

Filed on behalf of SurfCast, Inc.

By: Richard G. A. Bone ([RBone@VLPLawGroup.com](mailto:RBone@VLPLawGroup.com))

VLP Law Group LLP

555 Bryant Street, Suite 820

Palo Alto, CA 94301

Tel. (650) 419 2010

FAX (650) 353 5715

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

MICROSOFT CORPORATION

Petitioner

v.

SURFCAST, INC.

Patent Owner

---

Case IPR2013-00292<sup>1</sup>

Patent 6,724,403

---

DECLARATION OF GLENN E. WEADOCK  
IN SUPPORT OF PATENT OWNER'S REPLY TO OPPOSITION TO MOTION  
TO CONDITIONALLY AMEND  
UNDER 35 U.S.C. § 316(a)(9), AND 37 C.F.R. §§ 42.121 AND 42.23

---

<sup>1</sup> Cases IPR2013-00293, IPR2013-00294, and IPR2013-00295 have been consolidated with the instant proceeding.

## Table of Contents

1. Background and Qualifications .....	4
A. Scope of Work.....	4
B. Expertise.....	4
C. Patent Cases in Which I Have Offered Expert Testimony or Consulting .....	5
2. Patent Owner’s Substitute Claims recite terms that have been construed by Patent Owner, or that have plain and ordinary meanings .....	5
A. Grid of tiles Being Persistent .....	5
B. Session to Session .....	15
C. Call[ing] an application.....	16
D. Different .....	17
E. Selectable .....	19
F. Not Permitted to overlap .....	24
3. Patent Owner addressed the most relevant prior art .....	26

4. Patent Owner provided Written Description Support for Proposed Substitute Claims .....	28
5. Proposed Substitute Claims 53 – 59 Are Patentable Over the Prior Art .....	29
A. MSIE Kit .....	29
B. Kostas .....	41
C. Sapphire.....	47
6. Appendix (References and Information Considered).....	51

## **1. Background and Qualifications**

### **A. Scope of Work**

1. I have been asked by counsel for Patent Owner, SurfCast, Inc. (“SurfCast”), to support SurfCast’s “Motion For Conditional Amendment” of U.S. Patent no. 6,724,403 (“the ’403 Patent”).

2. The opinions provided are my own and are based on my analysis and work in this case and the education, experience, and skills I have acquired and developed throughout my career. Exhibit 2053 contains my CV.

3. In reaching my conclusions and opinions, I have relied upon my experience and training, my testing, and my review of the evidence produced in this case, and I have considered the documents and materials described in Appendix B in the process of forming my opinions.

4. My company is being paid \$300 per hour for my services, plus my direct expenses. I charge for my time and my compensation is not affected by the outcome in this matter.

### **B. Expertise**

5. Details of my professional qualifications and background are set out in my curriculum vitae, a copy of which is provided at Exhibit 2053, and in my previously submitted declarations, Exhibits 2004 and 2063.

**C. Patent Cases in Which I Have Offered Expert Testimony or Consulting**

6. A listing of patent cases in which I have offered testimony or consulted in is set forth in my previously submitted declarations, Exhibits 2004 and 2063.

**2. Patent Owner's Substitute Claims recite terms that have been construed by Patent Owner, or that have plain and ordinary meanings**

**A. Grid of Tiles Being Persistent**

7. Petitioner (but not Petitioner's expert, Dr. Karger) objects to Substitute Claim 53's "said grid of tiles being persistent from session to session" on the ground that "Patent Owner's construction ... points to attributes of individual **tiles**" while the Claim "requires the '**grid**' to be persistent, not the '**tiles.**'" (Paper No. 41, "Opposition", at 2-3.) Petitioner's allegation that the use of "tile" rather than "grid" in this construction creates ambiguity is without merit: if the tiles in the grid are persistent, then the grid of tiles is persistent, and vice-versa.

8. In my view, Patent Owner's proposed construction that "persistent," as used with respect to a tile, means that "at least the tile's position in a grid or array, the tile's refresh rate, and a source of information associated with the tile are maintained from session to session" (Paper No. 28, "Motion" at 5), is all that was required to understand the limitation "said grid of tiles being persistent from session to session".

9. A person of ordinary skill in the art at the time of the '403 Patent<sup>2</sup> would understand this language does not require that the grid be displayed on the screen between sessions, as Petitioner's expert implies in his "Reply Declaration of David R. Karger Regarding U.S. Patent No. 6,724,403." (Ex. 1110 at ¶¶ 168-169.) Every computer user understands that displays reset themselves when a user logs off, and go blank when a computer is turned off or placed into a sleep or hibernate mode.

10. Furthermore, the judge in the District Court proceeding (District of Maine, Docket No. 2:12-cv-00333-JAW) has ruled that "tile" includes "the graphical representation being persistent." His reasoning, which I believe is applicable under either the broadest reasonable interpretation in view of the specification standard or the standard applied by U.S. district courts, shows good common sense and mirrors arguments that Patent Owner and I have made in the past:

"This invention is, at its heart, a human interface concept that is meant to replace the icon-based paradigm with a tile-based paradigm. In order to accomplish this function, the invention must preserve critical

---

<sup>2</sup> Throughout this declaration, where I reference a person of ordinary skill in the art, or a person skilled (of skill) in the art, I am referring to such a person at the time that the '403 Patent was filed.

functionality of the old paradigm while also introducing something new and useful. Icons are indisputably “persistent” in the old paradigm, and this functionality is essential to human beings who must interface with computers. If icons were not persistent, a computer user would have to recreate his desktop—his “home base”—every time he started his computer. This would be an enormous aggregate waste of time, and would greatly impede the usefulness of the human interface. It follows that tiles could not replace icons without also being persistent, or providing some wholly new way of interfacing that did not make persistence a necessary feature. If Microsoft were correct in its objection, the ’403 Patent would purport to replace the icon paradigm without providing any alternative to persistence. This would be futile. One of skill in the art, after reading and digesting the patent as a whole, would understand that a “tile,” as a human interface concept, would be of little use were it not to persist between sessions.” Ex. 2082 at 22 – 23.

11. Tiles provide or enable “bookmarking” or “dynamic bookmarking.” (’403 Patent, Ex. 1001 at 8:30-34 and 11:50-52). These references indicate persistence because persistence is an essential characteristic of a bookmark<sup>3</sup>. In the

---

<sup>3</sup> In Dr. Karger’s deposition, the only example he provided of a bookmark not being persistent is when his laptop crashed. He then agreed that when the system worked correctly, “those bookmarks would be there later.” (Ex. 1113, 63:23-

physical world, a bookmark would be useless if it were gone (*i.e.*, if it were not persistent) the next time the user picked up the book. Similarly, bookmarks in the computer field (e.g., “favorites” in Microsoft’s Internet Explorer and bookmarks in Netscape Navigator; see Ex. 2083 (definitions of “bookmark” and “bookmark files” in the Microsoft Computer Dictionary, 5th ed.) are also necessarily persistent and would have been understood as such by those of skill in the art at the time of the ’403 Patent.

12. Moreover, because desktop icons are persistent, one of skill in the art would further understand the statement that the grid [of tiles] “...replaces the functionality of the computer ‘desktop’” (Ex. 1001 at 11:32-33) to indicate that tiles are persistent. This is because one of skill in the art would understand that replacement of the desktop includes replacement of the desktop icons’ function of providing a persistent mechanism to call an application program. Petitioner’s expert Dr. Karger understands this: “My understanding is that the grid of tiles is replacing the static icons and tool bars that are traditionally found on a desktop” (Ex. 1113, 71:4-6), and he replied “Yes, correct” to the question “And those icons that are being replaced by the grid of tiles were persistent, as we discussed earlier,

---

64:13). Dr. Karger also admitted that bookmarks “fulfill the sort of standard notion of persistence.” (Ex. 1113 at 64:23-24)



correct?” (Ex. 1113, 71: 15-20).

