A method for generating Lightweight Machine to Machine commands

Technical field

A method for use in a utility network for generating Lightweight Machine to Machine, LwM2M, commands.

Background

The Open Mobile Alliance (OMA) Lightweight Machine to Machine (LwM2M) protocol is a protocol for the management and service enablement of Internet of Things (IoT) devices, such as utility metering devices. Utility metering devices may comprise electricity meters, water meters, gas meters etc.

The device management capabilities provided by the LwM2M protocol include the remote provisioning of security credentials to enable a device to access a LwM2M server (bootstrapping), updating the firmware of the device, fault management, remote device diagnostics and troubleshooting, and connectivity management.

LwM2M may be particularly suitable for battery-powered utility metering devices, as LwM2M protocols can minimise energy consumption by optimising communications and supporting low-power sleep modes. As such, increasingly, utility networks (or resource distribution networks) are incorporating LwM2M utility metering devices.

Commands sent to LwM2M devices need to be structured according to the LwM2M protocol, which defines an object and resource model. Typically, a LwM2M command may specify a target resource, the operation(s) (i.e. read, write, execute, etc.) to be performed on the target resource, additional parameters required for the operation to be performed (e.g. the new value to be written in the case of a "write" operation), and/or any other relevant information. In order to perform a function, multiple operations may need to be performed on one or more resources within a LwM2M system, and these multiple operations may be defined in one or more LwM2M command(s).

Existing systems use less constrained devices to execute common requests/commands (Get Time, Read Meter Data, Read Event History) to specific standards (ANSI, DLMS).

These devices have enough resources to interpret the common requests/commands into the proper operations. The problem for LwM2M devices is that these common requests/commands do not easily map to the LwM2M object and resource model. Additionally, the LwM2M constrained device may not have the computational power to interpret and perform the common request/command and the system cannot simply construct a single atomic request to accomplish the same task in an LwM2M compatible way. To be able to integrate LwM2M devices into a system with less constrained devices, a solution is needed to allow common requests to perform on all device types (ANSI, LwM2M, DLMS, etc.).

Summary

According to the invention in a first aspect, there is provided a method for use in a utility network for generating Lightweight Machine to Machine, LwM2M, commands for implementation by one or more utility metering devices, the method comprising: generating, by a headend system, a command for implementation by a utility meter, wherein the utility meter comprises a LwM2M utility meter that implements a LwM2M protocol, and wherein the command generated by the headend system is in a format that is not suitable for execution by the LwM2M utility meter; transmitting, by the headend system, the command to an adaptor; generating, by the adaptor and based on the received command, a plurality of LwM2M commands, each LwM2M command defining one or more LwM2M operations for implementation by the LwM2M utility meter to satisfy the command generated by the headend system; executing, by the LwM2M utility meter, the plurality of LwM2M commands; and sending, by the adaptor and to the headend system, a response to the command generated by the headend system, based on the execution of the plurality of LwM2M commands by the LwM2M utility meter.

Optionally, the LwM2M operations comprise one or more of: a read operation, a write operation, an execute operation and a delete operation.

Optionally, the plurality of LwM2M commands are generated based on a single command received from the headend system. The single command may be an atomic command.

Optionally, the generation of the plurality of LwM2M commands comprises mapping, by a mapping module, the command generated by the headend system to the one or more LwM2M operations.

Optionally, the method further comprises queueing, by the adaptor, one or more of the plurality of LwM2M commands for execution by the LwM2M utility meter; and transmitting, by the adaptor and to the LwM2M utility meter, the one or more queued LwM2M commands for execution by the LwM2M utility meter.

Optionally, the method further comprises: receiving, by the adaptor and from the LwM2M utility meter, a wake-up signal, and wherein, the adaptor transmits the one or more queued LwM2M commands to the LwM2M utility meter for execution in dependence on receiving the wake-up signal.

Optionally, the method further comprises: determining, by the adaptor, a sequence of execution of the plurality of generated LwM2M commands, and wherein the first LwM2M command in the sequence of execution is queued by the adaptor for transmission to the LwM2M utility meter.

Optionally, the method further comprises: queuing, by the adaptor, a subsequent LwM2M command, as defined by the sequence of execution, in dependence on transmission of a previous LwM2M command in the sequence of execution to the LwM2M utility meter for execution.

Optionally, wherein the subsequent LwM2M command is queued in dependence on successful completion of the previous LwM2M command in the sequence of execution, by the LwM2M device.

Optionally, the method further comprises transmitting, by the adaptor and to the headend system, a response indicating successful completion of the command in dependence on successful completion of each of the plurality of LwM2M commands.

Optionally, the method further comprises: assigning, by the adaptor, a common correlation ID to each of the plurality of LwM2M commands generated based on the command received from the headend system.

Optionally, the method further comprises: transmitting, by the adaptor and to the headend system, a success response based on the successful execution of the plurality of LwM2M commands sharing the common correlation ID, by the LwM2M utility meter.

Optionally, the command generated by the headend system is a first command, and the plurality of LwM2M commands is a first plurality of LwM2M commands, and the method further comprises: generating, by the headend system, a second command for implementation by the LwM2M utility meter; generating, by the adaptor and based on the second command, a second plurality of LwM2M commands; assigning, by the adaptor and to each of the first and second plurality of LwM2M commands, a priority identifier; and queuing the first plurality of LwM2M commands and the second plurality of LwM2M commands, for execution by the LwM2M utility meter, in an order based on the priority identifiers of the first and second plurality of LwM2M commands.

Optionally, the method further comprises: re-ordering the queued first and second plurality of LwM2M commands, based on further information received by the adaptor and from the headend system.

Optionally, the further information comprises a third command generated by the headend system and received by the adaptor, and wherein the third plurality of LwM2M commands generated based on the third command comprise a priority identifier indicating that implementation of the third command is of higher priority than the first and/or second command.

Optionally, the method further comprises: receiving, by the adaptor and for each of the plurality of LwM2M commands transmitted to the LwM2M utility meter, a response message from the LwM2M utility meter indicating whether completion of a respective LwM2M command is successful or unsuccessful; and undertaking, by an error handler of the adaptor, an error management action, if the response message indicates unsuccessful completion of the respective LwM2M command.

