UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.

Petitioner,

v.

VOIP-PAL.COM, INC.,

Patent Owner

Case No. IPR2016-01201
U.S. Patent 8,542,815

DECLARATION IN SUPPORT PATENT OWNER RESPONSE TO INTER PARTES PETITION

Voip-Pal Ex. 2016
IPR2016-01201

META 1017
META V. VOIP-PAL

GOOGLE EXHIBIT 1017

I, William Henry Mangione-Smith, declare as follows:

I. INTRODUCTION

1. During the preparation of this Declaration, I have reviewed the following, as well as the other documents discussed in this Declaration:

a. U.S. Patent No. 8,542,815 ("the '815 Patent") and its file history;

b. U.S. Patent No. 9,179,005 ("the '005 Patent") and its file history;

c. Apple, Inc.'s Petition For *Inter Partes* Review Of U.S. Patent No. 8,542,815, including attached Declaration Of Henry H. Houh, PhD;

d. Apple, Inc.'s Petition For *Inter Partes* Review Of U.S. Patent No. 9,179,005, including attached Declaration Of Henry H. Houh, PhD;

e. Declaration of Ryan Purita (**Ex. 2011**);

f. Declaration of Johan Emil Viktor Bjorsell (**Ex. 2012**);

g. Declaration of Clay Perreault (**Ex. 2013**);

h. Declaration of David Terry (**Ex. 2018**);

i. Deposition Transcript of Henry H. Houh, Ph.D. Volumes I and I (**Ex. 2043, 2044**)

j. Subversion repository svn.tar file;

k. U.S. Patent No. 7,486,684 to Chu ("the Chu '684 Patent");

l. U.S. Patent No. 8,036,366 to Chu ("the Chu '366 Patent");

m. U.S. Patent Publication No. 2007/0064919 to Chen at al. ("the Chen Patent").

2. I have been retained by Voip-PAL.Com Inc. ("Voip-PAL") as an expert in the fields of computer science, computer communications, and related technologies. I am being compensated at my normal consulting rate. My compensation is not dependent on and in no way affects the substance of my statements in this Declaration. I have no financial interest in Voip-PAL, in the '815 Patent or in the '005 Patent.

II. QUALIFICATIONS

3. My technical background covers most aspects of computer system design, including low level circuitry, computer architecture, computer networking, graphics, application software, client-server application, Web technology, and system software (e.g., operating systems and compilers). I am a member of the Institute of Electrical and Electronics Engineers and the Association for Computing Machinery, which are the two most significant professional organizations in my profession. I have been employed as a design engineer, research engineer, professor and technical expert. Over my professional career I have been an active inventor with 109 issued U.S. patents, 196 published and pending U.S. patent applications and many unpublished U.S. patent applications.

4. From 1984 until 1991 I attended the University of Michigan in Ann Arbor, Michigan. I was awarded the degrees of Bachelor of Science and Engineering, Master of Science and Engineering, and Doctorate of Philosophy. My doctoral research focused on high performance computing systems including computer architecture, applications and operating system software, and compiler technology. One of my responsibilities during my graduate studies included teaching senior undergraduate students who were about to enter the profession.

5. After graduating from the University of Michigan I was employed by Motorola in Schaumburg, Illinois. While at Motorola, I was part of a team designing and manufacturing the first commercial battery-powered product capable of delivering Internet email over a wireless (i.e., radio frequency) link and one of the first personal digital assistants. I also served as the lead architect on the second-generation of this device. Part of my responsibilities at Motorola involved the specification, design, and testing of system control Application-Specific Integrated Circuits ("ASICs"). I conducted the initial research and advanced design that resulted in the Motorola M*Core embedded microprocessor. M*Core was designed to provide the high performance of desktop microprocessors with the low power of contemporaneous embedded processors. The M*Core received widespread use in a number of communications products including various

telephonic handsets, advanced pagers, and embedded infrastructure. While at Motorola I was the sole inventor on one U.S. patent.

6. From 1995 until 2005 I was employed by the University of California at Los Angeles ("UCLA") as a professor of Electrical Engineering. I was the director of the laboratory for Compiler and Architecture Research in Embedded Systems ("CARES") and served as the field chair for Embedded Computing Systems. The CARES research team focused on research, engineering and design challenges in the context of battery-powered and multi-media mobile computing devices. One of the key developments of my lab was the Mediabench software tool, which is widely used to design and evaluate multi-media embedded devices. Key elements of Mediabench include software that is essential for modern digital telephony. My primary responsibility, in addition to classroom teaching, involved directing the research and training of graduate students. I was a tenured member of the faculty, and had responsibilities for teaching as well as scholarly research. While at UCLA I was a named inventor on three U.S. patent applications, one of which issued as a patent. My colleagues at UCLA were some of the leading scientists and engineers in the world with a long list of innovations from computer network security devices to the nicotine patch. The graduate student researchers in my laboratory came from a diverse set of backgrounds, all with undergraduate degrees in computer engineering, electrical engineering or computer science, many

with multiple years of experience working as professional engineers in areas such as software development, computer system design and ASIC circuit design.

7. From 2005 until 2009, I was employed at Intellectual Ventures in Bellevue, Washington. My responsibilities at Intellectual Ventures included business development, technology assessment, market forecasting, university outreach, collaborative inventing, intellectual property licensing support, and intellectual property asset pricing. My colleagues and co-inventors at Intellectual Ventures included the former lead intellectual property strategist at Intel, Intel's lead IP council, Microsoft's chief software architect, the founder of Microsoft research, the designer of the Mach operating system, the architect of the U.S. Defense Department's Strategic Defense Initiative, the founder of Thinking Machines (a seminal parallel processing computer system), and Bill Gates. I had responsibility for hiring and managing over 15 staff members including multiple Ph.Ds. with degrees in electrical engineering and decades of experience in product design and engineering.

8. A brief summary of some of my qualifications for forming the opinions in this declaration are as follows: I have more than 27 years of experience as a computer architect, computer system designer, educator, and as an executive in the PC and electronics business. I am also a member of a number of professional associations, such as the ACM, IEEE and have been intimately

involved in professional research through the International Symposium on Microarchitecture (Program Chair for 26th and General Chair for 36th), IEEE Transactions on Computers (Associate Editor), ACM Transactions on Embedded Computing Systems (Associate Editor), and IEEE Computer (Associate Editor). I also have been on the program committees for ISCA, MICRO, ISLPED, Network Processors Workshop, FPL, Complexity-Effective Design, RAW, Workshop on Mediaprocessors, and DSP, FPT, and INTERACT.

9. For further details regarding my employment and academic history, please refer to my curriculum vitae (**Ex. 2045**).

III. RELEVANT LAW

10. I am not an attorney and I have no formal legal training. For the purposes of this Declaration, I have been informed about certain aspects of the law that are relevant to my opinions. My understanding of these aspects is summarized below.

11. I have been informed and understand that at the time of invention for the patents in question the United States employed a principle known as the "first to invent". It is my understanding that this principle gives priority to a first inventor who can convincingly claim to have invented a concept earlier than a second inventor even if circumstances indicate that the second inventor filed for patent protection earlier than the first inventor.

12. It is my understanding that an inventor can substantiate a date earlier than a patent filing date for purpose of priority and prior art by providing convincing evidence that the inventor had actually reduced the claimed invention to practice at a date prior to filing for patent protection. It is my understanding that the phrase "actual reduction to practice" means that the inventor had either practiced method claims or held in their possession a working artifact that practiced method or apparatus claims, and that the inventor was aware that the working artifact worked for its intended purpose.

13. It is my understanding that with regards to consideration of prior art for a particular patent U.S.C. 102(e) states:

> *(e) the invention was described in — (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for the purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.*

IV. LEVEL OF ORDINARY SKILL IN THE ART

14. Based on my review of the ’815 Patent and the ‘005 Patent and my background and experience in the field of computer science, it is my opinion that one of ordinary skill in the art as of the priority date would be someone with an undergraduate degree in either Computer Science, Computer Engineering, Electrical Engineering, or a closely related discipline. Furthermore, I believe that such a person would also have 2 years of experience in system-level software development. In my opinion a greater degree of professional experience could serve to replace some degree of formal education. I also believe that some greater degree of formal education could serve to replace some degree of professional work experience.

15. Based on my education and experience, I am familiar with the level of knowledge that one of ordinary skill would have possessed during the relevant period of time. By the priority date of the ‘815 Patent and the ‘005 Patent, I was a person who had more than ordinary skill in the art.

V. OVERVIEW OF THE ‘815 AND ‘005 PATENTS

16. The ‘815 and ‘005 Patents disclose a method and apparatus for routing communications, such as voice calls or video calls, between an internet network (i.e. a network based on the Internet Protocol, or IP) and a public switched telephone network (PSTN), and between two endpoints on an IP network. The

claimed routing methods and apparatus allow a call to be classified and routed as a "public network call" or as a "private network call" based on whether at least one calling attribute and at least a portion of the callee identifier meet certain network criteria.

17. A key aspect of the claimed inventions disclosed in the '815 and '005 patent is that they relate to specific calling individuals. The shared written description and claims make frequent use of the term "caller" and "callee". The specification make it apparent that a caller is an individual who initiates a call and a callee is an individual the caller wishes to communicate with. For example, the '815 Patent associates a caller or callee with a user (see for example Fig. 9, 10, '815@2:4-7, '815@2:48-50, '815@2:68-61, '815@3:1-2, '815@3:40-46, '815@15:5-23, '815@15:41-61, and '815@16:37-51). The detailed description also makes use of the word "subscriber" to refer to callers, i.e. individuals who subscribe to the claimed system and can initiate calls to designated callees. Thus, it is my belief that a person of ordinary skill in the art would understand that the '815 and '005 patents are directed to individual people (i.e. users or subscribers) who initiate calls to other individual people.

18. I have been asked to analyze Claims 1, 7, 27, 28, 34, 54, 72-74, 92, 93 and 111 of the '815 Patent. I have also been asked to analyze Claims 1, 24-26, 49, 50, 73-79, 83, 84, 88, 89, 92, 94-96, 98 and 99 of the '005 patent.

VI. CLAIM CONSTRUCTION ANALYSIS

19. I have been asked to apply the Petitioner's claim construction in my analysis for this opinion. While I have accepted these proposed constructions on their face for purposes of my analysis I have not formed an opinion as to the correct construction for each term.

VII. SOURCE CODE ANALYSIS

20. I have been asked to analyze version 361 of the RBR software. I have been provided with a file named "svn.tar", which contains a Subversion source code repository. Subversion is a commonly used tool to maintain and record changes to source code. A source code repository records the history of changes that are made to each file over time. I have reviewed the Declaration of Ryan Purita (**Ex. 2011**) regarding the file named "svn.tar" that he analyzed and I have confirmed that the file that I received has the same signatures identified in Mr. Purita's declaration for the algorithms MD5 (7c79ec3a4bd14ceaa0f8b8c0d0d3e483), SHA1 (87abf73113707e4262c24ffe663afe003742ed2b), CRC32 (954c203e), and SHA256 (4dbb53e51e68df305859181e34c1a17349c104190e98008946f2b0de1b577ebc). MD5 is an algorithm that creates a signature from a data file that is claimed to be uniquely associated with that file to the same degree as a human fingerprint is to an

individual person. CRC32 is like MD5 though not as robust. SHA-1 and SHA-256 are cryptographic one-way hash functions with no known collisions. Based on these it is virtually certain that the file I have is identical to the one that was analyzed by Mr. Purita. I also captured the checksum for all the php source files in the src directory using the following command from within the source directory: "cat * | shasum -p -a 256". This command produced the following output "214b815b4c8652f1d38d3a8e6c744aa7a28545eaa9fc28ef041737f1120ef9af ?- ".

21. I have extracted the Subversion repository from the svn.tar file, and I then extracted version "361" of the "rbr" directory. I utilized the "tar" and "svn" utilities that were available on my own computer to perform these functions. The Subversion repository indicates that Version 361 of the RBR software was last modified on June 6, 2005 at 09:22:59am.

