

*Second*, Nokia’s attempt to limit the listen string to a string that has “a fixed portion identifying a service name and a variable portion identifying one or more service types” is contrary to the claims—which expressly define the parameters of the listen string as “an application signature, an application signal type and an application signal port.” (*Id.*, 13:39-40, 14:45-47.) The claims do *not* require those parameters to be fixed or variable.

*Finally*, Nokia’s only support for its proposed additions is a single cite to a portion of the specification describing one particular embodiment. (D.I. 227 [Joint Chart] at 16 (citing ’034 patent, 5:28-40).) But Nokia ignores express language in the specification making clear that the disclosed embodiment is just an “example” of a possible implementation (i.e., the “QuickTime Conferencing” product). (’034 patent, 4:31-37, 5:10-14.) It would be improper to limit the claims to that disclosed example. *Netcraft Corp. v. eBay, Inc.*, 549 F.3d 1394, 1400 (Fed. Cir. 2008) (“An invention is not limited to its examples.”) (internal quotations omitted).

**D. U.S. Patent No. 5,455,854 (“the ’854 patent”)**

The ’854 patent is directed to systems and methods that use a particular type of “object-oriented” software for telephone systems. The parties dispute one term: “telephony element.”

**1. The Solution of the ’854 Patent**

Before the ’854 invention, telephone systems utilized software written with complex and proprietary coding techniques that were rigidly structured and difficult to exploit. These severe limitations impeded developers from designing applications compatible with the many different types of telephone systems—leaving users unable to make use of more than “a tiny fraction of the full capability of most modern phone systems.” (’854 patent, 1:45-46, 4:26-5:2.)

The ’854 inventors solved this problem by developing a new type of “object-oriented” software for use with telephone systems. This new approach—which was more flexible and easier to utilize than the prior art—permitted developers to create telephony software that was

not only adaptable to different proprietary hardware configurations, but also easier to upgrade with new features. This enhanced flexibility is particularly useful for telephone systems that have large hardware and feature variations between different phone models (e.g., cellular telephone systems). (*Id.*, Abstract, 2:28-30, 2:57-62, 5:36-39, 10:64-11:2.)

## 2. “Telephony Element” (Claims 1, 13)

Apple’s Construction	Nokia’s Construction
“Any identifiable aspect of the telephony system”	“A particular hardware element or protocol of the telephone system, with identifiable status and features”

Apple’s construction is taken verbatim from the specification of the ’854 patent, which broadly defines “[t]he *elements of the telephony system*” as “*any identifiable aspect of the telephony system.*” (*Id.*, Abstract.) In equally broad terms, the patent states that telephony elements can be anything that interfaces with application elements—including “physical equipment” (e.g., “a handset or a line”) and “[l]ess tangible elements” (e.g., “protocols” and “signals or procedures, including call progress tones, call setup, call hold, conference calls, or other call features”). (*Id.*, Abstract, 2:38-40, 2:50-53, 10:64-11:2.) See *Silicon Graphics, Inc. v. ATI Tech., Inc.*, 607 F.3d 784, 789 (Fed. Cir. 2010) (“If the specification reveals a special definition for a claim term, ‘the inventor’s lexicography governs.’”).

Ignoring this clear expression of the inventors’ intent to define “telephony element” broadly, Nokia improperly seeks to limit the term to “a particular hardware element or protocol ... with identifiable status and features.” Nothing in the intrinsic evidence requires the telephony element to be “particular” in any way, to have “identifiable status and features,” or to be limited to a “hardware element or protocol.” To the contrary, as detailed above, the specification repeatedly emphasizes that telephony elements may be “*any identifiable aspect* of the telephone system”—regardless of whether they have “identifiable status and features,” and not just a

“particular hardware element or protocol.”

**E. U.S. Patent No. 5,634,074 (“the ’074 patent”)**

The ’074 patent is directed to a “self-configuring startup procedure for communication devices, including telecommunication adapters,” that automatically enables a computer to identify an input/output (“I/O”) device attached to the computer’s serial port. (’074 patent, 1:49-57.) The parties dispute a single term: “beacon signal.”

**1. The Solution of the ’074 Patent**

In the prior art, it was difficult for users to configure a computer to operate with I/O devices attached to the computer’s serial port. That process required complex, time-consuming, and cumbersome manual configuration of devices by users, and, if not done properly, the device would not work properly. (*Id.*, 1:34-46.)