13. Furthermore, the '403 Patent states that “System 100 comprises a central processing unit 104, an input device 106 . . . and a main memory 112, all connected via bus 116. Residing in the memory is an operating system 120, a file system 122, a cache 124 for temporary storage of information, application programs 128-1 through 128-N and a graphical user interface (GUI) program 124 that is responsible for presenting information onto display 110.” (Ex. 1001 at 6:62 – 7:3.) The '403 Patent further states that “A representative tile data structure 500 is shown in Figure 4. It is important to understand that the tile itself is an image that at any given time is resident on the file system.” (Ex. 1001 at 9:57 – 59.) These passages indicate that tiles are persistent because they indicate that the file system is stored in a non-volatile memory, and that the tile image is stored on the file system in this non-volatile memory.

14. Two of Dr. Karger’s arguments against persistence in his reply declaration (Ex. 1110 at ¶¶168 – 172) appear to be new, as follows.

15. First, in ¶169 of Ex. 1110, Dr. Karger argues that “just because some or all of the information used to display a grid of tiles might be persistently stored in a database does not mean that the grid of tiles is persistently displayed.” While that is true when taken in isolation, it ignores the express disclosure in the '403 Patent that the grid of tiles “replaces the user’s desktop” (Ex. 1001 at 11:29-31)

SurfCast – Exhibit 2081, p. 9  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

and provides “bookmarking” (Ex. 1001 at 8:10 – 34 and 11:50-52). In light of this additional disclosure, a person of skill in the art would apply common sense and conclude that the information that is stored will be used for its obvious purpose, namely, to display the grid of tiles.

16. Second, in ¶170 of Ex. 1110, Dr. Karger points out that Patent Owner relies on a statement that information about the grid is stored in a “file system of the system’s main memory.” He then asserts that “main memory” means non-persistent memory, and points to two citations. Dr. Karger’s reasoning is flawed for several reasons.

17. First, for every file system I am aware of, none of them only resides in non-persistent memory (*e.g.* RAM). File systems, like operating systems and application programs, must be loaded into RAM from somewhere, and that “somewhere” is almost always a persistent storage device of some form (hard drive, flash drive, floppy disk, *etc.*). This is particularly true for personal computers. I note that when cross examined, Dr. Karger acknowledged that, in a typical system, most of the file system is likely to be on a device such as a hard disk, and only a portion of it may be in RAM when in use. (See Ex. 1113. at 125:22-126:7.) This testimony confirms my point that a file system is generally understood to reside in a non-volatile memory. Therefore, one of ordinary skill in the art would have understood that the term “main memory” used in the ’403

SurfCast – Exhibit 2081, p. 10  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

Patent refers to both non-volatile memory and volatile memory. I note that the example relied on in Dr. Karger's report states that the "main memory 22 is *typically* random access memory," (Ex. 1110, ¶ 170) which makes clear that the term "main memory" is sometimes used to refer to things other than RAM. The '403 Patent is one such instance.

18. Fig. 2 of the '403 Patent makes this clear. Box 112 is described as "a main memory" and it contains operating system, file system, cache, GUI, and multiple application programs. By Dr. Karger's logic, the operating system, GUI, and application programs also do not reside in any persistent storage — a scenario I have not run across.

19. It was (and is) common practice, when referring to something residing in a "file system", to presume non-volatile, persistent storage. The fact that the operating systems most commonly in use at the time of the '403 Patent (*e.g.*, Windows 98) would not entirely fit in the RAM of most computers further reinforces the understanding that "main memory" is being used in the '403 Patent and the Provisional Application to which it claims priority (Ex. 1005) to refer to non-volatile storage in addition to RAM. This understanding is further supported by the definition of "file system" in the Microsoft Press Computer Dictionary (2d. ed. 1994) (Ex. 2084): "In an operating system, the overall structure in which files are named, stored, and organized. A file system consists of files, directories, and

SurfCast – Exhibit 2081, p. 11  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

the information needed to locate and access these items. The term can also refer to the portion of the operating system that translates requests for file operations from an application program into low-level, sector oriented tasks that can be understood by the drivers that control the *disk drives*.” (emphasis added) This is because the reference to disk drives in connection with file system operations confirms the understanding of one of ordinary skill in the art that files systems are typically stored on non-volatile, persistent storage devices such as disk drives.

20. Dr. Karger has stated during deposition that if the operating system were only present in RAM, it would disappear when the computer got shut off. (Ex. 1113 at 121:20-23). He agreed that if the file system were only in RAM and were not stored on a hard disk, then the file system would be lost when the computer is turned off. (Ex. 1113 at 128:13-17.) These are both extremely impractical scenarios. Dr. Karger also stated that “in general, when I install things, I install – I would install them onto a disk ... for persistence, actually.” (Ex. 1113 at 121:6-11.)

21. In deposition testimony, in response to the question “Are you aware of any installation of Windows 95 in only RAM on any computer system?” Dr. Karger alluded to having seen “a variety of installations of various operating systems on virtual machines, and it’s quite plausible for some of these virtual machines to be operating entirely in RAM.” (Ex. 1113, 104:14-292:4) In my

SurfCast – Exhibit 2081, p. 12  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

experience with virtual machines (which is extensive), it is not at all plausible for them to be “operating entirely in RAM” with no associated non-volatile storage from which to load that RAM. Virtual machines may operate “entirely in RAM” once they are loaded, as Windows may, but, like Windows, virtual machines must start somehow, and they start from persistent, nonvolatile disk storage.

Furthermore, when a virtual machine is shut down, its contents are saved to that same non-volatile disk storage. The notion of re-creating a virtual machine “from scratch” every time a user boots it, is as far-fetched as the notion of re-installing the Windows operating system every time a user boots it. With respect to Dr. Karger’s assertion that “...certainly now you can boot off a network... you can fetch your operating system when you turn on the computer...” (Ex. 1113, 122:5-8), again, the operating system is loaded into RAM *from nonvolatile storage on the network, e.g., a network drive*. A disk drive is always involved.

22. The exhibits provided in Dr. Karger’s deposition (specifically, numbered as 2078 – 2080) make clear that the disk requirements for the Windows 95, Windows 98, and Windows XP operating systems are much greater than the RAM requirements, obviously indicating that it would have been infeasible to store the entire contents of these operating systems in RAM. I have written and taught seminars, and written a number of books, about all these operating systems, and I have never seen an instance in which the complete contents of the Windows CD-

SurfCast – Exhibit 2081, p. 13  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

ROM are loaded into RAM, nor would it have been technically possible at the time of the '403 Patent to do so.

23. With respect to the citations Dr. Karger makes to support his argument, the first one (a different patent from a different inventor) is a partial citation. When the full context is provided, Dr. Karger's analysis falls apart: "The main memory 22 is typically random access memory (RAM). *In operation, the platform 10 loads the operating system, for example Windows NT, into RAM from hard disk (not shown).*" Ex. 1107 at ¶ 22 (emphasis added).

24. The second citation (again, a different patent from a different inventor) is not even directed towards a computer, but rather towards a phone switch. Even so, the very first sentence of the summary of the invention refers to providing "a stored program controlled electronic switching system," the use of the word "stored" clearly indicating persistent storage, from which "main memory" may be populated. (Ex. 1108 at 1:59-64). The '364 patent discloses magnetic drum units/controllers in which copies of the contents of the main memory are stored. (Ex. 1108 at 6:34-38.)

25. Several references in the patent literature refer to "main memory" as including disk storage. International patent application WO 96/01032 (Ex. 2086; dated 1996) refers in claim 15 to a node "...wherein said main memory includes a hard disk or a read and/or write memory or a compact disc." U.S. Pat. No.

