Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2544032 C 2013/01/08

(11)(21) 2 544 032

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(22) Date de dépôt/Filing Date: 2006/04/18

(41) Mise à la disp. pub./Open to Public Insp.: 2006/10/18

(45) Date de délivrance/Issue Date: 2013/01/08

(30) Priorité/Priority: 2005/04/18 (EP05103043.5)

(51) Cl.Int./Int.Cl. *H04L 29/02* (2006.01),
H04W 28/06 (2009.01)

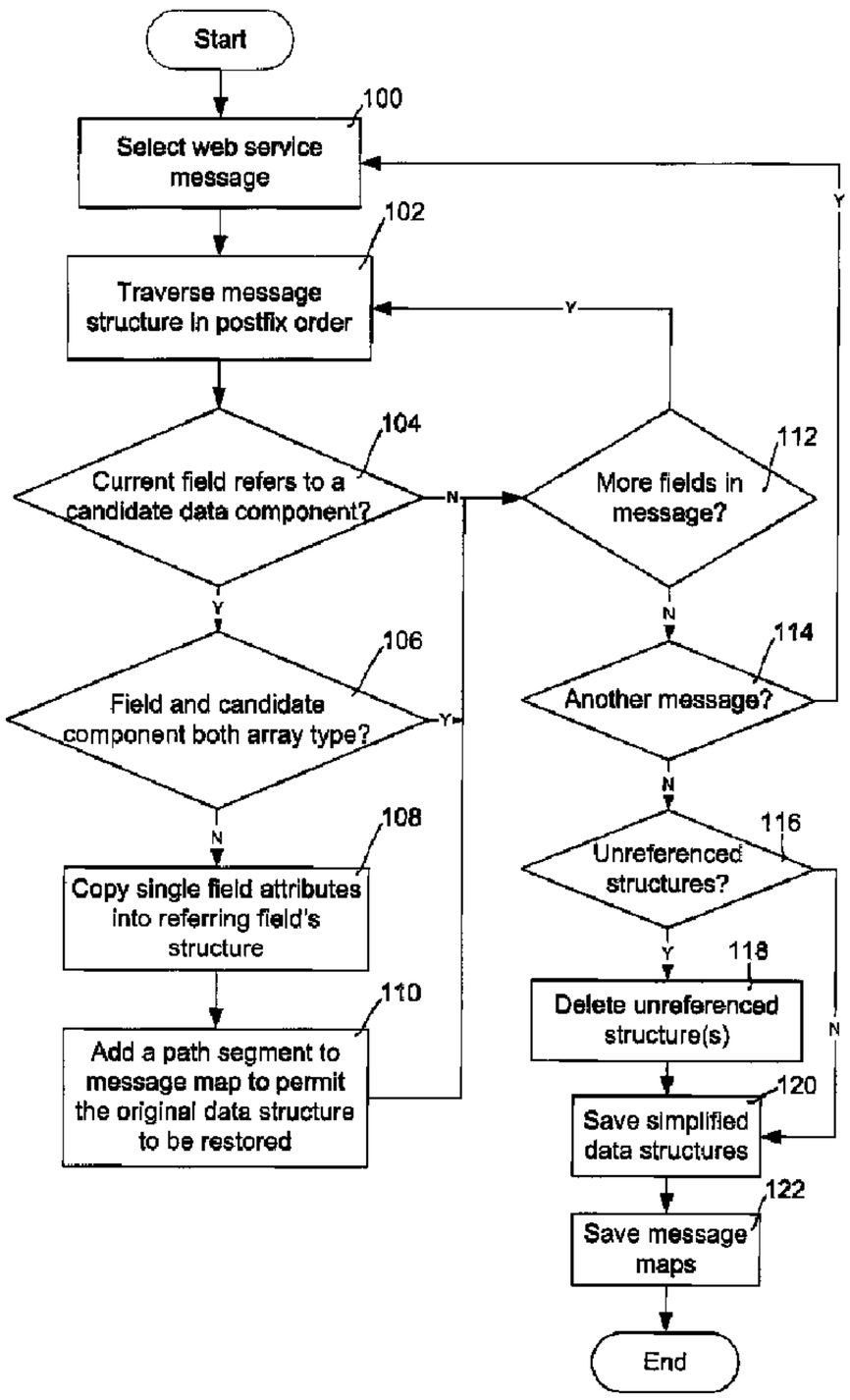
(72) Inventeurs/Inventors:
DEBRUIN, DAVID, CA;
BIBR, VIERA, CA;
FRITSCH, BRINDUSA L., CA;
SHENFIELD, MICHAEL, CA;
GORING, BRYAN R., CA

(73) Propriétaire/Owner:
RESEARCH IN MOTION LIMITED, CA

(74) Agent: SMART & BIGGAR

(54) Titre : SYSTEME ET METHODE PERMETTANT DE SIMPLIFIER LA STRUCTURE DE DONNEES DE COMMUNICATIONS SANS FIL

(54) Title: SYSTEM AND METHOD FOR SIMPLIFICATION OF DATA STRUCTURE IN WIRELESS COMMUNICATIONS



(57) Abrégé/Abstract:
System and method for data simplification in wireless communications are enabled by an application developer toolkit used to convert complex service messages to simplified data messages for storing information associated with a remote service that utilizes

CA 2544032 C 2013/01/08

(11)(21) 2 544 032

(13) C

(57) Abrégé(suite)/Abstract(continued):
complex data structures for conveying the information to devices that access the remote service. The application developer toolkit also outputs maps for converting the complex service messages to simplified wireless messages in a format of a simplified data structure, and maps simplified wireless messages to complex service messages. A proxy located in a communications path between a wireless device and the remote service uses the maps to convert complex service messages to simplified service messages, and vice versa.

ABSTRACT OF THE DISCLOSURE

System and method for data simplification in wireless communications are enabled by an application developer toolkit used to convert complex service messages to simplified data messages for storing information associated with a remote service that utilizes complex data structures for conveying the information to devices that access the remote service. The application developer toolkit also outputs maps for converting the complex service messages to simplified wireless messages in a format of a simplified data structure, and maps simplified wireless messages to complex service messages. A proxy located in a communications path between a wireless device and the remote service uses the maps to convert complex service messages to simplified service messages, and vice versa.

SYSTEM AND METHOD FOR SIMPLIFICATION OF DATA STRUCTURE IN WIRELESS COMMUNICATIONS

This application relates generally to wireless communications and wireless communications devices and, in particular, to a method and apparatus for simplification of data structure in wireless communications.

The acceptance of wireless devices as a preferred personal communications medium has created a growing demand for such devices. Users of such devices also increasingly expect more functionality and a broader range of services to be made available through such devices. Not only is there a demand for more functionality, there is also a demand for faster response times and more efficient access to remote services.

A major challenge is faced in exposing wireless devices to complex data sources, such as web services, due to the size and complexity of the data structures communicated from such sources. In wired networks and devices where resources and efficiency are not a significant concern, it is permissible to transmit, process and store large and complex data structures.

Complex data structures containing many levels of nesting introduce a significant memory overhead on wireless devices. This impacts performance when accessing such data in a memory store.

Fig. 1 is a block diagram of a prior art network in which wireless device users 10a, 10b operate wireless devices to send web service request messages via a public domain carrier 12 to an application gateway 14. The application gateway 14 forwards the web service request messages through the internet 16 to an appropriate web service 18a, 18b. The messages are processed by the appropriate web service 18a, 18b and returned through the internet 16 to the application gateway 14. The public domain carrier 12 forwards the response messages to the wireless device 10a, 10b which processes the response and displays response content to the wireless device users 10a, 10b.

FIG. 2 is a schematic diagram of an exemplary web service data structure 28 in accordance with the prior art. FIG. 2 schematically represents an exemplary complex data structure 20 named "GetSenatorInfoResponse." The complex data structure 20 contains a referring field 22 named "GetSenatorInfoResult" having a field type of "ArrayofSenatorInfo". The referring field 22 refers to a data component 24 that contains a single field "SenatorInfo" having a field type of array. Likewise, the referring field 26, refers to a candidate data component 28 "LocalAddress" having a field type of "ArrayofLocalAddress". As will be appreciated by those skilled in the art, this complex data structure requires significant computing resources to store the structure in memory or retrieve it from memory at a rate that is acceptable to most users. Since wireless devices 10a, 10b also have limited available data transfer capacity commonly referred to as "bandwidth", they generally cannot send and receive complex data structures at a rate that is considered acceptable to most users.

