

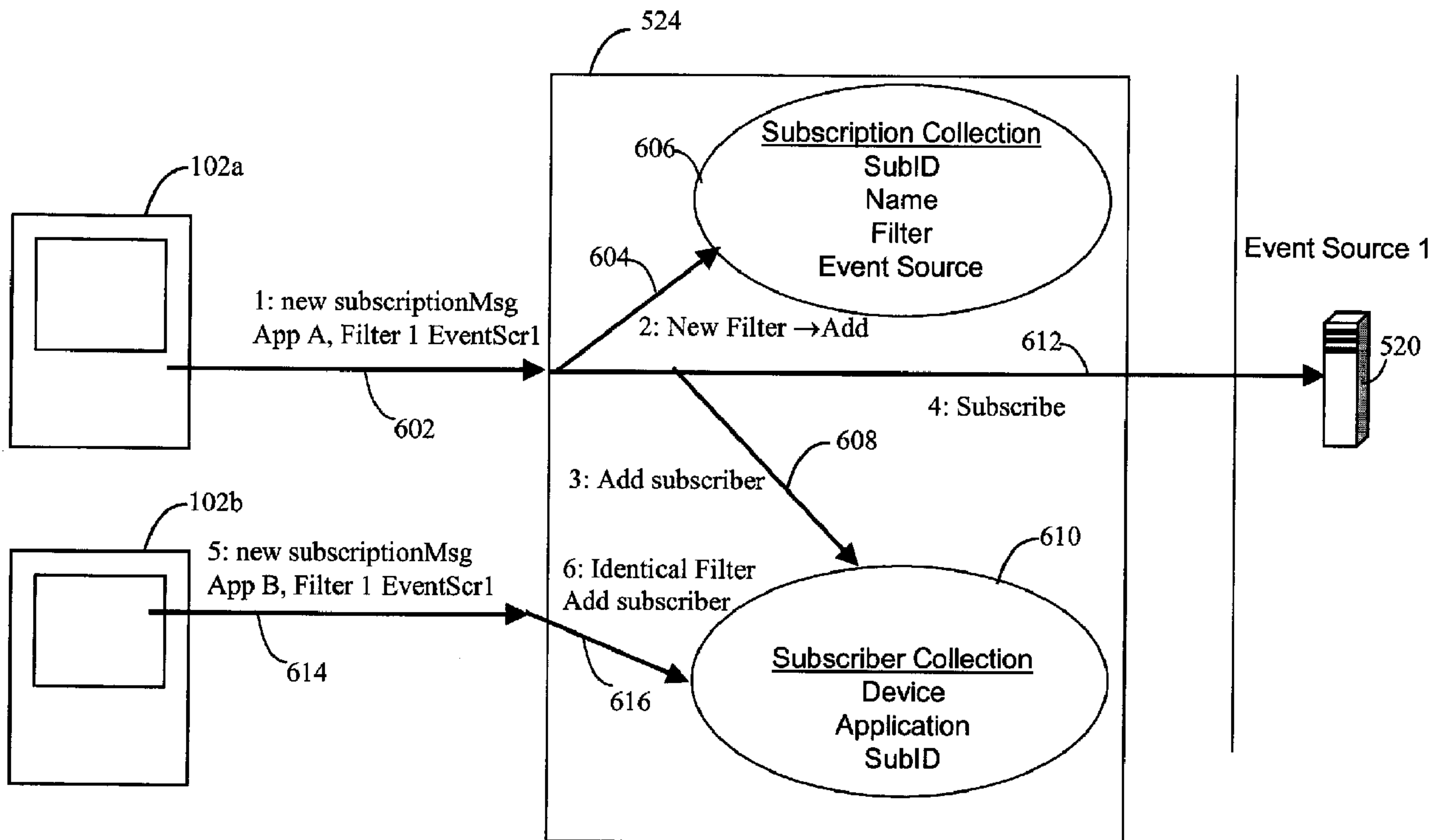


(86) Date de dépôt PCT/PCT Filing Date: 2006/04/18
 (87) Date publication PCT/PCT Publication Date: 2006/10/26
 (45) Date de délivrance/Issue Date: 2013/06/11
 (85) Entrée phase nationale/National Entry: 2007/10/11
 (86) N° demande PCT/PCT Application No.: CA 2006/000609
 (87) N° publication PCT/PCT Publication No.: 2006/111015
 (30) Priorité/Priority: 2005/04/18 (US60/672,035)

(51) Cl.Int./Int.Cl. *H04W 4/08* (2009.01)
 (72) Inventeurs/Inventors:
 FRITSCH, BRINDUSA, CA;
 BIBR, VIERA, CA;
 SHENFIELD, MICHAEL, CA
 (73) Propriétaire/Owner:
 RESEARCH IN MOTION LIMITED, CA
 (74) Agent: GOWLING LAFLEUR HENDERSON LLP

(54) Titre : SYSTEME ET PROCEDE PERMETTANT UN ABONNEMENT DE GROUPE POUR DES APPLICATIONS ASYNCHRONES FONDEES SUR LA TECHNOLOGIE DE DISTRIBUTION SELECTIVE SUR UN DISPOSITIF SANS FIL

(54) Title: SYSTEM AND METHOD FOR ENABLING GROUP SUBSCRIPTION FOR ASYNCHRONOUS PUSH-BASED APPLICATIONS ON A WIRELESS DEVICE



(57) Abrégé/Abstract:

A system and method of enabling group subscription for asynchronous push-based applications on wireless devices are provided. The system comprises an application gateway server (118) including an event sink for communicating between the system and a



(57) **Abrégé(suite)/Abstract(continued):**

device. The application gateway server is configured for communication with a back-end service via an event source and includes a subscription collection (606) for storing subscriptions to event sources and a subscriber collection (610) for identifying the device applications subscribed to an event source using one of the subscriptions stored in the subscription collection. The method comprises the steps of creating a subscription collection for storing subscriptions to event sources and creating a subscriber collection for identifying the device applications subscribed to an event source using one of the subscriptions stored in the subscription collection.

26 MARCH 2007 26.03.07

ABSTRACT

A system and method of enabling group subscription for asynchronous push-based applications on wireless devices are provided. The system comprises an application gateway server (118) including an event sink for communicating between the system and a device.

- 5 The application gateway server is configured for communication with a back-end service via an event source and includes a subscription collection (606) for storing subscriptions to event sources and a subscriber collection (610) for identifying the device applications subscribed to an event source using one of the subscriptions stored in the subscription collection. The method comprises the steps of creating a subscription collection for storing subscriptions to
- 10 event sources and creating a subscriber collection for identifying the device applications subscribed to an event source using one of the subscriptions stored in the subscription collection.

SYSTEM AND METHOD FOR ENABLING GROUP SUBSCRIPTION FOR
ASYNCHRONOUS PUSH-BASED APPLICATIONS ON A WIRELESS DEVICE

5 [0001] The present patent disclosure relates generally to a communication system for providing communication with a plurality of wireless devices and specifically to a system and method for enabling group subscription for asynchronous push-based applications on such devices.

BACKGROUND

10 [0002] Due to the proliferation of wireless networks, there are a continually increasing number of wireless devices in use today. These devices include mobile telephones, personal digital assistants (PDAs) with wireless communication capabilities, two-way pagers and the like. Concurrently with the increase of available wireless devices, software applications running on such devices have increased their utility. For example, the wireless device may include an application that retrieves a weather report for a list of desired cities or an application that allows a user to shop for groceries. These software applications take advantage of the ability to transmit data of the wireless network in order to provide timely and useful services to users, often in addition to voice communication. However, due to a plethora of different types of devices, restricted resources of some devices, and complexity of delivering large amounts of data to the devices, developing software applications remains a difficult and time-consuming task.

25 [0003] Currently, devices are configured to communicate with Web services through Internet-based browsers and/or native applications. Browsers have the advantage of being adaptable to operate on a cross-platform basis for a variety of different devices, but have a disadvantage of requesting pages (screen definitions in HTML) from the Web service, which hinders the persistence of data contained in the screens. A further disadvantage of browsers is that the screens are rendered at runtime, which can be resource-intensive. Applications for browsers are efficient tools for designing platform independent applications. Accordingly, different runtime environments, regardless of the platform, execute the same application.

