



- (51) **International Patent Classification:**
H04L 12/56 (2006.01) *H04L 29/12* (2006.01)
- (21) **International Application Number:**
PCT/US2012/035700
- (22) **International Filing Date:**
28 April 2012 (28.04.2012)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
13/096,194 28 April 2011 (28.04.2011) US
- (71) **Applicant (for all designated States except US):** MICROSOFT CORPORATION [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).
- (72) **Inventors:** ELSON, Jeremy E.; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). NIGHTINGALE, Edmund B.; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US).
- (81) **Designated States (unless otherwise indicated, for every kind of national protection available):** AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, 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, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) **Designated States (unless otherwise indicated, for every kind of regional protection available):** ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

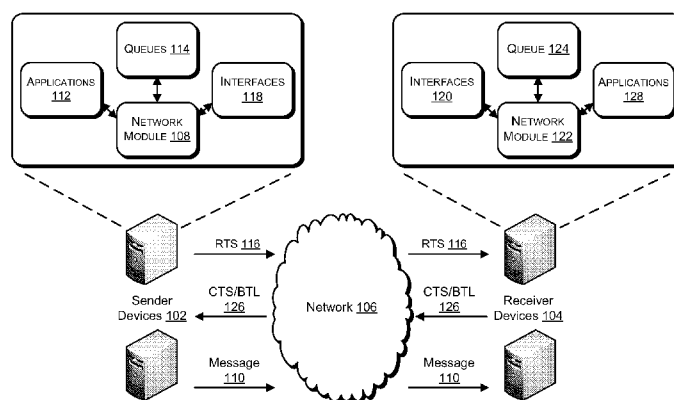
Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))

Published:

- without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: EFFECTIVE CIRCUITS IN PACKET-SWITCHED NETWORKS

**FIG. 1**

(57) **Abstract:** The creation of an effective circuit between a sender device and a receiver device over the packet-switched network is described herein. To establish the effective circuit, the sender device sends a request to the receiver device through the packet-switched network. The request is associated with a bandwidth reservation from the receiver device for reception of a message from the sender device. The receiver device receives multiple requests from multiple sender devices and reserves bandwidth for at least one of the sender devices. The receiver device then sends a response to the at least one sender device providing clearance to send the message to the receiver device using the reserved bandwidth, the request and response establishing the effective circuit. The receiver device may also decline the requests of the other sender devices, causing the other sender devices to send other requests to other receiver devices.

EFFECTIVE CIRCUITS IN PACKET-SWITCHED NETWORKS**BACKGROUND**

[0001] Messages are often transmitted through packet-switched networks using the Transmission Control Protocol/ Internet Protocol (TCP/IP) suite of protocols. TCP/IP determines the path bandwidth for message transmission by a sender device using a trial-and-error approach. The rate at which the sender device sends messages is increased until packet loss occurs. TCP/IP detects packet loss by noting that an acknowledgement was not received for a sent message. Packet loss often results from events such as buffer overflows that occur at intermediate or endpoint devices. Upon detecting packet loss, TCP/IP decreases the transmission rate and transmits messages at a gradually increasing rate until packet-loss is again detected. While this approach may work well for the transmission of a large number of messages between a device pair, it is not efficient for transmitting a small message or small number of messages because the messages may all be transmitted before TCP/IP has an opportunity to detect packet loss. If the initial rate used was too high, all of the messages sent may need to be retransmitted.

[0002] In addition to retransmission, packet loss may also result in a temporary halt of message transmission from a sender device. When TCP/IP detects a certain number of packet losses in a time period, the network is assumed to be congested and TCP/IP increases the delay between the transmission of messages from the sender device that is experiencing the packet loss. Such a delay can be far more costly to overall performance than retransmission. A sender device with ample bandwidth allowing parallel transmission of many small messages to many receivers may be at particular risk of having transmission halted when using TCP/IP.

[0003] The use of full bisection bandwidth networks can alleviate congestion in the core, but packet loss may still occur at end-stations. If TCP uses loss as a congestion signal, unnecessary retransmission delays can still occur..

[0004] Another approach for message transmission is to establish a circuit between two devices across the network. Circuits typically involve bandwidth allocation for a communication channel between the devices at the devices themselves and at each intermediate device. Because of the high cost of setting up and tearing down circuits, however, the short messages that typify many network applications cannot justify the cost of circuit set-up.

SUMMARY

[0005] Sender and receiver devices communicating through a packet-switched network establish an effective circuit between themselves by exchanging a request to send a message and a response providing clearance to send that message. Sufficient bandwidth
5 is ensured at the sender device by limiting the number of messages to be transmitted in parallel and is reserved at the receiver device responsive to receipt by the receiver device of a request to transmit a message. The packet-switched network between the sender and receiver devices may be a full bisection bandwidth network or a high bisection bandwidth network, making bandwidth reservation at intermediate devices unnecessary. The receiver
10 device may also be configured to decline sender requests when the received requests would utilize more bandwidth than is available to the receiver device. Responsive to having its request declined, a sender device may select a different message directed to a different receiver device and may transmit a request to send that message to the different receiver device. By switching to a different message and a different receiver device, the
15 sender device ensures that it does not sit idle while it has messages to transmit.

[0006] This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject
20 matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The detailed description is set forth with reference to the accompanying figures, in which the left-most digit of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in different
25 figures indicates similar or identical items or features.

[0008] FIG. 1 illustrates an example environment, in accordance with various embodiments.

[0009] FIGs. 2a-2b show example queues of a sender device, in accordance with various embodiments.

30 [0010] FIG. 3 shows an example queue of a receiver device, in accordance with various embodiments.

[0011] FIG. 4 is a flowchart showing a number of operations performed by a sender device to establish effective circuits with receiver devices over a packet-switched network, in accordance with various embodiments.

[0012] FIG. 5 is a flowchart showing a number of operations performed by a receiver device to establish an effective circuit with a sender device over a packet-switched network based on a reservation of receiver device bandwidth, in accordance with various embodiments.

5 [0013] FIG. 6 is a block diagram of an example computer system architecture, in accordance with various embodiments.

DETAILED DESCRIPTION

[0014] Described herein are techniques for establishing an effective circuit over a packet-switched network between a sender device and a receiver device. As used herein, the term “effective circuit” refers to a communication channel between two endpoint devices in which channel bandwidth is ensured. In contrast to a traditional circuit, the effective circuit discussed herein does not require substantial overhead to establish. By utilizing a full bisection bandwidth or high bisection bandwidth packet-switched network, reservation of bandwidth at intermediate devices becomes unnecessary. Bandwidth is ensured at the endpoints of the effective circuit by limiting message transmission at the sender device and by reserving bandwidth at the receiver device for messages of the sender device.

[0015] This disclosure leverages the observation that, in high- or full-bisection-bandwidth networks, the congestion occurs entirely at end-stations (sender and receiver devices), not in the network core. Consequently, it becomes feasible to do bandwidth allocation at the edges of the network, rather than the traditional method of watching for packet loss – which is an implicit signal from the network core.

[0016] To reserve receiver device bandwidth, the sender device transmits a request to send (RTS) message for each data message the sender device has to send to the receiver device. The sender device may have multiple data messages to transmit to one or more receiver devices and may limit the number of RTS messages sent based the bandwidth of the sender device for transmission of the data messages. In limiting the number of RTS messages, the sender device may assume negative responses to at least some of the RTS messages and thus request transmission of data messages that collectively would use greater bandwidth to transmit than the sender device has available. In some embodiments, the sender device may have a number of data messages to transmit to a single receiver device and may send one RTS message for the multiple data messages.

