

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

QUALCOMM INCORPORATED, and  
SAMSUNG ELECTRONICS CO., LTD.,  
Petitioners

v.

NETWORK SYSTEM TECHNOLOGIES  
Patent Owner

---

IPR2024-00338

U.S. Patent No. 7,373,449

---

**DECLARATION OF DR. TODD MOWRY IN SUPPORT OF  
PETITIONER'S REPLY**

## **TABLE OF CONTENTS**

|      |                                                                                                                                                                      |    |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| I.   | BACKGROUND .....                                                                                                                                                     | 1  |
| II.  | LEVEL OF ORDINARY SKILL IN the ART .....                                                                                                                             | 2  |
| III. | CLAIM CONSTRUCTION .....                                                                                                                                             | 3  |
| A.   | “First Module” and “Communication Manager” .....                                                                                                                     | 3  |
| B.   | “Communication Manager” and “Resource Manager” .....                                                                                                                 | 3  |
| C.   | The Preamble of Claim 10 .....                                                                                                                                       | 4  |
| D.   | “Independently Configurable” (claim 10).....                                                                                                                         | 4  |
| IV.  | CLAIMS 2, 6, AND 9-16 ARE UNPATENTABLE .....                                                                                                                         | 5  |
| A.   | My Opinions Are Not “Inconsistent” or “Unclear” .....                                                                                                                | 5  |
| B.   | GROUND II: Claims 2, 6, and 9-16 are Rendered Obvious by<br>Goossens2003 in View of Drake .....                                                                      | 7  |
|      | 1. A POSITA Would Have Been Motivated to Combine<br>Goossens2003 with Drake’s QoS Management System with a<br>Reasonable Expectation of Success .....                | 8  |
|      | a. Response to General Arguments .....                                                                                                                               | 8  |
|      | b. The Contentions Regarding Purported “Redundancy” and<br>“Teaching Away,” In Particular Regarding the “Communication<br>Manager” of Claim 2, Are Unsupported ..... | 28 |
|      | c. Dr. Bagherzadeh’s Assertions Regarding Claim 6 (And Claim<br>10) Are Unsupported .....                                                                            | 38 |
|      | 2. The Combination Teaches the Claim Limitations Regarding<br>“Communication Manager” and “Resource Manager” .....                                                   | 43 |
|      | a. “communication manager” (claims 2, 10[a]) .....                                                                                                                   | 43 |
|      | b. Elements 6[a] and 10[a] .....                                                                                                                                     | 46 |
|      | c. Elements 6[b] and 10[b] .....                                                                                                                                     | 48 |
|      | d. Elements 6[c] and 10[d] .....                                                                                                                                     | 51 |
|      | e. Elements 6[d] and 10[e] .....                                                                                                                                     | 55 |
| C.   | GROUND I: Claims 2, 6, and 9-16 are Rendered Obvious by<br>Goossens2002 in View of Drake .....                                                                       | 56 |

|                                                                                                                                                                      |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1. The Goossens2002 and Drake Combination Teaches the<br>“Independently Configurable [/Adjustable]” Claim Limitations...                                             | 56 |
| 2. A POSITA Would Have Been Motivated to Combine<br>Goossens2002 with Drake’s QoS Management System with a<br>Reasonable Expectation of Success .....                | 64 |
| a. Response to General Arguments .....                                                                                                                               | 64 |
| b. The Contentions Regarding Purported “Redundancy” and<br>“Teaching Away,” In Particular Regarding the “Communication<br>Manager” of Claim 2, Are Unsupported ..... | 65 |
| c. Dr. Bagherzadeh’s Assertions Regarding Claim 6 (and Claim<br>10) Are Unsupported .....                                                                            | 74 |
| 3. The Combination Teaches the Claim Limitations Regarding<br>“Communication Manager” and “Resource Manager” .....                                                   | 78 |
| a. “communication manager” (claims 2, 10[a]) .....                                                                                                                   | 78 |
| b. Elements 6[a] and 10[a] .....                                                                                                                                     | 80 |
| c. Elements 6[b] and 10[b] .....                                                                                                                                     | 82 |
| d. Elements 6[c] and 10[d] .....                                                                                                                                     | 85 |
| e. Elements 6[d] and 10[e] .....                                                                                                                                     | 90 |
| D. The Patent Owner’s Brief Mischaracterizes My Testimony .....                                                                                                      | 90 |
| V. ACKNOWLEDGMENT .....                                                                                                                                              | 94 |

I, Todd Mowry, declare as follows:

**I. BACKGROUND**

1. I submitted a prior declaration in this matter, which was dated December 18, 2023 and I understand was given the exhibit number “Exhibit 1003.” I refer to that declaration as my “First Declaration.”

2. I am a Professor in the Department of Computer Science at Carnegie Mellon University. I also have a courtesy appointment in the Department of Electrical and Computer Engineering. I have served on the faculty of Carnegie Mellon University for 27 years starting in 1997 through the present (2025).

3. Unless I state otherwise in this declaration, my background and qualifications are the same as I stated in paragraphs 1-15 and Exhibit 1004 to my First Declaration. My statements in paragraphs 16-18, and 29-58 regarding my review of U.S. Patent No. 7,373,449 (“the ’449 patent”) and related materials and the relevant technology remain unchanged, as do my understandings of the relevant legal principles stated in paragraphs 19-28.

4. Since my First Declaration, I have reviewed the Patent Owner’s Preliminary Response (“POPR”), the Board’s Institution Decision, the transcript of my November 4, 2024 deposition (which I understand was filed as Exhibit 2078), the Patent Owner’s Response dated November 19, 2024 (“POR”) and the materials cited in it, the Declaration of Dr. Nader Bagherzadeh dated November 19, 2024

(which I understand was filed as Exhibit 2070) and the materials cited in it, the deposition transcript of Dr. Bagherzadeh, and the materials I cite in this declaration.

## **II. LEVEL OF ORDINARY SKILL IN THE ART**

5. I understand that in the Institution Decision the Board adopted the level of skill I proposed in my First Declaration, and that Patent Owner and Dr. Bagherzadeh have agreed to apply that definition. Paper 35, 12; Exhibit 2070, ¶60. I understand that Dr. Bagherzadeh noted that in a different case (the “TI Litigation”), the district court adopted the following definition of a POSITA: “*a POSITA would have at least a Bachelor’s degree in electrical engineering, computer engineering, or a related field, and at least two years of work experience in the field of SoC and NoC technologies.*” Exhibit 2070, ¶57. I met the level of skill of a POSITA under my definition provided in my First Declaration, and also under the definition from the TI Litigation referenced by Dr. Bagherzadeh, both as of October 2003 (which I understand to be the relevant timeframe) and also today. I also was (and am) very familiar with others having the level of skill under each of those two definitions. I note that my opinions and analysis in both my First Declaration and this declaration would not change if the level of skill articulated in the TI Litigation were applied.

### **III. CLAIM CONSTRUCTION**

#### **A. “First Module” and “Communication Manager”**

6. I see that Dr. Bagherzadeh has offered opinions that as a matter of claim construction the terms “first module” and “communication manager” are separate and distinct. Exhibit 2070, ¶¶114-120.

7. I have not been asked to provide an opinion on this claim construction issue. I also note that it would not affect my opinions. As I stated in my First Declaration and also explain below, the claims are met by a combination of Goossens2002 with Drake’s QoS teachings, and also Goossens2003 with Drake’s QoS teachings. In each of the combinations, the QoS components (including the communication manager) are not positioned in the first module but rather where the Goossens references already teach to put QoS functionality: outside of the modules, with the NOC. As I explain further below, I also point to “first module” and “communication manager” components in the prior art that are separate and distinct.

#### **B. “Communication Manager” and “Resource Manager”**

8. I see that Dr. Bagherzadeh has offered the opinion that “communication manager” and “resource manager” would be understood by a POSITA as “separate and distinct.” Exhibit 2070, ¶¶121-125.

9. I have not been asked to provide an opinion on this issue. I also note

that it would not affect my opinions, as I point to separate and distinct components in the prior art I discuss that are a communication manager and a resource manager. I also note that I do not see that Dr. Bagherzadeh or the Patent Owner contend that any of the prior art I discuss fails to disclose a “communication manager” or “resource manager” because they are not separate and distinct from each other.

**C. The Preamble of Claim 10**

10. I see that Dr. Bagherzadeh has also offered the opinion that a POSITA would understand that the preamble of claim 10 is a limitation of the claim. Exhibit 2070, ¶¶126-129. While I have not been asked to provide an opinion on this issue, my opinions in both my First Declaration and this declaration assume that the preamble of claim 10 is a limitation of the claim.

**D. “Independently Configurable” (claim 10)**

11. I see that Dr. Bagherzadeh was asked to apply the definition of “independently configurable” from the Qualcomm Litigation, which was “able to be configured with connection properties that do not depend on those of another configuration.” Exhibit 2070, ¶¶140-141. I have not been asked to provide an opinion on the specific construction of this term or to apply the construction from the Qualcomm Litigation. However, my opinions would not change if the Qualcomm Litigation construction were applied in this proceeding.

#### IV. CLAIMS 2, 6, AND 9-16 ARE UNPATENTABLE<sup>1</sup>

##### A. My Opinions Are Not “Inconsistent” or “Unclear”

12. Dr. Bagherzadeh contends that I “appear to take inconsistent and/or unclear positions” as to “whether the claimed ‘first module’ purportedly corresponds to Drake’s source station 10 or only to application 11 therein.” Exhibit 2070, ¶¶147-149. That is not correct.

13. My First Declaration states that each of the *Goossens* references (Goossens2002, Goossens2003) provides the claimed “first module.” For example, see Exhibit 1003, ¶¶87, 128, 170, 192-193

14. I also note that Drake provides a module that similarly requests a communication service. For Drake, I explained that it was application 11 in source station 10 that was providing the request sent to Drake’s communication manager, like the claimed “first module” and the IP modules in the Goossens references. See for example, Exhibit 1003, ¶¶129-131 (specifically highlighting application 11), 177, 195. And there is nothing in the paragraphs of my First Declaration that Dr. Bagherzadeh cites (paragraphs 97, 134, or 198) that is inconsistent with this.

15. Drake does sometimes refer to the service request or response being

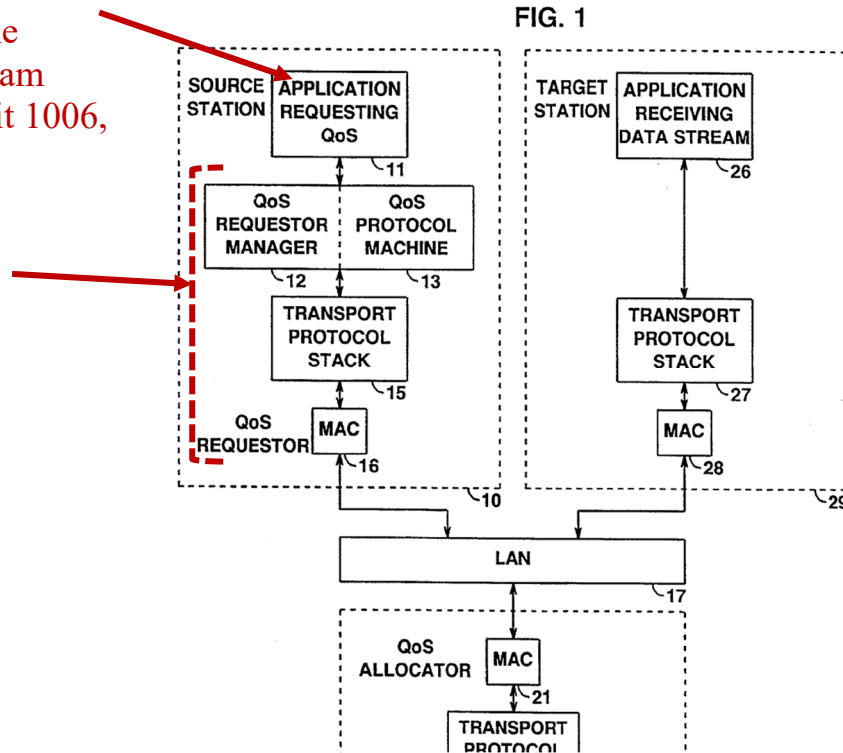
---

<sup>1</sup> My First Declaration also addressed claims 1 and 3-5, but I understand that after my First Declaration was filed the Patent Owner cancelled those claims.

sent from or to “source station 10.” This makes sense, because Drake shows that application 11 (which is the component making the original request and receiving the response) is in source station 10. For example, a POSITA would understand source station 10 could be something like a server. It is not wrong to say that in that context a service request comes from the server, but it actually originates from an application running on the server. A POSITA would understand the application running on the server to be a component (such as the main processor) that is separate from the interfacing “components” that are also part of the server. This is just like what Drake shows, where application 11 and the network interface “components” (such as QoS requestor manager 12) are shown and described as separate things. *See* Exhibit 1006, Fig. 1 and 4:26-36.

Application 11 “requests a QoS connection,” and also transmits the multimedia data stream (for example, Exhibit 1006, 4:27-32, 4:48-50)

Network interface “components” (for example, Exhibit 1006, 4:34-35)



**B. GROUND II: Claims 2, 6, and 9-16 are Rendered Obvious by Goossens2003 in View of Drake**

16. I understand that Dr. Bagherzadeh contends that claims 2, 6, and 9-16 are not obvious in view of Goossens2003 and Drake, because a POSITA would not have been motivated to combine their teachings to arrive at what is claimed (Exhibit 2070, ¶¶236-276), and also because even if they were combined, the combination would not meet the claim limitations regarding “communication manager” and “resource manager” (Exhibit 2070, ¶¶277-297). I do not agree, for at least the reasons I explain below.

**1. A POSITA Would Have Been Motivated to Combine Goossens2003 with Drake’s QoS Management System with a Reasonable Expectation of Success**

**a. Response to General Arguments<sup>2</sup>**

**(i) A POSITA Would Have Been Motivated to Look to Off-Chip Networking Technologies**

17. As I discussed in my First Declaration, “the parallels between off-chip networks and newer on-chip networks would have been well-known to POSITA.” Exhibit 1003, ¶¶163-166; *see also* Exhibit 1003, ¶¶30-34, 82 (Ground 1). Dr. Bagherzadeh similarly states that “NoCs share many of the same foundational principles as computer networks.” Exhibit 2070, ¶81. The ’449 patent likewise indicates that “many of the results in [off-chip network] fields are also applicable on chip.” Exhibit 1001, 1:61-66. Exhibit 2073, which Dr. Bagherzadeh quotes, similarly states that “[m]ost interconnect architectures used for NoC came from the parallel computing field.” Exhibit 2073, 7. A POSITA would understand “parallel computing field” in this quote to be reference to off-chip networks, because that was the common understanding and because the context, including the following sentences, indicated that the discussion is comparing on-chip and off-chip

---

<sup>2</sup> Dr. Bagherzadeh’s arguments regarding motivations (e.g., Exhibit 2070, pp. 23-34, pp. 71-104 (regarding Ground 1), pp. 118-148 (regarding Ground 2) are largely the same. I address both Grounds jointly in this section.

applications. *Id.*, 7. The known parallels between off-chip and on-chip networks are one reason why it would have been obvious for a POSITA to apply the teachings of Drake to Goossens2003 (and likewise to Goossens2002). Exhibit 1003, ¶¶79-81; *see also id.* ¶¶30-34 (Ground 1).

18. Dr. Bagherzadeh notes that there are differences between a NoC and a traditional off-chip network, such as a local area network (“LAN”) or a wide area network (“WAN”) and argues that because Drake is directed to a LAN, a POSITA would not have been motivated to use Drake to modify Goossens2002 or Goossens2003. Exhibit 2070, ¶¶242-245; *see also id.* ¶¶74-86, 168-171 (Ground 1). None of the differences between a traditional off-chip network and an on-chip network, however, contradict a POSITA’s general understanding at the time that teachings for off-chip networks were relevant to on-chip networks or the evidence showing that POSITA would understand that the specific teachings of Drake are applicable to Goossens2002 and Goossens2003.

