United States District Court,
N.D. California.

TV INTERACTIVE DATA CORPORATION,
Plaintiff.
v.
MICROSOFT CORPORATION,
Defendant.

No. C 02-2385 JSW

Dec. 9, 2003.

Michael David Okerlund, Becky R. Thorson, Ken R. Hall, Misti N. Okerlund, Niall Andrew MacLeod, Richard M. Martinez, Ronald J. Schutz, Sang Young Adam Brodie, Robins, Kaplan, Miller & Ciresi L.L.P., Nicholas Stephan Boebel, Minneapolis, MN, Samuel Ray Miller, Sidley Austin LLP, Tracy Eila Reichmuth, Folger Levin & Kahn, San Francisco, CA, Scott Ray Strand, St. Paul, MN, for Plaintiff.

B. Jefferson Boggs, Jr., R. Danny Huntington, Timothy A. Molino, Bingham McCutchen, Washington, DC, David Stewart Shuman, John E. Gartman, Juanita R. Brooks, Peter Douglas Weinstein, Todd Glen Miller, Fish & Richardson PC, San Diego, CA, Kelly C. Hunsaker, Michael Curley, Fish & Richardson PC, Redwood City, CA, T. Andrew Culbert, Microsoft Corporation, Redmond, WA, Thomas M. Melsheimer, Fish & Richardson P.C., Dallas, TX, for Defendants.

CLAIM CONSTRUCTION ORDER

JEFFREY S. WHITE**, District Judge.**

A claim construction hearing was held on August 5, 2003 pursuant to *Markman v. Westview Instruments, Inc.*, 517 U.S. 360 (1996), to construe ten disputed terms of four patents at issue in this case. Having carefully reviewed the parties' papers, heard their arguments and considered the relevant legal authority, and good cause appearing, the Court will now construe the disputed terms.

BACKGROUND

TV Interactive Data Corporation ("TVI") alleges that some of Microsoft Corporation's ("Microsoft") Windows(TM) operating system products infringe on four of TVI's U.S. Patents. Specifically, TVI alleges that the "AutoPlay" feature integrated into Microsoft's products infringes U.S. Patent Nos. 5,597,307 ("the '307 patent"), 5,795,156 ("the '156 patent"), 6,249,863 ("the '863 patent"), and 6,418,532 ("the '532 patent"). The AutoPlay feature allows a computer program or application to be run automatically after storage media is inserted into the computer.

The parties submitted a Joint Claim Construction statement setting forth the parties' proposed construction

of the ten disputed terms. On July 29, 2003 the parties and the Court participated in a technology tutorial to aid the Court in understanding the underlying technology related to the patents and the accused infringing products. Subsequently, a *Markman* hearing was held on August 5, 2003. The Court now construes the terms disputed by the parties.

ANALYSIS

A. Legal Standard.

Determination of the scope and meaning of disputed terms in patent claims is a matter of law for the court to decide. *Markman,* 517 U.S. at 372. In addition, the determination of whether a claim is written in means-plus-function format and the determination of the claimed function and corresponding structure in the specification of a means-plus-function claim limitation is a question of law. Kemco Sales, Inc. v. Control Papers Co., 208 F .3d 1352, 1360 (Fed.Cir.2000). In order to determine the meaning of a patent claim, the court considers three sources: the claims, the specification, and the prosecution history. Markman v. Westview Instruments, Inc., 52 F.3d 967, 979 (Fed.Cir.1995) (*en banc*), *aff'd,* Markman, 517 U.S. 370.

In construing the claims, the court must begin with an examination of the claim language itself. "[T]he analytical focus must begin and remain centered on the language of the claims themselves, for it is that language that the patentee chose to use to particularly point out and distinctly claim the subject matter which the patentee regards as his invention." Texas Digital Sys., Inc. v. Telegenix, Inc., 308 F.3d 1193, 1201-02 (Fed.Cir.2002) (internal quotations and citations omitted). "The terms used in the claims bear a 'heavy presumption' that they mean what they say and have the ordinary meaning that would be attributed to those words by persons skilled in the relevant art." Id. at 1202; *see also* Teleflex, Inc. v. Ficosa North American Corp., 299 F.3d 1313, 1325 (Fed.Cir.2002). "The claims define the scope of the right to exclude; the claim construction inquiry, therefore, begins and ends in all cases with the actual words of the claim." Renshaw PLC v. Marposs Societa' per Aziono, 158 F.3d 1243, 1248 (Fed.Cir.1998).

The words in the claim must then be interpreted "in light of the intrinsic evidence of record, including the written description, the drawings, and the prosecution history, if in evidence." Teleflex, 299 F.3d at 1324-25. "Such intrinsic evidence is the most significant source of the legally operative meaning of disputed claim language." Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed.Cir.1996).

A patentee is presumed to have intended the ordinary meaning of a claim term in the absence of an express intent to the contrary. York Prods., Inc. v. Central Tractor Farm & Family Ctr., 99 F.3d 1568, 1572 (Fed.Cir.1996). "The subjective intent of the inventor when he used a particular term is of little or no probative weight in determining the scope of a claim (except as documented in the prosecution history)." Markman, 52 F.3d at 985. "Rather the focus is on the objective test of what one of ordinary skill in the art at the time of the invention would have understood the term to mean." Id. at 986. Indeed, "unless compelled otherwise, a court will give a claim term the full range of its ordinary meaning as understood by persons skilled in the relevant art." Texas Digital, 308 F.3d at 1202 (citations omitted).