Consequently, there exists a need for a system and method that provides data and message optimization in order to ensure that wireless device response time and wireless messaging efficiency can keep up with demand, and that wireless device users remain satisfied with the performance of their wireless devices.

SUMMARY

It is desirable to provide a system and method for simplification of data structures used in wireless devices that promotes efficient data storage, efficient data access and retrieval, and transparent, efficient wireless messaging between wireless devices and remote services, such as worldwide web services.

In accordance with a first aspect of the disclosure there is provided a system for simplification of a data structure in wireless communications messages, comprising: a computer comprising a developer toolkit, the developer toolkit comprising computer executable code for accepting as input a service message containing complex data structures with multiple levels of nesting and outputting a simplified service message containing simplified data structures which contain fewer levels of nesting, and further outputting a message map for transforming the simplified service message to the complex service message, and vice versa; wherein the computer executable code is arranged to add a path segment to the message map to permit the input service message to be restored from the simplified service message.

In accordance with a further aspect of the disclosure there is provided a method for simplification of a data structure in wireless communications messages, comprising the steps of: provisioning the application developer toolkit with computer executable code for mapping complex service messages used by the remote service to simplified wireless messages used by the application, and the simplified wireless messages to the complex service messages; and provisioning a proxy in a communications path between the wireless device and the remote service with a message map comprising the mapped complex service messages, to permit the proxy to convert the complex service messages to the simplified wireless messages and the simplified wireless messages to the complex service messages, to enable wireless communications between the wireless device and the remote service;

wherein the method further comprises provisioning an application developer toolkit with computer executable code for adding a path segment to the message map to permit the service message to be restored from the simplified data message.

In accordance with a further aspect of the disclosure there is provided a system for simplification of data structure in wireless communications messages, the system comprising: a computing device comprising a processor and a computer-readable medium storing computer executable code for effecting a developer toolkit, the computer executable code, when executed by the processor, causing the device to: accept as input a complex service message containing a complex data structure with multiple levels of nesting; remove at least one field of the complex data structure to form a simplified data structure with fewer levels of nesting; output a simplified service message which contains the simplified data structure with fewer levels of nesting; and output a message map for transforming the simplified service message to the complex service message, and vice versa, wherein the removing of the at least one field comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: determining whether the current field refers to a candidate data component containing a single field; and if the determining is positive, copying a single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type, and wherein the outputting of the message map comprises adding, to the message map, a path segment identifying an original location of the removed at least one field for use in the transforming of the simplified service message to the complex service message.

In accordance with a further aspect of the disclosure there is provided

a method for simplification of wireless communications messages, comprising: provisioning an application developer toolkit with computer executable code that, upon execution by a processor: accepts as input a complex service message having a complex data structure, the complex data structure for communicating and storing information associated with a remote service that utilizes service messages for communicating information to wireless devices that access the remote service; removes at least one field of the complex data structure to form a simplified data structure with fewer levels of nesting; outputs a simplified service message with the simplified data structure; and outputs a message map for transforming the simplified service message to the complex service message, and vice versa, wherein the removing of the at least one field of the complex data structure comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: determining whether the current field refers to a candidate data component containing a single field; and if the determining is positive, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type, and wherein the outputting of the message map comprises adding, to the message map, a path segment identifying an original location of the removed at least one field for use in the transforming of the simplified service message to the complex service message.

In accordance with a further aspect of the disclosure there is provided a computer-readable medium storing computer executable code for effecting a developer toolkit for facilitating simplification of data structure in wireless communications messages, the computer executable code, when executed by a processor of a computing device of the system, causing the computing device to: accept as input a complex service message containing a complex data structure with multiple levels of nesting; remove at least one field of the complex data structure to form a simplified data structure with fewer levels of nesting; and

output a message map for transforming the simplified service message to the complex service message, and vice versa, wherein the removing of the at least one field comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: upon determining that the current field refers to a candidate data component containing a single field, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type, and wherein the outputting of the message map comprises adding, to the message map, a path segment identifying an original location of the removed at least one field for use in the transforming of the simplified service message to the complex service message.

In accordance with a further aspect of the disclosure there is provided a computing device in a system for simplification of data structure in wireless communications messages, the computing device comprising a processor and memory storing computer executable code for effecting a developer toolkit, the computer-executable code, when executed by the processor, causing the computing device to: accept as input a complex service message containing a complex data structure with multiple levels of nesting; remove at least one field of the complex data structure to form a simplified data structure with fewer levels of nesting; and output a message map for transforming the simplified service message to the complex service message, and vice versa, wherein the removing of the at least one field comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: upon determining that the current field refers to a candidate data component containing a single field, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type, and wherein the outputting of the message map comprises adding, to the message map, a path segment identifying an original location of the removed at

least one field for use in the transforming of the simplified service message to the complex service message.

BRIEF DESCRIPTION OF THE DRAWINGS

Further features and advantages of the present disclosure will become apparent from the following detailed description, taken in combination with the appended drawings, in which:

FIG. 1 is a block diagram of a prior art wireless network system;

FIG. 2 is a schematic diagram of an exemplary prior art web service data structure;

FIG. 3 is a schematic diagram illustrating a process for creating and enabling a system in accordance with the disclosure;

FIG. 4 is a block diagram of a proxy in accordance with the disclosure;

FIG. 5 is a block diagram of a wireless device in accordance with the disclosure;

FIG. 6 is a high level overview of messaging between a wireless device and a remote service in a system in accordance with the disclosure;

FIG. 7 is a flow chart of an algorithm in accordance with the disclosure for flattening complex data structures; and

FIG. 8 is a schematic representation of the data structure shown in FIG. 2 after it has been flattened using the algorithm shown in FIG. 7.

It will be noted that throughout the appended drawings, like features are identified by like reference numerals.

DETAILS

The disclosure provides a system and method for the simplification of data structures used for wireless communications. Wireless devices are provisioned with user applications for accessing remote services, such as World Wide Web services. The user applications are conditioned to use a simplified data structure for storing information associated with a remote service that utilizes complex data structures for conveying the information to devices that access the remote service. An algorithm processes the complex data structures to simplify them by removing structure not utilized in the wireless device.

System Overview

Fig. 3 is a schematic diagram illustrating a process for creating and enabling a system in accordance with the disclosure.

A remote service accessed by the wireless devices 10a, 10b shown in Fig. 1, such as web service 20 uses complex message structures to communicate information to users who access the remote service. The web service 20 likewise uses complex data structures for data storage and data retrieval. A wireless application developer 22 uses a specification of the web service 20 and an application developer toolkit to create wireless device applications 24 and message maps 26 for enabling a system in accordance with the disclosure. The wireless device applications 24 with simplified message formats and simplified data structures is created from a specification for the web service

20. As will be explained below with reference to FIG. 7, the simplified data structures are created by "flattening" data structures defined in the web service specification. The process of flattening the web service data structures involves removing any unnecessary structure from the complex data structures, such as the exemplary data structure 20 shown in FIG. 2. The flattening of the complex data structures is performed by the application developer 22 using an automated or semi-automated algorithm, as will be explained below with reference to FIGs. 7 and 8.

After the simplified data structures and message mapping is created for converting complex service messages to simplified wireless messages. The message mapping 26 is used by a proxy at an edge of the wireless network to convert the complex service messages to simplified wireless messages before the simplified wireless messages are sent wirelessly to the wireless device users 10a, 10b. In one embodiment of the disclosure the proxy that applies the data mapping 26 is an application gateway, as will be explained below with reference to Figs. 4 and 6.

FIG. 4 is a block diagram of proxy 40 in accordance with the disclosure. The proxy 40 is located in a communications path between the wireless device 10a, 10b and the remote service, for example, a worldwide web service 18a, 18b. In one embodiment of the disclosure the proxy 40 is an application gateway, and is hereinafter referred to as the application gateway 40.

The application gateway 40 supports a wireless network interface 46 having a link 42 to the wireless network. A message transformation function 48 receives messages from the wireless network interface 46 and processes the messages before forwarding the messages to a service network interface 50. The service network interface 50 has a link to a service network 44 (the Internet, for example) over which it forwards the messages to an appropriate web service(s). In accordance

with the disclosure, the application gateway 40 is provisioned with a plurality of message maps 52, 54. The message maps 52, 54 are created by the wireless application developer 22 and used by the message transformation function 48 to process service request and service response messages, as will be explained below in more detail with reference to FIG. 6. One message map 52, 54 is created by the application developer 22 for each message type used by each web service 18a, 18b.

