



(51) International Patent Classification:

H04L 12/14 (2006.01) *H04W 4/24* (2018.01)
H04M 15/00 (2006.01)

(21) International Application Number:

PCT/US2018/060561

(22) International Filing Date:

12 November 2018 (12.11.2018)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: **NOKIA TECHNOLOGIES OY** [FI/FI];
Karaportti 3, 02610 Espoo (FI).

(71) Applicant (for LC only): **NOKIA USA INC.** [US/US]; 200
S. Mathilda Avenue, Sunnyvale, California 94086 (US).

(72) Inventors: **SHARMA, Ranjan**; 4397 Haybury Drive,
New Albany, Ohio 43054 (US). **CAI, Yigang**; 23643
Royal Worlington Drive, Naperville, Illinois 60564 (US).
GARDELLA, Maryse; 3 Rue Louis Hubert, 78140 Velizy
(FR).

(74) Agent: **GOSNELL, Guy R.** et al.; ALSTON & BIRD LLP,
Bank of America Plaza, 101 South Tryon Street, Suite 4000,
Charlotte, North Carolina 28280-4000 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,
KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME,
MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,

(54) Title: METHOD AND APPARATUS FOR HANDLING FEATURE INCOMPATIBILITY IN A CHARGING SYSTEM OF A COMMUNICATION SYSTEM

		Service					
\	Fa	Fan	..	..	Fa2	Fa1	Fa0
IE							
	N1	-			M	-	M
	N2	M			M	M	M
	N3	-			M	M	-
	N4	M			-	-	M
	..						
	..				-	M	M
	<u>Nn</u>	-			M	-	-

Figure 3

(57) Abstract: A method, apparatus and computer program product cause a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features. Each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements. The method, apparatus and computer program product receive a first feature support response from the first charging server indicating whether each of the one or more required features is supported. The method, apparatus and computer program product determine whether the one or more required features are all supported by the second charging server. The method, apparatus and computer program product cause a second feature support query to be transmitted to a second charging server in a circumstance where the one or more required features are not all supported.



SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report (Art. 21(3))*

METHOD AND APPARATUS FOR HANDLING FEATURE INCOMPATIBILITY IN A CHARGING SYSTEM OF A COMMUNICATION SYSTEM

TECHNICAL FIELD

An example embodiment of the present invention generally relates to handling feature incompatibility in charging systems of a communication system, such as a fifth generation (5G) system.

5

BACKGROUND

Service providers provide voice and data services to end-users, via user equipment (UE), such as smart phones, laptop or tablet computers, video game systems or the like. Examples may include streaming audio, streaming video, Voice over Internet Protocol (VoIP), online gaming, etc. Charging for network service usage by a subscriber entails
10 reporting of usage for offline charging, which supports a post-paid charging model, or authorizing for network resource consumption prior to usage in online charging, which supports a pre-paid charging model.

In either case, the requests may be either offline – Accounting Request (ACR), or online – Credit Control Request (CCR), from the network elements with embedded
15 charging trigger functionality with the requests containing multiple information elements (IEs) that are sent to the (online or offline) charging system. The charging system needs to be able to parse and interpreted the IEs correctly.

Ideally, the charging system is capable of handling all IEs that are sent to the charging system for charging purposes. Each 3GPP release has a potential to add new
20 information elements in the charging specifications that a Network Element (NE)/Charging Triggering Function (CTF) may send and the charging domain must understand. When new IEs (or AVPs) are added to the charging specs, Diameter allows associating the “M-bit” such that 1) IEs with the M-bit set are interpreted and processed correctly by the receiver with a failure to do so resulting in the rejection of the message
25 and communication of the failure to the sender; and 2) IEs with the M-bit cleared are considered optional for interpretation and processing and the failure to do so does not result in rejection of the message containing the IE. New IEs may be introduced with their respective M-bits cleared, therefore, it does not mandate the charging domain to interpret and process the IEs correctly. Consequently, the charging domain can simply ignore those
30 IEs which are not supported. As a result, the extensibility of such a charging system is

limited.

BRIEF SUMMARY

A method, apparatus and computer program product are provided in accordance with an example embodiment to handle feature incompatibility in charging systems of a communication system, such as a fifth generation (5G) system.

In one example embodiment, a method is provided that includes causing a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features. Each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements. The method further includes receiving a first feature support response from the first charging server indicating whether each of the one or more required features is supported. The method further includes determining whether the one or more required features are all supported. The method further includes causing a second feature support query to be transmitted to a second charging server in a circumstance where the one or more required features are not all supported.

In some implementations of such a method, the method is utilized in conjunction with a charging triggering function that supports the one or more required features, and wherein information regarding one or more iterations of the one or more supported features is stored in the first charging server. In some embodiments, the method further includes receiving a second feature support response from the second charging server. The method further includes determining if the one or more required features are all supported by the second charging server. The method further includes in a circumstance where the one or more required features are not all supported, causing a third feature support query to be transmitted to a third charging server. In some embodiments, the first feature support response further comprises one or more support indication associated with each of the one or more required features. In some embodiments, each of the one or more support indication indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server. In some embodiments, the method further includes for each required feature: determining if the required feature is completely supported by the first charging server or if the required feature is not completely supported

and a predecessor feature of the required feature is supported; and causing the required feature to fall back to the predecessor feature in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported.

In another example embodiment, an apparatus is provided that includes at least one
5 processor and at least one memory including computer program code for one or more programs with the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus at least to cause a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features. Each of the one or more required features is
10 associated with a set of mandatory information elements and a set of non-mandatory information elements. The computer program code is further configured to, with the at least one processor, cause the apparatus to receive a first feature support response from the first charging server indicating if each of the one or more required features is supported. The computer program code is further configured to, with the at least one processor, cause
15 the apparatus to determine if the one or more required features are all supported by the second charging server. The computer program code is further configured to, with the at least one processor, cause the apparatus to cause a second feature support query to be transmitted to a second charging server in a circumstance where the one or more required features are not all supported.

20 In some implementations of such an apparatus, the apparatus is utilized in conjunction with a charging triggering function that supports the one or more required features; and information regarding one or more different iterations of the one or more supported features is stored in the first charging server. In some embodiments, the computer program code is further configured to, with the at least one processor, cause the
25 apparatus to receive a second feature support response from the second charging server. In some embodiments, the computer program code is further configured to, with the at least one processor, cause the apparatus to determine whether the one or more required features are all supported by the second charging server. In some embodiments, the computer program code is further configured to, with the at least one processor, cause the apparatus
30 to cause a third feature support query to be transmitted to a third charging server in a circumstance where the one or more required features are not all supported. In some embodiments, the first feature support response further comprises one or more support indication associated with each of the one or more required features. In some embodiments, each of the one or more support indication indicates one or more of:

whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server. In some embodiments, the computer program code is further configured to, with the at least one processor, cause the apparatus to: for each required feature: determine if the required feature is completely supported by the first charging server or if the required feature is not completely supported and a predecessor feature of the required feature is supported; and cause the required feature to fall back to the predecessor feature in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported.

In another example embodiment, a computer program product is provided that includes at least one non-transitory computer-readable storage medium having computer executable program code instructions stored therein with the computer executable program code instructions comprising program code instructions configured, upon execution, to cause a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features. Each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements. The computer executable program code instructions comprise program code instructions that are further configured, upon execution, to receive a first feature support response from the first charging server indicating whether each of the one or more required features is supported. The computer executable program code instructions comprise program code instructions that are further configured, upon execution, to determine whether the one or more required features are all supported. The computer executable program code instructions comprise program code instructions that are further configured, upon execution, to cause a second feature support query to be transmitted to a second charging server in a circumstance where the one or more required features are not all supported.