Recognizing the need “to provide easy, affordable access” to the “rapidly expanding” range of telecommunications services, such as fax, data, and voice, the ’074 inventors developed a “plug-and-play” method that permits a computer to automatically recognize and configure connected I/O devices—without user involvement or special purpose hardware. The method works, in part, by having the I/O device send a beacon signal to the computer that identifies the device and allows it to be automatically configured. (*Id.*, Abstract, 1:25-33, 1:49-67, 5:53-63.)

**2. “Beacon Signal” (Claim 1)**

<b>Apple’s Construction</b>	<b>Nokia’s Construction</b>
No construction required (or, “identifying signal”)	“A signal, from which a physical signaling characteristic for use in subsequent communications with the I/O device can be determined, that begins a self-configuring startup procedure”

As used in the ’074 patent, the term “beacon signal” has a plain meaning to a person of ordinary skill in the art, and thus does not require construction. Nevertheless, to the extent the

Court is inclined to construe the term, it should be defined as “identifying signal.”

That construction is consistent with the plain language of claim 1, which confirms that the “beacon signal” is the signal that *identifies* the attached I/O device to the computer: “sending from the I/O device to the computer a *beacon signal* comprising a sequence of bytes *identifying* the I/O device.” (*Id.*, 5:57-63 (further requiring use of the beacon signal to determine “a manner of interaction of the computer with the I/O device”).) The specification also repeatedly states that the beacon signal identifies the I/O device to the computer. For example, it explains that “the I/O device sends across the serial link to the computer a ‘beacon’ signal *identifying* the I/O device” (*id.*, Abstract, 1:60-62); and when describing the example of a “telecom adapter” as the I/O device, the specification explains that the telecom adapter “will send to the host computer a beacon signal that is used to *identify* the type of telecom adapter and to synchronize the computer and the telecom adapter” (*id.*, 3:10-14).

Notwithstanding this clear and consistent disclosure, Nokia seeks to limit the claims by embedding a host of unsupported limitations into “beacon signal.”

*First*, nothing in claim 1 requires the beacon signal to “begin[] a self-configuring startup procedure,” as Nokia would mandate. Indeed, the steps of claim 1 begin with “at least one of a power-up reset signal and a control signal from the computer”—*not*, as Nokia claims, with the I/O device sending a beacon signal to the computer. (*Id.*, 5:57-58.)

*Second*, nothing in the intrinsic record limits the beacon signal to a signal “from which a physical signaling characteristic for use in subsequent communications with the I/O device can be determined.” Nokia relies on a proposed amendment to claim 1 that would have added the limitation “including determining a physical signaling characteristic to apply during subsequent communications with the I/O device.” But the applicants did *not* pursue that amendment, and

instead proposed the language that ultimately issued as claim 1—which, in itself, confirms the amendment is not required. ('074 File History [8/1/95 Resp.] at 934; *id.* [9/3/96 Resp.] at 982; '074 patent, 5:53-63.) In fact, the proposed (but abandoned) language confirms Apple's position by using the open-ended modifier "including"—meaning that the "determining" step is not restricted to physical signaling characteristics, as Nokia claims (in any event, nothing in the cancelled amendment would have required the "physical signaling characteristic" to be "determined" from the beacon signal).

**F. U.S. Patent No. 5,315,703 ("the '703 patent")**

The '703 patent is directed to global notification of changes occurring in a system, such as a computer. The parties dispute three terms: (i) "notification source"; (ii) "connecting a plurality of objects to a notification source"; and (iii) "connection object."

**1. The Solution of the '703 Patent**

Components within a software system must be notified when certain data in another part of the system has changed (e.g., if a user withdraws funds online from a bank account, the new balance must be communicated to the system software used by the bank's ATM and local branches). To allow this, traditional software designs used hard-coded links between each component that changes data, and each component needing to be notified of a change. These restrictive hard-coded designs were difficult to expand when adding new components, each of which needed to be linked through new code to every other component requiring notification of a change. As these types of systems grew in complexity, making such changes became unwieldy.

