

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
21 December 2007 (21.12.2007)

PCT

(10) International Publication Number
WO 2007/147040 A2

(51) International Patent Classification:
H04L 12/56 (2006.01)

Florida 32903 (US). **KNAZIK, Robert J.**; 112 St. Croix Avenue, Cocoa Beach, Florida 32931 (US).

(21) International Application Number:
PCT/US2007/071191

(74) Agents: **YATSKO, Michael S.** et al.; Harris Corporation, 1025 W. NASA Blvd., MS A-11I, Melbourne, Florida 32919 (US).

(22) International Filing Date: 14 June 2007 (14.06.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/424,635 16 June 2006 (16.06.2006) US

(71) Applicant (for all designated States except US): **HARRIS CORPORATION** [US/US]; 1025 W. NASA Blvd., MS A-11I, Melbourne, Florida 32919 (US).

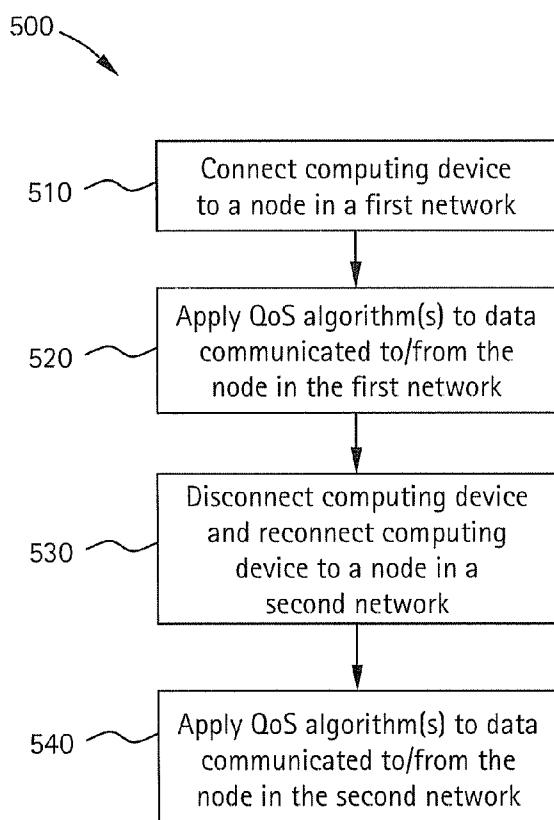
(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(72) Inventors: **SMITH, Donald L.**; 584 Hawksbill Island Drive, Satellite Beach, Florida 32937 (US). **GALLUSCIO, Anthony P.**; 549 Hummingbird Drive, Indialantic,

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,

[Continued on next page]

(54) Title: METHOD AND SYSTEM FOR NETWORK-INDEPENDENT QOS



(57) Abstract: The presently described technology provides a computing device (430) capable of being connected to a node (110) in a first network (410) and capable of applying one or more Quality of Service ("QoS") algorithms to one or more subsets of data communicated in the first network (410). The device (430) is also capable of applying the QoS algorithms to the subsets of data at or above a transport layer (240) of a protocol of the first network (410). The presently described technology also provides a method for applying one or more QoS algorithms. The method includes connecting a computing device (430) to a first network (410) and applying the QoS algorithms to one or more subsets of data independent of a protocol of the first network (410).



ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

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.

Published:

- *without international search report and to be republished upon receipt of that report*

METHOD AND SYSTEM FOR NETWORK-INDEPENDENT QOS

The presently described invention generally relates to communications networks. More particularly, the presently described inventions relates to systems and
5 methods for protocol filtering for Quality of Service.

Communications networks are utilized in a variety of environments. Communications networks typically include two or more nodes connected by one or more links. Generally, a communications network is used to support communication between two or more participant nodes over the links and intermediate nodes in the
10 communications network. There may be many kinds of nodes in the network. For example, a network may include nodes such as clients, servers, workstations, switches, and/or routers. Links may be, for example, modem connections over phone lines, wires, Ethernet links, Asynchronous Transfer Mode ("ATM") circuits, satellite links, and/or fiber optic cables.

15 A communications network may actually be composed of one or more smaller communications networks. For example, the Internet is often described as network of interconnected computer networks. Each network may utilize a different architecture and/or topology. For example, one network may be a switched Ethernet network with a star topology and another network may be a Fiber-Distributed Data
20 Interface ("FDDI") ring.

Communications networks may carry a wide variety of data. For example, a network may carry bulk file transfers alongside data for interactive real-time conversations. The data sent on a network is often sent in packets, cells, or frames. Alternatively, data may be sent as a stream. In some instances, a stream or
25 flow of data may actually be a sequence of packets. Networks such as the Internet provide general purpose data paths between a range of nodes and carrying a vast array of data with different requirements.

Communication over a network typically involves multiple levels of communication protocols. A protocol stack, also referred to as a networking stack or
30 protocol suite, refers to a collection of protocols used for communication. Each

protocol may be focused on a particular type of capability or form of communication. For example, one protocol may be concerned with the electrical signals needed to communicate with devices connected by a copper wire. Other protocols may address ordering and reliable transmission between two nodes separated by many intermediate
5 nodes, for example.

Protocols in a protocol stack typically exist in a hierarchy. Often, protocols are classified into layers. One reference model for protocol layers is the Open Systems Interconnection (“OSI”) model. The OSI reference model includes seven layers: a physical layer, data link layer, network layer, transport layer, session
10 layer, presentation layer, and application layer. The physical layer is the “lowest” layer, while the application layer is the “highest” layer. Two well-known transport layer protocols are the Transmission Control Protocol (“TCP”) and User Datagram Protocol (“UDP”). A well known network layer protocol is the Internet Protocol (“IP”).

15 At the transmitting node, data to be transmitted is passed down the layers of the protocol stack, from highest to lowest. Conversely, at the receiving node, the data is passed up the layers, from lowest to highest. At each layer, the data may be manipulated by the protocol handling communication at that layer. For example, a transport layer protocol may add a header to the data that allows for
20 ordering of packets upon arrival at a destination node. Depending on the application, some layers may not be used, or even present, and data may just be passed through.

One kind of communications network is a tactical data network. A tactical data network may also be referred to as a tactical communications network. A tactical data network may be utilized by units within an organization such as a
25 military (for example, army, navy, and/or air force). Nodes within a tactical data network may include, for example, individual soldiers, aircraft, command units, satellites, and/or radios. A tactical data network may be used for communicating data such as voice, position telemetry, sensor data, and/or real-time video.

An example of how a tactical data network may be employed is as
30 follows. A logistics convoy may be in-route to provide supplies for a combat unit in

the field. Both the convoy and the combat unit may be providing position telemetry to a command post over satellite radio links. An unmanned aerial vehicle ("UAV") may be patrolling along the road the convoy is taking and transmitting real-time video data to the command post over a satellite radio link also. At the command post, an analyst may be examining the video data while a controller is tasking the UAV to provide video for a specific section of road. The analyst may then spot an improvised explosive device ("IED") that the convoy is approaching and send out an order over a direct radio link to the convoy for it to halt and alerting the convoy to the presence of the IED.