19. One difference Dr. Bagherzadeh identifies is the “size of the components” and purported “cost” and “resource constraints,” arguing that because of this a POSITA “would not have implemented Drake’s QoS requestor manager somewhere on Goossens2003’s chip other than in the IPs and NOC.” Exhibit 2070, ¶263, *see also* ¶¶188 (Goossens2002). A POSITA would not have expected there to be meaningful size/cost/resource issues with the combinations, for the reasons I

explain below (for example, paragraphs 20 and 21). However, as I explained in my First Declaration, Goossens2003 itself also already teaches putting communication management functionality in the network interface of its NOC. Exhibit 1007, 76-77; Exhibit 1003, ¶¶185-186. So all a POSITA would have been doing is putting the communication manager right where Goossens2003 already teaches to put it on the chip—in the network interface of the NOC. Moreover, Goossens2002 notes a system-on-silicon (SOS/SOC) can contain “hundreds of IPs” (Exhibit 1005, 1), which is additional evidence that space in a NOC system would not have been an overriding concern for a POSITA. This is more evidence that purported size/cost/resource issues would not have dissuaded a POSITA from the combination.

20. Dr. Bagherzadeh argues that computer network components are “significantly larger” than NoC components and then makes a blanket assertion that because of this “[a] POSITA would understand that applying traditional networking architectures to a NoC is impossible.” Exhibit 2070, ¶75; *see also id.* ¶263, 101, 105. I disagree. Dr. Bagherzadeh’s assertion that traditional networking architectures are “impossible” in a NoC is unfounded. As discussed above and in my First Declaration (for example, Exhibit 1003, ¶¶30-34, 81-82, 165-166), a POSITA would understand that the teachings of off-chip networks are generally applicable to on-chip networks, because there are many parallels between the two.

Even if off-chip networks may have larger components, that would not mean that the general teachings of off-chip networks are not applicable, and Dr. Bagherzadeh provides no specific reason why a difference in size would render all teachings “impossible.”

21. In fact, Dr. Bagherzadeh and I agree that essentially everything integrated onto a chip is just a smaller version of the off-chip component. For example, Dr. Bagherzadeh acknowledges that a “system-on-chip,” which is the focus of Goossens2002, Goossens2003, and the ’449 patent, is a larger system implemented in smaller form onto a chip. Exhibit 2070, ¶62 (“A system-on-chip (‘SoC’) is an integrated circuit in which *the components of a system are integrated on the chip*, including memories, processors, interfaces, and logic.”) (emphasis added). I similarly stated in my First Declaration: “[a]s the capacity of integrated circuit (IC) technology has improved exponentially over the years, it has enabled *larger portions of computer systems* to be integrated onto single chips.” Exhibit 1003, ¶30 (emphasis added); *see also id.*, ¶31 (citing references). This applies equally to implementing components from networks, such as the components in Drake. For example, the ’449 patent, Dr. Bagherzadeh, Goossens2002, and Goossens2003 all acknowledge that it was known that components from networks could be integrated onto a chip to form “NoCs.” Exhibit 1001, 1:36-38; Exhibit 1003, ¶32; Exhibit 1037, 102:8-10; Exhibit 1005 (Goossens2002), Abstract;

Exhibit 1007 (Goossens2003), 61. Thus, any difference in size between off-chip components and on-chip components would not affect the motivation of a POSITA for these combinations.

22. Specific to the combinations of Drake with Goossens2002 or Goossens2003, Dr. Bagherzadeh relies on this “size” difference to suggest the proposed combination is faulty because “implementing a communication component outside of the IPs and NOS would require space and resources,” such as a separate circuit with its own network interface. Exhibit 2070, ¶¶263, 188. A POSITA knew that implementing a component anywhere on a chip (or off-chip) requires space and resources, and anything that communicates with a network typically required a network interface. These general facts do not mean that a POSITA would not be motivated to implement a component in a NOC, as Dr. Bagherzadeh appears to be suggesting. Moreover, as I noted above, Goossens2002 specifically explains that a system-on-silicon (SOS/SOC) can contain “hundreds of IPs” (Exhibit 1005, 1), indicating that space and resources (particularly for a chip containing only a small number of IPs) would not have been a dissuading factor.

23. Dr. Bagherzadeh also asserts that because of the size difference and corresponding “resource...constraints” a POSITA would not be motivated to replace the “single” quality-of-service manager in Goossens2003 (or Goossens2002) with “two” QoS components from Drake (QoS requestor manager

and QoS allocator). Exhibit 2070, ¶240; *see also id.* ¶¶165-166 (Ground 1). I disagree. I did not propose replacing a “single” QoS component with “two” QoS components. As I explained in my First Declaration, Goossens2002 and Goossens2003 did not provide detail on the implementation of the guaranteed services and connections, and Drake provides a more detailed example that would have been an obvious way to implement them. Exhibit 1003, ¶159; *see also id.* ¶75 (Ground 1). A POSITA would understand that accounting for the size difference and corresponding resource constraints would have been a routine part of the design process. This is particularly true given that Goossens2002 and Goossens2003 already contemplate using space on the chip (and *outside the IP modules*) for QoS functionality. For example, *see* Exhibit 1005, 3 (Goossens2002) (“network interfaces” that “implement multi-path and multi-channel ordering, and channel follow control” and a “quality-of-service manager” that can “observe and steer” the NOS); Exhibit 1007 (Goossens2003), 65, 67-69, 74 (describing “resource management”) and *id.*, 76-77 (describing “network interfaces” that allow “end-to-end” connection properties).

24. Another difference identified by Dr. Bagherzadeh is “the cost ratio between wiring and processing resources.” Exhibit 2070, ¶76. Dr. Bagherzadeh does not, however, argue that this difference affects any motivation to apply the teachings of Drake to Goossens2002 or Goossens2003. I also do not believe such a

difference would change the motivation of a POSITA in this regard.

25. Another difference identified by Dr. Bagherzadeh is that traditional off-chip networks had network interfaces embedded within the computers, whereas in a SoC, including network interfaces in an IP module was disfavored. Exhibit 2070, ¶¶77. Dr. Bagherzadeh later relies on a similar argument to suggest that Drake's teaching of QoS requestor manager 12 in source station 10 would teach a POSITA to include such a requestor in the IP module of Goossens2002 / Goossens2003. Exhibit 2070, ¶¶255-262; *see also id.* 184-187 (Ground 1). I strongly disagree. A POSITA would have been familiar with this "difference," but that would have motivated a POSITA to place the communication manager (such as Drake's QoS requestor manager 12) *outside* the IP module in the network-on-chip context, not inside it. In fact, this known "difference" between off-chip networks and on-chip networks is the reason why Drake showing the QoS requestor manager 12 located in source station 10 does not teach to include the QoS requestor manager 12 inside the IP module. I explain this in the following paragraphs.

26. As Dr. Bagherzadeh appears to agree that the general understanding in the art was that the network interfaces in an on-chip network context paralleled the network interfaces of a traditional off-chip network. Exhibit 2070, ¶¶77. That is, the network interfaces of a NoC were known to perform the same general function

as network interfaces of a traditional off-chip network. Both were understood to “decouple computation from communication,” by enabling their host/IP module to communicate without requiring specific knowledge of the low-level network details. Network interfaces in a traditional off-chip network functioned to interface a host processing module to a broader network. Network interfaces in the network-on-chip context were similarly understood to perform the same functionality for an IP module.

27. A POSITA would have understood that a “difference” between off-chip and on-chip network interfaces was that network interfaces in a traditional off-chip network were typically part of the host (e.g., a server), whereas in a NoC they were separate from the IP module (as the Goossens references show). Exhibit 2070, ¶77. A POSITA would also have understood that this “difference” is merely a box drawing exercise. The network interface in both off-chip and on-chip networks is functionally disposed in the *same location* – between the communicating component (e.g., application 11 in Drake; IP module in Goossens2002/Goossens2003) and the rest of the network. Each of Drake, Goossens2002, and Goossens2003 references the use of network interfaces to facilitate communication between users of the network and the network. Exhibit 1005 at 3 (Goossens2002) (“A channel implements a simplex point-to-point communication between two network interfaces . . . Network interfaces implement

multi-path and multi-channel ordering, and channel flow control.”); Exhibit 1007 at §4.2 (network contains “network interfaces”), §4.2.4, Fig. 4.5 (Goossens2003); Exhibit 1006 at 1:61-65 (Drake) (“These techniques typically involve the provision of a sophisticated user network interface through which the user requests a connection for multimedia transport and specifies the quality of service that is required.”). A POSITA would have understood that for network interfacing components in an off-chip network (often located within the host), the corresponding location in an on-chip network is in the network interface (which is *outside* of the IP module, just as Goossens2003 itself teaches). Exhibit 1007, ¶4.2.4, Figure 4.5.

28. A POSITA would have understood Drake to teach that the QoS requestor manager 12 is part of the network interface components for source station 10. This would have been understood because (A) network interfaces were commonly included as part of a host in an off-chip LAN system (*see* Exhibit 2070, ¶77); (B) Drake specifically teaches that the functionality of its QoS request manager 12 includes “assembl[ing]” and “format[ing]” the QoS request to enable communication with networking components like the QoS allocator 20 (Exhibit 1006, 4:32-37, 4:62-65), which a POSITA would recognize was interfacing; (C) the surrounding components 13-16 of source station 10 are also network interface-type components involved in enabling communication with network components in the

LAN (Exhibit 1006, 4:32-47); and (D) the location of the components 12 through 16—between the application 11 and LAN 17—is the location of a network interface (Exhibit 1006, FIG. 1). Thus, a POSITA would understand Drake to teach the QoS requestor manager 12 in the network interface.

29. Furthermore, (and as acknowledged by Dr. Bagherzadeh) Goossens2002 and Goossens2003 teach that it is desirable not to change the IP modules. Exhibit 1005, 1-2; Exhibit 1007, 65; Exhibit 2070, ¶¶255, 180. I explained in my First Declaration that Goossens2003 also discloses resource management functionality in its NoC (outside of the IP module), just not the details of a component to carry out that functionality. Exhibit 1003, ¶¶159, 173. I also explained that Goossens2003 similarly discloses communication management that is “implemented by” Goossens2003’s network interfaces.<sup>3</sup> Exhibit 1003, ¶¶185-186; Exhibit 1007, 76-77. Thus, Drake and the Goossens references, as well as the general understanding in the art, would all teach a POSITA that a natural place to put components corresponding to the network interface (such as Drake’s QoS

---

<sup>3</sup> I also explained that under Patent Owner’s district court definition of communication manager – which is just a “circuit” that “manages communication between modules,” Goossens2003 would also disclose a communication *manager*. Exhibit 1003, ¶¶185-186.

requestor 12), would have been in the network interfaces of the Goossens references. And in the network-on-chip context of Goossens2003, the interfaces are with the NOC, outside of the IP modules (and the same is true for Goossens2002).

30. Dr. Bagherzadeh also contends that network interface functionality is “typically software based.” Exhibit 2070, ¶79. Dr. Bagherzadeh relies on this to suggest that “a POSITA would have understood that Drake’s QoS requestor manager 12 is ... inseparably connected and coupled to the source station, such that QoS requestor manager 12 could not be outside of the source station.” *Id.*, ¶¶255, 180. I disagree. There is nothing in Drake indicating that those components are somehow inseparably connected and coupled to the source station 10.

31. Dr. Bagherzadeh appears to suggest that everything inside of source station 10 is software running on a common processing module. That is not what a POSITA would understand Drake to disclose. A POSITA would understand Drake to be disclosing and describing application 11 and QoS requestor manager 12 as separate components. See, for example, Exhibit 1006 (Drake) at FIG. 1 (showing application 11 and requestor manager 12 as separate components with arrows between them showing the data path); 4:34 (referring to “components 12 and 13”). A POSITA would understand from this description that QoS requestor manager 12 is already separate from application 11 in Drake, and there is no reason it could not be implemented apart from source station 10. A POSITA would have further

understood that it was common design practice to have the network interface components (e.g., QoS requestor manager 12) as separate components from the main processing module (for example, application 11) as I explained above. *Supra*, ¶¶15, 27-28. This is exactly what Drake shows.

32. Dr. Bagherzadeh also identifies differences in “reordering,” “time-related guarantees,” and “reliable communication” as purported differences between off-chip and on-chip networks, but again he does explain how any of these would affect any motivation to apply the teachings of Drake to Goossens2002 or Goossens2003. See his discussion at Exhibit 2070, ¶¶102-106. I also do not believe any of these would change the motivation of a POSITA in this regard.

**(ii) A POSITA Would Have Had A Reasonable Expectation of Success**

33. Dr. Bagherzadeh contends that a POSITA would not have reasonably expected success in making the combination. Exhibit 2070, ¶¶247, 264, 189 (citing ¶¶201-210). I disagree.

34. First, Dr. Bagherzadeh asserts that a POSITA as I’ve defined it, would not have had a reasonable expectation of success because they would not have had “specific knowledge regarding the design of a SoC.” Exhibit 2070, ¶¶ 201-210, 247. I note that Dr. Bagherzadeh does not propose a different definition of a POSITA. So, it does not appear that there is any dispute that my definition of a POSITA is accurate as to “the education level of active workers in the field,” “the

types of problems encountered in the art,” or “prior art solutions to these problems.” Exhibit 2070, ¶56.

35. In any case, a POSITA would have had sufficient experience to implement the teachings of Drake in Goossens2002 and Goossens2003. Goossens2002 and Goossens2003 each teach a POSITA about the IP modules, connections, network interfaces, and resource management, amongst other things. Drake teaches POSITA about the interaction between connections and QoS components, amongst other things. These teachings, combined with the general experience of a POSITA, would have been more than sufficient to allow a POSITA to reasonably expect success in making the combination.

36. Dr. Bagherzadeh argues that a POSITA “would have to have a thorough understanding of the constraints that exist when a SoC with hundreds of modules is being implemented on a two-dimensional grid.” Exhibit 2070, ¶205 The claims, however, do not require an SoC with “hundreds of modules” implemented “on a two-dimensional grid.” A POSITA would only need to understand that the combination could implement “a plurality of modules” on “an integrated circuit” along with the other claimed elements. See, for example, Exhibit 1001, claim 10. A POSITA would have known that components such as modules and QoS components could be integrated onto an integrated circuit (*see supra*, ¶¶20-21), and thus a POSITA would have had the required understanding to expect success in

such a combination.

37. Dr. Bagherzadeh further argues that a POSITA would have to understand how to “integrate their functionalities on a miniscule piece of hardware and arrange them within a network or within the network interfaces” of the SoC of Goossens2002 and Goossens2003. Exhibit 2070, ¶¶208. Dr. Bagherzadeh’s suggestion that a POSITA must possess sufficient personal knowledge to perform every step of fabricating an integrated circuit is inaccurate. A POSITA would have understood that integrating components onto an IC was routine and would have known that there would have been design automation tools and technical personnel available that could perform such integration.

38. And as far as arrangement, Drake illustrates the data path and arrangement of components. Applying that to the Goossens references would have been routine and straightforward. As I explained above and in my First Declaration, a POSITA at the time knew that the off-chip resource allocation solutions were applicable to on-chip networks. See, for examples, Exhibit 1003, ¶¶33 (quoting Exhibits 1015, 1010, and 1014), 163-166 (quoting Exhibit 1014 at 33 and 102). Implementing Drake’s QoS components and data path in Goossens2003 or Goossens2002’s NOC architecture would have been routine and straightforward. For example, as I explained at my deposition a natural and obvious place to implement the communication manager is in Goossens’ network

interface (which is where Goossens itself teaches to put communication management functionality, see Exhibit 1003 at ¶¶185-186) (Exhibit 2078, 204:7-8), and the resource manager could go anywhere on the chip that is connected to the network (Exhibit 2078, 212:20-213:3), just as Drake shows and which is consistent with Goossens2003 and Goossens2002's teaching of providing quality of service management separate from the IP modules.

