esoc

European Space Operations Centre

Robert-Bosch-Strasse 5

D-64293 Darmstadt

Germany

T +49 (0)6151 900

F +49 (0)6151 90495

www.esa.int

|  |
| --- |
|  |

OPS-SAT ICD Deviations

Approval

|  |  |
| --- | --- |
| Title OPS-SAT ICD Deviations | |
| Issue Number 1 | Revision Number 0 |
| Author José Luís Feiterinha, César Coelho, Dominik Marszk | Date 16/02/2017 |
| Approved By | Date of Approval |
|  |  |

Change Log

|  |  |  |  |
| --- | --- | --- | --- |
| Reason for change | Issue Nr. | Revision Number | Date |
|  |  |  |  |

Change Record

|  |  |  |  |
| --- | --- | --- | --- |
| Issue Number  1 | Revision Number  0 | | |
| Reason for change | Date | Pages | Paragraph(s) |
|  |  |  |  |

Distribution

|  |
| --- |
| Name/Organisational Unit |
|  |

Table of contents:

1 Introduction 4

2 MAL-SPP Transport Binding 4

3 Binary Encoding 4

4 Mapping Configuration Parameters (MCP) 5

5 Comparison of PoC SGICD with Nanomind OBSW ICD 6

5.1 Platform and Standard Services 7

5.1.1 FBO area 7

5.1.2 CPDU service (Area OPSSAT\_STD) 7

5.1.3 MemoryManagement service (Area OPSSAT\_STD) 7

5.1.4 ScheduleManagement service (Area OPSSAT\_STD) 8

5.1.5 PacketStore service (Area OPSSAT\_STD) 8

5.1.6 FileManagementExt service (Area OPSSAT\_STD) 8

5.1.7 MIC service (Area OPSSAT\_PF) 8

5.1.8 Power service (Area OPSSAT\_PF) 8

5.1.9 Mode service (Area OPSSAT\_PF) 9

5.1.10 TimeManagement service (Area OPSSAT\_PF) 9

5.1.11 ExperimentWD service (Area OPSSAT\_PF) 9

5.1.12 GPS service (Area OPSSAT\_PF) 9

5.1.13 CoarseADCS service (Area OPSSAT\_PF) 9

5.1.14 AdvancedOBC service (Area OPSSAT\_PF) 9

5.2 COM and M&C services 10

5.2.1 Event service (Area COM) 10

5.2.2 Archive service (Area COM) 10

5.2.3 Activity Tracking service (Area COM) 10

5.2.4 Action service (Area MC) 10

5.2.5 Parameter service (Area MC) 11

5.2.6 Alert service (Area MC) 11

5.2.7 Check service (Area MC) 11

5.2.8 Statistics service (Area MC) 12

5.2.9 Aggregation service (Area MC) 12

5.2.10 Conversion service (Area MC) 12

5.2.11 Group service (Area MC) 12

6 MO Framework 12

6.1 MO Messages 12

6.2 Publish-Subscribe interaction pattern 13

7 Others 13

8 Additional Considerations 13

9 Appendix 13

9.1 PUB-SUB Pattern Alternative Message Sequence 13

9.2 Automatic publish of aggregations 14

# Introduction

The document was produced to have a concise record of the deviations from the original Space to Ground ICD provided by the Proof of Concept for ESA’s OPS-SAT mission.

Furthermore, the document has been extended to capture the deviations regarding the applicable CCSDS Standards.

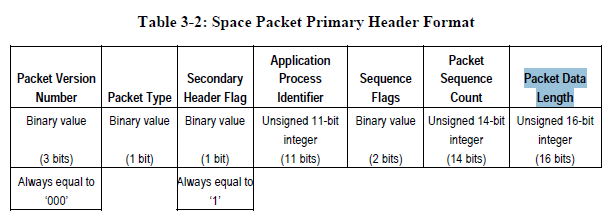
Complementing, the document holds the Mapping Configuration Parameters (MCP) of the MAL-SPP Binding. This is the so called “out-of-band agreement” between a consumer and a provider that uses this binding.

# MAL-SPP Transport Binding

The MAL-SPP Transport Binding used in OPS-SAT does not completely follow the “CCSDS Mission Operations - MAL-SPP Transport Binding and Binary Encoding” book. [insert reference]

There are 2 octets being appended at the end of the Space Packet containing the CRC checksum of the message.