Optionally, the error management action comprises determining, by the error handler and based on a type of error indicated within the response message, an alternative method for re-transmission of the LwM2M command.

Optionally, if the type of error indicates that the LwM2M command exceeds a packet size limit, the method further comprises: instructing, by the error handler, generation of a subset of LwM2M commands corresponding to the unsuccessfully completed LwM2M command, each LwM2M command of the subset of LwM2M commands comprising a packet size below the packet size limit; and transmitting, by the adaptor, the subset of LwM2M commands to the LwM2M utility meter for execution by the LwM2M utility meter. Optionally, the unsuccessful LwM2M command defines an operation to be performed on a plurality of resources of the LwM2M device, and each LwM2M command of the subset of LwM2M commands defines an operation to be performed on a subset of the plurality of resources.

Also disclosed herein is a system comprising: a LwM2M utility meter configured to implement a LwM2M protocol; a headend system configured to generate a command for implementation by the LwM2M utility meter, wherein the command generated by the headend system is in a format that is not suitable for execution by the LwM2M utility meter; and an adaptor configured to receive the command generated by the headend system and generate, based on the received command, a plurality of LwM2M commands, each LwM2M command defining one or more LwM2M operations for implementation by the LwM2M utility meter to satisfy the command generated by the headend system; and wherein the LwM2M utility meter is further configured to execute the plurality of LwM2M commands; and wherein the adaptor is further configured to send, to the headend system, a response to the command generated by the headend system, based on the execution of the plurality of LwM2M commands by the LwM2M utility meter.

Brief description of drawings

Figure 1 shows a schematic of an exemplary utility network;
Figure 2 shows a schematic of an exemplary adaptor;
Figure 3 shows an exemplary sequence diagram; and
Figure 4 shows an exemplary sequence diagram.

Detailed description

Figure 1 shows an exemplary utility network 100 according to the invention. The utility network 100 comprises a first network 102, a first device network 103 and a Lightweight Machine to Machine, LwM2M, network 104. The first network 102 may comprise a headend system 110. The headend system 110 may be in communication with the entities of the first device network 103. As shown in Figure 1, the first device network 103 may comprise a plurality of root nodes 106 (only one of which is shown in Figure 1) configured to communicate with a plurality of nodes 108a-n. The skilled person will appreciate that utility networks 100 may comprise hundreds of roots nodes each configured to communicate with thousands of nodes, and that Figure 1 shows a schematic representation of such a system (showing a single root node 106 and three nodes 108a-n) only. The root node 106 and/or the nodes 108a-n may comprise utility metering devices (or utility meter devices), and in some arrangements, ANSI or DLMS utility metering devices. The root node 106 may communicate with the nodes 108a-n to execute commands (e.g. collecting utility data from the nodes 108a-n) and forward the data obtained from execution of the command to a headend system 110 for processing by the headend system 110 (e.g. resource management, utility bill management etc.). The headend system 110 may generate the commands for execution by the root node 106 and/or the nodes 108a-n.

The utility network 100 comprises LwM2M devices, represented as single LwM2M device 114 in Figure 1, though the skilled person will understand that a large number of LwM2M devices may form part of the utility network 100. The LwM2M device 114 may comprise a LwM2M utility metering device.

A LwM2M client 116 is located on a the LwM2M device 114. The LwM2M client 116 may be integrated as a software library or a built-in function of the device 104.

The LwM2M client 116 comprises a plurality of objects 120a-c, wherein each object 120a-c represents a different concept of data. For example, a first object 120a may be for managing connections, a second object 120b may be for providing a GPS location of the LwM2M device 114, and a third object 120c may be for managing network connections. The objects 120a-c may be single-instance or multiple-instance. Each object 120a-c may comprise an object identifier (and information indicating the number of instances (i.e. single or multiple)). Each object instance of an object 120a-c comprises at least one resource. Each resource comprises an item of data. For example, a location

object may comprise six resources: a latitude, a longitude, an altitude, a velocity, an uncertainty and a timestamp. Each resource may comprise its own resource identifier. The skilled person will be familiar with the LwM2M object/resource model and no further information is given here.

The root node 106 and/or nodes 108a-n on the first device network 103 may be standardised devices (ANSI, DLMS) and the headend system 110 may communicate with them using various communication technologies, for example, to transmit common requests/commands (such as Get Time, Read Meter Data, Read Event History), as described above. The root node 106 and/or nodes 108a-n may each comprise a communications application, which may be embedded into the root node 106 and nodes 108a-n, configured to interpret the commands generated by the headend system 110 for execution by the root node 106 and/or nodes 108a-n.

In contrast, LwM2M devices, such as the LwM2M device 114, are constrained devices that do not have the computational power required to receive and interpret the commands generated by the headend system 110 to allow execution of the request/command by the LwM2M device 114.

In order for the LwM2M device 114 to execute a command, the command needs to be structured in a format that can be understood and executed by the LwM2M device 114. In particular, in order to execute a command (such as "Get Event Log" or "Get Meter Read Data"), the LwM2M device 114 will likely need to perform a plurality of operations, that is, a plurality of read, write, execute, delete etc. operations on various objects 120a-c and resources of the LwM2M client 116. Therefore, the LwM2M device 114 is unable to respond to commands as an atomic operation (i.e. as a single, indivisible unit of work), as is typically possible for the root node 106 and nodes 108a-n in the first network 102 (which comprise the communications application for interpreting commands generated by the headend system 110).

As mentioned above, the headend system 110 generates commands using protocols that are not compliant with the OMA LwM2M protocol. In particular, the headend system 110 may generate single (atomic) commands, which cannot be implemented by LwM2M devices due to the LwM2M protocol/specifications (that is, the single atomic command

cannot be divided into the multiple individual operations needed for implementation by a LwM2M device).

The utility system 100 of the invention therefore comprises an adaptor 126 configured to translate the command generated by the headend system 110 (which is in a format not compliant with the OMA LwM2M protocol, and may be a single (atomic) command) into a plurality of LwM2M commands suitable for implementation by the LwM2M device 114 (i.e. into a format that is compliant with the OMA LwM2M protocol). Each LwM2M command defines an operation that needs to be undertaken in order to execute the command generated by the headend system 110. Operations may include read, write, execute or delete operations.

