(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
3 July 2003 (03.07.2003)

PCT

(10) International Publication Number
WO 03/055185 A1

(51) International Patent Classification[7]: H04M 3/22

(21) International Application Number: PCT/US02/40952

(22) International Filing Date: 19 December 2002 (19.12.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/342,573 20 December 2001 (20.12.2001) US

(71) Applicant *(for all designated States except US)*: **TEKELEC** [US/US]; 26580 West Agoura Road, Calabasas, CA 91302 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **FREEDMAN, Steven, Michael** [US/US]; 3912 Bluffwind Drive, Raleigh, NC 27603 (US). **GALIZIA, Patrick, Joseph** [US/US]; 108H Northington Place, Cary, NC 27513 (US). **SPANGLER, Richard, James** [US/US]; 11409 North Radner Way, Raleigh, NC 27613 (US). **WAN, Joseph, Yu-Lung** [US/US]; 2677 Aster Drive, Richardson, TX 75082 (US).

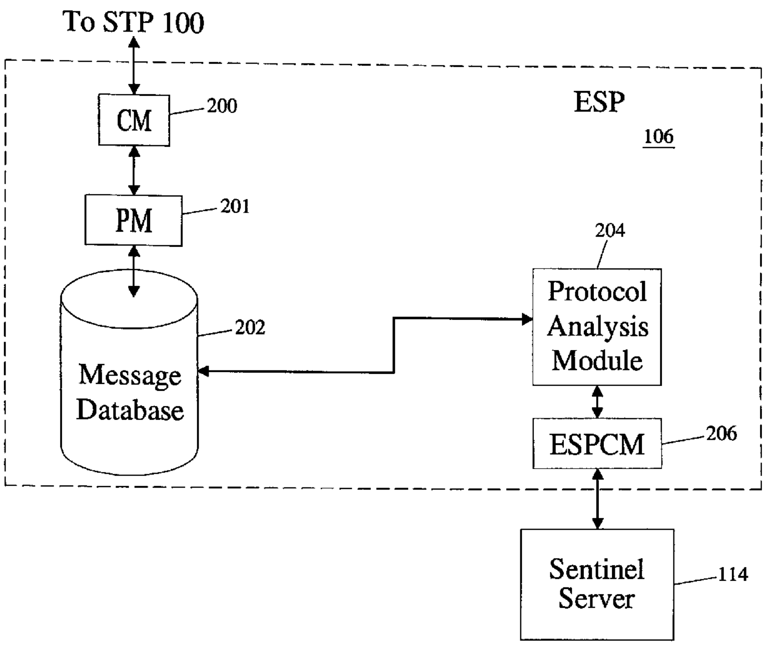
(74) Agent: **JENKINS, Richard, E.**; Jenkins & Wilson, P.A., University Tower, Suite 1400, 3100 Tower Boulevard, Durham, NC 27707 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SI, SK,

[Continued on next page]

(54) Title: DATABASE DRIVEN METHODS AND SYSTEMS FOR REAL TIME CALL TRACING



(57) Abstract: Database-driven methods and systems for database driven real time call tracing are disclosed. Signaling messages are copied from signaling link interface modules. The copied messages are sent to a database (202). A server (114) located remotely from the database (202) sends a query to the database (202) for a real time call trace. The database (202) is searched for messages that match the search criteria. If no messages are located in the historical data stored in the database, data entering the database (202) is analyzed in real time.

WO 03/055185 A1

TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Description

DATABASE DRIVEN METHODS AND SYSTEMS FOR REAL TIME CALL TRACING

Related Applications

This application claims the benefit of U.S. Provisional Patent Application Serial No. 60/342,573, filed December 20, 2001, the disclosure of which is incorporated herein by reference in its entirety.

Technical Field

The present invention relates to methods and systems for real time call tracing. More particularly, the present invention relates to database driven methods and systems for real time call tracing.

Related Art

In order to diagnosis problems, detect fraud, and perform security screening in telecommunications networks, it is often necessary to capture, in real time, signaling messages used to establish, maintain, and tear down calls. The capturing of signaling messages relating to a call in real time is referred in the telecommunications industry as real time call tracing. Real time call tracing can be difficult because the signaling links that carry signaling messages may operate at different speeds and signaling messages relating to the same call may be sent over different links.

Conventional real time call tracing systems utilize signaling link probes to copy and buffer messages from signaling links. For example, United States Patent Application Publication No. US 2002/0071530 (hereinafter, "the '530 Publication") discloses a real time call tracing system that uses link probes copy messages from signaling links. Each link probe includes a probe buffer. Messages received from a signaling link are stored in the probe buffer. Trigger criteria are applied to the signaling messages stored in each probe buffer. If a signaling message stored in a probe buffer matches a trigger condition, the message is sent to an element manager associated with the probe that discovered the signaling message. The element manager formulates and sends a new trigger condition to each of the signaling link probes. Each of the signaling link probes analyzes messages in its probe buffer based on the new trigger criteria.

One problem with conventional real time call trace systems, such as that described in the '530 Publication, is that the memory and processing capacity of signaling link probe devices is limited. Because of the limited storage capacity at conventional signaling link probes, many conventional real time call trace systems utilize circular buffers at signaling link probes. Using circular buffers at signaling link probes allows data to be continuously written to a signaling link probe buffer. However, if data at the beginning of the buffer is not read before the buffer becomes full, the data will be overwritten with new data, and the signaling messages needed to perform a real time call trace may be lost.

Another problem with probe-buffer-based call tracing systems, such as that described in the '530 Publication, is that the systems do not truly operate in real time. For example, analyzing messages stored in a buffer means that the messages have already been received and are not being analyzed in real time as they arrive. Moreover, if a message is not found in the historical data stored in a buffer, the call trace may fail.

Yet another problem associated with conventional probe-buffer-based call tracing systems is that signaling link probe buffers may utilize volatile memory for storing received signaling messages. Storing these signaling messages in volatile memory will result in loss of data if a signaling link probe loses power.

Yet another problem associated with conventional probe-buffer-based call tracing systems is that link probes typically consist of dedicated message copying hardware with proprietary interfaces on the message analysis side. Using dedicated hardware with proprietary interfaces limits the ability to write real time call trace applications that interface with the hardware and have access to the messages stored in the signaling link buffers.

