8.5 Component-to-Component Transfer (C2CX) Messages

The Component-to-Component Transfer (C2CX) Messages provide the ability to transfer control and status information between components. The messages are used for sending status, control, setup, initialization, heartbeat, security, a handshake, etc. from one component to another (one-to-one) or to multiple components (one-to-many).

The C2CX messages are typically unidirectional. If components need back and forth interaction or confirmation, it is recommended that the Directive Request and Response messages be used. Also, if it is desirable to know the progress or status of a C2CX message, the receiver of the message can issue a Log message at noteworthy points within its processing cycle. This means the C2CX messages act at a layer below general system knowledge that is accomplished through other messages such as the Log message and Directive messages. The C2CX messages seek to provide a communication mechanism just under the radar of the general system, but also easily viewable, as necessary.

8.5.1 Configuration Status Message

The **CFG – Configuration Status** C2CX message is used by software components to report their configuration information. (The **DEV – Device** C2CX message is used to report the status of devices.) Note that the Message Header already contains information about the component:

- Support of mission and satellites
- Name and location (facility and node)
- Class of capabilities provided

The additional information to be passed or reported is non-standard. The components reporting the configuration information and the components monitoring the configuration will need to establish:

- What configuration information is to be reported (and name the fields)
- When it is to be reported (upon change or periodically)
- What format to report the configuration information

A monitoring agent would collect Heartbeat messages and Configuration Status messages and possibly the Device messages to maintain a picture of what software is operating where, in what capacity, in what state, and with what physical and/or logical associations. The collected information can be:

- Presented in a graphical colored display depicting the operating environment and the logical associations of the components
- Used to determine what pre-determined actions to take in the event of a failure or degraded operations

As an example, a front-end telemetry and command processor would publish a CFG message whenever it is first run and thereafter when it associates (or disassociates) itself with another

cooperating component. In addition to the information already provided in the Message Header, it would also report the following configuration information:

- The role of the reporting component (PRIMARY, BACKUP)
- Number of associations

  Name of component or port associated with, such as:

  - Telemetry and command processor (decommutation and command verification)
  - External telemetry and command link / port
  - Planner and Scheduler - to direct the setup and operation of a pass
  - Flight dynamics component - to exchange downlink or other information

  Node of the associated component, if known

  Role of the associated component (PRIMARY, BACKUP)

If the monitoring agent is also a configuration manager, it can establish the present operating configuration and also prepare a contingent configuration. In the event of a component failure or processor failure, the configuration manager will know if it has the required and sufficient number of components to sustain operations. If not, it can also determine if it has the required and necessary numbers of components should a failover or restart procedure be invoked, and if so, automatically initiate that procedure.

The minimum and sometimes maximum configuration information a component can report is its own role. A single component with no associations would normally report its role as PRIMARY. Some configuration information may already be known and available if the components used a pre-registration or registration mechanism to disclose such information as:

- Nodes where they can execute
- If they are standalone or redundant, and if redundant, on what nodes could the redundant component operate

Furthermore, the Configuration Status Message may be used to report group associations. That is, to what group does this component belong, and is it a member and/or a manager of the group. Some groups of components may operate in a peer only association where other groups may require a group manager for organization, control, and direction.

Groups can be used for a number of purposes. These include, but are not limited to:

Message Exchange - groups can be formed to pass messages in a number of relationships and locations that include:

- Peer-to-Peer
- Client-Server
- Manager-Member
- Local and Distributed

Configuration Management - equipment can be logically associated to form groups (or suites or strings) that must operate together, failover together, have a minimum configuration (quorum), be addressed as a group, or other operating constraints or configurations.

Group formation can be pre-defined or dynamic. If dynamic, then additional group functions may be required, such as Create / Disband and Join / Leave.

Groups can also be hierarchical, as shown**Error! Reference source not found. Error! Reference source not found.**.

Table Error! No text of specified style in document.**-1. Group Hierarchical Associations**

Grouping Level	Space	Ground
Software	Software Application	Software Application
Hardware / Equipment	Processor	Computer
	Bus	Bus/LAN/WAN/Web
	Satellite	Facility/Center
	Constellation	Enterprise
Business	Mission	

The above discussion and illustrations provide a sampling of the ways groups may be employed within the messaging framework.

Table Error! No text of specified style in document.**-2. Configuration Status Message Summary**

Sender	Any C2MS compliant application
Senders Intended Usage	Publish
Receiver	Any C2MS compliant application
Receivers Intended Usage	Subscribe
What	Status and Control type information
When	As needed and depends upon the type of information being transferred

Example

1. A component needs information about another component's software configuration information from a logical or relational perspective.

8.5.1.1 Configuration Status Message Subjects

Table Error! No text of specified style in document.**-3. Configuration Status Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	CFG	[Component name of publisher]	[Component name of recipient(s)]	[Node]	[Facility]	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	CFG	APP1				
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	CFG	CFGMGR	AGENT3			
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	MSSN	*	SAT1	MSG	CFG	*	>			

Table Error! No text of specified style in document.**-4. Properties of the Miscellaneous Elements for the Configuration Status Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Optional	Component name of destination	DESTINATION-COMPONENT from header
ME3 *ME4*	Optional	Component destination: The two *miscellaneous elements* may be used to direct the message to a specific destination, as necessary, in Header String format. *ME3* = destination node *ME4* = destination facility	

Configuration status messages may be published for general consumption or they may be targeted to a central collector component. The second element, *ME2* (component of recipient), is used if necessary.

Examples

Publishing / Sending Configuration Status messages:

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CFG.APP1 or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CFG.APP1.COMPONENT or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CFG.APP1.COMPONENT.NODE.F
ACILITY
```

Subscribe to / receive a Configuration Status message:

```
C2MS.*.*.*.*.*.MSG.CFG.MYAGENT.CFGMGR.> or

C2MS.*.*.*.*.*.MSG.CFG.CFGMGR.>
```

8.5.1.2 Configuration Status Message Header

The abbreviated **Error! Reference source not found.Error! Reference source not found.** shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for the Configuration Status Message header.

Table Error! No text of specified style in document.**-5. Configuration Status Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	CFG	Unique message identifier, fixed for C2MS Standard Messages

Field Name	Value	Notes
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.5.1.3 Configuration Status Message Contents

Figure **Error! No text of specified style in document.**-1 below shows a UML object diagram of the Configuration Status Message with its required and optional fields.

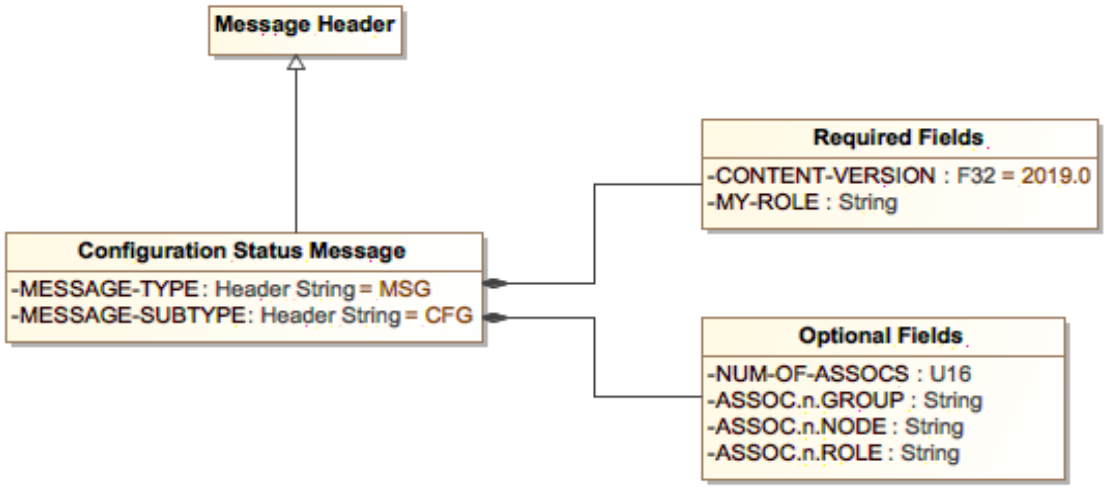

Figure Error! No text of specified style in document.**-1. Configuration Status Message Diagram**

Table **Error! No text of specified style in document.**-6 below describes additional field names, values, and notes for the Configuration Status Message.

Table Error! No text of specified style in document.**-6. Configuration Status Message Additional Information**

Field Name	Value/Description	Notes
CONTENT-VERSION	2019	Version Number for this message content description
MY-ROLE		Role the reporting component has in the configuration. E.g. PRIMARY, BACKUP, AGENT, SERVER, MEMBER, MGR, …
NUM-OF-ASSOCS	"n" starts at "1"	The number of associations to be reported.
ASSOC.n.GROUP		Name of component or group associated with
ASSOC.n.NODE		Location of associated component or group
ASSOC.n.ROLE		Role the associated component has, if known

8.5.2 Control Message

The Control messages are those that are typically used to request "under the hood" functions that, while essential for operations, are not of general interest.

The **Control** message is used to start, restart, reinitialize, or otherwise control another component. As an example, components may determine that they will not proceed with their processing until they have received a Control message with a CNTL-STRING of "INIT" or "START". Other components may require additional information in CNTL-STRING to begin processing. Still other components that have been performing their processing may allow themselves to be re-directed in their processing. Upon the reception of a Control message, a component will re-direct itself according to the supplied string of parameters.

For instance, a component could request another component to change its rate of publishing a Heartbeat message. The requesting component would send a Control message with a known decipherable command string in the CNTL-STRING field; for example: "SET HB 15".

Missions may want to have different processing modes or signals when the course of events changes what standard actions are to follow. For example, a system wide indicator may be sent using the CNTL messages with a value for the CNTL-STRING to signify that a pass has begun and the processing mode is 'PASS'. Or, the processing mode is 'PRE-PASS', 'POST-PASS', 'LIGHTS-OUT', 'LIGHTS-ON'. 'AUTONOMOUS', 'SIMULATION', 'LAUNCH', 'ECLIPSE-PERIOD', 'MANEUVER', 'SAFE-MODE', or any such state that could affect some components and result in conditional processing or decision making.

In these examples, a monitoring agent, criteria action agent, decision making component, script controller, or processing manager would monitor the events for state changes and then issue the CNTL message for all or a subset of the components.

A further example could involve distributed simulations. A key factor in these simulations is to know the simulated time. A CNTL message can be defined to set, distribute, or synchronize components to a simulated time. The CNTL message might be used to set the time or advance the simulated time by a delta time. This includes training, development, integration, and pre-launch / operations simulations. If necessary, a separate C2CX message may be developed with a called SETTIME with associated parameters.

Finally, a simple application could be a PING function. "PING" placed in the CNTL-STRING field would simply require the receiver to publish the same type of message but with "PING-ACK" in the CNTL-STRING field. Alternately, separate C2CX messages could be defined for the PING and PING-ACK functionality.

Table Error! No text of specified style in document.**-7. Control Message Summary**

Sender	Any C2MS compliant application
Senders Intended Usage	Publish
Receiver	Any C2MS compliant application
Receivers Intended Usage	Subscribe
What	Status and Control type information

When	As needed and depends upon the type of information being transferred

Example

1. A component can request another component to action with the CNTL message.

8.5.2.1 Control Message Subjects

Table Error! No text of specified style in document.**-8. Control Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	CNTL	[Component name of publisher]	[Component name of recipient(s)]	[Node]	[Facil ity]	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	CNTL	APP1				
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	CNTL	CFGMGR	AGENT3			
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	MSSN	*	SAT1	MSG	CNTL	*	>			

Table Error! No text of specified style in document.**-9. Properties of the *Miscellaneous Elements* for the Control Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Optional	Component name of destination	DESTINATION-COMPONENT from header
ME3 *ME4*	Optional	Component destination: The two *miscellaneous elements* may be used to direct the message to a specific destination, as necessary, in Header String format. *ME3* = destination node *ME4* = destination facility	n/a
ME5	Optional	A keyword the receiving process could use for filtering	"CNTL-KEYWORD" from msg content

Control messages may be published for general consumption or they may be targeted to a central collector component. The second element, *ME2* (component of recipient), is used if necessary.

Examples

Publishing / Sending Control messages:

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CNTL.APP1 or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CNTL.APP1.COMPONENT or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CNTL.APP1.COMPONENT.NODE.
FACILITY or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.CNTL.APP1.COMPONENT.NODE.
FACILITY.KEYWORD
```

Subscribe to / receive a Control message:

```
C2MS.*.*.*.*.*.MSG.CNTL.MYAGENT.CFGMGR.> or

C2MS.*.*.*.*.*.MSG.CNTL.CFGMGR.>
```

8.5.2.2 Control Message Header

The abbreviated **Error! Reference source not found.Error! Reference source not found.** shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for the Control Message header.

Table Error! No text of specified style in document.**-10. Control Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description

Field Name	Value	Notes
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	CNTL	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.5.2.3 Control Message Contents

Figure **Error! No text of specified style in document.**-2 below shows a UML object diagram of the Control Message with its required and optional fields.

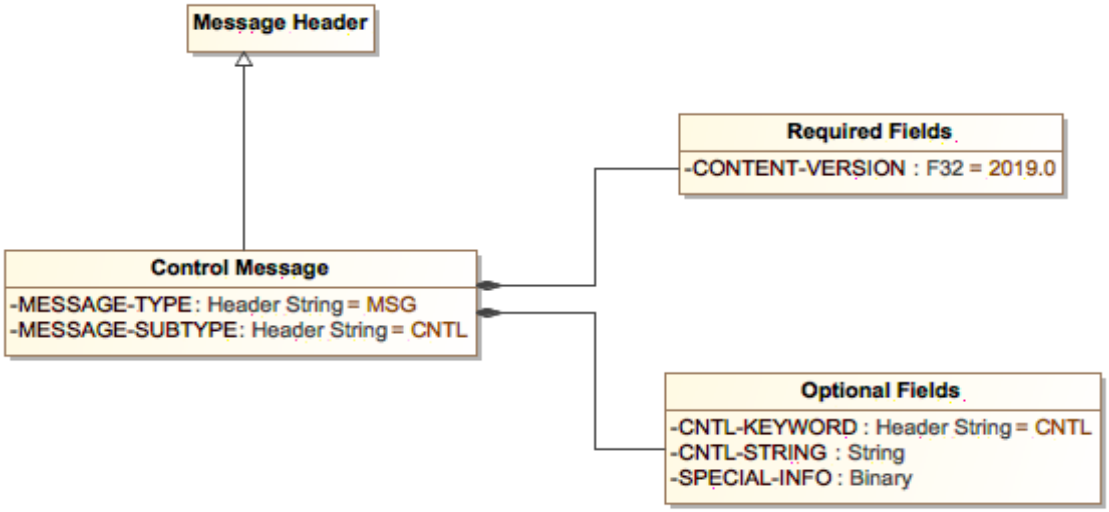

Figure Error! No text of specified style in document.**-2. Control Message Diagram**

Table **Error! No text of specified style in document.**-11 below describes additional field names, values, and notes for the Control Message.

Table Error! No text of specified style in document.**-11. Control Message Additional Information**

Field Name	Value/Description	Notes
CONTENT-VERSION	2019	Version Number for this message content description
CNTL-KEYWORD	Uppercase	Keyword extracted from the CNTL-STRING. Useful for routing/processing.
CNTL-STRING		Parameters to guide the component on further processing. E.g., INIT, Stop, Shutdown, Restart, Do X, Y, and Z.
SPECIAL-INFO		For application use. Any additional information can be provided here.

8.5.3 Device Message

The **Device** C2CX message is used to report the status of devices, physical or virtual. (The Heartbeat message and the Configuration Status message are used to report status on software components.) The Device message would typically be used to report the status of devices that would not be capable of reporting themselves. For example, a software component may interact with a specialized device or merely have access to the status of devices operating in the same environment. A designated software component would gather the status of the device(s) and publish the information with the Device C2CX message. This message is not intended to communicate with the device.

Thus, in conjunction with the Configuration Status and Heartbeat messages, a full story on the configuration can be gathered for reporting and subsequent actions when a system-wide re-configuration is implemented.

Of course, this message does not need to be restricted to physical devices. Virtual devices may be constructed and reported on as well. A physical device may be logically partitioned, or a logical device may be spread over a number of physical devices. Or, a virtual (or pseudo) device could be constructed or defined with no relation to any physical device. For example, a set of parameters, somehow related, could be grouped as a "device" and reported on for display and monitoring. A single reporting agent could be responsible for a virtual device and report on it. Or, a number of agents could report on separate parameters and the collector of the Device messages could effectively construct a virtual device from the disparate information. A set of key or critical parameters could be constructed and reported on using this method.

A hypothetical example for a communications data path could consist of a ground antenna, a ground station processor/controller, a data link, and a front-end processor. Together these devices could constitute a virtual data link device whose individual device statuses are collected (and logically ANDed together) to provide a GO/NOGO or Red/Yellow/Green status on the data link.