39. Dr. Bagherzadeh also argues that a POSITA would not have been capable of implementing Drake's QoS requestor manager on the chip of Goossens2002 or Goossens2003, particularly outside of both the IPs and NOS. Exhibit 2070, ¶¶264, 189, 208. As I already explained, in the proposed combination, the obvious thing to do is to implement Drake's QoS with the Goossens' references NOC/NOS. Drake already teaches the components and the data paths. Implementing that in Goossens2002 or Goossens2003 would have been routine design work for a POSITA. For example, Goossens2002 and Goossens2003 already teach implementing QoS functionality outside of the IP modules on the chip, and specifically teach communication management functionality in the network interfaces of the NOC/NOS. See, for example, Exhibit 1005 at 3 (Goossens2002) ("network interfaces" that "implement multi-path and multi-channel ordering, and channel follow control" and a "quality-of-service manager" that can "observe and steer" the NOS); Exhibit 1007 (Goossens2003) at 65, 67-69,

74 (describing “resource management”), 76-77 (describing “network interfaces” that allow “end-to-end” connection properties); *see also* Exhibit 1003, ¶¶159, 173, 185-186 (Goossens2003), ¶¶90, 136 (Goossens2002).

**(iii) The Only Two New Documents Dr. Bagherzadeh Cites (Exhibits 2072, 2073) Do Not Support His Contention that “Applying traditional networking architectures to a NoC is impossible”**

40. Dr. Bagherzadeh cites two documents that he contends support his assertion that a POSITA would have understood that “applying traditional networking architectures to a NoC is impossible.” Exhibit 2070, ¶¶75-86. Neither one supports his assertions.

41. The first one, Exhibit 2072, does not show or suggest that traditional off-chip network architectures are “impossible” on-chip. Exhibit 2070, ¶75. Exhibit 2072 just states that one design constraint that is present in a NoC, and less applicable in the off-chip network context, is area usage. The fact that NoCs can have different design constraints, such as potentially stricter area usage, does not mean that off-chip network architectures are “impossible” on-chip. Rather, as taught by the ’449 patent itself, such design constraints merely result in “reevaluat[ion]” of network design choices. Exhibit 1001, 1:66-2:1. For example, the size of a component could have been predictably reduced by shrinking buffer sizes, or by implementing a fixed-function custom hardware circuit rather than

using a general-purpose CPU. See, for example, Exhibit 1008, 9 (“A special-purpose PE may be much faster and smaller than a programmable processor.... Even if a programmable processor is used for a task, specialized CPUs can often improve performance while saving area. For example, matching the CPU datapath width to the native data sizes of the application can save a considerable amount of area.”).

42. The second one, Exhibit 2073, appears to be an unpublished, unsigned, and unreviewed thesis paper from a student. Dr. Bagherzadeh cites this unsigned thesis as support for his assertion that there are certain features of off-chip networks that are “impossible” on-chip. Exhibit 2070, ¶81. That is not what Exhibit 2073 states. A POSITA would understand Exhibit 2073 to be stating that an “ideal architecture” that provides the best possible outcome for each individual feature is impossible, because some features contradict each other. Exhibit 2073, 7. That is, there are trade-offs when using a NoC that cause compromises. That is unremarkable (and was understood by a POSITA). However, Exhibit 2073 does not suggest that anything relevant to the combination of Drake and Goossens2002/Goossens2003 is “impossible” (or even difficult), and Dr. Bagherzadeh does not identify anything. And Exhibit 2073 certainly does not show that any particular off-chip concept is “impossible” on-chip.

**(iv) I Explained the Combination**

43. Dr. Bagherzadeh asserts throughout his declaration that I failed to explain “why” or “how” certain aspects of the combination would work. For example, see Exhibit 2070, ¶¶236, 238-241, 245-246, 248-253, 266-272, 274, 276, 279. I disagree. In my First Declaration I explained “why” and “how” the Goossens2003/Drake and Goossens2002/Drake combinations would work. Exhibit 1003, §§IX.A, IX.B.

44. I explained, for example, that Goossens2002 and Goossens2003 do not include certain details regarding their implementation of guaranteed services and connections, and that Drake provides details regarding the implementation of these features. *See* Exhibit 1003, ¶¶ 75-82, 157-166. I further explain that a POSITA would have understood that using Drake’s quality of service management process would improve the efficiency and performance of Goossens2002 and Goossens2003 networks on chip by ensuring that the network resources required to provide guaranteed services are actually available and that the network will be able to fulfill its commitments to the IPs that have been granted such services. *Id.*, ¶¶79, 163. This combination would be straightforward in that it would retain the relevant components from Goossens2002 and Goossens2003 while adding details from Drake’s quality of service management process. None of the components from Goossens2002 or Goossens2003 that meet the claim elements of the ’449 patent

would be removed in this combination.

45. Dr. Bagherzadeh specifically suggests that I do not explain why a POSITA would be motivated to use the QoS allocator from Drake in the combinations in a way that meets the “resource manager” claim limitation or how that QoS allocator would work in the proposed combination. Exhibit 2070, ¶¶ 173-74, 248-249. That is not true. In the combination described above and in my First Declaration, a POSITA would be motivated to use Drake’s quality of service management process, which includes the QoS allocator, for the reasons described above. Moreover, the QoS allocator would work in the same way as it is described in Drake (e.g. *receive the QoS request from the requesting entity/application (forwarded by the QoS requestor manager 12), examine the resources of the network to determine whether it can be satisfied, and send a reply back through the QoS requestor manager 12 which indicates whether the requested QoS is available*). As described in my opening declaration, this would meet the resource manager claim limitation. For example, regarding Goossens2003, a POSA would have “looked to Drake’s disclosures regarding a QoS allocator for details regarding the implementation of resource management in Goossens2003’s NOC” and would have been motivated to implement the functionality of Drake’s QoS allocator “in a circuit on the same integrated circuit chip as Goossens2003’s IP blocks and NOC”. Ex-1003, ¶173, *see also id.* ¶¶162-163. And with respect to Goossens2002, I

explained that an obvious thing to do would have been to implement the functionality of Drake's QoS allocator "in Goossens2002's quality-of-service manager." Ex-1003, ¶¶96, *see also id.* ¶¶94. I provide additional detail throughout my declaration, but a POSITA would understand this to address the "why" (to provide additional details to aid in creating a SoC with "improved performance and efficiency") and "how" (in the manner described by Drake, but on a chip with the Goossens references' NOC).

46. Similarly, with respect to the communication manager (Drake's QoS requestor manager 12), I explained that Goossens2003 already discloses communication management functionality in its network interfaces. Exhibit 1003, ¶¶185-186. I also explained that Drake's QoS requestor manager 12, applied in the same way in Goossens2003, would provide the "benefits of Drake's known solution to ensuring the availability of network resources for guaranteed services," and that it receives a request for a QoS connection just like requests issued from IP modules in Goossens2003. Exhibit 1003, ¶¶187, 194-195, *see also id.* ¶¶165 (POSITA would expect using "Drake's known solution...would improve Goossen20034's NOC in the same way"), ¶¶162-164. And with respect to Goossens2002 (Ground 1), I explained that a POSA would be motivated to "implement[] the functionality of Drake's QoS requestor manager as a circuit on the same integrated circuit chip as Goossens2002's IP blocks and NoS". Exhibit

1003, ¶122, see also *id.* ¶¶78-79, 118-122. Again, I provide additional explanation both in my First Declaration and in this declaration, but a POSITA would understand this to address the why (to obtain the benefits of Drake’s QoS solution) and how (in the manner described by Drake, but on a chip with the Goossens references’ NOC).

**b. The Contentions Regarding Purported “Redundancy” and “Teaching Away,” In Particular Regarding the “Communication Manager” of Claim 2, Are Unsupported**

47. Dr. Bagherzadeh argues that I was “unclear” whether I was mapping the “communication manager” to Goossens2003’s NI or Drake’s QoS requestor manager 12, and that if it was Goossens2003’s NI then there was not motivation. Exhibit 2070, ¶¶250-251, 268-269. That is not correct.

48. As I explained in my declaration, *Drake* discloses a communication manager, as QoS requestor manager 12. Exhibit 1003, ¶187. I also explained that Goossens2003 discloses *communication management functionality* that was taught as located in the network interface—“network interfaces that manage the communication between different modules.” Exhibit 1003, ¶¶185-186.<sup>4</sup> To a

---

<sup>4</sup> I also stated that under Patent Owner’s district court case definition, Goossens2003 itself would also disclose a communication manager. Exhibit 1003, ¶186.

POSITA, this is yet more evidence of motivation to make exactly the combination I explained: to use the teachings of Drake regarding QoS management (including its QoS requestor manager 12 component, the communication manager) to fill in the details of the communication management that Goossens2003 already taught. And Goossens2003 already taught to implement communication management outside of the IP module, such as in the network interface of its NOC. *E.g.* Exhibit 1003, ¶¶185-186 (citing Exhibit 1007 (Goossens2003) at 76-77).

49. Dr. Bagherzadeh also argues that a POSITA would not have been motivated to use the QoS requestor manager 12 component from Drake in the proposed combination. Exhibit 2070, ¶¶ 252-265. Based on my review of his declaration, it appears he is making two basic arguments. He first argues that QoS requestor manager 12 would be redundant in Goossens2002. He then argues that Drake and/or Goossens2003 teaches away from the combination that includes QoS requestor manager 12. I do not agree with either of his arguments.

**No redundancy:**

50. Dr. Bagherzadeh first argues that I did not show that Drake's QoS requestor manager 12 was "needed" in Goossens2003 and he argues that its function is "redundant of Goossens2003's IP block." Exhibit 2070, ¶¶252-254, *see also, id.*, ¶237. As I explained in my First Declaration, Goossens2003 teaches a NOC and the importance of managing network resources to provide guaranteed

services, but it does not include certain details regarding implementation of its guaranteed services and connections. Exhibit 1003, ¶159. Drake, on the other hand, “provides details regarding the implementation of these features,” and “a POSITA would have found Drake’s known solution for ‘ensur[ing] selected levels of quality of services’ helpful in implementing an SoC like the one described by Goossens2003.” Exhibit 1003, ¶160. Therefore, as I explained in my First Declaration, a POSITA would have expected that using Drake’s “quality of service management process would improve the efficiency and performance of Goossens2003’s network on chip.” Exhibit 1003, ¶163. So, in the combination I already explained a POSITA would have been motivated to use, there is a communication manager. QoS requestor manager 12 was an important aspect to Drake’s quality of service management process, and a POSITA would not have simply removed it in the combination. *See* Exhibit 1003, ¶¶ 157-163.

51. In addition, in the combination, the “QoS requestor manager” would not be redundant of the functioning of Goossens2003’s IP blocks. As I discussed above, network interfaces “decouple computation from communication” by enabling their host/IP module to communicate without requiring specific knowledge of the low-level network details. A POSITA would understand that components such as the QoS requestor manager 12 would desirably be included in the NOC of Goossens2003.

52. A POSITA would understand that Goossens2003's NOC would desirably include a component to take the signal from the "re-use[d]" IP block and communicate with the network. Exhibit 1007, Abstract. Different re-used IPs would typically use different protocols to describe the type of service it wanted to send its data over. Exhibit 1010 at 1. Therefore, Goossens2003's NOC would have been expected to use a component such as Drake's QoS requestor manager 12 to assemble/format a request that could be understood by the network. That is exactly what Drake teaches the QoS requestor manager 12 is for. Exhibit 1006, 4:33-35 ("components 12 and 13 assemble a request"), 4:62-65 ("QoS requestor manager 12 interacts with application 11...to format the QoS request"). A POSITA would not expect such an interfacing component would already be included in the "re-used" IP block, which does the computation and sends messages without regard to the communication system it is using. The fact that this specific component was not disclosed in Goossens2003 is the result of Goossens2003's disclosure not including the details of the QoS components. A POSITA would not expect that Goossens2003's network would be able to perform its QoS functionality without a component to assemble the QoS request from the different re-used IP protocols. Therefore, to a POSITA the predictable, beneficial combination of Goossens2003's NOC with Drake's QoS teachings would be to simply use Drake's QoS components communicating with each other as Drake taught (including the QoS

requestor manager 12 that “assemble[s]”/“format[s]” the request).

**No teaching away:**

53. Dr. Bagherzadeh’s teaching away argument (Exhibit 2070, ¶¶ 255-265; *see also, id.*, ¶237) applies an overly rigid and unrealistic view of the prior art that ignores a POSITA’s understanding of both off-chip and on-chip networks. Goossens2003 does not “teach away,” and the proposed combination would not be a “modification of Drake’s teachings” as Dr. Bagherzadeh contends. For example, as I noted above and in my First Declaration, Goossens2003 already teaches putting communication management functionality outside the IP modules, such as in the “network interface” of Goossens2003. Exhibit 1007, ¶¶76-77; Exhibit 1003, ¶¶185-187. Also, as I discussed above, a POSITA would understand that the QoS requestor manager 12 was part of Drake’s QoS network interface components, and *in the network-on-chip context of Goossens2003*, a POSITA would simply put components corresponding QoS network interfacing (e.g., QoS requestor 12) where Goossens2003 taught to put QoS network interfacing components in its network-on-chip—outside its IP module, such as in the network interface of its NOC. *See supra*, ¶¶25-32. That Drake shows QoS network interfacing components (such as QoS requestor manager 12) in the source station (e.g., in a server) on the off-chip environment would not have dissuaded a POSITA from implementing such components in the on-chip environment in the place where Goossens2003

taught they would go: outside the IP modules, such as in the “network interface” of the NOC. Exhibit 1007 at 76-77; Exhibit 1003, ¶¶185-187.

54. Dr. Bagherzadeh attempts to suggest it is relevant that in one part of the specification Drake refers to source station 10 as “requestor station 10.” Exhibit 2010, ¶255. I do not understand why this is relevant or surprising. Source station 10 contains the components that are making the request for QoS, which is all a POSITA would understand from Drake’s reference to it as requestor station 10. This nomenclature would not suggest to a POSITA that all “QoS Requestor Managers” must be in that station, much less that it must be in an IP module in the combination with one of the Goossens reference. And Drake also discloses a QoS requestor manager (QoS Requestor Manager 23) outside of “requestor station 10.” See Exhibit 1006, Fig. 1.

55. Similarly, Dr. Bagherzadeh attempts to suggest it is relevant that QoS requestor manager 12 is not illustrated as being within “prior art [LAN] 17.” Exhibit 2070, ¶255. The fact that a block in Drake is described as “prior art [LAN] 17” does not suggest that the QoS components cannot be implemented outside IP modules in the on-chip context of Goossens2003. All that nomenclature would have suggested to the POSITA is that the portion of the system that is denoted by the box is a local area network and is non-novel; it does not suggest that the other components are not connected to the LAN or must be located in a “source station”

exactly as Drake shows. In any event, this would not dissuade a POSITA from implementing Drake's QoS requestor manager 12 in NOC of Goossens2003, such as in Goossens2003's network interface (just like it is part of the network interface components in Drake). Dr. Bagherzadeh cites Drake at 2:57-3:3 as purportedly teaching that QoS management should not be implemented in the LAN because doing so would be "complex," and contends this is evidence of teaching away. Exhibit 2070, ¶206. This discussion in Drake does not teach away, and in fact is entirely consistent with the combination a POSITA would be motivated to make. Drake references that in its system most of QoS management is "concentrated in a single QoS management station on the LAN" and that this "reduces complexity of introducing QoS management into existing Lan networks." Exhibit 2006, 2:66-3:3. A POSITA would understand this as a reference to Drake's *QoS allocator 20 station* (the resource manager). In the combination a POSITA would understand the resource manager (Drake's QoS allocator 20) could be implemented as single piece of centralized hardware on the NOC in Goossens2003 (just like in Drake QoS allocator 20 is shown as a single piece of centralized hardware "on the LAN" – meaning connected to the LAN). Exhibit 1003, 173 (a POSITA "would have been motivated to implement the functionality of Drake's QoS allocator in a circuit on the same integrated circuit chip as Goossens2003's IP blocks and NOC"); Exhibit 2078, 212:20-213:3. And nothing about this teaching (or any other teaching in

Drake) criticizes or discredits locating the separate component of the *QoS requestor manager 12* (the communication manager) outside of the IP module in the network interface network-on-chip context such as that disclosed by the Goossens references.