Each LwM2M command generated by the adaptor 126 may comprise a CoAP (Constrained Application Protocol) message. The CoAP messages may comprise one or more of: a read message (GET), a write message (PUT), an execute message (POST) and a delete message (DELETE). The skilled person will understand that a read message instructs the LwM2M device 114 to perform a read operation (i.e. read/get a resource value), a write message instructs the LwM2M device 114 to perform a write operation (i.e. update a resource value), an execute message instructs the LwM2M device 114 to perform an execute operation (i.e. execute a resource function), and a delete message instructs the LwM2M device 114 to perform a delete operation (i.e. delete a resource on a device).

The adaptor 126 may, along with the headend system 110, may form part of the first network 102. In exemplary arrangements, the adaptor 126 does not form part of (i.e. is not hosted by) the LwM2M device 114. The skilled person will appreciate however, that in alternative arrangements, the adaptor 126 may be hosted on the LwM2M device 114, or may form part of another network (i.e. a different network to the first network 102).

Figure 2 shows a schematic view of an exemplary adaptor 126.

The adaptor 126 comprises a communications module 132 configured for communication (which may be wireless or wired) with the headend system 110 and the LwM2M device 114. The communications module 132 may comprise a receiver and/or a transmitter. The communications module 132 may be configured to receive

communications from the headend system 110 according to a first (non-LwM2M) protocol, and transmit LwM2M communications to the LwM2M device 114 according to a LwM2M protocol. The communications module 132 may be in data communication with other entities, such as other utility metering devices (that are not located on the first network 102 or the LwM2M network 104), other servers etc. and is configured to transmit and receive data accordingly.

The adaptor 126 may further comprise a memory 134 and a processor 136. The memory 134 may comprise a non-volatile and/or a volatile memory. The memory 134 may have a computer program 138 stored therein. The computer program 138 may comprise instructions for performing the methods disclosed herein. The computer program 138 may be loaded in the memory 134 from a non-transitory computer readable medium 140, on which the computer program is stored.

The adaptor 126 may further comprise a controller 144, a LwM2M command generator 146, a staging module 148, a validation module 150, a command response generator 152 and an error handler 154. The processor 136 is configured by the computer program 138 to perform one or more of the functions of the controller 144, the LwM2M command generator 146, the staging module 148, the validation module 150, the command response generator 152 and the error handler 154.

A method of facilitating communications between entities (and specifically a headend system 110) on a first network 102 and LwM2M devices (such as LwM2M device 114) on a LwM2M network 104 is described below with reference to Figure 3.

302: The headend system 110 generates a command intended for execution by one or more LwM2M devices (LwM2M device 114 for the purpose of this example) of the LwM2M network 104. In some examples, the command may be intended for execution by a single LwM2M device, while in other examples, the command may be intended for execution by a defined plurality of LwM2M devices.

As mentioned above, the LwM2M devices comprise LwM2M utility meters, and the command generated by the headend system 110 may, for example, comprise a request for data from the LwM2M utility meter(s) (e.g. “Get Event Log”, or “Get Meter Read Data”, i.e. get processing/utility consumption data). In other

examples, the command generated by the headend system 110 may comprise a reconfiguration command, for altering one or more parameters or functions of the LwM2M utility meter(s). The skilled person will appreciate that these commands are given as examples only.

For the purpose of this example, the headend system 110 generates a "Get Event History" command, requesting information relating to past events recorded by the LwM2M utility meter (e.g. power outages, fluctuations, etc.).

The command generated by the headend system 110 is compliant with a first protocol, where the first protocol is not the OMA LwM2M protocol. As such, the command generated by the headend system 110 is not in a format suitable for execution by the LwM2M device 114. As described above, the command generated by the headend system 110 may be a single (atomic) command.

The command generated by the headend system 110 may comprise a command identifier.

The headend system 110 transmits, or otherwise inputs, the generated command to the adaptor 126.

304: The adaptor 126 receives the command generated by the headend system 110, and the controller 144 controls the LwM2M command generator 146 to generate a plurality of LwM2M commands based on the command generated by the headend system 110. The plurality of LwM2M commands are for implementation by the LwM2M device 114 to satisfy the command generated by the headend system 110 (in the first, non-OMA LwM2M compliant format/structure). In exemplary arrangements, the adaptor 126 may generate a plurality of LwM2M commands from a single command generated by the headend system.

The plurality of LwM2M commands may be considered to be correlated LwM2M commands. That is, the plurality of LwM2M commands may each define a step necessary to be executed (though not necessarily at the same time) to perform the specific function defined by the headend system command (e.g. data retrieval, reconfiguration, a firmware update etc.). The plurality of LwM2M

commands generated to satisfy the headend system command may be assigned a common correlation ID. The common correlation ID, may in some arrangements, match (or be based on) the command ID of the headend system command. Therefore the plurality of LwM2M commands also correlate to the command generated by the headend system 110.

The plurality of LwM2M commands generated by the LwM2M command generator 146 comprise one or more of: a read command, a write command, an execute command, and a delete command. Each LwM2M command may comprise an indication of the target LwM2M device to execute the command (in this case, LwM2M device 114), as well as an indication of the operation to be undertaken by the LwM2M device (read, write, execute, delete). Each LwM2M command may further comprise an object identifier and a resource identifier, identifying the object and/or resource that the command should be performed on. As explained above, the LwM2M commands may comprise a CoAP message.

The adaptor 126 may comprise or use a mapping module to generate the plurality of LwM2M commands. The mapping module may be implemented as a library (such as a .DLL, .JAR, .SO, .A etc.) or a plug in.

The mapping module may comprise mapping tables or mapping code, defining a set of rules for translating a command generated by the headend system in a first format (according to a first, non-LwM2M, protocol, e.g. a single atomic command) to corresponding LwM2M commands (comprising read/write/execute/delete operations). For example, the mapping module may define that a "Get Event History" command generated by the headend system 110 requires six LwM2M commands: three execute LwM2M commands and three read LwM2M commands. The mapping module may use the headend system 110 command title/type and utility device type (or model number) to determine the corresponding LwM2M commands.