Intent to limit the scope of a claim despite apparently broad language can be demonstrated in four ways. First, if the patentee "acted as his own lexicographer," and clearly set forth a definition of the disputed term in either the specification or the prosecution history, the court will defer to that definition. CCS Fitness, Inc. v. Brunswick Corp., 288 F.3d 1359, 1366 (Fed.Cir.2002) (citations omitted). Second, the court will adopt an altered meaning of a term "if the intrinsic evidence shows that the patentee distinguished that term from prior art on the basis of a particular embodiment, expressly disclaimed subject matter, or described a

particular embodiment as important to the invention." Id. at 1367. Third, a claim term will not take on its ordinary meaning "if the term chosen by the patentee so deprives the claim of clarity as to require resort to the other intrinsic evidence for a definite meaning." *Id*. Finally, a term in a means-plus-function claim is limited by statute to the structure or step described in the patent. 35 U.S.C. s. 112 para. 6.

Limitations from the specification (such as the preferred embodiment) cannot be read into the claims, absent and express intention to do so. *See, e.g*., Teleflex, 299 F.3d at 1326 ("The claims must be read in view of the specification, but limitations from the specification are not to be read into the claims.") (citations omitted); CCS Fitness, 288 F.3d at 1366 ("a patentee need not describe in the specification every conceivable and possible future embodiment of his invention"); Altris v. Symantec Corp., 318 F.3d 1363, 1372 (Fed.Cir.2003) ("resort to the rest of the specification to define a claim term is only appropriate in limited circumstances"). To protect against reading limitations from the specifications into the claims, the court should not consult the specification until after reviewing the claims in the light of the ordinary meaning of the words themselves. Texas Digital, 308 F.3d at 1204-05 (holding that to act otherwise "invites violation of our precedent counseling against importing limitations in to the claims") (citations omitted).

Only if the analysis of the intrinsic evidence fails to resolve any ambiguity in the claim language may the court rely on extrinsic evidence, such as expert declarations. Vitronics, 90 F.3d at 1583 ("[i]n those cases where the public record unambiguously describes the scope of the patented invention, reliance on any extrinsic evidence is improper.") Extrinsic evidence should be used only if needed to assist in determining the meaning or scope of technical terms in the claims, and may not be used to vary or contradict the terms of the claims. *Id*. (citing Pall Corp. v. Micron Separations, Inc., 66 F.3d 1211, 1216 (Fed.Cir.1995)).

In addition, the court has discretion to rely upon prior art, whether or not cited in the specification or file history, but only when the meaning of the disputed terms cannot be ascertained from a careful reading of the public record. Vitronics, 90 F.3d at 1584. Referring to prior art may make it unnecessary to rely upon expert testimony, because prior art may be indicative of what those skilled in the art generally understood certain terms to mean. *Id*. Unlike expert testimony, these sources are accessible to the public prior to litigation to aid in the determination of the scope of an invention. *Id*.

A patentee may write a claim in a means-plus-function format pursuant to 35 U.S.C. section 112, paragraph 6, by defining a particular function in the claims and a corresponding structure in the specification. Kemco Sales, 208 F.3d at 1360. Before the court can construe a claim that appears to be written in means-plus-function format, the court must determine whether the claim is subject to section 112, paragraph 6. Id. at 1361. Using the language "means" in the claimed language creates a presumption that the claim has invoked section 112, paragraph 6. *Id*. However, this presumption can be rebutted if the language of the claim itself recites a structure sufficient to support the claimed function. *Id*. Likewise, a claim not containing the words "means for" is presumed not to be subject to section 112, paragraph 6. However, this presumption is rebutted if the claim language does not recite a sufficiently definite structure to perform the claimed function. *Id*.

Once the court determines that a claim is subject to section 112, paragraph 6, the court must then construe the function recited in the claim and find the corresponding structure in the specification that performs that function. Chiuminatta Concrete Concepts, Inc. v. Cardinal Indus., 145 F.3d 1303, 1308 (Fed.Cir.1998). The patentee must describe in the specification the structure that performs the function described in the claims. Valmont Indus., Inc. v. Reinke Mfg. Co., 983 F.2d 1039, 1042 (Fed.Cir.1993). If the patentee fails to describe an adequate structure in the specification, the claim will be rendered invalid as indefinite under

section 112, paragraph 2. In re Donaldson Co., 16 F.3d 1189, 1195 (Fed.Cir.1994) (*en banc*).

B. Claim Construction.

1. Initialization file.

The term "initialization file" is found in claim 1 of each of the four patents at issue in this case. TVI proposes that the term "initialization file" should be construed as: "a file which, alone or in combination with other file(s), contains information or data used or referenced to start up or configure computer software and/or hardware." Microsoft's proposed construction of the term is: "an autostart driver, i.e. a device driver capable of automatically starting a process."

The parties agree that the term "initialization file" on its own has an ordinary meaning known by one who is skilled in the art of computer science. (Microsoft's Claim Construction Brief ("Microsoft's Br.") at 8; TVI's Opening Claim Construction Brief ("TVI's Opening Br.") at 8.) Typically, an initialization file is "a file that contains startup information required to launch a program or operating system." (Claim Const. Chart at 2-3.) The patent makes a reference to this ordinary meaning of initialization file: "[D]uring booting, host device 120 checks for initialization files such as startup files and configuration files. On finding a valid initialization file, host device 120 uses the initialization file during booting." (Declaration of Niall A. MacLeod ("MacLeod Decl.") Ex. 3 ("Specification") FN1 at Col. 22:19-23.) It is clear from both the evidence in the specification of the patent itself and the dictionary definitions that the term "initialization file" is well understood by those skilled in the art of computer science.

FN1. All of the patents at issue in this case are related, claim priority from the same patent application and have identical specifications. Therefore, the Court will follow the parties convention and cite to the specification of the '532 patent (attached as Exh. 3 to the Declaration of Niall A. MacLeod), when citing to the specification of the patents in suit.

