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

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2834046 C 2017/10/24

(11)(21) **2 834 046**

(12) BREVET CANADIEN / CANADIAN PATENT

(13) **C**

(86) **Date de dépôt PCT/PCT Filing Date:** 2012/05/24
(87) **Date publication PCT/PCT Publication Date:** 2012/11/29
(45) **Date de délivrance/Issue Date:** 2017/10/24
(85) **Entrée phase nationale/National Entry:** 2013/10/22
(86) **N° demande PCT/PCT Application No.:** KR 2012/004124
(87) **N° publication PCT/PCT Publication No.:** 2012/161535
(30) **Priorité/Priority:** 2011/05/24 (US61/489,652)

(51) **Cl.Int./Int.Cl.** ***H04N 7/08*** (2006.01),
H04N 7/173 (2011.01)

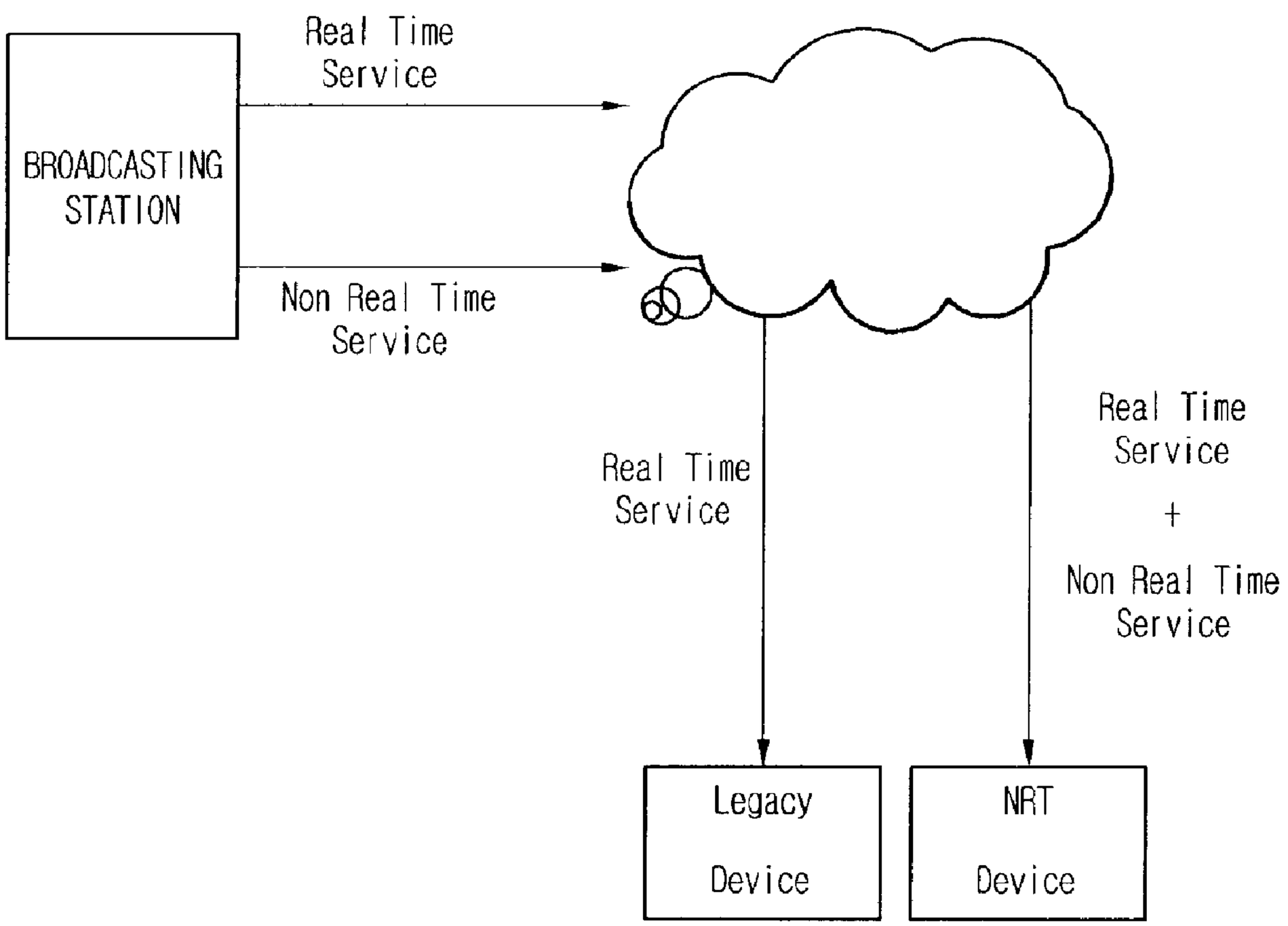
(72) **Inventeurs/Inventors:**
MOON, KYOUNGSOO, KR;
SUH, JONGYEUL, KR;
KIM, KWANSUK, KR;
KIM, JINPIL, KR;
KIM, SANGHYUN, KR;
JI, AETTIE, KR;
KIM, KYUNGHO, KR;
LEE, JOONHUI, KR

(73) **Propriétaire/Owner:**
LG ELECTRONICS INC., KR

(74) **Agent:** SMART & BIGGAR

(54) **Titre : PROCEDE PERMETTANT DE TRANSMETTRE UN SERVICE DE DIFFUSION, APPAREIL PERMETTANT DE LE RECEVOIR, ET PROCEDE PERMETTANT DE TRAITER UN SERVICE SUPPLEMENTAIRE EN UTILISANT L'APPAREIL PERMETTANT DE LE RECEVOIR**

(54) **Title: METHOD FOR TRANSMITTING A BROADCAST SERVICE, APPARATUS FOR RECEIVING SAME, AND METHOD FOR PROCESSING AN ADDITIONAL SERVICE USING THE APPARATUS FOR RECEIVING SAME**



(57) **Abrégé/Abstract:**
According to the present invention, a method for receiving a broadcast service comprises the steps of: selecting a non-real time service object; receiving service signaling data corresponding to the selected non-real time service object; determining a

CA 2834046 C 2017/10/24

(11)(21) **2 834 046**

(13) **C**

(57) **Abrégé(suite)/Abstract(continued):**
consumption model of the non-real time service object on the basis of the service signaling data; acquiring, if the consumption model is a predetermined consumption model, first content item reception information contained in the non-real time service object from the service signaling data; receiving a first content item on the basis of the first content item reception information; and executing the first content item in response to a request for execution from the selected non-real time service object.

ABSTRACT

According to the present invention, a method for receiving a broadcast service comprises the steps of: selecting a non-real time service object; receiving service signaling data corresponding to the selected non-real time service object; determining a consumption model of the non-real time service object on the basis of the service signaling data; acquiring, if the consumption model is a predetermined consumption model, first content item reception information contained in the non-real time service object from the service signaling data; receiving a first content item on the basis of the first content item reception information; and executing the first content item in response to a request for execution from the selected non-real time service object.

DESCRIPTION

METHOD FOR TRANSMITTING A BROADCAST SERVICE, APPARATUS FOR RECEIVING SAME, AND METHOD FOR PROCESSING AN ADDITIONAL SERVICE USING THE APPARATUS FOR RECEIVING SAME

TECHNICAL FIELD

[0001] The present disclosure relates to a method of transmitting a broadcast service, a receiving apparatus for receiving the broadcast service, and a method of processing an adjunct service for the receiving apparatus.

BACKGROUND ART

[0002] A digital television (DTV) is now presented to offer various services in addition to a television (TV)'s original function such as playing video and audio. For example, broadcasting information such as Electronic Program Guide (EPG) may be provided to a user, and also, broadcasting services from at least two channels may be simultaneously provided to a user. Especially, since a receiving system of the DTV includes a large capacity of a storage device, and is connected to a data communication channel and the internet (through which two-way communication is available), more services become accessible through broadcast signals. Additionally, since services offered through broadcast signals become more diversified, needs for providing adjunct services for the services through the broadcast signals, through various devices by using various methods are increased.

50514-84

SUMMARY

[0003] Embodiments provide a method of receiving a broadcast service and processing an adjunct service for the broadcast service, and a method of transmitting a broadcast service.

[0004] Embodiments also provide an adjunct service processing method by which an adjunct service for a broadcast service is provided by using an external device, and a device for receiving a broadcast service.

[0005] Embodiments also provide a broadcast service transmitting method by which information on an adjunct service linked to an external device is provided without interfering with an existing receiver, a device for receiving the broadcast service, and a method of processing an adjunct service for the receiving device.

[0005a] In one embodiment, there is provided a method for a broadcast receiving device to receive a broadcast, comprising: receiving uncompressed content from a broadcaster; extracting characteristic information from the uncompressed content, wherein the characteristic information includes location information on a server providing a trigger; receiving the trigger from the server, by using the location information, wherein the trigger includes an identifier for identifying a triggered declarative object (TDO), a trigger time indicating a timing for transmitting a command and a trigger action indicating a state of the TDO; receiving TDO information on the TDO for a second screen device; generating the command for the TDO based on the trigger and the TDO information, wherein the command only includes the trigger action; and transmitting the command for the TDO to the second screen device at a specific

time designated by the trigger time, wherein the second screen device acts on the command for the TDO.

[0005b] There is also provided in another embodiment an apparatus for receiving a broadcast service, comprising: a receiving unit for receiving uncompressed content from a broadcaster including characteristic information, for receiving a trigger from a server by using location information included in the characteristic information, and for receiving TDO information on the TDO for a second screen device, wherein the trigger includes an identifier for identifying a triggered declarative object (TDO), a trigger time indicating a timing for activating the TDO, and a trigger action indicating a state of the TDO; a service manager for generating a command for the TDO based on the trigger and the TDO information, wherein the command only includes the trigger action; and a network interface for transmitting the command for the TDO to the second screen device at a specific time designated by the trigger time, wherein the second screen device is to act on the command for the TDO.

[0006] In one embodiment, a method of processing an adjunct service for a broadcast service includes receiving signaling data for receiving the broadcast service; obtaining link information for linking the adjunct service to an external device; obtaining information on an external device providing the adjunct service and operation information on the adjunct service on a basis of the link information; providing the broadcast service on a basis of the broadcast service signaling data; transmitting the operation information on the adjunct service to an external device that is identified by the

50514-84

external device information, if an adjunct service request for the broadcast service occurs.

[0007] In another embodiment, a device for receiving a broadcast service includes a receiving unit receiving linkage information for linking a broadcast service, signaling data for receiving the broadcast service, and an adjunct service for the broadcast service to an external device; a service manager obtaining information on an external device for providing the adjunct service and operation information on the adjunct service on a basis of the link information and providing the broadcast service on a basis of the broadcast service signaling data, wherein if an adjunct service request for the broadcast service occurs, the service manager identifies an external device form the external device information and obtains operation information on the adjunct service; and a network interface unit transmitting operation information on the adjunct service to an external device that is identified by the external device information.

[0008] In further another embodiment, a method of transmitting a broadcast service includes generating broadcast service signaling data corresponding to a broadcast service; generating an adjunct service for the broadcast service; generating link information for linking the adjunct service to an external device; and transmitting the broadcast service, the broadcast service signaling data, the adjunct service, and the link information, wherein the link information comprises at least one of information on the external device and operation information on the adjunct service.

50514-84

[0009] According to an embodiment of the present invention, broadcast information may be received and an adjunct service therefor may be provided through an external device.

[0010] Additionally, according to an embodiment of the present invention, information on an adjunct service linked to an external device may be provided.

[0011] Furthermore, according to an embodiment of the present invention, an adjunct service may be efficiently provided and it is possible to enhance the convenience of a user who uses the adjunct service.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] Fig. 1 is a conceptual diagram illustrating how RT service and NRT service are provided.

[0013] Fig. 2 is a view illustrating a structure of NRT service according to an embodiment.

[0014] Fig. 3 is a view illustrating a protocol stack for NRT service according to an embodiment.

[0015] Fig. 4 is view illustrating one example of the protocol stack for mobile NRT service.

[0016] Fig. 5 is a view illustrating a bit stream section of a TVCT table section (VCT) according to an embodiment.

[0017] Figs. 6 and 7 are views illustrating how to define a value of a service_type field according to an embodiment.

[0018] Fig. 8 is view of data_service_table_section) for identifying an application of NRT service and bit stream syntax of data_service_table_bytes in a DST section.

[0019] Fig. 9 is a view illustrating a method of receiving and providing NRT service in a receiving system by using ATSC A/90 standard for transmitting data broadcasting stream and ATSC A/92 standard for transmitting IP multicast stream.

[0020] Figs. 10 and 11 are views illustrating a method of signaling a DSM-CC addressable section data by using VCT

according to another embodiment.

[0021] Fig. 11 is a view illustrating a method of signaling DSM-CC addressable section data by using VCT according to another embodiment of the present invention.

[0022] Figs. 12 and 13 are views illustrating a bit stream syntax of NST according to an embodiment.

[0023] Fig. 14 is a view illustrating a bit stream syntax of NRT_component_descriptor (MH_component_descriptor) according to an embodiment.

[0024] Fig. 15 is a view illustrating a bit stream syntax of NRT component descriptor including NRT_component_data according to an embodiment.

[0025] Fig. 16 is a view illustrating a bit stream syntax of NRT-IT section for signaling NRT application according to an embodiment.

[0026] Fig. 17 is a view illustrating a syntax structure of bit stream for NRT section (NRT_content_table_section) according to an embodiment.

[0027] Fig. 18 is a view illustrating a bit stream syntax structure of an SMT session providing signaling information on NRT service data according to an embodiment.

[0028] Fig. 19 is a view illustrating an FDT schema for mapping a file and content_id according to an embodiment.

[0029] Fig. 20 is a view illustrating an FDT schema for mapping a file and content_id according to another embodiment.

[0030] Fig. 21 is a flowchart illustrating an operation of a receiver according to an embodiment.

[0031] Figs. 22 and 23 are views illustrating a receiving system receiving, storing, and playing an NRT content for NRT service according to another embodiment.

[0032] Fig. 24 is a flowchart illustrating a method of a receiver to receive and provide NRT service according to an embodiment.

[0033] Fig. 25 is a view illustrating a bit stream syntax of a trigger according to an embodiment.

[0034] Fig. 26 is a view illustrating a PES structure according to a synchronized data stream method including a trigger according to an embodiment.

[0035] Fig. 27 is a view illustrating a synchronized data packet structure of PES payload for transmitting trigger as bit stream syntax according to an embodiment.

[0036] Fig. 28 is a view illustrating a content type descriptor structure in tap() on DST according to an embodiment

[0037] Fig. 29 is a view illustrating a syntax of PMT and service identifier descriptor according to an embodiment.

[0038] Fig. 30 is a view illustrating a trigger stream descriptor according to an embodiment.

[0039] Fig. 31 is a view of AIT according to an embodiment.

[0040] Fig. 32 is a view of STT according to an embodiment.

[0041] Fig. 33 is a block diagram illustrating a transmitter for transmitting TDO and a trigger according to an embodiment.

[0042] Fig. 34 is a block diagram illustrating a receiver for receiving TDO and a trigger according to an embodiment.

[0043] Fig. 35 is a flowchart illustrating a trigger transmitting method according to an embodiment.

[0044] Fig. 36 is a flowchart illustrating an operation of a receiver 300 according to an embodiment.

[0045] Fig. 37 is a flowchart illustrating a trigger receiving method by using a trigger table according to an

embodiment.

[0046] Fig. 38 is a flowchart illustrating an operation of a receiver when trigger signaling information and trigger are transmitted using DST according to an embodiment.

[0047] Fig. 39 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using a trigger stream descriptor according to an embodiment.

[0048] Fig. 40 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using a stream type according to an embodiment.

[0049] Fig. 41 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using AIT according to an embodiment.

[0050] Fig. 42 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using STT according to an embodiment.

[0051] Fig. 43 is a view illustrating a syntax of a linkage descriptor (link_descriptor) according to an embodiment of the present invention.

[0052] Figs. 44 and 45 are views illustrating contents of fields included in a link descriptor according to an embodiment of the present invention.

[0053] Fig. 45 is a timing diagram according to another embodiment of the present invention.

[0054] Figs. 46 and 47 are views illustrating a linkage between each table when a link descriptor of Fig. 43 is included in a descriptor of an event information table (EIT) in a PSIP table according to an embodiment of the present invention.

[0055] Fig. 47 is a view illustrating a maintenance trigger

receiving method according to an embodiment of the present invention.

[0056] Fig. 48 is a view illustrating a syntax of an event descriptor (Event_descriptor) and contents of fields in the event descriptor according to an embodiment of the present invention.

[0057] Fig. 49 is a view illustrating a method of identifying a linkage program through an event descriptor according to an embodiment of the present invention.

[0058] Fig. 50 is a flowchart illustrating an operation of receiving by the receiver 300 broadcast program or broadcast channel related contents by using a link descriptor according to an embodiment of the present invention.

[0059] Fig. 51 is a flowchart illustrating an operation of providing by the receiver 300 broadcast program related content by using an event descriptor according to an embodiment of the present invention.

[0060] Fig. 52 is a view illustrating a syntax of an NRT service descriptor (NRT_service_descriptor), that is, a service level descriptor according to an embodiment of the present invention.

[0061] Fig. 53 is a view illustrating a meaning according to each value of a consumption_model field in an NRT service descriptor according to an embodiment of the present invention.

[0062] Fig. 54 is a flowchart illustrating an operation of the receiver 300 when a TDO is transmitted by a TDO consumption model according to an embodiment of the present invention.

[0063] Fig. 55 is a flowchart illustrating a method of

allocating and managing a TDO storage area according to a TDO consumption model according to an embodiment of the present invention.

[0064] Fig. 56 is a view illustrating a TDO metadata descriptor according to an embodiment of the present invention.

[0065] Fig. 57 is a flowchart illustrating an operation of receiving by the receiver 300 TDO metadata according to an embodiment of the present invention.

[0066] Fig. 58 is a flowchart illustrating a method of the receiver 300 to manage a TDO depending on time information in TDO metadata according to an embodiment of the present invention.

[0067] Fig. 59 is a flowchart illustrating a method of the receiver 300 to manage a TDO depending on time information and priority information in TDO metadata according to another embodiment of the present invention.

[0068] Fig. 60 is a view illustrating a syntax of an internet location descriptor according to an embodiment of the present invention.

[0069] Fig. 61 is a flowchart illustrating an operation of the receiver 300 when an FDT is transmitted through an internet network according to an embodiment of the present invention.

[0070] Fig. 62 is a flowchart illustrating an operation of the receiver 300 when the URL of an FDT is transmitted through a link descriptor according to an embodiment of the present invention.

[0071] Fig. 63 is a flowchart illustrating an operation of the receiver 300 when the URL of an FDT is transmitted

through an NRT-IT according to an embodiment of the present invention.

[0072] Fig. 64 is a conceptual view illustrating an NRT service including an entry content item.

[0073] Figs. 65 and 66 are views illustrating an NRT-IT to transmit information on an entry content item according to an embodiment of the present invention.

[0074] Fig. 67 is a view illustrating an operation method of a receiver when an entry content item is transmitted according to an embodiment of the present invention.

[0075] Fig. 68 is a conceptual view of a plurality of NRT service objects transmitted according to an embodiment of the present invention.

[0076] Fig. 69 is a view illustrating a syntax of an NRT service descriptor included in an SMT according to an embodiment of the present invention.

[0077] Figs. 70 and 71 are views illustrating a syntax of another NRT-IT according to another embodiment of the present invention.

[0078] Fig. 72 is a view illustrating a syntax of an Other NRT location descriptor (Other_NRT_location_descriptor) according to another embodiment of the present invention.

[0079] Fig. 73 is a flowchart illustrating a method of receiving broadcast service according to an embodiment of the present invention.

[0080] Fig. 74 shows syntax of a companion device descriptor according to an embodiment;

[0081] Fig. 75 shows a companion device action descriptor according to an embodiment;

[0082] Fig. 76 shows an action type according to a value of an action type field of a

companion device action descriptor according to an embodiment;

[0083] Fig. 77 is a flow chart of a method of operating a receiver according to an embodiment;

[0084] Fig. 78 shows a case where link information is included in an SMT;

[0085] Fig. 79 shows a case where link information is included in an EIT, a VCT or a PMT;

[0086] Fig. 80 is a diagram for explaining how link information is included in non-real service signaling information according to an embodiment;

[0087] Fig. 81 is a diagram for explaining a capabilities descriptor according to an embodiment, and Fig. 82 shows a capabilities category code table according to an embodiment;

[0088] Fig. 83 is a flow chart of a method of obtaining link information by using a capabilities descriptor by a receiver according to an embodiment;

[0089] Fig. 84 shows a content items group descriptor according to another embodiment;

[0090] Fig. 85 is a flow chart of a method of providing a TDO service by using a content items group descriptor by a receiver;

[0091] Fig. 86 shows a companion device group descriptor according to another embodiment;

[0092] Figs. 87A and 87B show a companion device group descriptor according to another embodiment;

[0093] Figs. 88A and 88B show a companion device group descriptor according to another embodiment, and Fig. 89 shows the syntax for the descriptor of Figs. 88A and 88B;

[0094] Fig. 90 is a diagram for explaining a home network including a receiver and an external device according to an embodiment;

[0095] Fig. 91 is a diagram for explaining a home network including a receiver and an external device according to another embodiment;

[0096] Fig. 92 is a flow chart of a message delivery process on a network as shown in Fig. 91;

[0097] Fig. 93 shows a home network according to another embodiment, and Fig. 94 shows a message delivery process on a home network as shown in Fig. 93;

[0098] Fig. 95 is a flow chart of an event message delivery process of a home network including a receiver according to another embodiment;

[0099] Figs. 96 to 98 show the operations of a receiver when there is an adjunct service linkable to an external device;

[00100] Figs. 99A and 99B show the operations of an external device according to an embodiment;

[00101] Figs. 100 to 102 are diagrams for explaining how link information is transmitted by using DTC-CC according to an embodiment;

[00102] Fig. 103 is a block diagram of a network topology according to an embodiment;

[00103] Fig. 104 is a block diagram of a watermark based network topology according to an embodiment;

[00104] Fig. 105 is a ladder diagram of a data flow in a watermark based network topology according to an embodiment;

[00105] Fig. 106 shows a watermark based content recognition timing according to an embodiment;

[00106] Fig. 107 is a block diagram of a fingerprint based network topology according to an embodiment;

[00107] Fig. 108 is a ladder diagram of a data flow in a fingerprint based network topology according to an embodiment;

[00108] Fig. 109 is an ACR-Resulttype XML schema diagram that contains a query result according to an embodiment;

[00109] Fig. 110 is a block diagram of a watermark and fingerprint based network topology according to an embodiment;

[00110] Fig. 111 is a ladder diagram of a data flow in a watermark and fingerprint based network topology according to an embodiment; and

[00111] Fig. 112 is a block diagram of an image display apparatus according to another embodiment.

MODE FOR CARRYING OUT THE INVENTION

[00112] Preferred embodiments of the present invention will be described below in more detail with reference to the accompanying drawings. The configurations and operations of the present invention shown in and described with the accompanying drawings are explained as at least one example, and the technical idea of the present invention and its core configurations and operations are not limited thereby.

[00113] The terms used in the present invention are selected as currently used general terms if possible in the consideration of functions of the present invention but could vary according to intentions or conventions of those in the art or the advent of new technology. In certain cases, there are terms that are selected by an applicant arbitrarily, and in such a case, their meanings will be described in more detail in the specification. Accordingly, the terms used in the present invention should be defined on the basis of the meanings of the terms and contents over the present invention not the simple names of the terms.

[00114] Moreover, among the terms in the present invention, a real time (RT) service literally means a service in real time. That is, the service is time-restricted. In contrast, a non-real time (NRT) service is a service in NRT other than the RT service. That is, the NRT service is not restricted by time. Furthermore, data for NRT service is called NRT service data.

[00115] A broadcast receiver according to the present invention may receive NRT service through a medium such as a terrestrial wave, a cable, and the internet.

[00116] The NRT service may be stored in a storage medium of the broadcast receiver, and then may be displayed on a

display device according to a predetermined time or at the user's request. The NRT service is received in a file format, and is stored in a storage medium according an embodiment. The storage medium may be an HDD embedded in the broadcast receiver according to an embodiment. As another example, the storage medium may be a Universal Serial Bus (USB) memory or an external HDD, which is connected to the broadcast receiving system.

[00117] Signaling information is necessary to receive files constituting the NRT service, store them in a storage medium, and provide a service to a user. The present invention may designate the above signaling information as NRT service signaling information or NRT service signaling data.

[00118] The NRT service includes Fixed NRT service and Mobile NRT service according to a method of obtaining IP datagram including NRT service signaling data. Especially, the Fixed NRT service is provided to a fixed broadcast receiver, and the Mobile NRT service is provided to a mobile broadcast receiver.

[00119] Fig. 1 is a conceptual diagram illustrating how RT service and NRT service are provided.

[00120] A broadcasting station transmits the RT service according to a traditional way, that is, like current terrestrial broadcasting (or mobile broadcasting). At this point, the broadcasting station transmits the RT service, and then, by using a remaining bandwidth during the transmission or an exclusive bandwidth, may provide the NRT service. That is, the RT service and NRT service are transmitted through the same or different channel. Accordingly, in order for a broadcast receiver to separate the RT service and the NRT

service and store the separated NRT service in order to provide it to a user if necessary, service signaling information (or NRT service signaling data) is required. The NRT service signaling information (or NRT service signaling data) will be described in more detail later.

[00121] For example, a broadcasting station transmits broadcasting service data in real time and transmits news clip, weather information, advertisements, and Push VOD in non-real time. Additionally, the NRT service may be specific scenes, detail information of a specific program, and preview in real-time broadcasting stream in addition to news clip, weather information, advertisements, and Push VOD.

[00122] A typical broadcast receiver (i.e., a legacy device) may receive and process the RT service but may not receive and process the NRT service. That is, the typical broadcast receiver (i.e., a legacy device) is not influenced, in principle, by an NRT stream in a channel broadcasting RT service. That is, even when receiving NRT service, the typical broadcast receiver cannot process the received NRT service because it does not include a unit for processing it properly.

[00123] On the contrary, the broadcast receiver (i.e., an NRT device) of the present invention receives NRT service combined with RT service and properly processes the NRT service, so that it provides more various functions to a viewer than a typical broadcast receiver.

[00124] Fig. 2 is a view illustrating a structure of NRT service according to an embodiment.

[00125] The NRT service includes at least one content item (or content or NRT content) as shown in Fig. 2, and the

content item includes at least one file according to an embodiment. A file and object have the same meaning in the present invention.

[00126] The content item is a minimum unit playable independently. For example, news is provided in NRT. If the news includes business news, political news, and lift news, it may be NRT service, and each may be designated as a content item. Moreover, each of the business news, political news, and life news may include at least one file.

[00127] At this point, the NRT service may be transmitted in an MPEG-2 transport stream (TS) packet format through the same broadcasting channel as the RT service or an exclusive broadcasting channel. In this case, in order to identify the NRT service, a unique PID may be allocated to the TS packet of the NRT service data and then transmitted. According to an embodiment of the present invention, IP based NRT service data is packetized into an MPEG-2 TS packet and then transmitted.

[00128] At this point, NRT service signaling data necessary for receiving the NRT service data is transmitted through an NRT service signaling channel. The NRT service signaling channel is transmitted through a specific IP stream on an IP layer, and at this point, this specific IP stream may be packetized into an MPEG-2 TS packet and then transmitted. The NRT service signaling data transmitted through the NRT service signaling channel may include at least one of a Service Map Table (SMT), an NRT Service Table (NST), an NRT Content Table (NCT), an NRT Information Table (NRT-IT), and a Text Fragment Table (TFT). The NST or SMT provides access information on at least one NRT service operating on an IP

layer, or the content items or files constituting the NRT service. The NRT-IT or NCT provides access information on the content items or files constituting the NRT service.

[00129] Additionally, NRT service signaling data including SMT(or NST) and NRT-IT(or NCT) may be included in a PSIP table on MPEG-2 TS or may be transmitted through an NRT service signaling channel on an IP layer in a virtual channel. Moreover, a plurality of NRT service data may be provided through one virtual channel.

[00130] The NRT-IT includes information describing a content downloadable to be stored in a receiving device. Information provided to the NRT-IT may include a content title (for example, the name of a downloadable program), available time for downloading content, content recommendation, availability of caption service, content identification, and other metadata.

[00131] Additionally, the TFT provides detailed description on a content item or service. The TFT may include a data structure supporting multi languages and, as a result, may represent detailed descriptions (e.g., each string corresponds to one language) in different languages. The text fragment table may be included in private sections having a table_id value (TBD) and may be identified by TFT_id. A TFT section may be included IP packets in a service signaling channel, and a multicast IP address (224.0.23.60) and a port (4937) may be allocated to the service signaling channel by IANA.

[00132] First, a receiver may identify whether a corresponding service is the NRT service with reference to a service_category field in the SMT, for example. Additionally,

the receiver may uniquely identify the NRT service from the SMT through an NRT_service_id field.

[00133] Additionally, the NRT service may include a plurality of content items. The receiver may identify an NRT content item through a content_id field in the NCT or NRT-IT. In addition, the NRT content item and NRT service may be connected to each other by matching the NRT_channel_id field of the NCT to the NRT_service_id field.

[00134] Moreover, the NRT service may be transmitted through a FLUTE session and the receiver may extract FDT information from the FLUTE session. Then, content_id in the extracted FDT information is mapped into content_id of NCT or OMA-BCAST SG in order to confirm and receive the NRT service content that a user selects. If the mapping method is described briefly, for example, the receiver identifies each file constituting the NRT content item through the TOI and Content-Location fields in the FDT in the FLUTE session. Each TOI or the Content-Location and content item maps the content_ID of the FDT into the content_id field of the NCT or the content_id field of the OMA BCAST SG, so as to confirm and receive the NRT service content.

[00135] Fig. 3 is a view illustrating a protocol stack for NRT service according to an embodiment.

[00136] For Fixed NRT service, the NRT service of a file format is IP-packetized in an IP layer, and then, is transmitted in an MPEG-2 TS format through a specific channel.

[00137] Through an MPEG-2 based Program Specific Information (PSI) or Program and System Information Protocol (PSIP) table, for example, a VCT, it is determined whether there is NRT service in a virtual channel and identification information

of NRT service is signaled.

[00138] According to an embodiment, the NRT service signaling channel, which transmits NRT service signaling data signaling the access information of the IP based NRT service, is IP packetized into a specific IP stream in the IP layer, and then, is transmitted in an MEPG-2 TS format.

[00139] That is, a broadcasting station packetizes the NRT content item or files according to a file transfer protocol method as shown in Fig. 3, and then, packetizes the packetized NRT content item or files in an Asynchronous Layered Coding (ALC) or Layered Coding Transport (LCT) method. Then, the packetized ALC or LCT data are packetized according to a UDP method. Then, the packetized UDP data is packetized according to the IP method again, and then, becomes IP data. Here, the IP data may include a File Description Table (FDT) having information on a File Delivery over Unidirectional Transport (FLUTE) session. The packetized IP data may be designated as IP datagram for convenience of description in the present invention.

[00140] Additionally, the IP datagram of NRT service is encapsulated in an addressable section structure and is packetized again in an MPET-2 TS format. That is, one addressable section structure has a section header and CRC checksum, which are added to one IP datagram. The format of the addressable section structure is matched to a Digital Storage Media Command and Control (DSM-CC) section format for private data transmission in terms of a structure. Accordingly, the addressable section may be designated as a DSM-CC addressable section.

[00141] Moreover, NRT service signaling data including at

least one of SMT (or NST) and NRT-IT (or NCT) necessary for receiving NRT content/files may be transmitted through an NRT service signaling channel on an IP layer. Accordingly, the NRT service signaling data may be packetized according to an IP method in order to transmit it through the NRT service signaling channel on an IP layer. The NRT service signaling channel is encapsulated in the IP datagram having a well-known IP address and is multi-casted according to an embodiment.

[00142] Additionally, the NRT service signaling data may be included in Program Specific Information (PSI) or Program and System Information Protocol (PSIP) table section data and then transmitted. Moreover, the PSI table may include a Program Map Table (PMT) and a Program Association Table (PAT). The PSIP table may include a Virtual Channel Table (VCT), a Terrestrial Virtual Channel Table (TVCT), a Cable Virtual Channel Table (CVCT), a System Time Table (STT), a Rating Region Table (RRT), an Extended Text Table (ETT), a Direct Channel Change Table (DCCT), a Direct Channel Change Selection Code Table (DCCSCT), an Event Information Table (EIT), and a Master Guide Table (MGT).

[00143] Furthermore, as data for digital rights management and encryption of broadcasting service to protect the NRT service from illegal distribution and reproduction, BroadCast Services Enabler Suite Digital Rights Management (BCAST DRM) suggested by Open Mobile Alliance (OMA) may be used.

[00144] Moreover, the above mentioned Program Specific Information (PSI), Program and System Information Protocol (PSIP) table section data, DSM-CC addressable section data, and OMA BCAST DRM data are divided by a 184 byte unit, and

then, a 4 byte MEPG header is added to each 184 bytes in order to obtain a 188 byte MPEG-2 TS packet. At this point, a value allocated to the PID of the MPEG header is a unique value identifying a TS packet for transmitting the NRT service and NRT service signaling channel.

[00145] MPEG-2 TS packets may be modulated in a predetermined transmission method in a physical layer, for example, an 8-VSB transmission method, and then, may be transmitted to a receiving system.

[00146] Moreover, Fig. 4 is a view illustrating a protocol stack for NRT service according to another embodiment.

[00147] Fig. 4 is view illustrating one example of the protocol stack for mobile NRT service. As shown in Fig. 4, an adaption layer is included between an IP layer and a physical layer. As a result, without using an MPEG-2 TS format, the IP datagram of mobile service data and IP datagram of signaling information may be transmitted.

[00148] That is, a broadcasting station packetizes the NRT content/files according to a file transfer protocol method as shown in Fig. 4, and then, packetizes them according to an Asynchronous Layered Coding (ALC)/Layered Coding Transport (LCT) method. Then, the packetized ALC/LCT data are packetized according to a UDP method. Then, the packetized ALC/LCT/UDP data is packetized again according to the IP method and becomes ALC/LCT/UDP/IP data. The packetized ALC/LCT/UDP/IP data may be designated as IP datagram for convenience of description in the present invention. At this point, OMA BCAST SG information undergoes the same process as the NRT content/file to constitute IP datagram.

[00149] Additionally, when NRT service signaling information

(for example, SMT) necessary for receiving the NRT content/files is transmitted through a service signaling channel, the service signaling channel is packetized according to a User Datagram protocol (UDP) method, and the packetized UDP data is packetized again according to the IP method to become UDP/IP data. The UDP/IP data may be designated as IP datagram for convenience of description in the present invention. At the time, the service signaling channel is encapsulated in the IP datagram including Well-known IP destination address and well-known destination UDP port number, and is multi-casted according to an embodiment.

[00150] In addition, in relation to OMA BCAST DRM for service protection, a UDP header and an IP header are sequentially added to constitute one IP datagram.

[00151] The IP datagram of the NRT service, NRT service signaling channel, and mobile service data are collected in an adaption layer to generate a RS frame. The RS frame may include IP datagram of OMA BCAST SG.

[00152] The length (i.e., the number of rows) of a column in the RS frame is set by 187 bytes, and the length (i.e., the number of columns) of a row is N bytes (N may vary according to signaling information such as a transmission parameter (or TPC data).

[00153] The RS frame is modulated in a predetermined transmission method in a mobile physical layer (for example, VSB transmission method) and then is transmitted to a receiving system.

[00154] Moreover, whether the NRT service is transmitted is signaled through a PSI/PSIP table. As one example, whether the NRT service is transmitted is signaled to the VCT or TVCT.

[00155] Fig. 5 is a view illustrating a bit stream section of a TVCT table section (VCT) according to an embodiment.

[00156] Referring to Fig. 5, the TVCT table section has a table form of an MPEG-2 private section as one example, but is not limited thereto.

[00157] When the VCT and PID of the audio/video are parsed and then transmitted through the TVCT, the packet identification (PID) information may be obtained.

[00158] Accordingly, the TVCT table section includes a header, a body, and a trailer. A header part ranges from a table_id field to a protocol_version field. A transport_stream_id field is a 16 bit field and represents an MPEG-2 TS ID in a program association table (PAT) defined by a PID value of 0 for multiplexing. In a body part, a num_channels_in_section field is an 8 bit field and represents the number of virtual channels in a VCT section. Lastly, a trailer part includes a CRC_32 field.

[00159] First, the header part will be described as follows.

[00160] A table_id field (8 bits) is set with 0xC8 and identifies that a corresponding table section is a table section constituting TVCT.

[00161] A section_syntax_indicator field (1 bit) is set with 1 and represents that the section follows a general section syntax.

[00162] A private_indicator field (1 bit) is set with 1.

[00163] A section_length field (12 bits) describes that the number of bits remaining in the section to the last of the section from immediately after the section_length field. The value of the section_length field may not be greater than 1021.

[00164] A table_id_extension field (16 bits) may be set with 0x000.

[00165] A version_number field (5 bits) may have 0 and means the version number of VCT.

[00166] A current_next_indicator field (1 bit) represents that a corresponding table section is applicable currently if set with 1.

[00167] A section_number field (8 bits) indicates the number of corresponding table section among TVCT sections. In a first section of TVCT, section_number should be set with 0x00.

[00168] A last_section_number field (8 bits) means the table section of the last and highest number among TVCT sections.

[00169] A protocol_version field (8 bits) is a function that allows a table type delivering parameters having a different structure than one defined in a current protocol. Today, only one valid value of protocol_version is 0. The protocol_version having other than 0 may be used for the future version of the standard in order to recognize another table having a different structure.

[00170] Next, the body part will be described.

[00171] A num_channels_in_section field (8 bits) designates the numbers of virtual channels in the VCT section. The numbers are restricted by a table section length.

[00172] A short_name field (16 bits) represents the name of the virtual channel using 16 bit code value from 1 to 7 sequentially.

[00173] A major_channel_number field (10 bits) represents a major channel number related to a virtual channel defined by repetition in a "for" loop. Each virtual channel should relate to a major channel number and a minor channel number.

The major channel number together with the minor channel number serve as a reference number of a virtual channel of a user.

[00174] A minor_channel_number field (10 bits) represent minor or sub channel numbers ranging from '0' to '999'. This field together with major channel number serves as the second of the number or a channel number of second part representing the right portion. The minor_channel_number is set with 0 if service_type is an analog television. When the service_type is an ATSC_digital_television or an ATSC_audio_only, it uses a minor number ranging from 1 to 99. A value of the minor_channel_number does not overlap that of the major_channel_number in a TVCT.

[00175] A modulation_mode field (8 bits) represents a modulation mode for carrier related to a virtual channel.

[00176] A carrier_frequnecy field (32 bits) has a recommendation value of 0. Although the field is used to identify a carrier frequency, it is not recommended.

[00177] A channel_TSID field (16 bits) is an unsigned integer field representing an MPEG-2 TS ID related to a TS containing an MPEG-2 program, which is reference by a virtual channel in a range from '0x0000' to '0xFFFF'.

[00178] A program_number field (16 bits) identifies an unsigned integer number related to a virtual channel defined in an MPEG-2 program association table (PAT) and a TS program map table (PMT). A virtual channel corresponding to analog service includes program_number of '0xFFFF'.

[00179] An ETM_location field (2 bits) describes the existence and location of an extended text message (ETM).

[00180] An access_controlled field (1 bit) indicates an

access to events related to a virtual channel is controlled once it is set. If the flag is set with 0, an event access is not restricted.

[00181] A hidden field (1 bit) indicates that a user by a direct entry of a virtual channel number cannot access a virtual channel once it is set. A hidden virtual channel is omitted when a user surfs a channel, and is shown when the user accesses undefined or direct channel entry. A typical application of a hidden channel is a test signal and NVOD service. The hidden channel and its events may be shown on an EPG display according to a state of a hide_guide bit.

[00182] A hidden_guide field allows a virtual channel and its events to be displayed on an EPG display once it is set with 0 for a hidden channel. The bit is not related to a channel having no hidden bit set and thus non-hidden channels and their events are always displayed on an EPG display regardless of a state of a hide_guide bit. A typical application of a hidden channel, in which a hidden_guide bit set is set with 1, is a test signal and service easily obtainable through an application level pointer.

[00183] A service_type field (6 bits) represents a type of service transmitted from a virtual channel. Figs. 6 and 7 are views illustrating how to define a value of a service_type field according to an embodiment. According to an embodiment, a service_type value (i.e., '0x04') shown in Fig. 6 means that service_type is ATSC_data_only_service and NRT service is transmitted through a virtual channel. According to another embodiment, a service_type value (i.e., '0x08') shown in Fig. 7 means that service_type is ATSC_nrt_service and a virtual channel provides NRT service satisfying the ATSC

standard.

[00184] A source_id field (16 bits) represents the source of a program related to a virtual channel.

[00185] A descriptors_length field represents the total length (byte unit) of a descriptor for the following virtual channel.

[00186] A descriptor() field includes at least zero descriptor.

[00187] An additional_descriptors_length field represents a total length (byte unit) of the following VCT descriptor.

[00188] Lastly, in relation to the trailer part, a CRC_32 field is a 32 bit field and includes a cyclic redundancy check (CRC) value, which ensures zero output from registers of a decoder defined in an MPEG-2 system after processing an entire STT section.

[00189] Fig. 8 is view of data_service_table_section) for identifying an application of NRT service and bit stream syntax of data_service_table_bytes in a DST section. A broadcasting station NRT service data or NRT service signaling data, satisfying ASTC standard, may be transmitted through the DST table section of Fig. 8.

[00190] Hereinafter, semantic of fields including a data_service_table_section structure is as follows.

[00191] A table_id field (8 bits) as a field for type identification of a corresponding table section is a table section in which a corresponding table section constitutes DST through this field. For example, a receiver identifies that a corresponding table section is a table section constituting DST if a value of the field is 0XCF.

[00192] A section_syntax_indicator field (1 bit) is an

indicator defining a section format of DST, and the section format may be short-form syntax (0) of MPEG, for example.

[00193] A private_indicator field (1 bit) represents whether the format of a corresponding section follows a private section format and may be set with 1.

[00194] A private_section_length field (12 bits) represents a remaining table section length after a corresponding field. Additionally, a value of this field does not exceed '0xFFD'.

[00195] A table_id_extension field (16 bits) is dependent on a table, and may be a logical part of a table_id field providing a range of the remaining fields.

[00196] A version_number field (5 bits) represents the version number of DST.

[00197] A current_next_indicator field (1 bit) indicates whether a transmitted DST table section is applicable currently. If the field value is 0, it means that there is no table yet and the next table is valid.

[00198] A section_number field (8 bits) represents a section number in sections in which a corresponding table section constitutes a DST table. section_number of the first section in DST is set with '0x00'. The section_number is increased by one as the section of DST is increased.

[00199] A last_section_number field (8 bits) represents the last section number constituting a DST table, i.e., the highest section_number.

[00200] data_service_table_bytes represents a data block constituting DST, and its detailed structure will be described below.

[00201] A CRC_32 field is a 32 bit field and includes a cyclic redundancy check (CRC) value, which ensures zero

output from registers of a decoder defined in an MPEG-2 system after processing an entire DST section.

[00202] Hereinafter, semantic of fields including a data_service_table_bytes structure is as follows.

[00203] An sdf_protocol_version field (8 bits) describes the version of a Service Description Framework protocol.

[00204] An application_count_in_section field (8 bits) represents the number of applications listed in a DST section.

[00205] A compatibility_descriptor() field represents that a corresponding structure includes a DSM-CC compatible descriptor. Its purpose is to signal compatible requirements of an application in a receiving platform in order to use a corresponding data service after determining its ability.

[00206] An app_id_byte_length field (16 bits) describes the number of bytes used for identifying an application.

[00207] An app_id_description field (16 bits) describes the format and semantic of the following application identification bytes. For example, a value of an app_id_description may be defined as Table 1.

[00208] [Table 1]

Value	Application Identifier Format
0x0000	DASE application
0x0001-0x7FFF	ATSC reserved
0x8000-0xFFFF	User private

[00209]

[00210] An app_id_byte field (8 bits) represents a byte of an application identifier.

[00211] A tap_count field (8 bits) describes the number of

Tap() structures used for corresponding application.

[00212] A protocol_encapsulation field (8 bits) describes a protocol encapsulation type used for transmitting a specific data element referenced by a Tap() field. A value of the protocol_encapsulation field is defined as Table 2.

[00213] [Table 2]

Value	**Encapsulated Protocol**
0x00	Not in a MPEG-2 Transport Stream
0x01	Asynchronous non-flow controlled scenario of the DSM-CC Download protocol encapsulated in DSM-CC sections
0x02	Non-streaming Synchronized Download protocol encapsulated in DSM-CC sections
0x03	Asynchronous multiprotocol datagrams in Addressable Sections using LLC/SNAP header
0x04	Asynchronous IP datagrams in Addressable Sections
0x05	Synchronized streaming data encapsulated in PES
0x06	Synchronous streaming data encapsulated in PES
0x07	Synchronized streaming multiprotocol datagrams in PES using LLC/SNAP header
0x08	Synchronous streaming multiprotocol datagrams in PES using LLC/SNAP header
0x09	Synchronized streaming IP datagrams in PES
0x0A	Synchronous streaming IP datagrams in PES
0x0B	Proprietary Data Piping
0x0C	SCTE DVS 051 asynchronous protocol [19]
0x0D	Asynchronous carousel scenario of the DSM-CC Download protocol encapsulated in DSM-CC sections
0x0E	Reserved for harmonization with another standard body
0x0F-0x7F	ATSC reserved
0x80-0xFF	User defined

[00214]

[00215] An action_type field (7 bits) represents attribute of data referenced by a Tap().

[00216] A resource_location field (1 bit) describes a position of an association_tag field matching to an association tag value listed in the next Tap structure. When a corresponding field is set with 0, association_tag exists in PMT of a current MPEG-2 program. Like this, when the corresponding field is set with 1, a matching association_tag exits in DSM-CC Resource Descriptor in a Network Resources Table of a corresponding data service.

[00217] A Tap() field may include information on searching a data element of an application state in a communication channel of a lower layer. An association_tag field in a Tap() field may include correspondence information between data elements of an application state. A value of an association_tag field in one Tap structure corresponds to a value of an association_tag field of one association tag descriptor in a current PMT. For example, a Tap() field may have a specific structure including fields of Table 3.

[00218] [Table 3]

Syntax	**No. of bits**	**Format**
Tap () {		
tap_id	16	uimsbf
use	16	uimsbf
association_tag	16	uimsbf
selector()		
}		

[00219]

[00220] A tap_id field (16 bits) is used by an application to identify data elements. A value of tap_id has a range defined by values of app_id_byte fields related to Tap() in DST. A tap_id value is selected by a data service provider. Additionally, the tap_id value may be used for application to deal with a data element.

[00221] A Use field (16 bits) is used to specify a communication channel referenced by association_tag.

[00222] An association_tag field (16 bits) uniquely identifies one of a DSM-CC resource descriptor listed in a Network Resource Table or data elementary stream listed in PMT. A value of a corresponding field may be identical to an association_tag value of association_tag_descriptor.

[00223] A Selector() field describes a specific data element available in a communication channel or data elementary stream referenced by the association_tag field. Additionally, the selector structure may indicate a protocol required for a corresponding data element.

[00224] A tap_info_length field (16 bits) describes the number of bytes of descriptors in the next of a corresponding field.

[00225] A descriptor() field may include descriptor information according to a corresponding descriptor format.

[00226] An app_info_length field (8 bits) describes the number of bytes of the next descriptors of a corresponding field.

[00227] A descriptor() field may include descriptor information according to a corresponding descriptor format.

[00228] An app_data_length field (16 bits) describes the length of a byte unit of app_data_byte fields.

[00229] An app_data_byte (8 bits) field represents input parameters related to application and other private data fields in 1 byte.

[00230] A service_info_length field (8 bits) describes the number of byte units of the next descriptor.

[00231] A descriptor() field may include descriptor information according to a corresponding descriptor format.

[00232] A service_private_data_length field (16 bits) describes the length of a byte unit in private fields.

[00233] A service_private_data_byte field (8 bits) represents a private field in 1 byte.

[00234] Fig. 9 is a view illustrating a method of receiving and providing NRT service in a receiving system by using ATSC A/90 standard for transmitting data broadcasting stream and ATSC A/92 standard for transmitting IP multicast stream.

[00235] That is, information on stream constituting each virtual channel is signaled to service location descriptor of VCT or ES_loop of PMT. For example, as shown in Fig. 7 or 8, if VCT service type is 0x02(i.e., digital A/V/Data), 0x04(i.e., Data only), or 0x08(i.e., NRT Only service), NRT service stream may be transmitted to the virtual channel. At this point, if 0x95(i.e., DST transmission) is allocated to a stream_type field value in a service location descriptor (or ES loop of PMT), it means that broadcast is transmitted. If the stream_type field value has no value or is not 0x95, only typical A/V is transmitted. That is, if the stream_type field value in service location descriptor has 0x95, an Elementary_PID field value at this point is a PID value of a Data Service Table (DST). Accordingly, DST may be received through the Elementary_PID.

[00236] Through the DST, types of application and detailed information on data broadcasting stream transmitted through the channel may be obtained. The DST is used to identify NRT application (i.e., NRT service).

[00237] That is, the App_id_descrption field of DST defines the format and interpretation of the following application identification bytes. According to an embodiment, '0x0003' is allocated to the App_id_descrption field to identify NRT application. The above numerical value is just one example, and does not restrict the range of the rights of the present invention.

[00238] If the App_id_descrption field value is '0x0003', the next following Application_id_byte value becomes a Service ID value of the NRT application. A service ID for the NRT application may have a URI value uniquely identifying a corresponding service around the world.

[00239] After the NRT application is identified, PID of an MPEG-2 TS packet divided from the IP datagram of an NRT service signaling channel is searched through Tap information. Then, IP datagram transmitting a NRT service signaling channel may be obtained from MPEG-2 TS packets having PID obtained through the tap information, and NRT service signaling data may be obtained from the obtained IP datagram. At this point, the IP access information of the NRT service signaling channel may be well-known IP access information, i.e., well-known IP address and well-known UDP port number.

[00240] That is, if the Protocol_encapsulation field value in the DST is 0x04, asynchronous IP stream is transmitted, and if the Selector_type field value is 0x0102, a device_id value indicating destination address may be delivered through

selector_bytes. multiprotocol_encaplsulation_descriptor is used to accurately interpret the selector_bytes value and the number of valid bytes in the device_id value is signaled. As a result, through the Tap information, an IP Multicast address (or address range) of the NRT service signaling channel, transmitted to the corresponding PID, is obtained.

[00241] Accordingly, a receiver accesses the Multicast address (or address range) to receive IP stream, i.e., IP packet, and then, extracts NRT service signaling data from the received IP packet.

[00242] Then, the receiver receives NRT service data, i.e., NRT content item/files to store them in a storage medium or display them on a display device, on the basis of the extracted NRT service signaling data.

[00243] According to another embodiment, a Stream Type field value of DST may have new 0x96 instead of 0x95 to signal NRT service. This is because NRT service, i.e., new application, may malfunction when a typical receiver determines whether there is data broadcasting stream only on the basis of whether there is stream having a stream type of 0x95. In this case, with designating a stream newly, a typical receiver may disregard it to guarantee backwards compatibility.

[00244] Figs. 10 and 11 are views illustrating a method of receiving NRT service by using DSM-CC addressable section data according to another embodiment.

[00245] A data transmission method using DST is a standard for transmitting all kinds of IP datagram through digital broadcasting stream, and may be inefficient for the NRT service. Accordingly, Figs. 10 and 11 illustrate a method of receiving the NRT service by signaling the PID of a specific

stream including IP address information and section data of the IP datagram with respect to the NRT service through the data of the DSM-CC addressable section.

[00246] As shown in Fig. 10, the receiver may obtain information that NRT service stream is transmitted through the virtual channel when a service type of VCT (or TVCT) is 0x08 (i.e., NRT Only service). That is, the receiver may obtain information on whether there is NRT service according to service_type information by mapping the PID of a virtual channel into a channel number.

[00247] At this point, if 0x0D is allocated to a stream_type field value in service location descriptor of VCT (or ES loop of PMT), it means that DSM-CC stream is transmitted. An Elementary_PID field value at this point may be the PID value of a DSM-CC addressable section. Accordingly, the receiver receives a DSM-CC addressable section including NRT service data through Elementary_PID.

[00248] That is, the receiver may obtain the PID of the DSM-CC addressable section through VCT or PMT. Here, the receiver may obtain an NRT_IP_address_list_descriptor_A() field including an IP address of an NRT service signaling channel or an IP address of the FLUTE session for transmitting NRT service data, which corresponds to the PID obtained from PMT of the corresponding stream.

[00249] Moreover, the receiver may receive DSM-CC addressable section data from IP multicast stream or IP subnet on the basis of the IP address obtained from an NRT_IP_address_list_descriptor_A() field. The receiver may obtain a corresponding IP datagram including a specific NRT service (for example, A, B, or C) data by searching a DSM-CC

addressable section having PID corresponding to the obtained elementary_PID from the received DSM-CC addressable section data.

[00250] Fig. 11 is a view illustrating a method of signaling a DSM-CC addressable section data by using VCT according to another embodiment.

[00251] As mentioned above, the receiver may obtain information that NRT service stream may be transmitted when a service_type in VCT is 0X02, 0X04 of 0X08. Also, the receiver may obtain elementary_PID having a stream type of 0X0D from the service_location_descriptor() field to receive the DSM-CC stream. Here, the receiver may obtain an NRT_IP_address_list_descriptor_B() field including an IP address of an NRT service signaling channel or an IP address of the FLUTE session for transmitting NRT service data, which corresponds to the obtained elementary_PID.

[00252] Moreover, the receiver may receive DSM-CC addressable section data from IP multicast stream or IP subnet on the basis of the IP address obtained from an NRT_IP_address_list_descriptor_B() field. The receiver may obtain the IP datagram including specific NRT service (for example, A, B, or C) that it wants to receive from the received DSM-CC addressable section data by parsing the DSM-CC addressable section having PID corresponding to the obtained elementary_PID.

[00253] The processes for extracting NRT service signaling data and NRT service data are described as follows. Here, 0x08 is allocated to the service_type field value in VCT, and indicates that at least one NRT service is transmitted to a corresponding virtual channel.

[00254] That is, when the receiver is turned on and a channel is selected by default or a user through a tuner, the PSI/PSIP section handler obtains VCT and PMT from a broadcast signal received through the selected channel. Also, the PSI/PSIP section handler parses the obtained VCT to confirm whether there is NRT service. This is confirmed by checking the service_type field value in a virtual loop of the VCT. For example, when the service_type field value is not 0x08, the corresponding virtual channel does not transmit NRT service. At this point, since the virtual channel transmits existing service (i.e., legacy ATSC service), the receiver operates properly according to information in the virtual channel.

[00255] Additionally, in relation to a demultiplexing unit, if a service_type field value is 0x08 according to a control of a service manager, a corresponding virtual channel transmits NRT service. In this case, PID of DST is extracted by parsing a service location descriptor in a virtual channel loop of the VCT. Moreover, DST is received by using the extracted PID.

[00256] Moreover, the receiver confirms whether a corresponding service provided through a channel selected from the received DST is NRT service.

[00257] The NRT service is confirmed by an App_id_descrption field value.

[00258] According to an embodiment, '0x0003' is allocated to the App_id_descrption field to identify NRT application. The above numerical value is just one example, and does not restrict the range of the rights of the present invention.

[00259] If the App_id_descrption field value in the DST is

'0x0003', the next following Application_id_byte value becomes a Service ID value of the NRT application (i.e., NRT service). Therefore, the service manager or PSI/PSIP section handler extracts Tap() to PID of an MEGP-2 TS packet separated from the IP datagram of the NRT service signaling channel after identifying the NRT application (i.e., NRT service). Then, stream PID including association_tag of the extracted Tap is extracted from PMT.

[00260] Also, the addressable section handler may recover the DSM-CC addressable section by removing decapsulation, i.e., an MPEG-2 header, after receiving MPEG-2 TS packets corresponding to the extracted stream PID.

[00261] Then, the receiver recovers the IP datagram transmitting an NRT service signaling channel by removing a section header and CRC checksum from the DSM-CC addressable section and obtains NRT service signaling data from the recovered IP datagram. Here, access information on the IP datagram transmitting the NRT service signaling channel is a well-known destination IP address and a well-known destination UDP port number.

[00262] That is, if the Protocol_encapsulation field value in the DST is 0x04, asynchronous IP stream is transmitted, and if the Selector_type field value is 0x0102, a device_id value indicating a destination address may be delivered through selector_bytes. multiprotocol_encaplsulation_descriptor is used to accurately interpret the selector_bytes value and the number of valid bytes in the device_id value is signaled. As a result, through the Tap information, an IP Multicast address (or address range) of the NRT service signaling channel, transmitted to the corresponding PID, is obtained.

[00263] Accordingly, a receiver accesses the Multicast address (or address range) to receive IP stream, i.e., IP packet, and then, extracts NRT service signaling data from the received IP packet.

[00264] The receiver receives NRT service data, i.e., NRT content item/files to store them in a storage medium or display them on a display device, on the basis of the extracted NRT service signaling data.

[00265] Moreover, the NRT service may be provided Dynamic Content Delivery (DCD) service according to an embodiment. The DCD service is service for transmitting content to a receiver periodically or at the user request, and the content is selected from a server according to receiver information. The DCD service supports a point-to-point method and a broadcast method in a communication means for content delivery, and the above NRT service is transmitted through an OMA BCAST method and one of the broadcast methods of the DCD service.

[00266] NRT service data may be transmitted through the DCD service of the OMA BCAST method. In this case, the receiver may obtain the DCD channel information to receive NRT service and may receive the NRT service through a corresponding DCD channel on the basis of the DCD channel information.

[00267] Moreover, the DCD channel information may be included in the NST and transmitted. For example, the receiver receives NST, and obtains DCD channel information through DCD bootstrap.

[00268] Additionally, the NST may include DCD channel metadata, received through a DCD administrative channel, for signaling of the DCD channel information. Accordingly, the

receiver may obtain information on a channel for receiving NRT service and metadata through NST.

[00269] Accordingly, when NST including DCD channel information is transmitted, the receiver accesses the DCD channel through NST without transmission of the NRT service signal data, and then receives the NRT service.

[00270] Like this, if NST includes metadata of a channel for receiving NRT service, there are several advantages.

[00271] First, without receiving the NRT service signaling data on the basis of the service type of a virtual channel, service access speed may be increased by receiving channel metadata that directly receives NRT service from NST.

[00272] Additionally, update signaling for a channel change item may be performed in real time in a broadcast environment.

[00273] Moreover, access information in OMA BCAST SG may be obtained by referring to NST. For example, the receiver receives DCD channel meta data on the basis of the DCD channel information in NST, and obtains access information to receive NRT service on the basis of the NRT service signaling data and DCD channel metadata obtained from NST.

[00274] Lastly, NST including a list of NRT service related to another virtual channel may be transmitted. Accordingly, list information of the NRT service may be transmitted through a specific NRT service signaling channel on an IP layer not on a PSI or PSIP layer. Accordingly, in this case, backwards compatibility to PSI or PSIP may be reserved.

[00275] In addition, as mentioned above, the DCD channel information including the DCD channel metadata may be included in the access information of SG in OMA BCAST, and the access information corresponds to the NRT service

information in NST. In more detail, the receiver may obtain NRT service information in NST from an access fragment of OMA BCAST SG. Accordingly, the receiver may obtain information on receiving NRT service by receiving NST corresponding to the obtained NRT service information.

[00276] Moreover, the NRT service transmitted through the DCD channel may be divided by a service category allocated. For example, the service category of the NRT service transmitted through the DCD channel may be identified by 0X0F.

[00277] Figs. 12 and 13 are views illustrating a bit stream syntax of NST according to an embodiment.

[00278] Here, the corresponding syntax is created in an MPEG-2 private section format to help understanding, but the format of the corresponding data may vary. For example, the corresponding data may be expressed in a Session Description Protocol (SDP) format and signaled through a Session Announcement Protocol (SAP) according to another method.

[00279] NST describes service information and IP access information in a virtual channel for transmitting NST, and provides NRT broadcast stream information of a corresponding service by using an identifier of the NRT broadcast stream, i.e., NRT_service_id, in each service. Furthermore, the NST describes description information of each fixed NRT service in one virtual channel, and a descriptor area may include other additional information.

[00280] A table_id field (8 bits) as a field for type identification of a corresponding table section is a table section in which a corresponding table section constitutes NST through this field.

[00281] A section_syntax_indicator field (1 bit) is an

indicator defining a section format of NST, and the section format may be short-form syntax (0) of MPEG, for example.

[00282] A private_indicator field (1 bit) represents whether the format of a corresponding section follows a private section format and may be set with 1.

[00283] A section length field (12 bits) represents a remaining table section length after a corresponding field. Additionally, a value of this field does not exceed '0xFFD'.

[00284] A table_id_extension field (16 bits) is dependent on a table, and may be a logical part of a table_id field providing a range of the remaining fields. Here, a table_id_extension field includes an NST_protocol_version field.

[00285] The NST_protocol_version field (8 bits) shows a protocol version for notifying that NST transmits parameters having a different structure than other defined in a current protocol. Currently, this field value is 0. If the field value is designated with other than 0 later, it is for a table having a different structure.

[00286] A version_number field (5 bits) represents the version number of NST.

[00287] A current_next_indicator field (1 bit) indicates whether a transmitted NST table section is applicable currently. If the field value is 0, it means that there is no table yet and the next table is valid.

[00288] A section_number field (8 bits) represents a section number in sections in which a corresponding table section constitutes a NST table.

[00289] section_number of the first section of an NRT Service Table (NST) is set with '0x00'. The section_number is

increased by one each time a section of the NST is increased.

[00290] A last_section_number field (8 bits) represents the last section number constituting a NST table, i.e., the highest section_number. (Highest section_number)

[00291] A carrier_frequnecy field (32 bits) notifies a transmission frequency corresponding to a channel.

[00292] A channel_TSID field (16 bits) means a unique channel identifier of broadcast stream in which a corresponding NST section is currently transmitted.

[00293] A program_number field (16 bits) represents the number of a program related to a virtual channel.

[00294] A source_id field (16 bits) represents the source of a program related to a virtual channel.

[00295] A num_NRT_services field (8 bits) represents the number of NRT services in an NST section.

[00296] Additionally, NST provides information on a plurality of fixed NRT services by using a 'for' loop. Hereinafter, the same field information may be provided to each fixed NRT service.

[00297] An NRT_service_status field (2 bits) identifies a state of a corresponding mobile service. Here, MSB indicates whether a corresponding mobile service is active (1) or inactive (0), and whether the corresponding mobile service is hidden (1) or not (0). Here, if the mobile service is NRT service, a state of the corresponding NRT service is identified. Hidden service is mainly used for exclusive application and a typical receiver disregards it.

[00298] A SP_indicator field (1 bit) is a field representing service protection if the service protection applied to at least one of components necessary for providing meaningful

presentation of a corresponding mobile service is set.

[00299] A CP_indicator field (1 bit) represents whether content protection of a corresponding NRT service is set. If the CP_indicator field value is 1, it means that the content protection is applied to at least one of components required to provide a meaningful presentation of a corresponding NRT service.

[00300] An NRT_service_id field (16 bits) is an indicator that uniquely identifies a corresponding NRT service in a range of a corresponding NRT broadcast. The NRT_service_id is not changed during the corresponding service. Here, if the service is terminated, in order to evade confusion, NRT_service_id for the service may not be used for another service until an appropriate time elapses.

[00301] A Short_NRT_service_name field (8*8 bits) displays a short name of the NRT service. If there is no short name of the NRT service, the field may be filled with a null value (for example, 0x00).

[00302] An NRT_service_category field (6 bits) identifies a type of service in the corresponding NRT service.

[00303] A num_components field (5 bits) displays the number of IP stream components in the NRT service.

[00304] If an IP_version_flag field (1 bit) is set with 0, it indicates that a source_IP_address field, an NRT_service_destination_IP_address field, and a component_destination_IP_address field are IPv4 addresses. If set with 1, a source_IP_address field, an NRT_service_destination_IP_address field, and a component_destination_IP_address field are IPv6 addresses.

[00305] A source_IP_address_flag field (1 bit) indicates when

a flag is set that there is a source IP address value for corresponding NRT service to indicate source specific multicast.

[00306] An NRT_service_destination_IP_address_flag field (1 bit) indicates when a flag is set with 1 that there is an NRT service destination IP address field for providing a default IP address for components of a corresponding NRT service.

[00307] In relation to a source_IP_address field (128 bits), there is a corresponding field if source_IP_address_flag is set with 1, but there is no corresponding field if set with 0. If there is a corresponding field, the corresponding field includes a source IP address of all IP datagram transmitting components of the corresponding NRT service. A restricted use of a 128 bit long address of a corresponding field is for future use of IPv6, which is not currently used though. Source_IP_address becomes a source IP address of the same server transmitting all channels of a FLUTE session.

[00308] In relation to an NRT_service_destination_IP_address field (128 bits), if source_IP_address_flag is set with 1, there is a source_IP_address field, but if source_IP_address_flag is set with 0, there is no corresponding source_IP_address field. If there is no corresponding source_IP_address field, a component_destination_IP_address field exists for each component in a num_components loop. A restricted use of a 128 bit long address of a corresponding source_IP_address field is for future use of IPv6, which is not currently used though. NRT_service_destination_IP_Address is signaled if there is a destination IP address of a session level of the FLUTE

session.

[00309] Additionally, NST provides information on a plurality of components by using a 'for' loop. An essential_component_indicator field (1 bit) indicates when a value of a corresponding value is set with 1 that a corresponding component is a necessary component for NRT service. If not, the corresponding component is a selected component.

[00310] A port_num_count field (6 bits) indicates numbers of UDP ports related to a corresponding UDP/IP stream component. Values of the destination UDP port numbers are increased by one, starting from a component_destination_UDP_port_num field value.

[00311] A component_destination_IP_address_flag field (1 bit) is a flag representing that there is a component_destination_IP_address field for corresponding component if set with 1.

[00312] In relation to component_destination_IP_address field (128 bits), if component_destination_IP_address_flag is set with 1, there is corresponding field, but if component_destination_IP_address_flag is set with 0, there is no corresponding field. If there is a corresponding field, the corresponding field includes a source IP address of all IP datagram transmitting components of the corresponding NRT service. A restricted use of a 128 bit long address of a corresponding field is for future use of IPv6, which is not currently used though.

[00313] A component_destination_UDP_port_num field (16 bits) represents a destination UDP port number for corresponding UDP/IP stream component.

[00314] A num_component_level_descriptors field (4 bits) provides the number of descriptors providing additional information on corresponding IP stream component.

[00315] A component_level_descriptors field identifies at least one descriptor providing additional information on a corresponding IP stream component.

[00316] A num_NRT_service_level_descriptors field (4 bits) represents the number of NRT service level descriptors for corresponding service.

[00317] NRT_service_level_descriptor() identifies no or at least one descriptor providing additional information on corresponding NRT service. Here, a specific service type for NRT service may be provided. The specific service type includes a portal service providing web content, push VOD, and A/V download.

[00318] A num_virtual_channel_level_descriptors field (4 bits) describes the number of virtual channel level descriptors for a corresponding virtual channel.

[00319] virtual_channel_level_descriptor() represents a descriptor providing additional information on a virtual channel that a corresponding NST describes.

[00320] Moreover, NRT service is transmitted through FLUTE, and access information on the NST table is connected to FLUTE session information as follows.

[00321] Source_IP_address is a source IP address of the same server transmitting all channels of the FLUTE session.

[00322] NRT_service_destination_IP_Address is signaled if there is a destination IP address of a session level of the FLUTE session.

[00323] A component may be mapped into a channel in the FLUTE

session, and an additional destination IP address (which is different from an IP address signaled by session) is signaled through component_destination_IP_address at each channel.

[00324] Additionally, a destination port number is signaled through component_destination_UDP_port_num and the number of destination ports starting from component_destination_UDP_port_num may be additionally designated through port_num_count.

[00325] A plurality of channels may be configured for one destination IP address by designating a port in plurality. Here, one component designates a plurality of channels. However, it is desired to identify a channel through a destination IP address in general. Here, one channel is typically mapped into one component.

[00326] Content items/files for NRT service are transmitted through FLUTE, and corresponding FLUTE session information is signaled using access information on the NST table.

[00327] Fig. 14 is a view illustrating a bit stream syntax of NRT_component_descriptor (MH_component_descriptor) according to an embodiment.

[00328] NRT_component_descriptor() is shown in a component descriptor loop in each component of each NRT service in NST. Then, all parameters in a corresponding descriptor correspond to parameters used for components of NRT service.

[00329] Hereinafter, each field information transmitted through the NRT_component_descriptor of Fig. 14 will be described as follows.

[00330] A component_type field (7 bits) identifies an encoding format of a component. The identification value may be one of values allocated for payload_type of a RTP/AVP

stream. Additionally, the identification value may be a dynamic value ranging from 96 to 127. Values of the field for components constituting media transmitted through RTP are identical to those in payload_type in an RTP header of IP stream transmitting a corresponding component.

[00331] An adding value of a component type field in a range of 43 to 71 will be defined in the future version of the standard. When NRT service stream is transmitted based on FLUTE, in order to additionally signal parameters (described below) necessary for FLUTE session, 38 (which is component_type defined for a FLUTE component in ATSC) may be used, or 43 (i.e., an unallocated value) may be defined as component_type for new NRT transmission, and used.

[00332] A num_STKM_streams field (8 bits) identifies numbers of STKM streams related to a corresponding component.

[00333] A STKM_stream_id field (8 bits) identifies STKM stream having keys in order to decrypt the obtained corresponding protected component. Here, the STKM_stream_id field in the component descriptor for the STKM stream is referred.

[00334] An NRT_component_data (component_type) field provides at least one of encoding parameters necessary for expressing a corresponding component and other parameters. Here, a structure of an NRT_component_data element is determined by a value of a component_type field.

[00335] A File Delivery Table (FDT) of FLUTE sessions is used for delivering item lists of all content items, and provides sizes, data types, and other information of items related to obtain the items.

[00336] Accordingly, the present invention obtains

information for accessing the FLUTE session transmitting a corresponding content by using NST, in order to receive a selected content from SG obtained by using NRT-IT. Moreover, the present invention maps information in a file transmitted through a corresponding FLUTE session into information on a content item of NRT-IT. In this case, identification of service including the selected content item is resolved through NRT_service_id of the NST.

[00337] NRT service is transmitted through FLUTE, and access information on the NST table is connected to FLUTE session information as follows.

[00338] Source_IP_address is a source IP address of the same server transmitting all channels of the FLUTE session.

[00339] NRT_service_destination_IP_Address is signaled if there is a destination IP address of a session level of the FLUTE session.

[00340] A component may be mapped into a channel in the FLUTE session, and an additional destination IP address (which is different from an IP address signaled by session) is signaled through component_destination_IP_address at each channel. Additionally, a destination port number is signaled through component_destination_UDP_port_num and the number of destination ports starting from component_destination_UDP_port_num may be additionally designated through port_num_count.

[00341] A plurality of channels may be provided to one destination IP address by designating a plurality of ports, and in such a case, one component designates a plurality of channels. However, it is recommended that a channel be distinguished through a destination IP address, and in such a

case, one channel is mapped into one component.

[00342] component_attribute_byte may be used to signal an additional attribute of a component constituting a session. Additional parameters necessary for signaling a FLUTE session may be signaled through this.

[00343] In this regard, parameters for signaling the FLUTE session are required, and include definitely necessary required parameters and optional necessary parameters related to a corresponding FLUTE session. Firstly, the definitely necessary parameters include parameters such as a source IP address, the number of channels in the session, the destination IP address and port number for each channel in the session, the Transport Session Identifier (TSI) of the session, and the start time and end time of the session. The optional necessary parameters related to a corresponding FLUTE session include parameters such as FEC Object Transmission Information, some information that tells receiver in the first place, that the session contains files that are of interest and bandwidth specification.

[00344] The number of channels in the session may be explicitly provided, or may be obtained by adding up the number of streams constituting the session. Through the NST and component_descriptor, parameters such as start time and end time of the session, source IP address, destination IP address and port number for each channel in the session, Transport Session Identifier (TSI) of the session, and number of channels in the session may be signaled.

[00345] Fig. 15 is a view illustrating a bit stream syntax of NRT component descriptor including NRT_component_data according to an embodiment.

[00346] One NRT service may be included in multiple FLUTE sessions. Each session may be signaled using at least one NRT component descriptors depending on IP addresses and ports used for the session.

[00347] Hereinafter, each field of NRT_component_data will be described as follows.

[00348] A TSI field (16 bits) represents TSI of a FLUTE session.

[00349] A session_start_time field indicates a start time of the FLUTE session. If all values of the corresponding fields are 0, it means that a session started already.

[00350] A session_end_time field indicates an end time of the FLUTE session. If all values of the corresponding fields are 0, it means that a session continues infinitely.

[00351] A tias_bandwidth_indicator field (1 bit) indicates flags including Transport Independent Application Specific (TIAS) bandwidth information. If it indicates that the TIAS bandwidth field exists, a corresponding bit is set with 1, and if it indicates that the TIAS bandwidth field does not exist, the corresponding bit is set with 0.

[00352] In relation to an as_bandwidth_indicator field (1 bit), flags include Application Specific (AS) bandwidth information. If it indicates that the AS bandwidth field exists, a corresponding bit is set with 1, and if it indicates that the AS bandwidth field does not exist, the corresponding bit is set with 0.

[00353] An FEC_OTI_indicator field (1 bit) represents whether FEC object transmission information (OTI) is provided.

[00354] A tias_bandwidth field represents a TIAS maximum bandwidth.

[00355] An as_bandwidth field has an AS maximum bandwidth value.

[00356] An FEC_encoding_id field represents FEC encoding ID used in the corresponding FLUTE session.

[00357] An FEC_instance_id field represents FEC instance ID used in the corresponding FLUTE session.

[00358] Provided is a method of providing all Information necessary for receiving the FLUTE session by signaling the same parameters as above through FLUTE component data bytes, and of receiving files by obtaining information on all the files delivered through the FLUTE session that uses FDT received through the session.

[00359] This FLUTE component descriptor may be delivered through a Component_level_descriptor loop of NST. If the FLUTE channel is in plurality, since TSI and session_start_time, session_end_Time, i.e., parameters of a session level, should be signaled once, a FLUTE component descriptor may be transmitted only in one of components in several channels through a Component_level_descriptor loop.

[00360] Fig. 16 is a view illustrating a bit stream syntax of NRT-IT section for signaling NRT application according to an embodiment.

[00361] Information provided from NRT-IT includes a title of content (for example, a name of downloadable program), download available time and information, content advisories, caption service availability, content identification, and other metadata. One item of content may include at least one file. For example, an audio/video clip may be played in a JPEG thumbnail image used for displaying a screen.

[00362] An instance of NRT-IT may include data corresponding

to an arbitrarily predetermined period, or may describe a NRT content starting at a predetermined time and ends at the indefinite future. Each NRT-IT represents a start time and a duration period that may be indefinite. Each NRT-IT instance may be divided into 256 sections. Each section includes information on a plurality of content items. Information of a specific content item cannot be divided and stored in at least two sections.

[00363] The downloadable content item, which is more extended than a period that at least one NRT-IT instance takes, is the first of NRT-IT. The content item description is stored in NRT_information_table_section () in an availability order. Accordingly, when a value of last_section_number is greater than 0 (it means that NRT-IT is transmitted to a plurality of sections), all content item description in a specific section not the first section may have the same as or higher availability than the content item description of the next section.

[00364] Each NRT-IT identifies an NRT service related to a specific value of a valid service_id in a specific virtual channel during the period.

[00365] A table_id field (8 bits) is set with 0xTBD to identify a table section that a corresponding table section constitutes NRT-IT.

[00366] A service_id field (16 bits) describes a service_id field related to NRT service showing a content item that the section describes.

[00367] An NRT_IT_version_number field (5 bits) is defined as a set in at least one NRT_content_table_section() having a common value with respect to service_id,

current_next_indicator, protocol_version, and time_span_start fields. It identifies a version number of an NRT-IT instance. The version number is increased by 1 modulo 32 when a field of NRT-IT instance is changed.

[00368] A current_next_indicator field (1 bit) represents that a corresponding table section is applicable currently if set with 1.

[00369] A protocol_version field (8 bits) is set with 0. A function of protocol_version allows a table type having parameters in the future, which has a different structure than those defined in the current protocol. Currently, only one valid value of protocol_version is 0. A value other than 0 in protocol_version is used for the future version of standard to recognize other tables having different structures.

[00370] A time_span_start field (32 bits) represents a start time of an instance period represented in GPS sec from 00:00:00 UTC, January 6, 1980. A time of day of time_span_start is set to 00 min of the time. A value 0 of time_span_start represents a period of an NRT-IT instance starting from a negative past. A value of time_span is identical at each section of multi-sectioned NRT-IT instance. Values of time_span_start and time_span_length are set not to overlap another NRT-IT instance of an IP subnet at a specified period.

[00371] A time_span_length field (11 bits) identifies a number of min starting at the time recognized at time_span_start that the instance covers. Once it is set, a value of time_span_length does not change in a value of time_span_start. If a value of time_span_length is 0, an NRT-

IT instance covers an entire time starting from time_span_start at the indefinite future. When a value of time_span_start is 0, there is no meaning in time_span_length.

[00372] A value of time_span_start is identical at each section of multi-sectioned NRT-IT instance. Values of time span start and time span length are set not to overlap another NRT-IT instance of an IP subnet at a specified period.

[00373] A num_items_in_section field (8 bits) represents the number of content items described in an NRT-IT section.

[00374] A content_linkage field (16 bits) represents an identification number within a range from 0x0001 to 0xFFFF. 0x0000 is not used. content_linkage is a linkage function for two: this links at least one file of FLUTE FDT related to NRT service with metadata of NRT-IT and forms TF_id (identifier for Text Fragement in Text FragmentTable). A value of a content_linkage field corresponds to a value of an FDTCotent-Linkage element or a value of a File-Content-Linkage element in FLUTE FDT of each file related to a content item. A priority rule is applied when each content linkage value including a corresponding content linkage element in FLUTE FDT is matched.

[00375] A TF_availiable flag (Boolean flag) is set with 1 when Text Fragment exists in a Text Fragment Table of a service signaling channel. If Text Fragment is not included in a service signaling channel for the content item, a value of the TF_availiable field is set with 0.

[00376] If a low_lantency flag (Boolean flag) is set with 1, as a user waits, content is valid in a current digital transmission of sufficiently low delay time that collection attempts. If set with 0, a collection delay time becomes

longer and a user interface suggests a post view to a user.

[00377] A playback_length_in_seconds (20 bits) is an integer representing a playing time of a content in sec. A content including texts and/or still images has a value of 0. In relation to a content including audio or audio/video content, playback length in seconds represents a playing time of audio or audio/video content.

[00378] If a content_length_included flag (Boolean flag) is set with 1, a content_length field exists in the repetition in a 'for' loop. If set with 0, it indicates that the content_length field does not exist in the repetition in a 'for' loop.

[00379] If a playback_delay_included flag (Boolean flag) is set with 1, it indicates that a playback_delay field exists in the repetition in a 'for' loop. If set with 0, it indicates that the playback_delay field does not exist in the repetition in a 'for' loop.

[00380] If an expiration_included flag (Boolean flag) is set with 1, an expiration field exits in the repetition in a 'for' loop. If set with 0, it indicates that the expiration field does not exist in the repetition in a 'for' loop.

[00381] A duration (12 bits) field represents an expected cycle time of carousel including a referenced content item in a range of 1 to 2880 in min. A receiver uses a duration parameter determining a time taking for the referenced content capture.

[00382] playback_delay (20 bits) is represented with a number of the next sec of the first byte before playing a related content while incoming stream is buffered. A value of 0 represents playing starts immediately. When playback_delay is

not set, a receiver collects a complete file or a file before playing.

[00383] An expiration field (32 bits) represents expiration time expressed in GPS sec from 00:00:00 UTC, January 6, 1980. After expiration, the content is deleted from the memory. If it is not expired, the receiver uses a method that a company for managing a memory resource selects.

[00384] A content_name_length_ field (8 bits) represents the length (byte unit) of content_name_text.

[00385] A content_name_text() field represents a content item title in a system having a plurality of string structures.

[00386] A content_descriptors_length field (12 bits) represents an entire length (byte unit) of content_descriptor providing additional information on a content level.

[00387] content_descriptor is a descriptor that is additionally applied to each content item.

[00388] descriptor_length (10 bits) represents an entire length (byte unit) of a descriptor.

[00389] A descriptor is generally applied to all content items described in the current NRT-IT section.

[00390] Fig. 17 is a view illustrating a syntax structure of bit stream for NRT section (NRT_content_table_section) according to an embodiment. Detailed description of each field in the NCT section is as follows.

[00391] In Fig. 17, a table_id field (8 bits) as the identifier of a table includes an identifier identifying NCT.

[00392] A section_syntax_indicator field (1 bit) is an indicator defining a section format of NCT.

[00393] A private_indicator field (1 bit) represents whether NCT follows a private section.

[00394] A section_length field (12 bits) represents the section length of NST.

[00395] An NRT_channel_id field (16 bits) represents a value uniquely identifying NRT service including content described in NCT.

[00396] A version_number field (5 bits) represents the version number of NCT.

[00397] A current_next_indicator field (1 bit) represents whether information in a corresponding NCT section is applicable currently or in the future.

[00398] A section_number field (8 bits) represents the section number of a current NCT section.

[00399] A last_section_number field (8 bits) represents the last section number of NCT.