Accordingly, in light of these difficulties associated with conventional real time call tracing systems, there exists a need for improved methods and systems for real time call tracing.

Disclosure of the Invention

The present invention includes improved methods and systems that use a database rather than link probe buffers for real time call tracing. In one exemplary implementation, the real time call tracing system is completely probeless. Messages are copied directly from signaling link interface cards in a signaling message routing node, such as a signal transfer point. The copied messages are sent to a network monitoring processor associated with the signal transfer point. The network monitoring processor stores the messages in a relational database. A network monitoring server located remotely from the network monitoring processor sends a request to the network monitoring processor for a real time call trace. The network monitoring processor searches the relational database for historical data copied from the signaling link for the message. If the message is not located in the historical data, the network monitoring processor analyzes messages in real-time as the message are received.

Utilizing a database, rather than link probe buffers, greatly increases the scalability, reliability, and flexibility of a real time call tracing system. For example, the database may be stored in memory on a general-purpose computing platform separate from signaling link probe hardware. Because the database can be located on any suitable general-purpose computing platform, the number of messages that can be buffered for real time call tracing purposes is scalable based on hard disk size. The reliability of the system is also increased because the database is stored in nonvolatile memory located on one or more system disks. The flexibility of the system is increased because the database can be located on any suitable network element at any suitable location in the network.

Accordingly, it is an object of the present invention to provide improved methods and systems for real time call tracing that utilize a database rather than link probe buffers for real time call tracing.

It is another object of the invention to provide a method for real time call tracing that combines the analysis of historical data stored in the database with data received in real time.

Some of the objects of the invention having been stated hereinabove, and which are addressed in whole or in part by the present invention, other objects will become evident as the description proceeds when taken in connection with the accompanying drawings as best described hereinbelow.

Brief Description of the Drawings

Preferred embodiments of the invention will now be explained with reference to the accompanying drawings of which:

Figure 1 is a block diagram of a database driven system for real time call tracing according to an embodiment of the present invention;

Figure 2 is a block diagram of a network monitoring processor for database driven real time call tracing according to an embodiment of the present invention;

Figure 3 is a block diagram of a database for real time call tracing according to an embodiment of the present invention;

Figure 4 is a flow chart illustrating exemplary steps for real time call tracing according to an embodiment of the present invention; and

Figure 5 is a block diagram of an MSU block that may be included in the database illustrated in Figure 3.

Detailed Description of the Invention

The present invention may include a network monitoring platform, part of which may be integrated with an STP platform, such as the EAGLE® STP platform available from Tekelec of Calabasas, California. Figure 1 illustrates an exemplary architecture for a network monitoring platform on which embodiments of the present invention may be implemented. In Figure 1, reference numeral **100** represents an STP. STP **100** includes a plurality of link interface modules (LIMs) **102** that send and receive SS7 messages via SS7 signaling links. Although not illustrated in Figure 1, STP **100** may also include data communication modules (DCMs) for sending and receiving IP messages via IP signaling links. Sentinel transport cards (STCs) **104** route messages between LIMs **102** and network monitoring processors, such as EAGLE®/Sentinel processors (ESPs) **106**. LIMs **102** and STCs **104** are connected via IMT bus **108**. ESPs **106** each include a database that stores MSUs and alarm messages received from STP **100**.

From a hardware perspective, each LIM **102** and STC **104** may be a printed circuit board with an application processor and a communications processor. The application processor executes application level software for performing various functions. The communications processor communicates with other modules connected to inter-processor message transport (IMT) bus **108**. IMT bus **108** includes a pair of counter rotating rings. ESPs **106** may be general-purpose computing platforms located external to STP **100** and connected to STP **100** via Ethernet connections.

From a software perspective, LIMs **102** may include TCP/IP or SCTP/IP protocol software for establishing TCP/IP or SCTP/IP connections with ESPs **106** through STCs **104**, network monitoring client software for requesting network monitoring services from ESPs **106**, MSU copy functions for copying incoming and outgoing MSUs, and SS7 alarm functions for generating alarm notifications when certain events, such as signaling link failures, occur. LIMs **102** may encapsulate MSUs and alarm notifications in specialized packets that indicate the source of the MSUs or alarm messages. LIMs **102** may also communicate provisioning information to ESPs **106** to enable automatic configuration of ESPs **106** when a signaling link is added or deleted.

ESPs **106** may include server software that responds to service requests from LIMs **102**. Each ESP may be associated with a predetermined set of signaling links. For example, one ESP may be provisioned to handle signaling links 0-127 and another ESP may be provisioned to handle signaling links 128-255. When a LIM boots up with the network monitoring client software enabled, the LIM may broadcast a service request message to ESPs **106**. The ESP provisioned to handle the request for that particular LIM responds to the service request. If the response is a service acceptance, the requesting LIM establishes a TCP/IP connection with the responding ESP and begins sending copied messages and alarms to the ESP. The ESP receives and buffers the messages and alarms.

ESPs **106** may communicate with a server farm **110** via IP network **112**. Server farm **112** includes a Sentinel server **114**, a data gateway server **116**, an alarm server **118**, and a database server **120**. Sentinel server **114** may perform the following functions: real time signaling link status reporting, real time signaling link state reporting, real time protocol analysis, such as real time call tracing, filtering, and decoding, traffic report generation, and real time event reporting. Data gateway server **116** receives MSU fragments, formats the MSU fragments into call detail records (CDRs) and sends the CDRs to applications, such as fraud detection applications, billing applications, billing verification applications, etc. Alarm server **118** collects event message reports and other events that report signaling link errors and displays alarms to the user. Database server **120** is connected to Sentinel server **114**. Sentinel server **114** may generate standardized traffic reports in flat ASCII format. Some end users may desire to generate customized traffic reports. In order to allow the generation of customized traffic reports, database server **120** may store the data collected by Sentinel server **114** in a database, such as an Oracle database. A database front end, such as Crystal Reports available from Seagate Software may be used along with database server **120** to generated customized reports.