[0017] The receiver devices may receive RTS messages from multiple sender devices. These RTS messages may request permission to send data messages that

collectively would use greater bandwidth to receive than the receiver device has available. To ensure that it only receives as many simultaneous or concurrent data messages as it has bandwidth to receive, the receiver device may reserve bandwidth for one or more sender devices to transmit their data messages and respond to RTS messages of other sender devices with a message declining their requests to send data messages (these negative responses are referred to herein as “busy try later” (BTL) messages). The receiver device responds to sender devices for which it has reserved bandwidth with messages providing clearance to send data messages (these messages providing clearance to send are referred to herein as “CTS messages”). In selecting which sender devices to respond to with CTS messages, the receiver device may order the RTS messages in some fashion. For example, the receiver device may order the RTS messages based on priorities included in the RTS messages or based on numbers of BTL messages previously sent to the sender devices of the RTS messages.

[0018] Upon receiving a CTS messages, an effective circuit is established between the receiving sender device and the receiver device. The sender device then may proceed to send the data messages for which clearance was received to the receiver device. In some embodiments, the receiver device may only respond with CTS messages for a subset of the RTS messages sent to that receiver device by the sender device.

[0019] In response to receiving a BTL message, the sender device may select a different data message directed to a different receiver device and send an RTS message for that data message to that receiver device. In some embodiments, the data messages may be queued by the sender device and the sender device may select a first message from the queue that is directed to a different receiver device and for which an RTS has not previously been sent. By reserving bandwidth at an alternative receiver device, the sender device ensures that it does not have idle bandwidth while having data messages to send.

[0020] In some embodiments, the sender device need not send an RTS for every data message. Data messages that are control messages or that are smaller than a threshold size may simply be sent to receiver devices without establishing an effective circuit. This further reduces the overall overhead of the scheme.

Example Environment

[0021] FIG. 1 illustrates an example environment, in accordance with various embodiments. As shown in FIG. 1, sender devices 102 may communicate with receiver devices 104 through a packet-switched network 106. The sender devices 102 may be equipped with network modules 108 that receive messages 110 from applications 112 of

the sender devices 102 for transmission to the receiver devices 104. The network modules 108 may queue the messages 110 in queues 114 of the sender devices 102. Prior to transmitting the messages 110 to the receiver devices 104, the network modules 108 may transmit request to send (RTS) messages 116 for the messages 110 to the receiver devices 104 via interfaces 118 of the sender devices 102. These RTS messages 116 may then be received via interfaces 120 of the receiver devices 104 and processed by network modules 122 of the receiver devices 104. The network modules 122 may queue the RTS messages 116 in queues 124 of the receiver devices 104 for ordering of the RTS messages 116. The receiver devices 104 may then respond to the RTS messages 116 with messages 126 providing clearance to send (i.e., CTS messages) the messages 110 or with messages 126 declining the RTS messages 116 (i.e., BTL messages). Sender devices 102 receiving CTS messages 126 may then transmit the messages 110 to applications 128 of the receiver devices 104, the messages 110 being directed to those applications 128 by the applications 112 of the sender devices 102. Sender devices 102 receiving BTL messages 126 may select other receiver devices 104.

[0022] While the sender devices 102 and receiver devices 104 are shown herein as separate and distinct devices, it is to be understood that any computing device may be both a sender device 102 for some messages 110 and a receiver device 104 for other messages 110. Such a computing device would thus be equipped with the network module 108, applications 112, and queues 114 of a sender device 102 and with the network module 122, the queues 124, and the applications 128 of a receiver device 104. A computing device configured to be capable of acting as both a sender device 102 and a receiver device 104 may also have a single set of interfaces – in which case interfaces 118 and interfaces 120 would refer to a same set of interfaces – or two sets of interfaces – in which case one set (interfaces 118) may be dedicated to sending messages 110 and another set (interfaces 120) may be dedicated to receiving messages 110.

[0023] In various embodiments, the sender devices 102 and receiver devices 104 may be any sort of computing devices. For example, the sender devices 102 and receiver devices 104 may be or include personal computers (PC), laptop computers, servers or server farms, mainframes, tablet computers, work stations, telecommunication devices, personal digital assistants (PDAs), media players, media center devices, personal video recorders (PVR), televisions, or any other sort of device or devices. In some implementations, one or more of the sender devices 102 and receiver devices 104 represents one or more virtual machines implemented on one or more computing devices.

The nature and functions of the modules and data of the sender devices 102 and receiver devices 104 are described further herein. An example computing device for implementing one or both of a sender device 102 and a receiver device 104 is illustrated in FIG. 6 and is described below in greater detail with reference to that figure.

5 [0024] In various implementations, the network 106 may be any type or types of network, such as a wide area network (WAN), a local area network (LAN), or the Internet. Also, the network 106 may be public, private, or include both public and private networks. Further, the network 106 may be wired, wireless, or include both wired and wireless networks. The network 106 may utilize any one or more protocols for communication,
10 such as TCP/IP, other packet based protocols, or other protocols. In one embodiment, the network 106 may comprise a cloud computing environment that connects a plurality of nodes of the cloud computing environment. In such an embodiment, the sender devices 102 and receiver devices 104 may be nodes of the cloud computing environment.

[0025] As mentioned above, the network 106 may be a packet-switched network.

15 Such a network 106 may also be either a full bisection bandwidth or high bisection bandwidth network, allowing any node of the network to communicate with any other node at full speed without any delay in transmission introduced by intermediate nodes. In other embodiments, network 106 may be an infiniband network. The network 106 may also comprise one or more switches. The switches may be any sort of switches and may
20 each include network interface components, such as incoming and outgoing network interface components, each network interface component having a bandwidth. For example, a switch may have a number of incoming Ethernet ports and an incoming wireless port, as well as outgoing Ethernet and wireless ports. In some embodiments, the incoming bandwidth of a switch is proportioned to the outgoing bandwidth of the switch.
25 For instance, the collective incoming bandwidth of the network interfaces that serve devices “below” the switch in the network hierarchy may be ten gigabits per second, and the collective bandwidth of the network interface components up to core switches may also be ten gigabits per second. By proportioning the incoming and outgoing bandwidths of the switch, the introduction of bottlenecks associated with the switch is avoided. Such
30 switches with proportioned bandwidths are described in further detail in U.S. Patent Application No. 12/410,697, which is entitled “Data Center Without Structural Bottlenecks” and was filed on March 25, 2009, in U.S. Patent Application No. 12/410,745, which is entitled “Data Center Interconnect and Traffic Engineering” and was filed on

March 25, 2009, and in U.S. Patent Application No. 12/578,608, which is entitled “Agile Data Center Network Architecture” and was filed on October 14, 2009.

[0026] In some embodiments, as shown in FIG. 1, the sender devices 102 and receivers devices 104 may each include applications. Sender devices 102 may include applications 112 and receiver devices 104 may include applications 128. Applications 112 and 128 may be any sort of applications capable of reading and writing data, such as the data messages 110. The applications 112 and 128 may be the same type of application or different types. Also, applications 112 may represent a plurality of application and applications 128 may represent a corresponding plurality of applications. Applications 112 may be clients, servers, or peers with respect to the applications 128. From the perspective of the applications 112 and 128, the exchanged data messages 110 are sent directly without any awareness of the RTS/CTS message exchange or the effective circuit.

[0027] In various embodiments, data messages 110 may comprise any sort of data of any type and may have any size. Each data message 110 may be packetized into one or more packets for transmission over the network 106 by an application 112, a network module 108, or some other component of a sender device 102, such as a platform or network stack of the sender device 102. The number of packets comprising the data message 110 may vary with the size of the data message 110 and the network protocol used to transmit the data message 110. In addition to its data, each data message 110 may be associated with metadata. Such metadata may include identification of the receiver device 104 that the data message 110 is directed to and, if the data message 110 is part of a group of data messages 110, a sequence number identifying a position of the data message 110 within the group.