Table Error! No text of specified style in document.**-12. Device Message Summary**

Sender	Any C2MS compliant application
Senders Intended Usage	Publish
Receiver	Any C2MS compliant application
Receivers Intended Usage	Subscribe
What	Status and Control type information
When	As needed and depends upon the type of information being transferred

Example

1. A component reports its status, physical or virtual, or any collection of data.

8.5.3.1 Device Message Subjects

Table Error! No text of specified style in document.**-13. Device Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	DEV	[Component name of publisher]	[Component name of recipient(s)]	[Node]	[Facil ity]	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	DEV	APP1				
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	DEV	CFGMGR	AGENT3			
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	MSSN	*	SAT1	MSG	DEV	*	>			

Table Error! No text of specified style in document.**-14. Properties of the *Miscellaneous Elements* for the Device Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Optional	Component name of destination	DESTINATION-COMPONENT from header
ME3 *ME4*	Optional	Component destination: The two *miscellaneous elements* may be used to direct the message to a specific destination, as necessary, in Header String format. *ME3* = destination node *ME4* = destination facility	n/a

Device messages may be published for general consumption or they may be targeted to a central collector component. The second element, *ME2* (component of recipient), is used if necessary.

Examples

Publishing / Sending Device messages:

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.DEV.APP1 or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.DEV.APP1.COMPONENT or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.DEV.APP1.COMPONENT.NODE.F
ACILITY
```

Subscribe to / receive a Device message:

```
C2MS.*.*.*.*.*.MSG.DEV.MYAGENT.CFGMGR.> or

C2MS.*.*.*.*.*.MSG.DEV.CFGMGR.>
```

8.5.3.2 Device Message Header

The abbreviated table**Error! Reference source not found. Error! Reference source not found.** shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for the Device Message header.

Table Error! No text of specified style in document.**-15. Device Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	DEV	Unique message identifier, fixed for C2MS Standard Messages

Field Name	Value	Notes
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.5.3.3 Device Message Contents

Figure **Error! No text of specified style in document.**-3 below shows a UML object diagram of the Device Message with its required and optional fields.

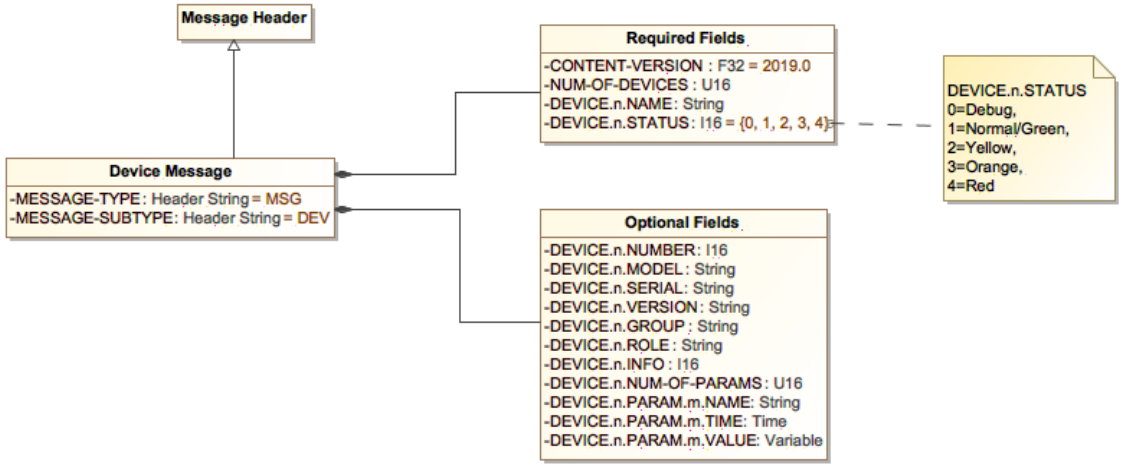

Figure Error! No text of specified style in document.**-3. Device Message Diagram**

Table **Error! No text of specified style in document.**-16 below describes additional field names, values, and notes for the Device Message.

Table Error! No text of specified style in document.**-16. Device Message Additional Information**

Field Name	Value/Description		Notes
CONTENT-VERSION	2019		Version Number for this message content description
NUM-OF-DEVICES	1+		Number of devices being reported in this message
DEVICE.n.NAME	"n" starts at 1		Name of the device
DEVICE.n.NUMBER			Number assigned to the device to distinguish it from identical devices.
DEVICE.n.MODEL			Model number of the device
DEVICE.n.SERIAL			Serial number of the device
DEVICE.n.VERSION			Version of the firmware operating within the device
DEVICE.n.GROUP			Name of group with which device is associated
DEVICE.n.ROLE			Role of the device, if known

Field Name	Value/Description		Notes
DEVICE.n.STATUS	Value	Description	Condition of the device being reported. The criteria for selecting the DEVICE.n.STATUS description is left to the reporting component.
	0	Debug	
	1	Normal / Green	
	2	Yellow	
	3	Orange	
	4	Red	
DEVICE.n.INFO			An additional status code that can be supplied that is specific to that device
DEVICE.n.NUM-OF-PARAMS	"n" starts at 1		Number of additional parameters being reported that are associated with the device
DEVICE.n.PARAM. m.NAME			Name of the additional parameter
DEVICE.n.PARAM. m.TIME			Time of parameter sampling
DEVICE.n.PARAM. m.VALUE			Value of the named parameter being reported

8.5.4 Heartbeat Message

The Heartbeat message is used to notify other components that the sending / publishing component is alive or active. Other components monitoring the Heartbeat messages can determine what action to take, if any, when a Heartbeat message fails to appear as scheduled, or if the COMPONENT-STATUS is not normal/green. If the component does not publish the heartbeat at the default rate, it can supply the publishing rate (PUB-RATE field) and a counter (COUNTER field) for a monitor to calculate when a heartbeat is expected or might be missing or late. Each component or system or mission can determine its own preferred heartbeat rate.

Not Using the COMPONENT-STATUS Field

If the COMPONENT-STATUS field is not to be used, then if the component is running but not 100%, then a component should cease publishing its Heartbeat message. The termination of the Heartbeat message will then make auto/re-configuration options possible. Which is to say, if the component is either 100% or 0%, or those are the only two states the component can report, then using the COMPONENT-STATUS field is not necessary, as long as the component can cease publishing the Heartbeat message in circumstances when it knows it is not 100%.

Using the COMPONENT-STATUS Field

A component may typically only supply the COMPONENT-STATUS of 1 – Normal/Green. However, when the status of the component is less than normal / green (100%), say yellow (75%), indicating a less than optimal operating state, it may also supply a status code in the COMPONENT-INFO field. This code would only have context within that component. A component may also issue a Log Message in conjunction with a change in COMPONENT-STATUS. The component would include the COMPONENT-INFO value in the subsequent Log Message so the Heartbeat message and the Log Message could be cross-referenced. Components can self-determine what constitutes a yellow, orange, or red state of processing. A component that ceases to send a heartbeat message will be presumed to be absent and in a red condition. If

applicable, a component will then be susceptible to a pre-determined recovery action, including failover and restart. A monitoring agent can use the COMPONENT-STATUS value to color code a display of the component's status.

Memory Leaks

A recurring and nagging problem in software development is the presence of memory leaks, i.e., the failure to return unused memory to the operating system. Over time, this causes a process (task) to use up all of its allotted memory and/or the total system memory resulting in a bogged down or inoperable system.

Having the component report its own memory usage in the Heartbeat message could be used to monitor for and discover memory leaks during execution. A separate monitoring agent could collect the Heartbeat messages, and among other responsibilities, monitor and track the memory usage for each process over time.

If the memory usage is determined to be steadily increasing over time, an action could be generated to issue a warning (Log) message. (An algorithm must be developed or found that would detect memory leaks for any process, regardless of the amount of memory it may require on any operating system.)

Table Error! No text of specified style in document.**-17. Heartbeat Message Summary**

Sender	Any C2MS compliant application
Senders Intended Usage	Publish
Receiver	Any C2MS compliant application
Receivers Intended Usage	Subscribe
What	Status and Control type information
When	As needed and depends upon the type of information being transferred

Example

1. All active components publish a heartbeat (aka keep-alive); a monitoring component checks on the ongoing presence of the components and detects their absence.

8.5.4.1 Heartbeat Message Subjects

Table Error! No text of specified style in document.**-18. Heartbeat Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	HB	[Component name of publisher]	[Component name of recipient(s)]	[Node]	[Facil ity]	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	HB	APP1				
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	HB	CFGMGR	AGENT3			
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	MSSN	*	SAT1	MSG	HB	*	>			

Table Error! No text of specified style in document.**-19. Properties of the *Miscellaneous Elements* for the Heartbeat Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Optional	Component name of destination	DESTINATION-COMPONENT from header
ME3 *ME4*	Optional	Component destination: The two *miscellaneous elements* may be used to direct the message to a specific destination, as necessary, in Header String format. *ME3* = destination node *ME4* = destination facility	n/a

Heartbeat messages may be published for general consumption or they may be targeted to a central collector component. The second element, *ME2* (component of recipient), is used if necessary.

Examples

Publishing / Sending Heartbeat messages:

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.HB.APP1 or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.HB.APP1.COMPONENT or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.HB.APP1.COMPONENT.NODE.FA
CILITY
```

Subscribe to / receive a Heartbeat message:

```
C2MS.*.*.*.*.*.MSG.HB.MYAGENT.CFGMGR.> or

C2MS.*.*.*.*.*.MSG.HB.CFGMGR.>
```

8.5.4.2 Heartbeat Message Header

The abbreviated table**Error! Reference source not found. Error! Reference source not found.** shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for the Heartbeat Message header.