In some implementations of such a computer program product, the computer program product is utilized in conjunction with a charging triggering function that supports the one or more required features, and information regarding one or more different iterations of the one or more supported features is stored in the first charging server. In some embodiments, the computer executable program code instructions comprise program code instructions that are further configured, upon execution, to receive a second feature support response from the second charging server. In some embodiments,

the computer executable program code instructions comprise program code instructions that are further configured, upon execution, to determine whether the one or more required features are all supported by the second charging server. In some embodiments, the computer executable program code instructions comprise program code instructions that are further configured, upon execution, to cause a third feature support query to be transmitted to a third charging server in a circumstance where the one or more required features are not all supported. In some embodiments, the first feature support response further comprises one or more support indications associated with each of the one or more required features. In some embodiments, each of the one or more support indications indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server. In some embodiments, the computer executable program code instructions comprise program code instructions that are further configured, upon execution, to: for each required feature: determine if the required feature is completely supported by the first charging server or if the required feature is not completely supported and a predecessor feature of the required feature is supported; and cause the required feature to fall back to the predecessor feature in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported.

In another example embodiment, an apparatus is provided that includes means for causing a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features. Each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements. The apparatus further includes means for receiving a first feature support response from the first charging server indicating whether each of the one or more required features is supported. The apparatus further includes means for determining whether each of the one or more required features are all supported. The apparatus further includes means for causing a second feature support query to be transmitted a second charging server in a circumstance where the one or more required features are not all supported.

In some implementations of such an apparatus, the apparatus is utilized in conjunction with a charging triggering function that supports the one or more required features; and information regarding one or more different iterations of the one or more

supported features is stored in the first charging server. In some embodiments, apparatus further includes means for receiving a second feature support response from the second charging server. . In some embodiments, apparatus further includes means for determining whether the one or more required features are all supported by the second charging server.

5 In some embodiments, apparatus further includes means for causing a third feature support query to be transmitted to a third charging server in a circumstance where the one or more required features are not all supported. In some embodiments, the first feature support response further comprises one or more support indication associated with each of the one or more required features. In some embodiments, each of the one or more support

10 indication indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server. In some embodiments, apparatus further includes means for: for each required feature: means for determining if the required feature is

15 completely supported by the first charging server or if the required feature is not completely supported and a predecessor feature of the required feature is supported; and means for causing the required feature to fall back to the predecessor feature in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported.

20

BRIEF DESCRIPTION OF THE DRAWINGS

Having thus described certain example embodiments of the present disclosure in general terms, reference will hereinafter be made to the accompanying drawings, which are not necessarily drawn to scale, and wherein:

25 Figure 1 is a block diagram of an apparatus that may be specifically configured in accordance with an example embodiment of the present disclosure;

Figures 2A and 2B illustrate example charging systems for offline charging and online charging in accordance with an example embodiment of the present disclosure;

Figure 3 illustrates an example matrix representing the supported features of charging peers in accordance with an example embodiment of the present disclosure; and

30

Figure 4 is a flowchart illustrating a set of operations performed, such as by the apparatus of Figure 1, in accordance with an example embodiment of the present disclosure.

DETAILED DESCRIPTION

Some embodiments will now be described more fully hereinafter with reference to the accompanying drawings, in which some, but not all, embodiments of the invention are shown. Indeed, various embodiments of the invention may be embodied in many different
5 forms and should not be construed as limited to the embodiments set forth herein; rather, these embodiments are provided so that this disclosure will satisfy applicable legal requirements. Like reference numerals refer to like elements throughout. As used herein, the terms “data,” “content,” “information,” and similar terms may be used interchangeably to refer to data capable of being transmitted, received and/or stored in accordance with
10 embodiments of the present invention. Thus, use of any such terms should not be taken to limit the spirit and scope of embodiments of the present invention.

Additionally, as used herein, the term ‘circuitry’ refers to (a) hardware-only circuit implementations (e.g., implementations in analog circuitry and/or digital circuitry); (b) combinations of circuits and computer program product(s) comprising software and/or
15 firmware instructions stored on one or more computer readable memories that work together to cause an apparatus to perform one or more functions described herein; and (c) circuits, such as, for example, a microprocessor(s) or a portion of a microprocessor(s), that require software or firmware for operation even if the software or firmware is not physically present. This definition of ‘circuitry’ applies to all uses of this term herein,
20 including in any claims. As a further example, as used herein, the term ‘circuitry’ also includes an implementation comprising one or more processors and/or portion(s) thereof and accompanying software and/or firmware. As another example, the term ‘circuitry’ as used herein also includes, for example, a baseband integrated circuit or applications processor integrated circuit for a mobile phone or a similar integrated circuit in a server, a
25 cellular network device, other network device, and/or other computing device.

As defined herein, a “computer-readable storage medium,” which refers to a non-transitory physical storage medium (e.g., volatile or non-volatile memory device), can be differentiated from a “computer-readable transmission medium,” which refers to an electromagnetic signal.

30 Service providers provide voice and data services to end-users, via user equipment (UE), such as smart phones, laptop or tablet computers, video game systems or the like. Examples may include streaming audio, streaming video, Voice over Internet Protocol (VoIP), online gaming, etc. Charging for network service usage by a subscriber entails reporting of usage for offline charging, which supports a post-paid charging model, or

authorizing for network resource consumption prior to usage in online charging, which supports a pre-paid charging model.

In either case, the requests may be either offline – Accounting Request (ACR), or online – Credit Control Request (CCR), from the network elements with embedded
5 charging trigger functionality containing multiple information elements (IEs) that are sent to the (online or offline) charging system. The charging system needs to be able to parse and interpret the IEs correctly.

Considering offline charging, network elements (NEs) with the embedded Charging Triggering Function (CTF) report usage consumption to the Charging Data
10 Functions (CDFs) responsible for generating Charging Data Records (CDRs). These CDRs provide details of user session and resource consumption during the session to enable a rating/billing system to post charges to the subscriber account. In contrast, an online charging system is responsible for prior authorization on a per subscriber basis of the resource consumption and an online charging system closely governs the duration and
15 Quality of Service (QoS) of the call/session, working with subscriber account balance and subscription class.

Irrespective of the mode of charging employed, the communication between the CTFs and the charging system is based on providing IE (information elements) from the CTF to the charging system, which are interpreted and processed by the charging system
20 before a response is sent back to the CTF. The request sent from the NE/CTF contains a header part and a payload part. In a 4th generation (4G) or 3rd generation (3G) communication system which uses Diameter as the protocol for charging, the request contains a Diameter Header, which essentially identifies the Diameter command, and a number of information elements named attribute-value pairs (AVPs). These IEs are
25 associated with flags, one of which is the M-bit, which when set signifies that in order to understand the charging data request, the charging system must understand and interpret the associated IE correctly. In a 5G system, Diameter and M-bit may not be utilized but protocols serving similar functions may be utilized. Although an example embodiment of the present disclosure is provided in conjunction with the Diameter protocol, the present
30 disclosure is not limited by the Diameter protocol and may be utilized in conjunction with other protocols serving similar functions in various communication systems.

The response from the charging system to the NE/CTF may be positive or negative. A negative response may indicate that the charging system is not able to interpret the charging data request correctly. For example, when the charging system cannot successfully parse and interpret one or more IEs that have the M-bit set, the charging system would return a negative response.

Ideally, the charging system must be capable of handling all IEs that are sent to the charging system for charging purposes. Each 3GPP release has a potential to add new information elements in the charging specifications that a NE/CTF may send and the charging domain must understand. When new IEs (or AVPs) are added to the charging specifications, Diameter allows associating the “M-bit” such that 1) IEs with the M-bit set must be interpreted and processed correctly by the receiver and a failure to do so must result in the rejection of the message and communication of the failure to the sender; and 2) IEs with the M-bit cleared are considered optional for interpretation and processing and the failure to do so does not result in rejection of the message containing the IE.

Network element vendors frequently choose to introduce new IEs with their M-bit cleared, and have these IEs associated with existing commands. This method of adding new IEs does not mandate the charging domain to interpret and process the IEs correctly. Consequently, the charging domain can simply ignore those IEs which are not supported. As a result, the extensibility of such a charging system is limited.