30 However, since different wireless devices have different capabilities and form factors, the application may not be executed or displayed as desired. Further, browser-based application

often require significant transfer bandwidth to operate efficiently, which may be costly or even unavailable for some wireless devices.

[0004] Current push applications available for mobile devices are proprietary implementations that have not been constructed using a standard specification for the subscription/notification model. Currently there is no known framework to build a mobile application generically, describe it as a 'notification enabled' application by pinpointing the notification components and put it in a context from which it can be available to the device for download and point and click execution.

10 SUMMARY

[0005] Systems and methods disclosed herein provide a communication system for enabling group subscription for asynchronous push-based applications on wireless devices and notification broadcast to obviate or mitigate at least some of the aforementioned disadvantages.

15 **[0006]** In accordance with an aspect of the present patent disclosure there is provided a system for enabling group subscription for asynchronous push-based applications on wireless devices. The system comprises an application registry, an application repository and an application gateway server including an event sink for communicating between the system and a device. The application gateway server is configured for communication with a back-
20 end service via an event source and including a subscription collection and a subscriber collection.

[0007] In accordance with another aspect of the present patent disclosure there is provided a method of enabling group subscription for asynchronous push-based applications on wireless devices. The method comprises the steps of reading a Web service interface, generating a
25 Web service interface therefrom conforming to a notification protocol, creating a notification-based application bundle for the Web service, publishing the notification-based application bundle and creating an event source.

[0008] In accordance with another aspect of the present patent disclosure there is provided a computer-readable medium storing instructions or statements for use in the execution in a
30 computer of a method of enabling group subscription for asynchronous push-based applications on wireless devices. The method comprises the steps of reading a Web service

interface, generating a Web service interface therefrom conforming to a notification protocol, creating a notification-based application bundle for the Web service, publishing the notification-based application bundle and creating an event source.

5 [0009] In accordance with another aspect of the present patent disclosure there is provided a propagated signal carrier carrying signals containing computer-executable instructions that can be read and executed by a computer. The computer-executable instructions are used to execute a method of enabling group subscription for asynchronous push-based applications on wireless devices. The method comprises the steps of reading a Web service interface, generating a Web service interface therefrom conforming to a notification protocol, creating a
10 notification-based application bundle for the Web service, publishing the notification-based application bundle and creating an event source.

BRIEF DESCRIPTION OF THE DRAWINGS

15 [0010] An embodiment of the patent disclosure will now be described by way of example only with reference to the following drawings in which:

Figure 1 shows in a schematic diagram an example of a network facilitating wireless component applications;

Figure 2 shows in a detailed component diagram an example of the application gateway shown in Figure 1;

20 Figure 3 shows in a flow diagram an example of a wireless component application communication model;

Figure 4 shows in an interface diagram an example of a communication sequence for the wireless component application communication model of Figure 3;

25 Figure 5 shows in flow diagram an example of a network as shown in Figure 1 for enabling asynchronous push-based applications on wireless devices in accordance with an embodiment of the present patent disclosure;

Figure 6 illustrates in a flow diagram an example of subscription in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure;

30 Figure 7 illustrates in a flow diagram an example of notification in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure;

Figure 8 illustrates in a flow diagram an example of un-subscription in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure; and

Figure 9 illustrates in a flow diagram an example of subscription ending in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0011] Systems and methods disclosed herein provide a communication system for enabling group subscription for asynchronous push-based applications on wireless devices and notification broadcast to obviate or mitigate at least some of the aforementioned disadvantages.

[0012] In the wireless device world, where resources are scarce and connectivity is intermittent, the user experience is best when information is 'pushed' to the user's device.

The user does not experience network latency since the user is not actively requesting the information, but rather the information is pushed to him as soon as it is available.

Applications that are suitable for push deliver dynamic data, information that is valuable in real-time (examples include email, messaging, weather, news, stock quotes, sports events, and schedules).

[0013] This patent disclosure describes a way to optimize traffic between device and event source for push-enabled applications. A proxy (in wireless component application terms this is the Application Gateway) sends a subscription to the event source on behalf of multiple devices (on some external event like administrator action). This subscription contains criteria that apply for multiple devices, thus reducing wireless traffic (saving each device from the need to subscribe individually with the same criteria). The proxy maintains the list of devices and applications that the subscription applies to. Traffic between event source and proxy is reduced as well, since the event source will only notify the proxy once and then the proxy will notify all subscribers for that subscription.

[0014] The method and system enables broadcast of notifications to multiple devices, while the devices are running same or different notification-based applications. This solution greatly optimizes the traffic between server proxy and event source by reducing the number of subscriptions and notifications initiated between the two parties.

[0015] Advantageously, the concept of a publish-subscribe mechanism is applied to reduce network traffic and associate through subscriptions disparate mobile applications without any apparent connections between them.

5 [0016] For convenience, like numerals in the description refer to like structures in the drawings. Referring to Figure 1, a communication infrastructure is illustrated generally by numeral 100. The communication infrastructure 100 comprises a plurality of wireless devices 102, a communication network 104, an application gateway 106, and a plurality of back-end services 108.

10 [0017] The wireless devices 102 are typically personal digital assistants (PDAs), such as a BlackBerry™ by Research in Motion for example, but may include other devices. Each of the wireless devices 102 includes a runtime environment capable of hosting a plurality of component applications.

15 [0018] Component applications comprise one or more data components, presentation components, and/or message components, which are written in a structured definition language such as Extensible Markup Language (XML). The component applications can further comprise workflow components which contain a series of instructions such as written in a subset of ECMAScript, and can be embedded in the XML in some implementations. Therefore, since the applications are compartmentalized, a common application can be written for multiple devices by providing corresponding presentation components without
20 having to rewrite the other components. Further, large portions of the responsibility of typical applications are transferred to the runtime environment for component application. The details of the component applications are described at the end of this description.

25 [0019] The wireless devices 102 are in communication with the application gateway 106 via the communication network 104. Accordingly, the communication network 104 may include several components such as a wireless network 110, a relay 112, a corporate server 114 and/or a mobile data server (MDS) 116 for relaying data between the wireless devices 102 and the application gateway 106.

30 [0020] The application gateway 106 comprises a gateway server 118 a provisioning server 120 and a discovery server 122. The gateway server 118 acts as a message broker between the runtime environment on the wireless devices 102 and the back-end servers 108. The gateway server 118 is in communication with both the provisioning server 120 and the

26 MARCH 2007 26.03.07

discovery server 122. The gateway server 110 is further in communication with a plurality of the back-end servers 108, such as Web services 108a, database services 108b, as well as other enterprise services 108c, via a suitable link. For example, the gateway server 110 is connected with the Web services 108a and database services 108b via Simple Object Access
5 Protocol (SOAP) and Java Database Connectivity (JDBC) respectively. Other types of back-end servers 108 and their corresponding links will be apparent to a person of ordinary skill in the art.

[0021] Each wireless device 102 is initially provisioned with a service book establishing various protocols and settings, including connectivity information for the corporate server
10 114 and/or the mobile data server 116. These parameters may include a Uniform Resource Locator (URL) for the application gateway server 118 as well as its encryption key. Alternatively, if the wireless device 102 is not initially provisioned with the URL and encryption key, they may be pushed to the wireless device 102 via the mobile data server 116. The mobile device 102 can then connect with the application gateway 106 via the URL
15 of the application gateway server 118.

[0022] Referring to Figure 2, a more detailed view of the application gateway 106 is shown. The application gateway server 118 includes three layers of service; a base services layer 202, an application gateway services layer 204 and an application services layer 206. The application gateway server 118 further includes an administration service 208.

20 [0023] The provisioning server 120 and discovery server 122 provide a provisioning service 210 and a discovery service 212, respectively.