FIG. 5 is a block diagram of a wireless device 56 in accordance with the disclosure. The wireless device 56 includes a network connection interface 58 that is well known in the art and used to communicate wirelessly with the public domain carrier 12. The wireless device 56 further includes a user interface 60, which may be a keypad, a touch sensitive screen, voice recognition software, or any other user interface for wireless devices. A device infrastructure 62 includes memory, processor(s) peripheral ports, keypad, display and other hardware components required to support the functionality of the wireless device 56. A run time environment 66 supports a plurality of simplified (optimized) data structures 68a, 68n that store corresponding application data in the simplified data structures, such as a simplified data structure 300, which will be explained below with reference to FIG. 8.

Operation Overview

FIG. 6 provides an overview of wireless messaging using simplified data structures in accordance with the disclosure. In step 70 a wireless device 10a formulates a service request message in a simplified format when a user of the wireless device 10a requests a service from web service 18a. The service request message is forwarded to the application gateway (step 72) which performs service request message mapping in step 74 to transform the service request message in the simplified format into a web service request message format required by the web service 18a. The application gateway 40 forwards the web service request

message to the web service 18a (step 76), which receives the web service request message and processes the message in a manner well known in the art (step 78). The web service 18a then formulates and returns a web service response message (step 80). The application gateway 40 receives the web service response message, correlates the web service response message with the web service request message sent earlier, and performs web service response message mapping in step 82 to transform the web service response message to a response message in the simplified format used by the wireless device 10a. The application gateway 40 forwards the response message in the simplified format to the wireless device 10a in step 84. The wireless device 10a receives the service response message and performs service response message processing in step 86. The wireless device 10a may generate a display of information to the user in step 88, depending on a content of the service response message processed in step 86.

Data Structure Simplification

FIG. 7 is a flow chart of an algorithm in accordance with the disclosure embodied in computer executable code for simplifying complex data structures by "flattening" the data structures. This is a post-mapping process in which general message mapping is performed in a manner that is not within the scope of the instant disclosure.

The algorithm shown in FIG. 7 is, for example, a part of the application developer toolkit 22 (FIG. 3). The algorithm accepts complex web service messages as input and outputs a simplified service message for wireless messaging and a message map that permits the complex web service message to be reconstructed using the simplified service message and the message map.

As shown in FIG. 7, a web service message is selected in step 100. The selected message is, for example, a web service message that contains a complex data structure as shown in FIG. 2 with several levels of

nesting that tax the resources of wireless transmission systems, as well as the limited processing power and storage memory of wireless devices 10a, 10b. The selected message is traversed in postfix order and each field of the message is examined (step 102). It is determined in step 104 whether a current field being examined refers to a candidate data component that permits the message to be flattened. A field that refers to a data structure that contains only a single field is a candidate data component. If the current field does not refer to a candidate data component, the algorithm checks to determine whether there are more fields in the message (step 112) and, if so, the algorithm returns to step 102. In step 106 it is determined whether the referring field and candidate data component are both of the array type. If both the referring field and the candidate data component are of the array type, the algorithm branches to step 112, and continues as described above.

However, if both the referring field and the candidate data component are not of the array type, the algorithm copies field attributes of the candidate data component up a level into the referring field's structure. A message map 52 (FIG. 4) is then updated to add a path segment to identify the original location of the referring field (step 110). The algorithm then returns to step 112.

When the entire message has been traversed in postfix order, it is determined in step 114 whether there is another message to process. If so, the algorithm returns to step 100. If not, the simplified (flattened) service messages are examined (step 116), to determine whether any of the simplified data structures are unreferenced in the data maps. For example, as shown in FIG. 8, and as will be explained below in detail with reference to FIG. 8, after flattening the GetSenatorInfoResult field 22 (FIG. 2) is changed to an array of SenatorInfo 204 (FIG. 8). If the type ArrayOfSenatorInfo is not referenced by any of the messages in a wireless application with which the messages are associated, then the algorithm deletes the

unreferenced data structure (step 118) The same applies to the ArrayOfLocalAddress type 208 (FIG. 8) . After any unreferenced data structure(s) are deleted, the remaining simplified data structures are saved (step 120). All modified message maps are then saved (step 122), and the message flattening algorithm terminates.

FIG. 8 illustrates the effects of employing the algorithm shown in FIG. 7 for simplifying, i.e. flattening, a web service message that contains the complex data structure shown in FIG. 2.

FIG. 8 schematically illustrates the complex data structure 20 shown in FIG. 2 after it has been simplified by flattening. The simplified data structure 200 does not contain the referring fields “GetSenatorInfoResult” 22 or “LocalAddress” 26. Rather, the field “GetSenatorInfoResult” 204 and “LocalAddress” 206 are respectively copied up into the position formally occupied by the respective referring fields 22, 26. With that exception, the simplified data structure shown in FIG. 8 is the same as the complex data structure shown in FIG. 7. Of course, as will be understood by those skilled in the art, for simplicity of illustration the exemplary data structures shown in FIGs. 2 and 8 are not necessarily representative of the complexity of data structures used by web services 18a, 18b.

As will be appreciated by those skilled in the art, the disclosure allows application developers to improve wireless device performance by reducing data processing overheads. It also allows application developers to reduce storage space requirements on the wireless devices 10a, 10b by reducing space-consuming complex data structures. The present disclosure also allows application developers to reduce bandwidth usage in the wireless leg of messaging required for accessing remote services, such as worldwide web services, while providing message maps for transparently delivering messages to the remote services in a native format that includes the complex data structures expected by the remote services.

Although the disclosure has been explained with explicit reference to web services, those skilled in the art will appreciate that the disclosure can be used to efficiently access any remote service that uses complex data structures for request/response messaging.

The embodiment(s) of the disclosure above is(are) intended to be exemplary only. The scope of the disclosure is therefore intended to be limited solely by the scope of the appended claims.

CLAIMS

1. A system for simplification of a data structure in wireless communications messages, comprising:

a computer comprising a developer toolkit, the developer toolkit comprising computer executable code for accepting as input a service message containing complex data structures with multiple levels of nesting and outputting a simplified service message containing simplified data structures which contain fewer levels of nesting, and further outputting a message map for transforming the simplified service message to the complex service message, and vice versa;

wherein the computer executable code is arranged to add a path segment to the message map to permit the input service message to be restored from the simplified service message.

2. The system as claimed in claim 1 wherein the computer executable code is arranged to traverse the service message in postfix order.

3. The system as claimed in claim 2 wherein the computer executable code is arranged to determine whether each current field encountered in the postfix order traverse refers to a candidate data component containing a single field.

4. The system as claimed in claim 3 wherein the computer executable code is arranged to determine whether each of the current field and the candidate data component are of an array type.

5. The system as claimed in claim 4 wherein the computer executable code is further arranged to copy single field attributes to a position occupied by the current field if only one of the current field and the field referred to are of an array type.

6. The system as claimed in any one of claims 2 to 5 wherein the computer executable code is further arranged to determine whether more fields remain to be examined in the postfix traverse.

7. The system as claimed in claim 6 wherein the computer executable code is arranged to save the simplified service

messages after all of the service messages have been examined.

8. The system as claimed in claim 7 wherein the computer executable code is further arranged to save the message maps that permits the service messages to be converted to the simplified messages, and vice versa.

9. A method for simplification of a data structure in wireless communications messages, comprising the steps of:

provisioning the application developer toolkit with computer executable code for mapping complex service messages used by the remote service to simplified wireless messages used by the application, and the simplified wireless messages to the complex service messages; and

provisioning a proxy in a communications path between the wireless device and the remote service with a message map comprising the mapped complex service messages, to permit the proxy to convert the complex service messages to the simplified wireless messages and the simplified wireless messages to the complex service messages, to

enable wireless communications between the wireless device and the remote service;

wherein the method further comprises provisioning an application developer toolkit with computer executable code for adding a path segment to the message map to permit the service message to be restored from the simplified data message.

10. The method as claimed in claim 9 further comprising provisioning the application developer toolkit with computer executable code for removing nesting structure from the service messages to output the simplified services messages.

11. The method as claimed in claim 9 or claim 10 wherein provisioning the proxy comprises provisioning an application gateway to serve as the proxy.

12. The method as claimed in claim 11 wherein provisioning the application gateway further comprises provisioning the application gateway with a message transformation function for applying the mapping.

13. The method as claimed in claim 12 further comprising conditioning the message transformation function to apply the mapping to the simplified wireless messages to produce complex service messages that are relayed via a wireline to the remote service, and to apply the mapping to the complex service messages to produce the simplified service messages that are relayed wirelessly to the wireless device.