56. Dr. Bagherzadeh further suggests that the QoS requestor manager is software running on source station 10 and suggests that this teaches away from including it in Goossen's NoC. Exhibit 2070, ¶255. As I explain above, I do not agree that a POSITA would understand QoS requestor manager 12 to be software. Instead, a POSITA would understand Drake to be disclosing and describing application 11 and QoS requestor manager 12 as separate "components." *See supra*, ¶¶30-32. Therefore, there would be no reason this component would not be located outside of the IP block in Goossens2003.

57. Additionally, whether or not QoS requestor manager 12 were software, it would have been well within the skill of a POSITA to implement such a communication manager in Goossen2003's network interface as hardware or software. Therefore, whether Drake's requestor manager 12 is software or hardware would not affect a POSA's motivation to incorporate such a communication manager in Goossens2003's NOC.

58. Dr. Bagherzadeh suggests he reads my First Declaration to suggest that a POSITA would modify the IP block of Goossens2003. Exhibit 2070, ¶257,

259. I do not understand how he reached this conclusion. As the Board recognized in its institution decision, my opinion was premised on implementing Drake's QoS management process outside of the Goossens2003 IP block. *See supra*, for example, ¶¶25, 28, 38-39, 52, 54; *see also* Exhibit 1003, ¶¶162-163, 173, 185-186 (Ground 1: *id.* ¶¶78-79, 94, 96).

59. Dr. Bagherzadeh also asserts that my First Declaration is "ambiguous and unclear as to specifically how the relied-upon components and functions of Drake's QoS system, particularly the QoS requestor manager, would purportedly work with Goossens2003's IP blocks and NoC." Exhibit 2070, ¶258. Dr. Bagherzadeh makes the same assertion with respect to Goossens2002. *Id.*, ¶183. I disagree with both. Among other things, my First Declaration stated:

a POSITA would have been particularly interested in Drake's disclosure of a "QoS requestor manager" to formulate requests for guaranteed services with particular properties, and a "QoS allocator" to determine whether the network resources required for the requested services are available.

Exhibit 1003, ¶76. I refer back to these statements in Ground 2. *Id.*, ¶187. This is a description of how the components would work with Goossens2003's IP blocks and NoC.

60. Dr. Bagherzadeh also asserts that my First Declaration did not assert that the QoS requestor manager would be implemented "as a circuit on the same integrated circuit chip as Goossens2003's IP blocks and NOS," like I stated with

respect to Goossens2002 in Ground 1. Exhibit 2070, ¶260. I disagree. The discussion of the QoS requestor manager in Ground 2 refers back to the statements made in Ground 1 with respect to Goossens2002. Exhibit 1003, ¶187. Dr. Bagherzadeh also states that this assertion fails for the same reasons as it did for Ground 1, citing his entire motivation section from Ground 1. As I explained above, his arguments are largely the same for Ground 1 as Ground 2, and I addressed those above. *See* Section VI.B.1.a. I addressed specific additional arguments related to Ground 1 below, and incorporate that discussion here to the extent Patent Owner suggests the discussion above does not address Dr. Bagherzadeh's point.

61. Dr. Bagherzadeh also asserts that Drake and Goossens2003 “teach completely opposite solutions” because QoS requestor manager 12 is in the source station 10 of Drake. Exhibit 2070, ¶262. As discussed above in paragraphs 25-29, 38-39, 52, and 54 above, Drake's inclusion of the QoS requestor manager 12 in the source station 10 teaches a POSITA that a natural, obvious place to put it would be in the network interface of Goossens2003. POSITAs did not typically view a “network,” “NoC” or “LAN” as having rigid boundaries like suggested by Dr. Bagherzadeh. To the contrary, POSITAs looked at components and how they interact, which a POSITA understood was *the same* for both on-chip and off-chip network. For an off-chip network: the components included an application, which

interacted with a network interface, which interacted with the transport components to transport a message. And for an on-chip network: the components also included an IP module, which interacted with a network interface, which interacted with transport components to transport a message. Because a POSITA understood it is the interaction of the components that matters, the fact that Drake labeled a particular part of its system “LAN” or the fact that Goossens2003 described a particular part of its system as “network” would not teach a POSITA that those are the only components that would be used. Thus, Drake does not teach either (a) putting the QoS requestor manager 12 in an IP module or (b) keeping the QoS requestor manager 12 out of the NoC as suggested by Dr. Bagherzadeh.

62. Drake’s reference to a “prior art” LAN also does not suggest that the network must “stay[] the same” and does not teach away from including QoS components with the Goossens2003’s NOC, as asserted by Dr. Bagherzadeh. *See* Exhibit 2070, ¶262. To the contrary, as I explained above, in the network-on-chip context a POSITA would have been motivated to implement QoS components with the NOC, outside of the IP modules, exactly as Goossens2003 teaches to do.

**c. Dr. Bagherzadeh’s Assertions Regarding Claim 6 (And Claim 10) Are Unsupported**

63. I disagree with Dr. Bagherzadeh’s assertion that I did not explain motivation for what is recited by claim 6 (and 10). *See* Exhibit 2070, ¶¶266-276.

64. Dr. Bagherzadeh appears to misunderstand the combination. In my

First Declaration, I explained why a POSITA would have been motivated to combine Goossens2003, which discloses a system-on-chip that provides guaranteed services using “connections,” with Drake’s description of components and mechanisms for “ensure[ing] selected levels of quality of services.” Exhibit 1003, ¶¶157-159, 160; *see also* Exhibit 1003, ¶¶155-166. Dependent claim 6 then recites specific details of quality-of-service management, which I explained was disclosed by the combination I already explained a POSITA would have been motivated to make: Goossens2003’s SoC having a “first module” (the IP module) requesting a service, combined with Drake’s QoS management system. *See* Exhibit 1003, ¶¶192-198 (referencing my discussion of the same claim elements for Ground 1 at ¶¶127-134).

65. Dr. Bagherzadeh also argues, at paragraphs 268-273, that there was “no need” to use a QoS requestor manager 12 (communication manager) in the Goossens/Drake combination, and that for the combination a POSITA would instead have “understood that Goossens2003’s IP would just send its request directly to the QoS allocator” and that the allocator “would have received the request directly from, and provided its response to, Goossens2003’s IP.” *E.g.* Exhibit 2070, ¶¶268, 269, 273. I do not agree, for at least two reasons.

66. First, as I discussed above, the combination I explain a POSITA would be motivated to make is just using Drake’s QoS management components (the QoS

management requestor 12 and the QoS allocator 20), as Drake discloses them, to provide the communication management functionality that Goossens teaches. In that combination, the most obvious, straightforward thing to do is to just use Drake's system as disclosed and without further alteration. And the Drake QoS system has a QoS requestor manager 12 (communication manager) that receives and forwards the service request and also receives the response back from the QoS allocator (resource manager).

67. Second, as I also explained above, this argument ignores the role of interfaces in networking (on-chip and off-chip). While it is true that the QoS originates from the requesting module (for example application 11 in Drake's source station), a POSITA at the time knew it was good design practice to raise the level of abstraction for requests so that the requesting module did not need to be redesigned for different implementations. A POSITA would have recognized that Drake's QoS requestor manager 12 was an interface that understood how to forward the QoS request from the requesting module to the resource manager, as that is what Drake explains requestor manager 12 did (e.g., formatting the request appropriately). Exhibit 1006, 4:33-38. A POSITA would understand that the QoS requestor manager 12 would have this same function in the Goossens references. In fact, the Goossens references themselves explain that "network interfaces" are used to interface with the (interchangeable/reusable) IP modules. Exhibit 1007

(Goossens2003), 74 (“Both networks contain two components: routers and network interfaces ... network interfaces are required to bridge the network layer and transport layer”), 76-77 (“IPs negotiate with network interfaces to obtain connections with certain properties.”), Fig. 4.5; Exhibit 1005 (Goossens2003), §3, §4.

68. Dr. Bagherzadeh attempts to suggest that I do not sufficiently describe why the response sent from the resource manager would go to the communication manager. Exhibit 2070, ¶ 272. This ignores that the only communication path disclosed by Drake from the QoS allocator (the resource manager) to the application requesting QoS goes through QoS Requestor Manager 12 (the communication manager). Exhibit 1006, Fig. 1. As I describe throughout this Declaration and my First Declaration, a POSITA would have been motivated to use the QoS system from Drake, and this would include the data path. It would not have made sense to a POSITA to add a separate path that would inform the application request QoS about whether the QoS was available but would avoid first sending the response message to the QoS requestor manager 12.

69. Dr. Bagherzadeh also argues that the Goossens2003/Drake combination is “actually comparable to communication between Drake’s QoS allocator and [Drake’s] source station 10.” Exhibit 2070, ¶273; *see also id.* ¶198 (citing Exhibit 1006, 4:50-52, 11:22-38, 14:22-27, Drake’s description of the QoS

allocator communicating back and forth with “source station 10”). This does not show that a POSITA would have been motivated to modify Drake’s QoS and *omit* the QoS requestor manager 12 (communication manager). To the contrary, it further confirms why a POSITA would be motivated to make the combination as I have described. A POSITA would have understood that the QoS allocator 20 communicating with “source station 10” is in fact the QoS allocator communicating with QoS requestor manager 12 (in source station 10). Exhibit 1006, Fig. 1.

70. I see that the Patent Owner cites a number of passages in my deposition transcript (e.g., Paper 35 at pp.37 and 59) and contends that I say that Drake’s QoS allocator 20 receives service requests from and sends replies to a first module such as Goossens2002’s IPs. That is not an accurate characterization of what I say. If one looks at the transcript, the Patent Owner’s attorney asks whether the IP block issues a QoS request, and I said that it does. That is true—the request for a QoS connection does *originate* from the IP block, but as I discuss above, the communication manager (Drake’s QoS requestor manager 12) is an interface that understands how to forward this to the resource manager (Drake’s QoS allocator 20). As I explained above, to a POSITA at the time having such an interfacing component (Drake’s QoS requestor manager 12) was common and considered beneficial in both on-chip and off-chip networks. A POSITA at the time knew that

when one was connecting things together in a network (either on-chip or off-chip), it was beneficial (and common) to use interfaces to do that.

71. Dr. Bagherzadeh relies on the same arguments for Claim 10. Exhibit 2070, ¶276. For the same reasons I explained above with respect to claim 6, I disagree with Dr. Bagherzadeh.

**2. The Combination Teaches the Claim Limitations Regarding “Communication Manager” and “Resource Manager”**

**a. “communication manager” (claims 2, 10[a])**

72. Dr. Bagherzadeh argues that the combination of Goossens2003 and Drake fails to disclose the “communication manager,” but that is incorrect. In the *combination*, the communication manager would be implemented on the chip outside of the IP module. The only reason Dr. Bagherzadeh provides is his decision to “assume” that I rely on Drake’s source station 10 for the claimed “first module.” Exhibit 2070, ¶278. Having “assumed” this, Dr. Bagherzadeh then just asserts that because, as a matter of claim construction, the communication manager “is separate from, and cannot be within, the ‘first module,’” Drake cannot disclose the communication manager because the QoS requestor manager 12 is shown in Drake’s Figure 1 in the source station 10. I disagree with this for several reasons.

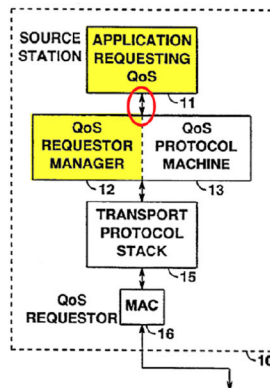
73. One, it is inconsistent with what a POSITA would understand from Drake’s disclosure and the combination with Goossens2003 that I explained in my First Declaration. Drake plainly teaches (and shows) the QoS requestor manager

12 as its own “component,” with its own function, that “interacts” with application 11. Exhibit 1006, 4:33-38 (for example, describing “component[] 12”), 4:62-65, Fig. 1. That Drake happens to show the QoS requestor manager 12 located in what Drake calls source station 10 does not change that. Dr. Bagherzadeh does not identify any reason that Drake’s QoS requestor manager 12, itself, is not or cannot be a communication manager.

74. Further, Dr. Bagherzadeh incorrectly “assume[s]” that I “rely” on “Drake’s source station 10 for the claimed ‘first module.’” *See* Exhibit 2070, ¶278 and note 10. As I explained above and in my First Declaration, I rely on the *Goossens2003* as disclosing the claimed “first module.” For examples, see my First Declaration at ¶170 (“Goossens2003 discloses this limitation”), ¶185 (claim 2), ¶¶193-194 (claim 6[a]), ¶205 (claim 10[a]); *see also id.* (Ground 1), ¶87 (claim 1[a]), ¶95 (claim 1[d]), ¶128 (claim 6[a]), ¶138 (claim 10.pre.1). Dr. Bagherzadeh doesn’t even acknowledge this.

75. Moreover, as I explained above, I pointed out in my First Declaration and at my deposition that the unit in Drake generating a request similar to the “first module” in Goossens was specifically *the application 11 in Drake’s source station*, not just Drake’s source station generally. For example, see paragraph 130 of my First Declaration:

130. Additionally, Drake also describes that the application in the source station (*i.e.*, first module) communicates this request to a QoS requestor manager (*i.e.*, communication manager) for subsequent transmission to a QoS allocator, as discussed further below. *See* Ex-1006 (Drake) at 3:21–28, 4:32–37, 4:62–65 (“The QoS requestor manager 12 interacts with application using internal messages as is well known in the art”), 5:5–8, Fig. 1. This is indicated by the arrows shown in Drake’s Figure 1 annotated below:



Ex-1006 (Drake) at Fig 1 (excerpted, annotated).

*See also* my First Declaration (Exhibit 1003) at ¶¶195 (same) and 177.

76. Dr. Bagherzadeh also contends that I was “unclear” or “fail[ed]” to specify where the “communication manager” is found in the prior art. Exhibit 2070, ¶279. That is not true. I explained that Drake discloses a communication manager, as QoS requestor manager 12. Exhibit 1003, ¶187 (citing *id.* ¶¶118-122). I also explained that Goossens2003 discloses communication management functionality—“network interfaces manage the communication between different modules.” Exhibit 1003, ¶¶185-186. I then explained that under Patent Owner’s definition in the district court case—which just requires a “circuit” that “manages

the communication between modules”—Goossens2003 would also disclose a communication manager. Exhibit 1003, ¶186.

77. Dr. Bagherzadeh argues that a POSITA would not have understood Goossens2003’s NIs to constitute a communication manager because the ’449 patent claims recite “communication manager” and “network interface means” separately. Exhibit 2070, ¶279. However, that does not mean one cannot be part of the other—the ’449 patent itself says that the communication manager can be in the network interface (Exhibit 1001, 10:5-6). Goossens2003 also teaches that communication management functionality can be implemented in Goossens2003’s network interface. Ex-1007, 76-77.

**b. Elements 6[a] and 10[a]**

78. Dr. Bagherzadeh argues that the “combination of Goossen2003 and Drake fails to meet limitations 6[a] and 10[a], which recite that a ‘first module [...] issue[s] a request [] for a connection with [said] second module[] to [said] communication manager,’” because he understands the Petition and myself to rely on “Drake’s source station 10 and QoS requester 12 therein for the claimed ‘first module’ and ‘communication manager,’” and Drake “does not teach that *source station 10*” issues a request as claimed. Exhibit 2070, ¶¶280-285. This is incorrect. The combination of Goossens2003 and Drake certainly does disclose limitations 6[a] and 10[a].