[0028] In some embodiments, a network module 108 may comprise any one or more modules, applications, processes, threads, or functions. The network module 108 may be a component of the platform of a sender device 102 or of the network stack of a sender device 102 or may be independent of other sender device 102 modules and applications. As mentioned above, the network module 108 may be configured to perform at least one of receiving data messages 110, bifurcating received data messages 110, queuing data messages 110, selecting data messages 110 for transmission, transmitting RTS messages 116 for the selected data messages 110, receiving CTS messages 126 or BTL messages 126 in response, transmitting a data message 110 in response to a CTS message 126, or selecting an alternate data message 110 in response to a BTL message 126. In addition to these operations, the network module 108 may also manage the

interfaces 118 of the sender device 102. These interfaces 118 may be selected by the network module 108 for the transmission of RTS messages 116 and data messages 110. Further, the network module 108 may perform these operations for any number of data messages 110 concurrently. For example, the network module 108 may receive a data message 110 from an application 112 while transmitting an RTS message 116 for another data message 110.

[0029] In some embodiments, upon reception of a data message 110 from an application 112, the network module 108 may determine whether the data message 110 is a control message or is smaller in size than a threshold in order to bifurcate the received data messages 110. The network module 108 provides data messages 110 that are control messages or smaller than a threshold to a network stack of the sender devices 102 for transmission using the interfaces 118 of the sender device 102. Data messages 110 that are not control messages and that are larger than the threshold are then queued by network module 108 for subsequent transmission.

[0030] In various embodiments, the network module 108 stores received data messages 110 in one queue 114 or several queues 114. The sender device 102 may have a single queue 114 in which data messages 110 for all receiver devices 104 are stored or may have a queue 114 associated with each receiver device 104. In other embodiments, the sender device 102 may utilize other numbers of queue 114 with other associations with receiver devices 104.

[0031] FIG. 2a, for example, illustrates an example of a queue 114 for storing data messages 110 that are directed to multiple receiver devices 104. As shown in FIG. 2a, a queue 202 (which may be an example of a queue 114) includes entries 204 for multiple data messages 110. While these entries 204 are shown as rows of a table, it is to be understood that the entries 204 may also be represented as columns or as items in some non-tabular form. Each entry 204 includes a message identifier 206, an indicator 208 of whether an RTS message 116 has been sent for the data message 110 represented by that entry 204, and an identifier 210 of a target receiver device 104. Each message identifier 206 may be a data value of any type – such as an integer or string – that uniquely identifies a data message 110. Each indicator 208 may be, for example, a binary value or one of a limited set of other value types indicating whether an RTS message 116 has been sent. Each identifier 210 of a target receiver device 104 may be a network identifier of the receiver device 104, such as an IP address or other form of address.

[0032] FIG. 2b illustrates examples of multiple queues 114 for storing data messages 110, one queue 114 for each of multiple receiver devices 104. As shown in FIG. 2b, each of queues 212, 214, and 216 (which may be examples of queues 114) include entries 204 for one or more data messages 110. While these entries 204 are shown as rows of a table, it is to be understood that the entries 204 may equally be represented as columns or as items in some non-tabular form. Each entry 204 includes a message identifier 206 and an indicator 208 of whether an RTS message 116 has been sent for the data message 110 represented by that entry 204. There is no need for a field representing an identifier 210 of the target receiver device 104 of a data message 110 because each of queues 212, 214, and 216 is associated with a specific receiver device 104. Each message identifier 206 may be a data value of any type – such as an integer or string – that uniquely identifies a data message 110. Each indicator 208 may be, for example, a binary value or one of a limited set of other value types indicating whether an RTS message 116 has been sent.

[0033] The queues 114 may be any sort of data structures known in the art, including data structures of modules, such as the network module 108, or other files or databases. Also, the queues 114 may be first-in-first-out (FIFO) data structures, as is known in the art, or may have data messages 110 added in or removed from the queues 114 in a different order. For example, data messages 110 may be added to the ends of the queues 114 but may be removed from the queues 114 in a different order. In such an example, the place of a data message 110 in the queue may be one of multiple factors in selecting and removing the data message 110 from a queue 114 for transmission.

[0034] In various embodiments, the network module 108 selects one or more data messages 110 from one or more of the queues 114 responsive to data messages 110 being added to a queue 114, responsive to a timer that estimates a time at which the sender device 102 will complete transmission of other data messages, or responsive to input from a platform of the sender device 102. The network module 108 may select data messages 110 from the queues 114 based on one or more factors. These factors may include a place of a data message 110 within a queue 114, which queue 114 that data messages 110 were previously selected from, sizes of data messages 110, available bandwidth of the sender device interfaces 118, an ability to chain together a number of data messages 110 for transmission to a single receiver device 104. These factors are provided only for the sake of illustration and any other number of factors may be used in the alternative.

[0035] In some embodiments, the network module 108 first determines an available bandwidth of the sender device interfaces 118. This available bandwidth may be a preset number known to the network module 108, such as a number representing eighty or ninety percent of the total bandwidth of the interfaces 118. In other embodiments, the available bandwidth may vary and may be determined repeatedly at different times.

[0036] The network module 108 may then select data messages 110 that will utilize as much of available bandwidth as possible for transmission. In one embodiment, the network module 108 selects data messages 110 from a head of a queue or queues 114 without reference to the receiver device 104 until the collective size of the selected data messages 110 approximates the available bandwidth. In other embodiments in which the sender device 102 has multiple queues 114 for multiple receiver devices 104, the network module 108 may service the queues in a round-robin fashion, selecting data messages 110 from a next queue 114 to be serviced. In further embodiments, the network module 108 may chain data messages 110 for a single receiver device 104, preferring to select multiple data messages 110 for the single receiver device 104 in order to reduce the number of receiver device bandwidth reservations that are needed. In such embodiments, the network module 108 tracks receiver devices 104 that chained data messages 110 have been transmitted to in order to prevent a single receiver device 104 from dominating message transmission. In yet other embodiments, the network module 108 prioritizes data messages 110 for which RTS messages 116 were previously sent and for which a threshold amount of time has passed. The reason for waiting a threshold amount of time is to allow the receiver device 104 that provided the BTL message 126 to complete its reception of other data messages 110 and free its reserved bandwidth.

[0037] In some embodiments, by considering the available bandwidth of the interfaces 118 in selecting the data messages 110, the network module 108 limits the number of data messages 110 that can be transmitted in parallel in order to avoid transmission bottlenecks on the sender device 102. Given that RTS messages 116 for a number of selected data messages 110 may be declined, however, the network module 108 may select a greater number of data messages 110 than there is available bandwidth to transmit in order to ensure full utilization of the available bandwidth.

[0038] In various embodiments, upon selecting the data messages 110, the network module 108 generates and transmits RTS messages 116 for the selected data messages 110. The network module 108 may generate and transmit an RTS message 116 for each selected data message 110 or may generate and transmit an RTS message 116 for a group

of data messages 110 that are to be transmitted in parallel to a single receiver device 104. RTS messages 116 may be small messages that identify a specific data message 110 that they are associated with, a size or sizes of the data message 110 or messages 110 associated with that RTS message, an identifier to the target receiver device 104, and/or a message priority. The message priorities may be determined by the network module 108 based on the ratio of data messages 110 that are queued for the target receiver device 104 of the RTS message versus data messages 110 that are queued for other receiver devices 104, as well as on an overall size of the queue 114 or queues 114. In embodiments where the RTS message 116 is associated with a chain of data messages 110, the RTS message 116 may identify the number of data messages 110 for that receiver device 104.