14. The method as claimed in any one of claims 9 to 13 further comprising provisioning the application developer toolkit with computer executable code for traversing the input service message in a postfix order to remove complex structure from the input service messages.

15. The method as claimed in claim 14 further comprising provisioning the application developer toolkit with computer executable code for examining each field in the postfix order traverse to determine whether a current field refers to a candidate data component.

16. The method as claimed in claim 15 further comprising provisioning the application developer toolkit with computer executable code for determining whether the

current field and the candidate data component are both fields of an array type.

17. The method as claimed in claim 16 further comprising provisioning the application developer toolkit with computer executable code for copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of the array type.

18. A machine readable medium comprising program code means executable on a computer having a programmable processor for implementing the method as claimed in any one of claims 9 to 17.

19. A system for simplification of data structure in wireless communications messages, the system comprising:

a computing device comprising a processor and a computer-readable medium storing computer executable code for effecting a developer toolkit, said computer executable code, when executed by said processor, causing said device to:

accept as input a complex service message containing a complex data structure with multiple levels of nesting;

remove at least one field of said complex data structure to form a simplified data structure with fewer levels of nesting;

output a simplified service message which contains said simplified data structure with fewer levels of nesting; and

output a message map for transforming the simplified service message to the complex service message, and vice versa,

wherein said removing of said at least one field comprises traversing the complex service message in postfix order and,

for a current field encountered in the postfix order traverse:

determining whether the current field refers to a candidate data component containing a single field; and

if said determining is positive, copying a single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type,

and wherein said outputting of said message map comprises adding, to said message map, a path segment identifying an original location of said removed at least one field for use in said transforming of said simplified service message to said complex service message.

20. The system as claimed in claim 19 wherein the computer executable code further determines whether more fields remain to be examined in the postfix traverse.

21. The system as claimed in claim 20 wherein the computer executable code saves the simplified service message after the complex service message has been examined.

22. The system as claimed in claim 21 wherein the computer executable code further saves the message map that permits the complex service message to be converted to the simplified service message, and vice versa.

23. A method for simplification of wireless communications messages, comprising:

provisioning an application developer toolkit with computer executable code that, upon execution by a processor:

accepts as input a complex service message having a complex data structure, said complex data structure for communicating and storing information associated with a remote service that utilizes service messages for communicating information to wireless devices that access the remote service;

removes at least one field of said complex data structure to form a simplified data structure with fewer levels of nesting;

outputs a simplified service message with said simplified data structure; and

outputs a message map for transforming the simplified service message to the complex service message, and vice versa,

wherein said removing of said at least one field of said complex data structure comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse:

determining whether the current field refers to a candidate data component containing a single field; and

if said determining is positive, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type,

and wherein said outputting of said message map comprises adding, to said message map, a path segment identifying an original location of said removed at least one field for use in said transforming of said simplified service message to said complex service message.

24. The method as claimed in claim 23 further comprising:

provisioning a proxy in a communications path between the wireless device and the remote service with the message map, to permit the proxy to convert the complex service message to the simplified service message and the simplified service message to the complex service message, to enable wireless communications between the wireless devices and the remote service.

25. The method as claimed in claim 24 wherein provisioning the proxy comprises provisioning an application gateway to serve as the proxy.

26. The method as claimed in claim 25 wherein provisioning the application gateway further comprises provisioning the

application gateway with a message transformation function for applying the message map.

27. The method as claimed in claim 26 further comprising conditioning the message transformation function to apply the message map to simplified service messages to produce complex service messages that are relayed via a wireline to the remote service, and to apply the message map to the complex service messages to produce the simplified service messages that are relayed wirelessly to the wireless devices.

28. The method as claimed in claim 23 further comprising provisioning the application developer toolkit with computer executable code for removing nesting structure from the complex service message to output the simplified service message.

29. A computer-readable medium storing computer executable code for effecting a developer toolkit for facilitating simplification of data structure in wireless communications messages, said computer executable code, when executed by a processor of a computing device of said system, causing said computing device to:

accept as input a complex service message containing a complex data structure with multiple levels of nesting;

remove at least one field of said complex data structure to form a simplified data structure with fewer levels of nesting; and

output a message map for transforming the simplified service message to the complex service message, and vice versa,

wherein said removing of said at least one field comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: upon determining that the current field refers to a candidate data component containing a single field, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type,

and wherein said outputting of said message map comprises adding, to said message map, a path segment identifying an original location of said removed at least

one field for use in said transforming of said simplified service message to said complex service message.

30. The computer-readable medium of claim 29 wherein the computer executable code further causes said computing device to determine whether more fields remain to be examined in the postfix traverse.

31. The computer-readable medium of claim 30 wherein the computer executable code causes said computing device to save the simplified service message after the complex service message has been examined.

32. The computer-readable medium of claim 31 wherein the computer executable code further causes said computing device to save the message map that permits the complex service message to be converted to the simplified service message, and vice versa.

33. The computer-readable medium of claim 29 wherein said computer executable code further causes said computing device to output said simplified service message which contains said simplified data structure with fewer levels of nesting.

34. The computer-readable medium of claim 33 wherein said computer executable code further causes said computing device to remove nesting structure from the complex service message to output the simplified service message.

35. A computing device in a system for simplification of data structure in wireless communications messages, said computing device comprising a processor and memory storing computer executable code for effecting a developer toolkit, said computer-executable code, when executed by said processor, causing said computing device to:

accept as input a complex service message containing a complex data structure with multiple levels of nesting;

remove at least one field of said complex data structure to form a simplified data structure with fewer levels of nesting; and

output a message map for transforming the simplified service message to the complex service message, and vice versa,

wherein said removing of said at least one field comprises traversing the complex service message in postfix order and, for a current field encountered in the postfix order traverse: upon determining that the current field refers to a candidate data component containing a single field, copying single field attributes of the candidate data component to a position occupied by the current field if only one of the current field and the candidate data component are of an array type,

and wherein said outputting of said message map comprises adding, to said message map, a path segment identifying an original location of said removed at least one field for use in said transforming of said simplified service message to said complex service message.

36. The computing device of claim 35 wherein the computer executable code further causes said computing device to determine whether more fields remain to be examined in the postfix traverse.

37. The computing device of claim 36 wherein the computer

executable code causes said computing device to save the simplified service message after the complex service message has been examined.

38. The computing device of claim 37 wherein the computer executable code further causes said computing device to save the message map that permits the complex service message to be converted to the simplified service message, and vice versa.

39. The computing device of claim 35 wherein said computer executable code further causes said computing device to output said simplified service message which contains said simplified data structure with fewer levels of nesting.

40. The computing device of claim 39 wherein said computer executable code further causes said computing device to remove nesting structure from the complex service message to output the simplified service message.