The '703 patent solved that problem by disclosing an innovative and flexible way to communicate critical details about system changes using an "object based system with a generic

framework for notification.” (’703 patent, Abstract, 1:24-26, 1:33-35.)<sup>10</sup> By using software objects, components that need to be notified of changes can register on-the-fly—something not possible using predefined, hard-coded links. The invention operates, in part, through the use of a “connection object” that interacts with both software components that make changes (i.e., sources) and software components that require notification of changes (i.e., receivers). The connection object stores information for a receiver that a source will later need to notify the receiver of a change. In this way, because the source already knows how to interact with the connection object, notifications can be sent to the receivers without hard-coded links. When a new receiver is added to the software, it only needs to store information about itself in the connection object, with no other software changes required.

**2. “Notification Source” (Claim 8)**

Apple’s Construction	Nokia’s Construction
No construction required; or “source of an indication of an event”	“A software data structure”

No construction is required of the term “notification source,” which has its plain and ordinary meaning of “source of an indication of an event.”

That construction is supported by the plain claim language, which is directed to the “source” of a “notification”—i.e., something that indicates an event has occurred. (Ex. 5 [Merriam-Webster] at 504 (**notify**, “1: to give notice of : report the occurrence of”).) Apple’s construction is also consistent with the specification, which consistently refers to a “notification” as the indication of an event that occurs due to, for example, a change in the source. (’703 patent, 1:6-8 (invention relates “to global *notification of changes* occurring in a system”); *id.*,

<sup>10</sup> The patent uses “object” to refer to object-oriented software or object-oriented programming, in which objects represent features in the computer system. (*Id.*, 3:11-24, 3:39-51.)

1:61-64 (“[A] document state is modified and *notification of the event* is sent to the system.”); *id.*, 17:17-18 (“When a passive collaborator’s selection changes a different *notification event* is sent.”); *id.*, Abstract (“[W]hen a *notification event* is detected...”).)

In its brief, Nokia contends that the Court should construe “notification source” as “a software data structure.” That argument fails for at least two reasons.

*First*, Nokia’s proposed construction finds no support in the intrinsic record, which does not even remotely suggest that the inventors intended to define “notification source” as a mere “software data structure.”

*Second*, beyond its lack of intrinsic support, Nokia’s broad “software data structure” construction would effectively give no meaning to the term “notification.” This too would be error. *See Merck & Co., Inc. v. Teva Pharm. USA, Inc.*, 395 F.3d 1364, 1372 (Fed. Cir. 2005) (“A claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so.”); *Datamize, LLC v. Plumtree Software, Inc.*, 417 F.3d 1342, 1355 (Fed. Cir. 2005) (rejecting proposed construction that “would improperly eliminate the word ‘pleasing’ from the phrase ‘aesthetically pleasing’”).

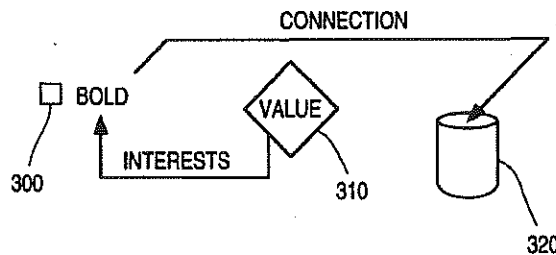
### 3. “Connecting a Plurality of Objects to a Notification Source” (Claim 8)

<b>Apple’s Construction</b>	<b>Nokia’s Construction</b>
No construction required; or “creating at least one connection for a plurality of objects to receive notification from a notification source.”	“At least one of the plurality of objects invoking a method in the connection object to connect to the notification source”

The Court does not need to construe the phrase “connecting a plurality of objects to a notification source,” which has its plain meaning of “creating at least one connection for a plurality of objects to receive notification from a notification source.” Nokia largely agrees with that construction, and also uses the terms “plurality of objects” and “notification source,” so the only remaining word for construction is “connecting.”

Apple’s construction is consistent with the plain language of claim 8, which makes clear that “*connecting* a plurality of objects to a notification source” means creating at least one connection to receive a notification from the source. Indeed, absent such a connection, the notification source could not even communicate with the plurality of objects, as required by other elements in the claim. (’703 patent, 38:5-11 (requiring “selectively dispatching notification to at least one of the plurality of objects,” and “receiving the notification by the at least one of the plurality of objects”).) It is the “connecting” step that allows this subsequent communication to occur—by creating a connection for the objects to receive notification.

