

---

## 8 Representation of primitives in data transfer

### 8.1 Introduction

This clause defines the representation of request and response primitives as XML documents, JSON texts or CBOR data format. The process of translating objects (i.e. primitives in the present context) into a format that can be stored or exchanged between network entities is commonly denoted as serialization or marshalling.

The serialization described here is used in two places:

- 1) It can be used when transmitting primitives over communication protocols such as HTTP, CoAP or MQTT. When applying a particular protocol binding, it is permitted to adapt the serialization approach, in order to make use of protocol-specific features. For example, a particular protocol binding may require that one or more primitive parameters be mapped to protocol-specific header fields rather than being included in the protocol-specific serialized JSON, XML or CBOR which represents the message body.
- 2) Certain instances of resource types, e.g. instances of the <delivery> resource, include serialized primitives embedded in one of their resource attributes.

In order to enable efficient communication, the short names introduced in clause 8.2 shall be applied in XML and JSON serializations to identify primitive parameters and resource attribute names. This implies that short names are applied in any communication over the Mca, Mcc and Mcc' reference points.

### 8.2 Short names

#### 8.2.1 Introduction

XML and JSON representations require the explicit encoding of the names of primitive parameters, resource attributes, (in the case of XML) resource types and complex data types members. Whenever a protocol binding transfers such a name over a oneM2M reference point, it shall use a shortened form of that name, rather than the full name that is used elsewhere in this and other oneM2M specifications. Short names enable payload reduction on involved telecommunication interfaces.

The mapping between the full names and their shortened form is given in the clauses 8.2.2 to 8.2.5.

#### 8.2.2 Primitive parameters

In protocol bindings primitive parameter names shall be translated into short names of Table 8.2.2-1.

**Table 8.2.2-1: Primitive parameter short names**

Parameter Name	XSD long name	Occurs in	Short Name
<b>Operation</b>	operation	Request	<b>op</b>
<b>To</b>	to	Request, Response	<b>to</b>
<b>From</b>	from	Request, Response	<b>fr</b>
<b>Request Identifier</b>	requestIdentifier	Request, Response	<b>rqi</b>
<b>Resource Type</b>	resourceType	Request	<b>ty</b>
<b>Content</b>	primitiveContent	Request, Response	<b>pc</b>
<b>Role IDs</b>	roleIDs	Request	<b>rids</b>
<b>Originating Timestamp</b>	originatingTimestamp	Request, Response	<b>ot</b>
<b>Request Expiration Timestamp</b>	requestExpirationTimestamp	Request	<b>rqet</b>
<b>Result Expiration Timestamp</b>	resultExpirationTimestamp	Request, Response	<b>rset</b>
<b>Operation Execution Time</b>	operationExecutionTime	Request	<b>oet</b>
<b>Response Type</b>	responseType	Request	<b>rt</b>
<b>Result Persistence</b>	resultPersistence	Request	<b>rp</b>
<b>Result Content</b>	resultContent	Request	<b>rcn</b>
<b>Event Category</b>	eventCategory	Request, Response	<b>ec</b>
<b>Delivery Aggregation</b>	deliveryAggregation	Request	<b>da</b>
<b>Group Request Identifier</b>	groupRequestIdentifier	Request	<b>gid</b>
<b>Filter Criteria</b>	filterCriteria	Request	<b>fc</b>
<b>Discovery Result Type</b>	discoveryResultType	Request	<b>drt</b>
<b>Response Status Code</b>	responseStatusCode	Response	<b>rsc</b>
<b>Tokens</b>	tokens	Request	<b>ts</b>
<b>Token IDs</b>	tokenIDs	Request	<b>tids</b>
<b>Token Request Indicator</b>	tokenReqIndicator	Request	<b>tqi</b>
<b>Local Token IDs</b>	localTokenIDs	Request	<b>ltids</b>
<b>Assigned Token Identifiers</b>	assignedTokenIdentifiers	Response	<b>ati</b>
<b>Token Request Information</b>	tokenReqInfo	Response	<b>tqi</b>
<b>Content Status</b>	contentStatus	Response	<b>cnst</b>
<b>Content Offset</b>	contentOffset	Response	<b>cnot</b>

XML serialized representations of primitives employ root element names to differentiate between request and response primitive types (see clause 8.3). These root element names shall be translated into short names as in Table 8.2.2-2.

**Table 8.2.2-2: Primitive root element short names**

Root Element Name	Occurs in	Short Name
<i>requestPrimitive</i>	Request	<b>reqp</b>
<i>responsePrimitive</i>	Response	<b>rsp</b>

## 8.2.3 Resource attributes

In protocol bindings, resource attributes names shall be translated into short names shown in the following tables.

**Table 8.2.3-1: Resource attribute short names (1/6)**

Attribute Name	Occurs in	Short Name
<i>accessControlPolicyIDs</i>	All except <i>accessControlPolicy</i> , <i>contentInstance</i>	<b><i>acpi</i></b>
<i>announcedAttribute</i>	<i>accessControlPolicy</i> , <i>AE</i> , <i>container</i> , <i>contentInstance</i> , <i>group</i> , <i>locationPolicy</i> , <i>mgmtObj</i> , <i>node</i> , <i>remoteCSE</i> , <i>schedule</i> , <i>semanticDescriptor</i> , <i>trafficPattern</i>	<b><i>aa</i></b>
<i>announceTo</i>	<i>accessControlPolicy</i> , <i>AE</i> , <i>container</i> , <i>contentInstance</i> , <i>group</i> , <i>locationPolicy</i> , <i>mgmtObj</i> , <i>node</i> , <i>remoteCSE</i> , <i>schedule</i> , <i>semanticDescriptor</i> , <i>trafficPattern</i>	<b><i>at</i></b>
<i>creationTime</i>	All	<b><i>ct</i></b>
<i>expirationTime</i>	All except <i>contentInstance</i> , <i>CSEBase</i>	<b><i>et</i></b>
<i>labels</i>	All (optional)	<b><i>lbl</i></b>
<i>lastModifiedTime</i>	All	<b><i>lt</i></b>
<i>Link</i>	All	<b><i>lnk</i></b>
<i>parentID</i>	All	<b><i>pi</i></b>
<i>resourceID</i>	All	<b><i>ri</i></b>
<i>resourceType</i>	All	<b><i>ty*</i></b>
<i>stateTag</i>	<i>container</i> , <i>contentInstance</i> , <i>delivery</i> , <i>request</i>	<b><i>st</i></b>
<i>resourceName</i>	All	<b><i>rn</i></b>
<i>privileges</i>	<i>accessControlPolicy</i>	<b><i>pv</i></b>
<i>selfPrivileges</i>	<i>accessControlPolicy</i>	<b><i>pvs</i></b>
<i>App-ID</i>	<i>AE</i>	<b><i>api</i></b>
<i>AE-ID</i>	<i>AE</i>	<b><i>aei</i></b>
<i>appName</i>	<i>AE</i>	<b><i>apn</i></b>
<i>pointOfAccess</i>	<i>AE</i> , <i>CSEBase</i> , <i>remoteCSE</i>	<b><i>poa</i></b>
<i>ontologyRef</i>	<i>AE</i> , <i>container</i> , <i>contentInstance</i> , <i>semanticDescriptor</i> , <i>flexContainer</i> , <i>timeSeries</i>	<b><i>or</i></b>
<i>nodeLink</i>	<i>AE</i> , <i>CSEBase</i> , <i>remoteCSE</i>	<b><i>nl</i></b>
<i>contentSerialization</i>	<i>AE</i>	<b><i>csz</i></b>
<i>creator</i>	<i>container</i> , <i>contentInstance</i> , <i>eventConfig</i> , <i>group</i> , <i>pollingChannel</i> , <i>statsCollect</i> , <i>statsConfig</i> , <i>subscription</i> , <i>semanticDescriptor</i> , <i>notificationTargetPolicy</i> , <i>flexContainer</i> , <i>timeSeries</i>	<b><i>cr</i></b>
<i>maxNrOfInstances</i>	<i>container</i> , <i>timeSeries</i>	<b><i>mni</i></b>
<i>maxByteSize</i>	<i>container</i> , <i>timeSeries</i>	<b><i>mbs</i></b>
<i>maxInstanceAge</i>	<i>container</i> , <i>timeSeries</i>	<b><i>mia</i></b>
<i>currentNrOfInstances</i>	<i>container</i> , <i>timeSeries</i>	<b><i>cni</i></b>