79. Dr. Bagherzadeh’s opinion is based on an incorrect assumption—that I rely on Drake’s source station 10 as the claimed “first module” that makes a “service request.” As I explained above and in my First Declaration, Goossens2003 discloses the claimed first module that is making the request. See for example Exhibit 1003, ¶193 (“Goossens2003’s master IPs (i.e., first module) issue requests...”), ¶194 (“Goossens2003 also explains that when IPs make a request...”). Dr. Bagherzadeh just ignores this.

80. Also, as I explained above, when I explained that in Drake a unit that could also be considered a “first module” sends a request to the communication manager (QoS requestor manager), I explained that it was application 11 in the source station 10 that was sending the request, not just the source station 10 generally. Exhibit 1003, ¶195 (“Drake similarly describes that an application in a source station (i.e., first module) ‘requests a QoS connection’...”). This is also what Drake’s Figure 1 shows, when it shows the Application 11 sending a request to QoS requestor 12. Dr. Bagherzadeh appears to agree with me on this point. Exhibit 2007, ¶¶281 (stating that the request in Drake comes from application 11); *see also* Exhibit 1037 at 53:4-18.

81. Dr. Bagherzadeh’s assumption that I rely just on Drake’s “source station 10” generally as the “first module” sending the request, and therefore the combination does not disclose the elements of 6[a] and 10[a] because Drake’s

“source station 10” cannot send a request to itself, is incorrect. It ignores both the combination that I am relying on and what Drake itself teaches.

**c. Elements 6[b] and 10[b]**

82. Dr. Bagherzadeh contends that the combination of Goossens2003 and Drake fails to meet limitations 6[b] and 10[b], which recite “[said] communication manager ... forward[s] [said] request ... to [a] resource manager.” Exhibit 2070, ¶286. Dr. Bagherzadeh contends that a POSITA would have understood that *“Drake teaches that the request received by QoS allocator 20 is sent from source station 10, not from QoS requestor manager 12 therein”* (Exhibit 2070, ¶287), and also that *“if a POSITA were to understand that any particular component(s) of source station 10 send or forward the request to allocator 20, it would be transport protocol stack 15 and/or MAC entity 16, not QoS requestor manager 12”* (Exhibit 2070, ¶288).

83. Neither of these assertions would make sense to a POSITA and neither is supported by Drake. As to the first—that because Drake describes the request received by the QoS allocator 20 as coming from “source station 10,” a POSITA would understand that it was *not* forwarded by the QoS requestor manager 12 *in* source station 10—Drake shows the opposite. Drake does not disclose an alternative path, where the request somehow bypasses QoS requestor manager 12. To the contrary, Drake’s Figure 1 shows one path—the request goes through (is

forwarded by) QoS requestor manager 12, and then ultimately is received by QoS allocator 20. The various lines with arrows illustrate the data path in this chain (i.e. 11↔12/13, 12/13↔15, 15↔16, 16↔17, etc.).

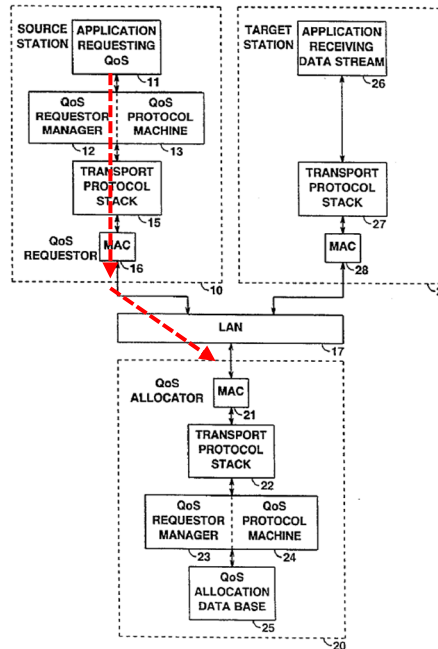


Exhibit 1006, Fig. 1 (annotated). Dr. Bagherzadeh confirms the request would follow this path from the application 11 requesting the QoS. Exhibit 1037 at 52:20-56:13. Drake also explains that it is the QoS requestor manager 12 specifically that “assemble[s] a request to reserve a QoS connection” (Exhibit 1006, 4:32–37) and that QoS requestor manager 12 “utilizes QoS protocol machine 13 to format the QoS request,” which Drake also refers to as a “QoS allocation request” (Exhibit 1006, 4:62–65, 8:31–49, 9:52–10:8).

84. A POSITA reading Drake’s description that the request received by QoS allocator 20 comes “from source station 10” would not ignore Drake’s other

teachings about where *in* source station 10 the request was forwarded from. And as I explained above, Drake's Figure 1 shows and explains the components in source station 10, and the path that the request takes through them. Saying that the request received by QoS allocator 20 is "from" source station 10 does not mean that it was not also forwarded by QoS requestor manager in source station 10.

85. As to Dr. Bagherzadeh's second assertion, that if a POSITA were to understand that any particular component in source station 10 was forwarding the request, it would be "transport protocol stack 15 and/or MAC entity 16, not QoS requestor manager 12" (Exhibit 2070, ¶288)—Drake does not support that either. A POSITA would understand the role of these components (protocol stack 15 and MAC entity 16) is to successfully transfer a data message that was received from QoS requestor manager 12 across the interconnect (LAN 17 in this case) to a destination. So, to the extent forwarding the request is just passing the message along, then every step in the chain in source station 10 is forwarding the message, including QoS requestor manager 12 and also then entities 15, and then 16. A POSITA would understand that for the request to ultimately arrive at QoS allocator 20 from source station 10, the QoS requestor manager 12 has to forward it.

86. To a POSITA, Drake also discloses that QoS requestor manager 12 forwards the request in a way that entities 15 and 16 do not. A POSITA would understand that application 11 wants to communicate with 26 in target station 29.

However, Drake teaches that the QoS requestor manager 12 realizes that it must first send a message somewhere else: to allocator station 20. Therefore, QoS requestor manager 12 is also forwarding in the sense of redirecting communication, in that it realizes the request should be sent to 20 (not 29).

87. Finally, I note that Dr. Bagherzadeh does not identify anything in the claim or the '449 patent that requires the claimed forwarding of the request “to” a resource manager to exclude a situation like Drake, where the request is forwarded by the QoS requestor manager 12 (communication manager), and then subsequently also goes to the resource manager through other components. And I do not see anything. The claim simply says that the communication manager forwards the request “to [the] resource manager.” Exhibit 1001, cl. 6[a], 10[a]. That is exactly what Drake shows the QoS requestor manager 12 doing—it forwards the request to QoS allocator 20 (through entities 15 and 16).

**d. Elements 6[c] and 10[d]**

88. Dr. Bagherzadeh contends that the combination of Goossens2003 and Drake fails to meet limitations 6[c] and 10[d], which recite “[said] resource manager [] responding with the availability of the requested [target] connection to [said] communication manager.” Exhibit 2070, ¶292. Dr. Bagherzadeh provides the opinion that a POSITA would understand that Drake “instead” teaches that QoS allocator 20 sends the QoS allocation reply to “source station 10 ... *in general*.”

Exhibit 2070, ¶293 (emphasis added). In doing so, Dr. Bagherzadeh cites teachings in Drake about sending the response to “source station 10.” For example, Dr. Bagherzadeh cites and quotes the following parts of Drake, at his Exhibit 2070 declaration, ¶293: Exhibit 1006, 4:54-57 (“station 10 is notified”), 8:20-29 (describing “QoS allocation reply” as a “message[] exchanged between the stations of the [LAN]”), 14:22-27 (“AllocationReply message is ... sent from allocator station 20 ... to source station 10,” and “source station 10 ... receives the AllocationReply message”), 11:22-48 (describing that allocation acceptance/denial messages in Fig. 5 are sent to, and received by, “source station 10”), 9:52-10:2 (describing a “QoS message,” such as QoS allocation reply, as including a “source address field 52 (Source MAC Address)” for the source station).

89. I disagree that these portions of Drake show that the QoS allocation reply in Drake is sent only to the source station 10 in general, and not to QoS requestor manager 12 in the source station. This would mean that the reply message would stop at the boundary of source station 10, which is nonsensical to a POSITA. To the contrary, the portions of Drake cited by Dr. Bagherzadeh reinforce a POSITA's understanding that the reply is sent to QoS requestor manager 12 (the communication manager) in source station 10.

90. Drake does not just disclose a black box source station 10. Instead, Drake shows and describes the components in source station 10, and the data path

through those components. A POSITA would understand that the only way Drake's QoS works is for the reply regarding the QoS request to get back to application 11 in source station 10—the entity that originally made the QoS request. And Drake shows that to get back to application 11, the reply is first sent to QoS requestor manager 12, which Drake teaches is the interface with application 11. *E.g.*, Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). This is also clearly shown in Drake's Figure 1 (double-headed arrows for the data path, indicating to a POSITA that the reply comes back through the same path as the request):

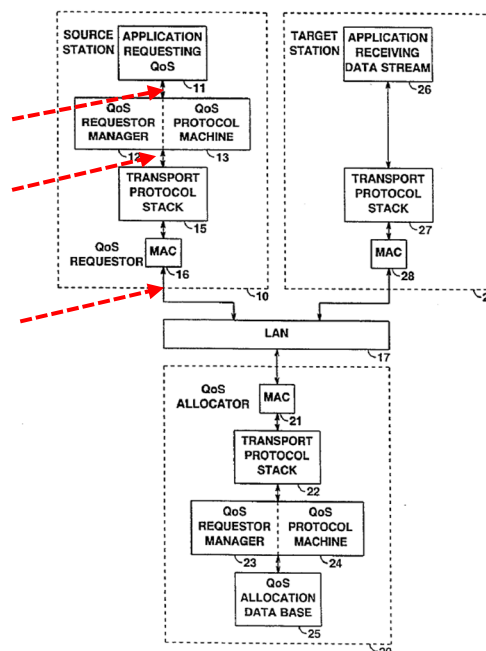


Exhibit 1006, Fig. 1 (annotated). A POSITA would understand that Drake's quality of service system would not work if the reply stopped at the boundary of source station 10. Instead, a POSITA would understand that Drake's teaching of sending a

reply “to source station 10” is teaching responding with availability of the connection to the communication manager (QoS requestor manager) as claimed.

91. Dr. Bagherzadeh also contends that to the extent the QoS allocation reply is sent “to an internal component of source station 10,” a POSITA would understand that it was sent “to application 11, not QoS requestor manager 12,” because “application 11 is the original requestor and is the component that [also] sends a multimedia file via the established connection.” Exhibit 2070, ¶294. To a POSITA, this is plainly wrong, and Drake does not support this understanding. I also note that Dr. Bagherzadeh agreed that the message does go from QoS allocator to the application requesting QoS using the same path described above. Exhibit 1037 at 59:22-60:8.

92. As I explained above Figure 1 shows the data path for communication (the lines with arrows at each end), and this indicates to a POSITA that the reply message goes back to component 12 before anything goes back to component 11. Exhibit 1006, Fig. 1. And Drake also expressly explains that QoS requestor manager 12 is the thing that interfaces with application 11—a POSITA would understand this is true for messages going in both directions. *See* Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). Moreover, there is no teaching in Drake of an alternative reply message route that bypasses the QoS requestor manager 12. Dr. Bagherzadeh states that the reply “could have

been sent to application 11 or some component not depicted in Drake's Figure 1" (Exhibit 2070, ¶294), but that is not what Drake shows. Drake shows the reply message traveling back through QoS requestor manager 12 and QoS Protocol machine 13. *See* Exhibit 1040 at 60:3-61:13.

93. Not only would a POSITA understand Drake to be teaching (and showing) that the QoS allocation reply goes back to the QoS requestor manager 12, a POSITA would also understand that is how Drake worked. A POSITA would know that for Drake's QoS system to function, the QoS allocation reply message would need to get back to the application 11 that made the original quality of service request (and that is the same component that then sends a multimedia file over the established link). Dr. Bagherzadeh does not appear to disagree with this. *See*, for example, Exhibit 2070, ¶294 (acknowledging that if the QoS allocation reply is sent to an internal component of source station 10, it "would be sent to application 11"). And any POSITA of common sense would realize that the QoS allocation reply sent "to station 10" (as Drake expressly discloses, e.g., Exhibit 1006, 4:54-57, 8:20-29, 14:22-27, 11:22-48, and 9:52-10:2) means that the reply is sent to internal components of station 10—it would not just stop at the boundary of station 10.

**e. Elements 6[d] and 10[e]**

94. As I explained in my First Declaration, Goossens2002 discloses a first

and second module that a connection is established between. Dr. Bagherzadeh does not appear to dispute this.

**C. GROUND I: Claims 2, 6, and 9-16 are Rendered Obvious by Goossens2002 in View of Drake**

95. As described in my First Declaration, in my opinion the claims of the '449 patent would have been obvious in light of Goossens2002 in view of Drake. Neither Dr. Bagherzadeh's declaration nor the Patent Owner's Response change my conclusion.

**1. The Goossens2002 and Drake Combination Teaches the "Independently Configurable [/Adjustable]" Claim Limitations**

96. As I explained in my First Declaration, Goossens2002 discloses channels that are independently configurable / adjustable as recited by Claims 2 and 10. Dr. Bagherzadeh contends that a POSITA reviewing Goossens2002 "would not assume that the channels of one connection are 'independently configurable,' under the agreed upon construction in the Qualcomm Litigation." Exhibit 2070, ¶155. That construction was "able to be configured with connection properties that do not depend on those of another configuration." Exhibit 2070, ¶155.

97. Dr. Bagherzadeh is not correct. Applying the construction from the Qualcomm Litigation would not change my opinion—Goossens2002 would still disclose channels that are independently configurable / adjustable as claimed.

98. As Dr. Bagherzadeh acknowledges, Goossens2002 discloses that “[a] channel can have a combined throughput and latency guarantee (GT) or best effort (BE), with an unspecified latency guarantee,” and that “[a] connection combines multiple channels.” Exhibit 1005 at 3. Dr. Bagherzadeh then just asserts that Goossens2002 does not “expressly” say that one channel can be GT while the other channel can be BE, and that a POSITA would not “assume” this was so. But he does not provide any *explanation* for that assertion. And I do not think there is one. Contrary to what Dr. Bagherzadeh asserts, a POSITA reading Goossens2002’s description of a system where a channel can have either GT or BE properties and that a connection combines multiple channels would immediately understand that each of those multiple channels is able to be configured with connection properties (e.g., GT or BE), that do not depend on those of another configuration.

99. Goossens2002 also expressly says that the Aethereal NOS “intends to support *both* guaranteed [GT] and best-effort [BE] services.” Exhibit 1005, 3 (emphasis added). At a minimum Goossens2002 discloses a mixture of GT and BE channels, which indicates to a POSITA that the configuration of one channel as GT does not depend on the configuration of another channel (e.g., as BE).

100. As I also explained in my First Declaration, Drake also discloses channels that are independently configurable / adjustable as claimed. Dr. Bagherzadeh contends that a POSITA would understand Drake’s discussion of

communicating in the forward direction and in the backward direction to disclose communication only along the “same” channel, and therefore (in his opinion) Drake does not disclose multiple channels. Exhibit 2070, ¶¶156-158. Dr. Bagherzadeh contends that a POSITA would not refer to communication on multiple channels as “forward” and “backward” communication, but rather as “request” and “response.” I disagree. It is not true that a POSITA would understand the words “forward” and “backward” to necessarily refer specifically to single-channel communication. In the context of Drake’s disclosure, a POSITA would understand that “forward” refers to going from source to target, and “backward” refers to going from target to source. *See* Exhibit 1006, 5:34-6:29. In other words, in this context, a POSITA would understand forward and backward refer to two channels.

