

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent of: Casimir M. Kaczmarczyk, et al.  
U.S. Patent No.: 6,950,441 Attorney Docket No.: 42360-0004IP3  
Issue Date: September 27, 2005  
Appl. Serial No.: 09/538,040  
Filing Date: March 29, 2000  
Title: METHOD AND SYSTEM TO INTERNETWORK  
TELECOMMUNICATION NETWORKS OF DIFFERENT  
PROTOCOLS

**Mail Stop Patent Board**

Patent Trial and Appeal Board  
U.S. Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450

**PETITION FOR *INTER PARTES* REVIEW OF UNITED STATES PATENT  
NO. 6,950,441 PURSUANT TO 35 U.S.C. §§ 311–319, 37 C.F.R. § 42**

## TABLE OF CONTENTS

I.	REQUIREMENTS FOR IPR UNDER 37 C.F.R. § 42.104.....	1
A.	Grounds for Standing Under 37 C.F.R. § 42.104(a).....	1
B.	Challenge Under 37 C.F.R. § 42.104(b) and Relief Requested .....	1
II.	LEVEL OF ORDINARY SKILL IN THE ART .....	3
III.	SUMMARY OF THE ‘441 PATENT .....	3
A.	Brief Description.....	3
B.	Claim Construction under 37 C.F.R. §§ 42.104(b)(3).....	4
C.	Summary of the Prosecution History of the ‘441 Patent.....	5
IV.	APPLICATION OF PRIOR ART TO CHALLENGED CLAIMS .....	6
A.	GROUND 1A – Claims 1, 3, 5-9, and 12-17 are obvious in view of Shankar and Huang.....	6
1.	Shankar .....	6
2.	Huang .....	11
3.	Combination of Shankar and Huang .....	12
B.	GROUND 1B – Claim 2 is obvious over Shankar in view of Huang, and Capers .....	47
1.	Capers .....	47
2.	Combination of Shankar, Huang, and Capers.....	50
3.	Application of Shankar, Huang, and Capers to Challenged Claims .....	54
C.	GROUND 1C – Claims 18-25, and 28-32 are obvious over Shankar in view of Huang, Capers, and Janning .....	54
1.	Janning.....	54
2.	Combination of Shankar, Huang, Capers, and Janning .....	55
D.	GROUND 1D –Claims 4 and 10 are obvious in view of Shankar, Huang, and Huitema. ....	65
1.	Huitema .....	65
2.	Combination of Shankar, Huang, and Huitema .....	67
3.	Application of Shankar, Huang, and Huitema to the Challenged Claims.....	71
E.	GROUND 1E–Claim 26 is obvious in view of Shankar, Huang, Capers, Janning, and Huitema. ....	73
1.	Combination of Shankar, Huang, Capers, Janning, and Huitema	73
V.	CONCLUSION.....	73
VI.	PAYMENT OF FEES – 37 C.F.R. § 42.103.....	74

VII.	MANDATORY NOTICES UNDER 37 C.F.R § 42.8(a)(1).....	74
A.	Real Party-In-Interest Under 37 C.F.R. § 42.8(b)(1).....	74
B.	Related Matters Under 37 C.F.R. § 42.8(b)(2) .....	74
C.	Lead And Back-Up Counsel Under 37 C.F.R. § 42.8(b)(3).....	74
D.	Service Information .....	75

## EXHIBITS

METASWITCH-1001	U.S. Patent No. 6,950,441 to Kaczmarczyk, et al. (“the ‘441 patent”)
METASWITCH-1002	Excerpts from the Prosecution History of the ‘441 Patent (“Prosecution History”)
METASWITCH-1003	Declaration of Dr. Vishal Sharma (“Sharma”)
METASWITCH-1004	U.S. Patent No. 6,570,869 to Shankar et al. (“Shankar”)
METASWITCH-1005	U.S. Patent No. 7,336,649 to Huang (“Huang”)
METASWITCH-1006	U.S. Patent No. 6,614,781 to Elliott et al. (“Elliott”)
METASWITCH-1007	U.S. Patent No. 6,111,893 to Volftsun et al. (“Volftsun”)
METASWITCH-1008	Christian Huitema, An Architecture For Residential Internet Telephony Service, IEEE Internet Computing, June 1999(“Huitema”)
METASWITCH-1009	RESERVED
METASWITCH-1010	U.S. Patent No. 6,418,205 to Capers et al. (“Capers”)
METASWITCH-1011	U.S. Patent No. 6,052,448 to Janning (“Janning”)
METASWITCH-1012	U.S. Provisional Application 60/127,169 (“’169 Provisional”)
METASWITCH-1013	Declaration of Gerard P. Grenier re Huitema
METASWITCH-1014	Declaration of Gerard P. Grenier re Sestak
METASWITCH-1015	RESERVED

- METASWITCH-1016 U.S. Patent No. 6,249,572 (“Brockman”)
- METASWITCH-1017 M. Sestak et al., Traffic Monitoring in a Network of PBXs, GLOBECOM ’90: IEEE Global Telecommunications Conference and Exhibition (“Sestak”)
- METASWITCH-1018 Media Gateway Control Protocol (MGCP), V1.0, The Internet Society (October, 1999) (“MGCP Specification”)

Metaswitch Networks Ltd. and Metaswitch Networks Corporation  
(collectively “Petitioner” or “Metaswitch”) petition for Inter Partes Review  
 (“IPR”) under 35 U.S.C. §§ 311–319 and 37 C.F.R. § 42 of claims 1-10, 12-26, 28-  
32 (“the Challenged Claims”) of U.S. Patent No. 6,950,441 (“the ‘441 patent”).

The Challenged Claims are unpatentable based on the references presented  
in this petition, and there exists a reasonable likelihood that Petitioner will prevail  
in proving unpatentability of the Challenged Claims.

Petitioner respectfully submits that an IPR should be instituted, and the  
Challenged Claims canceled as unpatentable.

**I. REQUIREMENTS FOR IPR UNDER 37 C.F.R. § 42.104**

**A. Grounds for Standing Under 37 C.F.R. § 42.104(a)**

Metaswitch certifies that the ‘441 Patent is available for IPR. The present  
petition is being filed within one year of service of a complaint against Metaswitch  
in 2:18-cv-00058 (E.D. Texas). Petitioner is not barred or estopped from  
requesting this review.

**B. Challenge Under 37 C.F.R. § 42.104(b) and Relief  
Requested**

Petitioner requests IPR on the grounds set forth below. Detail on how the  
claims are construed and where each element can be found in the cited prior art is  
provided herein. Additional support for each ground is provided in EX1003, the  
Declaration of Dr. Vishal Sharma (“Sharma”). Sharma, ¶12 *et seq.*

Ground	'441 Patent Claims	Basis for Rejection
Ground 1A	1, 3, 5-9, 12-17	103 - Shankar, Huang
Ground 1B	2	103 - Shankar, Huang, Capers
Ground 1C	18-25, 28-32	103 - Shankar, Huang, Capers, Janning
Ground 1D	4, 10	103 - Shankar, Huang, Huitema
Ground 1E	26	103 - Shankar, Huang, Capers, Janning, Huitema

The '441 patent issued from U.S. Application No. 09/538,040, filed March 29, 2000. The claims of the '441 patent all recite various features related to a “signaling agent” and “call agent,” and/or features related to a “network directory server,” “network gateway,” “network gateway agent,” “resource manager,” and specific configurations of these components, for example, that are not supported under 35 U.S.C. §112, or even mentioned, in provisional application no. 60/127,169 (“’169 provisional”). Thus, the claims of the '441 patent are entitled to an effective filing date no earlier than March 29, 2000. Sharma, ¶18.

Reference	Date	Section
Shankar(EX1004)	Sep. 30, 1998(filed)	§102(e)
Huang(EX1005)	Dec. 20, 1995(filed)	§102(e)
Volftsun(EX1007)	July 31, 1997(filed)	§102(e)
Huitema(EX1008)	June 1999(published)	§102(a)

Reference	Date	Section
Capers(EX1010)	May 7, 1998(filed)	§102(e)
Janning(EX1011)	Dec. 23, 1997(filed)	§102(e)

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

A person of ordinary skill in the art at the time of the alleged invention of the '441 Patent (a “POSITA”) would have a Bachelor’s or Master’s degree in computer science, computer engineering, electrical engineering, or the industry equivalent thereof, would have an understanding of software and three to five years’ experience with telecommunications systems, including circuit-switched and packet-switched networks, or equivalent qualifications through some combination of education, training, and experience. Additional education or work experience in a relevant field may compensate for a deficit in one of the other aspects of the requirements stated above. Sharma, ¶¶13-14.

## **III. SUMMARY OF THE ‘441 PATENT**

### **A. Brief Description**

The ‘441 patent relates generally to a device—referred to as a “softswitch”—interconnecting multiple networks. Sharma, ¶¶15-17. The softswitch is described as having six main “functional components,” including a “call agent” and a “signaling agent” that communicates with the “call agent” and a



network. '441 patent, 6:11-14. The “role of the signaling agent 138 is to receive any type of access protocol, communicate with call agent 140, and to terminate the call to the appropriate network using the applicable protocol as instructed by call agent 140,” and the “[c]all agent 140 is responsible for managing the end-to-end control of the sessions or calls between packet-switched elements.” *Id.*, 6:18-20. The softswitch may also include a “network directory server” for “managing user profiles, such as access authorization, call barring, dial plan, etc., and providing routing instruction for local and external (local number portability, 1-800) calls,” a “network gateway” for “interconnecting the softswitch network to an external network,” and a “resource manager” to “validate the incoming circuit, and to provide resource availability information for the outgoing call from the media switch.” *Id.*, 7:26-8:14. Notably, the '441 patent explicitly indicates that the network directory server “may reside on the same platform as call agent 140 or may be distributed.” *Id.*, 7:48-50; 8:48-51 (“Network directory service 200 accesses in-memory tables 202, which store user profiles, routing instructions, addresses, and subscriber services. Network directory service 200 further accesses external data servers 203.”).

**B. Claim Construction under 37 C.F.R. §§ 42.104(b)(3)**

Petitioner submits that all terms should be given their plain meaning, per BRI or Phillips, but reserves the right to respond to any constructions later offered

by Patent Owner or adopted by the Board. Sharma, ¶21. Petitioner provides the following specific constructions for clarity, and further details of how the claims are interpreted are discussed in the relevant sections below. No other terms need to be construed in order to resolve patentability over the prior art herein.

**“logic control” (5, 9, 16, 21, 25, 32)** – Consistent with the ’441 specification, this term refers to the “data configurable generic state machine processor that execute logic control programs.” ’441 Patent, 10:34-36; Sharma, ¶22.

**“[logic control] program” (5, 9, 16, 21, 25, 32)** – Consistent with the ’441 specification, this term refers to actions, transitions, and data as a series of data items or independent building blocks that define what a logic engine application is to perform. ’441 Patent, 10:42-45; Sharma, ¶23.

Petitioner’s comments on claim construction are solely for determining whether the prior art renders obvious the claims in this proceeding. Petitioner does not concede that any of the claims meet statutory standards.

### **C. Summary of the Prosecution History of the ’441 Patent**

The ’441 patent issued on September 27, 2005 from U.S. Patent Application No. 09/538,040 (“the ’040 application”), filed on March 29, 2000 with 51 claims. *See* Prosecution History. The record highlights several errors underpinning the Examiner’s conclusion that directly contradicted both the specification of the ’441

Patent and the description of the applied art.

For example, Elliott was applied in two office actions, but was applied incorrectly. The Examiner's conclusion relied on a legally erroneous claim interpretation that excluded a distributed arrangement of the network directory server and other components of the softswitch, even though the '441 Patent explicitly described such arrangements. Prosecution History, pp.191-195; '441 Patent, 7:48-50; 8:48-51; Sharma, ¶19. Additionally, the Examiner overlooked Elliott's teachings regarding co-located features that disclosed even the Examiner's narrow and legally erroneous interpretation of the claims. Elliott, 31:54-58; 49:12-18; 62:60-64:67; Sharma, ¶20. Likewise, regarding claim 9 (claim 10 during prosecution), the reasons for allowance erred by citing language that was not included in claim 9. Prosecution History, p. 193. In short, Elliott was not properly considered during original prosecution, and the claims would not have been allowed had they been interpreted consistent with the '441 specification and considered in view of Elliott's full disclosure.

#### **IV. APPLICATION OF PRIOR ART TO CHALLENGED CLAIMS**

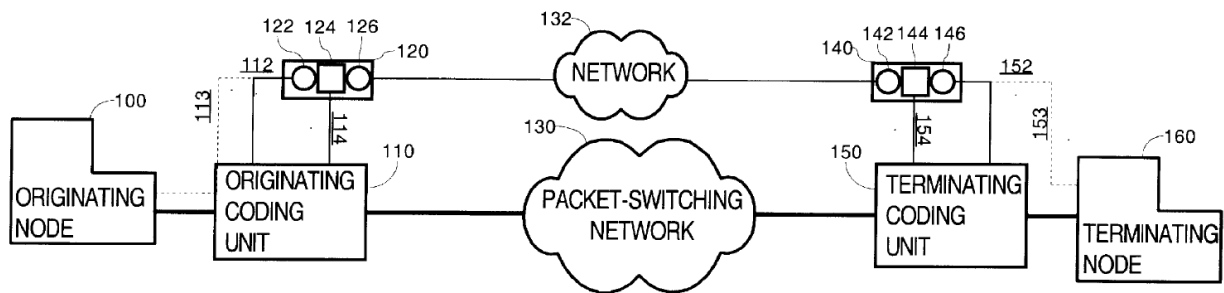
##### **A. GROUND 1A – Claims 1, 3, 5-9, and 12-17 are obvious in view of Shankar and Huang**

###### **1. Shankar**

Shankar describes a telecommunications system for communicating voice from a traditional telephone, or other communication device, over a packet-

switched network. Shankar, Abstract. Shankar's system utilizes one or more "signaling units" that "implement a 'virtual switch' and are responsible for processing and routing the signaling messages that are exchanged to set up and tear down a voice connection," and one or more "coding units" that "convert bearer voice traffic between legacy and packet formats." Shankar, 2:44-55; 5:14-32; *see also* Abstract; 1:44-48; 2:44-55; Sharma, ¶26. The signaling and coding units interconnect public branch exchanges, telephone switches, telephones, wireless public branch exchanges, or legacy telecommunications systems, for example, over the packet-switched network. Shankar, 3:65-4:3.

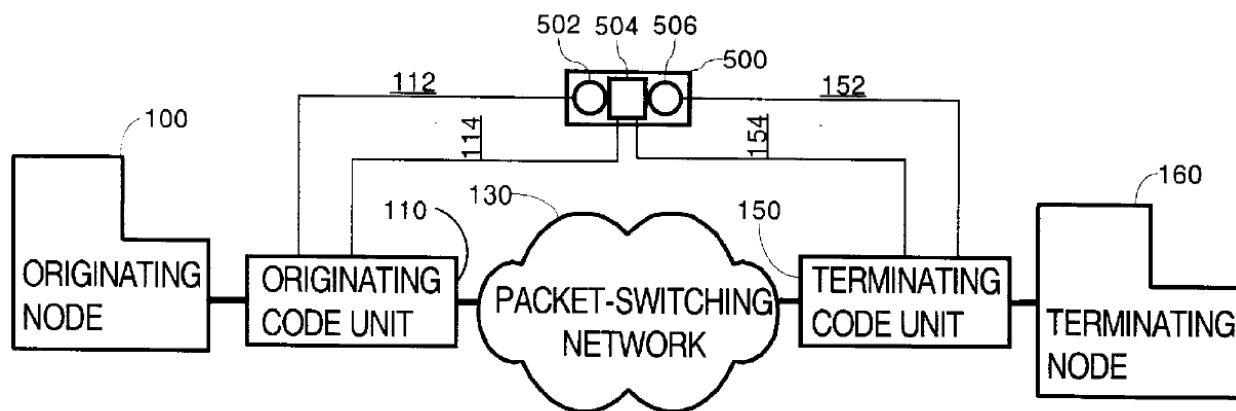
The system may be configured to include two "signaling units" and/or two "coding units" (e.g., originating and terminating), or may include a single "signaling unit" and/or single "coding unit." *See, e.g.* FIGS. 1, 5A-5C; Sharma, ¶27.