**Table 8.2.3-2: Resource attribute short names (2/6)**

Attribute Name	Occurs in	Short Name
<i>currentByteSize</i>	container	<i>cbs</i>
<i>locationID</i>	container	<i>li</i>
<i>disableRetrieval</i>	container	<i>disr</i>
<i>contentInfo</i>	contentInstance	<i>cnf</i>
<i>contentSize</i>	contentInstance	<i>cs</i>
<i>contentRef</i>	contentInstance	<i>conr</i>
<i>containerDefinition</i>	flexContainer	<i>cnd</i>
<i>primitiveContent</i>	request	<i>pc*</i>
<i>content</i>	contentInstance, timeSeriesInstance	<i>con</i>
<i>cseType</i>	CSEBase, remoteCSE	<i>cst</i>
<i>CSE-ID</i>	CSEBase, remoteCSE, service SubscribedNode	<i>csi</i>
<i>supportedResourceType</i>	CSEBase	<i>srt</i>
<i>notificationCongestionPolicy</i>	CSEBase	<i>ncp</i>
<i>source</i>	delivery	<i>sr</i>
<i>target</i>	delivery, request	<i>tg</i>
<i>lifespan</i>	delivery	<i>Ls</i>
<i>eventCat</i>	delivery	<i>ec</i>
<i>deliveryMetaData</i>	delivery	<i>dmd</i>
<i>aggregatedRequest</i>	delivery	<i>arq</i>
<i>eventID</i>	eventConfig, statsCollect	<i>evi</i>
<i>eventType</i>	eventConfig	<i>evt</i>
<i>evenStart</i>	eventConfig	<i>evs</i>
<i>eventEnd</i>	eventConfig	<i>eve</i>
<i>operationType</i>	eventConfig	<i>opt</i>
<i>dataSize</i>	eventConfig	<i>ds</i>
<i>execStatus</i>	execInstance	<i>exs</i>
<i>execResult</i>	execInstance	<i>exr</i>
<i>execDisable</i>	execInstance	<i>exd</i>
<i>execTarget</i>	execInstance, mgmtCmd	<i>ext</i>
<i>execMode</i>	execInstance, mgmtCmd	<i>exm</i>
<i>execFrequency</i>	execInstance, mgmtCmd	<i>exf</i>
<i>execDelay</i>	execInstance, mgmtCmd	<i>exy</i>
<i>execNumber</i>	execInstance, mgmtCmd	<i>exn</i>
<i>execReqArgs</i>	execInstance, mgmtCmd	<i>extra</i>
<i>execEnable</i>	mgmtCmd	<i>exe</i>
<i>memberType</i>	group	<i>mt</i>
<i>currentNrOfMembers</i>	group	<i>cnm</i>
<i>maxNrOfMembers</i>	group	<i>mnm</i>
<i>memberIDs</i>	group	<i>mid</i>
<i>membersAccessControlPolicyIDs</i>	group	<i>macp</i>
<i>memberTypeValidated</i>	group	<i>mtv</i>
<i>consistencyStrategy</i>	group	<i>csy</i>
<i>groupName</i>	group, subscription	<i>gn</i>
<i>locationSource</i>	locationPolicy	<i>los</i>
<i>locationUpdatePeriod</i>	locationPolicy	<i>lou</i>
<i>locationTargetID</i>	locationPolicy	<i>lot</i>
<i>locationServer</i>	locationPolicy	<i>lor</i>
<i>locationContainerID</i>	locationPolicy	<i>loi</i>
<i>locationContainerName</i>	locationPolicy	<i>lon</i>
<i>locationStatus</i>	locationPolicy	<i>lost</i>
<i>serviceRoles</i>	m2mServiceSubscriptionProfile	<i>svr</i>
<i>description</i>	mgmtCmd, mgmtObj, all management resources from firmware	<i>dc</i>
<i>cmdType</i>	mgmtCmd	<i>cmt</i>
<i>mgmtDefinition</i>	mgmtObj, all management resources from firmware	<i>mgd</i>
<i>objectIDs</i>	mgmtObj	<i>obis</i>

**Table 8.2.3-3: Resource attribute short names (3/6)**

Attribute Name	Occurs in	Short Name
<i>objectPaths</i>	mgmtObj	<i>obps</i>
<i>nodeID</i>	node	<i>ni</i>
<i>hostedCSELink</i>	node	<i>hcl</i>
<i>CSEBase</i>	remoteCSE	<i>cb*</i>
<i>M2M-Ext-ID</i>	remoteCSE	<i>mei</i>
<i>Trigger-Recipient-ID</i>	remoteCSE	<i>tri</i>
<i>requestReachability</i>	remoteCSE	<i>rr</i>
<i>triggerReferenceNumber</i>	remoteCSE	<i>trn</i>
<i>originator</i>	request	<i>org</i>
<i>metaInformation</i>	request	<i>mi</i>
<i>requestStatus</i>	request	<i>rs</i>
<i>operationResult</i>	request	<i>ors</i>
<i>operation</i>	request	<i>op*</i>
<i>requestID</i>	request	<i>rid</i>
<i>scheduleElement</i>	schedule	<i>se</i>
<i>deviceIdentifier</i>	serviceSubscribedNode	<i>di</i>
<i>ruleLinks</i>	serviceSubscribedNode	<i>rlk</i>
<i>statsCollectID</i>	statsCollect	<i>sci</i>
<i>collectingEntityID</i>	statsCollect	<i>cei</i>
<i>collectedEntityID</i>	statsCollect	<i>cdi</i>
<i>devStatus</i>	areaNwkDeviceInfo	<i>ss</i>
<i>statsRuleStatus</i>	statsCollect	<i>srs</i>
<i>statModel</i>	statsCollect	<i>sm</i>
<i>collectPeriod</i>	statsCollect	<i>cp</i>
<i>eventNotificationCriteria</i>	subscription	<i>enc</i>
<i>expirationCounter</i>	subscription	<i>exc</i>
<i>notificationURI</i>	subscription	<i>nu</i>
<i>groupID</i>	subscription	<i>gpi</i>
<i>notificationForwardingURI</i>	subscription	<i>nfu</i>
<i>batchNotify</i>	subscription	<i>bn</i>
<i>rateLimit</i>	subscription	<i>rl</i>
<i>preSubscriptionNotify</i>	subscription	<i>psn</i>
<i>pendingNotification</i>	subscription	<i>pn</i>
<i>notificationStoragePriority</i>	subscription	<i>nsp</i>
<i>latestNotify</i>	subscription	<i>ln</i>
<i>notificationContentType</i>	subscription	<i>nct</i>
<i>notificationEventCat</i>	subscription	<i>nec</i>
<i>subscriberURI</i>	subscription	<i>su</i>
<i>version</i>	firmware, software, token	<i>vr</i>
<i>URL</i>	firmware, software	<i>url</i>
<i>update</i>	firmware	<i>ud</i>
<i>updateStatus</i>	firmware	<i>uds</i>
<i>install</i>	software	<i>in</i>
<i>uninstall</i>	software	<i>un</i>
<i>installStatus</i>	software	<i>ins</i>
<i>activate</i>	software	<i>act</i>
<i>deactivate</i>	software	<i>dea</i>
<i>activeStatus</i>	software, areaNwkInfo	<i>acts</i>
<i>memAvailable</i>	memory	<i>mma</i>
<i>memTotal</i>	memory	<i>mmt</i>