101. Dr. Baghezadeh also argues that Drake “relates to Token Ring networks,” which employ “duplex” communication, and that “duplex” is a term of art that means only one channel. Exhibit 2070, ¶158. This is incorrect, for at least three reasons.

102. ***One***, a POSITA would not understand Drake’s disclosure to be *limited* to Token Ring networks. Token Ring networks are merely one *example*. Exhibit 1006, 4:21-23 (“LAN 17 may, for example, be a Token Ring network.”). Drake likely provided Token Rings as an example of a LAN because Drake was an IBM

patent, and IBM was the company that came up with the specific Token Ring system. Drake's Abstract and claims are broadly directed to "local area network[s]." *E.g.*, Exhibit 1006, Abstract and cl. 1. And a POSITA would have known that there were many LANs at the time, other than just specifically Token Ring networks. The functionality of Drake does not depend upon any other specific details on the LAN. For example, a POSITA would have understood that an Ethernet LAN (or any other type of packet-based LAN) would also work perfectly well with Drake.

103. **Two**, a POSITA would not understand that Token Ring networks "employ duplex *channels*," as the Patent Owner argues. What the Token Ring references Dr. Bagherzadeh cites (Exhibits 2076 and 2077) actually disclose is "duplex *communication*" or "duplex *mode*," not duplex *channels*. The fact these documents do not teach use of duplex *channels* is significant. While using a "duplex *channel*" would indeed indicate two-way communication using only one channel, merely using duplex *communication* or duplex *mode* would not. For example, Goosens2002 points out that "half-duplex" and "full-duplex" connections are implemented at the "session layer" (above the transport layer, which is where simplex channels naturally exist) by combining multiple connections: "In the session layer multiple connections are combined into services such as multicast, half duplex and full duplex connections." Exhibit 1005, 2.

104. Exhibit 2076 is a one-page document, that appears to be from 2021 (long after the relevant priority date that I understand applies to this matter), that mentions “duplex mode negotiation.” It talks about “duplex mode” (either half-duplex mode or full-duplex mode), but it does not discuss “duplex channels.” It certainly does not indicate or teach that anything using duplex communication is only single-channel communication.

105. In Exhibit 2077, the word “duplex” appears 23 times. The word “channel” appears 5 times, but it is unrelated to “duplex” (or communication across the ring); it instead refers to a different type of communication. “Duplex channel” never appears in Exhibit 2077. When “duplex” is used in Exhibit 2077, it is referring a specific “mode” of operation of the ring. This is discussed in the section entitled “Full-duplex dedicated Token Ring.” Exhibit 2077, 22. Figure 1 of Exhibit 2077 illustrates what is meant by “full duplex”:

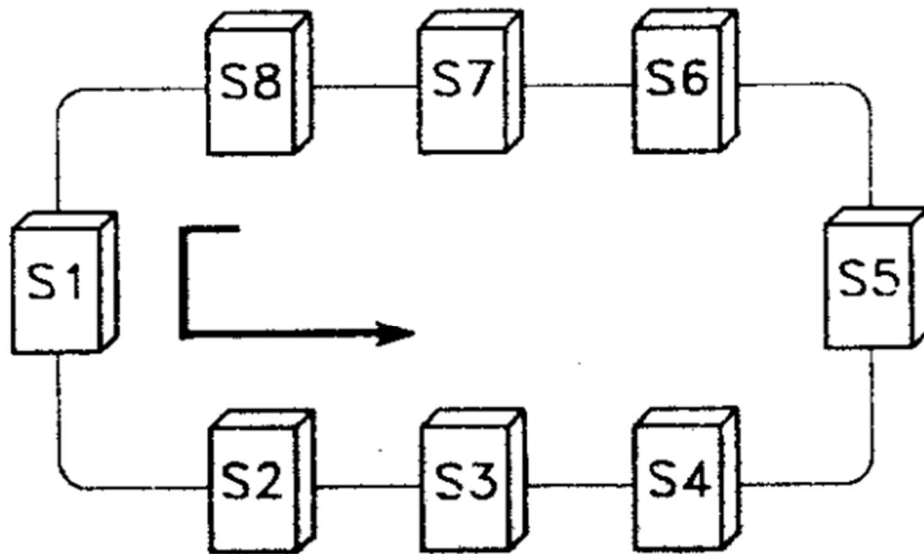


*Figure 1. Single token ring with eight ring stations*

106. In Figure 1, imagine that nodes S1 and S5 want to communicate with each other. Exhibit 2077 suggests that each link between nodes in the ring has a bandwidth of 16 Mbps. Exhibit 2077, at 16 (exhibit page 22). So if one thinks of a stream of data from S1 to S5, the communication rate for that data stream would be limited to 16 Mbps (since it is limited by the link speed between nodes in the ring). Also, from the perspective of node S2, as this communication stream is going through its connection in the ring, data would be flowing in (from S1) at 16 Mbps, and then getting forwarded on to S3 at 16 Mbps. Now, imagine that S1 is communicating a stream of data to S5, while S5 is simultaneously communicating another stream of data back to S1. It is possible to do this at full network speed because the stream of data from S1 to S5 follows the (unidirectional) path from left

to right on the bottom part of Figure 1, while the stream of data from S5 to S1 follows the right-to-left path along the top half of Figure 1. Because these two streams of data are using separate links, they can both co-exist simultaneously. This is the “full duplex” scenario Exhibit 2077 is describing. It is not describing “full-duplex channels,” but rather a scenario where the destination of a data stream (e.g., S5) is simultaneously receiving one data stream while also sending out a second data stream (to S1) at the same time. So, the Token Ring system Dr. Bagherzadeh cites does support duplex *communication* (e.g., where nodes S1 and S5 are both communicating with each other at the same time), but so do NOCs. And that does not indicate that the duplex communication is happening on only one channel.

107. **Three**, a POSITA would understand that the preferred embodiment of Drake supports multiple independent channels, and that it would not work with just a single full-duplex channel. For example, the algorithm for allocating a QoS transmission path (described at Exhibit 1006, 10:9-11:48, and Figs. 4-5) shows that it supports multiple of these QoS transmission paths (which would be channels). The algorithm is called with separate source and target nodes, and each invocation has its own QoS parameters. Also, the details of the allocation database 25, which are described Exhibit 1006, 6:35-55, shows that there are multiple channels. For each QoS Requestor (i.e. source), there is a “StreamCount: A count of the number

of data streams for which QoS is currently allocated by this requestor.” Exhibit 1006, 6:62-63. This means it supports more than one outgoing channel for each source in the network. Further, both LANAttachment 35 and BridgeAttachment 36 also have “StreamCount” parameters, which count the number of QoS reserved data streams that pass through that component. Exhibit 1006, 8:11-12, 8:18-19. In addition, Drake consistently uses “data streams” (plural). Exhibit 1006 Abstract (“A hierarchical data base containing all of the current parameters of all of the network resources and all of the currently supported data streams is used to support the allocation process.”), 5:8-14 (“In carrying out its management functions, manager 23 utilizes the data in QoS allocation data base 25 which, in general, contains all of the quality of service parameters for each of the components of LAN 17, as well as descriptions of all of the data streams already accepted by allocator 20 and the detailed constituents of the connection paths for each of those data streams.”).

108. A POSITA would also recognize that Drake would not work if the connection for transmitting the forward and backward communication was only a single full-duplex channel. (This is also clearly not what Drake discloses, for the reasons I explained above—for example, the database keeps track of “StreamCount” variables to record the number of active connections). But if it were the case that Drake somehow communicated via only a single full-duplex

channel, either Drake would have to have only one active connection at a time, or all connections would be forced to have the same QoS properties. Neither would work for what Drake discloses. If there were only one active channel at a time, subsequent allocation requests would have to wait until that initial connection was deallocated. A POSITA would recognize that this creates a problem—once the single-channel connection is established, it may be active for a long period of time. For example, if a user is streaming a movie, the single-channel connection (with QoS properties) might last for two hours. If only a single active connection is allowed at a time, it means that the network will effectively be shut down to other QoS-related requests for that (extremely long) period of time. A POSITA would immediately recognize that as an unworkable design. If, on the other hand, multiple connections were allowed but were all forced to use the same connection properties, a POSITA would recognize that as unworkably inefficient (and also requiring an additional mechanism that Drake does not describe or show).

**2. A POSITA Would Have Been Motivated to Combine Goossens2002 with Drake's QoS Management System with a Reasonable Expectation of Success**

**a. Response to General Arguments**

109. I incorporate by reference paragraphs 17-46 above, as the arguments Dr. Bagherzadeh makes are not supported for the same reasons.

**b. The Contentions Regarding Purported “Redundancy” and “Teaching Away,” In Particular Regarding the “Communication Manager” of Claim 2, Are Unsupported**

110. Dr. Bagherzadeh suggests that I do not adequately explain why QoS Requestor Manager 12 would be retained in the combination of Goossens2002 and Drake. Exhibit 2070, ¶176. I disagree. My First Declaration explains that a POSITA would have been motivated to combine Drake’s quality of service management process with Goossens2002 because Drake provided details that were missing from Goossens2002, and a POSITA would have understood that using Drake’s quality of service management process would have improved the performance of Goossens2002. *See* Exhibit 1003, ¶¶ 75-79. QoS requestor manager 12 is an important aspect to Drake’s quality of service management process, and a POSITA would not have simply removed it when they were using Drake’s quality of service management process with Goossens2002. *See* Exhibit 1003, ¶¶118-122.

111. Dr. Bagherzadeh then argues that a POSITA would not have been motivated to use the QoS requestor manager 12 component from Drake in the combination with Goossens2002. Exhibit 2070, ¶¶ 177-190. Based on my review of his declaration, it appears he is making two basic arguments. He first argues that QoS requestor manager 12 would be redundant in Goossens2002. He then argues that Drake and/or Goossens2003 teaches away from the combination that

includes QoS requestor manager 12. I do not agree with either of his arguments.

**No Redundancy**

112. Dr. Bagherzadeh first argues that I did not show that Drake's QoS requestor manager 12 was "needed" in Goossens2002 and he argues that its function is "redundant of Goossens2002's IP block." Exhibit 2070, ¶¶177-78. As I explained in my First Declaration, Goossens2002 teaches a NOC and the importance of managing network resources to provide guaranteed services, but it does not include certain details regarding implementation of its guaranteed services and connections. Exhibit 1003, ¶¶73-75. Drake, on the other hand, "provides details regarding the implementation of these features," and a POSITA would have found Drake's known solution for "ensur[ing] selected levels of quality of services" helpful in implementing an SoC like the one described by Goossens2002. Exhibit 1003, ¶76. Therefore, as I explained in my First Declaration, a POSITA would have been motivated to use Drake's example of a way to implement the QoS functionality that Goossens2002 already taught. Exhibit 1003, ¶¶73-79. And the example Drake provides uses a communication manager (QoS requestor manager 12) and a resource manager (QoS allocator 20). Exhibit 1003, ¶¶91-94, 118-22. So in the combination I already explained a POSITA would have been motivated to use, there is a communication manager. As I explained above, the QoS requestor manager 12 was an important aspect to Drake's quality of service

management process, and a POSITA would not have simply removed it in the combination.

113. In addition, in the combination, the “QoS requestor manager” would not be redundant of the functioning of Goossens2002’s IP blocks. As I discussed above, network interfaces “decouple computation from communication” by enabling their host/IP module to communicate without requiring specific knowledge of the low-level network details. A POSITA would understand that components such as the QoS requestor manager 12 would desirably be included in the NOC of Goossens2002.

114. Specifically, a POSITA would understand that Goossens2003’s NOC would desirably include a component to take the signal from the “re-use[d]” IP blocks and communicate with the network. Exhibit 1005 at 1. Different such re-used IPs would typically use different protocols to describe the type of service it wanted to send its data over. Exhibit 1010 at 1. Therefore, Goossens2002’s NOC would have been expected to use a component such as Drake’s QoS requestor manager 12 to assemble/format a request that could be understood by the network. That is exactly what Drake teaches the QoS requestor manager 12 is for. Exhibit 1006, 4:33-35 (“components 12 and 13 assemble a request”), 4:61-65 (“QoS requestor manager 12 interacts with application 11...to format the QoS request.”). A POSITA would not expect such an interfacing component would already be

included in the “bought or premade” IP block, which does the computation and sends messages without regard to the communication system it is using. The fact that this specific component was not disclosed in Goossens2002 is the result of Goossens2002’s disclosure not including the details of the QoS components. A POSITA would not expect that Goossens2002’s network would be able to perform its QoS functionality without a component to assemble the QoS request from the different bought or premade IP protocols. Therefore, to a POSITA the predictable, beneficial combination of Goossens2002’s NOC with Drake’s QoS teachings would be to simply use Drake’s QoS components communicating with each other as Drake taught (including the QoS requestor manager 12 that “assemble[s]”/“format[s]” the request).

### **No Teaching Away**

115. Dr. Bagherzadeh’s teaching away argument (Exhibit 2070, ¶¶ 179-189) applies an overly rigid and unrealistic view of the prior art that ignores a POSITA’s understanding of both off-chip and on-chip networks. Goossens2002 does not “teach away,” and the proposed combination would not be a “modification of Drake’s teachings” as Dr. Bagherzadeh contends. For example, as I discussed above, a POSITA would understand that the QoS requestor manager 12 was part of Drake’s network interface components, and *in the network-on-chip context of Goossens2002*, a POSITA would simply implement components for

QoS network interfacing (e.g., QoS requestor 12) where Goossens2002 taught to put network interfacing components in its network-on-chip—outside its IP module. *See* ¶¶ 25-32, *supra*. Therefore, the fact that Drake shows QoS network interfacing components (such as QoS requestor manager 12) in the source station (e.g., in a server) on the off-chip environment would not have dissuaded a POSITA from implementing such components in the on-chip environment in the place where Goossens2002 taught they would be: outside the IP modules.

116. Dr. Bagherzadeh attempts to suggest it is relevant that in one part of the specification Drake refers to source station 10 as “requestor station 10.” Exhibit 2010, ¶179. I do not understand why this is relevant or surprising. Source station 10 contains the components that are making the request for QoS, which is all a POSITA would understand from Drake’s reference to it as requestor station 10. This nomenclature would not suggest to a POSITA that all “QoS Requestor Managers” must be in an that station, much less that it must be in an IP module in the combination with one of the Goossens reference. And Drake also discloses a QoS requestor manager (QoS Requestor Manager 23) outside of “requestor station 10.” *See* Exhibit 1006, Fig. 1.

117. Similarly, Dr. Bagherzadeh attempts to suggest it is relevant that QoS requestor manager 12 is not illustrated as being within “prior art [LAN] 17.” Exhibit 2010, ¶179. The fact that a block in Drake is described as “prior art [LAN]

17” does not suggest that the QoS components cannot be implemented outside of IP modules in the on-chip context of Goossens2002. All that nomenclature would have suggested to the POSITA is that the portion of the system that is denoted by the box is a local area network and is non-novel; it does not suggest that the other components are not connected to the LAN. In any event, this would not dissuade a POSITA from putting the QoS requestor manager 12 in the network interface in Goossens2002, just like it is part of the network interface components in Drake. Dr. Bagherzadeh cites Drake at 2:57-3:3 as purportedly teaching that QoS management should not be implemented in the LAN because doing so would be “complex,” and contends this is evidence of teaching away. Exhibit 2070, ¶206. This discussion in Drake does not teach away, and in fact is entirely consistent with the combination a POSA would be motivated to make. Drake references that in its system most of QoS management is “concentrated in a single QoS management station on the LAN” and that this “reduces complexity of introducing QoS management into existing Lan networks.” Exhibit 2006, 2:66-3:3. A POSITA would understand this as a reference to Drake’s *QoS allocator 20 station* (the resource manager). In the combination a POSITA would understand the resource manager (Drake’s QoS allocator 20) could be implemented as single piece of centralized hardware on the NOC in Goossens2002 (just like in Drake QoS allocator 20 is shown as a single piece of centralized hardware “on the LAN” –

meaning connected to the LAN). Exhibit 1003, 173 (a POSITA “would have been motivated to implement the functionality of Drake’s QoS allocator in a circuit on the same integrated circuit chip as Goossens2002’s IP blocks and NOC”); Exhibit 2078, 212:20-213:3. And nothing about this teaching (or any other teaching in Drake) criticizes or discredits locating the separate component of the *QoS requestor manager 12* (the communication manager) outside of the IP module in the network interface network-on-chip context such as that disclosed by the Goossens references.