Using the mapping module, the LwM2M command generator 146 generates the plurality of LwM2M commands for execution by the LwM2M device 114.

Generation of the LwM2M commands by the LwM2M command generator 146, may further comprise defining an order (or sequence) in which the LwM2M commands should be implemented. The mapping module may define the order of the commands, according to the mapping tables/mapping code. Each LwM2M command may be assigned a sequence identifier indicating the position of each LwM2M command in the implementation sequence.

In this example, the LwM2M command generator 146 generates six LwM2M commands. Each LwM2M command may specify that type of operation to be performed (read/write/execute/delete) and the object and resource upon which the operation should be performed (i.e. an object ID and a resource ID). In this example the LwM2M commands generated are:

- A first LwM2M command comprising an execute command "General Log Start" (i.e. execute the start time of the General Log),
- A second LwM2M command comprising an execute command "Security Log Start" (i.e. execute the start time of the Security Log),
- A third LwM2M command comprising an execute command "Parameter Log Start", (i.e. execute the start time of the Parameter Log),
- A fourth LwM2M command comprising a read command "General Log" (i.e. read the data in the General Log),
- A fifth LwM2M command comprising a read command "Security Log" (i.e. read the data in the Security Log), and
- A sixth LwM2M command comprising a read command "Parameter Log" (i.e. read the data in the Parameter Log).

The skilled person will appreciate that in alternative examples, an alternative number of LwM2M commands may be generated for application to alternative objects/resources of the LwM2M device 114. The above number of commands, and objects/resources are given as examples only. Furthermore, the skilled person will appreciate that one or more of the generated LwM2M commands may define a plurality of objects and/or resources upon which the command is to be performed (for the purpose of this example however, it is assumed that each LwM2M command specifies a single resource upon which the command is to be performed)

306: The staging module 148 stages the plurality of LwM2M commands generated by the LwM2M command generator 146. In exemplary arrangements, the controller 144 may control the staging module 148 to stage the plurality of LwM2M commands, following the generation of the plurality of LwM2M commands by the command generator 146. For the purpose of this description, "staging" encompasses the process of preparing the LwM2M commands to be executed on the LwM2M device 114. This may include queueing one or more of the LwM2M commands, sending the LwM2M commands to the LwM2M device 114 immediately for execution by the LwM2M device 114, defining one or more conditions for implementation of the LwM2M command(s), and/or defining when the LwM2M commands should be executed (e.g. when the LwM2M device wakes up, during a certain time period etc.).

For the purpose of this description "queuing" encompasses preparing one or more LwM2M commands for transmission to the LwM2M device 114 for execution (e.g. when a condition is satisfied). In this example, the staging module 148 may queue each of the plurality of LwM2M commands generated by the LwM2M command generator 146 for implementation by the LwM2M device 114 once the LwM2M device 114 wakes up (e.g. once a wake-up signal is received from the device). The skilled person will appreciate however, that alternative implementations are possible. For example, alternative conditions for transmission of a queued LwM2M command may be defined, such as specifying a particular time, a particular network load condition, or alternatively specifying that the LwM2M commands should be transmitted immediately. The skilled person will be able to envisage further possibilities.

Where each LwM2M command has been assigned a sequence identifier, indicating the order in which the LwM2M commands should be transmitted to the LwM2M device 114, the staging module 148 is configured to queue the LwM2M commands in the order indicated by the sequence identifiers.

The queued commands may be stored in the memory 134 of the adaptor 126.

Advantageously, by queueing the LwM2M commands and storing the queued commands in the memory 134 of the adaptor 126, the LwM2M device 114 is not required to store any of the LwM2M commands in its own memory. This provides resiliency in the event that the LwM2M device 114 reboots.

308: The adaptor 126 is triggered to initiate communication of the one or more queued LwM2M commands to the LwM2M device 114 for execution by the LwM2M device 114.

In this particular example, the LwM2M device 114 is configured to transmit a "wake-up" signal to the adaptor 126, when the LwM2M device 114 is powered on. In exemplary arrangements, the LwM2M device 114 may be battery powered, and only operational for parts of the day in order to conserve battery life. In this example, the trigger to the adaptor 126 to begin to initiate communication of the one or more queued LwM2M commands to the LwM2M device 114 is receipt of the wake-up signal from the LwM2M device 114.

The skilled person will appreciate that in alternative arrangements, the trigger to the adaptor 126 to initiate communication of one or more of the generated LwM2M commands to the LwM2M device 114 may be different (and may comprise one or more of the examples listed in respect of step 306). For example, the adaptor 123 may transmit the one or more LwM2M commands immediately (such an arrangement may be more appropriate for non-battery powered LwM2M utility meters, which are always powered on).

310: In response to the trigger, the controller 144 of the adaptor 126 determines whether there are any LwM2M commands queued in respect of the LwM2M device 114 (i.e. the LwM2M device in respect of which the trigger/wake-up signal is received from).

If there are no LwM2M commands queued for implementation by the LwM2M device 114, no further action is taken.

312: In this example, the controller 144 of the adaptor 126 determines that there are six LwM2M commands queued for execution by the LwM2M device 114, and

therefore, the adaptor 126 begins communicating the queued LwM2M commands to the LwM2M device 114. Specifically, the controller 144 may control the communications module 132 to transmit the queued LwM2M commands to the LwM2M device 114.

The adaptor 126 transmits the first LwM2M command to the LwM2M device 114 for implementation by the LwM2M device 114. In this example, the first LwM2M command comprises an execute command that specifies "General Log Start" (i.e. execute the start time of the General Log).

The adaptor 126 may receive, from the LwM2M device 114, an acknowledgement that the first LwM2M command has been received by the LwM2M device 114 and/or a response message from the LwM2M device 114 indicating the outcome of the first LwM2M command execution (and any requested data).

314: One the adaptor 126 has received the acknowledgement and/or response message from the LwM2M device 114, the adaptor 126 transmits the second LwM2M command to the LwM2M device 114 for implementation by the LwM2M device 114. In this example, the second LwM2M command comprises an execute command "Security Log Start", i.e. a command to execute a particular object and resource.