SurfCast – Exhibit 2081, p. 14  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

RE43,119 E1 (Ex. 2085; having a 1998 priority date) discusses a hard disk serving as the “main memory unit” and also refers to “main memory” as RAM, using the term “main memory” to apply to both disk and RAM. Even more recently, U.S. Pat. No. 8,601,198 B2 (Ex. 2087; filed June 30, 2011) states in its description of Figure 1 that “Main memory 120 may include any combination of volatile and non-volatile memory.”

26. Even if one were to assume that a “main memory” were limited to RAM, the ’403 Patent still would indicate the presence of a hard disk or other non-volatile mass storage device to one of ordinary skill in the art. This is because a person of ordinary skill in the art would know that a hard drive would be needed as a source device in order for all the software shown in Fig. 2 of the ’403 Patent (Ex. 1001) to be loaded into “a main memory” RAM when needed by the computer. The disclosure of a RAM main memory itself would be a sufficient implied disclosure of a non-volatile storage device to one of ordinary skill in the art.

### **B. Session to Session**

27. The term “session to session” has a well-understood meaning in the art which would be understood to require the tiles to be persistent across sessions whether or not the computer is turned on and off.

28. Petitioner asserted that there is no objective meaning of “session”.

Patent Owner disagrees: the '403 Patent uses the term in a manner consistent with its ordinary meaning, that it is a period of continuous usage of the computing device, interrupted by, for example, a power cycle, a period of hibernation or sleep, or a change of user via a log-in or log-out instruction.

29. This is confirmed by Petitioner's expert, Dr. Karger, who has a "natural understanding" of session, as he testifies in his deposition: "Well, one kind of session might be where I leave the computer to go get some coffee and then I come back to the computer and start a new session of work. Another one might be where I turn off the computer and I start it again later to do some additional work." (Ex. 1113 at 65:17-23)

### **C. Call[ing] an application**

30. Dr. Karger says that the '403 Patent does not support the notion of calling an application program by selecting a tile (see Ex. 1110 at ¶173-179). I disagree. For example, the '403 Patent (cited in the Motion at 6) states at 11:23-24: "Therefore the application resides over existing applications without replacing any of them; rather it enables them to be called from the grid itself." In my opinion, this clearly indicates that application programs are called by selecting a tile.

31. Nevertheless, further support appears in the '403 Patent at 9:25-27: "When a tile is selected, whether by mouse click or otherwise, the tile instantly



provides the user with access to the underlying information....” and at 9:52-54, “Double-clicking tile 408 causes the mailbox window of the email utility to be sufficiently enlarged to allow new messages to be selected and read.” “Providing the user with access to the underlying information” clearly implies the invocation of an application program, and the reference to the “email utility” in the second citation reinforces the implication, as an “email utility” is a kind of application.

#### **D. Different**

32. Petitioner argues that Patent Owner has not offered an interpretation for “what makes two computer programs ‘different’” and this thereby “prejudices ... the public, and prevents a fair evaluation of the merits of its claim amendment.” Opposition at p. 4. Petitioner’s expert has offered that, in the context of two computer programs: “Internet Explorer 4 is a program that is ‘different from’ the program that implements the grid functionality of Active Desktop, or indeed from Active Desktop generally, because they are not the same. While there may be some overlap and they may even share certain instructions, it is clear that each employs different functionality and therefore different instructions.” (Ex. 1110 at ¶ 236.) Dr. Karger applies the same analysis when comparing Netscape browsers to the Active Desktop. (Ex. 1110 at ¶ 238.)

33. In my view, the meaning of the word “different” is so basic to the

English language that no further elaboration or special construction is necessary. This is equally true in the context of two computer programs. One of skill in the art would refer to two computer programs as being different from one another if, for example, they performed different functions, or if they were marketed under different names. Thus, a word processing program such as Microsoft Word and an Internet web-browser such as FireFox are different from one another because they have different functions. Additionally, two programs that have similar functions but which are sold under different names and present different interfaces to the user would also be “different” from one another: an example of two such programs would be Microsoft Word and Corel’s WordPerfect, both of which are word processing programs, but which a user would classify as being different from one another. Furthermore, two computer programs may share some functionality yet still be different from one another: for example, both Microsoft Word and Microsoft PowerPoint contain flow-chart tools within their functionalities but are considered to be different programs from one another.

34. Furthermore, the fact that the meaning of the term “different” is not itself vulnerable to ambiguity is demonstrated throughout Dr. Karger’s declaration where the term is used liberally without qualification or embellishment. For example, at ¶ 198, Dr. Karger notes that “multiple Desktop Items can be associated with different information sources” when analyzing Patent Owner’s proposed

SurfCast – Exhibit 2081, p. 18  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

claim limitation that a first and second information source are associated with respectively a first and second tiles. Dr. Karger further quotes directly from MSIE Kit to illustrate that “RealPlayer [can] play different media types” as well as to show that Outlook Express was an “‘application program being different from’ the portion of Active Desktop that arranges the display into the grid of tiles, and also different from Active Desktop generally.” (Ex. 1110 at ¶¶ 220, 221.) Dr. Karger also states, as a matter of fact, that “The email programs and the word processor program of Myers are clearly different programs because they perform different functions.” (Ex. 1110 at ¶ 345. See also Ex. 1113 at 98:4-7.)

35. Dr. Karger is also able to construe the term “different” when purporting to read Patent Owner’s claims on the references cited by Petitioner. For example, “Kostes discloses a replacement graphical environment for organizing information from different applications”. (Ex. 1110 at ¶ 273. See also Ex. 1110 at ¶¶ 308, 310, 311, and 315.) That Dr. Karger is able to understand the claim terms and measure them up to various references without requiring a specific construction indicates that Patent Owner has not fallen short in not offering a construction of the word “different”: it rests on its plain and ordinary meaning.

#### **E. Selectable**

36. In substitute claim 53, tiles are selectable, “wherein selecting a tile

calls an assigned application program to provide access to information.” As shown in Ex. 1001 at 10:44 – 49, the ’403 Patent discloses “[t]he overall hierarchy of a graphical user interface embodied by the present invention is summarized in FIG. 7 . . . [t]he grid can replace the functionality of a user’s computer desktop and offers similar and additional features.” From this statement, taken together with other statements in the ’403 Patent, such as description of the selection of a tile resulting in the provision of a window that can be used to communicate with the program or review the file (Ex. 1001 at 9:24-56), a person of ordinary skill would understand that tiles are selectable, and wherein “selecting a tile calls an assigned application program to provide access to information” as required by the substitute claims. This is because the description of the icon providing a window would be understood by one of ordinary skill in the art to refer to an application, called in response to the selection of the icon, running in the window; and because one of ordinary skill in the art would understand that replacing the functionality of the desktop would include replacing the functionality of the icons on the desktop, which would include the icon functionality of calling an application in response to selection of the tile to provide access to the information source (e.g., a word processing document) associated with the tile.

37. FIG. 13 of the ’403 Patent, (and Fig. 5 of the Provisional Application, Ex. 2066) also indicate that tiles are selectable. One of ordinary skill in the art

SurfCast – Exhibit 2081, p. 20  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

would recognize Fig. 13 as pseudo markup language, similar to HTML, that causes three tiles to be displayed. Each of the tiles includes an “HREF” instruction, which is known to those of ordinary skill in the art as an HTML instruction to signify a link that can be selected by the user to navigate to, in the case of first tile of Fig. 13, an html file with the URI [www.somewhere.com/sometarget.html](http://www.somewhere.com/sometarget.html).