Table Error! No text of specified style in document.**-20. Heartbeat Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	HB	Unique message identifier, fixed for C2MS Standard Messages

Field Name	Value	Notes
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.5.4.3 Heartbeat Message Contents

Figure **Error! No text of specified style in document.**-4 below shows a UML object diagram of the Heartbeat Message with its required, optional, and tracking fields.

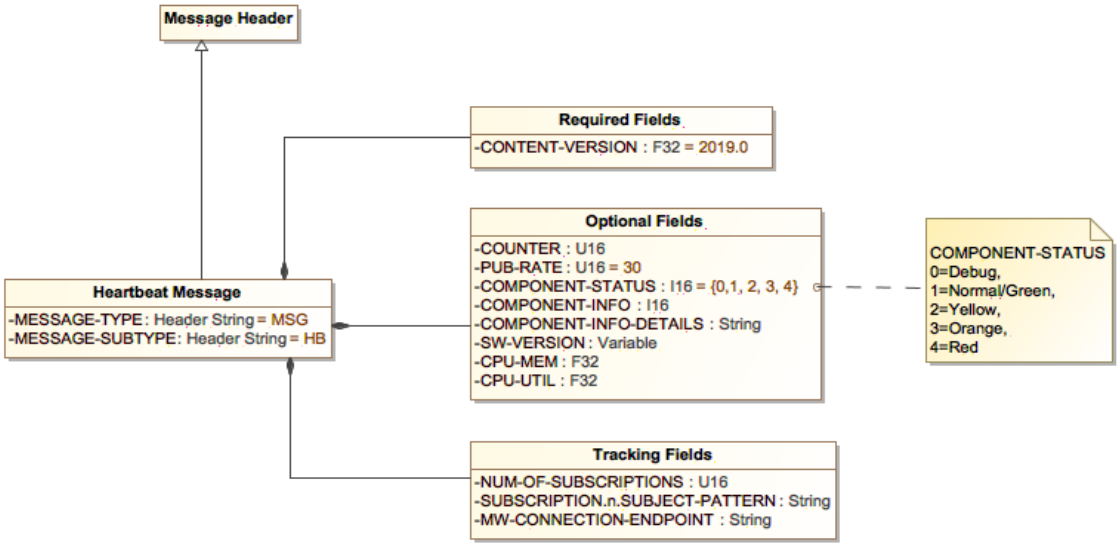

Figure Error! No text of specified style in document.**-4. Heartbeat Message Diagram**

Table **Error! No text of specified style in document.**-21 below describes additional field names, values, and notes for the Heartbeat Message.

Table Error! No text of specified style in document.**-21. Heartbeat Message Additional Information**

Field Name	Value/Description		Notes
CONTENT-VERSION	2019		Version Number for this message content description
COUNTER*	1+		Indicates the number of times that the C2CX heartbeat message has been published, including this message.
PUB-RATE			Indicates the rate, in number of seconds, which the C2CX heartbeat message is being published by the component. A rate of zero or less indicates that this C2CX message is not repeatedly published by the component. The default publishing rate of the C2CX heartbeat message is 30 seconds.
COMPONENT-STATUS	**Value**	**Description**	Indicates the condition of the component being monitored, typically itself, although it may be a proxy for a remote component. The
	0	Debug	

Field Name	Value/Description		Notes
	1	Normal / Green	component may choose the condition level based on its own criteria.
	2	Yellow	
	3	Orange	
	4	Red	
COMPONENT-INFO			An additional status code the component can supply that is specific to that component.
COMPONENT-INFO-DETAILS			Allows a component to detail its status in a verbose message
SW-VERSION			Version number identifier of the reporting component. Component must ascertain the data type before accessing the value (e.g. with a function call).
CPU-MEM	In megabytes		Amount of memory being used at this time by this component.
CPU-UTIL			Percentage of CPU being utilized.
NUM-OF-SUBSCRIPTIONS	U16		The number of active subscriptions set up across all connections held by the running application - reserved for use by implementation (PSM)
SUBSCRIPTION.n. SUBJECT-PATTERN	String		The nth subscription subject pattern held by the running application - reserved for use by implementation (PSM)
MW-CONNECTION-ENDPOINT	String		Broker(s) to which the client application is currently connected - reserved for use by implementation (PSM)

* Note: At a rate of two messages per minute, this counter will overflow after 22 days.

8.5.5 Resource Message

The C2CX **Resource** message is used to publish computer performance data. Resource data is organized per CPU, disk, and network port. It is intended that the data be a snapshot of the resources at the time of collection and not a cumulative summary. The snapshot of the resources can be paired with the time of publication of the message to produce a data point. After the collection of a number of data points, a trend /plot of the resources can be established.

All resource information has been marked as optional so that a component may provide any or all of the resource information as necessary. For example, an agent collecting and publishing data may determine to publish the CPU resources at a difference rate than the disk resources. Resource messages for CPUs might be published every 60 seconds, while disk Resource messages might be published every 300 seconds.

If a component is to be controlled or directed as to the frequency of resource publishing, it could use the CNTL message with a CNTL-STRING of "`SET RSRC CPU 60`" to set the CPU resources publication rate at 60 seconds, or disk resources at 300 seconds with "`SET RSRC DISK 300`".

Table Error! No text of specified style in document.**-22. Resource Message Summary**

Sender	Any C2MS compliant application
Senders Intended Usage	Publish
Receiver	Any C2MS compliant application
Receivers Intended Usage	Subscribe
What	Status and Control type information
When	As needed and depends upon the type of information being transferred

Example

1. This message is used to report a snapshot of computer performance data (CPU, memory, disk, and network usage).

8.5.5.1 Resource Message Subjects

Table Error! No text of specified style in document.**-23. Resource Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	RSRC	[Component name of publisher]	[Component name of recipient(s)]	[Node]	[Facil ity]	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	RSRC	APP1				
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	RSRC	CFGMGR	AGENT3			
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	MSSN	*	SAT1	MSG	RSRC	*	>			

Table Error! No text of specified style in document.**-24. Properties of the *Miscellaneous Elements* for the Resource Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Optional	Component name of destination	DESTINATION-COMPONENT from header
ME3 *ME4*	Optional	Component destination: The two *miscellaneous elements* may be used to direct the message to a specific destination, as necessary, in Header String format. *ME2* = destination component or group *ME3* = destination node *ME4* = destination facility	n/a

Resource messages may be published for general consumption or they may be targeted to a central collector component. The second element, *ME2* (component of recipient), is used if necessary.