**FIG. 1**

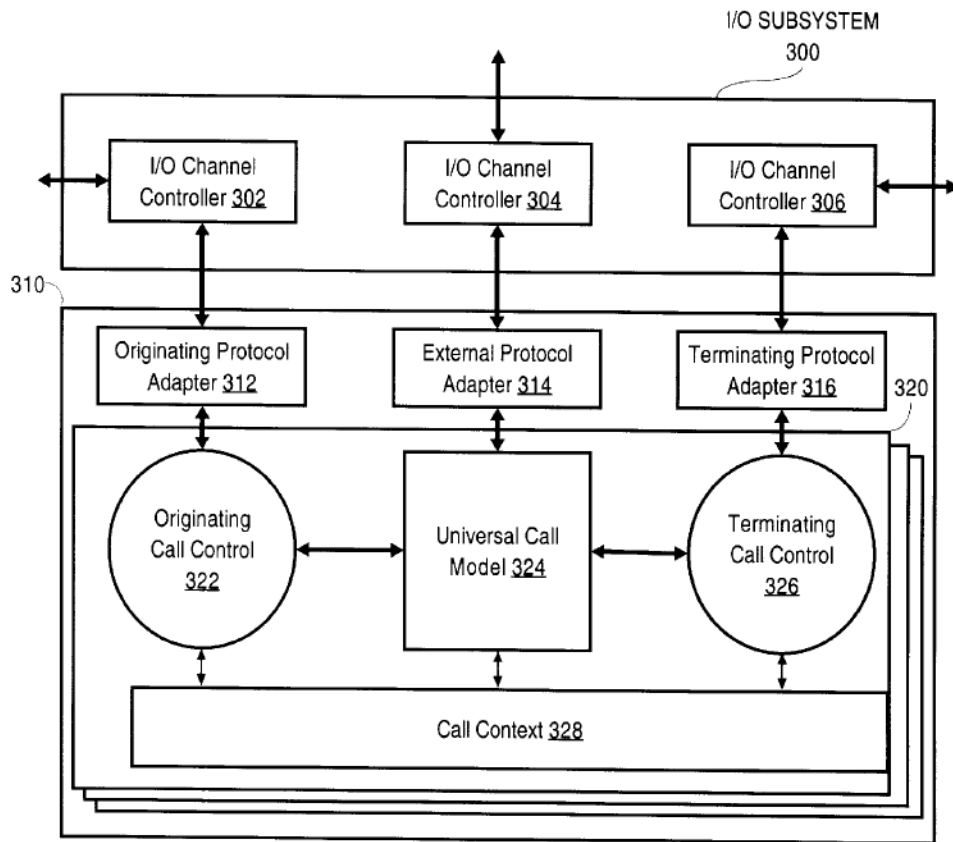
For example, Shankar illustrates an example system including an “originating signaling unit 120” that receives call signaling data, obtains the network address of the originating coding unit 110 that received the call, determines which “terminating signaling unit 140” should receive the call, and transmits a signaling message “including information for establishing the voice and the network address of the originating coding unit 110” to the “terminating signaling unit 140.” Shankar, 5:31-52; 12:65-14:18. The terminating signaling unit 140 “determines which bearer should receive the call, and controls the terminating coding unit 150 to establish a bearer channel through the packet-switching network. *Id.*, 5:53-63; 14:18-15:49. “In this manner, a voice call that is originated from a legacy system 100 or terminated by a legacy system 160 is seamlessly established over the packet-switching network 130 without having to upgrade or replace the legacy systems.” *Id.*; Sharma, ¶28.

Shankar also describes examples in which “a single signaling unit 500 handles the voice call signaling processing.” Shankar, 4:55-65; 15:50-16:13; Sharma, ¶29.



**FIG. 5(a)**

Shankar's "signaling unit" includes an "originating call control (OCC)," "universal call model (UCM)," and "terminating call control (TCC)." Shankar, 6:17-67; 8:57-12:64; 12:65-16:13. The OCC "models a call from the perspective of the originating protocol," including receiving messages in the originating protocol (from originating protocol adapter 312) and transitioning from one state to another state to output a non-protocol specific (i.e., universal protocol specific) message to UCM 324." Shankar, 11:16-22; Sharma, ¶30. The "TCC 326 receives non-protocol specific messages from UCM 324," and outputs terminating protocol specific messages to terminating protocol adapter 312, which are in turn routed to the "corresponding I/O channel controller 306, to a network node, and ultimately to the destination telephone." Shankar, 10:6-30.



**FIG. 3**

The UCM “manages the call.” Shankar, 11:38-52; Sharma, ¶31. For example, the UCM “handles calls in the converted universal protocol, arranges for messages to be routed ultimately to the other protocol converter, and controls the originating coding unit 110 over a control link 114.” Shankar, 6:29-33.

Shankar incorporates the “entire contents” of Volftsun by reference, which describes an example of a protocol converter, including its internal workings. Shankar, 6:53-67. Volftsun specifically describes additional details regarding the

structure and operation of its I/O subsystem, protocol adapters, OCC, UCM, and TCC, and communication between the respective components, which are discussed generally by Shankar. *See, e.g.* Volftsun, FIGS. 1-8b; Sharma, ¶32. As Volftsun is incorporated in Shankar by reference, it is treated with Shankar as a common document.

## **2. Huang**

Huang describes a hybrid system that, like Shankar, connects networks of different types, including packet-switched (e.g., IP) and circuit-switched (e.g., PSTN) networks. A “gateway computer (GC) or equivalent interconnects the packet-switched network to each of the circuit-switched networks, and converts voice signals into data packets and vice versa, and resolves the call destinations while routing the packets.” Huang, 2:24-34. By doing so, Huang describes that a user can benefit from the advantages of a packet-switched network while placing and receiving calls using traditional telephone equipment. Huang, 1:11-2:20; 2:53-67; Sharma, ¶33.

Each of Huang’s gateway computers have several “functional components,” including “media conversion modules 31,” “speech processing modules 48,” “routing resolution modules 39,” and “special services modules 38.” Huang, 3:49-4:36; FIGS. 3a-3b. “Routing modules 39 include an address resolution logic 45 implemented with CPU 21, and a network routing database also implemented with



CPU 21 and possibly shared with other GC's in a distributed manner.” *Id.*, 4:5-9;  
*see also id.*, claim 6 (“said routing components comprises an address resolution  
logic and a network routing database”).

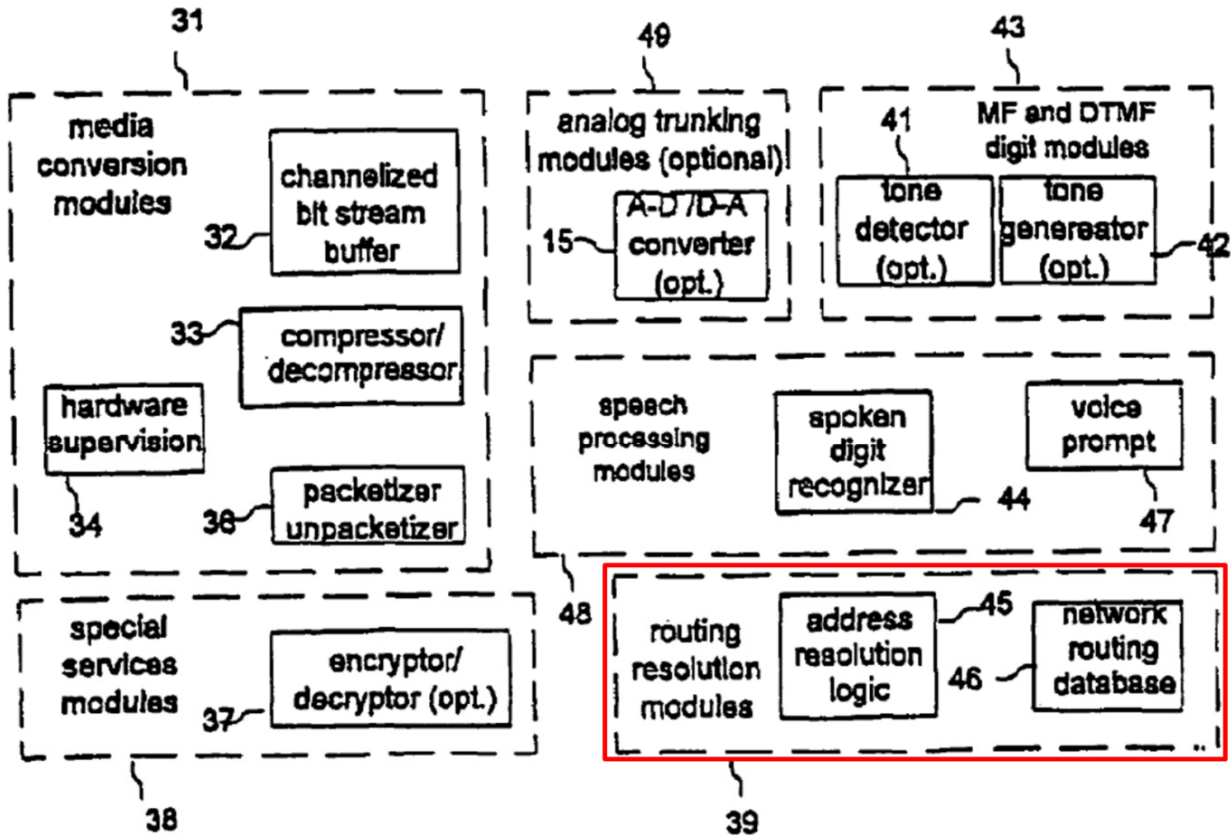


Fig. 3b

Sharma, ¶¶34-36; Huang, FIG. 3b (annotated).

### 3. Combination of Shankar and Huang

Shankar describes that the UCM of its signaling unit “resolves the call routing through the network 132,” and selects bearer channels to be used, indicating that the UCM receives routing information to make routing determinations. A POSITA would have found it obvious, based on Huang’s

teachings, that routing would have been effectively resolved by integrating features of a routing resolution module. Sharma, ¶37. Through this integration, a POSITA would have reasonably expected that Shankar's UCM (e.g., call agent), would have effectively requested routing information from the routing resolution module such that the route could be resolved through the IP network, to an appropriate network node, and/or destination telephone. *Id.* Indeed, dialed digits, for example, must be correlated to the address of the network node/destination telephone in order to execute the call. *Id.*

Additionally, Huang's routing resolution module would have enhanced Shankar's system by providing address resolution logic to facilitate resolution of an address, such as when processing a call between networks (e.g., between circuit-switched and packet-switched networks). Accordingly, based on Huang's disclosure, a POSITA would have been led to integrate Huang's address resolution modules in order to efficiently and effectively route calls, amplifying the advantages obtained by interconnecting traditional telephone equipment with packet-switched networks. Sharma, ¶38.

Ultimately, a POSITA would have been prompted to implement a routing resolution module suggested by Huang, and would have had a reasonable expectation of success, because doing so would have been merely the application of a known technique (e.g., using a routing resolution module to provide routing

information to facilitate routing between different networks) to a known system (e.g., Shankar's system that routes calls between different networks) ready for improvement to yield predictable results. Indeed, "when a patent 'simply arranges old elements with each performing the same function it had been known to perform' and yields no more than one would expect from such an arrangement, the combination is obvious." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417. Both Shankar and Huang describe functional modules that request information to resolve a route to process a call between different networks, and a POSITA would have found it obvious that applying Huang's routing resolution module would have effectively provided Shankar's UCM with routing information needed to resolve the route. Sharma, ¶39.

***[1.pre, 3.pre, 5.pre] A softswitch interconnecting networks of different transport protocols, comprising:***

If the preamble is limiting, it is disclosed by Shankar in view of Huang. Sharma, ¶40. For example, Shankar describes a softswitch provided by a "protocol converter" that implements a "signaling unit." The "signaling units" acts as a "virtual switch" and "are responsible for processing and routing the signaling messages that are exchanged to set up and tear down a voice connection." Shankar, 5:14-32; 5:65-6:67. Shankar describes that "the signaling units perform such functions as call resolution, call routing, bearer selection, and generation of call detail records (CDRs) for billing management." *Id.*; *see also* Abstract; 1:44-48;

2:44-55; 15:64-16:7 (describing a configuration “wherein a single signaling unit 500 handles the voice call signaling processing for both the originating coding unit 110 and the terminating coding unit 150.”).

Moreover, Shankar describes its “signaling apparatus implements signaling interworking and protocol conversion, if necessary, between the legacy systems and the packet-switching network,” and that the “coding units convert bearer voice traffic between legacy and packet formats,” further indicating that Shankar’s system interconnects networks of different protocols. Shankar, 2:44-55; 14:4-18; ’441 Patent, 2:4-9 (“The softswitch of the present invention allows service providers to interconnect intelligent control and signaling systems between networks.”); Sharma, ¶41.