[0024] At the lowest level, the base services layer 202 offers basic, domain-independent system services to other components in higher levels. Thus, for example, all subsystems in the application gateway services layer 204 and the application services layer 206 can utilize
25 and collaborate with the subsystems in the base services layer 202. In the present embodiment, the base services layer 202 includes a utilities subsystem 211, a security subsystem 213, a configuration subsystem 214, and a logging subsystem 216.

The application gateway services layer 204 provides wireless component application domain-specific services. These services provide efficient message transformation and
30 delivery to back-end services 108 and provide wireless device 102 and component application lifecycle management. In the present embodiment, the application gateway

services layer 204 includes a lifecycle subsystem 220, a connector subsystem 222, a messaging subsystem 224, and a transformation subsystem 226.

[0026] The application services layer 206 sits at the top of the architecture and provides external program interfaces and user interfaces using subsystems provided by the lower layers. For example, various applications such as a lifecycle application, a packaging application and a message listening application provide external program interfaces since they communicate primarily with applications on external systems. Similarly, an administration application provides a user interface by providing a user with the ability to access and potentially modify application gateway data and/or parameters.

10 [0027] The administration service 208 is responsible for administrative system messages, administration of the wireless devices 102, runtime administration of the application gateway subsystems, support and display system diagnostics, and administration of default implementations of the provisioning and discovery services.

15 [0028] The messaging listening application provides an interface for receiving messages from the wireless devices 102 as well as external sources and forwarding them to the messaging subsystem. Further, the message listening application typically authenticates that the source of the message is valid.

20 [0029] The security subsystem 212 providing services used by other subsystems for securing communications with the wireless device 102. In order to facilitate secure communications, the security subsystem 212 encrypts and decrypts messages, validates signatures and signs messages.

25 [0030] Referring to Figure 3 there is illustrated in a flow diagram an example of a wireless component application communication model. From a high-level perspective, the overall wireless component application infrastructure 300 includes a wireless component application runtime environment (Device RE) running on the device 102 and a wireless component application gateway (AG) 106 running on the server 118.

30 [0031] The AG 106 serves as a mediator between a wireless component application (sometimes referred to as application in this disclosure) executed by the RE and one or more back-end services 108 with which the application communicates. Often the back-end service is expected to be a Web service 108a using SOAP over HTTP or HTTPS as the transport

protocol. As Web services are the most commonly expected back-end service 108, the term Web service is used interchangeable with back-end service 108 throughout this disclosure. However, it is appreciated that other types of back-end services can also be adapted to the disclosure. Figure 3 exemplifies a synchronous link with a back-end service 108. However, it should be appreciated that the AG 106 can be in communication with back-end services 108 over asynchronous links.

[0032] The wireless component application communication model 300 is based upon an asynchronous messaging paradigm. In this model the application gateway (AG) 106 establishes and mediates the connection between the device 102 and the back-end service(s) 108 to:

1. Achieve greater flexibility in resource management.
2. Provide reliable communication link between device 102 and back-end service 108 to handle situations when wireless coverage is unstable.
3. Efficiently distribute workload between device RE 102 and AG 106.

[0033] Referring to Figure 4 there is illustrated in an interface diagram an example of a communication sequence for the wireless component application communication model of Figure 3. The diagram describes the communications sequence between the device 102 and the back-end service(s) 108:

- Upon receiving a request 402 from the device 102, via redirect request 404 the mobile data server (MDS) 116, AG 106 queues the request 406 and releases the connection to the device.
- Next, the request is retrieved from the queue 408, pre-processed and forwarded 410 to the Web service 108 through a synchronous communication channel.
- Any response from the previous request is processed by the application gateway (AG) 106 and a response message is sent asynchronously 412 and 414 back to the device.

[0034] Referring to Figure 5 there is illustrated in a flow diagram an example of a network as shown in Figure 1 for enabling asynchronous push-based applications on wireless devices in accordance with an embodiment of the present patent disclosure. In Figure 5, at Step 1, an existing Web service 500 has its Web service interface read 502 by a user of a (rapid application development) RAD tool 504. Then, at Step 2, the RAD tool user 504 generates

506 a Web service interface 508 conforming to a notification protocol. At Step 3, the RAD tool user 504 creates 510 a notification-based application bundle 512, then at Step 4 publishes 514 the notification-based application bundle 512 to a repository 516 associated with the application gateway 106. At the same time the RAD tool user 504 publishes a registry entry in a registry, the registry entry refers to the application bundle and where it is located. At Step 5 the RAD tool user 504 creates 518 an event source 520. If the mobile application developer (RAD user) has update rights to the back-end server that has the data and is to act as the event source 520, the event source is implemented according to the protocol supported by the application gateway 106. Otherwise, a proxy is created on the application gateway 106 to implement the required subscription/notification protocol. In this implementation of event source 520, the back-end infrastructure is polled for data and to detect data changes and triggers notifications when changes are found.

[0035] The following definitions are used with the detailed description:

15	<i>Notification</i>	message carrying data of interest to a specific user
	<i>Subscription</i>	message denoting an event originated by a user that expresses his interest to be notified when specific events occur
20	<i>Server Proxy</i>	server enabling a mobile application to communicate with back-end application servers and have access to data
25	<i>Notification-based Application Bundle</i>	XML application built with RAD tools that has been built for a framework that supports a notification protocol (for example, WS-Eventing). The application is accompanied by additional information to help the mobile framework map the application messages to notification protocol components.
30	<i>WS-Eventing</i>	Web Services Eventing standard for notification-based Web services (ftp://www6.software.ibm.com/software/developer/library/ws-eventing/WS-Eventing.pdf)
	<i>Event Source</i>	Back-end server accepting subscriptions, delivering notifications based on them and terminating subscriptions
35	<i>Event Sink</i>	Server accepting notifications from event source
	<i>Filter</i>	Expression that describes the type of data the user is interested in; it may have any format as long as the event sink and the event source both understand it

[0036] Referring to Figure 6 there is illustrated in a flow diagram an example of subscription in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure. Two mobile devices 102a and 102b communicate separately with a server proxy 524 to subscribe to services provided by the event source 520. At Step 1 the mobile device 102a sends a new subscription message 602 to the server proxy 524.

[0037] The server proxy 524 determines whether a subscription message with the same filter content and destination (event source endpoint) has been previously received and stored in the subscription collection. If a subscription matching this criteria is not found as shown at Step 2, a new filter is added 604 to a subscription collection 606 and at Step 3 a subscriber is added 608 to a subscriber collection 610. At Step 4, a subscribe message is sent 612 to the event source 520. If the subscription matching the criteria is found (yes) as shown at Steps 5 and 6 only information for a new subscriber 102b is preserved at 614 and 616 in the subscriber collection.

[0038] Subscribe Use Case: Device1 (102a) sends 602 the subscription SA with Filter 1. At the server proxy 524, Filter1 is assembled in accordance with the filter expression used by the event source 520 (or protocol) to form for example, "P1"+"P2"+"P3". This subscription is sent to event source 1 (520). At some other time, Device2 (102b) sends 614 another subscription for another application B that has the same Filter1 and is intended for the same event source 520. Since these two subscriptions have identical parameters, bandwidth savings can be realized by only sending the subscription once 612 to the event source 520. In other words, if any identical subscriptions occur, the proxy only preserves 616 the identity of the subscriber (in this case device and application).

[0039] The following are examples of two devices subscribing to two different applications that have common filter parameter values:

Device1

Application A

Subscription SA

Filter1

P1

P2

P3

Device2

Application B

Subscription SB

5 Filter1

P1

P2

P3

[0040] The following other scenarios can occur where the same subscription is applied:

10 (Filter1, E1), D1, App A -- Subscription sent from Device D1 (102a), application A, with Filter1 for event source E1 (520)

(Filter1, E1), D2, App A

(Filter1, E1), D1, App B

(Filter1, E1), D3, App B

15 [0041] The server proxy 524 maintains a subscription collection 606 including the following:

- Subscription Internal ID - The Subscription internal ID is generated by the proxy server 524 and sent to the event source 520; it is used by the event source 520 for notifications and subscription end.
- Subscription External ID (or Cancellation ID) – The Subscription external ID is sent by the event source 520 in subscription response; it is used by the proxy server 524 to send an unsubscribe message to the event source 520 (i.e., to cancel the subscription).
- Subscription Name is unique across an application or within a subscription type. The Subscription name is sent by the device 102.
- Filter expression is sent by device 102; expression is assembled by the server proxy 524.
- Event Source Endpoint – the server proxy 524 loads connectivity information for the event source endpoint when an application is installed on device 102.

