NATIONAL TRANSMISSION AND DESPATCH COMPANY

(NTDC)

SPECIFICATION DDS-110:2012

METER DATA COLLECTION (MDC) SERVER
TO BE USED IN
ADVANCED METERING INFRASTRUCTURE (AMI)
(Standards & Specifications)

Contents

SPECIFICATION FOR METER DATA COLLECTION (MDC) SERVER TO BE USED IN ADVANCED METERING INFRASTRUCTURE (AMI)

1. Foreword:

1.1. This standard specification has been prepared by the Engineering Directorate of Chief Engineer (D&S).

1.2. This specification is intended for procurement of material and not for contract.

1.3. This specification is subject to revision as and when required.

2. Scope:

2.1. This specification is being prepared for the introduction of a Meter Data Collection (MDC) server to be used with Advanced Metering Infrastructure (AMI).

2.2. This specification is intended for the introduction of MDC based Meter Reading System to be used with Meters as per relevant specification DDS-50:2007, DDS-60:2007 and DDS-65:03 having communication as per specification DDS-98:2011.

2.3. This specification covers the aspects of Meter Data Collection (MDC) server specification, MDC Software Specification, MDC integration with AMI and multi-vendor protocol.

2.4. This specification shall cover the multi-vendor compliance of the existing manufacturers as well as the prospective vendors.

2.5. This specification shall cover the interoperability between meters of different vendors with MDCs of different vendors.

2.6. This specification shall cover the interoperability between MDCs of different vendors with MDM of different vendors.

3. Definitions:

3.1. **AMI:**

Advanced Metering Infrastructure (AMI) are systems that measure, collect and analyze energy usage, and communicate with metering devices such as electricity

meters, either on request or on a schedule. These systems include hardware, software, communications and Meter Data Management (MDM) software etc.

3.2. **Meter Serial Number:**

Meter serial number is printed on meter label and stored in meter memory.

3.3. **Manufacturer Identifier**:

The unique numeric identification of each meter manufacturer.

3.4. **Meter Type:**

Meter Type shall correspond to design specification i.e. DDS-50:2007, DDS-60:2007 & DDS-65:03.

3.5. **MDC:**

Meter Data Collection Server is a system in Advanced Metering Infrastructure (AMI) used to obtain data from the meters and store it into the database.

3.6. **Wakeup:**

Wakeup is a request sent by Meter Data Collection (MDC) Server to the meter for making a connection.

3.7. **Pull:**

Pull is a mechanism in which MDC demands for specific information or data from meter and meter sends data in response.

3.8. **GPRS:**

General Packet Radio Service is a packet oriented communication technology. This technology is specifically used for communication between mobile phone and internet.

3.9. **SMS:**

Short Message Service is a communication service standardized in the GSM mobile communication system.

3.10. **Voice Call:**

Voice Call is a GSM service used to interchange speech data between two communication entities.

3.11. **TCP:**

Transmission Control Protocol is transport layer protocol for Internet Protocol (IP).

3.12. **IP:**

Internet Protocol is a networking layer protocol and forms the core of Internet all around the world.

3.13. **DVD:**

DVD is an optical disc storage media format, offering higher storage capacity than Compact Discs while having the same dimensions.

3.14. **RAID:**

RAID (an acronym for redundant array of independent disks; originally redundant array of inexpensive disks) is a storage technology that combines multiple disk drive components into a logical unit to increase the data storage reliability.

3.15. **OBIS:**

The OBIS OBject Identification System defines identification codes for all data in DLMS/COSEM compliant metering equipment.

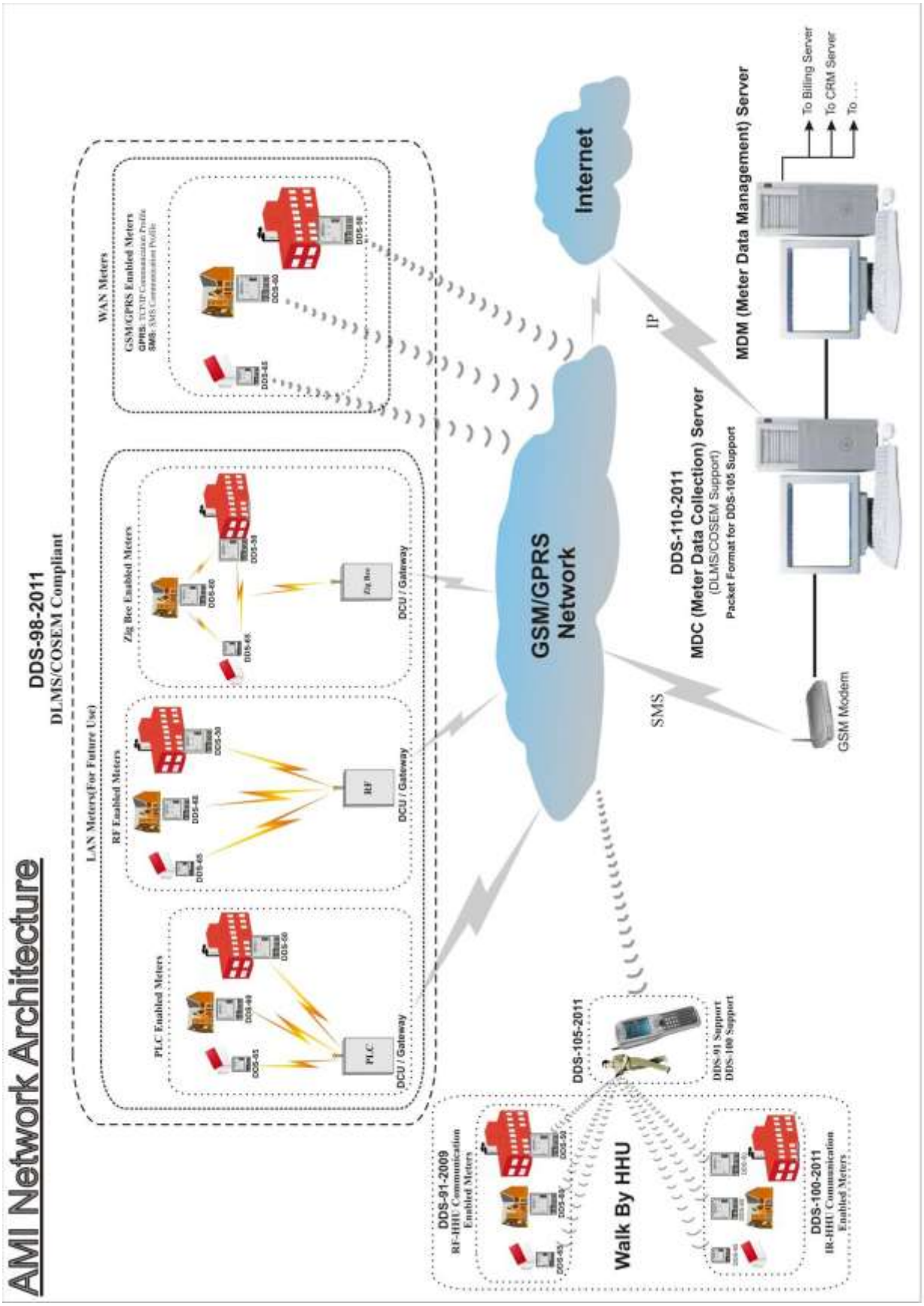
3.16. **Mockup:**

In manufacturing and design, a mockup is a scaled model of a design or device, used for teaching, demonstration, design evaluation, promotion, and other purposes. A mockup is a prototype if it provides at least part of the functionality of a system and enables testing of a design.