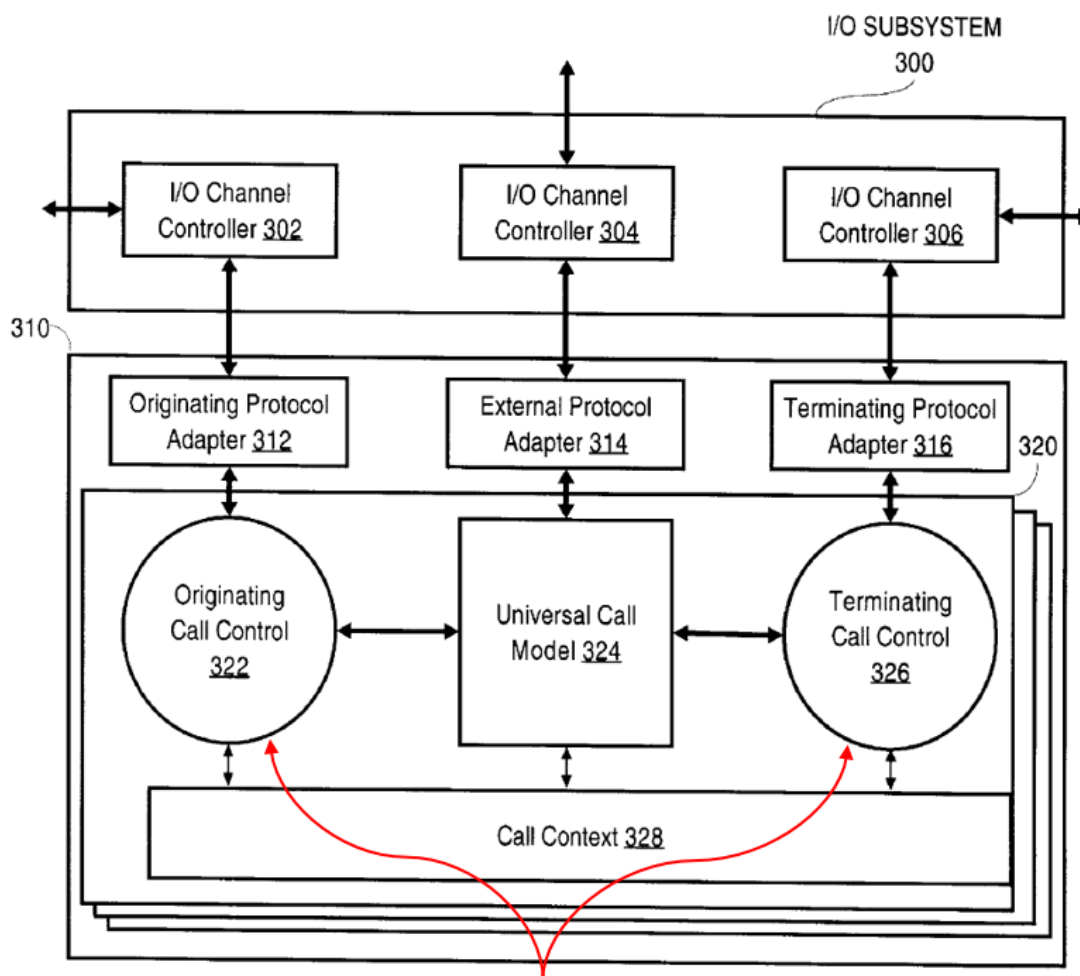
***[1.a, 3.a, 5.a,] a signaling agent coupled to the networks and operable to receive incoming signaling messages, translate the incoming signaling messages to a call model event, and route the call model event; and***

The combination discloses a signaling agent coupled to the networks as claimed. For example, Shankar describes a signaling agent provided by at least portions of the “originating call control (OCC)” and “terminating call control (TCC)” of its “signaling unit.” Shankar, 6:17-67; 8:57-12:64; 12:65-16:13; FIGS. 1, 5a-c; Volftsun, 9:22-39; FIG. 4. The ’441 patent describes “[t]he role of signaling agent 138 is to receive any type of access protocol, communicate with call agent 140, and to terminate the call to the appropriate network using the

applicable protocol as instructed by call agent 140.” ’441 Patent, 6:16-20.

Shankar’s “originating call control (OCC)” likewise is operable to receive incoming signaling messages and translate the incoming signaling messages to a call model event. Sharma, ¶42. Indeed, Shankar explicitly teaches that the “OCC 322 **models a call** from the perspective of the originating protocol.” Shankar, 11:16-22(emphasis added). The “OCC 322 **receives messages in the originating protocol** from originating protocol adapter 312 and, in response, transitions from one state to another state, resulting in **outputting a non-protocol specific (i.e., universal protocol specific) message to UCM 324.**” Shankar, 11:16-22(emphasis added); Volftsun, FIG. 7. In other words, the information in the non-protocol specific message corresponds to an event that is output in connection with modeling of the call by the OCC, and thus the information corresponds to a call model event. Moreover, the information is used by the UCM, which also models the call, further indicating that the OCC routes the call model event. Sharma, ¶42.

Shankar’s “TCC” is operable to route a call model event in a second manner by routing to a terminating destination. “The terminating protocol adapter 316 can route messages from associated call instances to the corresponding I/O channel controller 306, to a network node, and ultimately to the destination telephone.” Shankar, 10:6-25. Such routing by the “TCC” independently discloses that the signaling unit is “operable to route the call model event,” as claimed. Sharma, ¶43.



Portions of Shankar's originating call control and terminating call control that are operable to receive incoming signaling messages, translate the incoming signaling messages to a call model event, and route the call model event provide a signaling agent

Sharma, ¶43; Shankar, FIG. 3 (annotated); 8:58-61.

Shankar's signaling agent may also be considered to include the signaling unit's "protocol adapters" and corresponding "I/O channel controllers" in the "I/O subsystem," for example, which are involved in receiving and outputting signaling. *See, e.g.* Shankar, 13:12-14:3; Volftsun, FIG. 4; Sharma, ¶44.

As discussed above, the '441 Patent describes the "signaling agent" is itself

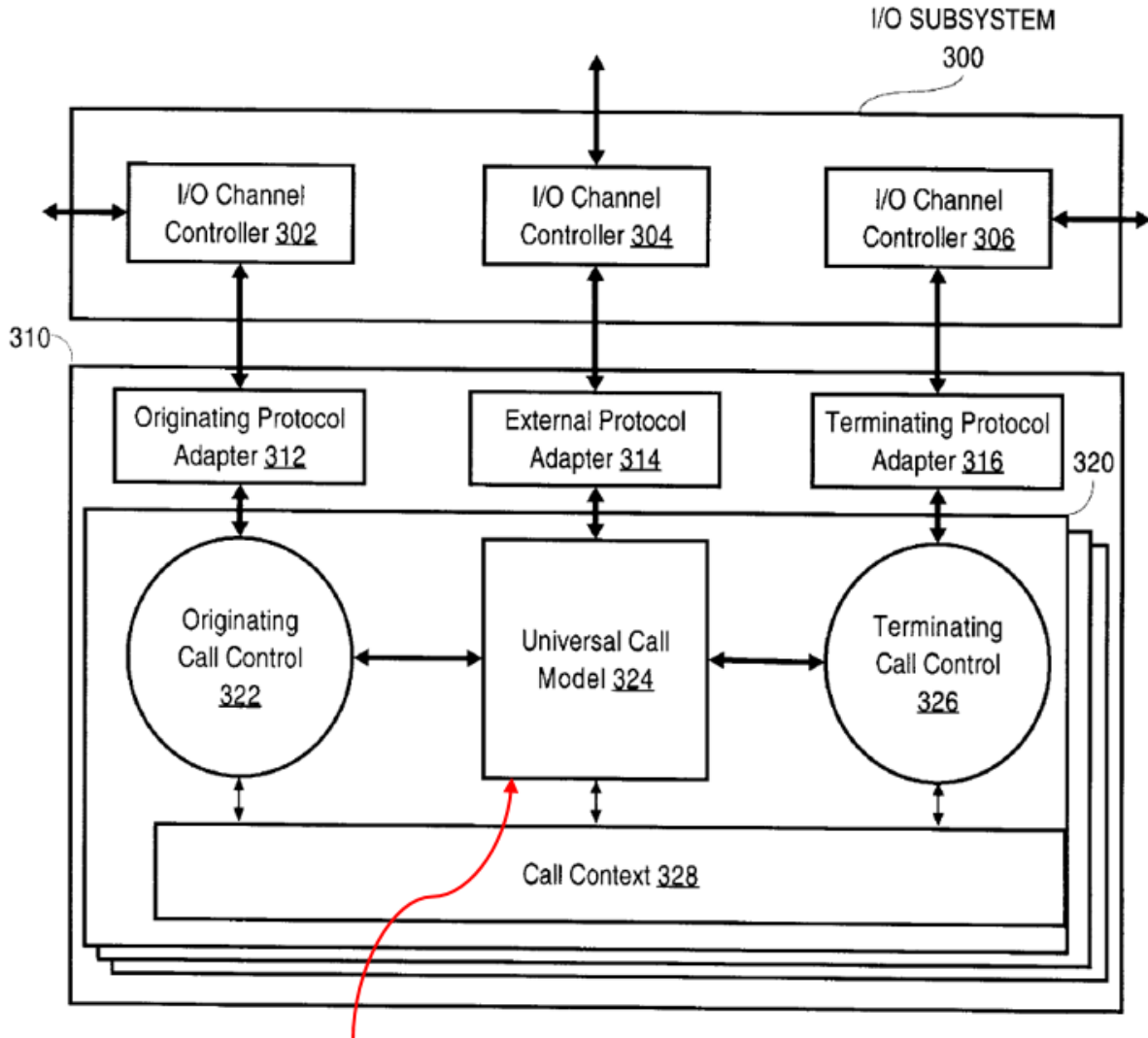
made up of multiple subcomponents. For example, the “signaling agent 138 includes a number of signaling agents 162-164 operable to” provide multiple different interfaces with entities external to the softswitch, each having a “respective logic control” and “logic control program,” “filter,” and “codec.” ’441 Patent, 8:19-26; 13:29-31. Likewise, Shankar’s signaling agent includes multiple subcomponents, the OCC and TCC, each of which provide different interfaces with external entities. In other words, Shankar’s signaling agent made up of multiple subcomponents is analogous to the ’441 Patent’s preferred embodiment in which signaling agent 138 is made up of multiple subcomponents (e.g., signaling agents 162-164). Sharma, ¶45. To the extent the claimed signaling agent is argued to require a single module, it would have been obvious to a POSITA to group the functionalities of the OCC and TCC in this manner because they both relate to translating the signaling between a protocol specific format and a protocol independent format.

Accordingly, components of the OCC and TCC, as taught by Shankar, disclose a signaling agent coupled to the networks and operable to receive incoming signaling messages, translate the incoming signaling messages to a call model event, and route the call model event.

***[1.b, 3.b, 5.b,] a call agent in communication with the signaling agent and operable to receive the call model event, request outgoing resources for establishing data sessions, generate outgoing signaling messages, and send the outgoing signaling messages to the signaling agent,***

The combination discloses a call agent in communication with the signaling agent as claimed. For example, Shankar describes a call agent provided by the universal call model (“UCM”), (e.g., alone or together with other components of the “OCC,” “TCC,” and/or other components of the “signaling unit.”). Sharma, ¶46. The UCM “handles calls in the converted universal protocol, arranges for messages to be routed ultimately to the other protocol converter, and controls the originating coding unit 110 over a control link 114.” Shankar, 6:29-33; FIGS. 1, 5a-c; Volftsun, 13:25-34.





The Universal Call Model, alone or together with components of the signaling unit that are operable to receive the call model event, request outgoing resources for establishing data sessions, generate outgoing signaling messages, and send the outgoing signaling messages to the signaling agent, provides a call agent.

Sharma, ¶46; Shankar, FIG. 3 (annotated). In other words, the universal call model is the state machine that handles call set-up and teardown. Sharma, ¶46. Volftsun, the “entire contents” of which are incorporated by Shankar, likewise illustrates a universal call model state machine, including the call states associated with call

set-up/teardown through which the UCM moves during call processing. *See, e.g.*, Volftsun, 20:31-60; FIG. 10; FIGS. 11-22; Shankar, 6:53-67.

The UCM is *operable to receive the call model event* from the OCC.  
Sharma, ¶47. The “UCM 124 of originating signaling unit 120 receive[s] the universal protocol ‘[Call]’ message.” Shankar, 14:4-18; *see also id.*, 11:16-22 (the OCC “output[s] a non-protocol specific (i.e., universal protocol specific) message to UCM 324.”).

The UCM is operable to *request outgoing resources for establishing data sessions*. Shankar explains that the universal call model (UCM) “handles calls in the converted universal protocol” and “arranges for messages to be routed ultimately to the other protocol converter [e.g., the terminating protocol converter].” Shankar, 6:28-33; FIG. 1.

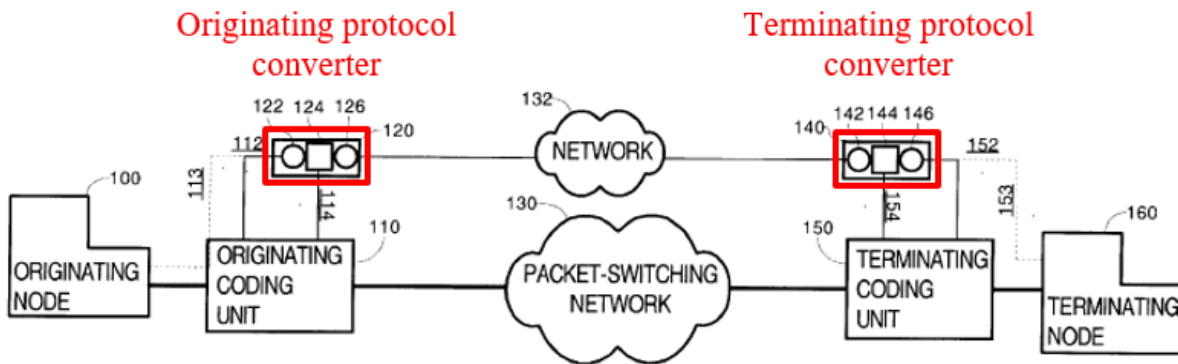


FIG. 1

Sharma, ¶48; Shankar, FIG. 1(annotated).

Shankar further describes the route determination and request for outgoing resources. For example, the “originating signaling unit 120” (which operates the UCM) makes a request for outgoing resources when “querying external systems” or by “accessing internal routing tables” to determine which terminating signaling unit 140 should receive the call. Shankar, 5:44-52; *see also* FIG. 3 (showing only the UCM as connected via an “external” adapter to external resources). Such requests result in a determination of at least a portion of the outgoing resources that will be used to terminate requested data sessions, such as the “terminating signaling unit,” and “terminating coding unit.” Sharma, ¶49.

The UCM (and/or the UCM and some components of the OCC) is involved in such requests for outgoing resources. As discussed above, the UCM is the universal call model state machine that handles call set-up/teardown, which includes outgoing resource requests required to set-up the call. Sharma, ¶50; Volftsun, 20:31-60; FIG. 10; FIGS. 11-22. Moreover, Shankar teaches that the TCC is “typically instantiated **after** routing analysis has determined the route,” further indicating that the UCM is responsible for outgoing resource requests. Shankar, 6:40-45 (emphasis added). Accordingly, the UCM alone or together with other functional components of Shankar’s signaling unit involved in these resource requests may be considered to collectively provide a call agent. Sharma, ¶50.

Shankar also discloses that the UCM is operable to *generate outgoing signaling messages, and send the outgoing signaling messages to the signaling agent*. The UCM generates outgoing signaling messages and sends these messages to the signaling agent, which in turn converts the universal protocol messages to an appropriate protocol. “The terminating call control (TCC) 126, typically instantiated after routing analysis has determined the route, converts signaling messages between the universal protocol and the protocol that provides connectivity to the terminating signaling unit 140.” Shankar, 6:40-53. In other words, after routing analysis has determined the route (e.g., including the routing analysis performed by the UCM), the outgoing signaling message to implement the routing is sent to the TCC for conversion from the universal protocol to the outgoing protocol. Sharma, ¶51. Indeed, Shankar describes that after performing the “call routing capability,” the “originating signaling unit 120 transmits a signaling message, including information for establishing the voice and the network address of the originating coding unit 110, through network 132 to terminating signaling unit 140.” *Id.* 5:44-52.

Accordingly, the UCM, alone or together with components of the OCC and/or other components of the signaling unit, as taught by Shankar, disclose a call agent in communication with the signaling agent and operable to receive the call model event, request outgoing resources for establishing data sessions, generate

outgoing signaling messages, and send the outgoing signaling messages to the signaling agent, as claimed.

***[1.c, 3.c, 5.c] the signaling agent further terminating the data sessions on the requested outgoing resources; and***

The combination discloses the signaling agent terminating the data sessions on the requested outgoing resources. Shankar describes that “[o]nce the appropriate terminating signaling unit 140 is identified, TCC 126 of the originating signaling unit 120 generates and transmits an XISUP IAM message 410 to the terminating signaling unit 140 with the standard ISUP IAM information plus the Universal Call Reference and **the network address of the bearer channel on the originating coding unit 110.**” Shankar, 14:33-45 (emphasis added); Abstract (“establish a bearer channel . . . for carrying the voice data.”); *see also* 15:64-16:7 (“the TCC 506 is responsible for communicating the signaling data with the terminating coding unit 150 over link 152.”); FIGS. 5a-c. The IAM message is based on the universal protocol message received from the UCM, and provides call set-up information that leads to termination of the call on the requested outgoing resources. Sharma, ¶52.