[00400] A protocol_version field (8 bits) indicates a protocol version for allowing NCT, which transmits parameters having different structures then those defined in a current protocol. (An 8-bit unsigned integer field whose function is to allow, in the future, this NRT Content Table to carry parameters that may be structured differently than those defined in the current protocol. At present, the value for the protocol_version shall be zero. Non-zero values of protocol_version may be used by a future version of this standard to indicate structurally different tables.)

[00401] A num_contents_in_section field (8 bits) indicates the number of contents in the NCT. At this point, the number of contents represents the number of contents transmitted through a virtual channel that source_id specifies.

[00402] Later, a 'for' loop (or a content loop) is performed as many as the number of contents corresponding to the

num_contents_in_section field value, to provide the detailed information of a corresponding content by each content.

[00403] A content_version field (32 bits) indicates the version number for content (or a file) having a specific content_id value. That is, let's assume that if content_id of a content that a receiver receives previously is 0x0010, the same content, i.e., its content_id value is 0x0010 is transmitted. At this point, if the content_version field value is different, the previously stored content is updated or replaced by receiving the newly announced content through the NCT. In this embodiment, the content_version field value means a series number representing a release version but may actually represent published (released) time directly. At this point, if the content_version field is difficult to represent publish time, a new field may be used to represent the published (released) time.

[00404] A content_id field (16 bits) indicates an identifier uniquely identifying the content (or file).

[00405] A content_available_start_time field (32 bits) and a content_available_end_time field (32 bits) represent a start time and end time of a FLUTE session transmitting the content.

[00406] An ETM_location field (2 bits) describes the existence and location of an extended text message (ETM).

[00407] A content_length_in_seconds field (30 bits) represents an actual play time of a corresponding content in sec unit when the content (or file) is an A/V file.

[00408] A content_size field (48 bits) represents the size of the content (or file) in byte unit.

[00409] A content_delivery_bit_rate field (32 bits) represents a bit rate at which the content (or file) is

transmitted, and means a target bit rate. That is, when a service provider or broadcasting station transmits a corresponding content, the content_delivery_bit_rate field displays how wide a bandwidth is to be allocated. Accordingly, if a receiver uses content_size and content_delivery_bit_rate, the minimum time for receiving a corresponding content (or file) is obtained. That is, the time for receiving content is estimated and provided to a user. Also, the minimum receiving time is obtained by calculating (conent_size * 8) / (content_delivery_bit_rate) and its unit is in sec.

[00410] A content_title_length field (8 bits) represents the length of content_title_text() in byte unit. If this field is used, the receiver knows how many bytes need to be read to obtain content_title_text () information.

[00411] A content_title_text() field represents a content title in the format of a multiple string structure.

[00412] That is, the receiver uses the NCT to obtain configuration information on NRT content/file, and provides a guide for the NRT/file on the basis of the obtained configuration information on NRT content/file. Moreover, the receiver obtains access information of FLUTE session, which transmits the content/file selected by the guide, from NST, and receives the selected content by using the obtained FLUTE session access information.

[00413] Moreover, the present invention may include container information, encoding information, and decoding parameters of media objects, necessary for rendering of the content/files constituting NRT service, in the NCT, and then transmit it. Accordingly, a receiving system extracts the container information, the encoding information, and the decoding

parameters of media objects by each content, necessary for rendering of the corresponding content/files, and uses them in rendering.

[00414] Fig. 18 is a view illustrating a bit stream syntax structure of an SMT session providing signaling information on NRT service data according to an embodiment.

[00415] Here, the corresponding syntax is created in an MPEG-2 private section format to help understanding, but the format of the corresponding data may vary.

[00416] The SMT describes signaling information (or signaling information of NRT service) and IP access information of a mobile service in Ensemble in which SMT is transmitted. The SMT uses Transport_Stream_ID, i.e., an identifier of broadcast stream including each service, and provides broadcasting stream information of a corresponding service. Furthermore, SMT includes description information of each mobile service (or NRT service) in one Ensemble, and includes other additional information in a descriptor area.

[00417] As mentioned above, the SMT session may be included as the IP stream format in the RS frame, and then, transmitted. In this case, RS frame decoders of a receiver describe later decode inputted RS frames, and outputs the decoded RS frames as a corresponding RS frame handler. Moreover, each RS frame handler divides the inputted RS frame by a row unit to constitute M/H TP, and outputs it as an M/H TP handler.

[00418] In addition, examples of fields transmitted through SMT are as follows.

[00419] A table_id field (8 bits) is a field indicating a table type, and through this, it is confirmed that this table

section is a table section in SMT. (table_id: An 8-bit unsigned integer number that indicates the type of table section being defined in Service Map Table (SMT)).

[00420] A section_syntax_indicator field (1 bit) is an indicator defining a session format of SMT, and its session format may be a short-form syntax ('0') of MPEG (section_syntax_indicator: This 1-bit field shall be set to '0' to always indicate that this table is derived from the "short" form of the MPEG-2 private section table).

[00421] A private_indicator field (1 bit) indicates whether SMT follows a private section (private_indicator: This 1-bit field shall be set to '1').

[00422] A section_length field (12 bits) represents the remaining session length of SMT after a corresponding field (section_length: A 12-bit field. It specifies the number of remaining bytes this table section immediately following this field. The value in this field shall not exceed 4093 (0xFFD)).

[00423] A table_id_extension field (16 bits) is dependent on a table, and may be a logical part of a table_id field providing a range of the remaining fields (table_id_extension: This is a 16-bit field and is table-dependent. It shall be considered to be logically part of the table_id field providing the scope for the remaining fields).

[00424] Here, a table_id_extension field includes an SMT_protocol_version field.

[00425] The SMT_protocol_version field (8 bits) shows a protocol version that allows SMT transmitting parameters having a different structure than those defined in a current protocol (SMT_protocol_version: An 8-bit unsigned integer field whose function is to allow, in the future, this SMT to

carry parameters that may be structured differently than those defined in the current protocol. At present, the value for the SMT_protocol_version shall be zero. Non-zero values of SMT_protocol_version may be used by a future version of this standard to indicate structurally different tables).

[00426] An ensemble id field (8 bits) includes values of '0x00' to '0x3F', as an ID value related to corresponding Ensemble (ensemble_id: This 8-bit unsigned integer field in the range 0x00 to 0x3F shall be the Ensemble ID associated with this Ensemble. The value of this field shall be derived from the parade_id carried from the baseband processor of physical layer subsystem, by using the parade_id of the associated Parade for the least significant 7 bits, and using '0' for the most significant bit when the Ensemble is carried over the Primary RS frame, and using '1' for the most significant bit when the Ensemble is carried over the Secondary RS frame).

[00427] A version_number field (5 bits) represents the version number of SMT. A current_next_indicator field (1 bit) indicates whether a transmitted SMT table session is applicable currently (current_next_indicator: A one-bit indicator, which when set to '1' shall indicate that the Service Map Table sent is currently applicable. When the bit is set to '0', it shall indicate that the table sent is not yet applicable and will be the next table to become valid. This standard imposes no requirement that "next" tables (those with current_next_indicator set to '0') must be sent. An update to the currently applicable table shall be signaled by incrementing the version_number field).

[00428] A section_number field (8 bits) represents a current

SMT session number (section_number: This 8-bit field shall give the section number of this NRT Service Signaling table section. The section_number of the first section in an NRT Service Signaling table shall be 0x00. The section_number shall be incremented by 1 with each additional section in the NRT Service Signaling table).

[00429] A last_section_number field (8 bits) represents the last session number constituting an SMT table.

[00430] (last_section_number: This 8-bit field shall give the number of the last section (i.e., the section with the highest section_number) of the Service Signaling table of which this section is a part).

[00431] A num_services field (8 bits) indicates the number of services in an SMT session. (num_services: This 8 bit field specifies the number of services in this SMT section.). At least one mobile service, at least one NRT service, or mobile and NRT services may be received through Ensemble having the SMT. If only NRT services are transmitted through the Ensemble having SMT, it may indicate the number of NRT services in the SMT.

[00432] Later, a 'for' loop (or a service loop) is performed as many times as the number of services corresponding to the num_service field value, to provide signaling information on a plurality of services. That is, signaling information of a corresponding service is displayed by each service in the SMT session. Here, the service may be mobile or NRT service. At this point, the following field information may be provided to each service.

[00433] A service_id field (16 bits) represents a value uniquely identifying a corresponding service (A 16-bit

unsigned integer number that shall uniquely identify this service within the scope of this SMT section.). The service_id of a service shall not change throughout the life of the service.

[00434] To avoid confusion, it is recommended that if a service is terminated, then the service id for the service should not be used for another service until after a suitable interval of time has elapsed. Here, if the service is NRT service, the service_id may identify the NRT service.

[00435] A Multi_ensemble_service field (2 bits) identifies whether a corresponding service is transmitted through at least one Ensemble.

[00436] Additionally, the corresponding field identifies whether service is rendered as a portion of the service transmitted through a corresponding Ensemble. That is, if the service is NRT service, the filed identifies whether NRT service is transmitted through at least one Ensemble (multi_ensemble_service: A two-bit enumerated field that shall identify whether the Service is carried across more than one Ensemble. Also, this field shall identify whether or not the Service can be rendered only with the portion of Service carried through this Ensemble.).

[00437] A service_status field (2 bits) identifies a state of a corresponding service. Here, MSB indicates whether a corresponding service is active (1) or inactive (0), and LSB indicates whether a corresponding service is hidden (1) or not (0). Here, when the service is NRT service, MSB of the service_status field indicates whether a corresponding NRT service is active (1) or inactive (0), and LSB indicates whether a corresponding NRT service is hidden (1) or not (0).

[00438] A SP_indicator field (1 bit) represents whether service protection of a corresponding service is set. If a SP_indicator field value is 1, service protection is applied to components required for providing meaningful presentation of a corresponding service.

[00439] A short_service_name_length field (3 bits) represents the length of a short service name in a short_service_name field in byte unit.

[00440] A short_service_name field represents a short name of a corresponding service (short_service_name: The short name of the Service, each character of which shall be encoded per UTF-8 [29]. When there is an odd number of bytes in the short name, the second byte of the last of the byte pair per the pair count indicated by the short_service_name_length field shall contain 0x00). For example, if the service is mobile service, a short name of the mobile service is displayed, and if it is NRT service, a short name of the NRT service is displayed.

[00441] A service_category field (6 bits) identifies a type category of a corresponding service. If a value of a corresponding field is set with a value indicating "informative only", it is dealt as an informative description for the category of the service. And, a receiver is required to test a component_level_descriptors() field of SMT in order to identify an actual category of the received service. The service_category field has an NTP time based component for services having video and/or audio component.

[00442] Especially, in regards to the present invention, if a service_category field value has '0x0E', a corresponding service indicates NRT service. In this case, it is indicated

that signaling information of service currently described in an SMT session is signaling information of NRT service.

[00443] A num_services field (5 bits) indicates the number of IP stream components in this service.

[00444] IP_version_flag field (1 bit), when set to '0', shall indicate that source IP address, service_destination_IP_address, and component_destination_IP_address fields are IPv4 addresses. The value of '1' for this field is reserved for possible future indication that source_IP_address, service_destination_IP_address, and component_destination_IP_address fields are for IPv6. Use of IPv6 addressing is not currently defined.

[00445] A source_IP_address_flag field (1 bit) shall indicate, when set, that a source IP address value for this Service is present to indicate a source specific multicast.

[00446] When a service_destination_IP_address_flag field (1 bit) is set, it indicates that a corresponding IP stream component is transmitted through IP datagram having a different target IP address than service_destination_IP_address.

[00447] Accordingly, if the flat is set, a receiving system uses component_destination_IP_address as destination_IP_address, and disregards a service_destination_IP_address field in a num_channels loop (service_destination_IP_address_flag: A 1-bit Boolean flag that indicates, when set to '1', that a service_destination_IP_address value is present, to serve as the default IP address for the components of this Service).

[00448] In relation to the source_IP_address field (32 or 128

bits), if source_IP_address_flag is set with 1, interpretation is required, but if not set with 0, no interpretation is required.

[00449] When the source_IP_address_flag field is set with '1' and the IP_version_flag field is set with '0', this field indicates a 32 but IPv4 address representing a source of a corresponding circuit channel. If the IP_version_flag field is set with '1', this field indicates a 32 bit IPv6 address representing a source of a corresponding virtual channel (source_IP_address: This field shall be present if the source_IP_address_flag is set to '1' and shall not be present if the source_IP_address_flag is set to '0'. If present, this field shall contain the source IP address of all the IP datagram carrying the components of this Service. The conditional use of the 128 bit-long address version of this field is to facilitate possible use of IPv6 in the future, although use of IPv6 is not currently defined).

[00450] If the service is NRT service, the Source_IP_address field becomes a source IP address of the same server transmitting all channels of the FLUTE session.

[00451] In relation to the service_destination_IP_address field (32 or 128 bits), if service_destination_IP_address_flag is set with 1, interpretation is required, but if set with 0, no interpretation is required. When the service_destination_IP_address_flag field is set with '1' and the IP_version_flag field is set with '0', this field indicates a 32 bit destination IPv4 address for a corresponding virtual channel.

[00452] When the service_destination_IP_address_flag field is

set with '1' and the IP_version_flag field is set with '1', this field indicates a 64 bit destination IPv6 address for a corresponding virtual channel. If the corresponding service_destination_IP_address cannot be interpreted, a component_destination_IP_address field in a num_components loop needs to be interpreted, and a receiving system uses component_destination_IP_address to access an IP stream component (service_destination_IP_address: This field shall be present if the service_destination_IP_address_flag is set to '1' and shall not be present if the service_destination_IP_address_flag is set to '0'. If this service_destination_IP_address is not present, then the component_destination_IP_address field shall be present for each component in the num_components loop. The conditional use of the 128 bit-long address version of this field is to facilitate possible use of IPv6 in the future, although use of IPv6 is not currently defined). If the service is NRT service, the service_destination_IP_Address field is signaled with a destination IP address of a session level of the FLUTE session.

[00453] Additionally, SMT provides information on a plurality of components by using a 'for' loop.

[00454] Later, a 'for' loop (or a component loop) is performed as many times as the number of components corresponding to the num_components field value, to provide access information on a plurality of components. That is, access information on each component in a corresponding service is provided. At this point, the following field information on each component may be provided. Here, one component corresponds to one FLUTE session according to an

embodiment.

[00455] An essential_component_indicator field (1 bit), when set to '1', shall indicate that this component is an essential component for the service. Otherwise, this field indicates that this component is an optional component).

[00456] A component destination IP address flag field (1 bit) shall indicate, when set to '1', that the component_destination_IP_address is present for this component.

[00457] A port_num_count field (6 bits) shall indicate the number of destination UDP ports associated with this UDP/IP stream component. The values of the destination UDP port numbers shall start from the component_destination_UDP_port_num field and shall be incremented by one, except in the case of RTP streams, when the destination UDP port numbers shall start from the component_estination_UPD_port_num field and shall be incremented by two, to allow for the RTCP streams associated with the RTP streams.

[00458] A component_destination_UDP_port_num (16 bits) represents the destination UDP port number for this UDP/IP stream component. For RTP streams, the value of component_estination_UDP_port_num shall be even, and the next higher value shall represent the destination UDP port number of the associated RTCP stream).

[00459] A component_destination_IP_address field (32 or 128 bits) shall be present if the component_destination_IP_address_flag is set to '1' and shall not be present if the component_destination_IP_address_flag is set to '0'. When this field is present, the destination

address of the IP datagram carrying this component of the M/H Service shall match the address in this field. When this field is not present, the destination address of the IP datagram carrying this component shall match the address in the M/H_service_destination_IP_address field. The conditional use of the 128 bit-long address version of this field is to facilitate possible use of IPv6 in the future, although use of IPv6 is not currently defined.

[00460] A num_component_level_descriptors field (4 bits) indicates the number of descriptors providing additional information on a component level.

[00461] component_level_descriptor() fields are included in the component loop as many as a number corresponding to the num_component_level_descriptors field value, so that additional information on the component is provided.

[00462] A num_service_level_descriptors field (4 bits) indicates the number of descriptors providing additional information on a corresponding service level.

[00463] service_level_descriptor() fields are included in the service loop as many as a number corresponding to the num_service_level_descriptors field value, so that additional information on the service is provided. If the service is mobile service, additional information on the mobile service is provided, and if it is NRT service, additional information on the NRT service is provided.

[00464] A num_ensemble_level_descriptors field (4 bits) indicates the number of descriptors providing additional information on an ensemble level.

[00465] ensemble_level_descriptor() fields are included in the ensemble loop as many as a number corresponding to the

num_ensemble_level_descriptors field value, so that additional information on the ensemble is provided.

[00466] Moreover, component_descriptor() as component_level_descriptors() may be provided to SMT of Fig. 18.

[00467] The component descriptor() is used as one of omponent_level_descriptors() of SMT, and describes additional signaling information of a corresponding component.

[00468] Accordingly, in relation to mobile NRT service, signaling information necessary for receiving a corresponding FLUTE session may be provided using the component descriptor of Fig. 14.

[00469] For example, if a component_type field value of the component descriptor of Fig. 14 is 38, a component_data (component_type) field provides data for FLUTE file delivery as shown in Fig. 15. Since each field description of Figs. 14 and 15 is made above, overlapping descriptions will be omitted.

[00470] Fig. 19 is a view illustrating an FDT schema for mapping a file and content_id according to an embodiment. Fig. 20 is a view illustrating an FDT schema for mapping a file and content_id according to another embodiment. They represent an FDT instant level entry file designating method. NRT content includes a plurality of files. However, since each file has no mark, it is difficult to search a file related to NRT content. Accordingly, as shown in Figs. 19 and 20, content_id is inserted into FDT in each file.

[00471] Hereinafter, an FDT instance level means, if a common attribute of all files declared in FDT needs to be defined, a level including a definition portion for the common attribute.

An FDT file level may mean a level including definition for an individual attribute of each file.

[00472] A receiver identifies whether a service transmitted through a corresponding channel is an SMT based NRT service. Additionally, the receiver identifies a content item and file of the corresponding NRT service.

[00473] As mentioned above, although the receiver may identify a file and content item in the NRT service, it does not have information on files of the content item and thus cannot match them. Accordingly, the receiver may not process the NRT service.

[00474] Accordingly, the present invention provides a method of identifying whether a content item is related. That is, a corresponding method shows what kinds of files are included in a content item. In this case, the receiver may properly process the received NRT service. Accordingly, the corresponding method may be designated on the basis of FDT information in FLUTE session transmitting NRT service. For example, each file constituting a content item is identified on the basis of a content-location and TOI field designated in the FLUTE session. content_id in FDT is matched to a content identifier (content_id) of NCT or a content identifier of content fragment in OMB BCAST SG.

[00475] Referring to Figs. 19 and 20, a portion indicated with 1 declares a content identifier in an FDT-Instance level, and this declared content identifier is assigned to all files declared in a corresponding FDT-Instance. Of course, this information may be overridden by assigning a new content identifier in a file level. Or, if a specific file belongs to another content item not a content item defined in the FDT-

Instance level, this may be notified through assigning a file level content_id described below. This embodiment expresses content_id in 16 bits.

[00476] In relation to a portion indicated with 2, when a file in the FDT Instance is included different content items with content id declaration in a file level, this method signals which file, all files of a content item and content, belongs to which entry.

[00477] A portion 3 is a method of notifying whether a corresponding file for each file is an entry file. That is, a file corresponding to a root file, which is played first among several files constituting a content item or is necessarily executed first to access a content item is called an entry file, and represents a method of notifying this information. An entry attribute may be omitted, and its default value is false. When it is omitted, it means that a corresponding file is not an entry file. “Entry” is a head of a file that needs to be processed to execute the file. For example, "index.html" may be an “entry”. Accordingly, an entry file may be set with ‘true” and other files are set with “false”. Through the entry file, transmitting the same file repeatedly may be effectively controlled. Once a file is downloaded, the entry file indicates a file of content for another reference, so that there is no need to download it in another or an additional instance.

[00478] A specific file functions as an entry in a specific group as a group related to a file level signals whether entry is possible, but its corresponding role may fail in another group. When a content identifier is assigned in an FDT-instance level, a method of notifying an entry file may

be considered as the following two methods.

[00479] 1) A method of additionally assigning a file level content identifier to a file corresponding to an entry file and setting its entry attribute with true: in this case, a content identifier is duplicated in an FDT-Instance level and a file level, but has the most flexible structure. That is, although one of the File-level and FDT-instance level may designate content_id, if another content_id is designated together in the File-level and FDT-instance, the content_id of the File level has priority to that of the FDT-instance level.

[00480] 2) like another embodiment of the FDT schema of Fig. 20, files functioning as an entry file may be directly referenced in content identifier definition in the FDT-instance level. For this, according to the embodiment of Fig. 20, FDT-Content-ID-Type is additionally defined for an FDT-instance level content identifier, and as shown in the portion 2, extends to include a content location of an entry file. In the case of the portion 2, an entry level is defined with its content_id. For example, each content_id shows which entry file exists.

[00481] In this method, content-location is duplicated so signaling may be problematic, but entry file configuration information may be immediately obtained by each content item.

[00482] Fig. 21 is a flowchart illustrating an operation of a receiver according to an embodiment.

[00483] Referring to Fig. 21, according to an embodiment, a receiver receives NRT service signaling data through an NRT service signaling channel, displays NRT guide information on the basis of the received NRT service signaling data, and

receives NRT service data for the selected NRT content, in order to provide NRT service.

[00484] First, once the receiver is turned on, a user selects a channel in operation S1000. Then, a physical transmission channel is turned according to the selected channel.

[00485] Then, VCT and PMT are obtained from a broadcast signal received through the tuned physical transmission channel in operation S1010. Then, it is confirmed in operation S1020 whether there is NRT service by parsing the obtained TVCT (VCT). This is confirmed by checking the service_type field value in a virtual loop of the VCT. For example, if a service_type field has 0x08, there is NRT service. Moreover, if not 0x08, since a corresponding virtual channel does not transmit the NRT service, a proper operation such as general A/V service may be performed according to information in the virtual channel in operation S1111.

[00486] Moreover, if it is determined that there is NRT service, since a corresponding virtual channel transmits NRT service, PID(PID=PID_NST) matching to a specific PID(PID_NST) of stream including a well known IP address for NRT service signaling channel address is obtained in operation S1030.

[00487] Moreover, the receiver receives a Transport Packet (TP) having the same PID as the obtained PID value (PID_NST) in operation S1040.

[00488] Then, the receiver extracts NRT service signaling data including a NRT service table (NST) from the received TP, or extracts an IP address for the NRT service signaling channel access from the received TP, in order to receive NRT service signaling data transmitted in another format through an IP layer in operation S1050.

[00489] Then, the receiver obtains channel information on NRT service data transmission by each NRT service from NST in operation S1060.

[00490] Then, the receiver obtains an NRT content table (NCT) including an NRT_channel_id field value identical to a value of Channel id, an identifier of the obtained channel information, from the NRT service signaling data in operation S1070.

[00491] Then, the receiver obtains content information on NRT content constituting each NRT service from each field of the obtained NCT in operation S1080. For example, the content information may include at least one of content_delivery_bit_rate, content_available_start_time, content_available_end_time and content_title_text() fields according to an embodiment of the NCT.

[00492] Then, the receiver displays NRT guide information by using content information in operation S1090. A user may select NRT content to use or be received, from the displayed NRT guide information.

[00493] Then, the receiver obtains NRT service access information having the selected NRT content from NST in operation S1100. The NRT service access information may include channel information or IP address information for receiving NRT service data, for example.

[00494] Moreover, the receiver receives a corresponding NRT content in operation S1110 by using the obtained NRT service access information after accessing a channel or server for transmitting NRT service, and performs a proper operation according to the NRT content.

[00495] Figs. 22 and 23 are views illustrating a receiving

system receiving, storing, and playing an NRT content for NRT service according to another embodiment.

[00496] The receiver of Fig. 23 may include an operation controlling unit 100, a baseband processing unit 110, a service demultiplexer 120, a stream component handler 130, a media handler 140, a file handler 150, a service manager 160, a PVR manager 170, a first storage unit 180, an SG handler 190, an EPG manager 191, an NRT service manager 192, an application manager 194, a middleware engine 193, a presentation manager 195, and a User Interface (UI) manager 196.

[00497] The baseband processing unit 110 may include a tuner 111 and a demodulator. The service demultiplexer 120 may include an MPEG-2 TP handler 121, a PSI/PSIP handler 122, an MPEG-2 TP demultiplexer 123, a descrambler 124, and a second storage unit 125.

[00498] The stream component handler 130 may include a Packetized Elementary Stream (PES) demodulator 131, an Elementary Stream (ES) demodulator 132, a PCR handler 133, a STC handler 134, a DSM-CC addressable section handler 135, an IP datagram handler 136, a descrambler 137, a UDP handler 138, a service signaling section handler 138-1, and a Conditional Access System (CAS) 139.

[00499] The media handler 140 may include an A/V demodulator 141. The file handler 150 may include an ALC/LCT stream handler 151, a file reconstruction buffer 152, an XML parser 153, an FDT handler 154, a decompressor 155, a third storage unit 156, and a file decoder 157.

[00500] In Fig. 23, the tuner 111 tunes a broadcast signal of a desired channel among broadcast signals received through a

terrestrial wave according to a control of the service manager 160, and then down-converts the tuned broadcast signal into an Intermediate Frequency (IF) signal to output it to the demodulator 112. The tuner 111 may receive real-time stream and non-real-time stream. The non-real-time stream is called an NRT stream in the present invention.

[00501] The demodulator 112 performs automatic gain control, carrier recovery, and timing recovery on a digital IF signal of a pass band inputted from the tuner 111, converts the digital IF signal into a baseband signal, and performs channel equalization. For example, when the broadcast signal is a VSB modulation signal, a VSB demodulation process is performed for automatic gain control, carrier recovery, and timing recovery.

[00502] The demodulated and channel-equalized data in the demodulator 112 is outputted to the MPEG-2 TP handler 121 in an MPEG-2 Transport Stream (TS) packet format.

[00503] The MPEG-2 TP handler 121 includes an MPEG-2 TP buffer and an MPEG-2 TP parser, and analyzes a TS header after temporarily storing an output of the demodulator 112. Then, if an output of the demodulator 112 is an A/V TS packet for real time or an NRT TS packet, it is outputted to the demultiplexer 123, and if it is a TS packet for PSI/PSIP table, it is outputted to the PSI/PSIP handler 122.

[00504] The PSI/PSIP handler 122 includes a PSI/PSIP section buffer and a PSI/PSIP parser, and after temporarily storing a TS packet outputted from the MPEG-2 TP handler 121, restores and parses a corresponding table from PSI/PSIP section data in a payload of the TS packet, with reference to a table identifier. At this point, it is determined whether one table

includes one section or a plurality of sections through a table_id field, a section_number field, and a last_section_number field in a corresponding section. Also, sections having the same table identifier are collected to complete a corresponding table. For example, sections having a table identifier allocated to VCT are collected to complete VCT. Moreover, the parsed information of each table is collected by the service manager 160 to be stored in the first storage unit 180. Table information such as VCT, PAT, PMT, and DST are stored in the first storage unit through the above processes. The service manager 160 stores the table information in the first storage unit 180 in a service map and guide data format.

[00505] The demultiplexer 123, if the inputted TS packet is an A/V TS packet in real time, divides the TS packet into an audio TS packet and a video TS packet, and then outputs them into the PES decoder 131. If the inputted TS packet is an NRT TS packet, it is outputted to the DSM-CC handler 135. Additionally, the demultiplexer 123, if the TS packet includes a Program Clock Reference (PCR), outputs it to the PCR handler 133, and if it includes Conditional Access (CA) information, outputs it to the CAS 139. An NRT TS packet includes a TS packet having NRT service data and a TS packet having NRT service signaling channel. A unique PID for identifying the NRT service is allocated to a TS packet of the NRT service data, and PID of a TS packet including the NRT service signaling channel is extracted using DST and PMT.

[00506] The demultiplexer 123, if a payload of the inputted TS packet is scrambled, outputs it to the descrambler 124, and then, the descrambler 124 receives information (control

words used for scramble) necessary for descramble from the CAS 139, and performs descramble on the TS packet.

[00507] The demultiplexer 123 stores an A/V packet in real time, inputted at the one request of temporary recording, scheduled recording, and time shift, in the second storage unit 125. The second storage unit 125 is a mass storage medium and may include HDD, for example. The second storage unit 125 performs downloading (i.e., storing) and updating (i.e., playing) according to a control of the PVR manager 170.

[00508] The demultiplexer 123 separates an audio TS packet and a video TS packet from the A/V TS packet updated from the second storage unit and then outputs them to the PES decoder 131 at the playing request.

[00509] The demultiplexer 123 is controlled by the service manager 160 and/or the PVR manager 170 to perform the above processes.

[00510] That is, if a service_type field value in VCT indicates that NRT service is transmitted, the service manager 160 extracts identification information of each NRT service from NRT_service_descriptor() received from a virtual channel loop of the VCT and stores it, and then extracts DST PID from a service location descriptor (or an ES loop of PMT) of the VCT to receive DST.

[00511] Then, NRT service is identified from the received DST, and PID of an MPEG-2 TS packet including the NRT service signaling channel is extracted to receive the identified NRT service by using DST and PMT. The extracted PID is outputted to the demultiplexer 123. The demultiplexer 123 outputs MPEG-2 TS packets corresponding to PID, outputted from the service manager 160, to the addressable section handler 135.

[00512] The PCR is a time reference value used for time synchronization of audio ES and video ES in the A/V decoder 141. The PCR handler 133 restores PCR in the payload of the inputted TS packet and outputs it to the STC handler 134. The STC handler 134 restores System Time Clock (STC), i.e., a reference clock of a system, from the PCR, and outputs it to the A/V decoder 141.

[00513] The PES decoder 131 includes a PES buffer and a PES handler, and after temporarily storing an audio TS packet and a video TS packet, removes a TS header from the TS packet to restore audio PES and video PES. The restored audio PES and video PES are outputted to the ES decoder 132. The ES decoder 132 includes an ES buffer and an ES handler, and removes each PES header from audio PES and video PES to restore audio ES and video ES, i.e., pure data. The restored audio ES and video ES are outputted to the A/V decoder 141.

[00514] The A/V decoder 141 decodes the audio ES and video ES through each decoding algorithm to restore a previous state of compression, and then outputs it to the presentation manager 195. At this point, time synchronization is performed when audio ES and video ES are decoded according to the STC. As one example, an audio decoding algorithm includes at least one an AC-3 decoding algorithm, an MPEG 2 audio decoding algorithm, an MPEG 4 audio decoding algorithm, an AAC decoding algorithm, an AAC+ decoding algorithm, an HE AAC decoding algorithm, an AAC SBR decoding algorithm, an MPEG surround decoding algorithm, and a BSAC decoding algorithm. A video decoding algorithm includes at least one of an MPEG 2 video decoding algorithm, an MPEG 4 video decoding algorithm, an H.264 decoding algorithm, an SVC decoding algorithm, and a

VC-1 decoding algorithm.

[00515] The CAS 139 includes a CA stream buffer and a CA stream handler, and after temporarily storing a TS packet outputted from the MPEG-2 TP handler or service protection data restored and outputted from a UDP datagram handler 138, restores information (for example, control words used for scramble) necessary for descramble from the stored TS packet or service protection data. That is, Entitlement Management Message (EMM) and Entitlement Control Message (ECM) in the payload of the TS packet are extracted and information necessary for descramble is obtained by analyzing the extracted EMM and ECM. The ECM may include a control word (CW) used in scramble. At this point, the control word may be encrypted using an encryption key. The EMM may include an encryption key and qualification information of corresponding data. Information necessary for descramble obtained from the CAS 139 is outputted to the descrambler 124 and 137.

[00516] The DSM-CC section handler 135 includes a DSM-CC section buffer and a DSM-CC section parser, and after temporarily storing a TS packet outputted from the demultiplexer 123, restores an addressable section in the payload of the TS packet. After restoring IP datagram by removing a header and CRC checksum of the addressable section, the restored IP datagram is outputted to the IP datagram handler 136.

[00517] The IP datagram handler 136 includes an IP datagram buffer and an IP datagram parser. After buffering IP datagram delivered from the DSM-CC section handler 135, the IP datagram handler 136 extracts and analyzes a header of the buffered IP datagram to restore UDP datagram from the payload

of the IP datagram, and then, outputs it to the UDP datagram handler 138.

[00518] At this point, if the IP datagram is scrambled, the scrambled UDP datagram is descrambled in the descrambler 137 and then is outputted to the UDP datagram handler 138. As one example, the descrambler 137 receives information (e.g., a control word used for scramble) necessary for descramble from the CAS 138 and performs descramble on the UDP datagram to output it to the UDP datagram handler 138.

[00519] The UDP datagram handler 138 includes an UDP datagram buffer and a UDP datagram parser. After buffering IP datagram delivered from the IP datagram handler 136 or the descrambler 137, the UDP datagram handler 138 extracts and analyzes a header of the buffered UDP datagram to restore the data included in the payload of the UDP datagram. At this point, if the restored data is service protection data, it is outputted to the CAS 139; if the restored data is NRT service signaling data, it is outputted to the service signaling section handler 138-1; and if the restored data is NRT service data, it is outputted to the ALC/LCT stream handler 151.

[00520] That is, access information on the IP datagram transmitting the NRT service signaling channel is a well-known destination IP address and a well-known destination UDP port number.

[00521] Accordingly, the IP datagram handler 136 and the UDP datagram handler 138 include a well-known destination IP multicast address and a well-known destination UDP port number, and extracts an IP multicast stream transmitting an NRT service signaling channel, i.e., NRT service signaling

data, to output it to the service signaling section handler 138-1.

[00522] Moreover, the service signaling section handler 138-1 includes a service signaling section buffer and a service signaling section parser, and restores and parses NST from the NRT service signaling data to output it to the service manager 160. When the NST is parsed, access information of the FLUTE session that transmits content/files constituting NRT service and signaling information necessary for rendering the NRT service may be extracted. For example, information necessary for rendering content/files of the NRT service, transmitted from the NST to each FLUTE session, may be extracted. Information necessary for rendering the content/files of the NRT service may include container information, encoding information, or decoding parameters of a media object.

[00523] The parsed information from the NST is collected by the service manager 160, and then, stored in the first storage unit 180. The service manager 160 stores the extracted information from the NST in the first storage unit 180 in a service map and guide data format. As another example, the NRT service manager 182 may serve as the service manager 160. That is, the parsed information from the NST is collected by the NRT service manager 192, and then, stored in the first storage unit 180.

[00524] The ALC/LCT stream hander 151 includes an ALC/LCT stream buffer and an ALC/LCT stream parser, and after buffering data having an ALC/LCT structure outputted fro the UDP datagram handler 138, analyzes a header and header extension of an ALC/LCT session from the buffer data. On the

basis of the analysis result of the header and header extension of the ALC/LCT session, if data transmitted to the ALC/LCT session has an XML structure, it is outputted to the XML parser 153. If the data has a file structure, after being temporarily stored in the file reconstruction buffer 152, it is outputted to the file decoder 157 or stored in the third storage unit 156. The ALC/LCT stream handler 151 is controlled by the NRT service manager 192 if data transmitted to the ALC/LCT session is data for NRT service. At this point, if data transmitted to the ALC/LCT session is compressed, after decompressed in the decompressor 155, it is outputted to at least one of the XML parser 153, the file decoder 157, and the third storage unit 156.

[00525] The XML parser 153 analyzes XML data transmitted through the ALC/LCT session, and if the analyzed data is for a file based service, it is outputted to the FDT handler 154. If the analyzed data is for service guide, it is outputted to the SG handler 190.

[00526] The FDT handler 154 analyzes and processes a file description table of the FLUTE protocol through an ALC/LCT session. The FDT handler 154 is controlled by the NRT service manager 192 if the received file is for NRT service.

[00527] The SG handler 190 collects and analyzes data for service guide transmitted in the XML structure and then output it to the EPG manager 191.

[00528] The file decoder 157 decodes a file outputted from the file reconstruction buffer 152, a file outputted from the decompressor 155, or a file uploaded from the third storage unit 156 through a predetermined algorithm, thereby outputting it to the middleware engine 193 or the A/V decoder

141.

[00529] The middleware engine 193 interprets and executes data having a file structure, i.e., application. Moreover, the application may be outputted to a screen or speaker through the presentation manager 195. The middleware engine 193 is a JAVA based middleware engine according to an embodiment.

[00530] The EPG manager 191 receives service guide data from the SG handler 190 according to a user input, and then, converts the received service guide data into a display format to output it to the presentation manager 195. The application manager 194 performs general managements on processing application data received in the format such as a file.

[00531] The service manager 160 collects and analyzes PSI/PSIP table data or NRT service signaling data transmitted to an NRT service signaling channel to create a service map, and then stores it in the first storage unit 125. Additionally, the service manager 160 controls access information on NRT service that a user wants, and also controls the tuner 111, the demodulator 112, and the IP datagram handler 136.

[00532] The operation controller 100 controls at least one of the service manager 160, the PVR manager 170, the EPG manager 191, the NRT service manager 192, the application manager 194, and the presentation manager 195 according to a user command, and thus, performs a function that a user wants.

[00533] The NRT service manager 192 performs general management on NRT service transmitted in a content/file format through the FLUTE session on an IP layer.

[00534] The UI manager 196 delivers a user input to the operation controller 100 through UI.

[00535] The presentation manager 195 provides to a user through at least one of a speaker and a screen at least one of audio/video data outputted from the A/V decoder 141, file data outputted from the middleware engine 193, and service guide data outputted from the EPG manager 191.

[00536] Moreover, one of the service signaling section handler 138-1, the service manager 160, and the NRT service manager 192 obtains content constituting the NRT service or IP access information on the FLUTE session transmitting a file, from a FLUTE session loop of NST (or an a component loop of NST). Additionally, the one obtains FLUTE level access information from component_descriptor() received in the component loop of the NST.

[00537] Then, the ALC/LCT stream handler and the file decoder 157 access the FLUTE file delivery session by using the obtained FLUTE level access information to collect files in the session. Once the files are collected, they constitute one NRT service. This NRT service may be stored in the third storage unit 156, or outputted to the middleware engine 193 or the A/V decoder 141 to be displayed on a display device.

[00538] The third storage unit 158, i.e., a storage medium storing a file such as NRT service data, may be shared with the second storage unit 125, or may be separately used.

[00539] Fig. 24 is a flowchart illustrating a method of a receiver to receive and provide NRT service according to an embodiment.

[00540] The receiver may obtain NRT service signaling information through an NRT service signaling channel or by

receiving IP datagram in the case of mobile NRT service, and obtains SMT from the NRT service signaling information in operation S2010.

[00541] Then, the receiver obtains NRT service information from SMT in operation S2020. The NRT service information may be obtained by parsing NRT service info descriptor in a service level descriptor loop. The obtained NRT service information may include requirement information on an application type for each NRT service or other NRT services.

[00542] Later, the receiver outputs NRT service guide on the basis of the obtained NRT service information in operation S2030. The NRT service guide may include application and service category information on each service. Additionally, detailed information may be further displayed on the basis of each field of NRT service info descriptor. The detailed information may include capacity information on corresponding NRT service according to a storage_requirement field or audio or video codec information on corresponding NRT service according to an audio_codec_type or video_codec_type field. A user may select NRT service to receive and use it on the basis of the information in the service guide.

[00543] Then, the receiver obtains identifier (content_id) for content items constituting the selected NRT service from NCT in operation S2040. The receiver obtains NRT_service_id corresponding to the selected NRT service from SMT, obtains NCT having the same NRT_channel_id value as the obtained NRT_service_id, and obtains an identifier (content_id) for content items constituting a corresponding NRT service through the obtained NCT.

[00544] Then, the receiver accesses the FLUTE session to

receive a file constituting the corresponding content item by using the obtained content item identifier (content_id) in operation S2050. Since each file constituting the content item is matched to TOI or a content location field of FDT in the FLUTE session, the receiver receives a file of a corresponding content item by using the FLUTE session in operation S2060. The receiving of the file may include receiving a corresponding file or object when a Content-ID attribute field for a corresponding file is identical to the obtained content_id after reading FDT in a corresponding FLUTE session.

[00545] Additionally, the receiver parses FDT instances in a corresponding FLUTE session to obtain a list of files corresponding to the content item. Moreover, the receiver obtains entry information including a list of files serving as an entry among lists of files.

[00546] Lastly, the receiver provides NRT service to a user on the basis of the receiver content item and the list of files corresponding thereto or entry information in operation S2080.

[00547] The content downloaded through the NRT service may be used at the timing that a user wants, being separated from real-time broadcasting.

[00548] Additionally, after transmitting NRT service in advance and storing it in a receiver, a broadcasting station may designate a content item of the corresponding NRT service, which is executed at the timing of when a specific real-time broadcasting is transmitted or the NRT service is displayed. According to an embodiment of the present invention, the NRT service may include content, which is downloaded in advance

linking with real-time broadcasting and executed at the specific timing. Additionally, according to an embodiment of the present invention, the NRT service may include content, which is prepared in advance to execute specific NRT service at the specific timing. An NRT service content triggered at the specific timing linked with real-time broadcasting to execute a specific action for a specific NRT service is called a Triggered Declarative Object (TDO). Accordingly, an NRT service application is classified as a non-real time declarative object (NDO) or a triggered declarative object (TDO) according to whether it is executed at the specific timing.

[00549] According to an embodiment of the present invention, a broadcasting station may transmit trigger information on trigging the TDO. The trigger information may include information on performing a specific action for a specific TDO at the specific timing.

[00550] Additionally, the trigger information may include trigger signaling data (trigger signaling information) for signaling a trigger, and trigger data constituting a trigger. Additionally, data stream transmitting trigger data may be designated as trigger stream. Also, the trigger data may mean itself.

[00551] Such a trigger may include at least one of a trigger identifier for identifying a trigger, a TDO identifier for identifying NRT service for trigger, and action information and trigger time on TDO.

[00552] The trigger identifier may be an identifier uniquely identifying a trigger. For example, a broadcasting station may include at least one trigger in broadcasting program

information of a predetermined time provided through EIT. In this case, the receiver may perform an action on the trigger target TDO at the timing designated for each trigger on the basis of at least one trigger. At this point, the receiver may identify each trigger by using a trigger identifier.

[00553] A TDO identifier may be an identifier for identifying an NRT service content, i.e., a target of trigger. Accordingly, the TDO identifier may include at least one of a trigger NRT service identifier (NRT_service_id), content linkage (content_linkage), and URI or URL of an NRT content item entry. Moreover, the TDO identifier may include a target identifier (target_service_id) for identifying a trigger target TDO described later.

[00554] Additionally, TDO action information may include information on action for TDO of a trigger target. The action information may be at least one of execution, termination, and extension commands of the target TDO. Additionally, the action information may include commands for generating a specific function or event in the target TDO. For example, if the action information includes the execution command of the target TDO, a trigger may request the activation of the target TDO to the receiver. Additionally, if the action information includes the extension command of the target TDO, a trigger may notify the receiver that the target TDO would extend. Additionally, if the action information includes the termination command of the target TDO, a trigger may notify the receiver that the target TDO would terminate. Thus, the broadcasting station may control a TDO operation in the receiver according to a real-time content through trigger.

[00555] Moreover, a trigger time may mean a time designated

for performing (trigging) an action designated for the target TDO. Additionally, the trigger time may be synchronized with video stream in a specific virtual channel in order to link NRT service with real-time broadcasting. Accordingly, the broadcasting station may designate a trigger time with reference to PCR that video stream refers. Accordingly, the receiver may trigger TDO at the timing that the broadcasting station designates with reference to PCR that video stream refers. Moreover, the broadcasting station may signal a trigger with a trigger identifier in a header of video stream in order to transmit accurate trigger time.

[00556] Additionally, the trigger time may be designated with UTC time. In the case of UTC time, the trigger time is not a relative time but an absolute time.

[00557] The trigger time may be accurate trigger timing or may include an approximate start time. Moreover, the receiver may prepare an action for target TDO in advance before accurate trigger timing by receiving approximate time. For example, the receiver may prepare TDO execution in advance so that TDO operates smoothly at the trigger time.

[00558] Fig. 25 is a view illustrating a bit stream syntax of a trigger according to an embodiment.

[00559] Here, trigger or trigger data is in a trigger table form, and a corresponding syntax is in an MPEG-2 private section form to help understanding. However, the format of corresponding data may vary. For example, the corresponding data may be expressed in a Session Description Protocol (SDP) format and signaled through a Session Announcement Protocol (SAP) according to another method.

[00560] A table_id field is set with 0XTBD arbitrarily, and

identifies that a corresponding table section is a table section constituting a trigger.

[00561] A section_syntax_indicator field is set with 1 and indicates that the section follows a general section syntax.

[00562] A private_indicator field is set with 1.

[00563] A section length field describes that the number of bits remaining in the section to the last of the section from immediately after the section_length field.

[00564] A source_id field represents the source of a program related to a virtual channel.

[00565] A TTT_version_number field represents version information of a trigger. Additionally, the version information of a trigger represents the version of a trigger protocol. The trigger version information may be used for determining where there is change in a trigger structure or a trigger itself. For example, the receiver determines that there is no trigger change if the trigger version information is identical. Additionally, the receiver determines that there a trigger change if the trigger version information is different. For example, the trigger version information may include a plurality of version numbers, and the receiver may determine whether there is a trigger change on the basis of some of the plurality of version numbers.

[00566] A current_next_indicator field represents that a corresponding table section is applicable currently if set with 1.

[00567] A section_number field indicates a number of a corresponding table section.

[00568] A last_section_number field means a table section of the last and highest number among sections.

[00569] A num_triggers_in_section field means the number of triggers in a corresponding table section. The number of triggers in one session may be one or in plurality. Additionally, the next 'for' loop is performed as many times as the number of triggers.

[00570] A trigger id field represents an identifier uniquely identifying a trigger.

[00571] A trigger_time field represents a time for which a trigger is performed. Moreover, this field may not be included in the session, and in this case, the trigger time may be a time designated from broadcasting stream as mentioned above.

[00572] A trigger_action field represents action information of a trigger performed at the trigger time. A trigger action may include at least one of a preparation command for target TDO, a target TDO execution command, a target TDO extension command, and a target TDO termination command. The trigger action may further include a command generating a specific command or event.

[00573] A trigger_description_length field represents the length of trigger_description_text.

[00574] A trigger_description_text field represents description for a corresponding trigger in a text format.

[00575] A service_id_ref field represents an identifier identifying a target TDO of a trigger. Accordingly, for example, a service_id_ref field may indicate an NRT_service_id field of SMT or NST to identify NRT service of a trigger target TDO.

[00576] A content_linkage field represents an identifier identifying a target TDO content item of a trigger. For

example, a content_linkage field may indicate a content_linkage field of NRT-IT or NCT to identify a target TDO content item of a trigger. Additionally, a service_id_ref field and a content_linkage field may be included in a class for indicating one target TDO.

[00577] A num_trigger_descriptors field represents the number of trigger descriptors.

[00578] A trigger_descriptor() field represents a descriptor including information on a trigger.

[00579] When a trigger is in a table format of the MPEG-2 private section, a broadcasting station may transmit one trigger according to a virtual channel.

[00580] A first method of a broadcasting station to transmit a trigger may include transmitting 0X1FF stream including the trigger table, i.e., PSIP basic PID. The first method may distinguish the trigger table from other tables by allocating table_id of the trigger table.

[00581] Moreover, a second method of transmitting a trigger includes allocating PID corresponding to a trigger table to a Master Guide Table (MGT) and transmitting a corresponding PID stream having the trigger table. The second method processes all tables in a corresponding PID stream by using the trigger table.

[00582] Moreover, according to an embodiment, at least one of trigger and trigger signaling information is transmitted through an MPEG-2 Packetized Elementary Stream (PES) in order to designate the accurate timing synchronized with video and audio as a trigger time.

[00583] Here, the video and audio synchronization of MPEG-2 PES will be described as follows. A receiver decoder operates

in synchronization with a time stamp of a transmitter encoder. The encoder has a main oscillator, called a System Time Clock (STC), and a counter. The STC is included in a specific program and a main clock of program for video and audio encoders.

[00584] Moreover, if a video frame or an audio block occurs in an encoder input, STC is sampled. A sampling value and a constant value as much as delay of the encoder and decoder buffers are added to generate display time information, i.e., Presentation Time Stamp (PTS) and then are inserted into the first portion of a picture or audio block. When frame reordering occurs, Decode Time Stamp (DTS) representing a time at which data needs to be decoded in a decoder is inserted. Except for the frame reordering of the B picture, DTS and PTS are same. DTS is additionally required in the case of the frame reordering. When DTS is used, there is PTS always. They may be inserted at an interval of less than about 700 msec. Additionally, it is defined in ATSC that PTS and DTS are inserted at the starting portion of each picture.

[00585] Moreover, an output of an encoder buffer includes a time stamp such as Program Clock Reference (PCR) in a transport packet level. Moreover, a PCT time stamp occurs at an interval of less than 100 msec, and is used for synchronizing STC of a decoder and STC of an encoder.

[00586] Moreover, video stream and audio stream may have each PTS or DTS corresponding to a common STC, for synchronization of audio stream and the decoder. Accordingly, PTS and DTS indicate when audio stream and video stream are played at each decoding unit, and are used to synchronize audio and video.

[00587] For example, a decoder of receiver outputs a PES packet in the received TS stream as a video PES depacketizer, and outputs a PCR value inserted in a TS packet header to a PCR counter. The PCR counter counts 100 of the PCR value and outputs it to a comparison unit. Moreover, the video PES depacketizer outputs a header of a PES packet to a DTS/PTS extractor, buffers Elementary Stream, i.e., image data to be displayed, in an Elementary Stream Buffer&Decoder. The DTS/PTS extraction unit extracts DTS and PTS values from the PES packet header and outputs them to the comparison unit. The comparison unit, if the PCR value inputted from the PCR counter becomes a DTS value or the PCR value of 100 becomes a PTS value, outputs each signal for that to a decoding/display control unit. The decoding/display control unit receives a signal that the PCR value becomes the DTS value from the comparison unit, and decodes the image data buffered in the elementary stream buffer & decoder to store them in a decoded stream memory. Additionally, the decoding/display control unit displays the decoded image data stored in the decoded stream memory through a display unit when receiving the signal that the PCR value becomes the PTS value from the comparison unit

[00588] Accordingly, MPEG-2 PES includes PTS and DTS in its header, which synchronize data transmitted during data transmission with one elementary stream (ES) or presentation time between a plurality of ES. This is called a synchronized data stream method.

[00589] That is, according to an embodiment, a broadcasting station includes trigger data or trigger stream in the payload of PES and designates trigger time as a PTS value of

the PES packet header by using the above synchronized data stream method. In this case, the receiver may trigger a target TDO at the accurate timing according to the PCR value that PTS of PES including a trigger refers. Accordingly, a broadcasting station may synchronize a trigger at the accurate timing of audio and video presentation that the broadcasting station is to trigger by using the PTS of the PES packet header designated as a trigger time and the PTS of the audio and video PES packet header.

[00590] Moreover, in relation to the header of the PES stream packet including a trigger, a stream_type value may be 0x06 to indicate a synchronized data stream method, stream_id may indicate a identifier of a predetermined stream, and PES_packet_length may indicate the length of PES stream including the payload of PES stream.

[00591] Fig. 26 is a view illustrating a PES structure according to a synchronized data stream method including a trigger according to an embodiment.

[00592] As shown in Fig. 26, PES of the synchronized data stream method may include a PES header and PES payload. The PES payload may include a synchronized data packet structure. As mentioned above, the trigger including a trigger table or another type of data may be included in the PES payload of Fig. 26 and then transmitted. Additionally, a broadcasting station may packetize the trigger in an IP datagram format, and may include and transmit the packetized trigger in an IP data area.

[00593] Fig. 27 is a view illustrating a synchronized data packet structure of PES payload for transmitting trigger as bit stream syntax according to an embodiment.

[00594] As shown in Figs. 26 and 27, the trigger may be included in the synchronized data packet structure and then transmitted. Detailed description of each field in the structure is as follows.

[00595] A data_identifier field is an identifier identifying a type of data included in a PES data packet. This may be set with 0X22 according to a type.

[00596] A sub_stream_id field is an identifier (user private) settable by a user.

[00597] A PTS_extention_flag field indicates whether there is a PTS_extention field. If this field value is 1, the PTS_extension field may be in the PES_data_packet field. Additionally, this field may be 0 when there is no PTS_extension field.

[00598] An output_data_rate_flag field may be set with 0.

[00599] A syncnronized_data_packet_header_length field represents the length of an optical field in the PES packet header. This field may be included If the PTS_extention_flag field is 1, and represents the length including synchroziced_data_privete_data_byte(s).

[00600] A PTS_extension field extends PTS delivered from the header of a corresponding PES packet. This field may include 9 bit Program Clock Reference (PCR) extension information. Additionally, a receiver may extend the PTS resolution of synchronized data from 11.1 μs (90 kHz), i.e., the MPEG-2 standard, to 37 ns (27 MHz).

[00601] A synchronized_data_private_data_byte field represents a payload byte of a synchronized PES packet. If the protocol_encapsulation of DST represents one of synchronized datagram, IP datagram not including LLC/SNAP,

and multiprotocol including LLS/SNAP, the synchronized_data_byte field may include one unique datagram. Accordingly, when LLC/SNAP is used, an 8 byte LLC/SNAP header may be shown in only the first 8 byte synchronized_data_byte of the PES packet.

[00602] Accordingly, if a broadcasting station includes a trigger in a synchronized data stream (stream_type) of PES and transmits it, a receiver may extract trigger stream from the payload of PES. Additionally, the receiver may perform an action on a target TDO by using the PTS value of the PES header as a trigger time. Accordingly, TDO may be trigged at the accurate timing of a frame unit by synchronizing a trigger on the basis of PTS, i.e., a reference time for presentation synchronization of video and audio. Additionally, when a trigger time is designated with PTS, video and audio synchronization may be easily obtained.

[00603] Moreover, trigger signaling information on obtaining trigger stream is transmitted according to an embodiment. A receiver receives trigger signaling information and obtains trigger stream in the synchronized data stream of PES on the basis of the received trigger signaling information.

[00604] A method of transmitting trigger signaling information to obtain trigger stream transmitted using synchronized data streaming may vary. One of the following methods is used to transmit trigger signaling information: 1. a transmission method through DST; 2. a transmission method through a service id descriptor; 3. a transmission method through a trigger stream descriptor; and 4. a transmission method by defining a stream type of trigger stream.

[00605] According to an embodiment, trigger signaling

information may be transmitted through DST for NRT service. DST is a table session for transmitting data service. Since its description and description for its data_service_bytes() are identical to those of Fig. 8, overlapping description will be omitted.

[00606] The DST may include signaling data for receiving each Elementary Stream (ES) constituting data service. Accordingly, trigger signaling data for receiving trigger stream may be included in DST.

[00607] Moreover, each data service may include at least one application, and each application may in an application identification structure including an application identifier such as app_id. Moreover, each application may include at least one data element constituting a corresponding application or data stream.

[00608] Accordingly, in order to transmit trigger stream through data service, a broadcasting station includes one trigger stream in a specific virtual channel and transmits it. Moreover, the broadcasting station may include one trigger stream in each application and transmit it. Accordingly, embodiments for transmitting trigger signaling information will be described according to two methods.

[00609] When one trigger stream is included a virtual channel, a data service for transmitting trigger stream is called a trigger service. In this case, a broadcasting station may allocate a fixed service identifier (service ID) to a trigger service.

[00610] Accordingly, a receiver may identify that one trigger stream is transmitted to a virtual channel when the service identifier has 0X01 as a fixed value.

[00611] Here, the broadcasting station may include trigger signaling information in an application identification structure in DST and transmit it.

[00612] For example, the broadcasting station adds 0x0001 as an App_id_description field value of DST to set a value that means interactive application for linking NT service such as TDO with a real-time broadcast Additionally, app_id_byte_length may use 3 bytes (0x0003) and app_id_byte may be allocated with 0x01 to indicate that corresponding data service includes trigger stream signaling information.

[00613] Accordingly, the receiver receives DST through the above method, and may identify tap() including trigger signaling information when app_id_byte_length is 0x0003, app_id_description is 0x0001, and app_id_byte is 0x01. The receiver extracts trigger signaling information including an association_tag value from the identified tap() structure, and association_tag_descriptor receives stream having the same PID as the extracted association_tag from data elementary stream (ES) listed in PMT extracted from broadcasting stream in order to receive trigger stream.

[00614] As mentioned above, NRT service is signaled through SMR or NST, and may be uniquely identified through 16 bit service identifier (sevice_id). Additionally, content items constituting NRT service may be identified through conent_lengate or a content identifier in NCT or NRT-IT. Accordingly, trigger service may be transmitted like NRT service by extending app_id_byte through DST. For example, app_id_byte may include data combining a service identifier (service id) field of trigger service and a content_linkage field. Accordingly, the first 16 bits of app_id_byte

correspond to a service id field in SMT or NST, and the later 32 bits correspond to a content linkage field in NCT or NRT-IT.

[00615] As above, the broadcasting station may include trigger signaling information in tap() and transmits it through an application identification structure of DST when one stream is included in each channel.

[00616] Moreover, according to an embodiment, trigger signaling information may be transmitted through a protocol_encapsulation field of DST. For example, if app_id_byte_length in DST is set with 0x0000, app id is not allocated. If protocol_encapsulation has 0x0F, it indicates that trigger signaling information is included in a corresponding tap() structure. Accordingly, a receiver may receive trigger signaling information from the corresponding tap() structure if app_id_byte_length is 0x0000 and protocol_encapsulation is 0x0F. Through this, a PID value on PMT indicating trigger stream is obtained and trigger stream is received as mentioned above.

[00617] Moreover, according to another embodiment, trigger signaling information may be transmitted through a content type descriptor field of DST.

[00618] As shown in Fig. 28, a content type descriptor structure in tap() on DST according to an embodiment is as follows.

[00619] A descriptorTag may have 0x72 to represent contentTypeDescriptor.

[00620] A descriptorLenth field represents the total length of a descriptor in a byte unit.

[00621] A contentTypeByte field represents a MIME media type

value of data referenced by tap connected to the descriptor. The MIME media type is defined in 5 of RFC2045 section [8].

[00622] Accordingly, a content type descriptor may be added to a tap() structure including trigger signaling information according to an embodiment. Accordingly, a receiver may receive trigger signaling information from the corresponding tap() structure if app_id_byte_length is 0x0000 and content type descriptor of the tap() structure corresponds to the predetermined content. Through this, a PID value on PMT indicating trigger stream is obtained and trigger stream is received as mentioned above. The MIME media type may be designated with a specific type to identify that there is trigger service signaling information through a content type descriptor.

[00623] As mentioned above, one NRT service may be a trigger service for transmitting trigger stream and may transmit respectively different stream to content items in the trigger service. In this case, each application may include one trigger stream.

[00624] Accordingly, an embodiment may include trigger stream in each content item of NRT service and may transmit it. In this case, the above-mentioned application identification structure may be used. For example, if app_id_byte_length is 0x0003, it indicates that trigger stream is transmitted through one NRT service by using one service identifier. If app_id_byte_length is 0x0007, it indicates that trigger stream is transmitted by each content item by using a service identifier and content linkage. If defined as above, each trigger stream may be transmitted in correspondence to each NRT service or content item. Since the next stage of a method

of transmitting and receiving trigger stream is identical to that of transmitting one trigger stream for each virtual channel, overlapping description will be omitted.

[00625] Fig. 29 is a view illustrating a syntax of PMT and service identifier descriptor according to an embodiment.

[00626] As shown in Fig. 29, a Program Map Table (PMT) represents information of a program broadcasted in each channel. A Program AssociationTable (PAT), in which 'packet ID' is defined as '0x00' and transmitted, may receive PMT by parsing 'packet ID' of PMT.

[00627] Moreover, a service identifier descriptor may be included in a descriptor loop for each ES of PMT. Then, it may include list information of services in each program element.

[00628] A structure of the service identifier descriptor will be described as follows.

[00629] A descriptor_tag field indicates that the descriptor is service_id_descriptor() and may have 0xC2.

[00630] A descriptor_length field represents a byte unit length from this field to the termination of the descriptor.

[00631] A service_count field indicates the number of services in a program element having the descriptor.

[00632] A service_id field indicates a service identifier in a program element having the descriptor.

[00633] According to an embodiment, trigger stream may be transmitted through a well-known IP address. Moreover, in order to signal a trigger, a broadcasting station may include a specific service identifier (service id, for example, 0x01) corresponding trigger stream in a service identifier descriptor and may transmit it. That is, trigger signaling

information on receiving trigger stream may be transmitted through a service identifier descriptor. Accordingly, if a service identifier of service_id_descriptor in an ES descriptor loop in an ES loop of PMT is 0x01, the receiver determines that elementray_PID in the ES loop is PID indicating trigger stream and receives the trigger stream through the PID.

[00634] Fig. 30 is a view illustrating a trigger stream descriptor according to an embodiment. According to an embodiment, a trigger may be signaled using a trigger stream descriptor. Like the above service identifier descriptor, the trigger stream descriptor may be included in an ES descriptor loop in an ES loop of PMT. Accordingly, if there is trigger stream, a trigger stream descriptor may exist in an ES descriptor loop. If identifying a trigger stream descriptor, a receiver may receive trigger stream by obtaining PID of the trigger stream from elementary_PID in a corresponding ES loop.

[00635] Like this, a trigger stream descriptor for transmitting trigger signaling information may include at least one of a service identifier (target service id) of TDO, a trigger target in trigger stream, and an IP address list transmitting trigger stream. The trigger stream descriptor of Fig. 30 is provided according to an embodiment and its structure will be described as follows.

[00636] A descriptor_tag field indicates a trigger_stream_descriptor if set with a predetermined value.

[00637] A descriptor_length field represents a byte unit length from this field to the termination of the descriptor.

[00638] A target_service_count field represents the number of target NRT service (TOD) of at least one trigger in trigger

stream.

[00639] A target_service_id field represents a service identifier (service_id) of target NRT service (TOD) of at least one trigger in trigger stream. A receiver may identify a service identifier (service_id) before receiving trigger stream by using the target service id field.

[00640] A target_content_item_count field represents the number of target NRT service content items of at least one trigger in trigger stream.

[00641] A target_content_linkage field represents a target NRT service content item linkage (content_linkage) of at least one trigger in trigger stream.

[00642] Moreover, a trigger stream descriptor is provided according to an embodiment, and thus, it is apparent that it may include additional information or have another configuration. For example, when one trigger stream is transmitted for each channel, a content item field may be omitted. Additionally, at least one of a trigger stream identification information field and a profile information field may be added to identify trigger stream.

[00643] A broadcasting station may transmit list information of trigger target NRT service such as TDO by using the trigger stream descriptor. Additionally, the broadcasting station may transmit trigger signaling information by using the target_service_id and targe_content_linkage fields if there is another trigger according to a content item. Additionally, a trigger stream descriptor may further include a list of IP address information or port numbers transmitting trigger stream.

[00644] According to an embodiment, a broadcasting station

designates a stream type and transmits trigger signaling information. A receiver extracts trigger signaling information by using a stream type from PMT and receives trigger stream through the trigger signaling information. For example, 0x96, one of stream types set preliminarily at the present, may be designated as trigger stream. In this case, a typical receiver has no information that a stream type is 0x96 and thus may not process trigger stream and disregard it. Accordingly, backwards compatibility for sub model receiver is guaranteed.

[00645] According to an embodiment, a trigger may be included in an Application information Table (AIT) for transmitting application information in data broadcasting such as Multimedia Home Platform (MHP) or Advanced Common application platform (ACAP), and may be transmitted. Fig. 31 is a view of AIT according to an embodiment.

[00646] Moreover, according to another embodiment a trigger may be included in a descriptor of STT to refer to a System Time Table (STT) as a trigger time, and then transmitted. Fig. 32 is a view of STT according to an embodiment.

[00647] Fig. 33 is a block diagram illustrating a transmitter for transmitting TDO and a trigger according to an embodiment.

[00648] Referring to Fig. 33, the transmitter 200 includes an NRT service transmitting unit 210, a trigger transmitting unit 220, a multiplexing unit 230, and a demodulation unit 240. The NRT service transmitting unit 210 includes an NRT service (TDO) generating unit 211 and an NRT service signaling data generating unit 212. The trigger transmitting unit 220 includes a trigger generating unit 221 and a trigger signaling data generating unit 222.

[00649] The NRT service (TDO) generating unit 211 receives data for NRT service generation from a service provider to generate the NRT service, packetizes the generated NRT service into IP datagram, and then packetized the packetized IP datagram into a transmission packet (TP). The packetized NRT service data is transmitted to the multiplexing unit 230.

[00650] The NRT service generating unit 211 transmits metadata including channel information about NRT service in transmission and service_id, to the NRT service signaling data generating unit 212. Additionally, if the generated NRT service is TDO, the NRT service generating unit 211 extracts trigger information including a trigger time for triggering TDO, identification information, and trigger action information of a target TDO, and then transmits it to the trigger generating unit 221.

[00651] The NRT service signaling data generating unit 212 generates NRT service signaling data for receiving NRT service by using the NRT service metadata, and packetizes the generated NRT service signaling data to the transmission packet (TP) to transmit it to the multiplexing unit 230.

[00652] Additionally, the trigger generating unit 221 generates trigger data by using trigger information of the TDO received from the NRT service (TDO) generating unit. The generated trigger data is packetized into a transmission packet to transmit it to the multiplexing unit 230. Moreover, the trigger generating unit 221 transmits metadata for receiving a trigger such as the packet identifier (PID) of the transmitted trigger data to the trigger signaling data generating unit 222.

[00653] The trigger signaling data generating unit 22

generates trigger signaling data on the basis of the received metadata, and packetizes the trigger signal in data into a transmission packet to transmit it to the multiplexing unit 230.

[00654] The multiplexing unit 230 multiplexes the received transmission packets by each channel, and then transmits the multiplexed signal to the modulation unit 240.

[00655] The modulation unit 240 modulates the multiplexed signal and transmits it to the external. The modulation method may vary, and the present invention is not limited thereto.

[00656] Fig. 34 is a block diagram illustrating a receiver for receiving TDO and a trigger according to an embodiment.

[00657] Referring to Fig. 34, the receiver 300 includes a demodulation unit 310, a demultiplexing unit 320, a trigger processing unit 330, an NRT service processing unit 340, and a service manager 350. The trigger processing unit 330 includes a trigger receiving unit 331 and a trigger signaling data receiving unit 332. The NRT service processing unit 340 includes an NRT service (TDO) receiving unit 341 and an NRT service signaling data receiving unit 342.

[00658] The demodulation unit 310 receives a modulated signal from the transmitter 200, and demodulates the received signal according to a predetermined demodulation method to transmit it to the demultiplexing unit 320.

[00659] The demultiplexing unit 320 demultiplexes the demodulated signal to restore an original transmission packet for each channel to transmit them to each receiving unit of the trigger processing unit 330 or the NRT service processing unit 340.

[00660] The NRT service signaling data receiving unit 342 receives and restores the packetized NRT service signaling data from the multiplexing unit 320 to extract information on NRT service, and then transmits it to the NRT service (TDO) receiving unit 341. The NRT service (TDO) receiving unit 341 receives transmission packets of NRT service from the multiplexing unit 320 by using information on receiving NRT service, and restores it as service data to transmit it to the service manager 350.

[00661] Moreover, the NRT service signaling data receiving unit 332 receives and restores the packetized trigger signaling data from the multiplexing unit 320, extract information on receiving a trigger, and then, transmits it to the trigger receiving unit 331. The trigger receiving unit 331 receives transmission packets including a trigger from the multiplexing unit 32 by using information on receiving a trigger, and restores trigger data to transmit it to the service manager 350.

[00662] The service manager 350 receives at least one of trigger data or NRT service (TDO) data from the trigger processing unit 330 or the NRT processing unit 340. Moreover, the service manager 350 performs and applies a trigger action on a trigger target TDO at the trigger timing, so that a trigger action on TDO is performed.

[00663] Fig. 35 is a flowchart illustrating a trigger transmitting method according to an embodiment.

[00664] Referring to Fig. 35, the NRT service generating unit 211 generates NRT service data by receiving NRT service data from external or on the basis of data received from the NRT service provider in operation S100. Moreover, the NRT service

generating unit 211 packets the generated data into a transmission packet. Additionally, the NRT service generating unit 211 transmits information on receiving transmission packets including NRT service to the NRT service signaling data generating unit 212.

[00665] Then, the NRT service signaling data generating unit 212 generates the above described NRT service signaling data and packetizes it into a transmission packet in operation S110.

[00666] Moreover, the NRT service generating unit 211 determines whether the generated NRT service is a trigger declarative object, i.e., TDO in operation S120.

[00667] Additionally, if the generated NRT service is TDO, the NRT service generating unit 211 transmits trigger information including a trigger time for triggering TDO, trigger action, target TDO identification information, to the trigger generating unit 221, and the trigger generating unit 211 generates trigger data by using the received triggered information in operation S130. The generated trigger data is packetized into a transmission packet and transmitted to the multiplexing unit. For example, a target service identifier for target TDO and trigger action information applied to a target service may be inserted into a packetized stream, i.e., the payload of PES, and then transmitted. Additionally, trigger time information is designated into a PTS or DTS format, inserted into the payload or header of PES, and then is transmitted. When the synchronized data streaming method is used, PTS of trigger stream and PTS of video and audio stream are synchronized to set the accurate play timing.

[00668] Moreover, the trigger signaling data generating unit

222 generates trigger signaling data for identifying and receiving a trigger transmitted from the trigger generating unit 221 and packetized the generated trigger signaling data into a transmission packet to transmit it to the multiplexing unit in operation S140. Here, the trigger signaling data may include a trigger stream descriptor or a service identifier descriptor, inserted in a program map table, and may include a packet identifier of trigger stream corresponding to each descriptor. Additionally, trigger signaling data may include a packet identifier of trigger stream in a TAP structure of DST.

[00669] Later, the multiplexing unit 230 multiplexes at least one of transmission-packetized NRT service data, NRT service signaling data, trigger data, and trigger signaling data by each transmission channel and then transmits it to the modulation unit 240.

[00670] Moreover, the modulation unit 240 performs modulation to transmit the multiplexed signal and transmits it to external receiver or a broadcasting network in operation S160.

[00671] Fig. 36 is a flowchart illustrating an operation of a receiver 300 according to an embodiment.

[00672] First, when the receiver 300 is turned on, a channel is selected by a user or a predetermined channel is selected in operation S200. The demodulation unit 310 demodulates the received signal from the selected channel, and the demultiplexing unit 320 demultiplexes the demodulated signal by each transmission channel. Also, the NRT service receiving unit 341 and the NRT service signaling data receiving unit 342 receive NRT service data and transmit it to the service manager 350 as described above.

[00673] Then, the trigger signaling data receiving unit 332 or the NRT service signaling data receiving unit 342 confirms whether trigger reception is possible in operation s220. The trigger reception confirmation may use one of the above-mentioned methods. That is, the trigger signaling data receiving unit 332 or the NRT service signaling data receiving unit 342 uses one of a method of confirming PID corresponding to a trigger in MGT or PSIP based PID, a method of using a tap structure of DST, a method of using a service identifier descriptor or a trigger stream descriptor, a method of using a trigger stream type, and a method of using AIT or STT, in order to confirm whether trigger reception is possible.

[00674] Moreover, when it is confirmed that trigger reception is possible, the trigger signaling data receiving unit 332 receives a transmission packet including trigger signaling data to restore the trigger signaling data, and then transmits it to the trigger receiving unit 331 in operation S230.

[00675] Later, the trigger receiving unit 331 extracts trigger data from the received transmission packet by using the trigger signaling data, and transmits it to the service manager 350 in operation S240. For example, the trigger receiving unit 331 may receive trigger stream by using a packet identifier corresponding to the trigger stream descriptor. Additionally, the trigger receiving unit 331 extracts trigger information from trigger stream and transmits it to the service manager 350. Additionally, if the received trigger stream is PES, PTS in the header of PES is extracted as a trigger time, and a target service identifier

and trigger action in the payload of PES are extracted, in order to transmit them to the service manager 350.j

[00676] Moreover, the service manager 350 performs a trigger action on a target TDO at the trigger timing, so that a trigger action on TDO is performed in operation S250. Especially, if the PTS of PES is a trigger time, the PTS of trigger stream is synchronized with the PTS in the header of audio and video stream, to satisfy the accurate play timing.

[00677] Fig. 37 is a flowchart illustrating a trigger receiving method by using a trigger table according to an embodiment.

[00678] The demodulation unit 310 receives and demodulates a broadcast signal for selected channel. Moreover, the trigger signaling data receiving unit 332 receives a PSIP table through the demultiplexing unit 320 and determines whether there is a trigger table in the received table to identify a trigger service in operation S310. The trigger signaling data receiving unit 332 searches PID allocated to a trigger table from an MGT or PSIP based table, or searches a table corresponding to Table_id allocated to a trigger table to identify a trigger service.

[00679] If the trigger service is not identified, the receiver 300 provides general broadcasting services.

[00680] Moreover, if the trigger service is identified, the trigger receiving unit 331 receives the searched trigger table and parses it in operations S320 and S330.

[00681] Then, the service manager 350 receives trigger information including trigger time, trigger action, and target TDO identification information parsed in the trigger table, and performs a corresponding trigger action on a

corresponding TDO at the corresponding trigger timing in operation S340.

[00682] Fig. 38 is a flowchart illustrating an operation of a receiver 300 when trigger signaling information and trigger are transmitted using DST according to an embodiment.

[00683] When a physical transmission channel is selected in operation S3000 and a channel selected by a tuner is tuned, the receiver 300 obtains VCT and PMT from a broadcast signal received through the tuned physical transmission channel by using the demodulation unit 310 and the demultiplexing unit 320 in operation S3010. Then, the PSI/PSIP section handler or the trigger signaling data receiving unit 332 or the NRT service signaling data receiving unit 342 parses the obtained VCT and PMT to confirm whether there is NRT service.

[00684] For example, when the service_type field value of VCT is not 0x04 or 0x08, since the corresponding virtual channel does not transmit NRT only service, the receiver 300 operates properly according to information in the virtual channel. However, even though the service_type field value does not mean NRT only service, the corresponding virtual channel may include NRT service. This case is called adjunct NRT service included in the corresponding virtual channel, and the receiver 300 may perform the same process as the case of receiving NRT service.

[00685] Then, the NRT service signaling data receiving unit 342 or the trigger signaling data receiving unit 332 determines that NRT service is received through a corresponding virtual channel if a service_type field value is 0x04 or 0x08. In this case, if a stream_type field value in a service location descriptor of VCT (or an ES loop of

PMT) is 0x95 (i.e., DST transmission), DST is received using an Elementary_PID field value in operation S3020. This may be performed in the demultiplexing unit 320 according to a control of the service manager 350.

[00686] Also, the trigger signaling data receiving unit 342 identifies a trigger service from the received DST in operation S3040. A method of identifying a trigger service uses one of a method of identifying a specific value allocated to app_id_description and app_id_byte by using an application identification structure, a method of identifying a specific value allocated to a protocol_encapsulation field, and a method of identifying tap including a content type descriptor.

[00687] If the trigger service is not identified from the received DST, since trigger data transmits general NRT service through a corresponding virtual channel, the receiver 300 operates properly according to NRT service in the corresponding virtual channel in operation S3030.

[00688] Moreover, when the trigger service is identified from DST, the trigger signaling data receiving unit 332 extracts tap from DST including trigger signaling information (PID of trigger stream) in operation S3060.

[00689] Then, the trigger signaling data receiving unit 332 extracts stream PID from PMT including association_tag of the extracted Tap in operation S3070.

[00690] The trigger receiving unit 331 receives MPEG-2 TS packets corresponding to the extracted stream PID, and removes decapsulation, i.e., TS header, to restore PES stream including trigger stream. The stream_type of a PES packet including trigger stream may be 0x06 representing

synchronized data stream. The trigger receiving unit 331 parses at least one of PTS of a PES packet header from the restored PES stream, a target TDO identifier in trigger stream, a trigger identifier, or trigger action information in operation S3070.

[00691] Then, the service manager 350 performs an action on the target TDO at the trigger timing by using the PTS of the PES packet header including a trigger as the trigger timing in operation S3080. Here, the target TDO may be NRT service indicated by the parsed target TDO identifier. Additionally, the action may be one of preparation, execution, extension, and termination commands provided from the parsed trigger action information.

[00692] Fig. 39 is a flowchart illustrating an operation of a receiver 300 when a trigger is transmitted using a trigger stream descriptor according to an embodiment.

[00693] When a physical transmission channel is selected in operation S3000 and a channel selected by a tuner is tuned, the receiver 300 obtains VCT and PMT from a broadcast signal received through the tuned physical transmission channel by using the demodulation unit 310 and the demultiplexing unit 320 in operation S4000. The broadcast signal includes VCT and PMT, and the trigger signaling data receiving unit 332 or the PSI/PSIP section handler parses the obtained VCT and PMT.

[00694] Also, the trigger signaling data receiving unit 332 confirms whether a trigger is transmitted from the VCT and PMT to a corresponding virtual channel. For this, the trigger signaling data receiving unit 332 determines whether there is the Trigger_stream_descriptor in the ES descriptor loop corresponding to a corresponding virtual channel in operation

S4020. Whether there is Trigger_stream_descriptor is determined by using whether a stream_type value is 0x06(synchronized data streaming) and a descriptor_tag field of a corresponding descriptor is identical to a value set to correspond to a trigger stream descriptor after searching descriptors in an ES descriptor loop.

[00695] If it is determined that Trigger_stream_descriptor is not identified from PMT and thus there is no Trigger_stream_descriptor, since a corresponding virtual channel does no transmit a trigger, the receiver 300 operates properly according to broadcast service in the corresponding virtual channel in operation S4025.

[00696] Then, if there is Trigger_stream_descriptor, the trigger signaling data receiving unit 332 extracts Elementary_PID in the corresponding ES loop of PMT in operation S4030. The extracted stream PID may be a PID value of stream including trigger stream.

[00697] Then, the trigger receiving unit 331 receives MPEG-2 TS packets corresponding to the extracted stream PID, and performs decapsulation (i.e., removes a TS header) to restore PES stream including trigger stream. The stream_type of a PES packet including trigger stream may be 0x06 representing synchronized data stream. The trigger receiving unit 331 parses at least one of PTS of a PES packet header from the restored PES stream, a target TDO identifier in trigger stream, a trigger identifier, or trigger action information in operation S4040.

[00698] Then, the service manager 350 performs an action on the target TDO at the trigger timing by using the PTS of the PES packet header including a trigger as the trigger timing

in operation S4050. Here, the target TDO may be NRT service indicated by the parsed target TDO identifier. Additionally, the action may be one of preparation, execution, extension, and termination commands provided from the parsed trigger action information.

[00699] Fig. 40 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using a stream type according to an embodiment.

[00700] When a physical transmission channel is selected and a channel selected by a tuner is tuned, the receiver 300 obtains VCT and PMT from a broadcast signal received through the tuned physical transmission channel by using the demodulation unit 310 and the demultiplexing unit 320. The broadcast signal includes VCT and PMT, and the trigger signaling data receiving unit 332 or the PSI/PSIP section handler parses the obtained VCT and PMT in operation S400.

[00701] Also, the trigger signaling data receiving unit 332 confirms whether a trigger is transmitted from the VCT and PMT to a corresponding virtual channel. For this, the trigger signaling data receiving unit 332 determines whether there is 0x96, i.e., the specific stream type in the ES descriptor loop corresponding to a corresponding virtual channel in operation S410.

[00702] If it is determined that 0x96 is not identified from stream type and thus there is no stream type, since a corresponding virtual channel does no transmit a trigger, the receiver 300 operates properly according to broadcast service in the corresponding virtual channel in operation S415.

[00703] Then, if the stream type is 0x96, the trigger signaling data receiving unit 332 extracts Elementary_PID in

the corresponding ES loop of PMT in operation S420. The extracted stream PID may be a PID value of stream including trigger stream.

[00704] Then, the trigger receiving unit 331 receives MPEG-2 TS packets corresponding to the extracted stream PID, and performs decapsulation (i.e., removes a TS header) to restore PES stream including trigger stream. The trigger receiving unit 331 parses at least one of PTS of a PES packet header from the restored PES stream, a target TDO identifier in trigger stream, a trigger identifier, or trigger action information in operation S430.

[00705] Then, the service manager 350 performs an action on the target TDO at the trigger timing by using the PTS of the PES packet header including a trigger as the trigger timing in operation S440. Here, the target TDO may be NRT service indicated by the parsed target TDO identifier. Additionally, the action may be one of preparation, execution, extension, and termination commands provided from the parsed trigger action information.

[00706] Fig. 41 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using AIT according to an embodiment.

[00707] The trigger signaling data receiving unit 332 receives AIT by using the demodulation unit 310 and the demultiplexing unit 320 in operation S500.

[00708] Also, the trigger signaling data receiving unit 332 confirms whether a trigger is transmitted from AIT. For this, the trigger signaling data receiving unit 332 confirms whether there is a trigger descriptor in AIT in operation S510.

[00709] If it is determined that there is no trigger descriptor, since a corresponding application does not include a trigger, the receiver 300 operates properly according to corresponding application service in operation S515.

[00710] Also, if there is a trigger descriptor, the trigger receiving unit 332 extracts trigger data from the trigger descriptor and parses the extracted trigger data to transmit it to the service manager 350 in operation S530.

[00711] Then, the service manager 350 performs an action on the target TDO at the trigger timing by using the parsed trigger data in operation S540. Here, the target TDO may be NRT service indicated by the parsed target TDO identifier. Additionally, the action may be one of preparation, execution, extension, and termination commands provided from the parsed trigger action information.

[00712] Fig. 42 is a flowchart illustrating an operation of a receiver when a trigger is transmitted using STT according to an embodiment.