10 The various networks that may exist within a tactical data network may have many different architectures and characteristics. For example, a network in a command unit may include a gigabit Ethernet local area network ("LAN") along with radio links to satellites and field units that operate with much lower throughput and higher latency. Field units may communicate both via satellite and via direct path
15 radio frequency ("RF"). Data may be sent point-to-point, multicast, or broadcast, depending on the nature of the data and/or the specific physical characteristics of the network. A network may include radios, for example, set up to relay data. In addition, a network may include a high frequency ("HF") network which allows long rang communication. A microwave network may also be used, for example. Due to
20 the diversity of the types of links and nodes, among other reasons, tactical networks often have overly complex network addressing schemes and routing tables. In addition, some networks, such as radio-based networks, may operate using bursts. That is, rather than continuously transmitting data, they send periodic bursts of data. This is useful because the radios are broadcasting on a particular channel that must be
25 shared by all participants, and only one radio may transmit at a time.

Tactical data networks are generally bandwidth-constrained. That is, there is typically more data to be communicated than bandwidth available at any given point in time. These constraints may be due to either the demand for bandwidth exceeding the supply, and/or the available communications technology not supplying
30 enough bandwidth to meet the user's needs, for example. For example, between some

nodes, bandwidth may be on the order of kilobits/sec. In bandwidth-constrained tactical data networks, less important data can clog the network, preventing more important data from getting through in a timely fashion, or even arriving at a receiving node at all. In addition, portions of the networks may include internal
5 buffering to compensate for unreliable links. This may cause additional delays. Further, when the buffers get full, data may be dropped.

In many instances the bandwidth available to a network cannot be increased. For example, the bandwidth available over a satellite communications link may be fixed and cannot effectively be increased without deploying another satellite.
10 In these situations, bandwidth must be managed rather than simply expanded to handle demand. In large systems, network bandwidth is a critical resource. It is desirable for applications to utilize bandwidth as efficiently as possible. In addition, it is desirable that applications avoid "clogging the pipe," that is, overwhelming links with data, when bandwidth is limited. When bandwidth allocation changes,
15 applications should preferably react. Bandwidth can change dynamically due to, for example, quality of service, jamming, signal obstruction, priority reallocation, and line-of-sight. Networks can be highly volatile and available bandwidth can change dramatically and without notice.

In addition to bandwidth constraints, tactical data networks may
20 experience high latency. For example, a network involving communication over a satellite link may incur latency on the order of half a second or more. For some communications this may not be a problem, but for others, such as real-time, interactive communication (for example, voice communications) for example, it is highly desirable to minimize latency as much as possible.

25 Another characteristic common to many tactical data networks is data loss. Data may be lost due to a variety of reasons. For example, a node with data to send may be damaged or destroyed. As another example, a destination node may temporarily drop off of the network. This may occur because, for example, the node has moved out of range, the communication's link is obstructed, and/or the node is
30 being jammed. Data may be lost because the destination node is not able to receive it

and intermediate nodes lack sufficient capacity to buffer the data until the destination node becomes available. Additionally, intermediate nodes may not buffer the data at all, instead leaving it to the sending node to determine if the data ever actually arrived at the destination.

5 Often, applications in a tactical data network are unaware of and/or do not account for the particular characteristics of the network. For example, an application may simply assume it has as much bandwidth available to it as it needs. As another example, an application may assume that data will not be lost in the network. Applications which do not take into consideration the specific
10 characteristics of the underlying communications network may behave in ways that actually exacerbate problems. For example, an application may continuously send a stream of data that could just as effectively be sent less frequently in larger bundles. The continuous stream may incur much greater overhead in, for example, a broadcast radio network that effectively starves other nodes from communicating, whereas less
15 frequent bursts would allow the shared bandwidth to be used more effectively.

Certain protocols do not work well over tactical data networks. For example, a protocol such as TCP may not function well over a radio-based tactical network because of the high loss rates and latency such a network may encounter. TCP requires several forms of handshaking and acknowledgments to occur in order to
20 send data. High latency and loss may result in TCP hitting time outs and not being able to send much, if any, meaningful data over such a network.

Information communicated with a tactical data network often has various levels of priority with respect to other data in the network. For example, threat warning receivers in an aircraft may have higher priority than position
25 telemetry information for troops on the ground miles away. As another example, orders from headquarters regarding engagement may have higher priority than logistical communications behind friendly lines. The priority level may depend on the particular situation of the sender and/or receiver. For example, position telemetry data may be of much higher priority when a unit is actively engaged in combat as
30 compared to when the unit is merely following a standard patrol route. Similarly,

real-time video data from a UAV may have higher priority when it is over the target area as opposed to when it is merely in-route.

There are several approaches to delivering data over a network. One approach, used by many communications networks, is a “best effort” approach. That is, data being communicated will be handled as well as the network can, given other demands, with regard to capacity, latency, reliability, ordering, and errors. Thus, the network provides no guarantees that any given piece of data will reach its destination in a timely manner, or at all. Additionally, no guarantees are made that data will arrive in the order sent or even without transmission errors changing one or more bits in the data.

Another approach is Quality of Service (“QoS”). QoS refers to one or more capabilities of a network to provide various forms of guarantees with regard to data that is carried. For example, a network supporting QoS may guarantee a certain amount of bandwidth to a data stream. As another example, a network may guarantee that packets between two particular nodes have some maximum latency. Such a guarantee may be useful in the case of a voice communication where the two nodes are two people having a conversation over the network. Delays in data delivery in such a case may result in irritating gaps in communication and/or dead silence, for example.

QoS may be viewed as the capability of a network to provide better service to selected network traffic. The primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency (required by some real-time and interactive traffic), and improved loss characteristics. Another important goal is making sure that providing priority for one flow does not make other flows fail. That is, guarantees made for subsequent flows must not break the guarantees made to existing flows.

Current approaches to QoS often require every node in a network to support QoS, or, at the very least, for every node in the network involved in a particular communication to support QoS. For example, in current systems, in order to provide a latency guarantee between two nodes, every node carrying the traffic

between those two nodes must be aware of and agree to honor, and be capable of honoring, the guarantee.

There are several approaches to providing QoS or QoS parameter/mechanisms/algorithms. One approach is Integrated Services, or

5 “IntServ.” IntServ provides a QoS system wherein every node in the network supports the services and those services are reserved when a connection is set up. IntServ does not scale well because of the large amount of state information that must be maintained at every node and the overhead associated with setting up such connections.

10 Another approach to providing QoS is Differentiated Services, or “DiffServ.” DiffServ is a class of service model that enhances the best-effort services of a network such as the Internet. DiffServ differentiates traffic by user, service requirements, and other criteria. Then, DiffServ marks packets so that network nodes can provide different levels of service via priority queuing or bandwidth allocation, or
15 by choosing dedicated routes for specific traffic flows. Typically, a node has a variety of queues for each class of service. The node then selects the next packet to send from those queues based on the class categories.

Existing QoS solutions are often network specific and each network type or architecture may require a different QoS configuration. Due to the
20 mechanisms existing QoS solutions utilize, messages that look the same to current QoS systems may actually have different priorities based on message content. However, data consumers may require access to high-priority data without being flooded by lower-priority data. Existing QoS systems cannot provide QoS based on message content at the transport layer.

25 As mentioned, existing QoS solutions require at least the nodes involved in a particular communication to support QoS. However, the nodes at the “edge” of network may be adapted to provide some improvement in QoS, even if they are incapable of making total guarantees. Nodes are considered to be at the edge of the network if they are the participating nodes in a communication (that is, the
30 transmitting and/or receiving nodes) and/or if they are located at chokepoints in the

network. A chokepoint is a section of the network where all traffic must pass to another portion. For example, a router or gateway from a LAN to a satellite link would be a choke point, since all traffic from the LAN to any nodes not on the LAN must pass through the gateway to the satellite link.