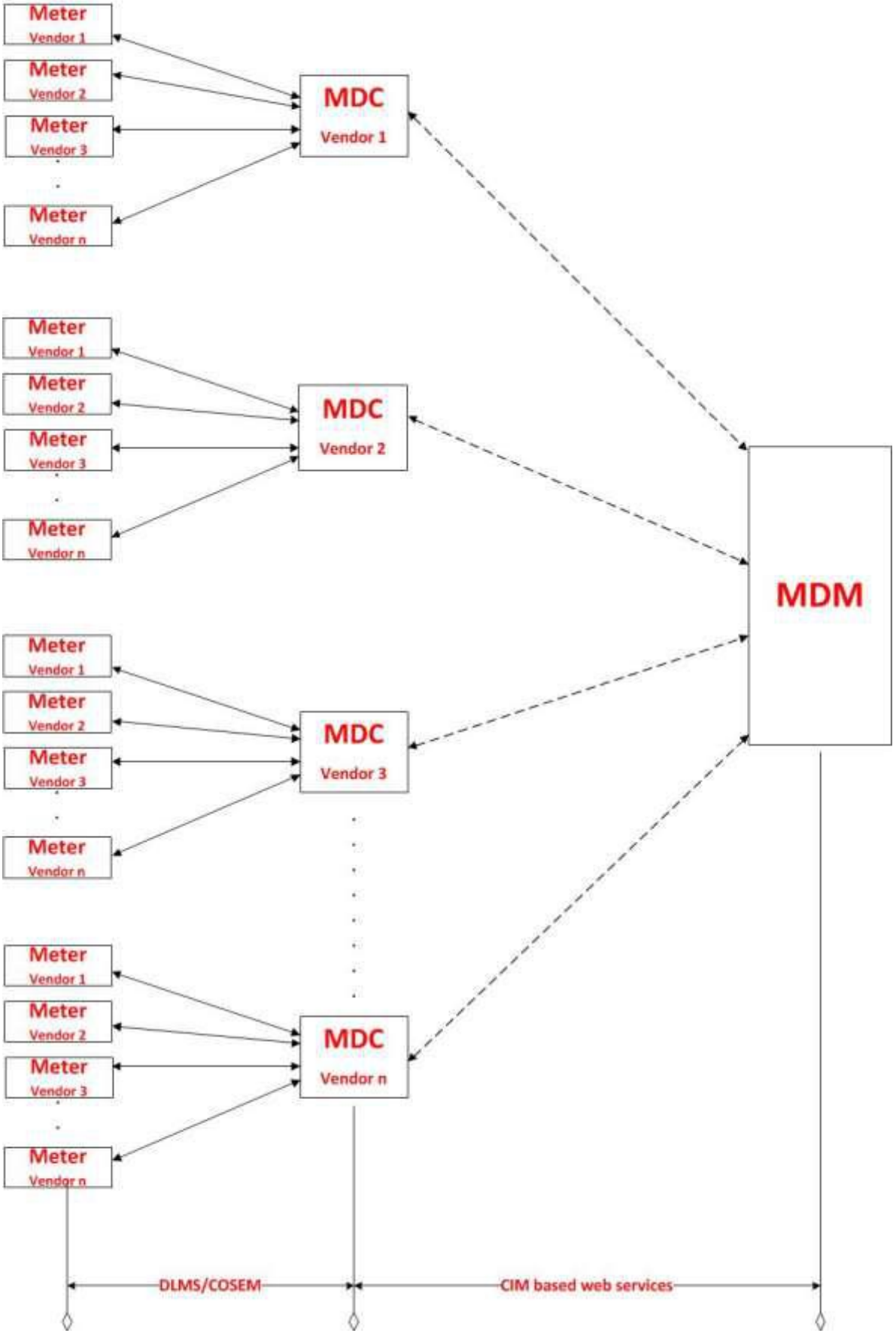
3.17. **CIM:**

The Common Information Model (CIM) is an abstract information model that provides data understanding through the identification of the relationships and associations of the data within a utility enterprise.

4. Network Architecture:



5. AMI Interoperability



Communication between meter and MDC over TCP/IP shall follow clause 11.1 “Communication protocol” and clause 11.2 “Communication modes” of this document. Web services for MDM shall follow clause 12 of this specification.

6. Software Architecture:

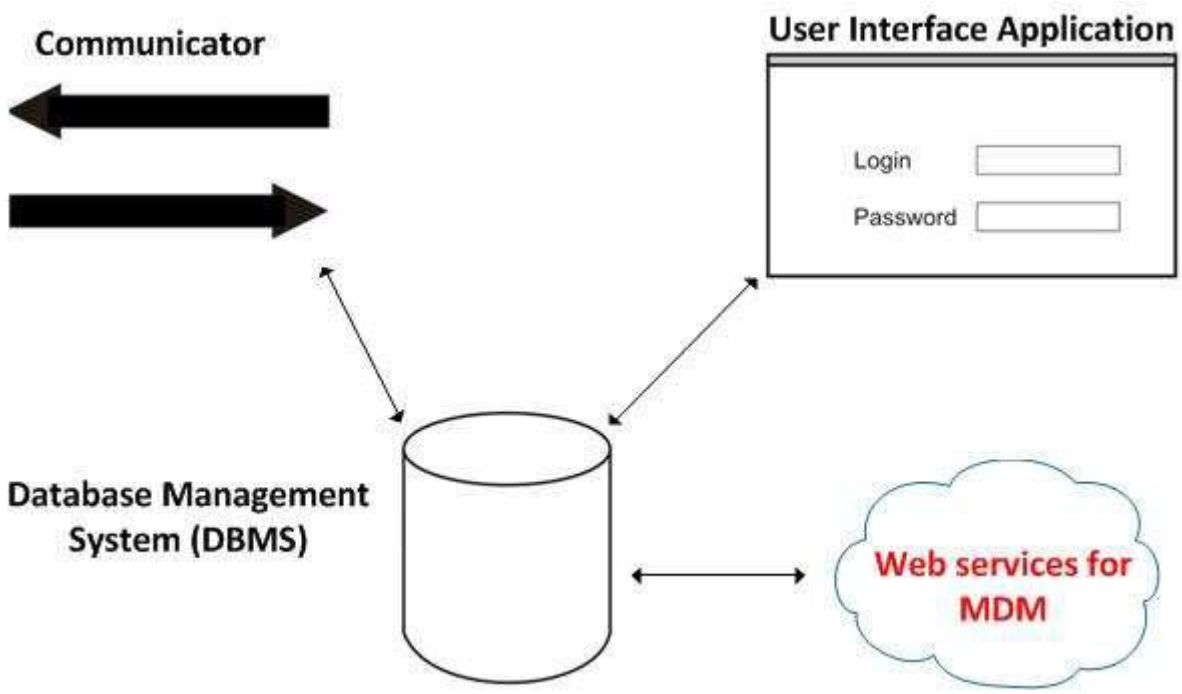
Meter Data Collection (MDC) Server shall have following:

6.1. User Interface Application

6.2. Database Management System (DBMS)

6.3. Communicator

6.4. Web Services for MDM



7. User Interface Application:

7.1. It is responsible for providing a graphical interface to the user to view the data received from the communicator; send meter configuration and other relevant data to the meters.

7.2. MDC software shall be capable of displaying all metering data as per relevant meter’s specifications (DDS-50:2007, DDS-60:2007 and DDS-65:03).

7.3. The interface application shall provide customer hierarchy for at least 3 levels. It shall also provide user management and user rights. User rights shall limit a user to a certain level of customer hierarchy.

7.4. The user interface application shall provide a mechanism for meter management functions. For example, adding or deleting a meter, meter configuration etc.

7.5. The user interface application shall provide a Graphical User Interface (GUI).

8. Database Management System (DBMS):

8.1. The Database Management System (DBMS) is a central repository for all metering data collected through communicator.

8.2. The host software shall have capabilities to store and report all metering data in a database management system as required in the relevant meter specifications like billing data, event data, load profiles etc.

8.3. It shall provide mechanism for periodic as well as on-demand backup; and also, it shall be able to restore data on request.

8.4. The Database server shall have backup interface for DVD.

8.5. The database shall be capable to hold the metering data as specified in “Schedule of Technical Data” for a period specified therein (Annex-I)

9. Hardware:

9.1. The Database Management Server shall have RAID hard drives.

9.2. There shall be at least one backup server (optional).

9.3. Server Class machines shall be used.

9.4. Online UPS shall be used.

9.5. Reliable Internet Connection with at least one static public IP address shall be used.

9.6. There shall be arrangement for at least two generators.

9.7. Data center room shall be properly air-conditioned.

9.8. There shall be at least one GSM Modem for Wakeup SMS/voice call to send one SMS/ make Voice call all at a time.

9.9. It shall be possible to use more than one server for load sharing when data load is high. The vender will propose system on the basis of requirements given by the customer in Annex-1.

10. GSM Modem:

GSM modem of following specifications shall be used with MDC

10.1. Quad-Band GSM/GPRS 850/900/1800/1900MHz

10.2. Compliant to GSM phase 2/2+

10.2.1. Class 4 (2W @ 850/900MHz)

10.2.2. Class 1(1W @ 1800/1900MHz)

10.3. The Modem shall have the capability for SMS and voice call communication

10.4. The modem shall have LED indicator visible from the distance.

10.5. The LED shall indicate the operational status of the modem like:

10.5.1. Fast blinking LED indicates absence of SIM card or no network service or insufficient signal strength.

10.5.2. OFF LED indicates modem is not working.

10.5.3. Slow blinking LED indicates healthiness of the modem and is registered to the GSM network

10.6. The SIM card shall be field replaceable

10.7. It shall have RS232 or USB interface for computer

11. Communicator:

It is responsible for handling all communications between meters and MDC and sending data into database for the user interface application.

11.1 Communication Protocols:

10.1.1. MDC shall support DLMS/COSEM communication protocol.

10.1.2. MDC shall also accept output data files as specified in DDS-105:2011 (if applicable).

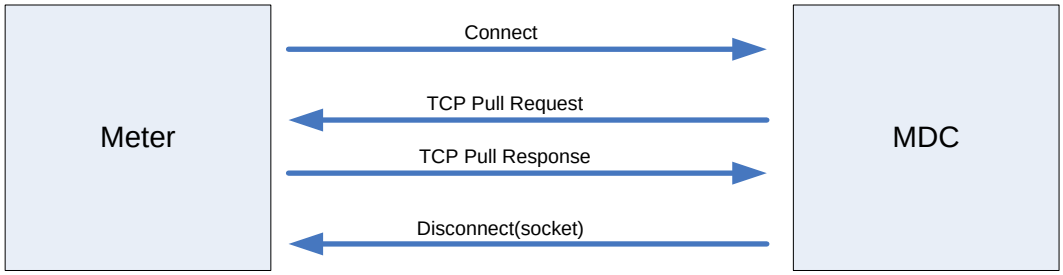
11.2. Communication Modes:

Meters shall be configurable to one of following Mode-I or Mode-II at a time

11.2.1. Mode I (GPRS as Primary Medium and SMS/Voice Call for Wakeup):

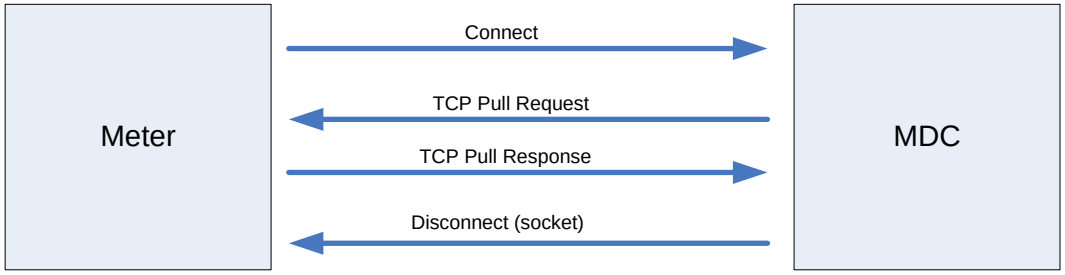
11.2.1.1. Pull at Preprogrammed Interval:

Meter shall establish a TCP connection with the MDC server at a predefined interval via GPRS. Number of retries and retry interval shall be programmable. The MDC shall pull data from the meter as shown in the figure below:



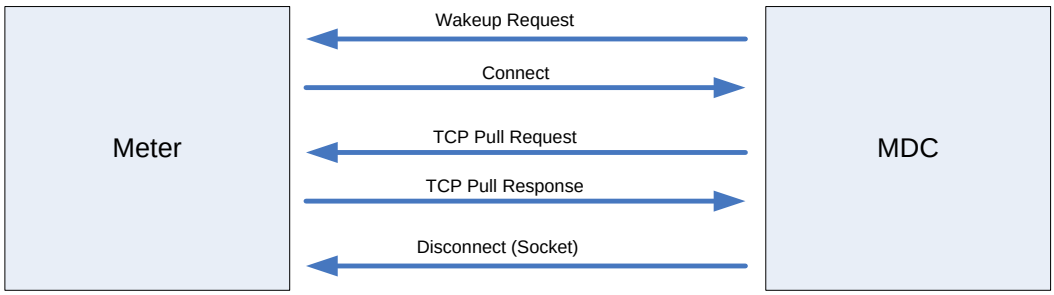
11.2.1.2. Pull at Events & Alarms:

The meter has a number of events/alarms; any of these events/alarms can be configured as a major alarm. If a major alarm/event occurs, meter shall establish a TCP/IP connection with the MDC via GPRS. Number of retries and retry interval shall be programmable. The MDC shall pull data from the meter as shown in the figure below:



11.2.1.3. Pull on-Demand:

For on-demand data the MDC shall send wakeup SMS or initiate a wakeup voice call for the meter. In response the meter shall establish a TCP/IP connection with the MDC server via GPRS. Number of retries and retry interval shall be programmable. The MDC shall pull data from the meter as shown in the figure below:



11.2.1.4. DLMS/COSEM application layer for mode I:

For IP based communication DLMS/COSEM specification suggest TCP communication profile. Meter shall support at least three associations as described below:

Association 0: (Public Association)

1. Client SAP: 0x10
2. Server SAP: 0x0001
3. Application context: Logical Name (LN) referencing
4. Authentication mechanism: Lowest Level Security (No security).
5. xDLMS context: At least meter shall support following services in association 0
    - get,
    - block-transfer-with-get-or-read,
6. Security suite: No security
7. Security context: No security

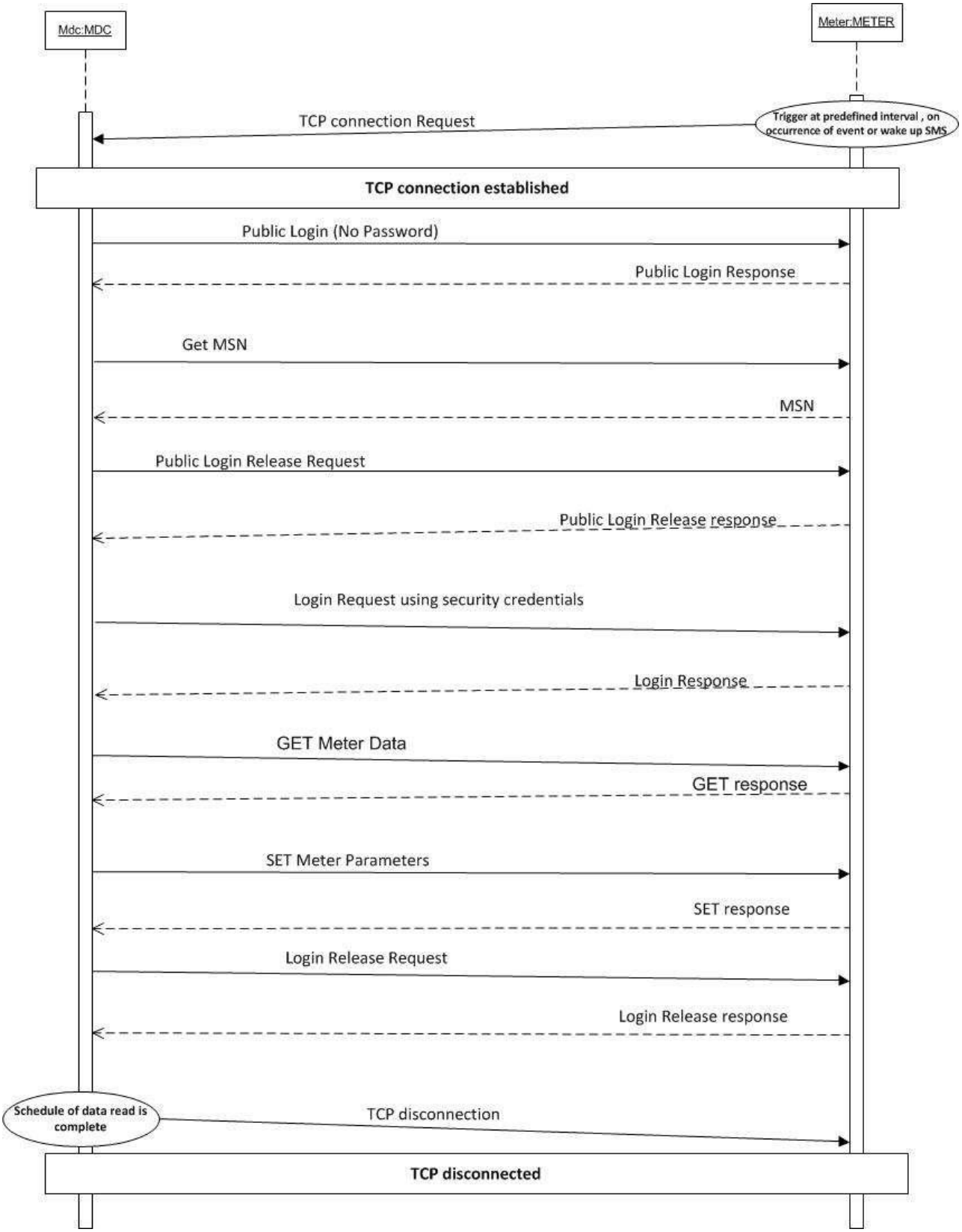
Association 1: (Association for Management user)

1. Client SAP: 0x01
2. Server SAP: 0x0001
3. Application context: Logical Name (LN) referencing with ciphering
4. Authentication mechanism: HLS using GMAC with mechanism id 5.
5. xDLMS context: At minimum meter must support following services in association 1
    - get,
    - set,
    - action,
    - block-transfer-with-get-or-read,
    - block-transfer-with-set-or-write,
    - block-transfer-with-action
6. Security suite: Security suite(0)
7. Security context: Authentication and encryption as per security suite 0.

Association 2: (Association for reader)

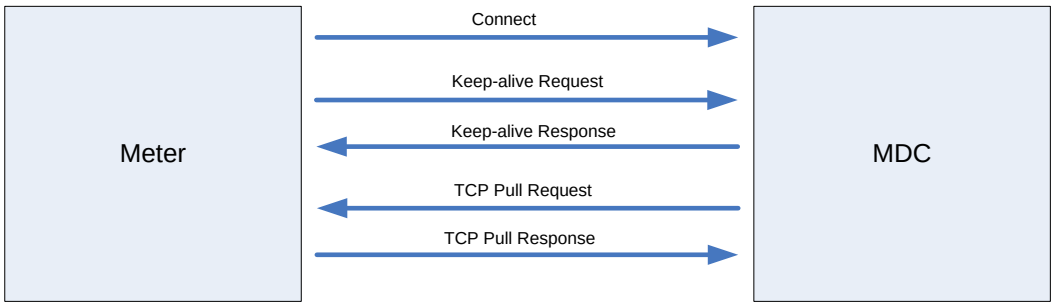
1. Client SAP: 0x12
2. Server SAP: 0x0001
3. Application context: Logical Name (LN) referencing with ciphering
4. Authentication mechanism: HLS using GMAC with mechanism id 5.
5. xDLMS context: At minimum meter shall support following services in association 2
    - get,
    - action,
    - block-transfer-with-get-or-read,
    - block-transfer-with-action
6. Security suite: Security suite(0)
7. Security context: Authentication and encryption as per security suite 0.