[00713] The trigger signaling data receiving unit 332 receives STT by using the demodulation unit 310 and the demultiplexing unit 320 in operation S600.

[00714] Also, the trigger signaling data receiving unit 332 confirms whether a trigger is transmitted from STT. For this, the trigger signaling data receiving unit 332 confirms whether there is a trigger descriptor in STT in operation S610.

[00715] If it is determined that there is no trigger descriptor, since a corresponding STT does not include a trigger, the receiver 300 operates properly according to a

broadcast signal in operation S615.

[00716] Also, if there is a trigger descriptor, the trigger receiving unit 332 extracts trigger data from the trigger descriptor and parses the extracted trigger data to transmit it to the service manager 350 in operation S630.

[00717] Then, the service manager 350 performs an action on the target TDO at the trigger timing by using the parsed trigger data in operation S540. Here, the target TDO may be NRT service indicated by the parsed target TDO identifier. Additionally, the action may be one of preparation, execution, extension, and termination commands provided from the parsed trigger action information.

[00718] Hereinafter, according to an embodiment of the present invention, a method of providing information on NRT service in linkage with NRT service, real-time broadcast, and an internet network will be described.

[00719] As mentioned above, in an actual broadcast system, one broadcast program may include at least one application executed in linkage with a corresponding program, and such an application may be stored in the receiver 300 and launched through the above-mentioned NRT service receiving method.

[00720] However, since a PSIP table does not include information on an NRT service linked with a broadcast program, when the receiver 300 is to receive an NRT service linked with a broadcast program, an NRT-IT of a channel including a specific broadcast program all needs to be parsed. Also, even when the receiver 300 receives the NRT-IT first to receive a content item of NRT service, it is difficult to identify a broadcast program linked with the content item. Additionally, since the PISP table does not include information on an

internet network, the receiver 300 receives only a real-time broadcast program, so that it is difficult to recognize related service transmitted through an internet network. Additionally, for the same reason as above, it is difficult for the receiver 300 to receive extended EPG on a specific real-time broadcast program, which is transmitted through an internet network. Accordingly, a systematic linkage between a real-time broadcast program, NRT service, and an internet network is required.

[00721] According to an embodiment of the present invention, the transmitter 200 may include information, which is necessary for linking to NRT service information such as an NRT or a TDO or an internet network, in a PSIP table corresponding to a specific channel or program, and then transmit it. Moreover, PSIP table information corresponding to specific real-time broadcast channel information or program information is included in NRT service, and then transmitted.

[00722] Accordingly, the receiver 300 receives a real-time broadcast channel or program, and parses the PSIP table, so that it may selectively receive an NRT service linked with a specific broadcast program. Furthermore, the receiver 300 receives a real-time broadcast channel or program, and parses the PSIP table, so that it may receive an NRT service linked with a specific broadcast channel or program, through an internet network. Then, the receiver 300 receives NRT service to obtain PSIP table information on a broadcast program linkable with the NRT service, and then provides it to a user.

[00723] According to an embodiment of the present invention, a descriptor providing information linked with real-time

broadcast, for example, one of NRT service, preview information, extended EPG information, highlight information, and related internet portal information, may be included in one of PSIP tables and then, transmitted. According to an embodiment of the present invention, such a descriptor may be referred to as a link descriptor.

[00724] Additionally, according to an embodiment of the present invention, information on a broadcast program linkable with NRT service, for example, broadcast channel information or program information, may be included in one of an SMT and an NRT such as an NRT-IT and then, transmitted. According to an embodiment of the present invention, such a descriptor may be referred to as an event descriptor.

[00725] Fig. 43 is a view illustrating a syntax of a link descriptor (link_descriptor) according to an embodiment of the present invention. Figs. 44 and 45 are views illustrating the contents of fields included in a link descriptor.

[00726] As shown in Fig. 43, the link descriptor includes a descriptor_tag field, a descriptor_length field, a number_of_links field, and a plurality of link data items. Each of the plurality of link data items includes a link_type field, a link_media field, a mime_type_length field, a mime_type field, a description_length field, a description, link_length field, and a link_byte field.

[00727] The descriptor_tag field may be an 8-bit unsigned integer to distinguish this descriptor as a link descriptor. For example, this field may have a 0xe8 value.

[00728] The descriptor_length field may be an 8-bit unsigned integer to define the length from a field immediately following this field to the end of the link descriptor.

[00729] The number_of_links field may be an 8-bit unsigned integer to indicate the number of the plurality of link data items.

[00730] The link_type field indicates the type of related service receivable based on the link_byte field. For example, as shown in Fig. 44, the link type field may indicate type information of specific service receivable based on the link_byte field included in a link data item.

[00731] For example, a link data item having a link_type field of 0X01 may relate to an HTML page of a broadcast program portal. For example, a link data item having a link_type field of 0X01 may relate to an HTML page of a broadcast program portal. A link data item having a link_type field of 0X03 may relate to a Preview Clip service of a broadcast program. A link data item having a link_type field of 0X04 may relate to a of an Extended EPG of a broadcast program. A link data item having a link_type field of 0X05 may relate to a highlight service of a broadcast program. A link data item having a link_type field of 0X06 may relate to a Multiview service of a broadcast program. Also, a link data item having a link_type field of 0X07 may relate to a TDO service linkable to a broadcast program.

[00732] The link_media field may be an 8-bit unsigned integer indicating a transmission medium to receive related service on the basis of a link data item. For example, as shown in Fig. 45, when the link_media field is 0x02, it indicates that related service receivable based on a link data item is receivable through NRT service. Additionally, when the link_media field is 0x03, it indicates that related service is receivable through internet service.

[00733] The MIME_type_length field may be an 8-bit unsigned integer indicating the length of MIME type information indicating the MIME protocol type of the link_byte field.

[00734] The MIME_type field may indicate the protocol type of the link_byte field. The MIME type may indicate one type of text, multipart, massage, application, image, audio, and video, for example.

[00735] The descriptor_length field may be an 8-bit unsigned integer to indicate a description field indicating detailed information of a corresponding link.

[00736] The description field may indicate detailed information of a corresponding link.

[00737] The link_length field may be an unsigned integer to indicate the length of the link_byte field.

[00738] The link_type field may be a field indicating data to receive the above-mentioned broadcast program related service. The link_byte field may include identification information of contents receivable through link medium. For example, as shown in Fig. 45, if the link medium is NRT service, the link_byte field may include service_id and content_linkage to receive NRT service. Additionally, if the link medium is internet service, the link_byte field may include an uniform resource identifier (URL), i.e., internet address information, or URL information.

[00739] Figs. 46 and 47 are views illustrating a relationship between each table when the link descriptor of Fig. 43 is included in a descriptor of an event information table (EIT) in a PSIP table.

[00740] As shown in Fig. 46, the link descriptor may be included in a descriptor loop of the EIT indicating broadcast

program information and then, transmitted.

[00741] The EIT in addition to a VCT is included in a PSIP and then, transmitted. Since the VCT includes source_id corresponding to a selected virtual channel_number, the receiver 300 parses an EIT including same source_id as the VCT to obtain information on each broadcast program provided from each virtual channel. Each broadcast program is identified with event_id

[00742] Accordingly, the receiver 300 extracts the link descriptor from a loop including event_id of a specific broadcast program in an event loop in the parsed EIT, and receives information linked with the specific broadcast program by using the extracted link descriptor through NRT service or an internet network.

[00743] For example, as shown in Fig. 46, if the link descriptor may include service_id and content_linkage of the NRT service linked with the specific broadcast program. In this case, the receiver 300 recognizes content item position information of the linked NRT service through an SMT, an NRT-IT, and an FDT, and downloads a content item through the recognized position information.

[00744] In more detail, the receiver 300 may obtain service signaling channel information corresponding to service_id included in a link_byte field of the link descriptor, from the SMT. At this point, the service signaling channel information may include an IP address and a port number. Moreover, the receiver 300 may obtain a list of content_id in a service corresponding to service_id included in a link_byte field of the link descriptor, from the NRT_IT. The receiver 300 may recognize the identifier of a content item to be

downloaded corresponding to service _id in the NRT_IT through a content linkage field in a link_byte field of the link descriptor. Then, the receiver 300 may recognize the positions of content item files corresponding to content_id in NRT_IT from an FLUTE FDT received through the IP address and port number of the SMT. Accordingly, the receiver 300 receives files configuring a content item of NRT service linked with a corresponding broadcast program by using the FLUTE FDT through a FLUTE session. Moreover, the receiver 300 extracts URL information of content items, for example, a URL list of files, linked with a broadcast program from an internet location descriptor described later in the NRT-IT, and based on the extracted list, receives files configuring a content item of an NRT service linked with a broadcast program through a network.

[00745] Furthermore, the link descriptor may include at least one of portal HTML page access information linked with a broadcast, Thumbnail reception information, Preview Clip reception information, Extended EPG reception information, highlight reception information, Multiview reception information, and linkable TDO reception information, in addition to NRT service linked with a broadcast program. The link descriptor may include information to receive such reception information through NRT service or an internet network.

[00746] Additionally, as shown in Fig. 47, a link_byte field of a link data item in the link descriptor may include URL to receive a service linked with a specific broadcast program. In this case, the receiver 30 accesses an internet site according to internet address information in the link_byte

field, and receives a broadcast program related service through an internet network. The broadcast program related service received through an internet network may include at least one of NRT service, thumbnail, preview clip, extended EPG, highlight, multiview, and TOD service.

[00747] In more detail, the receiver 300 obtains an EIT corresponding to a selected virtual channel, obtains the link descriptor from a descriptor loop of an event loop (i.e., a for loop including event_id) corresponding to a selected broadcast program in the obtained EIT, extracts an internet address from the obtained the link descriptor, and receives a service linked with a selected broadcast program in access to an internet network according to the extracted internet address. Accordingly, a broadcast service provider may provide a service linked with a specific broadcast program through real-time broadcast. Moreover, the receiver 300 may access an internet network by using internet address information, and may receive a service linked with a corresponding broadcast program through an internet network.

[00748] Fig. 48 is a view illustrating a syntax of an event descriptor (Event_descriptor) and contents of fields in the event descriptor according to an embodiment of the present invention.

[00749] As shown in Fig. 48, the event descriptor includes a descriptor_tag field, a descriptor_length field, and an ETM_id field.

[00750] The descriptor_tag field may be an 8-bit unsigned integer to distinguish this descriptor as a link descriptor. For example, this field may have a 0xe8 value.

[00751] The descriptor_length field may be an 8-bit unsigned

integer to define the length from a field immediately following this field to the end of the link descriptor.

[00752] The ETM_id field may be a 32-bit variable indicating broadcast channel information or broadcast program information linked with an NRT service including the event descriptor. The receiver 200 inserts a specific channel and program information into the event descriptor by using the ETM_id field, and includes the event descriptor in an NRT table signaling NRT service to transmit it. Additionally, the ETM_id field corresponds to ETM_id in an extended text table (ETT) providing additional information of a broadcast program and is used to provide additional information of a broadcast program.

[00753] In more detail, as shown in Fig. 48, when the ETM_id field includes broadcast channel information, it may be classified as channel ETM_id. In this case, the first 16 bits from the most significant bit (MSB) correspond to source_id of a corresponding virtual channel to identify a broadcast channel, and the remaining bits may be 0. Accordingly, the receiver 300 matches source_id with VCT of PSIP, so that it may identify a linkage channel of an NRT service including the event descriptor.

[00754] Additionally, when the ETM_id field includes all of broadcast program information and broadcast channel information, the least significant bit (LSB) is configured with 10. The first 16 bits from the MSB correspond to source_id of a corresponding channel and the 14 bits after that correspond to event_id of a corresponding program. Accordingly, the receiver 300 identifies a linkage channel of NRT service including the event descriptor and also

identifies a broadcast program linked with NRT service by matching event_id with an EIT of PSIP.

[00755] Fig. 49 is a view illustrating a method of identifying a linkage program through the event descriptor according to an embodiment of the present invention.

[00756] For example, as shown in Fig. 49, the service manager 350 of the receiver 300 identifies service_id through an SMT and receives and parses a content item of an NRT service corresponding to service_id identified by an NRT-IT.

[00757] Then, when the service manager 350 of the receiver 300 finds a specific part of the parsed NRT-IT, for example, the event descriptor (Event_descriptor) in a content loop, it may obtain ETM_id including broadcast program information linked with NRT service from the found event descriptor.

[00758] According to an embodiment of the present invention, the service manager 350 of the receiver 300 may obtain source_id of a channel including a broadcast program and event_id of a broadcast program from ETM_id. As mentioned above, the receiver 300 corresponds the first 16 bits of ETM_id to source_id, and 14 bits after that to event_id, so that it may obtain source_id and event_id.

[00759] Then, the service manager 350 of the receiver 300 may identify a virtual channel matching source_id from a VCT, and may identify an event loop of a broadcast program matching event_id from an EIT of a corresponding virtual channel. The service manager 350 of the receiver 300 receives NRT server related broadcast program information from the identified event loop and then, provides the information to a user. The broadcast program information may include at least one of a start time of a corresponding broadcast program, an ETT

position, a broadcasting time, title information, and description information in the event loop of an EIT. Accordingly, the service manager 350 of the receiver 300 provides to a user the information on the received NRT service related broadcast program on the basis of the received broadcast program information.

[00760] Fig. 50 is a flowchart illustrating an operation of receiving by the receiver 300 broadcast program or broadcast channel related contents by using the link descriptor according to an embodiment of the present invention.

[00761] Referring to Fig. 50, when the receiver 300 is turned on, it receives a VCT or an EIT by receiving a PSIP table through a PSIP/PSI handler or service manager in operation S7000. The PSIP/PSI handler or service manager 350 of the receiver 300 parses each table of the PSIP table and obtains VCT or EIT from the parsed tables.

[00762] Also, the service manager 350 of the receiver 300 searches for a descriptor loop from the received VCT or EIT in operation S7010. Then, the service manager 350 of the receiver 300 determines whether there is a link descriptor through searching in operation S7020. The service manager 350 of the receiver 300 searches for descriptors in a descriptor loop and if a descriptor_tag field is identical to that of a predetermined link descriptor, it is determined that there is a link descriptor.

[00763] Also, when it is determined that there is a link descriptor, the service manager 350 of the receiver 300 extracts link information from the link descriptor, and based on the extracted link information, displays a broadcast program including the link descriptor or a broadcast channel

related content list in operation S7030. Link information may include at least one of a link_type field, a link_media field, a mime_type field, a description field, and a link_byte field. The service manager 350 of the receiver 300 generates a list of contents linked with a specific broadcast channel or broadcast program on the basis of the extracted link information, for example, at least one of a related internet site address, thumbnail, a preview clip, extended EPG information, highlight information, multiview information, and TDO information, and then displays the list to a user. Therefore, a user may select content to be received from the displayed broadcast channel or broadcast program related content list.

[00764] Then, the service manager 350 of the receiver 300 determines whether content is selected and there is a receive instruction in operation S7040. The content may be selected by a user or a predetermined process. The selected content may be content displayed on the above-mentioned displayed broadcast channel or broadcast program related content list. Moreover, a user may input a receive instruction on the selected content, and also a receive instruction on the selected content may be performed by a predetermined process.

[00765] Then, when a receive instruction on the selected content is executed, the service manager 350 of the receiver 300 receives corresponding content immediately or perform reception reservation in operation S7050. The case that the reception reservation is performed includes the case that NRT service cannot be received because an SMT is not received, the case that a user sets to receive content after a predetermined time, or the case that NRT service to be

received is TDO service that is real-time linked with a broadcast program.

[00766] Moreover, when the link information of content to be received includes a site address accessible through internet, the service manager 350 of the receiver 300 accesses a corresponding site and displays a broadcast channel or program related home page. Furthermore, when the content to be received is one of thumbnail, preview clip, extended EPG information, highlight information, and multiveiw information, the service manager 350 of the receiver 300 may download it through internet or another path and then, may display it.

[00767] Then, when contents reserved for reception is NRT service, the service manager 350 of the receiver 300 receives an SMT to identify corresponding NRT service in operation S7060. Then, the service manager 350 of the receiver 300 accesses a FLUTE session through the above-mentioned method in operation S7070 and identifies a packet for the content reserved for reception among packets of a content item file configuring NRT service in operation S7080. Then, the service manager 350 of the receiver 300 receives the identified packets for the content reserved for reception through FLUTE or an internet network in operation S7090.

[00768] In more detail, since the link information extracted from the link descriptor includes service_id and content_linkage information of NRT service, the service manager 350 of the receiver 300 receives and identifies the content reserved for reception, that is, a reserved packetized content item of NRT service, through the FLUTE session on the basis of the service_id and content_linkage information. The service manager 350 of the receiver 300, for

example, extracts NRT service identification information from link_byte in the link descriptor as shown in Fig. 46, and receives an NRT service linked with a corresponding program through an SMT, an NRT-IT, and a FLUTE. Accordingly, the service manager 350 of the receiver 300 provides the information on content linked with a broadcast channel or broadcast program that is broadcasted currently or in the future through a VCT or an EIT. Moreover, the service manager 350 of the receiver 300 receives the selected content immediately through an internet network or NRT service or receives the content reserved for reception after a predetermined time. For example, as mentioned above, the receiver 300 obtains URL information of content items on the basis of NRT_IT and receives files configuring the selected content item of NRT service on the basis of the obtained URL information through an internet network.

[00769] Fig. 51 is a flowchart illustrating an operation of providing by the receiver 300 broadcast program related content by using an event descriptor according to an embodiment of the present invention.

[00770] According to an embodiment of the present invention, as shown in Fig. 51, the transmitter 200 inserts broadcast program information linked with NRT service into the event descriptor and inserts the event descriptor into a content loop of NRT-IT to be transmitted to the receiver 300 through a broadcast channel or NRT service signaling channel.

[00771] When NRT-IT including the event descriptor is transmitted in such a way, a PSIP/PSI handler or service manager 350 of the receiver 300 parses a PSIP table to obtain a VCT and an EIT in operation S7100. The receiver 300 may

provide information on a broadcast channel and broadcast program selected through a VCT and an EIT to a user.

[00772] Also, the service manager 350 of the receiver 300 determines whether NRT service is transmitted through an NRT service signaling channel or an internet network, and receives an SMT and an NRT-IT when NRT service is transmitted in operation S7120.

[00773] Then, the service manager 350 of the receiver 300 searches for a content loop of an NRT-IT and parses descriptor_tag of each descriptor to identify descriptors in the content loop in operation S7120. Then, the service manager 350 of the receiver 300 determines whether there is an event descriptor in the parsed descriptor in operation S7130.

[00774] If there is an event descriptor, the service manager 350 of the receiver 300 displays broadcast channel information or broadcast program information in the event descriptor (for example, a broadcast channel number based on source_id or broadcast program EPG information based on event_id) in operation S7140, and represents that the information is linked with receivable NRT service.

[00775] Then, the service manager 350 of the receiver 300 determines whether the content of NRT service linked with a broadcast program is selected and there is a receive instruction in operation S7150. The content may be selected by a user or a predetermined process. For example, when a broadcast program that a user wants to view is selected, a receive instruction on NRT service linked with a corresponding broadcast program is performed. Moreover, when an NRT service that a user wants to receive is selected, a

content receiving instruction is executed.

[00776] Then, when a receive instruction on the selected content is executed, the service manager 350 of the receiver 300 receives corresponding content immediately or perform reception reservation in operation S7160. According to an embodiment of the present invention, the selected content may be NRT service. Accordingly, the service manager 350 of the receiver 300 accesses a FLUTE session or internet network on the basis of the pre-received SMT and NRT-IT to receive the selected content such as NRT service in operation S7170. For example, as mentioned above, the receiver 300 obtains URL information of content items on the basis of an NRT_IT and then, confirms packets of the selected content item file.

[00777] Then, as shown in Fig. 50, the service manager 350 of the receiver 300 identifies a packet for content reserved for reception among packets of a content item file configuring NRT service in operation S7180. Then, the service manager 350 of the receiver 300 receives the identified packets for the content reserved for reception through FLUTE or an internet network in operation S7190.

[00778] In such a way, the transmitter 200 inserts the event descriptor into a content loop of an NRT-IT and also includes information on a broadcast program linked with NRT service in the event descriptor to transmit it. Moreover, the service manager 350 of the receiver 300 provides to a user the information on a broadcast channel or broadcast program that is currently transmitted based on the event descriptor of an NRT-IT.

[00779] Moreover, NRT service such as TDO may be signaled through an SMT or an NRT-IT, as mentioned above. Especially,

if the service_category field of the SMT is a specific value, for example, it indicates that 0x0E, a service signaled through the SMT is NRT service. Additionally, the SMT may include a Service level descriptor to indicate an attribute of NRT service transmitted. A service level descriptor in the SMT may be in plurality and, for example, may be at least one of a Protocol Version Descriptor, an NRT Service Descriptor, a Capabilities Descriptor, an Icon Descriptor, an ISO-639 Language Descriptor, a Receiver Targeting Descriptor, a Genre Descriptor, and an ATSC Private information Descriptor. By using such a service level descriptor, the transmitter 200 transmits information on NRT service and the receiver 300 operates according to the NRT service information. However, a currently used service level descriptor may not include information specified to the TDO suggested by the present invention. Therefore, a service level descriptor to deliver information on a TDO is required.

[00780] Fig. 52 is a view illustrating a syntax of NRT_service_descriptor, that is, a service level descriptor.

[00781] As shown in Fig. 52, NRT_service_descriptor in service level descriptors may include information on NRT service transmitted. The information on NRT service may include auto update, the length of content, save reservation information, or consumption_model information.

[00782] Especially, a consumption_model field in a box may include information on a service providing method of NRT service transmitted. Accordingly, the service manager 350 of the receiver 300 determines a proper processing method according to each NRT service on the basis of the consumption_model field, and provides NRT service according

to the determined method.

[00783] According to an embodiment of the present invention, the transmitter 200 presets an NRT service providing method applicable to an TDO and allocates a specific value to the consumption_model field to correspond to the TDO. Accordingly, the transmitter 200 may allocate a value corresponding to a TDO to the consumption_model field and then transmit it.

[00784] Then, the service manager 350 of the receiver 300 receives the consumption_model field and determines that an TDO is transmitted and then, receives the TDO according to a predetermined service providing method to provide it.

[00785] Therefore, according to an embodiment of the present invention, when the consumption_model field indicates a value corresponding to a TDO, for example, when the consumption_model field is 0x04, the service manager 350 of the receiver 300 determines that NRT service is TDO service, and operates according to a service providing method thereof. A service providing method according to a TDO consumption model of the receiver 300 will be described later.

[00786] Fig. 53 is a view illustrating a meaning according to each value of a consumption_model field in NRT_service_descriptor according to an embodiment of the present invention.

[00787] The consumption_model field is included in NRT_service_descriptor and is a field indicating which method for consumption_model NRT service that NRT_service_descriptor indicates uses. NRT service may be provided according to a consumption model of one of Browse&Download service, portal service, and push. Additionally, NRT service may be provided depending on a TDO consumption model according to an

embodiment of the present invention. Accordingly, the meaning of a value of a consumption_model field and an operation of a receiver suggested by the present invention will be described.

[00788] First, when a value of the consumption_model field is 0x00, it indicates that NRT service is not transmitted (forbidden).

[00789] Also, when a value of the consumption_model field is 0x01, it indicates that a corresponding NRT service is provided through a Browse&Download method. In this case, the service manager 350 of the receiver 300 browses a corresponding NRT service, and when content is selected, downloads the selected content.

[00790] When a value of the consumption_model field is 0x02, it indicates that a corresponding NRT service is provided through a Portal method. In this case, a corresponding NRT service may be provided through a method of accessing a web browser. Accordingly, files transmitted/received through a FLUTE session linked with a corresponding NRT service may include files for text or graphic rendering.

[00791] When a value of the consumption_model field is 0x03, it indicates that a corresponding NRT service is provided through a Push method. In this case, a corresponding NRT service may be provided based on a request of a user or the receiver 300. Additionally, the service manager 350 of the receiver 300 may allow a user to select whether to perform automatic update of content linked with a corresponding NRT service. If a user selects automatic update, the service manager 350 of the receiver 300 stores content linked with corresponding service to a cache, and automatically updates files to a new version. Also, when a user returns to the

requested Push service, the service manager 350 of the receiver 300 may display preloaded content.

[00792] In addition, when a value of the consumption_model field is 0x04, it indicates that a corresponding NRT service is provided according to a TDO consumption model. In this case, the service manager 350 of the receiver 300 determines that NRT service such as a TDO is transmitted based on the consumption_model field and operates the NRT service in linkage with real-time broadcast.

[00793] In more detail, when the consumption_model field is a TDO consumption model, the service manager 350 of the receiver 300 receives a corresponding NRT service (TDO), and by using at least one of the link descriptor or event descriptor, obtains NRT service related real-time broadcast channel or program information. Additionally, the service manager 350 of the receiver 300 may receive a trigger included in real-time broadcast to operate a TDO at a specific time or may download a content item in advance.

[00794] Moreover, in the case of a TDO consumption model, when each content item is displayed in a currently selected NRT service, it may be downloaded immediately. When an update version of each content item is displayed in a currently selected NRT service, it may be updated immediately. Also, each content item may be executed or terminated by a trigger.

[00795] Fig. 54 is a flowchart illustrating an operation of the receiver 300 when a TDO is transmitted by a TDO consumption model according to an embodiment of the present invention.

[00796] First, the receiver 300 receives an SMT in operation S8000, and parses a service level descriptor in operation

S8010. The receiver 300 may receive an SMT through a service signaling channel by using the service manager 350, and may parse service level descriptors including an NRT descriptor in a service level descriptor loop in each NRT service loop.

[00797] Then, the receiver 300 confirms NRT service descriptor in the parsed each service level descriptor, and determines whether a value of the consumption_model field in NRT_service_descriptor is 0x04, i.e., a TDO consumption model, in operation S8030. If the value is not 0x04, the receiver 300 operates according to another NRT service consumption model.

[00798] However, if the value is 0x04, the receiver 300 determines that an NRT service designated as a TDO consumption model is transmitted. In this case, the receiver 300 determines whether a TDO automatic reception is set in operation S8040. If the automatic reception is not set, the receiver 300 displays TDO service and a list of contents in the TDO service, and displays a TDO reception selection screen in operation S8045. Then, the receiver 300 receives a TDO service selected by a user in operation S8047.

[00799] Moreover, when the automatic reception is set, the receiver 300 receives a TDO without a user's selection in operation S8050. Here, the receiver 300 may receive a TDO at a specific time regardless of a user's selection. For example, the receiver 300 may automatically receive and store a TDO in the background without a user's selection while not displaying to a user that the TDO is being received. Whether to execute the TDO is determined by a user's input, or may be received by the receiver 300 before the trigger time of the TDO. Additionally, the receiver 300 receives the TDO without

a user's selection, so that it may smoothly operate in linkage with real-time broadcast during the execution of the TDO.

[00800] Then, the receiver 300 determines whether a trigger is received after receiving the TDO in operation S8060. The receiver 300 determines whether a trigger corresponding to the TDO is received through the trigger receiver 331. Also, when the trigger is received, the receiver 300 extracts a trigger time and a trigger action from the trigger through the service manager 350, and applies the trigger action to the TDO in ready at the trigger time in operation S8070. For example, the trigger target TDO may shift from one state of a non-ready state, a released state, a ready state, an active state, and a suspended state into another state. Accordingly, the receiver 300 may perform a shift operation according to a TDO state on the basis of the trigger action at the trigger time. As mentioned above, since the TDO is received automatically without a user's selection, and the trigger action is performed at the trigger time, the receiver 300 receives the TDO before the trigger time and prepares the TDO to be ready in the background before the trigger time.

[00801] Additionally, the receiver 300 receives and manages an NRT service that a user selects through the service manager 350. However, in the case of TDO service, since it operates in linkage with real-time broadcast and is triggered by a trigger, storing and managing the TDO service may be accomplished according to the intentions of real-time broadcast service providers.

[00802] Therefore, according to an embodiment of the present invention, in the case of a TDO consumption model type in

which TDO service is transmitted, the receiver 300 allocates a predetermined area of a storage unit as a TDO storage area, and then, receives TDO service, stores the received TDO service in the allocated area, and manages it. Furthermore, the TDO storage area is designated by the transmitter 200, and then, the TDO service is transmitted to the receiver 300. In this case, the receiver 300 may receive, store, and manage the TDO service in the allocated area.

[00803] Fig. 55 is a flowchart illustrating a method of allocating and managing by the receiver 300 a TDO storage area according to a TDO consumption model.

[00804] Referring to Fig. 55, first, the receiver 300 determines a consumption model of NRT service transmitted after operations S8000 to S8020 of Fig. 54 are performed. Then, the receiver 300 determines whether an NRT service of a TDO consumption model type, for example, TDO service, is transmitted based on a consumption model field in operation S8110.

[00805] When the consumption model field indicates a TDO consumption model, the receiver 300 extracts a storage_reservation field of NRT_service_descriptor in operation S8310. The storage_reservation field may be a value corresponding to a storage space essential for a corresponding TDO service. The transmitter 200 may designate a space value for TDO storage area in the storage_reservation field and transmits it.

[00806] Then, the receiver 300 allocates a storage area for TDO service reception and operation on the basis on the storage_reservation field value in operation S8140. The receiver 300 allocates an area designated in the

storage_reservation field or a predetermined TDO service reception area as a storage area for TDO service, and separates the storage area from another area of a storage unit and manages it. Additionally, the receiver 300 maintains the storage space size of the allocated area continuously and uniformly. Accordingly, a broadcast service provider provides extended NRT service such as a TDO continuously.

[00807] Then, the receiver 300 receives TDO service in an area allocated as a TDO service storage area in operation 8150 and stores and manages the received TDO service.

[00808] In such a way, the receiver 300 allocates a storage area for TDO service and manages it on the basis of the consumption model field and storage reservation field of NRT service. However, the above method is just one embodiment and is not limited to a storage method according to each value of a consumption model field and a storage reservation field. Accordingly, the receiver 300 determines whether to store TDO service on the basis of a consumption model field value and a storage reservation field value, and determine whether to receive TDO service on the basis of a current storage reservation field value and the size of a remaining space of a storage unit. Furthermore, when receiving a plurality of TDOs, the receiver 300 designates a predetermined area of a storage unit according to each TDO and manages it, or allocates a comprehensive area for TDO service in a storage unit to receive a plurality of TDOs.

[00809] Moreover, a TDO may mean an NRT service providing a TDO, or each of content items configuring TDO service. Moreover, each content item is signaled through an NRT-IT, and the receiver 300 receives data signaled through the NRT-

IT and recognizes information on a TDO. However, NRT-IT does not include information for TDO, for example, a trigger time, an execution termination time, an execution priority, and an additional data receiving path, besides information provided for general NRT service. Accordingly, it is difficult for the receiver 300 to understand the attribute of a TDO in detail and operates smoothly. Especially, when the receiver 300 manages a storage space, TDO services of a plurality of TDOs provided, or additional data of TDO service, information on TDO is insufficient.

[00810] Therefore, according to an embodiment of the present invention, a method of including specific property information of a TDO in TDO metadata and an operation of the receiver 300 according thereto are suggested.

[00811] Fig. 56 is a view illustrating a TDO metadata descriptor according to an embodiment of the present invention.

[00812] Referring to Fig. 56, TDO metadata includes a descriptor_tag field, a descriptor_length field, a scheduled_activation_start_time field, a scheduled_activation_end_time field, a priority field, an activation_repeat_flag field, and a repeat_interval field.

[00813] The descriptor_tag field may have a value to distinguish this descriptor as a TDO metadata descriptor.

[00814] The descriptor_length field may be an 8-bit unsigned integer to define the length from a field immediately following this field to the end of a TDO metadata descriptor.

[00815] The scheduled_activation_start_time field may be a 32-bit unsigned integer indicating the fastest scheduled activation start time obtained by calculating GPS sec from

the time 00:00:00 UTC on 6 January 1980. When the scheduled_activation_start_time field is 0, it indicates that an activation start time of TDO is already passed.

[00816] The scheduled_activation_end_time field may be a 32-bit unsigned integer indicating the fastest scheduled activation end time obtained by calculating GPS sec from the time 00:00:00 UTC on 6 January 1980. When the scheduled_activation_end_time field is 0, it indicates that an activation end time of TDO is not designated. Accordingly, when the scheduled_activation_end_time field is 0, the receiver 300 continuously provides service after the scheduled activation start time.

[00817] The scheduled_activation_start_time field and the scheduled_activation_start_time field may be used to allow the receiver 300 to synchronize a TDO with real-time broadcast and operate it. Additionally, the receiver 300 may selectively download a TDO on the basis of the scheduled_activation_start_time field and the scheduled_activation_start_time field and may determine the download order of TDOs. Additionally, the receiver 300 may determine the deletion order of TDOs on the basis of the scheduled_activation_start_time field and the scheduled_activation_start_time field. Especially, the receiver 300 identifies a TDO having a scheduled_activation_start_time field value closest to the current time (i.e., a TDO to be activated first), and downloads and stores the identified TDO first. Additionally, the receiver 300 may delete a TDO having a scheduled_activation_start_time field value that is the latest from the current time if a storage space is

insufficient.

[00818] Moreover, a priority field may be an 8-bit unsigned integer indicating the priority of a TDO service or a content item. The receiver 300 determines that as a value of the priority field is higher or lower, a higher priority is provided. Moreover, the receiver 300 may sequentially receive or store a plurality of TDOs on the basis of a priority field allocated to each TDO. Moreover, the receiver 300 may execute at least one of a plurality of TDOs on the basis of a priority field allocated to each TDO.

[00819] Also, since a target TDO executed by a trigger at a trigger time may be one but a TDO executable at one time may be in plurality, a priority designation for each TDO is necessary. Accordingly, when a TDO (service or a content item) is in plurality, the transmitter 200 designates each priority by using the above-mentioned priority field.

[00820] Moreover, the receiver 300 may sequentially download TDOs on the basis of the priority field. Moreover, the receiver 300 may display a plurality of TDOs executable at a specific time on the basis of a priority field. For example, although displaying a plurality of TDOs executable at a specific time, the receiver 300 positions and displays a TDO having a high priority on a top list. Accordingly, a user recognizes priority information through a displayed list, and efficient selection of a TDO according to priority information becomes possible.

[00821] Moreover, the receiver 300 may automatically execute a TDO on the basis of such a priority field. For example, if there are a plurality of TDOs to be executed within a specific time range, the receiver 300 selects a TDO having a

high or low priority field and automatically executes it on the basis of a priority field corresponding to each TDO.

[00822] An activation_repeat_flag field may be a 1-bit Boolean flag indicating whether a repeat_interval field is included in this descriptor.

[00823] The repeat_interval field may be an 8-bit unsigned integer to indicate the repeat interval of an activation schedule time. The repeat_interval field may indicate the execution period of a TDO. Accordingly, the receiver 300 may receive a TDO or manages a storage unit on the basis of the repeat_interval field. The repeat_interval field may mean the execution period of a TDO according to each value as shown in Table 4.

[00824] [Table 4]

VALUE	MEANING
0	Reserved
1	Every day
2	Every week 4-6 times
3	Every week 2-3 times
4	Every week
5	Biweekly
6	Every month
7	Once in two months
8	Once a quarter
9	Once in six months
10	Every year
11-255	Reserved

[00825]

[00826] In such a way, the transmitter 200 may transmit a TDO metadata descriptor through an NRT service signaling channel or an internet network. Additionally, the transmitter 200 inserts a TDO metadata descriptor into a content_level_descriptor loop of NRT-IT to transmit

additional information on a corresponding TDO.

[00827] Moreover, the receiver 300 may receive TDO metadata through an NRT service signaling channel or an internet network to obtain information on a TDO. The receiver 300 receives NRT-IT to search for a content level descriptor loop and obtains TDO metadata. Furthermore, when TDO metadata is transmitted through an internet network, the receiver 300 accesses a FLUTE session for receiving a corresponding TDO service and receives FDT, and also receives TDO metadata from an internet network through URL information of a content location field for each file.

[00828] Fig. 57 is a flowchart illustrating an operation of receiving by the receiver 300 TDO metadata according to an embodiment of the present invention.

[00829] Referring to Fig. 57, the receiver 300 receives an SMT and an NRT-IT transmitted through a service signaling channel in operation S8200.

[00830] According to an embodiment, the transmitter 200 includes a TDO metadata descriptor in a content level descriptor loop and then, transmits it. Accordingly, the receiver 300 searches for the content level descriptor loop of the received NRT-IT in operation S8220 and then, determines whether there is a TDO metadata descriptor in operation S8220. The receiver 300 determines whether the descriptor tag field value of each descriptor corresponds to TDO_metadata_descriptor, and thus, determines whether there is a TDO metadata descriptor.

[00831] Then, if there is a TDO metadata descriptor, the receiver 300 extracts TDO metadata from a TDO metadata descriptor in operation S8230. As mentioned above, the TDO

metadata may include at least one of scheduled activation start or end time information, priority information, related data reception information, and repeat interval information. Additionally, the related data reception information may include additional data on a corresponding TDO or reception path information of additional data.

[00832] Then, the receiver 300 obtains necessary information from the extracted TDO metadata and manages a TDO on the basis of the obtained information in operation S8240. The receiver 300 manages a TDO by receiving, storing, or deleting the TDO according to a predetermined condition on the basis of the TDO metadata. Moreover, the receiver 300 may receive related data that a service provider provides about a corresponding TDO on the basis of TDO metadata. For example, when a TDO is in execution, the receiver 300 extracts related data reception information from TDO metadata, receives additional action information on the TDO in execution through a broadcast channel or an internet network on the basis of the related data reception, and applies the received additional action to the TDO in execution.

[00833] Fig. 58 is a flowchart illustrating a method of the receiver 300 to manage a TDO depending on time information in TDO metadata according to an embodiment of the present invention.

[00834] Referring to Fig. 58, the receiver 300 determines whether a storage space for TDO is insufficient in a storage unit in operation S8300. As mentioned above, the storage space may be a predetermined area of a storage unit for storing TDOs. Additionally, the receiver 300 determines a storage space necessary for TDO reception on the basis of an

NRT service descriptor before receiving a TDO, and determines whether a storage space is insufficient in comparison to the remaining space of a current storage unit.

[00835] Then, if it is determined that a storage space is insufficient, the receiver 300 extracts time information from the TDO metadata on each TDO in operation S8310. Time information may include a scheduled activation start time or end time field of a TDO metadata descriptor. Moreover, the receiver 300 obtains TDO metadata on each TDO from a content level descriptor loop of NRT-IT as mentioned above, and extracts time information from the obtained TDO metadata.

[00836] Then, the receiver 300 first deletes a TDO having the largest value of a scheduled activation start time field on the basis of a scheduled activation start time for each TDO in operation S8320. As a scheduled activation start time is greater, the current needs may be reduced. Therefore, the receiver 300 first deletes a TDO to be executed at the time that is the farthest from the current time, so that storage space may be obtained.

[00837] Then, the receiver 300 first receives a TDO having the smallest value of a scheduled activation start time field on the basis of a scheduled activation start time for each TDO in operation S8330. As the scheduled activation start time value of a TDO is smaller, since the TDO is executed soon, if a storage space is insufficient, the receiver 300 first receives a TDO whose active start time reaches in the fastest time, so that reception is completed within a trigger time.

[00838] In addition, time information in TDO metadata may include a time slot descriptor for providing activation time

information of a TDO in addition to scheduled activation start and end time fields. The time slot descriptor may include a slot start time, a slot length, and repeat information, by using a time at which a TDO is performed as one slot. Accordingly, the receiver 300 extracts a time slot at which a TDO is activated, so as to predict a repeated execution and end time of the TDO.

[00839] Fig. 59 is a flowchart illustrating a method of the receiver 300 to manage a TDO depending on time information and priority information in TDO metadata according to another embodiment of the present invention.

[00840] Referring to Fig. 59, first, the receiver 300 receives TDO metadata, and determines whether there are a plurality of TDOs executable at a specific time in operation S8400. The receiver 300 extracts TDO metadata in a content descriptor loop of NRT-IT through a service signaling channel as mentioned above, and then receives TDO metadata corresponding to each TDO. Then, the receiver 300 may determine whether there are a plurality of TDOs executable at the same time on the basis of the scheduled activation start time field of the extracted each TDO metadata. When there are a plurality of TDOs having the same scheduled activation start time field value or a plurality of TDOs whose scheduled activation start time field value is within a predetermined time range, the receiver 300 determines that there are a plurality of TDOs executable at the same time.

[00841] Also, when it is determined that there are a plurality of TDOs, the receiver 300 extracts a priority value for each TDO from the TDO metadata in operation S8410. The priority value is extracted from a priority field in the TDO

metadata descriptor. Additionally, the receiver 300 matches a priority in the TDO metadata to each of a plurality of TDOs executable at the same time and then, stores them.

[00842] Then, the receiver 300 provides a list of a plurality of TDO executable at a specific time to a user on the basis of the extracted priority. The receiver 300 displays a list of a plurality of TDOs executable at a specific time or after a predetermined time in addition to a priority to a user, so that this leads to a user's selection. Additionally, the receiver 300 places a TDO having a high priority on the upper side of a TDO list and displays it so that priority information for each TDO may be provided together. Accordingly, a user sets which TDO is selected and performed at a specific time with reference to a TDO list, and selects which TDO is received and stored.

[00843] Then, when a TDO is selected by a user, the receiver 300 receives the selected TDO, and performs a trigger action on the selected TDO at a specific time, for example, a trigger time. The receiver 300 may omit an operation of receiving the selected TDO if the selected TDO is received and stored already. Moreover, the receiver 300 deletes the selected TDO from a storage unit, so that a storage space is obtained.

[00844] According to an embodiment, the receiver 300 receives and stores a TDO on the basis of priority information according to setting without a user's selection, and then performs a trigger action. Moreover, when there are a plurality of TDOs executable at the same time, the receiver 300 receives a TDO having a high priority first and stores it, and perform a trigger action on a TDO having a high priority

at a corresponding time.

[00845] Furthermore, according to an embodiment, the transmitter 200 may transmit files configuring a content item of NRT service through an internet network. In more detail, the transmitter 200 generates an FDT by including URL information of each file configuring a content item in a content location attribute, and transmits the generated FDT through a FLUTE session. Furthermore, the transmitter 200 may designate the URL of content item files transmitted through a broadcast network and an IP network by using an internet location descriptor. This internet location descriptor may be included in a content level descriptor loop of an NRT-IT. Accordingly, the transmitter 200 includes URL information on internet where each file placed in an internet location descriptor and transmits it.

[00846] One example of such an internet location descriptor is described with reference to Fig. 60.

[00847] Fig. 60 is a view illustrating a syntax of an internet location descriptor according to an embodiment of the present invention.

[00848] As shown in Fig. 60, the internet location descriptor includes a descriptor_tag field, a descriptor_length field, a URL_count field, a URL_length field, and a URL() field.

[00849] The descriptor_tag field may be an 8-bit unsigned integer to distinguish this descriptor as an internet location descriptor. For example, this field may have a 0xC9 value.

[00850] The descriptor_length field may be an 8-bit unsigned integer to define the length from a field immediately following this field to the end of an internet location

descriptor.

[00851] The URL_count field may be a 5-bit unsigned integer to indicate the number of pairs of URL length fields and URL fields in an internet location descriptor. That is, the internet location descriptor includes a plurality of URL length fields whose number corresponds to a URL count field and a plurality of URL fields whose number corresponds to a URL count field.

[00852] The URL_length field may be an 8-bit unsigned integer to indicate the length of the URL() field immediately following this field.

[00853] The URL() field is a character string indicating URL. When the URL() field indicates Relative URL or absolute tag URI, a corresponding URL may be seen as a content transmitted only through FLUTE of NRT. In other cases, a corresponding URL may be seen as a content only transmitted through a broadcast network, a content transmitted through an IP network, or a content transmitted through both a broadcast network and an IP network.

[00854] When the URLs of files configuring a content item are transmitted by using an internet location descriptor, a URL corresponding to each file is designated and needs to be included in an internet location descriptor. However, when the number of files is increased, the number of URLs to be designated is increased, so that transmission efficiency may be reduced. Moreover, the number of URLs is increased, their management becomes complex and difficult.

[00855] Accordingly, the transmitter 200 according to an embodiment of the present invention includes list information of URLs corresponding to each file in a FLUTE file

description table (FDT), and transmits the FDT through an internet network instead of a FLUTE session. The transmitter 200 may transmit list information of URLs or an FDT by using the internet location descriptor, and also may transmit listinformation of URLs or an FDT by using the link descriptor.

[00856] Also, related information between each file and a content item may be designated by a content linkage field in an FDT. Additionally, when the URL of the FDT is transmitted through an internet location descriptor, since a list of files configuring a content item is included in the FDT, the receiver 300 may determine a relationship between a content item and each file without a content linkage field. For example, the receiver 300 parses a content level descriptor loop of a content item to be received on the basis of NRT-IT, extracts the URL of the FDT from an internet location descriptor in the content level descriptor loop, and receives the FDT through an internet network, so that it obtains a list of files configuring the content item to be received.

[00857] The receiver 300 receives URL information of the FDT located through a link descriptor or an internet location descriptor, and receives the FDT on the basis of the received URL information through an internet network. Then, the receiver 300 receives files configuring a content item on the basis of URL information of each file in the FDT. In such a way, by transmitting the URL of FDT, it is not necessary to designate a URL to each of a plurality of files. As a result, transmission efficiency may be increased.

[00858] Fig. 61 is a flowchart illustrating an operation of the receiver 300 when FDT is transmitted through an internet

network according to an embodiment of the present invention.

[00859] Referring to Fig. 61, the receiver 300 first receives an SMT and an NRT-IT through the service manager 350 in operation S8500.

[00860] Then, the service manager 350 of the receiver 300 selects NRT service or a content item to be received in operation S8510. The service manager 350 of the receiver 300 selects NRT service to be received according to a predetermined condition. Additionally, the service manager 350 of the receiver 300 selects NRT service to be received according to a user's input.

[00861] Then, the service manager 350 of the receiver 300 determines whether to receive each file configuring a content item of the selected NRT service through internet in operation S8520. The service manager 350 of the receiver 300 determines whether to receive each file through internet on the basis of a connection state of the internet network. Additionally, the service manager 350 of the receiver 300 determines whether to receive each file through internet according to a user's setting.

[00862] Then, when it is determined that each file is not received through internet, the service manager 350 of the receiver 300 accesses a FLUTE session through an NRT service signaling channel, and receives each file configuring a content item of the selected NRT service in operation S8525. An operation of receiving NRT service through a FLUTE session is described above.

[00863] However, when it is determined that each file is received through internet, the service manager 350 of the receiver 300 receives FDT through internet in operation S8530.

The service manager of the receiver 300 receives URL information of FDT through a link descriptor or an internet location descriptor, and receives the FDT located on an internet network on the basis of the received URL information of the FDT.

[00864] Here, the FDT may include a list index of each file configuring a content item in a FLUTE session. A MIME type designation is necessary in order to additionally transmit the FDT through an internet network. The MIME type means a specific format to indicate the type of a content transmitted through an internet network. Various file formats are registered as MIME types and are used in internet protocols such as HTTP and SIP. The MIME type registration is managed by IANA. MIME may define a message format of a tree structure. According to an embodiment of the present invention, an MIME type corresponding to FDT may be defined as a format such as application/nrt-flute-fdt+xml. Additionally, the receiver 300 parses a URL having the MIME type and, based on this, receives FDT.

[00865] When receiving an FDT, the service manager 350 of the receiver 300 obtains URL information of files configuring each content item from the FDT in operation S8540. Then, the service manager 350 of the receiver 300 receives each file on the basis of the obtained URL information through internet in operation S8550. The service manager 350 of the receiver 300 connects the received files on the basis of a content linkage field of the FDT, so that it obtains a content item.

[00866] Then, the service manager 350 of the receiver 300 provides NRT service through the received content item in operation S8560.

[00867] As mentioned above, since the receiver 300 receives the FDT through internet, it is not necessary to receive URL information of each file and it is possible to efficiently receive NRT service through an internet network. Moreover, according to an embodiment of the present invention, an arbitrary file including a file list (index) may be transmitted instead of FDT.

[00868] Fig. 62 is a flowchart illustrating an operation of the receiver 300 when the URL of an FDT is transmitted through a link descriptor according to an embodiment of the present invention.

[00869] According to an embodiment of the present invention, the transmitter 200 includes the link descriptor in one of EIT, VCT, or PMT and transmits it, and the link descriptor includes URL information for obtaining FDT.

[00870] First, when the receiver 300 is turned on, a broadcast channel or a broadcast program is selected in operation S8600.

[00871] Then, the receiver 300 receives at least one of an EIT, a VCT, or a PMT through the service manager 350 or a PSIP/PSI handler in operation S8610, extracts a link descriptor in operation S8620, and obtains the URL information of an FDT from the link descriptor in operation S8630.

[00872] Here, the link descriptor may be included in the EIT and transmitted as shown in Fig. 46. In this case, the link descriptor includes information linked with a specific broadcast program. Additionally, a link descriptor may include URL information of an FDT in order to receive NRT service linked with the selected broadcast program through

internet. Accordingly, the receiver extracts a link descriptor from an event descriptor loop of an EIT corresponding to the selected broadcast program, and obtains URL information of an FDT included in the link descriptor. Especially, when the MIME type field in the link descriptor indicates the MIME type of an FDT, the receiver 300 determines that information indicated by a link byte field is the URL of a FDT file.

[00873] Additionally, the link descriptor may be included in a VCT or a PMT and transmitted. In more detail, the link descriptor is included in the descriptor loop of a VCT or the program descriptor loop of a PMT and includes information on a content linked with a specific channel or service. The receiver 300 extracts the link descriptor in a VCT or a PMT, and obtains URL information of a FDT file in the link descriptor through a method that is identical to that of the case that a link descriptor is included in the EIT.

[00874] Then, the service manager 350 of the receiver 300 receives an FDT file by using the obtained URL information of the FDT through internet in operation S8640.

[00875] Then, the service manager 350 of the receiver 300 obtains URL information of each file included in the FDT in operation S8650. The URL information of each file included in the FDT may include position information on internet of files configuring a content item of NRT service.

[00876] Then, the service manager 350 of the receiver 300 receives designated files according to the obtained URL information through internet in operation S8660 and stores the received files as a content item in operation S8670. The service manager 350 of the receiver 300 may store each file

as a content item of NRT service on the basis of the content linkage field of the FDT. Additionally, the service manager 350 of the receiver 300 provides to a user NRT service on the basis of the stored content item.

[00877] Through such a process, the receiver 300 may receive FDT including a file list configuring a content item of NRT service through an internet network. Moreover, since each file is received based on an FDT through internet, efficient transmission is possible.

[00878] Fig. 63 is a flowchart illustrating an operation of the receiver 300 when the URL of FDT is transmitted through NRT-IT according to an embodiment of the present invention.

[00879] According to an embodiment of the present invention, the transmitter 200 includes the internet location descriptor in a content descriptor loop of an NRT-IT and transmits it, and the internet location descriptor includes URL information for obtaining FDT.

[00880] First, the receiver 300 receives the NRT-IT through the service manager 350 in operation S8700.

[00881] Then, the receiver 300 extracts an internet location descriptor from a content descriptor loop content_descriptor() of the NRT-IT through the service manager 350 in operation S8610, obtains URL information of an FDT from the internet location descriptor in operation S8630, receives the FDT by using the URL information of the FDT through an internet network in operation S8740, and obtains URL information of a file configuring each content item from the FDT in operation S8750.

[00882] Here, the receiver 300 receives URL information of FDT through an internet location descriptor, and also

receives FDT in access to a FLUTE session through an SMT and an NRT-IT. Additionally, the URL of each file in an FDT may indicate the address of an internet network, and may also indicate a file position in a FLUTE session. Accordingly, the receiver 300 determines the URL format of each file from the FDT, and selectively receives each file through one of a FLUTE session and an internet network according to the determined format.

[00883] Additionally, the receiver 300 receives an FDT through an internet network, and receives files configuring a content item in access to a FLUTE session on the basis of the received FDT.

[00884] Moreover, the transmitter 200 designates a transmission preferred path when each file is transmittable through both a FLUTE session and an internet network. The transmission priority path may be designated as an attribute of File type: complex type by using xml schema. For example, it may be designated as <xs:attribute name="Internet_preferred" type="s:boolean" use="optional" default="false"/> . Here, internet_preferred means preferred transmission through an internet network, and when a type value is true, it means that preferred transmission through an internet network is performed. Additionally default is defined as false and this indicates that preferred transmission through a FLUTE session is performed by default. Accordingly, the receiver 300 confirms the attribute of each file type to selectively receive each file through one of a FLUTE session and an internet network.

[00885] Then, the service manager 350 of the receiver 300 receives designated files according to the obtained URL

information of the files through a FLUTE session or an internet network in operation S8760 and stores the received files as a content item in operation S8670.

[00886] The service manager 350 of the receiver 300 may store each file as a content item of NRT service on the basis of the content linkage field of the FDT as mentioned above. Additionally, the service manager 350 of the receiver 300 provides to a user NRT service on the basis of the stored content item.

[00887] Furthermore, NRT service may include the above-mentioned TDO and an NRT service declarative object (NDO). The receiver 300 may receive a TDO or an NDO to provide NRT service. Moreover, NRT service may include at least one content item, and the receiver 300 may receive at least one content item to provide NRT service. In providing NRT service, the receiver 300 extracts the above-mentioned consumption model field from an NRT service descriptor, and determines a consumption model of NRT service, so that service is provided through different methods according to a consumption model.

[00888] A service provider determines a consumption model corresponding to NRT service and then, allocates a consumption model field value corresponding thereto to NRT service and transmits it. A consumption model field may be included in an NRT service descriptor as shown in Fig. 52, and the receiver 300 may operate according to the extracted consumption model field. Although such a consumption model field value and an operation of the receiver 300 are shown in Fig. 53, a more detailed operation will be described below.

[00889] A consumption model of NRT service may be Browse and Download. In this case, NRT service may include a selectable

content for later download. Additionally, a user interface of NRT service provided from the receiver 300 may indicate selectable content. Also, a user may move between already-downloaded content items through a user interface of NRT service and may select a view command.

[00890] In the case of a browser and download consumption model, NRT service may include a plurality of content items. The receiver 300 may download at least one content item that a user selects to download. Here, a content item is downloaded in the background. Additionally, the receiver 300 monitors a content item downloaded for update in the background. The receiver 300 downloads a updated content item if update is confirmed based on a result of monitoring. Additionally, the receiver 300 may not update downloaded content items according to a setting.

[00891] Additionally, when at least one content that a user select is downloaded completely, the receiver 300 may launch or present it according to a user's selection. Then, the launched or presented at least one content item may be suspended or paused according to a user's instruction. Additionally, the receiver 300 may resume to launch or present the suspended or paused at least one content according to a user's instruction. In addition, when the launched or presented at least one content item is completed or a user instructs it to be terminated or a user selects to present another content item of browser and download service, the receiver 300 may terminate the launched or presented at least one content item. The receiver 300 may delete one content item when a user instructs to delete it.

[00892] Accordingly, a user may launch or present a content

item of a browser and download consumption model, or may suspend or pause the launching or presenting, or after a user exit the NRT service and comes back again, may resume the launching or presenting again.

[00893] Moreover, a consumption model of NRT service may be Push. In this case, NRT service may include a request based content. For example, the receiver 300 may launch or present a content item included in NRT service according to a request of a user subscribed to an NRT service of a push consumption model. The receiver 300 receives from a user an input on whether an NRT service of a received push model or a content relating thereto is automatically updated.

[00894] According to an embodiment of the present invention, the receiver 300 displays topics that a user or subscriber can select in correspondence to content items included in an NRT service of a push consumption model, receives a content item corresponding to the selected topic, and automatically performs updating. Accordingly, a user is allowed to select a content item to be updated automatically. The receiver 300 may provide an updated content item by using a predetermined area of NRT service displayed, and may continuously perform updating. If a user selects a content item for automatic update, the receiver 300 stores a received push model NRT service or a content relating thereto in a cache, and automatically updates files configuring the content to a new version. The receiver 300 may pre-load content and then display it when returning to a push model NRT service that a user requests. Additionally, in the case of a push consumption model, download, launch, suspend, resume, and shutdown are performed identical to those of the browser and

download service.

[00895] In the case of a push consumption model, NRT service may include one content item. The receiver 300 may download one content item in a push consumption model in the background. Then, the receiver 300 confirms update of the downloaded content item in the background. Additionally, the receiver 300 may launch the downloaded content item according to a user's NRT service select command. The receiver 300 may terminate the launched content item when a user makes an input for exiting NRT service. The receiver 300 may perform the automatic update when a user subscribes to a content item receiving service of a push consumption model. The receiver 300 determines whether to provide an automatic update option on the basis of an auto-update field in an NRT service descriptor in an SMT. Accordingly, the receiver 300 provides automatic update according to a user's selection on an automatic update option provided based on an NRT service descriptor and performs update in the background. When the automatic update option is off, for example, when a user closes the content item receiving service of the push consumption model, the receiver 300 may delete the downloaded content item.

[00896] Moreover, a consumption model of NRT service may be Portal. In this case, NRT service may include a service similar to accessing a web browser. Accordingly, a FLUTE session linked with an NRT service of a portal consumption model may include files for text or graphical components to configure a web page. Additionally, files providing portal NRT service are updated within near-real-time and then transmitted. Accordingly, the receiver 300 may generate a

screen of a portal consumption model NRT service to be displayed while a viewer waits.

[00897] In the case of a portal consumption model, NRT service may include one content item. When a user selects a portable consumption model NRT service, the receiver 300 may download one content item therein. Additionally, the receiver 300 may launch a content item when an entry file is downloaded among files of the content item that is being downloaded. The receiver 300 may terminate and delete the launched content item when a user makes an input for exiting NRT service. Additionally, in the case of a portal consumption model, download, launch, suspend, resume, and shutdown are performed identical to those of the browser and download service.

[00898] In the case of the above consumption models, the receiver 300 provides NRT service on the basis of one content item. However, with one content item, it is difficult to generate a user interface linked with real-time broadcast or provide a complex and diverse NRT service.

[00899] Accordingly, according to an embodiment of the present invention, an NRT service provider may provide a plurality of content items through the transmitter 300. Additionally, an NRT service provider may set one content item to control an operation of another content item. The receiver 300 receives this, and identifies a consumption model providing a plurality of content items to provide complex and various NRT services to a user. By using such a consumption model, an NRT service provider may generate a user interface that is interactive with NRT service (NDO or TDO) and transmit it. The receiver 300 generates a look and

feel user interface interactive with a user on the basis of a plurality of content items in NRT service and provides it to a user.

[00900] Such a consumption model of NRT service may be the above-mentioned TDO consumption model. In this case, NRT service transmitted may include the above-mentioned TDO.

[00901] In the case of a TDO consumption model, NRT service may include a plurality of content items. The receiver 300 may download each content item founded in the currently selected NRT service. The receiver 300 may update each content item whose updated version is founded in the currently selected NRT service. Each content item is launched or terminated by a trigger.

[00902] Also, according to an embodiment of the present invention, a consumption model of NRT service may be a scripted consumption model. In Such an interactive content item may be referred to as an entry content item that controls content items but is not limited the name of the content item. For example, an interactive content item may be a master content item of other content items.

[00903] The receiver 300 first receives an entry content item or master content item and then launches it. The entry content item may include a function for managing the access and download of NRT service content items. Additionally, the entry content item may include a function for displaying usable contents. The receiver 300 may display a user interface for allowing a user to select a usable content and reserve download, according to the launch of an entry content item. A user interface displayed may include a menu for playing previously-downloaded contents.

[00904] An NRT service object in an NRT service of a scripted consumption model is automatically launched when a user accesses the scripted consumption model NRT service. For example, a scripted consumption model NRT service object may be triggered by a user's access and then, launched.

[00905] The scripted consumption model NRT service may include a plurality of content items. One of the plurality of content items may include an entry content item having a user input available and interactive interface. The transmitter 300 inserts such a user interactive interface into an entry content item and transmits it, and the receiver 300 receives and launches an entry content item first. Then, the receiver 300 may control the download, update, or launch of another content item by launching the entry content item. Accordingly, an entry content item may include at least one file for downloading, updating, or launching other content items.

[00906] Also, an NDO including an entry content item may be launched when a user selects an NRT service including an NDO. Additionally, the launched NDO may be terminated when a NRT service exiting command of a user is inputted. An NRT service object including an entry content item is downloaded in advance and updated. Additionally, after the downloaded NRT service object is launched and terminated, the receiver 300 may maintain its storage. Through this, each time NRT service is selected, a user does not need to wait the receivers to download the NRT service, and is allowed to receive an NRT service updated by an entry content item, so that user's convenience may be enhanced.

[00907] According to an embodiment, in the case of a scripted consumption model, the receiver 300 may operate similar to a

push consumption model. An NRT service designated as a scripted consumption model may include a plurality of content items corresponding to a push consumption model or a portal consumption model. Accordingly, a scripted consumption model may mean a consumption model extending a push consumption model. Accordingly, a scripted consumption model may mean a consumption model extending a portal consumption model. Accordingly, a service provider may generate and transmit an NRT object by designating a push scripted consumption model or a portal scripted consumption model.

[00908] An NRT service object or NRT service of a scripted consumption model may include the above-mentioned plurality of content items and one of the plurality of content items is an entry content item. An entry content item is a content item that is launched first of all to provide by the receiver 300 NRT service according to an NRT service object. Then, files configuring each content item may include an entry file. The entry file is a file that is launched first of all when a content item is launched. Accordingly, a service provider generates an NRT service object through the transmitter 200, designates an entry content item to be launched first, and includes information on an entry content item in an NRT service object or transmits it through service signaling data. Then, a service provider may designate an entry file to be launched first of all among files configuring each content item through transmitter 200, and may transmit information on an entry file through service signaling data or a FLUTE session.

[00909] Moreover, an NRT service of a consumption model, push consumption model, or portal consumption model of the browser

and download consumption model may include a plurality of content items.

[00910] For example, an NRT service of a browser and download consumption model may include a plurality of content items. The plurality of content items may include a set of different information, and each may include an NRT service object with a unique look and feel interface to indicate a set of information. Each object may include the above-mentioned entry file by each content item. A user may select which content item is to be downloaded through the receiver 300, and also may select a content item for presentation after a predetermined time. When a content item for presentation is selected, the receiver 300 launches an object including a look and feel interface in the content item, outputs presentation, and manages the outputted presentation.

[00911] Additionally, an NRT service of a push consumption model may include an NRT service object including an entry file. The receiver 300 continuously updates files used in an NRT service object. When a user selects automatic update of service, the receiver 300 may maintain the update of the files. Additionally, the receiver 300 may update the files by an operation of an NRT service object itself. When a user selects NRT service, the receiver 300 may launch an NRT object to manage the presentation of the selected NRT service data.

[00912] Additionally, an NRT service of a portal consumption model may include an NRT service object including an entry file. When a user selects an NRT service of a portal consumption model, the receiver 300 obtains and launches an NRT service object including an entry file. The NRT service

object including an entry file may manage presentation for other files. In this case, an entry file is a simple HTML page, and each time service is provided, an obtaining process is necessary, so that a smooth operation may be guaranteed only when an acquisition time is very short. Accordingly, it is difficult for the receiver 300 to provide complex and various look and feel interactive interfaces by using only an entry file. Therefore, according to an embodiment, when an entry content item is designated and transmitted, the receiver 300 may store the entry content item in a predetermined storage space after an initial acquisition time of the entry content item. Since the receiver 300 does not require a following acquisition time for presentation, the presentation is possible using an interactive interface, and a smooth and natural operation is guaranteed. Accordingly, a service provider generates and transmits a look and feel interactive interface through the transmitter 200, and the receiver 300 provides a list of NRT service to a user through the interactive interface generated by the service provider without its own interface and launches a selected NRT service.

[00913] Fig. 64 is a conceptual view illustrating an NRT service including an entry content item.

[00914] Referring to Fig. 64, the NRT service may include a plurality of content items such as a content item 0, a content item 1, and a content item 3. Also, a service provider may designate each entry content item or entry file through the transmitter 200. As shown in Fig. 64, an entry content item is designated as the content item 0, and the entry file is designated as a file 0 and a file 5.

[00915] When an NRT service of Fig. 64 is received and a user

selects it, the receiver 300 launches the content item 0, i.e., an entry content item first of all. When launching the content item 0, the receiver 300 may execute the file 0, i.e., an entry file, first of all. Here, the entry content item may include a function for managing other content items. Accordingly, the receiver 300 may display the content item 1 and the content item 3 by an operation of the content item 0, and the content item 0 may launch or terminates at least one of the content item 1 and the content item 3 according to a user's selection. Moreover, when the content item 1 is launched, the file 5, i.e., an entry file, may be launched first.

[00916] Figs. 65 and 66 are views illustrating an NRT-IT to transmit information on an entry content item.

[00917] The NRT-IT or NCT shown in Figs. 65 and 66 is included in service signaling data, and the transmitter 200 includes information on an entry content item in service signaling data and transmits it. For example, the transmitter 200 may include a field, which indicates whether a specific content item is an entry content item, in the NRT-IT and then, may transmit it. Accordingly, the receiver 300 receives the service signaling data, extracts the NRT-IT, parses the field that indicates whether a specific content item is an entry content item, determines which content item is an entry content, and launches it first of all. As mentioned above, the NRT-IT may include individual information on a plurality of content items. Information of each of the plurality of content items may be included in a content item loop starting as for (j=0; j< num_content_items_in_section; j++). Then, each content item may correspond to each content item loop

identified by a content linkage field. Accordingly, the receiver 200 parses a content item loop to obtain a no_entry_flag field corresponding to each content item. Since descriptions for other fields in Figs. 65 and 66 are identical or similar to those of Fig. 16, their repeated descriptions are omitted.

[00918] A no_entry_flag field may be a 1-bit number indicating an entry content item. If this field is 0, a content item corresponding to a content item loop that this field belongs may be an entry content item of a service identified by service_id. If this field is 1, a content item corresponding to a content item loop that this field belongs may not be an entry content item of the above-mentioned service.

[00919] The transmitter 200 designates a content item to be launched first as an entry content item by using the no_entry_flag field and transmits it. The receiver 300 determines whether the no_entry_flag field is 0 or 1, and thus obtains information on a content item to be launched first.

[00920] Fig. 67 is a view illustrating an operation method of the receiver 300 when an entry content item is transmitted according to an embodiment of the present invention.

[00921] Referring to Fig. 67, the receiver 300 first receives an SMT and an NRT-IT from service signaling data in operation S8800, parses a service level descriptor loop from the SMT, and parses NRT_service_info_descriptor in the service level descriptor loop in operation S8810. The receiver 300 may receive an SMT through a service signaling channel by using the service manager 350, and may parse service level

descriptors including an NRT service information descriptor in a service level descriptor loop in each NRT service loop.

[00922] Also, the receiver 300 obtains information on an NRT service transmitted based on the parsed NRT service information descriptor, for example, at least one of an application type, a service category, service capacity, and video or audio codec information, to generate service guide and displays it in operation S8820. The receiver 300 generates and displays service guide through the service manager 350.

[00923] For example, the service manager 350 calculates a capacity of NRT service on the basis of a storage_requirement field in the NRT service information descriptor and displays it on the service guide. Moreover, the service manager 350 determines video or audio codec of NRT service on the basis of an audio_codec_type or video_codec_type field in the NRT service information descriptor, and displays information therefor on the service guide.

[00924] Then, the receiver 300 determines whether service reception on NRT service is selected in operation S8830. The service reception selection may be made by a user viewing the service guide. The service manager 350 determines which one of NRT service displayed on the service guide is selected by a user. The receiver 300 may maintain the service guide display when NRT service is not selected.

[00925] Moreover, when service reception on NRT service is selected, the receiver 300 obtains reception information of content items that a service to be received includes from an NRT-IT in operation S8840. The service manager 350 obtains the service identifier of the selected NRT service from an

SMT. Then, the service manager 350 obtains information on content items to receive the selected NRT service by comparing the service identifier with the NRT-IT.

[00926] For example, the service manager 350 extracts service_id corresponding to the NRT service to be received from an SMT. Then, the service manager 350 determines whether the service_id field of the NRT-IT corresponds to an NRT service to be received, and when they are matched, extracts a content_linkage field from a content item loop to obtain reception information of content items. Additionally, according to an embodiment, the service manager 350 obtains the no_entry_flag field from each content item loop, and determines in advance whether a content item corresponding to each content item loop is an entry content item before reception.

[00927] Then, the receiver 300 receives content items of an NRT service transmitted from the transmitter 200 on the basis of the obtained content item reception information in operation S8850. The service manager 350 of the receiver 300 may receive files configuring content items in access to a FLUTE session on the basis of the obtained service_id field and content_linkage field, as mentioned above.

[00928] Then, the receiver 300 determines whether there is a launch command on the received NRT service in operation S8860. The service manager 350 of the receiver 300 determines whether there is a user's launch command when the content item of NRT service is received completely. Additionally, the service manager 350 may automatically launch the received NRT service without a user's launch command.

[00929] When the NRT service is launched, the receiver 300

first determines whether a consumption model is scripted in operation S8870. The service manager 350 of the receiver 300 parses a service level descriptor loop from SMT, extracts NRT_service_descriptor from the service level descriptor loop, and determines whether the NRT service is a scripted consumption model on the basis of a value of the consumption_model field in NRT_service_descriptor. A value of a consumption model field for a scripted consumption model may be pre-designated. When its value is not a scripted consumption model, the receiver 300 operates in a different way according to another consumption model to provide service in operation S8890.

[00930] However, when the consumption model field corresponds to a value designated as a scripted consumption model, the receiver searches for an entry content item among the received content items in operation S8875. The receiver 300 searches for files in the entry content item among content items to execute it first. For this, the receiver 300 may search for an entry content item by using the no_entry_flag field of the NRT-IT. Additionally, the receiver 300 first receives and stores an entry content item in order for fast search and launch. For example, when content items configuring the selected NRT service are not all received, the receiver 300 receives, searches for, and launches an entry content item in advance, so that the NRT service may be partially provided.

[00931] Then, when the entry content item is searched, the receiver 300 launches an entry file first among files in the entry content item in operation S8880. When there is no entry file, the receiver 300 may execute a file according to a file

order or predetermined order.

[00932] Then, the receiver 300 operates by the executed file, and accordingly, provides NRT service to a user in operation S8885. The receiver 300 provides NRT service by sing an interactive interface in an entry content item. Additionally, the receiver 300 may launch or manage another content item according to a user input for an interactive interface in an entry content item.

[00933] In such a way, the receiver 300 launches an entry content item first to provide an interactive interface to a user. Additionally, the receiver 300 may manage content items of the selected NRT service through an interactive interface launched in an entry content item. The interactive interface may include a look and feel interface, and may be configured based on a graphic user interface for user's convenience. Moreover, the interactive interface may include a management menu such as launch, shut down, storage, or deletion of NRT service. For this, the interactive interface may include an icon image or video corresponding to each management menu. A user may conveniently manage another NRT service or other content items through an interface in an entry content item.

[00934] In such a way, the transmitter 200 designates a consumption mode corresponding to NRT service or an NRT service object to a consumption model or one of the other consumption models, allocates a value corresponding to the designated consumption model to a consumption model field in an NRT service descriptor of an NRT service information table, and transmits the NRT service information table through a service signaling channel.

[00935] However, a recognizable consumption model field value

may be limited according to the type or production time of the receiver 300 For example, the receiver 300 may operate by recognizing one of a push or portal consumption model value. In this case, when the transmitter 200 designates NRT service as a scripted consumption model and transmits it, the receiver 300 determines that a consumption model value is not designated, so that normal NRT service may not be provided.

[00936] Therefore, according to an embodiment of the present invention, the transmitter 200 generates a plurality of NRT service objects in correspondence to the same NRT service, and allocates different consumption model field values to the generated NRT service objects. The transmitter 200 may include allocation information of the consumption model field values in the above-mentioned NRT service table and then, may transmit it.

[00937] Fig. 68 is a conceptual view of a plurality of NRT service objects transmitted according to an embodiment of the present invention.

[00938] As shown in Fig. 68, an NRT service transmitted one virtual channel may include an NRT service object A and an NRT service object B. Also, A and B may provide the same NRT service. However, the NRT service object A may be designated as a browser and download consumption model and the NRT service object B may be designated as a scripted consumption model.

[00939] According to an embodiment of the present invention, a content item for providing the same NRT service may be the content item 1 to the content item 5 as shown in Fig. 68. Accordingly, the NRT service object A may include the content item 1 to the content item 5 provided by the browser and down

consumption model. Also, the NRT service object B may include the content item 6, i.e., the entry content item of the scripted consumption model. The content item 6 may include the above-mentioned interactive interface controlling the content item 1 to the content item 5.

[00940] In this way, when transmitting a specific NRT service including the content item 1 to the content item 5 as a browser and download consumption model, the transmitter 200 includes the content item 1 to the content item 5 in the NRT service A and transmits it. When transmitting them as a scripted consumption model, the transmitter 200 includes the content item 6, which is an entry content item to control the content item 1 to the content item 5, in the NRT service B and transmits it. The NRT service B includes only an entry content item, and the content item provides an interface by using content items included in the NRT service B, so that redundant transmission of a content item may be prevented and the waste of a bandwidth may be reduced.

[00941] Here, the entry content item may refer to an NRT-IT or an SMT in order to launch content items included in another NRT service. The NRT-IT or SMT may include relationship information between NRT services and information on content items according thereto. For example, the NRT-IT may include relationship information or reference information that the NRT service A and the NRT service B provide services by using the same content item.

[00942] According to an embodiment, the NRT service A and the NRT service B provide service by using the same content item, but the NRT service A and the NRT service B may include different NRT service objects or content items.

[00943] Fig. 69 is a view illustrating a syntax of an NRT service descriptor included in the SMT according to an embodiment of the present invention.

[00944] The above-mentioned relationship information may be included in an NRT service descriptor in a service descriptor loop of the SMT. According to an embodiment of the present invention, service is provided using the same content item but NRT services having different consumption model field values may be referred to as the same service or equivalent service. The relationship information on the equivalent services, as shown in Fig. 69, may include an equivalent_service_not_present field, a num_equivalent_services field, and an equivalent_service_id field.

[00945] The equivalent_service_not_present field may be a 1-bit flag to indicate whether there is equivalent service among transmitted NRT services. If this field is 0, it means that there is equivalent service.

[00946] The num_equivalent_services field may be an 8-bit unsigned integer to indicate the number if there is equivalent service.

[00947] The equivalent_service_id field may be an 16-bit unsigned integer to indicate a service identifier corresponding to equivalent service. Here, the equivalent_service_id field may include on a service identifier when equivalent service is transmitted through the same broadcast channel. However, when the same service is transmitted through different virtual channels, the equivalent_service_id field may include a service identifier and a broadcast identifier. Moreover, an NRT service

descriptor may further include an MH broadcast identifier field corresponding to a mobile broadcast identifier.

[00948] Since descriptions for other fields of an NRT service descriptor are identical to those of Fig. 52, repeated descriptions are omitted.

[00949] Figs. 70 and 71 are views illustrating a syntax of another NRT-IT according to another embodiment of the present invention. Since descriptions for other fields of the NRT-IT of Figs. 70 and 71 are identical to the above, repeated descriptions are omitted.

[00950] As shown in Figs. 70 and 71, the NRT-IT may further include an available_on_current_service field. When equivalent service is transmitted, this field indicates whether there is a content item transmitted in another NRT service. Additionally, even when there is no equivalent service, this field indicates whether a content item linked with a currently launched NRT service is transmitted through another NRT service in order for the receiver 300 to receive or launch a content item of another NRT service through an entry content item.

[00951] The available_on_current_service field may be a 1-bit number to indicate whether a currently transmitted NRT service (NRT service corresponding to a service identifier field of NRT-IT) includes a content item that this field belongs. If a value of this field is 1, the content item is included in the currently transmitted NRT service and transmitted, and if a value of this field is 0, the content item is included in another NRT service and transmitted.

[00952] The available_on_current_service field may be used for the receiver 300 to determine whether there is a content

item transmitted in another NRT service linked with the currently transmitted NRT service. For example, when NRT service is selected and an entry content item is launched in the receiver 300, an entry content item determines whether there is a content item transmitted in another NRT service on the basis of the available on current service field in order to provide a content item in another NRT service, and receives or launches a content item transmitted in another NRT service. The transmitter 200 may include information on a content item transmitted through another NRT service in NRT-IT and then, may transmit it. For example, the transmitter 200 may include information on a content item transmitted through another NRT service in a specific descriptor of NRT-IT and then, may transmit it.

[00953] Fig. 72 is a view illustrating a syntax of Other_NRT_location_descriptor according to another embodiment of the present invention.

[00954] Referring to Fig. 72, an other NRT location descriptor includes a descriptor_tag field, a descriptor_length field, a num_other_NRT_locations field, and an other_service_id field.

[00955] The descriptor_tag may be an 8-bit unsigned integer to distinguish this descriptor as a link descriptor. For example, this field may have a 0xe8 value.

[00956] The descriptor_length field may be an 8-bit unsigned integer to define the length from a field immediately following this field to the end of a link descriptor.

[00957] The num_other_NRT_locations field may be a 6-bit unsigned integer to indicate the number of other_service_id.

[00958] The other_service_id field may be an 16-bit unsigned

integer to indicate a service identifier of another NRT services in which a content item is transmitted.

[00959] Such an other NRT service location descriptor may be one of content level descriptors in a content level descriptor loop (a loop starting with for (i=0; j< num_content_descriptors; i++)) of the NRT-IT of Figs. 70 and 71. Accordingly, the NRT service location descriptor is included in the NRT-IT and then, transmitted.

[00960] The receiver 300 determines whether a content item is transmitted through another NRT services on the basis of the NRT-IT, and obtains the identifier of another NRT services in which a content item is transmitted on the basis of an NRT service location descriptor.

[00961] Fig. 73 is a flowchart illustrating a method of receiving broadcast service according to an embodiment of the present invention.

[00962] Referring to Fig. 73, the receiver 300 receives an SMT and an NRT-IT from the service signaling data in operation S8900, parses a service level descriptor loop from the SMT and parses NRT_service_info_descriptor in the service level descriptor loop in operation S8910, generates service guide on the basis of the parsed NRT service info descriptor and displays it in operation S8920, and determines whether service reception for NRT service is selected in operation S8930. The above operations may be performed by the service manager 350 of the receiver 300.

[00963] Moreover, when service reception on NRT service is selected, the receiver 300 obtains reception information of content items that a service to be received includes from the NRT-IT in operation S8940. The reception information on

content items may include content_linkage of a content item and service_id of NRT service that each content item belongs.

[00964] Then, the receiver 300 first determines whether a content item is provided through NRT service by using the reception information in operation S8950. Here, the reception information of a content item may include relationship information indicating whether a content item is transmitted through a currently selected service or whether a content item is transmitted through another NRT services. For example, the reception information may include the available_on_current_service field. Additionally, when a content item is transmitted through another NRT service, the reception information of the content item may include the identifier of the other NRT service and the identifier of the content item in order to receive the content item. Accordingly, the receiver 300 determines whether a content item is provided through another NRT service by using the relationship information.

[00965] Additionally, the relationship information may include other_NRT_location_descriptor, and the service manager 350 of the receiver 300 determines whether a content item of NRT service that a current user selects to receive is transmitted through a selected service or another NRT service by using the relationship information, and identifies an object corresponding to the other NRT service to receive a content item therein.

[00966] If it is determined that the content item is not transmitted through another NRT service, the service manager 350 of the receiver 300 accesses a FLUTE session according to the obtained content item reception information, and receives

files configuring each content item through the above method.

[00967] Moreover, if it is determined that the content item is transmitted, the service manager 350 of the receiver 300 obtains the service identifier for another NRT service from the reception information in operation S8960 and, by receiving SMT and NRT-IT corresponding thereto, obtains content item reception information in another NRT service in operation S8970. Then, the service manager 350 may receive files configuring a content item in access to a FLUTE session on the basis of the content item reception information.

[00968] Here, content items in another NRT service may be launched by an entry content item of the reception selected NRT service. The service manager 350 of the receiver 300 identifies the entry content item of the reception selected NRT service, receives and launches the entry content item first, and displays an interactive interface in the entry content item. Additionally, the receiver 300 may receive or launch a content item of another NRT service in correspondence to a user input for the displayed interface.

[00969] Additionally, as mentioned above, a content item of the same NRT service may be launched or received by an entry content item. When each content item is received or launched, the service manager 350 of the receiver 300 first receives and launches an entry file in files configuring each content item, as mentioned above. The service manager 350 of the receiver 300 may receive a list of files from a FLUTE session through FDT before receiving a file of a content item, and identifies an entry file first on the basis of the FDT.

[00970] On the other hand, as described above, the receiver 300 may receive a trigger and perform a trigger action on a

trigger target TDO (for example, a specific content item) at a trigger time based on the trigger.

[00971] Moreover, the receiver 300 may receive a trigger and perform a trigger action at a trigger time. In this case, the receiver 300 may perform a trigger action on a specific content item of the receiver 300 and enable a trigger action command for a specific content item included in external devices other than the receiver 300 to be transmitted and performed to and at the external device

[00972] Moreover, the receiver 300 will be able to deliver the received trigger itself to a nearby external device, and the external device will be able to perform a trigger action on a trigger target content item at a trigger time.

[00973] Thus, the target content item of a trigger may be included in the receiver 300 or the external device. The external device may be referred to as, for example, any one of a second screen device, a link device to the receiver 300, and a companion device. According to an embodiment, the external devices may be referred to as a companion device linkable the receiver 300.

[00974] By using such link devices, it is possible to link an adjunct service (application, non-real time service object or TDO) executed on a link target device, to a broadcast service that the receiver 300 receives.

[00975] To that end, the transmitter 200 needs to signal to the receiver 300 for information on an adjunct service to link to information on a link device.