There is a heavy presumption that claim terms carry their ordinary meaning as viewed by one skilled in the art. Texas Digital, 308 F.3d at 1202. However, this presumption can be overcome if the use of the ordinary meaning of a term in a claim makes the claims incomprehensible and their scope is entirely unclear. Bell Atl. Network Servs., Inc. v. Covad Communications Group, Inc., 262 F.3d 1258, 1268 (Fed.Cir.2001). Therefore, to determine the proper construction of a term that has a well established ordinary meaning, the claims containing this term must be read in light of the well established ordinary meaning to determine if they are comprehensible and their scope is clear. *Id* . If the claims are comprehensible and their scope is clear using the ordinary meaning of the term, it is that meaning that will govern construction of the claims. *Id*. However, if upon construing the claims using the ordinary meaning of the term the claims become uncomprehensible and their scope unclear, the claim term must be given some other meaning that is consistent with the claims. *Id*. When deciding the other meaning to be given to the term, the court is not at liberty to give a meaning that rewrites the claims just to save them from being invalid. Allen Eng'g Corp. v. Bartell Indus., Inc., 299 F.3d 1336, 1349 (Fed.Cir.2002).

The agreed upon ordinary meaning of the term "initialization file" is a file that contains configuration information used in the initialization (start-up or booting) of a computer system, and may also be used to determine the particular configuration of software when it is started. Therefore, it is clear from the ordinary meaning of the term that an initialization file is something that is only used to configure a program or system when it is started. An initialization file's role is complete once these instructions are read and used.

The result of these instructions can be the loading of a particular driver. (Declaration of Kelly C. Hunsaker ("Hunsaker Decl.") Exh. J at 98:20-99:12.) An initialization file is distinct from a driver because it is not "loaded" into memory and used to perform a further process. This is confirmed by the patent specification which points out the difference between an initialization file and a driver:

During booting, host device 120 checks for initialization files such as startup files and configuration files. On finding a valid initialization file, host device 120 uses the initialization file during booting. Then host device 120 ... installs an autostart driver (such as driver 436) in main memory [of the host device].

(Specification Col. 22:19-25.)

Applying this ordinary meaning of initialization file to the claim language makes some of the claims incomprehensible because some of the claims describe the initialization file as an active ongoing process that facilitates media detection, file checking and starting further processes each time a compatible file is found. For example, claims 6 and 14 of the '532 patent state that the initialization file may start up or spawn a new process to execute instructions each time the predetermined file is found. (Id. at Col. 52:8-13, 43-47). This function seems to be inconsistent with the initialization file being a startup or configuration file. On the other hand, some of the claims only make sense when the ordinary meaning of the term initialization files is used. For example, claims 1 and 20 of the '307 patent use the term initialization file consistently with its ordinary meaning. The claims recite the step of checking for and/or using an initialization file during the booting of an operating system on the host device. ('307 patent Col. 51:42-60; 52:48-54.) Therefore, there appears to be a conflict in the meaning of the term within the claim language of the patents.

Microsoft urges the Court to adopt a definition of initialization file as the "autostart driver" found in the specification. It is improper to import the claim limitation of "autostart driver" from the specification into the claims absent an intent by the patentee to do so. *See* Teleflex, 299 F.3d at 1326. In this case, the patentee did not act as his own lexicographer by defining the term initialization file in the specification as an autostart driver. In addition, as shown by the conflicting examples above, the patentee did not clearly redefine the term in the claim language, and the specification uses the term consistently with its ordinary meaning as agreed upon by both parties. When the patentee has not clearly redefined a term and where there is only one reasonable interpretation, even if the interpretation results in nonsensical construction, the court must construe the term according to its ordinary meaning. Process Control Corp. v. HydReclaim Corp., 190 F.3d 1350, 1357 (Fed.Cir.1999).

Accordingly, the Court construes the term "initialization file" according to its ordinary meaning in the art as: **a file which, alone or in combination with other file(s), contains information or data used or referenced to start up or configure software and/or hardware.**

2. Means for detecting insertion of a storage media into a peripheral.

The term "means for detecting insertion of a storage media into a peripheral" is found in claim 1 of both the '863 patent and the '156 patent. Both parties agree that the term is written in means-plus-function language and is therefore subject to 35 U.S.C. section 112, paragraph 6. (Microsoft's Br. at 11; TVI's Opening Br. at 9). The parties also agree that the function recited in the claim is "detecting insertion of a storage media into said peripheral." (Id.) The parties disagree as to what is the corresponding structure in the specification that performs the function. TVI proposes that the corresponding structure is "a host device (e.g.computer) programmed to perform the disclosed algorithm of detecting insertion of a storage media through polling or

use of an interrupt from a storage peripheral, and equivalents thereof." (TVI's Opening Brief at 12.) Microsoft proposes that the corresponding structure located in the specification is "an autostart driver that (a) enables an interrupt from a removable storage media peripheral, (b) waits for an interrupt from the removable storage media peripheral, and (c) receives the interrupt, for detecting insertion of a storage media into the peripheral." (Microsoft's Brief at 11.)

The specification describes the disclosed algorithms as being processed or "run" on a microcomputer or central processing unit. (Specification Col. 6:57-59.) TVI's contention is that the corresponding structure in the specification is a computer programmed with the disclosed algorithms. This position is entirely consistent with the Federal Circuit's reasoning in WMS Gaming, Inc. v. Int'l Game Tech., 184 F.3d 1339, 1349 (Fed.Cir.1999). "In a means-plus-function claim where the disclosed structure is a computer, or microprocessor, programmed to carry out an algorithm, the disclosed structure is not the general purpose computer, but rather a special purpose computer programmed to perform the disclosed algorithm." *Id*. In this case, the computer has been programmed with the software of the autostart driver. It is the autostart driver that instructs the computer how to detect the insertion of the storage media. Thus, a computer with the autostart driver loaded, is the structure disclosed that performs the claimed function. Therefore, the corresponding structure to "means for detecting insertion of a storage media into a peripheral" is a "host device (e.g.computer) programmed to perform the disclosed algorithm." However, "the structure of a microprocessor programmed to carry out an algorithm is limited by the disclosed algorithm." *Id*. The parties disagree as to what is the disclosed algorithm in the specification.