Examples

Publishing / Sending Resource messages:

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.RSRC.APP1 or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.RSRC.APP1.COMPONENT or
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.RSRC.APP1.COMPONENT.NODE.
FACILITY
```

Subscribe to / receive a Resource message:

```
C2MS.*.*.*.*.*.MSG.RSRC.MYAGENT.CFGMGR.> or

C2MS.*.*.*.*.*.MSG.RSRC.CFGMGR.>
```

8.5.5.2 Resource Message Header

The abbreviated **Error! Reference source not found.Error! Reference source not found.** shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for the Resource Message header.

Table Error! No text of specified style in document.**-25. Resource Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG

Field Name	Value	Notes
MESSAGE-SUBTYPE	RSRC	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.5.5.3 Resource Message Contents

Figure **Error! No text of specified style in document.**-5 below shows a UML object diagram of the Resource Message with its required and optional fields.

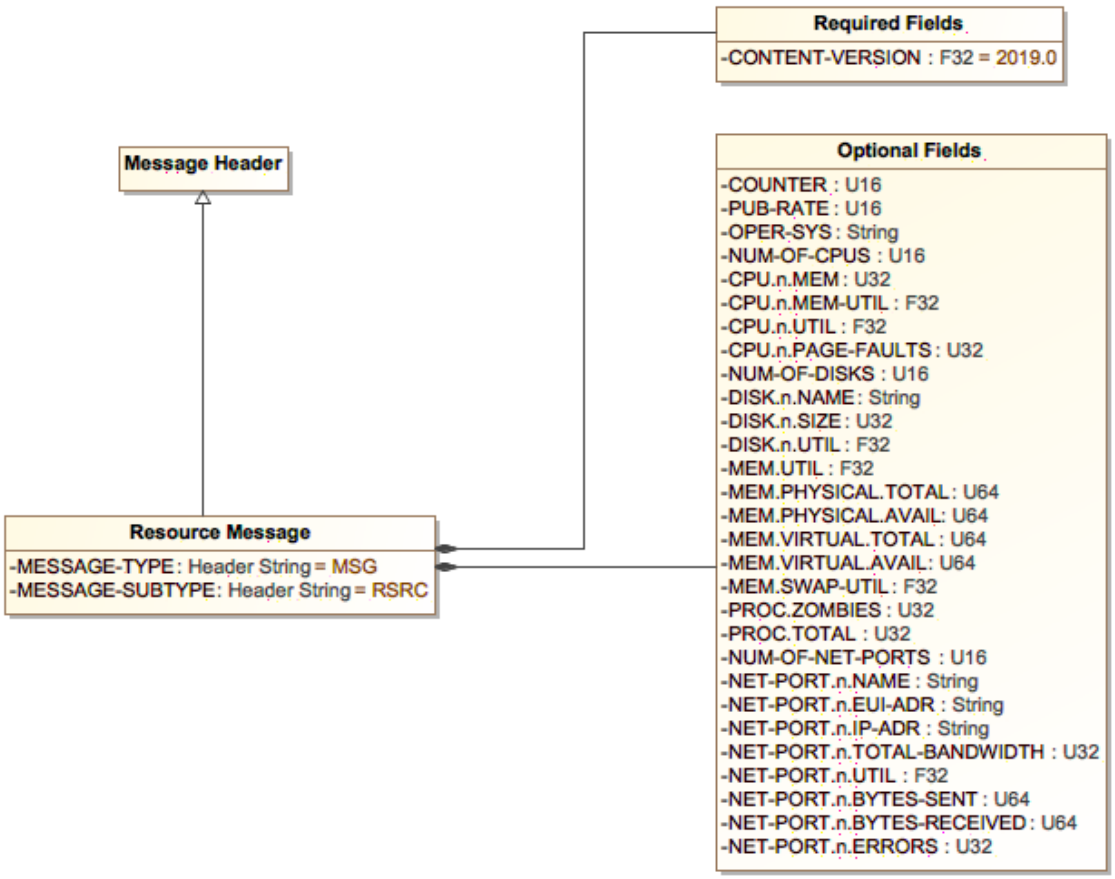

Figure Error! No text of specified style in document.**-5. Resource Message Diagram**

Table **Error! No text of specified style in document.**-26 below describes additional field names, values, and notes for the Resource Message.

Table Error! No text of specified style in document.**-26. Resource Message Additional Information**

Field Name	Value/Description	Notes
CONTENT-VERSION	2019	Version Number for this message content description
COUNTER*	1+	Indicates the number of times that the C2CX Resource message has been published, including this message.
PUB-RATE	Seconds	Rate the data is being collected and published. The default publishing rate is 30 seconds. A rate of zero indicates this message is not being repeatedly published.
OPER-SYS		Operating system component is using
NUM-OF-CPUS	1+	Number of CPUs being monitored
CPU.n.MEM	In megabytes. "n" starts at 1	Amount of memory for this CPU
CPU.n.MEM-UTIL	0-100	Memory utilization. Percentage of memory utilized.
CPU.n.UTIL	0-100	CPU utilization. Percentage of CPU utilized
CPU.n.PAGE-FAULTS	1+	Number of page faults
NUM-OF-DISKS	0+	Number of disks being monitored
DISK.n.NAME	"n" starts at "1"	Name of the disk
DISK.n.SIZE	In megabytes	Absolute size of the disk
DISK.n.UTIL	0-100	Disk space utilization. Percentage of Disk space utilized.
MEM.UTIL	0-100	Percent of main memory utilized
MEM.PHYSICAL.TOTAL	1+	Total amount of physical memory present, in bytes
MEM.PHYSICAL.AVAIL	0+	Total amount of physical memory available, in bytes
MEM.VIRTUAL.TOTAL	1+	Total amount of virtual memory present, in bytes
MEM.VIRTUAL.AVAIL	0+	Total amount of virtual memory available, in bytes
MEM.SWAP-UTIL	0-100	Percent of swap space used
PROC.ZOMBIES	0+	Number of zombie processes
PROC.TOTAL	0+	Number of total processes
NUM-OF-NET-PORTS	1+	Number of network ports
NET-PORT.n.NAME	"n" starts at "1"	Name of the network port
NET-PORT.n.EUI-ADR	Format of: 01-23-45-67-89-ab or 01:23:45:67:89:ab	Media Access Control (MAC) or Extended Unique Identifier (EUI) physical address. MAC-48, EUI-48, or EUI-64 format.
NET-PORT.n.IP-ADR	208.77.188.166 or 2001:0db8:85a3:08d3:1319:8a2e:0370:7334	Internet Protocol (IP) logical address. IPv4 (32-bit) or IPv6 (128-bit) format.
NET-PORT.n.TOTAL-BANDWIDTH	0+	Bandwidth of the port in Kbps
NET-PORT.n.UTIL	0-100	Percentage of Network port utilization
NET-PORT.n.BYTES-SENT	0+	Number of bytes sent over the port
NET-PORT.n.BYTES-RECEIVED	0+	Number of bytes received over the port
NET-PORT.n.ERRORS	0+	Number of errors encountered on the port
* Note: At a rate of 2 messages per minute, this counter will overflow after 22 days.		

8.6 Real-Time Telemetry Data Messages

Telemetry Messages are data packages that contain spacecraft health and safety data. In most ground systems, a Telemetry Message is packaged by the spacecraft and sent to the ground station. The ground station performs air-to-ground quality checking, adds ground station information, and routes the data to the ground system. Archive retrieval systems, simulators, and data generators can also provide Telemetry Messages in replay, simulation, and test modes.

Additionally, the data can be published "as is" (Raw), or after a degree or level of processing (Processed). The latter could involve a number of data scrubbing techniques plus conversion from binary values to engineering units (EU). Table **Error! No text of specified style in document.**-27 below lists the messages that have been defined to transport the various kinds of telemetry data.

Table Error! No text of specified style in document.**-27. Telemetry Messages**

Telemetry Message	Data Form (Level)	Data Format
Telemetry CCSDS Packet	Raw	CCSDS Packet
Telemetry CCSDS Frame	Raw	CCSDS Frame
Telemetry TDM Frame	Raw	TDM Frame
Processed Telemetry Data	Processed (Converted)	Data samples for one frame organized by mnemonic

The CCSDS Frame and CCSDS Packet are industry standard formats. Time Division Multiplexing (TDM) is the method and format for sending multiple digital signals along a single telecommunications transmission. Specific decommutation instructions for the frames and packets are documented in other resources.

Note: Additional telemetry message contents may be added as necessary.

8.6.1 Telemetry CCSDS Packet Message

The Telemetry CCSDS Packet Message is used for transferring CCSDS telemetry packets. The FORMAT field of the Telemetry Message Contents is set to CCSDSPKT. The Telemetry Message Contents consists of the raw CCSDS telemetry packet, the time of the packet and the quality of the data. The STREAM-MODE field is used to classify the kind or source of the CCSDS packets. The mode of the telemetry data can be real-time (RT), replay (RPY), simulation (SIM), or test (TEST).

Table Error! No text of specified style in document.**-28. Telemetry CCSDS Packet Message Summary**

Sender	A C2MS compliant application such as a ground station, simulator, archive component, or front-end processor
Senders Intended Usage	Publish
Receiver	Telemetry Decommutation System, Archive System, Trending System, Expert System
Receivers Intended Usage	Subscribe
What	Spacecraft health and safety data to be decommutated and/ or archived

When	As needed but usually dependent on data rate and/or replay rate

Example

- Spacecraft health and safety data sent from ground station to ground system

8.6.1.1 Telemetry CCSDS Packet Message Subjects

Table Error! No text of specified style in document.**-29. Telemetry CCSDS Packet Message Subject Naming**

	Subject Standard	**Domain Elements**		**Mission Elements**			**Message Elements**		***Miscellaneous Elements***				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	TLMPKT	Component of publisher	Stream-mode	Virtual Channel ID	AP ID	Collection Point
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMPKT	SATSIM	SIM	2	1	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMPKT	TFEP	RT	1	2	
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	*	*	SAT1	MSG	TLMPKT	*	RT	*	*	

Table Error! No text of specified style in document.**-30. Properties of the *Miscellaneous Elements* for the Telemetry CCSDS Packet Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Required	Identifies stream as real-time, playback, simulator, or test.	STREAM-MODE
ME3	Required	Virtual Channel ID	VCID from msg content; or from header portion of CCSDS frame
ME4	Required	AP ID – identifies a particular subsystem on the spacecraft	From header portion of data stream
ME5	Optional	Point on ground system where data was captured	COLLECTION-POINT

Example for Publisher / Sender of Telemetry Messages

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMPKT.SATSIM.SIM.2.1
```