[00976] For example, the receiver 300 or the link devices may include content items to which the same trigger action is applied by the same trigger. Moreover, companion devices may

have various screen sizes and various user interfaces. Thus, it is possible to assign separate content items to each device and designate each of the separate content items as the target content items of a trigger.

[00977] Thus, a signaling method for the link between a broadcast service received by the receiver 300 and a link device and a corresponding operation will be described in an embodiment.

[00978] According to an embodiment, the transmitter 200 may generate and deliver link information to the receiver 300. The link information may include information on an external device (companion device) which links to a (real time or non real time) broadcast service received by the receiver, and include action information (for example, trigger information or event information) on an adjunct service to be performed at a companion device. On the other hand, the adjunct service may include an application, an NRT service or a TDO that may link to a broadcast service received by the receiver 300.

[00979] According to an embodiment, link information may be included in signaling information on an adjunct service in a descriptor form. The link information may include, for example, at least one of a companion device descriptor and a companion device action descriptor.

[00980] Such link information may be included in at least one of the above-described PSIP table and PSI table in the descriptor form and transmitted to the receiver 300 or may be included in at least one of non-real time tables (SMT, NRT-IT) and transmitted to the receiver 300. For example, the link information may be included and transmitted in a

specific event descriptor loop of an EIT for a specific broadcast program.

[00981] Fig. 74 shows syntax of a companion device descriptor according to an embodiment.

[00982] The companion device descriptor according to an embodiment may be transmitted to the receiver through a broadcast service or an internet service and the transmitter may also modify the same data structure to an XML form and transmit the modified structure by using HTTP or other protocols.

[00983] As shown in Fig. 74, the companion device descriptor according to an embodiment includes a descriptor tag field, a descriptor length field, a companion identifier field, an action type field, a number of filters field, a plurality of filter items, a description field, an URL length field, and an URL field, and each of filter items includes a user agent filter length field, a user agent filter field, a version length field, and a version field.

[00984] The descriptor tag field may be an 8-bit unsigned integer for identifying the descriptor as the companion descriptor.

[00985] The descriptor length field may be an 8-bit unsigned integer to define the length from a field immediately following this field to the end of the link descriptor.

[00986] The companion identifier field may be a 32-bit unsigned integer to represent unique identification information for identifying a companion device. The companion identifier field may be a unique identifier for identifying the types of external devices of various types that are connected to a home network. Moreover, the

companion identifier field may include information for identifying a link device connected to the home network.

[00987] Here, the transmitter 200 may also group companion devices and transmit identifiable information. Grouping information on the companion devices may include detailed model information or serial information on each companion device according to a companion identifier. The related description will be made below.

[00988] The action type field may be an 8-bit unsigned integer to represent an action for controlling the operation of an adjunct service that is executed on the companion device.

[00989] The number of filters field may be an 8-bit unsigned integer to represent the number of filter items.

[00990] The description field may be multiple strings to represent a description on an adjunct service associated with the descriptor. For example, the description field may represent an introduction phrase for an application associated with the descriptor and the introduction phrase may be expressed by using various languages.

[00991] The URL length field may be an 8-bit unsigned integer to represent the length of an URL field.

[00992] The URL field may be a variable to represent URL that may receive an adjunct service associated with the descriptor. For example, the URL field may include URL information that is used for a companion device to receive an application associated with the descriptor.

[00993] On the other hand, the plurality of filter items that are included in the descriptor may include information on a user agent of each companion device. Each of the filter

items may be repetitively designated by using a 'for' phrase as many times as the number that is represented in the number of filters field.

[00994] The user agent filter length field may be an 8-bit unsigned integer to represent the length of a user agent filter field.

[00995] The user agent filter field may be a variable to represent a user agent of a companion device that is identified by the descriptor. The user agent may represent a user agent that is defined in an HTTP protocol (RFC2616). The user agent may represent a presentation application that requests an HTML page. The user agent may represent, for example, a platform of a web browser.

[00996] When an application executed in a companion device that is identified by the descriptor is a web application, the user agent field may have a value to represent a user agent that is used in the web browser. Moreover, when an application executed in a companion device that is identified by the descriptor is an application for a specific platform, the user agent field may have a value to represent the specific platform.

[00997] Thus, the receiver 300 may search all external devices connected to the home network and compare the user agent of each device with the value of a user agent filter field to determine whether the device is a companion device that has the user agent represented in the descriptor.

[00998] The version length field may be an 8-bit unsigned integer to represent the length of a version field.

[00999] The version field may be a variable to represent the version of a companion device. The receiver 300 may identify

only a companion device of a specific version among companion devices that are searched by the user agent filter field on the basis of the version field.

[001000] As such, by using the companion device descriptor, the transmitter 200 may signal to the receiver 300 for information on an external device to execute an adjunct service that may link to a broadcast service.

[001001] For example, the transmitter 200 may designate a companion device identifier to execute an adjunct service to the companion device descriptor, designate the user agent of a companion device to execute an adjunct service to a user agent filter, designate version information on a companion device to execute an adjunct service, insert and transmit URL information to receive an adjunct service to the receiver 300.

[001002] Thus, the receiver 300 may receive a companion device descriptor in association with an adjunct service, identify a companion device to execute a corresponding adjunct service, the user agent of the companion device, the version of the companion device, and URL to receive the corresponding adjunct service by using the companion device descriptor, and transmit to the companion device an event message that enables the corresponding adjunct service to be executed on the companion device.

[001003] Fig. 75 shows a companion device action descriptor according to an embodiment.

[001004] A service provider may designate an action for an adjunct service executed on a companion device by using a companion device action descriptor. The action may include, for example commands to start, prepare or stop a specific application, according to a type, and may also include a

command to generate an event for a particular application.

[001005] The companion action descriptor may include a descriptor tag field, a descriptor length field, a companion identifier field, an action type field, an event length field, and an event filed, as shown in Fig. 75.

[001006] The descriptor tag field may represent a companion action descriptor, and the descriptor length field and the companion identifier field may represent information as described on the companion device descriptor above.

[001007] The action type field may be an 8-bit unsigned integer to represent an action type for controlling an adjunct service of a companion device.

[001008] Fig. 76 shows an action type according to a value of such an action type field.

[001009] On the other hand, when the value of the action type filed represents an event, the even length field and the event field may be included in the companion device action descriptor.

[001010] The event length field may be an 8-bit unsigned integer to represent the length of the event field.

[001011] The event field may be a variable to represent event information that is delivered for a specific adjunct service. The event field may be delivered to a companion device in a string form. The event field may include, for example, control information for controlling a specific application that is executed on a companion device.

[001012] Moreover, the event field may correspond to, for example, a trigger. If receiving the trigger, the receiver 300 may transmit trigger information included in the event field to the companion device at a trigger time. Moreover,

the receiver 300 may also transmit the received trigger itself to the companion device in an event form. The companion device that receives the trigger information or the trigger may provide a TDO service by performing a trigger action on a specific content item that is a trigger target adjunct service.

[001013] Fig. 77 is a flow chart of a method of operating a receiver according to an embodiment.

[001014] Firstly, the receiver 300 receives signaling information on an adjunct service in step S9000. For example, the PSIP/PSI section handler of the receiver 300 may parse PSIP or PSI and receive signaling information (VCT, PMT, etc.) on a broadcast service. Moreover, the service manager 350 of the receiver 300 may receive signaling information (SMT, NRT-IT) on a non-real time service through a non-real time service signaling channel. Moreover, the service manager 350 of the receiver 300 may also receive signaling information on a TDO through a broadcast stream, an SMT, or an NRT-IT.

[001015] An adjunct service may include an application, a broadcast service (non-real time service or real time service), or a TDO (trigger declarative object) as described above. The adjunct service may include an adjunct service linkable to a real time broadcast and may also include an adjunct service linkable to a non-real time service.

[001016] The receiver 300 that receives adjunct service signaling information may receive an adjunct service according to a user input or a trigger based on adjunct service signaling information. In addition, the adjunct service may be received by a companion device in response to

a request through the receiver 300. The related description will be made below.

[001017] If the adjunct service signaling information is received, the receiver 300 obtains link information from the adjunct service signaling information in step S9005.

[001018] The link information may include information on the adjunct service of an external device that is linked to a broadcast service received by the receiver 300 as described above.

[001019] The link information may include identification information on the above-described linkable external device (companion device), the type and user agent of a companion device, protocol information and version information thereon.

[001020] Moreover, the link information may include information on an adjunct service to be executed on the companion device. For example, the link information may include a content linkage to be executed on the companion device. Moreover, the link information may include URL information that may receive a content item. In addition, the link information may include event information or action information to be performed on the content item.

[001021] Thus, such link information may include at least one of the above-described companion device descriptor and companion device action descriptor.

[001022] Subsequently, the receiver 300 determines whether there is a command to provide an adjunct service, in step S9010. The receiver 300 may determine based on a user input or a trigger whether there is an adjunct service providing command for a signaled adjunct service.

[001023] For example, the service manager 350 of the

receiver 300 may determine that there is an adjunct service providing command, if there is a user's selective input for a received non-real time service. Moreover, the service manager 350 of the receiver 300 may determine that there is an adjunct service providing command, if a trigger for a received TDO is received and a trigger time arrives.

[001024] If there is an adjunct service providing command, the receiver 300 determines whether an adjunct service to be provided is a link service, in step S9020. If there is an adjunct service that is provided by an external device (for example, a companion device) and not by the receiver 300, the receiver 300 may determine that there is a link service.

[001025] The receiver 300 may determine based on the link information whether there is a link service. For example, the receiver 300 may determine based on signaling information whether there is link information, the signaling information corresponding to an adjunct service for which a command is provided. The receiver 300 may determine that the adjunct service for which a command is provided is a link service, if identifying link information from at least one of PSIP/PSI, an SMT and an NRT-IT. Moreover, the receiver 300 may obtain link information from a trigger that generates an adjunct service providing command, and determine that a service is a link service.

[001026] If a service is not a link service, the receiver 300 provides an adjunct service in step S9040. The receiver 300 may provide a service by directly executing an adjunct service to be provided. For example, the receiver 300 may provide a non-real time service by executing a content object corresponding to a selected non-real time service. Moreover,

the receiver 300 may provide a TDO service in association with a real time broadcast by executing a target TDO of a trigger that is received at a trigger time.

[001027] If it is determined that the adjunct service is a link service, the receiver 300 identifies an external device based on the link information, in step S9025.

[001028] The receiver 300 may identify an external device to provide an adjunct service based on a companion device descriptor that is included in the link information. The receiver 300 may identify an external device to provide an adjunct service based on at least one of the unique identifier, version information, user agent information and protocol information on a companion device that is an external device to provide an adjunct service through each field of the above-described companion device descriptor.

[001029] In addition, the receiver 300 transmits to the identified external device a command for an adjunct service in step S9030.

[001030] Firstly, the receiver 300 may determine by using a network interface whether there is a companion device corresponding to the identified external device on a home network, and transmit to the device a command for an adjunct service if there is a companion device corresponding to the identified external device

[001031] In addition, if it is intended to extract a companion device action descriptor from link information and provide a corresponding adjunct service, the receiver 300 may determine an action type to be used for transmission, generate and transmit a command for the adjunct service according to an action type.

[001032] For example, the receiver 300 may transmit an execution command for a specific non-real time service that is included in a companion device, based on link information. Moreover, the receiver 300 may transmit a trigger action command for a TDO that is included in a companion device, based on link information. In addition, the receiver 300 may also transmit an event for a specific application that is executed on a companion device, based on link information.

[001033] On the other hand, if receiving a trigger, the receiver 300 may also transmit a trigger itself in stead of a command for an adjunct service.

[001034] In addition, if the command for the adjunct service is transmitted, an external device provides an adjunct service in step S9035. The external device may perform the execution of a non-real time service, the execution of a TDO, or an event operation on an application based on the received command. Moreover, if receiving a trigger, the external device may also perform a trigger action on a TDO at a trigger time based on the trigger.

[001035] As such, since the receiver 300 may receive link information and control the adjunct service providing operation of the external device in association with a broadcast service according to the link information, it is possible to effectively provide the broadcast service and it is possible to enhance the convenience of a user if the receiver 300 is used simultaneously with the external device.

[001036] As described above, such link information may be transmitted as a part of a PSIP/PSI table or a non-real time service signaling table.

[001037] Fig. 78 shows a case where link information is

included in an SMT.

[001038] As shown in Fig. 78, the link information may be transmitted as a part of a service level descriptor loop of the SMT in a descriptor form.

[001039] In Fig. 78, VCH0, a virtual channel, may be a channel over which an additional non-real time service stream is transmitted along with a broadcast stream. In an embodiment, the receiver 300 may receive a non-real time service stream based on a PMT, extract the SMT, an NRT-IT from the non-real time service stream, and obtain link information from the SMT. In this case, the receiver 300 may determine that a non-real time service associated with a specific virtual channel may be provided through a link at an external device. Moreover, the receiver 300 may transmit the reception or execution command of a corresponding non-real time service to an external device based on link information and an NRT-IT. Thus, a user may receive from the external device an additional non-real time service associated with a broadcast service that the user is enjoying.

[001040] Fig. 79 shows a case where link information is included in an EIT, a VCT or a PMT.

[001041] As shown in Fig. 79, the link information may be included in a PSIP/PSI table in a descriptor form. Moreover, the link information may represent that there is a specific program that is indicated by a table including the link information, or an external device application corresponding to a specific virtual channel.

[001042] For example, if the link information is included in the PMT, the receiver 300 parsing the PMT from PSI may determine that there is an external device application that

corresponds to a specific stream identified by the PMT.

[001043] Moreover, if the link information is included in the VCT, the receiver 300 parsing the VCT from PSIP may identify a specific channel loop in the VCT to which the link information belongs, and determine that there is an external device link application related to the identified specific channel.

[001044] On the other hand, if the link information is include in the EIT, the receiver 300 receiving the EIT may determine that there is an external device link application that corresponds to a specific event of the EIT that includes the link information.

[001045] Fig. 80 is a diagram for explaining how link information is included in non-real service signaling information according to an embodiment.

[001046] As shown in Fig. 80, the link information may be included in a non-real time service table (for example, an NRT-IT) in a descriptor form. The link information may correspond to a specific non-real time service or a specific content item of a non-real time service, and may be included in the SMT or the NRT-IT according to a corresponding relationship.

[001047] For example, if obtaining a companion device descriptor from a descriptor loop of an NRT-IT for the content item Content0 as shown in Fig. 79, the receiver 300 may determine that there is an adjunct service of an external device associated with Content0. Thus, if storing or playing Content0 is selected by a user input, the receiver 300 may display to a user an adjunct service of an external device linkable thereto, for example, information on a specific

application.

[001048] On the other hand, a companion device descriptor among the above-described link information may be included in the NRT-IT as shown in Fig. 80 and may be included in a content item descriptor loop that corresponds to a specific content item. In this case, the companion device descriptor may include at least two pieces of information that are described below.

[001049] Firstly, the companion device descriptor may include information on a target device to execute a content item corresponding to a content item descriptor loop that includes the companion device descriptor. In this case, the receiver 300 or the target device may obtain, by using an SMT or an NRT-IT, positional information for obtaining the content item corresponding to the content item descriptor loop that includes the companion device descriptor. The content item to be executed may be transmitted through a FLUTE session by using a broadcast service or may be transmitted by using an internet service. The target device may obtain identification information on the target device to execute the content item, by using a companion id that is included in the above-described companion device descriptor. Thus, the transmitter 200 may provide an adjunct service (reception and execution of a content item) at an external device by using a companion device descriptor.

[001050] Secondly, the companion device descriptor may include signaling information on an external device application that is equivalent to the content item corresponding to the content item descriptor loop that includes the companion device descriptor. A content item

that configures a non-real time service object transmitted through a broadcast network or an internet network may be optimized for the receiver 300. However, if the companion descriptor is transmitted, the content item may be executed at the external device. Thus, the transmitter 200 may separately transmit an application that has the same function as the content item and that is optimized for the external device, and signal by using the companion device descriptor.

[001051] Signaling information on such an external device application may include URL information for obtaining an application or URL information on a specific descriptor file that represents files configuring the application. In this case, the receiver 300 or the external device may obtain URL information from a companion device descriptor corresponding to a specific content item, extract application signaling information from the URL information, and receive an application corresponding to a specific content item by using the extracted application signaling information or display application information. Thus, the transmitter 200 may provide an adjunct service (reception and execution of an application equivalent to the content item) at an external device by using the companion device descriptor.

[001052] On the other hand, the transmitter 200 may signal information on a target device to execute a content item by using a capabilities descriptor used in an existing ATSC non-real time service without separately designating a companion device descriptor. In this case, the receiver 300 may receive a capabilities descriptor corresponding to a specific content item and identify an external device to execute the specific content item based on the capabilities descriptor.

[001053] Fig. 81 is a diagram for explaining a capabilities descriptor according to an embodiment, and Fig. 82 shows a capabilities category code table according to an embodiment.

[001054] Referring to Fig. 81, the capabilities descriptor according to an embodiment may be defined by modifying a capabilities descriptor specification for representing a receiver capabilities requirement in an existing ATSC NRT.

[001055] As shown in Fig. 81, the capabilities descriptor may be divided into an individual capabilities part and a sets of capabilities part but a description of each field of any part may be the same and all capabilities in the sets of capabilities part are interpreted as OR analysis and thus there is a difference in that the receiver 300 may determine that the receiver 300 satisfies the entire sets of capabilities if satisfying one capability,

[001056] Thus, only the individual capabilities part will be described. For example, each field may be defined as follows.

[001057] The descriptor tag may represent a value to identify this field as a capabilities descriptor. This value may be defined later.

[001058] The descriptor length may represent the byte length from a field immediately following this field to the end of this descriptor.

[001059] The capability code count may represent the number of capability code values to be described below.

[001060] The essential indicator may represent whether a support for a capability represented by a capability code following this field is essential for the receiver to

meaningfully present a content item or a service corresponding to this descriptor. For example, if the essential indicator has a value of 1, it may represent that a capability represented by a capability code is essential, and if 0, it may represent that a capability represented by a capability code is not essential.

[001061] The capability code may be a 7-bit unsigned integer that represents a specific capability defined by a table. For example, the capability code may be defined by a table, Table A.1 of an ATSC NRT service.

[001062] The format_identifier may be a 32-bit field that is defined in the registration descriptor of Section 2.6.9 of ISO/IEC 13818-1 [51]. A value of the format_identifier may be registered and recognized by SMPTE Registration Authority, LLC, a format identifier registration/management authority (It may be seen on Http://www.smpte-ra.org/mpegreg.html). This field may be used for identifying the capability code above.

[001063] The capability string count may be an 8-bit unsigned integer that represents the number of capability string values.

[001064] The essential indicator may represent whether a support for a capability represented by a capability string following this field is essential for the receiver to meaningfully present a content item or a service corresponding to this descriptor. For example, if the essential indicator has a value of 1, it may represent that a capability represented by a capability string is essential, and if 0, it may represent that a capability represented by a capability string is not essential.

[001065] The capability category code may be a 7-bit unsigned integer that represents a specific capability defined by a table. For example, the capability category code may be defined by a table as shown in Fig. 82.

[001066] As shown in Fig. 82, according to an embodiment, the capability category code field may include information on a target external device (companion device). Especially, if the capability category field has a value of 0x06 in Fig. 82, it may represent that a capability string value corresponding to the capability category code field is information on a companion device. Moreover, the capability string value corresponding to the capability category code field may include user filter information or version information on the companion device as information for identifying the companion device. The receiver 300 may obtain the user filter and version information from the capability string, determine user filter and version information on an HTTP protocol used in the companion device, and identify companion devices that use it.

[001067] Here, the capability string() for representing the above-described user filter and version information may be configured in the form of <user agent>/<version> or may configured in the form of a combination that "user agent" is separated from "version" by a space.

[001068] Fig. 83 is a flow chart of a method of obtaining link information by using a capabilities descriptor by a receiver according to an embodiment.

[001069] Firstly, the receiver 300 receives non-real time service signaling information and obtains an SMT and an NRT-IT in step 9100. A way of obtaining the SMT and NRT-IT has

been described above.

[001070] In addition, the receiver 300 parses a capabilities descriptor from the obtained SMT or NRT-IT in step S9105. Through the service manager 350, the receiver 300 may search a descriptor tag corresponding to the capabilities descriptor from a descriptor loop corresponding to a content item configuring a specific non-real time service object or a specific non-real time service and obtain the capabilities descriptor.

[001071] In addition, the receiver 300 determines from the capabilities descriptor whether there is a field where a value of a capability category code is 0x06, in step S9110. The receiver 300 may parse the capability category code field of the capabilities descriptor through the service manager 350 and determine whether the value is 0x06. As described with reference to Fig. 82, if the value of the capability category code is 0x06, it may represent that a capability string field corresponding to the capability category code is information on a companion device.

[001072] Thus, the receiver 300 parses a capability string () field corresponding to a field where a value of the capability category field is 0x06. The receiver 300 may determine from the value of the capability category code that the capability string represents information on a companion device.

[001073] In addition, the receiver 300 obtains information on a companion device from the capability string in step S9120. As described above, the companion device information in the capability string may include information on an external device linked to a non-real time service object

represented by the capabilities descriptor. The companion device information may represent user filter and version information that is used in an HTTP protocol of a device corresponding to a companion device among external devices, as described above. Thus, the receiver 300 may obtain companion device information and identify an external device to provide an adjunct service linked to a broadcast service. Moreover, such companion device information may further include a lot of information represented by the above-described link information, in addition to identification information on the companion device.

[001074] Fig. 84 shows a content items group descriptor according to another embodiment.

[001075] A service provider may group content items providing the same service through various link devices. The transmitter 200 may signal group information by grouping a group of content items providing the same service, by using the content items group descriptor.

[001076] As shown in Fig. 84, the content items group descriptor may include content items group id information, num content items information, and content item identification information, in addition to basic information such as a descriptor tag, a descriptor length, etc.

[001077] The content items group id information may represent an identifier that may uniquely identify a content item group for a service represented by this descriptor. The receiver 300 may identify a unique content item in a service range based on the content items group id information.

[001078] The num content items information may represent the number of content items that are included in a content

item group.

[001079] The content linkage may represent an identifier that indicates each of content items in a content item group.

[001080] The content items group descriptor may be included in non-real time service signaling information. Thus, the receiver may obtain the content items group descriptor from an SMT or an NRT-IT and determine that a non-real time service corresponding to the obtained descriptor loop may be executed on a plurality of devices.

[001081] In addition, the receiver 300 may determine that a plurality of content items in the content items group descriptor all executes the same service and executable devices are different from each other.

[001082] In addition, if a target object obtained from a trigger represents a content item group, the receiver 300 may determine that services for all TDOs (for example, performing a trigger action) represented by the content items group descriptor needs to be provided from each device, and perform corresponding proper processing.

[001083] Fig. 85 is a flow chart of a method of providing a TDO service by using a content items group descriptor by a receiver.

[001084] Referring to Fig. 85, the receiver 300 obtains link information in step S9200. The receiver 300 may obtain link information through the service manager 350. The receiver 300 may obtain link information from a non-real time service table or a broadcast table, and receive link information through the internet network. The link information may include link target device information and link target adjunct service information on which descriptions

have been made above. The link target device information may be transmitted as a part of, for example, a companion device descriptor and the link target adjunct service information may be transmitted as a part of, for example, a companion device action descriptor, or in the form of an application through the internet network.

[001085] In addition, the receiver 300 receives a trigger in step S9205. The receiver 300 may receive the trigger through the service manager 350 connected to a receiving unit. The trigger may include trigger time information, trigger action information, and trigger target object information, as described above. The trigger target object information may include content items id information on a specific TDO.

[001086] If receiving the trigger, the receiver 300 obtains a content item group identifier from the trigger target object information in step S9210.

[001087] The receiver 300 may obtain the trigger target object information through the service manager 350. In addition, it is possible to obtain a trigger target content item group identifier based on the trigger target object information. The content item group identifier may represent identification information for identifying a group of content items for identifying TDOs that provide the same service.

[001088] If obtaining the content item group identifier, the receiver 300 obtains the identifiers of content items corresponding to each device from a content items group descriptor. The receiver 300 may obtain the content items group descriptor including the content item group identifier from an SMT or an NRT-IT. In addition, the receiver 300 may obtain content linkage from the content items group

descriptor.

[001089] In addition, the receiver 300 transmits a trigger action on each of content items to each of link devices at a trigger time in step S9220. The receiver 300 may identify link devices that may provide content items corresponding to content linkages based on link information. Thus, the receiver 300 may transmit a trigger action to a link device or a companion device corresponding to the content item of a TDO at a trigger time.

[001090] Accordingly, each link device that receives each trigger action may provide a TDO service to a user by applying the trigger action to the TDO.

[001091] As such, if the TDO service is provided from an external device, a trigger that aims at the content item of the TDO service may include content items group id information on a target content item. Moreover, the trigger may include identification information on the content item of a TDO provided from the receiver 300, in addition to the content item of a TDO provided from the external device.

[001092] Thus, the receiver 300 may deliver the trigger action to the external device if obtaining the content items group id information.

[001093] Moreover, in an embodiment, the receiver 300 may also transmit a trigger itself and not a trigger action to a link device irrespective of a trigger time. In this case, a link device receiving the trigger may perform a trigger action on a TDO in a trigger time. The trigger reception and transmission operations of the receiver 300 may be equally applied to a TDO data stream. For example, if it is determined that the target content item of the TDO data

stream is a content item that is executed on an external device, the receiver 300 may transmit the TDO data stream itself to the external device.

[001094] On the other hand, a companion id that is included in the above-described link information and used for identifying a companion device may represent the type of a device as described above, more particularly the model name or serial number of a specific device.

[001095] To that end, the companion id may have various values and may be represented by suing a 32-byte string value. For example, the companion id may have a model name such as APA-A1332. Moreover, the companion id may have a unique value that depends on the screen resolution of a link device. In addition, the companion id may have a value to represent at least one of hardware version information and software version information, as a unique value of the link device.

[001096] Moreover, for example, companion devices that have similar functions or like hardware and software functions may be designated as one group or a plurality of groups. In this case, the transmitter 200 may group the companion devices by using such a companion id, assign a companion group id to each group and signal to the receiver for grouping information. Thus, if there are is plurality of link devices that may execute a specific content item, the transmitter 200 may simply signal by using the companion group id. Such grouping information may be transmitted as a part of a companion device group descriptor.

[001097] Fig. 86 shows a companion device group descriptor according to another embodiment.

[001098] The companion device group descriptor according

to an embodiment may include companion device group id, num companion devices, and companion id fields, in addition to descriptor tag and descriptor length fields for identifying this descriptor.

[001099] The companion device group id may represent identification information that may uniquely identify a companion device group in an adjunct service corresponding to the companion device group descriptor. The receiver 300 may identify a unique companion device group in a service range by using such identification information.

[001100] The num companion devices may represent the number of companion devices that are included in a companion device group.

[001101] The companion id may represent a value for identifying a companion device as described above, and may represent companion devices that are included in a companion device group corresponding to a specific companion device group id.

[001102] Such a companion device group descriptor may be transmitted as a part of non-real time service signaling data. For example, the companion device group descriptor may be placed in the non-real time service information level descriptor loop of an NRT-IT.

[001103] In addition, the receiver 300 may identify a non-real time service corresponding to a non-real time service information level descriptor that has obtained the companion device group descriptor, and may identify a corresponding companion device group.

[001104] Thus, the receiver 300 may determine based on the companion device group descriptor that companion devices

belonging to a companion device group may perform an adjunct service corresponding to the same content item or content item group. In addition, the receiver 300 may transmit, to executable companion devices, information required for providing an adjunct service.

[001105] Figs. 87A and 87B show a companion device group descriptor according to another embodiment.

[001106] As shown in Fig. 87A, the companion device group descriptor may further include companion id bytes information corresponding to the companion id of Fig. 86.

[001107] The companion id bytes may include 32-byte string information that represents the detailed model name of each companion device grouped by a companion device group id. For example, the companion id bytes may be designated to one to one correspond to the model name of each companion device such as LG-US8800 or APA-A1332, as shown in Fig. 87A. Thus, the receiver 300 may easily obtain detailed information such as the model name of a companion device belonging to a specific companion device group, from the companion id bytes field. Fig. 87B shows such a companion device group descriptor.

[001108] If the companion device group descriptor is defined as shown in Fig. 87B, the receiver 300 may configure and manage the companion id designated by a service provider and the companion id bytes as one array. Moreover, each companion identifier may be associated with each companion device group id. Thus, the receiver 300 may obtain model name information on companion devices in a descriptor through the companion device group descriptor of Fig. 87B and perform proper processing.

[001109] It is assumed that the companion id bytes field companion_id_bytes is herein a string and has 256 bytes, but the field may be adjusted if necessary. For example, a service provider may adjust the value of the num_companion_id_bytes to four bits so that the companion id bytes field companion id bytes represents a 16-byte string.

[001110] On the other hand, Figs. 88A and 88B show a companion device group descriptor according to another embodiment, and Fig. 89 shows the syntax for the descriptor of Figs. 88A and 88B.

[001111] As shown in Figs. 88A and 88B, in order to signal the detailed model name of a companion device, a service provider may separately designate a companion identification descriptor corresponding to a companion identifier that is included in the companion device group descriptor. Such a companion identification descriptor may be separately transmitted and included in the companion device group descriptor.

[001112] Fig. 88A represents when the companion identification descriptor is separately transmitted. The receiver 300 may receive the companion identification descriptor, extract a companion id byte corresponding to each companion identifier and obtain the model name of a companion device.

[001113] Fig. 88B represents when inserting and transmitting companion identification descriptors into the reserved region of the companion device group descriptor. The receiver 300 may extract the companion identification descriptors from the reserved region of the companion device group descriptor, extract a companion id byte from each

companion identification descriptor and obtain detailed information on a companion device. The detailed information may include string information such as LG-US8800 that is a model name.

[001114] Fig. 89 shows exemplary syntax for the companion identification descriptor of Figs. 88A and 88B.

[001115] As shown in Fig. 88, the companion identification descriptor may represent detailed identification information on a companion device by using a companion id bytes field.

[001116] In an embodiment, the companion identification descriptor may replace a companion identifier of syntax as shown in Figs. 86A and 86B. Thus, the receiver 300 may also identify a companion device by using only the companion identification descriptor and not the companion identifier.

[001117] Fig. 90 is a diagram for explaining a home network including a receiver and an external device according to an embodiment.

[001118] As shown in Fig. 90, the home network according to an embodiment may include the receiver 300, an external device 400, and an application server 500.

[001119] The receiver 300 may receive a terrestrial broadcast service from an external network. The receiver 300 may receive information on a broadcast program that a user is viewing, and receive link information for linking the broadcast program to the external device 400. Link information may include at least one of information on a linkable external device, information on an application to be executed on an external device when a link is made, and event information on an application. For example, the link information may include trigger target content information or

event information corresponding to a trigger action.

[001120] In addition, the receiver 300 may identify a list of linkable external devices from the link information, and deliver application information to a corresponding external device 400 if there is an external device 400 connected to a home network among the linkable external devices. The application information may include information on an application to be executed on the external device 400 or event information. In addition, the receiver 400 may also receive event information that is generated at the external device 400.

[001121] If there is no application to have to be executed, the external device 400 may check downloading from a user based on the received application information, and may receive a corresponding application from the application server if the user accepts. For example, the external device 400 may receive an application from the application server 500 based on a web site delivered from the receiver 300 or the URL of an app store.

[001122] On the other hand, the receiver 300 may transmit an event for a specific application to a corresponding external device, if receiving an event to be linked to an external device 400 through a broadcast service, such as a trigger. The receiver 300 may determine based on a response event from the external device 400 whether the specific application is being executed, downloaded but not executed, or not downloaded, and output the result to a user. In addition, based on the determination result, the receiver 300 may transmit an event to the external device, output a message that the external device 400 needs to execute an

application, or output a massage for querying whether to download.

[001123] Fig. 91 is a diagram for explaining a home network including a receiver and an external device according to another embodiment.

[001124] As shown in Fig. 91, the home network according to an embodiment may include the receiver 300, the external device 400, the application server 500, and a push notification server.

[001125] The receiver 300 may receive information on an application to be executed on the external device 400 by forming a link to a broadcast service. In addition, the receiver 300 may receive event (for example, a trigger) information on an application.

[001126] In addition, the receiver 300 may create a list of external devices 400. To this end, the receiver 300 may transmit a device query message to a home network and the external device 400 may transmit a respond to the query to the receiver 300.

[001127] On the other hand, the external device 400 may transmit a registration request as a device registration message to the receiver 400, and the receiver 400 may create a list of external devices 400 based on it. In this case, the external device 400 may perform a device registration process for the receiver 300 by using a specific application.

[001128] Moreover, the external device 400 may transmit device identification information for push notification to the receiver 300. The receiver 300 may request the push notification server 600 to perform an event for the external device 400 based on the device identification information for

push notification.

[001129] For example, if it is determined based on received link information that the external device 400 present on a home network is in a list of external devices linkable to a broadcast service, the receiver 300 may require the push notification server 600 to perform application download request push notification corresponding to device identification information on the external device 400.

[001130] Accordingly, the push notification server 600 may check whether the external device 400 corresponding to device identification information is registered, and transmit the application download request push notification to the checked external device 400. The external device 400 that receives push notification may put an application downloading query to a user, and download a corresponding application based on application information (URL, etc.) that is received from the receiver 300, if the user selects downloading.

[001131] On the other hand, if the receiver 300 receives an event (trigger, etc.) corresponding to an application through a broadcast service or an internet network, the receiver 300 may determine whether the external device 400 is executing or downloading an application corresponding to an event through the communication with the external device 400.

[001132] In addition, the receiver 300 may require the above-described push notification server 600 to perform application execution push notification corresponding to device identification information on the external device 400, if the application is not being executed. Accordingly, the push notification server 400 may transmit execution push notification to the external device 400.

[001133] In addition, if the execution is being executed, the receiver 300 may deliver event information (a trigger action, trigger itself, etc.) to a corresponding external device 400.

[001134] However, if the downloading is not performed, the receiver 400 may require the push notification server 600 to perform application download request push notification corresponding to device identification information on the external device 400, as described above. Accordingly, if downloading by the external device 400 is completed, the external device may transmit the start message of an application to the receiver 300 and the receiver 300 may transmit event information in response thereto.

[001135] Fig. 92 is a flow chart of a message delivery process on a network as shown in Fig. 91.

[001136] Referring to Fig. 92, firstly, the receiver 300 may be, for example, a TV, and the external device 400 may be, for example, a companion device and portable terminal such as a smart phone, a tablet, etc.

[001137] The receiver 300 requests the external device 400 to provide device information in step S9300 and transmits the device information to the receiver 300 in response thereto in step S9305.

[001138] In addition, the external device 400 transmits a device registration request to the receiver 300 in step S9310. If the receiver 300 directly performs device registration according to the setting and operation of the receiver 300, this step may be skipped.

[001139] In addition, the transmitter 200 transmits link information to the receiver 300 in step S9315. The link

information may include link target device information and link application information.

[001140] If receiving link information, the receiver 300 requests the push notification server 600 to perform push notification for adjunct service (for example, an application or a non-real time service) reception corresponding to the link information in step S9320.

[001141] In addition, the push notification server 600 transmits push notification of the adjunct service reception to the external device 400 in step S9325.

[001142] Subsequently, the external device 400 that receives push notification transmits a request for downloading an adjunct service to the application server 500 via a user's check in step S9330.

[001143] In addition, the application server 500 transmits, to the external device 400, enhanced service data corresponding to the request for downloading the adjunct service in step S9335. The enhanced service data may include binary data for driving an application.

[001144] Fig. 93 shows a home network according to another embodiment, and Fig. 94 is a flow chart of a message delivery process on a home network as shown in Fig. 93.

[001145] Referring to Fig. 93, the home network according to another embodiment may include the receiver 300, the external device 400, the application server 500, the push notification server 600, and a subscription server 700.

[001146] The subscription server 700 may perform the role of managing user subscription information and managing whether to receive and transmit a service by determining whether a user is a member after selecting member management

and some services. The subscription server 700 may be connected to a broadcast network or an internet network and transmit subscription information to the external device through the networks.

[001147] Moreover, the subscription server 700 may control the push notification server 600. For example, the subscription server 700 may control so that a subscription checked device is connected to a network and the push notification server 600 that is in a standby state pushes and notifies a software update or a new service introduction message to the subscription checked device. The push notification server 600 may transmit push notification such as a specific event dependent message or a download request message to the external device 400, accordingly.

[001148] In addition, the receiver 300 may transmit member information through the communication with the subscription server 700, and the subscription server 700 may check the member information, determine whether a user is a member, and perform adjunct service management.

[001149] Moreover, the external device 400 may also transmit member information through the communication with the subscription server 700, and the subscription server 700 may check the member information, determine whether a user is a member, and perform adjunct service management.

[001150] The subscription server 700 may manage the receiver 300 and the external device 400 on the same home network by using one account.

[001151] Moreover, if the external device 400 searched on the home network is a non-member device, the receiver 300 may also output a subscription recommending message for a

corresponding external device 400 to a user.

[001152] Referring to Fig. 94, firstly, the receiver 300 transmits member information on the receiver 300 to the subscription server 700 in step S9400. In addition, the external device also transmits member information on the external device 400 to the subscription server 700 in step S9405. The subscription server 700 may check based on the received member information whether the external device is a member device and manage the receiver 300 and the external device 400 that belong to the same network, as described above.

[001153] On the other hand, the receiver 300 requests the external device 400 searched on the home network to provide device information in step S9410 and the external device 400 transmits related device information to the receiver in step S9415. The receiver 300 may determine based on device information whether the external device is a member device of the subscription server 700, and output a subscription recommending message to a user if the external device is not a member device of the subscription server 700.

[001154] In addition, the receiver 300 receives link information that a service provider provides through the transmitter 100, in step S9420. The receiver 300 may determine based on the link information whether there is an external device linkable to a broadcast service or a non-real time service.

[001155] In addition, the receiver 300 transmits to the subscription server 700 an adjunct service download request message that includes linkable external device information (for example, a companion device descriptor) and external

device information connected currently to the home network, in step S9425.

[001156] The subscription server 700 selects an external device 400 capable of installing an adjunct service based on the linkable external device information and the external device information connected to the home network that are received, and transmits, to the push notification server 600, adjunct service download push notification including identification information on a corresponding external device 400.

[001157] In addition, the push notification server 600 transmits the adjunct service download push notification to the external device 400 in step S9435.

[001158] Subsequently, the external device 400 receiving the push notification transmits to the application server 500 a request for downloading an adjunct service via a user's check in step S9440.

[001159] In addition, the application server 500 transmits enhanced service data corresponding to the adjunct service download request to the external device in step S9445. The enhanced service data may include binary data for executing an application.

[001160] Fig. 95 is a flow chart of an event message delivery process of a home network including a receiver according to another embodiment.

[001161] Referring to Fig. 95, firstly, the transmitter 200 transmits an event message for an application to the receiver 300 in step S9500. The transmitter 200 may select an application in the external device 400 and transmit an event message for the application. For example the

application may be a TDO application and the event message may include a trigger.

[001162] The receiver 300 receiving an application event transmits the application event to the external device 400 corresponding to an application event message in step S9505. The application event that the receiver 300 transmits to the external device 400 may include information on a target application and operation information on an application. For example, in the case of a TDO, it may include at least one of information on a trigger target object, trigger action information and trigger time information.

[001163] The external device 400 receiving the application event determines whether a target application is being executed, in step S9510. If the target application is not being executed, the external device 400 transmits a reception rejection message to the receiver 300 in step S9515, and if the target application is being executed, the external device performs the received event on the target application in step S9550.

[001164] On the other hand, the receiver 300 receiving the reception rejection message delivers the application event request message to the subscription server 700 in step S9515. The application event request message may include application information and identification information on the external device 400.

[001165] If receiving the application event request message, the subscription server 700 checks whether the external device 400 is a member device, and transmits to the push notification server 600 an event performing push notification request message for the external device 400, in

step S9520.

[001166] In addition, the push notification server 600 transmits the push notification message for performing an event in response to the push notification request message in step S9525.

[001167] The external device 400 receiving the push notification message performs a user query on executing an application, and if a user selects executing the application, the external device 400 executes a corresponding application in step S9530.

[001168] In addition, if the application is executed, the external device 400 transmits to the receiver 300 an application executing message including information that the execution

[001169] On the other hand, after receiving the application executing message, the receiver 300 transmits an application event back to the external device 400 in step S9545 if an application event message is received from the transmitter in step S9550.

[001170] Subsequently, since the application is being executed as determined in step S9510, the external device 400 performs an operation corresponding to the received application event in step S9550 and transmits a response message generated from performing the operation to the receiver 300, in step S9555. Subsequently, if it is checked that an event due to a user input or the event reception form the transmitter 200 occurs, the receiver 300 may directly transmit the event to the external device 400.

[001171] In a method of providing an adjunct service linking an external device to a receiver in such a manner, a

service provider may easily perform service management through a subscription server.

[001172] Figs. 96 to 98 show the operations of a receiver 300 when there is an adjunct service linkable to an external device 400.

[001173] Fig. 96 shows a display screen of the receiver 300 for notifying a user that there is an external device application corresponding to a program that the user is currently viewing, when such an external device application is present. As shown in Fig. 96, the receiver 300 may display based on link information that there is an external device application corresponding to the Iris program that a user is currently viewing, and the receiver 300 may identify a plurality of external devices 400 connected currently to a home network and display a query screen for downloading to or execution on specific external devices 400.

[001174] For example, the receiver 300 may determine that an external device application corresponding to a companion device descriptor is linked to a program that a user is currently viewing, if obtaining the companion device descriptor from an EIT.

[001175] Moreover, the receiver 300 may obtain an identifier of an application included in the EIT, for example, a content item identifier, matches and determines the attributes of a corresponding content item with an NRT-IT, obtain link information corresponding to the content item from the NRT-IT, and determine that there is an application that is executable on an external device.

[001176] On the other hand, the receiver 300 identify external devices 400 corresponding to the user agent filter

of the external device obtained from the link information and display only executable external devices on the above-described query screen. In addition, a user may instruct to download or execute an application to or on a desired external device 400 through a menu on the screen.

[001177] Fig. 97 shows the operation of the receiver 300 displaying that there is an external device application linkable to a program that a user is currently viewing, when such an application is present. As shown in Fig. 97, the receiver 300 may display to a user whether there is an external device application, which is determined from link information. Moreover, the receiver 300 may display the type of external devices that may execute external device applications, in the form of a list that includes HDTV, CC, and iPhone.

[001178] On the other hand, Fig. 98 shows the operation of the receiver 300 representing such link information on EPG. As shown in Fig. 98, the receiver 300 may display an indicator 301 representing the presence of a link device application on EPG for a specific program that has a link device application. Thus, a user may easily determine from EPG that a specific program is linked to an external device. Moreover, the user may transmit a command to download or execute an application to or on an external device through the selective input of a specific program on EPG.

[001179] Figs. 99A and 99B show the operations of an external device 400 according to an embodiment.

[001180] As shown in Fig. 99A, if there is no event target application, the external device 400 receiving application information (for example, event information and application

information linked to the broadcast program Iris) from the receiver 300 may output a download query interface 401 for downloading the event target application. A user may perform a download input or a cancel input through the download query interface 401.

[001181] Moreover, Fig. 99B shows the operation of an external device 400 when there is an event target application according to application information received from the receiver 300 but the application is not currently executed. As shown in Fig. 99B, the external device 400 may output an execution query interface 402 for executing a program linked to the broadcast program Iris. A user may perform a download input or a cancel input through the execution query interface 402.

[001182] On the other hand, such link information may include action information on a TDO. The TDO action information may include the same information as the above-described companion device action descriptor or information corresponding thereto. In this case, TDO files or TDO action information may be transmitted through another network or another service. For example, the TDO action information may be transmitted through DTV-CC. The TDO action information may be transmitted by using an EXT1 code that is included in the DTV-CC.

[001183] Figs. 100 to 102 are diagrams for explaining how link information is transmitted by using DTC-CC according to an embodiment.

[001184] As shown in Fig. 100, the receiver may process the DTV-CC by using blocks. Each block performs the following function.

[001185] An MPEG-2 Transport Processor separates each elementary stream of single MPEG-2 TS. A Transport Processor driver manipulates some interrupts from the MPEG-2 Transport Processor, gets information and starts a DMA channel to copy segments of userdata including DTV-CC. It sends an event to a DTV-CC module when completing to copy single segment of userdata into a buffer. A DTV-CC module gets DTV-CC packets from the buffer (userdata) whenever the Transport processor driver sends the event of completeness of single copy process. If the DTV-CC module finds the service number for the ATSC 2.0 and the command is delivered through this service number, then it passes this command to an Adjunct service module.

[001186] The Adjunct service module processes the received ATSC 2.0 command and initiates a proper module in a DTV system (e.g, the proper module would be a web browser and/or an NRT module to get TDO(s)).

[001187] As shown in Fig. 100, the transmitter 200 may transmit command information that may be displayed on a DTV-CC screen through a DTV-CC stream.

[001188] Fig. 101 shows an example of command information.

[001189] The command information may be in the trigger message form of Fig. 101, represent whether there is an application for a companion device, and the ATSC 2.0 receiver 300 may determine by using the command information whether there is an application for a companion device, according to an embodiment.

[001190] In Fig. 101, the Trigger_id may represent the unique id of a trigger and the Version may represent the current version information on the trigger. In addition, the Target_device_type may be defined as the type of the ATSC 2.0

receiver. The TDO_action represents action information on the target TDO of the trigger as described above.

[001191] The command information of Fig. 101 transmitted to DTV-CC may include URI information, a combination of URI_character fields. The URI information may represent URI information on a server that has a trigger, and the receiver 200 or a companion device may access the server and obtain URL of a TDO to have to be received. Moreover, if a companion device requests a server to provide the URL of the TDO, the server may transmit a TDO suitable for the companion device. The companion device may request a TDO internet server with the Trigger_id, and the server may transmit a proper TDO according to the requested value.

[001192] By using such broadcast service transmission and reception methods, a service provider may efficiently provide an adjunct service for an external device linked to a broadcast service and it is possible to enhance the convenience of users who use the adjunct service linked to the broadcast service through the external device.

[001193] A method of receiving an adjunct service according to another embodiment will be described below.

[001194] The adjunct service may be service information, metadata, adjunct data, a compiled execution file, a web application, a Hypertext Markup Language (HTML) document, an XML document, a cascading style sheet (CSS) document, an audio file, a video file, an address such as Uniform Resource Locator (URL). Moreover, the adjunct service may be signaling data, link information, a trigger, an SMT, an NRT-IT, an EIT, a VCT, and a PMT that have been described above.

[001195] As a digital broadcast is activated, a

broadcasting station sends main audio/visual content along with enhanced service data that may be used in association with the main audio/visual content or that is independent of the main audio/visual content.

[001196] However, in the current broadcasting environment, there are little cases where each home's image display device directly receives a broadcast signal through airwaves. Rather, since each home's image display device is connected to a broadcast receiving device such as a set top box, there are many cases where uncompressed audio/visual content provided by the broadcasting receiving device is played.

[001197] On the other hand, the broadcast receiving device receives content from a server referred to as a multichannel video programming distributor (MVPD). The MVPD receives a broadcast signal from a broadcasting station, extracts content from the received broadcast signal, converts the content into a signal of a type suitable for sending and provides the converted signal to the broadcast receiving device. In this process, since the MVPD may exclude the extracted enhanced service data or add another enhanced service data, the broadcasting station may perform only the function of providing audio/visual content and not provide a broadcasting station-directed adjunct service.

[001198] Since the broadcast receiving device also extracts main AV data from the signal received the MVPD and provides only uncompressed AV data to the image display device, it is just possible to use an adjunct service under the control of the broadcast receiving device and it is not possible to provide an adjunct service under the control of the image display device. It disables the broadcasting

station to provide a broadcasting station-directed adjunct service.

[001199] In addition, even if the broadcasting station provides an adjunct service through a wireless channel, it is not possible to sufficiently provide an adjunct service due to the limit of a bandwidth.

[001200] Thus, an image display device for enabling a broadcasting station-directed adjunct service and a control method thereof will be described below.

[001201] A network topology according to an embodiment will be described below with reference to Figs. 103 to 111.

[001202] Fig. 103 is a block diagram of a network topology according to an embodiment.

[001203] As shown in Fig. 103, the network topology according to an embodiment includes a content providing server 10, a content recognition service providing server 20, a MVPD server 30, an adjunct service information providing server 40, a plurality of adjunct service providing servers 50, a set top box 60, a network 70, and a receiver 300.

[001204] The content providing server 10 may correspond to a broadcasting station and broadcasts a broadcast signal including main AV content. The broadcast signal may further include an adjunct service. The adjunct service may be related to the main AV content or not. There may be one or more content providing servers.

[001205] The content recognition service providing server 20 provides a content recognition service that allows the receiver 300 to recognize content based on the main AV content. The content recognition service providing server 20 may modify the main AV content or not. There may be one or

more content recognition service providing servers.

[001206] The content recognition service providing server 20 may be a watermark server that modifies the main AV content and inserts a visible watermark such as a logo into the main AV content. The watermark server may watermark a content provider's logo on the left or right upper end of each frame of the main AV content

[001207] Moreover, the content recognition service providing server 20 may be a watermark server that modifies the main AV content and inserts content information as an invisible watermark into the main AV content.

[001208] Moreover, the content recognition service providing server 20 may be a fingerprint server that extracts and stores characteristic information from some frames or some audio samples of the main AV content. The characteristic information is referred to as a signature.

[001209] The MVPD server 30 receives and muliplexes broadcast signals from a plurality of broadcast stations and provides the multiplexed signals to the set top box 60. Especially, the MVPD server 30 may perform demodulation and channel decoding on the received broadcast signals, extract main AV content and an adjunct service, perform channel encoding on the extracted main AV content and the extracted adjunct service and generate multiplexing signals for distribution. In this case, since the MVPD server 30 may exclude the extracted adjunct service or add another adjunct service, the broadcasting station may not provide a broadcasting station-directed service. There may be one or more MVPD servers.

[001210] The set top box 60 tunes a user-selected channel,

receives signals from the tuned channel, performs demodulation and channel decoding on the received signals and extracts main AV content. In addition, the set top box 60 decodes the extracted main AV content by using H.264/MPEG-4 AVC (Moving Picture Experts Group-4 advanced video coding) and Dolby AC-3, MPEG-2 AAC (Moving Picture Experts Group-2 Advanced Audio Coding) algorithms and generates uncompressed main AV content. The set top box 60 provides the generated uncompressed main AV content to the receiver 300 through the external input port of the receiver 300.

[001211] The adjunct service information providing server 40 provides adjunct service information for one or more available adjunct services related to the main AV content in response to a request from the receiver 300. The adjunct service information providing server 40 and the receiver 300 may communicate with each other through the network 70. There may be one or more adjunct service address providing servers. The adjunct service information providing server 40 may provide adjunct service information for the highest priority adjunct service among a plurality of available adjunct services.

[001212] The adjunct service providing server 50 provides one or more adjunct services available in relation to the main AV content in response to a request from the receiver 300. There may be one or more adjunct service providing servers. The adjunct service providing server 50 and the receiver 300 may communicate with each other through the network 70.

[001213] The network 70 may be an IP based network or an Advanced Television Systems Committee - Mobile/Handheld (ATSC

M/H) channel.

[001214] The receiver 300 may be a device with a display unit, such as a TV, a notebook computer, a feature phone, and a smart phone. The receiver 300 may receive uncompressed main AV content from the set top box 60 through an external input port, or receive a broadcast signal including encoded main AV content from the content providing server 10 or the MVPD server 30 through a broadcast network. The receiver 300 may receive a content recognition service from the content recognition service providing server 20 through the network 70, receive addresses for one or more adjunct services available in relation to the main AV content from the adjunct service information providing server 40 through the network 70, and receive one or more adjunct services available in relation to the main AV content from the adjunct service providing server 50 through the network 70.

[001215] Two or more of the content providing server 10, the content recognition service providing server 20, the MVPD server 30, the adjunct service information providing server 40, and the plurality of adjunct service providing servers 50 may be combined to be one server and may be operated by one company.

[001216] Fig. 104 is a block diagram of a watermark based network topology according to an embodiment.

[001217] As shown in Fig. 104, the network topology according to an embodiment further includes a watermark server 21.

[001218] The watermark server 21 as shown in Fig. 104 modifies main AV content and inserts content information into the modified main AV content. The MVPD server 30 receives

and distributes a broadcast signal including the modified main AV content. Especially, the watermark server may use a digital watermarking technology as will be described below.

[001219] Digital watermarking is a process of inserting information into a digital signal so that it is difficult to delete the information. For example, the digital signal may be a sound, a picture, or a video. If the digital signal is copied, inserted information is also contained in a copy. One digital signal may simultaneously carry several watermarks.

[001220] In the case of visible watermarking, inserted information is visible from a picture or a video. Typically, inserted information is a text or a logo to identify a media's owner. When a TV broadcasting station adds its logo to the corner of a video that is transmitted, the logo is a visible watermark.

[001221] In the case of invisible watermarking, information is added to a sound, a picture, or a video as digital data, and it is possible to detect that a certain amount of information is hidden but it is not possible to recognize such information. A secret message may be delivered through such invisible watermarking.

[001222] An application of watermarking resides in a copy right protection system for preventing illegal copying of digital media. For example, a copying device may obtain a watermark from digital media before copying the digital media, and determine based on the content of the watermark whether to copy or not.

[001223] Another application of the watermarking resides in tracing the source of the digital media. A watermark is

embedded in the digital media at each point on a distribution route. If such digital media is found later, the watermark may be extracted from the digital media and it is possible to find out a distribution source from the content of the watermark.

[001224] A description of digital media is another application of visible watermarking.

[001225] A file format for the digital media may include additional information referred to as metadata and a digital watermark is distinguished from metadata in that it is delivered as AV signal itself of the digital media.

[001226] Watermarking techniques include spread spectrum, quantization, and amplitude modulation.

[001227] If a marked signal is obtained by additional modification, the watermarking technique is spread spectrum. The spread spectrum watermark is known as quite robust but since it interferes with a host signal in which a watermark is embedded, a lot of information is not carried.

[001228] If the marked signal is obtained by quantization, the watermarking technique is a quantization type. A quantized watermark is low in robustness but may carry a lot of information.

[001229] If the marked signal is obtained by using an additional modification technique similar to the spread spectrum in a space domain, the watermarking technique is amplitude modulation.

[001230] Fig. 105 is a ladder diagram of a data flow in a watermark based network topology according to an embodiment.

[001231] Firstly, the content providing server 10 transmits a broadcast signal including an adjunct service and

main AV content in step S101A.

[001232] The watermark server 21 receives the broadcast signal provided by the content providing server 10, modifies the main AV content, inserts a visible watermark such as a log into the main AV content or watermark information as an invisible watermark into the main AV content, and provides the watermarked main AV content and the adjunct service to the MVPD 30 in step S103A.

[001233] The watermark information inserted through the invisible watermark may include one or more of a watermark purpose, content information, adjunct service information, and available adjunct service. The watermark purpose may represent one of pirate copy protection, an audience rating survey, and adjunct service obtainment.

[001234] The content information may include one or more of identification information on a content provider who provides the main AV content, identification information on the main AV content, rating information on the main AV content, time information on a content section used for obtaining content information, the name of a channel over which the main AV content is broadcast, the logo of a channel over which the main AV content is broadcast, the description of a channel over which the main AV content is broadcast, an usage information report address, a usage information report cycle, and minimum usage time for obtaining usage information.

[001235] When the receiver 300 uses a watermark to obtain content information, time information on a content section used for obtaining the content information may be time information on a content section in which the used watermark is embedded. When the receiver 300 uses a fingerprint to

obtain the content information, the time information on the content section used for obtaining the content information may be time information on a content section from which characteristic information is extracted. The time information on the content section used for obtaining the content information may include at least one of the start time of a content section used for obtaining the content information, the duration of a content section used for obtaining the content information, and the end time of a content section used for obtaining the content information.

[001236] The usage information report address may include one or more of a main AV content viewing information report address and an adjunct service usage information report address. The usage information report cycle may include one or more of a main AV content viewing information report cycle and an adjunct service usage information report cycle. The minimum usage time for obtaining usage information may include one or more of minimum viewing time for obtaining main AV content viewing information and minimum viewing time for extracting adjunct service usage information.

[001237] If the main AV content is viewed over minimum viewing time, the receiver 300 may obtain viewing information on the main AV content and report extracted viewing information to the main AV content viewing information report address on the main AV content viewing information report cycle.

[001238] If the adjunct service is used over minimum usage time, the receiver 300 may obtain adjunction service usage information and report extracted usage information to the adjunct service usage information report address on the

adjunct service usage information cycle.

[001239] The adjunct service information may include one or more of information on whether there is an adjunct service, an adjunct service address providing server address, a route for obtaining each available adjunct service, an address for each available adjunct service, the start time of each available adjunct service, the end time of each available adjunct service, the lifetime of each available adjunct service, a mode for obtaining each available adjunct service, a request cycle for each available adjunct service, priority information on each available adjunct service, a description of each available adjunct service, the category of each available adjunct service, an usage information report address, an usage information report cycle, and minimum usage time for obtaining usage information.

[001240] The route for obtaining an available adjunct service may represent an IP based network or an Advanced Television Systems Committee - Mobile/Handheld (ATSC M/H) channel. In a case where the route for obtaining the available adjunct service is the ATSC M/H, the adjunct service information may further include frequency information and channel information. The mode for obtaining each available adjunct service may represent push or pull.

[001241] On the other hand, the watermark server 21 may insert watermark information as an invisible watermark into the logo of main AV content.

[001242] For example, the watermark server 21 may insert a barcode into a certain part of a logo. In this case, the certain part of the logo may correspond to the first line of the lower end of a region on which the logo is displayed. If

receiving main AV content including a logo into which a barcode is inserted, the receiver 300 may not display the barcode.

[001243] Moreover, the watermark server 21 may insert watermark information in the form of metadata of a logo. In this case, the shape of the logo may be maintained.

[001244] Moreover, the watermark server 21 may insert N-bit watermark information to each frame of a logo with M frames. That is, the watermark server 21 may insert M*N watermark information through M frames.

[001245] The MVPD 30 receives a broadcast signal including watermarked main AV content and an adjunct service, generates multiplexed signals and provides them to the set top box 60 in step S105A. In this case, the multiplexed signal may exclude the received adjunct service or include a new adjunct service.

[001246] The set top box 60 tunes a user-selected channel, receives a signal from the tuned channel, demodulates, channel decodes and AV decodes the received broadcast signal, generates uncompressed main AV content, and provides the generated uncompressed main AV content to the receiver 300 in step S106A.

[001247] On the other hand, the content providing server 10 also broadcasts a broadcast signal including main AV content through a wireless channel in step S107A.

[001248] Moreover, the MVPD 30 may also transmit a broadcast signal including main AV content directly to the receiver 300, not through the set top box 60, in step S108A.

[001249] The receiver 300 may receive uncompressed main AV content through the set top box 60. Alternatively, the

receiver 300 may receive a broadcast signal through a wireless channel, demodulate and decodes the received broadcast signal and obtain main AV content. Alternatively, the receiver 300 may receive a broadcast signal from the MVPD 30, demodulate and decode the received broadcast signal and receive main AV content. The receiver 300 extracts watermark information from the audio samples of some sections or some frames of the obtained main AV content. If watermark information corresponds to a logo, the receiver 300 checks a watermark server address corresponding to a logo that is extracted from the corresponding relationship between a plurality of logos and a plurality of watermark server addresses. If the watermark information corresponds to the logo, the receiver 300 may not identify main AV content only with the logo. Moreover, even if the watermark information does not include content information, the receiver 300 may not identify main AV content but the watermark information may include content provider identification information or a watermark server address. If the watermark information includes the content provider identification information, the receiver 300 may check a watermark server address corresponding to content provider identification information that is extracted from the corresponding relationship between pieces of content provider identification information and a plurality of watermark server addresses. As such, if it is not possible to identify main AV content only with watermark information, the receiver 300 accesses the watermark server 21 corresponding to the obtained watermark server address and transmits a first query in step S109A.

[001250] The watermark server 21 provides a first response

to the first query in step S111A. The first response may include one or more of content information, adjunct service information, and an available adjunct service.

[001251] If the watermark information and the first response do not include an adjunct service address, the receiver 300 may not obtain an adjunct service. However, the watermark information and the first response may include an adjunct service address providing server address. As such, if the receiver 300 does not obtain the adjunct service address or the adjunct service through the watermark information and the first response but obtains the adjunct service address providing server address, the receiver 300 accesses the adjunct service information providing server 40 corresponding to the obtained adjunct service address providing server address and transmits a second query including content information in step S119A.

[001252] The adjunct service information providing server 40 searches one or more available adjunct services related to content information in the second query. Subsequently, the adjunct service information providing server 40 provides, to the receiver 300, adjunct service information for one or more available adjunct services as a second response to the second query in step S121A.

[001253] If the receiver 300 obtains one or more available adjunct service addresses through the watermark information, the first response or the second response, the receiver 300 accesses the one or more available adjunct service addresses and requests an adjunct service in step S123A and obtains the adjunct service in step S125A.

[001254] Fig. 106 shows a watermark based content

recognition timing according to an embodiment.

[001255] As shown in Fig. 106, if the set top box 60 is turned on to tune a channel and the receiver 300 receives main AV content of the channel tuned from the set top box 60 through an external input port 311, the receiver 300 may detect a content provider identifier (or a broadcasting station identifier) from the watermark of the main AV content. Subsequently, the receiver 300 may detect content information from the watermark of the main AV content based on the detected content provider identifier.

[001256] In this case, as shown in Fig. 106, the detectable cycle of the content provider identifier may be different from that of the content information. Especially, the detectable cycle of the content provider identifier may be shorter than that of the content information. Accordingly, the receiver 300 may have an efficient configuration for detecting only required information.

[001257] Fig. 107 is a block diagram of a fingerprint based network topology according to an embodiment.

[001258] As shown in Fig. 107, the network topology according to an embodiment further includes a fingerprint server 22.

[001259] The fingerprint server 22 as shown in Fig. 107 does not modify main AV content but extracts and stores characteristic information from audio samples of some sections or some frames of the main AV content. Subsequently, if the fingerprint server 22 receives the characteristic information from the receiver 300, it provides time information and identifier of AV content corresponding to the received characteristic information.

[001260] Fig. 108 is a ladder diagram of a data flow in a fingerprint based network topology according to an embodiment.

[001261] Firstly, the content providing server 10 transmits a broadcast signal including main AV content and an adjunct service in step S201A.

[001262] The fingerprint server 22 receives the broadcast signal provided by the content providing server 10, extracts pieces of characteristic information from a plurality of frame sections or a plurality of audio sections of the main AV content, and builds a database for a plurality of query results corresponding to each of pieces of characteristic information in step S203A. The query result may include one or more of content information, adjunct service information, and an available adjunct service.

[001263] The MVPD 30 receives a broadcast signal including the main AV content and the adjunct service, generates multiplexed signals and provides them to the set top box 60 in step S205A. In this case, the multiplexed signal may exclude the received adjunct service or include a new adjunct service.

[001264] The set top box 60 tunes a user-selected channel, receives a signal from the tuned channel, demodulates, channel decodes and AV decodes the received broadcast signal, generates uncompressed main AV content, and provides the generated uncompressed main AV content to the receiver 300 in step S206A.

[001265] On the other hand, the content providing server 10 also broadcasts a broadcast signal including main AV content through a wireless channel in step S207A.

[001266] Moreover, the MVPD 30 may also transmit a

broadcast signal including main AV content directly to the receiver 300, not through the set top box 60, in step S208A.

[001267] The receiver 300 may receive uncompressed main AV content through the set top box 60. Alternatively, the receiver 300 may receive a broadcast signal through a wireless channel, demodulate and decodes the received broadcast signal and obtain main AV content. Alternatively, the receiver 300 may receive a broadcast signal from the MVPD 30, demodulate and decode the received broadcast signal and receive main AV content. The receiver 300 extracts characteristic information from the audio samples of some sections or some frames of the obtained main AV content in step S213A.

[001268] The receiver 300 accesses the fingerprint server 22 corresponding to a preset fingerprint server address and transmits a first query including extracted characteristic information in step S215A.

[001269] The fingerprint server 22 provides a query result as a first response to the first query in step S217A. If the first response corresponds to failure, the receiver 300 may access the fingerprint server 22 corresponding to another fingerprint server address and transmit the first query including extracted characteristic information.

[001270] The fingerprint server 22 may provide an Extensible Markup Language (XML) document as the query result. An example of the XML document that contains the query result will be described with reference to Fig. 7 and Table 1.

[001271] Fig. 109 is an ACR-Resulttype XML schema diagram that contains a query result according to an embodiment.

[001272] As shown in Fig. 109, the ACR-Resulttype

containing the query result has the attribute ResultCode and the elements ContentID, NTPTimestamp, SignalingChannelInformation, and ServiceInformation.

[001273] For example, if the attribute ResultCode has a value of 200, it may mean that the query result is successful. If the attribute ResultCode has a value of 404, it may mean that the query result is unsuccessful.

[001274] The element SignalingChannelInformation has the element SignalingChannelURL which has the attributes UpdateMode and PollingCycle. The attribute UpdateMode may have a value of Pull or Push.

[001275] The element ServiceInformation has the elements ServiceName, ServiceLogo, and ServiceDescription.

[001276] Table 5 shows the ACR-ResultType XML Schema that contains such a query result.

[001277] [Table 5]

```
<xs:complexType name="ACR-ResultType">
    <xs:sequence>
      <xs:element name="ContentID" type="xs:anyURI"/>
      <xs:element name="NTPTimestamp" type="xs:unsignedLong"/>
      <xs:element name="SignalingChannelInformation">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="SignalingChannelURL" maxOccurs="unbounded">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:anyURI">
                    <xs:attribute name="UpdateMode">
                      <xs:simpleType>
                        <xs:restriction base="xs:string">
                          <xs:enumeration value="Pull"/>
                          <xs:enumeration value="Push"/>
                        </xs:restriction>
                      </xs:simpleType>
                    </xs:attribute>
                    <xs:attribute name="PollingCycle" type="xs:unsignedInt"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="ServiceInformation">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ServiceName" type="xs:string"/>
            <xs:element name="ServiceLogo" type="xs:anyURI" minOccurs="0"/>
            <xs:element name="ServiceDescription" type="xs:string"
minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:any namespace="##other" processContents="skip" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="ResultCode" type="xs:string" use="required"/>
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
```

[001278] As the element ContentID, an ATSC content identifier as shown in Table 2 below may be used.

[001279] [Table 6]

Syntax	The Number of bits	format
ATSC_content_identifier() {		
TSID	16	uimsbf
reserved	2	bslbf
end_of_day	5	uimsbf
unique_for	9	uimsbf
content_id	var	
}		

[001280] As shown in Table 6, the ATSC content identifier has a structure that includes TSID and house numbers.

[001281] The 16-bit unsigned integer TSID carries a transport stream identifier.

[001282] The 5-bit unsigned integer end_of_day is set as the hour of a day when a content_id value may be reused after a broadcast ends.

[001283] The 9-bit unsigned integer unique_for is set as the number of days when the content_id value may not be reused.

[001284] The content_id represents a content identifier. The receiver 300 may decrease the unique_for everyday by 1 from the hour corresponding to the end_of_day, and if the unique_for does not become zero, it may be considered that the content_id is unique.

[001285] On the other hand, as the element ContentID, a global service identifier for an ATSC-M/H service as described below may be used.

[001286] The global service identifier has the following form.

[001287] -

urn:oma:bcast:iauth:atsc:service:<region>:<xsid>:<serviceid>

[001288] Here, <region> is an international country code including two characters as regulated by ISO 639-2. <xsid> for a local service is a decimal number for TSID as defined

in <region>, and <xsid> for a regional service (major > 69) is "zero". <serviceid> is defined as <major> or <minor>. <major> represents a major channel number and <minor> represents a minor channel number.

[001289] An example of the global service identifier is as follows.

[001290] - urn:oma:bcast:iauth:atsc:service:us:1234:5.1

[001291] - urn:oma:bcast:iauth:atsc:service:us:0:100.200

[001292] On the other hand, as the element ContentID, an ATSC content identifier as described below may be used.

[001293] The ATSC content identifier has the following form.

[001294] urn:oma:bcast:iauth:atsc:content:<region>:<xsidz>:<contentid>:<unique_for>:<end_of_day>

[001295] Here, <region> is an international country code including two characters as regulated by ISO 639-2. <xsid> for a local service is a decimal number for TSID as defined in <region> and may be followed by <serviceid>. <xsid> for a regional service (major > 69) is <serviceid>. <content_id> is a base64 sign for the content_id field defined in Table 2, <unique_for> is a decimal number sign for the unique_for field defined in Table 2, and <end_of_day> is a decimal number sign for the end_of_day field defined in Table 2.

[001296] Referring back to Fig. 108, descriptions will be made below.

[001297] If the query result does not include an adjunct service address or an adjunct service but includes an adjunct service address providing server address, the receiver 300 accesses the adjunct service information providing server 40 corresponding to the obtained adjunct service address

providing server address and transmits a second query including content information in step S219A.

[001298] The adjunct service information providing server 40 searches one or more available adjunct services related to the content information on the second query. Subsequently, the adjunct service information providing server 40 provides, to the receiver 300, adjunct service information for one or more available adjunct services as a second response to the second query in step S221A.

[001299] If the receiver 300 obtains one or more available adjunct service addresses through the first response or the second response, it accesses the one or more available adjunct service addresses, requests adjunct services in step S223A, and obtains them in step S225A.

[001300] If the attribute UpdateMode has a value of Pull, the receiver 300 transmits an HTTP request to the adjunct service providing server 50 through SignalingChannelURL, and receives an HTTP response including a PSIP binary stream from the adjunct service providing server 50 in response to that. In this case, the receiver 300 may transmit the HTTP request according to a polling cycle designated by the attribute PolyingCycle. Moreover, the element SignalingChannelURL may also have an update time attribute. In this case, the receiver 300 may transmit the HTTP request at an update time designated by the update time attribute.

[001301] If the attribute UpdateMode has a value of Push, the receiver 300 may asynchronously receive an update from a server by utilizing XMLHTTPRequest API. In this case, if there is a change in signaling information after the receiver 300 makes an asynchronous request to the server through

XMLHTTPRequest object, the server 300 provides the signaling information as a response through a channel. If there is a limit on the standby time of a session, a session timeout response is generated and the receiver immediately recognizes it and makes a request again so that it is possible to always maintain a signaling channel between the receiver and the server.

[001302] Fig. 110 is a block diagram of a watermark and fingerprint based network topology according to an embodiment.

[001303] As shown in Fig. 110, the network topology according to an embodiment further includes a watermark server 21 and a fingerprint server 22.

[001304] The watermark server 21 as shown in Fig. 110 inserts content provider identification information into main AV content. The watermark server 21 may insert the content provider identification information as a visible watermark such as a logo, into the main AV content or may insert the content provider identification information as an invisible watermark into the main AV content.

[001305] The fingerprint server 22 does not modify the main AV content does not modify main AV content but extracts and stores characteristic information from audio samples of some sections or some frames of the main AV content. Subsequently, if the fingerprint server 22 receives the characteristic information from the receiver 300, it provides time information and identifier of AV content corresponding to the received characteristic information.

[001306] Fig. 111 is a ladder diagram of a data flow in a watermark and fingerprint based network topology according to an embodiment.

[001307] Firstly, the content providing server 10 transmits a broadcast signal including main AV content and an adjunct service in step S301A.

[001308] The watermark server 21 receives the broadcast signal provided by the content providing server 10, modifies the main content, inserts a visible watermark such as a logo into the main AV content or inserts watermark information an invisible watermark into the main AV content, and provides the watermarked main AV content and the adjunct service to the MVPD 30 in step S303A. The watermark information inserted through the invisible watermark may include one or more of content information, adjunct service information, and an available adjunct service. The content information and the adjunct service information have been described above and thus their descriptions are skipped.

[001309] The MVPD 30 receives a broadcast signal including the watermarked main AV content and the adjunct service, generates multiplexed signals and provides them to the set top box 60 in step S305A. In this case, the multiplexed signal may exclude the received adjunct service or include a new adjunct service.

[001310] The set top box 60 tunes a user-selected channel, receives a signal from the tuned channel, demodulates, channel decodes and AV decodes the received broadcast signal, generates uncompressed main AV content, and provides the generated uncompressed main AV content to the receiver 300 in step S306A.

[001311] On the other hand, the content providing server 10 also broadcasts a broadcast signal including main AV content through a wireless channel in step S307A.

[001312] Moreover, the MVPD 30 may also transmit a broadcast signal including main AV content directly to the receiver 300, not through the set top box 60, in step S308A.

[001313] The receiver 300 may receive uncompressed main AV content through the set top box 60. Alternatively, the receiver 300 may receive a broadcast signal through a wireless channel, demodulate and decodes the received broadcast signal and obtain main AV content. Alternatively, the receiver 300 may receive a broadcast signal from the MVPD 30, demodulate and decode the received broadcast signal and receive main AV content. The receiver 300 extracts watermark information from the audio samples of some sections or some frames of the obtained main AV content. If watermark information corresponds to a logo, the receiver 300 checks a watermark server address corresponding to a logo that is extracted from the corresponding relationship between a plurality of logos and a plurality of watermark server addresses. If the watermark information corresponds to the logo, the receiver 300 may not identify main AV content only with the logo. Moreover, even if the watermark information does not include content information, the receiver 300 may not identify main AV content but the watermark information may include content provider identification information or a watermark server address. If the watermark information includes the content provider identification information, the receiver 300 may check a watermark server address corresponding to content provider identification information that is extracted from the corresponding relationship between pieces of content provider identification information and a plurality of watermark server addresses. As such, if it is

not possible to identify main AV content only with watermark information, the receiver 300 accesses the watermark server 21 corresponding to the obtained watermark server address and transmits a first query in step S309A.

[001314] The watermark server 21 provides a first response to the first query in step S311A. The first response may include one or more of a fingerprint server address, content information, adjunct service information, and an available adjunct service. The content information and the adjunct service information have been described above, so their descriptions are skipped.

[001315] If the watermark information and the first response include a fingerprint server address, the receiver 300 extracts characteristic information from the audio samples of some sections or some frames of main AV content in step S313A.

[001316] The receiver 300 accesses the fingerprint server 22 corresponding to a fingerprint server address in the first response and transmits a second query including the extracted characteristic information in step S315A.

[001317] The fingerprint server 22 provides a query result as a second response to the second query in step S317A.

[001318] If the query result does not include an adjunct service address or an adjunct service but includes an adjunct service address providing server address, the receiver 300 accesses the adjunct service information providing server 40 corresponding to the obtained adjunct service address providing server address and transmits a third query including content information in step S319A.

[001319] The adjunct service information providing server

40 searches one or more available adjunct services related to the content information on the third query. Subsequently, the adjunct service information providing server 40 provides, to the receiver 300, adjunct service information for one or more available adjunct services as a third response to the third query in step S321A.

[001320] If the receiver 300 obtains one or more available adjunct service addresses through one of the first to third responses, it accesses the one or more available adjunct service addresses, requests adjunct services in step S323A, and obtains them in step S325A.

[001321] The receiver 300 according to an embodiment will be described below with reference to Fig. 112.

[001322] As shown in Fig. 112, the receiver 300 according to an embodiment includes a broadcast signal receiving unit 301, a demodulation unit 310, a channel decoding unit 305, a demultiplexing unit 320, an AV decoding unit 309, an external input port 311, a play control unit 313, a player 320, a service manager 350, a data transceiving unit 341, and a memory 350.

[001323] The broadcast signal receiving unit 301 receives a broadcast signal from the content providing server 10 or the MVPD 30.

[001324] The demodulation unit 310 demodulates the received broadcast signal and generates a demodulated signal.

[001325] The channel decoding unit 305 channel decodes the demodulated signal and generates channel decoded data.

[001326] The demultiplexing unit 320 separates main AV content and an adjunct service from the channel decoded data. The separated adjunct service is stored in an adjunct service

storage unit 352.

[001327] The AV decoding unit 309 AV decodes the separated main AV content and generates uncompressed main AV content.

[001328] On the other hand, the external input port 311 receives uncompressed main AV content from the set top box 60, a digital versatile disk (DVD) player, and a Blue-ray disc player. The external input port 311 may include one or more of a DSUB port, a High Definition Multimedia Interface (HDMI) port, a Digital Visual Interface (DVI) port, a composite port, a component port, and an S-Video port.

[001329] The play control unit 313 plays at least one of uncompressed main AV content generated by the AV decoding unit 309 and uncompressed main AV content received from the external input port 311, by user selection.

[001330] The player 320 includes a display unit 321 and a speaker 323. The display unit 321 may include at least one of a liquid crystal display (LCD), a thin film transistor-liquid crystal display (TFT LCD), an organic light-emitting diode (OLED), a flexible display, and a 3D display.

[001331] The service manager 350 obtains content information on main AV content and obtains an available adjunct service based on the obtained content information. Especially, the service manager 350 may obtain identification information on main AV content based on the audio samples of some sections or some frames of uncompressed main AV content as described above, which is also referred to as automatic contents recognition (ACR) in the specification.

[001332] The service manager 350 may correspond to the service manager 350 of Fig. 34.

[001333] The data transceiving unit 341 may include an

Advanced Television Systems Committee - Mobile/Handheld (ATSC-M/H) channel transceiving unit 341a and an IP transceiving unit 341b.

[001334] The ATSC-M/H channel transceiving unit 341a communicates with other devices or servers through an ATSC-M/H channel.

[001335] The IP transceiving unit 341b communicates with other devices or servers through an IP based network.

[001336] The memory 350 may include at least one type of storage mediums including a flash memory type, a hard disk type, a multimedia card micro type, a card-type memory (for example, an SD or XD memory, etc.), a Random Access Memory (RAM), a Static RandomAccess Memory (SRAM), a Read-Only Memory (ROM), an Electrically Erasable Programmable Read-Only Memory (EEPROM), a Programmable Read-Only memory (PROM), a magnetic memory, a magnetic disk, and an optical disk. The receiver 300 may operate as a web storage that performs the storage function of the memory 350 on the internet.

[001337] The memory 350 may include a content information storage unit 351, an adjunct service storage unit 352, a logo storage unit 353, a configuration information storage unit 354, a bookmark storage unit 355, a user information storage unit 356, and a usage information storage unit 357.

[001338] The content information storage unit 351 stores pieces of content information corresponding to pieces of characteristic information.

[001339] The adjunct service storage unit 352 may store a plurality of adjunct services corresponding to the pieces of characteristic information or store a plurality of adjunct services corresponding to the pieces of content information.

[001340] The logo storage unit 353 stores a plurality of logos. Moreover, the logo storage unit may further store content providing identifiers corresponding to the plurality of logos and watermark server addresses corresponding to the plurality of logos.

[001341] The configuration information storage unit 354 stores configuration information for ACR.

[001342] The bookmark storage unit 355 stores bookmarks.

[001343] The user information storage unit 356 stores user information. The user information may include one or more of one or more pieces of account information for one or more services, regional information, family member information, favorite genre information, image display device information, and a usage information providing range. The account information may include account information for a usage information measuring server, and account information for a social network service such as twitter and facebook. The regional information may include address information and a zip code. The family member information may include the number of family members, and each member's age, sex, religion and job. The favorite genre information may be set as one or more of sports, a movie, a drama, education, news, entertainment and other genres. The image display device information may include information on the type, manufacturer, firmware version, resolution, model name, OS and browser of the image display device, the presence/absence and capacity of a storage device, and a network speed. If the usage information providing range is set, the receiver 300 may collect and report main AV content viewing information and adjunct service usage information within the set range. The

usage information providing range may be set for each of virtual channels. Moreover, a usage information measuring tolerance range may be set for all of physical channels.

[001344] The usage information storage unit 357 stores main AV content viewing information and adjunct service usage information that are collected by the receiver 300. Moreover, the receiver 300 may analyze a service usage pattern based on the collected main AV content viewing information and the collected adjunct service usage information and store the analyzed service usage pattern in the usage information storage unit 357.

[001345] The service manager 350 may obtain content information on main AV content from the fingerprint server 22 or the content information storage unit 351. If the content information storage unit 351 does not include any or enough content information corresponding to the extracted characteristic information, the service manager 350 may receive additional content information through the data transceiving unit 341. Moreover, the service manager 350 may continue to update content information.

[001346] The service manager 350 may an available adjunct service from the adjunct service providing server 50 or the adjunct service storage unit 353. If the adjunct service storage unit 353 does not include any or enough adjunct services, the service manager 350 may update adjunct services through the data transceiving unit 341. Moreover, the service manager 350 may continue to update the adjunct services.

[001347] The service manager 350 may extract a logo from main AV content, query the logo storage unit 355 and obtain a

watermark server address or a content provider identifier corresponding to the extracted logo. If the logo storage unit 355 does not include any or enough logos that match the extracted logo, the service manager 350 may receive an additional logo through the data transceiving unit 341. Moreover, the service manager 350 may continue to update logos.

[001348] The service manager 350 may perform various techniques for decreasing calculation load when comparing the logo extracted from main AV content with a plurality of logos in the logo storage unit 355.

[001349] For example, the service manager 350 may perform a comparison based on color characteristics. That is, the service manager 350 may compare the color characteristic of the extracted logo with that of the logo in the logo storage unit 355 and determine whether they match each other.

[001350] Moreover, the service manager 350 may perform a comparison based on character recognition. That is, the service manager 350 may compare characters recognized from the extracted logo with those recognized from the logo in the logo storage unit 355 and determine whether they match each other.

[001351] In addition, the service manager 350 may perform a comparison based on the contour shape of a logo. That is, the service manager 350 may compare the contour shape of the extracted logo with that of the logo in the logo storage unit 355 and determine whether they match each other.