**Table 8.2.3-4: Resource attribute short names (4/6)**

Attribute Name	Occurs in	Short Name
<i>areaNwkType</i>	areaNwkInfo	<i>ant</i>
<i>listOfDevices</i>	areaNwkInfo	<i>ldv</i>
<i>devId</i>	areaNwkDeviceInfo	<i>dvd</i>
<i>devType</i>	areaNwkDeviceInfo	<i>dvt</i>
<i>areaNwkId</i>	areaNwkDeviceInfo	<i>awi</i>
<i>sleepInterval</i>	areaNwkDeviceInfo	<i>sli</i>
<i>sleepDuration</i>	areaNwkDeviceInfo	<i>sld</i>
<i>listOfNeighbors</i>	areaNwkDeviceInfo	<i>lnh</i>
<i>batteryLevel</i>	battery	<i>btl</i>
<i>batteryStatus</i>	battery	<i>bts</i>
<i>deviceLabel</i>	deviceInfo	<i>dlb</i>
<i>manufacturer</i>	deviceInfo	<i>man</i>
<i>model</i>	deviceInfo	<i>mod</i>
<i>deviceType</i>	deviceInfo	<i>dtv</i>
<i>fwVersion</i>	deviceInfo	<i>fwv</i>
<i>swVersion</i>	deviceInfo	<i>swv</i>
<i>hwVersion</i>	deviceInfo	<i>hwv</i>
<i>capabilityName</i>	deviceCapability	<i>can</i>
<i>attached</i>	deviceCapability	<i>att</i>
<i>capabilityActionStatus</i>	deviceCapability	<i>cas</i>
<i>enable</i>	deviceCapability, allJoynSvcObject	<i>ena</i>
<i>disable</i>	deviceCapability	<i>dis</i>
<i>currentState</i>	deviceCapability	<i>cus</i>
<i>reboot</i>	reboot	<i>rbo</i>
<i>factoryReset</i>	reboot	<i>far</i>
<i>logTypeId</i>	eventLog	<i>lgt</i>
<i>logData</i>	eventLog	<i>lgd</i>
<i>logStatus</i>	eventLog	<i>lgst</i>
<i>logStart</i>	eventLog	<i>lga</i>
<i>logStop</i>	eventLog	<i>lgo</i>
<i>firmwareName</i>	firmware	<i>fwnnam</i>
<i>softwareName</i>	software	<i>swn</i>
<i>cmdhPolicyName</i>	cmdhPolicy	<i>cpn</i>
<i>mgmtLink</i>	cmdhPolicy, activeCmdhPolicy, cmdhDefaults, cmdhNetworkAccessRules, cmdhNwAccessRule	<i>cmlk</i>
<i>activeCmdhPolicyLink</i>	activeCmdhPolicy	<i>acmlk</i>
<i>order</i>	cmdhDefEcValue, cmdhLimits	<i>od</i>
<i>defEcValue</i>	cmdhDefEcValue	<i>dev</i>
<i>requestOrigin</i>	cmdhDefEcValue, cmdhLimits	<i>ror</i>
<i>requestContext</i>	cmdhDefEcValue, cmdhLimits	<i>rct</i>
<i>requestContextNotification</i>	cmdhDefEcValue, cmdhLimits	<i>rctn</i>
<i>requestCharacteristics</i>	cmdhDefEcValue, cmdhLimits	<i>rch</i>
<i>applicableEventCategories</i>	cmdhNetworkAccessRules	<i>aecs</i>
<i>applicableEventCategory</i>	cmdhEcDefParamValues, cmdhBuffer	<i>aec</i>
<i>defaultRequestExpTime</i>	cmdhEcDefParamValues	<i>dqet</i>
<i>defaultResultExpTime</i>	cmdhEcDefParamValues	<i>dset</i>
<i>defaultOpExecTime</i>	cmdhEcDefParamValues	<i>doet</i>
<i>defaultRespPersistence</i>	cmdhEcDefParamValues	<i>drp</i>
<i>defaultDelAggregation</i>	cmdhEcDefParamValues	<i>dda</i>
<i>limitsEventCategory</i>	cmdhLimits	<i>lec</i>
<i>limitsRequestExpTime</i>	cmdhLimits	<i>lqet</i>
<i>limitsResultExpTime</i>	cmdhLimits	<i>lset</i>
<i>limitsOpExecTime</i>	cmdhLimits	<i>loet</i>
<i>limitsRespPersistence</i>	cmdhLimits	<i>lrp</i>
<i>limitsDelAggregation</i>	cmdhLimits	<i>lda</i>
<i>targetNetwork</i>	cmdhNwAccessRule, trafficPattern	<i>ttn</i>