22. Version 361 of the RBR software consists a number of PHP files that are run on a server and communicate using an XMLRPC interface. The invite.php file is the entry point for the messages that are sent from the B2BUA server. The functions in the invite.php file rely upon functions in the call_routes.class.php file, which in turn rely upon functions in the call_ttl.class.php file, which in turn uses code in the call_e164.class.php file. The source code in the call_routes.class.php file and the call_e164.class.php file also utilizes a configuration file named

config.php, a sample of which is provided in the file named config-dist.php in the same directory.

23. I have prepared in **Ex. 2014** printouts of the files that make up Version 361 of the RBR software. I have also prepared a log of the files in the "trunk" directory up through November 2, 2006 in **Ex. 2015**. The logs were generated by first updating to the source code as it existed on November 2, 2006 using the "svn update –r 1879" command and then using the "svn log -v" command from within the trunk directory. This shows the log messages for all RBR source files within that directory up through November 2, 2006.

24. Based upon my review of the source code provided it is my opinion that an RBR server running RBR source code Version 361 would implement a call routing controller, which corresponds to the Routing Controller 16 illustrated in Fig. 1 of the '815 Patent, that received messages when a call was initiated, processed those messages and responded with a routing message. In the tables below I have indicated how the RBR server source code maps to certain claims of the '815 Patent.

Patent 8,542,815	**RBR Source Code Version 361**
1. [1p] A process for operating a call routing controller to facilitate communication between callers and callees in a system comprising a	The RBR server implements a call routing controller. Callers and callees, which may be internet (IP) phones or endpoints on the PSTN, are associated with nodes. Within a Digifonica supernode, the RBR server facilitates communication by responding to requests from the B2BUA server and providing routing

plurality of nodes with which callers and callees are associated, the process comprising:	messages back to the B2BUA server.

[1a] in response to initiation of a call by a calling subscriber, receiving a caller identifier and a callee identifier;	In response to a SIP INVITE message from an IP phone, the B2BUA server sends a USER_INVITE message to the RBR server.
	
	The inviteClient() function is the entry point for the RBR software that receives this message. The inviteClient function receives five parameters in an XML_RPC_Message (**Ex. 2014 at 42**, "invite.php" line 27):
	
	`* This function is called by the XML-RPC Server expecting type (integer * like 4 is THIRD_PARTY_INVITE) ,username,callee,caller,callid (In that * order in the XML_RPC_Message)`
	
	A type value of 2 represents a "USER_INVITE" message (**Ex. 2014 at 42**, "invite.php" line 35):
	
	`$type_arr[2] = 'USER_INVITE';`
	
	Besides the type parameters, the USER_INVITE message contains four other parameters identified as "username", "callee", "caller" and "called" (**Ex. 2014 at 42**, "invite.php" lines 44-47):
	
	`$username = $params->getParam(1)->scalarval();`
	`$callee = $params->getParam(2)->scalarval();`
	`$caller = $params->getParam(3)->scalarval();`
	`$callid = $params->getParam(4)->scalarval();`
	
	The "caller" variable corresponds to the caller identifier and the "callee" parameter corresponds to the callee

	identifier.

[1b] locating a caller dialing profile comprising a username associated with the caller and a plurality of calling attributes associated with the caller;	The RBR server loads a calling profile using the "caller" parameter.
	
	The inviteClient() function first constructs a new call_routes() object (**Ex. 2014 at 43**, "invite.php" line 60):
	
	`$call = new call_routes($caller,$caller_domain,$callee, $callee_domain,$type,$forced_cid);`
	
	The constructor for call_routes() is defined at lines 59-74 of call_routes.class.php (**Ex. 2014 at 19**):
	
	`/** * Constructor... the class must be called with the defined parameters. Callee domain is optional and the class will try to guess it. * Special is a flag (THIRD_PARTY_INVITE) when set, the caller is not set and concidered a incoming PSTN call. * * @param String $caller * @param String $domain * @param String $callee * @param String $callee_domain * @param String $special * @return call_routes */ function call_routes($caller,$domain,$callee,$callee_domain='',$special='') { $this->special_request = $special; $this->call_ttl($caller,$domain,$callee,$callee_domain); $this->genereate_routes(); }`
	
	The call_routes() constructor calls the call_ttl() function (**Ex. 2014 at 19**, "call_routes.class.php" line

72):

```
$this-
>call_ttl($caller,$domain,$callee,$callee_d
omain);
```

The call_ttl() constructor is defined at lines 131-153 of call_ttl.class.php (**Ex. 2014 at 27**):

```
/**
 * Class constructor must be initiated with
at least caller, domain (callee_domain and
special flag are optional.)<br>
 * Once it is initiated, it will
automatically execute proper functions and
do its due diligence, to create TTL.
 * $this->ttl is where you can get the
Total time to live for this call. can be
accessed byt extending classes and <br>
 * through the function get_ttl().
 *
 * @param string $caller
 * @param string $domain
 * @param string $callee
 * @param string $callee_domain
 * @param string $special
 * @return call_ttl
 */
function
call_ttl($caller,$domain,$callee,$callee_do
main='',$special='') {
    if(!$callee_domain){
        //try to guess
        if($temp_domain = $this-
>guess_user_domain($callee)) $this-
>callee_domain=$temp_domain;
    }
    $this-
>set_caller_data($caller,$domain,$special);

    $this->set_callee($callee,$this-
>callee_domain);
    $this->generate_ttl();
}
```

The call_ttl() function calls the set_caller_data()

function (**Ex. 2014 at 27**, “call_ttl.class.php” line 149):

```
$this-
>set_caller_data($caller,$domain,$special);
```

The set_caller_data() function is defined at lines 187-189 of call_ttl.class.php and calls the set_caller() function (**Ex. 2014 at 28**, “call_ttl.class.php” line 188):

```
$this->set_caller($caller,$domain);
```

The set_caller() function is defined at lines 325-340 of call_e164.class.php (**Ex. 2014 at 7**):

```
/**
 * Takes care of sequence of execution to
create a caller profile
 *
 * @param String $caller
 * @param String $domain
 */
public function
set_caller($caller,$domain){

    $this->caller = $caller;

    if(! ($this->special_request ==
'THIRD_PARTY_INVITE')){
        $this->caller_domain = $domain;
        $this->create_caller_profile();
    }
    $this->caller_is_set = true;
}
```

The set_caller() function calls the create_caller_profile() function (**Ex. 2014 at 7**, “call_e164.class.php” line 337):

```
$this->create_caller_profile();
```

The create_caller_profile() function is defined at lines 342-391 of call_e164.class.php (**Ex. 2014 at 7-8**) (lines 342-346 shown):

	```/** * All aspects of the caller Flags and data needed to identify a caller is set here. including overlapping numbers. * */ private function create_caller_profile(){```
	
	The create_caller_profile() function queries an SQL database to load a caller profile (**Ex. 2014 at 7**, "call_e164.class.php" line 348):
	
	```$user_data = mysql_fetch_array($this->sql_query("SELECT * FROM subscriber_dialing_profile WHERE username='$this->caller' and domain= '$this->caller_domain'"));```
	
	The data returned from the SQL query include the caller's idd (international direct dial digits), ndd (national direct dial digits) and area code (**Ex. 2014 at 7**, "call_e164.class.php" lines 350-356):
	
	```$this->caller_idd = $user_data['idd']; $this->caller_ndd = $user_data['ndd']; $this->caller_areacode = $user_data['area_code']; $this->caller_countrycode = $user_data['country_code']; $temp = explode(",",$user_data['local_length']); $this->caller_min_local_length = $temp[0]; $this->caller_max_local_length = $temp[1];```

[1c] determining a match when at least one of said calling attributes matches at least a portion of said callee identifier;	The RBR server matches attributes in the caller profile values against the callee identifier. The call_ttl() constructor, after calling the set_caller_data() function calls the set_callee() function (**Ex. 2014 at 27**, "call_ttl.class.php" line 151): ```$this->set_callee($callee,$this->callee_domain);```

	The set_callee() function is defined at lines 393-418 of call_e164.class.php (**Ex. 2014 at 8-9**) (lines 393-399 shown):
	
	`/**`
	` * Takes care of the sequence and logic needed to create a callee`
	` *`
	` * @param String $callee`
	` * @param String $callee_domain`
	` */`
	`public function set_callee($callee,$callee_domain=''){`
	
	The set_callee() function calls the create_callee_profile() function (**Ex. 2014 at 8**, "call_e164.class.php" line 407):
	
	`$this->create_callee_profile();`
	
	The create_callee_profile() function is defined at lines 450-539 of call_e164.class.php (**Ex. 2014 at 9-11**) (lines 450-454 shown):
	
	`/**`
	` * Creates a callee profile, all aspects of the callee is set here including the e164 number is also set here. NOTE: the e164 number can change when call forwarding occures.`
	` *`
	` */`
	`private function create_callee_profile(){`
	
	The create_callee_profile() function matches values that were previously retrieved as part of the caller profile with the callee identifier. For example, a check is made to see if the callee identifier begins with the international dialing digits (idd) (**Ex. 2014 at 9-10**, "call_e164.class.php" lines 467-476):
	
	`case (isset($this->caller_idd) &&`

	`preg_match("/^".$this->caller_idd."/",$this->callee)):`
	`$idd_length = strlen($this->caller_idd);`
	`$this->set_ml_id($idd_length);`
	`if($this->callee_data['ml_id']){ //callee was found and set`
	`$this->callee_type = 1;`
	`$this->check_callee_length($idd_length);`
	`$this->set_e164_formed_number(substr($this->callee,$idd_length));`
	`}else{`
	`  throw new Exception(4);`
	`}`
	
	Further, create_callee_profile() checks for matching national dialing digits (**Ex. 2014 at 10**, "call_e164.class.php" lines 479-494), checks for matching area codes (**Ex. 2014 at 10**, "call_e164.class.php" lines 495-501), checks for a matching local number (**Ex. 2014 at 10-11**, "call_e164.class.php" lines 506-513), and checks for a network number (**Ex. 2014 at 11**, "call_e164.class.php" lines 520-527).

[1d] classifying the call as a public network call when said match meets public network classification criteria and classifying the call as a private network call when said match meets private network classification criteria;	The RBR server classifies the call as a public network call or a private network call.
	
	The create_callee_profile() function classifies the call by setting the callee_type variable to a value from 1 to 7. In the case of a public classification, the callee_type variable is set to one of the values 1, 2, 3 or 6 (**Ex. 2014 at 10**, "call_e164.class.php" lines 471, 483, 485, 497 and 510)"
	
	`$this->callee_type = 1;`
	`$this->callee_type = 6;`
	`$this->callee_type = 2;`
	`$this->callee_type = 3;`
	`$this->callee_type = 3;`
	
	In the case of a private classification, the callee_type variable is set to one of the values 4 or 7 (**Ex. 2014 at**

11, “call_e164.class.php” lines 523 and 525 respectively):

```
$this->callee_type = 4;
$this->callee_type = 7;
```

The callee_type variable is set to a value of 5 in the case of a failure to classify the call (**Ex. 2014 at 11**, “call_e164.class.php” line 529):

```
$this->callee_type = 5; //fail
```

The set_callee() function calls the check_for_did() function. The check_for_did() function is defined at lines 420-441 of call_e164.class.php (**Ex. 2014 at 9**):