There are two possible approaches to design an algorithm to detect insertion of a storage media into a peripheral. These are termed the "device interrupt" and the "polling" or "timer interrupt." The "device interrupt" approach utilizes an interrupt to indicate that storage media has been inserted into a peripheral. In this approach it is the peripheral that sends a signal to indicate that a storage media has been inserted into the peripheral. The algorithm of Figure 5B contains the details of the "device interrupt" approach: the autostart driver (1) enables interrupts from removable storage media peripherals; (2) waits for an interrupt from any removable storage media peripheral; and (3) receives the interrupt, which indicates a removable storage media has been inserted into a peripheral. (Specification Col. 23:9-21.) The parties agree that this algorithm is a means for detecting insertion of a storage media into a peripheral.

The second approach that can be used to detect the insertion of a storage media into a peripheral is the "polling" or "timer interrupt" method. In this method the peripheral is queried at a point in time to determine if storage media has been inserted. The distinction between the "device interrupt" and "polling" or "timer interrupt" methods is set forth in the specification:

To facilitate automatic detection of insertion of a storage media, a host device is installed during booting with an autostart driver which periodically polls one or more peripherals of the host device. In an alternative embodiment, the autostart driver is invoked by an interrupt from a removable storage media peripheral caused by insertion of a storage media into the peripheral.

(Specification Col 21:59-65.)

Figure 5C discloses the "polling" or "timer interrupt" method to detect insertion of a storage media into a peripheral. A summary of the description of the function of the method disclosed in Figure 5C is as follows: the autostart driver (a) sets a timer period to one second, (b) enables a timer interrupt, (c) waits for the timer interrupt, (d) on receipt of the timer interrupt, sets the current peripheral to first removable storage media,

and then (e) checks to see if a predetermined file is present in the current peripheral. (*Id.* Fig. 5C). The algorithm disclosed can detect insertion of storage media in a peripheral by checking for a particular file on the storage media. However, because the algorithm only checks for a specified file, it is unable to detect insertion of storage media that does not contain the specified file. Therefore, the algorithm disclosed in Figure 5C only detects insertion some of the time.

The specification also contains the disclosure quoted above that "to facilitate automatic detection of a storage media ... an autostart driver periodically polls one or more peripherals of the host device." (*Id.* at Col 21:51-54.) TVI contends that this disclosure is sufficient to perform the claimed function because it contains the structure necessary to perform the claimed function. *See* Northrop Grumman Corp. v. Intel Corp., 325 F.3d 1346, 1352-53 (Fed.Cir.2003). Microsoft argues that the algorithm corresponding to the function in the claim must be described in the specification in sufficient detail to determine if the claim limitation is met by Microsoft's accused products. To determine whether sufficient structure has been disclosed to support a means-plus-function limitation, the Court must look at the disclosed structure and determine if "one skilled in the art" would believe that the structure disclosed was sufficient. Amtel Corp. v. Info. Storage Devices, Inc., 198 F.3d 1374, 1379 (Fed.Cir.1999). According to the deposition of Christian Hicks, one of ordinary skill in the art would know at least three ways to detect media insertion using a "polling" or "timer interrupt" method described in the specification. (MacLeod Deck, Exh. 9 at 162:20-163:17; 164:4-165:23.) These include: (1) checking a register; (2) checking shared memory; and (3) trying to access the media, including attempts to access a particular file, and by examining the different kinds of error codes being returned. (*Id.*) Therefore, one of ordinary skill in the art at the time of the invention would believe that the structure disclosed was sufficient to perform the claimed function.

Accordingly, the Court finds that the structure clearly linked to the function recited in the claim of detecting insertion of a storage media into said peripheral is: **a host device (e.g.computer) programmed to perform the disclosed algorithm of detecting insertion of a storage media through polling or use of an interrupt from a storage peripheral, and equivalents thereof.**FN2

FN2. 35 U.S.C. Section 112, paragraph 6, provides that the corresponding structure includes the structure disclosed in the specification and equivalents thereof.

3. Detecting insertion of a storage media into a peripheral.

The term "detecting insertion of a storage media into a peripheral" is found in claims 1 and 9 of the '532 patent. The parties dispute whether this term is written in step-plus-function format and thus governed by 35 U.S.C. section 112, paragraph 6. TVI asserts that the term is not subject to section 112, paragraph 6 and should be construed by the Court as "detecting insertion of a storage media into a peripheral." Conversely, Microsoft argues that the claim is written in step-plus-function format and is subject to 35 U.S.C. section 112, paragraph 6. Microsoft asserts the function of the term is "detecting insertion of a storage media into a peripheral." Microsoft sets out the corresponding structure in the specification that performs this function as "the acts of (a) enabling an interrupt from a removable storage media peripheral, (b) waiting for an interrupt from the removable storage media peripheral, and (c) receiving the interrupt, for detecting insertion of a storage media into the peripheral."

A patentee may write a claim in a step-plus-function format by describing a step in a method and not writing into the claims the acts necessary to carry out that step. O.I. Corp. v. Tekmar Co., Inc., 115 F.3d

1576, 1583 (Fed.Cir.1997). As is the case with to means-plus-function claims, in a step-plus-function claim the court will look to the specification to determine the corresponding acts necessary to accomplish the step disclosed. *Id*. Analysis of a possible step-plus-function claim involves two inquiries. First, it must be determined if the claim is written in step-plus-function format and therefore subject to section 112, paragraph 6. Second, if the claim is subject to section 112, paragraph 6, the corresponding acts in the specification must be identified.