30 [0042] The server proxy 524 maintains a subscriber collection 610 with a reference to the subscription ID including the following:

- Device
- Application

- Expiry (optional) - Expiry: date when subscription for this subscriber will expire; if expiry is specified, the proxy has to consider it and delete subscribers after this time is elapsed
- Subscription Internal ID is reference to subscription collection
- 5 • Subscription name is the unique identifier on the device; it is used for notification, unsubscribe, subscription end.

[0043] Referring to Figure 7 there is illustrated in a flow diagram an example of notification in a network configured for group subscription and notification broadcast to notification-based applications in accordance with an embodiment of the present patent disclosure.

10 [0044] When notification 702 for the subscription arrives at Step 1, the server proxy 524 looks up 704 the subscription at Step 2 and finds 706 all the subscribers for the 'service' at Step 3 and broadcasts 708 the message to all of them 102 at Step 4. The subscriber is represented by a (device, application) pair. The server proxy 524 builds and sends this message to different mobile applications hosted by the runtime environment on the mobile
15 device.

[0045] Referring to Figure 8 there is illustrated in a flow diagram and example of un-subscription in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure. An unsubscribe 802 is initiated by the subscriber 102a, at Step 1. The proxy server removes 804
20 the subscriber from the subscriber collection 610, at Step 2. Only if there are no longer any instances in the subscriber collection, the subscription identified by the subscription internal ID is removed 806 from the subscription collection 606, at Step 3 and an unsubscribe message is sent 808 to the event source 520, at Step 4.

[0046] Referring to Figure 9 there is illustrated in a flow diagram an example of subscription
25 ending in a network configured for group subscription to notification-based applications in accordance with an embodiment of the present patent disclosure. The event source 520 initiates Subscription End 902, at Step 1. When received, the proxy server 524 sends 904 a subscription End message to all subscribers 102 for this subscription ID, at Step 2. Note that the server proxy 524 builds and sends this message to different applications hosted by the
30 same runtime environment. The proxy server 524 empties 906 the subscriber collection 610,

at Step 3 and removes 908 the subscription with this ID from the subscription collection 606, at Step 4. The subscriber instance is represented by a (device, application) pair.

[0047] The system and methods according to the present patent disclosure may be implemented by any hardware, software or a combination of hardware and software having
5 the above described functions. The software code, either in its entirety or a part thereof, may be stored in a computer-readable memory. Further, a computer data signal representing the software code which may be embedded in a carrier wave may be transmitted via a communication network. Such a computer-readable memory and a computer data signal are also within the scope of the present patent disclosure, as well as the hardware, software and
10 the combination thereof.

[0048] While particular embodiments of the present patent disclosure have been shown and described, changes and modifications may be made to such embodiments without departing from the true scope of the patent disclosure.

WHAT IS CLAIMED IS:

1. An application gateway server for providing a proxy for a group subscription , the application gateway server comprising:

a subscription collection for storing the group subscription from a plurality of wireless devices to notification-based applications provided by a back-end server, the back-end server hosting an event source and accepting a subscription sent from the application gateway server, the subscription collection including a plurality of filters describing a plurality of subscriptions from the plurality of wireless devices, each of the plurality of filters representing a unique filter expression; and

a subscriber collection for identifying each of the plurality of wireless devices and applications of the each of the plurality of wireless devices, the applications being subscribed to the event source accepting the group subscription;

wherein the application gateway server receives a first subscription request having a filter content and a destination, and determines if a subscription message having the filter content and the destination has been previously received by comparing the filter content and the destination with the plurality of filters in the subscription collection;

wherein a new filter is added to the subscription collection if the subscription message has not been previously received;

wherein a new subscriber is added to the subscriber collection if the subscription message has been previously received.

2. The application gateway server as claimed in claim 1, wherein the application gateway server is part of a system comprising an application registry for hosting a wireless application identifier; and an application repository for hosting a wireless application.

3. The application gateway server as claimed in claim 1, wherein the subscription collection includes:

a subscription Internal ID;

a subscription External ID;

a subscription name;

a filter expression; and

an event source endpoint.

4. The application gateway server as claimed in claim 1, wherein the subscriber collection includes:

a subscription Internal ID;

a subscribed device;

a subscribed application; and

a subscription name.

5. The application gateway server as claimed in claim 1, wherein the

subscribed device and the subscribed application form pairs.

6. The application gateway server as claimed in claim 1, wherein a subscription is removed from the subscription collection when there are no subscribers in the subscriber collection associated with the subscription.

7. The application gateway server as claimed in claim 2, wherein the application registry includes an entry for a component application in the application repository.

8. The application gateway server as claimed in claim 2, wherein the application repository includes a component application for accessing the back-end service via the event source.

9. The application gateway server as claimed in claim 2, wherein the system enables notification message broadcast to multiple subscribers providing data content.

10. A method of providing a group subscription, the method comprising:

creating a subscription collection for storing the group subscription from a plurality of wireless devices to notification-based applications provided by a

back-end server, the back-end server hosting an event source and accepting a subscription provided by an application gateway server, the subscription collection including a plurality of filters describing a plurality of subscriptions from the plurality of wireless devices, each of the plurality of filters representing a unique filter expression; creating a subscriber collection for identifying each of the plurality of wireless devices and applications of the each of the plurality of wireless devices, the applications being subscribed to the event source accepting the group subscription;

receiving a first subscription request having a filter content and a destination;

determining if a subscription message having the filter content and the destination has been previously received by comparing the filter content and the destination with the plurality of filters in the subscription collection; and

adding a new filter to the subscription collection if the subscription message has not been previously received; and

adding a new subscriber to the subscriber collection if the subscription message has been previously received.

11. The method as claimed in claim 10, wherein the subscription collection and the subscriber collection are on an application gateway server.

12. The method as claimed in claim 10, wherein the subscriber collection contains a plurality of (device, application) pairs.

13. The method as claimed in claim 10, further comprising the steps of:

receiving a notification having a filter content and a subscription internal ID from the event source;

comparing the filter content and the subscription internal ID with subscriber information in the subscriber collection; and

sending the notification to subscribers that have the filter content and the subscription Internal ID.

14. A computer-readable medium storing instructions or statements for use in the execution in a computer of a method of providing a group subscription , the method comprising the steps of:

creating a subscription collection for storing the group subscription from a plurality of wireless devices to notification-based applications provided by a back-end server, the back-end server hosting an event source and accepting a subscription provided by an application gateway server, the subscription collection including a plurality of filters describing a plurality of subscriptions from the plurality of wireless devices, each of the plurality of filters representing a unique filter expression;

creating a subscriber collection for identifying each of the plurality of wireless devices and applications of the each of the plurality of wireless devices, the applications being subscribed to the event source accepting the group subscription;

receiving a first subscription request having a filter content and a destination;

determining if a subscription message having the filter content and the destination has been previously received by comparing the filter content and the destination with the plurality of filters in the subscription collection; and

adding a new filter to the subscription collection if the subscription message has not been previously received; and

adding a new subscriber to the subscriber collection if the subscription message has been previously received.