Accordingly, signaling sent by the TCC, as taught by Shankar, discloses the signaling agent further terminating the data sessions on the requested out-going resources, as claimed.

***[1.d] further comprising a network directory server operable to receive requests for routing information to establish data sessions from the call agent, and provide routing information to the call agent.***

The combination discloses a network directory server operable to receive requests for routing information to establish data sessions from the call agent, and provide routing information to the call agent. Sharma, ¶53. The '441 patent describes that “[s]oftswitch 100 includes six main functional components [including] . . . network directory server [144],” and that “[n]etwork directory server 144 may reside on the same platform as call agent 140 or may be distributed.” '441 patent, 7:48-50. The '441 patent further describes that the “network directory server 144” includes a “network directory service 200” that “**accesses in-memory tables 202**, which store user profiles, routing instructions, addresses, and subscriber services,” and “further accesses external data servers 203.” '441 patent, 8:48-53.

Like the '441 Patent's “in-memory tables,” Shankar describes a network directory server provided by “lookup tables” and the resource/medium that stores the “lookup tables.” Shankar's signaling unit “resolves the call routing through the network 132 to the terminating signaling unit 140, based on dynamic techniques, **static techniques such as provisioned lookup tables**, or a combination [thereof],” and, in turn, routes messages to a “network node,” and “destination telephone.” Shankar, 10:19-22 (emphasis added) (“The terminating protocol adapter 316 can

route messages from associated call instances to the corresponding I/O channel controller 306, to a network node, and ultimately to the destination telephone.”); 14:32-45. In other words, Shankar’s signaling unit (e.g., the UCM handling call set-up) queries the “lookup tables” to obtain routing information to resolve the call routing through the network 132, which in turn facilitates route determination for user communications over network 130. The “lookup tables” are thus operable to receive requests for routing information to establish data sessions, and provide routing information to the call agent (e.g., UCM), so that a route can be determined. Sharma, ¶¶53-54. Shankar’s “lookup tables,” and the resource that stores them, disclose the preferred configuration of “in-memory tables” recited by the ’441 Patent. *Id.*

Accordingly, Shankar’s “lookup tables” and storage medium operable to receive queries for routing information and provide routing information, as taught by Shankar, discloses [1.d]. Additionally, such features would have been obvious to a POSITA in view of a predictable combination with Huang. *See, supra*, IV.A.2-3 (Huang and the combination of Shankar and Huang). For example, Huang provides an example of a “gateway computer” that interconnects different network types. Huang, 4:5-10. The “gateway computer” includes a network directory server provided by a “routing resolution module.” *Id.* In particular, Huang teaches that its “routing resolution module” includes a “network routing

database 46,” which stores network routing information. *Id.*; Sharma, ¶55.

Shankar’s UCM and related components would have requested routing information from the “routing resolution module,” including information from the “network routing database 46,” to determine a route through the network, in accordance with the teachings of Shankar and Huang.

Accordingly, Shankar’s signaling units, including a “routing resolution module” in accordance with Huang’s teachings, disclose [1.d].

***[3.d] further comprising a network directory server operable to receive requests for address resolution to establish data sessions from the call agent, and provide address resolution responses to the call agent.***

The combination discloses [3.d]. Shankar explicitly indicates that the signaling unit (e.g., the UCM responsible for call set-up/teardown) “**resolves** the call routing through the network 132 to the terminating signaling unit 140, based on dynamic techniques, static techniques such as provisioned lookup tables, or a combination [thereof],” and the messages are, in turn, routed to a “network node,” and “destination telephone.” Shankar, 10:19-22 (emphasis added). In other words, “lookup tables” are operable to receive queries for address resolution, and provide address resolution responses, so that the UCM can “resolve the call routing.” Sharma, ¶56. Here again, Shankar’s “lookup tables,” and the resource that stores them, disclose the preferred configuration of “in-memory tables” recited by the ’441 Patent. *Id.* Notably, the plain language of the claim is not limited to any



particular type of address resolution, nor does the specification of the '441 Patent provide specific implementations. Accordingly, “lookup tables” and associated storage medium operable to receive queries for address resolution provide routing information, as taught by Shankar, discloses [3.d].

Additionally, Shankar describes a network directory server provided by a “domain name server (DNS)” that is queried for address resolution (e.g., of the originating coding unit, which is used to establish the call). Sharma, ¶57. Shankar, 5:32-43. For example, the “originating signaling unit 120 obtains the network address of the originating coding unit 110 within the packet-switching network 130” by “inquiring another computer system (not shown) such as domain name server (DNS).” The “domain name server” thus receives queries for the “network address” (e.g., address resolution requests) and responds with the associated “network address” (e.g., address resolution responses), so that the call can be set-up using the appropriate coding unit and associated resources. Shankar’s DNS is like the embodiment of the '441 patent, which describes that “Network directory server 144 may reside on the same platform as call agent 140 or **may be distributed.**” '441 Patent, 7:48-50.

To the extent [3.d] is narrowly interpreted such that Shankar is considered to not expressly disclose the claimed network director server, such features would have been obvious to a POSITA in view of a predictable combination with Huang.

*See, supra*, IV.A.2-3 (Huang and the combination of Shankar and Huang).

Huang's "routing resolution module" includes an "address resolution logic 45."

The call agent of Shankar's system (e.g., including the UCM and related components) in view of Huang would have made requests for address resolution when processing a call, and the "address resolution logic" would have in turn provided address resolution responses, in accordance with the call routing objectives suggested by Shankar and Huang. Sharma, ¶58. Such responses would have provided address resolution to allow a data session to be established between an origination and termination.

Accordingly, Shankar's signaling units, including a "routing resolution module" having "address resolution logic," in accordance with Huang's teachings, disclose [3.d].

***[5.d] wherein the signaling agent comprises: a logic control executing a logic control program and operable to process signaling messages of a particular signaling protocol;***

Shankar's signaling unit discloses [5.d]. Shankar teaches that the "OCC" and "TCC" are implemented as "state machines" by "objects in an object-oriented programming language or by other data structures," indicating that the "OCC" and "TCC" provide a logic control. Shankar; 11:4-15; 11:16-63. These state machines execute a logic control program, transitioning between states to model a call from the perspective of the originating protocol, a universal protocol, and the

terminating protocol, for example. *Id.*; Sharma, ¶59; *see also* 9:15-19. Likewise, the '441 patent describes that “[l]ogic controls 315-321 are the data-configurable generic state machine processors that execute logic control programs,” and a “logic control program defines how the logic engine is to operate.” ’441 Patent, 10:34-37; *see also id.*, 8:38-41.

Shankar’s logic control is operable to process signaling messages of a particular signaling protocol. “The OCC 322, UCM 324, and TCC 326 state machines model a call from the perspective of the originating protocol, a universal protocol, and the terminating protocol, respectively.” Shankar, 11:12-15; *see also* 8:58-61; 6:17-28; 9:15-24; 9:34-55; 11:16-25; Sharma, ¶60.

Accordingly, the OCC and TCC of Shankar’s signaling unit disclose [5.d].

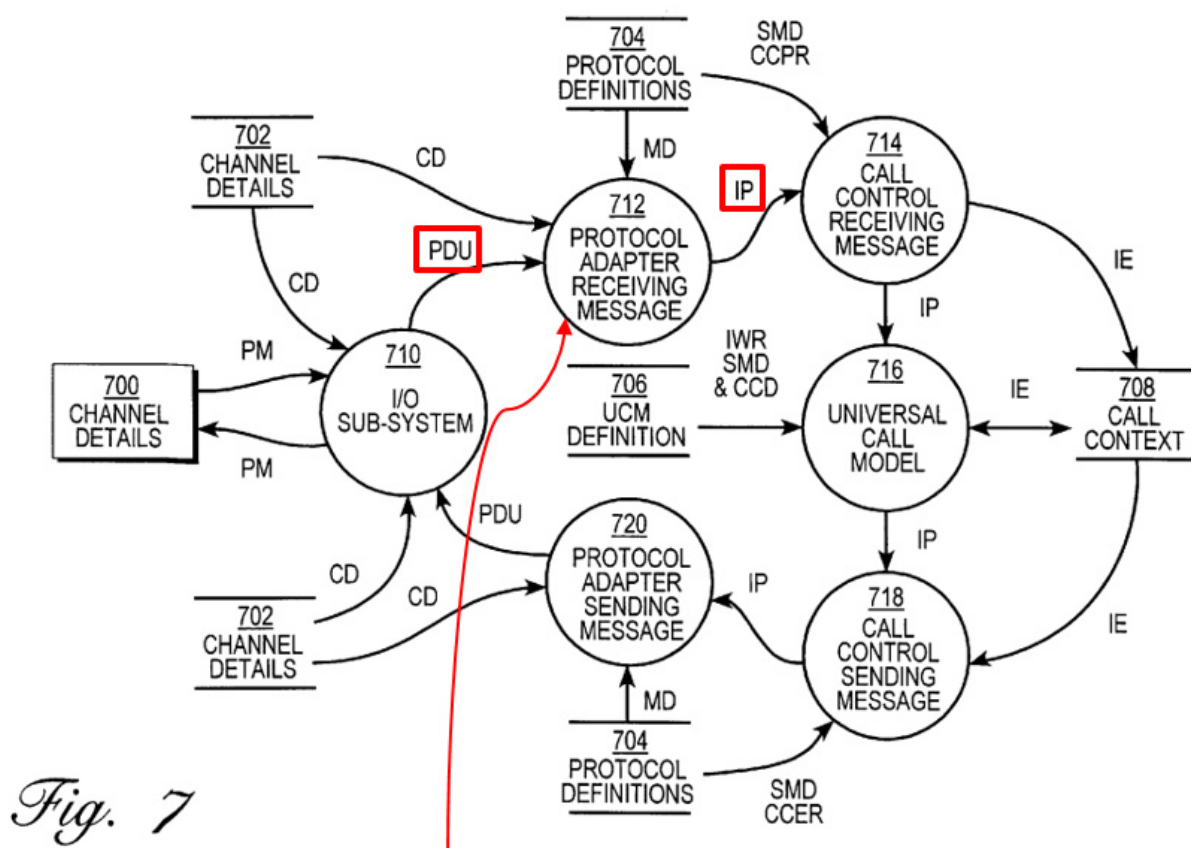
***[5.e] a codec specialized in the signaling protocol of an access network and operable to parse and format signaling messages according to the signaling protocol; and***

Shankar discloses a codec specialized in the signaling protocol of an access network and operable to parse and format signaling messages according to the signaling protocol. Shankar’s “protocol adapter” is configured for parsing and formatting messages according to a particular signaling protocol, indicating the “protocol adapter” includes a codec that is ***specialized*** for the particular protocol. *See, e.g.* Shankar, 9:15-55; Sharma, ¶61. The “protocol adapter is a software module responsible for interfacing the protocol conversion engine 310 with the I/O

subsystem 300,” and “is capable of decoding an incoming message to determine the call with which the message is associated.” *Id.* The protocol adapter functions to decode incoming protocol-specific signaling messages, and vice versa. *Id.* (“bi-directional”). The protocol adapter is thus specialized to parse and format signaling messages according to particular signaling protocols, and effectuate transmission and reception with the outside world. Sharma, ¶61. Likewise, the ’441 patent describes its codecs merely as “coders/decoders” that are “specialized communication modules.” ’441 Patent, 8:25-26.

Moreover, Volfstun provides further detail of a protocol adapter that parses and formats signaling messages according to a particular signaling protocol. “Protocol Adaptors 404 and 408 respectively take the signaling messages coming from or going to the originating party and take the signaling messages coming from or going to the terminating party and **parse and assemble the signaling messages.**” Volfstun, 9:45-49 (emphasis added). As illustrated in FIG. 7, the protocol adapters execute a “Protocol Adapter Receiving Message 712” process in which the protocol adapter receives incoming signaling Protocol Data Units (“PDU”), parses and formats the signaling PDUs, and outputs protocol-specific “Internal Primitives” that are in turn routed to a “Call Control Receiving Message 714” process of the OCC. Volfstun, 14:2-8. Internal Primitives are “vehicles for carrying signaling information data and/or signals,” and “may contain State Signals

and Information Elements.” Volftsun, 10:20-27; 5:40-45.



*Fig. 7*

The “protocol adapter” includes a codec operable to parse and format incoming protocol data units (PDU) into “internal primitives”

Sharma, ¶62; Volftsun, FIG. 7(annotated). As further illustrated in FIG. 7, the Protocol Adapter carries out the translation to generate Internal Primitives in part using “message definitions,” which are “the collection of information about messages including format, content, and coding as would be required for a software process to receive, analyze, and reconstruct messages.” Volftsun, 13:13-17. In other words, the protocol adapter includes a codec because it is specialized in a particular signaling protocol of an access network and generates “Internal

Primitives” by parsing and formatting the incoming signaling it receives. Sharma, ¶62. To the extent that these aspects of the protocol adapters are not explicitly disclosed in Shankar’s general disclosure of the protocol adapter, a POSITA would have found it obvious to adapt internal communication in accordance with Volftsun’s more specific description. Indeed, as discussed previously, Shankar’s protocol converter likewise includes an I/O subsystem, protocol adapters, and OCC, UCM, and TCC call control state machines that communicate with each other to effectuate a call. Incorporating specific implementation details in accordance with Volftsun’s teachings would have facilitated communication between the components (e.g., which is taught by Shankar itself), such that incoming signaling messages could be effectuated, and a call executed. Sharma, ¶63. Moreover, a POSITA would have had a reasonable expectation of success, especially in view of the common architecture, components, and functions described by Shankar and Volftsun. *Id.*

Shankar describes its protocol adapters are “implemented to **support respective protocols** or protocol families.” Sharma, ¶64; Shankar, 9:15-17; Volftsun, 10:50-57(“Protocol Adapters 510, 512, 514, 516, and 518, each support a unique protocol or protocol family (comprised of a group of protocols which differ only by minor variance.”); 11:16-19.

Alternatively, even if Shankar’s protocol adapter were considered not to

explicitly disclose a “codec,” a POSITA would have found it obvious to use a codec to implement translation between incoming signaling and Internal Primitives, because codecs are, by definition, communication modules that carry out translation. Codecs were widely known modules for parsing and formatting signaling messages such that the signaling messages could be processed and the signaling effectuated by the system. Sharma, ¶65.