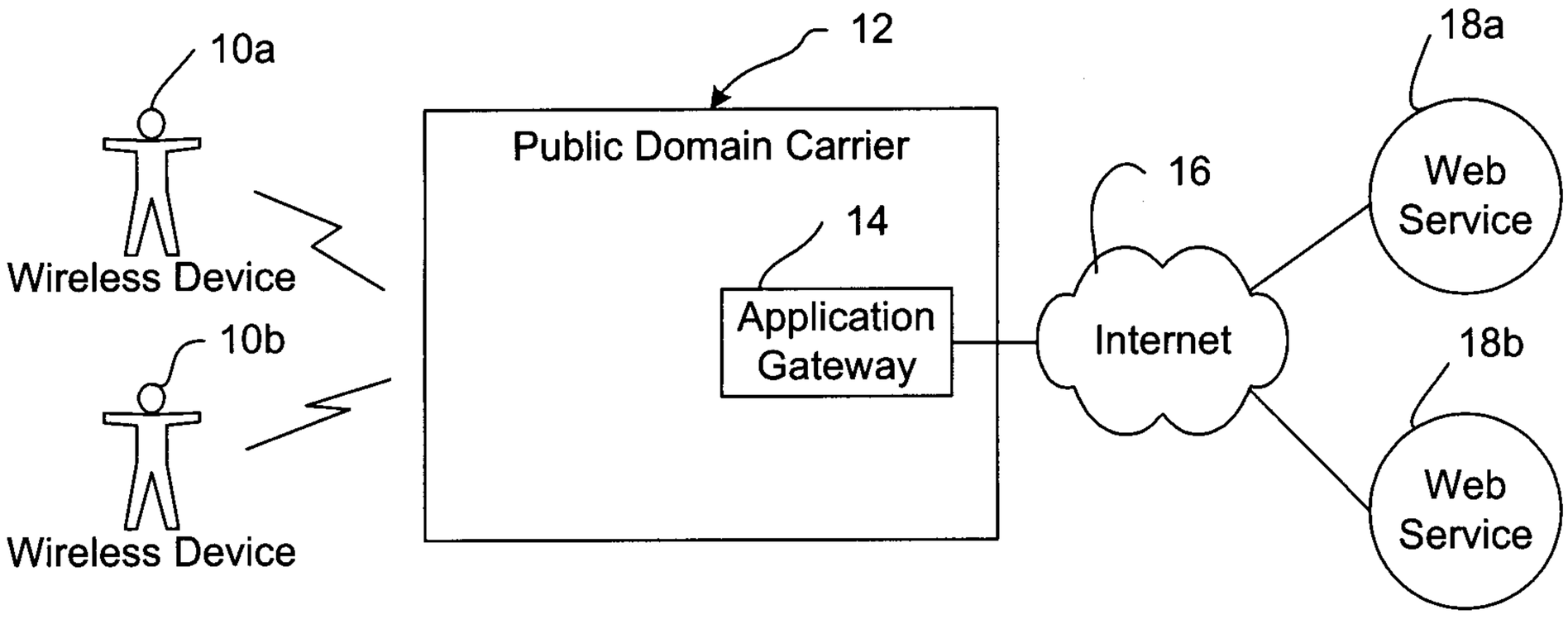


FIG. 1
Prior Art

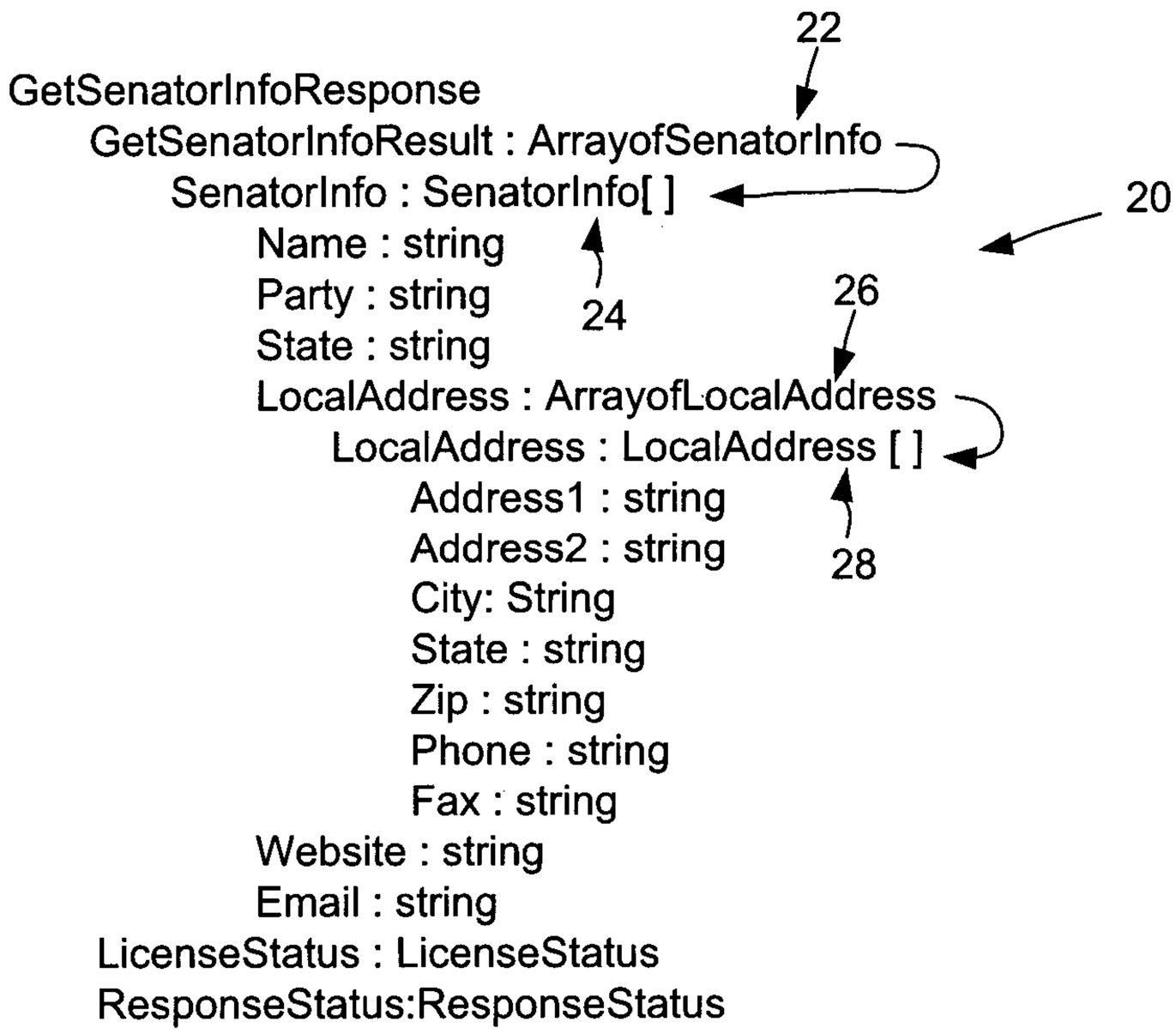


FIG. 2
Prior Art

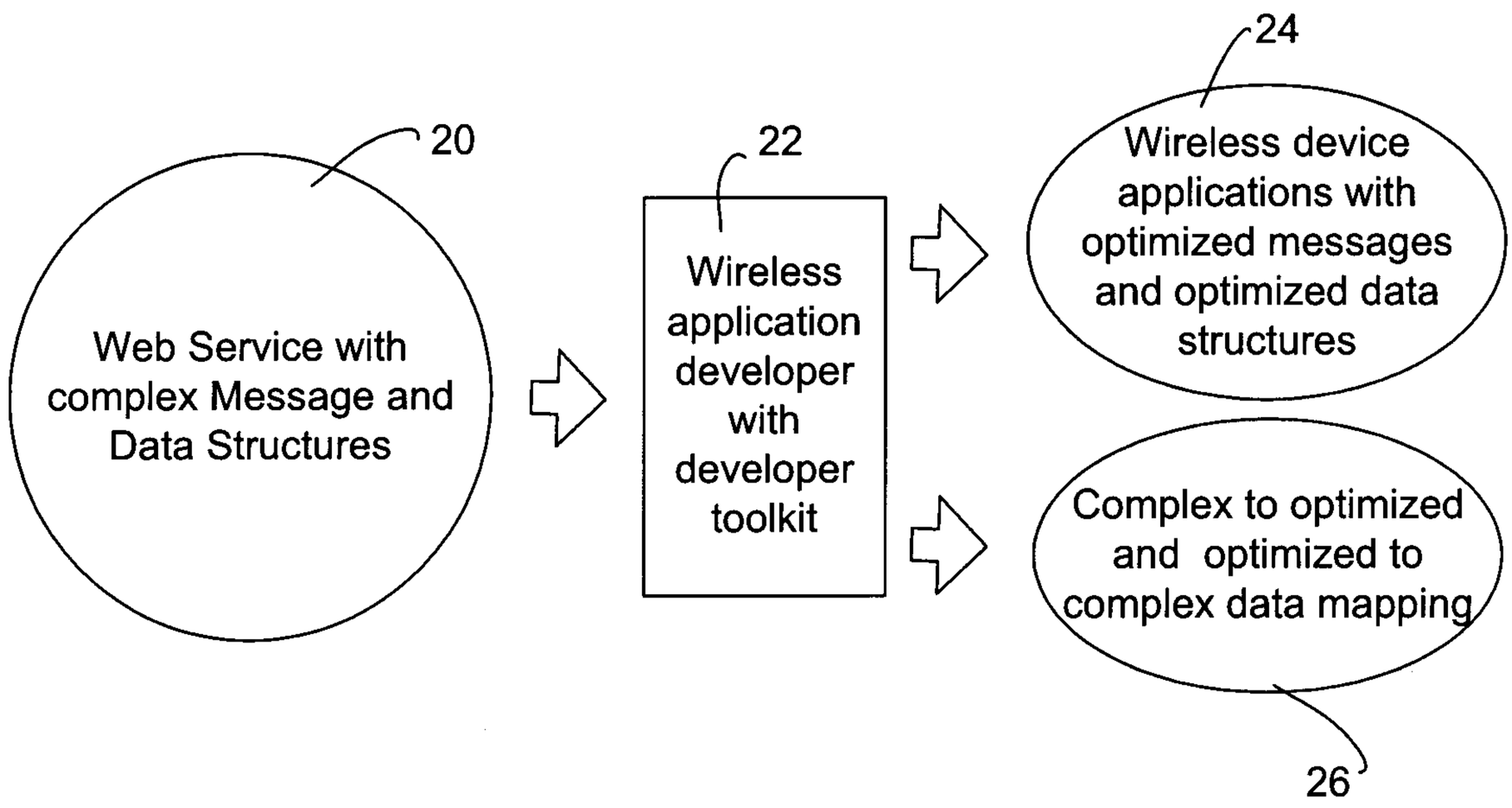

20
Web Service with complex Message and Data Structures
22
Wireless application developer with developer toolkit
24
Wireless device applications with optimized messages and optimized data structures
Complex to optimized and optimized to complex data mapping
26