118. Dr. Bagherzadeh further suggests that the QoS requestor manager is software running on source station 10 and suggests that this teaches away from including it in Goossen’s NoC. Exhibit 2070, ¶180. As I explain above, I do not agree that a POSITA would understand QoS requestor manager 12 to be software. Instead, a POSITA would understand Drake to be disclosing and describing application 11 and QoS requestor manager 12 as separate “components.” *See supra*, ¶¶30-31. Therefore, there would be no reason this component would not be located outside of the IP block in Goossens2002.

119. Additionally, whether or not QoS requestor manager 12 were software, it would have been well within the skill of a POSITA to implement such a communication manager in Goossen2002’s network interface as hardware or software. Therefore, whether Drake’s requestor manager 12 is software or

hardware would not affect a POSA's motivation to incorporate such a communication manager in Goossens2002's NOC.

120. Dr. Bagherzadeh suggests he reads my First Declaration to suggest that a POSITA would modify the IP block of Goossens2002. Exhibit 2070, ¶182, 184. I do not understand how he reached this conclusion. As the Board recognized in its institution decision, my opinion was premised on implementing Drake's QoS management process outside of the Goossens2002 IP block. *See supra*, ¶¶2528-29, 38-39, 52, 55.

121. Dr. Bagherzadeh also asserts that my First Declaration is "ambiguous and unclear as to specifically how the relied-upon components and functions of Drake's QoS system, particularly the QoS requestor manager, would purportedly work with Goossens2002's IP blocks and NOS." Exhibit 2070, ¶183. I disagree. Among other things, my First Declaration stated:

A POSITA would have been particularly interested in Drake's disclosure of a "QoS requestor manager" to formulate requests for guaranteed services with particular properties, and a "QoS allocator" to determine whether the network resources required for the requested services are available.

Exhibit 1003, ¶76 (emphasis added). This is a description of how the components would work with Goossens2002's IP blocks and NoC.

122. Dr. Bagherzadeh also asserts that Drake and Goossens2002 "teach completely opposite solutions" because QoS requestor manager 12 is in the source

station 10 of Drake. Exhibit 2070, ¶187. As discussed above in paragraphs 25-29 above, Drake's inclusion of the QoS requestor manager 12 in the source station 10 teaches a POSITA that an obvious place to put it would be in the network interface of Goossens2002. POSITAs did not typically view a "network," "NoC" or "LAN" as having rigid boundaries like suggested by Dr. Bagherzadeh. To the contrary, POSITAs looked at components and how they interact, which a POSITA understood was *the same* for both on-chip and off-chip network. For an off-chip network: the components included an application, which interacted with a network interface, which interacted with the transport components to transport a message. And for an on-chip network: the components also included an IP module, which interacted with a network interface, which interacted with transport components to transport a message. Because a POSITA understood it is the interaction of the components that matters, the fact that Drake labeled a particular part of its system "LAN" or the fact that Goossens2002 described a particular part of its system as "network" would not teach a POSITA that those are the only components that would be used. Thus, Drake does not teach either (a) putting the QoS requestor manager 12 in an IP module or (b) keeping the QoS requestor manager 12 out of the NOSs suggested by Dr. Bagherzadeh.

123. Drake's reference to a "prior art" LAN also does not suggest that the network must "stay[] the same" and does not teach away from including QoS

components with the Goossens2002's NOS, as asserted by Dr. Bagherzadeh. See Exhibit 2070, ¶187. To the contrary, as I explained above, in the network-on-chip context a POSITA would have been motivated to implement QoS components with the NOC, outside of the IP modules, exactly as Goossens2002 teaches to do.

**c. Dr. Bagherzadeh's Assertions Regarding Claim 6 (and Claim 10) Are Unsupported**

124. I disagree with Dr. Bagherzadeh's assertion that I did not explain motivation for what is recited by claim 6 (and claim 10). *See* Exhibit 2070, ¶¶191-200.

125. Dr. Bagherzadeh appears to misunderstand the combination. In my First Declaration, I explained why a POSITA would have been motivated to combine Goossens2002, which discloses a system-on-chip (referred to in Goossens2002 as a system on silicon) that provides guaranteed services using "connections," with Drake's description of components and mechanisms for "ensure[ing] selected levels of quality of services." Exhibit 1003, ¶¶73-79. Dependent claim 6 recites specific details of quality-of-service management, which I explained was disclosed by the combination I already explained a POSITA would have been motivated to make: Goossens2002's SoC (having a "first module" (the IP module) requesting a service, combined with Drake's QoS management system. *See* Exhibit 1003, ¶¶127-134.

126. Dr. Bagherzadeh also argues, at paragraphs 193-198, that there was

“no need” to use a QoS requestor manager 12 (communication manager) in the Goossens/Drake combination, and that for the combination a POSITA would instead have “understood that Goossens2002’s IP would just send its request directly to the QoS allocator” and that the allocator “would have received the request directly from, and provided its response to, Goossens2002’s IP.” *E.g.* Exhibit 2070, ¶¶194, 195, 198. I do not agree, for at least two reasons.

127. First, as I discussed above, the combination I explain a POSITA would be motivated to make is just using Drake’s QoS management components (the QoS management requestor 12 and the QoS allocator 20), as Drake discloses them, to provide the communication management functionality that Goossens teaches. In that combination, the most obvious, straightforward thing to do is to just use Drake’s system as disclosed and without further alteration. And the Drake QoS system has a QoS requestor manager 12 (communication manager) that receives and forwards the service request and also receives the response back from the QoS allocator (resource manager).

128. Second, as I also explained above, this argument ignores the role of interfaces in networking (on-chip and off-chip). While it is true that the QoS originates from the requesting module (for example application 11 in Drake’s source station), a POSITA at the time knew it was good design practice to raise the level of abstraction for requests so that the requesting module did not need to be

redesigned for different implementations. A POSITA would have recognized that Drake's QoS requestor manager 12 was an interface that understood how to forward the QoS request from the requesting module to the resource manager, as that is what Drake explains requestor manager 12 did (e.g., formatting the request appropriately). Exhibit 1006, 4:33-38. A POSITA would understand that the QoS requestor manager 12 would have this same function in the Goossens references. In fact, Goossens2002 itself explains that "network interfaces" are used to interface with the (interchangeable/reusable) IP modules. Exhibit 1005 (Goossens2003), 3 §3.

129. Dr. Bagherzadeh attempts to suggest that I do not sufficiently describe why the response sent from the resource manager would go to the communication manager. Exhibit 2070, ¶197. This ignores that the only communication path disclosed by Drake from the QoS allocator (the resource manager) to the application requesting QoS goes through QoS Requestor Manager 12 (the communication manager). Exhibit 1006, Fig. 1. As I describe throughout this Declaration and my First Declaration, a POSITA would have been motivated to use the QoS system from Drake, and this would include the data path. It would not have made sense to a POSITA to add a separate path that would inform the application request QoS about whether the QoS was available but would avoid first sending the response message to the QoS requestor manager.

130. Dr. Bagherzadeh also argues that the Goossens2002/Drake combination is “actually comparable to communication between Drake’s QoS allocator and [Drake’s] source station 10.” Exhibit 2070, ¶198 (citing Exhibit 1006, 4:50-52, 11:22-38, 14:22-27, Drake’s description of the QoS allocator communicating back and forth with “source station 10”). This does not show that a POSITA would have been motivated to modify Drake’s QoS and *omit* the QoS requestor manager 12 (communication manager). To the contrary, it further confirms why a POSITA would be motivated to make the combination as I have described. A POSITA would have understood that the QoS allocator 20 communicating with “source station 10” is in fact the QoS allocator communicating with QoS requestor manager 12 (in source station 10). Exhibit 1006, Fig. 1.

131. I see that the Patent Owner cites a number of passages in my deposition transcript (e.g., Paper 35 at 37) and contends that I say that Drake’s QoS allocator 20 receives service requests from and sends replies to a first module such as Goossens2002’s IPs. That is not an accurate characterization of what I say. If one looks at the transcript, the Patent Owner’s attorney asks whether the IP block issues a QoS request, and I said that it does. That is true—the request for a QoS connection does *originate* from the IP block, but as I discuss above, the communication manager (Drake’s QoS requestor manager 12) is an interface that

understands how to forward this to the resource manager (Drake's QoS allocator 20). As I explained above, to a POSITA at the time having such an interfacing component (Drake's QoS requestor manager 12) was common and considered beneficial in both on-chip and off-chip networks. A POSITA at the time knew that when one was connecting things together in a network (either on-chip or off-chip), it was beneficial (and common) to use interfaces to do that.

132. Dr. Bagherzadeh relies on the same arguments for Claim 10. Exhibit 2070, ¶200. For the same reasons I explained above with respect to claim 6, I disagree with Dr. Bagherzadeh.

**3. The Combination Teaches the Claim Limitations Regarding “Communication Manager” and “Resource Manager”**

**a. “communication manager” (claims 2, 10[a])**

133. Dr. Bagherzadeh argues that the combination of Goossens2002 and Drake fails to disclose the “communication manager,” but that is incorrect. In the combination, the communication manager would be implemented on the chip outside of the IP module. The only reason Dr. Bagherzadeh provides is his decision to “assume” that I rely on Drake's source station 10 for the claimed “first module.” Exhibit 2070, ¶212 and note 5. Having “assumed” this, Dr. Bagherzadeh then just asserts that because, as a matter of claim construction, the communication manager “is separate from, and cannot be within, the ‘first module,’” Drake cannot disclose the communication manager because the QoS requestor manager 12 is shown in

Drake's Figure 1 in the source station 10. Exhibit 2070, ¶212. I disagree with this for several reasons.

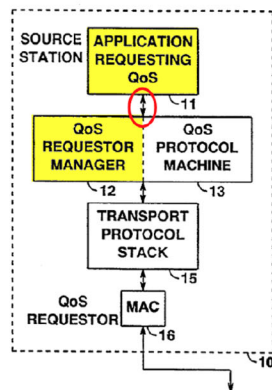
134. One, it is inconsistent with what a POSITA would understand from Drake's disclosure and the combination with Goossens2002 that I explained in my First Declaration. Drake plainly teaches (and shows) the QoS requestor manager 12 as its own unit, with its own function. Exhibit 1006, 4:33-38, 4:61-65, Fig. 1. That Drake happens to show the QoS requestor manager 12 located in what Drake calls source station 10 does not change that. Dr. Bagherzadeh does not identify any reason that Drake's QoS requestor manager 12, itself, is not or cannot be a communication manager.

135. Further, Dr. Bagherzadeh *incorrectly* "assume[s]" that I rely on "Drake's source station 10 for the claimed 'first module.'" See Exhibit 2070, ¶¶212, 233, 280, 283, and note 5. As I explained above and in my First Declaration, I rely on Goossens2002 as disclosing the claimed "first module." For examples, see my First Declaration at ¶87 (claim 1[a]), ¶95 (claim 1[d]), ¶128 (claim 6[a]), ¶138 (claim 10.pre.1). Dr. Bagherzadeh doesn't even acknowledge this, and instead just "assumes" that I rely on Drake's source station for the claimed "first module" element.

136. Moreover, as I explained above, I pointed out in my First Declaration and at my deposition that the unit in Drake generating a request similar to the "first

module” in Goossens was specifically *the application 11* in Drake’s source station, not just Drake’s source station generally. For example, see paragraph 130 of my First Declaration:

130. Additionally, Drake also describes that the application in the source station (*i.e.*, first module) communicates this request to a QoS requestor manager (*i.e.*, communication manager) for subsequent transmission to a QoS allocator, as discussed further below. *See* Ex-1006 (Drake) at 3:21–28, 4:32–37, 4:62–65 (“The QoS requestor manager 12 interacts with application using internal messages as is well known in the art”), 5:5–8, Fig. 1. This is indicated by the arrows shown in Drake’s Figure 1 annotated below:



Ex-1006 (Drake) at Fig 1 (excerpted, annotated).

See also my First Declaration, at ¶¶129, 131 (same).

### **b. Elements 6[a] and 10[a]**

137. Dr. Bagherzadeh argues that the “combination of Goossen2002 and Drake fails to meet limitations 6[a] and 10[a], which recite that a ‘first module [...] issue[s] a request [] for a connection with [said] second module[] to [said] communication manager,’” because he understands the Petition and myself to rely

on “Drake’s source station 10 and QoS requester 12 therein for the claimed ‘first module’ and ‘communication manager,’” and Drake does not teach the *source station 10* issuing a request as claimed. Exhibit 2070, ¶213. This is incorrect. The combination of Goossens2002 and Drake certainly does disclose limitations 6[a] and 10[a].

138. Dr. Bagherzadeh’s opinion is based on an incorrect assumption—that I rely on Drake’s source station 10 as the claimed “first module” that makes a “service request.” As I explained above and in my First Declaration, the *Goossens* references disclose the claimed first module that is making the request. Exhibit 1003, ¶128 (“The [Goossens] IP that makes a service request is a ‘first module’”). Dr. Bagherzadeh just ignores this.

139. Also, as I explained above, when I explained that in Drake a unit that could also be considered a “first module” sends a request to the communication manager (QoS requestor manager), I explained that it was application 11 in the source station 10 that was sending the request, not just the source station 10 generally. Exhibit 1003, ¶129; *see also id.* ¶¶130, 131, 177, 195 (same). This is also what Drake’s Figure 1 shows, when it shows the Application 11 sending a request to QoS requestor 12. Dr. Bagherzadeh appears to agree with me on this point. Exhibit 2007, ¶¶214-215 (stating that the request in Drake comes from application 11); *see also* Exhibit 1037 at 53:4-18.

140. Dr. Bagherzadeh’s assumption that I rely just on Drake’s “source station 10” generally as the “first module” sending the request, and therefore the combination does not disclose the elements of 6[a] and 10[a] because Drake’s “source station 10” cannot send a request to itself, is incorrect because it ignores both the combination that I am relying on and what Drake itself teaches.

**c. Elements 6[b] and 10[b]**

141. Dr. Bagherzadeh contends that the combination of Goossens2002 and Drake fails to meet limitations 6[b] and 10[b], which recite “[said] communication manager ... forward[s] [said] request ... to [a] resource manager.” Exhibit 2070, ¶219. Dr. Bagherzadeh contends that a POSITA would have understood that “*Drake teaches that the request received by QoS allocator 20 is sent from source station 10, not from QoS requestor manager 12 therein*” (Exhibit 2070, ¶220), and also that “*if a POSITA were to understand that any particular component(s) of source station 10 send or forward the request to allocator 20, it would be transport protocol stack 15 and/or MAC entity 16, not QoS requestor manager 12*” (Exhibit 2070, ¶¶286-290, 221).

142. Neither of these assertions would make sense to a POSITA and neither is supported by Drake. As to the first—that because Drake describes the request received by the QoS allocator 20 as coming from “source station 10,” a POSITA would understand that it was *not* forwarded by the QoS requestor manager 12 *in*

source station 10—Drake shows the opposite. Drake does not disclose an alternative path, where the request somehow bypasses QoS requestor manager 12. To the contrary, Drake’s Figure 1 shows one path—the request goes through (is forwarded by) QoS requestor manager 12, and then ultimately is received by QoS allocator 20. The various lines with arrows illustrate the data path in this chain (i.e. 11↔12/13, 12/13↔15, 15↔16, 16↔17, etc.).