**Table 8.2.3-5: Resource attribute short names (5/6)**

Attribute Name	Occurs in	Short Name
<i>minReqVolume</i>	cmdhNwAccessRule	<i>mrw</i>
<i>spreadingWaitTime</i>	cmdhNwAccessRule	<i>swt</i>
<i>backOffParameters</i>	cmdhNwAccessRule	<i>bop</i>
<i>otherConditions</i>	cmdhNwAccessRule	<i>ohc</i>
<i>maxBufferSize</i>	cmdhBuffer	<i>mbfs</i>
<i>storagePriority</i>	cmdhBuffer	<i>sgp</i>
<i>applicableCredIDs</i>	serviceSubscribedAppRule	<i>apci</i>
<i>allowedApp-IDs</i>	serviceSubscribedAppRule	<i>aai</i>
<i>allowedAEs</i>	serviceSubscribedAppRule	<i>aae</i>
<i>notificationTargetURI</i>	notificationTargetMgmtPolicyRef	<i>ntu</i>
<i>notificationPolicyID</i>	notificationTargetMgmtPolicyRef	<i>npi</i>
<i>action</i>	notificationTargetPolicy	<i>ac</i>
<i>policyLabel</i>	notificationTargetPolicy	<i>plbl</i>
<i>rulesRelationship</i>	notificationTargetPolicy	<i>rrs</i>
<i>creator</i>	notificationTargetPolicy	<i>cr</i>
<i>deletionRules</i>	policyDeletionRules	<i>dr</i>
<i>deletionRulesRelation</i>	policyDeletionRules	<i>drr</i>
<i>dynamicAuthorizationConsultationIDs</i>	All resources having an accessControlPolicyID attribute	<i>daci</i>
<i>dynamicAuthorizationEnabled</i>	dynamicAuthorizationConsultation	<i>dae</i>
<i>dynamicAuthorizationPoA</i>	dynamicAuthorizationConsultation	<i>dap</i>
<i>dynamicAuthorizationLifetime</i>	dynamicAuthorizationConsultation	<i>dal</i>
<i>descriptorRepresentation</i>	semanticDescriptor	<i>dcrp</i>
<i>semanticOpExec</i>	semanticDescriptor	<i>soe</i>
<i>descriptor</i>	semanticDescriptor	<i>dsp</i>
<i>relatedSemantics</i>	semanticDescriptor	<i>rels</i>
<i>periodicInterval</i>	timeSeries	<i>pei</i>
<i>missingDataDetect</i>	timeSeries	<i>mdd</i>
<i>missingDataMaxNr</i>	timeSeries	<i>mdn</i>
<i>missingDataList</i>	timeSeries	<i>mdlt</i>
<i>missingDataCurrentNr</i>	timeSeries	<i>mdc</i>
<i>missingDataDetectTimer</i>	timeSeries	<i>mdt</i>
<i>dataGenerationTime</i>	timeSeriesInstance	<i>dgt</i>
<i>sequenceNr</i>	timeSeriesInstance	<i>snr</i>
<i>providedToNSE</i>	trafficPattern	<i>ptn</i>
<i>periodicIndicator</i>	trafficPattern	<i>pri</i>
<i>periodicDurationTime</i>	trafficPattern	<i>pdT</i>
<i>periodicIntervalTime</i>	trafficPattern	<i>pit</i>
<i>stationaryIndication</i>	trafficPattern	<i>sti</i>
<i>dataSizeIndicator</i>	trafficPattern	<i>dsi</i>
<i>validityTime</i>	trafficPattern	<i>vdT</i>
<i>roleID</i>	role	<i>rlid</i>
<i>roleName</i>	role	<i>rlnm</i>
<i>tokenLink</i>	role	<i>rltl</i>
<i>tokenID</i>	token	<i>tkid</i>
<i>tokenObject</i>	token	<i>tkob</i>
<i>issuer</i>	token, role	<i>tkis</i>
<i>holder</i>	token, role	<i>tkhd</i>
<i>notBefore</i>	token, role	<i>tknb</i>
<i>notAfter</i>	token, role	<i>tkna</i>
<i>tokenName</i>	token	<i>tknm</i>
<i>audience</i>	token	<i>tkau</i>
<i>permissions</i>	token	<i>tkps</i>
<i>extension</i>	token	<i>tkex</i>
<i>e2eSecInfo</i>	CSEBase, remoteCSE, AE	<i>esi</i>

**Table 8.2.3-6: Resource attribute short names (6/6)**

<b>Attribute Name</b>	<b>Occurs in</b>	<b>Short Name</b>
<i>serviceName</i>	genericInterworkingService	<b><i>gisn</i></b>
<i>operationName</i>	genericInterworkingOperationInstance	<b><i>gion</i></b>
<i>inputDataPointLinks</i>	genericInterworkingService, genericInterworkingOperationInstance	<b><i>giip</i></b>
<i>outputDataPointLinks</i>	genericInterworkingService, genericInterworkingOperationInstance	<b><i>giop</i></b>
<i>inputLinks</i>	genericInterworkingOperationInstance	<b><i>giil</i></b>
<i>outputLinks</i>	genericInterworkingOperationInstance	<b><i>giol</i></b>
<i>operationState</i>	genericInterworkingOperationInstance	<b><i>gios</i></b>
<i>direction</i>	allJoynApp	<b><i>dir</i></b>
<i>objectPath</i>	allJoynSvcObject	<b><i>ajop</i></b>
<i>interfaceIntrospectXmlRef</i>	allJoynInterface	<b><i>ajir</i></b>
<i>input</i>	allJoynMethodCall	<b><i>inp</i></b>
<i>callStatus</i>	allJoynMethodCall	<b><i>clst</i></b>
<i>output</i>	allJoynMethodCall	<b><i>out</i></b>
<i>currentValue</i>	allJoynProperty	<b><i>crv</i></b>
<i>requestedValue</i>	allJoynProperty	<b><i>rqv</i></b>
NOTE: * marked short names have been already assigned in Table 8.2.2-1.		



## 8.2.4 Resource types

In protocol bindings resource type names shall be translated into short names of Table 8.2.4-1.

**Table 8.2.4-1: Resource and specialization type short names**

<b>Resource Type Name</b>	<b>Short Name</b>
accessControlPolicy	<b>acp</b>
accessControlPolicyAnnc	<b>acpA</b>
AE	<b>ae</b>
AEAnnc	<b>aeA</b>
container	<b>cnt</b>
containerAnnc	<b>cntA</b>
latest	<b>la</b>
oldest	<b>ol</b>
contentInstance	<b>cin</b>
contentInstanceAnnc	<b>cinA</b>
CSEBase	<b>cb</b>
delivery	<b>dlv</b>
eventConfig	<b>evcg</b>
execInstance	<b>exin</b>
fanOutPoint	<b>fopt</b>
group	<b>grp</b>
groupAnnc	<b>grpA</b>
locationPolicy	<b>lcp</b>
locationPolicyAnnc	<b>lcpA</b>
m2mServiceSubscriptionProfile	<b>mssp</b>
mgmtCmd	<b>mgc</b>
node	<b>nod</b>
nodeAnnc	<b>nodA</b>
pollingChannel	<b>pch</b>
pollingChannelURI	<b>pcu</b>
remoteCSE	<b>csr</b>
remoteCSEAnnc	<b>csrA</b>
request	<b>req</b>
schedule	<b>sch</b>
scheduleAnnc	<b>schA</b>
serviceSubscribedAppRule	<b>asar</b>
serviceSubscribedNode	<b>svsn</b>
statsCollect	<b>stcl</b>
statsConfig	<b>stcg</b>
subscription	<b>sub</b>
firmware	<b>fwr</b>
firmwareAnnc	<b>fwrA</b>
software	<b>swr</b>
softwareAnnc	<b>swrA</b>
memory	<b>mem</b>
memoryAnnc	<b>memA</b>
areaNwkInfo	<b>ani</b>
areaNwkInfoAnnc	<b>aniA</b>
areaNwkDeviceInfo	<b>andi</b>
areaNwkDeviceInfoAnnc	<b>andiA</b>
battery	<b>bat</b>
batteryAnnc	<b>batA</b>
deviceInfo	<b>dvi</b>
deviceInfoAnnc	<b>dviA</b>
deviceCapability	<b>dvc</b>
deviceCapabilityAnnc	<b>dvcA</b>
reboot	<b>rbo *</b>
rebootAnnc	<b>rboA</b>
eventLog	<b>evl</b>
eventLogAnnc	<b>evlA</b>
cmdhPolicy	<b>cmp</b>
activeCmdhPolicy	<b>acmp</b>
cmdhDefaults	<b>cmdf</b>
cmdhDefEcValue	<b>cmdv</b>
cmdhEcDefParamValues	<b>cmpv</b>
cmdhLimits	<b>cml</b>
cmdhNetworkAccessRules	<b>cmnr</b>