In exemplary arrangements in which the adaptor 126 is configured to transmit the subsequent LwM2M command in response to receipt of a response message from the LwM2M device 114, the adaptor 126 may transmit the subsequent LwM2M command to the LwM2M device 114 irrespective of the outcome indicated within the response message (i.e. irrespective of whether the response message indicates a successful or unsuccessful outcome of the previously transmitted LwM2M command). The skilled person will understand that in alternative arrangements, the adaptor 126 need not receive an acknowledgement/response message from the LwM2M device 114 in order to transmit the next LwM2M command. In alternative arrangements, the adaptor 126 may transmit the next LwM2M command after a predefined amount of time has expired, or in response to a different trigger (such as a further wake-up signal).

The adaptor 126 may receive, from the LwM2M device 114, an acknowledgement that the second LwM2M command has been received by the LwM2M device 114 and/or a response message from the LwM2M device 114 indicating the outcome of the second LwM2M command execution (and any requested data).

316-320: A similar method to that undertaken for step 314 (in respect of the second LwM2M command) is undertaken for steps 316 to 320, which relate to the third, fourth and fifth LwM2M commands being transmitted by the adaptor 126, and to the LwM2M device 114, for implementation by the LwM2M device 114.

322: One the adaptor 126 has received the acknowledgement and/or response message from the LwM2M device 114 (in respect of the penultimate LwM2M command – in this case the fifth LwM2M command), the adaptor 126 transmits the sixth, and in this case, final, LwM2M command to the LwM2M device 114 for implementation by the LwM2M device 114. In this example, the final LwM2M command comprises a read LwM2M command relating to the "Parameter Log".

Again, the adaptor 126 may receive, from the LwM2M device 114, an acknowledgement that the final LwM2M command has been received by the LwM2M device 114 and/or a response message from the LwM2M device 114 indicating the outcome of the final LwM2M command execution (and any requested data – in this case the Parameter Log data).

324: Once the final LwM2M command has been transmitted to the LwM2M device 114 for execution, the validation module 150 of the adaptor 126 may determine whether each of the plurality of LwM2M commands have been successfully completed by the LwM2M device 114. Specifically, the validation module 150 may determine whether each of the plurality of LwM2M commands which share a common correlation ID have been successfully completed by the LwM2M device 114.

The validation module 150 may determine whether the LwM2M commands have completed successfully based on the response messages received by the adaptor 126 and from the LwM2M device 114. Each response message may

comprise one or more of: an indication of whether the corresponding LwM2M command was executed successfully (and may be in the form of a status code), any errors encountered during execution of the corresponding LwM2M command (such as error type), and any relevant data/status updates that have resulted from the execution of the corresponding LwM2M command.

326: If the validation module 150 determines that all of the plurality of LwM2M commands sharing the common correlation ID have been successfully completed, the controller 144 controls the command response generator 152 to generate a command response for transmission to the headend system 110. A single command response may be generated by the command response generator 152 for transmission to the headend system 110, regardless of how many correlated LwM2M commands are executed by the LwM2M device 114, and therefore regardless of how many LwM2M commands and response messages are exchanged between the adaptor 126 and the LwM2M device 114. The single response may comprise the correlation ID, or the original headend system command ID (which may be the same as the correlation ID in some examples), to allow the headend system 110 to identify the command that the response relates to.

The command response may comprise an indication of the successful completion of the command by the LwM2M device 114, and any relevant object and/or resource updates that have resulted from the execution of the command so that the headend system 110 may update any relevant records (e.g. the object and resource database). In this particular example, the command response may comprise the data contained in the read responses (i.e. the responses to the Read General Log, Read Security Log, and Read Parameter Log LwM2M commands).

The controller 144 controls the communications module 132 of the adaptor 126 to transmit the generated command response to the headend system 110.

In the event that the validation module 150 determines that one or more of the LwM2M commands (sharing the common correlation ID) have not successfully completed, the controller 144 controls the error handler 154 to undertake an error management action.

The error management action may comprise one or more of: ordered dequeueing of one or more of the LwM2M commands, implementation of a timeout condition (in the event that a response message is not received by the adaptor 126 and from the LwM2M device 114), and individual retries of the one or more failed LwM2M commands. In exemplary arrangements, the error handler 154 may be configured to determine an alternative retry method, based on the error type indicated in the response message. That is, the error handler 154 may be configured to instruct the controller 144 to implement the retry (e.g. re-transmission of the failed LwM2M command) in a different way, based on the error response message received. For example, the response message received from the LwM2M device 114 may indicate that a LwM2M command failed to be executed due to the LwM2M command exceeding a packet size limit. If the failed LwM2M command specifies multiple objects and/or resources on which the command is to be performed, in response to such an error message, the error handler 154 may instruct the controller 144 to control the LwM2M command generator 146 to generate a subset of LwM2M commands based on the single, failed LwM2M command. Each of the LwM2M commands of the subset may specify fewer objects and/or resources upon which the command is to be performed, when compared to the single, failed LwM2M command, to comply with the packet size limit. The controller 144 may then control the communications module 132 to transmit the subset of LwM2M commands to the LwM2M device 114 (as described above in respect of the plurality of LwM2M commands) for implementation by the LwM2M device 114. In exemplary arrangements, this process may be recursive. That is, the error handler 154 may not determine whether each of the subset of LwM2M commands complies to the packet size limit ahead of generation/transmission, and instead may instruct generation of a first subset of LwM2M commands for the adaptor 126 to transmit to the LwM2M device 114 for implementation, and if the error persists (i.e. a second response message is received indicating that the one or more of the LwM2M commands of the first subset of LwM2M commands exceeds the packet size limit), the error handler 154 may instruct generation of a second subset of LwM2M commands for transmission to the LwM2M device 114. The second subset of LwM2M commands may comprise a larger number of LwM2M commands than the first subset of LwM2M commands. This process may repeat until each LwM2M command complies with the packet size limit (or until it is not possible to split the LwM2M commands any further, in which case an error response may be transmitted). The skilled person will appreciate that alternative retry methods may be implemented in dependence on the error type.