5 Thus, there is a need for systems and methods providing QoS in a tactical data network. There is a need for systems and methods for providing QoS on the edge of a tactical data network. Additionally, there is a need for adaptive, configurable QoS systems and methods in a tactical data network.

 In addition, there is a need for a technology that provides QoS to a
10 variety of networks, independent of the protocols employed by the networks and/or the hardware used in the network. In other words, a need exists for a QoS solution that is network protocol and/or hardware independent. Such a need can be met by providing QoS solutions to data (or subsets of data) at or above the transport layer in the OSI seven-layer model. By providing the QoS solutions at or above this layer, the
15 underlying hardware and/or protocols of the networks becomes less relevant. That is, the QoS solution can easily be connected and disconnected to a variety of networks employing different protocols and/or hardware in order to provide QoS to data communicated in and among the networks.

 The presently described technology provides a computing device
20 capable of being connected to a node in a first network and capable of applying one or more Quality of Service ("QoS") algorithms to one or more subsets of data communicated in the first network. The device is also capable of applying the QoS algorithms to the subsets of data at or above a transport layer of a protocol of the first network.

25 The presently described technology also provides a method for applying one or more QoS algorithms. The method includes connecting a computing device to a first network and applying the QoS algorithms to one or more subsets of data independent of a protocol of the first network.

 The presently described technology also provides a computer-readable
30 storage medium including a set of instructions for a computer. The instructions

include a QoS routine that is configured to apply one or more QoS algorithms to one or more subsets of data communicated in a network at or above a transport layer of a protocol of the network.

FIG. 1 illustrates a tactical communications network environment
5 operating with an embodiment of the presently described technology.

FIG. 2 illustrates a schematic diagram of the OSI seven-layer model and the operation of the set of instructions for computing device in accordance with an embodiment of the presently described technology.

FIG. 3 illustrates an example of multiple networks facilitated using the
10 data communications system in accordance with an embodiment of the presently described technology.

FIG. 4 illustrates a system for applying QoS algorithms to data independent of network hardware and/or protocols in accordance with an embodiment of the presently described technology.

15 FIG. 5 illustrates a flowchart of a method for applying QoS algorithms independent of network hardware and/or protocols in accordance with an embodiment of the presently described technology

The foregoing summary, as well as the following detailed description of certain embodiments of the presently described technology, will be better
20 understood when read in conjunction with the appended drawings. For the purpose of illustrating the presently described technology, certain embodiments are shown in the drawings. It should be understood, however, that the presently described technology is not limited to the arrangements and instrumentality shown in the attached drawings.

FIG. 1 illustrates a tactical communications network environment 100
25 operating with an embodiment of the presently described technology. The network environment 100 includes a plurality of communication nodes 110, one or more networks 120, one or more links 130 connecting the nodes and network(s), and one or more communication systems 150 facilitating communication over the components of the network environment 100. The following discussion assumes a network

environment 100 including more than one network 120 and more than one link 130, but it should be understood that other environments are possible and anticipated.

Communication nodes 110 may be and/or include radios, transmitters, satellites, receivers, workstations, servers, and/or other computing or processing devices, for example. Nodes 110 can be sources and/or recipients of information or data.

Network(s) 120 may be hardware and/or software for transmitting data between nodes 110, for example. Network(s) 120 may include one or more nodes 110, for example. Link(s) 130 may be wired and/or wireless connections to allow transmissions between nodes 110 and/or network(s) 120.

The communications system 150 may include software, firmware, and/or hardware used to facilitate data transmission among the nodes 110, networks 120, and links 130, for example. As illustrated in FIG. 1, communications system 150 may be implemented with respect to the nodes 110, network(s) 120, and/or links 130. In certain embodiments, every node 110 includes a communications system 150. In certain embodiments, one or more nodes 110 include a communications system 150. That is, in certain embodiments, one or more nodes 110 may not include a communications system 150.

The communication system 150 provides dynamic management of data to help assure communications on a tactical communications network, such as the network environment 100. As shown in FIG. 2, in certain embodiments, the system 150 (or a set of instructions operating on a computing device in system 150) operates as part of and/or at the top of the transport layer 240 in the OSI seven layer protocol model (described in more detail below). By operating at or above the transport layer 240, system 150 (or the set of instructions operating on a computing device in system 150) can be used in a variety of networks employing a variety of different hardware and/or protocols. That is, by operating at or above the transport layer 240, the presently described technology operates independent of the network hardware and/or network protocol. Therefore, the presently described technology can easily be connected to a node in a first network to provide QoS solutions to data or subsets of

data communicated by one or more nodes in the first network, disconnected from the node, connected to a node in a second network (having different hardware and/or protocols than the first network) to provide QoS solutions to data/subsets of data communicated by node(s) in the second network. The presently described technology
5 becomes, in a sense, network “agnostic.”

The system 150 may give precedence to higher priority data in the tactical network passed to the transport layer, for example. The system 150 may be used to facilitate communications in a single network, such as a LAN or wide area network (“WAN”), or across multiple networks. An example of a multiple network
10 system is shown in FIG. 3. The system 150 may be used to manage available bandwidth rather than add additional bandwidth to the network, for example.

In certain embodiments, the system 150 is a software system, although the system 150 may include both hardware and software components in various embodiments. As described above, the system 150 can be network hardware
15 independent, for example. That is, by operating at or above the transport layer 240, the system 150 may be adapted to function on a variety of hardware and software platforms. In certain embodiments, the system 150 operates on the edge of the network rather than on nodes in the interior of the network. However, the system 150 may operate in the interior of the network as well, such as at “choke points” in the
20 network.

The system 150 can use rules and modes or profiles to perform throughput management functions such as optimizing available bandwidth, setting information priority, and managing data links in the network (for example, QoS parameters/mechanisms/algorithms). By “optimizing” bandwidth, it is meant that the
25 presently described technology can be employed to increase an efficiency of bandwidth use to communicate data in one or more networks. Optimizing bandwidth usage can include removing functionally redundant messages, message stream management or sequencing, and message compression, for example. Setting information priority can include differentiating message types at a finer granularity
30 than IP based techniques and sequencing messages onto a data stream via a selected

rule-based sequencing algorithm, for example. Data link management can include rule-based analysis of network measurements to affect changes in rules, modes, and/or data transports, for example. A mode or profile can include a set of rules related to the operational needs for a particular network state of health or condition.

- 5 The system 150 provides dynamic, “on-the-fly” reconfiguration of modes, including defining and switching to new modes on the fly.

The communication system 150 can be configured to accommodate changing priorities and grades of service, for example, in a volatile, bandwidth-limited network. The system 150 can be configured to manage information for
10 improved data flow to help increase response capabilities in the network and reduce communications latency. Additionally, the system 150 can provide interoperability via a flexible architecture that is upgradeable and scalable to improve availability, survivability, and reliability of communications. The system 150 supports a data communications architecture that may be autonomously adaptable to dynamically
15 changing environments while using predefined and predictable system resources and bandwidth, for example.