11.2.1.5. Sequence diagram for data read in Mode I:



11.2.2. Mode-II (GPRS always alive):

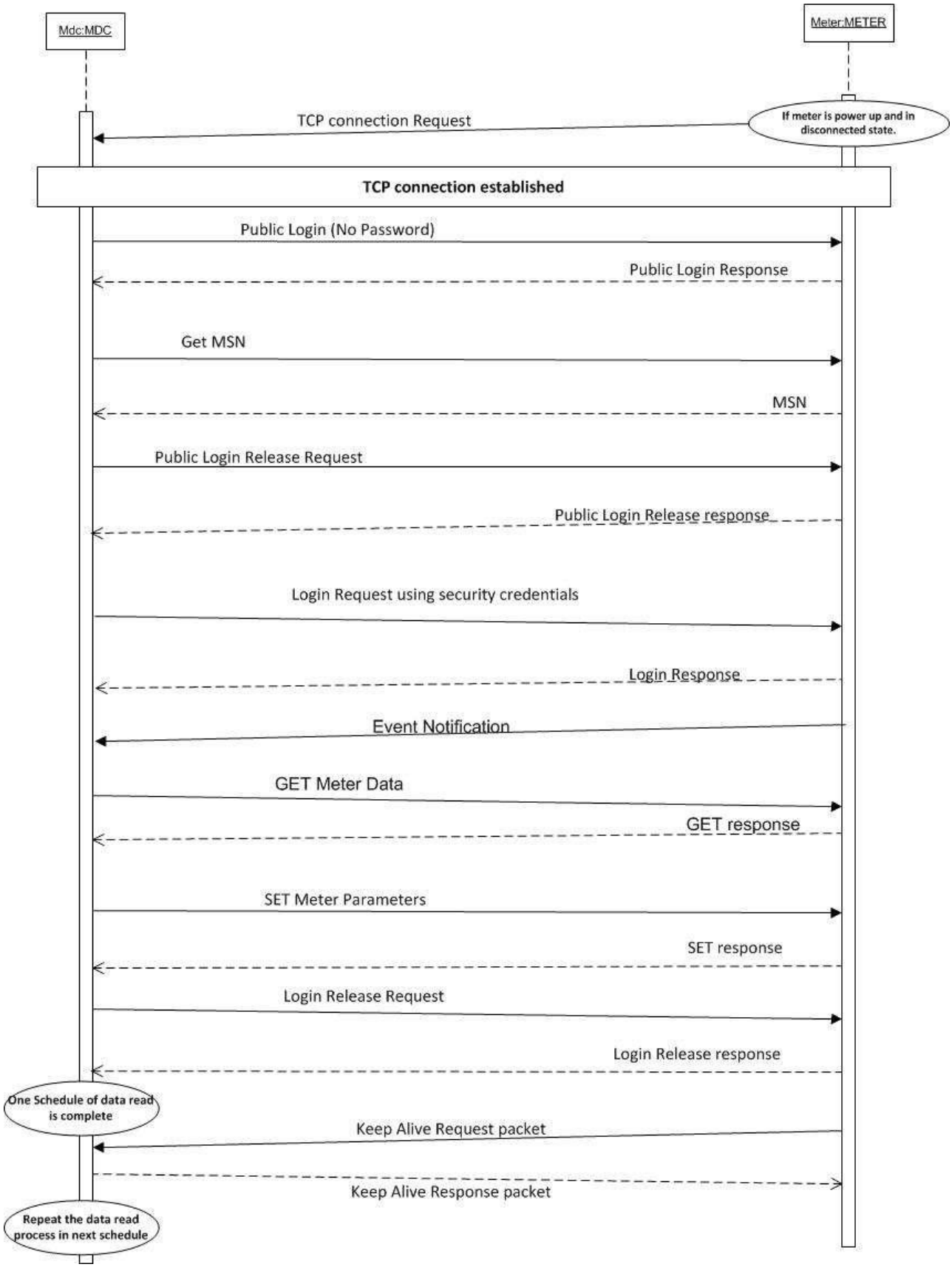
Meter shall always maintain a TCP/IP connection with the MDC server via GPRS. In this process the meter shall send periodic heart beat/keep-alive packets to the MDC server. The MDC shall pull data from the meter as and when required as shown in the figure below:



11.2.2.1. DLMS/COSEM application layer for mode II:

Application layer in mode II shall follow all points mentioned in section 10.2.1.4.for application layer for mode I

11.2.2.2. Detailed Sequence Diagram for data read in Mode II:



11.3. Communication Channels:

Meter shall have two programmable IP addresses. For communication via GPRS one of two programmed IP address shall be used as primary address and the other IP address shall be used as secondary address. The meter shall try to establish a TCP/IP connection with the primary IP address if it fails to do so then the meter shall try to establish TCP/IP connection with secondary IP address.

Meter shall have at least three programmable GSM Numbers. In case of Wakeup command via SMS or voice call the meter shall only respond to the wakeup commands originated from the GSM number programmed in it.

For configuring and reading of communication mode, two IP addresses and three programmable GSM numbers, following OBIS codes shall be used.

Quantity	**OBIS code**	**Interface Class**	**Attribute**
Two IP addresses	0.0.2.2.0.255	**Auto connect (class_id = 29, version = 2)**	6
Three GSM numbers	0.0.2.1.0.255	**Auto answer (class_id = 28, version = 2)**	7
Communication mode (Mode I and Mode II)	0.0.2.2.0.255	**Auto connect (class_id = 29, version = 2)**	2

These OBIS code shall have read & write access in association 1 and read only access in association 2.

11.4. Communication OBIS codes:

10.4.1. Energy and Demand:

Energy and Demand quantities OBIS codes shall only be accessible in Association 1 and Association 2. For representing decimal values in interface classes Register (class Id=3, version 0) and Extended Register (class Id=4, version 0), Attribute 2 (value) and Attribute 3(Scalar_unit) shall be used together i.e. Value x 10^{scalar}.

Standard OBIS codes for quantities used for billing purposes are shown below:

Cumulative Active energy absolute (Abs KWH)		**Interface Class**	**Choice of Data type for Attribute 2**
Total	1.0.15.8.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 1	1.0.15.8.1.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 2	1.0.15.8.2.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 3	1.0.15.8.3.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 4	1.0.15.8.4.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned

Cumulative Reactive Energy KVarh Q1 + Q3 in single packet (Pakistan Specific)		**Interface Class** (These are country specific OBIS codes, any suitable class id can be used)	**Choice of Data type for Attribute 2**
Total	1.0.94.92.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned

Tariff 1	1.0.94.92.1.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 2	1.0.94.92.2.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 3	1.0.94.92.3.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 4	1.0.94.92.4.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned

Maximum Demand Active absolute (MDI-KW abs)		**Interface Class**	**Choice of Data type for Attribute 2**
Total	1.0.15.6.0.255	**Extended Register (class_id = 4, version = 0)**	Double-long-unsigned
Tariff 1	1.0.15.6.1.255	**Extended Register (class_id = 4, version = 0)**	Double-long-unsigned
Tariff 2	1.0.15.6.2.255	**Extended Register (class_id = 4, version = 0)**	Double-long-unsigned
Tariff 3	1.0.15.6.3.255	**Extended Register (class_id = 4, version = 0)**	Double-long-unsigned
Tariff 4	1.0.15.6.4.255	**Extended Register (class_id = 4, version = 0)**	Double-long-unsigned

Cumulative Maximum Demand Active absolute (Cum-MDI-KW abs)		**Interface Class**	**Choice of Data type for Attribute 2**
Total	1.0.15.2.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 1	1.0.15.2.1.255	**Register**	Double-long-unsigned

		(class_id = 3, version = 0)	
Tariff 2	1.0.15.2.2.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 3	1.0.15.2.3.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Tariff 4	1.0.15.2.4.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned

Associated data		**Interface Class**	**Choice of Data type for Attribute 2**
MDI Reset Date and Time	1.0.0.1.2.255	**Data (class_id = 1, version = 0)**	Octet String(12)
MDI Reset Count	1.0.0.1.0.255	**Data (class_id = 1, version = 0)**	Long-unsigned

11.4.2. Instantaneous:

These instantaneous quantities OBIS codes shall only be accessible in Association 1 and Association 2. For representing decimal values in interface classes Register (class Id=3, version 0) and Extended Register (class Id=4, version 0), Attribute 2 (value) and Attribute 3(Scalar_unit) shall be used together i.e. Value x 10^{scalar}.

Standard OBIS codes instantaneous data elements are shown below;

Quantity	**OBIS code**	**Interface Class**	**Choice of Data type for Attribute 2**
Aggregate Active power Import (KW+)	1.0.1.7.0.255	**Register (class_id = 3,**	Double-long-unsigned

		version = 0)	
Aggregate Active Power Export (KW-)	1.0.2.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Aggregate Reactive power Import	1.0.3.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Aggregate Reactive power export	1.0.4.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Voltage Phase A	1.0.32.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Voltage Phase B	1.0.52.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Voltage Phase C	1.0.72.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Current Phase A	1.0.31.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Current Phase B	1.0.51.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Current Phase C	1.0.71.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Average Power Factor	1.0.13.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Frequency	1.0.14.7.0.255	**Register (class_id = 3, version = 0)**	Double-long-unsigned
Time	1.0.0.9.1.255	**Data (class_id = 1, version = 0)**	Octet String(4)
Date	1.0.0.9.2.255	**Data (class_id = 1, version = 0)**	Octet String(5)
Current Tariff	0.0.96.14.0.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned
CT ratio Numerator	1.0.0.4.2.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned
CT ratio Denominator	1.0.0.4.5.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned
PT ratio Numerator	1.0.0.4.3.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned
PT ratio Denominator	1.0.0.4.6.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned
Customer Code	0.0.96.1.10.255	**Data (class_id = 1, version = 0)**	Octet String

11.4.3. Events Codes and Event notification Packet:

For Event notification standard packet of DLMS/COSEM with Tag 0xC2 should be used. Referred to IEC-62056-53 OBIS code to be used in this Packet is 0.0.96.11.0.255.. This OBIS code shall be implemented with Interface Class **Data (class_id = 1, version = 0)** and choice of data type for attribute 2 should be long-unsigned. Only attribute 2 of event code object shall be sent in event notification packet. Alternatively, MDC can PULL events data using GET command with 0xC0. This OBIS code contains the event code of most recent event occurred. Event code list is implementation specific. List of Event codes for all necessary quantities is given below.

List of event Codes					
Quantity	**Event Code**	**DDS-50**	**DDS-60**	**DDS-65 Multi Tariff**	**DDS-65 Single Tariff**
MDI reset	101	Y	Y	Y	N
Parameterization	102	Y	Y	Y	N
Power fail start (will be reported when power is restored)	111	Y	Y	Y	N
Power fail end	112	Y	Y	Y	N
Phase failure	113	Y	Y	N	N
Over Volt	114	Y	N	N	N
Under Volt	115	Y	N	N	N
Demand Over Load	116	Y	N	N	N
Reverse Energy (Active energy)	117	Y	Y	N	N
Reverse Polarity	118	Y	Y	N	N
CT Bypass	121	Y	N	N	N

11.4.4. Keep-alive Packet Format:

The Transport Layer payload of Keep-alive request is defined in the table as follows:

Tag	**Length of Meter Number**	**Meter Number**
1 byte **(0xDD)**	1 byte	As Specified in Section 10.5

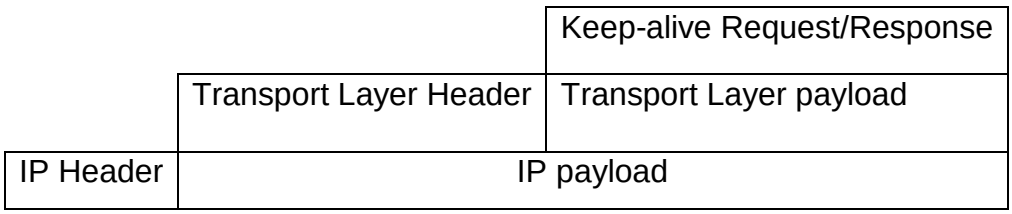
The Transport Layer payload of Keep-alive response is defined in the table as follows:

Tag
1 byte **(0xDA)**

The above request and response packets of Keep-alive commands are payload of Transport Layer. The reasons behind choosing it as Transport Layer payload are given below:

- To keep GPRS data traffic low.
- The purpose served by this packet is beyond the DLMS protocol specification.
- Tags used for Keep-alive request and response packets are utilized by DLMS specification for some other purpose.