[0039] Upon generating the RTS messages 116, the network module 108 selects one or more available interfaces 118 for transmission of the RTS messages 116. The interfaces 118 may be any sort of network interfaces known in the art, such as wired or wireless interfaces for WANs or LANs, wireless interfaces for PANs or Bluetooth® networks, or any other sort of interfaces. For example, interfaces 118 may comprise one or more network cards. These interfaces 118 may comprise a pool of interfaces that is managed by the network module 108 or another component as a collective resource that shares a same set of queues and queue states (i.e., queues 114). In one embodiment, the bandwidth of the interfaces 118 may be allocated to different processes, including one or more processes of the network module 108. This may include independent use of a single network interface 118 by multiple processes. As long as the processes have divided the network bandwidth statically ahead of time, the processes may use the allocated bandwidth without coordination. Upon selecting the one or more available interfaces 118, the network module 108 transmits the RTS messages 116 over the one or more available interfaces 118.

[0040] In response to sending RTS messages 116, the sender device 102 receives CTS messages 126, BTL messages 126, or both. One CTS message 126 or BTL message 126 may be returned to the sender device 102 for each RTS message 116 that it sends.

[0041] Upon receiving a CTS message 126 via an interface 118, the network module 108 determines which data message 110 or data messages 110 the CTS message 126 provides clearance for. As mentioned above, the receipt of a CTS message 126 from a receiver device 104 establishes an effective circuit between the sender device 102 and that receiver device 104. The CTS message 126 may identify a specific data message 110 by a message identifier 206 or may refer to a specific RTS message 116, which may in

turn be related by the network module 108 to a specific data message 110. The CTS message 126 may also identify multiple data messages 110 that are cleared for transmission to the receiver device 104 that sent the CTS message 126. For example, the CTS message 126 may be responsive to an RTS message 116 that requested permission to send several data messages 110 to the receiver device 104. Such a CTS message 126 may provide clearance to send one, some, or all of the data messages 110 identified in the RTS message 116.

[0042] In various embodiments, upon determining the data messages 110 identified by the CTS messages 126, the network module 108 may select one or more of the interfaces 118 to transmit the data messages 110 and transmit the data messages 110 via the selected interfaces 118. In one embodiment, the interface 118 used to transmit a data message 110 may be different from the interface 118 used to transmit the RTS message 116 for that data message 110. Upon transmitting a data message 110, the network module 108 may remove the data message 110 from the queue or queues 114.

[0043] In addition to or instead of receiving one or more CTS messages 126, the sender device 102 may receive, via an interface 118, one or more BTL messages 126. Upon receiving a BTL message 126, the network module 108 determines which data message 110 or data messages 110 the BTL message 126 has declined. The network module 108 may then update the indicators 208 of those data messages 110 in the queues 114 to reflect that RTS messages 116 for those data messages 110 have been declined.

[0044] The network module 108 may then proceed to select one or more alternative data messages 110 for transmission in place of the declined data messages 110. In some embodiments, the network module 108 may utilize the above-described factors used to select the now-declined data messages 110 in selecting the alternative data messages 110. The network module 108 may also consider the size of the declined data message 110 or the collective size of multiple declined data messages 110 in selecting one or more alternative data messages 110. Also, in selecting the one or more alternative messages, the network module 108 may filter out data messages 110 for which an RTS message 116 has been declined and data messages 110 that are directed to the receiver device 104 from which the BTL message 126 was received. Once the alternative data messages 110 are selected, the network module 108 generates RTS messages 116 for the alternative data messages 110, selects interfaces 118 to use in transmitting the RTS messages 116, and transmits the RTS messages 116. By selecting alternative data messages 110 rather than waiting to transmit previously selected data messages 110, the

network module 108 ensures that the sender device 102 does not sit idle while it has data messages 110 waiting for transmission.

[0045] In some embodiments, as mentioned above, the sender device 102 may utilize a timer to determine when to select further data messages 110 for transmission.

5 The timer may be a part of the network module 108 or a separate component. The timer is initiated either when data messages 110 are selected or when transmission of data messages 110 begins and estimates the time for transmitting the data messages 110 based on the available bandwidth and the sizes of the data messages 110. Once that estimated time is reached, the timer informs the network module 108 that the time has been reached
10 to enable the network module 108 to select further data messages 110 for transmission.

[0046] In various embodiments, a network module 116 may comprise any one or more modules, applications, processes, threads, or functions. The network module 116 may be a component of the platform of a receiver device 104 or of the network stack of a receiver device 104 or may be independent of other receiver device 104 modules and
15 applications. As mentioned above, the network module 116 may be configured to perform at least one of receiving RTS messages 116, ordering the RTS messages 116, reserving bandwidth for one or more sender devices 102 of RTS messages 116, generating and sending CTS messages 126 to sender devices 102 for which bandwidth has been reserved, generating and sending BTL messages 126 to other sender devices 102, tracking the
20 number of times BTL messages 126 have been sent to a sender device 102, and receiving data messages 110 using the reserved bandwidth. Further, the network module 116 may perform these operations for any number of data messages 110 concurrently. For example, the network module 116 may receive a data message 110 while transmitting a CTS message 126 responsive to an RTS message 116 for another data message 110.

25 [0047] In some embodiments, when a network module 116 receives RTS messages 116 via the interfaces 120, the network module 116 may queue these RTS messages in a queue 124 for ordering. The RTS messages 116 may be received from a plurality of sender devices 102, and the interfaces 120 receiving those RTS messages 116 may be any sort of interfaces, such as interfaces of the same or different types as the interfaces 118
30 described above. The queue 124 may be any sort of data structure, such as a data structure of the network module 116.

[0048] For example, FIG. 3 illustrates an example of a queue 124 that includes RTS message entries that are used to order the received RTS messages 116. As shown, a queue 302 (which may be an example of the queue 124) includes entries 304 for multiple

RTS messages 116. While these entries 304 are shown as rows of a table, it is to be understood that the entries 304 may equally be represented as columns or as items in some non-tabular form. Each entry 304 includes a message identifier 306, a count 308 of BTL messages 126 sent to a sender device 102 responsive to RTS messages 116 associated with a specific data message 110, an indicator 310 of a priority of a data message 110 associated with the RTS message 116 of the entry 304, and an identifier 312 of a sender device 102 that sent the RTS message 116 of the entry 304. Each message identifier 306 may be a data value of any type – such as an integer or string – that uniquely identifies either the RTS message 116 or the data message 110 associated with that RTS message 116. Each count 308 may be an integer or some other number value that is retrieved from another data structure of the receiver device 104 which is used to track BTL messages 126 sent responsive to RTS messages 116 associated with a specific data message 110. When an RTS message 116 is received, the network module 116 checks this data structure for the number of BTL messages 126 associated with the data message 110 referenced by the received RTS message 116. Each indicator 310 may be, for example, a binary value or one of a limited set of other value types indicating whether has been prioritized. Alternatively, the indicator 310 for each entry 304 may indicate a specific priority level of a data message 110 if several priority levels are available. Each identifier 312 of a sender device 102 may be a network identifier of the sender device 102, such as an IP address or other form of address.

[0049] In various embodiments, the network module 116 may order the RTS messages 116 in the queue 124 based on any number of factors. For example, the network module 116 may order the RTS messages 116 based on message priorities and/or based on the count of BTL messages 126 mentioned above. The network module 116 may also consider the receiver device bandwidth available for reservation and the sizes of the data messages 110 corresponding to the RTS messages 116.