[001352] The service manager 350 may obtain content information on main AV content based on the audio samples of some sections or some frames of uncompressed main AV content,

obtain signaling data for an adjunct service based on the obtained content information and then obtain the adjunct service based on the signaling data.

[001353] A method of transmitting and receiving a broadcast service according to an embodiment may be manufactured as a program to be executed by a computer and may be stored in the computer readable recording medium that includes read-only memory (ROM), random-access memory (RAM), CD-ROMs, magnetic tapes, floppy disks, optical data storage devices, and carrier waves (such as data transmission through the Internet).

[001354] The computer readable recording medium can also be distributed over network coupled computer systems so that the computer readable code is stored and executed in a distributed fashion. Moreover, functional programs, codes, and code segments for accomplishing the present invention can be easily construed by programmers skilled in the art to which the present invention pertains.

[001355] In addition, although exemplary embodiments have been illustrated and described above, the present disclosure is not limited specific embodiments described above but may be varied by those skilled in the art without departing from the subject matter of the present disclosure claimed in the following claims. Further, these variations should not be understood individually from the technical spirit or perspective of the present disclosure.

[001356]

50514-84

CLAIMS:

1. A method for a broadcast receiving device to receive a broadcast, comprising:

receiving uncompressed content from a broadcaster;

extracting characteristic information from the uncompressed content, wherein the characteristic information includes location information on a server providing a trigger;

receiving the trigger from the server, by using the location information,

wherein the trigger includes an identifier for identifying a triggered declarative object (TDO), a trigger time indicating a timing for transmitting a command and a trigger action indicating a state of the TDO;

receiving TDO information on the TDO for a second screen device;

generating the command for the TDO based on the trigger and the TDO information, wherein the command only includes the trigger action; and

transmitting the command for the TDO to the second screen device at a specific time designated by the trigger time,

wherein the second screen device acts on the command for the TDO.

2. The method of claim 1, wherein the TDO information includes a target device type for the TDO, and wherein a value

50514-84

of the target device type represents one of the broadcast receiving device, the second screen device and both.

3. The method of claim 2, wherein the command for the TDO includes the target device type for the TDO.

4. The method of claim 1, further comprising:

receiving a capabilities descriptor which provides a list of capabilities used for the TDO,

wherein the generating the command for the TDO based on the trigger and the TDO information comprises:

generating the command for the TDO based on the capabilities descriptor,

wherein a value of capability category code in the capabilities descriptor represents one of download protocol, FEC algorithm, wrapper/archive formats, compression algorithms, and media type.

5. The method of claim 1, wherein the TDO information includes a uniform resource identifier (URL) which is used for downloading a file which is part of the TDO, and wherein the command for the TDO includes the URL.

6. The method of claim 1, further comprising:

receiving a subscription request from the second screen device,

wherein the transmitting the command for the TDO to the second screen device comprises:

50514-84

transmitting the command for the TDO based on the subscription request.

7. An apparatus for receiving a broadcast service, comprising:

a receiving unit for receiving uncompressed content from a broadcaster including characteristic information, for receiving a trigger from a server by using location information included in the characteristic information, and for receiving TDO information on the TDO for a second screen device, wherein the trigger includes an identifier for identifying a triggered declarative object (TDO), a trigger time indicating a timing for activating the TDO, and a trigger action indicating a state of the TDO;

a service manager for generating a command for the TDO based on the trigger and the TDO information, wherein the command only includes the trigger action; and

a network interface for transmitting the command for the TDO to the second screen device at a specific time designated by the trigger time,

wherein the second screen device is to act on the command for the TDO.

8. The apparatus of claim 7, wherein the TDO information includes a target device type for the TDO, and wherein a value of the target device type represents one of a broadcast receiving device, the second screen device and both.

50514-84

9. The apparatus of claim 7, wherein the receiving unit is for receiving a capabilities descriptor which provides a list of capabilities used for the TDO,

wherein the service manager is for generating the command for the TDO based on the capabilities descriptor,

wherein a value of capability category code in the capabilities descriptor represents one of download protocol, FEC algorithm, wrapper/archive formats, compression algorithms, and media type.

10. The apparatus of claim 7, wherein the TDO information includes a uniform resource identifier (URL) which is used for downloading a file which is part of the TDO, and wherein the command for the TDO includes the URL.

11. The apparatus of claim 7, wherein the receiving unit is for receiving a subscription request from the second screen device, and wherein the network interface is for transmiting the command for the TDO based on the subscription request.

FIG.1

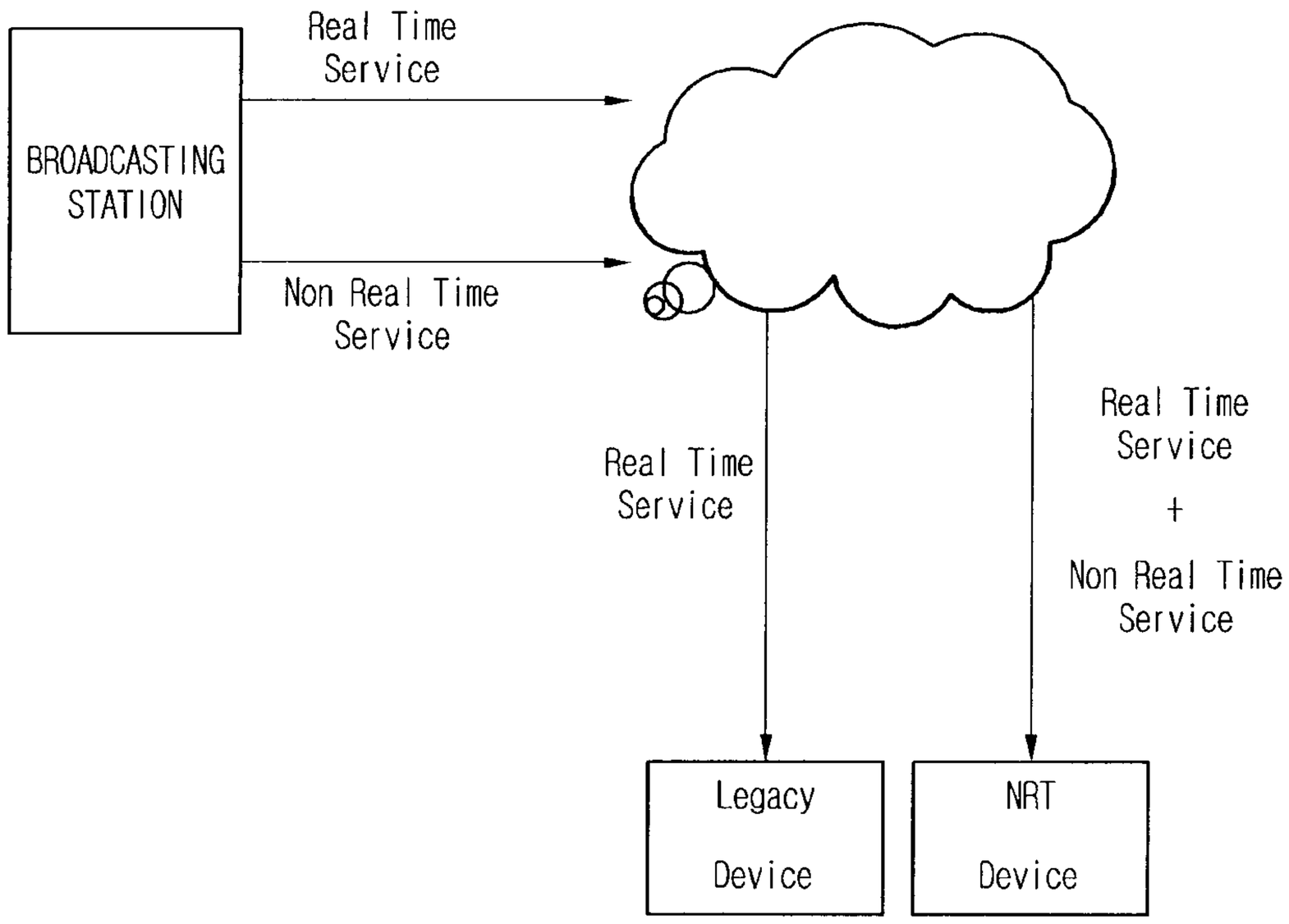

BROADCASTING STATION
Real Time Service
Non Real Time Service
Real Time Service
Real Time Service + Non Real Time Service
Legacy Device
NRT Device

FIG.2

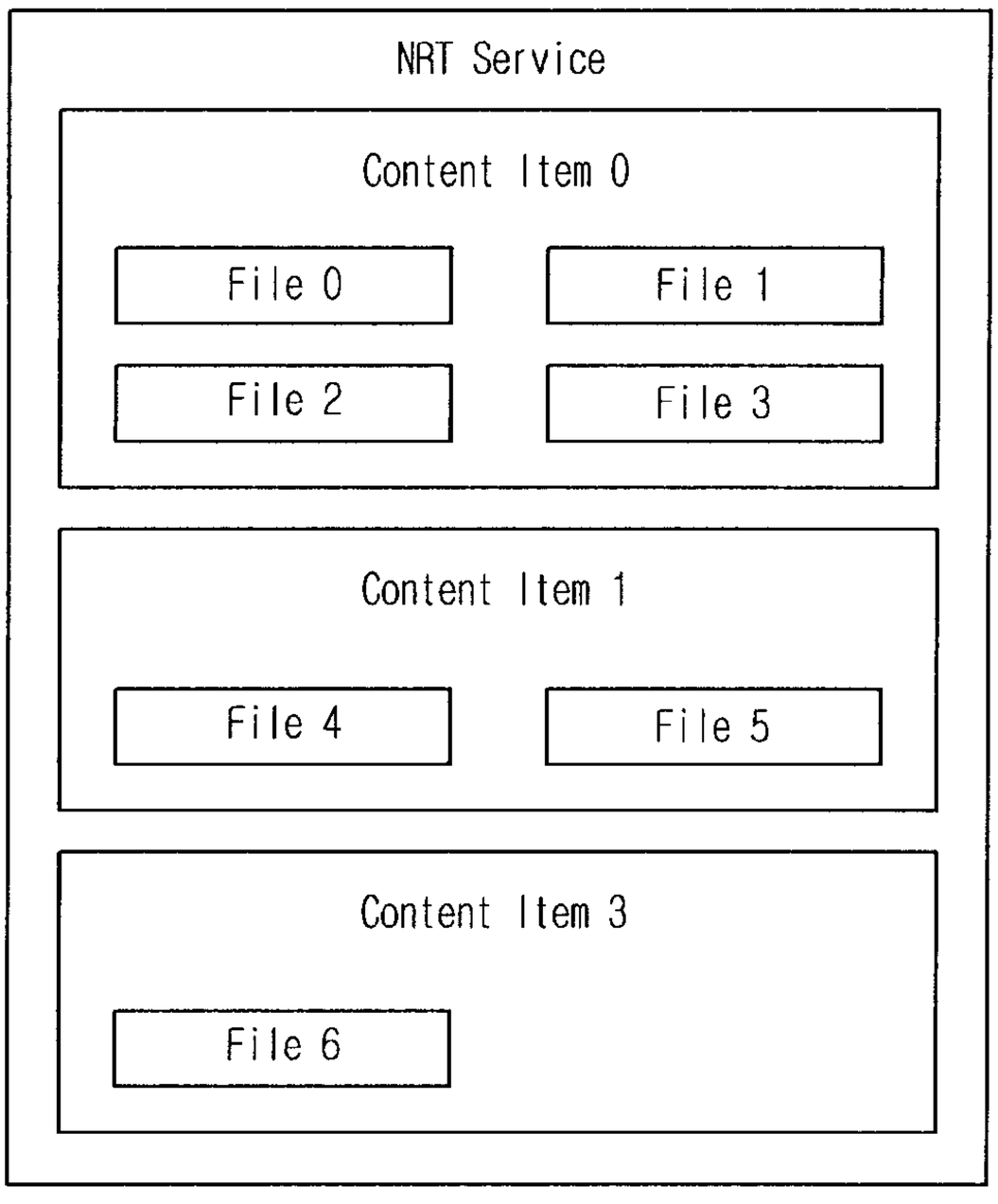

NRT Service
Content Item 0
File 0
File 1
File 2
File 3
Content Item 1
File 4
File 5
Content Item 3
File 6

FIG.3

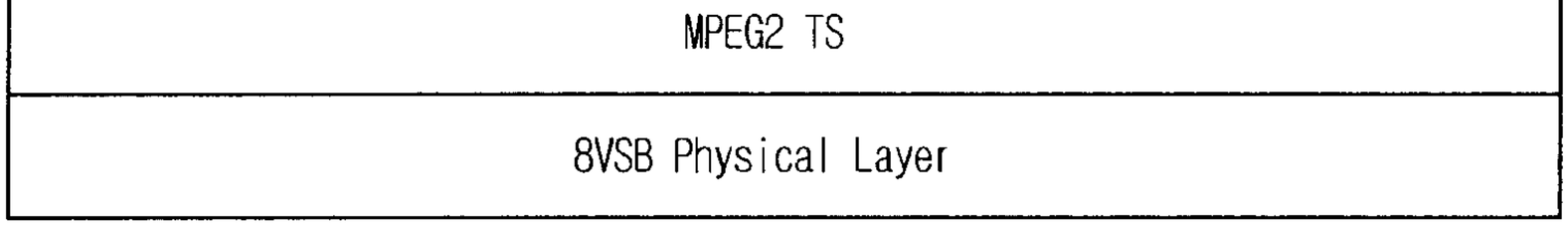

NRT Content Items / Files
FLUTE
ALC / LCT
UDP
IP layer signaling-SMT + NRT-IT
IP
Service protection-OMA BCAST DRM
PSI/PSIP
DSMCC Addressable Sections
MPEG2 TS
8VSB Physical Layer

FIG.4

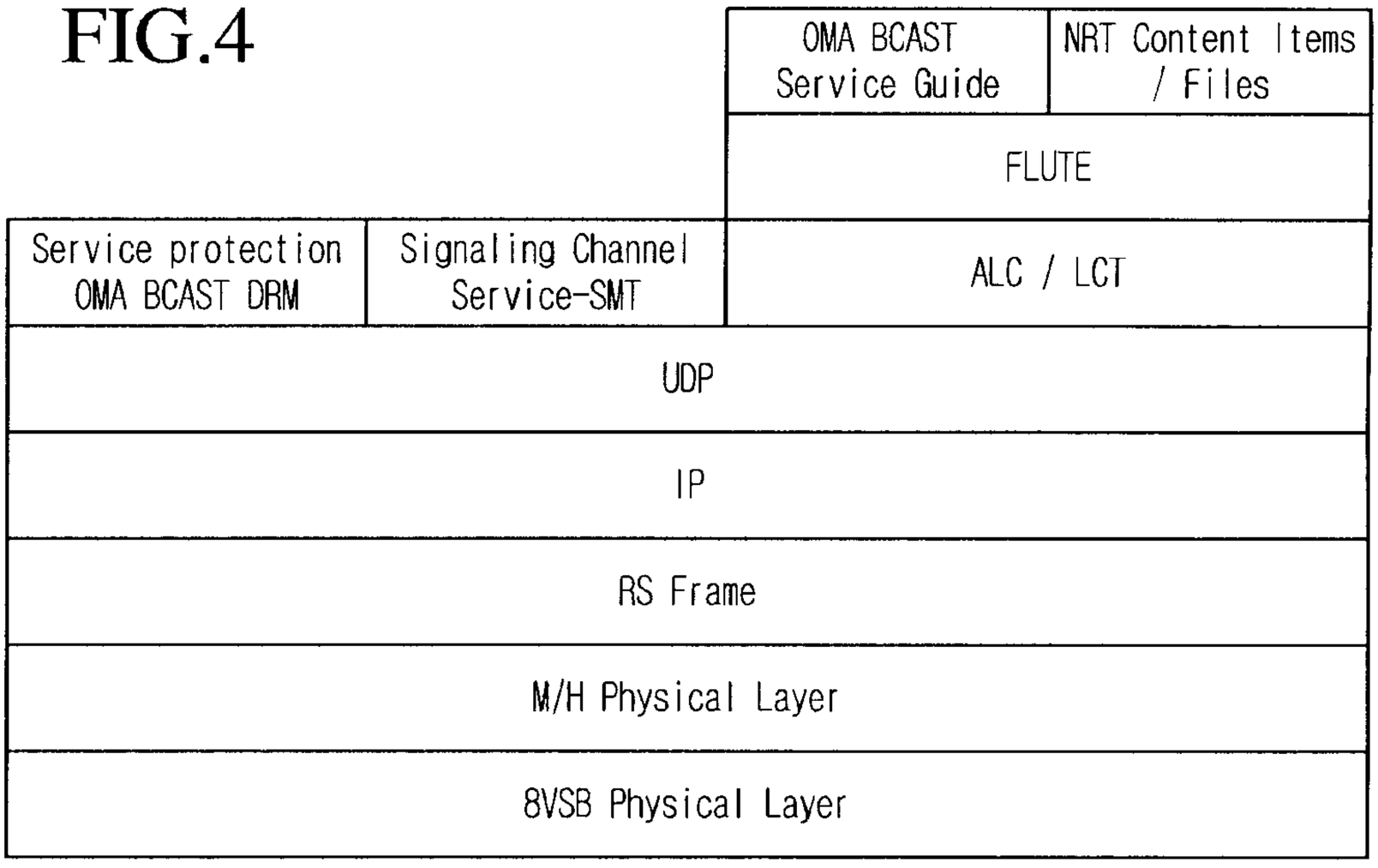

OMA BCAST Service Guide
NRT Content Items / Files
FLUTE
Service protection OMA BCAST DRM
Signaling Channel Service-SMT
ALC / LCT
UDP
IP
RS Frame
M/H Physical Layer
8VSB Physical Layer

FIG.5

Syntax	No. of Bits	Format
virtual_channel_table_section(){		
table_id	8	0xC8
section_syntax_indicator	1	'1'
Private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
transport_stream_id	16	uimsbf
reserved	2	'11'
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
protocol_version	8	uimsbf
num_channels_in_section	8	uimsbf
for(i=0;i<num_channels_in_section;i++){		
short_name	7*16	uimsbf
reserved	4	'1111'
major_channel_number	10	uimsbf
minor_channel_number	10	uimsbf
modulation_mode	8	uimsbf
carrier_frequency	32	uimsbf
channel_TSID	16	uimsbf
program_number	16	uimsbf
ETM_location	2	uimsbf
access_controlled	1	bslbf
hidden	1	bslbf
reserved	2	'11'
hide_guide	1	bslbf
reserved	3	'111'
service_type	6	uimsbf
source_id	16	uimsbf
reserved	6	'111111'
descriptors_length	10	uimsbf
for)i=0;i<N;i++){		
descriptor()		
}		
}		
reserved	6	'111111'
additional_descriptors_length	10	uimsbf
for(j=0;j<N;j++){		
additional_descriptor()		
}		
CRC_32	32	rpchof
}		

FIG.6

service_type	Meaning
0x00	[Reserved]
0x01	analog_television - The virtual channel carries analog television programming
0x02	ATSC_digital_television - The virtual channel carries television programming (audio, video and optional associated data) conforming to ATSC standards
0x03	ATSC_audio - The virtual channel carries audio programming (audio service and optional associated data) conforming to ATSC standard s.
0x04	ATSC_data_only_service - The virtual channel carries a data service conforming to ATSC standards, but no video of stream_type 0x02 or audio of stream_type 0x81.
0x05-0x3F	[Reserved for future ATSC use]

FIG.7

Value	Meaning
0x00	[Reserved]
0x01	Analog_television-The virtual channel carries analog television programming
0x02	ATSC_digital_television-The virtual channel carries television programming (audio, video and optional data) conforming to ATSC standards.
0x03	ATSC_audio-The virtual channel carries audio programming (audio service and optional data) conforming to ATSC standards.
0x04	ATSC_data_only_service-The virtual channel carries a data service conforming to ATSC standards, but no video of stream_type 0x02 or audio of stream_type 0x81.
0x05	Software Download Data Service see A/97
0x06	Unassociated/small screen service see A/65C Amendment 1
0x07	Parameterized Service New A/V CODEC
0x08	ATSC_nrt_service-The virtual channel carries a NRT service conforming to ATSC standards.
0x09-0x7F	[Reserved for future]
0x80-0xFF	[User Private]

FIG.8

Syntax	No. of bits	Format
data_service_table_section() {		
table_id	8	0xCF
section_syntax_indicator	1	bslbf
private_indicator	1	bslbf
reserved	2	'11'
private_section_length	12	uimsbf
table_id_extension	16	uimsbf
reserved	2	'11'
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
data_service_table_bytes()		
CRC_32	32	rpchof
}		
Syntax	No. of bits	Format
data_service_table_bytes() {		
sdf_protocol_version	8	uimsbf
application_count_in_section	8	uimsbf
if(application_count_in_section > 0) {		
for(j = 0; j <application_count_in_section; j++) {		
compatibility_descriptor()		
app_id_byte_length	16	uimsbf
if(app_id_byte_length > 1) {		
app_id_description	16	uimsbf
for(i=0;i< app_id_byte_length-2;i++) {		
app_id_byte	8	bslbf
}		
}		
tap_count	8	uimsbf
for(i = 0; i < tap_count; i++) {		
protocol_encapsulation	8	uimsbf
action_type	7	uimsbf
resource_location	1	bslbf
Tap()		
tap_info_length	16	uimsbf
for(k=0; k<N; k++) {		
descriptor()		
}		
}		
app_info_length	16	uimsbf
for(i=0; i< M; i++) {		
descriptor()		
}		
app_data_length	16	uimsbf
for(i = 0; i < app_data_length; i++) {		
app_data_byte	8	bslbf
}		
}		
}		
service_info_length	16	uimsbf
for(j=0; j<K;j++) {		
descriptor()		
}		
service_private_data_length	16	uimsbf
for(j = 0; j < service_private_data_length; j++) {		
service_private_data_byte	8	bslbf
}		
}		

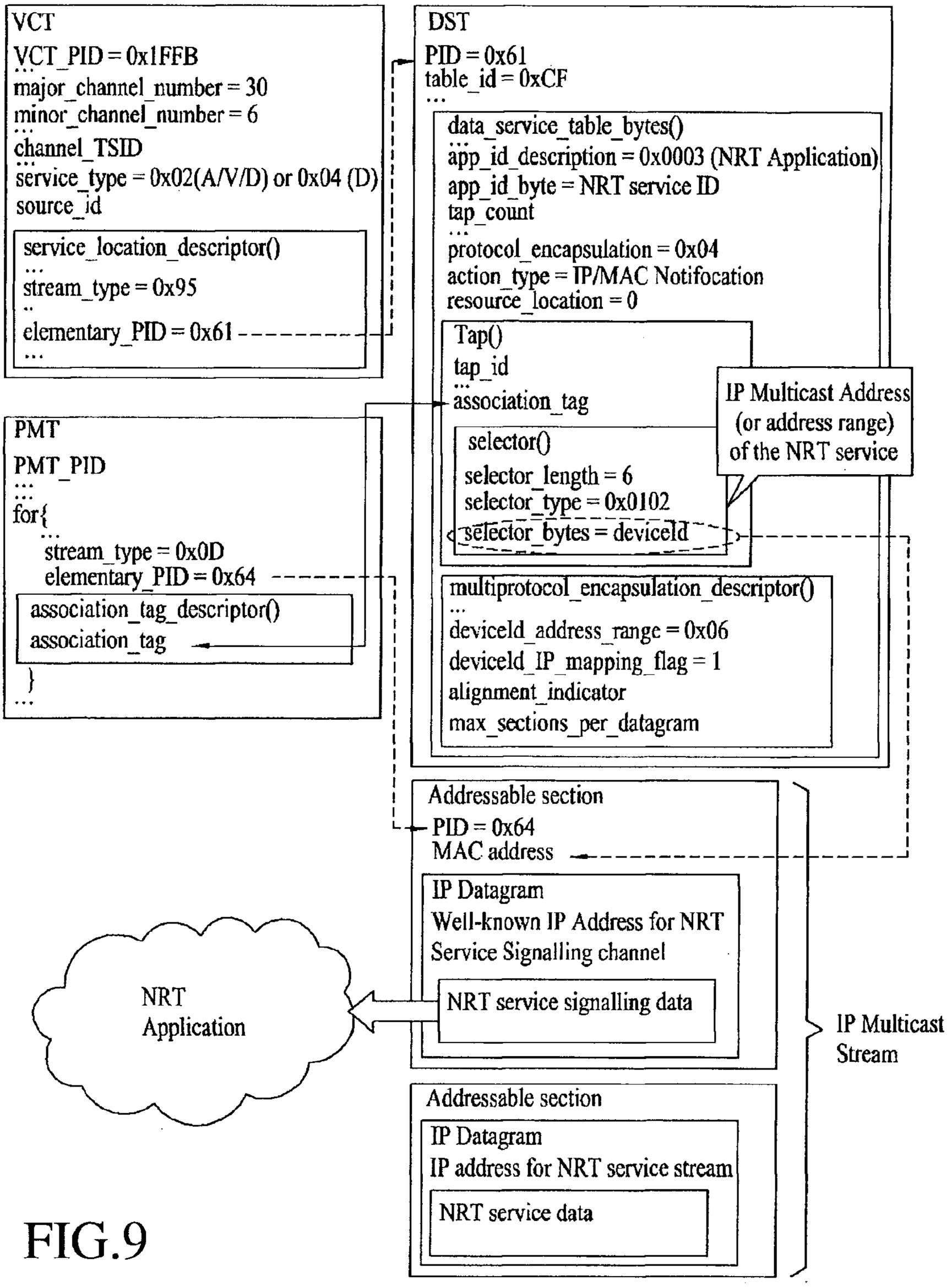

VCT
VCT_PID = 0x1FFB
major_channel_number = 30
minor_channel_number = 6
channel_TSID
service_type = 0x02(A/V/D) or 0x04 (D)
source_id
service_location_descriptor()
stream_type = 0x95
elementary_PID = 0x61
DST
PID = 0x61
table_id = 0xCF
data_service_table_bytes()
app_id_description = 0x0003 (NRT Application)
app_id_byte = NRT service ID
tap_count
protocol_encapsulation = 0x04
action_type = IP/MAC Notifocation
resource_location = 0
Tap()
tap_id
association_tag
selector()
selector_length = 6
selector_type = 0x0102
selector_bytes = deviceId
IP Multicast Address (or address range) of the NRT service
multiprotocol_encapsulation_descriptor()
deviceId_address_range = 0x06
deviceId_IP_mapping_flag = 1
alignment_indicator
max_sections_per_datagram
PMT
PMT_PID
for{
stream_type = 0x0D
elementary_PID = 0x64
association_tag_descriptor()
association_tag
}
Addressable section
PID = 0x64
MAC address
IP Datagram
Well-known IP Address for NRT Service Signalling channel
NRT service signalling data
NRT Application
Addressable section
IP Datagram
IP address for NRT service stream
NRT service data
IP Multicast Stream

FIG.9

FIG.10

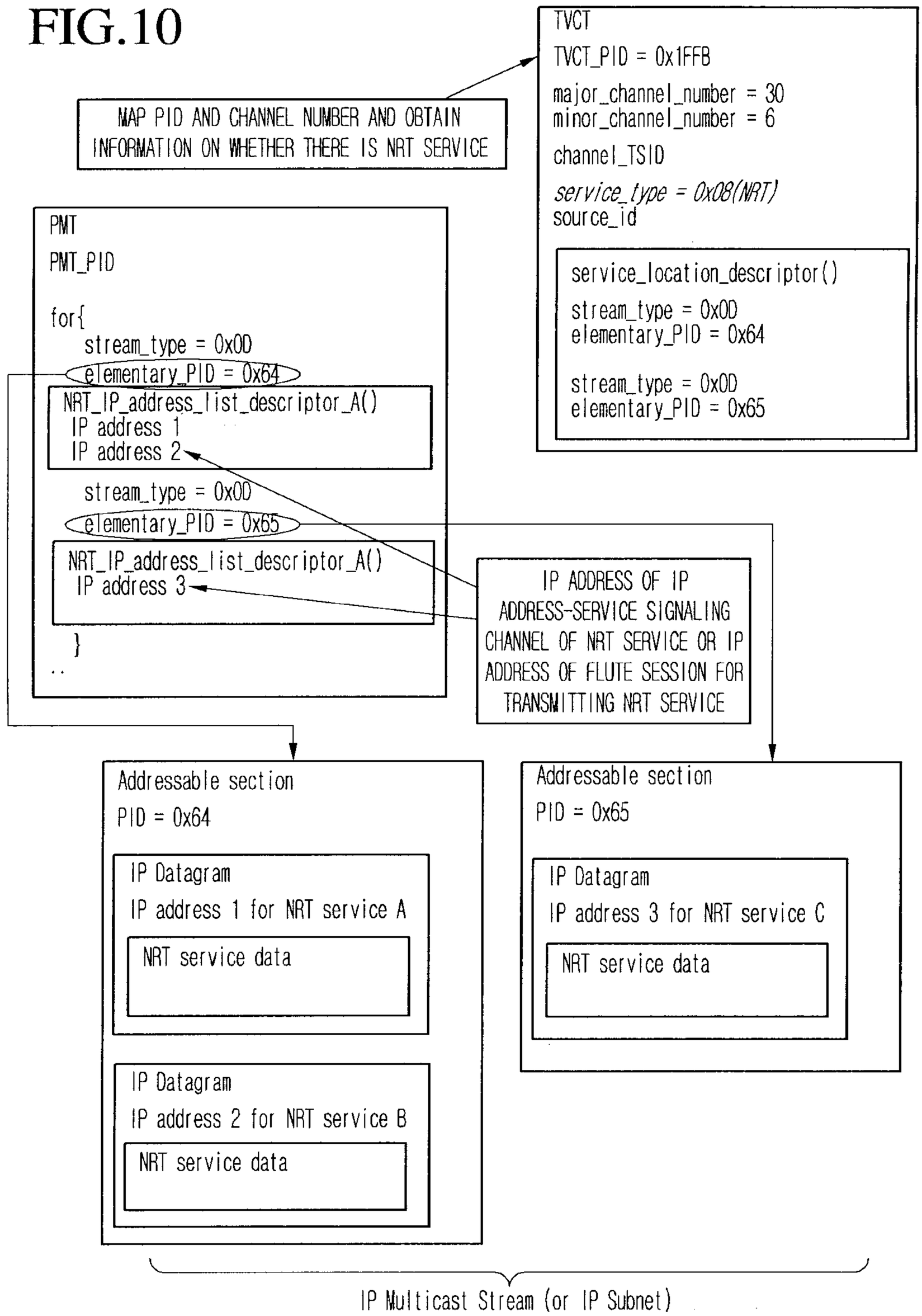

MAP PID AND CHANNEL NUMBER AND OBTAIN INFORMATION ON WHETHER THERE IS NRT SERVICE
TVCT
TVCT_PID = 0x1FFB
major_channel_number = 30
minor_channel_number = 6
channel_TSID
service_type = 0x08(NRT)
source_id
service_location_descriptor()
stream_type = 0x0D
elementary_PID = 0x64
stream_type = 0x0D
elementary_PID = 0x65
PMT
PMT_PID
for{
stream_type = 0x0D
elementary_PID = 0x64
NRT_IP_address_list_descriptor_A()
IP address 1
IP address 2
stream_type = 0x0D
elementary_PID = 0x65
NRT_IP_address_list_descriptor_A()
IP address 3
}
IP ADDRESS OF IP ADDRESS-SERVICE SIGNALING CHANNEL OF NRT SERVICE OR IP ADDRESS OF FLUTE SESSION FOR TRANSMITTING NRT SERVICE
Addressable section
PID = 0x64
IP Datagram
IP address 1 for NRT service A
NRT service data
IP Datagram
IP address 2 for NRT service B
NRT service data
Addressable section
PID = 0x65
IP Datagram
IP address 3 for NRT service C
NRT service data
IP Multicast Stream (or IP Subnet)

FIG.11

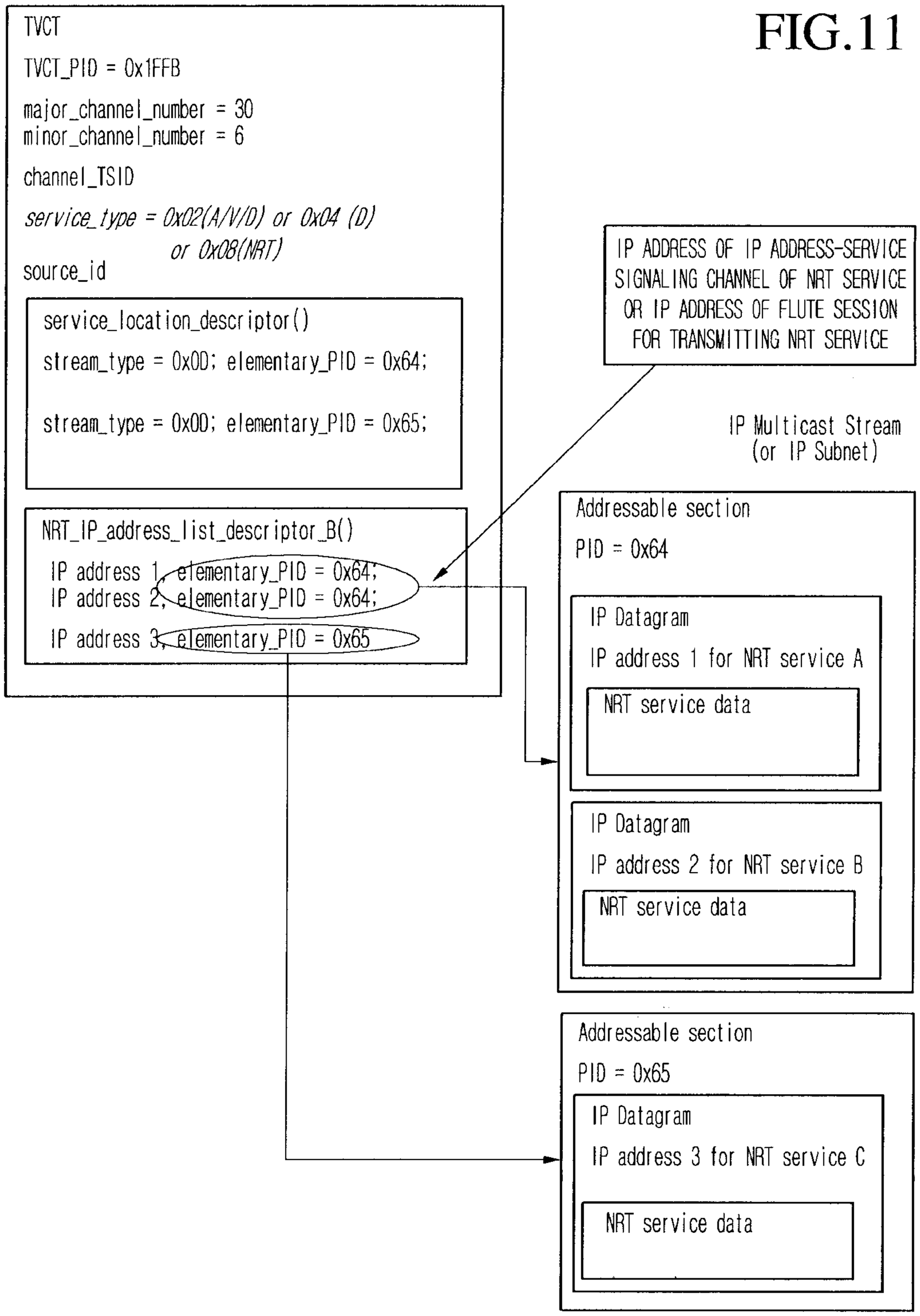

TVCT
TVCT_PID = 0x1FFB
major_channel_number = 30
minor_channel_number = 6
channel_TSID
service_type = 0x02(A/V/D) or 0x04 (D)
or 0x08(NRT)
source_id
service_location_descriptor()
stream_type = 0x0D; elementary_PID = 0x64;
stream_type = 0x0D; elementary_PID = 0x65;
NRT_IP_address_list_descriptor_B()
IP address 1, elementary_PID = 0x64;
IP address 2, elementary_PID = 0x64;
IP address 3, elementary_PID = 0x65
IP ADDRESS OF IP ADDRESS-SERVICE SIGNALING CHANNEL OF NRT SERVICE OR IP ADDRESS OF FLUTE SESSION FOR TRANSMITTING NRT SERVICE
IP Multicast Stream (or IP Subnet)
Addressable section
PID = 0x64
IP Datagram
IP address 1 for NRT service A
NRT service data
IP Datagram
IP address 2 for NRT service B
NRT service data
Addressable section
PID = 0x65
IP Datagram
IP address 3 for NRT service C
NRT service data

FIG.12

Syntax	No. of Bits	Format
NRT_service_map_table_section() {		
table_id	8	0xDB
section_syntax_indicator	1	'0'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension {		
NST_protocol_version	8	uimsbf
reserved	8	uimsbf
}		
reserved	2	'11'
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved	4	'1111'
carrier_frequency	32	uimsbf
channel_TSID	16	uimsbf
program_number	16	uimsbf
source_id	16	uimsbf
num_NRT_services	8	uimsbf
for (i=0; i<num_NRT_services; i++)		
{		
reserved	4	'1111'
NRT_service_status	2	uimsbf
SP_indicator	1	bslbf
CP_indicator	1	bslbf
NRT_service_id	16	uimsbf
short_NRT_service_name	8*8	
reserved	2	'11'
NRT_service_category	6	uimsbf
num_components	5	uimsbf
IP_version_flag	1	bslbf
source_IP_address_flag	1	bslbf
NRT_service_destination_IP_address_flag	1	bslbf
if (source_IP_address_flag)		
source_IP_address	32 or 128	uimsbf
if (NRT_service_destination _IP_address_flag)		
NRT_service_destination_IP_address	32 or 128	uimsbf
for (j=0; j<num_components; j++)		
{		
reserved	1	'1'
essential_component_indicator	1	bsblf
component_destination_IP_address _flag	1	bslbf
port_num_count	5	uimsbf
component_destination_UDP_port_num	16	uimsbf
if (component_destination_IP_address _flag)		
component_destination_IP_address	32 or 128	uimsbf
Reserved	4	'1111'

FIG.13

Syntax	No. of Bits	Format
num_component_level_descriptors	4	uimsbf
for (k=0; k<num_component_level_descriptors; k++)		
{		
component_level_descriptor()	var	
}		
}		
reserved	4	'1111'
num_NRT_service_level_descriptors	4	uimsbf
for (m=0; m<num_MH_service_level_descriptors; m++)		
{		
NRT_service_level_descriptor()	var	
}		
}		
reserved	4	'1111'
num_virtual_channel_level_descriptors	4	uimsbf
for (n=0; n<num_virtual_channel_level_descriptors; n++) {		
{		
virtual_channel_level_descriptor()	var	
}		
}		

FIG.14

Syntax	No. of Bits	Format
MH_component_descriptor() {		
descriptor_tag	8	0x8D
descriptor_length	8	uimsbf
component_type	7	uimsbf
component_encryption_flag	1	bsblf
if (component_encryption_flag == '1') {		
num_STKM_streams	8	uimsbf
for (i=0; i<num_STKM_streams; i++) {		
STKM_stream_id	8	uimsbf
}		
MH_component_ data(component_type)	var	
}		

FIG.15

Syntax	No. of Bits	Format
NRT_component_ data() {		
TSI	16	uimsbf
session_start_time	32	uimsbf
session_end_time	32	uimsbf
reserved	5	'11111'
tias_bandwidth_indicator	1	bslbf
as_bandwidth_indicator	1	bslbf
FEC_OTI_indicator	1	bslbf
if (tias_bandwidth_indicator == '1'){		
tias_bandwidth	16	uimsbf
}		
if (as_bandwidth_indicator == '1'){		
as_bandwidth	16	uimsbf
}		
if (FEC_OTI_indicator == '1'){		
FEC_encoding_id	8	uimsbf
FEC_instance_id	16	uimsbf
}		
}		

FIG.16

Syntax	No.of Bits	Format
NRT_information_table_section() {		
table_id	8	0xTBD
section_syntax_indicator	1	'1'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
service_id	16	uimsbf
reserved	2	'11'
NRT_IT_version_number	5	uimsbf
current_next_indicator	1	'1'
section_number	8	uimsbf
last_section_number	8	uimsbf
protocol_version	8	uimsbf
time_span_start	32	uimsbf
reserved	5	'11111'
time_span_length	11	uimsbf
num_items_in_section	8	uimsbf
for(j=0; j<num_items_in_section; j++) {		
content_linkage	32	uimsbf
updates_available	1	bslbf
reserved	1	'1'
TF_available	1	bslbf
low_latency	1	bslbf
playback_length_in_seconds	20	uimsbf
content_length_included	1	bslbf
playback_delay_included	1	bslbf
expiration_included	1	bslbf
reserved	1	'1'
duration	12	uimsbf
if(content_length_included=1) {	40	uimsbf
content_length		
}		
if(playback_delay_included=1) {		
reserved	4	'1111'
playback_delay	20	uimsbf
}		
if(expiration_included=1) {		
expiration		
}	32	uimsbf
content_name_lingth		
content_name_text()	8	uimsbf
reserved	var	
content_descriptors_length	4	'1111'
for(i=-;i<N; i++) {	12	uimsbf
content_descriptor()		
}		
}		
reserved	6	'111111'
descriptors_length	10	uimsbf
for(i=0; i<M; i++) {		
descriptor()		
}		
}		

FIG.17

Syntax	No. of bits	Format
NRT_content_table_section ( ) {		
table_id	8	uimsbf
section_syntax_indicator	1	'1'
private_indicator	1	'1'
reserved	2	bslbf
section_length	12	uimsbf
NRT_Channel_ID	16	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	1
section_number	8	uimsbf
last_section_number	8	uimsbf
protocol_version	8	uimsbf
num_contents_in_section	8	uimsbf
for (j=0; j < num_contents_in_section; j++) {		
content_version	32	uimsbf
content_id	32	uimsbf
content_available_start_time	32	uimsbf
content_available_end_time	32	uimsbf
ETM_location	2	uimsbf
content_length_in_seconds	30	uimsbf
content_size	48	uimsbf
content_delivery_bit_rate	32	uimsbf
content_title_length	8	uimsbf
content_title_text ( )	var	
descriptors_length	16	uimsbf
for (i=0; i < N; i++) {		
descriptor ( )		
}		
}		
}		

Syntax	No.of Bits	Format
service_map_table_section() {		
table_id	8	0xDB
section_syntax_indicator	1	'0'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension {		
SMT_protocol_version	8	uimsbf
ensemble_id	8	uimsbf
}		
reserved	2	'11'
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
num_services	8	uimsbf
for (j=0;j< num_services; j+ +)		
{		
service_id	16	uimsbf
multi_ensemble_service	2	uimsbf
service_status	2	uimsbf
SP_indicator	1	bslbf
short_service_name_length	3	uimsbf
short_service_name	16*m	
reserved	2	'11'
service_category	6	uimsbf
num_components	5	uimsbf
IP_version_flag	1	bslbf
source_IP_address_flag	1	bslbf
service_destination_IP_address_flag	1	bslbf
if (source_IP_address_flag)		
source_IP_address	32 or 128	uimsbf
if (service_destination_IP_address_flag)		
service_destination_IP_address	32 or 128	uimsbf
for (j=0;j< num_components; j+ +)		
{		
reserved	1	'1'
essential_component_indicator	1	bslbf
component_destination_IP_address_flag	1	bslbf
port_num_count	5	uimsbf
component_destination_UDP_port_num	16	uimsbf
if (component_destination_IP_address_flag)		
component_destination_IP_address	32 or 128	uimsbf
reserved	4	'1111'
num_component_level_descriptors	4	uimsbf
for (k=0;k< num_components_level_descriptors; k++)		
{		
component_level_descriptor()	var	
}		
}		
reserved	4	'1111'
num_service_level_descriptors	4	uimsbf
for (m=0; m<num_service_level_descriptors; m++)		
{		
service_level_descriptor()	var	
}		
}		
reserved	4	'1111'
num_ensemble_level_descriptors	4	uimsbf
for (n=0; n<num_ensemble_level_descriptors; n++) {		
{		
ensemble_level_descriptor()	var	
}		
}		