Because the system illustrated in Figure 1 copies signaling messages directly from link interface modules within the network node, such as STP **100**, the need for link probes is reduced. However, the real time call tracing methods of the present invention are not limited to use with completely probeless network monitoring systems. For example, the real time call tracing methods of the present invention may be used with probe based network monitoring systems, such as the i2000 or i3000 shelves available from Tekelec of Calabasas, California. In addition, the real time call tracing methods of the present invention may be used with network monitoring systems that include both probeless and probe based message copy functions. However, unlike conventional probe based real time call tracing systems, the methods and systems of the present invention preferably use a database located on a computing platform separate from signaling link probe hardware.

Call Tracing

As stated above, one of the potential applications of the network monitoring system architecture illustrated in Figure 1 is call tracing. The present invention includes both real time call tracing where messages are examined immediately as they are received into a database and non-real time call tracing where historical data is scanned for messages of interest. In a preferred embodiment of the invention, real-time and non-real time call tracing may be used together to increase the likelihood that all messages relating to a call are captured. Call tracing can be used for a variety of different applications, including security, problem diagnosis, and fraud detection.

Figure 2 illustrates exemplary components of ESPs **106** and Sentinel server **114** that may be associated with real time and non-real time call tracing according to an embodiment of the present invention. In Figure 2, ESP **106** may include a communications module **200** for managing resources of ESP **106** for monitoring ports of STP **100**. In particular, when a port on STP **100** is activated for monitoring, that port may send a service request via a UDP broadcast to a well known port. Communications module **200** receives the request, checks to see whether this port is provisioned on the ESP and forwards the request to port monitor **201**. Port monitor **201** processes the service request message and sends back a service accept message. The port requesting service on STP **100** may then establish a TCP/IP connection with port monitor **201**, send provisioning information to port monitor **201**, and begin sending copies of MSUs to port monitor **201**. Port monitor may **201** store the MSUs in a real time MSU database **202**.

A protocol analysis module **204** may poll database **202** in response to requests received from Sentinel server **114**. For example, protocol analysis module **204** may extract MSUs having a particular timestamp value from database **202** in response to requests from Sentinel server **114**. Protocol analysis module **204** may filter and examine the MSUs in database **202**. An

ESP communication manager **206** controls communications between ESP **106** and external applications, such as Sentinel server **114**.

In a preferred embodiment, database **202** comprises a relational database divided into tables according to signaling links being monitored. Figure 3 illustrates a relational database for real time call tracing according to an embodiment of the present invention. Referring to Figure 3, database **202** includes a plurality of link tables **300A-300N**. Each link table stores messages copied from a particular signaling link. Within each link table, the signaling messages are preferably arranged in blocks **302**. Each block includes a plurality of messages. Each block may be indexed by a beginning timestamp corresponding to the first message in the block. Subsequent messages in each block may be indexed using an offset from the initial time stamp.

Indexing messages using blocks and offsets as illustrated in Figure 3 improves the efficiency of searching for messages when performing a call trace. For example, in conventional methods that use the link buffers, each message is required to be searched sequentially in order to locate a message having a particular timestamp. In contrast, in database **302** illustrated in Figure 3, if a timestamp of interest is not within a message block, messages within that block can be skipped. As a result, the efficiency of searching the database for messages to be included in a real time call trace is improved.

In a preferred embodiment of the invention, a portion of database **202** is stored in random access memory on a general purpose computing device and another portion of the database is stored in disk memory. When protocol analysis module **204** attempts to query database **202**, the query is preferably first directed to the portion of database **202** stored in random access memory. If the query to the portion of database **202** stored in random access memory results in a hit, the message is returned from random access memory. If the query to the portion of database **202** stored in random access memory does not result in a hit, the portion of database **202** stored in disk memory is searched. If the search of the portion of database **202** stored in disk memory results in a hit, the link table that included the message that resulted in the hit is moved to random access memory to speed up subsequent accesses to the same range of messages.

Figure 4 is a flow chart illustrating exemplary steps performed by Sentinel server **114** and ESP **106** in performing a real time call trace according to an embodiment of the present invention. Referring to Figure 4, in step **ST1**, Sentinel server **114** sends a request for a real time call trace to ESP **106**. For example, if a customer calls into a service provider's call center and indicates that he or she is unable to make a call from a particular directory number, the request may specify a real time call trace for a call from a calling party number corresponding to the customer's directory number. After step **ST1**, the service provider may instruct the customer to place a call from the particular directory number.

In step **ST2**, ESPCM **208** receives the request for a real time call trace and forwards the request to protocol analysis module **206**. For example, in the failed call scenario, the customer service representative employed by the service provider may instruct the customer to reattempt the call at 12pm. In this case, protocol analysis module **204** may attempt to locate MSUs having the particular calling party number and a timestamp greater that 12pm. Accordingly, in step **ST3**, protocol analysis module **204** queries the database for MSUs having the particular calling party number and a timestamp greater than 12pm. In steps **ST5A** and **5B**, protocol analysis module **204** continues querying the database until the MSU is located. Once the MSU is located, in step **ST5C**, protocol analysis module **204** sends the message to Sentinel server **114**.

In step **ST6**, Sentinel server **114** receives the message and sends the timestamp of the message to all other ESPs participating in the real time call trace. In step **ST7**, the other ESPs participating in the real time call trace search their respective message databases until the timestamp is reached. In this example, it is assumed that the other ESPs participating in the call trace do not find an MSU with a time stamp earlier than the time stamp located by the ESP that first detected a match on the calling party number. Accordingly, in step **ST7**, the other ESPs notify Sentinel server **114** that match with a timestamp earlier than that of the first-detected message was not located with a timestamp earlier than that of the first-detected message. If any of the other ESPs had located a matching MSU prior to the timestamp of the MSU detected by the first ESP, the ESPs would send that MSU to Sentinel server **114**, and Sentinel server **114** would keep the earliest matching MSU having the particular timestamp.

Once Sentinel server **114** receives the first message in the call trace having the earliest time stamp, in step **ST8**, Sentinel server **114** sends new filters to the ESPs. Different filtering instructions may be sent to the ESP that detected the first message than to the other ESPs. For example, to the ESP that detected the first message, it is known exactly where in the database the matching MSU is stored. Accordingly, Sentinel server **114** may instruct that ESP to begin searching at the location in the database immediately following the first detected MSU. If the real time call trace is an ISUP call trace, the key or filter for searching the database may be a combination of originating point code (OPC), destination point code (DPC), and circuit identification code (CIC). For databases on ESPs that did not detect the first message, Sentinel server **114** may instruct these ESPs to begin searching at the timestamp of the first received message. The filter criteria sent to the other ESPs may also be the combination of OPC, DPC, and CIC.