```
/**
 * Checks if the number being called is a
mapped DID in our system, if it is we rout
it internally
 *
 * @param String $user
 * @return Boolean or
DidMap(DigifonicaNumber#Domain)
 */
protected function check_for_did($user=''){
     $num =  $user?$user:$this->callee;
     $did_check = @mysql_fetch_array($this-
>sql_query("SELECT * FROM did_bank WHERE
did='$num'"));
     if($did_check[0]){
          try{
               if($did_check['mapped_to']
and $did_check['domain']){
                    return
$did_check['mapped_to']."#".$did_check['dom
ain'];
               }else{
                    throw new Exception(8);
               }
          }catch (Exception $e){
               $this-
>e164_exception_handler($e->getMessage());
          }
     }
```

	`return false;`
	`}`
	
	The check_for_did() function queries an SQL database to determine if the reformatted callee number is contained in the "did_bank" table. This check determines if the callee is mapped to a Digifonica subscriber account. In the case that the check_for_did() function returns a value, the set_callee() function will make a call to itself at line 415 of call_e164.class.php (**Ex. 2014 at 8**). This will cause a second call to the create_callee_profile() function, which will cause the callee_type to be changed to 4 or 7 depending on whether the destination is associated with the same supernode or a different supernode as the caller.

[1e] when the call is classified as a private network call, producing a private network routing message for receipt by a call controller, said private network routing message identifying an address, on the private network, associated with the callee;	The RBR server produces a private network routing message in the case of a private network call classification.
	
	The call_routes() constructor, after calling the call_ttl() function calls the genereate_routes() function (**Ex. 2014 at 19**, "call_routes.class.php" line 73):
	
	`$this->genereate_routes();`
	
	The genereate_routes() function is defined at lines 91-148 of call_routes.class.php (**Ex. 2014 at 19-21**) (lines 91-95 shown):
	
	`/**`
	` * Generates the routes and the SIP messages including VM data.`
	` *`
	` */`
	`protected function genereate_routes(){`
	
	The genereate_routes() function tests the value of callee_type and in the case of 4 and 7 sets the routes[] array (**Ex. 2014 at 20**, "call_routes.class.php" line 146):

	`/**`
	` * Generates the routes and the SIP messages including VM data.`
	` *`
	` */`
	`protected function genereate_routes(){`
	
	The routes[] array is returned to the inviteClient() function and stored into a "$response" variable (**Ex. 2014 at 43**, "invite.php" line 62):
	
	`$response = $call->get_routes();`
	
	The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php, line 70):
	
	`return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`

[1f] when the call is classified as a public network call, producing a public network routing message for receipt by the call controller, said public network routing message identifying a gateway to the public network.	The RBR server produces such a public network routing message in the case of a public network call classification. The genereate_routes() function tests the value of callee_type and in the case of 1, 2, 3 or 6 queries an SQL database to get 'supplier' addresses (**Ex. 2014 at 19**, "call_routes.class.php" line 99): *`$routes_q = $this->sql_query("SELECT * FROM suppliers WHERE ml_id=".$this->callee_data['ml_id']." ORDER BY sup_price ASC");`* The genereate_routes() function then processes the results from the SQL query and builds a routes[] array (**Ex. 2014 at 20**, "call_routes.class.php" line 133): `$this->routes[] =  $c."h323-ivr-in = 'Routing:".$my_routes_info[$curr_route]['prepend'].$my_routes_info[$curr_route]['ndd_i`

	`dd_replacement'].$this->e164_formed_nubmer."@".$curr_route.$my_routes_info[$curr_route]['port'].";credit-time=$this->ttl".$my_routes_info[$curr_route]['auth'].$my_routes_info[$curr_route]['expires'].";cli=".$caller_id."'";`
	
	The routes[] array is returned to the inviteClient() function and stored into a "$response" variable (**Ex. 2014 at 43**, "invite.php" line 62):
	
	`$response = $call->get_routes();`
	
	The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php" line 70):
	
	`return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`

7. The process of claim 1 further comprising formatting said callee identifier into a pre-defined digit format to produce a re-formatted callee identifier.	The RBR server formats the callee identifier into an E.164 compliant number.
	
	The create_callee_profile() function calls the function set_e164_formed_number() to format and store a callee identifier (**Ex. 2014 at 10-11**, "call_e164.class.php" lines 473, 490, 500 and 513):
	
	`$this->set_e164_formed_number(substr($this->callee,$idd_length));`
	`$this->set_e164_formed_number($this->callee);`
	`$this->set_e164_formed_number($this->caller_countrycode.$this->callee);`
	`$this->set_e164_formed_number($this->caller_countrycode.$this->caller_areacode.$this->callee);`

27. [27p] A non-transitory computer	See claim element [1p].

readable medium encoded with codes for directing a processor to execute a method of operating a call routing controller to facilitate communication between callers and callees in a system comprising a plurality of nodes with which callers and callees are associated, the method comprising:	
[27a] in response to initiation of a call by a calling subscriber, receiving a caller identifier and a callee identifier;	See claim element [1a].
[27b] locating a caller dialing profile comprising a username associated with the caller and a plurality of calling attributes associated with the caller;	See claim element [1b].
[27c] determining a match when at least one of said calling attributes matches at least a portion of said callee identifier;	See claim element [1c].
[27d] classifying the call as a public network call when said match meets public network classification criteria and	See claim element [1d].

classifying the call as a private network call when said match meets private network classification criteria;	
[27e] when the call is classified as a private network call, producing a private network routing message for receipt by a call controller, said private network routing message identifying an address, on the private network, associated with the callee; and	See claim element [1e].
[27f] when the call is classified as a public network call, producing a public network routing message for receipt by a call controller, said public network routing message identifying a gateway to the public network.	See claim element [1f].
28. [28p] A call routing apparatus for facilitating communications between callers and callees in a system comprising a plurality of nodes with which callers and callees are associated, the apparatus comprising:	The RBR server implements a call routing controller. Callers and callees, which may be internet (IP) phones or endpoints on the PSTN, are associated with nodes. Within a Digifonica supernode, the RBR server facilitates communication by responding to requests from the B2BUA server and providing routing messages back to the B2BUA server.
[28a] receiving means	In response to a SIP INVITE message from an IP

for receiving a caller identifier and a callee identifier, in response to initiation of a call by a calling subscriber;	phone, the B2BUA server sends a USER_INVITE message to the RBR server.
	
	The inviteClient() function is the entry point for the RBR software that receives this message. The inviteClient function receives five parameters in an XML_RPC_Message (**Ex. 2014 at 42**, "invite.php" line 27):
	
	`* This function is called by the XML-RPC Server expecting type (integer * like 4 is THIRD_PARTY_INVITE) ,username,callee,caller,callid (In that * order in the XML_RPC_Message)`
	
	A type value of 2 represents a "USER_INVITE" message (**Ex. 2014 at 42**, "invite.php" line 35):
	
	`$type_arr[2] = 'USER_INVITE';`
	
	Besides the type parameters, the USER_INVITE message contains four other parameters identified as "username", "callee", "caller" and "called" (**Ex. 2014 at 42**, "invite.php" lines 44-47):
	
	`$username = $params->getParam(1)->scalarval();`
	`$callee = $params->getParam(2)->scalarval();`
	`$caller = $params->getParam(3)->scalarval();`
	`$callid = $params->getParam(4)->scalarval();`
	
	The "caller" variable corresponds to the caller identifier and the "callee" parameter corresponds to the callee identifier.

[28b] means for locating a caller dialing profile comprising a username associated with the caller	The RBR server loads a calling profile using the "caller" parameter. The inviteClient() function first constructs a new

and a plurality of calling attributes associated with the caller;	call_routes() object (**Ex. 2014 at 43**, "invite.php" line 60):
	
	$call = new call_routes($caller,$caller_domain,$callee, $callee_domain,$type,$forced_cid);
	
	The constructor for call_routes() is defined at lines 59-74 of call_routes.class.php (**Ex. 2014 at 19**):
	
	/**
	* Constructor... the class must be called with the defined parameters. Callee domain is optional and the class will try to guess it.
	* Special is a flag (THIRD_PARTY_INVITE) when set, the caller is not set and concidered a incoming PSTN call.
	*
	* @param String $caller
	* @param String $domain
	* @param String $callee
	* @param String $callee_domain
	* @param String $special
	* @return call_routes
	*/
	function call_routes($caller,$domain,$callee,$callee_domain='',$special='') {
	$this->special_request = $special;
	$this->call_ttl($caller,$domain,$callee,$callee_domain);
	$this->genereate_routes();
	}
	
	The call_routes() constructor calls the call_ttl() function (**Ex. 2014 at 19**, "call_routes.class.php" line 72):
	
	$this->call_ttl($caller,$domain,$callee,$callee_domain);
	
	The call_ttl() constructor is defined at lines 131-153 of

call_ttl.class.php (**Ex. 2014 at 27**):

```
/**
 * Class constructor must be initiated with
at least caller, domain (callee_domain and
special flag are optional.)<br>
 * Once it is initiated, it will
automatically execute proper functions and
do its due diligence, to create TTL.
 * $this->ttl is where you can get the
Total time to live for this call. can be
accessed byt extending classes and <br>
 * through the function get_ttl().
 *
 * @param string $caller
 * @param string $domain
 * @param string $callee
 * @param string $callee_domain
 * @param string $special
 * @return call_ttl
 */
function
call_ttl($caller,$domain,$callee,$callee_do
main='',$special='') {
    if(!$callee_domain){
        //try to guess
        if($temp_domain = $this-
>guess_user_domain($callee)) $this-
>callee_domain=$temp_domain;
    }
    $this-
>set_caller_data($caller,$domain,$special);

    $this->set_callee($callee,$this-
>callee_domain);
    $this->generate_ttl();
}
```

The call_ttl() function calls the set_caller_data() function (**Ex. 2014 at 27**, “call_ttl.class.php” line 149):

```
$this-
>set_caller_data($caller,$domain,$special);
```

The set_caller_data() function is defined at lines 187-189 of call_ttl.class.php and calls the set_caller()

function (**Ex. 2014 at 28**, “call_ttl.class.php” line 188):

```
$this->set_caller($caller,$domain);
```

The set_caller() function is defined at lines 325-340 of call_e164.class.php (**Ex. 2014 at 7**):

```
/**
 * Takes care of sequence of execution to
create a caller profile
 *
 * @param String $caller
 * @param String $domain
 */
public function
set_caller($caller,$domain){

    $this->caller = $caller;

    if(! ($this->special_request ==
'THIRD_PARTY_INVITE')){
        $this->caller_domain = $domain;
        $this->create_caller_profile();
    }
    $this->caller_is_set = true;
}
```

The set_caller() function calls the create_caller_profile() function (**Ex. 2014 at 7**, “call_e164.class.php” line 337):

```
$this->create_caller_profile();
```

The create_caller_profile() function is defined at lines 342-391 of call_e164.class.php (**Ex. 2014 at 7-8**) (lines 342-346 shown):