The newly incorporated IEs are not understood and acted upon by the receiver. Therefore, it is questionable if a successful Diameter response indicates if the charging domain has understood the charging data request completely and correctly. When an NE/CTF receives a DIAMETER_SUCCESS response, it is not certain if the charging domain has duly considered and processed the additional IEs.

The changes to IEs are initiated at the CTFs and cascaded downstream. Therefore, the addition of IEs by a NE/CTF with their M-bit cleared for compatibility reasons does not support extensibility and the charging system can silently discard such “optional” IEs while reporting success in the response to the NE/CTFs. Therefore, a NE/CTF would not be able to determine such an issue because there is no error response from the charging domain. The NE/CTF may continue as if the added IEs have been accepted and processed appropriately at the receiver, which creates an issue.

When the CDF generates a CDR for the charging event, currently, the standards require only the record type field of the CDR to be mandatory (plus a few more for the PS domain CDRs), while the rest of the fields are deemed optional. Unless additional checks

are deployed for checking the presence of certain fields in the CDR, the fact that newly added IEs do not manifest in the CDR goes undetected and is not considered an error. Such an undetected incompatibility error creates an issue in charging.

The 3rd Generation Partnership Project (3GPP) release 6 introduced the notion of supported features in Diameter via the Supported-Feature AVP. Technical Specification (TS) 29.229 section 7.2, which defines the principles associated with this AVP. The basic idea is that Diameter applications are expected to support a base set of IEs (AVPs) for the application in question. Evolution of the application may entail tuning the application without modifying the charging aspects, or via adding “features”, that require adding to the charging interface. Features are not expected to be backward-compatible; so if the CTFs and the charging domain are not in lockstep, it cannot be presumed that the features would be supported at CDF or OCS (Online Charging System).

The Supported-Features AVP (code 628) is a grouped AVP of the following syntax:

```

15          { Vendor-Id }
           { Feature-List-ID }
           { Feature-List }
           *[AVP]

```

The Feature-List AVP contains a list of supported features of the origin host. The Vendor-Id AVP and the Feature-List-ID AVP may together identify which feature list is carried in the Supported-Features AVP for the Application-ID present in the command header. Where a Supported-Features AVP is used to identify features that have been defined by 3GPP, the Vendor-Id AVP may contain the vendor ID of 3GPP. Vendors may define proprietary features, but it is strongly recommended that the possibility is used only as the last resort. The Supported-Features AVP is used to identify features that have been defined by a vendor other than 3GPP and it may contain the vendor ID associated with the vendor. If there are multiple feature lists defined by the same vendor and the same application, the Feature-List-ID AVP may differentiate those lists from one another. The destination host may use the value of the Feature-List-ID AVP to identify the feature list. TS 29.229 further defines that, if new AVPs are added to the commands because of the new feature, the new AVPs may have the ‘M’ bit cleared and the AVP shall not be defined mandatory in the command ABNF. The support for a feature may be defined to be mandatory behaviour of a node. However, this mechanism in TS 29.229 does not alleviate

the extensibility problem. If a feature adds mandatory IEs, the feature evolves into a different feature which is not backward compatible. On the other hand, if a feature adds IEs with the M-bit cleared, then the charging domain is not obliged to interpret the added IEs. Either way, the current state of the art is insufficient to address the extensibility and compatibility paradigms.

Accordingly, a method, apparatus and computer program product are provided in accordance with an example embodiment to handle feature incompatibility in charging systems, such as a charging system of a 5G communication system. Even though some example embodiments are described in conjunction with a charging system in a 5G system, other example embodiments may be utilized with other charging systems with minor modifications.

Figure 1 illustrates an example apparatus that may be provided to embody the various components in a charging system, such as a charging system in a 5G communication system, for example, a CTF and a charging server (CS). As illustrated in Figure 1, the apparatus 10 of an example embodiment includes, is associated with or is otherwise in communication with processing circuitry 12, a memory 14, and a communication interface 16. As shown in Figure 1, the apparatus of an example embodiment includes, is associated with or is otherwise in communication with a processor 12, an associated memory 14 and a communication interface 16.

The processor 12 (and/or co-processors or any other circuitry assisting or otherwise associated with the processor) may be in communication with the memory device 14 via a bus for passing information among components of the apparatus 10. The memory device may be non-transitory and may include, for example, one or more volatile and/or non-volatile memories. In other words, for example, the memory device may be an electronic storage device (e.g., a computer readable storage medium) comprising gates configured to store data (e.g., bits) that may be retrievable by a machine (e.g., a computing device like the processor). The memory device may be configured to store information, data, content, applications, instructions, or the like for enabling the apparatus to carry out various functions in accordance with an example embodiment of the present disclosure. For example, the memory device could be configured to buffer input data for processing by the processor. Additionally or alternatively, the memory device could be configured to store instructions for execution by the processor.

The apparatus 10 may, in some embodiments, be embodied in various computing devices as described above. However, in some embodiments, the apparatus may be

embodied as a chip or chip set. In other words, the apparatus may comprise one or more physical packages (e.g., chips) including materials, components and/or wires on a structural assembly (e.g., a baseboard). The structural assembly may provide physical strength, conservation of size, and/or limitation of electrical interaction for component circuitry included thereon. The apparatus may therefore, in some cases, be configured to implement an embodiment of the present invention on a single chip or as a single “system on a chip.” As such, in some cases, a chip or chipset may constitute means for performing one or more operations for providing the functionalities described herein.

The processor 12 may be embodied in a number of different ways. For example, the processor may be embodied as one or more of various hardware processing means such as a coprocessor, a microprocessor, a controller, a digital signal processor (DSP), a processing element with or without an accompanying DSP, or various other circuitry including integrated circuits such as, for example, an ASIC (application specific integrated circuit), an FPGA (field programmable gate array), a microcontroller unit (MCU), a hardware accelerator, a special-purpose computer chip, or the like. As such, in some embodiments, the processor may include one or more processing cores configured to perform independently. A multi-core processor may enable multiprocessing within a single physical package. Additionally or alternatively, the processor may include one or more processors configured in tandem via the bus to enable independent execution of instructions, pipelining and/or multithreading.

In an example embodiment, the processor 12 may be configured to execute instructions stored in the memory device 14 or otherwise accessible to the processor. Alternatively or additionally, the processor may be configured to execute hard coded functionality. As such, whether configured by hardware or software methods, or by a combination thereof, the processor may represent an entity (e.g., physically embodied in circuitry) capable of performing operations according to an embodiment of the present disclosure while configured accordingly. Thus, for example, when the processor is embodied as an ASIC, FPGA or the like, the processor may be specifically configured hardware for conducting the operations described herein. Alternatively, as another example, when the processor is embodied as an executor of instructions, the instructions may specifically configure the processor to perform the algorithms and/or operations described herein when the instructions are executed. However, in some cases, the processor may be a processor of a specific device (e.g., an image processing system) configured to employ an embodiment of the present invention by further configuration of

the processor by instructions for performing the algorithms and/or operations described herein. The processor may include, among other things, a clock, an arithmetic logic unit (ALU) and logic gates configured to support operation of the processor.

The communication interface 16 may be any means such as a device or circuitry embodied in either hardware or a combination of hardware and software that is configured to receive and/or transmit data from/to a network. In this regard, the communication interface may include, for example, an antenna (or multiple antennas) and supporting hardware and/or software for enabling communications with a wireless communication network. Additionally or alternatively, the communication interface may include the circuitry for interacting with the antenna(s) to cause transmission of signals via the antenna(s) or to handle receipt of signals received via the antenna(s). In some environments, the communication interface may alternatively or also support wired communication. As such, for example, the communication interface may include a communication modem and/or other hardware/software for supporting communication via cable, digital subscriber line (DSL), universal serial bus (USB) or other mechanisms.