Section 112, paragraph 6 is invoked when steps-plus-function language is used in a method claim without describing the corresponding acts in the claim. *Id*. When the patentee uses the words "step for" in a method claim there is a presumption that the claim is subject to section 112, paragraph 6. Masco Corp. v. United States, 303 F.3d 1316, 1327 (Fed.Cir.2002). However, the use of the words "step of" does not give rise to a presumption. *Id*. In addition, "where a method claim does not contain the term 'step for,' a limitation of that claim cannot be construed as a step-plus-function limitation without a showing that the limitation contains no act." *Id*. Functional steps describe what the claim element accomplishes, whereas acts describe how the function is accomplished. Seal-Flex, Inc. v. Athletic Track and Court Const., 172 F.3d 836, 849-50 (Fed.Cir.1999) (Raider J., concurring).

The claim term of "detecting insertion of a storage media into a peripheral" is not entitled to the presumption that it is subject to section 112, paragraph 6 because it is not written in step-plus-function format. The preamble of the claim contains the words "steps of" and not "steps for," and therefore, does give rise to a presumption that section 112, paragraph 6 is invoked. *See* Masco Corp., 303 F.3d at 1327. In addition, the claim element begins with a gerund ending in "ing," that indicates that the step is performing an act. In construing claim terms beginning with words that end with the active "ing," the Federal Circuit has stated:

If we were to construe every process claim containing steps described by a "ing" verb, such as passing, heating, reacting, transferring, etc. into a step-plus-function limitation, we would be limiting process claims in a manner never intended by Congress.

O.I. Corp., 115 F.3d at 1583.

In addition, Microsoft asserts that the term is written in step-plus-function format because the term "detecting insertion of a storage media into a peripheral" is similar to the term used in other related patents of "means for detecting insertion of a storage media into said peripheral." However, "interpretation of claims would be confusing indeed if claims that are not means- or step-plus-function claims were to be interpreted as if they were, only because they use language similar to that used in other claims subject to this provision." Id. at 1583-84.

Accordingly, the Court holds the term "detecting insertion of a storage media into a peripheral" is not written in step-plus-function format and is not subject to section 112, paragraph 6. Therefore, the term is construed as written as: **detecting insertion of a storage media into a peripheral.**

4. Means for checking ... for a file other than said initialization file.

The term "means for checking ... for a file other than said initialization file" is found in claim 1 of both the '156 patent and the '863 patent. Both parties agree that this term is subject to 35 U.S.C. section 112, paragraph 6 and that the function recited in the term is "checking for a file other than said initialization file."

TVI argues that the corresponding structure in the specification is "a host device (e.g.computer) programmed to perform the disclosed algorithm of determining if a file is present in or accessible from the storage media, and equivalents thereof." Microsoft believes that the corresponding structure is "an autostart driver, that on receipt of an interrupt, checks the storage media inserted in the peripheral for a file other than the autostart driver."

As stated above in the discussion of the term "means for detecting insertion of a storage media into said peripheral," the structure in the specification that performs the claimed function is a computer programmed to perform the function of determining if a file is present in or accessible from the storage media. The Court again follows the reasoning of the Federal Circuit in WMS Gaming, 184 F.3d at 1349, and construes the disputed term to mean: **a host device (e.g.computer) programmed to perform the disclosed algorithm of determining if a file is present in or accessible from the storage media, and equivalents thereof.**

5. Predetermined name.

The term "predetermined name" is found in claim 1 of the '863 and '532 patents and claim 9 of the '532 patent. TVI asserts that the term should be construed as "any name which is determined ahead of time." Microsoft believes the term should be construed as "any name which is determined ahead of time and which is consistently used in the autostart driver and in compatible storage media."

Both parties agree that "predetermined name" is not a term of art in computer science. In addition, "predetermined name" has an ordinary definition of "a name determined ahead of time." The parties disagree as to whether TVI modified the ordinary definition of the term by acting as their own lexicographer. Microsoft contends that TVI changed the ordinary definition by acting as their own lexicographer when describing an embodiment in the detailed description section of the specification. In describing Figure 5A, the patent defines predetermined name as follows: "A predetermined name is any name which is determined ahead of time and which is consistently used in the autostart driver and also in compatible storage media." (Specification Col. 22:39-42.)

It would be improper for the Court to read into the claims the additional limitations that the name is used consistently between the autostart driver and the storage media from a preferred embodiment in the specification. It is a basic principle of claim construction that while claim terms are read in light of the specification, limitations from the specification should not be read into the claims. *See* Comark Communications, Inc. v. Harris Corp., 156 F.3d 1182, 1186 (Fed.Cir.1998). Limitations from the specification (such as the preferred embodiment) cannot be read into the claims, absent an express intention to do so. *See, e.g.* Teleflex, 299 F.3d at 1326. TVI's definition of "predetermined name" was used in describing one embodiment of the invention. However, "a patentee need not describe in the specification every conceivable and possible future embodiment of his invention." CCS Fitness, 288 F.3d at 1366. The "definition" of the term "predetermined name" appears as a part of the description of a particular embodiment, and as such only served to explain that embodiment. In addition, the term was not defined along with other general definitions at the beginning of the detailed description.

The term "predetermined name" is a general term and has no special meaning in the art. TVI could have easily added additional limitations in the claims if they intended a more specific definition of the term. In construing claims, the Court follows the well settled rule that "the analytical focus must begin and remain centered on the language of the claims themselves, for it is that language that the patentee chose to use to particularly point out and distinctly claim the subject matter which the patentee regards as his invention."

Texas Digital Sys., 308 F.3d at 1201-02 (internal quotations and citations omitted). For the foregoing reasons, the Court construes the term "predetermined name" consistently with its plain meaning as: **any name which is determined ahead of time.**

6. Sequence of instructions to be executed.

The term "sequence of instructions to be executed" is found in claim 1 of the '532 and '863 patent and in claim 9 of the '532 patent. TVI's proposed construction of the term is "a discrete set of related directions to be executed." Microsoft construes the term as "two or more executable commands that must be executed in a particular order. The term contains three parts that the parties dispute as to their meaning. First, the parties dispute whether the term "instructions" includes both executable commands and declarative statements. Second, the parties disagree if the term requires more than one instruction to be present. Third, the parties dispute whether the "instructions" must be executed in a particular order." The Court will address each of these disputed issues in turn.

a. Instructions include both declarations and executable commands.