```
/**
 * All aspects of the caller Flags and data
needed to identify a caller is set here.
including overlapping numbers.
 *
 */
private function create_caller_profile(){
```

	The create_caller_profile() function queries an SQL database to load a caller profile (**Ex. 2014 at 7**, "call_e164.class.php" line 348):
	
	`$user_data = mysql_fetch_array($this->sql_query("SELECT * FROM subscriber_dialing_profile WHERE username='$this->caller' and domain= '$this->caller_domain'"));`
	
	The data returned from the SQL query include the caller's idd (international direct dial digits), ndd (national direct dial digits) and area code (**Ex. 2014 at 7**, "call_e164.class.php" lines 350-356):
	
	`$this->caller_idd = $user_data['idd'];`
	`$this->caller_ndd = $user_data['ndd'];`
	`$this->caller_areacode = $user_data['area_code'];`
	`$this->caller_countrycode = $user_data['country_code'];`
	`$temp = explode(",",$user_data['local_length']);`
	`$this->caller_min_local_length = $temp[0];`
	`$this->caller_max_local_length = $temp[1];`

[28c] means for determining a match when at least one of said calling attributes matches at least a portion of said callee identifier;	The RBR server matches attributes in the caller profile values against the callee identifier.
	
	The call_ttl() constructor, after calling the set_caller_data() function calls the set_callee() function (**Ex. 2014 at 27**, "call_ttl.class.php" line 151):
	
	`$this->set_callee($callee,$this->callee_domain);`
	
	The set_callee() function is defined at lines 393-418 of call_e164.class.php (**Ex. 2014 at 8-9**) (lines 393-399 shown):
	
	`/**`
	` * Takes care of the sequence and logic`

	needed to create a callee
	*
	* @param String $callee
	* @param String $callee_domain
	*/
	public function
	set_callee($callee,$callee_domain=''){
	
	The set_callee() function calls the create_callee_profile() function (**Ex. 2014 at 8**, "call_e164.class.php" line 407):
	
	$this->create_callee_profile();
	
	The create_callee_profile() function is defined at lines 450-539 of call_e164.class.php (**Ex. 2014 at 9-11**) (lines 450-454 shown):
	
	/**
	* Creates a callee profile, all aspects of the callee is set here including the e164 number is also set here. NOTE: the e164 number can change when call forwarding occures.
	*
	*/
	private function create_callee_profile(){
	
	The create_callee_profile() function matches values that were previously retrieved as part of the caller profile with the callee identifier. For example, a check is made to see if the callee identifier begins with the international dialing digits (idd) (**Ex. 2014 at 9-10**, "call_e164.class.php" lines 467-476):
	
	case (isset($this->caller_idd) &&
	preg_match("/^".$this->caller_idd."/",$this->callee)):
	$idd_length = strlen($this->caller_idd);
	$this->set_ml_id($idd_length);
	if($this->callee_data['ml_id']){ //callee was found and set
	$this->callee_type = 1;
	$this->check_callee_length($idd_length);

	`$this->set_e164_formed_number(substr($this->callee,$idd_length));`
	`}else{`
	`  throw new Exception(4);`
	`}`
	
	Further, create_callee_profile() checks for matching national dialing digits (**Ex. 2014 at 10**, "call_e164.class.php" lines 479-494), checks for matching area codes (**Ex. 2014 at 10**, "call_e164.class.php" lines 495-501), checks for a matching local number (**Ex. 2014 at 10-11**, "call_e164.class.php" lines 506-513), and checks for a network number (**Ex. 2014 at 11**, "call_e164.class.php" lines 520-527).

[28d] means for classifying the call as a public network call when said match meets public network classification criteria;
 	
[28e] means for classifying the call as a private network call when said match meets private network classification criteria;
	The RBR server classifies the call as a public network call or a private network call.
	
	The create_callee_profile() function classifies the call by setting the callee_type variable to a value from 1 to 7. In the case of a public classification, the callee_type variable is set to one of the values 1, 2, 3 or 6 (**Ex. 2014 at 10**, "call_e164.class.php" lines 471, 483, 485, 497 and 510)"
	
	`$this->callee_type = 1;`
	`$this->callee_type = 6;`
	`$this->callee_type = 2;`
	`$this->callee_type = 3;`
	`$this->callee_type = 3`
	
	In the case of a private classification, the callee_type variable is set to one of the values 4 or 7 (**Ex. 2014 at 11**, "call_e164.class.php" lines 523 and 525 respectively):
	
	`$this->callee_type = 4;`
	`$this->callee_type = 7;`
	
	The callee_type variable is set to a value of 5 in the case of a failure to classify the call (**Ex. 2014 at 11**,

	"call_e164.class.php" line 529):
	
	`$this->callee_type = 5; //fail`
	
	The set_callee() function calls the check_for_did() function. The check_for_did() function is defined at lines 420-441 of call_e164.class.php (**Ex. 2014 at 9**):
	
	`/**`
	` * Checks if the number being called is a mapped DID in our system, if it is we rout it internally`
	` *`
	` * @param String $user`
	` * @return Boolean or DidMap(DigifonicaNumber#Domain)`
	` */`
	`protected function check_for_did($user=''){`
	`  $num =  $user?$user:$this->callee;`
	`  $did_check = @mysql_fetch_array($this->sql_query("SELECT * FROM did_bank WHERE did='$num'"));`
	`  if($did_check[0]){`
	`  try{`
	`  if($did_check['mapped_to'] and $did_check['domain']){`
	`  return $did_check['mapped_to']."#".$did_check['domain'];`
	`  }else{`
	`  throw new Exception(8);`
	`  }`
	`  }catch (Exception $e){`
	`  $this->e164_exception_handler($e->getMessage());`
	`  }`
	`  }`
	`  return false;`
	`}`
	
	The check_for_did() function queries an SQL database to determine if the reformatted callee number is contained in the "did_bank" table. This check determines if the callee is mapped to a Digifonica subscriber account. In the case that the check_for_did()

	function returns a value, the set_callee() function will make a call to itself at line 415 of call_e164.class.php (**Ex. 2014 at 8**). This will cause a second call to the create_callee_profile() function, which will cause the callee_type to be changed to 4 or 7 depending on whether the destination is associated with the same supernode or a different supernode as the caller.

[28f] means for producing a private network routing message for receipt by a call controller, when the call is classified as a private network call, said private network routing message identifying an address, on the private network, associated with the callee; and	The RBR server produces a private network routing message in the case of a private network call classification.
	
	The call_routes() constructor, after calling the call_ttl() function calls the genereate_routes() function (**Ex. 2014 at 19**, "call_routes.class.php" line 73):
	
	`$this->genereate_routes();`
	
	The genereate_routes() function is defined at lines 91-148 of call_routes.class.php (**Ex. 2014 at 19-21**) (lines 91-95 shown):
	
	`/**`
	` * Generates the routes and the SIP messages including VM data.`
	` *`
	` */`
	`protected function genereate_routes(){`
	
	The genereate_routes() function tests the value of callee_type and in the case of 4 and 7 sets the routes[] array (**Ex. 2014 at 20**, "call_routes.class.php" line 146):
	
	`/**`
	` * Generates the routes and the SIP messages including VM data.`
	` *`
	` */`
	`protected function genereate_routes(){`
	
	The routes[] array is returned to the inviteClient()

	function and stored into a "$response" variable (**Ex. 2014 at 43**, "invite.php" line 62):
	
	`$response = $call->get_routes();`
	
	The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php, line 70):
	
	`return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`

[28g] means for producing a public network routing message for receipt by a call controller, when the call is classified as a public network call, said public network routing message identifying a gateway to the public network.	The RBR server produces such a public network routing message in the case of a public network call classification. The genereate_routes() function tests the value of callee_type and in the case of 1, 2, 3 or 6 queries an SQL database to get 'supplier' addresses (**Ex. 2014 at 19**, "call_routes.class.php" line 99): *`$routes_q = $this->sql_query("SELECT * FROM suppliers WHERE ml_id=".$this->callee_data['ml_id']." ORDER BY sup_price ASC");`* The genereate_routes() function then processes the results from the SQL query and builds a routes[] array (**Ex. 2014 at 20**, "call_routes.class.php" line 133): `$this->routes[] = $c."h323-ivr-in = 'Routing:".$my_routes_info[$curr_route]['prepend'].$my_routes_info[$curr_route]['ndd_idd_replacement'].$this->e164_formed_nubmer."@".$curr_route.$my_routes_info[$curr_route]['port'].";credit-time=$this->ttl".$my_routes_info[$curr_route]['auth'].$my_routes_info[$curr_route]['expires'].";cli=".$caller_id."'";` The routes[] array is returned to the inviteClient()

	function and stored into a "$response" variable (**Ex. 2014 at 43**, "invite.php" line 62):
	
	`$response = $call->get_routes();`
	
	The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php" line 70):
	
	`return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`

34. The apparatus of claim 28 further comprising formatting means for formatting said callee identifier into a predefined digit format to produce a re-formatted callee identifier.	The RBR server formats the callee identifier into a E.164 compliant number.
	
	The create_callee_profile() function calls the function set_e164_formed_number() to format and store a callee identifier (**Ex. 2014 at 10-11**, "call_e164.class.php" lines 473, 490, 500 and 513):
	
	`$this->set_e164_formed_number(substr($this->callee,$idd_length));`
	`$this->set_e164_formed_number($this->callee);`
	`$this->set_e164_formed_number($this->caller_countrycode.$this->callee);`
	`$this->set_e164_formed_number($this->caller_countrycode.$this->caller_areacode.$this->callee);`

54. [54p] A process for operating a call routing controller to establish a call between a caller and a callee in a communication system, the process comprising:	See claim element [1p].
[54a] in response to initiation of a call by a	See claim elements [1a] and [1b].

calling subscriber, locating a caller dialing profile comprising a plurality of calling attributes associated with the caller; and	
[54b] when at least one of said calling attributes and at least a portion of a callee identifier associated with the callee match and when the match meets a private network classification criterion, producing a private network routing message for receipt by a call controller, said private network routing message identifying an address, on a private network, the address being associated with the callee; and	See claim elements [1c], [1d] and [1e].
[54c] when at least one of said calling attributes and said at least said portion of said callee identifier associated with the callee match and when the match meets a public network classification criterion, producing a public network routing message for receipt by a call controller, said public network routing message	See claim elements [1c], [1d] and [1f].

identifying a gateway to a public network.	
72. The process of claim 54 further comprising causing the private network routing message or the public network routing message to be communicated to a call controller to effect routing of the call.	The RBR server returns the routing message to the B2BUA server, which then effects the routing of the call. The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php" line 70): `return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`
73. A non-transitory computer readable medium encoded with codes for directing a processor to execute the method of claim 54.	The RBR server is implemented on a computer with storage for the computer software executed by a processor.
74. [74p] A call routing controller apparatus for establishing a call between a caller and a callee in a communication system, the apparatus comprising:	See claim element [1p].
[74a] a processor operably configured to: access a database of caller dialing profiles wherein each dialing profile associates a plurality of calling attributes with a respective subscriber, to locate a dialing profile	See claim elements [1a] and [1b].

associated with the caller, in response to initiation of a call by a calling subscriber; and	
[74b] produce a private network routing message for receipt by a call controller, said private network routing message identifying an address, on a private network, through which the call is to be routed, when at least one of said calling attributes and at least a portion of a callee identifier associated with the callee match and when the match meets a private network classification criterion, the address being associated with the callee; and	See claim elements [1c], [1d] and [1e].
[74c] produce a public network routing message for receipt by a call controller, said public network routing message identifying a gateway to a public network, when at least one of said calling attributes and said at least said portion of said callee identifier associated with the callee match and when the match meets a public	See claim elements [1c], [1d] and [1f].

network classification criterion.	
92. The apparatus of claim 74 wherein said processor is further operably configured to cause the private network routing message or the public network routing message to be communicated to a call controller to effect routing of the call.	See claim 72.
93. [93p] A call routing controller apparatus for establishing a call between a caller and a callee in a communication system, the apparatus comprising:	See claim element [1p].
[93a] means for accessing a database of caller dialing profiles wherein each dialing profile associates a plurality of calling attributes with a respective subscriber, to locate a dialing profile associated with the caller, in response to initiation of a call by a calling subscriber; and	See claim elements [28a] and [28b].
[93b] means for producing a private	See claim elements [28c], [28d], [28e] and [28f].

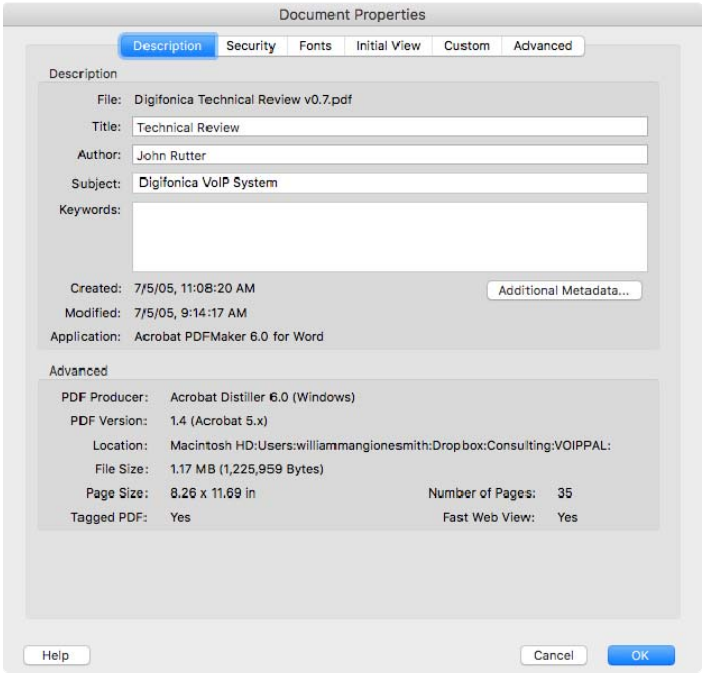
network routing message for receipt by a call controller, said private network routing message identifying an address, on a private network, through which the call is to be routed, when at least one of said calling attributes and at least a portion of a callee identifier associated with the callee match and when the match meets a private network classification criterion, the address being associated with the callee; and	
[93c] means for producing a public network routing message for receipt by a call controller, said public network routing message identifying a gateway to a public network when at least one of said calling attributes and said at least said portion of said callee identifier associated with the callee match and when the match meets a public network classification criterion.	See claim elements [28c], [28d], [28e] and [28g].
111. The apparatus of claim 93 further	The RBR server returns the routing message to the B2BUA server, which then effects the routing of the

comprising means for causing the private network routing message or the public network routing message to be communicated to a call controller to effect routing of the call.	call. The inviteClient() function returns the $response variable to the B2BUA server over the XML_RPC connection (**Ex. 2014 at 43**, "invite.php" line 70): `return new XML_RPC_Response(new XML_RPC_Value($response, "string"));`

25. Based on the foregoing, it is my opinion that Claims 1, 7, 27, 28, 34, 54, 72-74, 92, 93 and 111 of the '815 Patent were practiced by the RBR server running version 361 of the RBR software within the Digifonica system.

VIII. DIGIFONICA DOCUMENT REVIEW

26. I was asked to review a computer file titled "Digifonica Technical Review v0.7.pdf" (**Ex. 2003**) and comment on any information that I found related to its date of creation. I used the program Adobe Acrobat to open the file and then review the file properties. This process informed me that the file contains metadata that indicates it was created on July 5th of 2005. I have included a screenshot of this below:



27. I was also asked to retrieve the document "DF_Technology_Overview_June 05.sxw" from the SVN software repository discussed above. I retrieved this document from the directory "documentation/DF_Authored_Whitepapers" using the svn software tools. I then opened this document using the program OpenOffice and produced a PDF file (**Ex.2020**). The Subversion repository indicates that this file was last modified on June 5, 2005.

28. I have reviewed the Declaration of David Terry (**Ex. 2018**). Mr. Terry describes the operation of the Digifonica system in paragraph four. I have reviewed the Declaration of Clay Perreault (**Ex. 2013**). Mr. Perreault describes the operation of the Digifonica system in paragraphs 12-13. I have reviewed the

Declaration of Johan Emil Viktor Bjorsell (**Ex. 2012**). Mr. Bjorsell describes the operation of the Digifonica system in paragraphs 4-5 and 16-20. Mr. Terry's description, Mr. Perreault's description and Mr. Bjorsell's description all confirm that the RBR software was deployed in a system that operated consistent with my analysis of Version 361 above.

29. The contents of Ex. 2003, Ex. 2020, Ex. 2018, Ex. 2013 and Ex. 2012 support my opinion that the Version 361 software that I reviewed was operational in June 2005, and that the features present in the Version 361 software successfully practiced the claims of the '815 Patent analyzed above. Thus, it is my opinion that the subject matter of the claims of the '815 Patent analyzed above had been reduced to practice by Digifonica in June 2005.

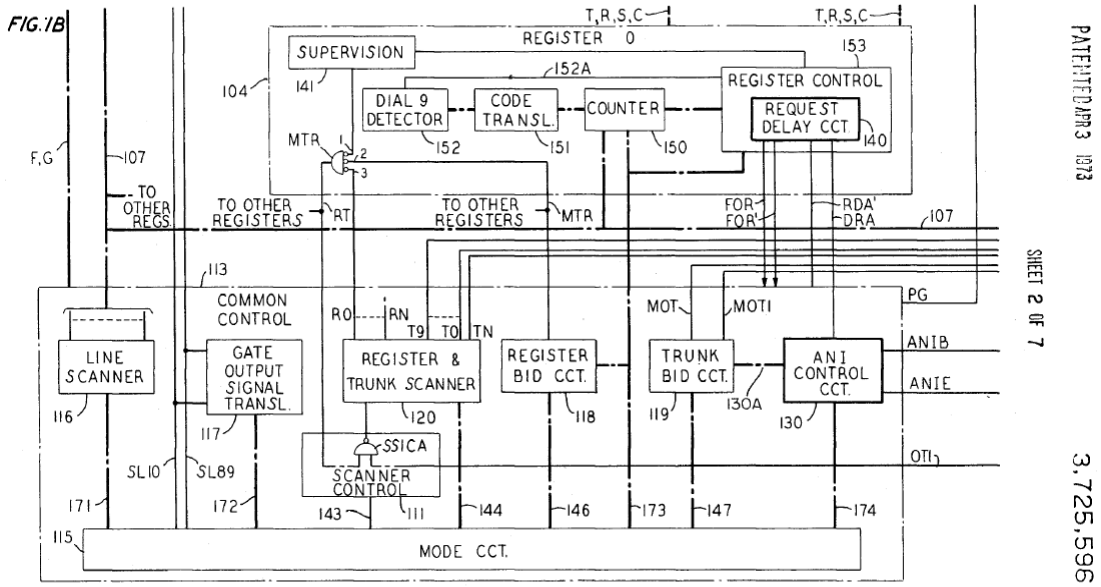
IX. OVERVIEW OF DIAL PLANS

30. U.S. Patent No. 3,725,596 ("Maxon") (**Ex. 2046**) discloses an early private branch exchange (PBX) having equipment for automatically generating and transmitting calling station and trunk number information to a central office on outgoing calls. See the Maxon Abstract. The Maxon patent was filed in 1971 and has some features which would have been well known at the time of the priority date of the '815 and '005 Patents.

31. Maxon discloses: "Our invention is disclosed as embodied in a PBX having an end marked network together with a wired logic system controller. An

outgoing call is initiated when a station user removes his subset, is connected to a register, and dials a prefix digit, such as 9, to obtain a connection to a central office trunk circuit. The register detects the dialing of the prefix digit 9 and advises the system controller that a connection to an outgoing trunk circuit is required." (Maxon@2:1-9).

32. Regarding Maxon's FIGS 1A and 1B (the latter of which is reproduced below), Maxon discloses at column 9, line 66 to column 10, line 6, that "a calling party at station ST10... dials a prefix digit, such as the conventional prefix digit 9, to initiate an outgoing call to the central office. The digit 9 is registered... translated... and detected by the dial 9 detector 152. Upon the detection of this digit, the register control circuit 153 advises common control that the digit 9 has been dialed for a central office call." (Maxon@9:66-10:6) (emphasis added).