38. I note that in his Reply Declaration, Dr. Karger has cited deposition testimony from Mr. Santoro and myself regarding a ’403 Patent tile providing access to information before a tile is selected. (Ex. 1110 at ¶ 77-78.) As discussed below (in ¶ 64), it is my understanding that the U.S. Court of Appeals for the Federal Circuit has indicated that inventor testimony is irrelevant to claim construction, and this testimony of Mr. Santoro is clearly relied on by Dr. Karger for what it implies about Mr. Santoro’s construction of the claim terms.

Furthermore, it is not clear to me why this testimony is being cited. It may be that Dr. Karger is arguing that the fact that a tile provides access to information before it is selected means that the tile cannot be selectable (to call an assigned application program) as required by the substitute claim. If that is the case, then I disagree with such an argument. There is no requirement that the *only* access to information must be the result of selecting the tile. Such a requirement would be contrary to the clear disclosure in the ’403 Patent that 1) tiles display refreshed content (e.g., Ex. 1001 at 8:35-37), and 2) that “when a tile is selected ... the tile

instantly provides the user with access to the underlying information.” (Ex. 1001 at 9:24-26).

39. Furthermore, although the '403 Patent makes clear to one of ordinary skill in the art at the time of the '403 Patent that applications can be called from tiles in a grid by selecting a tile in the same manner that applications can be called by selecting icons (Ex. 1001 at 11:15-31), nothing in this passage indicates that tiles are the sole mechanism for calling applications. It was well known to those of ordinary skill in the art at the time of the '403 Patent that Microsoft's operating systems, such as Windows 95 and Windows 98, included a Start button on the taskbar (typically located along the bottom of the screen) that, when pressed, provided a menu-driven mechanism for calling applications to provide access to underlying information. This ability to call an application to provide access to underlying information was an alternative to the desktop shortcuts/icons. Furthermore, those versions of Windows also included a command line interface through which a user could call an application, providing a second alternative to desktop icons/shortcuts for calling applications. Furthermore, one could open a “Run” dialog from the Start menu, type in the path and name of an application program, and thereby execute it. Still other ways existed to call applications, but these examples suffice to prove that one of ordinary skill in the art would understand that replacing the desktop and its icons/shortcuts with an array of tiles

as disclosed in the '403 Patent would not eliminate these other methods of providing calling applications to provide access to underlying information.

40. I note that in paragraphs 68 and 69 of Exhibit 1110, Petitioner's expert attempts to further construe and add ambiguity to, the Board's claim construction of a "tile" as "a graphical user interface element . . . that, when selected, provides access to an information source." Specifically, Petitioner's expert re-interprets the word "when" by volunteering that it has two "common" meanings, allowing the phrase "when selected, provide access to an information source" to mean either: "(i) the provision of access to the information source **at the time that** the tile is selected, or (ii) the provision of access to the information source **during the time that** (i.e., while) the tile is selected. Ex. 1110 at ¶ 69 (emphasis in the original).

41. Microsoft's expert does not allege that he reviewed the '403 Patent to determine which of the alternative interpretations of "when selected" are actually supported by the '403 Patent. He cites to nothing in the '403 Patent to support interpretation (ii). The '403 Patent clearly supports interpretation (i). For example, the '403 Patent states that "When a tile is selected, whether by mouse click or otherwise, the tile instantly provides the user with access to the underlying information." (Ex. 1001 at 9:25-27.)

42. It is my opinion that interpretation (ii), *i.e.*, allowing the mere display  
SurfCast – Exhibit 2081, p. 23  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

of information in a tile as providing access to the information source, would render the “when selected, provide access to an information source” portion of the Board’s claim construction meaningless because access would be provided without the tile being selected.

43. Nevertheless, the Substitute Claims’ language that “the tiles are selectable, and wherein selecting a tile calls an assigned application program to provide access to information” makes it clear to one of skill in the art that the alleged ambiguity raised by Microsoft’s expert would not apply to the Substitute Claims.

#### **F. Not permitted to overlap**

44. Patent Owner seeks to amend claim 46 to recite a “grid of tiles that are not permitted to overlap.” Dr. Karger, in his declaration, has interpreted this term to be a “negative limitation,” which “may be satisfied by silence in the prior art.” (Ex. 1110 at ¶ 16.) Therefore, Dr. Karger interprets the claim limitation to “require[] the absence of functionality rather than the presence of functionality”. Ex. 1110 at ¶ 285. I disagree with this analysis.

45. First, the requirement “not permitted” speaks to an active process, or a positive feature, not to a negative limitation as Dr. Karger supposes. This is because any software that implements a grid of tiles in which the tiles are con-



strained from overlapping must have the requisite code in place to test for overlap and to reject it when it occurs, for example, when a user attempts to relocate a tile in an overlapping position. It is true that there are some display systems, as Dr. Karger points out in deposition (Ex. 1113 at p.17 line 15, to p. 18, line 5) that use boundary lines to separate windows from one another. A boundary line acts as an edge shared between two windows in these systems; nevertheless the properties of a boundary line that define the edges of pairs of windows are themselves positive features, coded in such a way that the two abutting windows do not encroach upon one another. And, as Dr. Karger states, there are other ways of implementing a tiled user interface than a system with the boundary lines that he describes, in which case “the system itself just takes care of re-laying out the windows” which is a positive limitation performed by “the system” (Ex. 1113, 28:17-24).

46. Second, Dr. Karger’s example of a covered window system that crashes when windows overlap because it omits the code that prevents overlap (Ex. 1113 at p.32, lines 9 – 13) is in itself not to be taken seriously. Such a system might not allow windows to overlap but it would also not provide a properly functioning window system to a user.

47. In fact, under cross examination, Dr. Karger had previously admitted that “with respect to a tiled windows system, ..., one way in which that system could be implemented would be to include some boundaries that are *enforced by*

SurfCast – Exhibit 2081, p. 25  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

*the software*”, which is in itself a positive limitation. Ex. 1113 at p.24 line 24 – p.25, line4.

48. In my view, in the context of tiles placed on a grid, the constraint that those tiles not be permitted to overlap is a real and specific limitation that would have to be coded in, and could be implemented in a number of different ways. For example, if cells in the grid were of a fixed size, the same size as that of a tile, then the software could require that a tile, when placed in the grid, is fitted exactly into a cell, aligned with each edge of the cell. This would be accomplished via software that prevented a tile from overlapping cell boundaries in the grid, and thereby overlapping other tiles.

49. Another way of implementing a constraint that tiles in the grid do not overlap would be to prevent a user from moving or resizing a tile in any way that caused it to occupy a cell in the grid that was already occupied by another tile. A further way to implement this feature would be to — optionally — force an existing tile to move to another cell if a user desired to place a new tile in a potentially overlapping position.

### **3. Patent Owner addressed the most relevant prior art**

50. Petitioner alleges that Myers is “highly relevant prior art” that was not addressed in Patent Owner’s Motion for Conditional Amendment. Opp. at 4. It is

my opinion, however, that Myers is clearly less relevant than other references analyzed by Patent Owner in the Motion, for several reasons summarized in the following paragraph and explored further in Section 5.C herein, and that there was no reason to specifically address Myers in the Motion.

51. As Petitioner's expert admitted on cross examination, apparently contradicting his statement in his Reply Declaration (Ex. 1110 at ¶340), Myers' icons are not selectable to call assigned application programs as is required by what Petitioner has identified as the "principle [*sic*] amendment" (Paper No. 46 at 2). Ex. 1113 at 170:7-171:3.<sup>4</sup>

52. Petitioner's expert also admitted under cross examination that Myers' icons are not persistent. (Ex. 1113, at 197:14-19)

53. Petitioner's expert further admitted that his invalidity position based on Myers is one of obviousness that relies on MSIE Kit, whereas he contends that two other references analyzed in Patent Owner's motion, MSIE Kit and Kostas, are anticipatory. (Ex. 1113 at 182:20-24)

---

<sup>4</sup> It is inconsistent for Dr. Karger to simultaneously point to a reference that describes an unusual and creative system, and then state that a person of skill would have imparted traditional behaviors to that system. The fact that Myers' "icons" vanish when their associated window is closed (Ex. 1066, p. 7) is evidence that these icons do not behave in a traditional manner.