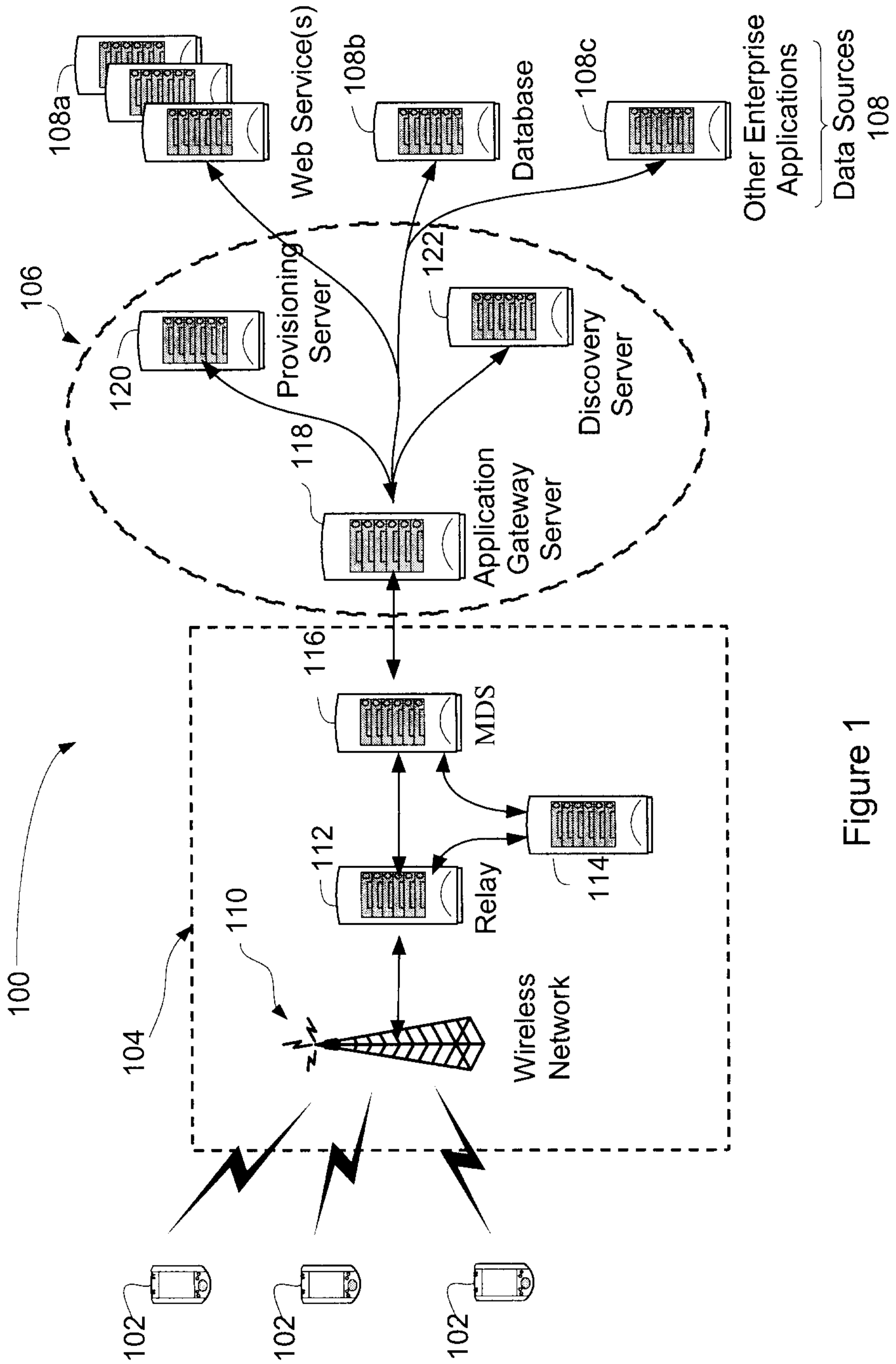


Figure 1

106

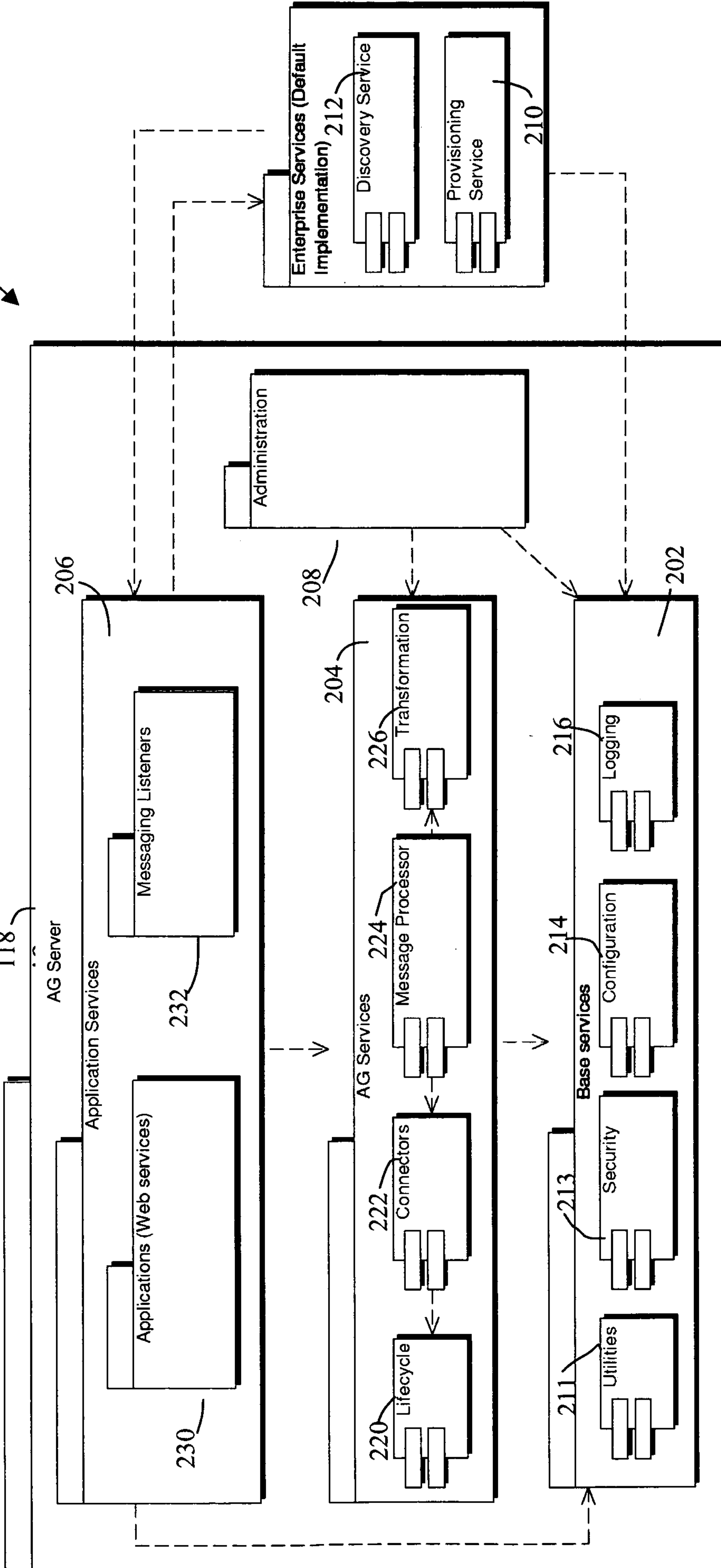