Referring now to Figures 2A and 2B, example charging systems for offline charging (Figure 2A) and online charging (Figure 2B) are depicted. Rf(s) in Figure 2A and Ro(s) in Figure 2B denote reference points from the CTF 20 to the CDF 22A and the OCF 22B, respectively. The Rf(s) and Ro(s) may be utilized for the transport of charging events. The CTF 20 may be an integrated component of the core network (CN) domain, service codes, or sub-systems.

A charging event may be a set of charging information that would be forwarded by the CTF 20 towards a CS, such as the CDF 22A (offline charging) or towards the OCS 22B (online charging). A set of charging data records may be associated with the charging event.

The CDF and OCS may generate CDRs based on the charging events received. A CDR is a formatted collection of information about a chargeable event (e.g. time of call set-up, duration of the call, amount of data transferred, or the like) for later use in billing and accounting. For each party to be charged for parts of or all charges of a chargeable event a separate CDR is generated. More than one CDR may be generated for a single chargeable event, e.g. because of its long duration, or because more than one charged party is to be charged. The CDRs generated may be transferred to the Charging Gateway Function (CGF) via the Ga reference points. The CGFs 24 act as gateways between the

3GPP network and the Billing Domain 26. The Billing Domain 26 may be a billing system or a billing mediation device configured to generate bills according to pre-defined rules.

In some embodiments, when NE/CTFs are deployed, it is not known a-priori if the NE/CTF would serve a specific feature segment (such as WLAN charging) or if the
5 NE/CTF would be used as a general-purpose NE/CTF serving multiple features. From a service provider (or operator) perspective, it is advantageous to opt for the deployment of general-purpose network components and building blocks. Such general-purpose network components are able to address the feature incompatibility issue previously discussed.

In some embodiments, when an NE/CTF starts session reporting for offline
10 charging purposes (or starts asking for authorization in case of online charging), it is not known if the session being reported is going to use specific features and if so, which features. Therefore, it is not possible for an NE/CTF to check with the charging system as to whether one or more features are supported by the latter beforehand. When a charging session/event begins, the NE/CTF cannot anticipate the feature-set that would be used by
15 the UE and the NE/CTF cannot therefore ask if the charging system supports that feature-set at the beginning of the session. For offline charging, the consumed resources are reported post-facto, that is, the NE/CTF informs the charging system about resource consumption related to a specific feature after the consumption has occurred. If the charging system is unable to provide a satisfactory response to the charging data request,
20 one possible option for the NE is to seek support for the features at an alternative charging system. However, this approach would result in splitting the charging data report across more than one charging system, and therefore this approach is computationally demanding at the billing mediation that must handle incomplete CDRs and reconcile the incomplete CDRs subsequently.

In some embodiments, the Supported-Features mechanism along with a fallback
25 mechanism are utilized at the NE/CTF and the charging domain to handle such feature incompatibility. The charging peers may exchange the supported feature capability at the time of initial conversation in the event that a charging event occurs, outside of a charging session reporting. Once the supported feature capability exchange finishes, the charging
30 peers decide on the supported features, or their fallback features, or decide to seek alternative peers that are better suited because of feature compatibility.

A feature is denoted herein as F_{xy}, where both x and y are represented by integers. In some alternative embodiments, x and y may be represented by other characters instead. x denotes an identifier associated with the feature and y denotes an iteration/version

associated with the feature with a feature with a larger value of y being a successor feature to a feature with a smaller value of y and, conversely, a feature with a smaller value of y being a predecessor feature to a feature with a larger value of y . $y=0$ denotes the first feature iteration, $y=1$ denotes the evolution of the first iteration, $y=2$ denotes the evolution of the second iteration and so on. In some embodiments, an example feature Fa1 is not backwards-compatible with Fa0 and Fa2 is not backwards-compatible with Fa1. If Fa1 is backwards-compatible with an earlier iteration, such iteration may be treated as the same iteration as Fa1 for the sake of simplicity. In some embodiments, Fa1 is backwards-compatible with an earlier iteration such as Fa0.

In some embodiments, from the perspective of the charging system/server, Fa0, Fa1, and Fam (m denotes any integer) are considered as separate features with no apparent connection available to deduce the relationship between these iterations. The feature names may not be expected to be indicative of their functionality, nor is the evolution of the features going to provide a hint at the relationship. Therefore, unless it is specifically maintained by the corresponding peers (CTFs and charging domain), the sequence deduction is not possible.

In some embodiments, each charging peer in the form of a charging system/server or a CTF may store support indication information for each feature. For example, considering 4 features only, the following support indication information at 2 NE/CTF and

2 charging servers are provided as examples:

- NE/CTF1 – Requires support for features Fa3, Fb2, Fc5 and Fd1
- NE/CTF2 – Requires support for features Fa2, Fb1, Fc5
- Charging server 1 (CS1) – Provides support for features Fa2, Fb2 and Fc6 with implicit support of a predecessor feature of Fb2, such as Fb1. In some embodiments, support of Fb2 means implicit support of Fb0 and Fb1 (such as by utilizing a fallback mechanism described later)
- Charging server 2 (CS2) – Provides support for features Fa2, Fb3, Fc5 and Fd2 with implicit support of the predecessor features.

Figure 3 illustrates an example matrix representing the supported features of each of these entities. For a service feature, Fa in the example, a column represents the current supported feature present at the charging peer (CTF or charging host function (CDF) which may be a CS). The right-most column represents the first iteration of the service feature, the row adjacent and to the left represents the next iteration until the leftmost column, which then denotes the latest successor feature of a service feature. In this case, the current supported feature is Nth iteration, denoted by 'Fan' for the service Fa0.

The list of IEs required in each service feature implementation is denoted by the mandatory presence ('M' against the IE in the supported column) or non-mandatory or optional presence ('-' against the IE in the supported column). Accordingly, for Fa0, the mandatory IEs are IE N1, N2, N4, ..., and optional IEs are N3, ..., Nn. Fa1 requires the mandatory presence of IE N2, N3 ... and the optional presence of N1, N4, ..., Nn. In some embodiments, additional supported services at the peers would have their individual Service-IE matrices. Therefore, if a peer supports 10 services, the peer may maintain 10 Service-IE matrices. Note that matrices are provided merely as an example and another structure that provides effectively the same functionality for the purpose of keeping accurate information may also be utilized.

Referring now to Figure 4 the operations performed by a CTF which may be embodied by the apparatus illustrated in Figure 1 in accordance with an example embodiment are illustrated.

As shown in block 402, the CTF includes means, such as the communication interface 16 and/or the processing circuitry 12, for transmitting a first feature support query to a first charging server querying whether the first charging server supports the one or more required features. A required feature is a feature required and supported by the CTF for the purpose of generating charging events. For example, NE/CTF1, may send such query to CS1 first. If there are multiple CSs available, the order in which the query is sent is customizable. In addition, the logic which is used to choose a CS to be queried first is customizable as well. Implementations may differ according to different embodiments. A round-robin approach, or a weighted approach, or a preferential approach based on parameters of their choice are all possible choices and other approaches may be possible.

As shown in block 404, the CTF includes means, such as the communication interface 16 and/or the processing circuitry 12, for receiving a first feature support response from the first charging server indicating if each of the one or more required features is supported. In some embodiments, a feature support response, such as the first feature support response, may further include one or more support indications associated with each of the one or more supported features. In some embodiments, each of the one or more support indications is a Boolean response indicating whether a feature associated with the support indication is supported or not supported. In some embodiments, the one or more support indications indicates whether the required feature is completely supported, or if the charging server supports a predecessor feature or successor feature of

the required feature, or if no predecessor feature or successor feature of the required feature is supported.

Using the example in Figure 3, the query may ask if CS1 supports features Fa3, Fb2, Fc5 and Fd1 (which are all required features). A response from CS1 may indicate that CS1 can only provide support for Fa2, Fb2 and Fc5. Notice that Fd1 is not supported at CS1, nor is Fd0 (using the previously described notation convention). In some embodiments, the NE/CTF1 may choose to query another CS as an alternative.