The skilled person will appreciate that the error management action taken will depend on the type and severity of the error encountered (an indication of which may be given in the response messages received from the LwM2M device 114). The controller 144 may alternatively, or further, control the communications module 132 of the adaptor 126 to transmit an error response to the headend system 110.

The above-mentioned method may advantageously allow LwM2M devices, such as LwM2M utility meters, to be implemented in utility networks where the command/headend system is not configured for LwM2M communications. By implementing an "adaptor" as disclosed above, which translates the communications issued by a headend system (according to a non-LwM2M protocol), long-established and complex headend systems that control thousands and thousands of non-LwM2M utility meter devices, do not need to be reconfigured in order to communicate with LwM2M devices.

Figure 4 shows an alternative method of facilitating communications between entities (and specifically a headend system 110) on a first network 102 and LwM2M devices (such as LwM2M device 114) on a LwM2M network 104.

Steps 402 and 404 correspond substantially to steps 302 and 304 described above in respect of Figure 3, and so a description of these steps is not repeated here. For the purposes of this example however, it is assumed that three LwM2M commands (instead of six LwM2M commands) are generated for implementation by the LwM2M device 114.

406: Following generation of the plurality of LwM2M commands, by the LwM2M command generator 146, the staging module 148 of the adaptor 126 determines a sequence of execution for the LwM2M commands. In this example, the staging module 148 determines the following order of execution for the LwM2M commands: execution of the first LwM2M command, execution of the second LwM2M command, and execution of the third (and final) LwM2M command. As mentioned above, the plurality of LwM2M commands may be assigned a sequence identifier indicating the position of each LwM2M command in the sequence of execution, and the staging module 148 may order the LwM2M commands based on the sequence identifiers.

The staging module 148 queues the LwM2M command that is determined to be the first in the sequence of execution: in this case, the first LwM2M command. The remaining LwM2M commands in the sequence of execution are not queued.

408: The adaptor 126 is triggered to initiate communication of the queued first LwM2M command to the LwM2M device 114.

This step may substantially correspond to step 308 described above in respect of the method of Figure 3. That is, in this particular example, the adaptor 126 may receive a wake-up signal from the LwM2M device 114, and this may trigger transmission of the queued LwM2M commands to the LwM2M device 114 (however the skilled person will appreciate that alternative triggers may be used in alternative methods).

410: Similarly to step 310, described above in respect of the method outlined in Figure 3, in response to the trigger, the controller 144 of the adaptor 126 determines whether there are any LwM2M commands queued in respect of the LwM2M device 114 (i.e. the LwM2M device in respect of which the trigger/wake-up signal is received from).

412: In this example, the controller 144 determines that the first LwM2M command is queued for transmission to the LwM2M device 114, and therefore, the controller 144 controls the communications module 132 to transmit the first LwM2M command to the LwM2M device 114 for execution of the first LwM2M command.

The adaptor 126 may receive, from the LwM2M device 114, a response message from the LwM2M device 114 indicating the outcome of the first LwM2M command execution (and any requested data).

414: On receipt of the response message from the LwM2M device 114, the controller 144 may control the staging module 148 to queue the next LwM2M command in the sequence of execution, which in this case is the second LwM2M command. The staging module 148 may determine the next LwM2M command to queue based on the sequence ID of the remaining generated LwM2M commands.

In some arrangements, the controller 144 may control the validation module 150 to determine whether the first LwM2M command has been successfully completed by the LwM2M device 114 based on the response message. The staging module 148 may queue the next LwM2M command in the sequence of execution in dependence on the successful completion of the previous LwM2M command. That is, in some arrangements, the controller 144 may control the staging module 148 to queue the second LwM2M command, only if the first LwM2M command has been successfully completed by the LwM2M device 114.

416: The controller 144 controls the communications module 132 to transmit the queued second LwM2M command to the LwM2M device 114 for execution of the second LwM2M command.

In some arrangements, the adaptor 126 may receive polling communications from the LwM2M device 114 at regular intervals following powering on of the LwM2M device 114. The polling communications may query whether there are any queued LwM2M commands for transmission, from the adaptor 126, and to the LwM2M device 114. In response to the polling communications, the controller 144 of the adaptor 126 determines whether there are any LwM2M commands queued in respect of the LwM2M device 114. If the controller 144 determines that there is a queued LwM2M command, then the controller 144 controls the communications module 132 to transmit the queued command to the LwM2M device 114.

In this example, the controller 144 determines that the second LwM2M command is queued, and therefore controls the communications module 132 to transmit the second LwM2M command to the LwM2M device 114.

The adaptor 126 may receive, from the LwM2M device 114, a response message from the LwM2M device 114 indicating the outcome of the second LwM2M command execution (and any requested data).

418: On receipt of the response message from the LwM2M device 114, the controller 144 may control the staging module 148 to queue the next LwM2M command in

the sequence of execution, which in this case is the final LwM2M command. The staging module 148 may determine the next LwM2M command to queue based on the sequence ID of the remaining generated LwM2M commands.

Similarly to the process undertaken in step 414, the staging module 148 may queue the final LwM2M command in dependence on the successful completion of the previous LwM2M command (in this case the second LwM2M command).

420: The controller 144 controls the communications module 132 to transmit the queued final LwM2M command to the LwM2M device 114 for execution of the final LwM2M command.

Similarly to the process undertaken in step 416, the queued final LwM2M command may be transmitted to the LwM2M device 114 in response to a polling communication received by the adaptor 126 from the LwM2M device 114.

422: Once the final LwM2M command has been transmitted to the LwM2M device 114 for execution, and the response message had been received by the adaptor 126 and from the LwM2M device 114, the validation module 150 may determine whether each of the plurality of LwM2M commands (sharing the common correlation ID) have been successfully completed by the LwM2M device 114.

In arrangements in which the staging module 148 queues the next LwM2M command in the execution sequence in dependence on the successful completion of the previous LwM2M command (in this case the second LwM2M command), then the validation module 150 may simply determine the successful completion of the final LwM2M command by the LwM2M device 114. This is because in such arrangements, the validation module 150 has already determined whether the previously undertaken LwM2M commands have been successful (and has only progressed with subsequent LwM2M commands if the previous LwM2M commands have been successful).