In certain embodiments, the system 150 provides throughput management to bandwidth-constrained tactical communications networks while remaining transparent to applications using the network. The system 150 provides
20 throughput management across multiple users and environments at reduced complexity to the network. As mentioned above, in certain embodiments, the system 150 runs on a host node in and/or at the top of layer four (the transport layer) of the OSI seven layer model and does not require specialized network hardware. The system 150 may operate transparently to the layer four interface. That is, an
25 application may utilize a standard interface for the transport layer and be unaware of the operation of the system 150. For example, when an application opens a socket, the system 150 may filter data at this point in the protocol stack. The system 150 achieves transparency by allowing applications to use, for example, the TCP/IP socket interface that is provided by an operating system at a communication device on the
30 network rather than an interface specific to the system 150. System 150 rules may be

written in extensible markup language (“XML”) and/or provided via custom dynamic link libraries (“DLL”), for example.

In certain embodiments, the system 150 provides QoS on the edge of the network. The system’s QoS capability offers content-based, rule-based data prioritization on the edge of the network, for example. Prioritization can include differentiation and/or sequencing, for example. The system 150 can differentiate messages into queues based on user-configurable differentiation rules, for example. The messages are sequenced into a data stream in an order dictated by the user-configured sequencing rule (for example, starvation, round robin, relative frequency, etc.). Using QoS on the edge, data messages that are indistinguishable by traditional QoS approaches can be differentiated based on message content, for example. Rules can be implemented in XML, for example. In certain embodiments, to accommodate capabilities beyond XML and/or to support extremely low latency requirements, the system 150 allows dynamic link libraries to be provided with custom code, for example.

Inbound and/or outbound data on the network may be customized via the system 150. Prioritization protects client applications from high-volume, low-priority data, for example. The system 150 helps to ensure that applications receive data to support a particular operational scenario or constraint.

In certain embodiments, when a host is connected to a LAN that includes a router as an interface to a bandwidth-constrained tactical network, the system can operate in a configuration known as QoS by proxy. In this configuration, packets that are bound for the local LAN bypass the system and immediately go to the LAN. The system applies QoS on the edge of the network to packets bound for the bandwidth-constrained tactical link.

In certain embodiments, the system 150 offers dynamic support for multiple operational scenarios and/or network environments via commanded profile switching. A profile can include a name or other identifier that allows the user or system to change to the named profile. A profile may also include one or more identifiers, such as a functional redundancy rule identifier, a differentiation rule

identifier, an archival interface identifier, a sequencing rule identifier, a pre-transmit interface identifier, a post-transmit interface identifier, a transport identifier, and/or other identifier, for example. A functional redundancy rule identifier specifies a rule that detects functional redundancy, such as from stale data or substantially similar data, for example. A differentiation rule identifier specifies a rule that differentiates messages into queues for processing, for example. An archival interface identifier specifies an interface to an archival system, for example. A sequencing rule identifier identifies a sequencing algorithm that controls samples of queue fronts and, therefore, the sequencing of the data on the data stream. A pre-transmit interface identifier specifies the interface for pre-transmit processing, which provides for special processing such as encryption and compression, for example. A post-transmit interface identifier identifies an interface for post-transmit processing, which provides for processing such as de-encryption and decompression, for example. A transport identifier specifies a network interface for the selected transport.

A profile can also include other information, such as queue sizing information, for example. Queue sizing information identifies a number of queues and amount of memory and secondary storage dedicated to each queue, for example.

In certain embodiments, the system 150 provides a rules-based approach for optimizing bandwidth. For example, the system 150 can employ queue selection rules to differentiate messages into message queues so that messages can be assigned a priority and an appropriate relative frequency on the data stream. The system 150 can use functional redundancy rules to manage functionally redundant messages. A message is functionally redundant if it is not different enough (as defined by the rule) from a previous message that has not yet been sent on the network, for example. That is, if a new message is provided that is not sufficiently different from an older message that has already been scheduled to be sent, but has not yet been sent, the newer message can be dropped, since the older message will carry functionally equivalent information and is further ahead in the queue. In addition, functional redundancy may include actual duplicate messages and newer messages that arrive before an older message has been sent. For example, a node can

receive identical copies of a particular message due to characteristics of the underlying network, such as a message that was sent by two different paths for fault tolerance reasons. As another example, a new message can contain data that supersedes an older message that has not yet been sent. In this situation, the system
5 150 can drop the older message and send only the new message. The system 150 can also include priority sequencing rules to determine a priority-based message sequence of the data stream. Additionally, the system 150 can include transmission processing rules to provide pre-transmission and post-transmission special processing, such as compression and/or encryption.

10 In certain embodiments, the system 150 provides fault tolerance capability to help protect data integrity and reliability. For example, the system 150 can use user-defined queue selection rules to differentiate messages into queues. The queues are sized according to a user-defined configuration, for example. The configuration specifies a maximum amount of memory a queue can consume, for
15 example. Additionally, the configuration can allow the user to specify a location and amount of secondary storage that may be used for queue overflow. After the memory in the queues is filled, messages can be queued in secondary storage. When the secondary storage is also full, the system 150 can remove the oldest message in the queue, logs an error message, and queues the newest message. If archiving is enabled
20 for the operational mode, then the de-queued message can be archived with an indicator that the message was not sent on the network.

Memory and secondary storage for queues in the system 150 can be configured on a per-link basis for a specific application, for example. A longer time between periods of network availability may correspond to more memory and
25 secondary storage to support network outages. The system 150 can be integrated with network modeling and simulation applications, for example, to help identify sizing to help ensure that queues are sized appropriately and time between outages is sufficient to help achieve steady-state and help avoid eventual queue overflow.

Furthermore, in certain embodiments, the system 150 offers the
30 capability to meter inbound (“shaping”) and outbound (“policing”) data. Policing and

shaping capabilities help address mismatches in timing in the network. Shaping helps to prevent network buffers from flooding with high-priority data queued up behind lower-priority data. Policing helps to prevent application data consumers from being overrun by low-priority data. Policing and shaping are governed by two parameters:

5 effective link speed and link proportion. The system 150 may form a data stream that is no more than the effective link speed multiplied by the link proportion, for example. The parameters may be modified dynamically as the network changes. The system may also provide access to detected link speed to support application level decisions on data metering. Information provided by the system 150 may be

10 combined with other network operations information to help decide what link speed is appropriate for a given network scenario.

FIG. 4 illustrates a system 400 for applying QoS algorithms to data independent of network hardware and/or protocols in accordance with an embodiment of the presently described technology. System 400 includes a first network 410, a

15 second network 420, a computing device 430, and a plurality of nodes 110 in one or more of first and second networks 410, 420.

First and second networks 410, 420 can include any network employed to communicate information via data or subsets of data. For example, first and second networks 410, 420 can be similar to networks 120 described above. In an

20 embodiment, first and/or second networks 410, 420 each comprise a low or high speed network. A low speed network can include any network with a limited bandwidth capability or availability. For example, a low speed network can comprise a LAN such as a military tactical network. In an embodiment, a low speed network is a tactical network such as a Tactical Satellite ("TACSAT") network or a tactical HF

25 network. In another example, a low speed network can include a radio or IP based radio network. A high speed network can include any network with a large bandwidth capability or availability. Generally, a high speed network has a greater bandwidth or throughput than a low speed network. For example, a high speed network can comprise one or more networks with traditionally large bandwidth

connections and high data throughputs. In an embodiment, a high speed network comprises Ethernet connections and/or an EPLRS network.

In an embodiment, first and second networks 410, 420 use different hardware and/or different protocols to communicate data between nodes 110 within
5 each network 410, 420 or between a node 110 in network 410 with a node 110 in network 420. While only two networks are illustrated in FIG. 4, this is not intended as a limitation on the presently described technology. The presently described technology is applicable to a larger or smaller number of networks. For example, the presently described technology can be applied to a single network, or three or more
10 networks. In addition, the number of nodes 110 illustrated in each network 410, 420 is not intended as a limitation on the presently described technology. The presently described technology is applicable to a larger or smaller number of nodes 110 in a network.