<b>Resource Type Name</b>	<b>Short Name</b>
cmdhNwAccessRule	<b>cmwr</b>
cmdhBuffer	<b>cmbf</b>
notificationTargetMgmtPolicyRef	<b>ntpr</b>
notificationTargetPolicy	<b>ntp</b>
policyDeletionRules	<b>pdr</b>
notificationTargetSelfReference	<b>ntsr</b>
dynamicAuthorizationConsultation	<b>dac</b>
semanticDescriptor	<b>smd</b>
semanticDescriptorAnnc	<b>smdA</b>
semanticFanOutPoint	<b>sfop</b>
timeSeries	<b>ts</b>
timeSeriesAnnc	<b>tSA</b>
timeSeriesInstance	<b>tsi</b>
timeSeriesInstanceAnnc	<b>tsia</b>
trafficPattern	<b>trpt</b>
trafficPatternAnnc	<b>trptA</b>
role	<b>rol</b>
token	<b>tk</b>
genericInterworkingService	<b>gis</b>
genericInterworkingServiceAnnc	<b>gisa</b>
genericInterworkingOperationInstance	<b>gio</b>
genericInterworkingOperationInstanceAnnc	<b>gioa</b>
svcObjWrapper	<b>ajsw</b>
svcObjWrapperAnnc	<b>ajswa</b>
svcFwWrapper	<b>ajfw</b>
svcFwWrapperAnnc	<b>ajfwa</b>
allJoynApp	<b>ajap</b>
allJoynAppAnnc	<b>ajapa</b>
allJoynSvcObject	<b>ajso</b>
allJoynSvcObjectAnnc	<b>ajsoa</b>
allJoynInterface	<b>ajif</b>
allJoynInterfaceAnnc	<b>ajifa</b>
allJoynMethod	<b>ajmd</b>
allJoynMethodAnnc	<b>ajmda</b>
allJoynMethodCall	<b>ajmc</b>
allJoynMethodCallAnnc	<b>ajmca</b>
allJoynProperty	<b>ajpr</b>
allJoynPropertyAnnc	<b>ajpra</b>
NOTE: * marked short names have been already assigned in attribute Tables 8.2.3-1 to 8.2.3-5.	

## 8.2.5 Complex data types members

In protocol bindings complex data types member names shall be translated into short names of Table 8.2.5-1.

**Table 8.2.5-1: Complex data type member short names**

Member Name	Occurs in	Short Name
createdBefore	filterCriteria, eventNotificationCriteria	<b>crb</b>
createdAfter	filterCriteria, eventNotificationCriteria	<b>cra</b>
modifiedSince	filterCriteria, eventNotificationCriteria	<b>ms</b>
unmodifiedSince	filterCriteria, eventNotificationCriteria	<b>us</b>
stateTagSmaller	filterCriteria, eventNotificationCriteria	<b>sts</b>
stateTagBigger	filterCriteria, eventNotificationCriteria	<b>stb</b>
expireBefore	filterCriteria, eventNotificationCriteria	<b>exb</b>
expireAfter	filterCriteria, eventNotificationCriteria	<b>exa</b>
labels	filterCriteria, eventNotificationCriteria	<b>lbl *</b>
resourceType	filterCriteria	<b>ty *</b>
sizeAbove	filterCriteria, eventNotificationCriteria	<b>sza</b>
sizeBelow	filterCriteria, eventNotificationCriteria	<b>szb</b>
contentType	filterCriteria	<b>cty</b>
limit	filterCriteria	<b>lim</b>
attribute	filterCriteria, eventNotificationCriteria	<b>atr</b>
contentFilterSyntax	filterCriteria	<b>cfs</b>
contentFilterQuery	filterCriteria	<b>cfq</b>
level	filterCriteria	<b>lvl</b>
offset	filterCriteria	<b>ofst</b>
notificationEventType	eventNotificationCriteria	<b>net</b>
operationMonitor	eventNotificationCriteria, notificationEvent	<b>om</b>
representation	notificationEvent	<b>rep</b>
filterUsage	filterCriteria	<b>fu</b>
eventCatType	eventCat	<b>ect</b>
eventCatNo	eventCat	<b>ecn</b>
number	batchNotify	<b>num</b>
duration	batchNotify	<b>dur</b>
notification	aggregatedNotification, Request Primitive Content	<b>sgn</b>
notificationEvent	notification	<b>nev</b>
verificationRequest	notification	<b>vrq</b>
subscriptionDeletion	notification	<b>sud</b>
subscriptionReference	notification	<b>sur</b>
creator	notification	<b>cr*</b>
notificationForwardingURI	notification	<b>nfu*</b>
IPEDiscoveryRequest	notification	<b>idr</b>
filterCriteria	IPEDiscoveryRequest	<b>fc*</b>
operation	operationMonitor, dynAuthDasRequest	<b>op*</b>
originator	operationMonitor, IPEDiscoveryRequest, dynAuthDasRequest	<b>or*</b>
accessId	externalID	<b>aci</b>
MSISDN	externalID	<b>msd</b>
action	actionStatus	<b>can</b>
status	actionStatus	<b>sus</b>
childResource	All except execInstance, announced resource, management resources from firmware	<b>ch</b>
accessControlRule	privileges, selfPrivileges	<b>acr</b>
accessControlOriginators	accessControlRule	<b>acor</b>
accessControlOperations	accessControlRule	<b>acop</b>
accessControlContexts	accessControlRule	<b>acco</b>
accessControlWindow	accessControlContexts	<b>actw</b>
accessControlIpAddresses	accessControlContexts	<b>acip</b>
ipv4Addresses	accessControlIpAddress	<b>ipv4</b>
ipv6Addresses	accessControlIpAddress	<b>ipv6</b>
accessControlLocationRegion	accessControlContexts	<b>aclr</b>
countryCode	accessControlLocationRegion	<b>acc</b>
circRegion	accessControlLocationRegion	<b>accr</b>
name	attribute, anyArgType, mgmtLinkRef, childResourceRef, contentRef	<b>nm*</b>
specializationID	childResourceRef	<b>spid</b>
value	attribute	<b>val</b>
type	anyArgType	<b>typ</b>