33. FIG. 1B of Maxon (above) shows a circuit labeled "DIAL 9 DETECTOR" 152, which was referred to above. Notably, Maxon discloses that dialing a prefix digit of "9" was already considered "conventional" at the time Maxon was filed in 1971.

34. Dialing, detecting, or processing a prefix digit (in this case, "9") is mentioned also at Maxon 8:58-68, 9:21, 9:38-40, 13:3-6, 14:6-7 and at 14:59.

35. Maxon Claim 1 in column 14 recites, "...means for detecting the receipt from a calling one of said line circuits of a dialed prefix digit signifying a request for a network connection to one of said trunk circuits, first means responsive to said detection for identifying said calling line circuit and for supplying a signal identifying said line circuit to said network, second means responsive to said detection for selecting an idle one of said trunk circuits and for

supplying a signal identifying said selected trunk circuit to said network, said network being responsive to the receipt of said identification signals from said first and second means for establishing a connection between said calling line circuit and said selected trunk circuit, and means for transmitting information from said first and second means to a central office identifying said calling line circuit and said selected trunk circuit.."

36. The functionality embodied in Claim 1 of Maxon involves detecting a prefix digit as part of establishing a PSTN call to a "central office". While Maxon discloses an outdated implementation of detecting a user-dialed prefix digit and, in response to the detection, routing the call to a public PSTN network destination, different variants of this technique have been implemented in nearly all PBX systems during the past 40 years (and continue to be implemented today).

37. The prefix digit for PSTN dialing in North America is typically "9" or "8" but in other countries may be a different digit (e.g., "0"). The prefix digit that is used by the PBX to direct calls to the PSTN is typically a system-wide setting in the PBX, and commonly would be part of the "dial plan" of the PBX.

38. In a PBX system, the "dial plan" (or "dialing plan") defines the types of dialing patterns that are expected or permissible and the function or action that would result in response to the dialing patterns.

39. The document entitled "MERLIN Communications System Centrex/PBX Connection" (**Ex. 2047**) is a user guide published by Lucent Technologies in 1985, which also was responsible for the Chu '684 patent. The MERLIN device and provided a telephone network within an office which was also connected to a PBX or Centrex system. At page 15, the MERLIN user guide refers to using prefix digits for classifying calls in the context of a PBX, for example, "Contact people outside the building by dialing to access an outside line (usually 9) and the number". Other prefixes could also be used in the alternative or in addition, such as dialing 8 to access special network lines of a company network.

40. In some cases, it was expected for the user to pause after dialing the prefix digit for accessing the PSTN. For example, page 18 of the MERLIN user guide, in describing how to program outside auto dial buttons, states, "if your Centrex/PBX service does not immediately return a dial tone after dialing to access an outside line (usually by dialing 9), they must program a pause by using Hold (Pause) after the initial digit to give the system time to return a dial tone." Notably, dialing 9 was "usually" the method used to access an outside line at the time Lucent's MERLIN user guide was published, in 1985.

41. A Quick Reference Guide for a "Definity" system was published by Rockefeller Group Telecommunications Services Inc. (**Ex. 2048**), and provided

instructions about how to dial various PSTN numbers by dialing a prefix digit of 9. Per the internal metadata of the PDF file for the quick reference guide, the document was created in 2000 and relates to a "Definity G3", which I believe is a reference to Lucent's Definity G3 PBX product, which was available in 2000 or earlier.

42. A textbook published in 2006 by Valdar entitled "Understanding Telecommunications Networks" describes the basic functions of a PBX system at page 38 (**Ex. 2049**). Valdar states, "Businesses which have more than a few telephones use a private branch exchange system, known as a PBX, to provide call connections between each telephone (which become 'extensions') and links into the PSTN [7]. The PBX is really a small version of the PSTN exchanges, typically ranging in sizes from 10 up to 5,000 extensions. A private numbering scheme is required to enable extension to extension dialing, also special codes (e.g. 'dial 9') are required to enable calls to be made to the PSTN. Incoming calls from the PSTN have to be answered by a receptionist or operator at a manual console so that the appropriate (privately numbered) extension can be contacted. Alternatively, the extension numbers can form part of the public numbering scheme (see Chapter 10) so that calls from the PSTN can be directly switched by the PBX to the required extension (known as DDI), so avoiding the need for manual intervention where the caller knows the number of the wanted extension. Only the calls to the PSTN are

charged. The corporate customer owns and pays for its PBX. In the case where a company extends over two or more sites (e.g. office or factory buildings) the PBXs on each site can be linked by private circuits, thus enabling calling between all the extensions. This is known as a 'private corporate network' (or just 'private network'). In this case the private numbering scheme extends across all the PBXs and usually each PBX is linked to the PSTN. Charging only applies to calls leaving the private network for the PSTN, although, of course, a rental charge is made by the network operator for the lease of the private circuits. A virtual private network (VPN) provides an alternative to the use of private circuits between each PBX, as shown in Fig. 2.12(a). The VPN exchange switches calls between die PBXs connected to it (e.g. between PBXs 'a' and 'b') as well as to trunk links to the other PBXs in the VPN corporate network. However, the VPN is provided over public exchanges, either as special business exchanges or as part of the PSTN. Each VPN exchange switches the private network calls of several private corporate networks, although each operates in isolation, using its own numbering scheme. Thus, each corporate network appears to have the benefits of a private set of links between their PBXs, even though connectivity is provided over public exchanges - hence the use of the word 'virtual' in VPN. The VPN customer is charged a subscription for the VPN service based on a certain level of inter-PBX calls (traffic); there are

usually charges made when the level of calls between any two PBXs exceeds the agreed threshold."

43. Webster's New World Telecom Dictionary provides a definition for dial tone which makes a distinction between dialing an "internal PBX station number" and an "external number" (**Ex. 2050**). In the latter case, "the user must dial an access code in order to gain access to an external trunk connected to the public switched telephone network (PSTN)... The conventional access code is nine (9) in the United States and Canada, and zero (0) in most other countries".

44. The PBX dialing rules, such as those from the Rockefeller Group discussed above, are referred to as a private dial plan. The PBX company will offer a range of options (such dialing 8 or 9 for an outside line) and the IT department of an enterprise will finalize the ultimate dial plan used. The purpose of a private dial plan is to provide uniform access to enterprise services, such as intercoms and short extension numbers, in a manner that is consistent and thus easy to manage across the enterprise.

45. In contrast the E.164 standard dial plan is a fixed international standard that specifies how to identify the nation called, region within a nation, etc. The private dial plan is often referred to as an internal dial plan while the E.164 is the external dial plan. They serve different purposes and are not equivalent.

46. It has long been a convention to use a simple single digit to differentiate between an internal private dial plan and the external dial plan, as discussed above with the common use of dialing a single digit, e.g. 8 or 9, to differentiate between the two plans. Clearly there could not be a method for categorizing a call as either internal or external that was easier or quicker for a user than dialing a single digit.

47. The documents discussed above, Ex. 2046, Ex. 2047, Ex. 2048, Ex. 2049 and Ex. 2050, are naturally only examples, a few from many, and I would expect that a person of ordinary skill in the art would likely have been employed in an enterprise that used exactly such a system, or have been exposed to dial 8 or 9 type approach during their education or work colleagues.

X. OPINIONS CONCERNING CHU '684 AND CHU '366

A. A Person of Ordinary Skill Would Not Have Combined Chu '684 and Chu '366

48. Dr. Houh presents a number of arguments in section VII of his declaration regarding the combination of Chu '684 with Chu '366. I will endeavor to respond to a number of those arguments that I believe are either misguided or incorrect in the order that they appear.

49. He states in paragraph 36 of his declaration that it would have been obvious "to one of skill in the art to modify the system described by Chu '684 with

the specific dialed digit reformatting teachings of Chu '366." I believe his conclusion is incorrect and I will explain my reasoning.

50. His first argument is that the two pieces of art are "in the same technological field and substantially overlap in relevant subject matter". They are not and do not. For example, Chu '684 is categorized by the USPTO as within patent class H04L 65/1069 which relates to call setup within a network infrastructure while Chu '366 resides in H04M7/0075 "Details of addressing, directories or routing tables" and H04M1/2535 "Telephone sets using digital voice transmission adapted for voice communication over an Internet Protocol [IP] network". The USPTO, after the entire review process had completed with the inventors of these two pieces of art, placed them into completely disjoint technical categories. They share a common parent only under H04 which is "Electric Communication Technique".

51. These categorizations place Chu '684 in the realm of a network infrastructure. This fact is highlighted by the Field of the Invention portion of Chu '684 which states:

> ***The invention relates to the field of communications systems and more specifically to the management and control of voice-over Internet Protocol (VoIP) virtual private networks (VPNs) in an IP-based public branch exchange (PBX) environment.***

52. Thus, Chu '684 is intended to address problems in a network switch where a multiplicity of virtual private networks provide traffic. Certainly, aspects of Chu '684 may have broader or narrower application however it is clear that the focus of the problems Chu '684 is attempting to address present themselves within a network infrastructure.

53. Chu '684 is furthermore directed to organizations that subscribe to services in order to support multiple locations, each with multiple telephones. "The subject invention specifies a network architecture for providing a voice over IP virtual private network (VoIP VPN) service to an enterprise and a method of establishing such a VoIP VPN. The VoIP VPN service connects all the IP-PBXs 55 of an enterprise into a single logical network." Chu '684 at Column 3 Lines 52-56. Chu '684 contains multiple passages and figures that refer to an enterprise that operates at multiple physical locations, for example Subscriber Location 105 shown in Fig. 4. Furthermore, Chu '684 specifically refers to a situations where a call connects a single enterprise at two different locations:

> *The above description is for establishing a call between two IP phones at two locations of the same subscriber. Many subscribers, each with multiple locations, can be served by the same packet-switch/soft-switch network. Each subscriber can use their own IP address plan as well as their own dial plan. To each subscriber, it appears that all their locations are connected by*

a private network, although the same network is used to serve multiple subscribers. (Chu '684 Column 12 Lines 59-66)

54. Thus, Chu '684 is clearly focused on a system where network infrastructure supports multiple enterprises (which, Chu '684 calls "subscribers"), each enterprise may have multiple locations. Thus a call may connect one enterprise telephone to another telephone of the same enterprise. It seems highly unlikely that a single person in Los Angeles would intend to call themselves in Philadelphia, though an individual working at IBM in Los Angeles may call another IBM employee in Philadelphia. Thus, the word "subscriber" in Chu '684 is referring to an organizational entity such as an enterprise.

55. In contrast, Chu '366 is directed to problems faced by an individual person who is attempting to initiate a telephone call. Chu '366 is focused on processing and preparing a dialing number at the caller's location for presentation to a VOIP system rather than as part of a VOIP system. For example, the abstract of Chu '366 refers to a user interface that a user employs to dial a phone. Figures 3-5 and 8-9 of Chu '366 disclose user interfaces. Chu '366 discusses the limitations of the prior art within this passage:

By contrast, there is no such concept of local, long distance or national calls when making a call via Internet telephony. VoIP calls use the Internet, which is world-wide and not tied to any single location. While a

VoIP call may be made between two local points, that call may be routed through servers located across the globe. Consequently, existing global VoIP service providers require users to enter fully formatted E.164 telephone numbers. Even when making a local call, within for example the United States, instead of simply entering the seven digit telephone number, the VoIP subscriber must enter "+," then the country code, then the area code, then the telephone number. This is time consuming and bothersome, especially when subscribers are used to the concept of abbreviated telephone numbers for local and national calls with traditional PSTN systems. **(Chu '366 @ 1:44-58)**

According to the present system, a user is able to enter telephone numbers for VoIP telephone calls as they would according to a traditional telephone numbering plan for land-line telephone calls. (Chu '366@1:67-4)

Upon launching the application program according to the present system, the user may be presented with the presentation and user interface layer, which may include a variety of graphical objects allowing the user to initiate and terminate VoIP telephone calls. The user interface layer may

provide the user the ability to set up and store one or more call origin location profiles which identify geographic location information as well as the country and/or area code (if applicable) for the one or more locations from which the user may regularly place VoIP telephone calls, e.g., home, office, satellite office, etc. (Chu '366@2:5-15).

56. While Chu '366 uses the word "subscriber" the context of this passage, along with the rest of the Chu '366 specification, would make it clear to a person of ordinary skill in the art that a Chu '366 means a user who desires to dial a telephone in order to initiate a personal call to another individual. The Chu '366 user interface is presented to a specific person on a specific telephone not to all of IBM at all locations simultaneously. Thus, while both Chu '684 and Chu '366 use the term "subscriber", it would be plain that their usage is fundamentally distinct. In Chu '684, "subscriber" is an enterprise, not an individual, whereas Chu '366 uses the same term to mean individual user.

57. Based upon my review of Chu '684 and Chu '366, and in particular the excerpts cited above, it is my opinion that, contrary to the arguments of Dr. Houh, a person of ordinary skill in the art would conclude that Chu '684 and '366 are in totally different technical fields and are attempting to solve different problems. Furthermore, Chu '684 is directed to enterprise-level problems relating to PBX systems, whereas Chu '366 (and Chen) do not relate to PBX systems or

enterprise dial plans. The teachings of Chu '366 and Chen regarding how to address problems faced by individual persons who are attempting to initiate a telephone call, provide no meaningful teaching relevant to enterprise-level problems for PBX's, let alone any guidance as to how to manage or modify an enterprise dial plan for a PBX. Changing individual settings for a device which only serves one individual as in Chu '366 or Chen, is completely different than implementing an IT telephony policy for a multiplicity of users in an enterprise dial plan. Thus there is no way that a skilled person would look to Chu '366 or Chen for information about how to implement PBX systems or dial plans.

58. Dr. Hough moves on to argue for combining Chu '684 and Chu '366 by stating:

> ***Additionally, both references teach a process in which dialed digits and caller attributes are used to determine where the call should be routed. Compare Ex. 1003, Chu '684 at 8:65-9:1 ("At step 608, after receiving all the dialed digits from the phone 101, server 110 consults its dial plan to determine whether the call is local, to another on-net phone, or to a phone that is on the PSTN.") with Ex. 1004,***

59. This passage is based on a key misunderstanding at the heart of Dr. Houh's declaration and the petition for review, i.e. the mistaken notion that a Chu '684 subscriber is an individual caller, with the implication that a "dial plan" in

Chu ‘684 is caller-specific. It is not. There is one dial plan used by an enterprise, shared by all callers, thus the statement that Chu ‘684 teaches a “process in which dialed digits and caller attributes are used to determine where the call should be routed” is mistaken. Dr. Houh corrected this error during his deposition by revising his position to state that Chu ‘684 is referring to an enterprise when it uses the word subscriber (Ex. 2043, Houh deposition at 15:11 – 17:4). As I discussed above, the subscriber in Chu ‘684 is not a person attempting to place a telephone call (i.e. a caller) but rather an enterprise that may comprise many telephones at many separate physical locations each with many callers.

60. During his deposition Dr. Houh recognized and acknowledged this mistake when he stated that Chu ‘684 was directed to enterprises rather than users. Thus, Chu ‘684 does not relate in any way to “caller attributes”:

```
11        Q.   Okay.  Thank you.
12             There is some confusion on my part relative
13  to the use of the term "subscriber" in Chu '684.
14  The -- at one point you quote, indicating that the --
15  there could be many subscribers, each with multiple
16  locations, that could be served by the same
17  packet-switch network.  Each subscriber uses their own
18  IP address as well as their own dial plan.
19             The question, specifically, is -- "many
20  subscribers, each with multiple locations."  That
21  sounds like the subscribers that are being referenced
```

22 are a collective -- an enterprise, rather than an
23 individual. Could you clarify that, please?
24 MR. HART: Hey, Ryan, could you -- for my
25 benefit and for Dr. Houh's, could you point us to
1 where you were quoting, just to see if I need to
2 object -- enter an objection or not?
3 MR. THOMAS: Okay. Yes. That is page 16,
4 paragraph 36 and page 17, paragraph 37.
5 MR. HART: And that's from the Exhibit 1,
6 the declaration from the '005 patent proceeding?
7 MR. THOMAS: Yes, that's correct. And the
8 specific language is from paragraph 37. It's about
9 the 5th line in that.
10 MR. HART: Got it.
11 A. Okay. I see that. So I think there is --
12 part of the confusion may arise out of the fact that
13 the '815 patent also uses the term "subscriber," and
14 the '684 patent uses the term "subscriber."
15 So -- and they use them in -- I -- I
16 suppose, different senses, and -- and that the '815
17 patent uses the term "subscriber" as a -- as an
18 individual user, and the discussion in '684 does talk
19 about subscriber as an enterprise. And I -- I -- I
20 think the confusion arises out of that, but I've tried
21 to use the terminology of the -- of the '815 patent,
22 but it -- it -- it leads to -- it leads to this issue