TVI contents that the term "instructions" should be construed as "directions" that include both executable commands and declarative statements. Microsoft argues that the term should be construed as "executable commands," therefore excluding any declarative statements. Microsoft attempts to establish the meaning of an instruction by referring to dictionary definitions, the language of the claims themselves, and the language of the specification. Microsoft cites two computer dictionary definitions in order to establish the ordinary meaning of the term instructions to one skilled in the art of computer science. These definitions both state that an instruction is a "command to the processor of a computer system" and take the "form of statements in a programming language that are subsequently translated by a compiler or interpreter into the instructions executed by the processor." (Hunsaker Deck, Exh. O at MST018749 and MST018739.) Both of the definitions cited by Microsoft support their construction that an instruction is a "command," however, this construction is inconsistent with the use of the term "instructions" in both the language of the claims and the language of the specification. This inconsistency is important because while there is a heavy presumption that claim terms carry their ordinary meaning as viewed by one skilled in the art, this presumption can be overcome if the use of the ordinary meaning of a term in a claim makes the claims incomprehensible and their scope is entirely unclear. Texas Digital Sys., 308 F.3d at 1202; Bell Atl. Network Servs., 262 F.3d at 1268.

The language of independent claim 1 of the '863 patent describes the "sequence of instructions" used to start a process. In addition to being included in the independent claim, the term "sequence of instructions" is also included in several subsequent dependent claims in the patent. Under the doctrine of claim differentiation, dependent claims are presumed to be narrower in scope than independent claims on which they depend. Ak Steel Corp. v. Sollac, 344 F.3d 1234, 1242 (Fed.Cir.2003). By this reasoning, an independent claim is necessarily broader than its dependent claims and is able to encompass all of the limitations in the dependent claims. *Id*. The dependent claims in the '863 patent define different possibilities of the content of the "sequence of instructions" described in the independent claim. For example, the various dependent claims describe:

said sequence of instructions comprises:

(claim 2) instructions for starting said process either from said inserted storage media or from a remote server's storage media;

(claim 3) instructions for accessing a specified location on said inserted storage media;

(claim 4) instructions for retrieving and displaying certain selections contained at a location on said inserted storage media;

(claim 5) instructions for executing certain instructions located in either of a host device memory or on said inserted storage media; and

(claim 7) instructions for executing an application for displaying one or more selections encoded on said inserted storage media.

('863 Patent Col. 52:9-33.)

The language of the dependent claims demonstrates that the "sequence of instructions" encompasses two distinct types of instructions. Dependant claims 2, 3, and 4 show that "sequence of instructions" includes instructions that command the computer to perform certain processes, such as starting up a process on a storage media or on a remote server, accessing a specified location on a storage media, or retrieving and displaying certain selections on a storage media. However, dependent claim 5 demonstrates the other type of instructions included in the "sequence of instructions." The claims include in the term "sequence of instructions" instructions to explain how to execute instructions to start up a process or, in other words, directions on how to execute instructions which start up a process. This use of instructions is different from instructions that actually start up the process; they instruct or direct how to execute the other instructions or commands to start up the process. Because these dependent claims define possibilities of what is contained in the "sequence of instructions" of the independent claim, the independent claim must be read to encompass all of these possibilities described in the dependent claims. Therefore, "sequence of instructions" includes both directions and commands.

The use of "directions" in this instance is akin to a declarative statement in a computer program. Microsoft states in their brief that declarative statements "set variables or provide information." (Microsoft's Br. at 18.) Since these "directions" are providing information about how to execute instructions or commands to start up a process they meet the definition of a declarative statements.

The language of the specification is not inconsistent with the term "instructions" including both commands and declarative statements. Parts of the specification describe the contents of the sequence of instructions (DISGO.BAT) containing commands for starting up an application. (Specification Col. 6:60-62.) However, there is nothing in the specification that is inconsistent with the sequence of instructions also containing declarative statements. Likewise, as shown above, the claims are not limited to commands and contemplate a sequence of instructions including declarative statements. Therefore, the specification is not inconsistent with the term "instructions" containing both commands and declarative statements. Accordingly, the Court construes the term "instructions" to include both commands and declarative statements.

b. A "sequence of instructions" must contain more than one instruction.

The parties also disagree whether a "sequence of instructions" must contain more than one instruction. When construing claim language of this type, the court must pay attention to the rules of grammar and syntax. In re Hyatt, 708 F.2d 712, 714 (Fed.Cir.1983). The term "instructions" is plural as used in the claim

language. In addition, the language "sequence" immediately proceeding "instructions" implies that there is more than one instruction. "Sequence of instructions" is also described in the dependent claims as "comprising instructions." Therefore more than one instruction is present in the sequence. This interpretation is also consistent with the specification which consistently speaks of "instructions" in the sequence of instructions. The only place that the singular term "instruction" is used is in the specification is describing what is included in the term "application startup instructions." (Specification Col. 22:53-59.) This description does not require that the "sequence of application start-up instructions" be singular, but rather only describes a list of several instructions that can be present in the sequence.

TVI asserts that "the number of instructions included in performing a function depends on the level of the programming language." (TVI's Opening Br. at 16.) This assertion does not aid the Court in resolving the issue of the meaning of the claim term, rather, the assertion is intended to show that it does not matter how many instructions are used to perform a function, the result is the same. The argument that the same function can be obtained by a substantially similar sequence of instructions (i.e., a single instruction or multiple instructions) is one that must be reserved for a determination of infringement of the accused device under the doctrine of equivalents. *See* Graver Tank & Mfg. Co., Inc. v. Linde Air Products Co., 339 U.S. 605 (1950). Therefore, based on the plain language of the claims, the Court holds that "sequence of instructions" contains more than one instruction.

c. The instructions need not be executed in a particular order .