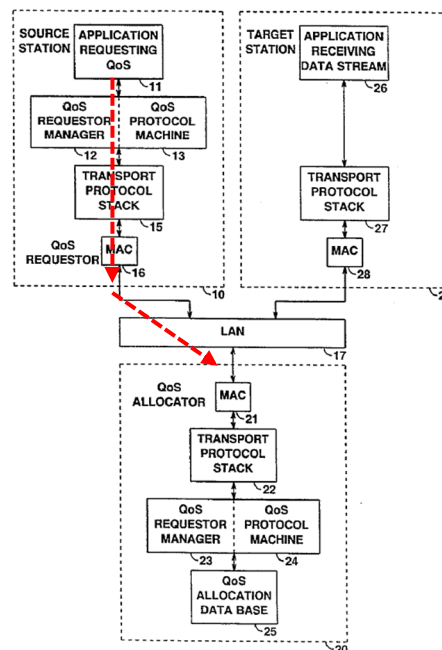


Exhibit 1006, Fig. 1 (annotated). Dr. Bagherzadeh confirms the request would follow this path from the Application Requesting QoS. Bagherzadeh Depo at 52:20-56:13. Drake also explains that it is the QoS requestor manager 12 specifically that “assemble[s] a request to reserve a QoS connection” (Exhibit 1006, 4:32–37) and that QoS requestor manager 12 “utilizes QoS protocol machine 13 to format the QoS request,” which Drake also refers to as a “QoS allocation

request” (Exhibit 1006, 4:62–65, 8:31–49, 9:52–10:8).

143. A POSITA reading Drake’s description that the request received by QoS allocator 20 comes “from source station 10” would not ignore Drake’s other teachings about where *in* source station 10 the request was forwarded from. And as I explained above, Drake’s Figure 1 shows and explains the components in source station 10, and the path that the request takes through them. Saying that the request received by QoS allocator 20 is “from” source station 10 does not mean that it was not also forwarded by QoS requestor manager in source station 10.

144. As to Dr. Bagherzadeh’s second assertion, that if a POSITA were to understand that any particular component in source station 10 was forwarding the request, it would be “transport protocol stack 15 and/or MAC entity 16, not QoS requestor manager 12” (Exhibit 2070, ¶221)—Drake does not support that either. A POSITA would understand the role of these components (protocol stack 15 and MAC entity 16) is to successfully transfer a data message that was received from QoS requestor manager 12 across the interconnect (LAN 17 in this case) to a destination. So, to the extent forwarding the request is just passing the message along, then every step in the chain in source station 10 is forwarding the message, including QoS requestor manager 12 and also then entities 15, and then 16. A POSITA would understand that for the request to ultimately arrive at QoS allocator 20 from source station 10, the QoS requestor manager 12 has to forward it.

145. To a POSITA, Drake also discloses that QoS requestor manager 12 forwards the request in a way that entities 15 and 16 do not. A POSITA would understand that application 11 wants to communicate with 26 in target station 29. However, Drake teaches that the QoS requestor manager 12 realizes that it must first send a message somewhere else: to allocator station 20. Therefore, QoS requestor manager 12 is also forwarding in the sense of redirecting communication, in that it realizes the request should be sent to 20 (not 29).

146. Finally, I note that Dr. Bagherzadeh does not identify anything in the claim or the '449 patent that requires the claimed forwarding of the request “to” a resource manager to exclude a situation like Drake, where the request is forwarded by the QoS requestor manager 12 (communication manager), and then subsequently also goes to the resource manager *through* other components. And I do not see anything. The claim simply says that the communication manager forwards the request “to [the] resource manager.” Exhibit 1001, cl. 6[a], 10[a]. That is exactly what Drake shows the QoS requestor manager 12 doing—it forwards the request to QoS allocator 20 (through entities 15 and 16).

**d. Elements 6[c] and 10[d]**

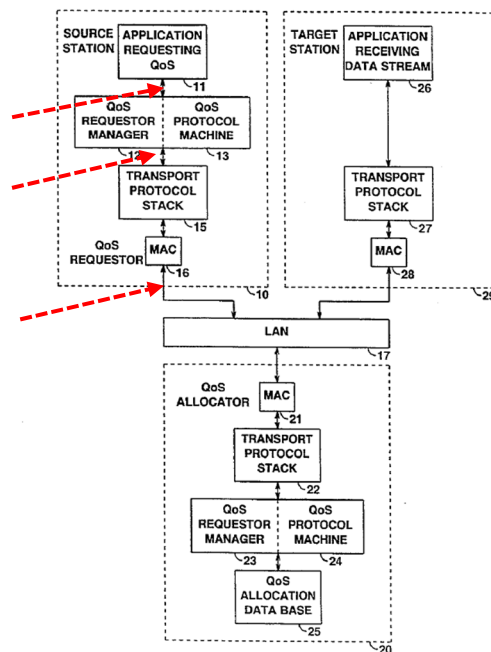
147. Dr. Bagherzadeh contends that the combination of Goossens2002 and Drake fails to meet limitations 6[c] and 10[d], which recite “[said] resource manager [] responding with the availability of the requested [target] connection to

[said] communication manager.” Exhibit 2070, ¶225. Dr. Bagherzadeh provides the opinion that a POSITA would understand that Drake “instead” teaches that QoS allocator 20 sends the QoS allocation reply to “source station 10 ... *in general*.” Exhibit 2070, ¶¶226. In doing so, Dr. Bagherzadeh cites teachings in Drake about sending the response to “source station 10.” For example, Dr. Bagherzadeh cites and quotes the following parts of Drake, at his Declaration (Exhibit 2070), ¶226: Exhibit 1006, 4:54-57 (“station 10 is notified”), 8:20-29 (describing “QoS allocation reply” as a “message[] exchanged between the stations of the [LAN]”), 14:22-27 (“AllocationReply message is ... sent from allocator station 20 ... to source station 10,” and “source station 10 ... receives the AllocationReply message”), 11:22-48 (describing that allocation acceptance/denial messages in Fig. 5 are sent to, and received by, “source station 10”), 9:52-10:2 (describing a “QoS message,” such as QoS allocation reply, as including a “source address field 52 (Source MAC Address)” for the source station).

148. I disagree that these portions of Drake show that the QoS allocation reply in Drake is sent only to the source station 10 in general, and not to QoS requestor manager 12 in the source station. This would mean that the reply message would stop at the boundary of source station 10, which is nonsensical to a POSITA. To the contrary, the portions of Drake cited by Dr. Bagherzadeh reinforce a POSITA's understanding that the reply is sent to QoS requestor manager

12 (the communication manager) in source station 10.

149. Drake does not just disclose a black box source station 10. Instead, Drake shows and describes the components in source station 10, and the data path through those components. A POSITA would understand that the only way Drake's quality of service management works is for the reply regarding the QoS request to get back to application 11 in source station 10—the entity that originally made the QoS request. Dr. Bagherzadeh agreed this was the case. Exhibit 2070 at 59:22-60:5. And Drake shows that to get back to application 11, the reply is first sent to QoS requestor manager 12, which Drake teaches is the interface with application 11. *E.g.*, Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). This is also clearly shown in Drake's Figure 1 (double-headed arrows for the data path, indicating to a POSITA that the reply comes back through the same path as the request):



150. Exhibit 1006, Fig. 1 (annotated). A POSITA would understand that Drake's quality of service system would not work if the reply stopped at the boundary of source station 10. Instead, a POSITA would understand that Drake's teaching of sending a reply "to source station 10" is teaching responding with availability of the connection to the communication manager (QoS requestor manager) as claimed.

151. Dr. Bagherzadeh also contends that to the extent the QoS allocation reply is sent "to an internal component of source station 10," a POSITA would understand that it was sent "to application 11, not QoS requestor manager 12," because "application 11 is the original requestor and is the component that [also] sends a multimedia file via the established connection." Exhibit 2070, ¶227. To a POSITA, this is plainly wrong, and Drake does not support this understanding. I

also note that Dr. Bagherzadeh agreed that the message does go from QoS allocator to the application requesting QoS using the same path described above. Exhibit 1037 at 59:22-60:8.

152. As I explained above Figure 1 shows the data path for communication (the lines with arrows at each end), and this indicates to a POSITA that the reply message goes back to component 12 before anything goes back to component 11. Exhibit 1006, Fig. 1. And Drake also expressly explains that QoS requestor manager 12 is the thing that interfaces with application 11—a POSITA would understand this is true for messages going in both directions. *See* Exhibit 1006, 4:63-65 (“QoS requestor manager 12 interacts with application 11”). Moreover, there is no teaching in Drake of an alternative reply message route that bypasses the QoS requestor manager 12. Dr. Bagherzadeh states that the reply “could have been sent to application 11 or some component not depicted in Drake’s Figure 1” (Exhibit 2070, ¶227), but that is not what Drake shows. Drake shows the reply message traveling back through QoS requestor manager 12 and QoS Protocol machine 13. *See* Exhibit 1037 at 60:3-61:13.

153. Not only would a POSITA understand Drake to be teaching (and showing) that the QoS allocation reply goes back to the QoS requestor manager 12, that would also be the way a POSITA would have understood Drake works. A POSITA would know that for Drake’s QoS system to function, the QoS allocation

reply message would need to get back to the application 11 that made the original quality of service request (and that is the same component that then sends a multimedia file over the established link). Dr. Bagherzadeh does not appear to disagree with this. See for example, Exhibit 2070, ¶227 (acknowledging that if the QoS allocation reply is sent to an internal component of source station 10, it “would be sent to application 11”). And any POSITA of common sense would realize that the QoS allocation reply sent “to station 10” (as Drake expressly discloses, e.g., Exhibit 1006, 4:54-57, 8:20-29, 14:22-27, 11:22-48, and 9:52-10:2) means that the reply is sent to internal components of station 10—it would not just stop at the boundary of station 10.

**e. Elements 6[d] and 10[e]**

154. As I explained in my First Declaration, Goossens2002 discloses a first and second module that a connection is established between. Dr. Bagherzadeh does not appear to dispute this.

**D. The Patent Owner’s Brief Mischaracterizes My Testimony**

155. The Patent Owner’s brief, at pages 31-32 of Paper 35, mischaracterizes my testimony regarding implementing Drake’s communication manager in the Goossens systems.

156. I actually testified that Drake shows its QoS requestor manager (*communication manager*) “in the network interface” and “[t]hat would be the

natural place to put it” in Goossens’ system. Exhibit 2078, 203:24-204:9 (emphasis added, see also the green highlighted text below). Then, as the testimony shows, I thought that I had misunderstood the question and that Patent Owner’s attorney was instead asking about Drake’s *resource* manager (not Drake’s communication manager), and so I changed my response to start talking about Drake’s *resource manager* (blue highlighted text):

23 BY MR. CHRISTOPHER:  
24 Q. You have not explicitly defined in paragraph  
25 122 where Drake's QoS requestor manager would be

---

Page 204

1 implemented as a circuit on the integrated circuit chip  
2 in Goossens2002?  
3 MR. KOHLHEPP: Object to form.  
4 THE WITNESS: Well, there are -- I'm referring  
5 to the teachings of Drake, and there would be several --  
6 I mean, there would be several options for where this  
7 might be implemented. It's shown here in the network  
8 interface. That would be a natural place to put it. I  
9 believe that the '449 patent mentions that the -- that  
10 this could be either -- well, sorry. Sorry. Resource  
11 manager.  
12 This could be either -- I believe the '449  
13 patent discusses that this could either be distributed  
14 or centralized. And those are -- are all options. You  
15 just need the -- the concepts here, that are described  
16 in Drake. So there's some flexibility in terms of how  
17 this is implemented. But a POSITA would understand  
18 this.

157. For the subsequent deposition testimony (that Patent Owner cites at pages 31-32 of its brief), I was talking about the *resource manager*. This is clear not only from my testimony above, but also because I repeatedly referred to the “resource manager” in my answers in the testimony the Patent Owner’s brief cites—see Exhibit 2076, 205:14-209:13:

14 Q. Okay. Would one option be to implement Drake's  
15 QoS requestor manager in Goossens2002's IP block?

16 A. Well, that's not the most natural choice. It  
17 would probably be separate from the IP blocks. I don't  
18 think that putting it in the IP block would be the most  
19 natural choice.

20 Q. Can you think of any reason why a person of  
21 skill in the art would put the QoS requestor manager  
22 from Drake in the Goossens2002 IP block?

23 A. I mean, as I mentioned, I believe the '449  
24 patent says that the resource manager could be  
25 implemented either in a centralized or distributed

Page 206

1 manner. If it's centralized, then it wouldn't be in a  
2 particular IP block unless you count being your own IP  
3 block as an option. I mean, effectively that's what it  
4 would be. But in terms of the existing IP blocks, if it  
5 was distributed, the natural place for it to be is in  
6 the network interface, or near the network interface.  
7 So there are always trade-offs with whether you put  
8 things in IP blocks or not. It -- it's -- like I said,  
9 I don't think it's the most natural choice. But I don't  
10 remember shaping a specific opinion on that. If I did  
11 in my report, I'm happy to talk about that. But I don't  
12 remember discussing that exact issue.

13 Q. So I think you've testified that one option  
14 would be to implement Drake's QoS requestor manager in  
15 the network interface of Goossens2002. Is that right?

16 A. Well, I believe that the -- I believe that the  
17 '449 patent mentioned that it could be -- the resource  
18 manager could be implemented either centralized or  
19 distributed. If it was distributed, then it could be  
20 done in the -- you could -- it could be that there are  
21 parts of that, at least, that are in the network  
22 interface, although, you know, it would have to be  
23 coordinating somehow with the other distributed  
24 components of the resource manager.

25 It's easy to think about doing it as a

1 centralized manager, because it needs to have a -- a

2 global view of what's happening with the resources.

3 It's easier to implement that if you don't need

4 distributed communication.

5 **Q. All right. Just so I get a clear answer to**  
 6 **this, somebody skilled in the art could implement the**  
 7 **Drake QoS requestor manager in the Goossens network**  
 8 **interface. That's possible, right?**

9 MR. KOHLHEPP: Object to form.

10 THE WITNESS: There are -- I mean, the resource

11 manager is -- there are requirements for what the

12 resource manager does, as described in the '449 patent.

13 And the functionality for that could be distributed or

14 it could be centralized.

15 It -- it doesn't make sense for it to exist --

16 I mean, if it's distributed, then it introduces some

17 more complications. You would need to coordinate across

18 the different distributed nodes. It's -- it's easier to

19 think of it if it's centralized, because it eliminates

20 the communication problem of just understanding the

21 global allocation of all the resources. But there --

22 but there are multiple ways to implement this.

23 BY MR. CHRISTOPHER:

24 **Q. Okay. Can you list for me in your obviousness**  
 25 **combination all of the potential places that the Drake**

1 **QoS requestor manager could be implemented in**

2 **Goossens2002?**

3 MR. KOHLHEPP: Object to form.

4 THE WITNESS: Well, my understanding is that

5 when doing obvious analysis, I don't need to show the

6 universe of possibilities, but, rather, just show that

7 there is a possibility that matches the patent claims.

8 And one option that would, would be to have a

9 centralized implementation of the resource manager,

10 similar -- very much like what is shown in figure 1 from

11 Drake. And that would be a straightforward thing for an

12 engineer to implement. And the -- the teachings of

13 Drake would show that as an option. As I said, I

14 believe that you could -- rather than having just a

15 simply centralized version of it, you could

16 decentralize -- you could distribute it. But it's

17 easiest to think of it as just a centralized piece of

18 hardware.

19 BY MR. CHRISTOPHER:

20 **Q. In the centralized implementation that you just**  
 21 **described, tell me specifically where in Goossens2002**  
 22 **that the QoS requestor manager would be implemented.**

23 A. On the chip somewhere. It doesn't really

24 matter where it is. You just need to be able to

25 communicate with it.

## V. ACKNOWLEDGMENT

158. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the full knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States code.

Dated: January 27, 2025

  
\_\_\_\_\_  
Dr. Todd Mowry