As shown in block 406, the CTF includes means, such as the communication interface 16 and/or the processing circuitry 12, for determining if the one or more required features are all supported. Since Fd1 is not supported at CS1, nor is Fd0, the features are not all supported. In the event that the features are all supported, the CTF is paired with the first charging server. In some embodiments, the CTF requires the CS1 to completely support all required features to be considered that all features are supported. In some embodiments, the CTF only requires the CS1 to support a predecessor feature of each of the required features. In some embodiments, the CTF only requires the CS1 to support a successor feature of each of the required features. In some embodiments, the CTF only requires the CS1 to support either a successor feature or a predecessor feature of each of the required features. In such embodiments, the CTF may calculate an overall support rank of the CS1 and compare it with the overall support rank of other CSs available to the CTF. One example of calculating a support rank may be calculating the number of features completely supported. In some embodiments, in the event that a charging event is generated based on the observation of network resource usage, the CTF may transfer the charging event to the first charging server if the CS1 met the requirement of supporting all required features.

As shown in block 408, the CTF includes means, such as the communication interface 16 and/or the processing circuitry 12, for transmitting a second feature support query to a second charging server in the event that not all of the features are supported by the first charging server. In some embodiments, operations 402 to 408 may be repeated until all charging servers available to the CTF are queried or a CS meeting the requirement is found. In some embodiments, in the event that a charging event is generated based on the observation of network resource usage, the CTF may transfer the charging event to the charging server that met the requirement of supporting all required features.

For example, the NE/CTF1 may then send a query to CS2 and ask for support of features Fa3, Fb2, Fc5 and Fd1. Since CS2 provides support for predecessor and successor

features of the required features in the form of Fa2, Fb3, Fc5 and Fd2 and the desired features are not completely supported at NE/CTF1, CS2 responds such that:

- 5 - If the required feature is supported in an identical manner, the response indicates that the feature is completely supported. In this example, Fc5 would be indicated as part of the response.
- If a successor feature of the required feature is supported at the CS, then the response indicates the feature required by the NE/CTF. In this example, Fb2 and Fd1 would be indicated as part of the response
- 10 - If the predecessor feature of required feature is supported at the CS, then the CS response may provide the supported feature at the CS. In this example, Fa2 would be an instance of this rule and included as part of the response.

In some embodiments, the feature support query may take the format of a Capability-Feature-Request (CFR), indicated by the command code set to 325 (subject to
15 standardization) and the command flag's 'R' bit set. The charging peer (CTF) may send the CFR to a charging host (CDF/CS) to inquire about the latter's capabilities in regards to supporting specific service features. An example CFT is provided below:

Message Format
 20 <CFR> ::= < Diameter Header: 325, REQ >
 { Origin-Host }
 { Origin-Realm }
 1* { Host-IP-Address }
 { Vendor-Id }
 { Product-Name }
 25 { Supported-Features }
 [Origin-State-Id]
 * [Supported-Vendor-Id]
 * [Auth-Application-Id]
 * [Acct-Application-Id]
 30 * [Vendor-Specific-Application-Id]
 [Firmware-Revision]
 * [AVP]

In some embodiments, the feature support response may take the format of a Capability-Feature-Answer (CFA), indicated by the command code set to 325 (subject to
35 standardization) and the command flag's 'R' bit cleared. The charging host (CDF/CS) may send the CFA to the CTF in response to a CFR command initiated by the charging peer (CTF). An example CFA is provided below:

Message Format
 40 <CFA> ::= < Diameter Header: 325 >
 { Origin-Host }
 { Origin-Realm }
 { Result-Code }
 { Supported-Features }
 [Error-Message]

* [AVP]

In the CFR, the CTF will use the Supported-Features AVP to query the CDF about its support of the required features. The Supported-Features grouped AVP may be defined as:

5 Supported-Features ::= < AVP header: 628 10415 >
 { Vendor-Id }
 { Feature-List-ID }
 { Feature-List }
 *[AVP]

10

The Vendor-Id may identify 3GPP with its vendor ID for identifying features defined by 3GPP, or may indicate proprietary features developed by a vendor. The Feature-List is an m-bit field (m may equal to 32 in some embodiments), where each bit represents a feature. Feature-List-ID is a list of unique identifiers corresponding to the features supported in the Feature-List. If there are multiple feature lists defined by a vendor, the Feature-List-ID AVP differentiates those lists from one another. The destination host will use the value of the Feature-List-ID AVP to identify the feature list.

For example, for the sake of brevity, considering feature Fa5 only, the CTF desires support for Fa5, while the CDF can provide support for Fa4, if Fa4 is defined in Feature-List-Id 1 as the MSB of the Feature-List and Fa5 is defined in Feature-List-Id 2 as the LSB of the Feature-List. The CFR from the CTF to the CDF will contain the Supported-Features AVP corresponding to Fa5 as follows:

20 Supported-Features ::= < AVP header: 628 10415 >
 { Vendor's assigned Vendor-Id }
 { 0001h in the Feature-List-ID }
 { 2 in the Feature-List }
 *[AVP]

The CFA from the CDF to the CTF may contain the Supported-Features AVP corresponding to Fa4 as follows:

30 Supported-Features ::= < AVP header: 628 10415 >
 { Vendor's assigned Vendor-Id }
 { 1000h in the Feature-List-ID }
 { 1 in the Feature-List }
 *[AVP]

The operations of 402 to 408, that is, the operations of transmitting queries to find a CS that supports all required features may be repeated for multiple times. For example, the operations may be repeated until all CSs available to the CTF are queried or until a CS that supports all required features is found. In some embodiments, falling back from a feature to a predecessor feature may be supported by a CTF or a CS.

In the example previously described, the demands of NE/CTF1 cannot be fully met with either CS1 or CS2. However, CS2 provides a close match to what is required by NE/CTF1. With the feature association and sequence of evolution maintained at both the NE/CTF and the CS, the query-response becomes more intelligent than a simple Boolean function.

The fallback may occur in this example in the response from CS2 in that Fa2 is indicated instead of non-support of Fa3 which was required by the NE/CTF1. The NE/CTF1 may be capable of falling back from Fa so as to go from Fa3 to Fa2. In some embodiments, the CTF may send an updated CFR. One example is provided below:

Supported-Features ::= < AVP header: 628 10415 >
 { Vendor's assigned Vendor-Id }
 { 1000h in the Feature-List-ID }
 { 1 in the Feature-List }
 *[AVP]

In some embodiments, CFA response from the CDF reflects the accepted/negotiated service feature support. One example is provided below:

Supported-Features ::= < AVP header: 628 10415 >
 { Vendor's assigned Vendor-Id }
 { 1000h in the Feature-List-ID }
 { 1 in the Feature-List }
 *[AVP]

In this embodiment, the peers may need to maintain a sequence of service feature evolution (e.g., Fa0 → Fa1 → Fa2 ...). In this regard, the evolution is maintained with the 2 IEs that form part of the Supported-Features AVP, namely, the Feature-List-ID and the Feature-List. So, Fa0 may be represented as a {Feature-List-ID, Feature-List} tuple and Fa1 likewise. In the example above, Fa4 is {1, 1000h} while Fa5 is {2, 0001h}. An example evolution link is provided below:

... → {1, 1000h} → {2, 0001h} → ...

Because NE/CTF1 agreed to the support provided by the CS2, the set of IEs in the charging event to be transmitted would pertain to Fa2 (instead of Fa3), Fb2, Fc5 and Fd1. A similar adjustment is performed at the CS2, where CS2 expects to apply the necessary checks for syntax in relation to Fa2, Fb2 (instead of Fb3), Fc5 and Fd1 (instead of Fd2).