[0050] Based on the ordering, the network module may reserve bandwidth for one or more of the sender devices 102 associated with received RTS messages 116. These may be the sender devices 102 corresponding to the highest ordered RTS messages 116. In some embodiments, the network module 116 may apply a maximum bandwidth reservation for sender devices 102 such that no sender device 102 may be allocated the entire available bandwidth of the receiver device 102. This maximum bandwidth reservation may prevent a sender device 102 from monopolizing the receiver device bandwidth.

[0051] To effect the bandwidth reservations, the network module 116 sends CTS messages 126 to the sender devices 102 for which bandwidth is reserved. The CTS messages 126 may identify the RTS messages 116 that they are responsive to or the data messages 110 for which clearance is being provided. In some embodiments, the CTS messages 126 may also identify a number or identities of multiple data messages 110 in the case where an RTS message 116 identified multiple data messages 110.

[0052] The network module 116 may then generate and send BTL messages 126 in response to RTS messages 116 that did not result in a bandwidth reservation and CTS message 126 response. The BTL messages 126 may identify the RTS messages 116 that they are responsive to or the data messages 110 for which clearance is being declined. In some embodiments, the BTL messages 126 may also identify a number or identities of multiple data messages 110 in the case where an RTS message 116 identified multiple data messages 110. After sending the BTL messages 126, the network module 116 may update the counts of BTL messages for specific data messages 110 that are stored in a data structure of the receiver device 104, as discussed above. The network module 116 may also flush the queue 124 of the entries 304 in the queue 124 after each entry 304 is responded to with a CTS message 126 or a BTL message 126.

[0053] In various embodiments, the network module 116 may then receive the data messages 110 using the reserved bandwidth. Because the receiver device 104 only receives data messages 110 that it has provided clearance to send, the risk of buffer overflow at the receiver device 104 is averted. Upon receiving the data messages 110, the network module 116 provides the data messages 110 to the applications 128 to which they are directed and releases the bandwidth reservation(s) associated with the data messages 110. In some embodiments, completion of receiving the data messages 110 and release of the bandwidth reservations(s) triggers the sending of a next CTS message 126.

Example Operations

[0054] FIGs. 4 and 5 are flowcharts illustrating example operations of processes. The example operations of the processes are illustrated in individual blocks and summarized with reference to those blocks. These processes are illustrated as logical flow graphs, each operation of which may represent a set of operations that can be implemented in hardware, software, or a combination thereof. In the context of software, the operations represent computer-executable instructions stored on one or more computer-readable storage media that, when executed by one or more processors, perform the recited operations. Generally, computer-executable instructions include routines, programs,

objects, components, data structures, and the like that perform particular functions or implement particular abstract data types. The order in which the operations are described is not intended to be construed as a limitation, and any number of the described operations can be combined in any order and/or in parallel to implement the process.

5 [0055] FIG. 4 is a flowchart showing a number of operations performed by a sender device to establish effective circuits with receiver devices over a packet-switched network, in accordance with various embodiments. As illustrated at block 402, a sender device may receive a message for transmission from one or more applications of that sender device to a receiver device. The sender device and receiver device may be
10 connected by a packet-switched network, such as a full bisection bandwidth network or a high bisection bandwidth network. At block 404, upon receiving the message, the sender device may determine if the message is a control message or has a size that is smaller than a threshold size. At block 406, if the message is a control message or is smaller than the threshold size, the sender device may transmit the message to the receiver device specified
15 for the message. At block 408, if the message is not a control message or is larger than the threshold size, the sender device queues the message for subsequent transmission.

[0056] At block 410, the sender device selects a message for transmission to a receiver device. The sender device may select a message from a sender device queue, such as the first message in the queue. In some embodiments, the sender device may
20 select multiple messages for transmission, such as multiple messages to a same receiver device for chained transmission or multiple messages to multiple receiver devices. At block 410a, the sender device limits the number of messages selected based on an available transmission bandwidth of the sender device so that all messages selected may be transmitted concurrently without waiting for any others of the selected messages to
25 complete sending. At block 410b, the sender device may select a message from a queue that is associated with a previous request to send (RTS). The sender device may have previously sent an RTS for that message and may have received a response declining permission to send the message.

[0057] At block 412, the sender device sends an RTS for the selected message.
30 The RTS may include the size of the message. In some embodiments, sending comprises sending the multiple requests via multiple interfaces of the sender device, the multiple interfaces sharing a same set of message queues and queue states. In one embodiment, the RTS may be sent via a different interface than the message. At block 412a, the sender device sends multiple RTSs to multiple receiver devices in parallel. At block 412b, the

sender device chains RTSs to a single receiver device, sending the chained RTSs to the receiver device concurrently. At 412c, the chaining may include indicating in a single RTS the number of messages that the sender device has to send to the receiver device.

[0058] In response to the RTS, the sender device receives a clearance to send (CTS) at block 414 or response declining the RTS (referred to herein as a “busy try later” (BTL) message) at block 422. At block 414, the sender device receives a CTS from the receiver device. The RTS and CTS create an effective circuit between the sender device and receiver device over the packet-switched network. The receiver device sends CTS messages to sender devices that it has reserved bandwidth for. At block 416, the sender device determines which messages were cleared by the CTS. For example, if the sender device indicated at block 412c that the sender device has ten messages to transmit, the receiver device might only respond with a CTS or CTSs clearing five of the messages. At block 418, the sender device sends the cleared message or messages to the receiver device to be received by the receiver device using the reserved band width. By utilizing its full bandwidth and the reserved bandwidth of receiver devices, the sender device ensures that the messages are sent without any delay, packet loss, or unused bandwidth.

[0059] At block 420, the sender device utilizes a timer to estimate when a next message should be transmitted to a receiver device. The timer estimates a time to send the selected message or messages and thus a time when message transmission will be complete and sender device bandwidth will be available.

[0060] At block 422, rather than receiving a CTS, the sender device receives a BTL from the receiver device declining the request to send the message. At block 424, responsive to the BTL, the sender device selects another message to send to another receiver device. In some embodiments, the sender device selects the first message in the queue for which an RTS has not been sent and which is directed to another receiver device. An RTS message for the other message is then sent at block 412. The sender device selects another message for transmission with the RTS is declined to ensure that the sender device does not sit idle while it has messages to transmit and available bandwidth to transmit those messages.

[0061] FIG. 5 is a flowchart showing a number of operations performed by a receiver device to establish an effective circuit with a sender device over a packet-switched network based on a reservation of receiver device bandwidth, in accordance with various embodiments. As illustrated at block 502, a receiver device may receive a plurality of requests to send (RTS) messages from a plurality of sender devices. In some

embodiments, the RTS messages include sizes of the messages that the sender devices are requesting to send. At block 504, upon receiving these RTS messages, the receiver device may order the RTS messages based at least on one of priorities included in one or more of the RTS messages or on a number of times a sender device has received responses declining its RTS messages.

[0062] At block 506, the receiver device may reserve bandwidth for at least one sender device for receiving messages from the at least one sender device. The at least one sender device may have its RTS message selected from the ordered RTS messages. In some embodiments, the receiver device reserves bandwidth for receiving multiple messages from a sender device corresponding to multiple RTS messages. Also, in further embodiments, the number of sender devices for whom bandwidth is reserved may be related to sizes of the messages the sender devices are requesting to send. At block 508, the receiver device then responds to the RTS message or RTS messages of the at least one sender device, providing that at least one sender device with clearance to send (CTS) a message or messages using the reserved bandwidth.

[0063] At block 510, the receiver device responds to the other sender devices with responses declining the RTS messages of the other sender devices. These negative responses are referred to herein as “busy try later” (BTL) messages. The BTL messages indicate to the other sender devices that they should send RTS messages to other receiver devices in order to fully utilize their sender device bandwidths. In some embodiments, when the receiver device sends a BTL message to a sender device, it increments a representation of a number of BTL messages sent to that sender device. The representation is then used for subsequent ordering operations.