In step **ST9**, each ESP searches historical data for the next message satisfying the filter criteria. In step **ST10**, if the message is located, control proceeds to step **ST11** where the message is forwarded to sentinel server **114**. In step **ST12**, if the message is the last message in the call trace, real time call tracing ends. If the message is not the last message, steps **ST8**-**ST11** are repeated for the next message.

Unlike conventional probe-buffer-based call trace applications, if the message is not located in the database, ESPs **106** of the present invention do not stop the real time call trace and indicate that a failure occurred. Rather, once the historical data has been searched and no match is indicated, ESPs

106 preferably continue searching data as it arrives. This combination of searching historical data and searching data in real time increases the likelihood that all of the messages relating to a call trace will be captured. Accordingly, in step **ST10**, if no message is located in the historical data, control proceeds to step **ST13** where ESPs **106** query their respective databases for real time data. In step **ST14**, ESPs **106** determine whether a matching message is located. If a matching message is not located, ESPs **106** continue to query the database in real time. The databases may be queried for a predetermined time period after which ESPs **106** send an error message to Sentinel server **114**. Once a matching message is located, control proceeds to step **ST11** where the message is forwarded to Sentinel server **114**. Thus, using the steps illustrated in Figure 4, the reliability of real time call tracing is increased.

Figure 5 illustrates the concept of searching historical data followed by searching real time data. In Figure 5 message block **304**, which may be located in database **202**, stores messages received from STP **100** in sequential memory locations. Location **500** stores the first message received for the call trace, which for calls established using SS7 signaling may be an IAM message. Locations above location **500** in block **304** store messages received before the IAM message, and locations below location **500** in block **304** store messages received after the IAM message. According to the present embodiment, the search for the next message preferably begins at location **502**, since this message stores the next message received after the IAM message and continues through location **504**, which stores the last message in the block. If additional message blocks are present in the database **202**, these blocks should also be searched. Because database **202** may include separate tables for each signaling link, each table for each signaling link may be searched for messages matching the OPC/DPC/CIC condition with a timestamp later than that of the IAM message. If multiple ESPs **106** are used to monitor a set of signaling links, the database at each ESP may be searched for the matching signaling message.

In Figure 5, message block **304** includes a block timestamp **506** corresponding to the time that the first message in block **304** was received. Each message in block **304** has a block timestamp indicating the offset in microseconds from block timestamp **506**. For example, the offset of the first message in the block is 0 because this message corresponds to the block timestamp. Subsequent messages, such as IAM message **500** have non-zero offset values corresponding to the times that the messages are received relative to block timestamp **506**. Indexing messages using a block timestamp and offsets from the block timestamp greatly decreases search time in database **202**.

Using a database versus using link buffers greatly enhances the efficiency, flexibility, and reliability of real time call tracing over conventional methods. For example, because messages are stored in a relational database, the database can be accessible using applications written using standard database query languages, such as structural query language (SQL). In addition, because the database may be stored in non-volatile memory and/or on one or more system disks, the storage capability of the database is scalable based on system disk size. Decoupling the database from link probes allows the database to be located on any suitable computer platform at any suitable location in the network. Finally, using a database rather than dedicated hardware to store messages facilitates interaction with third party applications. For example, database **202** may include an open database connectivity front end for applications to access database **202** over a network.

Although the examples described herein have related primarily to real time call tracing where the calls are set up using ISUP messaging, the present invention is not limited to tracing calls set up using ISUP messaging. The methods and systems described herein for real time call tracing may be used to trace call signaling messages in any suitable call signaling protocol, such as session initiation protocol (SIP), H.323, media gateway control protocol (MGCP), or other call signaling protocol.

It will be understood that various details of the invention may be changed without departing from the scope of the invention. Furthermore, the foregoing description is for the purpose of illustration only, and not for the purpose of limitation, as the invention is defined by the claims as set forth hereinafter.

CLAIMS

What is claimed is:

1. A method for real time call tracing, the method comprising:
   (a) sending, from a location remote from a signaling link, a filter condition for locating a message in a real time call trace;
   (b) searching, in a database, historical data copied from the signaling link for a message matching the filter condition;
   (c) in response to failing to locate the message in the historical data, polling the database in real time for the message matching the filter condition; and
   (d) in response to locating the message in steps (b) or (c), forwarding the message to the remote location.
2. The method of claim 1 wherein step (b) includes searching historical data in a database separate from signaling link probe hardware.
3. The method of claim 1 wherein step (b) includes searching historical data in a database including messages copied by a message copy function located in a network routing node.
4. The method of claim 1 wherein step (b) includes searching historical data in a relational database, the relational database including a plurality of tables, each table corresponding to a signaling link, each table including a plurality of message blocks, each message block including blocks of copied signaling messages indexed according to a timestamp of a first message in each block and offsets from the timestamp.
5. The method of claim 1 wherein step (b) includes searching historical data stored on a general purpose computing platform.
6. The method of claim 5 wherein a first portion of the database is stored in random access memory on a general purpose computing platform and a second portion of the database is stored in disk storage on the general purpose computing platform, and wherein step (b) includes searching random access memory for the message and in response to failing to locate the message in random access memory, searching disk storage for the message, and in response to locating the message in disk

storage, moving a block of memory including the message to random access memory.

7. The method of claim 1 wherein searching historical data includes searching data in the database beginning at a database record corresponding to a previously received message in the call trace.

8. The method of claim 1 comprising, at the remote location, in response to receiving the message, formulating and sending a request for a subsequent message based on information in the received message.

9. A method for real time call tracing, the method comprising:

   (a) receiving messages copied from a plurality of signaling links;

   (b) storing the messages in a database; and

   (c) performing a real time call trace based on the information stored in the database.

10. The method of claim 9 wherein receiving messages copied from a plurality of signaling links includes receiving messages copied directly from the signaling links by a message copy function located internal to a signal transfer point.

11. The method of claim 9 wherein receiving signaling messages copied from a signaling link includes receiving signaling messages copied by signaling link probes.