FIG.18

WO 2012/161535 PCT/KR2012/004124 CA 02834046 2013-10-22

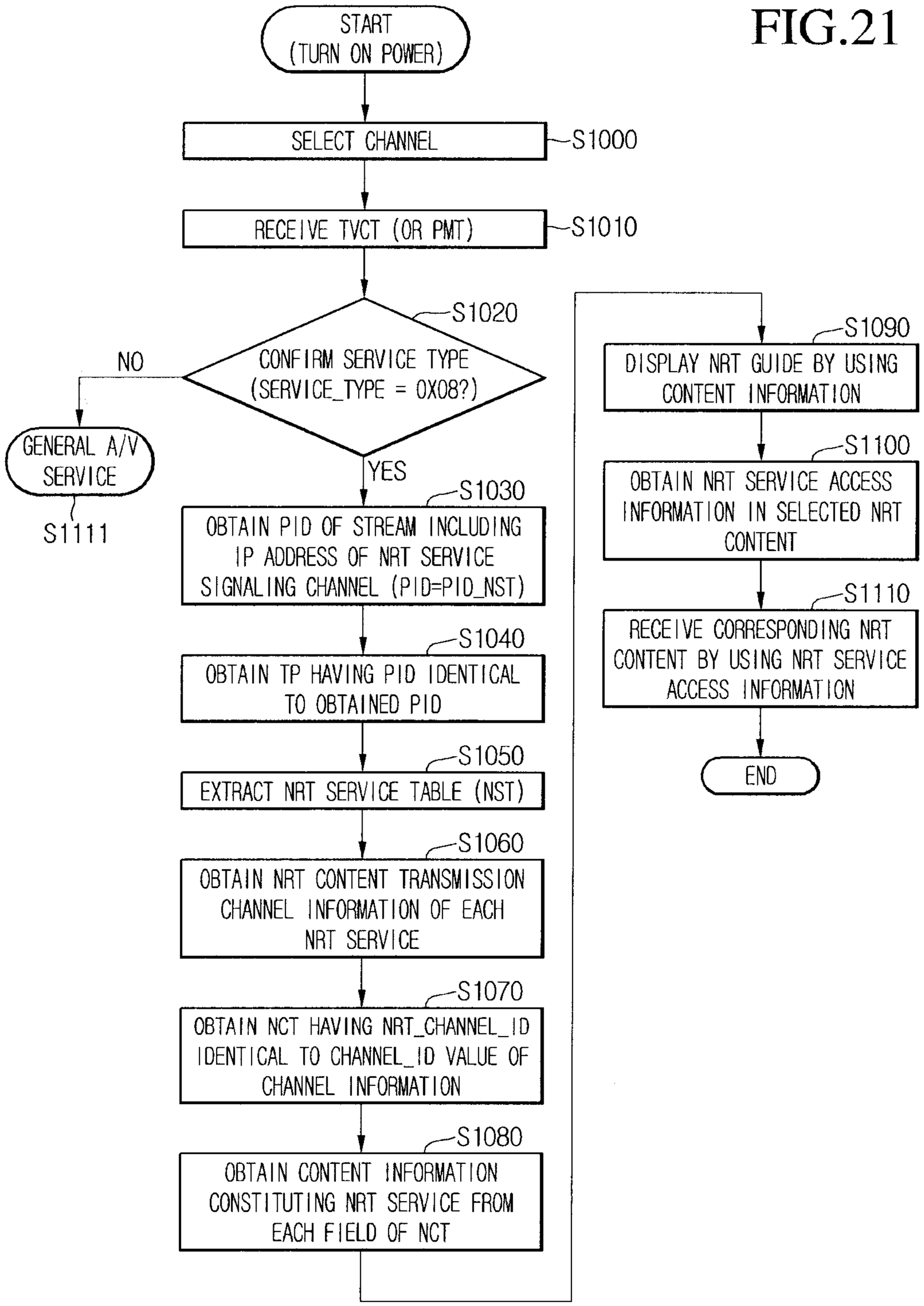
FIG.19

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="urn:atsc:nrt:flute:fdt:2009" xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:atsc:nrt:flute:fdt:2009" elementFormDefault ="qualified">
  <xs:element name="FDT-Instance" type="FDT-InstanceType"/>
  <xs:complexType name="FDT-InstanceType">
    <xs:sequence>
      <xs:element name="File" type="File-Type" maxOccurs ="unbounded"/>
      <xs:element name="FDT-Content-ID" type="xs:usignedShort" minOccurs="0" maxOccurs="unbounded"/>①
      <xs:anynamespace="##other" processContents="skip" minOccurs="0"maxOccurs ="unbounded"/>
    </xs:sequence>
    <xs:attribute name="Expires" type="xs:string" use="required"/>
    <xs:attribute name="Complete" type="xs:boolean" use="optional"/>
    <xs:attribute name="Content-Type" type="xs:string" use="optional"/>
    <xs:attribute name="Content-Encoding" type="xs:string" use="optional"/>
    <xs:attribute name="FEC-OTI-FEC-Encoding-ID" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-FEC-Instance-ID" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Maximum-Source-Block-Length" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Encoding-Symbol-Length" type="xs:unsignedLong" use="optional"/>
    <xs:attributename="FEC-OTI-Max-Number-of-Encoding-Symbols" type="-xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Scheme-Specific-Info" type="xs:base64Binary" use="optional"/>
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
  <xs:complexType name="File-Type">
    <xs:sequence >
      <xs:element name="File-Content-ID" type="Content-ID-Type" minOccurs="0" maxOccurs ="unbounded"/>②
      <xs:any namespace="##other"processContents ="skip"minOccurs ="0"maxOccurs ="unbounded"/>
    </xs:sequence >
    <xs:attribute name="Content-Location" type="xs:anyURI" use="required"/>
    <xs:attribute name="TOI" type="xs:positiveInteger" use="required"/>
    <xs:attribute name="Content-Length" type="xs:unsignedLong" use="required"/>
    <xs:attribute name="Transfer-Length" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="Content-Type" type="xs:string" use="optional"/>
    <xs:attribute name="Content-Encoding" type="xs:string" use="optional"/>
    <xs:attribute name="Content-MD5" type="xs:base64Binary" use="optional"/>
    <xs:attribute name="FEC-OTI-FEC-Encoding-ID" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-FEC-Instance-ID" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Maximum-Source-Block-Length" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Encoding-Symbol-Length" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Max-Number-of-Encoding-Symbols" type="xs:unsignedLong" use="optional"/>
    <xs:attribute name="FEC-OTI-Scheme-Specific-Info" type="xs:base64Binary" use="optional"/>
    <xs:anyAttribute processContents ="skip"/>
  </xs:complexType >
  <xs:complexType name="Content-ID-Type">
    <xs:attribute name="Content-ID" type="xs:unsingedShort" use="required"/>
    <xs:attribute name="entry" type="xs:boolean " default="false" use="optional"/>③
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
</xs:schema>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="urn:atsc:nrt:flute:fdt:2009" xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:atsc:nrt:flute:fdt:2009" elementFormDefault="qualified">
 <xs:element name="FDT-Instance" type="FDT-InstanceType"/>
 <xs:complexType name="FDT-InstanceType ">
  <xs:sequence >
   <xs:element name="File" type="File-Type"maxOccurs ="unbounded"/>
   <xs:element name="FDT-Content-ID" type="FDT-Content-ID-Type" minOccurs="0" maxOccurs="unbounded"/>
   <xs:any namespace="##other" processContents ="skip"minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>                                                                          ①
  <xs:attribute name="Expires" type="xs:string" use="required"/>
  <xs:attribute name="Complete" type="xs:boolean" use="optional"/>
  <xs:attribute name="Content-Type" type="xs:string" use="optional"/>
  <xs:attribute name="Content-Encoding" type="xs:string" use="optional"/>
  <xs:attribute name="FEC-OTI-FEC-Encoding-ID" type="xs:unsignedLong " use="optional"/>
  <xs:attribute name="FEC-OTI-FEC-Instance-ID" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Maximum-Source-Block-Length" type=" xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Encoding-Symbol-Length" type=" xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Max-Number-of-Encoding-Symbols" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Scheme-Specific-Info" type="xs:base64Binary" use="optional"/>
  <xs:anyAttribute processContents ="skip"/>
</xs:complexType>
<xs:complexType name="File-Type">
  <xs:sequence>
   <xs:element name="File-Content-ID" type="Content-ID-Type" minOccurs="0"maxOccurs ="unbounded"/>
   <xs:any namespace="##other"processContents="skip"minOccurs="0"maxOccurs="unbounded"/>
  </xs:sequence >
  <xs:attribute name="Content-Location" type="xs:anyURI" use="required"/>
  <xs:attribute name="TOI" type="xs:positiveInteger" use="required"/>
  <xs:attribute name="Content- Length" type="xs:unsignedLong" use="required"/>
  <xs:attribute name="Transfer-Length" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="Content- Type" type="xs:string" use="optional"/>
  <xs:attribute name="Content- Encoding" type="xs:string" use="optional"/>
  <xs:attribute name="Content- MD5" type="xs:base64Binary" use="optional"/>
  <xs:attribute name="FEC-OTI-FEC-Encoding-ID" type=" xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-FEC-Instance-ID" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Maximum-Source-Block-Length" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Encoding-Symbol-Length" type="xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Max-Number-of-Encoding-Symbols" type=" xs:unsignedLong" use="optional"/>
  <xs:attribute name="FEC-OTI-Scheme-Specific-Info" type="xs:base64Binary" use="optional"/>
  <xs:anyAttribute processContents ="skip"/>
 </xs:complexType >
 <xs:complexTypename="FDT-Content-ID-Type">②
  <xs:sequence>
   <xs:elementname="Entry-Content-Location" type=xs:anyURI"minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="Content-ID" type="xs:unsingedShort" use="required"/>
  <xs:anyAttribute processContents="skip"/>
 <xs:complexType>
 <xs:complexType name="Content-ID-Type">
  <xs:attribute name="Content-ID" type="xs:unsingedShort" use="required"/>
  <xs:attribute name="entry" type="xs:boolean" default="false" use="optional"/>
  <xs:anyAttribute processContents ="skip"/>
 </xs:complexType >
</xs:schema >
```

FIG.20

FIG.21


START
(TURN ON POWER)
SELECT CHANNEL
S1000
RECEIVE TVCT (OR PMT)
S1010
S1020
CONFIRM SERVICE TYPE
(SERVICE_TYPE = 0X08?)
NO
GENERAL A/V
SERVICE
S1111
YES
S1030
OBTAIN PID OF STREAM INCLUDING
IP ADDRESS OF NRT SERVICE
SIGNALING CHANNEL (PID=PID_NST)
S1040
OBTAIN TP HAVING PID IDENTICAL
TO OBTAINED PID
S1050
EXTRACT NRT SERVICE TABLE (NST)
S1060
OBTAIN NRT CONTENT TRANSMISSION
CHANNEL INFORMATION OF EACH
NRT SERVICE
S1070
OBTAIN NCT HAVING NRT_CHANNEL_ID
IDENTICAL TO CHANNEL_ID VALUE OF
CHANNEL INFORMATION
S1080
OBTAIN CONTENT INFORMATION
CONSTITUTING NRT SERVICE FROM
EACH FIELD OF NCT
S1090
DISPLAY NRT GUIDE BY USING
CONTENT INFORMATION
S1100
OBTAIN NRT SERVICE ACCESS
INFORMATION IN SELECTED NRT
CONTENT
S1110
RECEIVE CORRESPONDING NRT
CONTENT BY USING NRT SERVICE
ACCESS INFORMATION
END

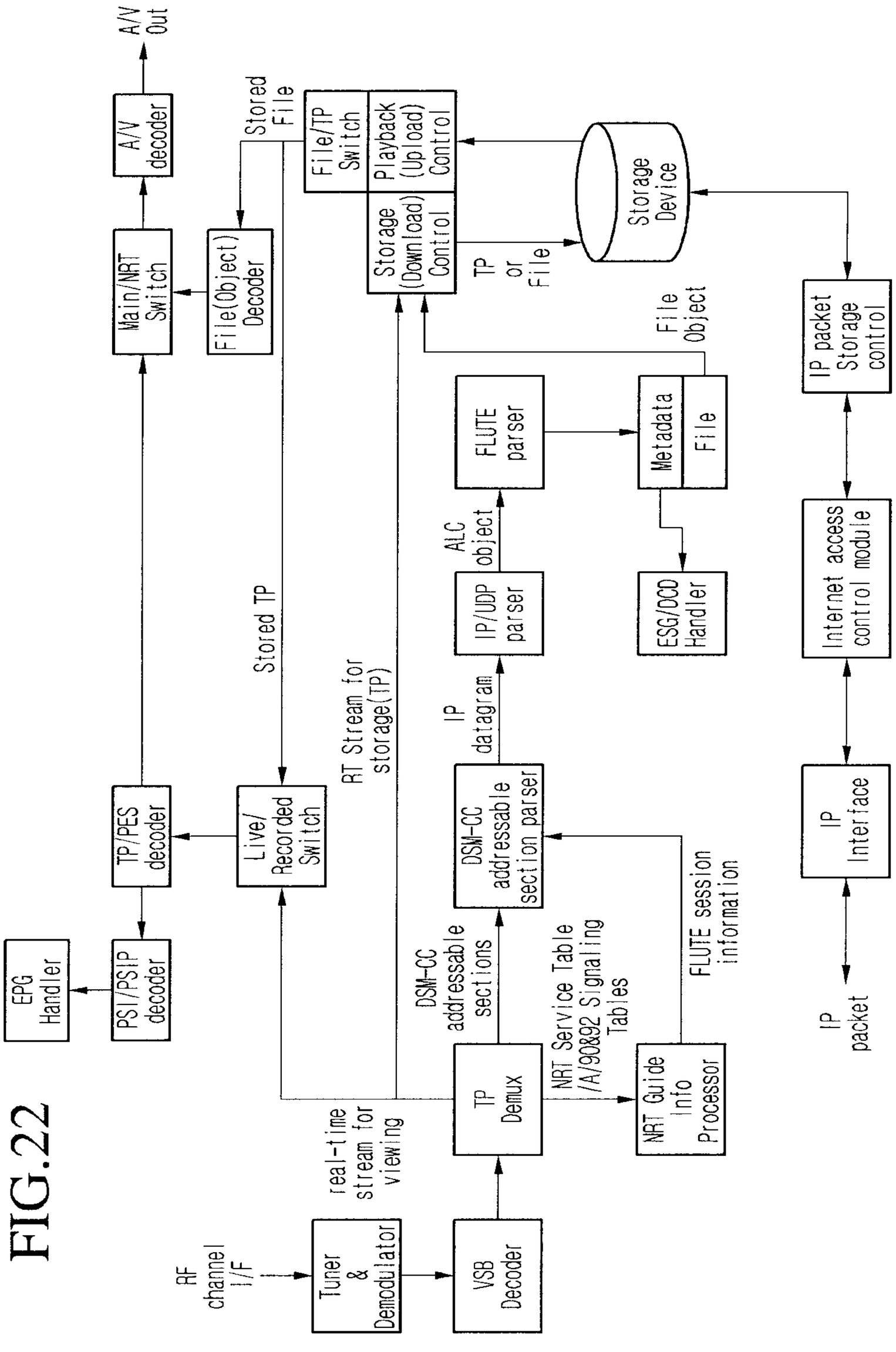

A/V Out
A/V decoder
Main/NRT Switch
File(Object) Decoder
Stored File
File/TP Switch
Playback (Upload) Control
Storage (Download) Control
Storage Device
TP or File
File Object
IP packet Storage control
Stored TP
RT Stream for storage(TP)
IP datagram
IP/UDP parser
ALC object
FLUTE parser
Metadata
File
ESG/DCD Handler
Internet access control module
IP Interface
IP packet
TP/PES decoder
Live/ Recorded Switch
DSM-CC addressable section parser
EPG Handler
PSI/PSIP decoder
DSM-CC addressable sections
NRT Service Table /A/90&92 Signaling Tables
FLUTE session information
real-time stream for viewing
TP Demux
NRT Guide Info Processor
RF channel I/F
Tuner & Demodulator
VSB Decoder

FIG.22

FIG.23

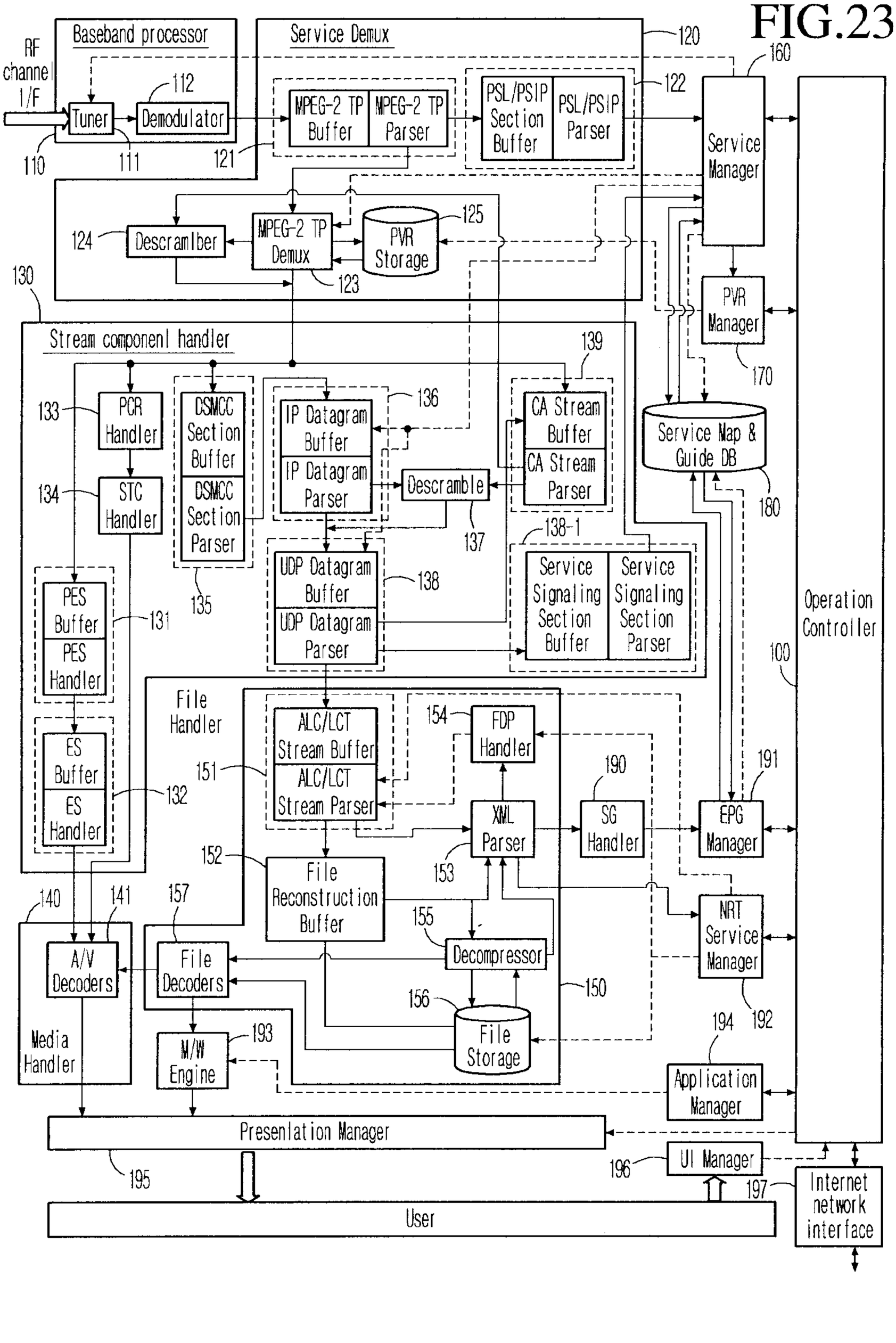

RF channel I/F
Baseband processor
Service Demux
120
160
112
Tuner
Demodulator
MPEG-2 TP Buffer
MPEG-2 TP Parser
PSL/PSIP Section Buffer
PSL/PSIP Parser
122
Service Manager
110
111
121
124
Descrambler
MPEG-2 TP Demux
PVR Storage
125
123
PVR Manager
130
Stream component handler
139
170
133
PCR Handler
DSMCC Section Buffer
DSMCC Section Parser
IP Datagram Buffer
IP Datagram Parser
136
CA Stream Buffer
CA Stream Parser
Service Map & Guide DB
134
STC Handler
Descramble
180
137
138-1
135
UDP Datagram Buffer
UDP Datagram Parser
138
Service Signaling Section Buffer
Service Signaling Section Parser
PES Buffer
PES Handler
131
Operation Controller
100
File Handler
ALC/LCT Stream Buffer
ALC/LCT Stream Parser
151
154
FDP Handler
190
191
ES Buffer
ES Handler
132
XML Parser
SG Handler
EPG Manager
152
153
File Reconstruction Buffer
140
141
157
155
NRT Service Manager
A/V Decoders
File Decoders
Decompressor
150
156
194
192
Media Handler
M/W Engine
193
File Storage
Application Manager
Presentation Manager
196
UI Manager
195
197
Internet network interface
User

FIG.24

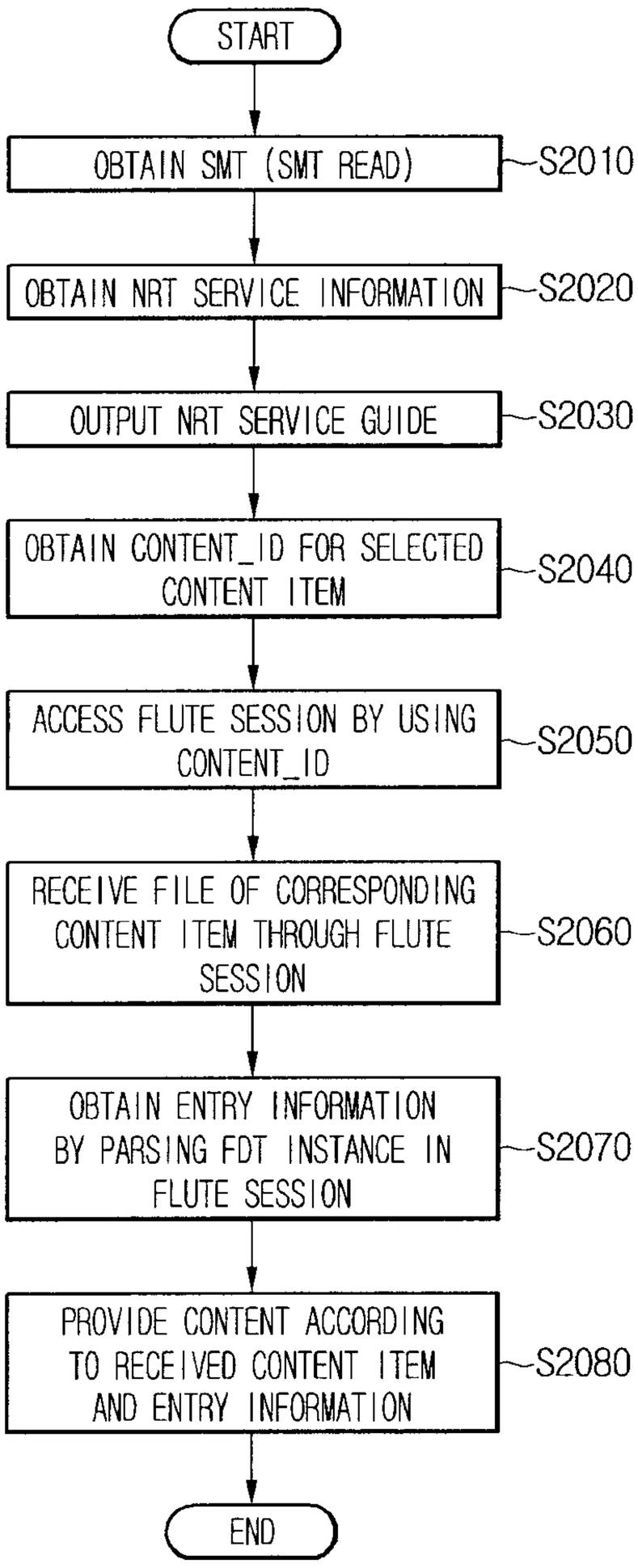

START
OBTAIN SMT (SMT READ)
S2010
OBTAIN NRT SERVICE INFORMATION
S2020
OUTPUT NRT SERVICE GUIDE
S2030
OBTAIN CONTENT_ID FOR SELECTED CONTENT ITEM
S2040
ACCESS FLUTE SESSION BY USING CONTENT_ID
S2050
RECEIVE FILE OF CORRESPONDING CONTENT ITEM THROUGH FLUTE SESSION
S2060
OBTAIN ENTRY INFORMATION BY PARSING FDT INSTANCE IN FLUTE SESSION
S2070
PROVIDE CONTENT ACCORDING TO RECEIVED CONTENT ITEM AND ENTRY INFORMATION
S2080
END

FIG.25

Syntax	No. of Bits	Format
tdo_trigger_table_section() {		
table_id	8	0xTBD
section_syntax_indicator	1	0
private_indicator	1	1
reserved	2	11
section_length	12	uimsbf
source_id	16	uimsbf
reserved	2	11
TTT_version_number	5	uimsbf
current_next_indicator	1	1
section_number	8	uimsbf
last_section_number	8	uimsbf
num_triggers_in_section	8	uimsbf
for (i=0; i< num_triggers_in_section; i++) {		uimsbf
trigger_id	32	uimsbf
trigger_time	32	uimsbf
trigger_action	16	uimsbf
trigger_description_length	8	uimsbf
trigger_description_text	var	
service_id_ref	16	uimsbf
content_linkage	32	uimsbf
num_trigger_descriptors	8	uimsbf
for (j=0; j<num_bundle_descriptors; j++) {		
trigger_descriptor ()	var	
}		
}		
}		

FIG.26

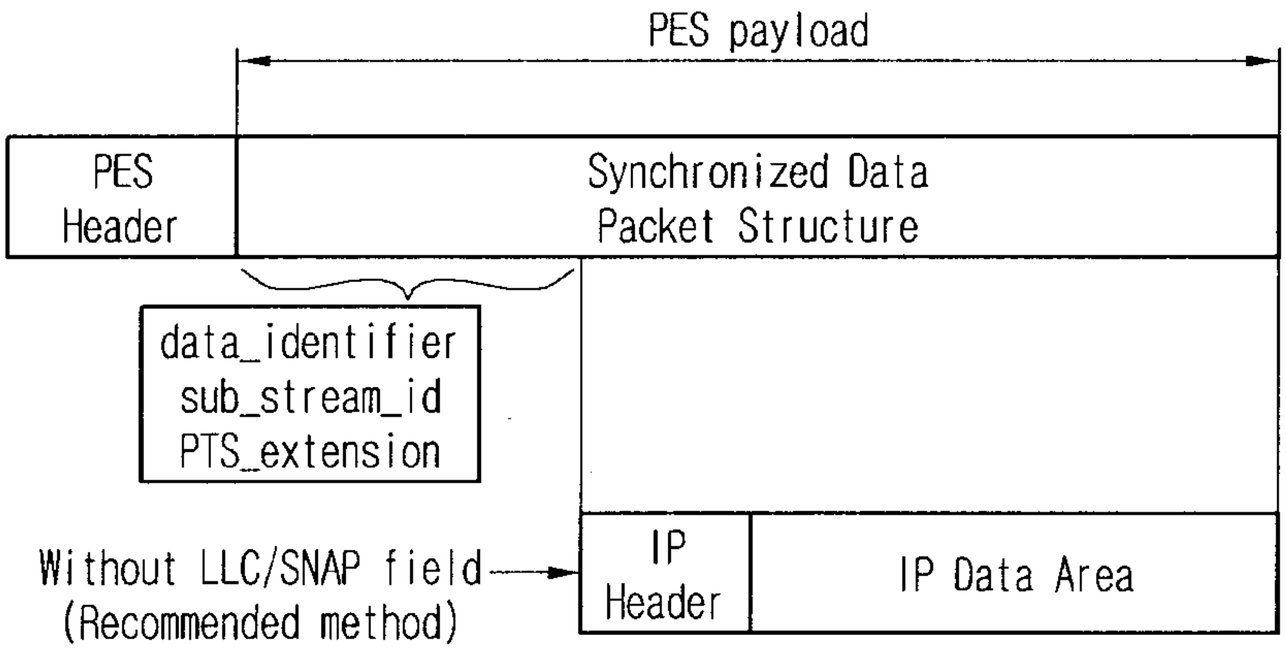

PES payload
PES Header
Synchronized Data Packet Structure
data_identifier
sub_stream_id
PTS_extension
Without LLC/SNAP field
(Recommended method)
IP Header
IP Data Area

FIG.27

Syntax	No. of bits	Format
synchronized_data_packet(){		
data_identifier	8	uimsbf
sub_stream_id	8	uimsbf
PTS_extension_flag	1	bslbf
output_data_rate_flag	1	bslbf
reserved	2	'11'
synchronized_data_packet_header_length	4	uimsbf
if (PTS_extension_flag=='1'){		
reserved	7	'1111111'
PTS_extension	9	uimsbf
}		
for (i=0;i<N1;i++){		
synchronized_data_private_data_byte	8	bslbf
}		
for (i=0;i<N2;i++){		
synchronized_data_byte	8	bslbf
}		
}		

FIG.28

Syntax	No. of bits	Format
contentTypeDescriptor(){		
descriptorTag	8	0x72
descriptorLength	8	uimsbf
for(i=0;i<descriptorLength;i++){		
contentTypeByte	8	bslbf
}		
}		

FIG.29

Syntax	No. Of bits
TS_program_map_section(){	
table_id	8
section_syntax_indicator	1
'0'	1
reserved	2
section_length	12
program_number	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
reserved	3
PCR_PID	13
reserved	4
program_info_length	12
for(i=0;i<N;i++){	
descriptor()	
}	
for(i=0;i<N1;i++){	
steam_type	8
reserved	3
elementary_PID	13
reserved	4
ES_info_length	12
for(i=0;i<N2;i++){	
descriptor()	
}	
}	
CRC_32	32
}	

Syntax	No. of Bits	Format
service_id_descriptor() {		
descriptor_tag	8	0xC2
descriptor_length	8	uimsbf
service_count	8	uimsbf
for (i=0; i<service_count; i++)		
service_id	16	bslbf
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.30

Syntax	No. Of bits
TS_program_map_section(){	
table_id	8
section_syntax_indicator	1
'0'	1
reserved	2
section_length	12
program_number	16
reserved	2
version_number	5
current_next_indicator	1
section_number	8
last_section_number	8
reserved	3
PCR_PID	13
reserved	4
program_info_length	12
for(i=0;i<N;i++){	
descriptor()	
}	
for(i=0;i<N1;i++){	
steam_type	8
reserved	3
elementary_PID	13
reserved	4
ES_info_length	12
for(i=0;i<N2;i++){	
descriptor() ←	
}	
}	
CRC_32	32
}	

Syntax	No. of Bits	Format
Trigger_stream_descriptor() {		
descriptor_tag	8	0xC2
descriptor_length	8	uimsbf
target_service_count	8	uimsbf
for (i=0; i<service_count; i++) {		
target_service_id	16	bslbf
target_content_item_count	8	uimsbf
for(k=0; k<target_content_item_count; k++)		
target_content_linkage	32	bslbf
}		
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.31

	No. of bits	Identifier
application_information_section(){		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
test_application_flag	1	bslbf
application_type	15	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved_future_use	4	bslbf
common_descriptors_length	12	uimsbf
for(i=0;i<N;i++){		
descriptor()		
}		
reserved_future_use	4	bslbf
application_loop_length	12	uimsbf
for(i=0;i<N;i++){		
application_identifier()		
application_control_code	8	uimsbf
reserved_future_use	4	bslbf
application_descriptors_loop_length	12	uimsbf
for(j=0;j<N;j++){		
descriptor()		
}		
}		
CRC_32	32	rpchof
}		

FIG.32

Syntax	No. of Bits	Format
System_time_table_section(){		
table_id	8	uimsbf
section_syntax_indicator	1	'1'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension	16	0x0000
reserved	2	'11'
version_number	5	'00000'
current_next_indicator	1	'1'
section_number	8	0x00
last_section_number	8	0x00
protocol_version	8	uimsbf
system_time	32	uimsbf
GPS_UTC_offset	8	uimsbf
daylight_savings	16	uimsbf
for (i=0;i<N;i++){		
descriptor()		
}		
CRC_32	32	rpchof
}		

FIG.33

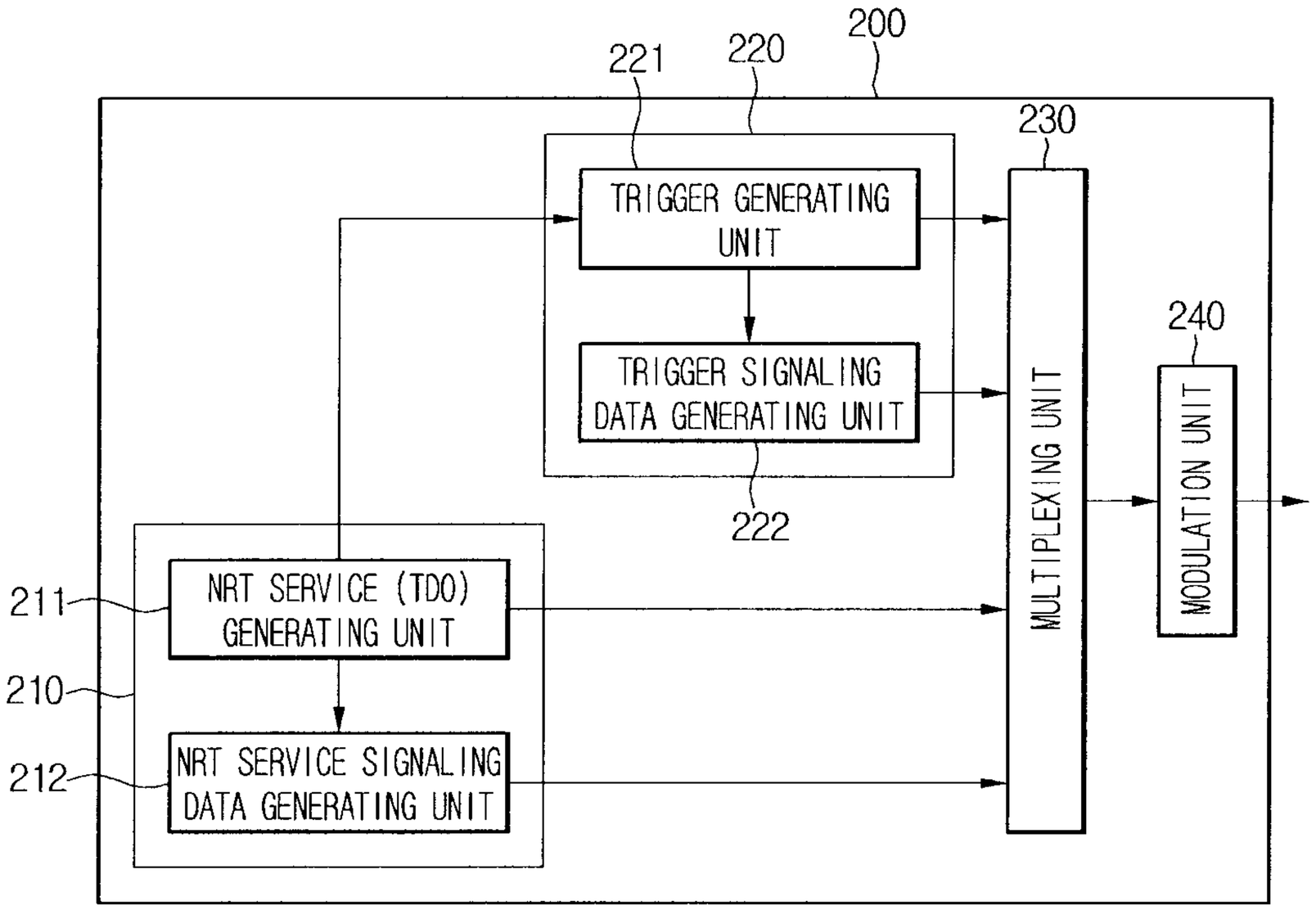

200
221
220
230
TRIGGER GENERATING UNIT
TRIGGER SIGNALING DATA GENERATING UNIT
222
240
MODULATION UNIT
MULTIPLEXING UNIT
211
NRT SERVICE (TDO) GENERATING UNIT
210
212
NRT SERVICE SIGNALING DATA GENERATING UNIT

FIG.34

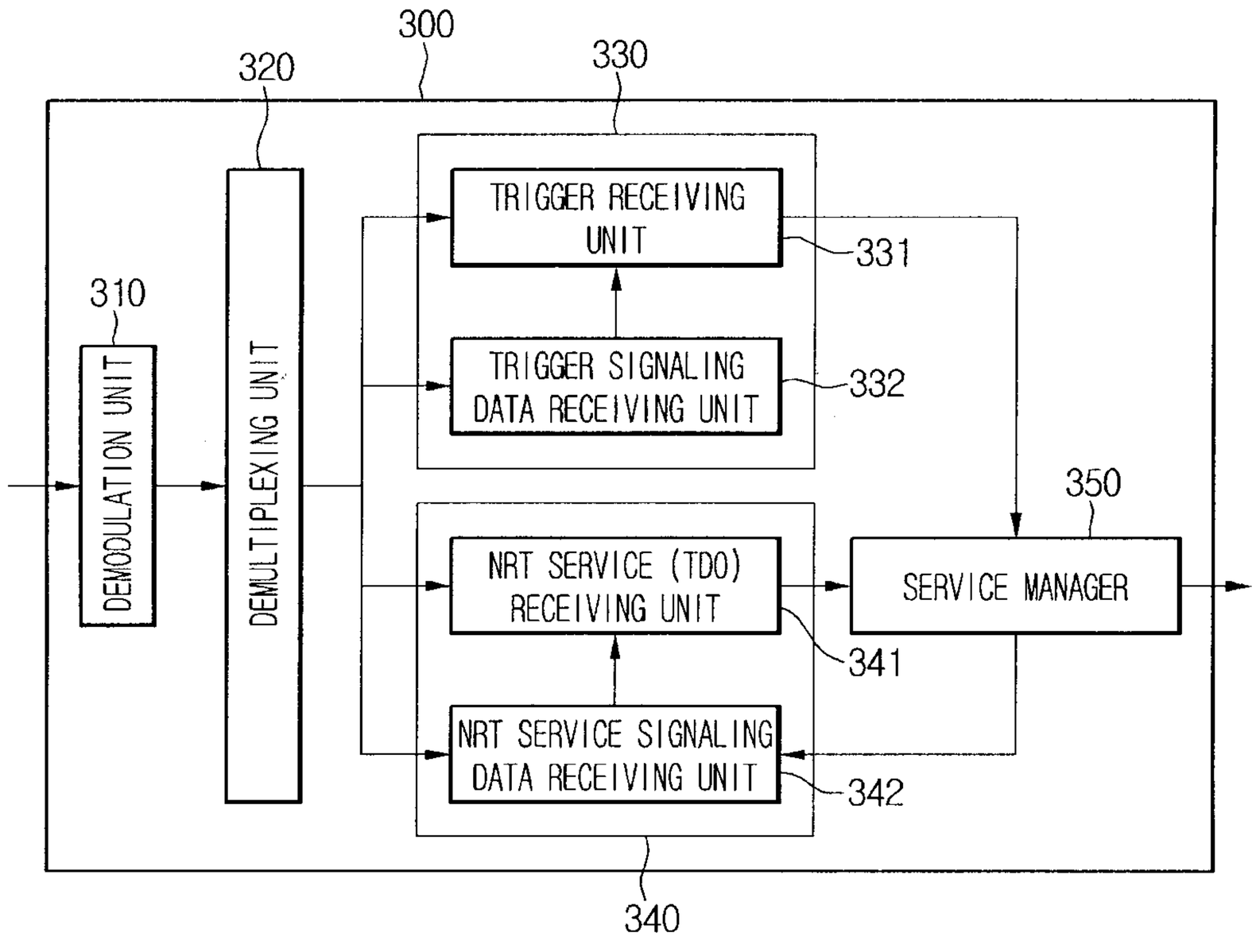

300
320
330
TRIGGER RECEIVING UNIT
331
310
DEMODULATION UNIT
DEMULTIPLEXING UNIT
TRIGGER SIGNALING DATA RECEIVING UNIT
332
350
NRT SERVICE (TDO) RECEIVING UNIT
SERVICE MANAGER
341
NRT SERVICE SIGNALING DATA RECEIVING UNIT
342
340

FIG.35

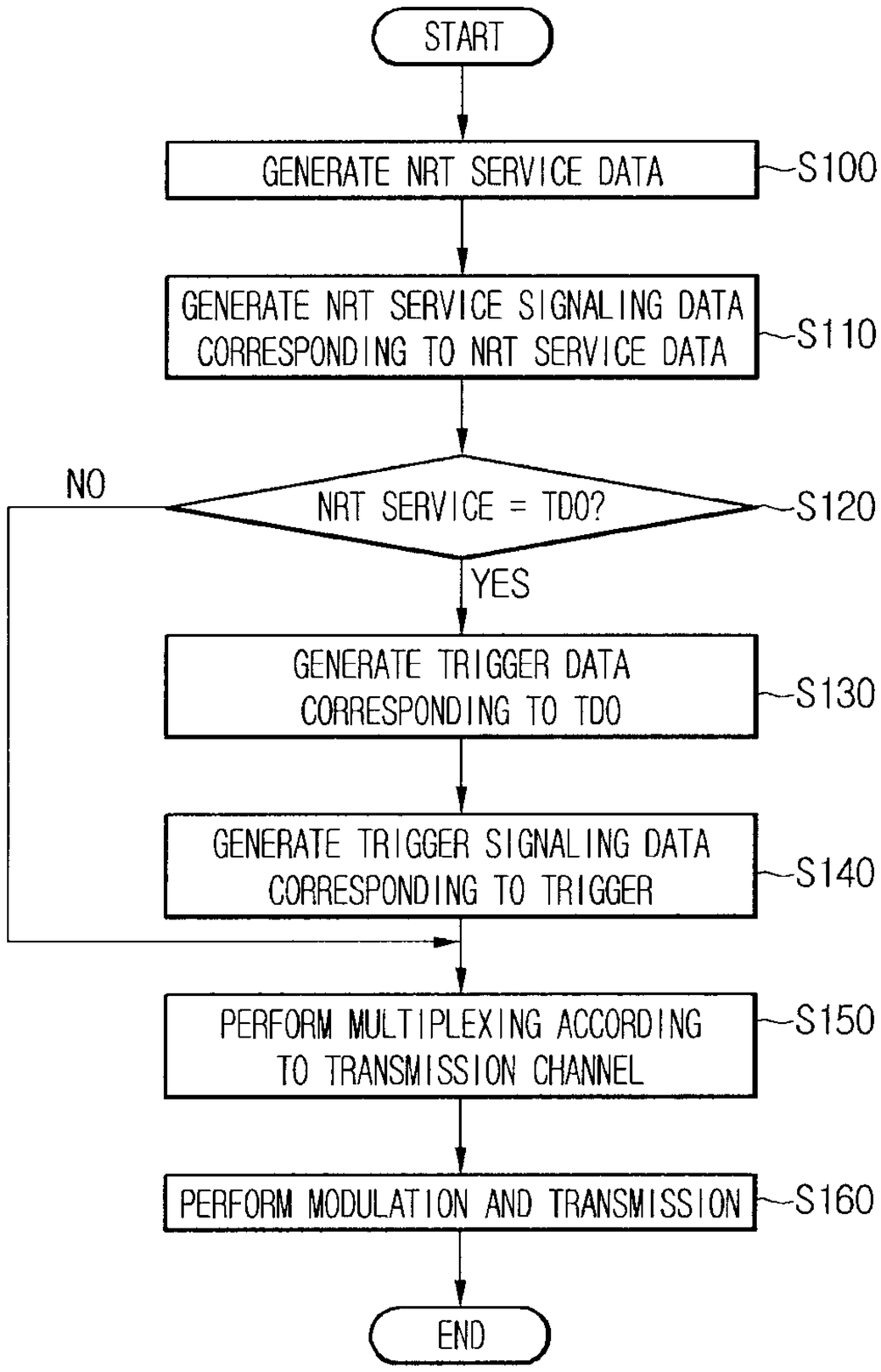

START
GENERATE NRT SERVICE DATA
S100
GENERATE NRT SERVICE SIGNALING DATA CORRESPONDING TO NRT SERVICE DATA
S110
NO
NRT SERVICE = TDO?
S120
YES
GENERATE TRIGGER DATA CORRESPONDING TO TDO
S130
GENERATE TRIGGER SIGNALING DATA CORRESPONDING TO TRIGGER
S140
PERFORM MULTIPLEXING ACCORDING TO TRANSMISSION CHANNEL
S150
PERFORM MODULATION AND TRANSMISSION
S160
END

FIG.36

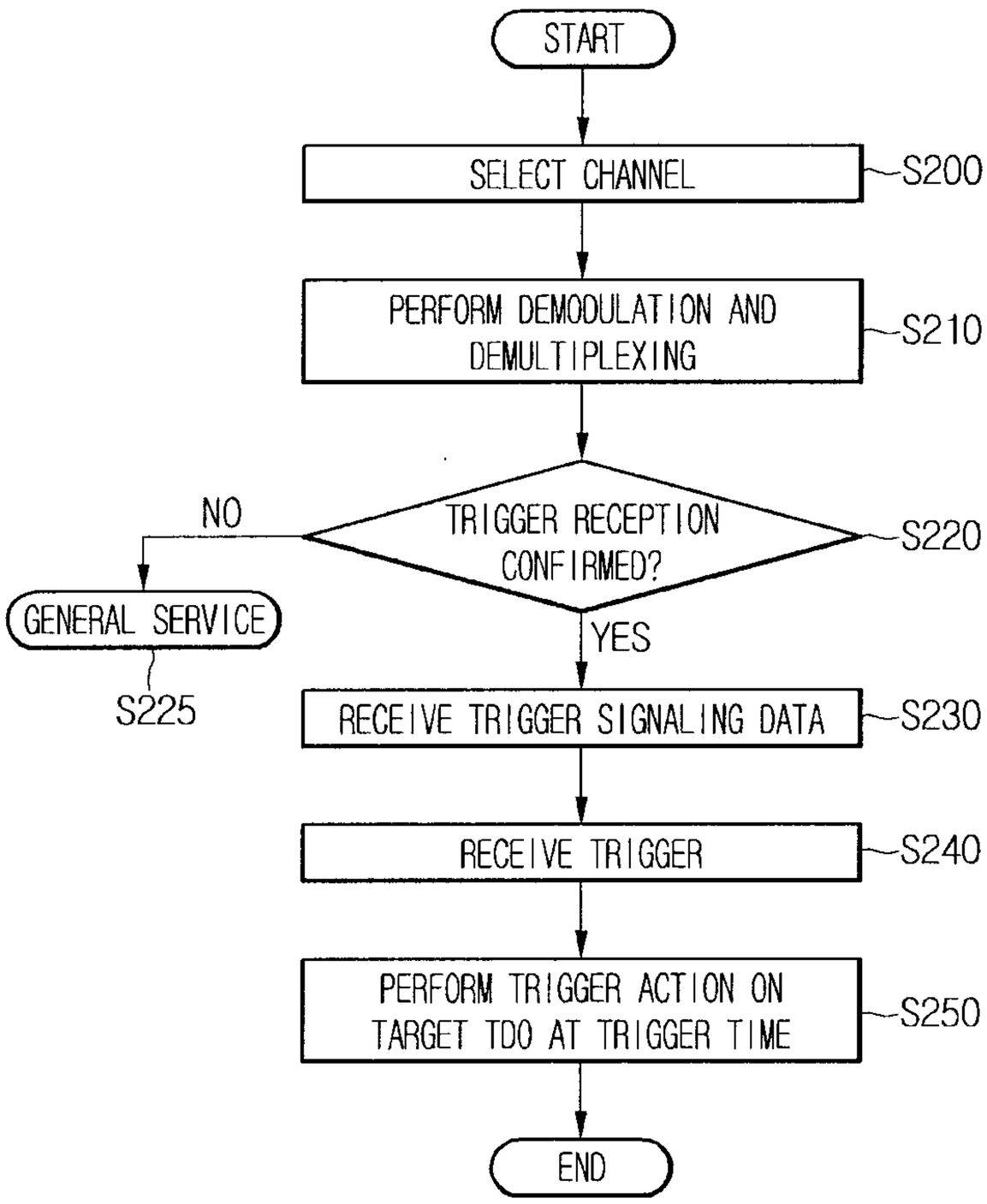

START
SELECT CHANNEL
S200
PERFORM DEMODULATION AND DEMULTIPLEXING
S210
TRIGGER RECEPTION CONFIRMED?
S220
NO
GENERAL SERVICE
S225
YES
RECEIVE TRIGGER SIGNALING DATA
S230
RECEIVE TRIGGER
S240
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME
S250
END

FIG.37

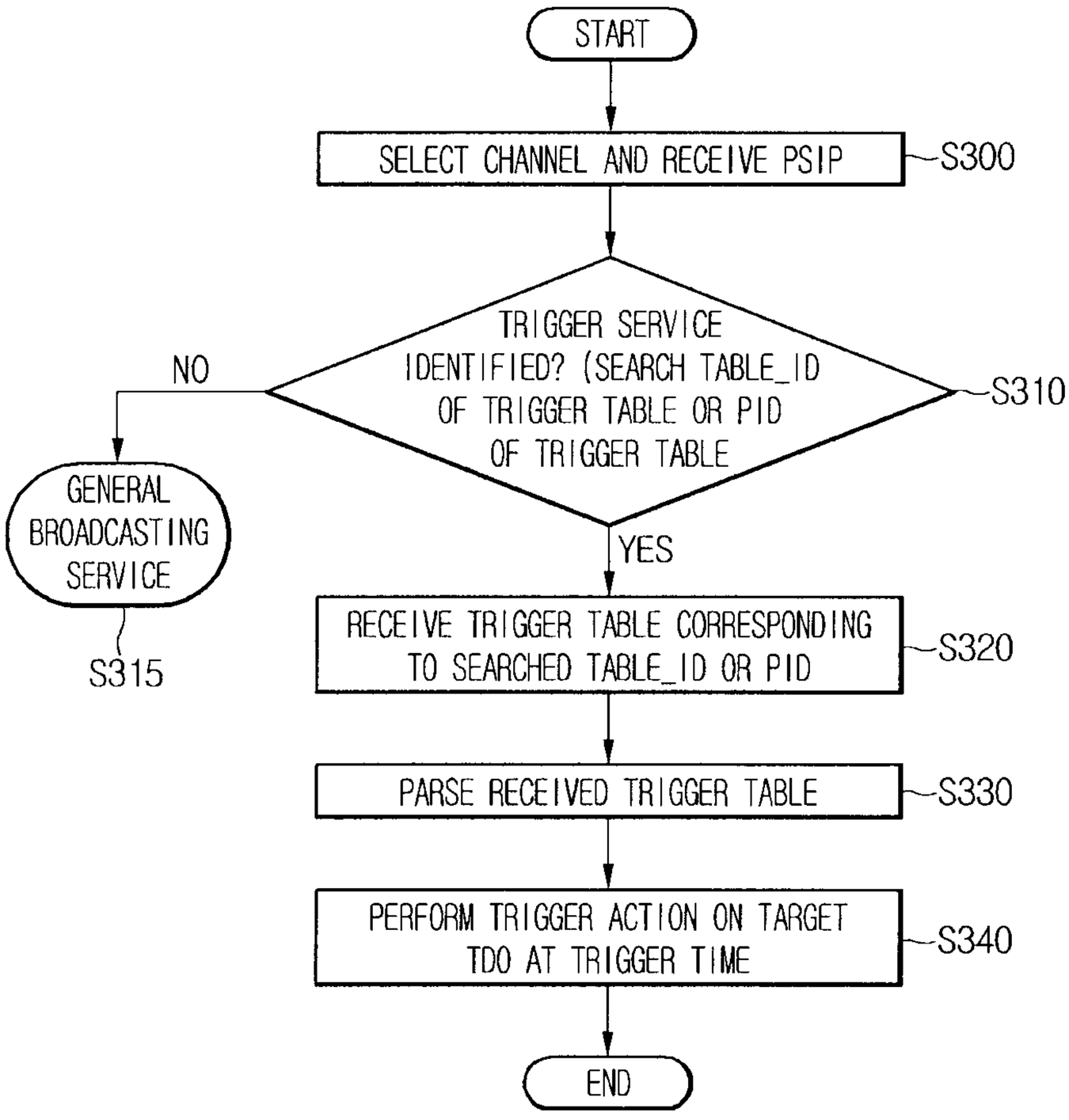

START
SELECT CHANNEL AND RECEIVE PSIP
S300
TRIGGER SERVICE IDENTIFIED? (SEARCH TABLE_ID OF TRIGGER TABLE OR PID OF TRIGGER TABLE
S310
NO
YES
GENERAL BROADCASTING SERVICE
S315
RECEIVE TRIGGER TABLE CORRESPONDING TO SEARCHED TABLE_ID OR PID
S320
PARSE RECEIVED TRIGGER TABLE
S330
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME
S340
END

FIG.38

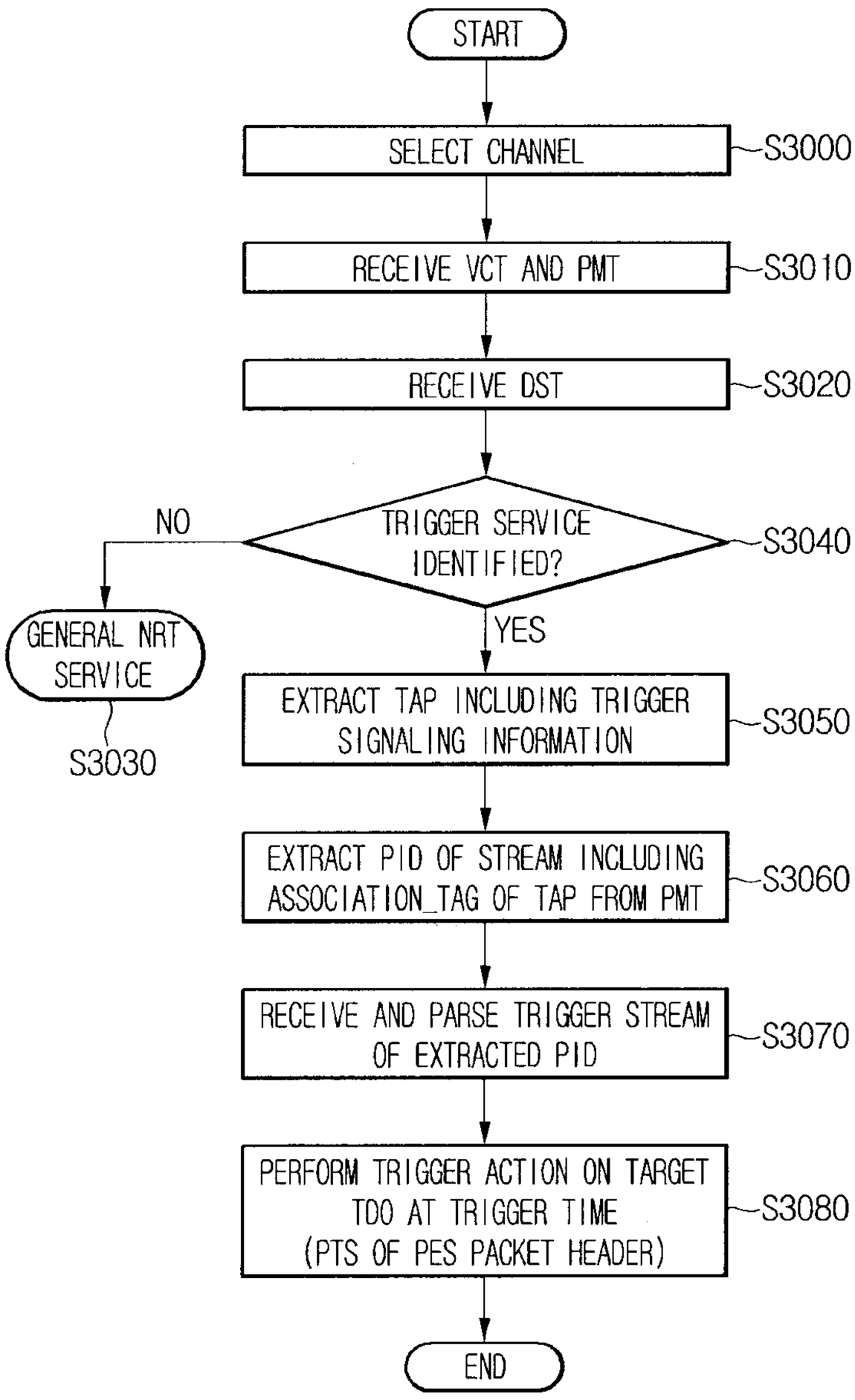

START
SELECT CHANNEL
S3000
RECEIVE VCT AND PMT
S3010
RECEIVE DST
S3020
TRIGGER SERVICE IDENTIFIED?
S3040
NO
YES
GENERAL NRT SERVICE
S3030
EXTRACT TAP INCLUDING TRIGGER SIGNALING INFORMATION
S3050
EXTRACT PID OF STREAM INCLUDING ASSOCIATION_TAG OF TAP FROM PMT
S3060
RECEIVE AND PARSE TRIGGER STREAM OF EXTRACTED PID
S3070
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME (PTS OF PES PACKET HEADER)
S3080
END

FIG.39

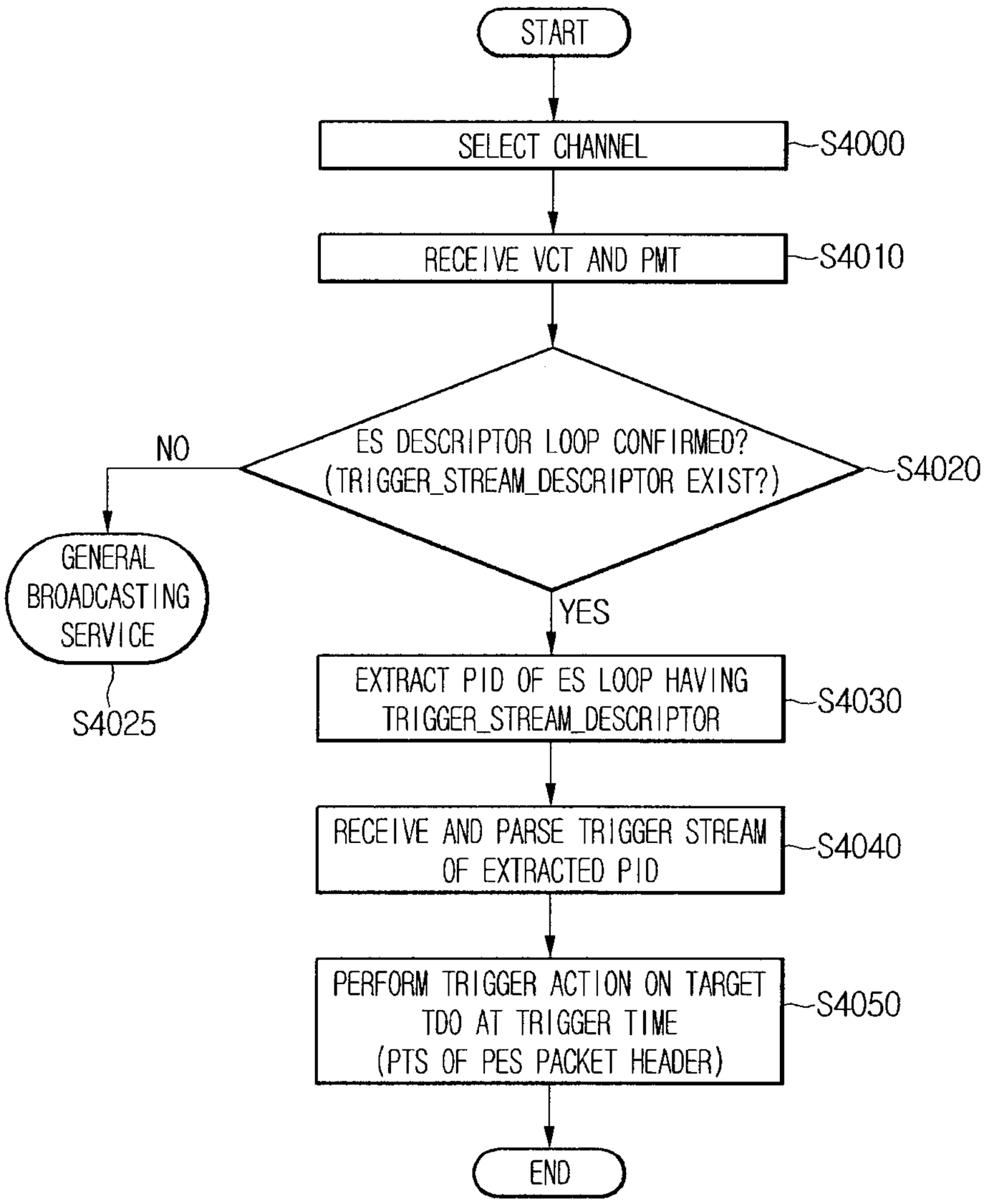

START
SELECT CHANNEL
S4000
RECEIVE VCT AND PMT
S4010
ES DESCRIPTOR LOOP CONFIRMED?
(TRIGGER_STREAM_DESCRIPTOR EXIST?)
S4020
NO
YES
GENERAL BROADCASTING SERVICE
S4025
EXTRACT PID OF ES LOOP HAVING TRIGGER_STREAM_DESCRIPTOR
S4030
RECEIVE AND PARSE TRIGGER STREAM OF EXTRACTED PID
S4040
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME
(PTS OF PES PACKET HEADER)
S4050
END

FIG.40

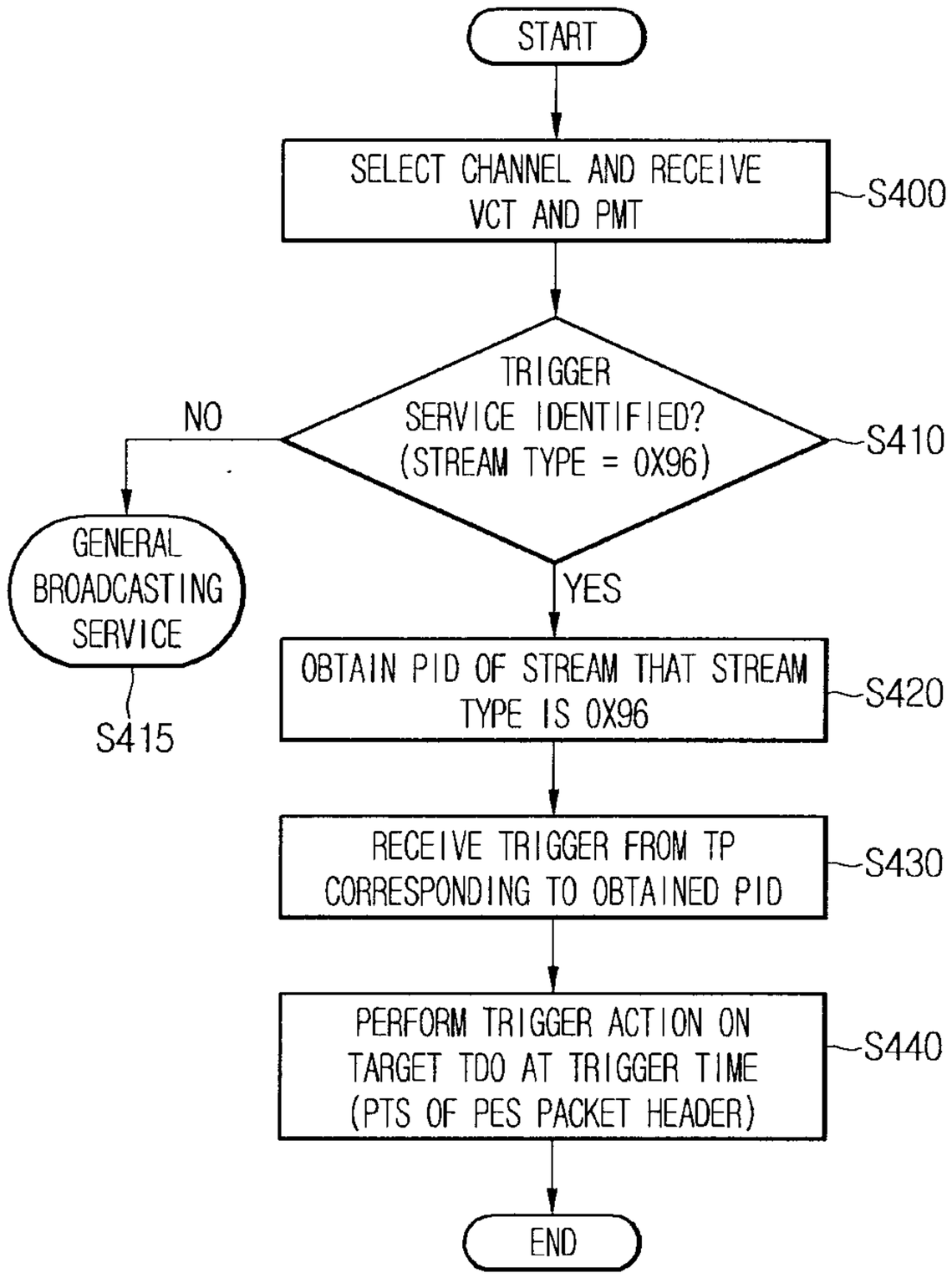

START
SELECT CHANNEL AND RECEIVE VCT AND PMT
S400
TRIGGER SERVICE IDENTIFIED? (STREAM TYPE = 0X96)
S410
NO
YES
GENERAL BROADCASTING SERVICE
S415
OBTAIN PID OF STREAM THAT STREAM TYPE IS 0X96
S420
RECEIVE TRIGGER FROM TP CORRESPONDING TO OBTAINED PID
S430
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME (PTS OF PES PACKET HEADER)
S440
END

FIG.41

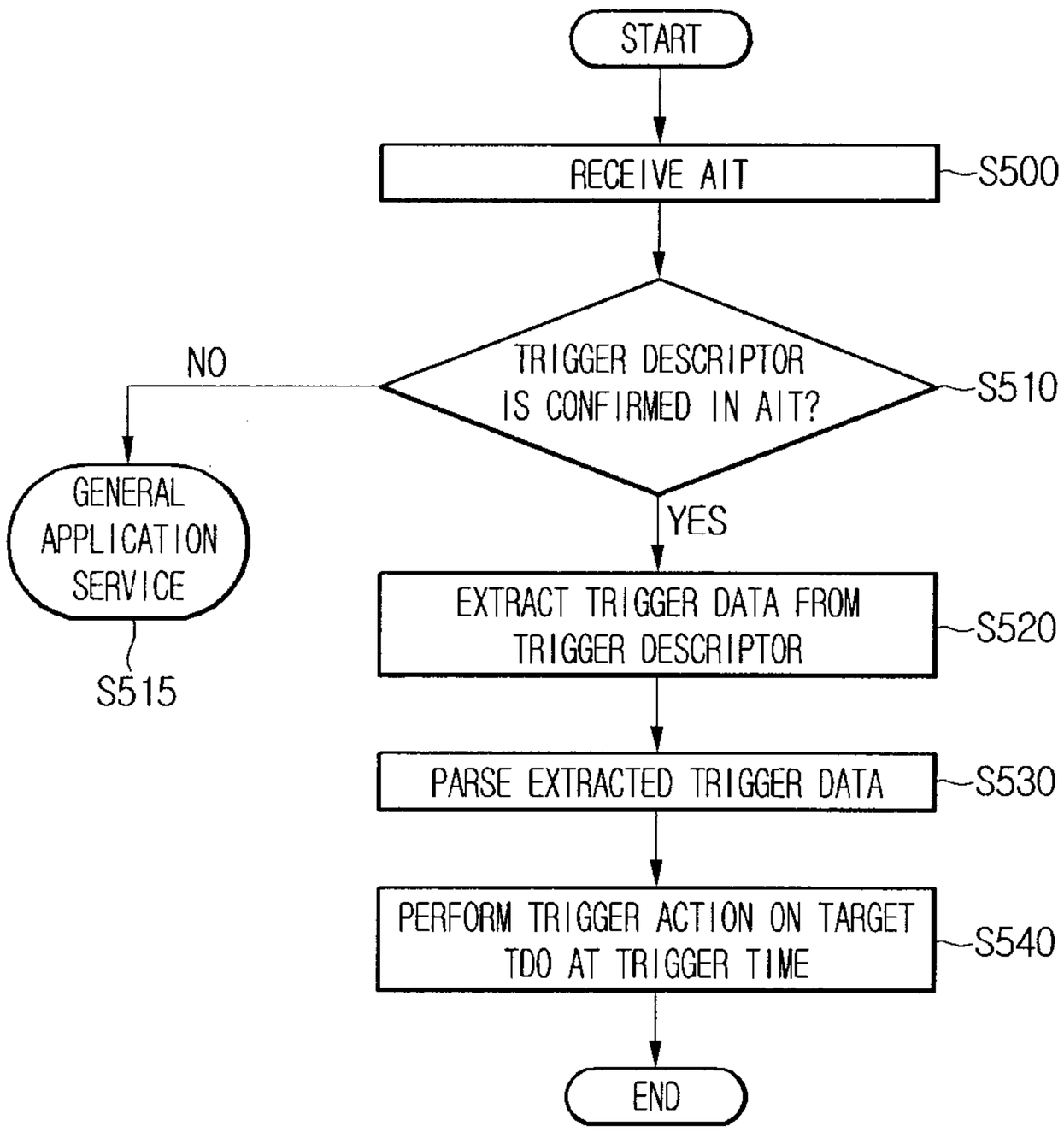

START
RECEIVE AIT
S500
TRIGGER DESCRIPTOR IS CONFIRMED IN AIT?
S510
NO
YES
GENERAL APPLICATION SERVICE
S515
EXTRACT TRIGGER DATA FROM TRIGGER DESCRIPTOR
S520
PARSE EXTRACTED TRIGGER DATA
S530
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME
S540
END

FIG.42

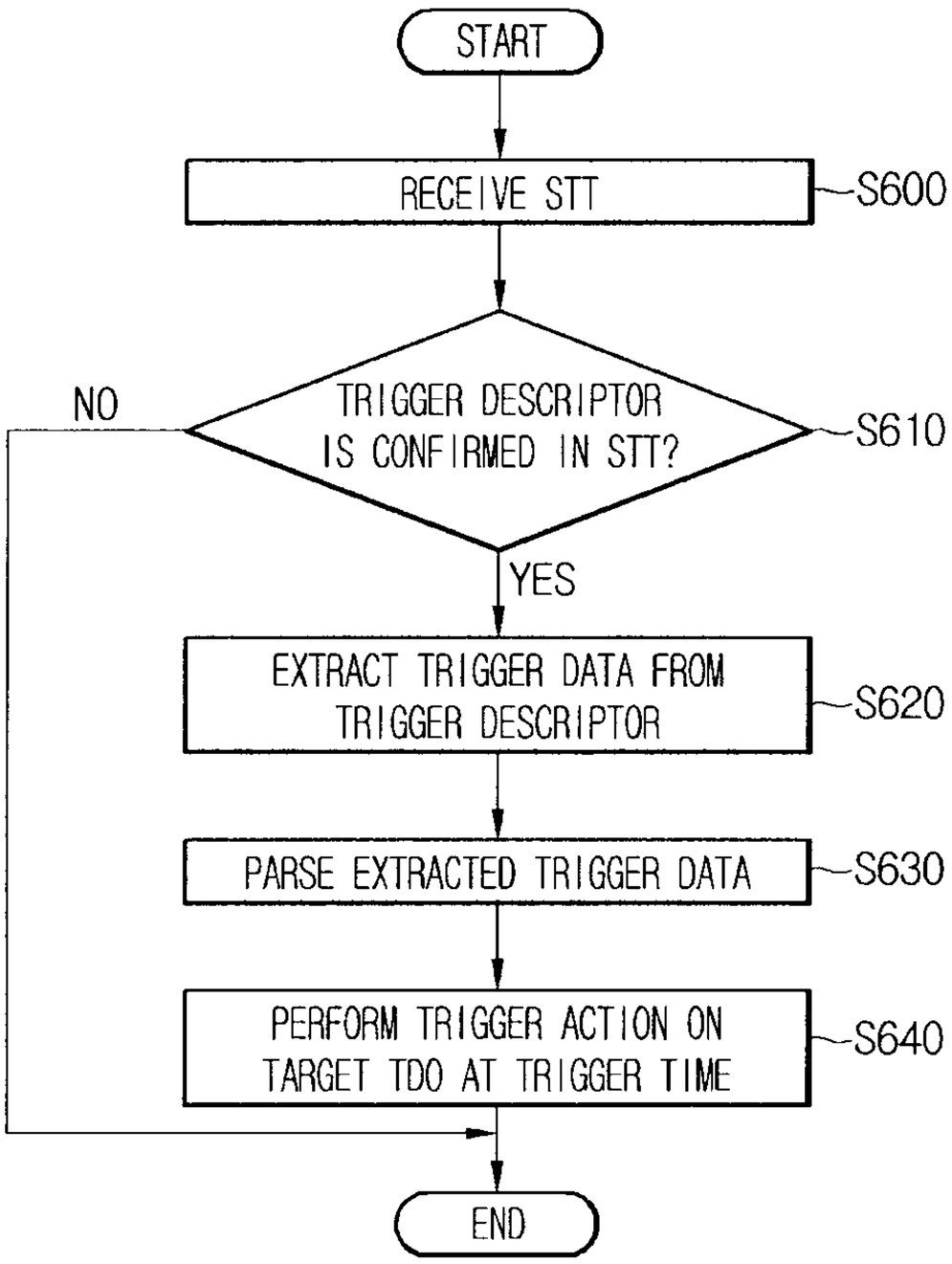

START
RECEIVE STT
S600
TRIGGER DESCRIPTOR IS CONFIRMED IN STT?
S610
NO
YES
EXTRACT TRIGGER DATA FROM TRIGGER DESCRIPTOR
S620
PARSE EXTRACTED TRIGGER DATA
S630
PERFORM TRIGGER ACTION ON TARGET TDO AT TRIGGER TIME
S640
END

FIG.43

link_descriptor() {		
descriptor_tag	8	0xe8
descriptor_length	8	uimsbf
number_of_links	8	uimsbf
for (i=0; i<number_of_links; i++) {		
link_type	8	uimsbf
link_media	8	uimsbf
mime_type_length	8	uimsbf
mime_type	var	
description_length	8	uimsbf
description	var	
link_length	uimsbf	
link_byte	var	
}		
}		

FIG.44

	link_type
Html Portal	0x01
Thumbnail	0x02
Preview Clip	0x03
EPG	0x04
Highlight	0x05
MultiView	0x06
TDO	0x07

FIG.45

	link_media	link_byte
NRT service	0x02	service_id content_linkage
Internet service	0x03	URI

FIG.46

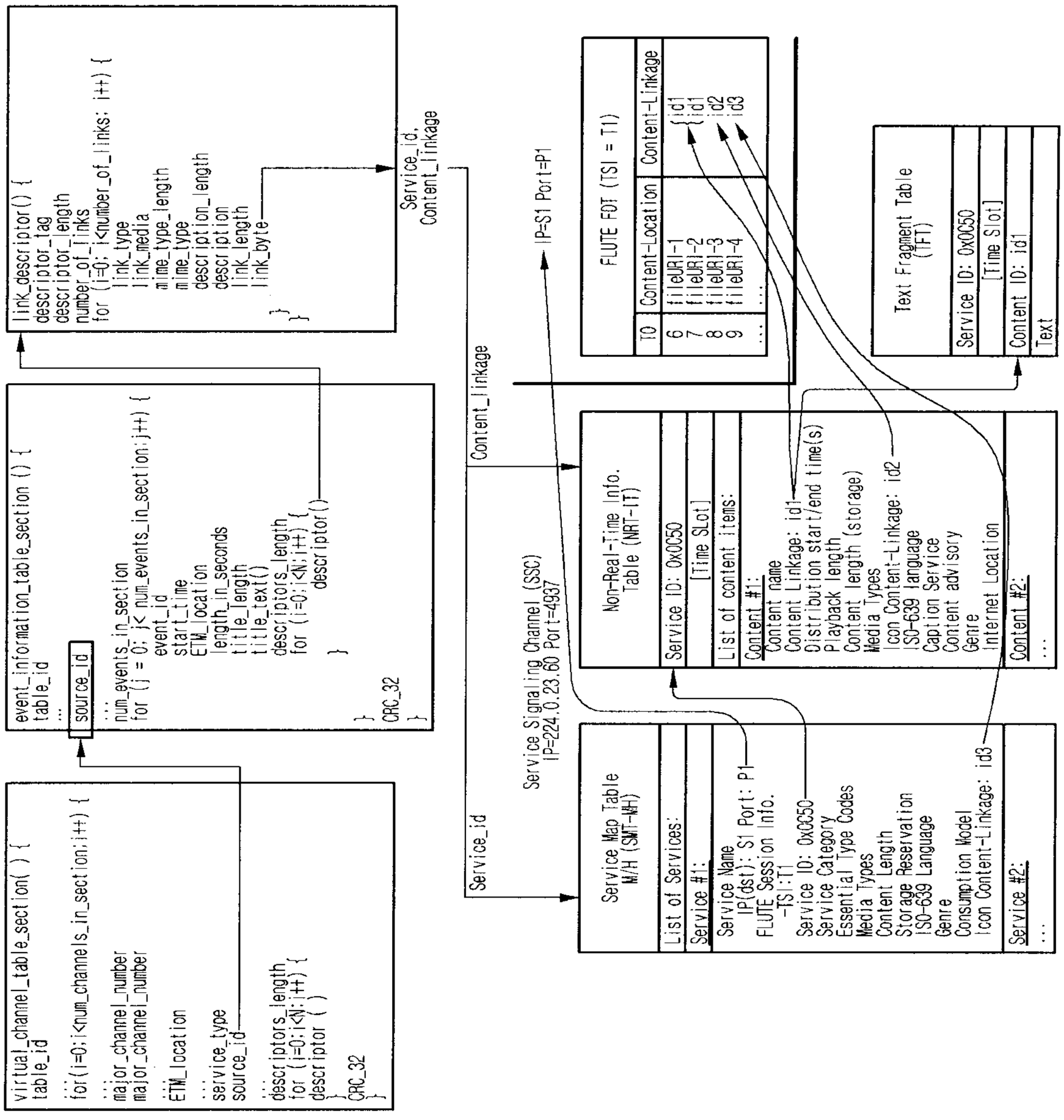

link_descriptor() {
descriptor_tag
descriptor_length
number_of_links
for (i=0; i<number_of_links; i++) {
link_type
link_media
mime_type_length
mime_type
description_length
description
link_length
link_byte
}
}
Service_id,
Content_Linkage
event_information_table_section () {
table_id
...
source_id
...
num_events_in_section
for (j = 0; j< num_events_in_section;j++) {
event_id
start_time
ETM_location
length_in_seconds
title_length
title_text()
descriptors_length
for (i=0;i<N;i++) {
descriptor()
}
}
CRC_32
}
virtual_channel_table_section( ) {
table_id
...
for(i=0;i<num_channels_in_section;i++) {
...
major_channel_number
major_channel_number
...
ETM_location
...
service_type
source_id
...
descriptors_length
for (i=0;i<N;i++) {
descriptor ( )
}
CRC_32
}
IP=S1 Port=P1
Content_Linkage
FLUTE FDT (TSI = T1)
TO
Content-Location
Content-Linkage
6 fileURI-1 {id1
7 fileURI-2 {id1
8 fileURI-3 id2
9 fileURI-4 id3
....
Text Fragment Table
(TFT)
Service ID: 0x0C50
[Time Slot]
Content ID: id1
Text
Service Signaling Channel (SSC)
IP=224.0.23.60 Port=4937
Non-Real-Time Info.
Table (NRT-IT)
Service ID: 0x0C50
[Time SLot]
List of content items:
Content #1:
Content name
Content Linkage: id1
Distribution start/end time(s)
Playback length
Content length (storage)
Media Types
Icon Content-Linkage: id2
ISO-639 language
Caption Service
Content advisory
Genre
Internet Location
Content #2:
...
Service_id
Service Map Table
M/H (SMT-MH)
List of Services:
Service #1:
Service Name
IP(dst): S1 Port: P1
FLUTE Session Info.
-TSI:T1
Service ID: 0x0C50
Service Category
Essential Type Codes
Media Types
Content Length
Storage Reservation
ISO-639 Language
Genre
Consumption Model
Icon Content-Linkage: id3
Service #2:
...

FIG.47

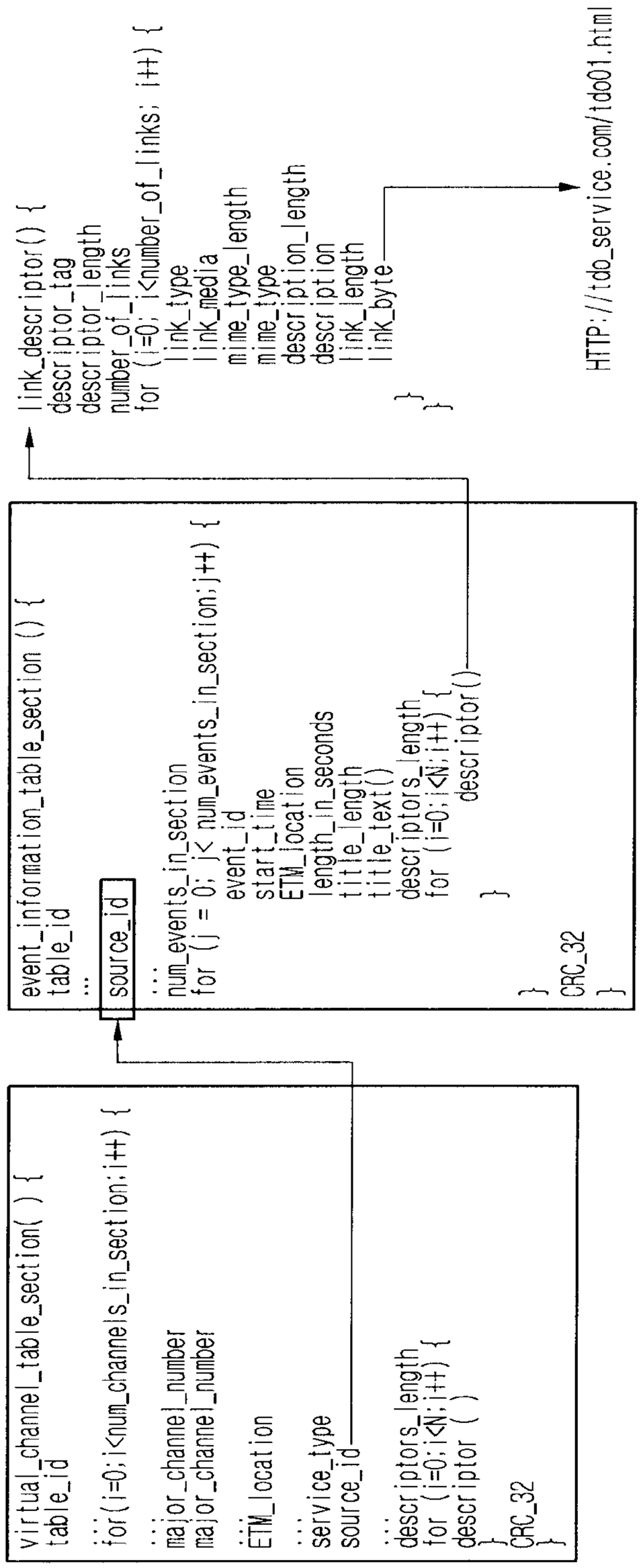

event_information_table_section () {
table_id
...
source_id
...
num_events_in_section
for (j = 0; j< num_events_in_section;j++) {
event_id
start_time
ETM_location
length_in_seconds
title_length
title_text()
descriptors_length
for (i=0;i<N;i++) {
descriptor()
}
}
CRC_32
}
virtual_channel_table_section( ) {
table_id
for(i=0;i<num_channels_in_section;i++) {
major_channel_number
major_channel_number
ETM_location
service_type
source_id
descriptors_length
for (i=0;i<N;i++) {
descriptor ( )
}
CRC_32
}
HTTP://tdo_service.com/tdo01.html
link_type
link_media
mime_type_length
mime_type
description_length
description
link_length
link_byte
}
}

FIG.48