54. For all the above reasons, and as explored further in Section 5.C below, Myers is far less relevant than either of the MSIE Kit or Kostas references, which were analyzed in Patent Owner’s Motion (Paper No. 28 at 8 and 13).

#### **4. Patent Owner provided Written Description Support for Proposed Substitute Claims**

55. The limitation “calling an application by selecting a tile” is clearly supported by the portions of the written description cited in the Motion for Amendment (Paper No. 28 at 5), and for the reasons set forth above in paragraphs 29 – 30. I also find support in the ’403 Patent at least at 12:24-30 (“Fig. 11 shows an arrangement of tiles in which are depicted different application programs associated with three of the tiles. Tile 802-2-2 links to a file viewer displaying a specific tile; tile 802-N-1 presents streaming video; tile 802-M-N depicts a page of information on the world wide web”).

56. The persistent display of a grid of tiles is also supported by the specification of the ’403 Patent, which discloses that the grid of tiles is stored in a file system (Ex. 1001 at 12:66-13:1), and Petitioner’s expert admitted that one of ordinary skill in the art would understand that the file system is typically stored in a non-volatile hard disk. (See *e.g.*, Ex. 1113 at 125:8-9 and 126:2-7). See also the discussion at ¶¶ 7 – 26 above.

## 5. Proposed Substitute Claims 53 – 59 Are Patentable Over the Prior Art

### A. MSIE Kit

57. The Opposition alleges that MSIE Kit (Ex. 1007) anticipates substitute claims 53 – 59. (Opp. at 9-12.) This section addresses each of Petitioner’s expert’s arguments, including errors of logic as well as errors of fact.

58. **No selecting a tile to call an assigned application program.** MSIE Kit does not disclose “wherein selecting a tile calls an assigned application program.” The Active Desktop items in MSIE Kit do not call any application program when they are selected, and neither Petitioner nor its expert identify anything in MSIE Kit to the contrary. Instead, Petitioner and its expert rely on selecting a hot spot, link, or control *within* an Active Desktop item to satisfy this limitation. (Opp. at 10; Ex. 1110 ¶¶ 120-139 and 213-216.) Selecting a link or some other object *in* an item is different from selecting the item *itself*. (We will see that the same fundamental distinction, among others, exists for the Kostas reference (Ex. 1051), the other primary reference presented by Dr. Karger.)

59. As I have stated above, selecting a link or some other object in an Active Desktop item is different from selecting the Active Desktop item itself. For example, if an Active Desktop item displays a website without making any links visible within that item, then such an item could not be selected to call an application as required by the Substitute Claims. In fact, nothing would happen if

SurfCast – Exhibit 2081, p. 29  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

a user clicked on an Active Desktop item that did not contain a link visible in that item. This is admitted by Microsoft's 30(b)(6) witness Joseph Belfiore in his deposition testimony in the concurrent District Court litigation. Specifically, when asked if "anything" would happen if a user clicked on an active desktop item that did not have a link within it, Mr. Belfiore said "No." (Ex. 2088 at 172:7-173:4.)

60. Dr. Karger conflates or confuses "selecting an item" with "selecting something inside the item" at many points in his Reply Declaration, and he also goes through a variety of verbal gymnastics to gloss over this fundamental distinction between this allegedly anticipatory reference and the amended claims of the '403 Patent. For example, in ¶ 213 he writes that "Active Desktop Items are selectable in several ways, including by clicking on a link or hot spot or using keyboard navigation," but that statement is false in the first instance and off-point in the second: the Active Desktop *item* is not selectable "by clicking on a link or hot spot," and selecting an Active Desktop item "using keyboard navigation" (as he describes in ¶216 with TAB and SHIFT+TAB) does not select it *to call an assigned application program* and is therefore irrelevant, not being what the claim requires. For another example, in ¶215, he states that "Active Desktop also allows users to interact with content displayed in a Desktop Item...", which may be true but which has nothing to do with the claim language. For yet another example, Dr. Karger equates "selecting an item" with "selecting something inside the item" in

SurfCast – Exhibit 2081, p. 30  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

¶ 230 (“...when someone selected a mailto link in a Desktop Item, and thus selected the Desktop item...”). Equating a *contained* object to the *containing* object is fundamentally incorrect analysis, no matter how many times Dr. Karger repeats it.

61. For example, in ¶ 242, Dr. Karger refers to “application programs described above that may be called when an Active Desktop item is selected.” However, none of the programs he mentions are “called when an Active Desktop item is selected,” and Dr. Karger’s statement here is incorrect. Dr. Karger posits a scenario in which “...the system is configured to call a Netscape Browser as the web browser called when an Item is selected...” (Ex. 1110 at ¶ 218) and ignores the fact that *no* application is *ever* called “when an Item is selected.” The same error affects Ex. 1110 at ¶ 217, in which Dr. Karger discusses a link that calls Internet Explorer.

62. The same problem affects the various “helper” applications that Dr. Karger discusses in ¶¶ 219-231 of Ex. 1110. The helper applications, be they Microsoft Chat, VDO Live Player, RealPlayer, Outlook Express, NetShow, NetMeeting, etc. – are not Active Desktop items. They may be invocable from within the Active Desktop items, but they are not themselves Active Desktop items, and the basic problem is the same as with Dr. Karger’s attempt to characterize a link within an Active Desktop item as the same thing as the Active Desktop item itself. Again, Dr. Karger glosses over the issue with wording such as

SurfCast – Exhibit 2081, p. 31  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

“...it would have been obvious to employ helper applications with Active Desktop...” (Ex. 1110 at ¶222) but, again, the claim language says nothing about “employing” anything with anything else.

63. Furthermore, the “helper” applications are not assigned to the Active Desktop item. They may be said, instead, to be assigned to the link (or other object) *inside* the Active Desktop item. So these examples of helper applications fail the “assigned” limitation as well as the “selecting” limitation, and statements such as Dr. Karger makes in Ex. 1110 at ¶ 225 (“...the Netshow streaming video application would be ‘assigned’ to a Desktop Item ...”) are incorrect on their face. If an Active Desktop item displays a website without making any hot spots, links, or controls visible within it, then such an item could not be used in any way to call any application. In contrast, a tile according to Substitute Claim 53 is a significant improvement, because it will call an application program regardless of whether the information in the tile includes any links or has any hotspots or other objects.

64. I should also note that ¶ 133 of Ex. 1110 (referenced and relied on in ¶ 213 thereof to support his assertion that MSIE Kit discloses the “wherein tiles are selectable, and wherein selecting a tile calls an assigned application program to provide access to information, the assigned application program being different from a program that arranges the display into the grid of tiles” amendment) cites

the testimony of one of the inventors, Mr. Lagermann, to support his (Dr. Karger’s)  
SurfCast – Exhibit 2081, p. 32  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292



interpretation of that limitation. I have been informed by counsel that the U.S. Court of Appeals for the Federal Circuit has held that the inventor's testimony regarding the meaning of claim terms is irrelevant to, and unreliable for, claim construction, regardless of whether that testimony is helpful or harmful to any interest the inventor may have in a proceeding. The portions of Mr. Lagermann's transcript relied on by Dr. Karger in ¶ 133 of his report (Ex. 1110) are nothing more than an illustration of Mr. Lagermann's views on the construction of this term, and therefore I will follow the Federal Circuit's teaching by not relying on this testimony.