12. The method of claim 9 wherein storing the signaling messages in a database includes storing the signaling messages in a relational database including a plurality of tables, each table corresponding to a signaling link, each table being arranged in message blocks, each message block being indexed by a timestamp of a first received message in the block and offsets from the timestamp.

13. The method of claim 9 wherein storing the messages in a database include storing the messages in a database located on a general purpose computing platform remote from the signaling links.

14. The method of claim 9 wherein performing a real time call trace based on the data stored in the database includes searching historical data in the database for messages relating to the real time call trace and, in

response to failing to locate the messages in the database, searching data that arrives in the database in real time.

15. A system for real time call tracing, the system comprising:

(a) a computing platform operatively associated with a signal transfer point or signaling gateway for receiving signaling messages copied from signaling links connected to the signal transfer point or signaling gateway;

(b) a real time database located on the computing platform for storing the copied messages;

(c) a protocol analysis module located on the computing platform for polling the database for messages based on filter criteria, wherein the protocol analysis module is adapted to search historical data in the database and to search messages received in the database in real time; and

(d) a network monitoring server located remotely from the computing platform for requesting real time call trace information by sending queries including filter criteria to the protocol analysis module.

16. The system of claim 15 wherein the computing platform comprises a general purpose computing platform.

17. The system of claim 16 wherein the computing platform includes random access memory and disk memory, and wherein the real time database includes a first portion stored in the random access memory and a second portion stored in the disk memory.

18. The system of claim 17 wherein the protocol analysis module is adapted to poll the portion of the real time database stored in the random access memory, and in response to failing to locate messages in the random access memory, the protocol analysis module is adapted to poll the disk memory.

19. The system of claim 15 wherein the real time database includes a plurality of tables, each table corresponding to a signaling link and including a plurality of message blocks, each message block storing a plurality of signaling messages copied from one of the signaling links.

20. The system of claim 19 wherein each message block is indexed by a timestamp of a first message in each block and subsequent messages in each block are indexed by offsets from the timestamp of the first message in each block.

21. The system of claim 15 wherein the protocol analysis module is adapted to search the historical data for real time call trace information, and, in response to failing to locate the information in the historical data, to poll the database for the real time call trace information.

22. The system of claim 15 wherein the network monitoring server is adapted to formulate real time call trace requests based on real time call trace information received from the computing platform in response to prior requests.

23. A computer program product comprising computer executable instructions embodied in a computer-readable medium for performing steps comprising:

(a) storing signaling messages copied from telecommunications signaling links in a relational database;

(b) receiving filter criteria for performing a real time all trace based on signaling messages stored in the database;

(c) querying historical data in the database based on the filter criteria; and

(d) in response to failing to locate message matching the filter criteria, querying the database in real time for messages matching the filter criteria.

24. The computer program product of claim 23 wherein storing signaling messages copied from telecommunications signaling links in a relational database includes storing signaling messages received from a probeless network monitoring system associated with a signal transfer point.

25. The computer program product of claim 23 wherein storing signaling messages copied from telecommunications signaling links includes storing signaling messages copied by link probes from

telecommunications signaling links in a relational database separate from the link probes.

26. The computer program product of claim 23 wherein storing signaling messages copied from telecommunications signaling links in a relational database includes storing signaling messages in a database including a plurality of tables, each table corresponding to a signaling link and including a plurality of message blocks, each message block being indexed by a timestamp of a first signaling message stored in each message block and subsequent messages in each message block being indexed by an offset from the timestamp of the first message.

27. The computer program product of claim 23 wherein receiving filter criteria for performing a real time call trace includes receiving message parameters associated with the call trace.

28. The computer program product of claim 23 wherein querying historical data in the database includes querying data from a last received signaling message associated with a call trace until the end of the database.

29. The computer program product of claim 23 wherein querying the historical data includes searching a first portion of the database stored in random access memory and, in response to failing to locate a message matching the filter criteria in the random access memory, searching a portion of the database stored in disk memory.

30. The computer program product of claim 29 comprising in response to locating a message matching the filter criteria in disk memory, moving a portion of the database stored in disk memory containing the matching message to random access memory.