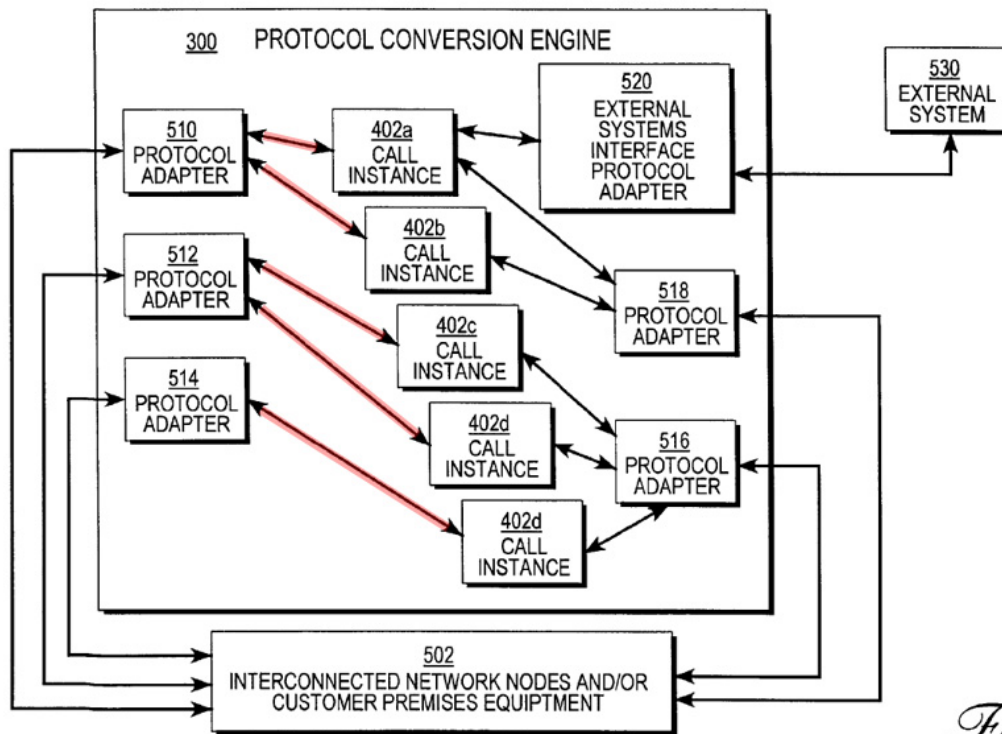
Accordingly, protocol adapters that parse and format signaling to internal primitives, as taught by Shankar, disclose [5.e].

***[5.f] a filter operable to filter and route signaling messages from the codec to the logic control.***

The combination discloses a filter provided by the “protocol adapter,” which is operable to filter and route signaling messages from the codec to the logic control. After “internal primitives” are generated in conjunction with the protocol adapter’s codec, the protocol adapter filters the internal primitives to the associated OCC call instance (i.e., to the logic control). Sharma, ¶66. “An originating protocol adapter 312 is capable of decoding an incoming message to determine the call with which the message is associated.” Shankar, 9:34-55; 9:21-24 (“route protocol-specific messages between an I/O channel and the appropriate call instance.”); 10:22-26 (“pass them to the appropriate call instance”).

Volftsun further describes that the protocol adapter filters and routes the signaling messages from the codec to the logic control. As illustrated in FIG. 5, a

logic control (e.g., “OCC,” “TCC”) is instantiated for each call instance, and the “protocol adapter” routes “internal primitives” from its codec to the instance of the logic control associated with the call.



*Fig. 5*

The “protocol adapter” routes messages from its codec to the logic control (e.g., OCC) of the call instance associated with the message

Sharma, ¶67; Volftsun, FIG. 5(annotated). Here again, a POSITA would have found it obvious to adapt internal communication in accordance with Volftsun’s more specific description of filtering and routing signaling messages between the codec and logic control. *Id.*

Accordingly, a protocol adapter that filters and routes signaling from its codec to the logic control of a particular call instance, as taught by Shankar,



discloses [5.f].

***[6] The softswitch, as set forth in claim 5, wherein the signaling protocol is SS7.***

The combination discloses [6]. For example, Shankar describes that “the signaling units also convert the legacy protocols of the originating node 100 and the terminating node 160, such as . . . SS7/C7 . . . , into messages for communicating with one another and for controlling a coding unit over control links 114 and 154” Shankar, 5:20-26 (emphasis added); 13:12-32 (“SS7”); Sharma, ¶68.

***[7] The softswitch, as set forth in claim 5, wherein the signaling protocol is session initiation protocol.***

The combination discloses [7]. Shankar describes that “the signaling units also convert the legacy protocols of the originating node 100 and the terminating node 160, such as . . . SIP . . . , into messages for communicating with one another and for controlling a coding unit over control links 114 and 154” Shankar, 5:20-26 (emphasis added); Sharma, ¶68.

***[8] The softswitch, as set forth in claim 5, wherein the signaling protocol is H.323.***

The combination discloses [8]. Shankar describes that “the signaling units also convert the legacy protocols of the originating node 100 and the terminating node 160, such as . . . H.323 . . . , into messages for communicating with one another and for controlling a coding unit over control links 114 and 154” Shankar, 5:20-26 (emphasis added); 13:12-32; Sharma, ¶68.

***[9.pre] A softswitch interconnecting networks of different transport protocols, comprising:***

*Supra*, [1.pre].

***[9.a.] a signaling agent coupled to the networks and operable to receive incoming signaling messages, translate the incoming signaling messages to a call model event, and route the call model event; and***

*Supra*, [1.a].

***[9.b] a call agent in communication with the signaling agent and operable to receive the call model event, request outgoing resources for establishing data sessions, generate outgoing signaling messages, and send the outgoing signaling messages to the signaling agent,***

*Supra*, [1.b].

***[9.c] the signaling agent further terminating the data sessions on the requested outgoing resources; and***

*Supra*, [1.c].

***[9.d] further comprising a network gateway which comprises: a logic control executing a logic control program and operable to process messages of a particular transport protocol;***

The combination discloses [9.d]. The '441 patent describes that “[n]etwork gateway 146 includes gateway agents 180 for interfacing with MGCP media gateway switch, IPDC media gateway switch, etc. Each gateway agent 180 each also includes a logic control 182 executing a logic control program, a filter processor 183, and a codec 184.” 441 patent, 8:26-34; Sharma, ¶69.

Shankar likewise describes that its “signaling units” including functional units for interfacing with its “coding units.” For example, Shankar describes “the

UCM 124 generates and transmits a create connection message 406, specifying the bearer channel, through external protocol adapter 314, I/O channel controller 304, control link 114 to originating coding unit 110.” Shankar, 14:10-13. “The create connection message 406 can be implemented according to an appropriate protocol such as a CRCX message in the Simple Gateway Control Protocol (SGCP), a proposed International Engineering Task Force (IETF) standard submitted by Bellcore and Soliant Internet Systems.” *Id*; *see also* Shankar, 14:46-54 (“the terminating signaling unit 140 issues a CRCX message 412 via SGCP and control link 154 to terminating coding unit 150 for setting up a connection from the endpoint for terminating node 160 to the bearer channel port of the originating coding unit 110.” In other words, the UCM and related components are operable to process messages with the coding unit (e.g., media gateway) of a particular transport protocol. Sharma, ¶70.

Notably, like the OCC and TCC described above (*see, e.g., supra* [5.d]), Shankar teaches that the UCM is implemented as a “state machine” by “objects in an object-oriented programming language or by other data structures,” indicating that the UCM provides logic control executing a logic control program. Shankar, 11:4-15; 11:16-63. The state machines execute a logic control program that models a call from the perspective of the universal protocol, and is able to interface with coding units using an appropriate transport protocol. Sharma, ¶71; *Id.*; 14:13-

18(“Simple Gateway Control Protocol (SGCP)”); *see also* 9:15-19; 14:19-32.

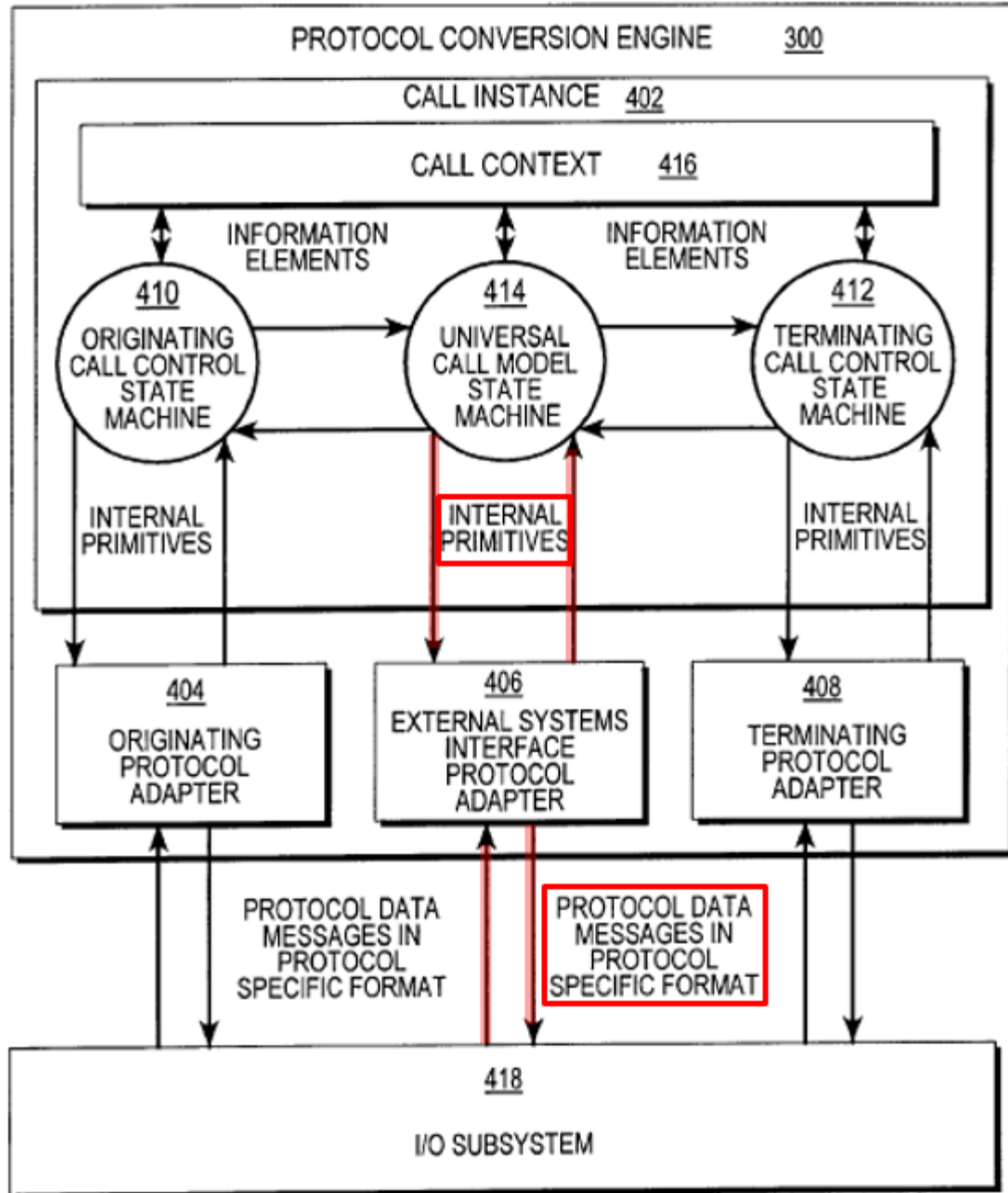
Accordingly, the UCM, as taught by Shankar, discloses [9.d].

***[9.e] a codec specialized in the transport protocol of a media gateway and operable to parse and format the messages according to the transport protocol; and***

Shankar discloses [9.e]. Shankar’s “protocol adapter” is configured for parsing and formatting messages according to the transport protocol of a media gateway. “For implementing voice over packet-switching networks, the external protocol adapter 314 is used for real-time communication with a coding unit such as originating coding unit 110 and terminating coding unit 150.” Shankar, 10:41-45. Shankar further explains that the “protocol adapter is a software module responsible for interfacing the protocol conversion engine 310 with the I/O subsystem 300,” and is “responsible for connecting with the appropriate I/O channel controller 304 in the I/O subsystem 300 for sending and receiving messages with the coding unit and routing the messages to and from the proper call instance 320.” *Id.*, 10:45-50. In other words, the protocol adapter functions to code/decode protocol-specific signaling messages. *Id.*; 9:34-36. The protocol adapter is thus specialized to parse and format signaling messages according to particular signaling and/or transport protocols, and effectuate transmission and reception of such messages with the outside world. Sharma, ¶72. Likewise, the ’441 patent describes its codecs merely as “coders/decoders” that are “specialized

communication modules.” ’441 Patent, 8:25-26.

Moreover, Volfstun provides further detail of an external protocol adapter that parses and formats messages according to the transport protocol of a media gateway. As discussed previously, “[p]rotocol Adaptors 404 and 408 respectively take the signaling messages coming from or going to the originating party and take the signaling messages coming from or going to the terminating party and **parse and assemble the signaling messages.**” Volfstun, 9:45-49; *supra*, [5.e]. Volfstun explains that “external” protocol adapter 406 operates “in a manner similar to Protocol Adapters 404 and 408,” and “provides means for communicating to external systems and/or entities.” *Id.*, 9:8-12. In other words, protocol adapter 406 likewise receives incoming “protocol data messages,” parses and formats these incoming messages, and outputs “internal primitives” that are in turn routed to the UCM of the appropriate call instance. FIG. 4; *supra*, [5.e]; Sharma, ¶703



*Fig. 4*

Sharma, ¶73; Volftsun, FIG. 4(annotated).

Here again, the protocol adapters are specialized in a particular signaling protocol of the access network. Shankar describes its protocol adapters are “implemented to **support respective protocols** or protocol families.” Shankar, 9:15-17; Volftsun, 10:50-57 (“Protocol Adapters 510, 512, 514, 516, and 518, each” support a unique protocol or protocol family); 11:16-19. The same is true of the external protocol adapter. Sharma, ¶74.

Alternatively, even if Shankar’s protocol adapter were considered not to explicitly disclose a “codec,” a POSITA would have found it obvious to use a codec to implement translation according to the transport protocol of the coding unit, because codecs are, by definition, communication modules that carry out such translation. Codecs were widely known modules for parsing and formatting messages such that the messages could be processed by the system. Sharma, ¶75.

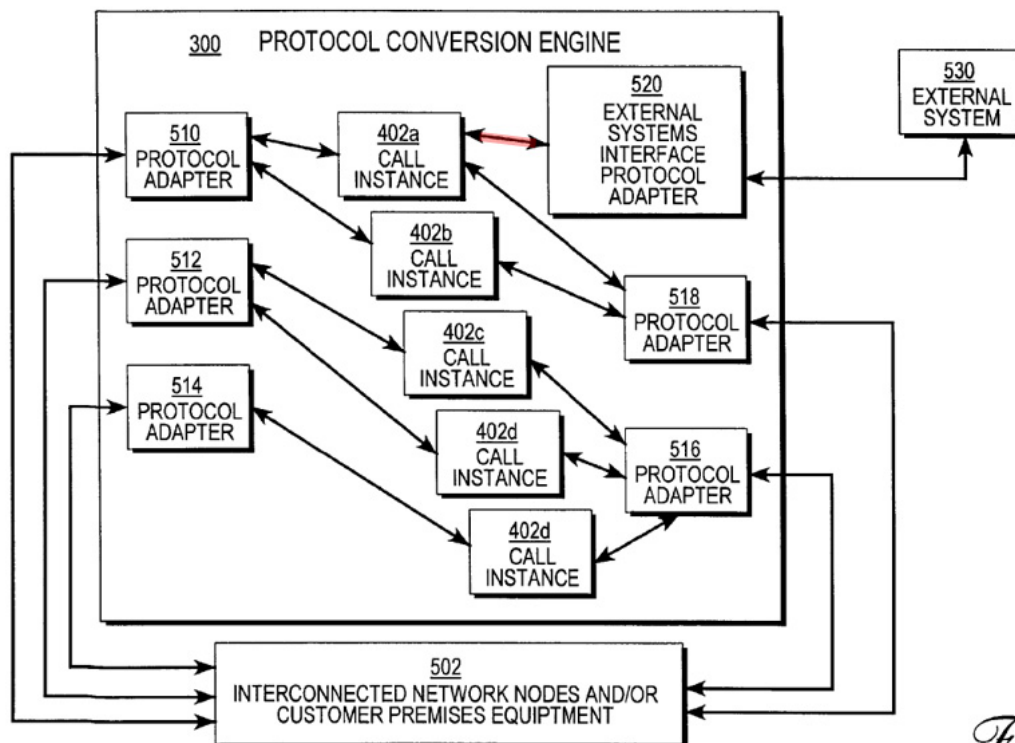
Accordingly, protocol adapters that parse and format signaling messages of transport protocols to internal primitives, as taught by Shankar, disclose [9.e]

***[9.f] a filter operable to filter and route signaling messages from the codec to the logic control.***

The combination discloses a filter provided by the “protocol adapter.” Sharma, ¶76. After “internal primitives” are generated in conjunction with the protocol adapter’s codec, the protocol adapter filters the internal primitives to the associated OCC call instance (i.e., to the logic control). “The external protocol adapter 314 is responsible for connecting with the appropriate I/O channel

controller 304 in the I/O subsystem 300 for sending and receiving messages with the coding unit and routing the messages to and from the proper call instance 320.” Shankar, 10:45-50; 9:34-55; 9:21-24; 10:22-26 (“pass them to the appropriate call instance”).