65. **No grid.** Furthermore, MSIE Kit does not unambiguously disclose a grid. MSIE Kit consists of a book and a CDROM containing Internet Explorer 4, with Active Desktop functionality built into it. When I installed Active Desktop from the CDROM, I did not in fact see a grid of non-overlapping Active Desktop items when I added items to the Active Desktop; on the contrary, the "default" behavior was for overlapping to occur. At best, therefore, taking MSIE Kit as it is packaged (book plus CDROM), it is self-contradictory on this point, and a person of skill in the art would not take the printed book's comments regarding grid functionality at face value, given that the accompanying software contradicts them.

66. **No desktop replacement.** In an effort to show that Active Desktop is similar to the invention of the amended claims, Dr. Karger writes (Ex. 1110 at  
SurfCast – Exhibit 2081, p. 33  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

¶ 181) that Active Desktop “can replace the existing desktop graphical user interface,” but this statement is not supported by the MSIE Kit and it is also not true. Icons, the Start menu, and the taskbar all still exist (and are needed) with Active Desktop. See, for example, MSIE Kit, at p. 177: “[Active] Desktop items live on the Windows desktop, alongside existing desktop shortcuts.” My own testing of Active Desktop demonstrates that desktop shortcuts, icons, the Start menu, and the Taskbar all remain present when Active Desktop is installed.

67. **No requirement of non-overlapping grid.** Dr. Karger’s reasoning for why Active Desktop items are “tiles” has already been addressed and remains incorrect. As for his reasoning that p. 183 of MSIE Kit “discloses a non-overlapping grid when the first six items are displayed” (Ex. 1110 at ¶188), I respond as follows:

- A “non-overlapping grid” is not what the revised claim language requires. It requires “tiles that are not permitted to overlap.” These are not the same things; for example, a person who doesn’t drive is not the same as a person who is not *permitted* to drive.<sup>5</sup>

---

<sup>5</sup> Dr. Karger’s logic regarding “negative limitations” (e.g. in ¶ 338 of Ex. 1110) is inapplicable here, as has previously been discussed (¶¶ 44-45, herein). An invention that *prohibits* behavior – such as, here, the overlapping of tiles – is not an

- The figure on p. 176 of MSIE Kit (Ex. 1007) shows two desktop components – significantly less than the six in the purported “grid” – and they are shown to be overlapping. Therefore, Dr. Karger’s presumption of non-overlapping with six or fewer items seems to be contradicted by his own reference.

68. In a new argument, Dr. Karger points to ways of locking down the Active Desktop configuration (Ex. 1110 at ¶190-191). “Not permitted to overlap” is not the same thing as “not permitted to change,” any more than “not permitted to eat meat” is the same thing as “not permitted to eat.” A total lockdown of the grid of tiles in the ’403 Patent (which is what the “prohibit changes to Active Desktop” setting would do) would sacrifice many of the benefits of the invention by not permitting a user to modify their primary computing environment in any respect

---

invention that *ignores* that behavior. Some cars do not permit the driver to start the car without placing a foot on the brake. That safety feature is inventive and proactive, and requires circuitry, logic, and hardware to implement. The fact that its effect can be stated in the negative – just as the amended claims’ requirement that tiles not overlap can be stated in the negative – does not mean that the result happens “by default,” or is equivalent to any system that does not include a similar proactive functionality. In fact, Dr. Karger’s position is internally inconsistent with his allegation that MSIE Kit discloses non-overlapping tiles through the provision of administrator “lockdown” functionality (Ex. 1110, ¶ 191), a functionality not necessarily found in a system that allows tiles to overlap.

whatsoever; preventing tiles from *overlapping*, on the other hand, sacrifices none of those benefits and permits a wide range of useful and desirable customizing possibilities, including rearranging tiles, changing tile refresh rates, adding tiles, deleting tiles, *etc.*

- The only way for Dr. Karger’s scenario to even come close to the claim limitation is if every user in the company had already configured his or her Active Desktop setups to be non-overlapping when the network administrator enforced the “prohibit changes” rule. That seems like an unlikely scenario, given that even with as few as two Active Desktop items, MSIE Kit shows overlapping (p. 176); and even then, as noted above, the resulting environment would be substantially different (and much more restrictive) than an environment in which only the specific limitation of non-overlapping tiles is in force.

69. **No “different” program.** Dr. Karger makes various attempts to put forth the hypothesis that Active Desktop items, when selected, call a different program, as required by the language of the substitute claims. (Ex. 1110 at ¶235-237.) They are all incorrect, as will be demonstrated below.

70. First, Dr. Karger has not shown that *any* application is called as a result of Active Desktop items being selected, as discussed above. And, in fact, I find no such disclosure in MSIE Kit.

71. Second, as to the “different” aspect of the claim limitation, Dr. Karger declares (Ex. 1110, at ¶ 235), after listing some of the Dynamic Link Libraries (DLL’s) that contain Web functionality, that “A person of ordinary skill in the art would understand these disclosures as showing that the Active Desktop gridding functionality is a different set of software instructions from those that are called when the web browser is launched when a Desktop Item is selected.” This statement is incorrect for multiple reasons:

- a. None of the DLL’s listed by Dr. Karger (Ex. 1110 at ¶234) say anything about “Active Desktop gridding functionality.”
- b. Nothing, in fact, in MSIE Kit points to “Active Desktop gridding functionality” as residing in any separate or different program, or even program module.
- c. The “Active Desktop gridding functionality” appears from my own testing to be a fiction, at least with respect to overlapping.
- d. DLL’s are not programs, but rather program modules, that may be combined to constitute programs.
- e. The substitute claims do not refer to “set of software instructions”, but refer to “programs,” which are not the same thing.
- f. No web browser is ever “launched when a Desktop Item is selected,” as explained in previous sections of this declaration.

SurfCast – Exhibit 2081, p. 37  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

72. Microsoft goes to considerable effort to point out that Active Desktop and IE4 are merely alternative views of the same thing, as the following observations and citations demonstrate (emphasis added):

- a. The code modules that are resident in memory on an Active Desktop PC are the same ones used by Internet Explorer.
- b. The first line of the Active Desktop chapter (#14) in MSIE Kit describes “The *Internet Explorer 4* Active Desktop interface.” (p. 174)
- c. The second paragraph of the Active Desktop chapter in MSIE Kit begins “Active Desktop technology for Internet Explorer 4...”.
- d. MSIE Kit also refers to “The Active Desktop in Internet Explorer 4...” (p. 175).
- e. MSIE Kit states on p. 175 that “Active Desktop items are hosted in an instance of the WebBrowser control” which calls IE code.
- f. MSIE Kit states on p. 176 that “Active Desktop items and Active Desktop wallpaper are, essentially, Web sites that appear directly, in borderless frames, on users’ desktops.”
- g. MSIE Kit states on p. 180 under the heading “Configuring  
SurfCast – Exhibit 2081, p. 38  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

Active Desktop Items” that “When you subscribe to a URL, site information is automatically cached on your computer for offline viewing, and *Internet Explorer 4 will automatically update the desktop item* on a schedule you choose.”

- h. The citation that Dr. Karger relies on to provide evidence of grid functionality states that: “By default, *Internet Explorer* lays out new Active Desktop items on a 3 by 2 grid” (p. 183).
- i. Dr. Karger states that MSIE Kit “...describes features of Microsoft’s Internet Explorer 4, including Microsoft’s Active Desktop technology” (¶180).