Consequently, the Packet Data Length of the Space Packet Primary Header is incremented by 2:



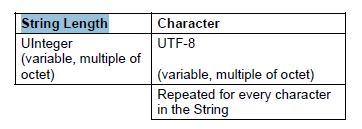
# Binary Encoding

The Binary Encoding used in OPS-SAT does not completely follow the “CCSDS Mission Operations - MAL-SPP Transport Binding and Binary Encoding” book. [insert reference]

The following MAL Data Types do not follow the Binary Encoding prescriptions: MAL::Identifier, MAL:: String, MAL::URI.

The String length is not being encoded as UInteger according to requirement 5.21.3:

***5.21.3*** *The field ‘String Length’ shall be encoded as a UInteger (see 5.18).*



In OPS-SAT, the Binary Encoding does not follow this specification and instead the String Length field is encoded as an unsigned 16-bit integer.

According to GMV’s Design Document, the reason for this change is to conform with the S2K data generation/processing. Further assessment should be taken to guarantee that this is still valid.

Note: UInteger is encoded as an unsigned 32-bit integer according to requirement 5.18.2:

***5.18.2*** *If the MCP VARINT\_SUPPORTED is FALSE, then a MAL::UInteger shall be encoded as an Unsigned 32-bit Integer (see 5.25).*

Unsigned 16-bit integer corresponds to MAL::UShort type (per 5.16.2):

***5.16.2*** *If the MCP VARINT\_SUPPORTED is FALSE, then a MAL::UShort shall be*

*encoded as an Unsigned 16-bit Integer (see 5.25).*



# Mapping Configuration Parameters (MCP)

MCP’s values should be agreed upon and put into OBSW ICD.

The MCP’s used in OPS-SAT are following:

| Parameter Name | Type | Description | Value |
| --- | --- | --- | --- |
| AUTHENTICATION\_ID | MAL:Blob | Value to be assigned to the MAL header field ‘Authentication Id’ if the QoS property AUTHENTICATION\_ID\_FLAG is FALSE |  |
| DOMAIN | List<MAL::Identifier> | Value to be assigned to the MAL header field ‘Domain’ if the QoS property DOMAIN\_FLAG is FALSE |  |
| DURATION\_CODE\_FORMAT | MAL::Blob | P-Field of the CUC Time Code Format to be applied to the MAL::Duration elements | 00101111 |
| DURATION\_UNIT | MAL::String | Name of the time unit used by the Time Code Format for MAL::Duration, e.g., second, millisecond. | seconds |
| FINE\_TIME\_CODE\_FORMAT | MAL::Blob | P-Field of the Time Code Format to be applied to the MAL::FineTime elements | 00101111 |
| FINE\_TIME\_EPOCH | MAL::String | Epoch in ISO-8601 format to be applied to the MAL::FineTime elements | 2000-01-01T00:00:00.000 |
| FINE\_TIME\_EPOCH\_TIMESCALE | MAL::String | Name of the time scale for the MAL::FineTime epoch, either “TAI” or “UTC”. | TAI |
| FINE\_TIME\_UNIT | MAL::String | Name of the time unit used by the Time Code Format for MAL::FineTime, e.g., second, millisecond. | Seconds |
| NETWORK\_ZONE | MAL::Identifier | Value to be assigned to the MAL header field ‘Network Zone’ if NETWORK\_ZONE\_FLAG is FALSE |  |
| PRIORITY | MAL::UInteger | Value to be assigned to the MAL header field ‘Priority’ if PRIORITY\_FLAG is FALSE |  |
| TIME\_CODE\_FORMAT | MAL::Blob | P-Field of the Time Code Format to be applied to the MAL::Time elements | 00101111 |
| TIME\_EPOCH | MAL::String | Epoch in ISO-8601 format to be applied to the MAL::Time elements | 2000-01-01T00:00:00.000 |
| TIME\_EPOCH\_TIMESCALE | MAL::String | Name of the time scale for the MAL::Time epoch, either “TAI” or “UTC”. | TAI |
| TIME\_UNIT | MAL::String | Name of the time unit used by the Time Code Format for MAL::Time, e.g., second, millisecond. | seconds |
| SESSION\_NAME | MAL::Identifier | Value to be assigned to the MAL header field ‘Session Name’ if SESSION\_NAME\_FLAG is FALSE |  |
| VARINT\_SUPPORTED | MAL::Boolean | Boolean indicating whether the ‘Varint’ encoding format is supported or not | false |
| SOURCE\_ID | MAL::String | Value to be assigned to the MAL header field ‘Source Id’ if SOURCE\_ID\_FLAG is FALSE |  |
| DESTINATION\_ID | MAL::String | Value to be assigned to the MAL header field ‘Destination Id’ if DESTINATION\_ID\_FLAG is FALSE |  |

# Comparison of PoC SGICD with Nanomind OBSW ICD

This document tries to provide a comparison of the changes performed to the initial SGICD for the OPS-SAT OBSW.