1/5

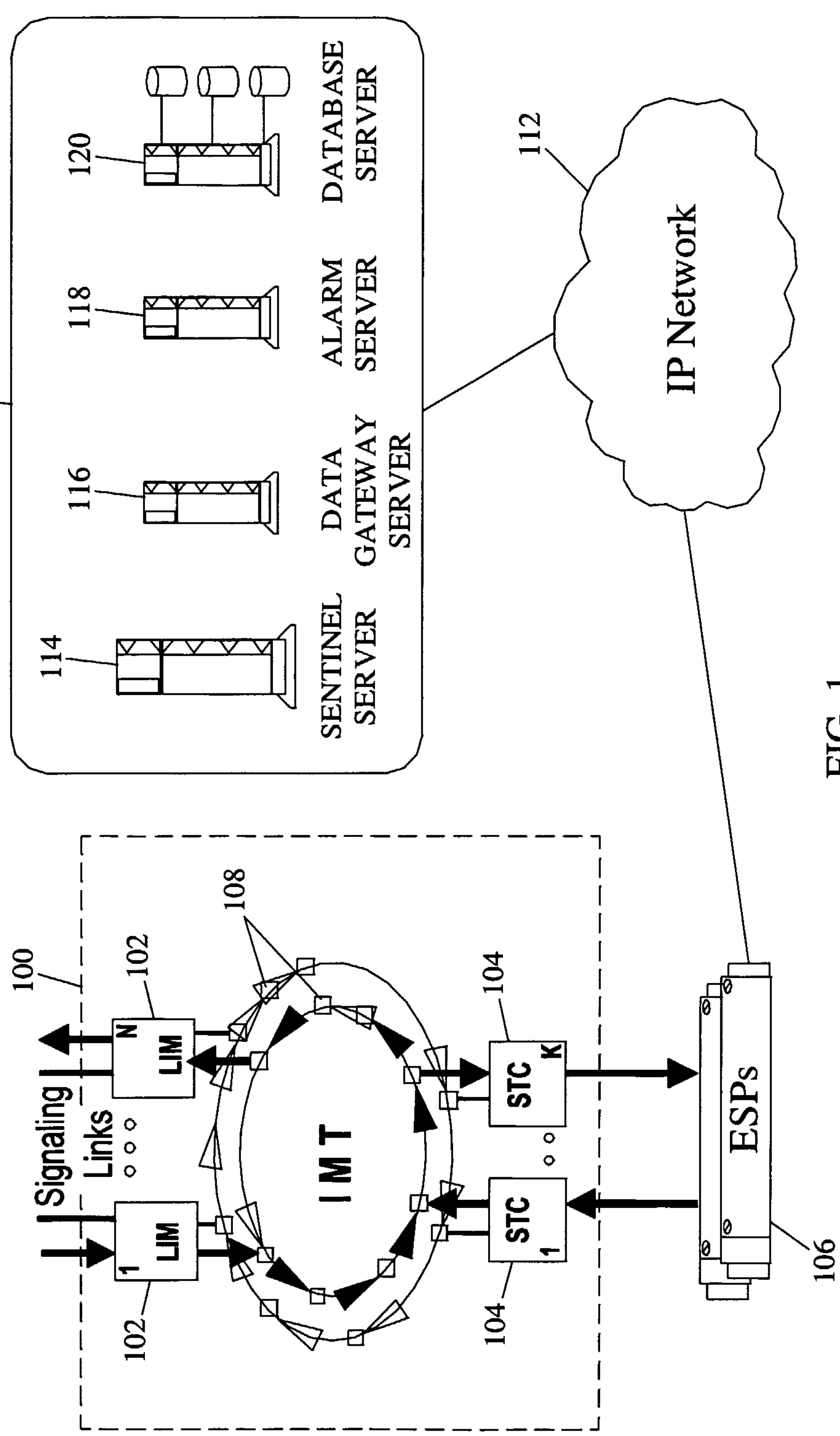

110
114
SENTINEL SERVER
116
DATA GATEWAY SERVER
118
ALARM SERVER
120
DATABASE SERVER
112
IP Network
100
102
102
108
104
104
Signaling Links
N
LIM
1
LIM
IMT
STC
K
STC
1
ESPs
106

FIG. 1

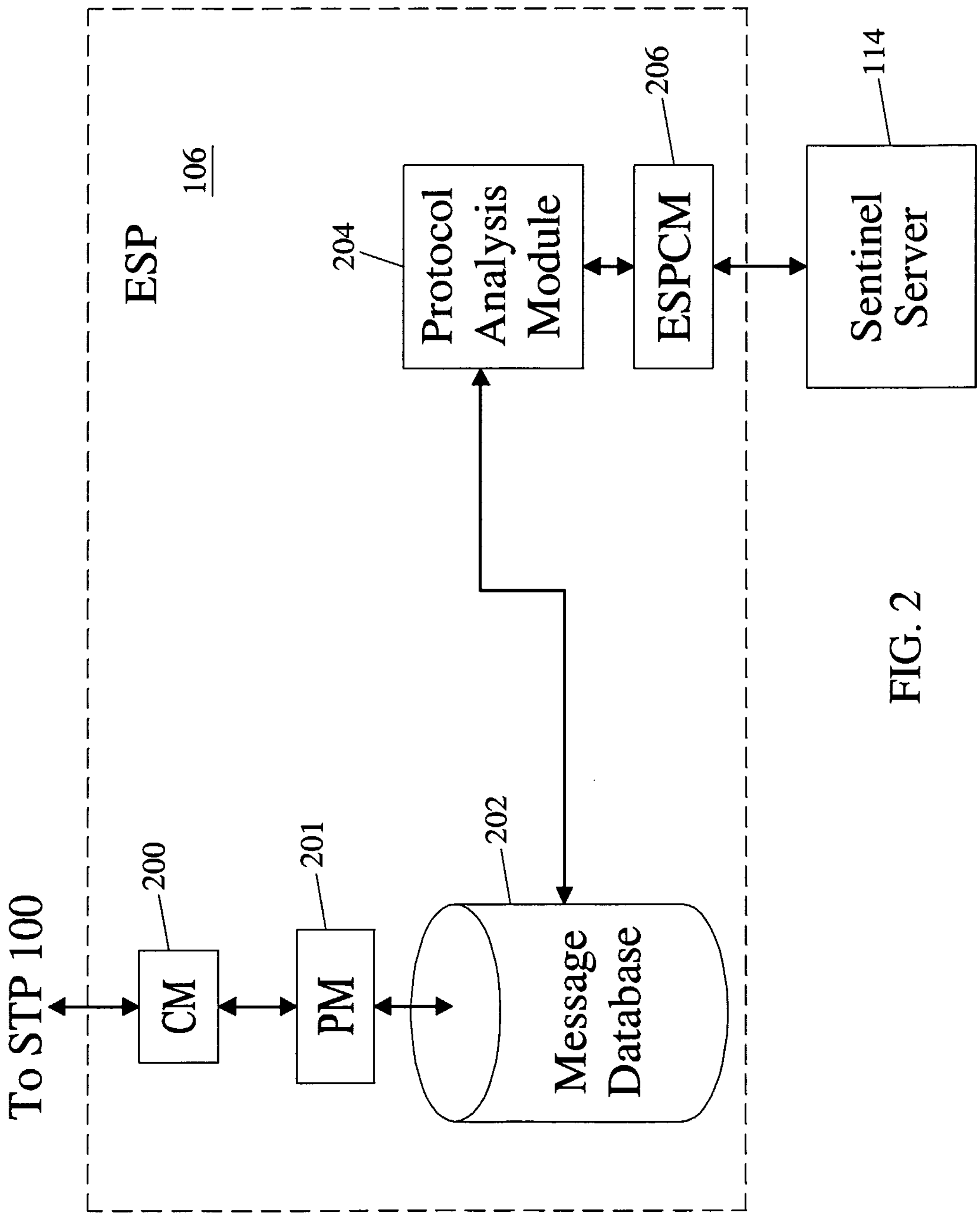

To STP 100
CM
200
PM
201
Message Database
202
ESP
106
Protocol Analysis Module
204
ESPCM
206
Sentinel Server
114