73. The fact that Active Desktop items appear in borderless frames and do not have the usual window attributes (scroll bars, menus, etc.) does not mean that they are not browser windows. It has long been possible to remove menus, toolbars, scrollbars, etc. by invoking Internet Explorer with the appropriate parameters, e.g., with JavaScript. Developers of the day even coined a term for this type of browser display: “chromeless.” I further remember that Microsoft’s own “HTML Applications,” which bore the filename extension **.hta**, were basically browser windows without toolbars, scrollbars, or other typical window appurtenances. It was also possible to run Internet Explorer “full screen” in something known as “kiosk mode” – again, it looked different, but it was still a

browser. A browser without a menu, scrollbars, etc. is still a browser, running browser code, displaying HTML, and so forth. Therefore, Dr. Karger's analysis in ¶236 of Ex. 1110 is incorrect.

74. Dr. Karger's incorrect conclusion (Ex. 1110 at ¶236) is in no way supported either by Klaus Lagermann or by me. With respect to Mr. Lagermann, the view that Active Desktop is a portion of the Windows operating system has no bearing on whether Active Desktop is "different" from Internet Explorer, and with respect to my testimony in *U.S. v. Microsoft*, my testimony that Internet Explorer is an application distinct from the Windows operating system equally has no bearing on whether Active Desktop is "different" from Internet Explorer.

75. **No uniform assignment of attributes.** Dr. Karger applies incorrect logic in Ex. 1110 at ¶248. He argues that if it is possible to specify a retrieval rate for Active Desktop items, but a user does not do so, then there must be some default retrieval rate. That reasoning ignores the possibility that *no* retrieval rate will be applied if the user does not select one.

76. Furthermore, the "lack of assigning" an attribute cannot be deemed to be the same as the assigning of an attribute. Dr. Karger fails to show an active assignment of an attribute (his choice being refresh rate) and therefore he cannot argue that Active Desktop anticipates substitute claim 55.

77. In Ex.1110 at ¶250, Dr. Karger states that "Chen discloses providing a  
SurfCast – Exhibit 2081, p. 40  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292



default retrieval rate for tiles,” but this is incorrect, because Chen does not disclose tiles. Chen discloses windows. They are non-overlapping, but they are not tiles, at least because they are not selectable in the language of the amended claims.

78. For the same reason, Dr. Karger’s statement in Ex. 1110 ¶251 regarding Chen and MSIE Kit showing “schemes to display tiles” is incorrect, as neither Chen nor MSIE Kit disclose tiles, as required by the substitute claims.

79. Regarding Dr. Karger’s statements regarding obviousness (Ex. 1110 at ¶251-2), I have already addressed why it would not have been obvious to combine Chen and MSIE Kit; those arguments refute these paragraphs.

#### **B. Kostas**

80. Like MSIE Kit, Kostas (Ex. 1051) does not disclose “wherein selecting a tile calls an assigned application program.” Petitioner’s expert admitted in cross examination that his opinion was based on the alleged tiles in Kostas being the items in the “montage view”, not the “netchannels” or “main view.” Ex. 1113 at 137:10-18. However, Kostas does not disclose that the items in the montage view call an application program when selected.

81. Additionally, the netchannels themselves cannot be selected to call an application program. Rather, a link in the netchannels can be selected to call an application program. But this is different from selecting the netchannel itself to

SurfCast – Exhibit 2081, p. 41  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

call an application program, for the same reasons discussed above in connection with MSIE Kit. This distinction alone is enough to disqualify Kostas as anticipatory of the substitute claims, just as it is with MSIE Kit. Nevertheless, there are other problems with Dr. Karger's analysis that render the Kostas reference even less anticipatory.

82. Throughout Dr. Karger's analysis of Kostas, he conflates the upper part of Kostas' display (the "main view" or "netchannels") with the lower part (the "montage view"). The upper part (main view) has netchannels and, potentially, links. The lower part (montage view) has non-overlapping regions, called "panes," that are described as presenting applications. Dr. Karger talks about the "rectangular regions" in both the context of the main view and the montage view, but in fact these two kinds of "rectangular regions" occur in two distinct and distinctly different parts of the display that behave differently. Dr. Karger recognizes that that "you have two different windows here" and that they are "separate thing[s]" in his deposition testimony (Ex. 1113 at 209:18-22). No single "rectangular region" in Kostas meets the requirements of the amended claim language, and it would not be obvious to "cherry-pick" attributes of "main view" rectangular regions and combine them with attributes of "montage view" rectangular regions in an attempt to construct the "tiles" of the substitute claims.

Certain functionality of those features is dictated by the reference itself, which has

SurfCast – Exhibit 2081, p. 42  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

set out a scheme in which functionality is divided between two different types of item. One of ordinary skill in the art would not think to make the montage view panes selectable in the sense of Patent Owner's substitute claims, because they already have applications running in them.

83. On the contrary, I believe that a person of skill in the art would regard Kostas' division of the display into a "main view" containing netchannels, which themselves contain content that may be selectable, and a "montage view" containing content that appears not to be selectable, contrary to the invention of the substitute claims, with its unified grid of tiles, rather than anticipating it.

84. In Ex. 1110 at ¶270, Dr. Karger states that "Each rectangular region can display content from a different information source... and each such information from such information source refreshed at a specified interval." This is incorrect. There is no showing in Kostas that the "rectangular regions" in the montage view can "refresh at a specified interval." There is an example of a "market minder" application in the montage view that updates "continuously" but there is no showing in Kostas of updates in the montage view according to a schedule. In deposition, Dr. Karger corrected his report and stated that this sentence in ¶ 270 of Ex. 1110 "...is referring to the panes of the montage view." (Ex. 1113, 137:10-15) Dr. Karger also admitted in deposition that there is no

explicit disclosure in Kostas that the applications running in the montage view use

SurfCast – Exhibit 2081, p. 43  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

the “model view services” referred to on page 20 of Kostas (Ex. 1113, 142:14-330:25).

85. In ¶ 285 of Ex. 1110, Dr. Karger applies his “negative limitation” theory in an attempt to equate the display of Kostas, which Dr. Karger says “has no discussion of functionality to allow its rectangular regions to overlap,” with the requirement of the substitute claims, which proactively prevent tiles from overlapping. My analysis on this point was presented above, in ¶¶ 44 – 49.

86. Ex. 1110 at ¶ 286 states that “Each rectangular region may be refreshed.” However, this is incorrect. Kostas does not disclose that each “rectangular region” of the montage view is able to be refreshed. Second, Dr. Karger states that “Each rectangular region may also be selected.” This, too, is incorrect; there is no disclosure in Kostas that *any* rectangular region may be selected.

87. I have already discussed that selecting a link within a netchannel is not the same as selecting a netchannel. As to the montage view, Kostas is silent on how the applications represented by its “rectangular regions” are invoked. This is most likely because the rectangular regions of the montage view are already activated, that is, launched and running. In other words, the montage view is essentially a collection of windows with applications running in them. See for example Kostas on p. 11: “The montage view makes it easy for the user to manage two or more applications running concurrently.” See also Kostas at p. 13: “Rather

SurfCast – Exhibit 2081, p. 44  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

than floating each stand-alone application in its own window, the montage view contains a two-dimensional segmentable workspace in which applications can be placed.” The fact that montage view applications are already running would also explain why Kostas discloses nothing about updating the montage view applications according to a schedule; each pane is effectively a window, and the application window handles its own updates.

88. Therefore, Dr. Karger’s conclusion that one of ordinary skill in the art would understand that in order to interact with email a user must have selected the pane of Kostas, is not supported. Furthermore, Kostas cannot anticipate the substitute claims, because in Kostas there is no need to call an application assigned to the tile, given that the applications are already running and fully visible.

89. Ex. 1110 at ¶ 289 provides evidence for the persistence of the arrangement of “panes,” but does not provide evidence for the persistence of the arrangement of netchannels.