Indeed, Volftsun further describes that the protocol adapter filters and routes the signaling messages from the codec to the logic control. A logic control (e.g., “UCM,”) is instantiated for each call instance, and as illustrated in FIG. 5, the “protocol adapter” routes “internal primitives” from its codec to the instance of the logic control associated with the call.



*Fig. 5*

The “protocol adapter” routes messages from its codec to the logic control (e.g., UCM) of the call instance associated with the message



Sharma, ¶77; Volftsun, FIG. 5(annotated).

Accordingly, a protocol adapter that filters and routes signaling from its codec to the logic control of a particular call instance, as taught by Shankar, discloses [9.f].

***[12] The softswitch, as set forth in claim 9, wherein the transport protocol is simple gateway control protocol.***

Shankar describes the transport protocol may be “Simple Gateway Control Protocol (SGCP).” Shankar, 14:13-18; Sharma, ¶78.

***[13] The softswitch, as set forth in claim 1, wherein the networks comprise a public switched telephone network.***

The combination discloses [13]. For example, Shankar describes a need addressed by its system of carrying voice calls that are originated or terminated by legacy systems over a packet-switching network, and explicitly identifies public switched telephone networks. Shankar, 1:11-25 (“Public Switched Telephone Network (PSTN)”); 1:44-48; 1:49-65 (“integrate the PSTN with the Internet.”); Sharma, ¶79. Moreover, Shankar indicates that the “originating” or “terminating node” can be a “telephone switch” or “legacy telecommunications system.” Shankar, 3:65-4:3; *see also* 4:4-20 (“packet-switching network 130 may even be overlaid on the PSTN.”).

***[14] The softswitch, as set forth in claim 1, wherein the networks comprise a packet network.***

The combination discloses [14]. Shankar describes its system carries voice

calls from an originating node 100 to a terminating node 160 over a “packet-switching network 130.” Shankar, 3:38-49; Title; 1:5-2:67; FIGS. 1-5C; Sharma, ¶80.

***[15] The softswitch, as set forth in claim 1, wherein the networks comprise a wireless network.***

The combination discloses [15]. Shankar describes that “originating node 100” can be implemented as a “wireless PBX” (private branch exchange).

Shankar, 3:65-4:3; Sharma, ¶81.

***[16] The softswitch, as set forth in claim 1, wherein the call agent comprises a protocol-independent logic engine operable to execute a function-specific logic control program.***

The combination discloses [16]. For example, Shankar’s call agent provided by the UCM (e.g., and related components of the OCC and TCC that handle the universal protocol) manages the call using a “universal protocol” according to a “universal call model.” Shankar, 11:38-52; Sharma, ¶82. Moreover, Shankar describes that the UCM is “implemented as [a] state machine[] by objects in an object-oriented programming language,” further indicating that the UCM executes function-specific logic control programs. Shankar, 11:4-15. Indeed, the ’441 patent likewise describes the call agent logic engine as a “**protocol-independent, data-configurable, multi-threaded state machine processor.**” ’441 Patent, 8:34-56. Additionally, Shankar’s call agent may be considered to include components of the OCC and TCC that handle the universal call model.

Shankar's call agent executes multiple logic control programs responsible for specific functions, such as call resolution, call routing, bearer selection, generation of CDRs, communication with coding units, etc. Each of these are functions for specific tasks carried out, at least in part, by the call agent (e.g., portions of the UCM, which provides the logic for call set-up/teardown, as discussed above). Shankar, 5:14-32 ("the signaling units perform such functions as call resolution, call routing, bearer selection, and generation of call detail records (CDRs) for billing management."); Sharma, ¶83.

Moreover, Volftsun describes a "Call Context" that is "a collection of data elements that describe the contents of the messages passed through Protocol Conversion Engine 300. The description is in a physical representation independent, **protocol-independent form.**" Volftsun, 9:51-55 (emphasis added); 10:7-11 ("In particular UCM 414 is comprised of states, primitives, alternatives, decisions, saves, and procedures that are invoked in response to Internal Primitives sent by OCC 410 and TCC 412 in such a way that a Call Instance 402 is **processed in a protocol independent fashion.**"); Sharma, ¶84.

Accordingly, Shankar's UCM comprises a protocol-independent logic engine operable to execute a function-specific logic control program, as claimed.

*[17] The softswitch, as set forth in claim 1, further comprising a billing sub-system in communication with the call agent and operable to generate call detail records.*

The combination discloses [17]. For example, Shankar describes that its “signaling units perform such functions as call resolution, call routing, bearer selection, and **generation of call detail records (CDRs) for billing management**,” indicating that the signaling unit includes a billing sub-system. Shankar, 5:14-32 (emphasis added). A POSITA would have found it obvious that the billing sub-system is in communication, directly or indirectly, with the UCM involved in routing and other call processing. *Id.*; Sharma, ¶¶85-86. Moreover, Shankar explicitly teaches that “[s]ome call-related information that persists beyond the duration of the call may be stored in a database in main memory 206 or storage device 208 to implement billing records,” further indicating storage of information in a billing server. 10:56-11:3.

**B. GROUND 1B – Claim 2 is obvious over Shankar in view of Huang, and Capers**

**1. Capers**

Capers describes a telecommunications network including a signaling gateway that translates between signaling protocols of interconnected networks and manages resource and call state management to handle calls on the network. Capers, Abstract; 1:37-42; 2:36-3:7; 4:44-61; FIG. 1-8; Sharma, ¶87. The signaling gateway includes a transaction control layer (e.g., TCL 214) that performs call and resource state management functions, including “the capability of tracking different types of states for both calls and network resources (i.e.,

circuits) while using industry standards for call processing.” *Id.*

In more detail, Capers describes that “state tracking is performed by the TCL 214’s state-dependent processing 304 using a call/circuit state data store 306,” which is a “database” that includes information about the availability of a particular resource (e.g., circuit) to complete the call. Capers, 10:10-40; Table 4; FIGS 2-3(illustrating call/circuit state data store 306 of TCL within the signaling gateway). Via the “call/circuit state data store 306,” the “signaling gateway 110 knows which circuits are available” and which are “unavailable,” such as due to an administrative block, processing of a current call, etc. *Id.* An available circuit may be appropriately selected “so that the signaling gateway 110 may map calls to resources.” *Id.*; Sharma, ¶88.

Capers additionally describes verifying that an incoming circuit is valid to handle an incoming call. First, Capers describes verifying the validity of an incoming circuit by verifying that the incoming circuit is not in a blocked state. Sharma, ¶89. For example, Capers describes that circuits can be administratively blocked through a maintenance command, in which case the circuit is not valid to handle a call. Capers, 9:11-17; 10:1-17. The “TCL 214 includes a single state machine process that tracks the states of both calls and resources,” specifically including the “circuits between the bridging switch 104 and the NGSN 108,” (i.e., an incoming circuit of an inbound call). Capers, 10:18-26. The “resource states

are managed based on” SS7 ISUP messages “from the bridging switch 104.” *Id.*

The various states are tracked “so that the signaling gateway 110 may map calls to resources.” *Id.*, 10:27-41.

Second, Capers describes verifying the validity of an incoming circuit by verifying the circuit is equipped on its platform. Sharma, ¶90. “Initial processing 302 for SS7 messages from bridging switch 104 includes general validation” that “ensures that the SS7 ISUP message sent to the signaling gateway 110 does not include a circuit identification code (CIC) of a circuit not equipped on the NGSN 108 platform.” Capers, 9:18-25; Table 2 (“Unequipped Circuit Identification message”); Table 4.

Notably, Capers was applied during original prosecution for its teaching of “verifying an incoming circuit is in a valid state.” Prosecution History, pp.190-191 (citing Capers, 9:21-24); Sharma, ¶91. Applicant did not dispute that Capers disclosed this feature, nor did Applicant dispute the Examiner’s combination of this feature with Elliott. *Id.*, pp. 237-238. Indeed, the Examiner’s analysis explicitly set forth example motivation for such a combination, which was uncontested by Applicant in the record. Prosecution History, pp.190-191 (describing Capers’ “general validation” and motivation to “improve the system by ensuring that the soft switch is capable of processing the SS7 ISUP messages sent to it.”).

The examiner withdrew the rejection based on Elliott and Capers based on a perceived deficiency of Elliott<sup>1</sup>, which Shankar does not have. For at least this reason, Shankar, Huang, and Capers is not “the same or substantially the same prior art or arguments” previously presented to the Office. 35 U.S.C. § 325(d).

## **2. Combination of Shankar, Huang, and Capers**

A POSITA would have been motivated to apply Capers’ resource availability monitoring to enhance Shankar and Huang’s system with additional resource availability information during call processing. Sharma, ¶92. Shankar describes selecting bearer channels based on “channel availability,” although Shankar does not provide exhaustive implementation details of the selection process. Shankar, 19:10-15 (“such factors as channel availability.”); 14:4-18 (“based upon . . . an available channel.”). Capers illustrates how channel availability information can be monitored and made available to the entity handling the call.

Based on Shankar’s, Huang’s, and Caper’s disclosures, a POSITA would have been led to include a “call/circuit state data store,” similar to that suggested by Capers, that stores information about the availability of a particular resource

---

<sup>1</sup> As discussed above, the examiner’s application of Elliott, and reasons for allowance, were also based on legal and factual error.

(e.g., circuit) to complete the call. Capers, 10:10-4. The combined system would have thus included a database that receives resource requests (e.g., outgoing circuit/bearer channels), and responds with circuit/bearer channel availability which can be used by the UCM to map the call to particular resources. Sharma, ¶93.

Multiple reasons would have prompted a POSITA to incorporate Capers' suggestions as described above. First, resource management was a known feature of conventional call process that Shankar itself indicates should be included in its system, as indicated by Shankar's use of "availability" information in selecting a bearer channel. A POSITA would thus have had a reasonable expectation of success incorporating Capers' resource management features to perform bearer selection. Sharma, ¶94.

Moreover, a POSITA would have been led to incorporate Capers' suggestions as described above to improve system reliability and efficiency. A POSITA would have found it obvious, based on Capers' teachings regarding resource management, that call set-up would be enhanced by basing channel selection in part on resource availability to ensure that the selected resources were available to effectively handle the call. If unavailable resources were inadvertently assigned to the call, the call would not be effectively completed, and the parties to the call unable to connect with one another. Sharma, ¶95.



Additionally, by tracking availability of specific resources, the reason a circuit is unavailable can be known, as explicitly taught by Capers, which could in turn be used to enhance system performance, troubleshoot, etc. Capers, 3:1-7; Sharma, ¶96.

Relatedly, a POSITA would have been led by Capers' teachings to verify the validity of incoming circuits to handle inbound calls in at least two ways. First, in a "blocked" state, the circuit is not in a state appropriate to handle transmission of the voice data to execute a call. Sharma, ¶97. A POSITA would have found it obvious, based on Capers' teachings, that validating the incoming circuit would have enhanced reliability and efficiency by ensuring that the circuit was usable for the inbound call before proceeding further with call processing. Indeed, Capers indicates that appropriate responses can be generated by TCL 214, and/or appropriate remedial action taken, due to resource limitations. Capers, 9:48-52 (describing an error message "when requests may not be accomplished due to resource limitations."). Additionally, a POSITA would have been motivated to incorporate advantages associated with identifying and knowing the circuit is not valid, rather than the call simply failing or otherwise not being connected. Capers, 3:1-7 (describing knowing reasons for resource unavailability as an advantage of its system).

Second, a POSITA would have been led to verify the validity of the

incoming circuit by ensuring that the CIC identifying the incoming circuit is not “unequipped” to handle calls via the signaling unit platform, in accordance with Capers’ teachings. Here again, a POSITA would have found it obvious, based on Capers’ teachings, that validating the incoming circuit would have enhanced reliability by ensuring that the circuit was usable for the inbound call before proceeding further with call processing. Sharma, ¶¶98; Prosecution History, pp.190-191.

Ultimately, a POSITA would have been led to implement Capers’ resource management and circuit validation as discussed above, and would have had a reasonable expectation of success, because doing so would have been merely the application of a known technique (e.g., tracking and storing resource availability in a database accessible by a call processing entity/verifying the validity of an incoming circuit) to a known system (e.g., Shankar’s system that selects bearer channels based on “availability) ready for improvement to yield predictable results. *KSR*, 550 U.S. at 417; Sharma, ¶¶99. Shankar and Capers both describe bearer channel/circuit selection based on the circuit’s ability to handle transmission required for call execution. It would have been obvious to a POSITA that applying the suggestions of Capers to Shankar and Huang’s system would have led to predictable results without significantly altering or hindering the functions performed by Shankar/Huang’s system. Sharma, ¶¶99.

### **3. Application of Shankar, Huang, and Capers to Challenged Claims**

***[2] The softswitch, as set forth in claim 1, further comprising a resource manager operable to receive outgoing resource requests from the call agent, and provide outgoing resource availability responses to the call agent.***

A POSITA would have found it obvious, in view of predictable combination with Capers as discussed above, to include resource tracking and management in Shankar/Huang's system. *Supra* IV.B.1-2 (Capers and combination with Capers). The result would have included a "call/circuit state data store" database that provides a resource manager operable to receive outgoing resource requests from the UCM, and provides bearer channel/circuit availability responses (e.g., outgoing resource availability responses) to the UCM to use on call-setup. Capers, 10:27-40 ("map calls to resources."); Sharma, ¶100.

Accordingly, a "call/circuit state data store" operable to receive outgoing resource requests from the UCM, and provide outgoing resource availability responses to the UCM, discloses [2].

### **C. GROUND 1C – Claims 18-25, and 28-32 are obvious over Shankar in view of Huang, Capers, and Janning**

#### **1. Janning**

Janning describes call detail record (CDR) generation and formatting in a telecommunications system. Janning, Abstract. Janning describes first "collecting raw call information for calls," and then "producing a CDR" from the raw call information based on an "optimal data structure" determined from the raw call

information.” Abstract; Sharma, ¶101. A “formatting agent” selects the “optimal data structure” and produces the CDR. Janning describes that its system provides flexibility and efficiency in creating CDRs by using a format based on the raw call information. Janning, Title; 1:43-61; 2:28-59.

Janning teaches disconnecting the call serves as a trigger for generation of the CDR. **“When the call is disconnected,** the unformatted raw information” collected during that call is “transmitted to a formatter 28” where the unformatted raw information is formatted “into a selected data structure commonly referred to as a call detail record (or ‘CDR’).” Janning, 4:4-11 (emphasis added). In other words, the CDR is generated in response to disconnecting the call. Sharma, ¶102.