Communication network layers stack, showing the generation of Keep-alive request and response, is shown below.



11.5. Meter Identifier:

The MDC shall communicate with meter using a unique 4-Byte Meter Identifier. The office of the Chief Engineer Design and Standards shall allocate prefix to each manufacturer a 4 byte meter identifier as mentioned below:

Meter Serial Number Format		
00-40	00-99	000001-999999
Manufacturer	Meter Type	Meter Serial Number
40 SBL	99 DDS-50	Start =000001 , End = 999999
39 PEL	98 DDS-60	
38 Escorts	97 DDS-65	
37 KBK	Etc.	
36 MicroTech		
35 CTI		
33 Ace Indigo		
32 Creative Electronics		
Etc.		

Meter identifier shall be readable from meter with OBIS code given below**;**

Quantity		**Interface Class**	**Choice of Data type for Attribute 2**
Meter Identifier	0.0.96.1.0.255	**Data (class_id = 1, version = 0)**	Double-long-unsigned

Meter identifier OBIS code shall be accessible in association 0, association 1 and association 2.

11.6 Load Profile

OBIS code to be used for load profile is:

Quantity	OBIS code	Attribute	Interface Class	Selective access
Load Profile data	1.0.99.1.0.255	2	Profile generic (class_id = 7, version = 1)	Range Descriptor and date&time as restricting object

Restricting_object in range descriptor shall be date&time with OBIS code given below.

Restricting object	OBIS code	Interface Class	Attribute
Date&time	0.0.1.0.0.255	Clock (class_id = 8, version = 0)	2

Load Profile data shall only be accessible in Association 1 and Association 2.

Load profile channels can be accessed using attribute 3 of same OBIS code.

Quantity	OBIS code	Attribute	Interface Class
Load Profile channels	1.0.99.1.0.255	3	Profile generic (class_id = 7, version = 1)

Load Profile channels shall have read and write access in Association 1 and read only access in Association 2

11.7 Billing data

OBIS code to be used for billing data is:

Quantity	OBIS code	Attribute	Interface Class	Selective access
Billing data	1.0.98.1.0.255	2	Profile generic (class_id = 7, version = 1)	Entry descriptor

Billing data shall only be accessible in Association 1 and Association 2.

Billing data quantities list shall be read only. It can be read using attribute 3 of same OBIS code.

Quantity	OBIS code	Attribute	Interface Class
Billing data quantities list	1.0.98.1.0.255	3	Profile generic (class_id = 7, version = 1)

Billing data quantities list shall have read only access in Association 1 and Association 2.

Along with the energy and demand quantities billing data quantities list shall contain MDI reset counter to support selective access using entry descriptor.

Restricting object	OBIS code	Interface Class	Attribute
MDI Reset Count	1.0.0.1.0.255	**Data (class_id = 1, version = 0)**	2

11.8 Events data

Meter shall store relevant event code as mentioned in section 10.4.3 along with occurrence date&time.

OBIS code for event code is also mentioned in section 10.4.3.

OBIS code to be used for events data is:

Quantity	OBIS code	Attribute	Interface Class	Selective access
Events data	0.0.99.98.0.255	2	Profile generic (class_id = 7, version = 1)	Entry descriptor

Events data shall only be accessible in Association 1 and Association 2.

OBIS codes captured in each entry of events log shall be:

Associated data	OBIS code	Interface Class	Attribute
Date &Time	0.0.1.0.0.255	**Clock (class_id = 8, version = 0)**	2
Events counter	0.0.96.15.0255	**Data (class_id = 1, version = 0)**	2
Events code	0.0.96.11.0.255	**Data (class_id = 1, version = 0)**	2

Data type for attribute 2 of Events counter OBIS code shall be Double-long-unsigned.

11.9 Contactor control

For remote ON and OFF of the contactor following OBIS code and method shall be used.

Restricting object	OBIS code	Interface Class	method
Remote OFF	0.0.96.3.10.255	Disconnect control (class_id = 70, version = 0)	1::remote_disconnect(0)
Remote ON	0.0.96.3.10.255	Disconnect control (class_id = 70, version = 0)	2::remote_connect(0)

These methods shall only be accessible in Association 1.

12. Web Services in MDC for MDM

MDC shall expose web services which are based on CIM IEC 61968-9 and IEC 61968-100.

- These web services shall be “SOAP” based and shall be using “*HTTPS*” as transport protocol.
- Messages exchanged between MDC and MDM shall follow Common Message Envelope given in IEC 61968-100.
- WS-Security is security extension to SOAP for web services. Event SOAP Header shall contain Username and Password for authentication purpose.
- MDC and MDM shall exchange SOAP messages over secure VPN (if not co-located) using HTTPS.
- Strongly typed web services shall be used for data exchange. The strongly-typed web service integration pattern is intended for use to implement semantic based interfaces in support of a SOA integration strategy. The strongly-typed pattern has the following characteristics:

    1) Uses SOAP-based web services, where fine-grained WSDLs are used to define a contract.

    2) Enables stronger payload validation by defining operation messages using strongly typed payloads

- WSDLs of web services shall follow the templates provided in Annex C of IEC 61968-100.
- XML schemas (XSDs) used in web services shall follow the Annex H of IEC 61968-9.
- WSDLs and their corresponding XML schemas (XSD) of web services shall be made accessible for MDM using VPN (if not co-located).
- XML serializer used in both MDC and MDM should be compliant to World Wide Web Consortium recommendation XSD 1.1.

12.1 Common message envelope

XML file formed on the basis of Common message envelope of IEC 61968-100 shall come under Body of SOAP envelope. XML schema for common message envelope is given in Annex A of IEC 61968-100. Major elements of the message are:-

- Header
- Request
- Reply
- Payload

Request is only present in request message. Reply element is only present in response messages. Data in Payload depends on Header element which is compulsory. There are two fields in *Header* element that must be populated: Verb and Noun. These two fields are explained below:-

Verb: This field is used to identify a specific action to be taken. There are an enumerated set of valid verbs, where commonly used values include ‘get’, ‘create’, ‘change’, ‘cancel’, ‘close’, ‘execute’ and ‘reply’. Within event notification messages ‘past tense’ verbs are used, which can include ‘created’, ‘changed’, ‘canceled’, ‘closed’ and ‘executed’.

Noun: to identify the subject of the action and/or the type of the payload, such as MeterReadings, Notification, etc.

Optional field of Header element are Revision, Context, Replaydetection, Timestamp, Source, AsyncReplyFlag, ReplyAddress, AckRequired, User, MessageID, CorrelationID, Comment, Property. Their use is explained in IEC 61968-100.

Payload element of a message depends on the combination of verb and noun used in header element.

In this specification only Verb, Noun and their corresponding payloads are explained in detail and those guidelines shall be followed in design of web services for communication between MDC and MDM.

12.2 WS-Security

As explained above WS-Security is security extension to SOAP services. Web Services used for communication between MDC and MDM shall follow username password authentication based type of WS-Security. For this purpose SOAP Header of every message shall contain "*Username*" and "*Password*" for authentication purpose. Password shall be exchanged in form of message digest (Hashed) form. Kindly note SOAP Header is different from Header of Common message envelope explained in section 12.1 of this specification.

12.3 Web Services

There are four main categories of web services that are used for data exchange between MDC and MDM. These categories are explained below.

12.3.1 Services to request Meter Readings from MDC

Only two web services: will be used for exchanging/exporting data to MDM from MDC.

1) RequestMeterReadings

2) ExecuteMeterReadings

Entity	***Web service***	***Operations***	***Messages use in operation***	***Type of message (Schema)***
MDM	RequestMeterReadings	getMeterReadings	**Output:**GetMeterReadingsRequestMessage **input:**GetMeterReadingsResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	• GetMeterReadings.xsd[1] • MeterReadings.xsd[2] • Common_message_envelope.xsd[3]
MDC	ExecuteMeterReadings	getMeterReadings	**input:**GetMeterReadingsRequestMessage **Output:**GetMeterReadingsResponseMessage	• GetMeterReadings.xsd[1] • MeterReadings.xsd[2] • Common_mess

			Fault: Fault message (request- response pattern of message exchange in WSDL)	age_envelope.xsd[3] Response and fault messages do not contain payloads.

1. GetMeterReadings.xsd is given in Annex H.15 of IEC 61968-9
2. MeterReadings.xsd is given in Annex H.23 of IEC 61968-9
3. Common_message_envelope.xsd is given in Annex A of IEC 61968-100

MDC should provide Meter Readings present in its database as per parameters in GetMeterReadingsRequestMessage.