The final disagreement between the parties on this term is whether a "sequence of instructions to be executed" must be executed in a particular order. Microsoft argues that the instructions "must be executed in a particular order." However, this construction is not helpful to the understanding of the claim language because the construction does not recite a particular order for the instructions to be executed, but merely states that they are executed in a particular order. All computer instructions are executed in a particular order and making a statement to that effect does not clarify the language of the claims. Microsoft next argues that TVI's expert admitted that the instructions contained in the file in TVI's preferred embodiment would not function properly if it was executed out of order. (Microsoft's Br. at 19; Hunsaker Decl., Exh. J at 200:8-203:16.) This argument simply underscores the fact that execution in some order is necessary for a computer program. However, the order of execution can be different under different conditions, including simultaneous execution. The plain language of the claim does not have a limitation that the instructions be executed in a particular order precisely because the exact sequence of instructions is not spelled out in the claims of the patent. It is not a limitation that was intended by the patentee and therefore it would be improper for the Court to read it into the claims. Therefore, Microsoft's proposed limitation of execution in "a particular order" is not adopted by the Court.

Accordingly, the Court construes the term "sequence of instructions to be executed" as: **two or more declarations or commands to be executed.**

7. Means for starting up said process.

The term "means for starting up said process" is found in claim 1 of the ' 863 patent. Both parties agree that this term is written in means-plus-function format and subject to 35 U.S.C. section 112, paragraph 6, however, they disagree on both what function the term embodies and the corresponding structure in the specification. TVI proposes that the function is "starting up a program or part of a program (e.g.code)," while Microsoft asserts the function is "starting up a process." TVI asserts that the corresponding structure in the specification is "a host device (e.g .computer) programmed to perform the disclosed algorithm of

performing a set of discrete, related directions to initiate the running of a program or part of a program (e.g.code), and equivalents thereof." Microsoft construes the corresponding structure as "an autostart driver that executes the file of a predetermined name which, in turn, executes the sequence of instructions contained within that file, for starting up said process."

When determining the function of a term written in means-plus-function format section 112, paragraph 6, does not permit limitation of a function by adopting a function different from that explicitly recited in the claim. Micro Chem., Inc. v. Great Plains Chem. Co., 194 F.3d 1250, 1258 (Fed.Cir.1999). The function explicitly recited in the claim is "starting up said process." A computer program is a type of process that may be started up. It would be improper for the Court to further limit the recited function to "starting up a program or part of a program." Therefore the function explicitly recited in the claim and the function performed by the disputed limitation is "starting up a process."

For the reasons stated above in the two previous disputed terms subject to section 112, paragraph 6, the corresponding structure in the specification is a computer programmed with the algorithm disclosed in the specification that starts up a process. The specification discloses two possible algorithms for starting up a process. One is a computer programmed to execute a file of predetermined name that contains instructions that are used to start up a process. (Specification Col. 23:55-68.) The other is the computer programmed to execute a new process which, in turn, executes a sequence of instructions within the file of predetermined name to start up a process. (*Id*. at Col 24: 7-11.) This distinction is made clear in the specification:

In another embodiment of this invention, in step 529, instead of the autostart driver executing the instructions in DISGO.BAT as shown in FIG. 5B, the autostart driver starts up or spawns a new process which executes the instructions in DISGO.BAT.

(*Id*.)

Therefore, the Court construes the corresponding structure in the specification as: **a host device (e.g.computer) programmed to perform the disclosed algorithms of starting up a process by either executing a file of predetermined name which contains directions to start up a process or executing a new process which, in turn, executes directions in a file of predetermined name to start up a process, and equivalents thereof.**

8. Automatically.

The term "automatically" is found in claim 1 of the '532, '863, '156, and ' 307 patents, in addition, it is found in claim 4 of the '156 patent, and claims 2 and 14 of the '307 patent. TVI's proposed construction of "automatically" is "wholly or largely controlled by a device (e.g.computer)." Microsoft's construction of the term is "without user input."

Both parties agree that the term "automatically" is a relative term and that the proper definition of the term depends on the context it is used in. Therefore, it is necessary to look at the use of the term in the claim language and the specification to determine the proper meaning of automatically. *See* Renishaw PLC, 158 F.3d at 1250 (where a term has several possible meanings "the patent disclosure serves to point away from the improper meanings and toward the proper meaning."). Automatically is used in the claims of the patents to characterize "checking for a file," "loading a portion of the electronic content," "executing an application," and "loading an initialization file." In addition, it is used in various other ways in the claim

language. "Automatically" is also used in the specification to describe a preferred embodiment:

Therefore when an autostart driver is installed in a host device, this invention allows applications encoded in compatible storage media (as described above) to start up automatically without any additional user input, soon after a storage media is inserted into a peripheral of a host device. Therefore, once a host device is powered up, booted and installed with an autostart driver, a user need not touch any keys switches of a host device, and can merely insert storage media to start an application in accordance with this invention.

(Specification Col. 26:11-20.)

It is apparent from the specification that the term "automatically" is used to start up an application "without any additional user input." *Id*.