Figure 2

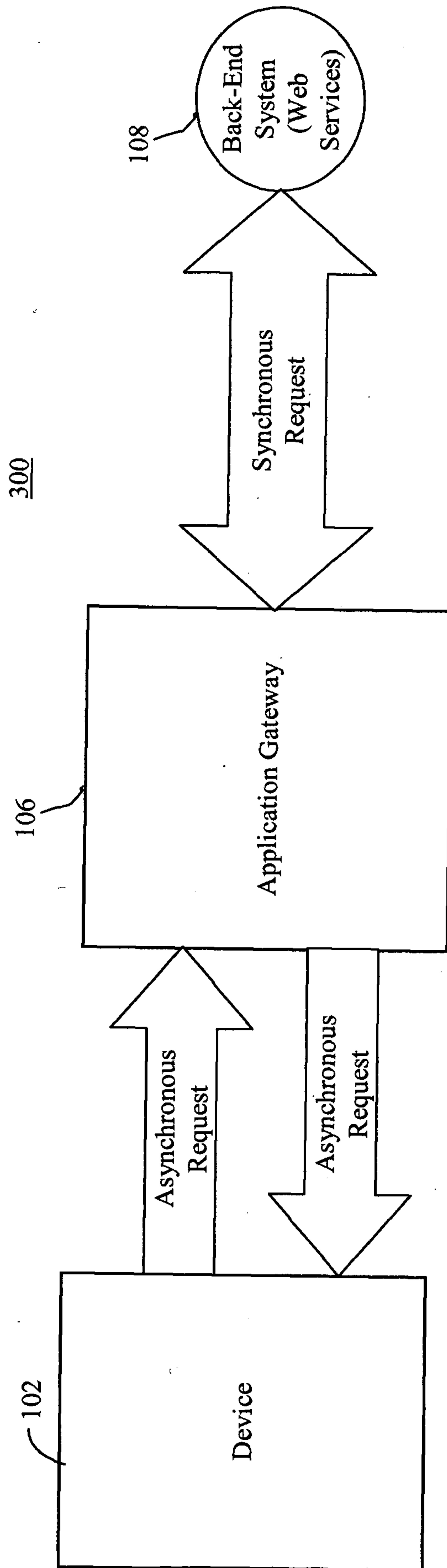
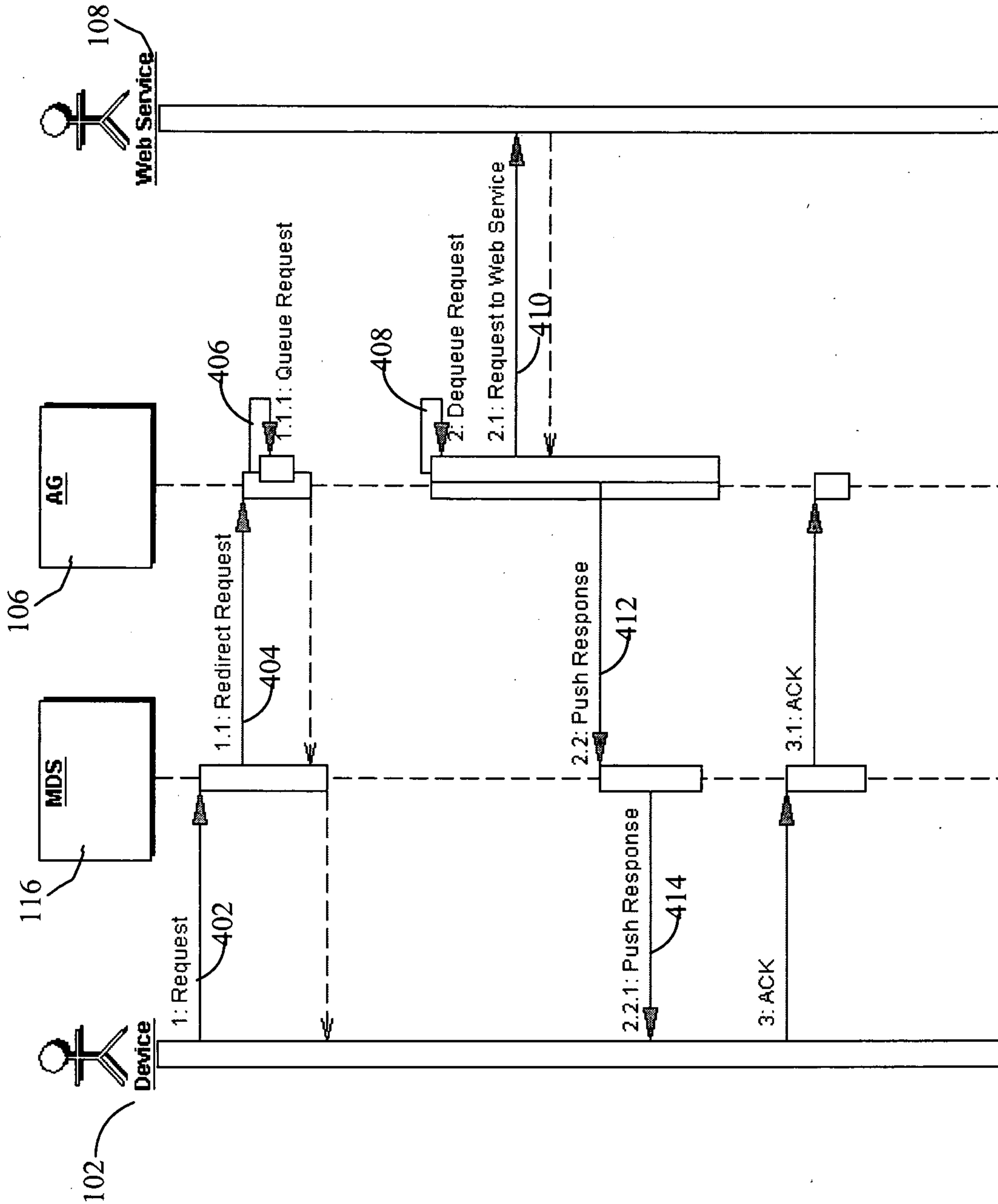