In some arrangements, the staging module 148 queues the next LwM2M command in the execution sequence without determining whether the previous command has been successfully completed. In such arrangements, the

validation module 150 may determine whether each of the plurality of LwM2M commands sharing the common correlation ID have been successfully completed by the LwM2M device 114 in a similar manner to that outlined above in respect of method step 318.

424: If the validation module 150 determines that all of the plurality of LwM2M commands sharing the common correlation ID have been successfully completed, the controller 144 controls the command response generator 152 to generate a command response for transmission to the headend system 110. A single command response may be generated by the command response generator 152 for transmission to the headend system 110, regardless of how many correlated LwM2M commands are executed by the LwM2M device 114, and therefore regardless of how many LwM2M commands and response messages are exchanged between the adaptor 126 and the LwM2M device 114. The single response may comprise the correlation ID, or the original headend system command ID (which may be the same as the correlation ID in some examples), to allow the headend system 110 to identify the command that the response relates to.

The command response may comprise an indication of the successful completion of the command by the LwM2M device 114, any and any relevant data/status updates that have resulted from the execution of the command so that the headend system 110 may update any relevant records. Alternatively, the command response may comprise an error message, indicating that the command could not be successfully completed, and details relating to the error (e.g. the error type, severity, etc.).

The controller 144 controls the communications module 132 of the adaptor 126 to transmit the generated command response to the headend system 110.

In alternative arrangements, the LwM2M commands generated may simply all be queued and transmitted to the LwM2M device in response to the trigger (e.g. the wake-up signal). That is, transmission of a subsequent command by the adaptor 126 may not be dependent on receipt of an acknowledgement or a response message from the LwM2M device 114. This may lead to more advantageous de-queuing times.

The skilled person will appreciate that in some situations, the head end system 110 may generate a plurality of commands for implementation by the LwM2M device 114. As such, multiple sets of correlated LwM2M commands may be generated by the LwM2M command generator 146 for implementation by the LwM2M device 114. For example, a first set of correlated LwM2M commands may be generated based on a to a first headend system command; a second set of correlated LwM2M commands may be generated based on a second headend system command, etc.

The adaptor 126 may further comprise a prioritisation module configured to prioritise the sets of correlated LwM2M commands and define an order for transmission to the LwM2M device 114 for implementation. That is, the prioritisation module may assign a priority ID to each set of correlated LwM2M commands, indicating the order in which each set of correlated LwM2M commands should be transmitted to the LwM2M device 114 for implementation. The priority ID may be assigned based on the order in which the adaptor 126 receives the commands from the head end system 110 and/or an urgency indication within the headend system command. The sets of correlated LwM2M commands may be queued for transmission to the LwM2M device 114 in a particular order based on the priority IDs.

The prioritisation module may be configured to re-prioritise the order in which each set of correlated LwM2M commands should be transmitted to the LwM2M device 114 for implementation (i.e. re-queue the sets of LwM2M commands), for example in response to receipt of a subsequent set of correlated LwM2M commands, or an input that the existing queued sets of correlated LwM2M commands should be re-prioritised. For example, if a subsequent command is received from the headend system, with an urgency level that indicates that the subsequent command should be implemented before any other commands, the prioritisation module may be configured to reorder the queued sets of correlated LwM2M commands to take account of the new, more urgent, set of correlated LwM2M commands (which may comprise reassigning priority IDs).

It will be appreciated by the person of skill in the art that various modifications may be made to the above described embodiments without departing from the scope of the invention. The word “exemplary” is used herein to mean “an example”. Any embodiment

described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other embodiments.

A computer program may be configured to provide any of the above described methods. The computer program may be provided on a computer readable medium. The computer program may be a computer program product. The product may comprise a non-transitory computer usable storage medium. The computer program product may have computer-readable program code embodied in the medium configured to perform the method. The computer program product may be configured to cause at least one processor to perform some or all of the method.

Various methods and apparatus are described herein with reference to block diagrams or flowchart illustrations of computer-implemented methods, apparatus (systems and/or devices) and/or computer program products. It is understood that a block of the block diagrams and/or flowchart illustrations, and combinations of blocks in the block diagrams and/or flowchart illustrations, can be implemented by computer program instructions that are performed by one or more computer circuits. These computer program instructions may be provided to a processor circuit of a general purpose computer circuit, special purpose computer circuit, and/or other programmable data processing circuit to produce a machine, such that the instructions, which execute via the processor of the computer and/or other programmable data processing apparatus, transform and control transistors, values stored in memory locations, and other hardware components within such circuitry to implement the functions/acts specified in the block diagrams and/or flowchart block or blocks, and thereby create means (functionality) and/or structure for implementing the functions/acts specified in the block diagrams and/or flowchart block(s).

Computer program instructions may also be stored in a computer-readable medium that can direct a computer or other programmable data processing apparatus to function in a particular manner, such that the instructions stored in the computer-readable medium produce an article of manufacture including instructions which implement the functions/acts specified in the block diagrams and/or flowchart block or blocks.

A tangible, non-transitory computer-readable medium may include an electronic, magnetic, optical, electromagnetic, or semiconductor data storage system, apparatus,

or device. More specific examples of the computer-readable medium would include the following: a portable computer diskette, a random access memory (RAM) circuit, a read-only memory (ROM) circuit, an erasable programmable read-only memory (EPROM or Flash memory) circuit, a portable compact disc read-only memory (CD-ROM), and a portable digital video disc read-only memory (DVD/Blu-ray).

The computer program instructions may also be loaded onto a computer and/or other programmable data processing apparatus to cause a series of operational steps to be performed on the computer and/or other programmable apparatus to produce a computer-implemented process such that the instructions which execute on the computer or other programmable apparatus provide steps for implementing the functions/acts specified in the block diagrams and/or flowchart block or blocks.

Accordingly, the invention may be embodied in hardware and/or in software (including firmware, resident software, micro-code, etc.) that runs on a processor, which may collectively be referred to as "circuitry," "a module" or variants thereof.