The specification is also consistent with Apple’s construction. For example, with respect to Figure 3, the specification describes a “connection” between item 300 (a checkbox of a user interface) and item 320 (data in the software system indicating the checkbox should be checked):



**FIG. 3**

(*Id.*, Fig. 3; *id.*, 18:58-66 (“Everyone *gets to know each other* a little better as shown in FIG. 3.... The checkbox 300, in turn, *connects* to the data 320 for the notifications ...”).) Items 300 and 320 must “get to know each other” before notification can occur. It is the “connecting” step when the creation of this connection occurs.

Nokia’s proposed construction is improper for several reasons.

*First*, Nokia attempts to read an entirely new limitation into the claim by requiring the step of “invoking a method in the connection object.” That “method” appears nowhere in the claim, none of the intrinsic evidence that Nokia cites even refers to a “method” (let alone a



“method in the connection object”), and the specification does not require such a method. The Court should decline to re-write the claim by inserting that new requirement now. *See Quantum Corp. v. Rodime, PLC*, 65 F.3d 1577, 1584 (Fed. Cir. 1995) (“[I]t is well settled that ... courts do not redraft claims.”).

*Second*, Nokia further compounds its error by seeking to require that “one of the plurality of objects” must invoke Nokia’s new “method in the connection object.” Claim 8 does not require the “plurality of objects” to perform the “connecting” step; thus, the Court should not require the “plurality of objects” to perform steps involved with such “connecting.”

#### 4. “Connection Object” (Claim 8)

Apple’s Construction	Nokia’s Construction
“Object that provides the dispatch of a notification from the notification source to at least one of the plurality of objects”	“An object containing a method for selectively dispatching notifications from the notification source to the specific ones of the plurality of objects that have identified to the connection object an interest in specific source events”

Apple’s construction is confirmed by the plain language of claim 8, which explicitly requires “dispatching notification to at least one of the plurality of objects based on the connection registration information stored in the connection object.” (’703 patent, 38:5-8.) That is, the claim makes clear that the “connection object” itself must be involved in dispatching notification to at least one of the plurality of objects, just as Apple proposes.

Apple’s construction also is grounded in the ’703 specification, which explains that a “connection object” acts as an intermediary between the “notifier” (the “notification source”) and the “notification receiver objects” (the “plurality of objects”): “Connection objects provide the dispatch of notifications from the notifier to specific notification receiver objects.” (*Id.*, 11:63-65.) Apple’s proposed construction mirrors this language, but simply replaces “notifier” with “notification source” and “notification receiver objects” with “plurality of objects.”

Ignoring this clear reading of the intrinsic record, Nokia proposes to import two new limitations into claim 8—neither of which is required.

*First*, just as it did for “connecting a plurality of objects to a notification source,” Nokia would require the “connection object” to include “a method”; this time, a method that does the act of dispatching. Again, the intrinsic record does not require that additional method.

*Second*, Nokia’s construction also requires the “plurality of objects” to “identif[y] to the connection object an interest in specific source events.” But once again, there is no support in the claim language or intrinsic record for this limitation.

#### IV. CONCLUSION

For the foregoing reasons, the Court should adopt Apple’s proposed constructions for the patents-in-suit.

<p>OF COUNSEL:</p> <p>William F. Lee WILMERHALE 60 State Street Boston, MA 02109 Tel: 617 526 6000</p> <p>Mark D. Selwyn WILMERHALE 950 Page Mill Road Palo Alto, CA 94304 Tel: (650) 858-6000</p> <p>Kenneth H. Bridges Michael T. Pieja BRIDGES &amp; MAVRAKAKIS LLP 540 Cowper Street, Suite 100 Palo Alto, CA Tel: (650) 681-4475</p> <p>Dated: March 15, 2011 1005066v1 / 35035</p>	<p>POTTER ANDERSON &amp; CORROON LLP</p> <p>By: <u>/s/ David E. Moore</u> Richard L. Horwitz (#2246) David E. Moore (#3983) Hercules Plaza, 6<sup>th</sup> Floor 1313 N. Market Street Wilmington, DE 19899 Tel: (302) 984-6000 <a href="mailto:rhoorwitz@potteranderson.com">rhoorwitz@potteranderson.com</a> <a href="mailto:dmoore@potteranderson.com">dmoore@potteranderson.com</a></p> <p><i>Attorneys for Defendant/Counterclaim-Plaintiff Apple Inc.</i></p>
--	---

**IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF DELAWARE**

**CERTIFICATE OF SERVICE**

I, David E. Moore, hereby certify that on March 15, 2011, the attached document was electronically filed with the Clerk of the Court using CM/ECF which will send notification to the registered attorney(s) of record that the document has been filed and is available for viewing and downloading.