12.3.1.1 GetMeterReadingsRequestMessage

HeaderType

Verb	get
Noun	MeterReadings
Revision	2.0
Timestamp	Validate as per request—Not to be older than 24 hours
Correlation Id	Response must contain same id as in request

RequestType

other	As per GetMeterReadings.xsd of Annex H.15 of IEC 61968-9

GetMeterReadings important fields in RequestType

Element	**Usage**
EndDevice	List of MSNs
EndDeviceGroup	List of subdivisions
Reading	Timestamp and ReadingReasonKind: billing, inquiry, loadManagement, service connect and service disconnect etc
Reading Type	Only mRID to be used and it must contain type of data “Monthly Billing data”, “Daily reads” , “Instantaneous data” and “Interval Data”
UsagePoint	List of Reference numbers
UsagePointGroup	Batch numbers
TimeSchedule	Reading Date&time interval parameter

PayloadType
Not used for GetMeterReadingsRequestMessage
.

12.3.1.2 GetMeterReadingsResponseMessage

HeaderType

Verb	reply
Noun	MeterReadings
Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

RequestType
Not used for GetMeterReadingsResponseMessage.

PayloadType

other	As per MeterReadings.xsd given in Annex H.23 of IEC 61968-9.

Fields used in MeterReadings schema.

Field	**Type**
MeterReading	MeterReading

Definition of MeterReading.

Field	**Type**
reportedDateTime	Meter reading dateTime
timestamp	dateTime (meter timestamp corresponding to data)
Meter	Meter (Contains MSN in mRID).
Readings	Reading (meter Reading structure contains more than one Reading) Reading type further contains Reading Value, timestamp, ReadingType and ReadingReasonkind as in request.

ReadingType for different data are to be used as per Annex C IEC 61968-9.

12.3.2 Services to request Meter Events from MDC

Two web services: will be used for exchanging/exporting data to MDM from MDC.

3) RequestEndDeviceEvents

4) ExecuteEndDeviceEvents

Entity	***Web service***	***Operations***	***Messages use in operation***	***Type of message (Schema)***
MDM	RequestEndDeviceEvents	getEndDeviceEvents	**Output:**GetEndDeviceEventsReques	• GetEndDeviceEvents.xsd[4] • EndDeviceEvent

			tMessage **input:**GetEndDeviceEventsResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	s.xsd[5] • Common_message_envelope.xsd
MDC	ExecuteMeterReadings	getEndDeviceEvents	**input:**GetMeterReadingsRequestMessage **Output:**GetMeterReadingsResponseMessage **Fault:** Fault message (request-response pattern of message exchange in WSDL)	• GetEndDeviceEvents.xsd[4] • EndDeviceEvents.xsd[5] • Common_message_envelope.xsd Response and fault messages do not contain payloads.

4. GetEndDeviceEvents.xsd is given in Annex H.13 of IEC 61968-9
5. EndDeviceEvents.xsd is given in Annex H.7 of IEC 61968-9

MDC should provide Meter events present in its database as per parameters in GetEndDeviceEventsRequestMessage.

12.3.2.1 GetEndDeviceEventsRequestMessage

HeaderType

Verb	get
Noun	EndDeviceEvents
Revision	2.0
Timestamp	Validate as per request—Not to be older than 24 hours
Correlation Id	Response must contain same id as in request

RequestType

other	As per GetEndDeviceEvents.xsd of Annex H.13 of IEC 61968-9

GetEndDeviceEvents important fields in RequestType

Element	**Usage**
EndDevice	List of MSNs
EndDeviceGroup	List of subdivisions
EndDeviceEventType	Only mRID to be used and it must contain name of event “Power fail End”, “Over load” as per table given in section 11.4.3 of this specification.
UsagePoint	List of Reference numbers
UsagePointGroup	Batch numbers
TimeSchedule	Reading Date&time interval parameter

PayloadType
Not used for GetEndDeviceEventsRequestMessage.

12.3.2.2 GetEndDeviceEventsResponseMessage

HeaderType

Verb	reply
Noun	EndDeviceEvents

Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

PayloadType

other	As per EndDeviceEvents.xsd given in Annex H.7 of IEC 61968-9.

Required Fields used in EndDeviceEvents schema.

Field	**Type**
EndDeviceEvent	EndDeviceEvent
EndDeviceEventType	EndDeviceEvent type for above event As per Annex E of IEC 61968-9.

Required elements and their definition for EndDeviceEvent

Field	**Type**
mRID	Name of event as per table 11.4.2
createdDateTime	DateTime of event created in meter
issuerID	MSN of meter

EndDeviceEventType for different data are to be used as per Annex E of IEC 61968-9.

12.3.3 Services for On demand relay control and MDI reset

To perform meter control on request using CIM based web services, two web services shall be made in MDC namely:-

1) ExecuteEndDeviceControls

2) SendEndDeviceEvents.

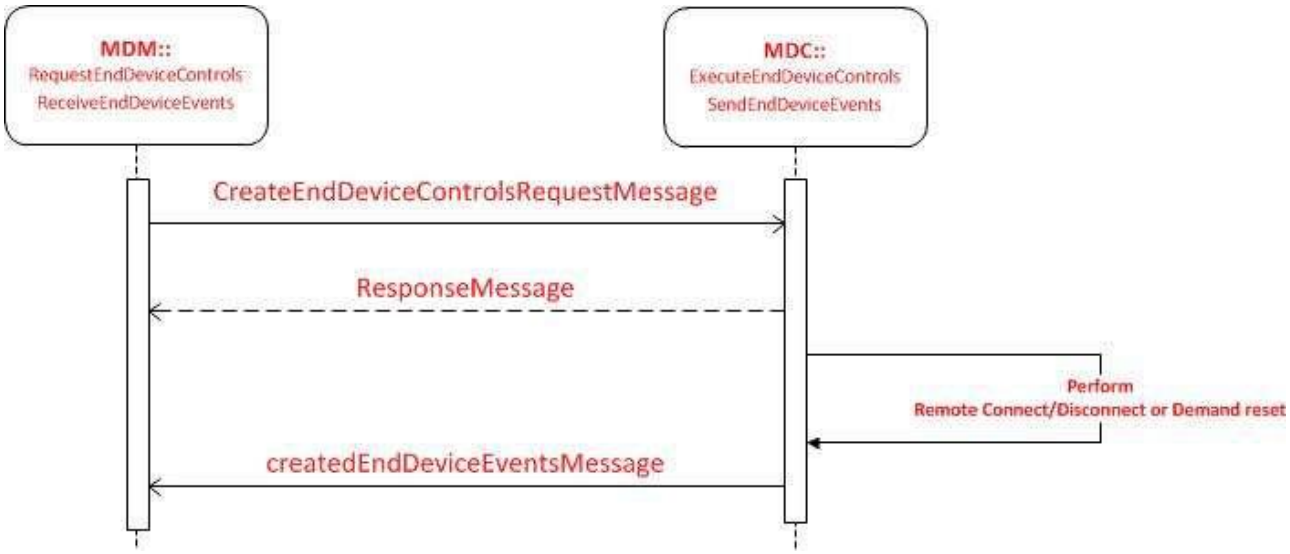
Corresponding services in MDM shall be:-

1) RequestEndDeviceControls

2) ReceiveEndDeviceEvents

Entity	*Web service*	*Operations*	*Messages use in operation*	*Type of message (Schema)*
MDM	RequestEndDeviceControls	**createEndDeviceControls**	**Output:**CreateEndDeviceControlsRequestMessage **input:**ResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	• EndDeviceControls.xsd[6] • Common_message_envelope.xsd Response and fault messages do not contain payload.
MDC	ExecuteEndDeviceControls	**createEndDeviceControls**	**Input:**CreateEndDeviceControlsRequestMessage **Output:**ResponseMessage **Fault:** Fault message (request- response pattern of message exchange in WSDL)	• EndDeviceControl.xsd[6] • Common_message_envelope.xsd Response and fault messages do not contain payloads.
MDC	SendEndDeviceEvents	**createdEndDeviceEvents**	**Output:**createdEndDeviceEventsmessage (Notification pattern of message exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd
MDM	ReceiveEndDeviceEvents	**createdEndDeviceEvents**	**Input:**createdEndDeviceEventsmessage (one way pattern exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd

6. EndDeviceControls.xsd is given in Annex H.6 of IEC 61968-9



12.3.3.1 CreateEndDeviceControlsRequestMessage

HeaderType

Verb	create
Noun	EndDeviceControls
Revision	2.0
Timestamp	Validate as per request—Not to be older than 24 hours
Correlation Id	Response must contain same id as in request

RequestType
Not used for GetEndDeviceControlsRequestMessage.

PayloadType

other	As per EndDeviceControls.xsd given in Annex H.6 of IEC 61968-9.

Required Fields used in EndDeviceControls schema.

Field	**Type**
EndDeviceControl	EndDeviceControl
EndDeviceControlType	EndDeviceControl type for above event As per Annex F of IEC 61968-9.

EndDeviceControlType for different data are to be used as per Annex F of IEC 61968-9.

Required elements and their definition for EndDeviceControl

Field	**Type**
mRID	Name of event as per table 11.4.2
reason	On demand

12.3.3.2 ResponseMessage

HeaderType

Verb	reply
Noun	EndDeviceControls
Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

ReplyType

Result	An enumeration of results:- OK, PARTIAL and FAILED

PayloadType
Not used for ResponseMessage.

12.3.3.3 createdEndDeviceEventsMessage

HeaderType

Verb	created
Noun	EndDeviceEvents
Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

RequestType
Not used for createdEndDeviceEventsMessage

PayloadType

other	As per EndDeviceEvents.xsd given in Annex H.7 of IEC 61968-9.

Required Fields used in EndDeviceEvents schema.

Field	**Type**
EndDeviceEvent	EndDeviceEvent
EndDeviceEventType	EndDeviceEvent type for above event As per Annex E of IEC 61968-9.

EndDeviceEventType for different data are to be used as per Annex E of IEC 61968-9

Required elements and their definition for EndDeviceEvent

Field	**Type**
mRID	Name of event as per table 11.4.2
createdDateTime	DateTime of event created in meter
issuerID	MSN of meter

EndDeviceControlType values for on demand Load control and Demand reset and their corresponding EndDeviceEventType are extracted from Annex E and F of IEC 61968-9 and given below.