## **2. Combination of Shankar, Huang, Capers, and Janning**

A POSITA would have been motivated to have Shankar’s call agent (e.g., UCM and related functional components) generate a call detail record in response to disconnecting the call, based on Janning’s suggestions. Shankar describes that its “signaling units perform such functions” as “generation of call detail records (CDRs) for billing management,” although little implementation detail is provided. Shankar, 5:14-32; 10:56-11:3. Janning describes CDR generation that provides flexibility and efficiency, and that uses disconnecting the call as a trigger. The combination of Shankar, Huang, and Capers, in view of Janning, would likewise include a “formatter” as part of the call agent (e.g., as part of the UCM or within

the signaling agent in communication with the UCM) operable to generate a CDR in response to disconnecting the call. Sharma, ¶103.

Multiple reasons would have prompted a POSITA to incorporate Janning's suggestions as described above. First, CDR generation was a known feature that Shankar, itself, teaches its "signaling units" should include, as was the timing of generating the CDR in response to disconnecting the call. *See, e.g.*

V.B.3(Janning's CDR generation); Sestak, pp.1-6 ("generates a CDR at call termination); Brockman, 3:4 ("Typically, the CDR is generated when a call is completed."); Sharma, ¶104. A POSITA would therefore have had a reasonable expectation of success generating CDRs in response to disconnecting the call.

Further, Janning teaches its techniques for generating CDRs provide efficiency and flexibility. Janning, Title; 1:43-61 (describing prior techniques as "inefficient"); 2:28-59. Moreover, information to populate the CDR and select an appropriate format, such as call duration, is available only upon disconnecting the call. Sharma, ¶105. A POSITA would have been led to generate CDRs in accordance with Janning's teachings to avail Shankar's system to such efficiency and flexibility.

Ultimately, a POSITA would have been led to implement Janning's CDR generation in response to disconnecting the call as described above, and would have had a reasonable expectation of success, because doing so would have been

merely the application of a known technique (e.g., CDR generation in response to disconnecting a call) to a known system (e.g., Shankar's system in which the signaling units generate CDRs associated with a call) ready for improvement to yield predictable results. *KSR*, 550 U.S. at 417; Sharma, ¶106. Shankar and Janning both describe CDR generation, and Janning describes a specific manner of doing so in response to disconnecting the call that can promote efficiency and flexibility. It would have been obvious to a POSITA that applying the suggestions of Janning to the system of Shankar, Huang, and Capers would have led to predictable results without significantly altering or hindering the functions performed by Shankar/Huang/Capers' system.

***[18.pre] A softswitch interconnecting networks of different transport and signaling protocols, comprising:***

*Supra*, Ground 1A, [1.pre].

***[18.a] a signaling agent coupled to the networks and operable to receive incoming signaling messages, translate the incoming signaling messages to a call event, and route the call event;***

*Supra*, Ground 1A, [1.a].

***[18.b] a call agent in communication with the signaling agent and operable to receive the call event, verify the validity of incoming circuits of inbound calls, generate a request for an outgoing resources for establishing data sessions;***

As discussed above, Shankar discloses a call agent in communication the signaling agent *operable to receive the call event*, and *generate a request for an outgoing resources for establishing data sessions*. *Supra*, Ground 1A, [1.b].

Moreover, the combination discloses a call agent (e.g., the UCM alone or together with related components), is operable to ***verify the validity of incoming circuits of inbound calls***. For example, Shankar indicates that the UCM can select an “available” bearer channel on the trunk associated with the incoming call setup message. Shankar, 14:4-9 (“upon an available channel on the associated trunk.”); 10:10-14 (“dynamic logic may be set up at run-time based on such factors as channel availability.”). In other words, the UCM is operable to verify the validity of incoming circuits by selecting an “available” bearer channel that is valid to be used for the call. Sharma, ¶107.

Additionally, Shankar discloses that the call agent is operable to verify the validity of incoming circuits of inbound calls by carrying out the connection set-up process with the originating coding unit. The UCM “generates and transmits a create connection message 406, specifying the bearer channel,” to “originating coding unit 110.” Shankar, 14:10-18. In response, the originating coding unit 110 “seizes an endpoint” for the specified bearer channel, “associates a network address for the selected bearer channel,” and “responds back to the originating signaling unit 120 over the control link 114 with a message 408 that includes the network address of the originating coding unit 110.” *Id.*, 14:19-31. These exchanges serve to verify the incoming bearer channel such that a valid bearer channel is identified for use to complete the call. Sharma, ¶108.

Accordingly, the UCM, alone or together with components of the OCC and/or other components of the signaling agent, as taught by Shankar, disclose [18.b].

To the extent [18.b] is narrowly interpreted such that Shankar is considered to not expressly disclose this feature, it would have been obvious to do so in view of predictable combination with Capers.<sup>2</sup> *Supra* IV.B.1-2 (Capers and combination with Capers). In the resulting combination, the UCM would have been operable to verify the validity of incoming circuits by verifying that the incoming circuit has not been “blocked” and is not “unequipped” on the platform, in accordance with Capers’ teachings. Sharma, ¶109. Indeed, such validation is like the preferred embodiment described by the ’441 Patent, which “confirms the call is being processed from a circuit in a valid state” to “prevent processing a call from a circuit that has been administratively locked or through fault management has been disabled.” ’441 Patent, 9:9-13. The UCM would have been operable to verify the

---

<sup>2</sup> Notably, Capers was applied during original prosecution as disclosing this feature. Prosecution History, pp.190-191. Applicant did not dispute that Capers disclosed this feature, and did not dispute the Examiner’s proposed combination with a device interconnecting networks of different transport protocols (Elliott’s soft switch). *Id.*, pp.237-238.



validity of incoming circuits using information from the incoming signaling messages, such as by ensuring inbound messages do not include a circuit identification code (CIC) of a circuit not equipped on Shankar's signaling unit/coding unit system. Capers, 9:18-25.

Accordingly, validating an incoming call to verify the circuit has not been "blocked," and validation based on the circuit identification information in an IAM message, as taught by Shankar, Huang, and Capers, each independently disclose verifying the validity of incoming circuits of inbound calls, as claimed.

***[18.c] a network directory server in communication with the call agent and operable to receive the request for an outgoing resource, and provide information on the outgoing resource;***

*Supra*, Ground 1A, [1.c].

***[18.d] a network gateway agent operable to receive a request to establish a data session on the selected outgoing resource, and set up an open session; and***

The combination discloses [18.d]. Shankar explains that the "signaling unit" controls the "coding unit" over a control link in order to "establish a bearer channel for the voice data." Shankar, 5:14-32; Shankar, 6:28-33. The control messages from the UCM requesting a data session are handled via the "external protocol adapter 314" and "I/O channel control 304 in the I/O subsystem 300." Shankar, 14:10-18 ("UCM 124 generates and transmits a create connection message 406, specifying the bearer channel, through external protocol adapter 314, I/O channel controller 304, control link 114 to originating coding unit 110."); 10:31-55. In

other words, the “external protocol adapter 314,” “I/O channel control 304” (e.g., and/or related components of the UCM) are operable to receive a request to establish a data session from the call agent (e.g., portions of the UCM) and set up an open session by communicating with the coding unit, thereby providing a network gateway agent. Sharma, ¶110.

The configuration and operation of Shankar’s UCM, “external protocol adapter 314,” and “I/O channel controller 304” to control the coding unit and establish a data session is like the call agent and the network gateway agent described by the ’441 patent. ’441 patent, 8:26-29. Likewise, the “external protocol adapter 314” and “I/O channel control 304,” interface with the coding unit (e.g., media gateway) to set up data sessions using the bearer channels and other resources indicated by the UCM. Sharma, ¶111.

Accordingly, the “external protocol adapter 314” and “I/O channel control 304,” alone or together with related components of the UCM, as taught by Shankar, disclose [18.d].

***[18.e] the call agent operable to terminate the data sessions on the requested outgoing resources, and generate a call detail record in response to disconnecting the data session.***

The combination discloses a ***call agent operable to terminate the data sessions on the requested outgoing resources.*** For example, Shankar describes that, after communicating with the originating coding unit to establish an incoming

bearer channel (e.g., for voice transmission), the UCM of the originating soft switch resolves the route to a terminating signaling unit. Then, “[o]nce the appropriate terminating signaling unit 140 is identified, TCC 126 of the originating signaling unit 120 generates and transmits an XISUP IAM message 410 to the terminating signaling unit 140 with the standard ISUP IAM information plus the Universal Call Reference and **the network address of the bearer channel on the originating coding unit 110.**” Shankar, 14:33-45(emphasis added); Abstract (“establish a bearer channel . . . for carrying the voice data.”); *see also* 15:64-16:7 (“the TCC 506 is responsible for communicating the signaling data with the terminating coding unit 150 over link 152.”); FIGS. 5a-c. The IAM message is based on the universal protocol message received from the UCM, and provides call set-up information that leads to termination of the call on the requested outgoing resources. Sharma, ¶112.

Additionally, the “create connection message” generated by the UCM specifying the bearer channel (e.g., “requested resource”) discloses that the UCM is operable to terminate the data sessions on the requested outgoing resources. The message leads to a complete connection used to transmit voice/data to terminate the call. Sharma, ¶112.

The combination also discloses the call agent is operable to ***generate a call detail record in response to disconnecting the data session.*** Doing so was

conventional, and a POSITA would have been led to implement CDR generation in this manner in Shankar's system, as taught by Janning. *Supra* IV.C.1-2. In the resulting system, the "formatter" would have been considered part of the call agent, and would have generated a CDR *in response to* disconnecting the data session, in accordance with the teachings of Shankar, Huang, Capers, and Janning. Sharma, ¶113.

Accordingly, the combination discloses [18.e].

***[19] The softswitch, as set forth in claim 18, further comprising a resource manager operable to receive outgoing resource requests from the call agent, and provide outgoing resource availability responses to the call agent.***

*Supra*, Ground 1B, [2].

***[20] The softswitch, as set forth in claim 18, wherein the network directory server is further operable to receive requests for address resolution to establish data sessions from the call agent, and provide address resolution responses to the call agent.***

*Supra*, Ground 1A, [3.d].

***[21.a] The softswitch, as set forth in claim 18, wherein the signaling agent comprises: a logic control executing a logic control program and operable to process signaling messages of a particular signaling protocol;***

*Supra*, Ground 1A, [5.d].

***[21.b] a codec specialized in the signaling protocol of an access network and operable to parse and format signaling messages according to the signaling protocol; and***

*Supra*, Ground 1A, [5.e].

***[21.c] a filter operable to filter and route signaling messages from the codec to the logic control.***

*Supra*, Ground 1A, [5.f].

***[22] The softswitch, as set forth in claim 21, wherein the signaling protocol is SS7.***

*Supra*, Ground 1A, [6.a].

***[23] The softswitch, as set forth in claim 21, wherein the signaling protocol is session initiation protocol.***

*Supra*, Ground 1A, [7.a].

***[24] The softswitch, as set forth in claim 21, wherein the signaling protocol is H.323.***

*Supra*, Ground 1A, [8].

***[25.a] The softswitch, as set forth in claim 18, wherein the network gateway comprises: a logic control executing a logic control program and operable to process messages of a particular transport protocol;***

*Supra*, Ground 1A, [9.d].

***[25.b] a codec specialized in the transport protocol of a media gateway and operable to parse and format the messages according to the transport protocol; and***

*Supra*, Ground 1A, [9.e].

***[25.c] a filter operable to filter and route signaling messages from the codec to the logic control.***

*Supra*, Ground 1A, [9.f].

***[28] The softswitch, as set forth in claim 25, wherein the transport protocol is simple gateway control protocol.***

*Supra*, Ground 1A, [12].

***[29] The softswitch, as set forth in claim 18, wherein the networks comprise a public switched telephone network.***

*Supra*, Ground 1A, [13].

***[30] The softswitch, as set forth in claim 18, wherein the networks comprise a packet network.***

*Supra*, Ground 1A, [14].

***[31] The softswitch, as set forth in claim 18, wherein the networks comprise a wireless network.***

*Supra*, Ground 1A, [15].

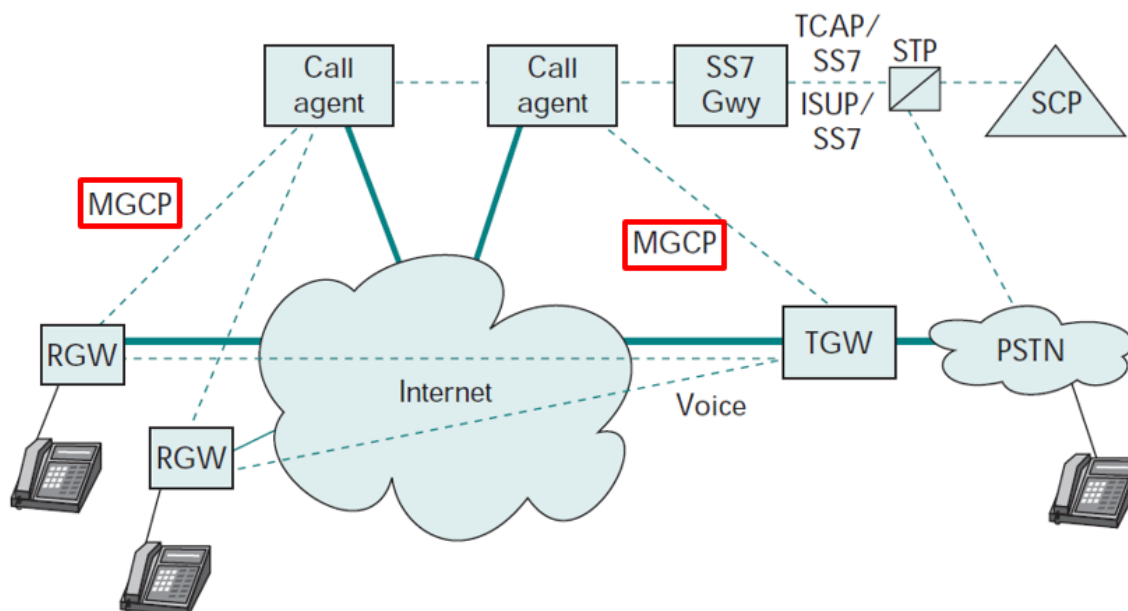
***[32] The softswitch, as set forth in claim 18, wherein the call agent comprises a protocol-independent logic engine operable to execute a function-specific logic control program.***

*Supra*, Ground 1A, [16].

**D. GROUND 1D –Claims 4 and 10 are obvious in view of Shankar, Huang, and Huitema.**

**1. Huitema**

Huitema describes a telephony system that interconnects PSTN and Internet (IP) networks. The system includes media gateways (e.g. residential (RGW), trunking (TGW), SS7, etc.) that translate voice communication between “PSTN and Internet telephony voice formats.” Huitema, p.1-3. “Call agents” handle call signaling from trunks that interconnect the PSTN and the IP network, and use “MGCP for controlling the residential and trunking gateways.” Huitema, pp.2-3.



**Figure 1. Proposed network architecture. Call agents use the Media Gateway Control Protocol to control residential and trunking gateways.**

Sharma, ¶114; Huitema, Figure 1(annotated).

The “call agent controls both residential gateways and trunking gateways via the MGCP.” Huitema, p.3; p. 5 (“merger of” SGCP and IPDC protocols).