It is an attempt at a raw comparison, it does not perform any type of evaluation over the changes.

In some cases error messages were changed (some new, some removed). This is expected because the SG ICD had to make several conjectures (e.g. regarding list of payloads) for this reason error messages were not compared.

**Nomenclature**

*OBSW ICD* - In this document Nanomind OBSW (OnBoard SoftWare) ICD provided as part of the CDR datapack.

*SGICD* – This refers to the Space-to-Ground ICD provided with the OPS-SAT Proof-of-concept.

**Remarks**

Past & Present references shall be interpreted as SGICD (past) and OBSW ICD (present). Example: if the text states "a new operation has been added", it means that this operation was not in the SGICD but has since been added to the OBSW ICD.

Both SGICD and OBSW ICD specify (among others) the following reference documents:

(01/2014 in SGICD and in OBSW ICD)

The COM and M&C services were not defined in SGICD. Therefore, the referenced books were used for the comparison with the OBSW.

## Platform and Standard Services

This chapter goes through each service of the SGICD and compares it to the respective OBSW implementation. It tries to identify the divergence points and when possible it will add technical information regarding those changes.

### FBO area

*Not applicable to OBSW*

This area has two services in the ICD (FileManagement and FileTransfer). This is not applicable to the OBSW since the FBO OBSW application will not run on the NanoMind.

### CPDU service (Area OPSSAT\_STD)

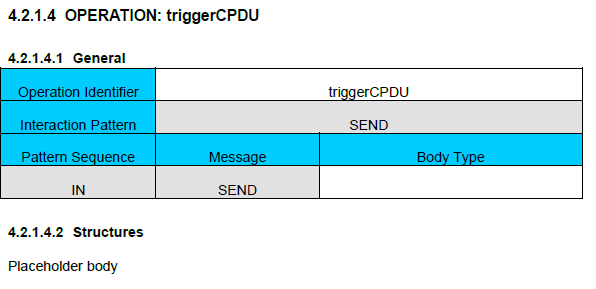
The CPDU service provided low level operations:

Figure 1 - Sample operation showing that body type and structures were only placeholders

* distributeOnOff
* registerLoad
* triggerCPDU

The service in the SGICD was *only a placeholder* and none of the operations was truly defined, there was no body type of structure defined for any of the operations.

It was removed in the OBSW ICD.

### MemoryManagement service (Area OPSSAT\_STD)

This service is for memory management functionality.

On the SGICD there were two operations defined, the OBSW ICD adds one more:

* loadMemory (kept as is on the OBSW ICD)
* dumpMemory (removed body type of the [final] response, all information is delivered via update messages)
* checkMemory operation was introduced to allow checksums[[1]](#footnote-1).

Other differences:

* MemoryDevice enumeration was changed to reflect the real devices.
* Memoryblock ‘CRC’ enumeration field name was renamed to ‘checksum’.

### ScheduleManagement service (Area OPSSAT\_STD)

*GMV has already agreed to revert any changes and to respect the SGICD. For this reason this service is not covered in this document.*

### PacketStore service (Area OPSSAT\_STD)

Both operations are kept:

* downlinkPacketStoreContent, changed from PROGRESS to SUBMIT interaction pattern.
* stopPacketStoreDownlink

Note: The solution proposed by GMV is to have the *downlinkPacketStoreContent* initiate the downlink of the packet-store over a separate virtual channel. This is more in line with how other ESA missions work and shall require less (or none at all) changes to SCOS than what was proposed in the SGICD which require each dumped packet to be embed in a MAL message.

### FileManagementExt service (Area OPSSAT\_STD)

*Removed in the OBSW ICD. Not applicable to OBSW.*

### MIC service (Area OPSSAT\_PF)

Service was renamed to ‘BusSwitches’, both Area and Service number values were kept.

This service has a single operation setAccessMask which was kept, however the body type has changed.

* The APID field has been added so that now access masks are per APID.
* The *devices* and *accessAllowed* body parts have been changed to be lists.