Figure 3

26 MARCH 2007 26.03.07

Figure 4



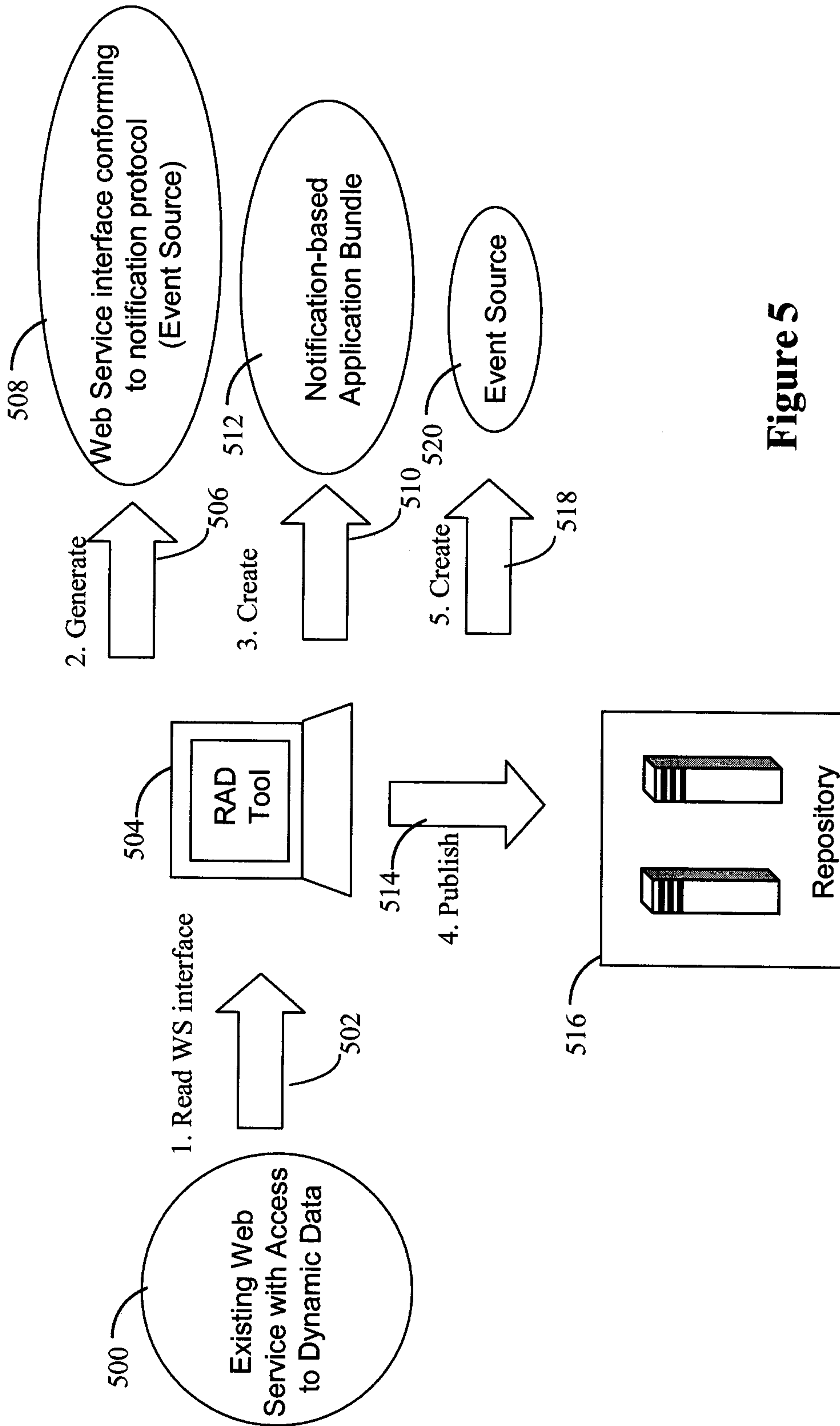


Figure 5

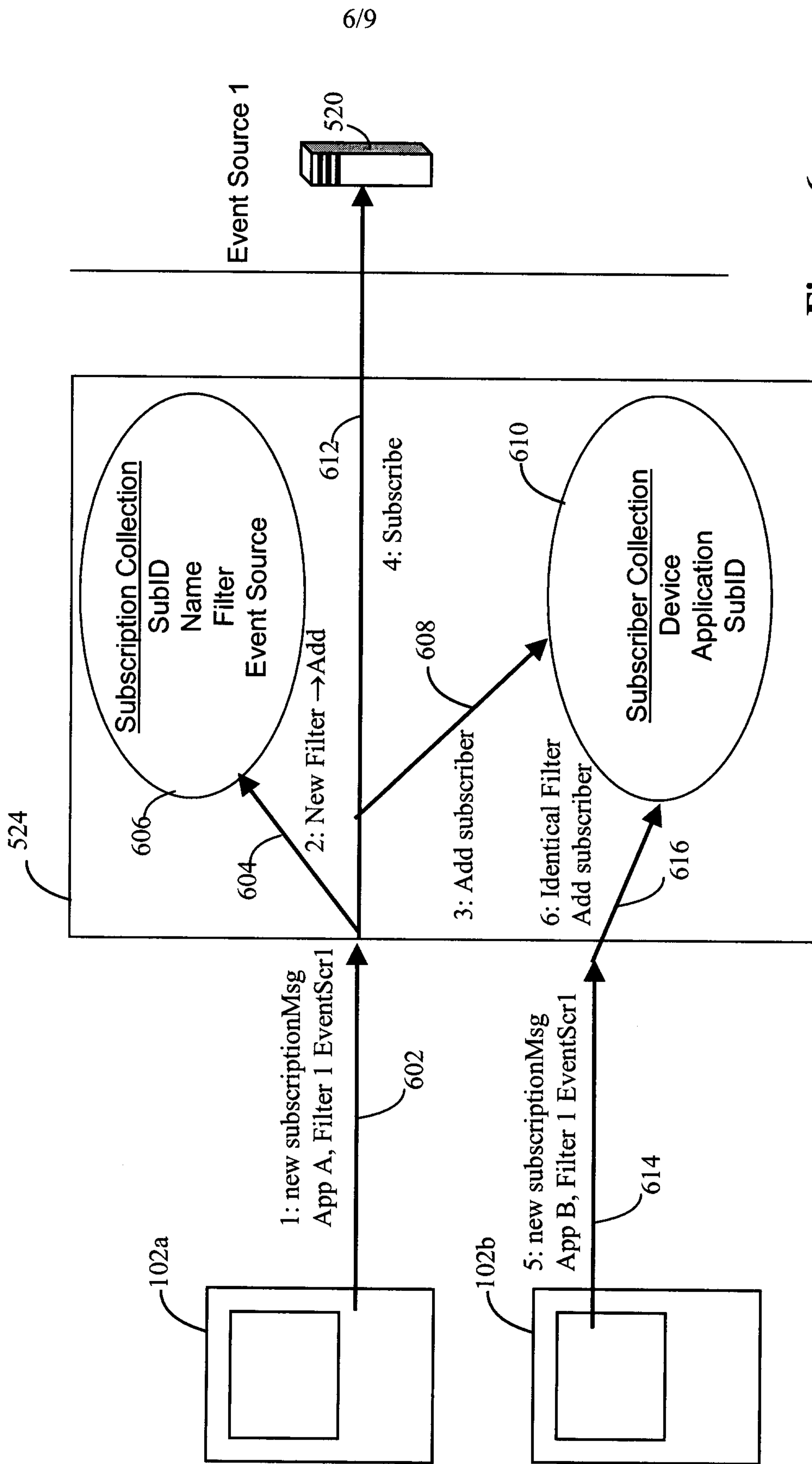


Figure 6