FIG. 2

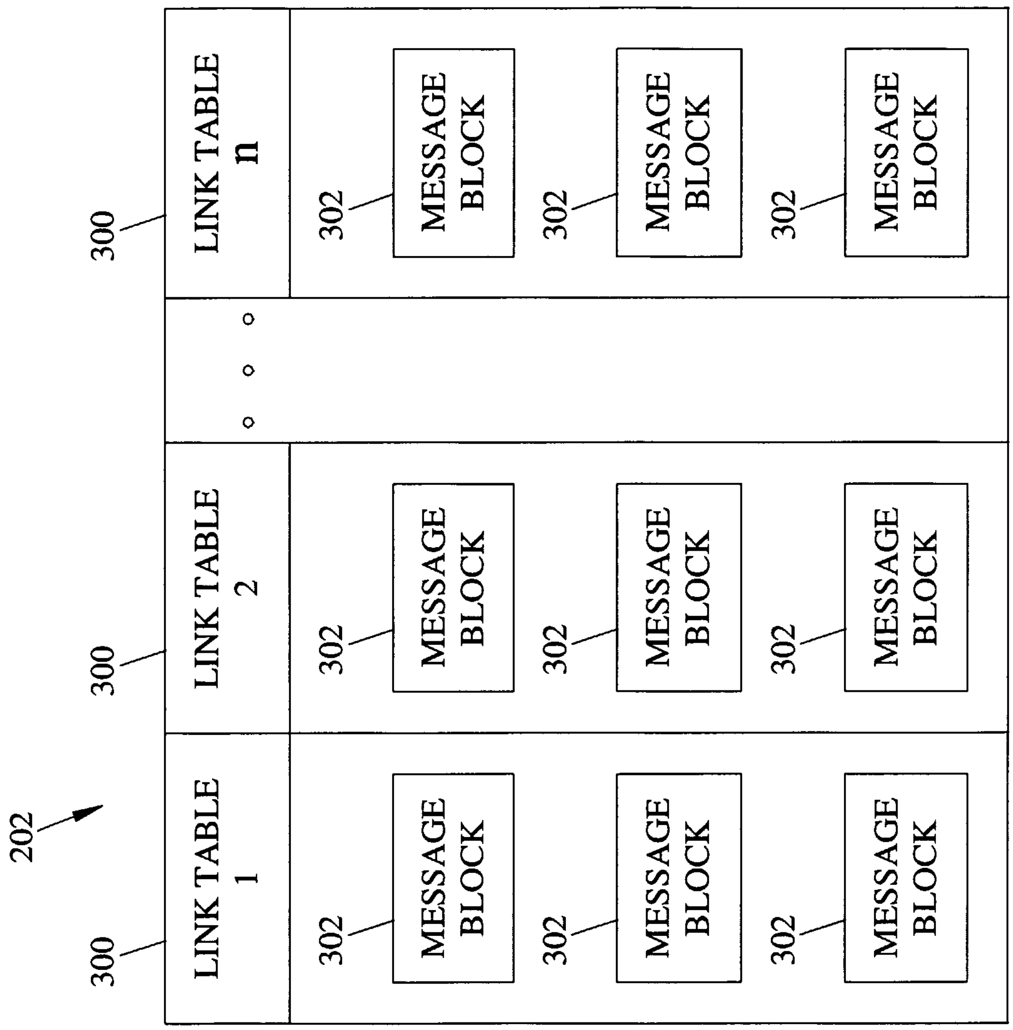
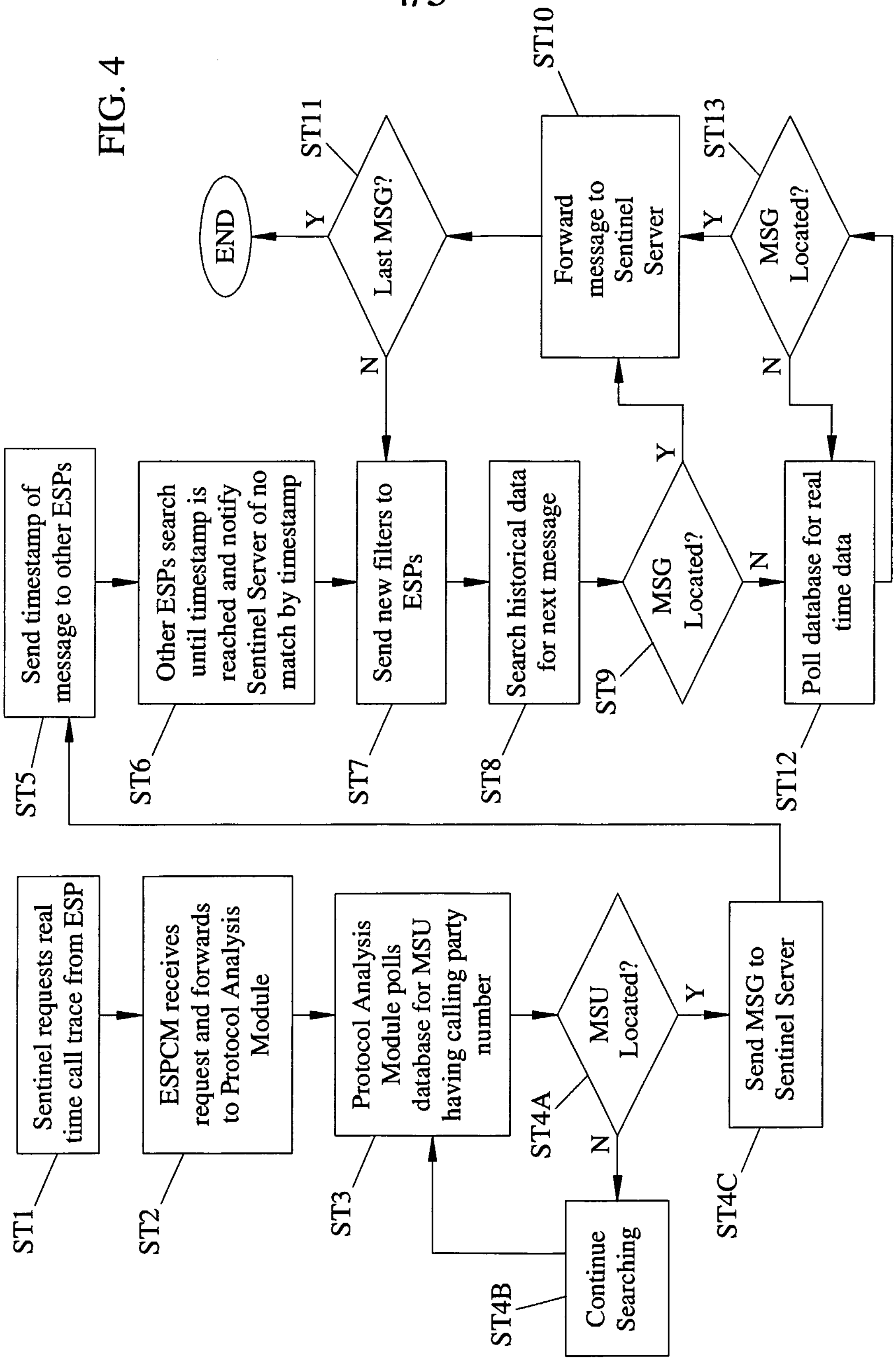
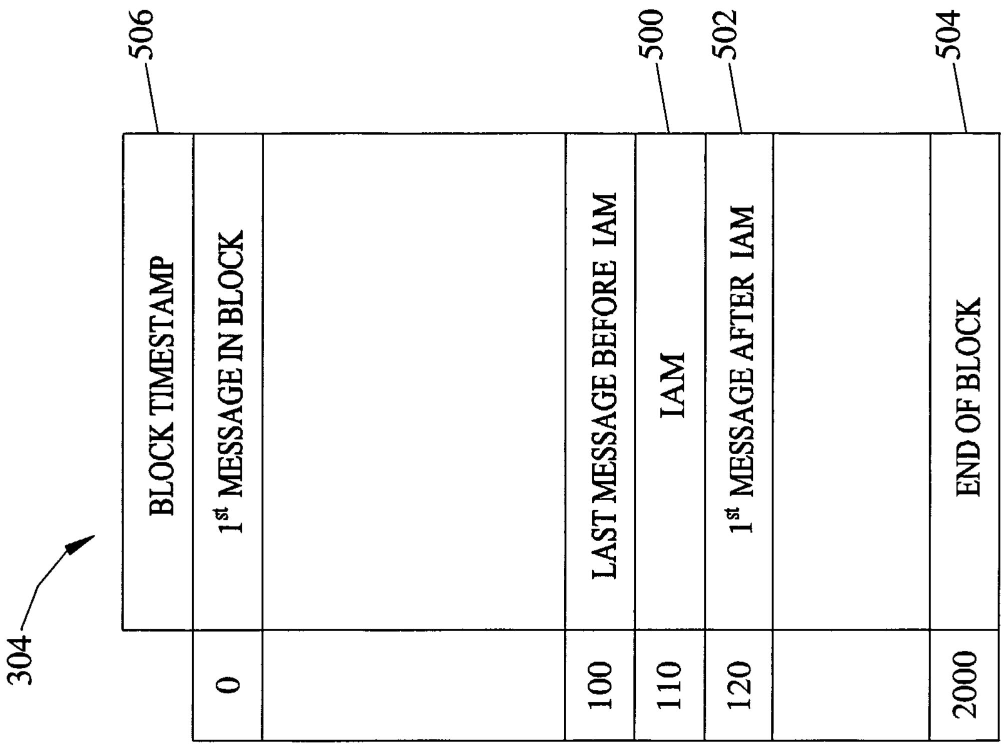