Example for Subscriber / Receiver of Telemetry Messages

```
C2MS.*.*.MSSN.*.*.MSG.TLMPKT.TFEP.RT.>

C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMPKT.TFEP.RT.2.1
```

8.6.1.2 Telemetry CCSDS Packet Message Header

The abbreviated Table **Error! No text of specified style in document.**-31. Telemetry CCSDS Packet Message Header below shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for Telemetry CCSDS Packet Message header.

Table Error! No text of specified style in document.**-31. Telemetry CCSDS Packet Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	TLMPKT	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.6.1.3 Telemetry CCSDS Packet Message Contents

Figure **Error! No text of specified style in document.**-6 below shows a UML object diagram of the Telemetry CCSDS Packet Message with its required and optional fields.

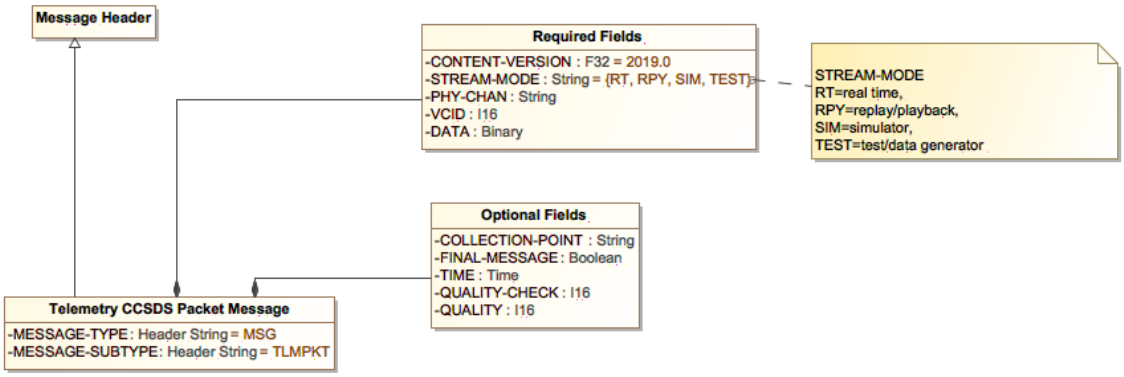

Figure Error! No text of specified style in document.**-6. Telemetry CCSDS Packet Message Diagram**

Table **Error! No text of specified style in document.**-32. Telemetry CCSDS Packet Message Additional Information below describes additional field names, values, and notes for the Telemetry CCSDS Packet Message.

Table Error! No text of specified style in document.**-32. Telemetry CCSDS Packet Message Additional Information**

Field Name	Value		Notes
CONTENT-VERSION	2019		Version Number for this message content description
COLLECTION-POINT			Receiver, device, point, path, etc. where data was received. Used to distinguish data simultaneously received at multiple collection points.
STREAM-MODE	**Value**	**Description**	Identifies the mode of the stream of telemetry as either Real-time, Replay, Simulator, or Test.
	RT	Real-time	
	RPY	Replay	
	SIM	Simulator	
	TEST	Test/Data Generator	
FINAL-MESSAGE	**Value**	**Description**	When true (and known, especially for replay data), indicates the last message in the stream.
	0	No/False	
	1	Yes/True	
TIME			Time of packet, usually ground receipt time
PHY-CHAN			Physical channel on which data is received
VCID			Virtual Channel ID
QUALITY-CHECK	**Value**	**Description**	Indicates quality checking was performed for reason indicated.
	Bit 0	Partial Packet	
QUALITY	**Value**	**Description**	Indicates quality state if checking was performed
	Bit 0	Partial Packet	
DATA			Raw telemetry data

8.6.2 Telemetry CCSDS Frame Message

The Telemetry CCSDS Frame Message is used to transfer CCSDS telemetry frames. The FORMAT field of the Telemetry Message Contents is set to CCSDSFRAME. The Telemetry Message Contents consists of the raw CCSDS telemetry frame, the time of the frame, quality checking performed, and the resulting quality of the data. The STREAM-MODE field is used to classify the kind or source of the CCSDS frame.

A frame with a Frame Sync pattern at the front that includes Reed-Solomon check symbols is called a Coded Virtual Channel Data Unit (CVCDU) in the CCSDS documentation.

Table Error! No text of specified style in document.**-33. Telemetry CCSDS Frame Message Summary**

Sender	A C2MS compliant application such as a ground station, simulator, archive component, or front-end processor
Senders Intended Usage	Publish
Receiver	Telemetry Decommutation System, Archive System, Trending System, Expert System
Receivers Intended Usage	Subscribe
What	Spacecraft health and safety data to be decommutated and/ or archived
When	As needed but usually dependent on data rate and/or replay rate

Example

- Spacecraft health and safety data sent from ground station to ground system

8.6.2.1 Telemetry CCSDS Frame Message Subjects

Table Error! No text of specified style in document.**-34. Telemetry CCSDS Frame Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	TLMFRAME	Component of publisher	Stream-mode	Virtual Channel ID	AP ID	Collection Point
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMFRAME	SATSIM	SIM	2	1	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMFRAME	TFEP	RT	1	2	
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	*	*	SAT1	MSG	TLMFRAME	*	RT	*	*	

Table Error! No text of specified style in document.**-35. Properties of the *Miscellaneous Elements* for the Telemetry CCSDS Frame Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Required	Identifies stream as real-time, playback, simulator, or test.	STREAM-MODE
ME3	Required	Virtual Channel ID	VCID from msg content; or from header portion of CCSDS frame
ME4	Optional	AP ID – identifies a particular subsystem on the spacecraft	From header portion of data stream
ME5	Optional	Point on ground system where data was captured	COLLECTION-POINT

Example for Publisher / Sender of Telemetry Messages

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMFRAME.SATSIM.SIM.2.1
```

Example for Subscriber / Receiver of Telemetry Messages