It should also be noted that in some alternate implementations, the functions/acts noted in the blocks may occur out of the order noted in the flowcharts. For example, two blocks shown in succession may in fact be executed substantially concurrently or the blocks may sometimes be executed in the reverse order, depending upon the functionality/acts involved. Moreover, the functionality of a given block of the flowcharts and/or block diagrams may be separated into multiple blocks and/or the functionality of two or more blocks of the flowcharts and/or block diagrams may be at least partially integrated. Finally, other blocks may be added/inserted between the blocks that are illustrated.

CLAIMS:

1. A method for use in a utility network for generating Lightweight Machine to Machine, LwM2M, commands for implementation by one or more utility metering devices, the method comprising:

generating, by a headend system, a command for implementation by a utility meter, wherein the utility meter comprises a LwM2M utility meter that implements a LwM2M protocol, and wherein the command generated by the headend system is in a format that is not suitable for execution by the LwM2M utility meter;

transmitting, by the headend system, the command to an adaptor;

generating, by the adaptor and based on the received command, a plurality of LwM2M commands, each LwM2M command defining one or more LwM2M operations for implementation by the LwM2M utility meter to satisfy the command generated by the headend system;

executing, by the LwM2M utility meter, the plurality of LwM2M commands; and

sending, by the adaptor and to the headend system, a response to the command generated by the headend system, based on the execution of the plurality of LwM2M commands by the LwM2M utility meter.

2. A method according to claim 1, wherein the LwM2M operations comprise one or more of: a read operation, a write operation, an execute operation and a delete operation.

3. A method according to claim 1 or 2, wherein the plurality of LwM2M commands are generated based on a single command received from the headend system.

4. A method according to any preceding claim, wherein the generation of the plurality of LwM2M commands comprises mapping, by a mapping module, the command generated by the headend system to the one or more LwM2M operations.

5. A method according to any preceding claim, further comprising:

queueing, by the adaptor, one or more of the plurality of LwM2M commands for execution by the LwM2M utility meter; and

transmitting, by the adaptor and to the LwM2M utility meter, the one or more queued LwM2M commands for execution by the LwM2M utility meter.

6. A method according to claim 5, further comprising:

receiving, by the adaptor and from the LwM2M utility meter, a wake-up signal, and

wherein, the adaptor transmits the one or more queued LwM2M commands to the LwM2M utility meter for execution in dependence on receiving the wake-up signal.

7. A method according to claim 5 or 6, wherein the method further comprises:

determining, by the adaptor, a sequence of execution of the plurality of generated LwM2M commands, and

wherein the first LwM2M command in the sequence of execution is queued by the adaptor for transmission to the LwM2M utility meter.

8. A method according to claim 7, further comprising:

queuing, by the adaptor, a subsequent LwM2M command, as defined by the sequence of execution, in dependence on transmission of a previous LwM2M command in the sequence of execution to the LwM2M utility meter for execution.

9. A method according to claim 8, wherein the subsequent LwM2M command is queued in dependence on successful completion of the previous LwM2M command in the sequence of execution, by the LwM2M device.

10. A method according to any preceding claim, further comprising:

transmitting, by the adaptor and to the headend system, a response indicating successful completion of the command in dependence on successful completion of each of the plurality of LwM2M commands.

11. A method according to any preceding claim, further comprising:

assigning, by the adaptor, a common correlation ID to each of the plurality of LwM2M commands generated based on the command received from the headend system.

12. A method according to claim 11, further comprising:

transmitting, by the adaptor and to the headend system, a success response based on the successful execution of the plurality of LwM2M commands sharing the common correlation ID, by the LwM2M utility meter.

13. A method according to any of claims 4 to 12, wherein the command generated by the headend system is a first command, and the plurality of LwM2M commands is a first plurality of LwM2M commands, the method further comprising:

generating, by the headend system, a second command for implementation by the LwM2M utility meter;

generating, by the adaptor and based on the second command, a second plurality of LwM2M commands;

assigning, by the adaptor and to each of the first and second plurality of LwM2M commands, a priority identifier; and

queuing the first plurality of LwM2M commands and the second plurality of LwM2M commands, for execution by the LwM2M utility meter, in an order based on the priority identifiers of the first and second plurality of LwM2M commands.

14. A method according to claim 13, further comprising:

re-ordering the queued first and second plurality of LwM2M commands, based on further information received by the adaptor and from the headend system.

15. A method according to claim 14, wherein the further information comprises a third command generated by the headend system and received by the adaptor, and wherein the third plurality of LwM2M commands generated based on the third command comprise a priority identifier indicating that implementation of the third command is of higher priority than the first and/or second command.

16. A method according to any preceding claim, further comprising:

receiving, by the adaptor and for each of the plurality of LwM2M commands transmitted to the LwM2M utility meter, a response message from the LwM2M utility meter indicating whether completion of a respective LwM2M command is successful or unsuccessful; and

undertaking, by an error handler of the adaptor, an error management action, if the response message indicates unsuccessful completion of the respective LwM2M command.

17. A method according to claim 16, wherein the error management action comprises determining, by the error handler and based on a type of error indicated within the response message, an alternative method for re-transmission of the LwM2M command.

18. A method according to claim 17, wherein if the type of error indicates that the LwM2M command exceeds a packet size limit, the method further comprises:

instructing, by the error handler, generation of a subset of LwM2M commands corresponding to the unsuccessfully completed LwM2M command, each LwM2M command of the subset of LwM2M commands comprising a packet size below the packet size limit; and

transmitting, by the adaptor, the subset of LwM2M commands to the LwM2M utility meter for execution by the LwM2M utility meter.

19. A system comprising:

a LwM2M utility meter configured to implement a LwM2M protocol;

a headend system configured to generate a command for implementation by the LwM2M utility meter, wherein the command generated by the headend system is in a format that is not suitable for execution by the LwM2M utility meter; and

an adaptor configured to receive the command generated by the headend system and generate, based on the received command, a plurality of LwM2M commands, each LwM2M command defining one or more LwM2M operations for implementation by the LwM2M utility meter to satisfy the command generated by the headend system,

wherein the LwM2M utility meter is further configured to execute the plurality of LwM2M commands, and

wherein the adaptor is further configured to send, to the headend system, a response to the command generated by the headend system, based on the execution of the plurality of LwM2M commands by the LwM2M utility meter.