[0064] At block 512, the receiver device receives the message or messages from the at least one sender device using the reserved bandwidth.

Example System Architecture

[0065] FIG. 6 is a block diagram of an example computer system architecture for a sender device 102 or a receiver device 104, in accordance with various embodiments. As shown, the sender device 102/receiver device 104 may comprise at least a memory 602 (including a cache memory) and one or more processing units (or processor(s)) 604. The processor(s) 604 may be implemented as appropriate in hardware, software, firmware, or combinations thereof. Software or firmware implementations of the processor(s) 604 may include computer-executable or machine-executable instructions written in any suitable

programming language to perform the various functions described. Processor(s) 604 may also or alternatively include one or more graphic processing units (GPUs).

[0066] Memory 602 may store program instructions that are loadable and executable on the processor(s) 604, as well as data generated during the execution of these programs. Depending on the configuration and type of computing device, memory 602 may be volatile (such as random access memory (RAM)) and/or non-volatile (such as read-only memory (ROM), flash memory, etc.). The computing device or server may also include additional removable storage 606 and/or non-removable storage 608 including, but not limited to, magnetic storage, optical disks, and/or tape storage. The disk drives and their associated computer-readable media may provide non-volatile storage of computer readable instructions, data structures, program modules, and other data for the computing devices. In some implementations, the memory 602 may include multiple different types of memory, such as static random access memory (SRAM), dynamic random access memory (DRAM), or ROM.

[0067] Computer-readable media includes, at least, two types of computer-readable media, namely computer storage media and communications media.

[0068] Computer storage media includes volatile and non-volatile, removable and non-removable media implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules, or other data. Computer storage media includes, but is not limited to, RAM, ROM, erasable programmable read-only memory (EEPROM), flash memory or other memory technology, compact disc read-only memory (CD-ROM), digital versatile disks (DVD) or other optical storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other non-transmission medium that can be used to store information for access by a computing device.

[0069] In contrast, communication media may embody computer readable instructions, data structures, program modules, or other data in a modulated data signal, such as a carrier wave, or other transmission mechanism. As defined herein, computer storage media does not include communication media.

[0070] The sender device 102/receiver device 104 may also contain communications connection(s) 610 that allow the sender device 102/receiver device 104 to communicate with a stored database, another computing device or server, user terminals, and/or other devices on a network, such as network 106. The sender device 102/receiver device 104 may also include input device(s) 612, such as a keyboard, mouse, pen, voice

input device, touch input device, etc., and output device(s) 614, such as a display, speakers, printer, etc.

[0071] The sender device 102/receiver device 104 may further include an interface 118/120 for sending and receiving data across the network 106. As mentioned above, the interface 118/120 may comprise any number of network interfaces, each having an associated bandwidth for transmitting and receiving data.

[0072] Turning to the contents of the memory 602 in more detail, the memory 602 may include platform 616. The platform 616 may comprise an operating system and/or one or more application programs or services. If the memory 602 belongs to a sender device 102, the memory 602 may also include a network module 108 and applications 112, which may each represent any one or more modules, applications, processes, threads, or functions, and one or more queues 114, which may represent any storage formats or data structures. The network module 108, applications 112, and queues 114 are described above in greater detail. If the memory 602 belongs to a receiver device 104, the memory 602 may also include a network module 122 and applications 128, which may each represent any one or more modules, applications, processes, threads, or functions, and one or more queues 124, which may represent any storage formats or data structures. The network module 122, applications 128, and queues 124 are described above in greater detail.

Conclusion

[0073] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as exemplary forms of implementing the claims.

CLAIMS

WE CLAIM:

1. A method comprising:

5 sending, by a sender device, a request to a receiver device connected to the sender device by a packet-switched network, the request associated with a bandwidth reservation from the receiver device for reception of a message from the sender device;

10 receiving, by the sender device, a response from the receiver device providing clearance to send the message to the receiver device using reserved bandwidth of the receiver device, the request and the response creating an effective circuit between the sender device and receiver device over the packet-switched network.

2. The method of claim 1, further comprising:

determining if another message is a control message or is smaller than a threshold size; and

15 responsive to determining that the other message is a control message or is smaller than the threshold size, sending the other message to the receiver device without sending a prior request associated with a bandwidth reservation.

3. The method of claim 1, further comprising sending multiple requests to multiple receiver devices in parallel.

20 4. The method of claim 3, further comprising limiting the number of requests sent based on an available bandwidth of the sender device.

5. The method of claim 1, further comprising concurrently sending multiple requests for multiple messages to the receiver device.

25 6. The method of claim 1, wherein the request indicates a number of messages that the sender device has to send to the receiver device and method further comprises receiving a response providing clearance for some or all of the messages.

7. A sender device comprising:

a processor; and

a network module configured to be operated by the processor to perform operations including:

30 sending to a first receiver device a first request to send (RTS) a first message, the first RTS seeking a first bandwidth reservation from the first receiver device for reception of the first message from the sender device;

receiving a response from the first receiver device declining the first RTS;
and

based on the response, sending a second RTS to a second receiver device, the second RTS seeking a second bandwidth reservation from the second receiver device for reception of a second message from the sender device.

8. The sender device of claim 7, further comprising one or more queues storing
5 messages for transmission to receiver devices.

9. The sender device of claim 8, wherein the network module is further configured to select as the second message from the one or more queues a message in the queues that is not directed to the first receiver device and for which an RTS has not been sent.

10. One or more computer-readable storage media comprising computer-executable instructions configured to program a receiver device to perform operations including:
10

receiving a plurality of requests to send messages from a plurality of sender devices;

reserving bandwidth for at least one of the sender devices; and

15 sending the at least one of the sender devices a response providing clearance to send one or more messages to the receiver device using the reserved bandwidth.