```
Event_descriptor() {
  descriptor_tag                8          0xe9
  descriptor_length             8          uimsbf
  ETM_id                        32
}
```

	MSB			LSB
Bit	31 ··· 16	15 ··· 2		1 0
Channel ETM_id	source_id	0 ··· 0		0 0
Event ETM_id	source_id	event_id		1 0

FIG.49

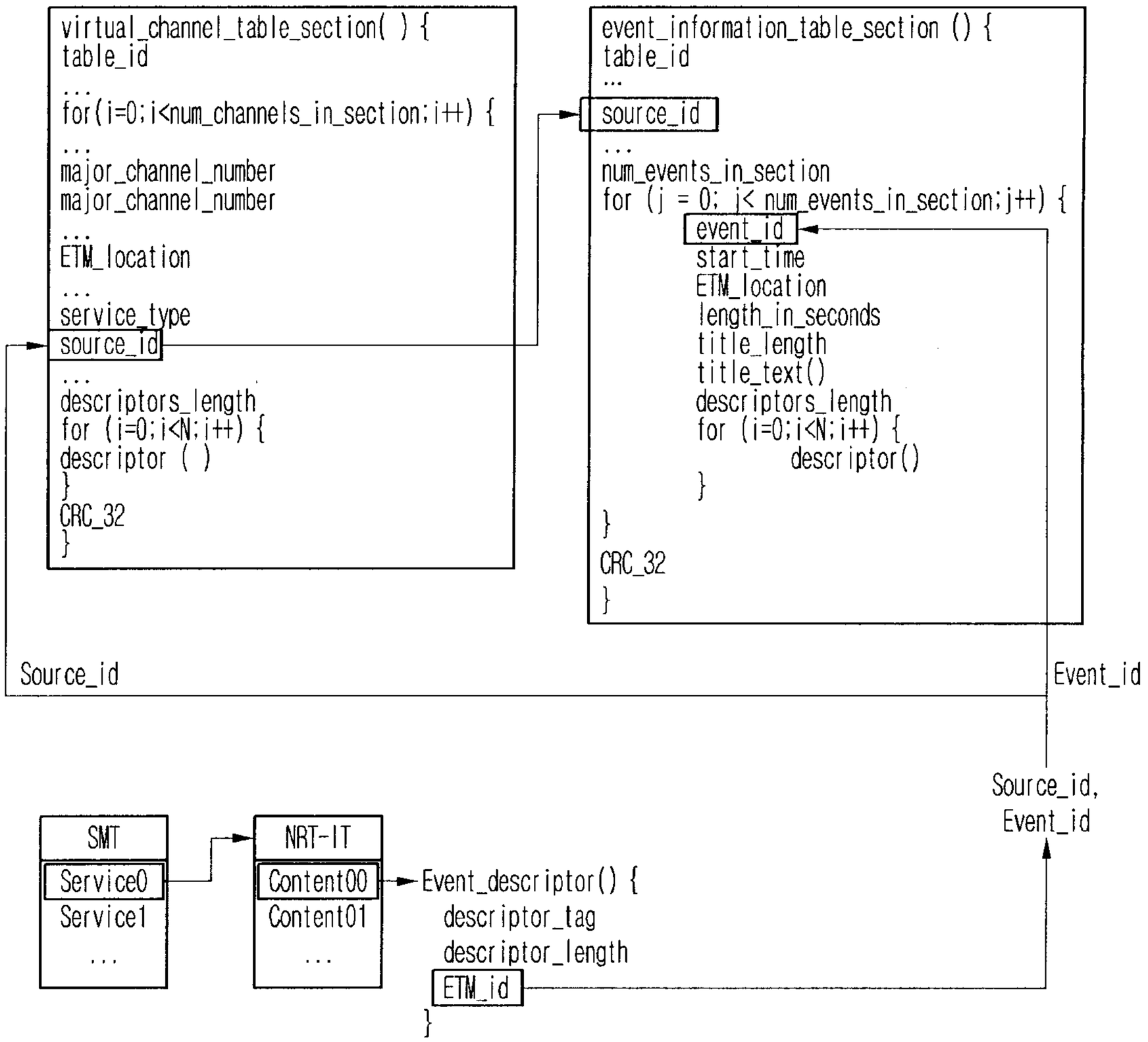

virtual_channel_table_section( ) {
table_id
...
for(i=0;i<num_channels_in_section;i++) {
...
major_channel_number
major_channel_number
...
ETM_location
...
service_type
source_id
...
descriptors_length
for (i=0;i<N;i++) {
descriptor ( )
}
CRC_32
}
event_information_table_section () {
table_id
...
source_id
...
num_events_in_section
for (j = 0; j< num_events_in_section;j++) {
event_id
start_time
ETM_location
length_in_seconds
title_length
title_text()
descriptors_length
for (i=0;i<N;i++) {
descriptor()
}
}
CRC_32
}
Source_id
Event_id
Source_id,
Event_id
SMT
Service0
Service1
...
NRT-IT
Content00
Content01
...
Event_descriptor() {
descriptor_tag
descriptor_length
ETM_id
}

FIG.50

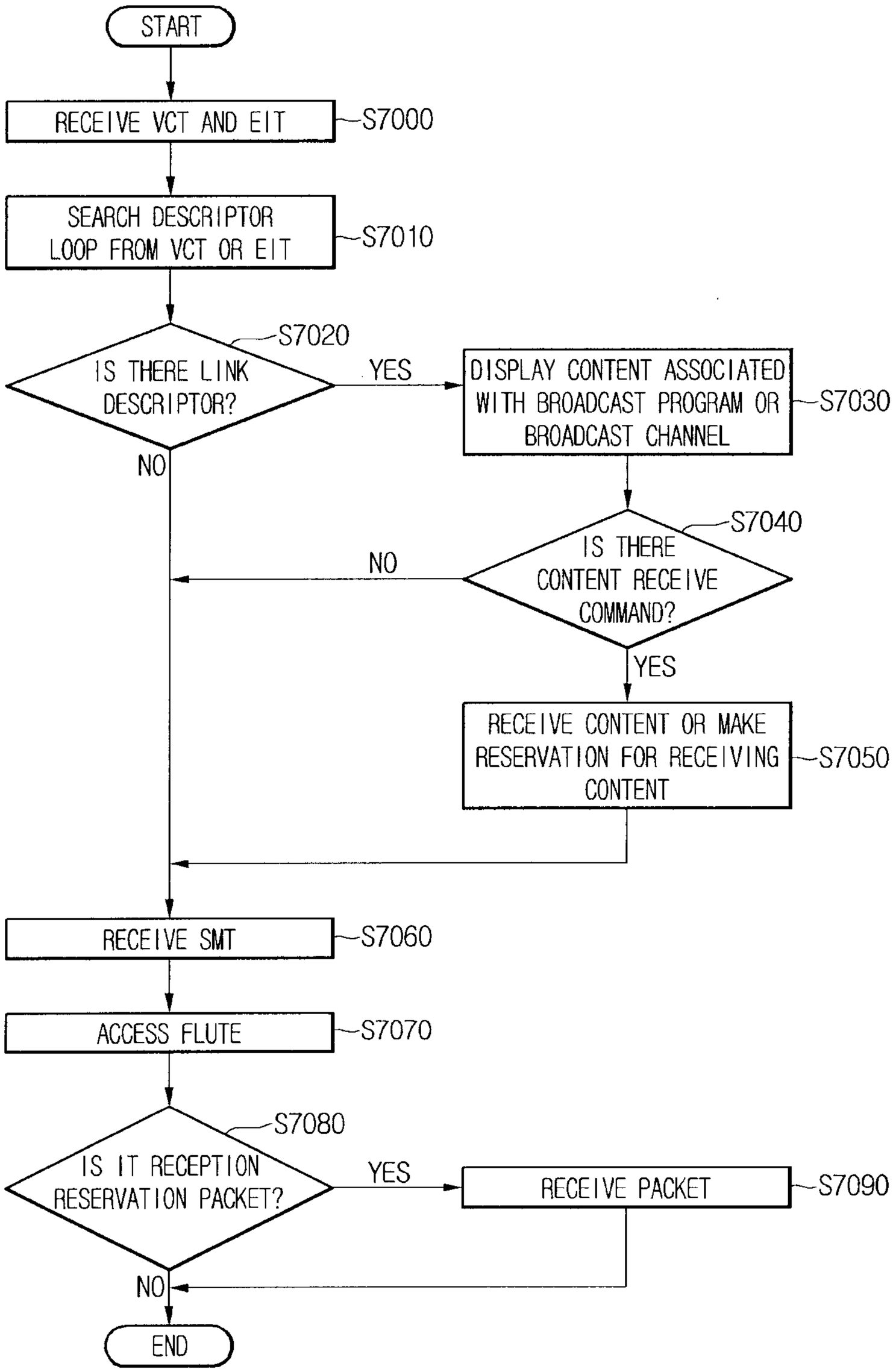

START
RECEIVE VCT AND EIT
S7000
SEARCH DESCRIPTOR LOOP FROM VCT OR EIT
S7010
S7020
IS THERE LINK DESCRIPTOR?
YES
DISPLAY CONTENT ASSOCIATED WITH BROADCAST PROGRAM OR BROADCAST CHANNEL
S7030
NO
S7040
IS THERE CONTENT RECEIVE COMMAND?
NO
YES
RECEIVE CONTENT OR MAKE RESERVATION FOR RECEIVING CONTENT
S7050
RECEIVE SMT
S7060
ACCESS FLUTE
S7070
S7080
IS IT RECEPTION RESERVATION PACKET?
YES
RECEIVE PACKET
S7090
NO
END

FIG.51

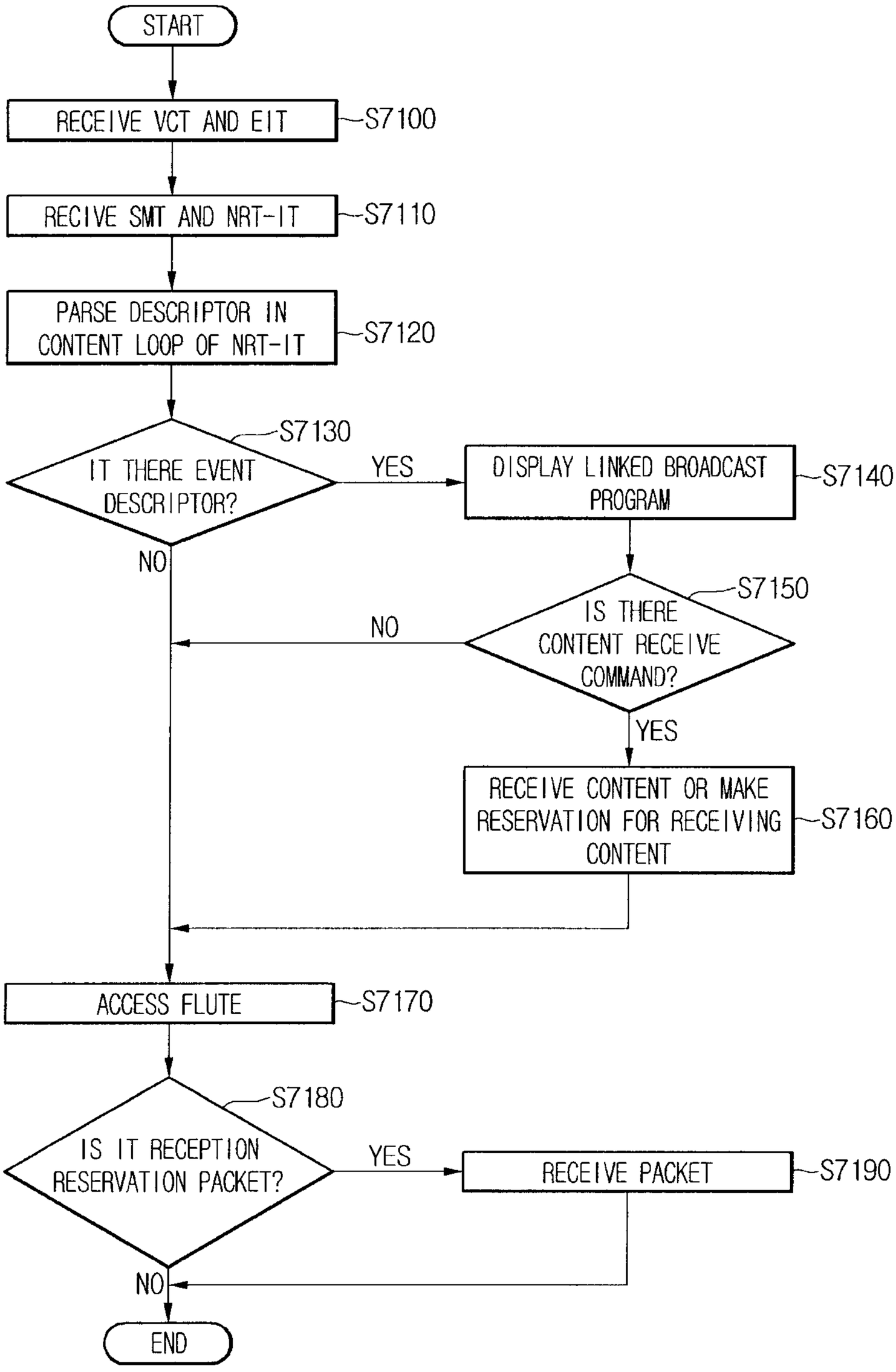

START
RECEIVE VCT AND EIT
S7100
RECIVE SMT AND NRT-IT
S7110
PARSE DESCRIPTOR IN CONTENT LOOP OF NRT-IT
S7120
S7130
IT THERE EVENT DESCRIPTOR?
YES
DISPLAY LINKED BROADCAST PROGRAM
S7140
NO
S7150
IS THERE CONTENT RECEIVE COMMAND?
NO
YES
RECEIVE CONTENT OR MAKE RESERVATION FOR RECEIVING CONTENT
S7160
ACCESS FLUTE
S7170
S7180
IS IT RECEPTION RESERVATION PACKET?
YES
RECEIVE PACKET
S7190
NO
END

FIG.52

Syntax	No. of Bits	Format
NRT_service_descriptor() {		
descriptor_tag	8	0xC4
descriptor_length	8	uimsbf
reserved	2	'11'
consumption_model	6	uimsbf
auto-update	1	bslbf
storage_reservation_present	1	bslbf
default_content_length_present	1	bslbf
reserved	5	'11111'
if (storage_reservation_present==1) {		
storage_reservation	24	uimsbf
if (default_content_length_present==1) {		
default_content_length	32	uimsbf
}		
for (j=0; j< N; j++) {		bslbf
reserved	8	
}		
}		

FIG.53

Consumption_model	Meaning
0x00	Forbidden.
0x01	Browse & Download - The NRT service describes content that can be selected for later download.
0x02	Portal - The NRT service provides an experience similar to a web browser access. Files needed to support text/graphics rendering are available in the associated FLUTE session.
0x03	Push - The NRT service offers request-based content. Receivers are expected to offer the user a choice whether or not to automatically update content associated with the service. For such services, if the user selects the auto-update option, the receiver caches and service-related content and automatically updates files as new versions are made available. When the user returns to a requested Push service, content that had been pre-loaded is displayed.
0x04	TDO - The NRT service offers TDO(Triggered Declarative Object).
0x05 - 0x3F	Reserved for use by ATSC or other SDOs who register the use with ATSC.

FIG.54

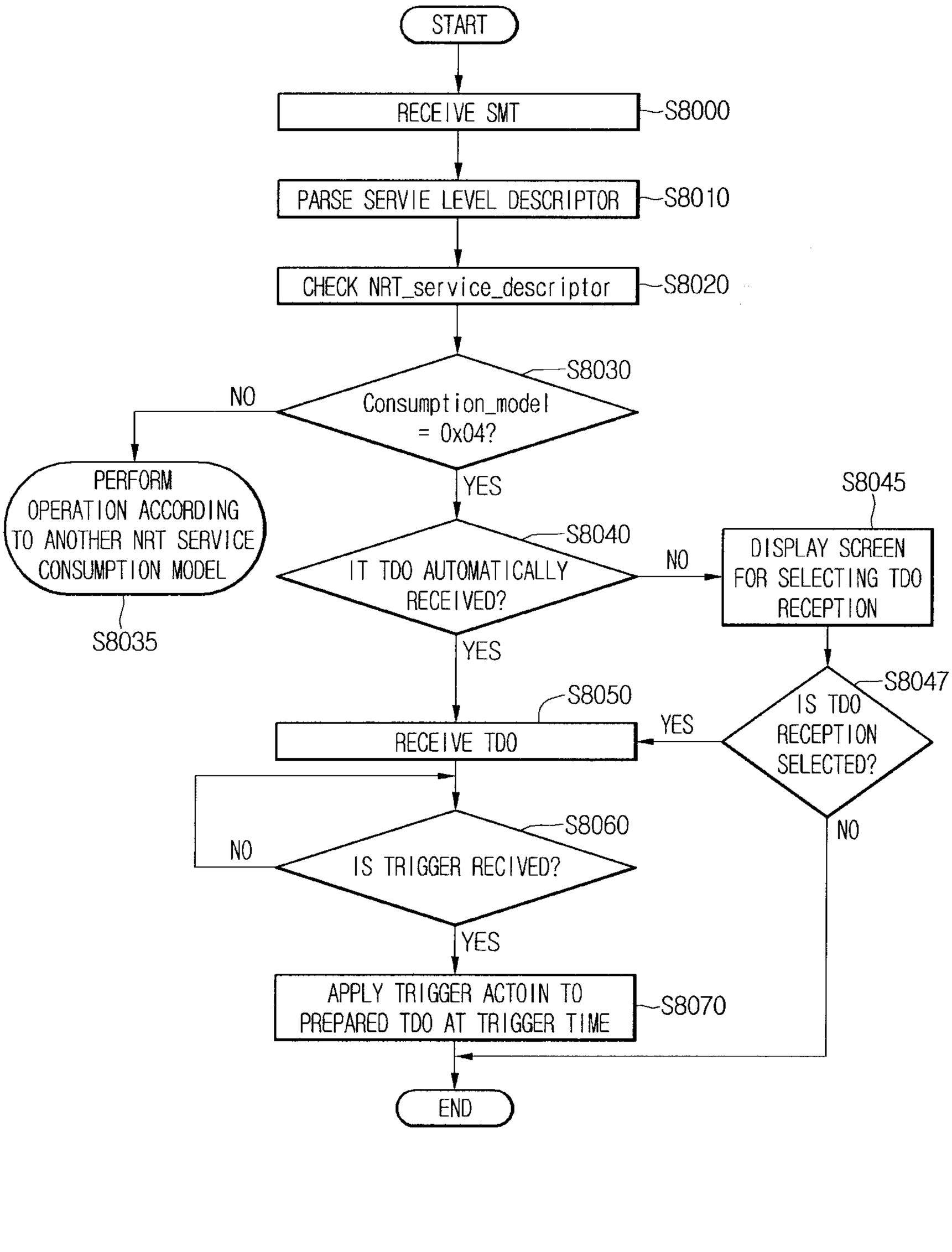

START
RECEIVE SMT
S8000
PARSE SERVIE LEVEL DESCRIPTOR
S8010
CHECK NRT_service_descriptor
S8020
S8030
Consumption_model = 0x04?
NO
YES
PERFORM OPERATION ACCORDING TO ANOTHER NRT SERVICE CONSUMPTION MODEL
S8035
S8040
IT TDO AUTOMATICALLY RECEIVED?
NO
YES
S8045
DISPLAY SCREEN FOR SELECTING TDO RECEPTION
S8047
IS TDO RECEPTION SELECTED?
YES
NO
S8050
RECEIVE TDO
S8060
IS TRIGGER RECIVED?
NO
YES
APPLY TRIGGER ACTOIN TO PREPARED TDO AT TRIGGER TIME
S8070
END

FIG.55

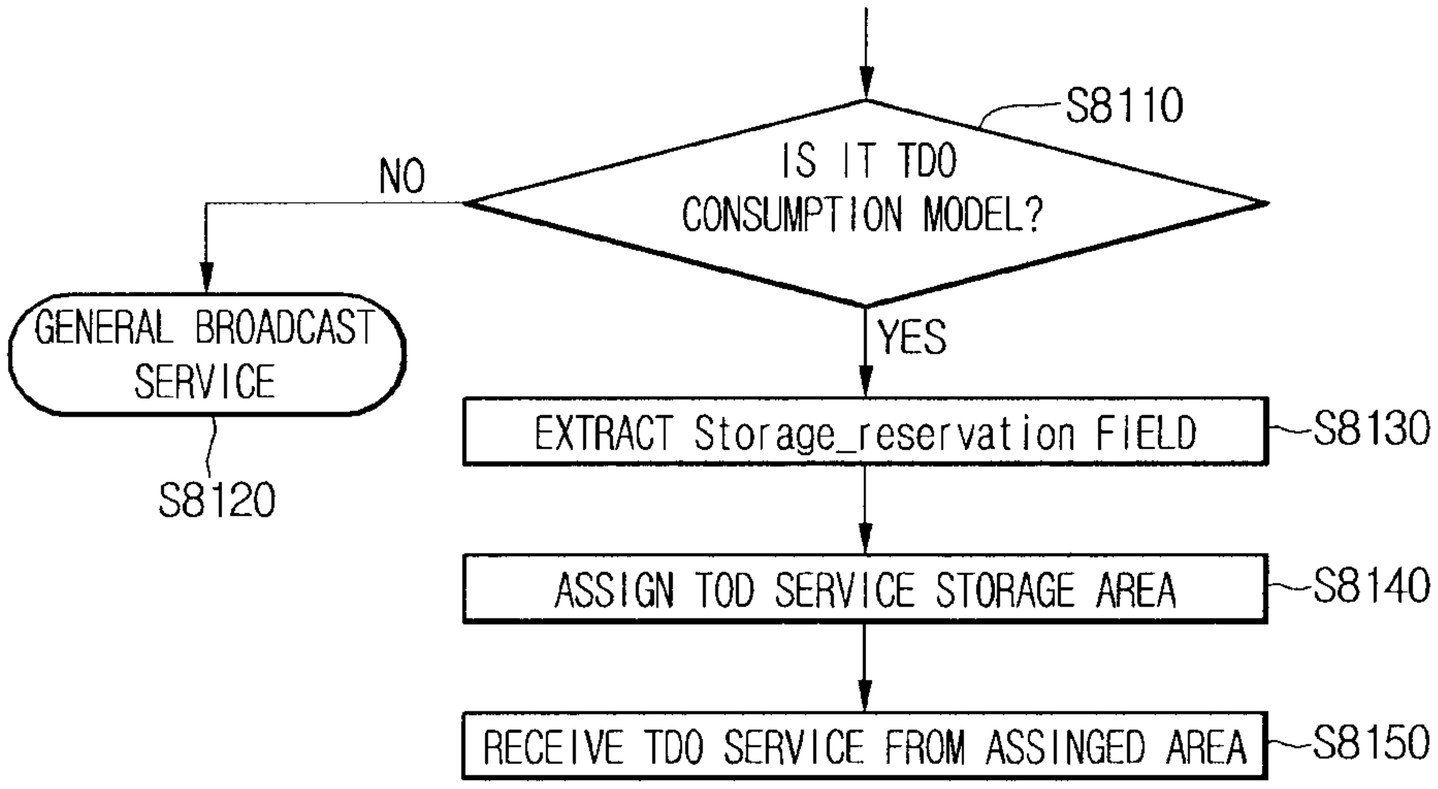

S8110
IS IT TDO CONSUMPTION MODEL?
NO
YES
GENERAL BROADCAST SERVICE
S8120
EXTRACT Storage_reservation FIELD
S8130
ASSIGN TOD SERVICE STORAGE AREA
S8140
RECEIVE TDO SERVICE FROM ASSINGED AREA
S8150

FIG.56

Syntax	No. of Bits	Format
TDO_metadata_descriptor() {		
descriptor_tag	8	0xTBD
descriptor_length	8	Uimsbf
scheduled_activation_start_time	32	uimsbf
scheduled_activation_end_time	32	uimsbf
priority	8	uimsbf
activation_repeat_flag	1	bslbf
reserved	7	'11111'
if (repeating==1) {		
repeat_interval	8	uimsbf
}		
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.57

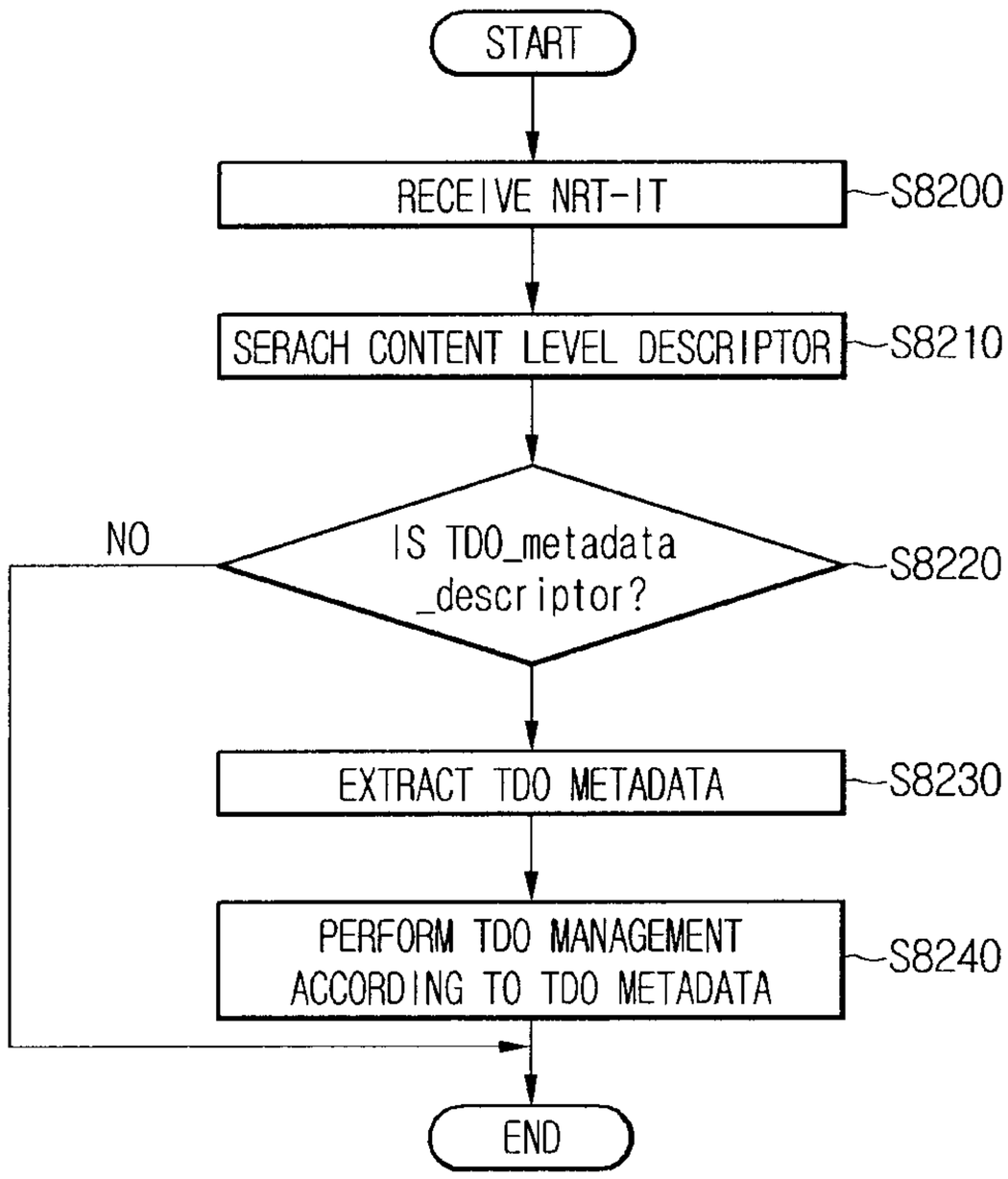

START
RECEIVE NRT-IT
S8200
SERACH CONTENT LEVEL DESCRIPTOR
S8210
IS TDO_metadata _descriptor?
S8220
NO
EXTRACT TDO METADATA
S8230
PERFORM TDO MANAGEMENT ACCORDING TO TDO METADATA
S8240
END

FIG.58

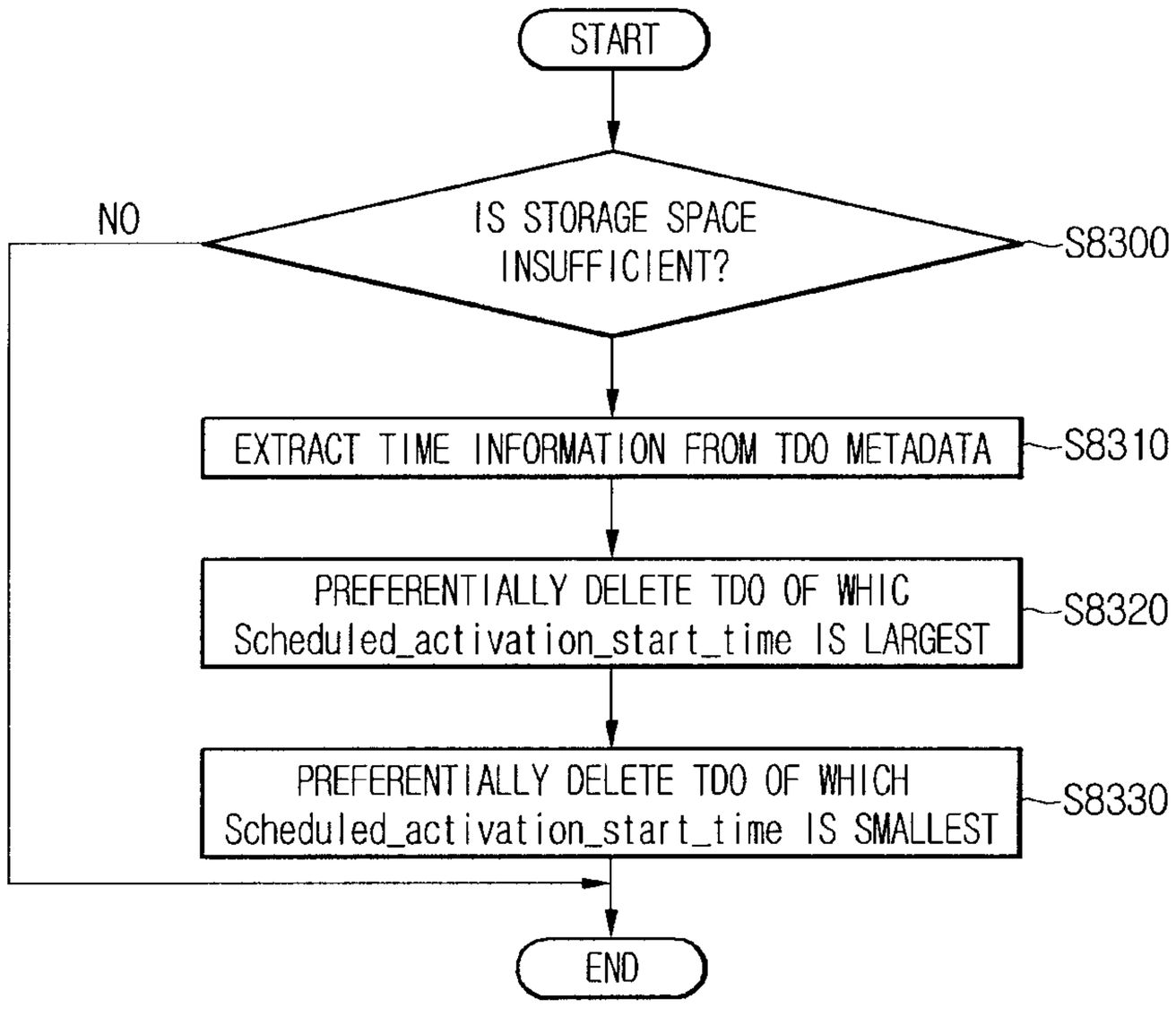

START
IS STORAGE SPACE INSUFFICIENT?
NO
S8300
EXTRACT TIME INFORMATION FROM TDO METADATA
S8310
PREFERENTIALLY DELETE TDO OF WHIC Scheduled_activation_start_time IS LARGEST
S8320
PREFERENTIALLY DELETE TDO OF WHICH Scheduled_activation_start_time IS SMALLEST
S8330
END

FIG.59

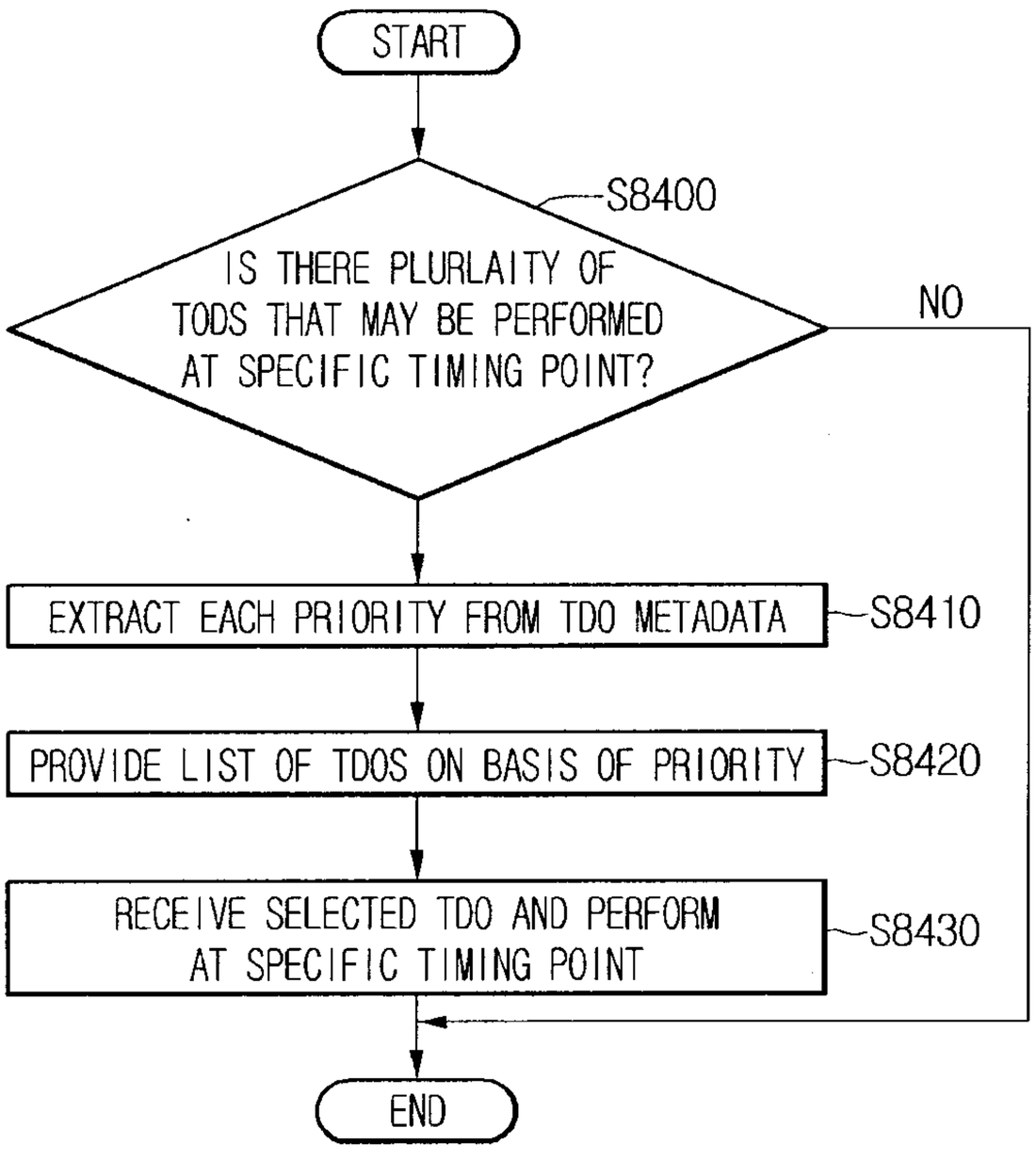

START
S8400
IS THERE PLURLAITY OF TODS THAT MAY BE PERFORMED AT SPECIFIC TIMING POINT?
NO
EXTRACT EACH PRIORITY FROM TDO METADATA
S8410
PROVIDE LIST OF TDOS ON BASIS OF PRIORITY
S8420
RECEIVE SELECTED TDO AND PERFORM AT SPECIFIC TIMING POINT
S8430
END

FIG.60

Syntax	No. of Bits	Format
internet_location_descriptor() {		
descriptor_tag	8	0xC9
descriptor_length	8	uimsbf
reserved	3	'111'
URL_count	5	uimsbf
for (i=0; i<URL_count; i++) {		
URL_length	8	uimsbf
URL()	var	
}		
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.61

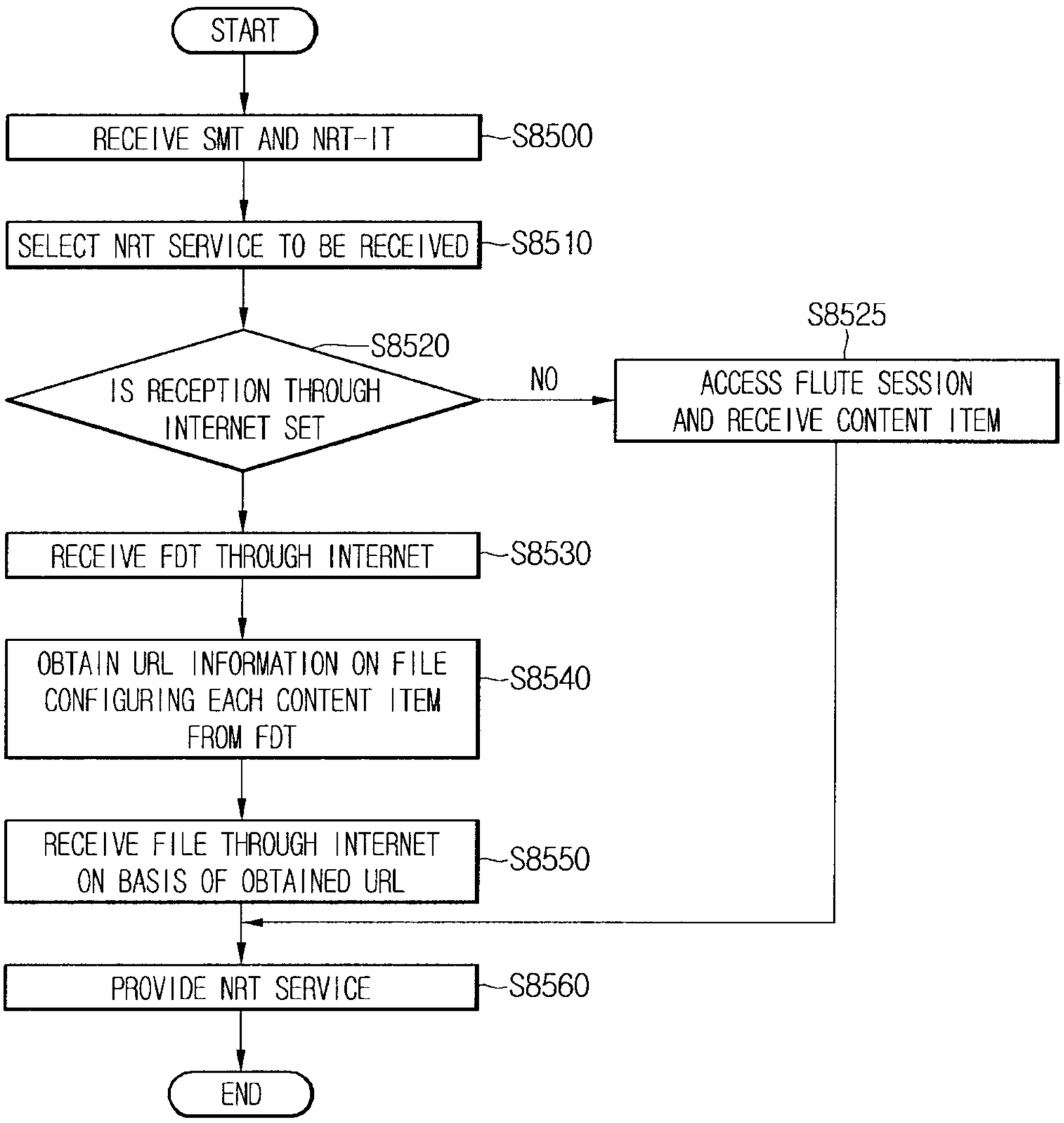

START
RECEIVE SMT AND NRT-IT
S8500
SELECT NRT SERVICE TO BE RECEIVED
S8510
S8520
IS RECEPTION THROUGH INTERNET SET
NO
S8525
ACCESS FLUTE SESSION AND RECEIVE CONTENT ITEM
RECEIVE FDT THROUGH INTERNET
S8530
OBTAIN URL INFORMATION ON FILE CONFIGURING EACH CONTENT ITEM FROM FDT
S8540
RECEIVE FILE THROUGH INTERNET ON BASIS OF OBTAINED URL
S8550
PROVIDE NRT SERVICE
S8560
END

FIG.62

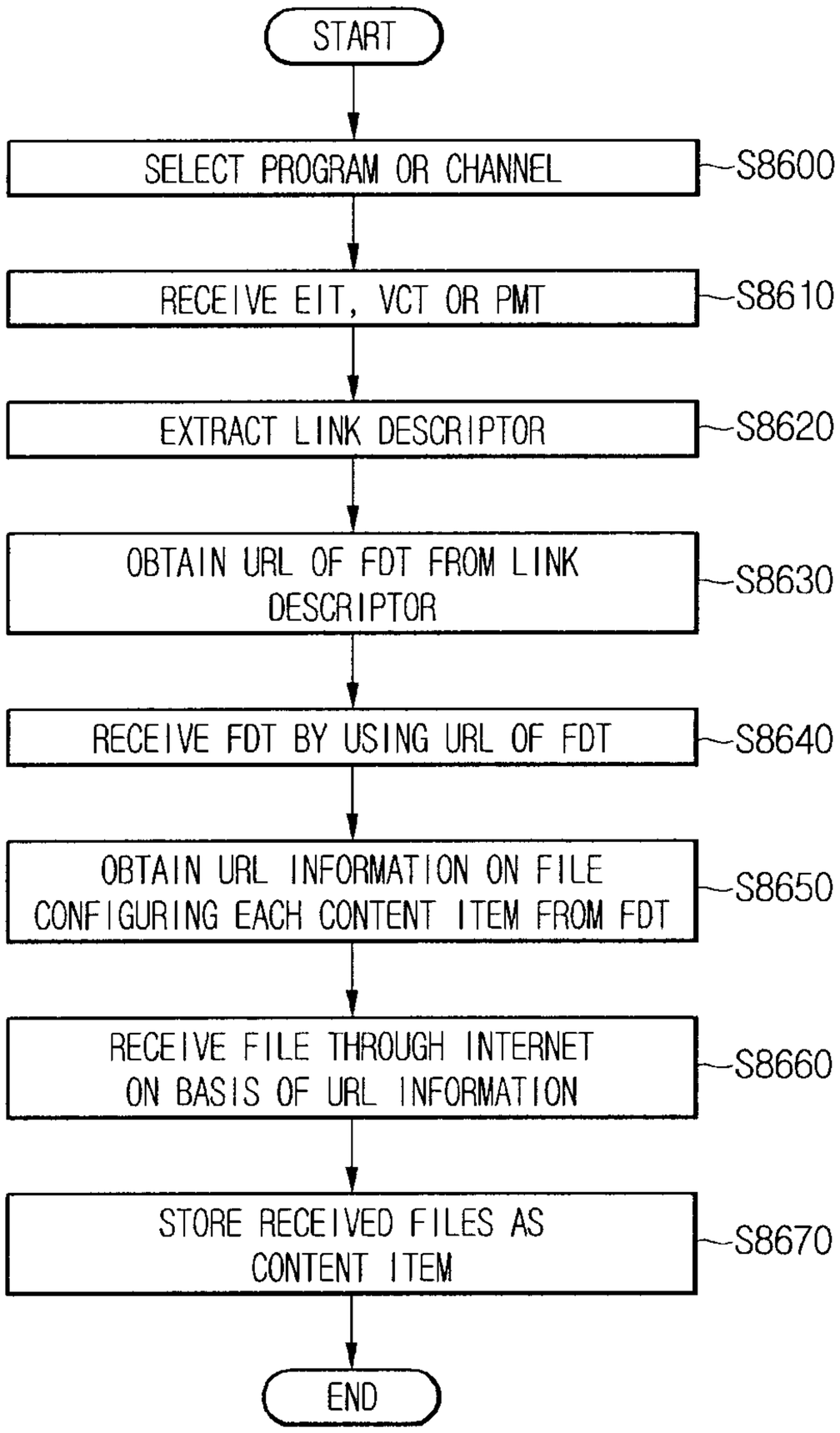

START
SELECT PROGRAM OR CHANNEL
S8600
RECEIVE EIT, VCT OR PMT
S8610
EXTRACT LINK DESCRIPTOR
S8620
OBTAIN URL OF FDT FROM LINK DESCRIPTOR
S8630
RECEIVE FDT BY USING URL OF FDT
S8640
OBTAIN URL INFORMATION ON FILE CONFIGURING EACH CONTENT ITEM FROM FDT
S8650
RECEIVE FILE THROUGH INTERNET ON BASIS OF URL INFORMATION
S8660
STORE RECEIVED FILES AS CONTENT ITEM
S8670
END

FIG.63

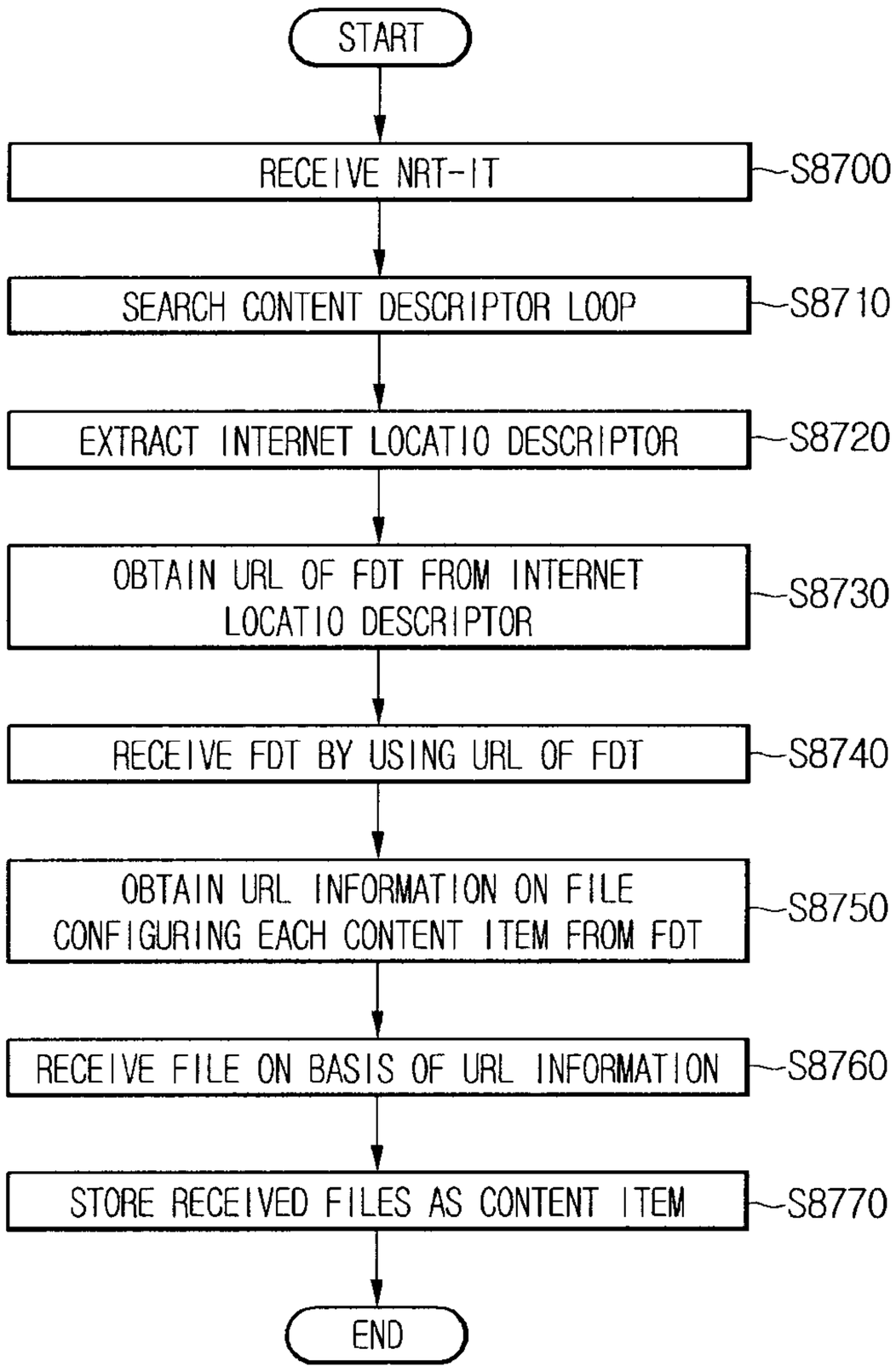

START
RECEIVE NRT-IT
S8700
SEARCH CONTENT DESCRIPTOR LOOP
S8710
EXTRACT INTERNET LOCATIO DESCRIPTOR
S8720
OBTAIN URL OF FDT FROM INTERNET LOCATIO DESCRIPTOR
S8730
RECEIVE FDT BY USING URL OF FDT
S8740
OBTAIN URL INFORMATION ON FILE CONFIGURING EACH CONTENT ITEM FROM FDT
S8750
RECEIVE FILE ON BASIS OF URL INFORMATION
S8760
STORE RECEIVED FILES AS CONTENT ITEM
S8770
END

FIG.64

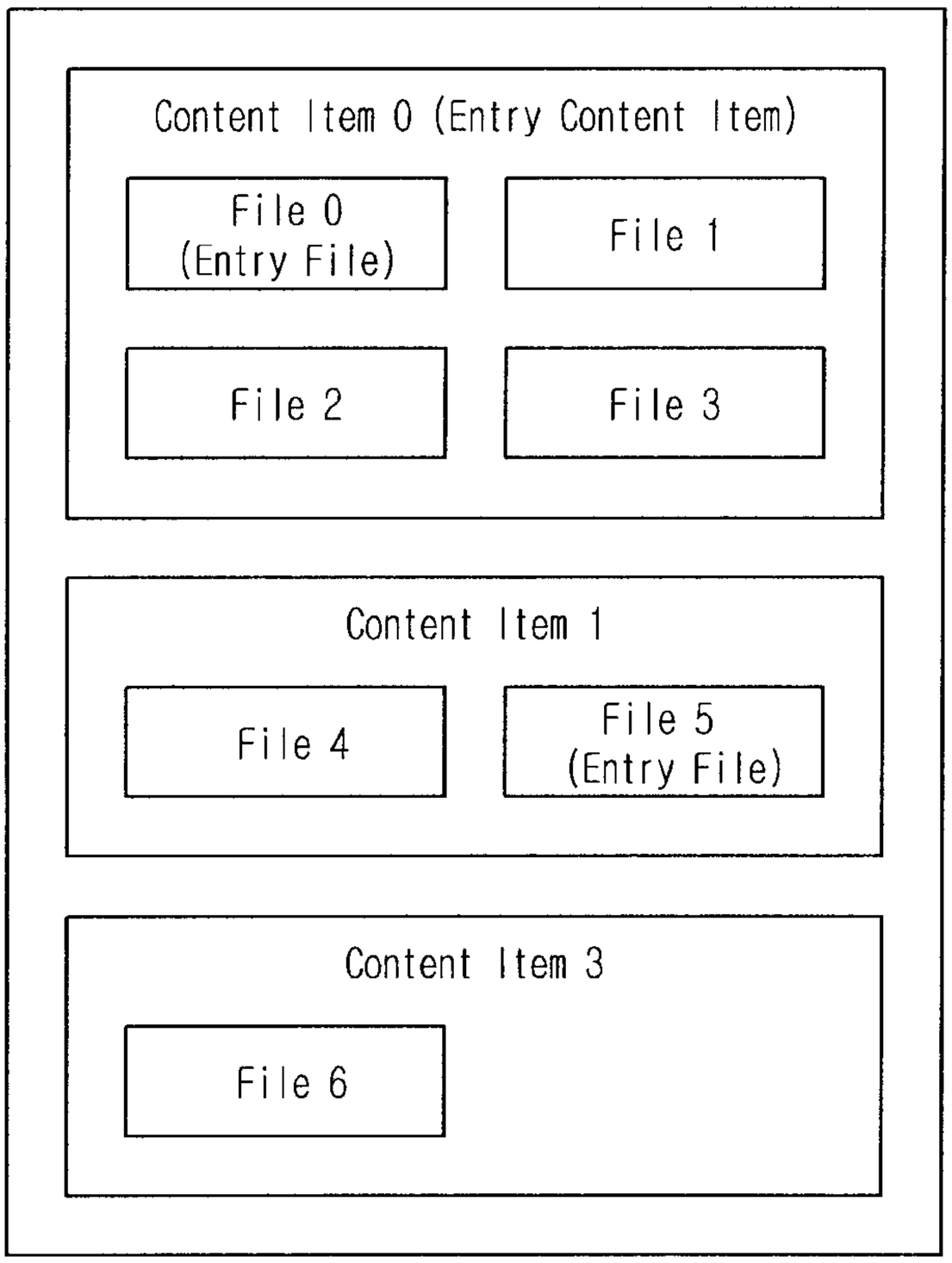

Content Item 0 (Entry Content Item)
File 0 (Entry File)
File 1
File 2
File 3
Content Item 1
File 4
File 5 (Entry File)
Content Item 3
File 6

FIG.69

Syntax	No. of Bits	Format
NRT_service_descriptor() {		
descriptor_tag	8	0xC4
descriptor_length	8	uimsbf
reserved	2	'11'
consumption_model	6	uimsbf
auto-update	1	bslbf
storage_reservation_present	1	bslbf
default_content_length_present	1	bslbf
equivalent_service_not_present	1	bslbf
reserved	4	'1111'
if (storage_reservation_present==1) {		
storage_reservation	24	uimsbf
if (default_content_length_present==1) {		
default_content_length	32	uimsbf
}		
if (equivalent_service_not_present==0) {		
num_equivalent_services	8	uimsbf
for (i=0; i< num_equivalent_services ; i++) {		
equivalent_service_id	16	uimsbf
}		
}		
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.65

Syntax	No. Bits	Format
NRT_information_table_section() {		
table_id	8	0xDF
section_syntax_indicator	1	'0'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension {		
protocol_version	8	uimsbf
subnet_id	8	uimsbf
}		
reserved	2	'11'
NRT_IT_version_number	5	uimsbf
current_next_indicator	1	'1'
section_number	8	uimsbf
last_section_number	8	uimsbf
service_id	16	uimsbf
time_span_start	32	uimsbf
reserved	5	'11111'
time_span_length	11	uimsbf
num_content_items_in_section	8	uimsbf
for (j=0; j< num_content_items_in_section; j++) {		
content_linkage	32	uimsbf
updates_available	1	bslbf
TF_available	1	bslbf
content_security_conditions_indicator	1	bslbf
available_on_internet	1	bslbf
playback_length_included	1	bslbf
playback_delay_included	1	bslbf
expiration_included	1	bslbf
content_length_included	1	bslbf
no_entry_flag	1	bslbf
reserved	3	'111'

FIG.66

acquisition_time	12	uimsbf
if (playback_length_included==1) {		
reserved	4	'1111'
playback_length_in_seconds	20	uimsbf
}		
if (playback_delay_included==1) {		
reserved	4	'1111'
playback_delay	20	uimsbf
}		
if (expiration_included==1) {		
expiration	32	uimsbf
}		
if (content_length_included==1) {		
content_length	40	uimsbf
}		
content_name_length	8	uimsbf
content_name_text()	var	
num_content_descriptors_length	8	uimsbf
for (i=0; i<num_content_descriptors; i++) {		
content_descriptor()	var	
}		
}		
num_descriptors	8	uimsbf
for (i=0; i<num_descriptors; i++) {		
descriptor()	var	
}		
}		

FIG.67

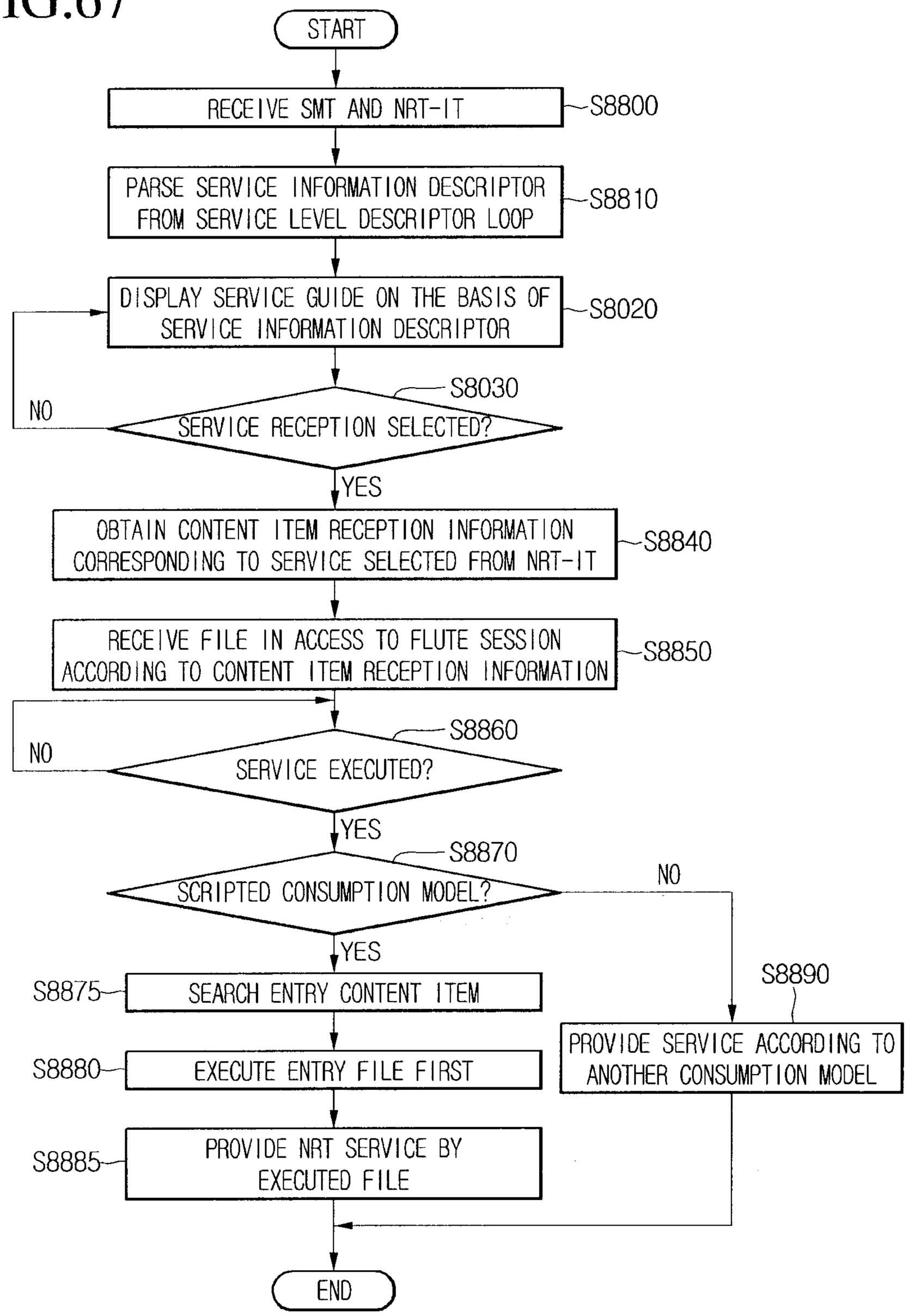

START
RECEIVE SMT AND NRT-IT
S8800
PARSE SERVICE INFORMATION DESCRIPTOR FROM SERVICE LEVEL DESCRIPTOR LOOP
S8810
DISPLAY SERVICE GUIDE ON THE BASIS OF SERVICE INFORMATION DESCRIPTOR
S8020
S8030
SERVICE RECEPTION SELECTED?
NO
YES
OBTAIN CONTENT ITEM RECEPTION INFORMATION CORRESPONDING TO SERVICE SELECTED FROM NRT-IT
S8840
RECEIVE FILE IN ACCESS TO FLUTE SESSION ACCORDING TO CONTENT ITEM RECEPTION INFORMATION
S8850
S8860
SERVICE EXECUTED?
NO
YES
S8870
SCRIPTED CONSUMPTION MODEL?
NO
YES
S8875
SEARCH ENTRY CONTENT ITEM
S8880
EXECUTE ENTRY FILE FIRST
S8885
PROVIDE NRT SERVICE BY EXECUTED FILE
S8890
PROVIDE SERVICE ACCORDING TO ANOTHER CONSUMPTION MODEL
END

FIG.68

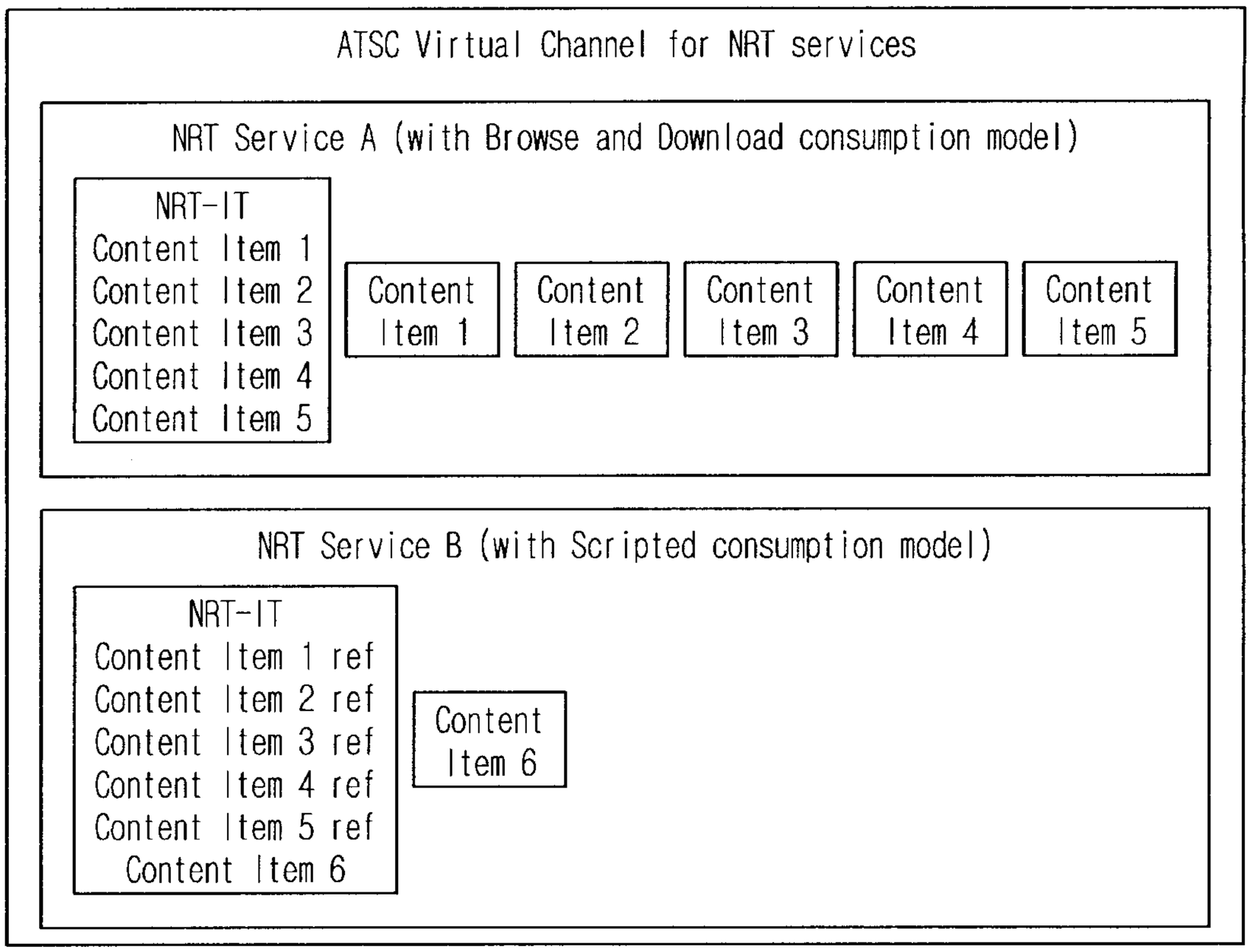

ATSC Virtual Channel for NRT services
NRT Service A (with Browse and Download consumption model)
NRT-IT
Content Item 1
Content Item 2
Content Item 3
Content Item 4
Content Item 5
Content Item 1
Content Item 2
Content Item 3
Content Item 4
Content Item 5
NRT Service B (with Scripted consumption model)
NRT-IT
Content Item 1 ref
Content Item 2 ref
Content Item 3 ref
Content Item 4 ref
Content Item 5 ref
Content Item 6
Content Item 6

FIG.70

Syntax	No. Bits	Format
NRT_information_table_section() {		
table_id	8	0xDF
section_syntax_indicator	1	'0'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension {		
protocol_version	8	uimsbf
subnet_id	8	uimsbf
}		
reserved	2	'11'
NRT_IT_version_number	5	uimsbf
current_next_indicator	1	'1'
section_number	8	uimsbf
last_section_number	8	uimsbf
service_id	16	uimsbf
time_span_start	32	uimsbf
reserved	5	'11111'
time_span_length	11	uimsbf
num_content_items_in_section	8	uimsbf
for (j=0; j< num_content_items_in_section; j++) {		
content_linkage	32	uimsbf
updates_available	1	bslbf
TF_available	1	bslbf
content_security_conditions_indicator	1	bslbf
available_on_internet	1	bslbf
playback_length_included	1	bslbf
playback_delay_included	1	bslbf
expiration_included	1	bslbf
content_length_included	1	bslbf
no_entry_flag	1	bslbf
available_on_current_service	1	bslbf
reserved	3	'111'

FIG.71

reserved	3	'111'
acquisition_time	12	uimsbf
if (playback_length_included==1) {		
reserved	4	'1111'
playback_length_in_seconds	20	uimsbf
}		
if (playback_delay_included==1) {		
reserved	4	'1111'
playback_delay	20	uimsbf
}		
if (expiration_included==1) {		
expiration	32	uimsbf
}		
if (content_length_included==1) {		
content_length	40	uimsbf
}		
content_name_length	8	uimsbf
content_name_text()	var	
num_content_descriptors_length	8	uimsbf
for (i=0; i<num_content_descriptors; i++) {		
content_descriptor()	var	
}		
}		
num_descriptors	8	uimsbf
for (i=0; i<num_descriptors; i++) {		
descriptor()	var	
}		
}		

FIG.72

Syntax	No. Bits	Format
Other_NRT_location_descriptor() {		
descriptor_tag	8	TBD
descriptor_length	8	uimsbf
reserved	2	'11'
num_other_NRT_locations	6	uimsbf
for (i=0; i< num_other_NRT_locations ; i++) {		
other_service_id	16	uimsbf
}		
}		

FIG.73

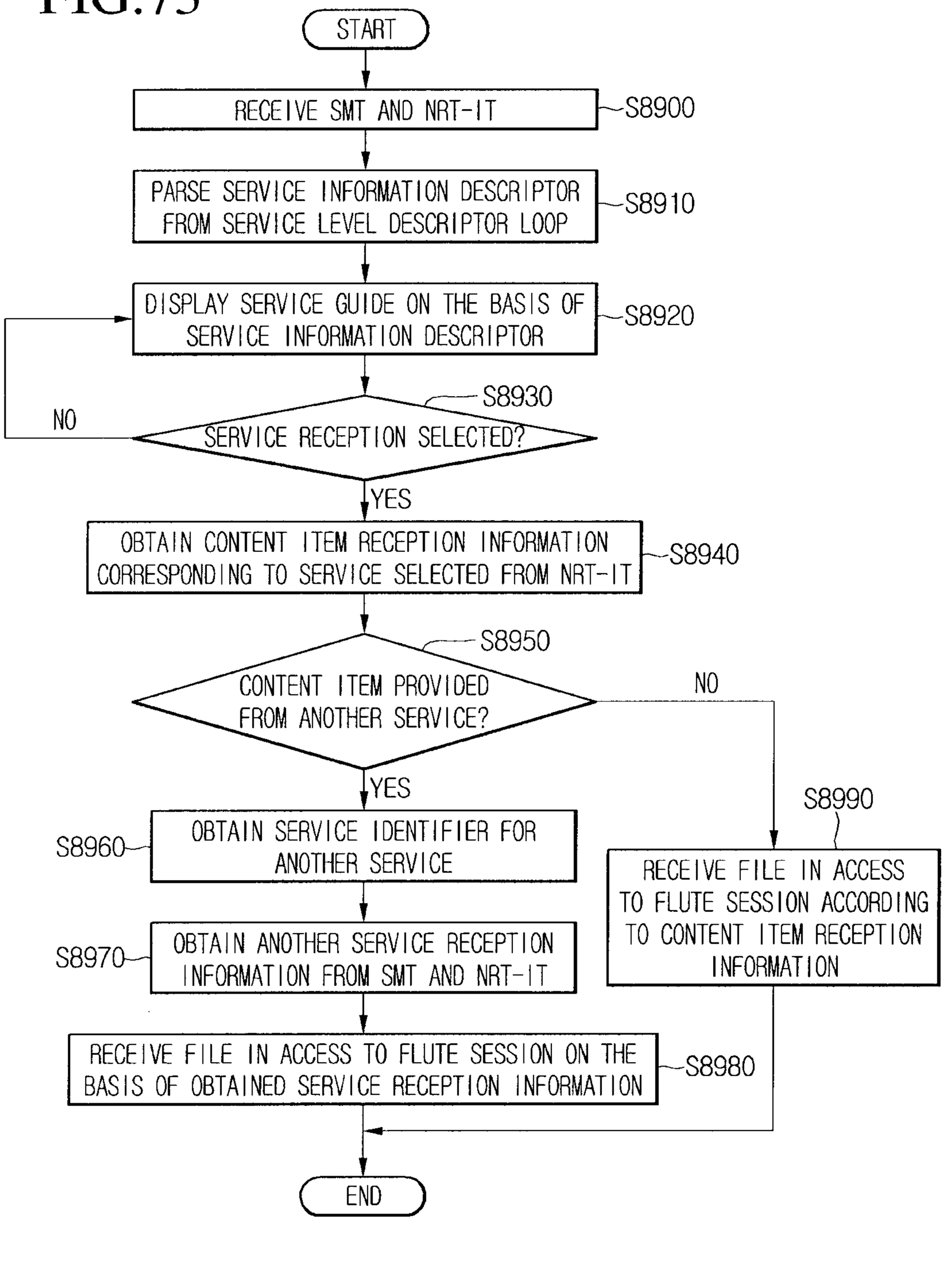

START
RECEIVE SMT AND NRT-IT
S8900
PARSE SERVICE INFORMATION DESCRIPTOR FROM SERVICE LEVEL DESCRIPTOR LOOP
S8910
DISPLAY SERVICE GUIDE ON THE BASIS OF SERVICE INFORMATION DESCRIPTOR
S8920
S8930
SERVICE RECEPTION SELECTED?
NO
YES
OBTAIN CONTENT ITEM RECEPTION INFORMATION CORRESPONDING TO SERVICE SELECTED FROM NRT-IT
S8940
S8950
CONTENT ITEM PROVIDED FROM ANOTHER SERVICE?
NO
YES
S8960
OBTAIN SERVICE IDENTIFIER FOR ANOTHER SERVICE
S8970
OBTAIN ANOTHER SERVICE RECEPTION INFORMATION FROM SMT AND NRT-IT
S8990
RECEIVE FILE IN ACCESS TO FLUTE SESSION ACCORDING TO CONTENT ITEM RECEPTION INFORMATION
RECEIVE FILE IN ACCESS TO FLUTE SESSION ON THE BASIS OF OBTAINED SERVICE RECEPTION INFORMATION
S8980
END

FIG.74

companion_device_descriptor(){	
descriptor_tag	8 TBD
descriptor_length	8 uimsbf
companion_id	32 uimsbf
action_type	8 uimsbf
number_of_filters	8 uimsbf
for(i=0;i<number_of_filters;i++){	
user_agent_filter_length	8 uimsbf
user_agent_filter	var
version_length	8 uimsbf
version	var
}	
description	multiplestring
URL_length	8 uimsbf
URL	var
}	

FIG.75