5 To elaborate, consider the set of IEs that are supported for Fb2 and Fb3:

Fb2 = {Fb20, Fb21, Fb22, ..., Fb2n}

Fb3 = {Fb30, Fb31, Fb32, ..., Fb3m}

Between these, the following relationships apply:

10 M may or may not be equal to N

Fb2 $\neq$ Fb3, or Fb2 $\cap$ Fb3 $\neq$ NULL

Fb2 $\subset$ Fb3 may be true, but Fb2 $\subseteq$ Fb3 may not be true

15 In some embodiments, Fb2 and Fb3 may require use of some of the same IEs, while Fb3 as an evolution of Fb2 may require support of additional IEs, or deprecate some IEs, or both. In addition, the syntax for Fb3 may be different from the syntax of Fb2, such that one or more IEs in Fb2 with their M-bit set are deprecated in Fb3, so the CS cannot mandate their presence when handling Fb3 in lieu of Fb2. Conversely, because CS2 is configured to handle Fb3 and must fall-back to Fb2, CS2 may need to adhere to the syntax
20 needed to support Fb2 and must undo any deprecations that were used in Fb3.

In some embodiments, an CFR-CFA exchange may carry information pertaining to a single list and one or more service features, since the Feature-List-ID allows carriage of up to m distinct features at once (m may be equal to 32). When a CTF desires features from multiple Feature-Lists to be supported, the CTF may send multiple CFRs. Each CFR
25 may query/negotiate the support with the CDF to completion, and may be clubbing features that belong to the same Feature-List in the same request as an optimization measure. By way of example, the exchange may follow the pattern below:

CTF $\rightarrow$ CFR, with Feature-List='x' and all Feature-List-IDs

CDF $\rightarrow$ CFA, with Feature-List='x' and supported Feature-List-IDs

30 If the CDF supports one or more requested features in a different iteration belonging to a different Feature-List for the vendor of the CTF, that information may not be provided in the response. The CTF may parse the response and inquire via separate individual messages (CFRs) for each of those features. After concluding the negotiations

for Feature-List='x', the CTF may proceed to repeat the process for Feature-List='y' features, as needed for charging.

As described above, Figure 4 includes a flowchart of an apparatus 10, method, and computer program product according to certain example embodiments. It will be understood that each block of the flowchart, and combinations of blocks in the flowchart, may be implemented by various means, such as hardware, firmware, processor, circuitry, and/or other devices associated with execution of software including one or more computer program instructions. For example, one or more of the procedures described above may be embodied by computer program instructions. In this regard, the computer program instructions which embody the procedures described above may be stored by a memory device 14 of an apparatus employing an embodiment of the present invention and executed by processing circuitry 12 of the apparatus. As will be appreciated, any such computer program instructions may be loaded onto a computer or other programmable apparatus (e.g., hardware) to produce a machine, such that the resulting computer or other programmable apparatus implements the functions specified in the flowchart blocks. These computer program instructions may also be stored in a computer-readable memory that may direct a computer or other programmable apparatus to function in a particular manner, such that the instructions stored in the computer-readable memory produce an article of manufacture, the execution of which implements the function specified in the flowchart blocks. The computer program instructions may also be loaded onto a computer or other programmable apparatus to cause a series of operations to be performed on the computer or other programmable apparatus to produce a computer-implemented process such that the instructions which execute on the computer or other programmable apparatus provide operations for implementing the functions specified in the flowchart blocks.

A computer program product is therefore defined in those instances in which the computer program instructions, such as computer-readable program code portions, are stored by at least one non-transitory computer-readable storage medium with the computer program instructions, such as the computer-readable program code portions, being configured, upon execution, to perform the functions described above, such as in conjunction with the flowchart of Figure 4. In other embodiments, the computer program instructions, such as the computer-readable program code portions, need not be stored or otherwise embodied by a non-transitory computer-readable storage medium, but may, instead, be embodied by a transitory medium with the computer program instructions,

such as the computer-readable program code portions, still being configured, upon execution, to perform the functions described above.

Accordingly, blocks of the flowchart support combinations of means for performing the specified functions and combinations of operations for performing the specified functions for performing the specified functions. It will also be understood that one or more blocks of the flowchart, and combinations of blocks in the flowchart, may be implemented by special purpose hardware-based computer systems which perform the specified functions, or combinations of special purpose hardware and computer instructions.

10 In some embodiments, certain ones of the operations above may be modified or further amplified. Furthermore, in some embodiments, additional optional operations may be included, such as represented by the blocks outlined in dashed lines. Modifications, additions, or amplifications to the operations above may be performed in any order and in any combination.

15 Many modifications and other embodiments of the inventions set forth herein will come to mind to one skilled in the art to which these inventions pertain having the benefit of the teachings presented in the foregoing descriptions and the associated drawings. Therefore, it is to be understood that the inventions are not to be limited to the specific embodiments disclosed and that modifications and other embodiments are intended to be included within the scope of the appended claims. Moreover, although the foregoing descriptions and the associated drawings describe example embodiments in the context of certain example combinations of elements and/or functions, it should be appreciated that different combinations of elements and/or functions may be provided by alternative embodiments without departing from the scope of the appended claims. In this regard, for
20 example, different combinations of elements and/or functions than those explicitly described above are also contemplated as may be set forth in some of the appended claims. Although specific terms are employed herein, they are used in a generic and descriptive sense only and not for purposes of limitation.

THAT WHICH IS CLAIMED:

1. A method comprising:
 - causing a first feature support query to be transmitted to a first charging server
 - 5 querying whether the first charging server supports the one or more required features, wherein each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements;
 - receiving a first feature support response from the first charging server indicating whether each of the one or more required features is supported;
 - 10 determining whether the one or more required features are all supported; and
 - in a circumstance where the one or more required features are not all supported, causing a second feature support query to be transmitted to a second charging server.
2. A method according to Claim 1 wherein the method is utilized in
15 conjunction with a charging triggering function that supports the one or more required features, and wherein information regarding one or more different iterations of the one or more supported features is stored in the first charging server.
3. A method according to any of Claim 1 to 2 further comprising:
 - 20 receiving a second feature support response from the second charging server;
 - determining whether the one or more required features are all supported by the second charging server; and
 - in a circumstance where the one or more required features are not all supported, causing a third feature support query to be transmitted to a third charging server.
- 25 4. A method according to any of Claims 1 to 3 wherein the first feature support response further comprises one or more support indications associated with each of the one or more required features, and wherein each of the one or more support indications indicates one or more of: whether the required feature is completely supported,
30 whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server.

5. A method according to Claim 4 further comprising:

for each required feature:

determining whether the required feature is completely supported by the first charging server or whether the required feature is not completely supported and a predecessor feature of the required feature is supported; and

in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported, causing the required feature to fall back to the predecessor feature.

6. An apparatus comprising at least one processor and at least one memory including computer program code for one or more programs, the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus at least to:

cause a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features, wherein each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements;

receive a first feature support response from the first charging server indicating whether each of the one or more required features is supported; and

determine whether the one or more required features are all supported by the second charging server;

in a circumstance where the one or more required features are not all supported, cause a second feature support query to be transmitted to a second charging server.

7. An apparatus according to Claim 6 wherein the apparatus is utilized in conjunction with a charging triggering function that supports the one or more required features, and wherein information regarding one or more different iterations of the one or more supported features is stored in the first charging server.

8. An apparatus according to any of Claims 6 to 7 wherein the at least one memory and the computer program code are further configured to, with the at least one processor, cause the apparatus at least to:

receive a second feature support response from the second charging server;

determine whether the one or more required features are all supported by the second charging server; and

in a circumstance where the one or more required features are not all supported, cause a third feature support query to be transmitted to a third charging server.

5

9. An apparatus according to any of Claims 6 to 8 wherein the first feature support response further comprises one or more support indications associated with each of the one or more required features, and wherein each of the one or more support indications indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server.

10. An apparatus according to Claim 9 wherein the at least one memory and the computer program code are further configured to, with the at least one processor, cause the apparatus at least to:

for each required feature:

determine whether the required feature is completely supported by the first charging server or whether the required feature is not completely supported and a predecessor feature of the required feature is supported; and

in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported, causing the required feature to fall back to the predecessor feature.