202
300
LINK TABLE 1
300
LINK TABLE 2
300
LINK TABLE n
302
MESSAGE BLOCK

FIG. 3

FIG. 4


Sentinel requests real time call trace from ESP
ST1
ESPCM receives request and forwards to Protocol Analysis Module
ST2
Protocol Analysis Module polls database for MSU having calling party number
ST3
MSU Located?
ST4A
N
Continue Searching
ST4B
Y
Send MSG to Sentinel Server
ST4C
Send timestamp of message to other ESPs
ST5
Other ESPs search until timestamp is reached and notify Sentinel Server of no match by timestamp
ST6
Send new filters to ESPs
ST7
Search historical data for next message
ST8
MSG Located?
ST9
N
Y
Poll database for real time data
ST12
MSG Located?
ST13
N
Y
Forward message to Sentinel Server
ST10
Last MSG?
ST11
N
Y
END



304
506
500
502
504
BLOCK TIMESTAMP
0
1st MESSAGE IN BLOCK
100
LAST MESSAGE BEFORE IAM
110
IAM
120
1st MESSAGE AFTER IAM
2000
END OF BLOCK

FIG. 5

INTERNATIONAL SEARCH REPORT	International application No. PCT/US02/40952

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : H04M 3/22
US CL : 379/32.05
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 379/32.05, 32.03, 88.19, 112.01, 112.06, 114.01, 126, 133, 213.01

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 4,464,543 (KLINE et al) 07 August 1984 (07.08.1984), Abstract, column 2, lines 7-17, column 4, lines 19-64.	1-30
Y	US 4,959,849 (BHUSRI) 25 September 1990 (25.09.1990), Abstract, column 2, lines 15-65, column 4, lines 7-65, column 5, lines 44-66.	1-30
Y	US 5,438,570 A (KARRAS et al) 01 August 1995 (01.08.1995), Abstract, column 4, lines 25-68, column 10, lines 5-46.	1-30
Y	US 5,592,530 A (BROCKMAN et al) 07 January 1997 (07.01.1997), Abstract, column 3, lines 20-62, column 9, lines 15-50.	1-30
Y	US 5,638,431 A (EVERETT et al) 10 June 1997 (10.06.1997), column 6, lines 45-58, column 7, lines 42-67, column 8 lines 40-67.	1-30
Y	US 5,757,895 A (ARIDAS et al) 26 May 1998 (26.05.1998), column 2, lines 26-63, columns 4-9.	1-30
Y	US 5,809,121 A (ELLIOTT et al) 15 September 1998 (15.09.1998), column 3, lines 10-64, Figure 2.	1-30
Y	US 6,137,876 A (WONG et al) 24 October 2000 (24.10.2000), Abstract, column 6, lines 39-65, column 7, lines 29-65, column 8, lines 16-65, columns 8-9.	1-30

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier application or patent published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search	Date of mailing of the international search report
09 May 2003 (09.05.2003)	02 JUN 2003
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450 Alexandria, Virginia 22313-1450 Facsimile No. (703)305-3230	Authorized officer Curtis Kuntz Telephone No. (703) 305-4700