Computing device 430 can be included in a communication system 150
15 of FIG. 1 described above, for example. Computing device 430 comprises any system, device, apparatus or group of systems, devices or apparatuses capable of performing electronic processing or communication of data. For example, computing device 430 can comprise a personal computer, such as a desktop or laptop computer or a server computer. In an embodiment, computing device 430 can also comprise a
20 network hardware component, such as a network router or a network switch, for example.

Computing device 430 includes a computer-readable storage medium. For example, computing device 430 includes a hard drive, a CD drive, a DVD drive, or a memory key in communication with device 430. In another embodiment,
25 computing device 430 is in communication with a remote computer-readable storage medium. For example, computing device 430 can have a wired or wireless communications link with a server or other memory remote from device 430.

The computer-readable storage medium in communication with computing device 430 includes a set of instructions for a computer. In an
30 embodiment, the set of instructions is embodied in one or more software applications

capable of being run or executed on computing device 430. The set of instructions can include one or more software routines for enabling computing device 430 to apply one or more QoS algorithms to data or subsets of data communicated to or from a node 110 connected to computing device 430 at or above the transport layer.

5 The set of instructions enable computing device 430 to provide dynamic management of data throughput for data communicated within, to and/or from a network 410, 420 independent of the hardware of the network and/or a protocol used by the network to communicate data. Therefore, one technical effect of the set of instructions is to improve the throughput of data communicated to, from or
10 within a given network.

 In operation, computing device 430 is connected to a node 110 in first network 410. For example, a laptop computer can be connected via a serial port to a radio in a military tactical network. The node 110 to which computing device 430 is connected can transmit and/or receive information communicated via data or subsets
15 of data. For example, the information can be communicated via data packets, cells, or frames or as a data stream. The data or subsets of data can be communicated to and/or from other nodes 110 in the same or different network 410, 420.

 In an embodiment, a plurality of devices 430 can be connected to a plurality of nodes 110. For example, two or more nodes 110 can each be connected to
20 a computing device 430.

 As data or data subsets are communicated to or from a node 110 to which computing device 430 is connected, the set of instructions stored on the computer-readable storage medium (in communication with computing device 430) applies one or more QoS algorithms. The set of instructions can include one or more
25 routines, such as a QoS routine. The set of instructions and routine(s) can be embodied in one or more software applications, for example.

 In a preferred embodiment, the set of instructions are written in standard Extensible Markup Language ("XML"). In another embodiment, the set of instructions are provided to computing device 430 via customized dynamic link

libraries (“DLL”). The use of customized DLLs can be preferred to XML where extremely low latency requirements must be supported.

The set of instructions can filter data at the top of the protocol stack when an application opens a network socket to transmit data. The set of instructions
5 can be transparent to users as the instructions use the TCP/IP socket interface provided by the operating system of device 430.

Once computing device 430 is connected to a node 110 in first network 410, node 110 communicates one or more subsets of data to at least one other node 110 in first and/or second network 410, 420. In addition, node 110 can receive one or
10 more subsets of data from at least one other node 110 in first and/or second network 410, 420. The QoS routine in the set of instructions in communication with computing device 430 applies one or more QoS algorithms to the subsets of data.

The QoS algorithms are applied to the data or data subsets at or above the transport layer of the OSI seven-layer model to maintain the network hardware
15 independence and/or the protocol independence of the set of instructions. As mentioned above, FIG. 2 illustrates a schematic diagram of the OSI seven-layer model 200 (described above) and the operation of the set of instructions for computing device 430 in accordance with an embodiment of the presently described technology. The OSI model 200 includes seven layers, namely an application layer 210, a
20 presentation layer 220, a session layer 230, a transport layer 240, a network layer 250, a data link layer 260 and a physical layer 270. A transmitting user 280 (using a node 110 connected to computing device 430, for example) communicates data 290 to a receiving user 292 (at another node 110, for example) over a link 294.

In accordance with the presently described technology, the set of
25 instructions operating on computing device 430 implements one or more QoS algorithms at a level 296 at or above transport layer 240 of OSI model 200. In doing so, the set of instructions is able to optimize the bandwidth available a network while providing the network independence of the presently described technology. As described above, by “optimize,” it is meant that the QoS algorithms can be used to
30 increase an efficiency of bandwidth usage in a network.

In other words, traditional QoS solutions are network specific with a different configuration of QoS solutions for each network type. However, according to the presently described technology, computing device 430 is able to apply QoS algorithms to data transmitted to, from or within a network 410, 420 without being
5 “hard-wired” or confined to the hardware of a given network 410, 420.

The QoS algorithm(s) can include any rule or parameter based adjustment of the priority or order in which data is transmitted to a given destination. In other words, a QoS algorithm can include one or more rules or parameters that give precedence to higher-priority data. In doing so, the QoS algorithm(s) can optimize
10 bandwidth, establish or set priority on the information contained in the data, and manage a data link as bandwidth becomes constrained over a given data link or within a given network, as described above. Again, by “optimize,” it is meant that the QoS algorithms can be used to increase an efficiency of bandwidth usage in a network. For example, optimizing bandwidth usage can include removing functionally
15 redundant messages, message stream management or sequencing, and message compression, for example. Setting information priority can include differentiating message types at a finer granularity than IP based techniques and sequencing messages onto a data stream via a selected rule-based sequencing algorithm, for example. Data link management can include rule-based analysis of network
20 measurements to affect changes in rules, modes, and/or data transports, for example.

The QoS algorithms can also include the prioritization of data based on user-configurable rules, as described above. For example, messages can be sequenced into a data stream in an order dictated by a user-configured sequencing rule (for example, starvation, round robin, relative frequency, etc.). Data messages that are
25 indistinguishable by traditional QoS approaches can be differentiated based on message content, for example.

QoS algorithms can also be employed to manage a data link by dynamically modifying a link according to a selected mode. A mode comprises of a collection of rules and configuration information for controlling data propagation to
30 and from the transport layer on a network link. The mode can specify throughput

management rules, archival configuration, pre- and post-transmission rules, and transport selection.

By applying the QoS algorithms, the subsets of data communicated to or from node 110 (to which computing device 430 is connected) can be
5 communicated and/or received in a priority order. That is, the subsets of data can be prioritized and communicated or received by node 430 (to which computing device 430 is connected) in the prioritized order determined by the QoS algorithms. In communicating and/or receiving the subsets of data in a priority order, the QoS algorithms can increase an efficiency of bandwidth use within the network 410 that
10 includes node 110 to which computing device 430 is connected. In addition, the QoS algorithms can manage one or more data links of network 410, as described above.

In an embodiment, computing device 430 can be disconnected from the node 110 to which it was connected in first network 410 and connected to another, different node 110 in second network 430. Second network 420 can employ different
15 hardware and/or communication protocols than first network 410.

Once computing device 430 is connected to node 110 in second network 430, the QoS routine of the set of instructions in communication with computing device 430 can again apply one or more QoS algorithms to one or more subsets of data communicated to and/or from node 110 (connected to device 430) at
20 or above the transport layer, as described above. Again, as computing device 430 and QoS routine apply the QoS algorithm(s) at or above the transport layer, the different hardware of networks 410, 420 and/or different communication protocols of networks 410, 420 becomes less relevant. That is, the QoS algorithms can be applied by computing device 430 regardless of the different network 410, 420 hardware and/or
25 protocols employed. Therefore, computing device 430 is able to apply QoS algorithms independent of network hardware and/or protocols.