90. The discussion of ¶ 296-299 in Ex. 1110 apply only to netchannels in the main view. The same is true of ¶ 301. As noted above, there is no disclosure in Kostas that the netchannels may be selected, merely that the netchannels can contain selectable links.

91. Dr. Karger is again incorrect in ¶304 of Ex. 1110 when he states that “...users can call various applications... by way of the rectangular panes of the

SurfCast – Exhibit 2081, p. 45  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

Kostes display.” Again, Kostas is silent on how the user can “call various applications” except by being able to click a link inside a netchannel, and I have already noted that Kostas applications appear to be already running, with no need of being “called”. Furthermore, the amended patent claims do not state that an application can be run “by way of” a rectangular pane.

92. In ¶305 of Ex. 1110, Dr. Karger says it would have been obvious to use selection of the Kostas panes to call an application program. I do not agree, and Dr. Karger has not made a persuasive case. Active Desktop did not work that way. Chen did not work that way. Sapphire did not work that way. And these are the references that Dr. Karger deems most persuasive, among all possible prior art, to render the substitute claims unpatentable.

93. In Ex. 1110 at ¶312, Dr. Karger is factually incorrect. Selecting a link in a netchannel is not the same as “selecting a rectangular region.”

94. In ¶313 of Ex.1110, again, Dr. Karger is incorrect. Nothing in his citation refers to selecting a rectangular region. Kostas provides no disclosure of how a user would activate the described “unified inbox.”

95. In the discussion in ¶¶321 – 325 of Ex. 1110, Dr. Karger seems to be arguing that because software classes can exhibit inheritance, some unnamed attribute is assigned uniformly to each “tile” and then it somehow becomes a “default retrieval rate.” By this reasoning, every software product built on a

SurfCast – Exhibit 2081, p. 46  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

programming language (like Java) that supports class inheritance would uniformly assign an attribute to its user interface objects, which is clearly not the case. Kostas shows no disclosure of uniformly assigning any attribute of any “tile” (setting aside that no one rectangular region in Kostas satisfies the requirements of a “tile”).

### **C. Sapphire**

96. Dr. Karger cites the Myers article (Ex. 1066) as an obviousness reference when combined with MSIE Kit (Ex. 1007). This article presents certain details about an operating environment including a window manager and related functionality. In my opinion, Myers describes a very different system from the amended claims, even when combined with MSIE Kit.

97. *Sapphire icons are not persistent.* As Dr. Karger states in his deposition of April 2014, “But if I turned off a Sapphire machine and turned it back on, those icons are probably gone.” (Ex. 1113 at 197:17-19). Myers states that when a user closes a window, the icon associated with that window disappears. (“Thus when a window is deleted, its icon is deleted, but the hole is not filled until another window is created.” Myers, Ex. 1066 p. 7.) This is a reflection of the fact that, as Dr. Karger explains, “[a]gain, with Sapphire, there is this one-to-one association between applications and icons” (Ex. 1113 at, 175:23-25).

98. *Sapphire icons are not selectable to run an application.* In fact, it is not clear to me from the Myers reference how a user could run an application.

SurfCast – Exhibit 2081, p. 47  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

(Changing the focus of the environment by designating one program as the “listener,” as Dr. Karger discusses in his deposition (Ex. 1113 at 194:5-13), is not the same as running an application.)

99. One could certainly not run an application by somehow selecting an icon, if the icon is absent when the application is not running, as described in the preceding paragraph. As additional evidence that the Sapphire icons are not used to run applications, Myers writes, “Thus, the icons as a group can be moved around or removed altogether if the user does not like icons.” (Ex. 1066, at p. 7.) The icons therefore appear to be an optional, and dispensable, part of the Sapphire user interface.

100. The “pop-up menus” discussed in Myers p. 8 appear to be application-specific, such as “Help” (see Ex. 1066, Fig. 6), as further suggested by Myers’ statement on p. 9 of the Exhibit that “The pop-up menu provides all of the commands that can be given directly from the title line” (i.e., the title line of a specific window). There is also no disclosure of Myers’ “accelerators” being able to run an application (Ex. 1066, p. 8).

101. Dr. Karger shares my puzzlement on this point, stating in his deposition that “Now, as far as I can tell, the document does not describe in general how applications will be launched. And in the question that you asked, if I close all of my icons – I’m sorry, if I close all of my applications, it’s not clear whether

SurfCast – Exhibit 2081, p. 48  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292



Sapphire comes with any kind of launcher interface, anything that I can use to launch a new application.” (Ex. 1113 at 167:25-168:7) and “So I’m afraid the document – I’m afraid this document is basically moot – mute on the question you asked about how you will start up a new application” (Ex. 1113 at 168:25-169:4).

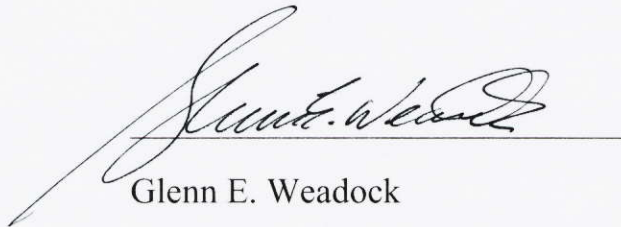
102. Given that MSIE Kit also does not disclose how Active Desktop items are selectable to run an application, the combination of Myers and MSIE Kit together also fails to disclose this limitation of the amended claims.

103. *No refresh rates.* I find no disclosure in the Myers article of the assigning of refresh rates to Sapphire icons. (This appears to be the reason that Dr. Karger proposes a combination of Myers with MSIE Kit.)

104. *No requirement of non-overlapping.* Myers does not disclose that Sapphire icons are not permitted to overlap; in fact, he discloses that Sapphire icons can be obscured, at least by other windows; see, for example, his Fig. 5 (Ex. 1066 p. 6), whose caption begins “A typical Sapphire icon window. Note that this window can be covered by other windows.”

105. Given that MSIE Kit also does not disclose that Active Desktop items are not permitted to overlap, the combination of Myers and MSIE Kit together also fails to disclose this limitation of the amended claims.

106. I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct.

A handwritten signature in black ink, appearing to read "Glenn E. Weadock", is written over a horizontal line.

Glenn E. Weadock

May 9, 2014

## **Appendix**

### **References and Information Considered**

#### Patents

1. U.S. Patent 6,724,403 (Ex. 1001; and its prosecution history and related applications, including references cited therein)
2. WO99/26127 (Kostes; Ex. 1051)
3. U.S. Pat. No. RE 43,119 (Ex. 2085)
4. WO96/01032 (Ex. 2086)
5. U.S. Pat. No. 8,601,198 (Ex. 2087)

#### Depositions

6. David R. Karger, January 14, 2014, and all associated exhibits
7. David R. Karger, April 24, 2014, and all associated exhibits
8. Joseph Belfiore, July 11, 2013.

#### Publications and Resources

9. The New IEEE Standard Dictionary of Electrical and Electronics Terms, Fifth Edition, Gediminas P. Kurpis, Institute of Electrical and Electronics Engineers, 1993
10. Myers (Ex. 1066)
11. MSIE Kit (Ex. 1007)
12. Windows 95 Installation Requirements (Ex. 2078)

SurfCast – Exhibit 2081, p. 51  
Microsoft Corp. v. SurfCast, Inc. IPR2013-00292

13. Minimum Hardware Requirements for a Windows 98 Installation (Ex. 2079)
14. System Requirements for a Windows XP Operating System (Ex. 2080)
15. Memorandum Opinion, March 14, 2014 – District Court of Maine, Case 2:12-cv-00333-JAW (Ex. 2082)
16. Microsoft Computer Dictionary - Definition – Bookmark (Ex. 2083)
17. Microsoft Press Computer Dictionary - Definition – File System (Ex. 2084)