FIG. 3

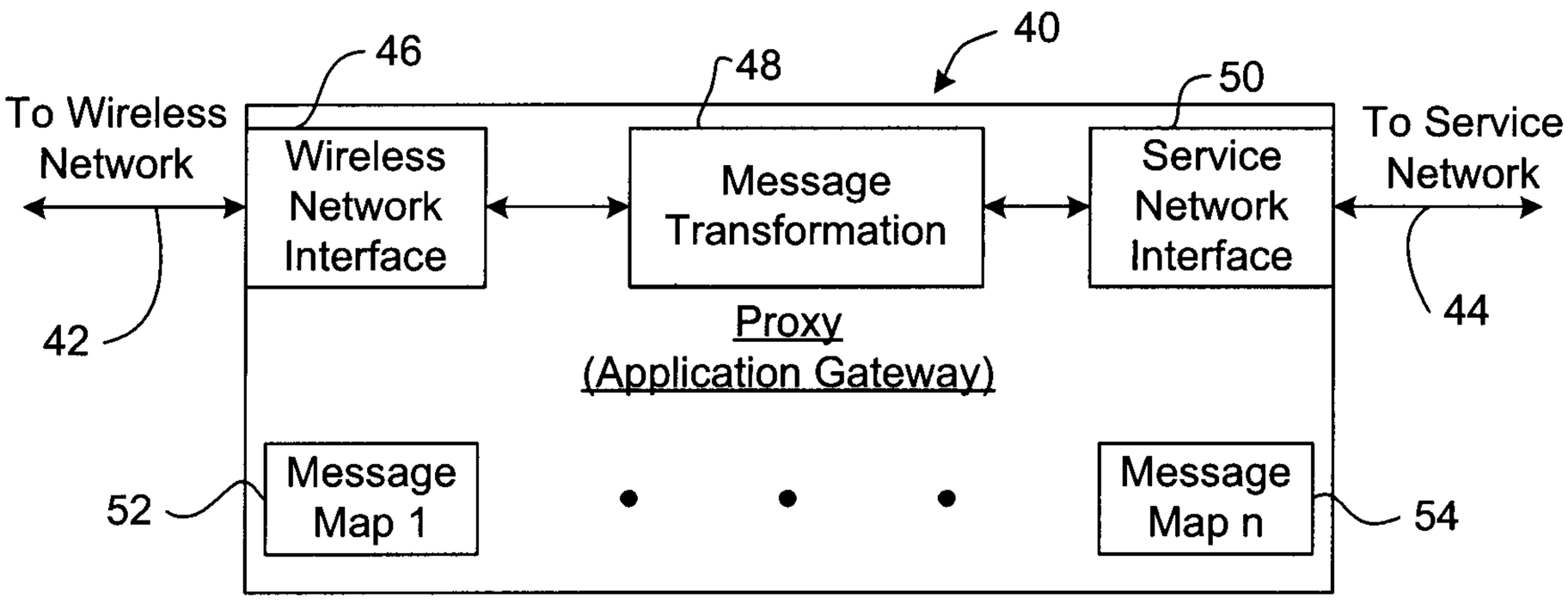

40
46
48
50
To Wireless Network
Wireless Network Interface
Message Transformation
Service Network Interface
To Service Network
42
44
Proxy
(Application Gateway)
52
Message Map 1
Message Map n
54

FIG. 4

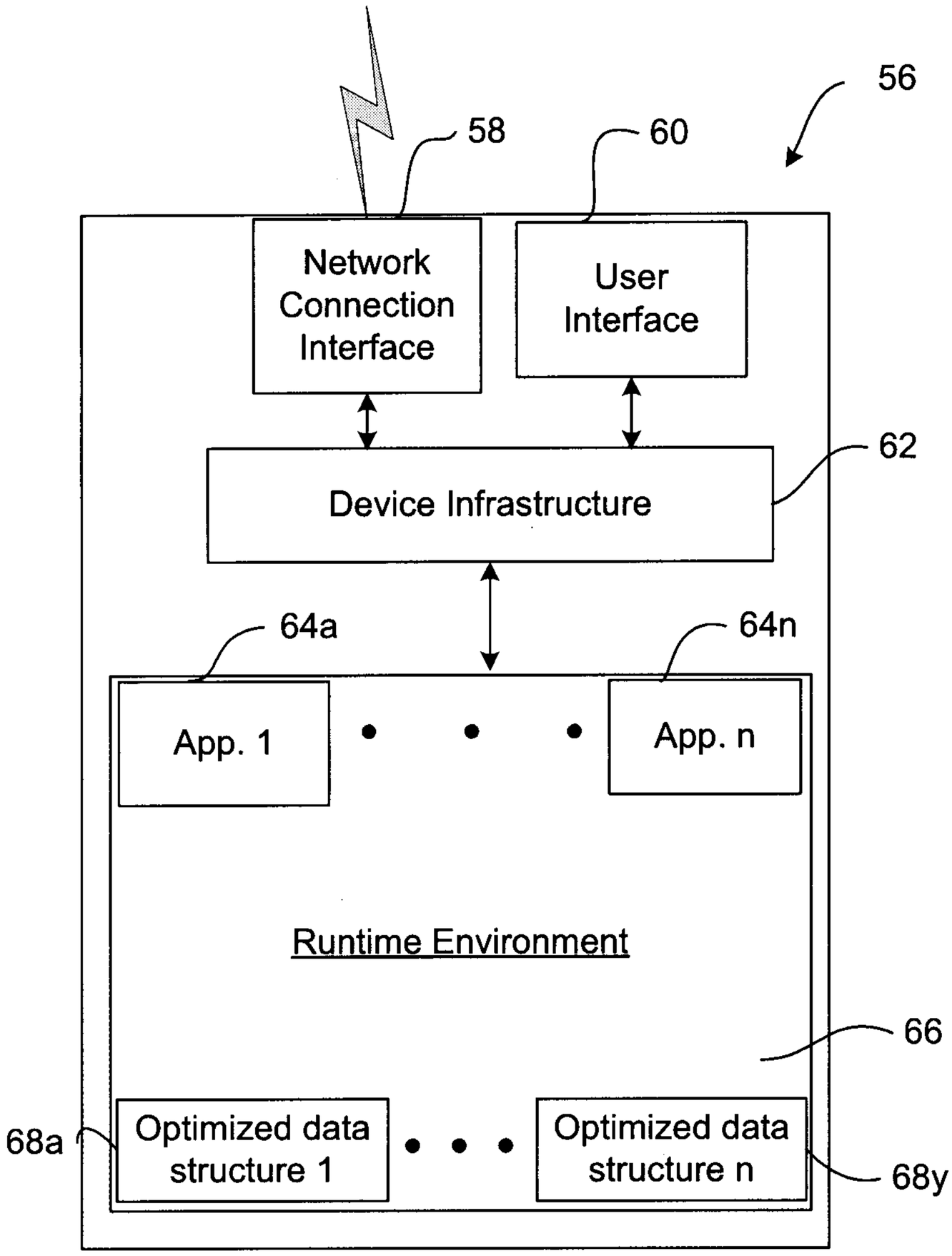

56
58
60
Network Connection Interface
User Interface
62
Device Infrastructure
64a
64n
App. 1
App. n
Runtime Environment
66
68a
Optimized data structure 1
Optimized data structure n
68y