FIG. 5 illustrates a flowchart of method 500 for applying QoS algorithms independent of network hardware and/or protocols in accordance with an embodiment of the presently described technology. According to method 500, first at
30 step 510, a computing device 430 is connected to a node 110 in a first network 410.

Node 110 to which computing device 430 is connected can be a source and/or recipient of one or more subsets of data communicated to, from or within network 410.

5 Next, at step 520, one or more QoS algorithms are applied to one or more subsets of data communicated to and/or from node 110 to which computing device 430 is connected. As described above, the QoS algorithms are applied to the data at or above a transport layer of the OSI seven layer model. In doing so, method 500 enables the QoS algorithms to be applied to data communicated to, from or within a network independent of the network's hardware and/or protocols.

10 Also as described above, applying the QoS algorithms can result in an increased efficiency of bandwidth usage in network 410, can establish a priority of information communicated in the data, and/or can manage one or more data links of network 410.

15 Next, at step 530, computing device 430 is disconnected from a node 110 in first network 410 and connected to a different node 110 in second network 420. As described above, first and second networks 410, 420 can include different network hardware and/or communication protocols to communicate data to, from or within the respective networks.

20 Next, at step 540, one or more QoS algorithms are applied to one or more subsets of data communicated to and/or from node 110 to which computing device 430 is connected in second network 420. As described above, the QoS algorithms are applied to the data at or above a transport layer of the OSI seven layer model. Also as described above, applying the QoS algorithms can result in an increased efficiency of bandwidth usage in network 420, can establish a priority of
25 information communicated in the data, and/or can manage one or more data links of network 420.

30 In an example of the presently described technology, computing device 430 can be connected to a node 110 in a first LAN 310 illustrated in the several networks of FIG. 3. The node 110 to which computing device 430 is connected can be a soldier's satellite radio, for example. Computing device 430 can be connected to

the soldier's radio via a connection such as a USB or serial connection. Once computing device 430 is connected to the soldier's radio, computing device 430 and the QoS routine of the set of instructions in communication with device 430 can apply QoS routines to data transmitted from the soldier's radio.

5 For example, the QoS algorithms can be applied to positional data transmitted from the soldier's radio in LAN 310 to an air combat unit 340 (such as an airplane) and/or a ground combat unit 330 (such as a tank). As described above, computing device 430 and QoS routine apply the algorithms to data at or above the transport layer to provide for network independence.

10 By applying the QoS algorithms to the positional data, the order in which the data is transmitted can be established according to the priority of the positional information. For example, positional information regarding the location of enemy troops can receive a higher priority than positional information regarding the location of an enemy, unmanned ground combat unit. If the soldier's radio is
15 attempting to transmit both of these types of positional information, the positional information regarding the location of enemy troops can be given a higher priority by the QoS algorithms and transmitted before other types of less important positional information.

 In a similar example, the QoS algorithms can be applied to positional
20 data transmitted to the soldier's radio in LAN 310 from an air combat unit 340 (such as an airplane) and/or a ground combat unit 330 (such as a tank). Again, by applying the QoS algorithms to the positional data, the order in which the data is transmitted can be established according to the priority of the positional information. For example, positional information regarding the location of enemy troops can receive a
25 higher priority than positional information regarding the location of an enemy, unmanned ground combat unit. If a reconnaissance aircraft 340 is attempting to warn the soldiers in LAN 310 of the location of enemy troops, the positional information regarding the location of enemy troops can be given a higher priority by the QoS algorithms and received by the soldier's radio before other types of less important
30 positional information.

Computing device 430 can then be disconnected from the soldier's radio in LAN 310 and connected to a ground combat unit (such as a tank) or another soldier's radio in a second LAN 320. Computing device 430 and QoS routine can then apply QoS algorithms to adjust the priority in which data is transmitted and/or received by the ground combat unit or soldier's radio, as described above. Even if second LAN 320 employs hardware and/or communication protocols that differ from first LAN 310, the same computing device 430 that applied QoS algorithms to data communicated to and/or from the soldier's radio in LAN 310 can also apply QoS algorithms to data communicated to and/or from the soldier's radio in LAN 320. Again, as the QoS algorithms are applied at or above the transport layer, the application of the QoS algorithms by computing device 430 is network hardware and/or protocol independent. Therefore, a change in network hardware and/or protocol does not impact computing device's 430 ability to apply QoS algorithms to communicated data. Moreover, a change in the node 110 to which computing device 430 is connected also does not impact computing device's 430 ability to apply QoS algorithms to communicated data.

By applying QoS algorithms to a variety of networks employing different hardware and/or protocols, the presently described technology enables QoS algorithms to be applied to data independent of the network hardware and/or protocols. In other words, the presently described technology provides the application of QoS algorithms in a network-independent manner. While current QoS solutions may be limited to tied to the hardware of a network, the presently described technology can be applied to a wide variety of networks independent of the networks' respective hardware and/or protocols.