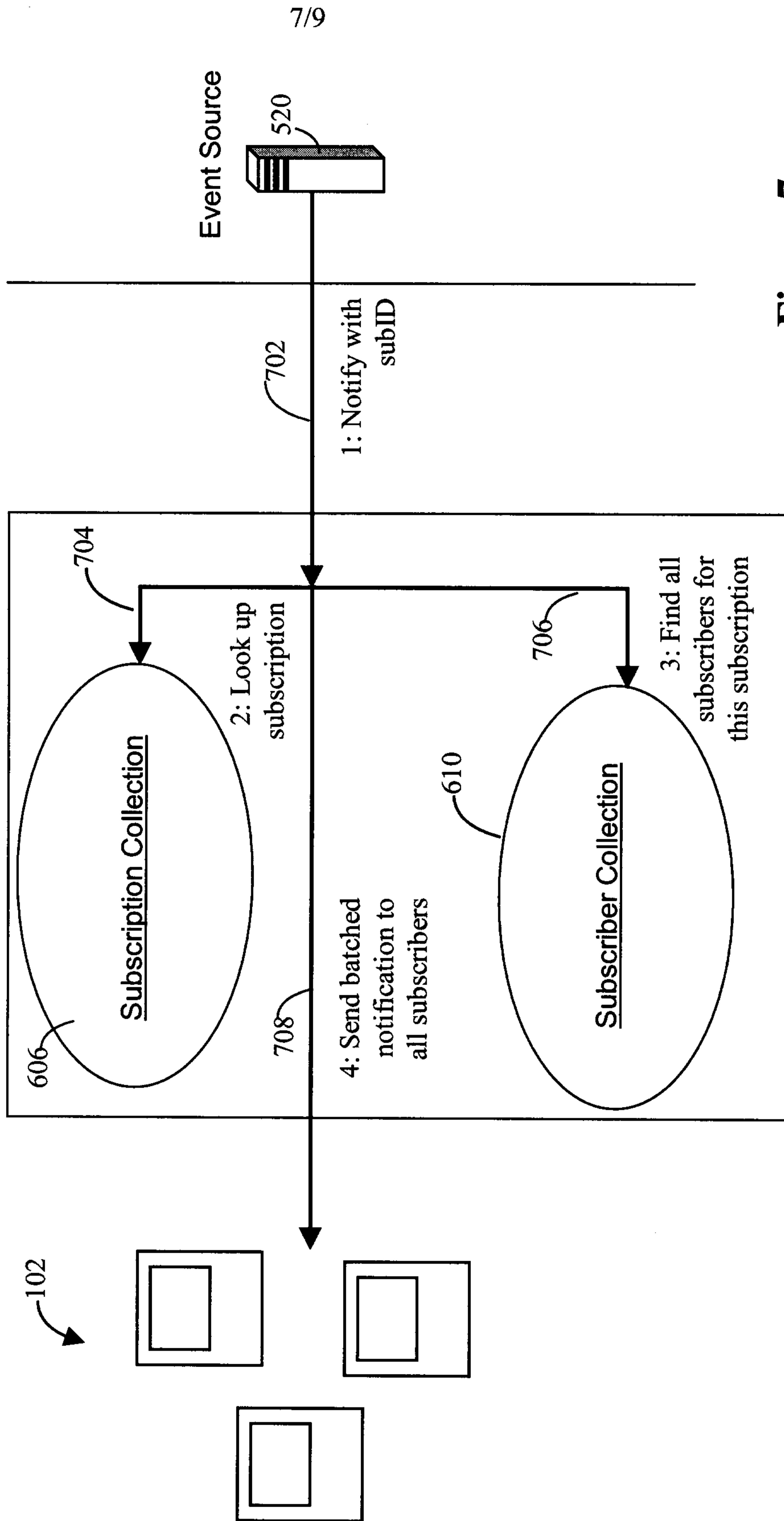


Figure 7

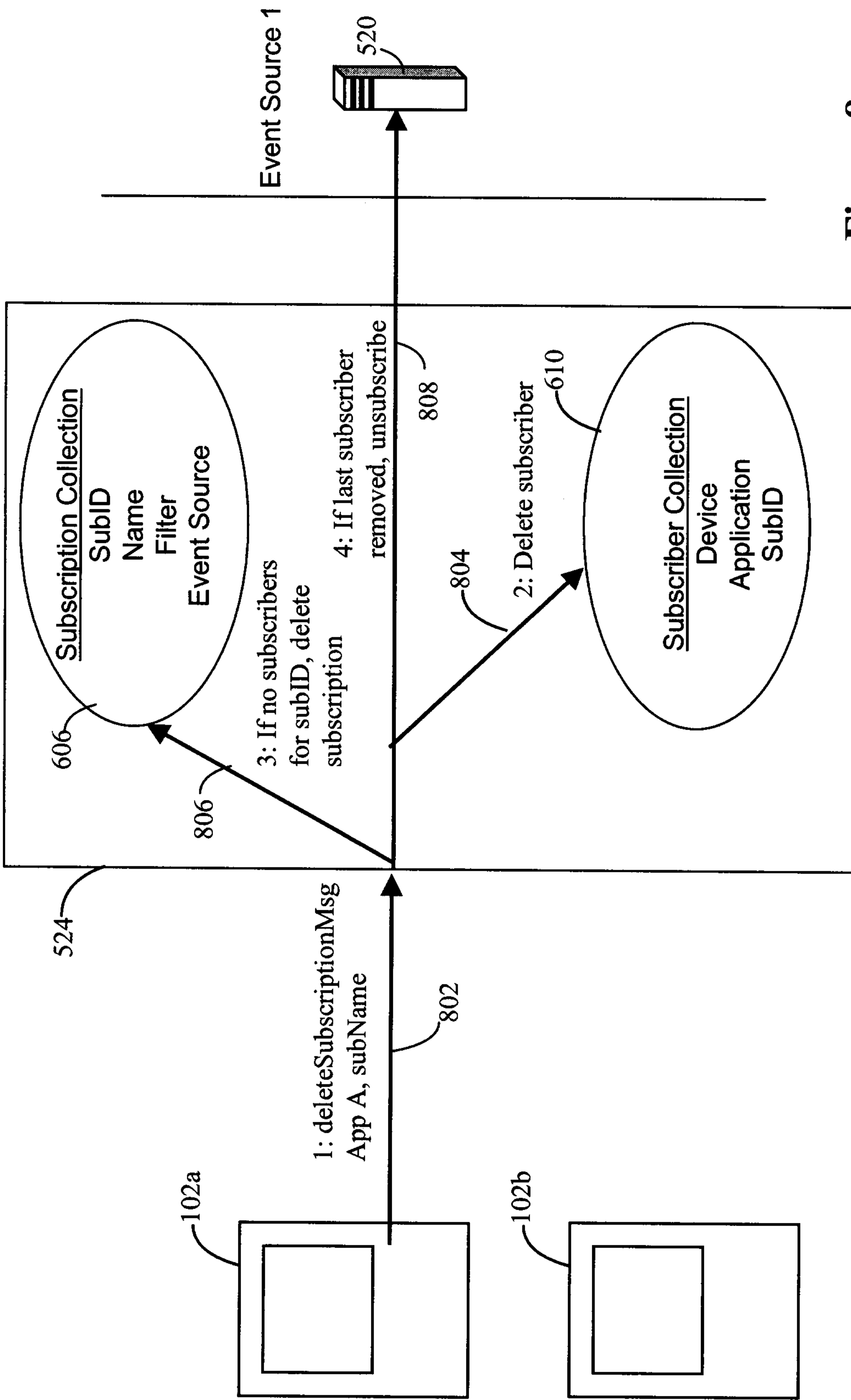


Figure 8

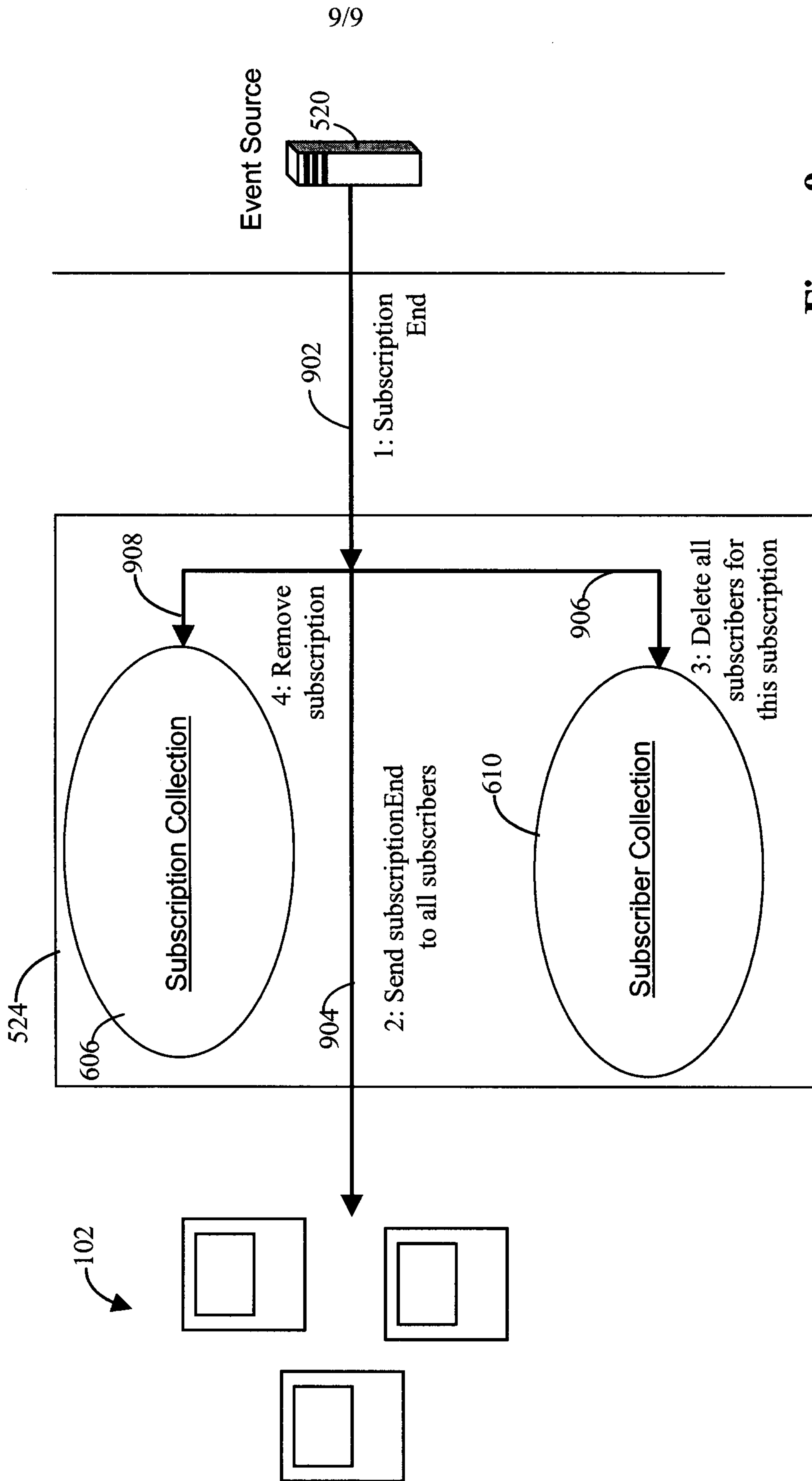


Figure 9