```
companion_device_action_descriptor() {
 descriptor_tag                                  8 TBD
 descriptor_length                               8 uimsbf
 companion_id                                    32 uimsbf
 action_type                                     8 uimsbf
 if ( action_type == EVENT )
 {
  event_length
  event                                          8 uimsbf
  }                                              var
}
```

FIG.76

action type		
AUTOSTART	0x00	IMMEDIATELY START CORRESPONDING APP
USERSTART	0x01	START BY USER
SHOW	0x02	START CORRESPONDING APP
READY	0x03	WAIT WITHOUT EXECUTING CORRESPONDING APP
STOP	0x04	STOP CORRESPONDING APP
EVENT	0x05	DELIVER SPECIFIC EVENT TO APP

FIG.77

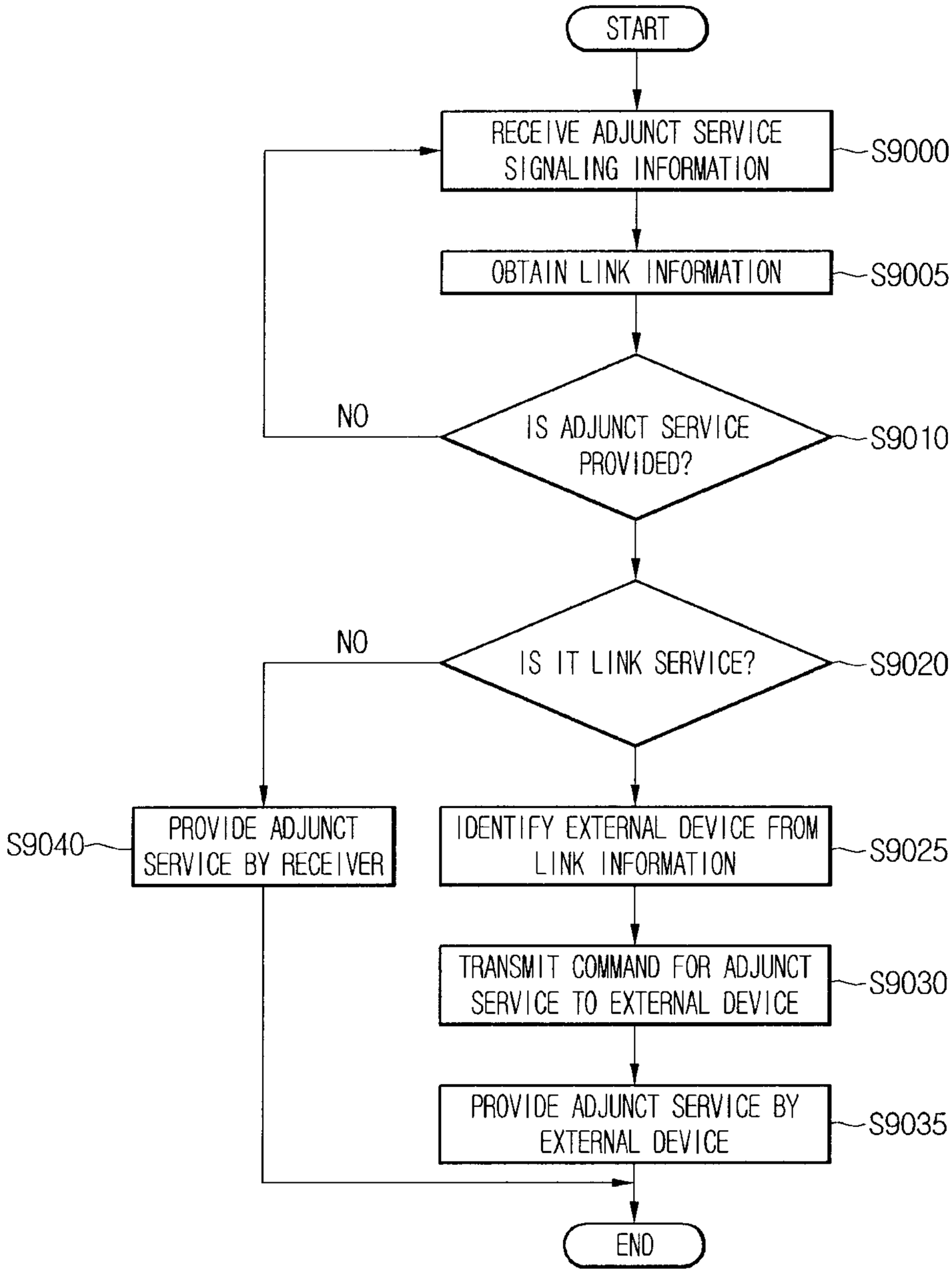

START
RECEIVE ADJUNCT SERVICE SIGNALING INFORMATION
S9000
OBTAIN LINK INFORMATION
S9005
IS ADJUNCT SERVICE PROVIDED?
S9010
NO
IS IT LINK SERVICE?
S9020
NO
PROVIDE ADJUNCT SERVICE BY RECEIVER
S9040
IDENTIFY EXTERNAL DEVICE FROM LINK INFORMATION
S9025
TRANSMIT COMMAND FOR ADJUNCT SERVICE TO EXTERNAL DEVICE
S9030
PROVIDE ADJUNCT SERVICE BY EXTERNAL DEVICE
S9035
END

FIG.78

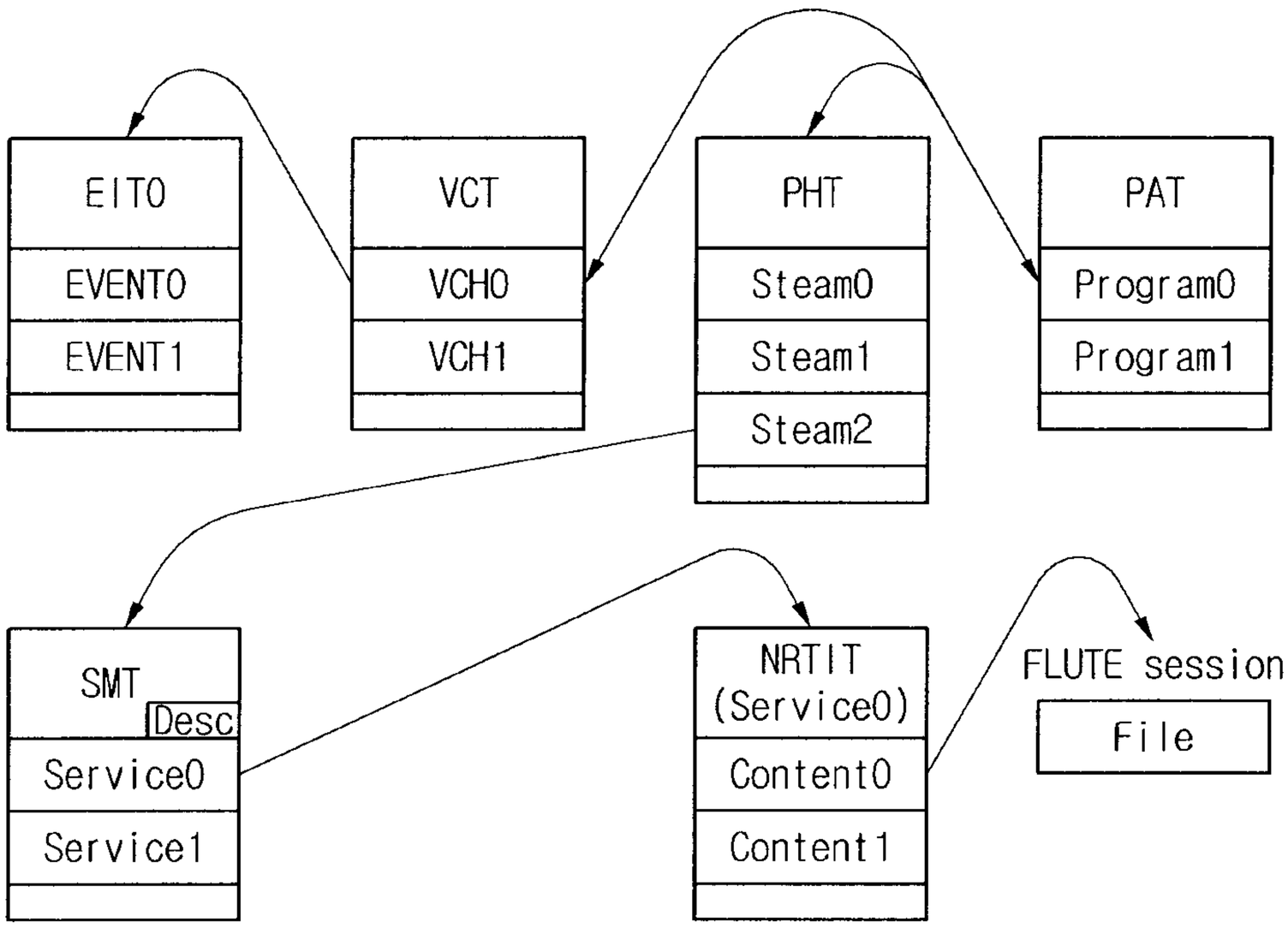

EIT0
EVENT0
EVENT1
VCT
VCH0
VCH1
PHT
Steam0
Steam1
Steam2
PAT
Program0
Program1
SMT
Desc
Service0
Service1
NRTIT
(Service0)
Content0
Content1
FLUTE session
File

FIG.79

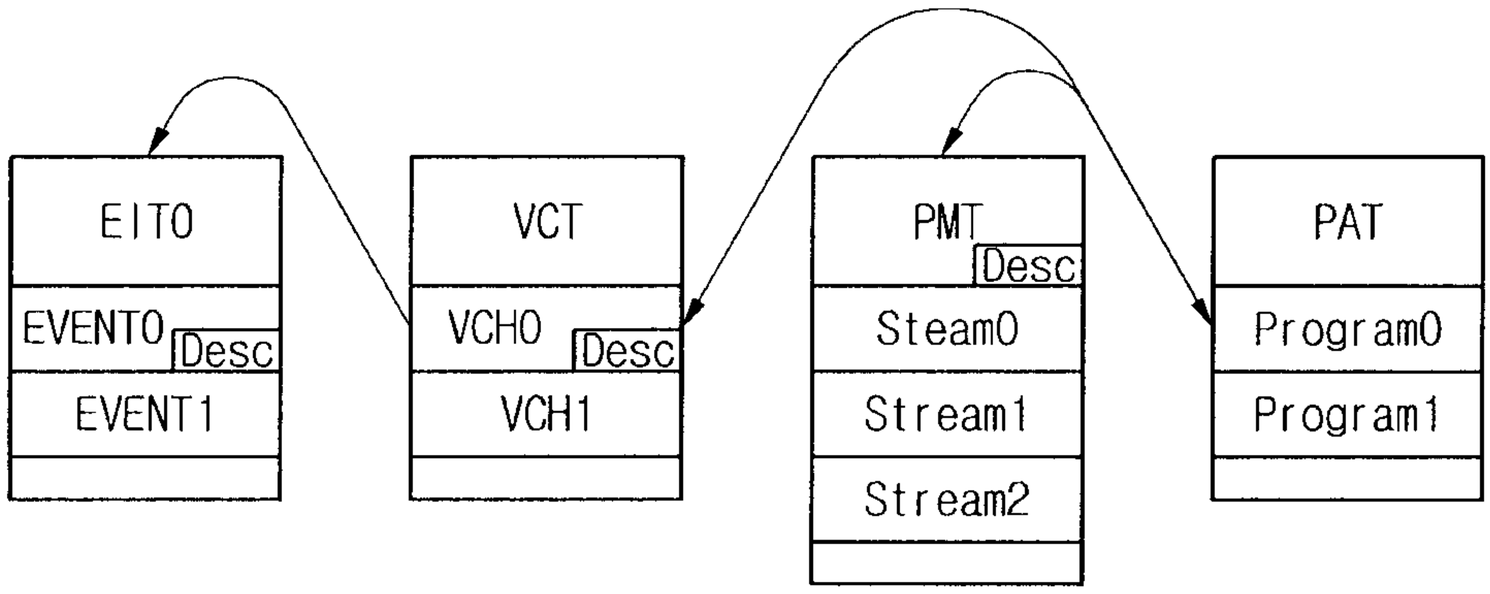

EIT0
EVENT0
Desc
EVENT1
VCT
VCH0
Desc
VCH1
PMT
Desc
Steam0
Stream1
Stream2
PAT
Program0
Program1

FIG.80

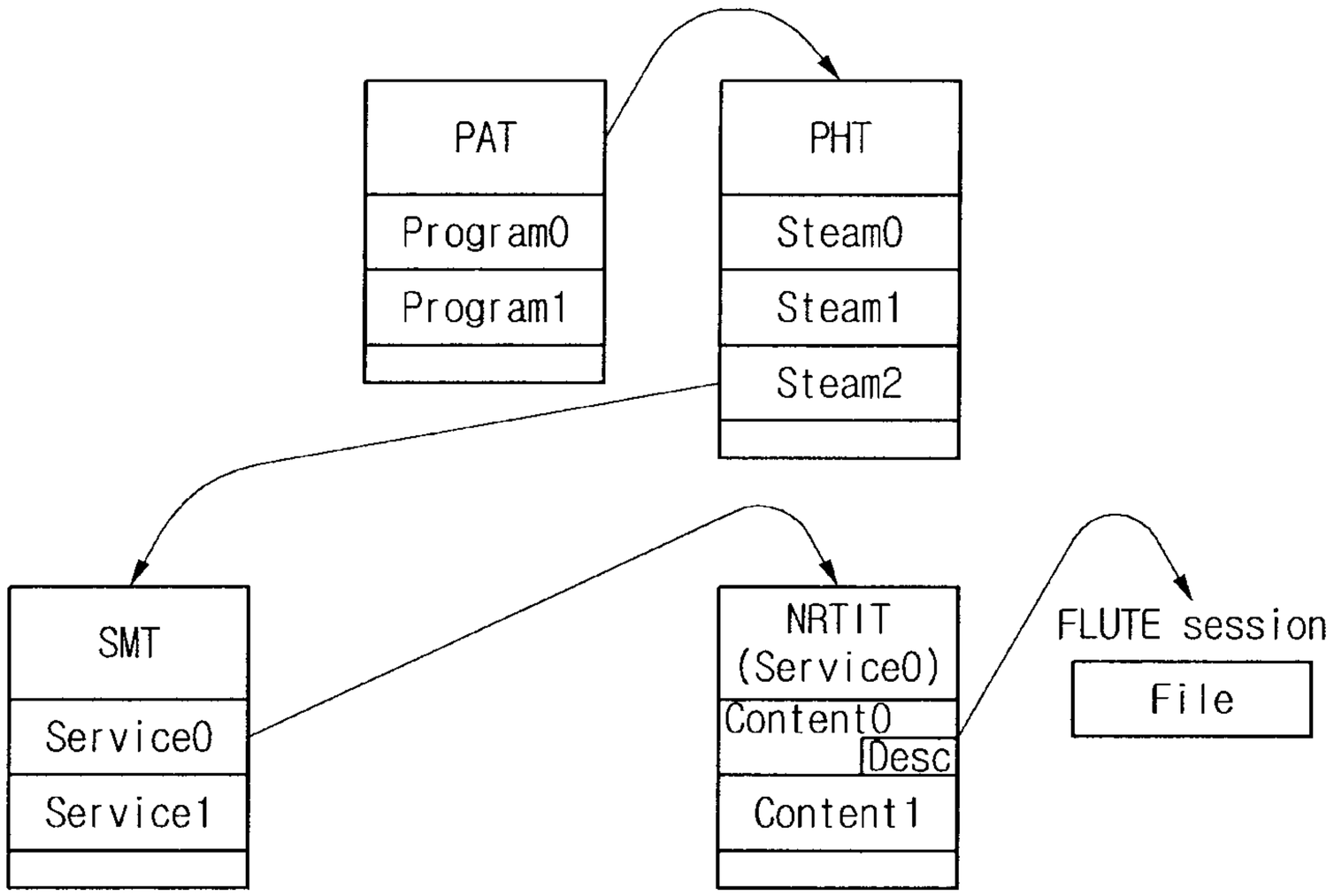

PAT
Program0
Program1
PHT
Steam0
Steam1
Steam2
SMT
Service0
Service1
NRTIT
(Service0)
Content0
Desc
Content1
FLUTE session
File

FIG.81

Syntax	Bits	Format
capabilities_descriptor() {		
descriptor_tag	8	0xTBD
descriptor_length	8	uimsbf
Individual_capabilities() {		
capability_code_count	8	uimsbf
for (i=0; i<capability_code_count; i++) {		
essential_indicator	1	bslbf
capability_code	7	uimsbf
if (capability_code > 0x6F) {		
format_identifier	32	
}		
}		
capability_string_count	8	uimsbf
for (i=0; i<capability_string_count; i++) {		
essential_indicator	1	
capability_category_code	7	
capability_string_length	8	uimsbf
capability_string()	var	uimsbf
}		
}		
sets_of_capabilities() {		
sets_of_capabilities_count	8	bslbf
for (k=0; k<sets_of_capabilities_count; k++) {		
essential_indicator	1	bslbf
capability_codes_in_set_count	7	uimsbf
for (i=0; i<capability_codes_in_set_count; i++) {		
reserved	1	
capability_code	7	uimsbf
if (capability_code > 0x6F) {		
format_identifier	32	uimsbf
}		
}		
capability_strings_in_set_count	8	uimsbf
for (i=0; i<capability_strings_in_set_count; i++) {		
reserved	1	
capability_category_code	7	
capability_string_length	8	uimsbf
capability_string()	var	uimsbf
}		
}		
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.82

capability_category_code	Capability Category	Registry
0x00	reserved	
0x01	Download Protocol	No registry ?use widely used industry name
0x02	FEC Algorithm	IANA registry of FEC encoding IDs and instance IDs [63]
0x03	Wrapper/Archive Format	IANA registry of media types and subtypes [62]
0x04	Compression Algorithm	IANA registry of HTTP Content-Coding values [61]
0x05	Media Type	IANA registry of media types and subtypes [62]
0x06	Companion Device (Second screen device)	User filter and version used in HTTP protocol
0x07-0x7F	reserved	

FIG.83

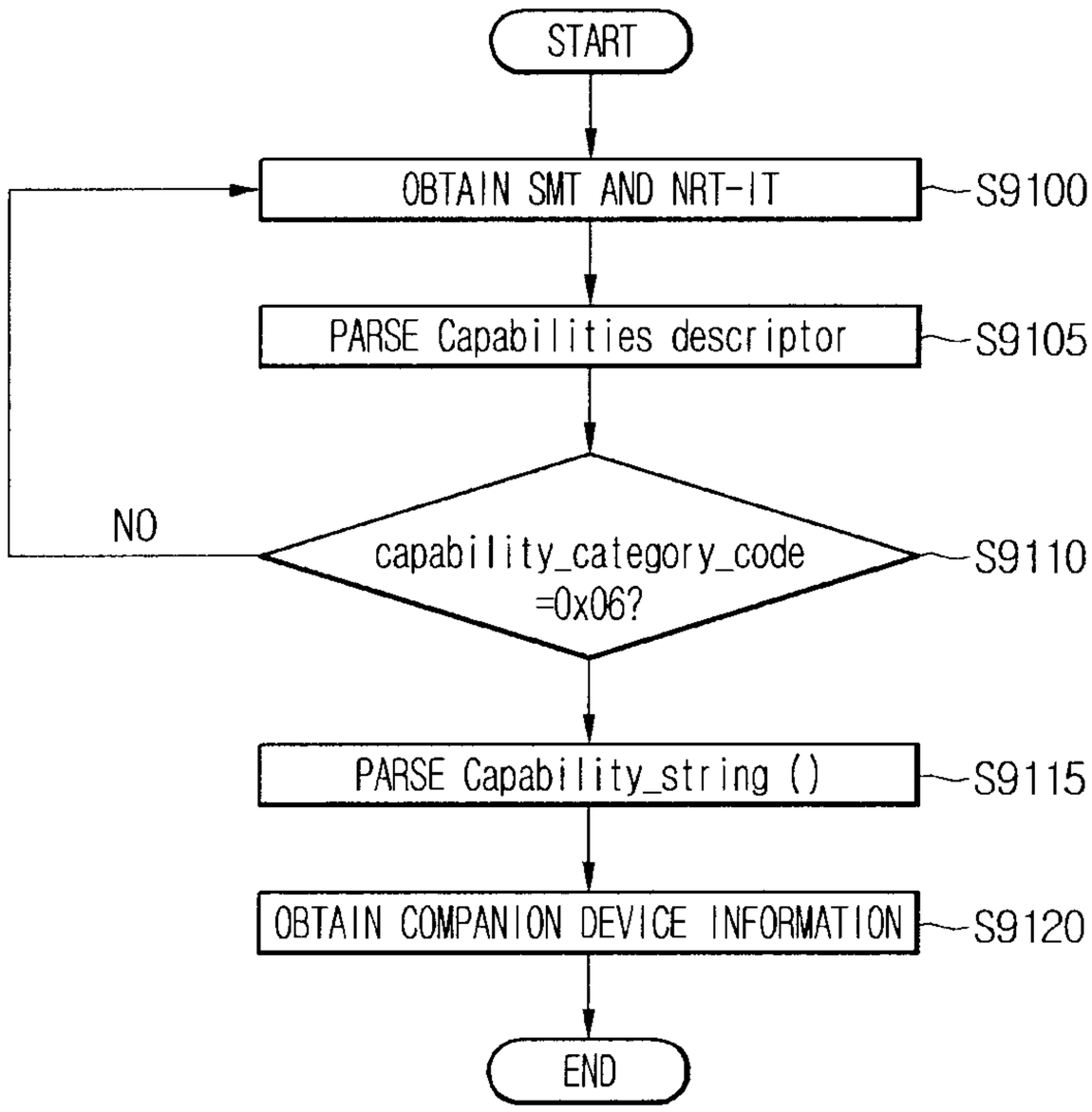

START
OBTAIN SMT AND NRT-IT
S9100
PARSE Capabilities descriptor
S9105
NO
capability_category_code =0x06?
S9110
PARSE Capability_string ()
S9115
OBTAIN COMPANION DEVICE INFORMATION
S9120
END

FIG.84

Syntax	No. of Bits	Format
content _items_group_descriptor() {		
descriptor_tag	8	TBD
descriptor_length	8	uimsbf
content_items_group_id	32	uimsbf
num_content_items	8	Uimsbf
for (i=0; i<num_content_items; i++)		
content_linkage	32	bslbf
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		
}		

FIG.85

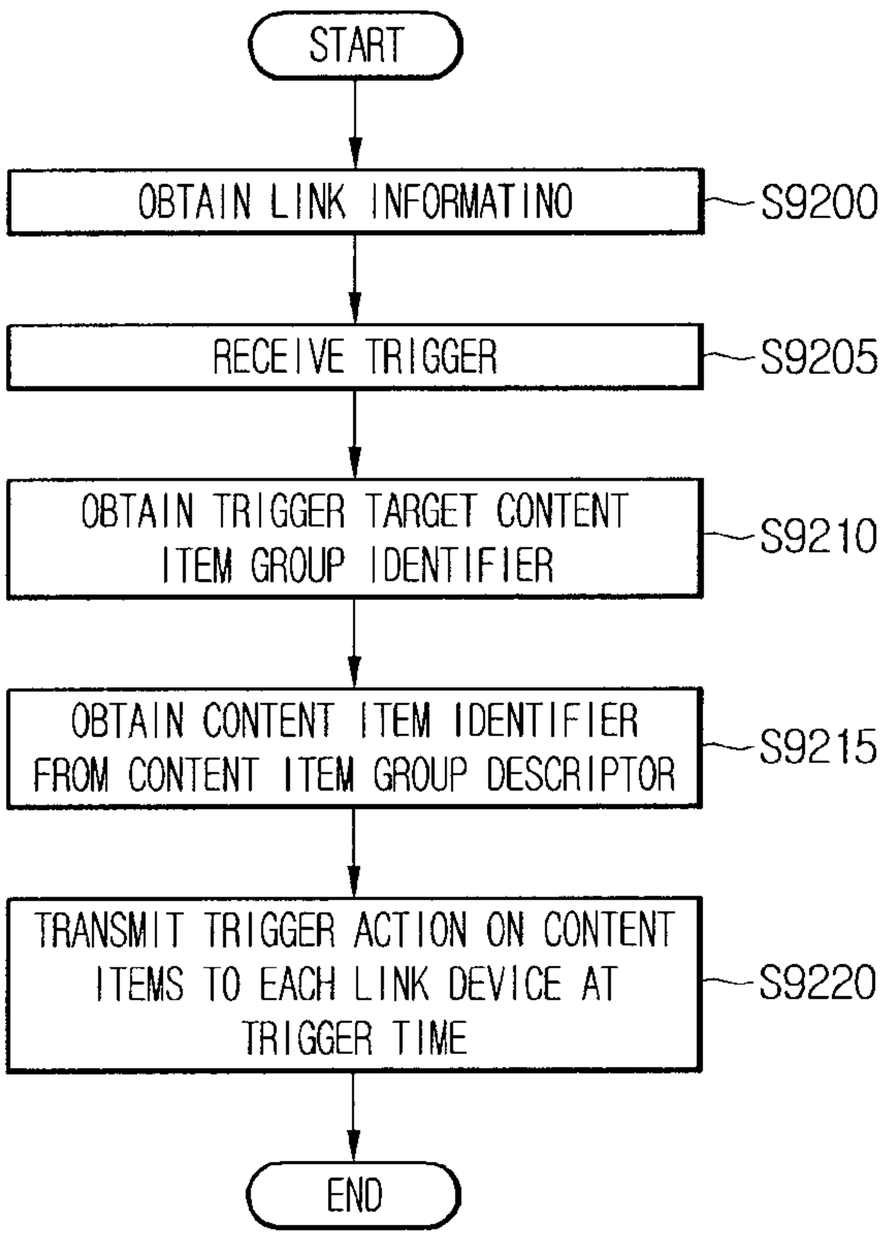
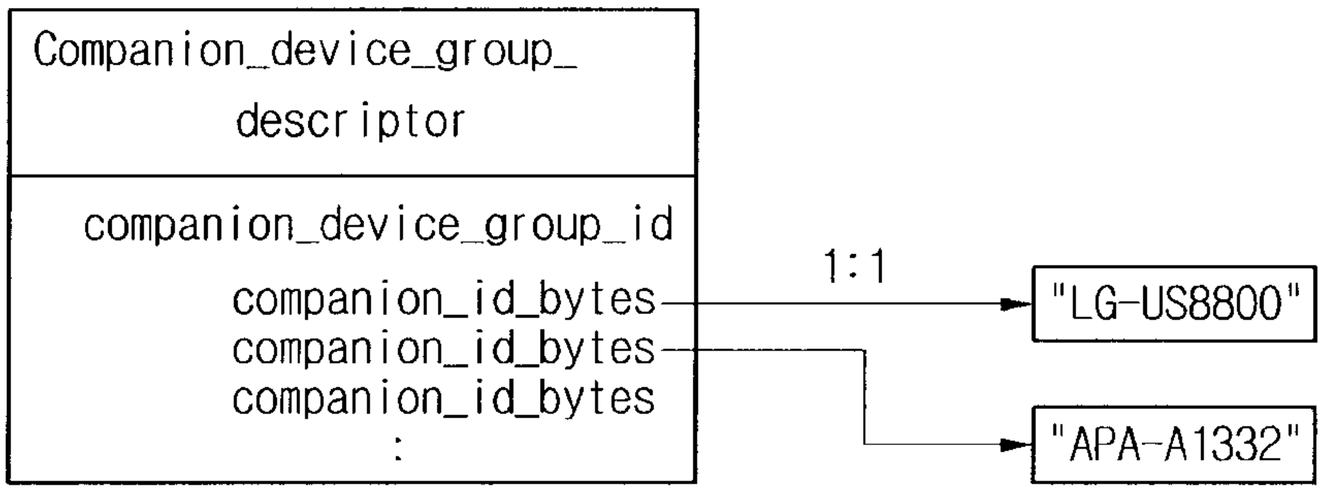

START
OBTAIN LINK INFORMATINO
S9200
RECEIVE TRIGGER
S9205
OBTAIN TRIGGER TARGET CONTENT ITEM GROUP IDENTIFIER
S9210
OBTAIN CONTENT ITEM IDENTIFIER FROM CONTENT ITEM GROUP DESCRIPTOR
S9215
TRANSMIT TRIGGER ACTION ON CONTENT ITEMS TO EACH LINK DEVICE AT TRIGGER TIME
S9220
END

FIG.86

Syntax	No. of Bits	Format
companion_device_group_descriptor() {		
descriptor_tag	8	TBD
descriptor_length	8	uimsbf
companion_device_group_id	32	uimsbf
num_companion_devices	8	Uimsbf
for (i=0; i<num_companion_devicess; i++)		
companion_id	32	bslbf
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.87

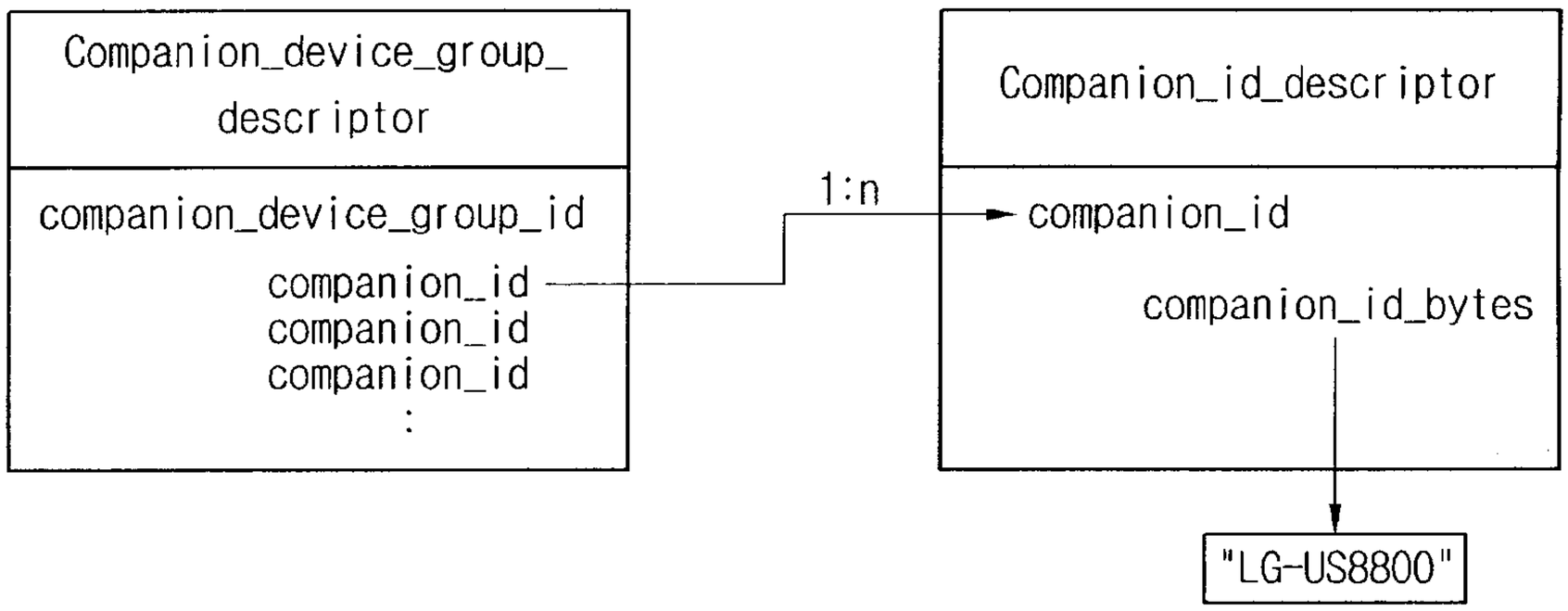


(A)

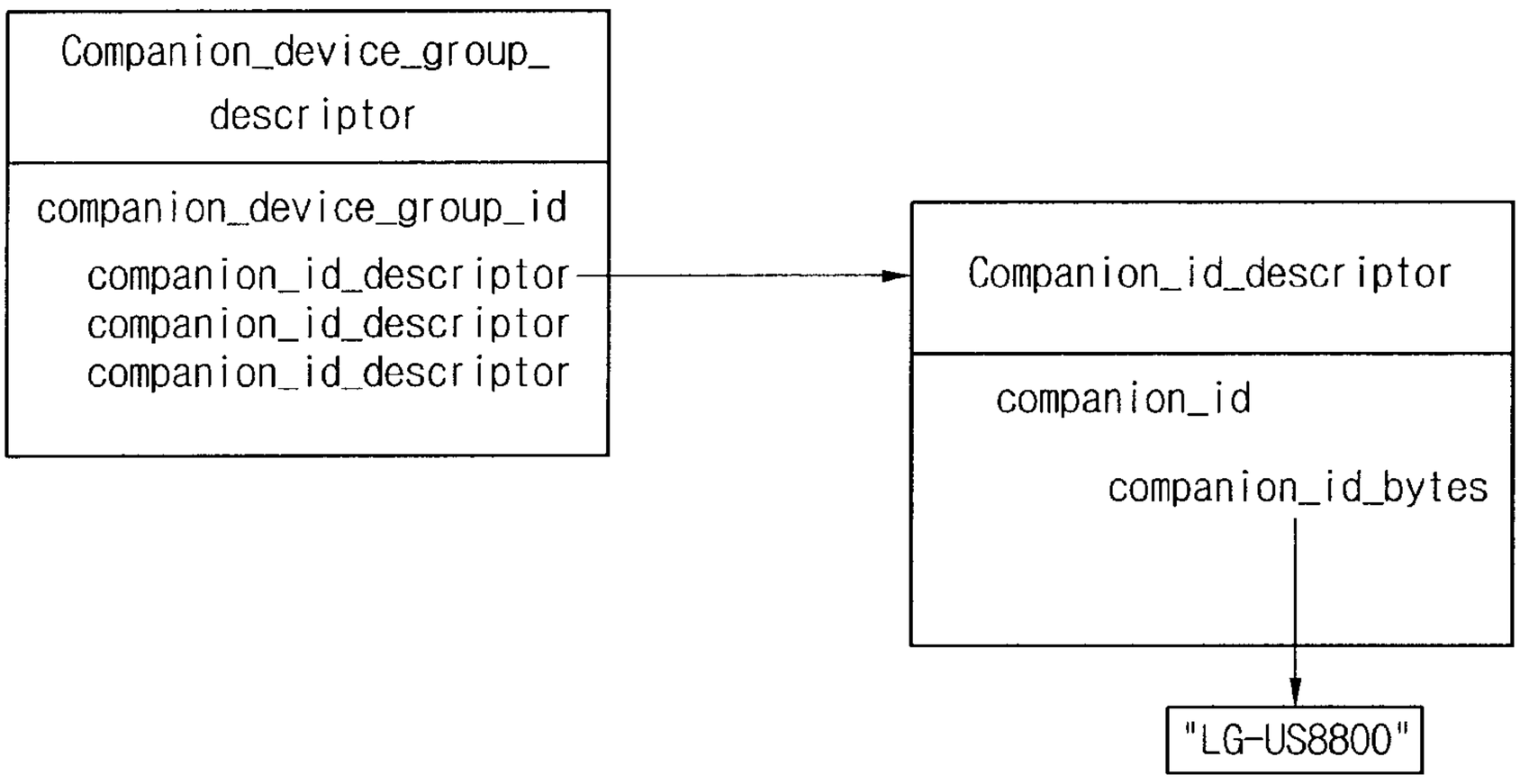
Syntax	Bits	Format
companion_device_group_descriptor() {		
descriptor_tag	8	TBD
descriptor_length	8	uimsbf
companion_device_group_id	32	uimsbf
num_companion_devices	8	Uimsbf
for (i=0; i<num_content_items; i++)		
{		
companion_id	32	bslbf
num_companion_id_bytes	8	uimsbf
for (j=0; j<num_companion_id_bytes; j++)		
companion_id_bytes	8	bslbf
}		
for (k=0; k< N; k++) {		
reserved	8	bslbf
}		

(B)

FIG.88


Companion_device_group_
descriptor
companion_device_group_id
companion_id
companion_id
companion_id
:
1:n
Companion_id_descriptor
companion_id
companion_id_bytes
"LG-US8800"

(A)


Companion_device_group_
descriptor
companion_device_group_id
companion_id_descriptor
companion_id_descriptor
companion_id_descriptor
Companion_id_descriptor
companion_id
companion_id_bytes
"LG-US8800"

(B)

FIG.89

Syntax	Bits	Format
companion_id_descriptor() {		
descriptor_tag	8	TBD
descriptor_length	8	uimsbf
companion_id	32	bslbf
num_companion_id_bytes	8	uimsbf
for (i=0; i<num_companion_id_bytes; i++)		
companion_id_bytes	8	bslbf
for (j=0; j< N; j++) {		
reserved	8	bslbf
}		
}		

FIG.90

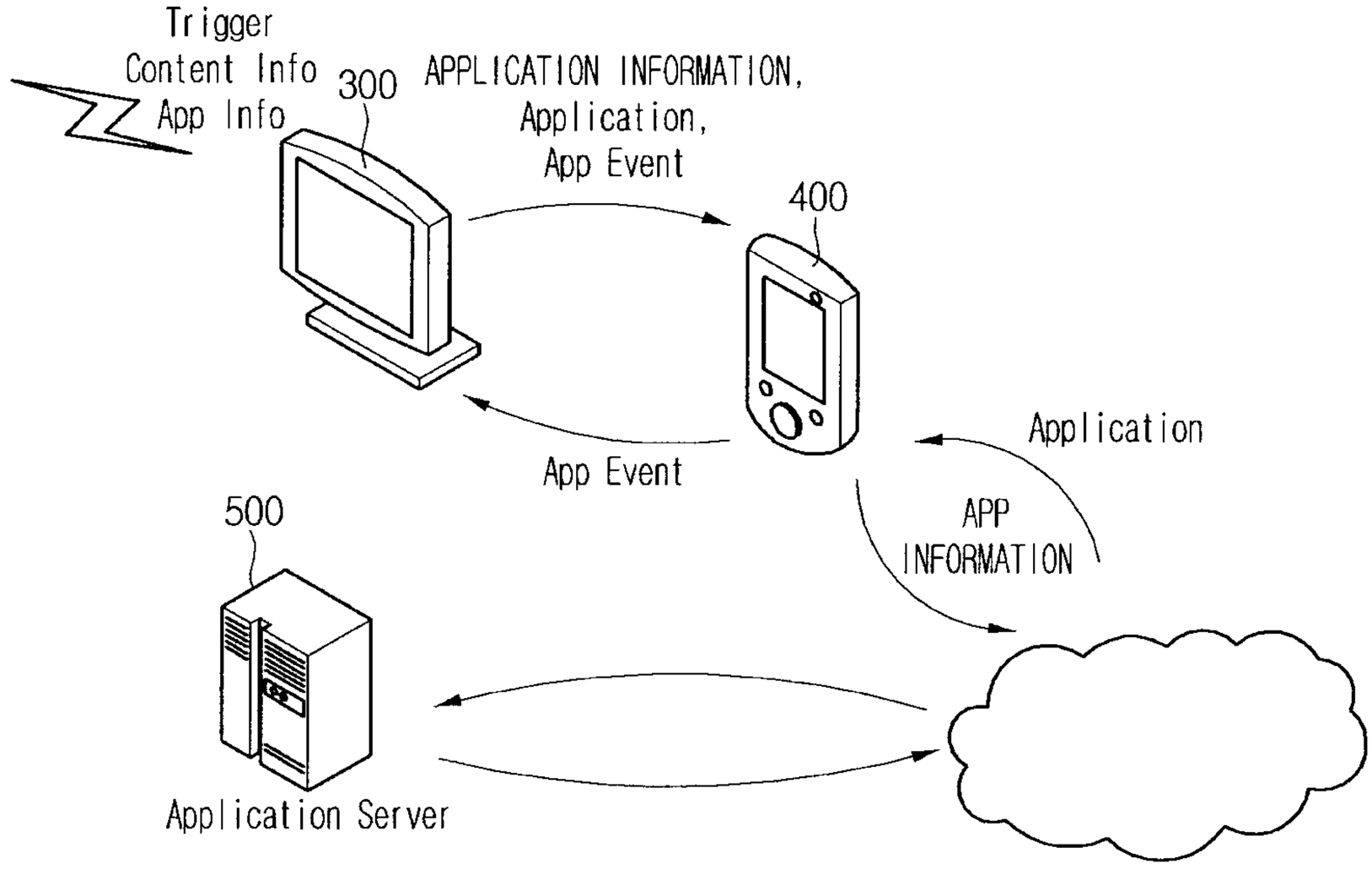

Trigger
Content Info
App Info
300
APPLICATION INFORMATION,
Application,
App Event
400
App Event
Application
APP
INFORMATION
500
Application Server

FIG.91

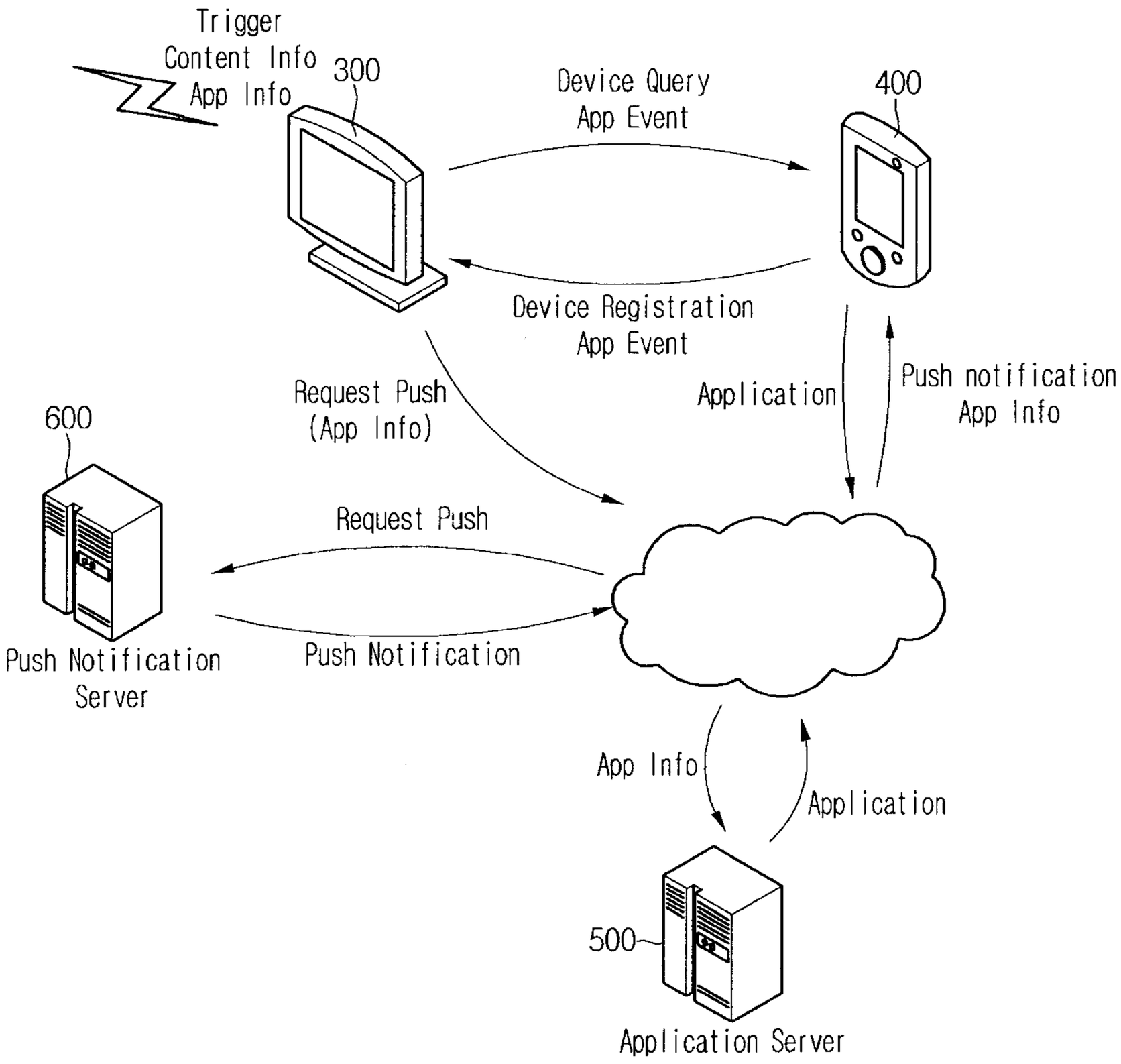

Trigger
Content Info
App Info
300
Device Query
App Event
400
Device Registration
App Event
Request Push
(App Info)
Application
Push notification
App Info
600
Request Push
Push Notification
Push Notification
Server
App Info
Application
500
Application Server

FIG.92

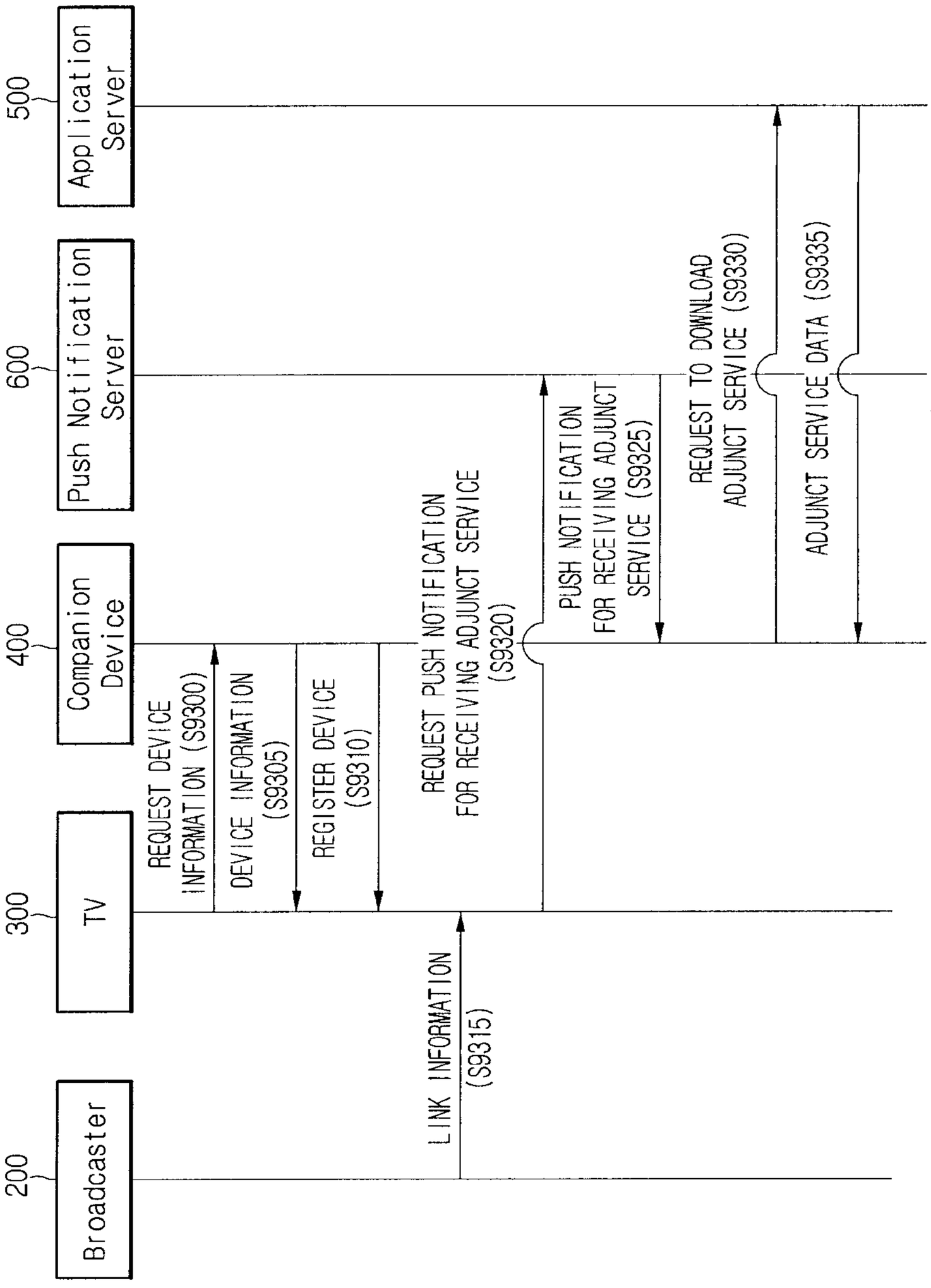

Broadcaster
200
TV
300
Companion Device
400
Push Notification Server
600
Application Server
500
LINK INFORMATION (S9315)
REQUEST DEVICE INFORMATION (S9300)
DEVICE INFORMATION (S9305)
REGISTER DEVICE (S9310)
REQUEST PUSH NOTIFICATION FOR RECEIVING ADJUNCT SERVICE (S9320)
PUSH NOTIFICATION FOR RECEIVING ADJUNCT SERVICE (S9325)
REQUEST TO DOWNLOAD ADJUNCT SERVICE (S9330)
ADJUNCT SERVICE DATA (S9335)

FIG.93

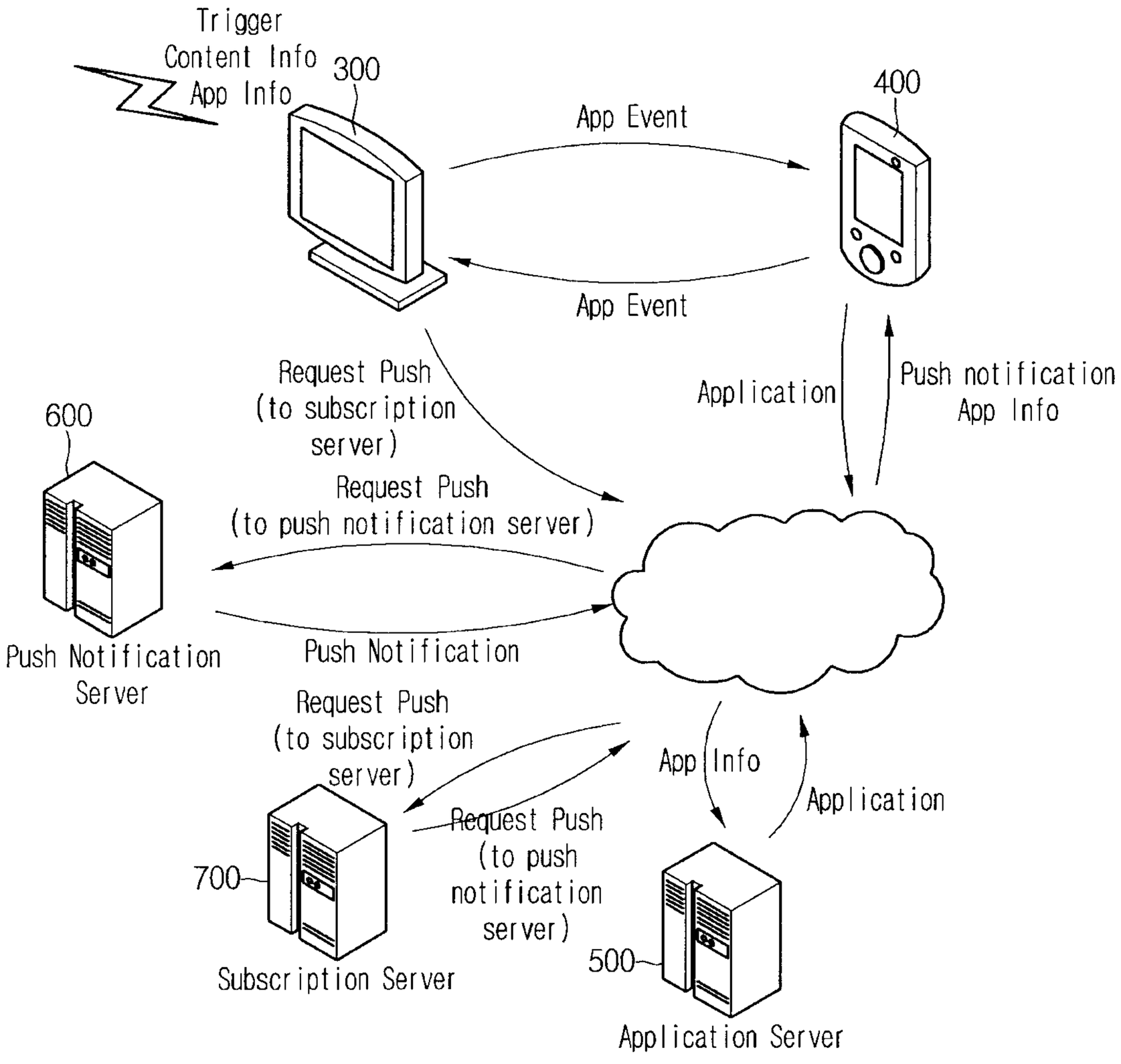

Trigger
Content Info
App Info
300
400
App Event
App Event
Request Push
(to subscription
server)
Application
Push notification
App Info
600
Request Push
(to push notification server)
Push Notification
Server
Push Notification
Request Push
(to subscription
server)
App Info
Application
Request Push
(to push
notification
server)
700
500
Subscription Server
Application Server

FIG.94

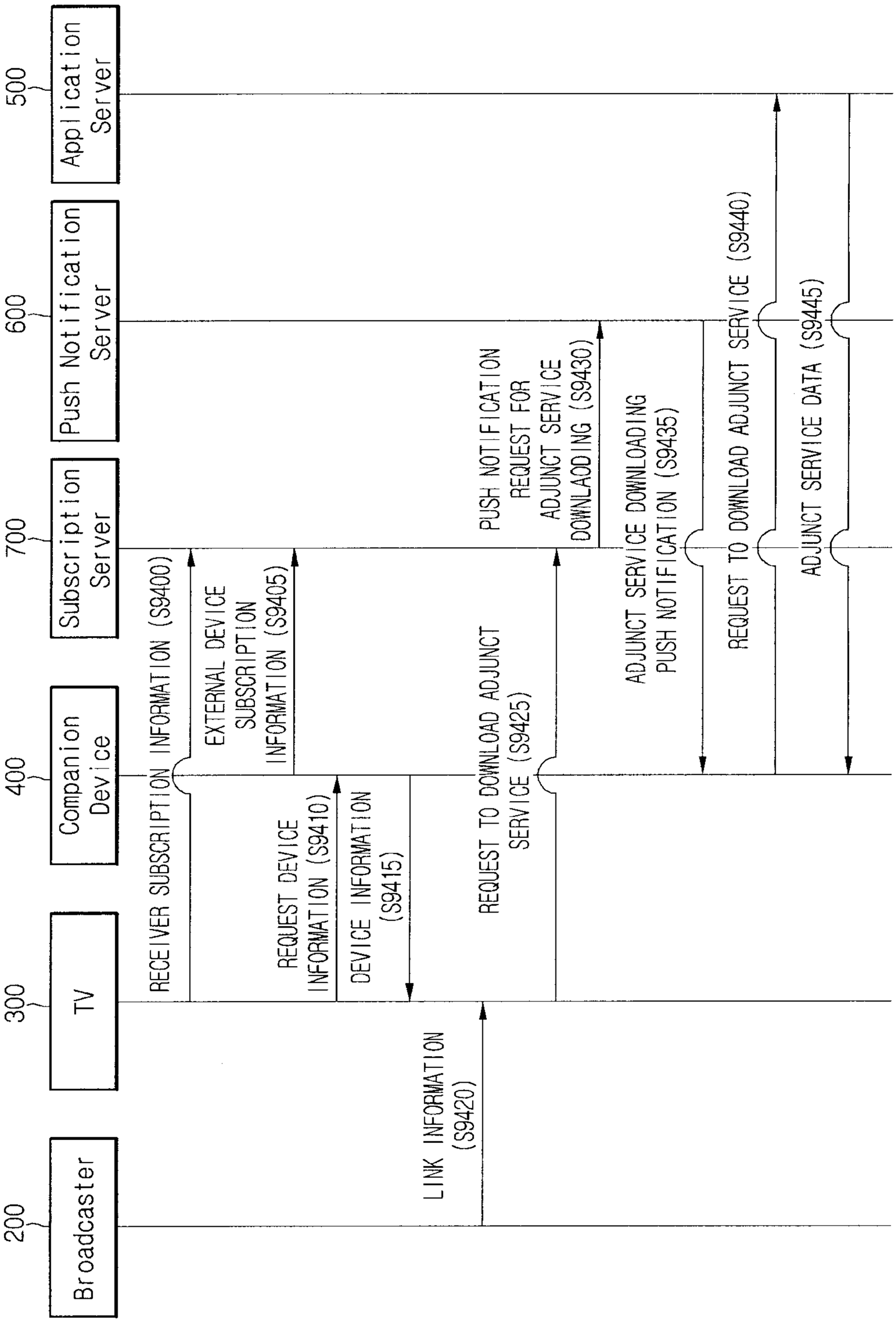

200
Broadcaster
300
TV
400
Companion Device
500
Application Server
600
Push Notification Server
700
Subscription Server
LINK INFORMATION (S9420)
RECEIVER SUBSCRIPTION INFORMATION (S9400)
REQUEST DEVICE INFORMATION (S9410)
DEVICE INFORMATION (S9415)
EXTERNAL DEVICE SUBSCRIPTION INFORMATION (S9405)
REQUEST TO DOWNLOAD ADJUNCT SERVICE (S9425)
PUSH NOTIFICATION REQUEST FOR ADJUNCT SERVICE DOWNLOADING (S9430)
ADJUNCT SERVICE DOWNLOADING PUSH NOTIFICATION (S9435)
REQUEST TO DOWNLOAD ADJUNCT SERVICE (S9440)
ADJUNCT SERVICE DATA (S9445)

FIG.95

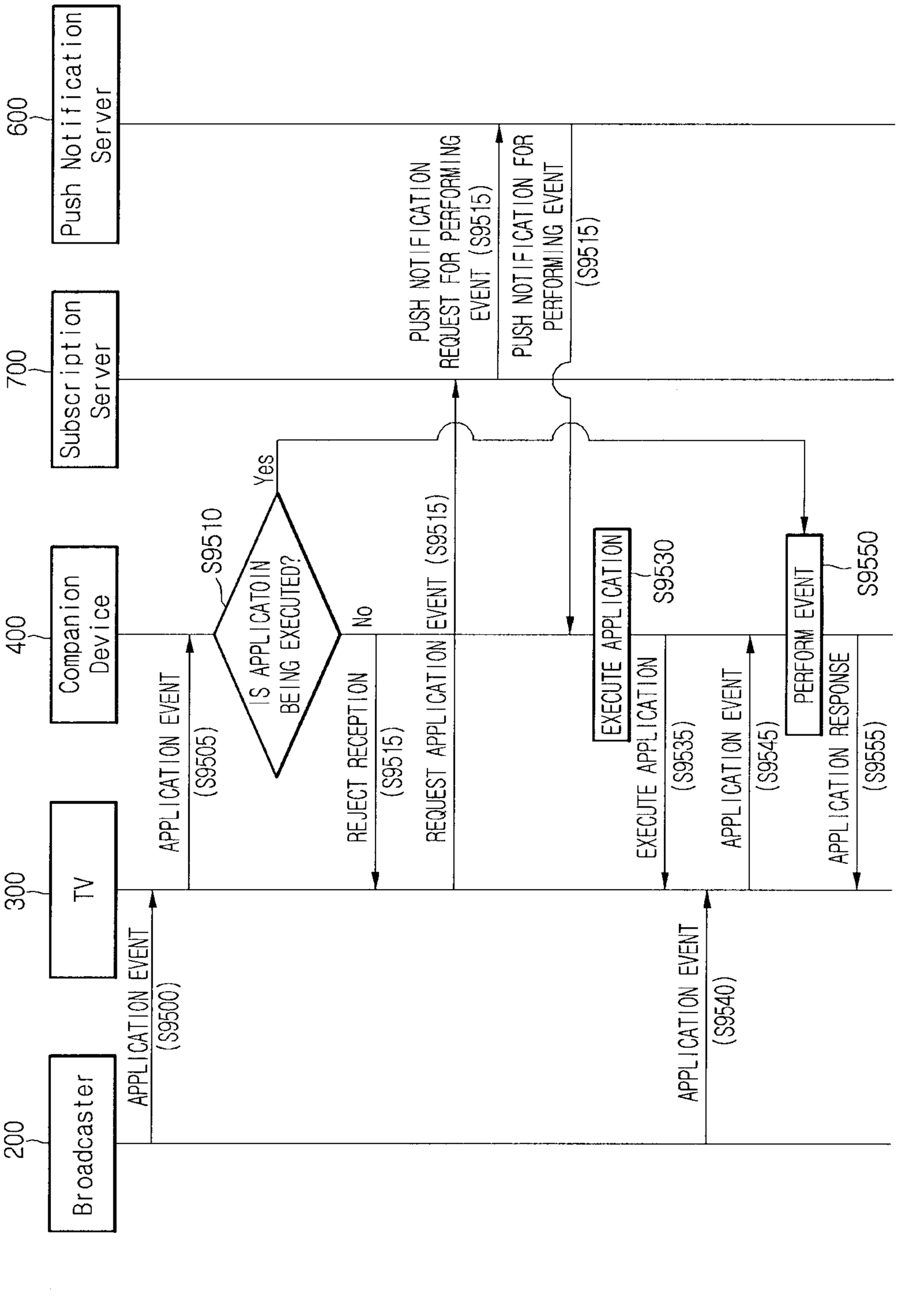

Broadcaster
200
TV
300
Companion Device
400
Subscription Server
700
Push Notification Server
600
APPLICATION EVENT (S9500)
APPLICATION EVENT (S9505)
S9510
IS APPLICATOIN BEING EXECUTED?
Yes
No
REJECT RECEPTION (S9515)
REQUEST APPLICATION EVENT (S9515)
PUSH NOTIFICATION REQUEST FOR PERFORMING EVENT (S9515)
PUSH NOTIFICATION FOR PERFORMING EVENT (S9515)
EXECUTE APPLICATION
S9530
EXECUTE APPLICATION (S9535)
APPLICATION EVENT (S9540)
APPLICATION EVENT (S9545)
PERFORM EVENT
S9550
APPLICATION RESPONSE (S9555)

FIG.96

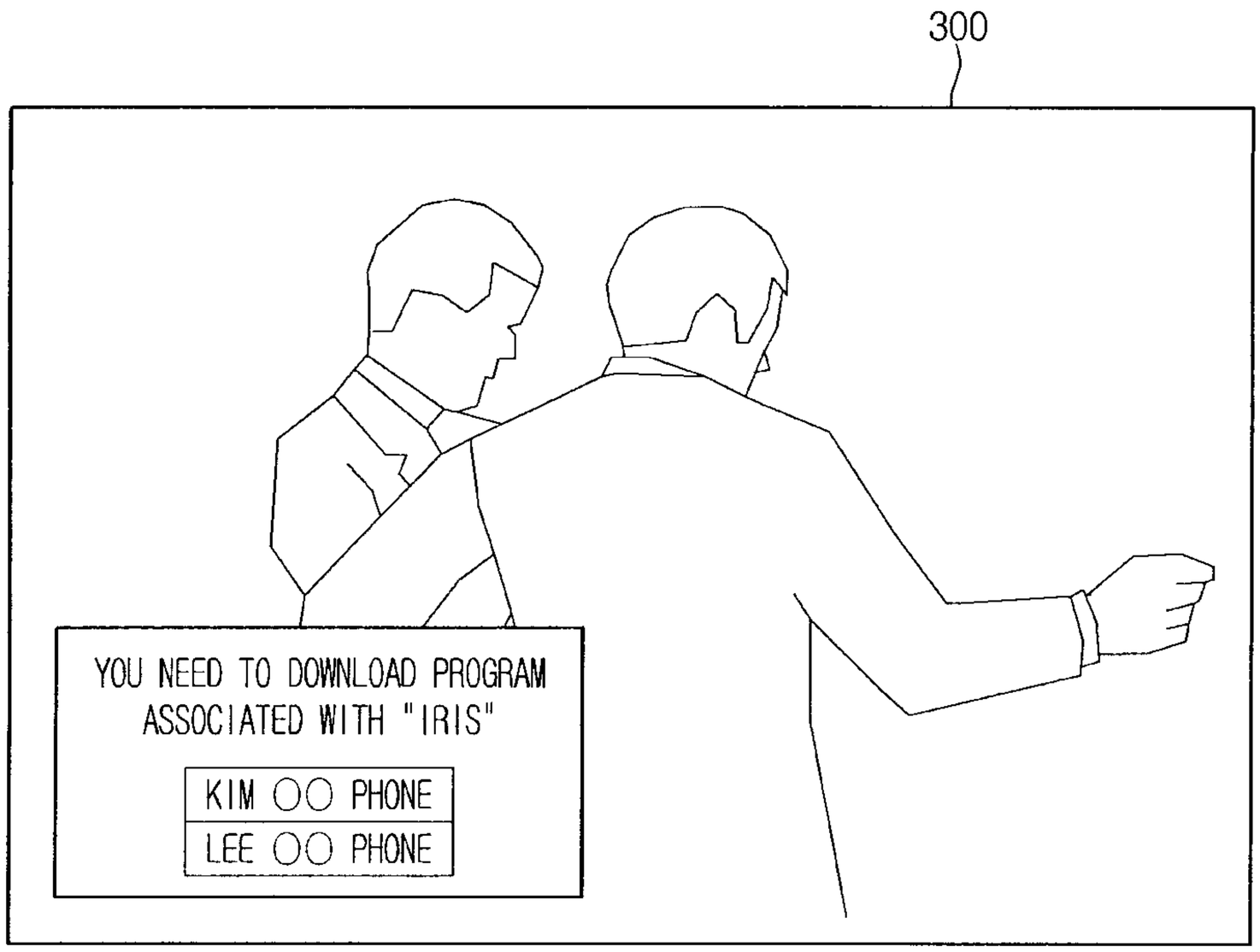

300
YOU NEED TO DOWNLOAD PROGRAM
ASSOCIATED WITH "IRIS"
KIM OO PHONE
LEE OO PHONE

FIG.97

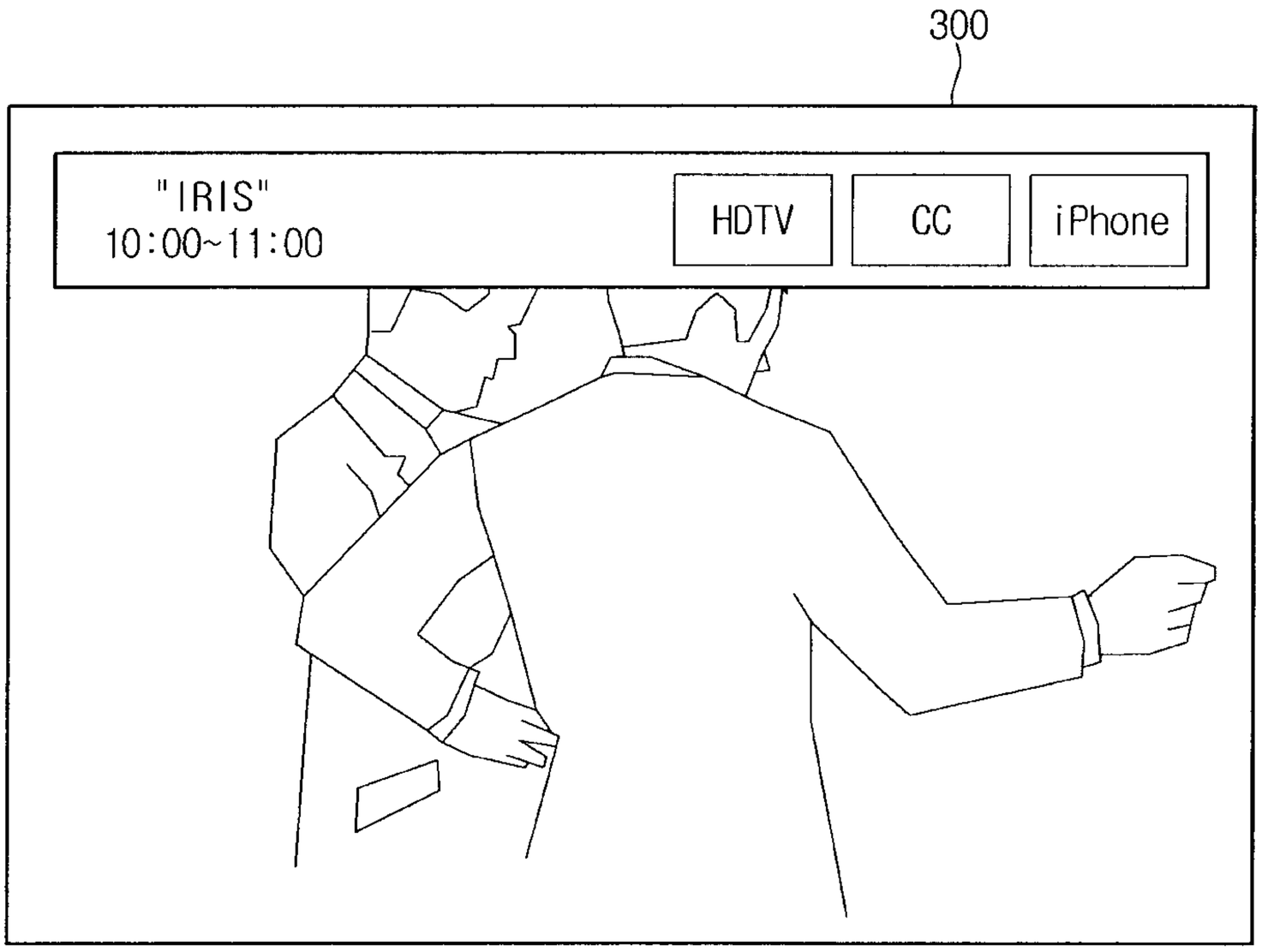


FIG.98

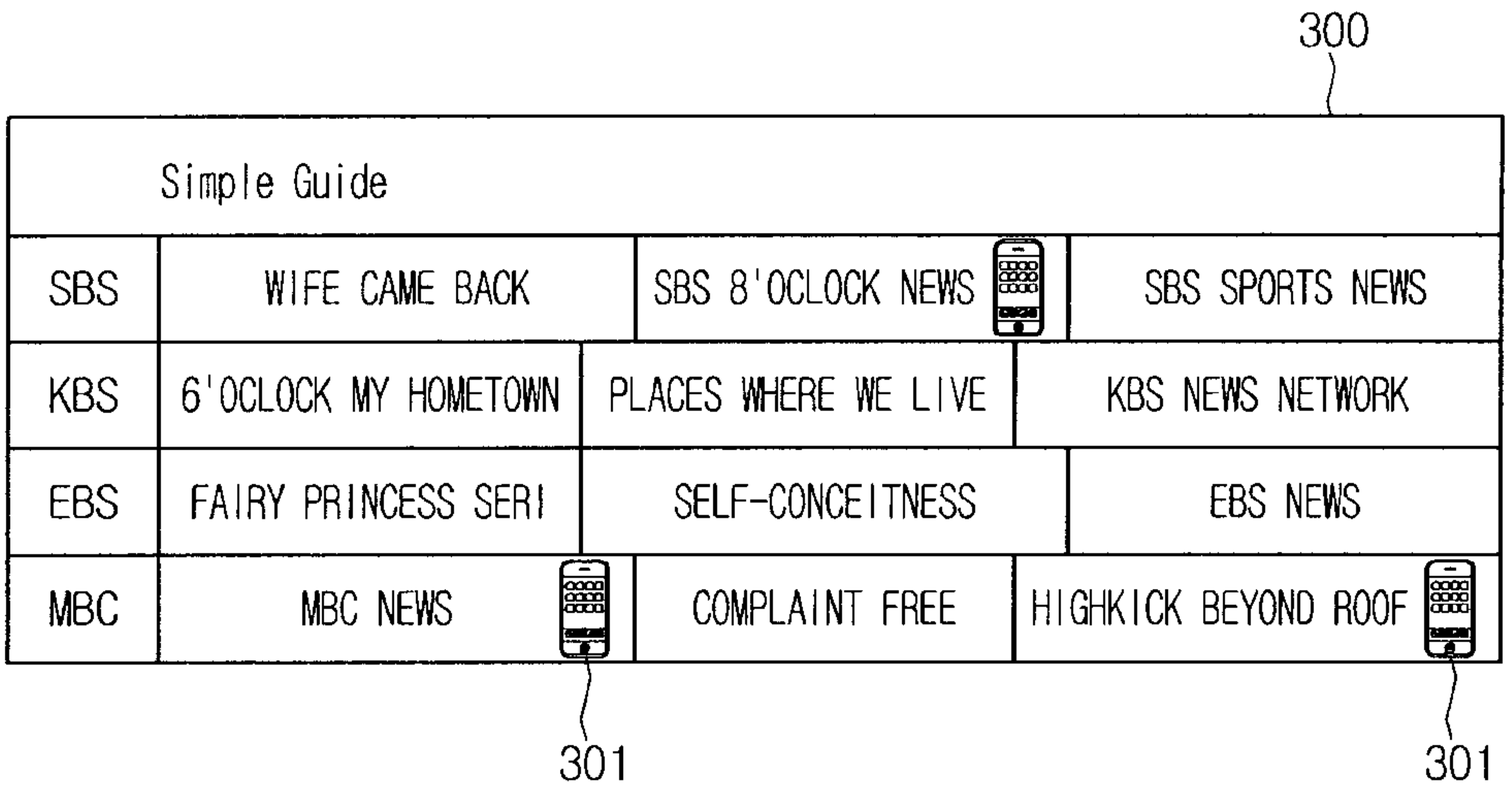

300
Simple Guide
SBS
WIFE CAME BACK
SBS 8'OCLOCK NEWS
SBS SPORTS NEWS
KBS
6'OCLOCK MY HOMETOWN
PLACES WHERE WE LIVE
KBS NEWS NETWORK
EBS
FAIRY PRINCESS SERI
SELF-CONCEITNESS
EBS NEWS
MBC
MBC NEWS
COMPLAINT FREE
HIGHKICK BEYOND ROOF
301
301

FIG.99

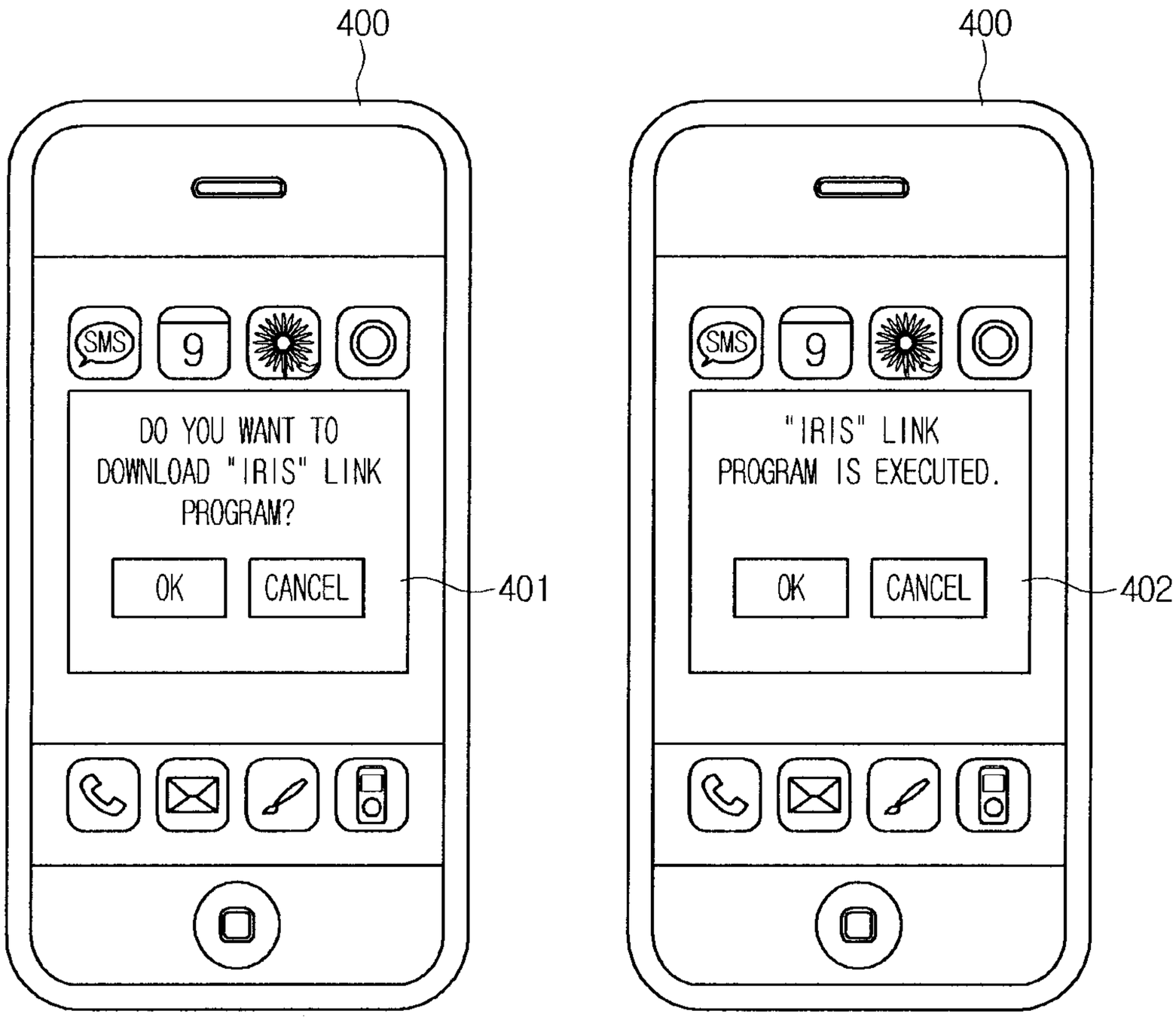

400
SMS
9
DO YOU WANT TO DOWNLOAD "IRIS" LINK PROGRAM?
OK
CANCEL
401
(A)
400
SMS
9
"IRIS" LINK PROGRAM IS EXECUTED.
OK
CANCEL
402
(B)

FIG.100

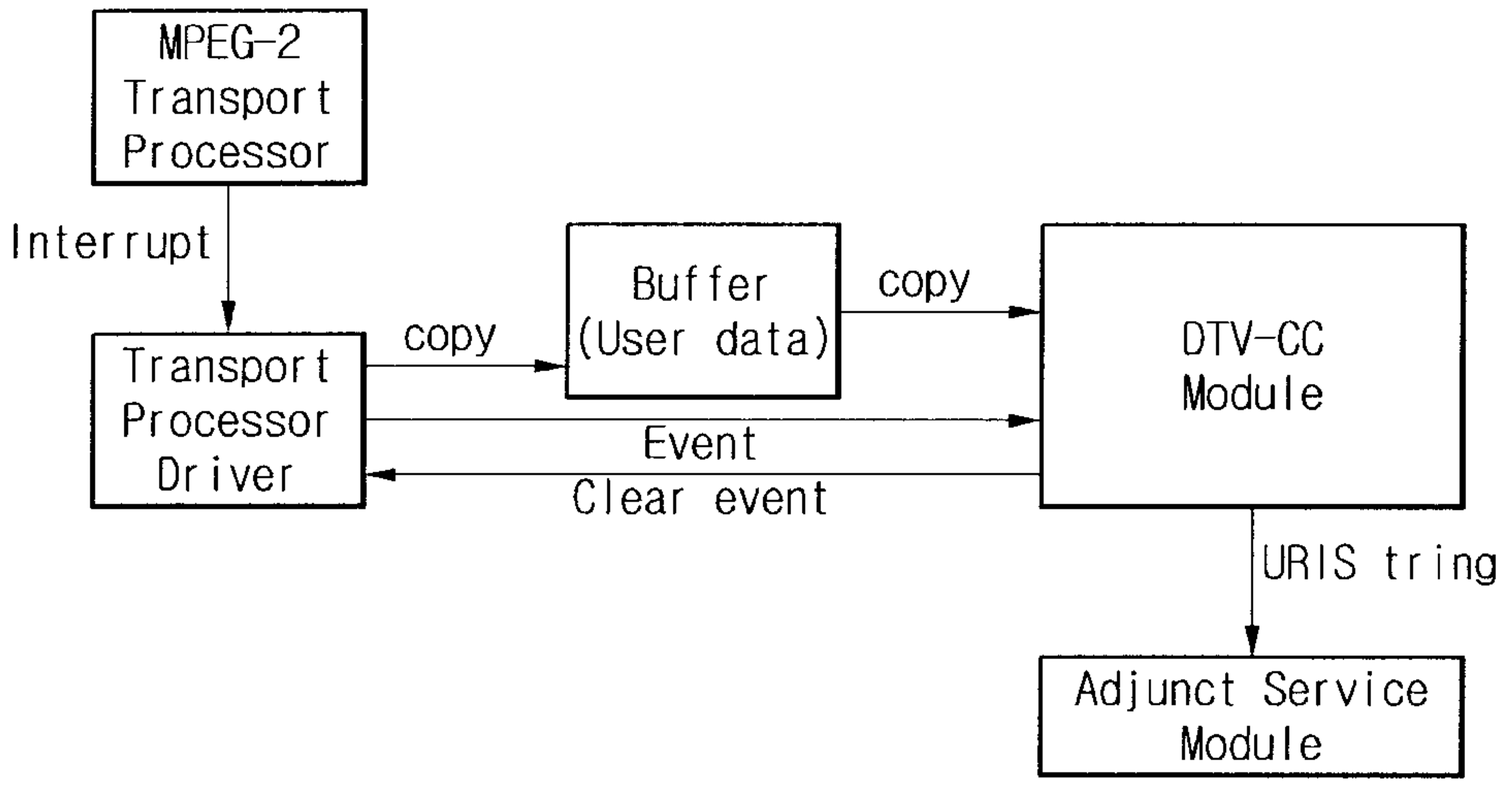

MPEG-2
Transport
Processor
Interrupt
Transport
Processor
Driver
copy
Buffer
(User data)
copy
DTV-CC
Module
Event
Clear event
URIS tring
Adjunct Service
Module

FIG.101

	No. of bits	Format	No. of Bytes
Trigger_message() {			
Trigger_Id	8	uimsbf	1
Version	4	uimsbf	1
Target_device_type	2	uimsbf	
Reserved	2	uimsbf	
TDO_Action	8	uimsbf	1
for (k=0; k<L-1; k++) {			
URI_character	8	uimsbf	1
}			
}			

FIG.102

Target_device_type	Description
00	TRIGGER FOR UTILIZING ATSC NRT 1.0 AND TRIGGER INFORMATION ON CONTENT COMPATIBLE WITH IT ARE INCLUDED
01	ONLY TRIGGER INFORMATION THAT MAY BE EXECUTED ON ATSC 2.0 RECEIVER IS INCLUDED.
10	ONLY TRIGGER THAT MAY BE EXECUTED ON COMPANION DEVICE IS INCLUDED
11	TRIGGER INFORMATION THAT MAY BE EXECUTED ON BOTH ATSC 2.0 RECEIVER AND COMPANION DEVICE IS INCLUDED

FIG.103

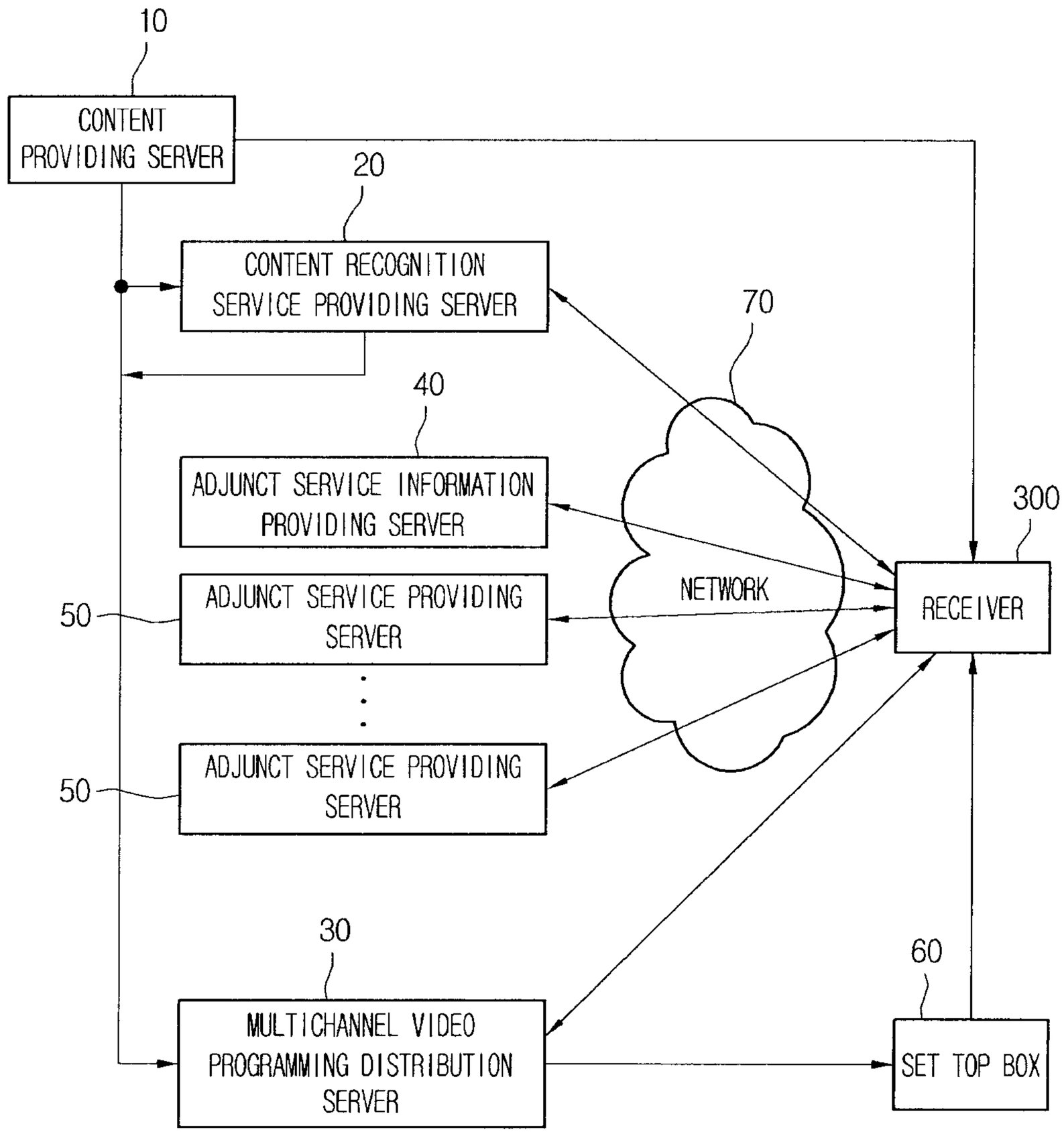

10
CONTENT PROVIDING SERVER
20
CONTENT RECOGNITION SERVICE PROVIDING SERVER
70
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
300
NETWORK
RECEIVER
50
ADJUNCT SERVICE PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
30
60
MULTICHANNEL VIDEO PROGRAMMING DISTRIBUTION SERVER
SET TOP BOX

FIG.104

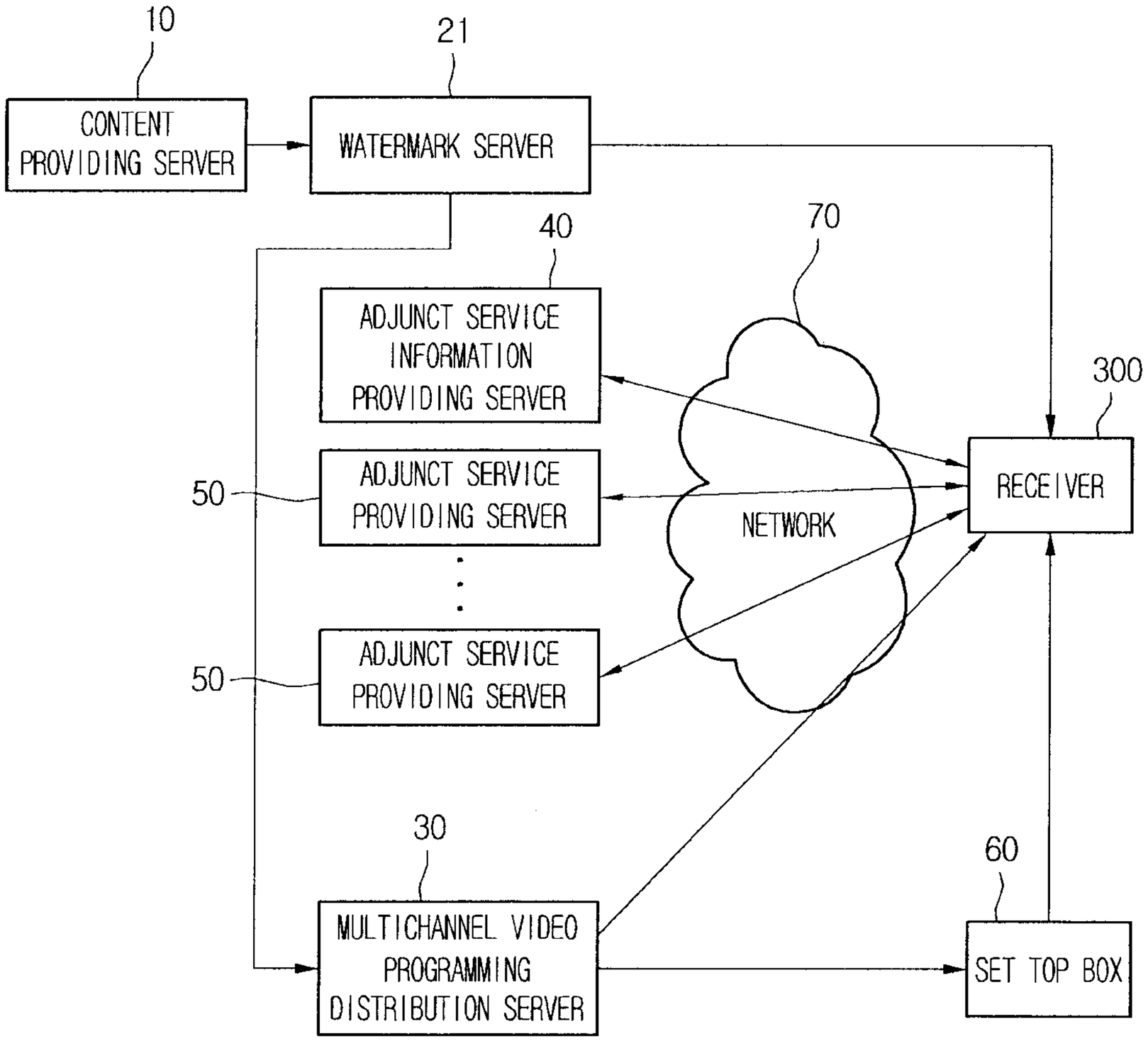

10
CONTENT PROVIDING SERVER
21
WATERMARK SERVER
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
70
NETWORK
300
RECEIVER
50
ADJUNCT SERVICE PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
30
MULTICHANNEL VIDEO PROGRAMMING DISTRIBUTION SERVER
60
SET TOP BOX

FIG.105

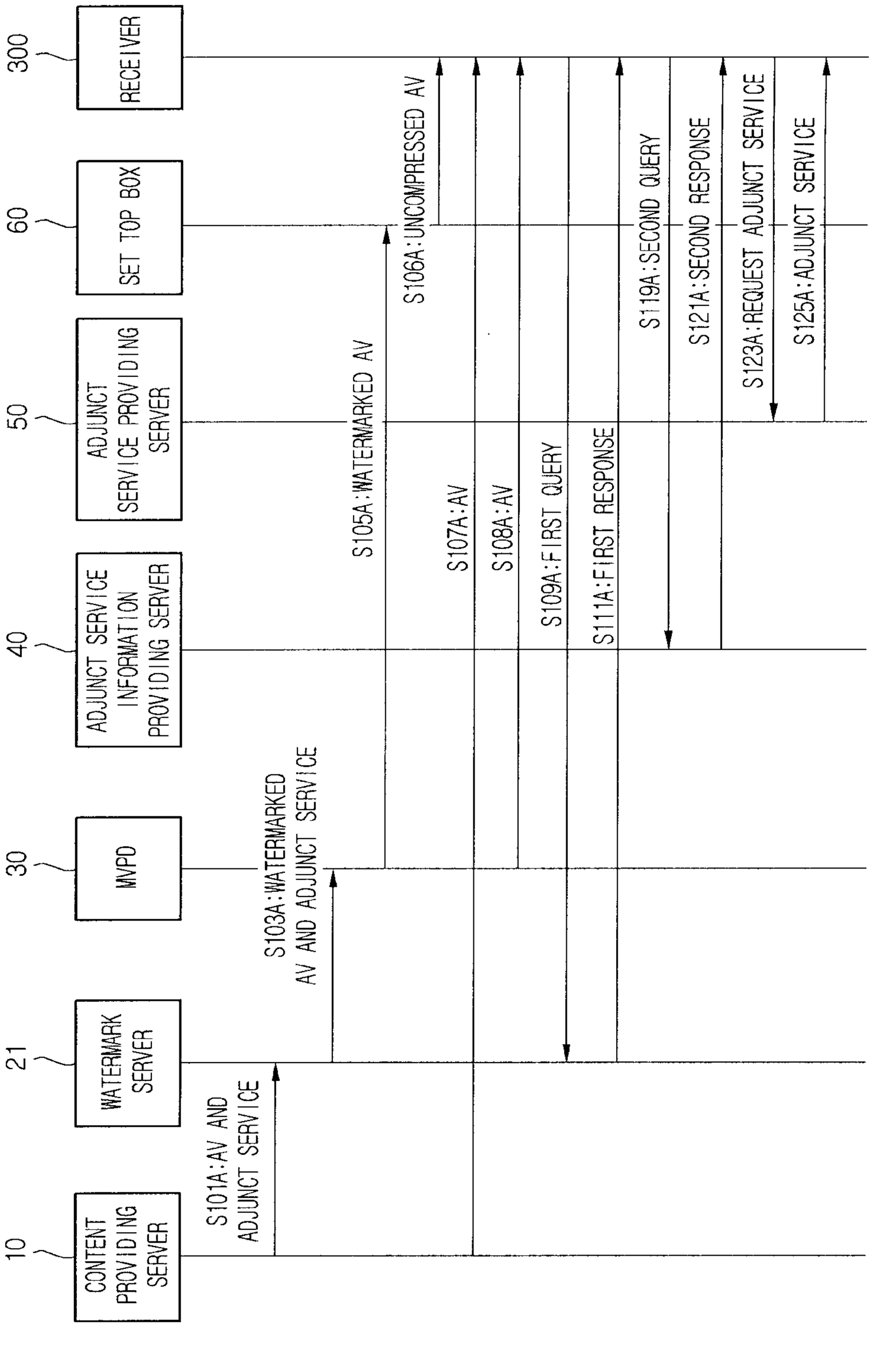

10
CONTENT PROVIDING SERVER
21
WATERMARK SERVER
30
MVPD
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
60
SET TOP BOX
300
RECEIVER
S101A:AV AND ADJUNCT SERVICE
S103A:WATERMARKED AV AND ADJUNCT SERVICE
S105A:WATERMARKED AV
S106A:UNCOMPRESSED AV
S107A:AV
S108A:AV
S109A:FIRST QUERY
S111A:FIRST RESPONSE
S119A:SECOND QUERY
S121A:SECOND RESPONSE
S123A:REQUEST ADJUNCT SERVICE
S125A:ADJUNCT SERVICE

FIG.106

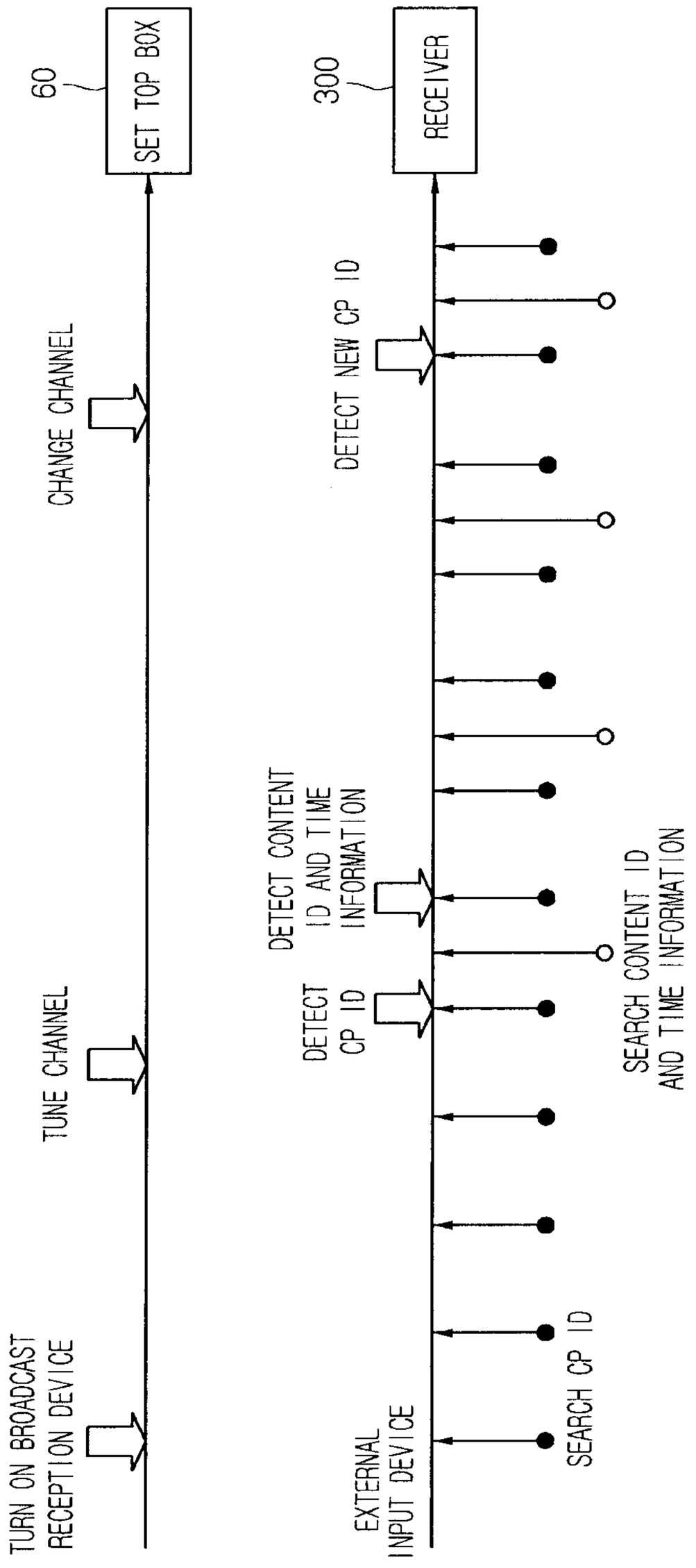

60
SET TOP BOX
300
RECEIVER
CHANGE CHANNEL
DETECT NEW CP ID
DETECT CONTENT ID AND TIME INFORMATION
TUNE CHANNEL
DETECT CP ID
SEARCH CONTENT ID AND TIME INFORMATION
TURN ON BROADCAST RECEPTION DEVICE
EXTERNAL INPUT DEVICE
SEARCH CP ID

FIG.107

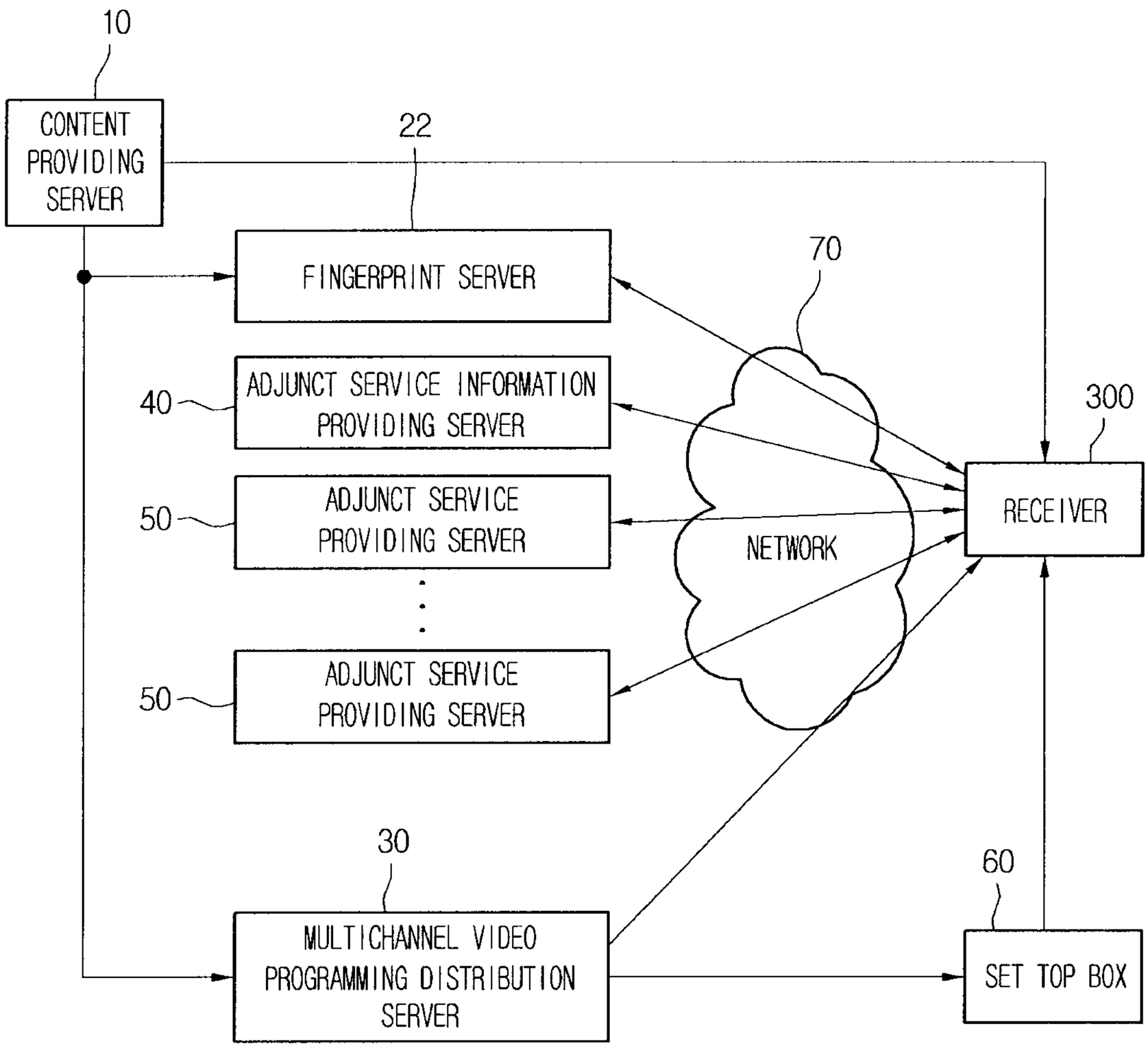

10
CONTENT PROVIDING SERVER
22
FINGERPRINT SERVER
70
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
300
50
ADJUNCT SERVICE PROVIDING SERVER
RECEIVER
NETWORK
50
ADJUNCT SERVICE PROVIDING SERVER
30
60
MULTICHANNEL VIDEO PROGRAMMING DISTRIBUTION SERVER
SET TOP BOX

FIG.108

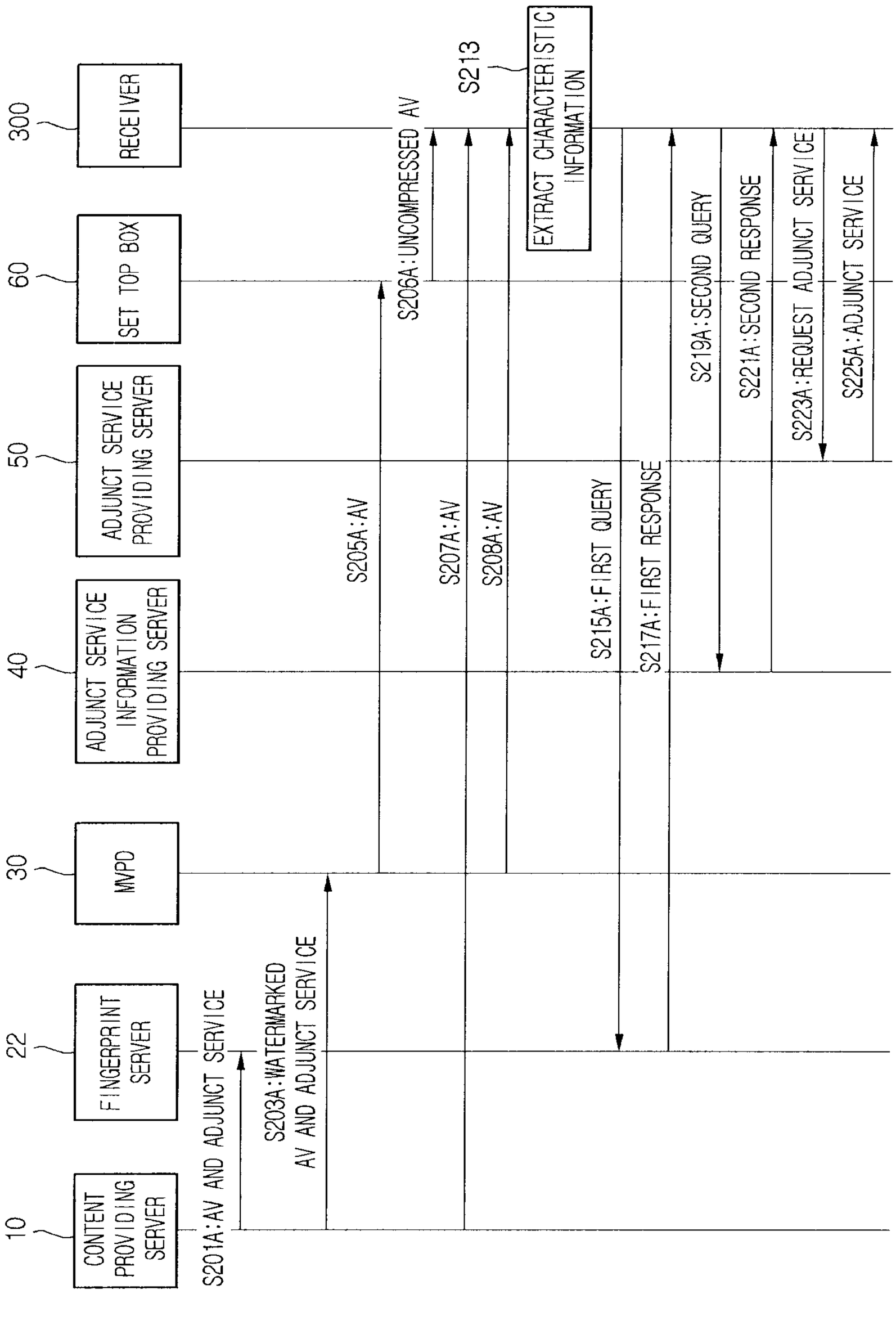

10
CONTENT PROVIDING SERVER
22
FINGERPRINT SERVER
30
MVPD
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
60
SET TOP BOX
300
RECEIVER
S201A:AV AND ADJUNCT SERVICE
S203A:WATERMARKED AV AND ADJUNCT SERVICE
S205A:AV
S206A:UNCOMPRESSED AV
S207A:AV
S208A:AV
S213
EXTRACT CHARACTERISTIC INFORMATION
S215A:FIRST QUERY
S217A:FIRST RESPONSE
S219A:SECOND QUERY
S221A:SECOND RESPONSE
S223A:REQUEST ADJUNCT SERVICE
S225A:ADJUNCT SERVICE

FIG.109

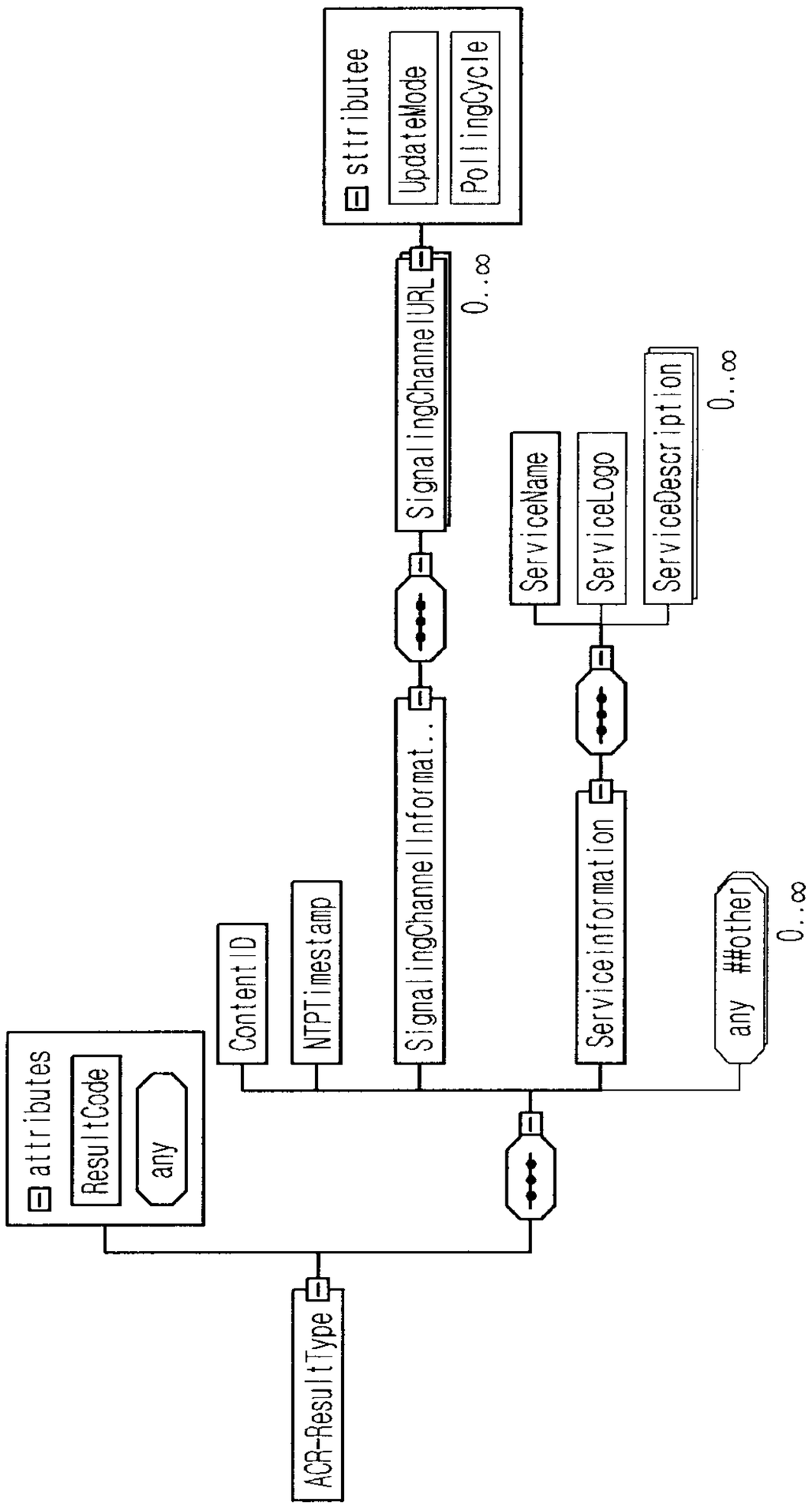

sttributee
UpdateMode
PollingCycle
8
attributes
ResultCode
any
ContentID
NTPTimestamp
SignalingChannelInformat..
SignalingChannelURL
0..∞
ServiceName
ServiceLogo
ServiceDescription
0..∞
Serviceinformation
any ##other
0..∞
ACR-ResultType

FIG.110

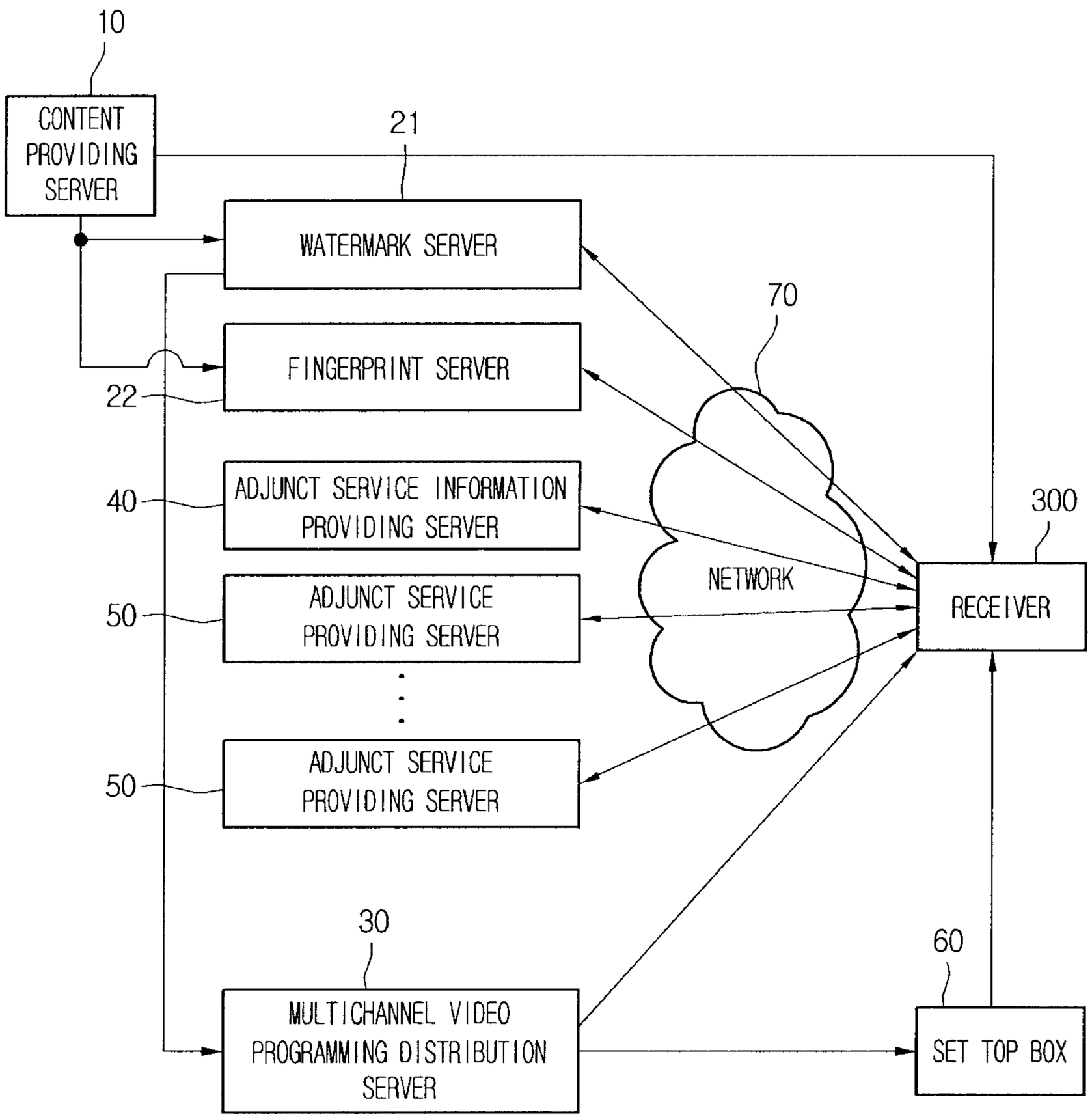

10
CONTENT PROVIDING SERVER
21
WATERMARK SERVER
FINGERPRINT SERVER
22
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
40
ADJUNCT SERVICE PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
50
70
NETWORK
300
RECEIVER
30
MULTICHANNEL VIDEO PROGRAMMING DISTRIBUTION SERVER
60
SET TOP BOX

FIG.111

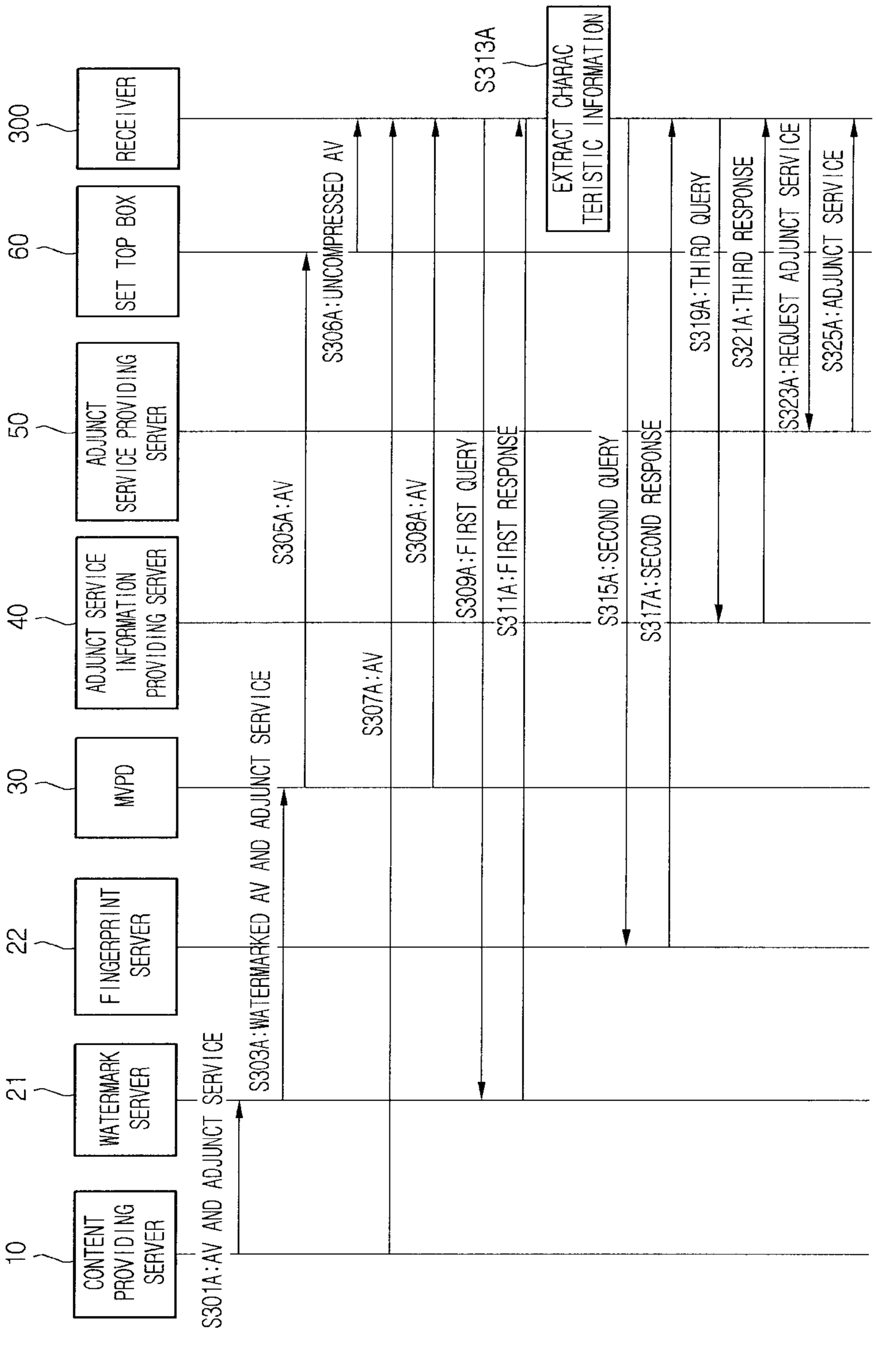
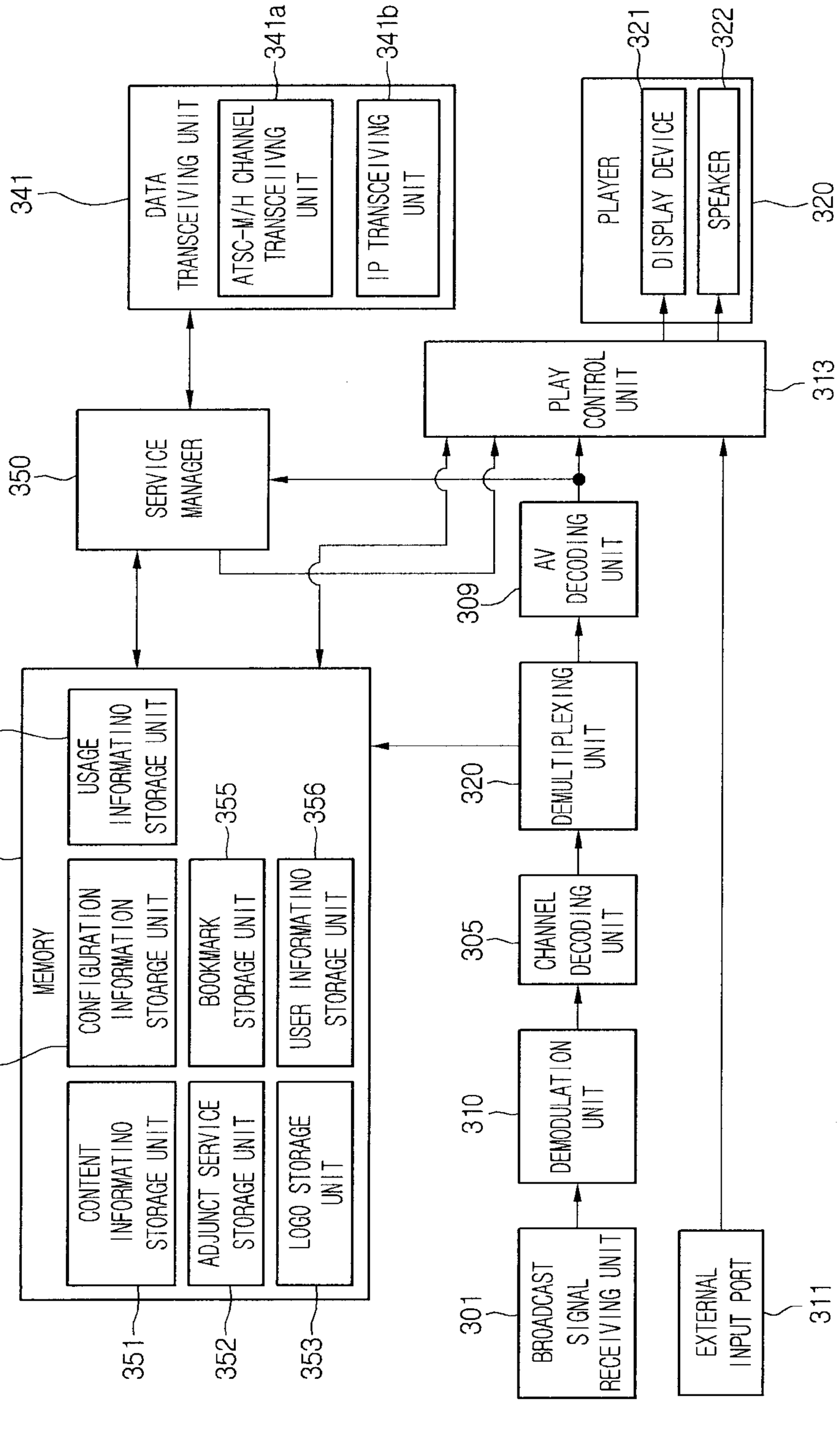

10
CONTENT PROVIDING SERVER
21
WATERMARK SERVER
22
FINGERPRINT SERVER
30
MVPD
40
ADJUNCT SERVICE INFORMATION PROVIDING SERVER
50
ADJUNCT SERVICE PROVIDING SERVER
60
SET TOP BOX
300
RECEIVER
S301A:AV AND ADJUNCT SERVICE
S303A:WATERMARKED AV AND ADJUNCT SERVICE
S305A:AV
S306A:UNCOMPRESSED AV
S307A:AV
S308A:AV
S309A:FIRST QUERY
S311A:FIRST RESPONSE
S313A
EXTRACT CHARACTERISTIC INFORMATION
S315A:SECOND QUERY
S317A:SECOND RESPONSE
S319A:THIRD QUERY
S321A:THIRD RESPONSE
S323A:REQUEST ADJUNCT SERVICE
S325A:ADJUNCT SERVICE

FIG.112


341
DATA TRANSCEIVING UNIT
341a
ATSC-M/H CHANNEL TRANSCEIVNG UNIT
341b
IP TRANSCEIVING UNIT
321
322
PLAYER
DISPLAY DEVICE
SPEAKER
320
313
PLAY CONTROL UNIT
350
SERVICE MANAGER
309
AV DECODING UNIT
320
DEMULTIPLEXING UNIT
305
CHANNEL DECODING UNIT
310
DEMODULATION UNIT
301
BROADCAST SIGNAL RECEIVING UNIT
311
EXTERNAL INPUT PORT
350
MEMORY
357
USAGE INFORMATINO STORAGE UNIT
355
356
354
CONFIGURATION INFORMATION STOARGE UNIT
BOOKMARK STORAGE UNIT
USER INFORMATINO STORAGE UNIT
CONTENT INFORMATINO STORAGE UNIT
ADJUNCT SERVICE STORAGE UNIT
LOGO STORAGE UNIT
351
352
353

BROADCASTING STATION

Real Time Service

Non Real Time Service

Real Time Service

Real Time Service + Non Real Time Service

Legacy Device

NRT Device