```
C2MS.*.*.MSSN.*.*.MSG.TLMFRAME.TFEP.RT.>

C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMFRAME.TFEP.RT.2.1
```

8.6.2.2 Telemetry CCSDS Frame Message Header

The abbreviated Table **Error! No text of specified style in document.**-36. Telemetry CCSDS Frame Message Header below shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for Telemetry CCSDS Frame Message header.

Table Error! No text of specified style in document.**-36. Telemetry CCSDS Frame Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	TLMFRAME	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.6.2.3 Telemetry CCSDS Frame Message Contents

Figure **Error! No text of specified style in document.**-7 below shows a UML object diagram of the Telemetry CCSDS Frame Message with its required and optional fields.

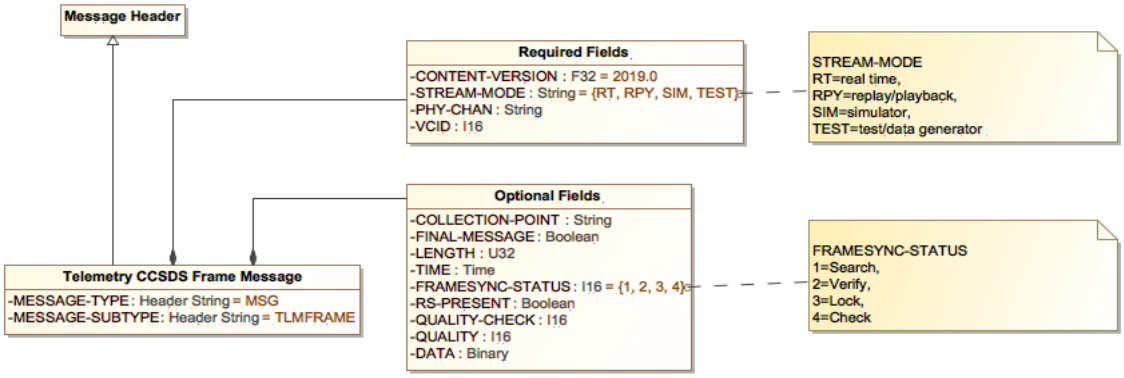

Figure Error! No text of specified style in document.**-7. Telemetry CCSDS Frame Message Diagram**

Table **Error! No text of specified style in document.**-37. Telemetry CCSDS Frame Message Additional Information below describes additional field names, values, and notes for the Telemetry CCSDS Frame Message.

Table Error! No text of specified style in document.**-37. Telemetry CCSDS Frame Message Additional Information**

Field Name	Value/Description		Notes
CONTENT-VERSION	2019		Version Number for this message content description
COLLECTION-POINT			Receiver, device, point, path, etc. where data was received. Used to distinguish data simultaneously received at multiple collection points.
STREAM-MODE	**Value**	**Description**	Identifies the kind or source of the stream of telemetry as either Real-time, Replay, Simulator, or Test.
	RT	Real-time	
	RPY	Replay	
	SIM	Simulator	
	TEST	Test/Data Generator	
FINAL-MESSAGE	**Value**	**Description**	When true (and known, especially for replay data), indicates the last message in the stream.
	0	No/False	
	1	Yes/True	
LENGTH	Bytes		Length of frame
TIME			Time of frame, usually ground receipt time
PHY-CHAN			Physical channel on which data is received
VCID			Virtual Channel ID
FRAMESYNC-STATUS	**Value**	**Description**	State of frame synchronization from equipment when frame is ingested
	1	Search	
	2	Verify	
	3	Lock	
	4	Check	
RS-PRESENT	**Value**	**Description**	Indicates if the Reed-Solomon codes are present in the data.
	0	No/False	

Field Name	Value/Description		Notes
	1	Yes/True	
QUALITY-CHECK	**Value**	**Description**	Indicates quality checking performed, if applicable. If the bit is set the particular quality check was performed.
	Bit 0	CRC Quality Check	
	Bit 1	Reed-Solomon Quality Check	
	Bit 2	Turbo Code Quality Check	
QUALITY	**Value**	**Description**	Indicates quality state if checking is performed. If the bit is set the particular quality check failed.
	Bit 0	CRC Quality State	
	Bit 1	Reed-Solomon Quality State	
	Bit 2	Turbo Code Quality State	
DATA			Raw telemetry data

8.6.3 Telemetry TDM Frame Message

The Telemetry Time-Division Multiplexing (TDM) Frame Message is used to transfer TDM frames. The Telemetry Message Contents simply consists of the raw TDM frame, length of the frame, and the time of the frame. The STREAM-MODE field is used to classify the kind or source of the TDM frame.

Table Error! No text of specified style in document.**-38. Telemetry TDM Frame Message Summary**

Sender	A C2MS compliant application such as a ground station, simulator, archive component, or front-end processor
Senders Intended Usage	Publish
Receiver	Telemetry Decommutation System, Archive System, Trending System, Expert System
Receivers Intended Usage	Subscribe
What	Spacecraft health and safety data to be decommutated and/ or archived
When	As needed but usually dependent on data rate and/or replay rate

Example

- Spacecraft health and safety data sent from ground station to ground system

8.6.3.1 Telemetry TDM Frame Message Subjects

Table Error! No text of specified style in document.**-39. Telemetry TDM Frame Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	TLMTDM	Component of publisher	Stream-mode	Virtual Channel ID	AP ID	Collection Point
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMTDM	SATSIM	SIM	2	1	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMTDM	TFEP	RT	1	2	
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	*	*	SAT1	MSG	TLMTDM	*	RT	*	*	

Table Error! No text of specified style in document.**-40. Properties of the *Miscellaneous Elements* for the Telemetry TDM Frame Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Required	Identifies stream as real-time, playback, simulator, or test.	STREAM-MODE
ME3	Required for all CCSDS	Virtual Channel ID	VCID from msg content; or from header portion of CCSDS frame
ME4	Required for CCSDS Packet only	AP ID – identifies a particular subsystem on the spacecraft	From header portion of data stream
ME5	Optional	Point on ground system where data was captured	COLLECTION-POINT

Example for Publisher / Sender of Telemetry Messages

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMTDM.SATSIM.SIM.2.1
```

Example for Subscriber / Receiver of Telemetry Messages

```
C2MS.*.*.MSSN.*.*.MSG.TLMTDM.TFEP.RT.>

C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMTDM.TFEP.RT.2.1
```

8.6.3.2 Telemetry TDM Frame Message Header

The abbreviated Table Error! No text of specified **style in document.**-41. Telemetry TDM Frame Message Header shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for Telemetry TDM Frame Message header.

Table Error! No text of specified style in document.**-41. Telemetry TDM Frame Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	TLMTDM	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.6.3.3 Telemetry TDM Frame Message Contents

Figure **Error! No text of specified style in document.**-8 below shows a UML object diagram of the Telemetry TDM Frame Message with its required and optional fields.

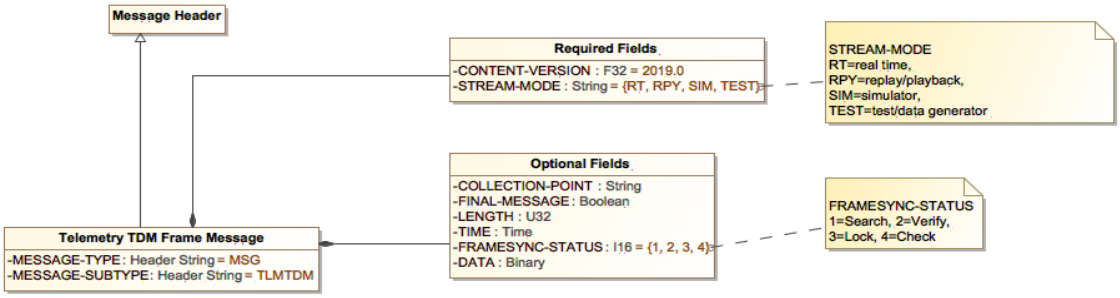

Figure Error! No text of specified style in document.**-8. Telemetry TDM Frame Message Diagram**

Table **Error! No text of specified style in document.**-42. Telemetry TDM Frame Message Additional Information below describes additional field names, values, and notes for the Telemetry TDM Frame Message.

Table Error! No text of specified style in document.**-42. Telemetry TDM Frame Message Additional Information**

Field Name	Value/Description		Notes
CONTENT-VERSION	2019		Version Number for this message content description
COLLECTION-POINT			Receiver, device, point, path, etc. where data was received. Used to distinguish data simultaneously received at multiple collection points.
STREAM-MODE	**Value**	**Description**	Identifies the kind or source of the stream of telemetry as either Real-time, Replay, Simulator, or Test.
	RT	Real-time	
	RPY	Replay	
	SIM	Simulator	
	TEST	Test/Data Generator	
FINAL-MESSAGE	**Value**	**Description**	When true (and known, especially for replay data), indicates the last message in the stream.
	0	No/False	
	1	Yes/True	
LENGTH	Bytes		Length of frame
TIME			Time of frame, usually ground receipt time
FRAMESYNC-STATUS	**Value**	**Description**	State of frame synchronization from equipment when frame is ingested
	1	Search	
	2	Verify	
	3	Lock	
	4	Check	
DATA			Raw telemetry data

8.6.4 Telemetry Processed Frame Message

The Telemetry Processed Frame Message is a hybrid between the unprocessed (raw) Telemetry Message and the Mnemonic Value Data Message. It contains both raw and converted data for a frame of telemetry data that is organized in the message by mnemonic. Thus, it is frame-based as are the telemetry messages, but mnemonic-organized as are the Mnemonic Value Data Messages. It serves to provide all the telemetry data in a raw and processed format. Therefore, many consumers could be provided with a substantial amount of data without needing to specifically request a custom selected mnemonic data set.

When this message is published is to be determined by the mission or data provider. It could be published "alongside" or in accordance with the raw telemetry data messages or by itself. Also, it could be published automatically or only by request, as a replay.

The STREAM-MODE field is used to classify the kind or source of the frames (or packets). The mode of the telemetry data can be real-time (RT), replay (RPY), simulation (SIM), or test (TEST).