During the prosecution of the patent TVI distinguished prior art based on the automated feature of the current invention. "It is well established that 'the prosecution history limits the interpretation of claim terms so as to exclude any interpretation that was disclaimed during prosecution.' " Springs Window Fashions L.P. v. Novo Indus. L.P., 323 F.3d 989, 994 (Fed.Cir.2003) (quoting *Pall Corp.,* 259 F.3d at 1392). The claim term can be limited by arguments in the prosecution history used to distinguish a claim over prior art. Hughes Aircraft Co. v. United States, 717 F.2d 1351, 1362 (Fed.Cir.1983). However, this disclaimer must be done with "reasonable clarity and deliberateness ." *N*. Telecom Ltd. v. Samsung Elecs. Co., 215 F.3d 1281, 1294 (Fed.Cir.2000). The patentee distinguished the Willman et al. reference during the prosecution of the patent as follows:

Willman et al. fail to disclose or suggest "automatically checking for a *file* in a removable storage media ..." and the step of "automatically loading at least a portion of the electronic content ... in response to finding said *file* " as recited by Claim 63. Applicants submit that modifying or adding "file systems" as disclosed by Willman et al. merely involves determining the format of files encoded on a storage media, and requires a user to actually input the name of a file to be loaded (as discussed in the previous paragraph). In contrast, a method according to Claim 63 performs the step of "checking" as well as the step of "loading" automatically, without user input of a file name.

(Hunsaker Deck, Exh. Q at 16-17.)

This argument made during the prosecution of the patent to distinguish the Willman reference because the steps of "checking" and "loading" are done without user input of a file name does not eliminate all possible user input, it only eliminates user input of a file name. Therefore, there is still a possibility of user input reserved by the patentee during prosecution. The prosecution history of the patent serves to clarify what the patentee meant by "automatically" and "without any additional user input." The Court adopts the definition of automatically used in the specification as clarified by the prosecution history as: **without user input of a file name.**

9. Application.

The term "application" is found in claim 7 of the '532 and '863 patents, claim 15 of the '532 patent and claim 1 of the '307 patent. TVI proposes the Court construe the term as "a program designed to assist in the performance of a specific task, such as word processing, accounting, or inventory management." Microsoft proposes the term be construed as "a program which interprets button codes from a remote control."

The term "application" has a well established meaning that one of ordinary skill in the art of computer science would know. An application is "a program designed to assist in the performance of a specific task, such as word processing, accounting, or inventory management." (Claim Const. Chart at 51.) This definition is confirmed by Microsoft's expert Dr. Long. (MacLeod Deck, Exh. 6 at 27:5-15). A term is presumed to have its ordinary meaning unless a patentee acts as his own lexicographer and clearly sets out an alternate definition in the patent specification. CCS Fitness, 288 F.3d at 1366. In one embodiment described in the specification the patents at issue in this case set forth an alternate definition of "application" that departs from the ordinary meaning. "As used herein, the term 'application' is intended to mean code and/or data which interprets button codes from a remote control." (Specification Col 9:18-21). This definition limits "application" to a program that interprets button codes from a remote control and is used in numerous places in the specification consistently with this modified definition. However, this definition is explicitly expanded later in the specification as follows:

Although an autostart driver is used to start an application for a remote control in one embodiment of this invention, an autostart driver can also automatically start other applications such as LOTUS 1-2-3 and Word Perfect if so programmed.

(Specification Col. 25:20-24.)

Thus, it is clear that the term application is not limited only to a program that interprets button codes from a remote control. The specification clearly states that a program that interprets button codes from a remote control is only "one embodiment" of an application and puts forth other types of applications that may be used. It would be improper for the Court to read into the claims the limitations contained in an embodiment into the claims. *See Teleflex,* 299 F.3d at 326. Therefore, the Court interprets the term "application" consistent with its ordinary meaning in the art and use in the specification as: **a program designed to assist in the performance of a specific task, such as word processing, accounting, or inventory management.**

10. Returning to said step of automatically enabling.

The term "returning to said step of automatically enabling" is found in claim 1 of the '307 patent. TVI proposes that this term be construed as "returning to a state in which the host device (e.g .computer) is prepared to process requests for attention to or from hardware and/or software." Microsoft construes the term as "after having previously disabled the interrupt, re-enabling the interrupt." Microsoft's construction adds two additional limitations into the claims from the specification: disabling the interrupt and re-enabling the interrupt. These limitations do not appear in the claims and it would be improper for the Court to read the limitations into the claims. *See* Altris, 318 F.3d at 1370.

Microsoft's proposed construction is also inconsistent with the doctrine of claim differentiation. Under this doctrine, "each claim in a patent is presumptively different in scope." Wenger Mfg., Inc. v. Coating Mach. Sys., Inc., 239 F.3d 1225, 1233 (Fed.Cir.2001). The doctrine is particularly useful where "there is a dispute over whether a limitation found in a dependent claim should be read into an independent claim, and that limitation is the only meaningful difference between the two claims." Intermatic, Inc. v.. Lamson Sessions Co., 273 F.3d 1355, 1364 (Fed.Cir.2001). This is precisely the case when dependent claim 1 is compared to independent claim 16 of the '307 patent. Dependent claim 16 reads:

"The method of claim 1 further comprising disabling said interrupt prior to said step of automatically

executing said file." (Specification Col. 52:38-40.) Dependent claim 16 adds the limitation of disabling the interrupt, a step not found in independent claim 1. Therefore, under the doctrine of claim differentiation, independent claim 1 does not include the limitation of disabling the interrupt. If the limitation of disabling the interrupt is not present in claim 1, then there is no need for the limitation of re-enabling the interrupt.

For the foregoing reasons, the Court construes the term "returning to said step of automatically enabling" as: **returning to a state in which the host device (e.g.computer) is prepared to process requests for attention to or from hardware and/or software.**

CONCLUSION

Based on the analysis set forth above, the Court adopts the foregoing constructions of the disputed terms. The parties are ordered to submit a further joint case management report pursuant to Patent Standing Order paragraph 13 within 21 days of the filing of this Order. The parties are specifically ordered to propose a stipulated date to schedule a settlement conference before Magistrate Judge Zimmerman.

IT IS SO ORDERED.

N.D.Cal.,2003.
TV Interactive Data Corp. v. Microsoft Corp.