Multiple call agents may be used in a single call to determine a route and complete the call when different call agents service the origination and termination of the call, for example. “If the two RGWs participating in a two-party call are controlled by different call agents, the agents will need to coordinate to set up the call.” *Id.* The call agents have an interface to communicate with one another (e.g., using ISUP) “so that information about the IP addresses, ports, and codecs” for the “originating and terminating party can be exchanged between the two call agents.” *Id.* Indeed, “each gateway participating in an Internet telephony call must have”

this information, and “call agents are responsible for ensuring that this is so.” *Id.*; Sharma, ¶115.

## **2. Combination of Shankar, Huang, and Huitema**

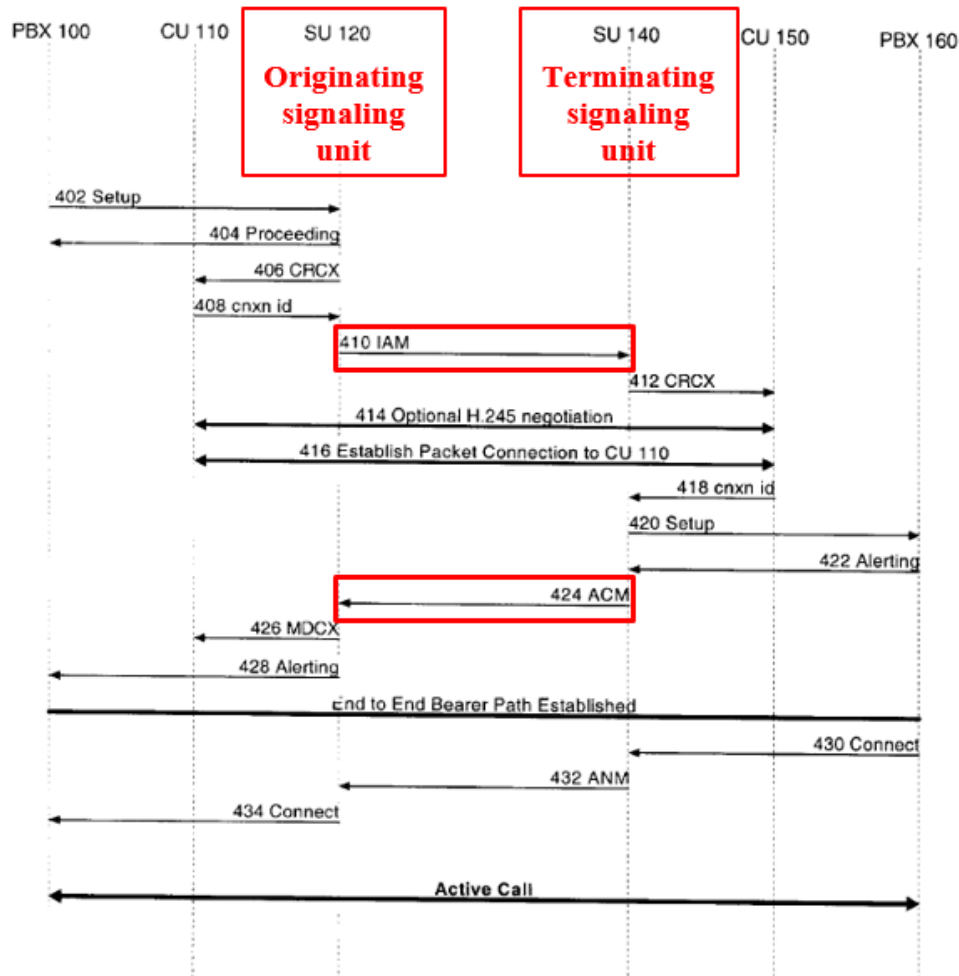
A POSITA would have been motivated to combine Huitema with Shankar and Huang to improve the system in at least two ways. First, a POSITA would have been motivated to use MGCP to communicate between the signaling agent and coding agent. Shankar already describes use of SGCP, which was the predecessor of MGCP. Shankar, 14:13-18 (“Simple Gateway Control Protocol (SGCP).”); *id.*, 9:39-47. MGCP was a combination of the IPDC and SGCP protocols. Huitema, p.5; MGCP Specification, 126 (“MGCP version 0.1 results from the fusion of the SGCP and IPDC proposals.”).

Based on the disclosures of Shankar and Huitema, a POSITA would have found it obvious to use MGCP (e.g., in place of SGCP), in order to use the newest protocol and keep current. Moreover, a POSITA would have been motivated to utilize the advantages provided by the later protocol (e.g., as compared to the earlier developed SGCP), and to obtain additional functionality and enhancements included in MGCP (e.g., such as functionality of the IPDC protocol added to the existing SGCP protocol). *Id.*; Sharma, ¶116.

Second, Huitema would have led a POSITA to use a network gateway that interfaces between signaling units to exchange routing information for call set-up.



Shankar describes a preferred embodiment that uses “originating” and “terminating” signaling units 120, 140 in order to determine a route a complete a call to a destination in a network external to that serviced by the “originating” signaling unit. “Originating” signaling unit 120 selects the incoming bearer channel, but not having access to routing information to select the terminating bearer channel, the “originating” signaling unit communicates with the “terminating” signaling unit which in turn resolves routing to the terminating node. Shankar, 14:4-45; Sharma, ¶117. “TCC 126 of the originating signaling unit 120 generates and transmits an XISUP IAM message 410 to the terminating signaling unit 140.” *Id.*, 14:38-45. After setup of the outgoing routing by the “terminating” signaling unit 140, the “terminating signaling unit 140” transmits an “XISUP ACM message 424 to the originating signaling unit 120” that contains “ISUP ACM information” and “the network address of the terminating coding unit 150,” among other information useful for call setup and completion. In other words, the TCC (e.g., including the XISUP interface) that generates, transmits, and receives messages between “originating” and “terminating” signaling units provides a “network gateway.”



Sharma, ¶117; Shankar, FIG. 4(annotated).

Huitema provides additional details of the communication between signaling units/call agents during call set-up. When calling a destination external to an originating call agent, the originating call agent must communicate with a terminating call agent that services the destination party: “If the two RGWs participating in a two-party call are controlled by different call agents, the agents will need to coordinate to set up the call,” so that “information about the IP addresses, ports, and codecs . . . for the originating and terminating party can be

exchanged between the two call agents.” Huitema, p.4. “[E]ach gateway participating in an Internet call must have” this information, and “the call agents are responsible for ensuring that this is so.” *Id.*

Accordingly, based on the disclosures of Shankar, Huang, and Huitema, it would have been obvious for the UCM of an “originating” signaling unit to request address locations needed to reach the called party via the unit of the TCC that interfaces with a “terminating” signaling unit, and to receive the address locations in response so that the call could be set-up. Sharma, ¶¶118-119. The combination would have promoted effective communication between an “originating” signaling unit and a “terminating” signaling unit that supports an external network so that a call could be completed to the external network. Moreover, such features facilitate a “large network” that includes a “very large number” of gateways supported by multiple signaling units/call agents—an advantage explicitly described by Huitema. Huitema, p.4. Such a configuration facilitates the ability of users to call a large number of destination numbers. Moreover, a POSITA would have found it obvious that such a configuration promotes reliability and scalability. Huitema, p.2; Sharma, ¶120.

Ultimately, a POSITA would have been led to implement MGCP in Shankar/Huang’s system, and would have had a reasonable expectation of success, because doing so would have been merely the application of a known technique

(e.g., using MGCP to communicate between a call agent responsible for call set-up and a media gateway) to a known system (e.g., Shankar's system that uses SGCP to communicate between a signaling agent responsible for call set-up and coding unit) ready for improvement to yield predictable results. *KSR*, 550 U.S. at 417; Sharma, ¶121. Likewise a POSITA would have been led to interface between signaling units using a network gateway that can receive requests for address locations and respond with address location information, and here again would have had a reasonable expectation of success, because doing so would have been merely the application of a known technique (e.g., communication between originating and terminating signaling units/call agents using an appropriate interface) to a known system (e.g., Shankar's system in which originating and terminating signaling units communicate routing information during call set-up) ready for improvement to yield predictable results. *KSR*, 550 U.S. at 417.

### **3. Application of Shankar, Huang, and Huitema to the Challenged Claims**

***[4.pre] A softswitch interconnecting networks of different transport protocols, comprising:***

*Supra*, Ground 1A, [1.pre].

***[4.a] a signaling agent coupled to the networks and operable to receive incoming signaling messages, translate the incoming signaling messages to a call model event, and route the call model event; and***

*Supra*, Ground 1A, [1.a].

***[4.b] a call agent in communication with the signaling agent and operable to receive the call model event, request outgoing resources for establishing data sessions, generate outgoing signaling messages, and send the outgoing signaling messages to the signaling agent,***

*Supra*, Ground 1A, [1.b].

***[4.c] the signaling agent further terminating the data sessions on the requested outgoing resources; and***

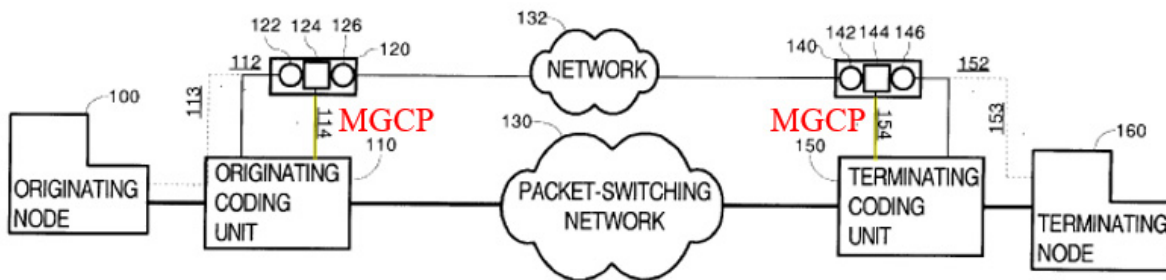
*Supra*, Ground 1A, [1.c].

***[4.d] further comprising a network gateway operable to receive request for address locations of called parties in external networks to establish data sessions from the call agent, and provide the address locations to the call agent.***

The combination discloses [4.d]. *Supra*, IV.D.1-2 (Huitema and combination of Huitema/Shankar/Huang). To the extent claim 4 is narrowly interpreted such that Shankar is considered to not expressly disclose this feature, it would have been obvious to employ a network gateway to interface with a terminating signaling unit, such that the network gateway could receive requests for address locations associated with a called party in an external network (e.g., from the portions of the UCM handling call-setup), and provide the address locations in response. *Id.*; Sharma, ¶122 Indeed, Huitema explicitly teaches signaling units/call agents exchange this information to complete a call to an external called party. Accordingly, the portion of Shankar's TCC that interfaces with another signaling unit, in accordance with Huitema's teachings, discloses [4.d].

**[10] The softswitch, as set forth in claim 9, wherein the transport protocol is media gateway control protocol.**

The combination discloses [9]. *Supra* IV.D.1-2 (Huitema and combination of Huitema/Shankar/Huang). The result would have included MGCP to communicate between the signaling agent and coding unit (e.g., over control link 114).



Sharma, ¶123; Shankar, FIG. 1(annotated).

**E. GROUND 1E—Claim 26 is obvious in view of Shankar, Huang, Capers, Janning, and Huitema.**

**1. Combination of Shankar, Huang, Capers, Janning, and Huitema**

A POSITA would have been led to combine these references for the same reasons set forth above in IV.A-D, *supra*. Sharma, ¶124.

**[26] The softswitch, as set forth in claim 25, wherein the transport protocol is media gateway control protocol.**

*Supra*, Ground 1D, [10].

**V. CONCLUSION**

The challenged claims are unpatentable as set forth in Grounds 1A-1E.

## **VI. PAYMENT OF FEES – 37 C.F.R. § 42.103**

Petitioner authorizes charge to Deposit Account No. 06-1050 for the necessary fees.

## **VII. MANDATORY NOTICES UNDER 37 C.F.R § 42.8(a)(1)**

### **A. Real Party-In-Interest Under 37 C.F.R. § 42.8(b)(1)**

In addition to Petitioners, DC Holdco SARL and Data Connection Second Employee Benefit Trust are being identified as real parties-in-interest.

### **B. Related Matters Under 37 C.F.R. § 42.8(b)(2)**

Petitioner is not aware of any disclaimers, reexamination certificates or petitions for inter partes review for the '441 Patent. The '441 patent is also the subject of IPR2019-00396. The '441 patent is the subject of a number of civil actions including: *Sonus Networks Inc. v. Metaswitch Networks Ltd. et al.* 2:18-cv-00058 E.D. Texas.

### **C. Lead And Back-Up Counsel Under 37 C.F.R. § 42.8(b)(3)**

Metaswitch provides the following designation of counsel.

<b>Lead Counsel</b>	<b>Backup counsel</b>
Joshua A. Griswold, Reg. No. 46,310 Fish & Richardson P.C. 3200 RBC Plaza 60 South Sixth Street Minneapolis, MN 55402 Tel: 214-747-5070 Fax: 877-769-7945 Email: <a href="mailto:IPR42360-0004IP3@fr.com">IPR42360-0004IP3@fr.com</a>	David B. Conrad, Reg. No. 60,788 Craig A. Deutsch, Reg. No. 69,264 Fish & Richardson P.C. 3200 RBC Plaza, 60 South Sixth Street Minneapolis, MN 55402 Tel: 214-747-5070 Fax: 877-769-7945 <a href="mailto:PTABInbound@fr.com">PTABInbound@fr.com</a>

**D. Service Information**

Please address all correspondence and service to the address listed above.

Petitioner consents to electronic service by email at [IPR42360-0004IP3@fr.com](mailto:IPR42360-0004IP3@fr.com)

(referencing No. 42360-0004IP3 and cc'ing [PTABInbound@fr.com](mailto:PTABInbound@fr.com),

[griswold@fr.com](mailto:griswold@fr.com), [conrad@fr.com](mailto:conrad@fr.com) and [deutsch@fr.com](mailto:deutsch@fr.com).

Respectfully submitted,

Dated December 5, 2018

/Joshua A. Griswold/

Joshua A. Griswold, Reg. No. 46,310

David B. Conrad, Reg. No. 60,788

Craig A. Deutsch, Reg. No. 69,264

Fish & Richardson P.C.

3200 RBC Plaza, 60 South Sixth Street

Minneapolis, MN 55402

T: 214-747-5070

F: 877-769-7945

(Control No. IPR2019-00395)

Attorneys for Petitioner



**CERTIFICATION UNDER 37 CFR § 42.24**

Under the provisions of 37 CFR § 42.24(d), the undersigned hereby certifies that the word count for the foregoing Petition for *Inter partes* Review totals 13,969 words, which is less than the 14,000 allowed under 37 CFR § 42.24.

Dated December 5, 2018

/Joshua A. Griswold/  
Joshua A. Griswold, Reg. No. 46,310  
David B. Conrad, Reg. No. 60,788  
Craig A. Deutsch, Reg. No. 69,264  
Fish & Richardson P.C.  
3200 RBC Plaza, 60 South Sixth Street  
Minneapolis, MN 55402  
T: 214-747-5070  
F: 877-769-7945

Attorneys for Petitioner

**CERTIFICATE OF SERVICE**

Pursuant to 37 CFR §§ 42.6(e)(4)(i) *et seq.* and 42.105(b), the undersigned certifies that on December 5, 2018, a complete and entire copy of this Petition for *Inter partes* Review and all supporting exhibits were provided via Federal Express, to the Patent Owner by serving the correspondence address of record as follows:

Straub & Straub - Sonus Correspondence  
788 Shrewsbury Avenue  
Tinton Falls NJ 07724

/Diana Bradley/  
Diana Bradley  
Fish & Richardson P.C.  
60 South Sixth Street, Suite 3200  
Minneapolis, MN 55402  
(858) 678-5667