Member Name	Occurs in	Short Name
maxNrOfNotify	rateLimit	<i>mn</i>
timeWindow	rateLimit	<i>tw</i>
scheduleEntry	scheduleElement	<i>sce</i>
aggregatedNotification	Request Primitive Content	<i>agn</i>
attributeList	Request Primitive Content	<i>atrl</i>
securityInfo	Request Primitive Content, Response Primitive Content	<i>seci</i>
aggregatedResponse	Response Primitive Content	<i>agr</i>
resource	Response Primitive Content	<i>rce</i>
URIList	Response Primitive Content	<i>uril</i>
debugInfo	Response Primitive Content	<i>dbg</i>
anyArg	resetArgsType, rebootArgsType, uploadArgsType, downloadArgsType, softwareInstallArgsType, softwareUpdateArgsType, softwareUninstallArgsType, execReqArgsListType	<i>any</i>
fileType	downloadArgsType	<i>ftyp</i>
URI	resourceWrapper, dynAuthTokenReqInfo	<i>uri</i>
URL	downloadArgsType	<i>url*</i>
username	uploadArgsType, downloadArgsType, softwareUpdateArgsType, softwareUninstallArgsType,	<i>unm</i>
password	uploadArgsType, downloadArgsType, softwareUpdateArgsType, softwareUninstallArgsType,	<i>pwd</i>
filesize	downloadArgsType	<i>fsi</i>
targetFile	downloadArgsType	<i>tgf</i>
delaySeconds	downloadArgsType	<i>dss</i>
successURL	downloadArgsType	<i>surl</i>
startTime	downloadArgsType	<i>stt</i>
completeTime	downloadArgsType	<i>cpt</i>
UUID	softwareInstallArgsType, softwareUpdateArgsType, softwareUninstallArgsType,	<i>uuid</i>
executionEnvRef	softwareInstallArgsType, softwareUpdateArgsType, softwareUninstallArgsType,	<i>eer</i>
version	softwareUninstallArgsType, tokenClaimSet	<i>vr*</i>
reset	execReqArgsListType	<i>rst</i>
reboot	execReqArgsListType	<i>rbo*</i>
upload	execReqArgsListType	<i>uld</i>
download	execReqArgsListType	<i>dld</i>
softwareInstall	execReqArgsListType	<i>swin</i>
softwareUpdate	execReqArgsListType	<i>swup</i>
softwareUninstall	execReqArgsListType	<i>swun</i>
tracingOption	deliveryMetaData	<i>tcop</i>
tracingInfo	deliveryMetaData	<i>tcin</i>
responseTypeValue	responseTypeInfo	<i>rtv</i>
notificationURI	responseTypeInfo	<i>nu</i>
timeOfDay	deletionContexts	<i>tod</i>
locationRegions	deletionContexts	<i>lr</i>
URIReference	contentRef	<i>urir</i>
semanticsFilter	filterCriteria	<i>smf</i>
missingDataList	timeSeries	<i>mdl</i>
missingData	eventNotificationCriteria	<i>md</i>
tokenId	tokenClaimSet, dynAuthLocalTokenIdAssignments	<i>tkid</i>
holder	tokenClaimSet	<i>tkhd*</i>
issuer	tokenClaimSet	<i>tkis*</i>
notBefore	tokenClaimSet	<i>tknb*</i>
notAfter	tokenClaimSet	<i>tkna*</i>
tokenName	tokenClaimSet	<i>tknm*</i>
audience	tokenClaimSet	<i>tkau*</i>

Member Name	Occurs in	Short Name
permissions	tokenClaimSet	<i>tkps*</i>
extension	tokenClaimSet	<i>tkex*</i>
permission	tokenPermissions	<i>pm</i>
resourceIDs	tokenPermission	<i>ris</i>
privileges	tokenPermission	<i>pv*</i>
roleIDs	tokenPermission	<i>rids*</i>
localTokenIdAssignment	dynAuthLocalTokenIdAssignments	<i>ltia</i>
localTokenID	dynAuthLocalTokenIdAssignment	<i>lti</i>
dasInfo	dynAuthTokenReqInfo	<i>dasi</i>
dasRequest	dynAuthTokenReqInfo	<i>daq</i>
securedDasRequest	dynAuthTokenReqInfo	<i>sdr</i>
filterOperation	filterCriteria	<i>fo</i>
targetedResourceType	dynAuthDasRequest	<i>trt</i>
originatorIP	dynAuthDasRequest	<i>oip</i>
ipv4Address	dynAuthDasRequest	<i>ip4</i>
ipv6Address	dynAuthDasRequest	<i>ip6</i>
originatorLocation	dynAuthDasRequest	<i>olo</i>
originatorRoleIDs	dynAuthDasRequest	<i>orid</i>
requestTimestamp	dynAuthDasRequest	<i>rts</i>
targetedResourceID	dynAuthDasRequest	<i>trid</i>
proposedPrivilegesLifetime	dynAuthDasRequest	<i>ppl</i>
roleIDsFromACPs	dynAuthDasRequest	<i>rfa</i>
tokenIDs	dynAuthDasRequest	<i>tids</i>
dynamicACPIInfo	dynAuthDasResponse	<i>dai</i>
grantedPrivileges	dynAuthDasResponse	<i>gp</i>
privilegesLifetime	dynAuthDasResponse	<i>pl</i>
tokens	dynAuthDasResponse	<i>tkns</i>
securityInfoType	securityInfo	<i>sit</i>
dasRequest	securityInfo	<i>dreq</i>
dasResponse	securityInfo	<i>dres</i>
esprimRandObject	securityInfo	<i>ero</i>
esprimObject	securityInfo	<i>epo</i>
escertkeMessage	securityInfo	<i>eckm</i>
resourceRef	listOfChildResourceRef	<i>rrf</i>
resourceRefList	Response Primitive Content	<i>rri</i>
esprimRandID	originatorESPrimRandObject, receiverESPrimRandObject,	<i>esri</i>
esprimRandValue	originatorESPrimRandObject, receiverESPrimRandObject,	<i>esrv</i>
esprimRandExpiry	originatorESPrimRandObject, receiverESPrimRandObject,	<i>esrx</i>
esprimKeyGenAlgID	originatorESPrimRandObject,	<i>esk</i>
esprimKeyGenAlgIDs	receiverESPrimRandObject,	<i>esks</i>
esprimProtocolAndAlgIDs	originatorESPrimRandObject, receiverESPrimRandObject,	<i>espa</i>
supportede2ESecFeatures	e2eSecInfo	<i>esf</i>
certificates	e2eSecInfo	<i>escert</i>
sharedReceiverESPrimRandObject	e2eSecInfo	<i>esro</i>
networkAction	backOffParameters	<i>nwa</i>
initialBackoffTime	backOffParameters	<i>ibt</i>
additionalBackoffTime	backOffParameters	<i>abt</i>
maximumBackoffTime	backOffParameters	<i>mbt</i>
optionalRandomBackoffTime	backOffParameters	<i>rbt</i>
backOffParametersSet	backOffParameters	<i>bops</i>
dataLink	listOfDataLinks	<i>dali</i>
attributeName	dataLink	<i>atn</i>
dataContainerID	dataLink	<i>dcid</i>
accessControlAuthenticationFlag	accessControlRule	<i>acaf</i>
dataLinkEntry	listOfDataLinks	<i>dle</i>
NOTE: * marked short names have been already assigned in attribute Table 8.2.3-1.		

## 8.3 XML serialization

### 8.3.1 Method

XML serialization of request or response primitives refers to the process of representing the primitive as an XML document.

The XML document shall be a well-formed XML document compliant with W3C XML 1.0 [1]. It shall be restricted to Unicode characters and encoded using UTF-8 as described in RFC 3629 [21].