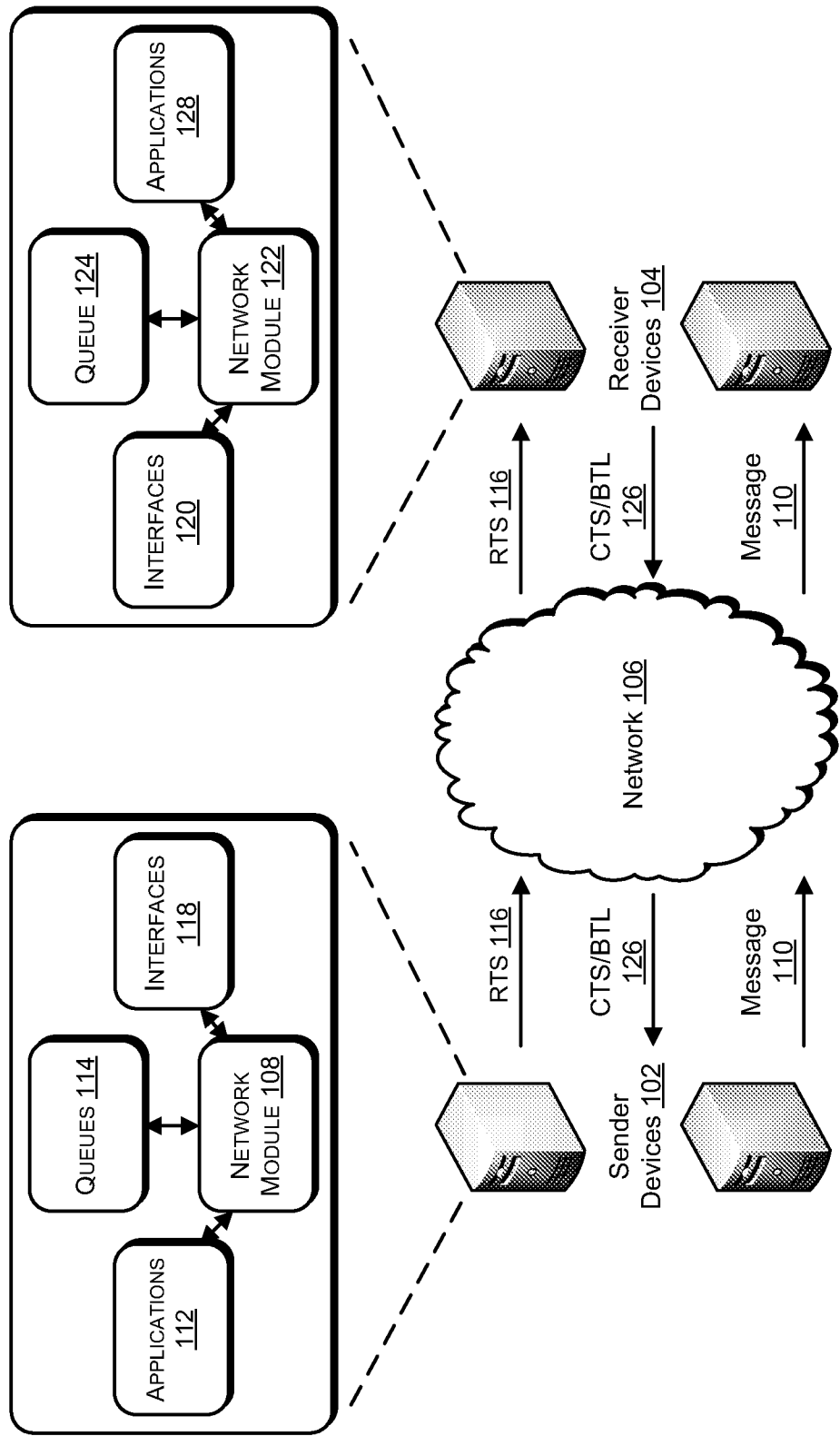


FIG. 1

2/6

Messages <u>204</u>	Msg ID <u>206</u>	RTS? <u>208</u>	Receiver Device <u>210</u>
	0001	Y	Receiver 1
	0010	N	Receiver 2
	0011	N	Receiver 1
	0100	N	Receiver 3

Queue 202

FIG. 2a

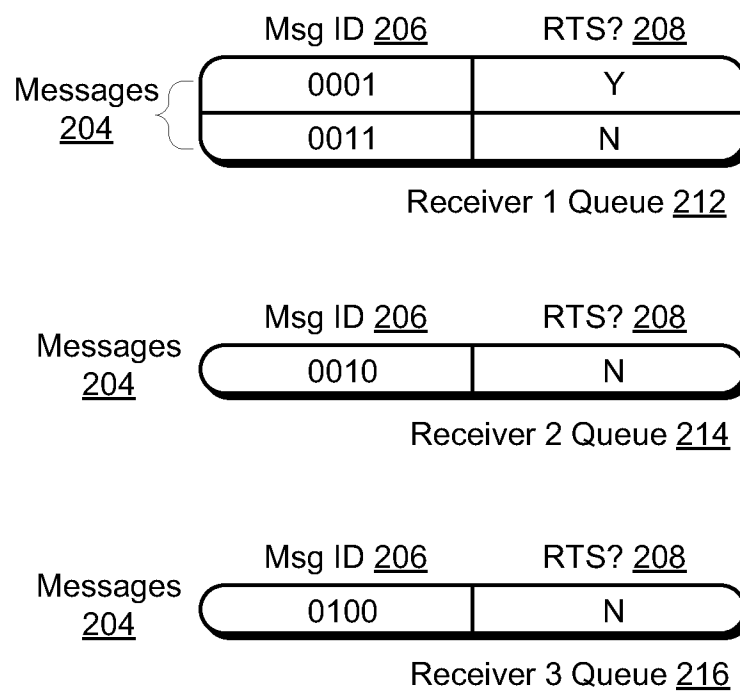


FIG. 2b

Messages
304

Msg ID <u>306</u>	Num. BTLs <u>308</u>	Priority <u>310</u>	Sender Device <u>312</u>
1010	5	N	Sender 1
0010	0	Y	Sender 2
1001	1	N	Sender 3
0101	0	N	Sender 2