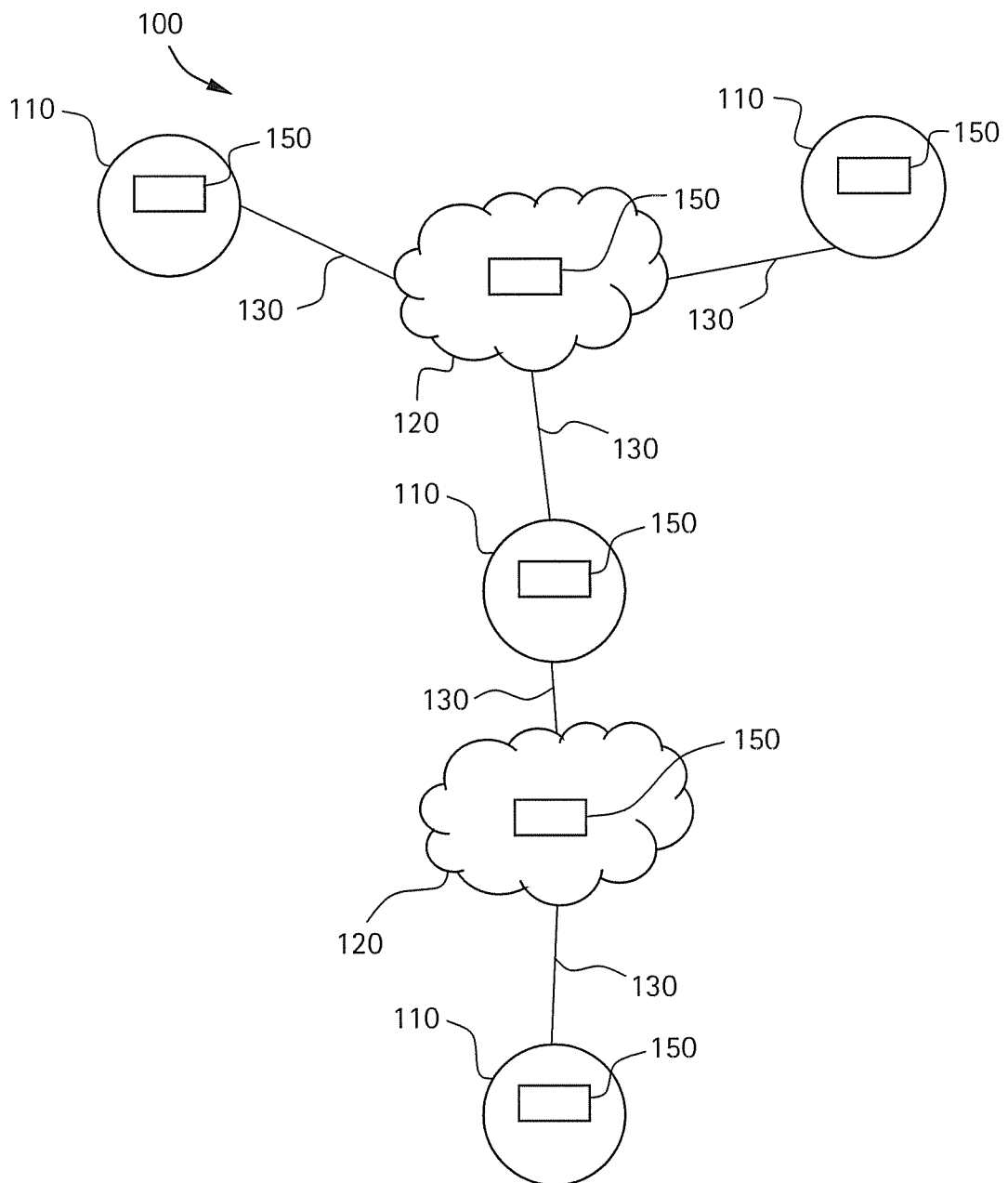
CLAIMS

1. A computing device capable of being connected to a node in a first network and capable of applying one or more Quality of Service (“QoS”) algorithms to one or more subsets of data communicated to said node or from said node, wherein said device is capable of applying said QoS algorithms to said subsets of data at or above a transport layer of a protocol of said first network.
2. The device of claim 1, wherein said device is capable of applying said QoS algorithms to said subsets of data independent of said protocol of said first network.
3. The device of claim 2, wherein said device is capable of being disconnected from said first network and connected to a node in a second network to apply one or more QoS algorithms to one or more subsets of data communicated in said second network.
4. The device of claim 3, wherein said protocol of said first network differs from a protocol of said second network.
5. The device of claim 1, wherein said node is a source of said subsets of data.
6. The device of claim 1, wherein said QoS algorithms are applied to said subsets of data to increase an efficiency of bandwidth use in said first network, establish priority of information of said subsets of data, and/or manage one or more data links in said first network.
7. The device of claim 1, wherein said device comprises one or more of a computer, a router of said second network, and a switch of said second network.

8. A method for applying one or more Quality of Service (“QoS”) algorithms, said method including:
- connecting a computing device to a first network; and
- 5 applying said QoS algorithms to one or more subsets of data independent of a protocol of said first network.
9. The method of claim 8, wherein said applying step includes applying said QoS algorithms to said subsets of data at or above a transport layer of said
- 10 protocol of said first network.
10. The method of claim 8, further including:
- connecting said computing device to a second network; and
- applying one or more QoS algorithms to one or more subsets of data
- 15 communicated in said second network,
- wherein a protocol of said second network differs from said protocol of said first network.