11. A computer program product comprises at least one non-transitory computer-readable storage medium having computer executable program code instructions stored therein, the computer executable program code instructions comprising program code instructions configured, upon execution, to:

cause a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features, wherein each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements;

receive a first feature support response from the first charging server indicating whether each of the one or more required features is supported; and

determine whether the one or more required features are all supported;

in a circumstance where the one or more required features are not all supported, cause a second feature support query to be transmitted to a second charging server.

5 12. A computer program product according to Claim 11 wherein the computer program product is utilized in conjunction with a charging triggering function that supports the one or more required features, and wherein information regarding one or more different iterations of the one or more supported features is stored in the first charging server.

10

 13. A computer program product according to any of Claims 11 to 12 wherein the computer executable program code instructions further comprise program code instructions configured, upon execution, to:

 receive a second feature support response from the second charging server;

15 determine whether the one or more required features are all supported by the second charging server; and

 in a circumstance where the one or more required features are not all supported, cause a third feature support query to be transmitted to a third charging server.

20 14. A computer program product according to any of Claims 11 to 13 wherein the first feature support response further comprises one or more support indications associated with each of the one or more required features, and wherein each of the one or more support indications indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a
25 successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server.

 15. A computer program product according to Claim 14 wherein the computer executable program code instructions further comprises program code instructions
30 configured, upon execution, to:

 for each required feature:

 determine whether the required feature is completely supported by the first charging server or whether the required feature is not completely supported and a predecessor feature of the required feature is supported; and

in a circumstance where the required feature not completely supported and the predecessor feature of the required feature is supported, cause the required feature to fall back to the predecessor feature.

5 16. An apparatus comprising:

means for causing a first feature support query to be transmitted to a first charging server querying whether the first charging server supports the one or more required features, wherein each of the one or more required features is associated with a set of mandatory information elements and a set of non-mandatory information elements;

10 means for receiving a first feature support response from the first charging server indicating whether each of the one or more required features is supported;

means for determining whether each of the one or more required features are all supported; and

15 means for in a circumstance where the one or more required features are not all supported, causing a second feature support query to be transmitted a second charging server.

17. An apparatus according to Claim 16 wherein the apparatus is utilized in conjunction with a charging triggering function that supports the one or more required features, and wherein information regarding one or more different iterations of the one or more supported features is stored in the first charging server.

18. An apparatus according to any of Claims 16 to 17 further comprising:

25 means for receiving a second feature support response from the second charging server;

means for determining whether the one or more required features are all supported; by the second charging server and

30 means for in a circumstance where the one or more required features are not all supported, causing a third feature support query to be transmitted to a third charging server.

19. An apparatus according to any of Claims 16 to 18 wherein the first feature support response further comprises one or more support indications associated with each of the one or more required features, and wherein each of the one or more support

indications indicates one or more of: whether the required feature is completely supported, whether the first charging server supports a predecessor feature or a successor feature of the required feature, or no predecessor feature or successor feature of the required feature is supported by the first charging server.

5

20. An apparatus according to Claim 19 further comprising:
for each required feature:

means for determining whether the required feature is completely supported
by the first charging server or whether the required feature is not completely
10 supported and a predecessor feature of the required feature is supported; and

means for in a circumstance where the required feature not completely
supported and the predecessor feature of the required feature is supported, causing
the required feature to fall back to the predecessor feature.