Queue 302

FIG. 3

4/6

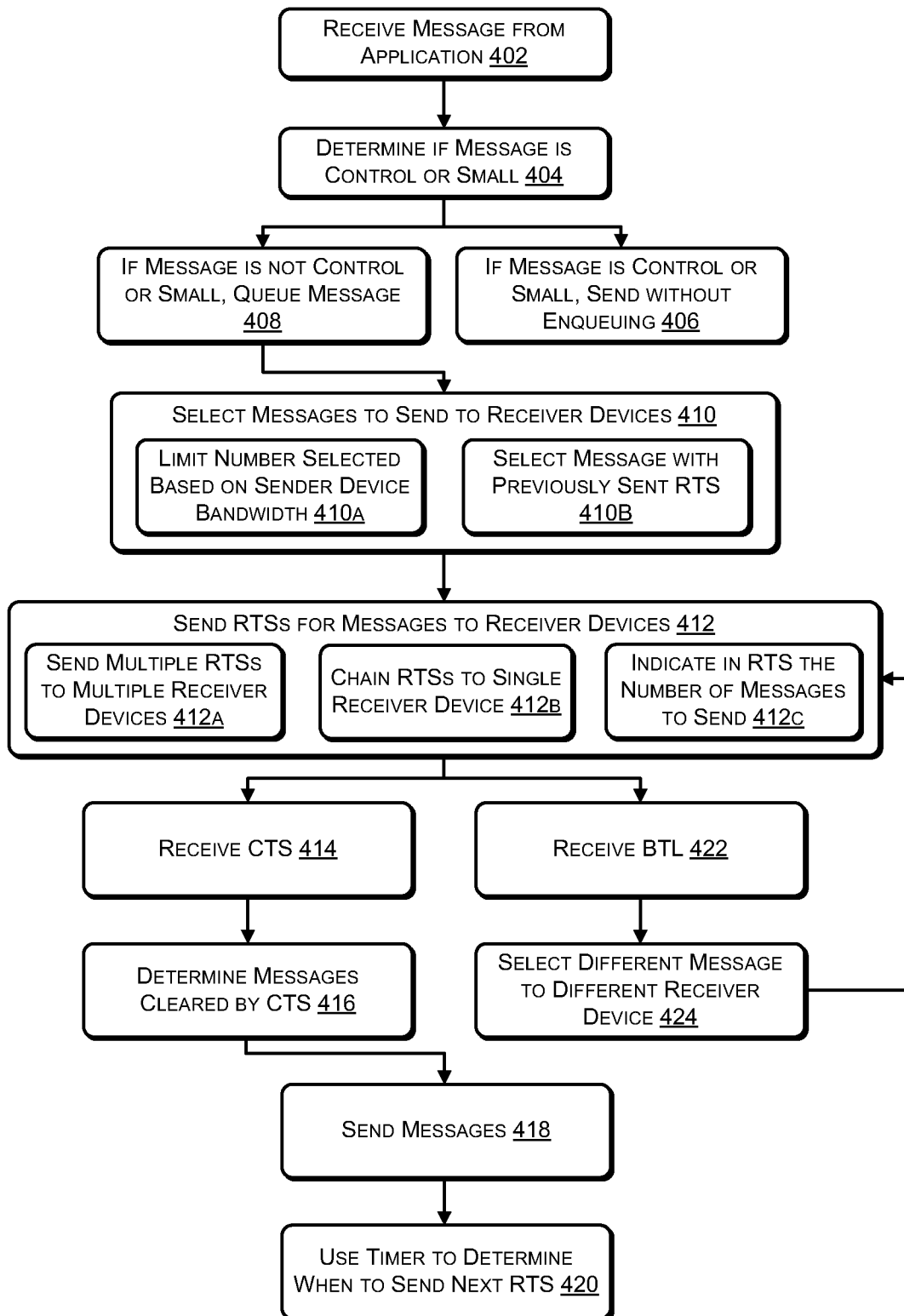
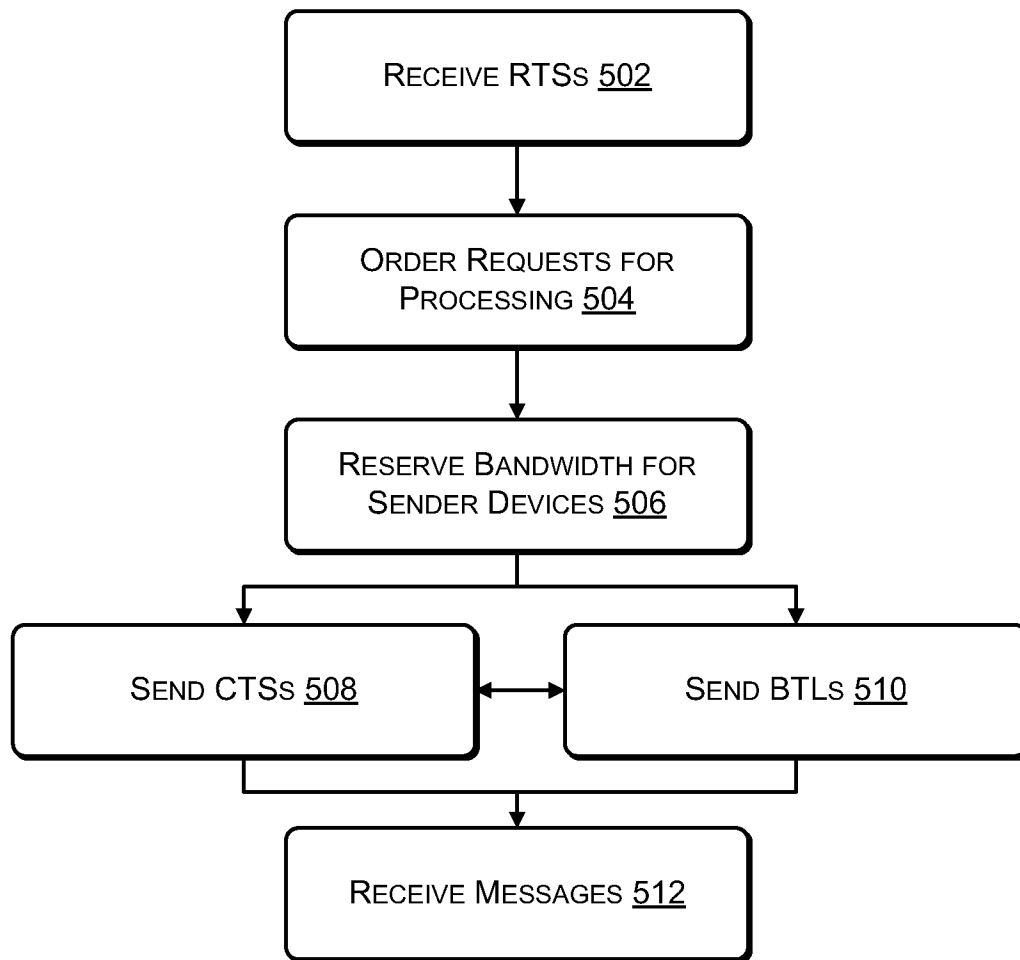
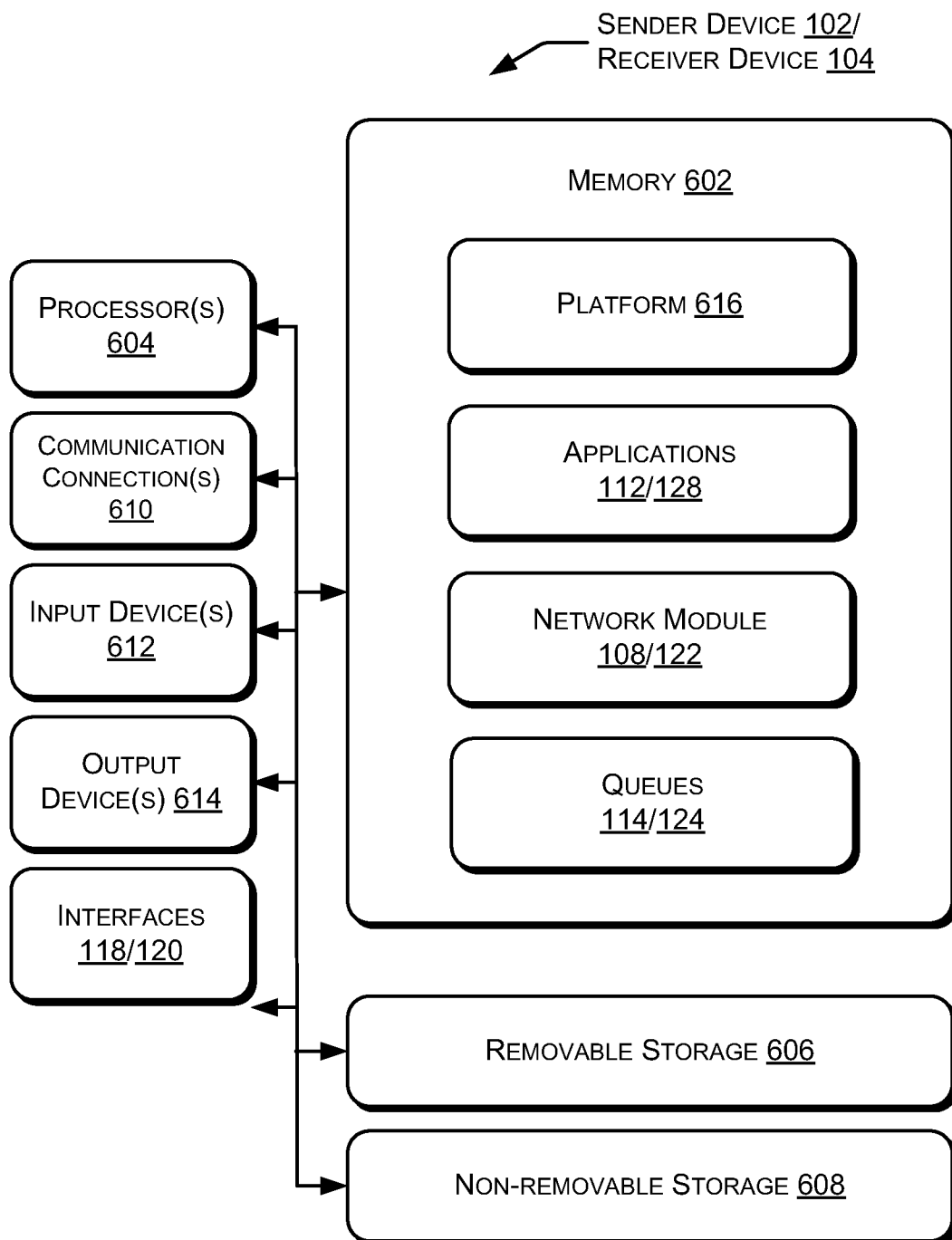


FIG. 4

5/6

**FIG. 5**

6/6

**FIG. 6**