FIG. 5

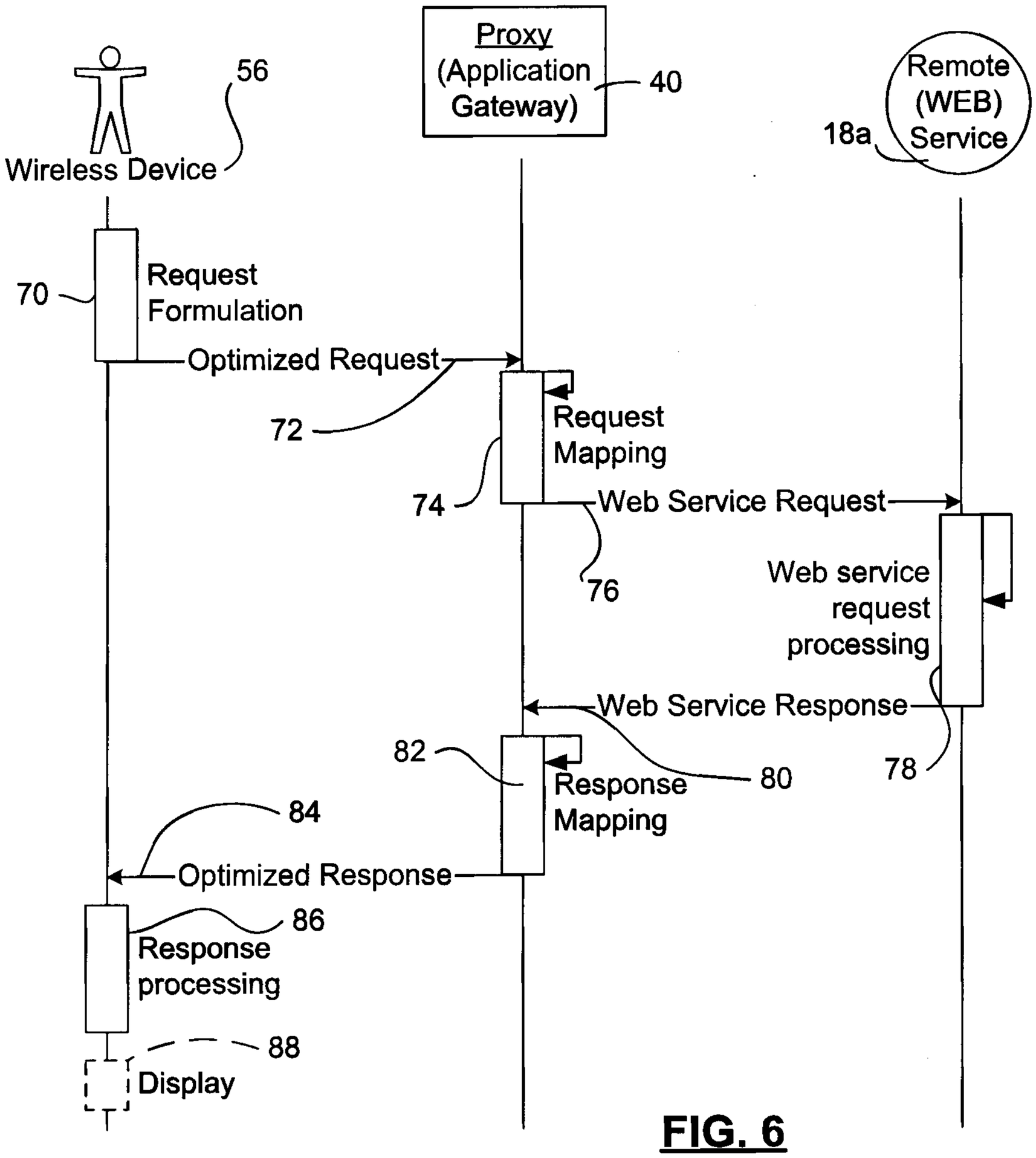

Proxy
(Application
Gateway)
40
Remote
(WEB)
Service
18a
56
Wireless Device
70
Request
Formulation
Optimized Request
72
Request
Mapping
74
Web Service Request
76
Web service
request
processing
Web Service Response
82
80
78
Response
Mapping
84
Optimized Response
86
Response
processing
88
Display

FIG. 6

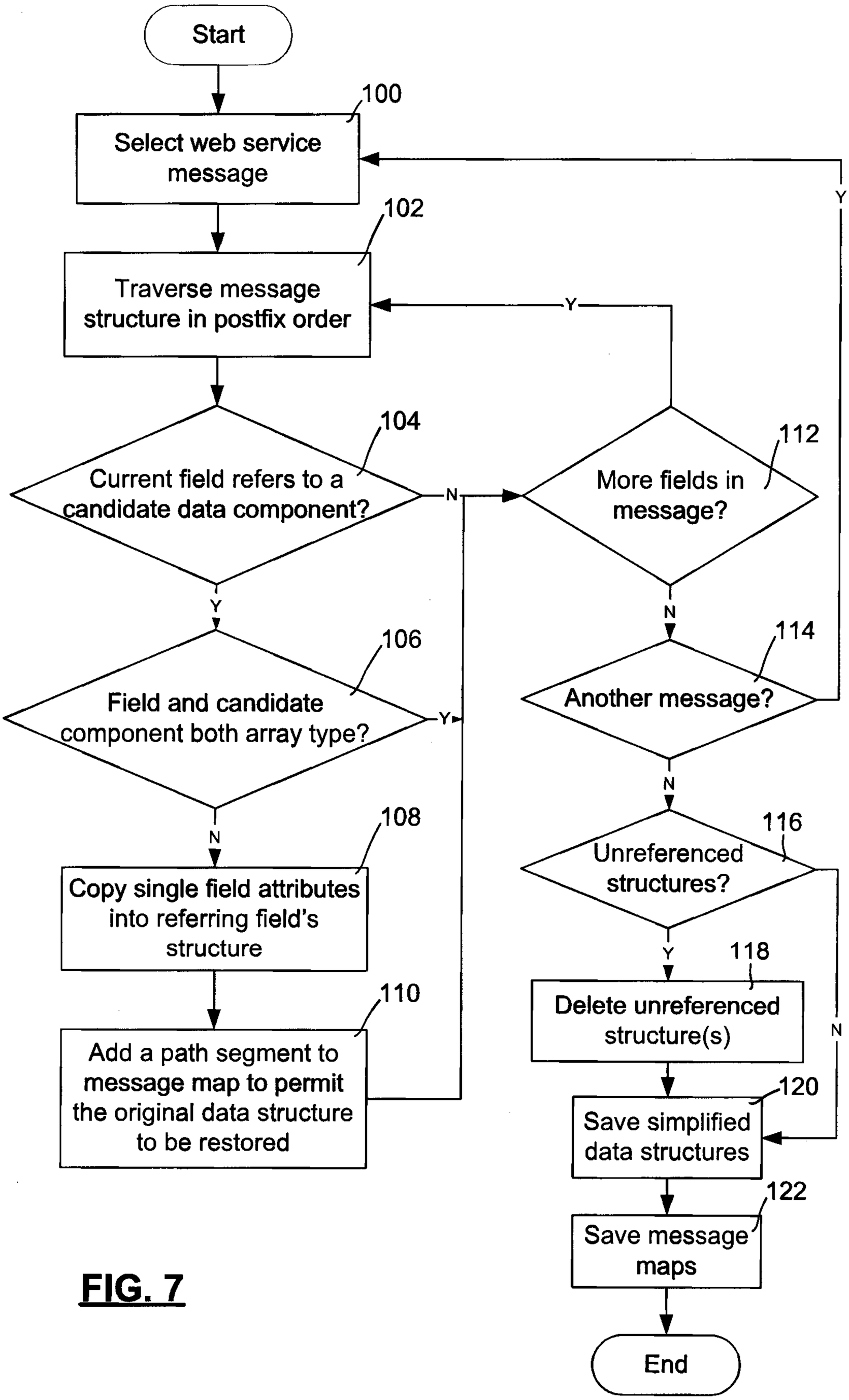

Start
100
Select web service message
102
Traverse message structure in postfix order
104
Current field refers to a candidate data component?
Y
N
106
Field and candidate component both array type?
Y
N
108
Copy single field attributes into referring field's structure
110
Add a path segment to message map to permit the original data structure to be restored
112
More fields in message?
Y
N
114
Another message?
Y
N
116
Unreferenced structures?
Y
N
118
Delete unreferenced structure(s)
120
Save simplified data structures
122
Save message maps
End