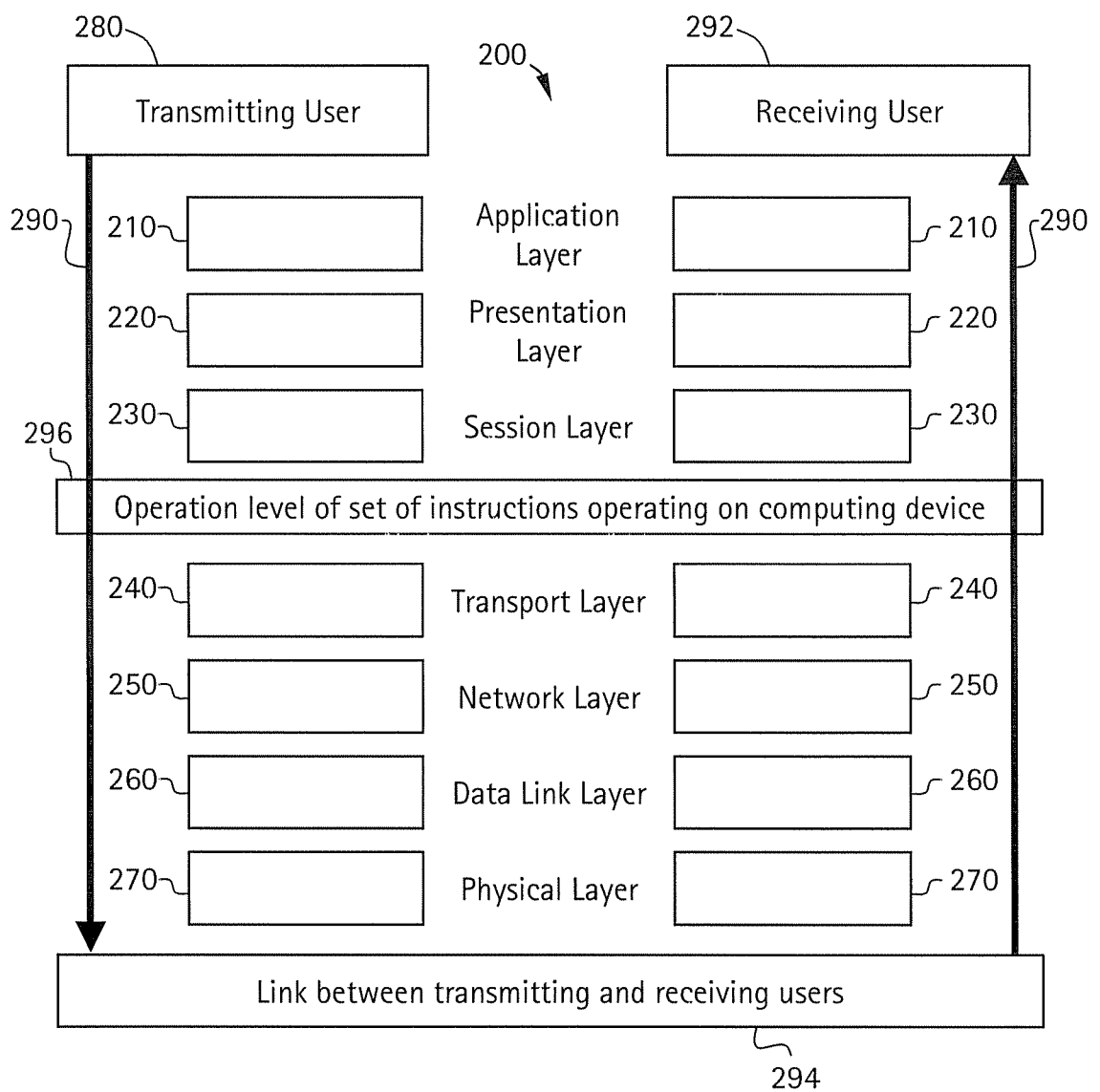
1/5

FIG. 1



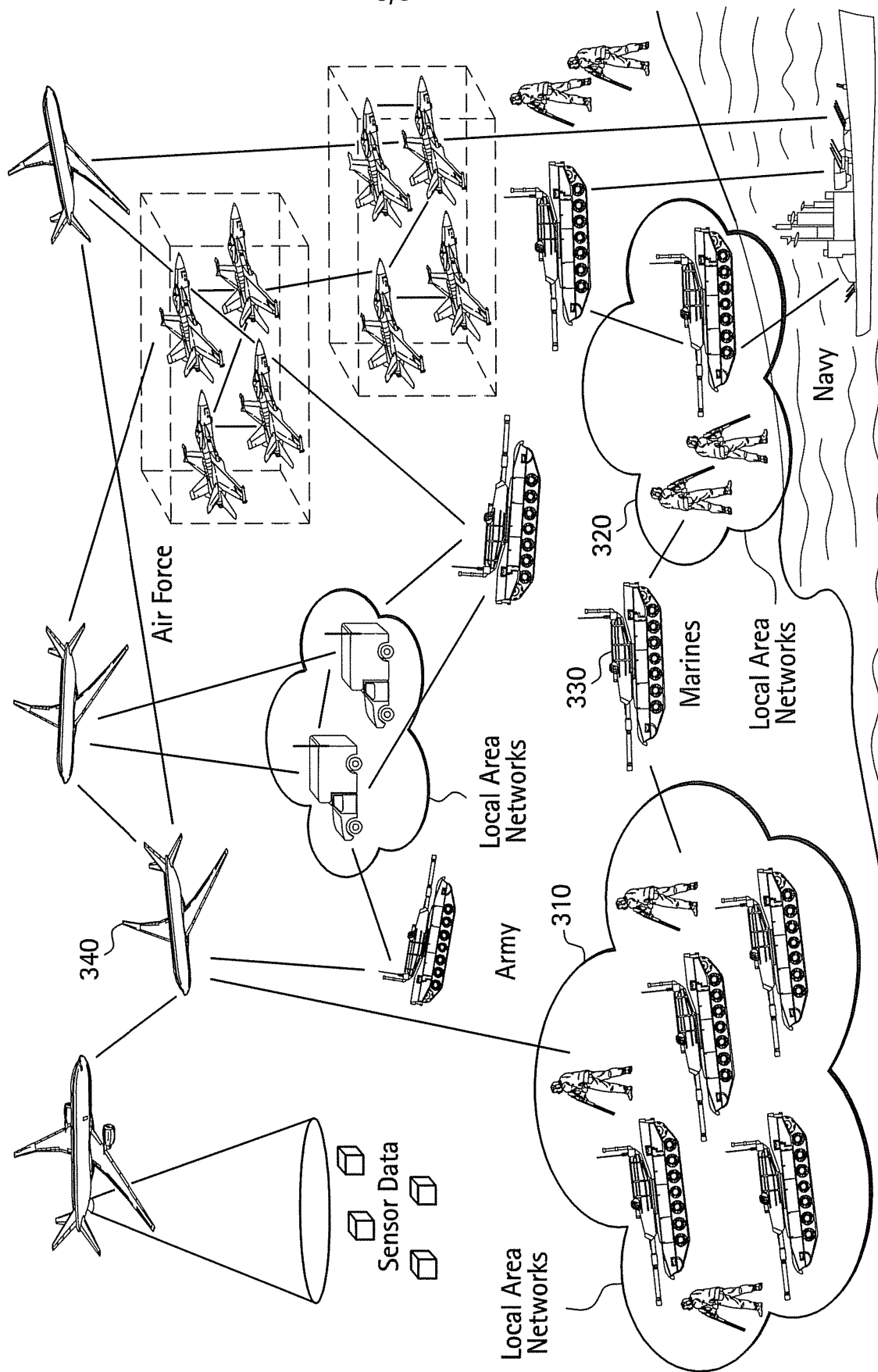
2/5

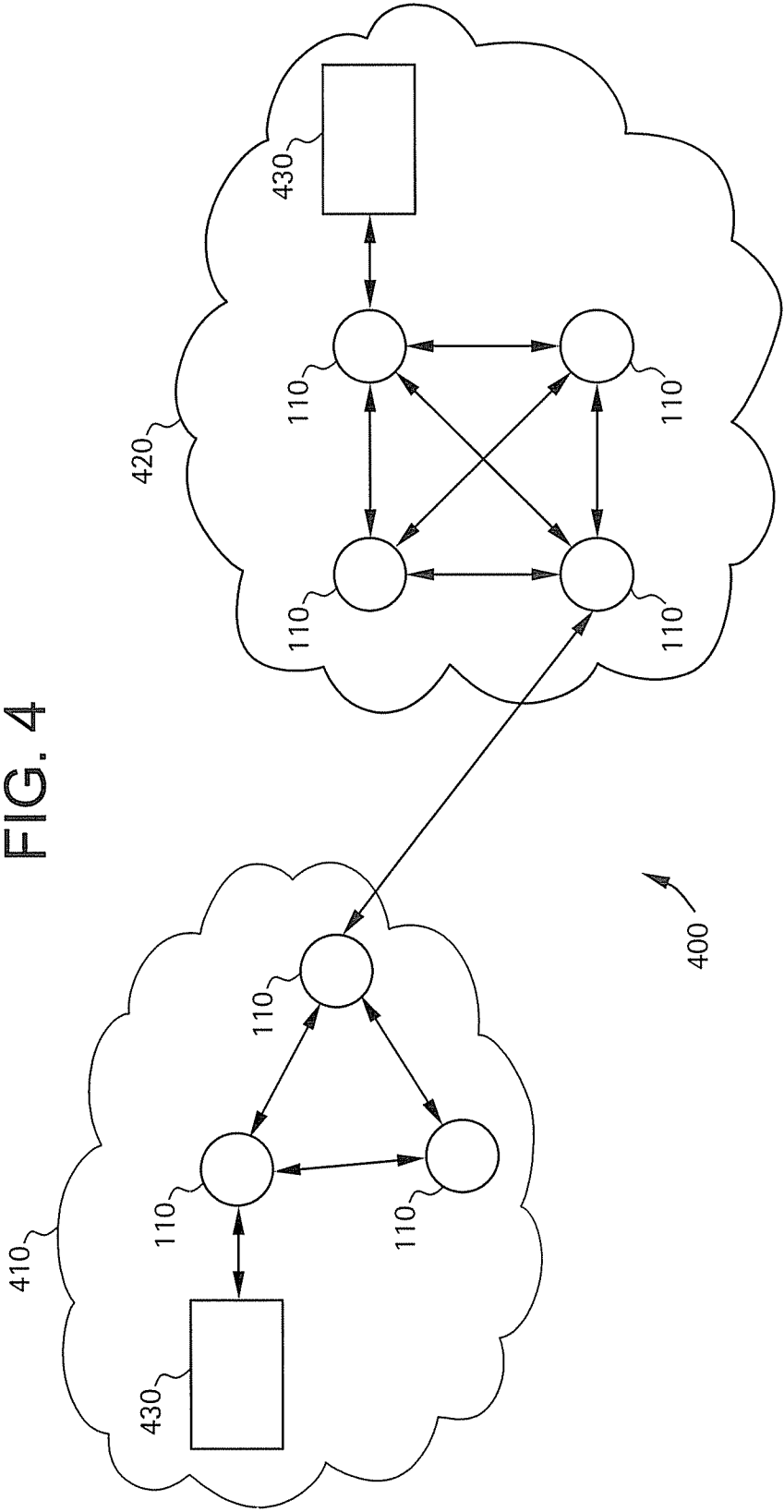
FIG. 2



3/5

FIG. 3





5/5

FIG. 5