The structure and data types of XML serialized request and response primitives shall be consistent with the XSD defined in CDT-requestPrimitive-v2\_7\_0.xsd and CDT-responsePrimitive-v2\_7\_0.xsd, respectively. The data types used in these XSD files comply with the definitions in clause 6 and clause 7 of the present document.

XML serializations shall comply with the order of resource attributes and elements imposed by the XML schema definition. If an implementation uses modified XSD modified from the original files for schema validation of partial resource representations (see NOTE 2 in clause 6.1), the order of resource attributes shall not be changed.

Note that the XSD files included in the present release employ the long names for primitive parameters and other XML elements and attributes, but the primitive serialization is required to use the corresponding short names (as defined clause 8.2 of the present document).

NOTE: XML Schema files are available with both long and short names.

The primitive **Content** parameter is serialized just like any other element of complex type. Generally, the **Content** parameter may include only a partial set of attributes specified for the resource type as indicated in the **Resource Type** parameter, e.g. for partial Update or Retrieve Request procedures. For Notification Request primitives, the **Content** parameter includes a Notification data object as defined in clause 7.5.1.1 and the datatype definition given in CDT-notification-v2\_7\_0.xsd.

### 8.3.2 Examples

An example that shows a request primitive serialized into an XML document is shown below. This example shows the create request for an instance of a <contentInstance> resource. Only mandatory primitive parameters and resource attributes are shown.

```
<?xml version="1.0" encoding="UTF-8"?>
<m2m:rqp xmlns:m2m="http://www.onem2m.org/xml/protocols">
  <op>1</op>
  <to>//cse1.mym2msp.org</to>
  <fr>/cse1234/app567</fr>
  <rqi>0002bf63</rqi>
  <ty>4</ty>
  <pc>
    <m2m:cin>
      <cnf>application/xml:1</cnf>
      <con>PHRpbWU+MTc4ODkzMdk8L3RpbWU+PHRlbXA+MjA8L3RlbXA+DQo=</con>
    </m2m:cin>
  </pc>
</m2m:rqp>
```

The XML elements have the following meaning:

- rqp: Root element of the Request primitive, which includes a reference to an XSD file which defines its datatype.
- op: **Operation** parameter of datatype m2m:operation: in this example value = 1 indicates a "Create" operation.
- to: **To** parameter of type m2m:anyURI: URI of the target resource.



- fr: **From** parameter of type m2m:ID: ID of the Originator (either AE-ID or CSE-ID).
- rqi: **Request Identifier** parameter of type m2m:requestID: this could e.g. represent a counter number.
- ty: **Resource Type** parameter of datatype m2m:resourceType: indicating type of the resource to be created (value = 4 indicates that a <contentInstance> resource shall be created).
- pc: **Content** parameter of datatype m2m:primitiveContent: the attributes of the resource to be provided by the Originator.
- cin: Root element of the <contentInstance> resource of datatype m2m:contentInstance: this includes the mandatory attributes (and optional attributes not shown in this example) supplied by the request Originator. In this example, the **Content** parameter includes an instance of a <contentInstance> resource which consists of two attributes: contentInfo (cnf) – which specifies base64 encoding - and the content (con) itself.

## 8.4 JSON serialization

### 8.4.1 Terminology

The following conventions are used in the clause that follows.

- The italicized terms *object*, *member*, *name*, *array*, *number*, *string*, *boolean* and *null* are to be interpreted as in RFC 7159 [19]
- The italicized term *element* is to be interpreted to encompass oneM2M Primitive Parameters, Resource Attributes and other elements or attributes used inside oneM2M complex type definitions

### 8.4.2 Method

The primitive shall be encoded as a JSON *object*, conforming to the requirements of RFC 7159 [19]. This JSON *object* shall be restricted to Unicode characters defined in The Unicode Standard and encoded using UTF-8 as described in RFC 3629 [21]. The names in each *object* in the JSON shall be unique.

The structure of the top-level primitive *object* shall be determined by the data type definitions in clause 6 and clause 7 of the present document, as follows:

1. All *member's names* shall be the short name defined in clause 8.2.
2. If an *element* is defined in the present document as having a complex type, then it is serialized in the JSON *member* as an *object* and its children are recursively serialized as members of that *object*, using short names as defined in clause 8.2.
3. The membership of each nested *object* shall respect the cardinality constraints from the corresponding XSD complex type definition,
4. If an *element* is defined in the present document as having an atomic data type that is numeric (including enumeration data types in clause 6.3.4) then its value is serialized into the JSON *member* as a *number*.
5. If an *element* is defined as having an atomic data type that is non-numeric then its value is serialized into the JSON *member* as a *string*.
6. If an *element* is defined as xs:boolean (or a type derived from xs:boolean) then it is serialized in the JSON *member* as a *boolean*.
7. If an *element* is defined as having an xs:list type in the corresponding XSD then it is serialized in the JSON *member* as an *array*.
8. If an *element* instance has a null value then it is serialized into the JSON *member* as a *null*, regardless of the data type that it has in the corresponding XSD.

9. If an *element* is defined as having `maxOccurs > 1` in the corresponding XSD then its occurrences are serialized in a single JSON *member* as an *array*.
10. If an *element* has an XSD data type that is a simple type with XML attributes, then it is serialized in the JSON member as an *object*. The XML attributes appear as *members* of that object (using their short names) and the value of the *element* is serialized as a *member* of that *object* with the special name "val".
11. The *members* (at each level) may be serialized in any order. The order in which they appear in the corresponding XSD file is immaterial.
12. If an *element* has an XSD data type that is a complex type with XML attributes, then it is serialized in JSON as an *object*. The XML attributes appear as *members* of that object (using their short names) as do the XML elements.

The **Content** parameter is treated just like any other parameter of complex type. It is serialized as an object and its members are the attributes and/or child resource references of the Resource that is being transferred. The **Content** parameter is not required to contain all the attributes of the Resource. The JSON representation of the **Content** parameter shall be encapsulated by a member name as defined in the first column of Tables 7.5.2-1 and 7.5.2-2.

### 8.4.3 Examples

Here is an example that shows the payload of a request message serialized using JSON:

```
{ "op": "1", "fr": "//xxxxx/2345", "to": "//xxxxx/99", "rqi": "A1234", "pc": { "m2m:sch": { "se": "* 0-5 2,6,10 * * * *" } }, "ty": 18 }
```

- op: operation (in this case it is Create)
- fr: ID of the Originator (either the AE or CSE)
- to: URI of the target resource
- rqi: request identifier (this is a string)
- pc: attributes of the <schedule> resource with member name "m2m:sch" to be provided by Originator. This is serialized as a nested JSON object
- ty: type of resource to be created (in this case a Schedule resource). This is a number.

Note that the Operation (op) parameter is present only in Request primitives. The presence of this parameter in JSON serialized primitive representations allows to differentiate Request primitives from Response primitives.

The example below shows an <AE> resource serialized using JSON where m2m:ae is a Global Element having an XML attribute "rn" defined in the XSD file with short names associated with the <AE> resource:

```
{
  "m2m:ae": {
    "rn": "appname",
    "aei": "CAE01",
    "ct": "20160404T132648",
    "et": "20160408T004648",
    "lt": "20160404T132648",
    "pi": "ONET-CSE-02",
    "ri": "REQID1",
    "ty": 2
  }
}
```

## 8.5 CBOR serialization

### 8.5.1 Method

Concise Binary Object Representation (CBOR) is a binary serialization format of structured data specified in RFC7049 [38]. CBOR provides unambiguous encoding of structured data into a binary representation and reverse decoding.