FIG. 7

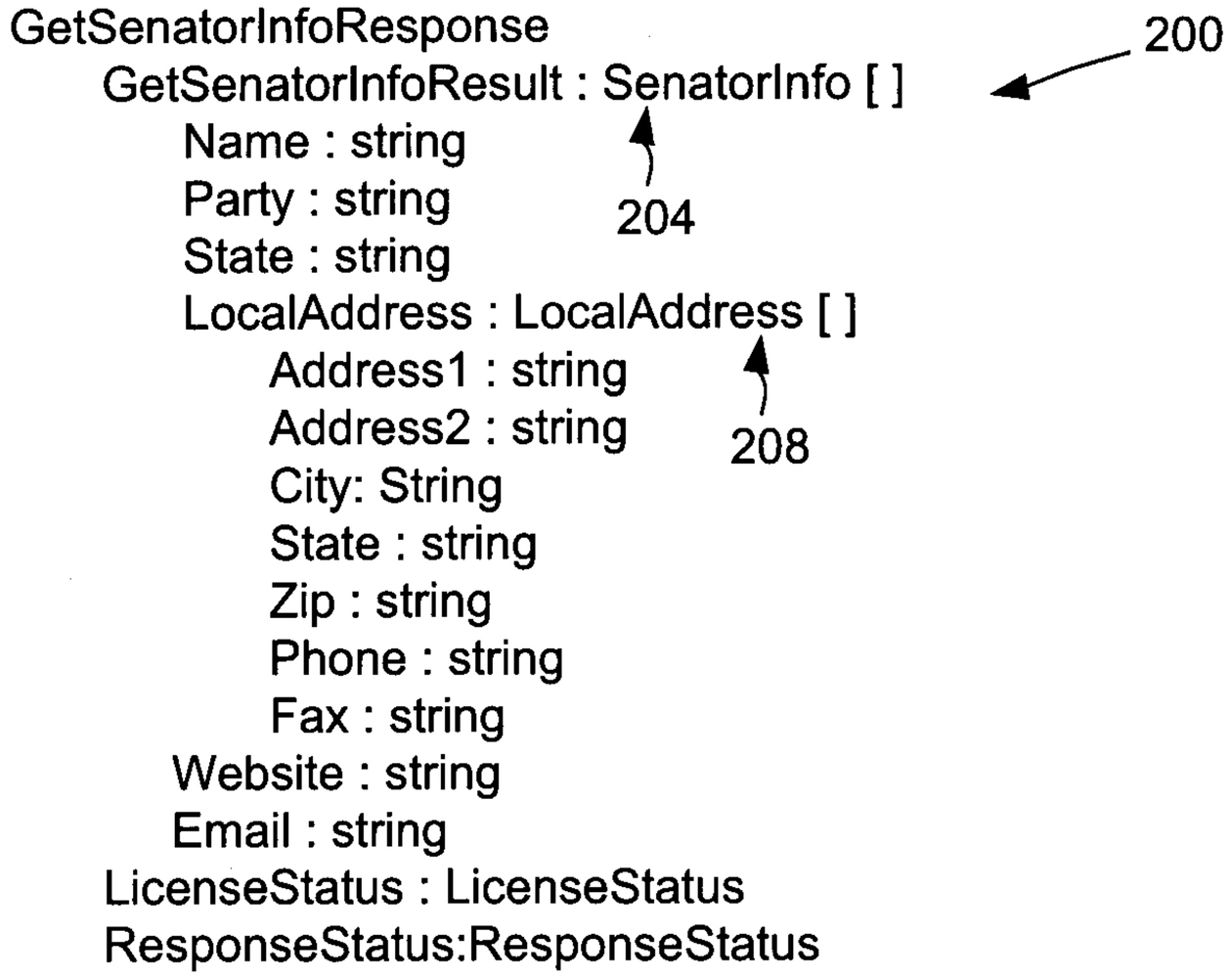
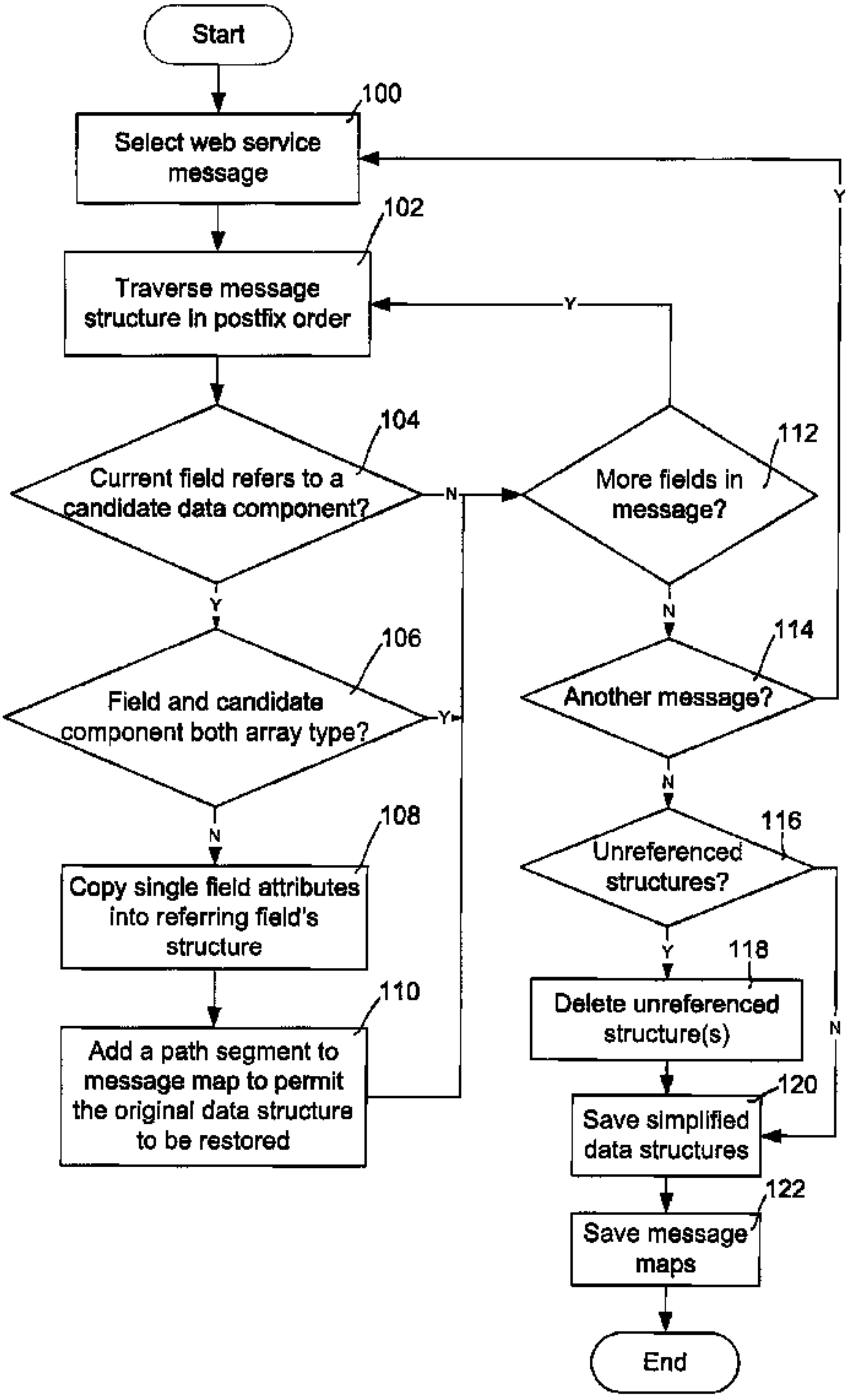

GetSenatorInfoResponse
GetSenatorInfoResult : SenatorInfo [ ]
Name : string
Party : string
State : string
LocalAddress : LocalAddress [ ]
Address1 : string
Address2 : string
City: String
State : string
Zip : string
Phone : string
Fax : string
Website : string
Email : string
LicenseStatus : LicenseStatus
ResponseStatus:ResponseStatus
200
204
208

FIG.8


Start
100
Select web service message
102
Traverse message structure in postfix order
104
Current field refers to a candidate data component?
Y
N
106
Field and candidate component both array type?
Y
N
108
Copy single field attributes into referring field's structure
110
Add a path segment to message map to permit the original data structure to be restored
112
More fields in message?
Y
N
114
Another message?
Y
N
116
Unreferenced structures?
Y
N
118
Delete unreferenced structure(s)
120
Save simplified data structures
122
Save message maps
End