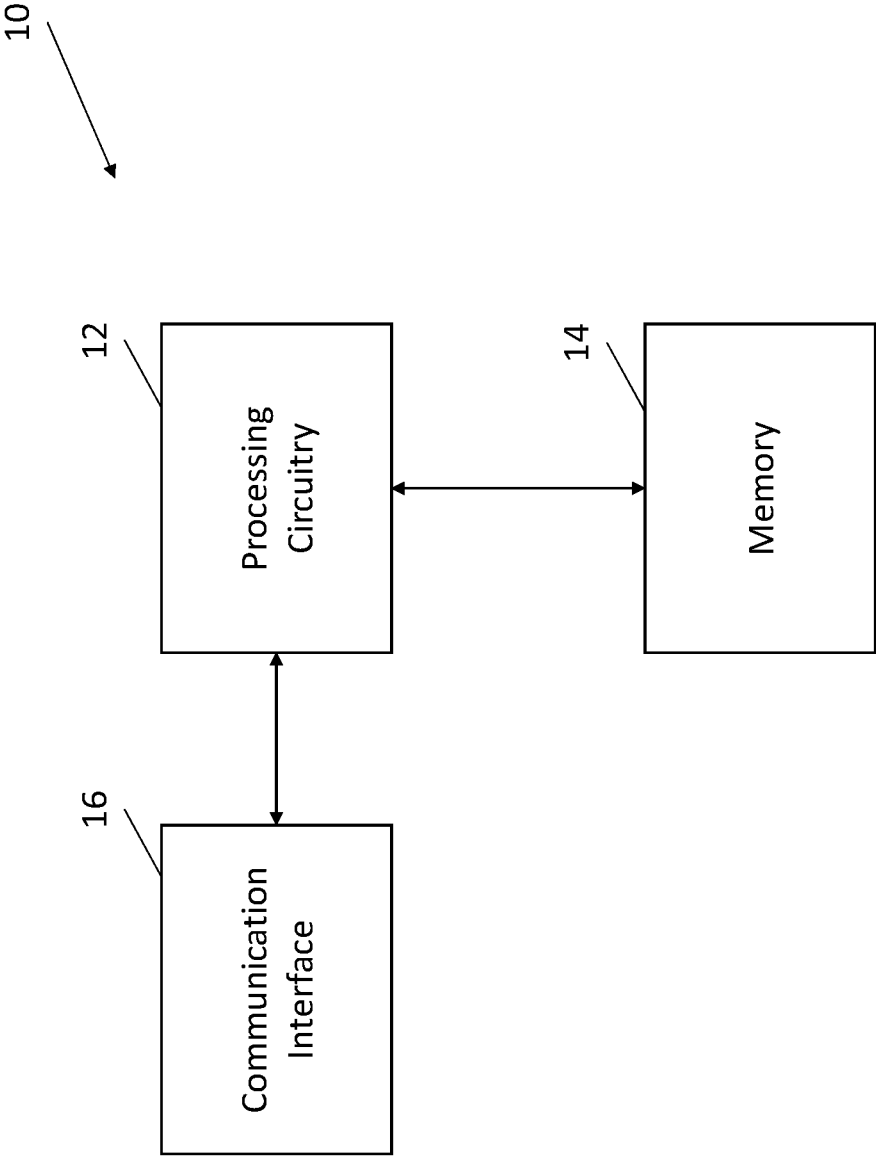


Figure 1

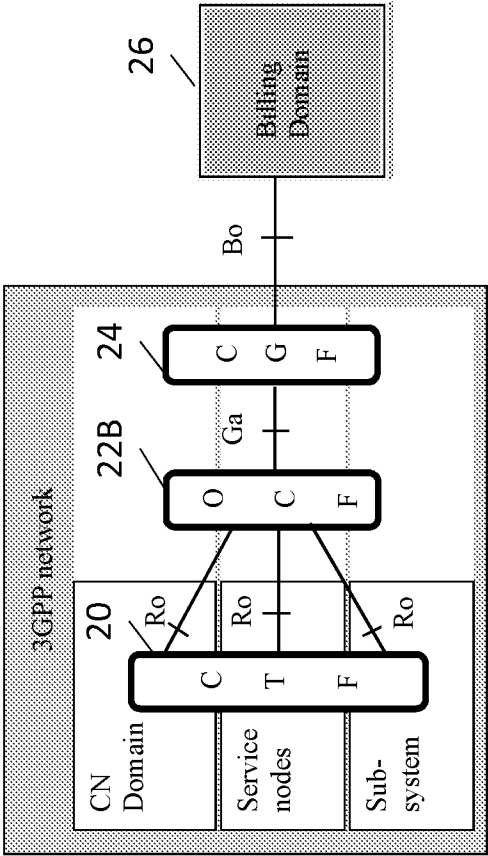


Figure 2B

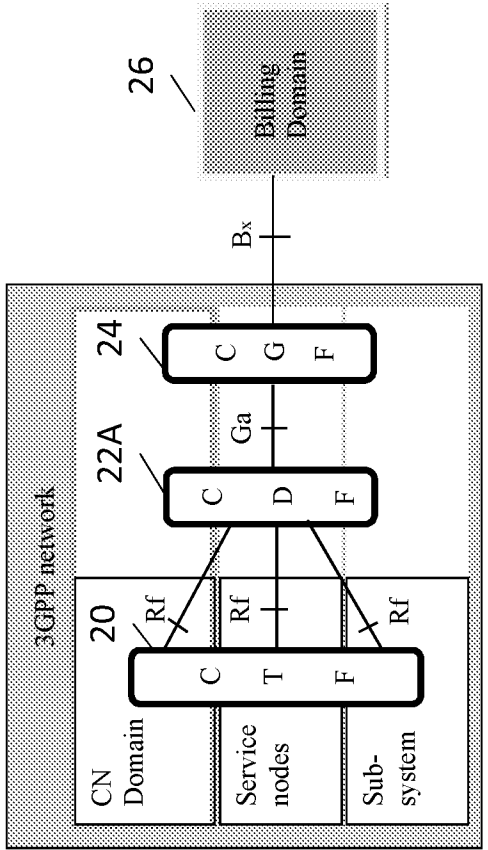


Figure 2A

Service		Fa	Fan	..	..	Fa2	Fa1	Fa0
IE	N1	-				M	-	M
	N2	M				M	M	M
	N3	-				M	M	-
	N4	M				-	-	M
	..							
..						-	M	M
	Nn	-				M	-	-

Figure 3

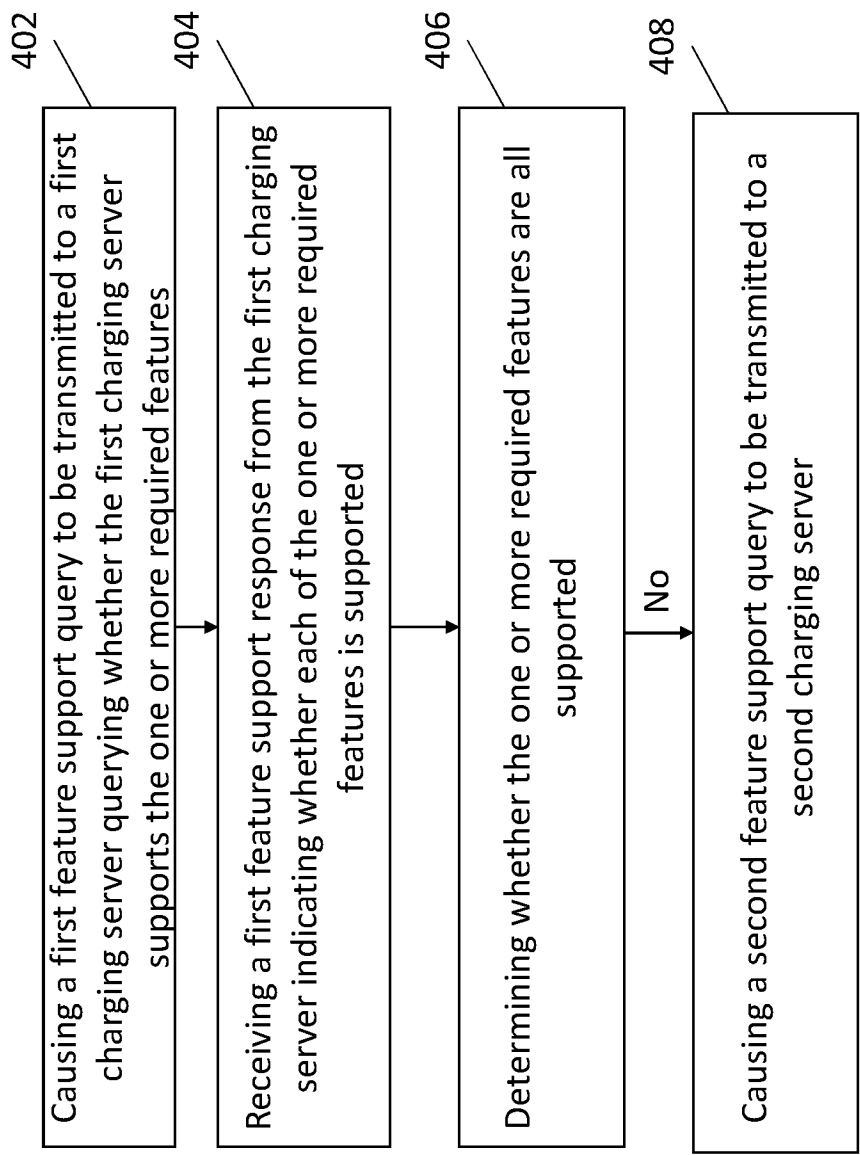


Figure 4

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2018/060561

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04L12/14 H04M15/00 H04W4/24
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L H04M H04W

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	"3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; Charging management; Diameter charging applications (Release 15)", 3GPP DRAFT; 32299-F40, 3RD GENERATION PARTNERSHIP PROJECT (3GPP), MOBILE COMPETENCE CENTRE ; 650, ROUTE DES LUCIOLES ; F-06921 SOPHIA-ANTIPOLIS CEDEX ; FRANCE 21 September 2018 (2018-09-21), XP051477466, Retrieved from the Internet: URL: http://www.3gpp.org/ftp/3guInternal/3GPP%5Fultimate%5Fversions%5Fto%5Fbe%5Ftransposed/sentToDpc [retrieved on 2018-09-21] paragraph [6.5.10] - paragraph [6.5.10.3] -/--	1-3,6-8, 11-13, 16-18
Y		4,5,9,



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

11 July 2019

Date of mailing of the international search report

19/07/2019

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Patlaka, Efi

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2018/060561

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	paragraph [7.1.8] - paragraph [7.1.8A] paragraph [7.2.144] -----	10,14, 15,19,20
Y	NOKIA ET AL: "pCR TR 32.870 New solution and evaluation key issue 6", 3GPP DRAFT; S5-176377 REL-15 PCR 32.870 NEW SOLUTION AND EVALUATION KEY ISSUE 6, 3RD GENERATION PARTNERSHIP PROJECT (3GPP), MOBILE COMPETENCE CENTRE ; 650, ROUTE DES LUCIOLES ; F-06921 SOPHIA-ANTIPOLI ', vol. SA WG5, no. Reno, US; 20171127 - 20171201 1 December 2017 (2017-12-01), XP051366305, Retrieved from the Internet: URL:http://www.3gpp.org/ftp/Meetings%5F3GPP%5FSYNC/SA5/Docs/ [retrieved on 2017-12-01] paragraph [5.6.3.4] -----	4,5,9, 10,14, 15,19,20
A	US 2012/134286 A1 (BHALLA RAJESH [US]) 31 May 2012 (2012-05-31) paragraph [0119] - paragraph [0132] -----	1-20
A	"3rd Generation Partnership Project; Technical Specification Group Core Network; Cx and Dx Interfaces based on the Diameter protocol; Protocol details; (Release 5)", 3GPP STANDARD; 3GPP TS 29.229, 3RD GENERATION PARTNERSHIP PROJECT (3GPP), MOBILE COMPETENCE CENTRE ; 650, ROUTE DES LUCIOLES ; F-06921 SOPHIA-ANTIPOLIS CEDEX ; FRANCE, no. V2.0.0, 1 May 2002 (2002-05-01), pages 1-22, XP050372491, paragraph [07.1] -----	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2018/060561

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2012134286 A1	31-05-2012	CN 102480765 A	30-05-2012
		EP 2458798 A2	30-05-2012
		JP 5421341 B2	19-02-2014
		JP 2012120170 A	21-06-2012
		RU 2011148333 A	10-06-2013
		US 2012134286 A1	31-05-2012