The specifics on how CBOR can be negotiated between protocol endpoints is protocol specific and defined by the individual bindings.

This clause defines the relationship between JSON objects as defined in clause 8.4 and CBOR representations.

Section 2 of RFC7049 [38] specifies the applicable CBOR encoding rules.

In particular, the following rules shall apply when using CBOR serialization:

- Text strings (i.e. any names/keys and text string values) shall be encoded as UTF-8 strings, CBOR major type 3.
- Integer numbers shall be encoded as CBOR major types 0 or 1.
- Floating point numbers shall be encoded as CBOR major type 7 with Additional Information 26 for single precision (32-bit) and Additional Information 27 for double precision (64-bit) formats.

Note that CBOR ignores whitespace characters (including space, LF/CR) if used for formatting of JSON objects in textual representations.

If decoding of CBOR serializations results in values not compliant with the underlying XSD, this shall be interpreted as an error by the receiver of the primitive.

### 8.5.2 Examples

This clause presents some examples of CBOR serialized primitives. Note that due to given encoding options, a CBOR encoder may produce somewhat different binary serializations. However, in any case the CBOR decoding shall produce an equivalent representation in JSON format as shown in the examples below.

#### *Example 1:*

JSON representation (a request primitive of message length: 173 bytes):

```
{ "m2m:rqp": { "op": 1, "to": "//example.net/mncsel234", "rqi": "A1000",  
  "rcn": 7, "pc": { "m2m:ae": { "rn": "SmartHomeApplication", "api": "Na56",  
    "apn": "app1234" } }, "ty": 2 } }
```

CBOR representation as sequence of hexadecimal characters (length: 117 bytes):

```
a1476d326d3a727170a6427063a1466d326d3a6165a342726e54536d617274486f6d654170706c69636174696f6e436  
17069444e6135364361706e47617070313233344274790242746f572f2f6578616d706c652e6e65742f6d6e63736531  
3233344372636e07426f700143727169454131303030
```

#### *Example 2:*

JSON representation (a response primitive of message length: 234 bytes):

```
{ "m2m:rsp": { "rsc": 2001, "rqi": "A1000", "pc": { "m2m:ae": { "rn": "SmartHomeApplication", "ty":  
  2, "ri": "ae1", "api": "Na56", "apn": "app1234", "pi": "cb1", "ct": "20160506T153208",  
  "lt": "20160506T153208", "acpi": [ "acp1", "acp2" ], "et": "20180506T153208",  
  "aei": "S_SAH25" } } } }
```

CBOR representation as sequence of hexadecimal characters (length: 187 bytes):

```
a1476d326d3a727370a3427063a1466d326d3a6165ab43617069444e6135364361706e47617070313233344265744f  
3230313830353036543135333230384263744f3230313630353036543135333230384274790242726943616531426  
c744f3230313630353036543135333230384361656947535f534148323542726e54536d617274486f6d654170706c6
```

9636174696f6e427069436362314461637069824461637031446163703243727169454131303030437273631907d1

**Example 3:**

JSON representation (request primitive of message length: 187 bytes):

```
{ "m2m:rsp": { "rsc": 2001, "rqi": "A1000", "pc": { "m2m:ae": { "rn": "SmartHomeApplication", "ty": 2, "ri": "ae1", "api": "Na56", "apn": "app1234", "pi": "cb1", "ct": "20160506T153208", "lt": "20160506T153208", "acpi": ["acp1", "acp2"], "et": "20180506T153208", "aei": "S_SAH25" } } } }
```

CBOR representation as sequence of hexadecimal characters (length: 133 bytes):

a1476d326d3a727170a6427063a1476d326d3a636e74a3436d6e691901f442726e52536d617274486f6d65436f6e7461696e6572436d62731a000186a04274790342746f582c2f2f6578616d706c652e6e65742f6d6e637365313233342f536d617274486f6d654170706c69636174696f6e4372636e07426f700143727169454131303031

**Example 4:**

JSON representation (response primitive of message length: 306 bytes):

```
{ "m2m:rsp": { "rsc": 2001, "rqi": "A1001", "pc": { "m2m:cnt": { "rn": "SmartHomeContainer", "ty": 3, "ri": "cnt1", "pi": "ae1", "ct": "20160506T154048", "lt": "20160506T154048", "acpi": ["acp1"], "et": "20180506T154048", "cr": "S_SAH25", "st": 0, "mni": 500, "mbs": 100000, "cni": 0, "cbs": 0, "mia": 3600 } } } }
```

CBOR representation as sequence of hexadecimal characters (length: 197 bytes):

a1476d326d3a727370a3427063a1476d326d3a636e74af436362730042726944636e7431436d6e691901f442637247535f53414832354265744f3230313830353036543135343034384263744f323031363035303654313534303438436d62731a000186a042747903436d6961190e1042737400426c744f32303136303530365431353430343842726e52536d617274486f6d65436f6e7461696e657242706943616531446163706981446163703143636e690043727169454131303031437273631907d1

---

# Annex A (Normative): Binding Mch to Diameter for Charging

## A.1. Introduction

Present clause provides Diameter binding of Mch.

## A.2. Diameter Commands on Mch

### A.2.1. Accounting Request Command

The ACR command is sent from the Charging Function (CHF included within the SCA CSF) embedded within the M2M IN to the Charging Server using the Mch reference point. This command is used for Event Based requests.

The ACR message format is defined according to the Diameter Base Protocol in RFC 3588 [13] as follows:

```
<ACR> ::= < Diameter Header: 271, REQ, PXY >
    < Session-Id >
    { Origin-Host }
    { Origin-Realm }
    { Destination-Realm }
    { Accounting-Record-Type }
    { Accounting-Record-Number }
    [ Acct-Application-Id ]
    [ Destination-Host ]
    [ Origin-State-Id ]
    [ Event-Timestamp ]
    * [ Proxy-Info ]
    * [ Route-Record ]
    [ Service-Context-Id ]
    [ Service-Information ]
    * [ AVP ]
```

### A.2.2. Accounting Answer Command

The ACR command is sent from the Charging Server to the Charging Function (CHF included within the SCA CSF) embedded within the M2M IN in response to the ACR command and is used to acknowledge reception of the charging data. This command is used for Event Based responses.

The ACA message format is defined according to the Diameter Base Protocol in RFC 3588 [13] as follows:

```
<ACA> ::= < Diameter Header: 271, PXY >
    < Session-Id >
    { Result-Code }
    { Origin-Host }
    { Origin-Realm }
    { Accounting-Record-Type }
    { Accounting-Record-Number }
    [ Acct-Application-Id ]
    [ User-Name ]
    [ Origin-State-Id ]
    [ Event-Timestamp ]
    * [ Proxy-Info ]
    * [ AVP ]
```

## A.3. Mapping of M2M Recorded Information Elements to AVPs

The following table describes the mapping of the M2M Recorded Information Elements identified in TS-0001 to the Diameter AVPs.