Table Error! No text of specified style in document.**-43. Telemetry Processed Frame Message Summary**

Sender	A C2MS compliant application such as a ground station, simulator, archive component, or front-end processor
Senders Intended Usage	Publish
Receiver	Telemetry Decommutation System, Archive System, Trending System, Expert System
Receivers Intended Usage	Subscribe
What	Spacecraft health and safety data to be decommutated and/ or archived
When	As needed but usually dependent on data rate and/or replay rate

Example

- Spacecraft health and safety data sent from ground station to ground system

8.6.4.1 Telemetry Processed Frame Message Subjects

Table Error! No text of specified style in document.**-44. Telemetry Processed Frame Message Subject Naming**

	Subject Standard	Domain Elements		Mission Elements			Message Elements		*Miscellaneous Elements*				
Subject Element	**Speci-fication**	**DOMAIN1**	**DOMAIN2**	**MISSION**	**CONST**	**SAT**	**TYP**	**SUBTYP**	***ME1***	***ME2***	***ME3***	***ME4***	***ME5***
Subject Content	C2MS	[domain 1 – system specific]	[domain 2 – system specific]	[mission]	[constell ation]	[sat]	MSG	TLMPROC	Component of publisher	Stream-mode	Virtual Channel ID	AP ID	Collection Point
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMPROC	SATSIM	SIM	2	1	
Example for Publisher / Sender	C2MS	DOM1	DOM2	MSSN	CNS1	SAT1	MSG	TLMPROC	TFEP	RT	1	2	
Example for Subscriber / Receiver	C2MS	DOM1	DOM2	*	*	SAT1	MSG	TLMPROC	*	RT	*	*	

Table Error! No text of specified style in document.**-45. Properties of the *Miscellaneous Elements* for the Telemetry Processed Frame Message**

Miscellaneous Element	Required / Optional	Description	Field in Msg, if applicable
ME1	Required	Component name of Publisher	COMPONENT from header
ME2	Required	Identifies stream as real-time, playback, simulator, or test.	STREAM-MODE
ME3	Optional	Virtual Channel ID	VCID from msg content
ME4	Optional	AP ID – identifies a particular subsystem on the spacecraft	From header portion of data stream
ME5	Optional	Point on ground system where data was captured	COLLECTION-POINT

Example for Publisher / Sender of Telemetry Messages

```
C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMPROC.SATSIM.SIM.2.1
```

Example for Subscriber / Receiver of Telemetry Messages

```
C2MS.*.*.MSSN.*.*.MSG.TLMPROC.TFEP.RT.>

C2MS.DOM1.DOM2.MSSN.CNS1.SAT1.MSG.TLMPROC.TFEP.RT.2.1
```

8.6.4.2 Telemetry Processed Frame Message Header

The abbreviated Table **Error! No text of specified style in document.**-46. Telemetry Processed Frame Message Header shows the required values of the MESSAGE-TYPE and MESSAGE-SUBTYPE fields for Telemetry Processed Frame Message header.

Table Error! No text of specified style in document.**-46. Telemetry Processed Frame Message Header**

Field Name	Value	Notes
HEADER-VERSION	2019	Version Number for this message description
MESSAGE-TYPE	MSG	Message type identifier: REQ, RESP, or MSG
MESSAGE-SUBTYPE	TLMPROC	Unique message identifier, fixed for C2MS Standard Messages
More … Please refer to **Error! Reference source not found. Error! Reference source not found.** for a complete definition.	…	…

8.6.4.3 Telemetry Processed Frame Message Contents

Figure **Error! No text of specified style in document.**-9 below shows a UML object diagram of the Telemetry Processed Frame Message with its required, optional, and dependent fields.

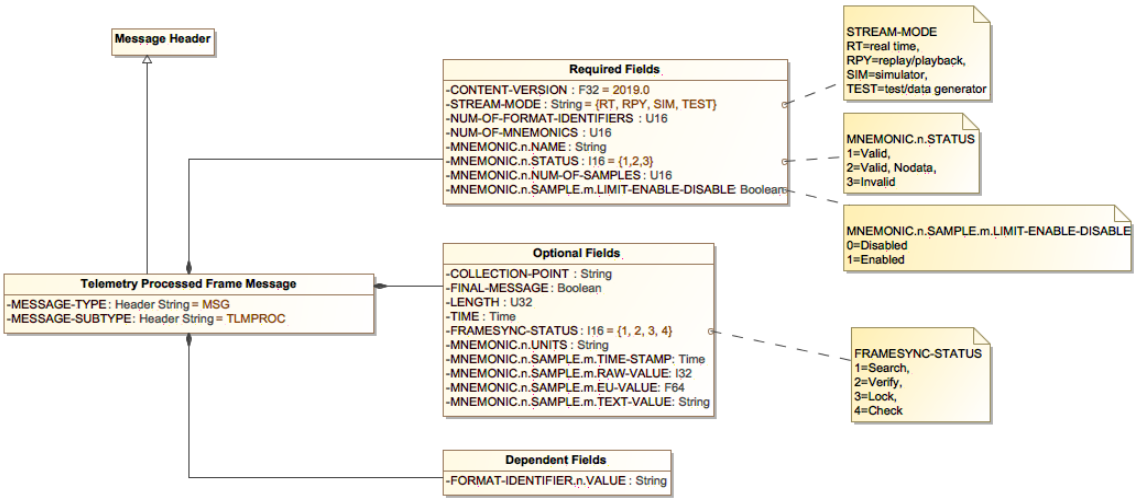

Figure Error! No text of specified style in document.**-9. Telemetry Processed Frame Message Diagram**

Table **Error! No text of specified style in document.**-47. Telemetry Processed Frame Message Additional Information describes additional field names, values, and notes for the Telemetry Processed Frame Message Additional Information

Table Error! No text of specified style in document.**-47. Telemetry Processed Frame Message Additional Information**

Field Name	Value/Description		Notes
CONTENT-VERSION	2019		Version Number for this message content description
COLLECTION-POINT			Receiver, device, point, path, etc. where data was received. Used to distinguish data simultaneously received at multiple collection points.
STREAM-MODE	**Value**	**Description**	Identifies the kind or source of the stream of telemetry as either Real-time, Replay, Simulator, or Test.
	RT	Real-time	
	RPY	Replay	
	SIM	Simulator	
	TEST	Test/Data Generator	
FINAL-MESSAGE	**Value**	**Description**	When true (and known, especially for replay data), indicates the last message in the stream.
	0	No/False	
	1	Yes/True	
LENGTH	Bytes		Length of frame
TIME			Time of frame, usually ground receipt time
FRAMESYNC-STATUS	**Value**	**Description**	State of frame synchronization from equipment when frame is ingested
	1	Search	
	2	Verify	
	3	Lock	
	4	Check	

Field Name	Value/Description		Notes
NUM-OF-FORMAT-IDENTIFERS	0+		Number of fields used to identify the frames (e.g. TDM major/minor frames would have a value of 2). Zero is only permissible for vehicles with one telemetry format with a single type of frame.
FORMAT-IDENTIFIER.n.VALUE			Value of the nth field used to identify the telemetry. If the message is used with XTCE, this is the nth comparison in a comparison list in the restriction criteria in an XTCE container.
NUM-OF-MNEMONICS	1+		Total number of mnemonics in this message
MNEMONIC.n.NAME			Name of the 'nth' mnemonic
MNEMONIC.n.STATUS	**Value**	**Description**	Status of the 'nth' mnemonic: valid mnemonic, or valid mnemonic with no data, or invalid mnemonic
	1	Valid	
	2	Valid, Nodata	
	3	Invalid	
MNEMONIC.n.UNITS			Units associated with the raw value converted to engineering units for the 'nth' mnemonic
MNEMONIC.n.NUM-OF-SAMPLES			Number of data samples for the 'nth' mnemonic. This value should equal the number of times the mnemonic appears in the telemetry frame (e.g. will be greater than 1 for super-commutated telemetry points.
MNEMONIC.n.SAMPLE .m.TIME-STAMP			Time stamp for the 'nth' data sample of the 'nth' mnemonic
MNEMONIC.n.SAMPLE .m.RAW-VALUE			Raw value for the 'nth' data sample of the 'nth' mnemonic
MNEMONIC.n.SAMPLE .m.EU-VALUE			Raw value converted to Engineering Units if engineering units conversion is present for the 'nth' data sample of the 'nth' mnemonic
MNEMONIC.n.SAMPLE .m.TEXT-VALUE			Raw value converted to a text string if text conversion is present for the 'nth' data sample of the 'nth' mnemonic
MNEMONIC.n.SAMPLE .m.LIMIT-ENABLE-DISABLE	**Value**	**Description**	Indicates the limit checking state for the 'nth' data sample of the 'nth' mnemonic
	0	Disabled	
	1	Enabled	