```
23  you described.
24              But the '684 patent does talk about
25  enterprises and enterprise IP-PBX.  I -- let me see if
 1  I can find a good citation for that.  Yes.  In the
 2  description of the background art, it does talk about,
 3  you know -- like, around 42, that -- or sorry.
 4  Starting at line 33.
```

Ex. 2043, Houh deposition at 15:11 – 17:4.

61. Dr. Houh states "For example, Chu '684 teaches that each subscriber is assigned their own dial plan, a unique IP address, and a unique E.164-compliant telephone number Ex. 1003, Chu '684 at 3:56-64." (Houh Declaration at paragraph 45). This statement is simply wrong. That portion of Chu '684 states that each IP-phone, not each subscribing enterprise, may have its own E.164 telephone number. Furthermore, the passage states that such a telephone number, along with the IP address and dial plan, are determined by the enterprise while the language in the Petition suggests that these values are assigned to the enterprise by some third party.

62. Thus, a person of ordinary skill would conclude that Chu '684 and Chu '366 do not "***teach a process in which dialed digits and caller attributes are used to determine where the call should be routed".*** More significantly Dr. Houh has agreed with my conclusion here during his deposition testimony, thus

correcting a portion of his erroneous declaration and pointing to a key error in the Petition.

63. Dr. Houh then moves on to state that:

Second, the system of Chu '684 already contains the infrastructure needed to support such reformatting. Chu '684 expressly discloses geographically dispersed subscribers each of whom may use subscriber-specific dial plans. Ex. 1003, Chu '684 at 12:60-64. ("Many subscribers, each with multiple locations, can be served by the same packet-switch/soft-switch network. Each subscriber can use their the [sic] own IP address plan as well as their own dial plan."). One of skill in the art would understand that the purpose of assigning subscriber-specific dial plans, rather than dial plans applicable to all PBX subscribers, is to include subscriber-specific information such as geographic location, area codes, etc. Thus, like as expressly disclosed in Chu '366, the infrastructure of Chu '684 would support dialed digit reformatting based on attributes of the caller such as location and area code. (Hough declaration @ paragraph 37).

64. This passage mischaracterizes what Chu '684 discloses. To begin with I will note that Chu '684 discloses two different types of "dial plans". The first, cited previously in this report, is the "public E.164 number plan" (Chu '684 at 9:17). The second type of dial plan disclosed by Chu '684, is the *enterprise*

internal dial plan (Chu '684 at 3:56-58). Chu '684's statement that each "subscriber" has their own dial plan means that each *enterprise* has its own dial plan. This internal enterprise dial plan would be applicable to all users of an enterprise's PBX. To be clear, Chu '684 does not disclose "user-specific" dial plans, only enterprise dial plans. An enterprise dial plan is intended to be shared by many users, thus by its very nature it cannot be user-specific. The server 110 would know and use the internal dial plan for one enterprise (Chu '684 at 8:65-9:1), whereas the soft-switch 220 would know the internal dial plans of multiple enterprises that subscribe to the service provider's network (Chu '684 at 4:59-62). The soft-switch 220 would use the dial plan for a particular enterprise *after* it received a call from the server 110 (Chu '684 at 9:30-31). Dr. Houh acknowledges that Chu '684 does not disclose reformatting based upon a dial plan because he relies on Chu '366 as providing that teaching. However, he then asserts that Chu '684 contains the infrastructure for implementing reformatting. Dr. Houh relies upon his belief of what a person of ordinary skill in the art would understand as to the purpose of dial plans. I disagree with his characterization. Chu '684 discloses no desire, need or method for reformatting a dialed number within the PBX based upon the enterprise internal dial plan, and Chu '366/Chen provide no teaching whatsoever about integrating reformatting into a PBX enterprise dial plan such as that disclosed in Chu '684.

65. Dr. Houh's next argument regarding the combination of Chu '684 and Chu '366 is presented in paragraph 38:

> ***Third, the proposed modification to Chu '684 would be straightforward, would not require undue experimentation, and would produce predictable results. Prepending information such as country codes and area codes to dialed digits has a long history in the PSTN. Upon reading the disclosure of Chu '684, a person of ordinary skill in the art would have recognized that allowing users to place calls as if they were dialing from a standard PSTN phone would be desirable, creating a system capable of supporting a more intuitive and user-friendly interface. One of ordinary skill would also have appreciated that these improvements to Chu'684 could be achieved by merely programming the system of Chu '684 to analyze the dialed digits and reformat as necessary using caller attributes such as national and area code. Such modifications are simply a combination of the system of Chu '684 with elements of Chu '366 that would have yielded predictable results without requiring undue experimentation. Thus, it would have been natural and an application of nothing more than ordinary skill and common sense to combine Chu '684 with the number reformatting of Chu '366.***

66. I disagree with Dr. Houh's conclusion here. To begin with I will point out again that Chu '366 discloses technology that resides at a telephone while Chu '684 discloses technology that resides within a networked PBX, but there is no need to "combine" the two in any meaningful manner. Dr. Houh suggests that an improvement in Chu '684 is desired, but fails to adequately explain the deficiency being addressed. PBX phone users worldwide did not find their phones to be inadequate or unintuitive. PBX's provided all the features of ordinary phones connected to the PSTN, and *in addition*, supported the dialing of private extension numbers. The use of a prefix digit such as "9" was not a deficiency but rather a simple way of supporting both PSTN dialing and extension dialing. Dr. Houh stated during his deposition (**Ex. 2043** at pages 40-43) that he personally would have liked to dial in a way that was inconsistent with the public numbering plan in his area (which switched from 7-digit dialing to 10-digit dialing), but his subjective personal preference is not a deficiency in Chu '684's system. Nor is there any teaching in any of the cited references that would direct one of ordinary skill in the art to his proposed configuration. Further, there is no compelling reason to justify his proposed changes to Chu '684's system. Many people, including users of PBX phone systems, would prefer to dial in the same way as everyone else is dialing for reasons of simplicity and consistency. A skilled person in the field of PBX phone systems would consider that the most "intuitive" and "user-friendly" way of

providing PSTN dialing for most people is simply to support dialing according to the dialing conventions of their geographic area. Thus, in Chu '684, the users would dial PSTN "public numbers" based on the location of the customer premises 105 (see Chu '684 at FIG. 2 and 16:53-54), dial "private numbers" based on the enterprise "private numbering plan" (Chu '684 at 16:49-52), and use a prefix digit (e.g., "9") to conveniently select between the two methods of dialing in an easy and familiar way. These methods of supporting dialing in a PBX have been popular for many decades and still remain popular. I am not aware of any PBX enterprise systems available even today which would allow each user to dial however they wanted.

67. Furthermore, Dr. Houh's testimony in his own deposition makes it clear that significant experimentation would be required for his proposed combination. See Ex. 2044 at 133:15-156:5. It is far from clear how the proposed combination would operate. In his testimony, Dr. Houh presented multiple inconsistent versions of the combination in different scenarios that were presented to him to analyze in response to various problems identified by his questioner. Dr. Houh did not adhere to the specific reformatting algorithms that were disclosed in Chu '366 or Chen, but instead, proposed the addition and deletion of many algorithmic steps in the different scenarios. His testimony includes discussion regarding the difficulty with handling ordinary telephone numbers such as 911,

411 and 101 (Ex. 2044 at pages 135-141). I personally worked for an organization that used the "dial 9 for an outside line" dial plan, which had to be changed because an employee from Canada would dial "911" late at night to try and get an outside line then the number he was used to, only to have emergency services respond. The dial plan was changed to "dial 8 for an outside line" – which resulted in him dialing "811" and getting the "don't dig" service. Of course, Chu '366 indicates that its benefit is that a user can dial their phone as if it is an ordinary PTSN phone line – so it is not clear how a user could even dial an inside extension number based on Chu '366 without exiting the VOIP system. The proposed method of mating a Chu '366 phone method to a Chu '684 network service creates multiple sets of problems, which Dr. Houh had to ward off in his testimony, despite his opportunity to prepare for such an obvious line of questions. Dr. Houh's testimony failed to show how any combination of these two pieces of prior art would provide satisfactory service to any user, let alone how the combination would have been obvious and achieved without undue experimentation.

B. <u>Chu '684 Does Not Disclose a Caller Dialing Profile</u>

68. As I explained above Chu '684 discloses a system that deals with enterprises and each enterprise may have multiple physical locations, each location with multiple individuals and multiple telephones. Thus, in those cases where the

Chu '864 system is able to identify the enterprise that is originating a telephone call it is not consequently identifying an individual user.

69. Dr. Houh argues against this point in paragraph 45 of his declaration. He believes that "Chu '684 discloses a caller dialing profile comprising a username, as claimed in the '815 Patent Challenged Claims". Dr. Houh begins by saying:

> *Based on this example in the specification, I agree with the construction proposed in the Petition that 'username' should be interpreted, at least under the broadest reasonable interpretation, to include any unique identifier associated with a user. Applying this definition, Chu '684 expressly, or at least inherently, discloses a caller dialing profile comprising a username. For example, Chu '684 teaches that each subscriber is assigned their own dial plan, a unique IP address, and a unique E.164-compliant telephone number.* Houh Declaration at ¶45

As I have explained Chu '684 clearly indicates that the subscriber is an enterprise comprising multiple users. Thus, identifying a specific enterprise does not identify a unique user. There is no evidence in Chu '684 that a <u>user</u> is assigned a dial plan – Dr. Houh has constructed this idea out of an initial misunderstanding of the meaning of "subscriber". Contrary to Dr. Houh's statements, Chu '684 discloses that the dial plan and IP addresses are selected by the enterprise, not assigned.

"Each subscriber [i.e., enterprise] can use their own IP address plan as well as their own dial plan." (Chu '684 at 12:62-64). Chu '684 also directly contradicts Dr. Houh by stating that the prior art was deficient because it required an enterprise to be assigned IP addresses while Chu '684 overcomes that limitation (Chu '684 at 2:14-29). Dr. Houh has mischaracterized Chu '684 to suggest that the network service provider assigns these various data elements to the enterprise and thus the Chu '684 service provider has access to this information inherently – but this theory is not supported by Chu '684.

C. <u>Chu '684 Does not Disclose Reformatting</u>

70. Page 11 of the Petition states:

> *Additionally, both Chu '684 and the '815 Patent describe call setup procedures in which a call processor analyzes attributes of the caller and information identifying the callee (e.g., dialed digits) to determine whether the call should be routed to a destination on the private packet network or the public PSTN. Compare* ***Ex. 1003****, Chu '684 at 8:65-9:1 ("At step* ***608****, after receiving all the dialed digits from the phone* ***101****, server* ***110*** *consults its dial plan to determine whether the call is local, to another on-net phone, or to a phone that is on the PSTN.") with* ***Ex. 1001****,*

71. This characterization is wrong – the cited passage in Chu '684 at 8:65-9:1 (describing "step 608") does indicate that the dial plan is consulted based using knowledge of the enterprise – but the analysis of the dialed digits does not require knowledge of the attributes of a caller. Furthermore, the dialed digits are considered to determine if the call is "local, to another on-net phone, or to a phone that is on the PSTN" which does not need to know anything about the callee (including the callee's telephone number). This aspect of Chu '684 does not need to rely upon callee information, for example, the dial plan needs only to specify patterns against which the dialed digits could be compared to determine the type of call. I will explain further why the Petitioner's conclusion is incorrect.

72. As I discussed above at ¶¶ 30-47, PBX systems well known at the time of invention had a well known and widely used method for enterprise users to distinguish between a local (inside or "on-net") and a PSTN (outside or "off-net") number. The practice typically involved dialing the digit "9" and then a PSTN number (i.e., public number) to go outside, or skipping the initial "9" to dial an internal "on net" number (i.e., private number). This practice is still widely used though often the initial number "8" is used for various reasons. I have reviewed the document "MERLIN COMMUNICATIONS SYSTEM CENTREX/PBX CONNECTION" dated March 1985 (**Ex. 2047**), which stated on page 18 "However, if your Centrex/PBX service does not immediately return a dial tone