I hereby certify that on March 15, 2011, the attached document was electronically mailed to the following person(s)

**VIA ELECTRONIC MAIL**

Jack B. Blumenfeld  
Rodger D. Smith II  
Morris, Nichols, Arsht & Tunnell LLP  
1201 North Market Street  
Wilmington, DE 19899  
[jblumenfeld@mnat.com](mailto:jblumenfeld@mnat.com)  
[rsmith@mnat.com](mailto:rsmith@mnat.com)

Steven D. Hemminger  
Alston & Bird LLP  
275 Middlefield Road, Suite 150  
Menlo Park, CA 94025  
[steve.hemminger@alston.com](mailto:steve.hemminger@alston.com)

Alan L. Whitehurst  
Scott J. Pivnick  
Thomas W. Davison  
Alston & Bird LLP  
The Atlantic Building  
950 F Street, NW  
Washington, DC 20004-1404  
[alan.whitehurst@alston.com](mailto:alan.whitehurst@alston.com)  
[scott.pivnick@alston.com](mailto:scott.pivnick@alston.com)  
[tom.davison@alston.com](mailto:tom.davison@alston.com)

Kirk T. Bradley  
S. Benjamin Pleune  
Alston & Bird LLP  
Bank of America Plaza, Suite 4000  
101 South Tryon Street  
Charlotte, NC 28280-4000  
[kirk.bradley@alston.com](mailto:kirk.bradley@alston.com)  
[ben.pleune@alston.com](mailto:ben.pleune@alston.com)

Marsha E. Mullin  
Alston & Bird LLP  
333 South Hope Street, 16<sup>th</sup> Floor  
Los Angeles, CA 90071  
[marsha.mullin@alston.com](mailto:marsha.mullin@alston.com)

Michael J. Newton  
Alston & Bird LLP  
Chase Tower, Suite 3601  
2200 Ross Avenue  
Dallas, TX 75201  
[mike.newton@alston.com](mailto:mike.newton@alston.com)

Adam J. Biegel  
Keith E. Broyles  
Patrick J. Flinn  
John D. Haynes  
Ryan W. Koppelman  
Mark A. McCarty  
Peter Kontio  
Matthew D. Richardson  
Byron Holz  
Coby Nixon  
Matthew J. McNeill  
Rohan Kale  
Labriah Lee  
Matthew J. Urbanawiz  
Alston & Bird LLP  
One Atlantic Center  
1201 West Peachtree Street  
Atlanta, GA 30309  
[adam.biegel@alston.com](mailto:adam.biegel@alston.com)  
[keith.broyles@alston.com](mailto:keith.broyles@alston.com)  
[patrick.flinn@alston.com](mailto:patrick.flinn@alston.com)  
[john.haynes@alston.com](mailto:john.haynes@alston.com)  
[ryan.koppelman@alston.com](mailto:ryan.koppelman@alston.com)  
[mark.mccarty@alston.com](mailto:mark.mccarty@alston.com)  
[peter.kontio@alston.com](mailto:peter.kontio@alston.com)  
[matt.richardson@alston.com](mailto:matt.richardson@alston.com)  
[byron.holz@alston.com](mailto:byron.holz@alston.com)  
[steve.mcneiff@alston.com](mailto:steve.mcneiff@alston.com)  
[coby.nixon@alston.com](mailto:coby.nixon@alston.com)  
[matt.mcneill@alston.com](mailto:matt.mcneill@alston.com)  
[rohan.kale@alston.com](mailto:rohan.kale@alston.com)  
[labriah.lee@alston.com](mailto:labriah.lee@alston.com)  
[matt.urbanawiz@alston.com](mailto:matt.urbanawiz@alston.com)

*/s/ David E. Moore*

---

Richard L. Horwitz  
David E. Moore  
POTTER ANDERSON & CORROON LLP  
(302) 984-6000  
[rhorwitz@potteranderson.com](mailto:rhorwitz@potteranderson.com)  
[dmoore@potteranderson.com](mailto:dmoore@potteranderson.com)

941557/35035