Event or Control Description	EndDeviceControl type	Expected Events
Load Control initiate(ON→OFF)	3.15.0.54	Success :3.15.17.242 Failure: 3.15.17.217
Load Control stop (OFF→ ON)	3.15.0.55	Success: 3.15.17.216 Failure: 3.15.17.218
Demand reset	3.8.0.214	Success: 3.8.0.215 Failure: 3.8.0.65

12.3.4 Services for Master Data Management

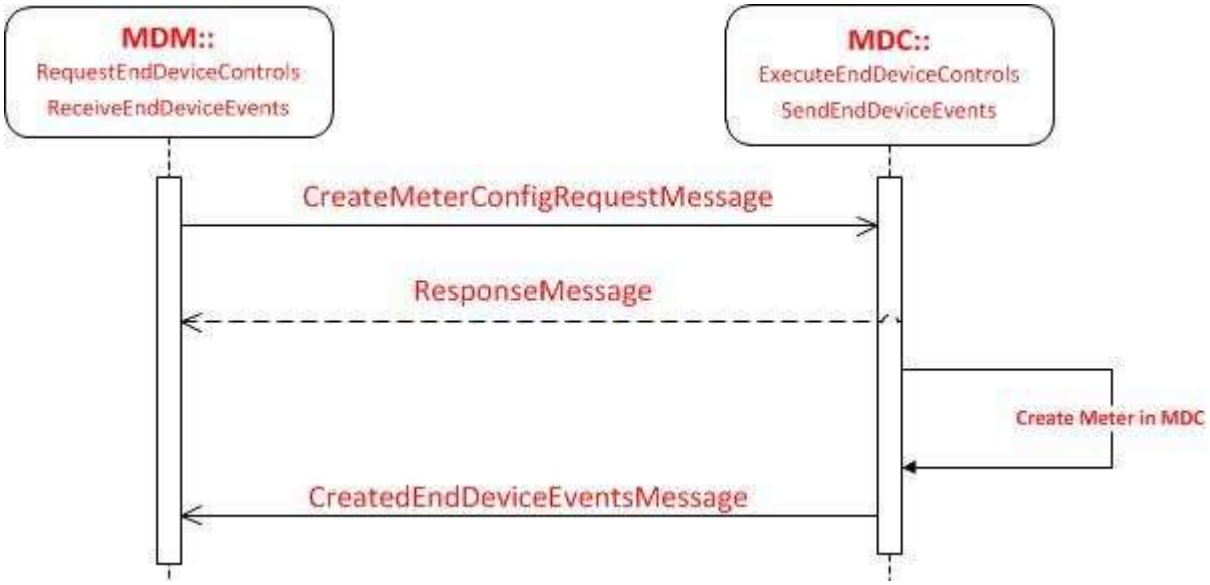
Services for Master data management as per section 5.10, Annex K and Annex L of IEC 61968-9 shall be used. Web Services in this category which are relevant to requirements in Pakistan are given below and shall be supported by MDC and MDM:

12.3.4.1 Services to create new meter in MDC

Entity	*Web service*	*Operations*	*Messages use in operation*	*Type of message (Schema)*
MDM	RequestMeterConfig	• **createMeterConfig**	**Output:**CreateMeterConfigRequestMessage **input:**ResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	• MeterConfig.xsd[7] • Common_message_envelope.xsd Response and fault messages do not contain payload.
MDC	ExecuteMeterConfig	•**createMeterConfig**	**Input:** CreateMeterConfigRequestMessage **Output:**ResponseMessage **Fault:** Fault message (request- response pattern of message exchange in WSDL)	• MeterConfig.xsd[7] • Common_message_envelope.xsd Response and fault messages do not contain payloads.
MDC	SendEndDeviceEvents	**createdEndDeviceEvents**	**Output:**createdEndDeviceEventsmessage (Notification pattern	• EndDeviceEvents.xsd • Common_message_envelope

			of message exchange in WSDL)	.xsd
MDM	ReceiveEndDeviceEvents	**createdEndDeviceEvents**	**Input:**createdEndDeviceEventsmessage (one way pattern exchange in WSDL)	• EndDeviceEvents.xsd[11] • Common_message_envelope.xsd[12]

7. MeterConfig.xsd is given in Annex H.22 of IEC 61968-9



12.3.4.1.1 CreateMeterConfigRequestMessage

HeaderType

Verb	create
Noun	MeterConfig
Revision	2.0
Timestamp	Validate as per request—Not to be older than 24 hours
Correlation Id	Response must contain same id as in request

RequestType

Not used for CreateMeterConfigRequestMessage.

PayloadType

other	As per MeterConfig.xsd given in Annex H.22 of IEC 61968-9.

Required Fields in MeterConfig schema.

Field	**Type**
Meter.mRID	Contains MSN of meter.
Meter.formnumber	Contains order number from Disco.
Meter.ConfigurationEvents	ConfigurationEvent is field in structure of Meter element Provides following information • the date and time at which the configuration or configuration change became effective or is to become effective, • the entity performing the configuration operation, and • Free-form remarks providing additional business information related to the configuration event.
Meter.EndDeviceInfo	Contains ratings of meter

12.3.4.1.2 ResponseMessage

HeaderType

Verb	reply
Noun	MeterConfig
Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

ReplyType

Result	An enumeration of results:- OK, PARTIAL and FAILED

PayloadType
Not used for ResponseMessage.

12.3.4.1.3 createdEndDeviceEventsMessage

HeaderType

Verb	created
Noun	EndDeviceEvents
Revision	2.0
Timestamp	Timestamp of system for message generation
Correlation Id	Same as in Request message

RequestType
Not used for createdEndDeviceEventsMessage

PayloadType

other	As per EndDeviceEvents.xsd given in Annex H.7 of IEC 61968-9.

Required Fields used in EndDeviceEvents schema.

Field	**Type**
EndDeviceEvent	EndDeviceEvent
EndDeviceEventType	EndDeviceEvent type for above event As per Annex E of IEC 61968-9.

EndDeviceEventType for different data are to be used as per Annex E of IEC 61968-9

Required elements and their definition for EndDeviceEvent

Field	**Type**
mRID	Name of event as per table 11.4.2
createdDateTime	DateTime of event created in meter
issuerID	MSN of meter

Event or Control Description	**Expected Events**
New meter created	Success :3.7.0.83 Failure: 3.7.0.297

12.3.4.2 Services to create new UsagePoint/Reference number in MDC

Entity	*Web service*	*Operations*	*Messages use in operation*	*Type of message (Schema)*
MDM	RequestUsagePointConfig	• **createUsagePointConfig**	**Output:**CreateUsagepointConfigRequestMessage **input:**ResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	• UsagePointConfig.xsd[8] • Common_message_envelope.xsd Response and fault messages do not contain payload.
MDC	ExecuteUsagePointConfig	•**createUsagePointConfig**	**Input:**CreateUsagepointConfigRequestMessage **Output:**ResponseMessage **Fault:** Fault message (request- response pattern of message exchange in WSDL)	• UsagePointConfig.xsd[8] • Common_message_envelope.xsd Response and fault messages do not contain payloads.
MDC	SendEndDeviceEvents	**createdEndDeviceEvents**	**Output:**createdEndDeviceEventsmessage (Notification pattern of message exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd
MDM	ReceiveEndDeviceEvents	**createdEndDeviceEvents**	**Input:**createdEndDeviceEventsmessage (one way pattern exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd

8. UsagePointConfig.xsd is given in Annex H.27 of IEC 61968-9

Rest of the detail is same as for service for creating meter only difference is of schema UsagePointConfig.xsd instead of MeterConfig.xsd.

12.3.4.3 Services to create new Customer in MDC

Entity	*Web service*	*Operations*	*Messages use in operation*	*Type of message (Schema)*
MDM	RequestCustomerConfig	• **createCustomerConfig**	**Output:** CreateCustomerConfigRequestMessage	• CustomerConfig.xsd[9] • Common_mes

			input: ResponseMessage **Fault:** Fault message (solicit-response pattern of WSDL message exchange)	sage_envelope .xsd Response and fault messages do not contain payload.
MDC	ExecuteCustomerConfig	• **createCustomerConfig** •	**Input:**CreateCustomerConfigRequestMessage **Output:**ResponseMessage **Fault:** Fault message (request- response pattern of message exchange in WSDL)	• CustomerConfig.xsd[9] • Common_message_envelope .xsd Response and fault messages do not contain payloads.
MDC	SendEndDeviceEvents	**createdEndDeviceEvents**	**Output:**createdEndDeviceEventsmessage (Notification pattern of message exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope .xsd
MDM	ReceiveEndDeviceEvents	**createdEndDeviceEvents**	**Input:**createdEndDeviceEventsmessage (one way pattern exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope .xsd

9. CustomerConfig.xsd is given in Annex H.4 of IEC 61968-9

Rest of the detail is same as for service for creating meter only difference is of schema CustomerConfig.xsd instead of MeterConfig.xsd.

12.3.4.4 Services to associate a usage point and customer in MDC

Entity	***Web service***	***Operations***	***Messages use in operation***	***Type of message (Schema)***
MDM	RequestMasterDataLinkageConfig	• **createMasterDataLinkageConfig**	**Output:** CreateMasterDataLinkageConfigRequestMessage **input:**ResponseMessage	• MasterDataLinkageConfig.xsd[10] • Common_message_envelope .xsd

			Fault: Fault message (solicit-response pattern of WSDL message exchange)	Response and fault messages do not contain payload.
MDC	ExecuteMasterDataLinkageConfig	• **createMasterDataLinkageConfig** • **deleteMasterDataLinkageConfig**	**Input:**CreateCustomerConfigRequestMessage **Output:**ResponseMessage **Fault:** Fault message (request- response pattern of message exchange in WSDL)	• MasterDataLinkageConfig.xsd [10] • Common_message_envelope.xsd Response and fault messages do not contain payloads.
MDC	SendEndDeviceEvents	**createdEndDeviceEvents**	**Output:**createdEndDeviceEventsmessage (Notification pattern of message exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd
MDM	ReceiveEndDeviceEvents	**createdEndDeviceEvents**	**Input:**createdEndDeviceEventsmessage (one way pattern exchange in WSDL)	• EndDeviceEvents.xsd • Common_message_envelope.xsd

10. MasterDataLinkageConfig.xsd is given in Annex H.21 of IEC 61968-9

Rest of the detail is same as for service for creating meter only difference is of schema MasterDataLinkageConfig.xsd instead of MeterConfig.xsd.