after dialing to access an outside line (usually by dialing 9)". I have reviewed the document titled "Quick Reference Guide Definity" dated May 4, 2000 (**Ex. 2048**) which shows the use of dialing "9" and then adding a PSTN number to call an outside line (see figure below). I have reviewed a portion of "Understanding Telecommunications Networks" by Andy Valdar with a copyright date of 2006 (**Ex. 2049**), which also established on page 38 that the practice of dialing a "9" to indicate the following number is an outside PSTN number was well understood. Based upon these references, my own experience and my understanding of a person of ordinary skill in the art I feel quite comfortable in stating that PBX systems were commonly known to "examine" a dialed number by looking for a leading "9" and using that information to determine if the callee was to be reached via an inside line (on net) or via an outside PSTN number. In fact, a person of ordinary skill in the art would understand Chu '684 to be referring to the almost universally used process of dialing a "9" to indicate the following number is an outside PSTN telephone number, based in part on the disclosures referenced here. This approach, which is implemented in the clear majority of PBX systems, does not require considering the callee number at all. An example of user dialing instructions from a Definity PBX quick reference guide (**Ex. 2048**) is shown below.

DIALING INSTRUCTIONS

To Place a Local Call
Dial 9 + 7 digit telephone number

To Place a Long Distance Call
Dial 9 + 1 + area code + 7 digit telephone number

To Place an International Call
Dial 9 + 011 + country code + city code + telephone number

To Call Another Extension within Your Company
Dial the last 4 digits of the telephone number

73. Thus, based upon my review of Chu '684 and Chu '366 along with the filed Petition and Dr. Houh's declaration and testimony, it is my opinion that a person of ordinary skill in the art would not have found Chu '366 if searching for prior art to inspire a solution to those problems faced by the inventors of the '815 Patent. Furthermore, if Chu '366 were found, they would not consider the combination of Chu '366 and Chu '684 to be advantageous or desirable. If they did pursue such a combination they would not have achieved an operable system without undue experimentation. And finally, the most likely such solution would not have practiced the challenged claims of the '815.

II. OPINIONS CONCERNING CHU '684 AND CHEN

74. Chen and Chu '366 are the secondary references relied upon by Dr. Houh in discussing obviousness. Like Chu '366, Chen would relate to a different technical category than Chu '684 as it solves completely different problems in a different context (see my comments above in the section entitled "A Person of

Ordinary Skill Would Not Have Combined Chu '684 and Chu '366"). Dr. Houh essentially reuses his arguments regarding the combination of Chu '684 with Chu '366 to form his arguments regarding the combination of Chu '684 with U.S. Patent Publication No. 2007/0064919 (i.e. Chen). I will avoid repeating my comments regarding Chu '684 as presented in the section above related to the combination with Chu '366 as they are applicable here as well. Indeed, the Petitioner states that "the substance between the two secondary references is largely identical." Petition at 37. They further explain that Ground 2 is presented "to account for the possibility that the Patent Owner may attempt to 'swear behind' the Chu '366 reference" while Chen predates the '815 Patent's priority date "by a significant amount." In my view, the comments and faults found with regards to Chu '366 generally apply to Chen and demonstrate the shortcomings of the Petition's arguments regarding combining Chu '684 and Chen.

A. <u>Reformatting</u>

75. I will describe a deficiency in the combination of Chen and Chu '684 proposed by the Petitioners and endorsed by Dr. Houh. The key aspect of this discussion is that Chen reformats a dialed number local to a user before sending the resulting number into the telco system. Chu '366's approach to reformatting is sufficiently similar that my analysis of the defects of the combination presented below apply equally to the combination of Chu '684 and Chu '366.

76. Chen describes his invention thusly: "[0015] A primary object of the present invention is to provide a communication method for placing phone calls by using a fixed dial plan, so that a user in any region or country may place phone calls in a familiar manner." Thus, the Chen system receives a dialed number from a user, reformats that number as appropriate for the given local Telco PSTN switch, and sends the reformatted number to the PSTN switch. The Petitioners rely upon a system combining Chu '684 with Chen where the Chen reformatted number is fed into step 608 in Chu '684. For example, in the context of analyzing Claim 1 of the '815 Patent the Petitioners state on page 27:

> ***<u>Once the callee identifier is reformatted</u>, Chu '684 determines whether the callee is a private packet network subscriber or a public PSTN customer (i.e., whether the call "meets public network classification criteria" or "private network classification criteria").***

Thus, the Petition describes a system in which a callee identifier is reformatted (by the method of Chu '366 or Chen), and once the callee identifier is reformatted, the result is used by Chu '684 to determine the type of call. The Petition indicates that the determining step in Chu '684 is performed by step 608 (Chu '684 at 8:65-9:1). Thus, the output of reformatting becomes the input of Chu '684's step 608. Such a system presents several problems that are not addressed by the Petition or Dr. Houh and would not have obvious solutions to a person of ordinary skill in the art.

77. Chen and Chu '366 are systems designed for exclusively for PSTN calling. Chen states in [0005] that a telephone number in Chen is used to identify the destination in a PSTN. For example, the destination of a call in Chen is always a PSTN destination, which proceeds through a PSTN Gateway (e.g., 502) and/or a PSTN Telco Switch (e.g., 504). PSTN numbers are also called public numbers, because they follow a public numbering plan and the conventions of the public switched telephone network (PSTN).

78. Unlike Chu '684, Chen and Chu '366 are not PBX systems and provide no teaching about PBX dial plans. Unlike a PBX, which would support both private and public numbers in its dial plan, Chen and Chu '366 have no teaching of a private number or a private numbering plan. Private numbers are different than public numbers (e.g., PSTN numbers) in that they do not follow the conventions of the PSTN. Rather, private numbers follow the conventions of the "private numbering plan" (just as "public numbers" follow the conventions of a "public numbering plan"). For example, enterprises may assign short extension numbers, possibly in one or more specific ranges, to phones for internal use through the PSTN (e.g.. extensions 1000-1999) as part of their internal private numbering plan, which would be distinct from the PSTN public numbering plan. Those short extension numbers would be considered private numbers.

79. Chen uses a reformatting algorithm to transform telephone numbers from one format to another. Chen involves receiving a dialed telephone number, transforming this telephone number into E.164 format as shown in FIG. 6, and then transforming the E.164 format number as shown in FIG. 6 into a final format for the PSTN switch at the actual physical local (Chen [0016]).

80. Reformatting in Chen and Chu '366 is only ever practiced on PSTN telephone numbers (i.e., public numbers), as Chen and Chu '366 have no concept of private numbers or private network calls. The reformatting methods taught by Chen and Chu '366 could be applied in other systems, but they are only valid for PSTN numbers. The reason is that the reformatting relies on specific assumptions about the properties of the numbers being formatted, such as whether they follow certain PSTN conventions. For example, reformatting may involve matching a leading portion of a number, removing portions of a number, adding digits to the number by prepending, etc. See, e.g., FIG. 6 of Chen (NDD, IDD, etc). Reformatting in the sense of Chu '366 or Chen, does not have any validity for private numbers. This is because private numbers don't follow the PSTN conventions. Thus, the reformatting algorithm in Chen or Chu '366 assumes that any input that it may receive is a valid public number (i.e., PSTN phone number). For example, one possible assumption is that if the leading digit is "1", then this number represents a long-distance PSTN call (e.g., NDD in FIG. 6). While that

may be true in some jurisdictions (e.g., the U.S.), this assumption is invalid in *private* numbers and *private* numbering plans, which do not follow the PSTN conventions. For example, in a private numbering scheme or plan where extension ranges 1000-1999 are defined, providing the private number of 1111 into formatting will make the reformatting method (mistakenly) think that the leading 1 is an NDD. However, in this instance the leading 1 is not an NDD. The number may be an alternative way of dialing a destination within a PBX.

81. A skilled person would appreciate that a private number should not be fed into an algorithm or method which is designed for matching or manipulating public numbers, because the private numbers could be misclassified. If there are manipulations involved in the algorithm or method, again, the private number should not be processed because it could get corrupted. If such a corrupted private number were used to place a call, the result would be unpredictable: it would likely result in an invalid call or connecting to a "wrong number". This is an error that is committed in the combinations proposed by the Petitioner.

82. In a PBX system (unlike in Chu '366 or Chen), a phone may dial either a private number (extension) or a public number (PSTN phone). The PBX cannot validly do reformatting of the number if is a private number; the PBX can only reformat PSTN phone numbers. Thus, if one needed reformatting (for some reason) in a PBX, one would only apply reformatting to the public numbers; one

would not apply reformatting to the private numbers. If the type of a number that was received in a PBX is unknown, then a skilled person would know to classify the number first (as public or private). Once the number is classified, then one would reformat only the public numbers, while not altering the private numbers. A private number that is changed (e.g., corrupted) has a different "meaning" in the private numbering plan that defined it, whereas a public number usually can be transformed between two or more predetermined valid different formats, all of which have the same "meaning" within the public numbering plan that defined the public number.

83. Chen is not a PBX system and thus has no concept of an outside line. Nor does it have private numbers, or a private numbering scheme. In most situations a leading 9 will indicate to Chen a number that has no E.164 equivalent. Or a Chu '684 user could dial 8 to broadcast over the intercom to all telephones in an enterprise, or an extension (e.g. "384"). Both examples are too short for a corresponding E.164 number. Chen would wait for additional digits to be dialed (thus frustrating the user) or, if directed to simply pass the numbers on using some mechanism not disclosed, not be able to generate a corresponding E.164 number. If somehow Chen did generate an E.164 number there is no disclosed method for Chu '684 to determine if the number is an internal number (such as the two

examples above) or a number that can only be reached by leaving the PBX/VOIP system.

84. Thus, the Petitioner's reasoning is flawed. If Chen does reformat the dialed number it will send Chu '684 an E.164 number that will by definition be considered a PTSN number by Chu '684. In either of Petitioner's proposed combinations, the modified system does not first reformat and then decide whether the dialed digits are internal or external.

B. <u>Multiple Dial Plans</u>

85. The proposed combinations of either Chu '366 or Chen with Chu '684 are unreasonable at least because they combine a user specific dial plan, an enterprise dial plan, and the E.164 final dial plan in an untenable and unmanageable manner.

86. As I discussed above, enterprises design dial plans to serve business goals such as ease of use and uniformity across the enterprise. Having a single dial plan makes it easier for an enterprise's IT department to manage training by having uniform and reliable directions for dialing. Providing technical support to users is also easier because each telephone is configured to operate in a consistent manner. An enterprise dial plan is not user-specific by design.

87. Having a single dial plan also eliminates the possibility of having dial plans with conflicting requirements (e.g., one dial plan defining a digit "0" as

being the digit to call the attendant, a second dial plan defining the digit "0" as being the PSTN prefix digit or outside line access code, a third dial plan interpreting the digit "0" as a request to call a telephone company operator, and a fourth dial plan interpreting the digit "0" as the German national dialing digit (NDD) prefix of "0" for long-distance calling). An enterprise would find it undesirable to support user-specific dial plans in a PBX in order to avoid such conflicts.

88. Chu '366 and Chen upend the cart by allowing the individual user to change the dial plan of a specific telephone system, whereas in a PBX system such as Chu '684 it is paramount for the enterprise to control its own (enterprise-wide) PBX dial plan, for the reasons I have given above. Dr. Houh was unable to explain in his deposition testimony how the combination of these two different types of dial plans would work together because they were not designed to work together and the combinations introduce unexpected behavior and operational problems. Enterprise dial plans are carefully designed to meet enterprise goals such as ease of use with no exceptions allowed that would bar users from accessing necessary services or unnecessarily burden an enterprise's IT department. The proposed introduction of user-specific dial plans into a PBX having an enterprise-specific dial plan would result in an unpredictable and often inoperative system.

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

Dated: 2/10/17 By: _____