13. Data Export Formats:

13.1 Billing_Data.csv

Sr #	Item Name	Type (ASCII Text)	Description
1	Reading Date	YY/MM/DD	
2	Reading Time	HH:MM:SS	
3	Reference Number	XXXXXXXXXXXXX	
4	Meter Serial Number	NNNNNNNNNN	
5	T1 Active kWh	XXXXXX.XX	kWh
6	T2 Active kWh	XXXXXX.XX	kWh
7	T3 Active kWh	XXXXXX.XX	kWh

8	T4 Active kWh	XXXXXX.XX	kWh
9	TL Active kWh	XXXXXX.XX	kWh
10	T1 Reactive kVArh	XXXXXX.XX	kVArh
11	T2 Reactive kVArh	XXXXXX.XX	kVArh
12	T3 Reactive kVArh	XXXXXX.XX	kVArh
13	T4 Reactive kVArh	XXXXXX.XX	kVArh
14	TL Reactive kVArh	XXXXXX.XX	kVArh
15	T1 Active MDI	XXX.XXX	kW
16	T2 Active MDI	XXX.XXX	kW
17	T3 Active MDI	XXX.XXX	kW
18	T4 Active MDI	XXX.XXX	kW
19	TL Active MDI	XXX.XXX	kW
20	T1 Cumulative Active MDI	XXXXXX.XX	kW
21	T2 Cumulative Active MDI	XXXXXX.XX	kW
22	T3 Cumulative Active MDI	XXXXXX.XX	kW
23	T4 Cumulative Active MDI	XXXXXX.XX	kW
24	TL Cumulative Active MDI	XXXXXX.XX	kW
25	MDI Reset Date	YY/MM/DD	
26	MDI Reset Time	HH:MM:SS	
27	MDI Reset Count Number	XXXX	

Note: Unavailable quantities shall be appended by blank spaces

13.2 Instantaneous_Data.csv

Sr #	Item Name	Type (ASCII Text)	Description
1	Reading Date	YY/MM/DD	
2	Reading Time	HH:MM:SS	
3	Reference Number	XXXXXXXXXXXXX	
4	Meter Serial Number	NNNNNNNNNN	
5	Aggregate Active power Import (KW+)	XXXXXX.XXX	kW

6	Aggregate Active Power Export (KW-)	XXXXXX.XXX	kW
7	Aggregate Reactive power Import	XXXXXX.XXX	kW
8	Aggregate Reactive power export	XXXXXX.XXX	kW
9	Voltage Phase A	XXXXXX.XXX	Volts
10	Voltage Phase B	XXXXXX.XXX	Volts
11	Voltage Phase C	XXXXXX.XXX	Volts
12	Current Phase A	XXXXXX.XXX	Amperes
13	Current Phase B	XXXXXX.XXX	Amperes
14	Current Phase C	XXXXXX.XXX	Amperes
15	Average Power Factor	X.XX	
16	Frequency	XX.XX	Hz
17	Time	YY/MM/DD	
18	Date	HH:MM:SS	
19	Current Tariff	X	Number

Note: Unavailable quantities shall be appended by blank spaces

14. User Acceptance Criteria:

Following tests shall only to be performed when MDC is fully installed at utility designated location.

For energy meter prototype testing as per DDS-98 required communication verification as per Annex-II shall be performed using MDC software mockup.

14.1 Software Architecture

Criteria	Pass	Fail
Provision of User Interface Application		
Provision of Database Management System		

Provision of Communicator Application		

14.2 User Interface Application

Criteria	Pass	Fail
Capable of displaying all metering data as per relevant WAPDA/PEPCO meter's specifications i.e. DDS-50,DDS-60 and DDS-65		
Provides customer hierarchy for at least 3 levels		
Provides user management and user rights		
User rights limit a user to a certain level of customer hierarchy		
Provides a GUI		

14.3 Database Management System

Criteria	Pass	Fail
Stores and reports all metering data in DBMS		
Provides mechanism for periodic as well as on-demand backup; and also restores data on request		
Database server has a backup interface for DVD		
Capable to hold the metering data as specified in "Schedule of Technical Data"		

14.4 Hardware:

Criteria	Pass	Fail
RAID Hard Drives(if applicable)		
At least one backup server(if applicable)		
Server Class machines(if applicable)		
Online UPS(if applicable)		
Arrangement for at least two generators(if applicable)		
At least one GSM Modem.		
More than one server for load sharing(if applicable)		

14.5 Server side GSM Modem:

Criteria	Pass	Fail
Quad Band (verification through datasheet)		
SMS Capability(if applicable)		
Voice Call Capability(if applicable)		
LED Indicator		
Field Replaceable SIM		
RS232 or USB Interface		

14.6 Communicator

Criteria	Pass	Fail
Supports DLMS/COSEM communication protocol		
Supports output data files as specified in DDS-105:2011(if applicable)		

14.7 Communication Modes

Criteria	Pass	Fail
Mode I (Pull at preprogrammed Interval)		
Mode I (Pull at Events & Alarms)		
Mode I (Pull on Demand)		
Mode II(GPRS always alive)		

14.8 Communication Channels

Criteria	Pass	Fail
Provision of two Programmable IP addresses		
Provision of primary and secondary IP addresses		
Provision of three Programmable GSM Numbers		

14.9 Meter Identifier

Criteria	Pass	Fail
Supports Meter Identifier		

14.10 Meter to MDC interoperability verification

Criteria	Pass	Fail
Through CTT test report (Version of CTT shall 3.0 or higher)		

14.10 MDC to MDM interoperability verification

Criteria	Pass	Fail
Verifying the WSDL and XSD of web services on LAN		

14.12 Data Export Format

Criteria	Pass	Fail
Billing Data Export Format as per relevant specifications		
Instantaneous Data Export Format as per relevant specifications		

Engr. Atif Shahzad DY. MANAGER (D&S)	Engr. Shabir Ahmed Shaikh Addl. Chief Engineer (D&S)	Engr. Tahir Mahmood CHIEF ENGINEER (S&S)

15. Annex-I: Schedule of Technical Data:

Following data is to be filled by the customer as per their requirement

13.1. Billing Data to be transferred from meter to MDC to be filled for each meter/customer type.

S.No.	Item	Type	Required (mark Yes/No)
1	TL kWh	Billing	
2	T1 kWh	Billing	
3	T2 kWh	Billing	
4	T3 kWh	Billing	
5	T4 kWh	Billing	
6	TL kVArh	Billing	
7	T1 kVArh	Billing	
8	T2 kVArh	Billing	
9	T3 kVArh	Billing	
10	T4 kVArh	Billing	
11	TL MDI kW	Billing	
12	T1 MDI kW	Billing	
13	T2 MDI kW	Billing	
14	T3 MDI kW	Billing	
15	T4 MDI kW	Billing	
16	TL CUM MDI kW	Billing	
17	T1 CUM MDI kW	Billing	
18	T2 CUM MDI kW	Billing	
19	T3 CUM MDI kW	Billing	
20	T4 CUM MDI kW	Billing	

13.2. Instantaneous Data to be transferred from meter to MDC to be filled for each meter/customer type

S.No.	Item	Type	Required (mark Yes/No)
1	Aggregate Active power Import (KW+)	Instantaneous	
2	Aggregate Active Power Export (KW-)	Instantaneous	
3	Aggregate Reactive power Import	Instantaneous	
4	Aggregate Reactive power export	Instantaneous	
5	Voltage Phase A	Instantaneous	
6	Voltage Phase B	Instantaneous	
7	Voltage Phase C	Instantaneous	
8	Current Phase A	Instantaneous	
9	Current Phase B	Instantaneous	
10	Current Phase C	Instantaneous	
11	Average Power Factor	Instantaneous	
12	Frequency	Instantaneous	
13	Time	Instantaneous	
14	Date	Instantaneous	
15	Current Tariff	Instantaneous	
16	CT ratio Numerator	Instantaneous	
17	CT ratio Denominator	Instantaneous	
18	PT ratio Numerator	Instantaneous	
19	PT ratio Denominator	Instantaneous	
20	Customer Code	Instantaneous	

13.3. Frequency and Duration of Data to be filled for each meter/customer type used.

S.No.	Item	Count/Frequency/Duration
1	Billing Data interval	
2	Instantaneous Data interval	
3	Number of meters/customers in this category	
4	Data to be stored inside server for how many number of months	

13.4. IT Infrastructure requirement.

S.No.	IT Infrastructure	Required from bidder (Yes/No)
1	Servers	
2	Switched Router	
3	Online UPS	
4	Reliable Internet Connection	
5	Generators	
6	Data Center Room	
7	GSM Modems	
8	Tower	
9	Others(to be specified)	

13.5. Optional Features:

S.No.	Feature Names	Required from bidder (Yes/No)
1	Wake up on SMS	
2	Wake on voice call	
3	Backup server	
4	DDS-105:2011 integration required	

13.6. Service Level Agreement (SLA):

S.No.	Feature Names	
1	Meters to be integrated (specify make and model)	
2	After sales support duration	
3	Quantity to be integrated for each type and make of meters	
4		

16. Annex-II Parameters list for communication verification:

S.No.	Quantity Name
1	Meter Serial Number
2	Total kWh
3	Total kvarh