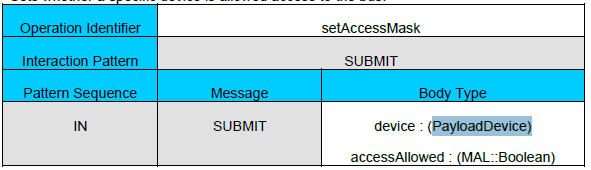


Figure 2 - OBSW ICD setAccessMask operation

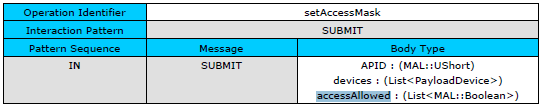


Figure 3 - SGICD setAccessMask operation

Both versions should work with SCOS.

PayloadDevice enumeration has been updated to reflect the real devices.

### Power service (Area OPSSAT\_PF)

All operations were kept:

* powerOffSBandTX, no changes
* powerOnSBandTX, no changes
* powerOffSBandRX, no changes
* powerOnSBandRX, no changes
* setPowerState, OBSW ICD replaced values with lists (same as for setAccessMask on 5.1.7 above)

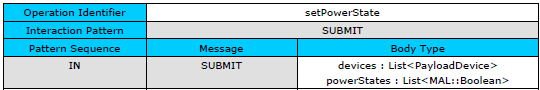


Figure 4 - OBSW ICD setPowerState

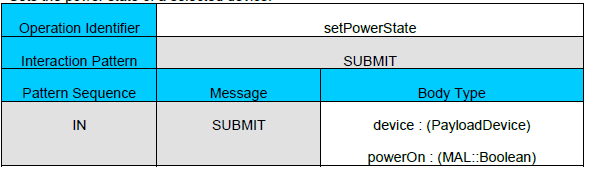


Figure 5 - SGICD setPowerState

### Mode service (Area OPSSAT\_PF)

No changes on operations.

SystemMode enumeration was updated: Idle and Extended safe modes have been removed.

### TimeManagement service (Area OPSSAT\_PF)

All operations were kept:

* useAutomaticTime, no changes
* getTimeMode, no changes
* applyManualTimeShift, no changes
* setManualTime, resolution changed from Picoseconds (MAL:FineTime) to Miliseconds (MAL:Time).

### ExperimentWD service (Area OPSSAT\_PF)

No changes.

### GPS service (Area OPSSAT\_PF)

SGICD provided only an empty placeholder (no operations defined).

One service operation has been defined on the OBSW ICD:

* getGPSData - a PROGRESS operation allowing to issue a NMEA command to the GPS receiver and receive a response.

### CoarseADCS service (Area OPSSAT\_PF)

SGICD provided only an empty placeholder (no operations defined).

Four SUBMIT operations have been defined:

* setADCSMode, allows to change the coarse ADCS mode.
* setNavTransMode, allows to switch the translational navigation flag of the coarse ADCS.
* uploadOrbitTLE, allows to upload new orbit TLE values to be used by the coarse ADCS.
* resetADCS, allows to stop and reinitialize the coarse ADCS component of the OBSW.

### AdvancedOBC service (Area OPSSAT\_PF)

Two operations were kept, one removed, three new were added:

* performHealthCheck – modified:
  + Has been extended to support multiple checks by introducing a “test identification number”
  + Test result set to Attribute (SGICD provided only a True/False results)[[2]](#footnote-2)
* loadSoMImage – modified:
  + removed SoM id, specifying which SoM the image would be uploaded to,
  + imageID has been renamed to imageNumber.
* uploadOrbitTLE – removed (moved to CoarseADCS service).
* resetOBSW – added, resets OBSW.
* readI2CPayload – added, allows to read (receive) data from the I2C payload device.
* writeI2CPayload – added, allows to write (send) data to the I2C payload device.

## COM and M&C services

This chapter compares the OBSW ICD against the CCSDS MO Common Object Model and Monitoring and Control Services books (It uses the books CCSDS 521.1-B-1 and 522x1r3 respectively).

### Event service (Area COM)

This service has only a single operation ‘monitorEvent’ which did not change.

Note:

For OPS-SAT there is an out-of-band agreement that the ground segment is by default subscribed to all events. So, although the operation is provided as-is in the ICD, its use-case from Ground Segment perspective is limited to listening to the published messages (see 6.2 for information). This behaviour should be described in the ICD.

### Archive service (Area COM)

*Not defined in SGICD. Not implemented in OBSW.*

### Activity Tracking service (Area COM)

No changes.

### Action service (Area MC)

Not defined in SGICD. The OBSW ICD is out of date considering current agreements.

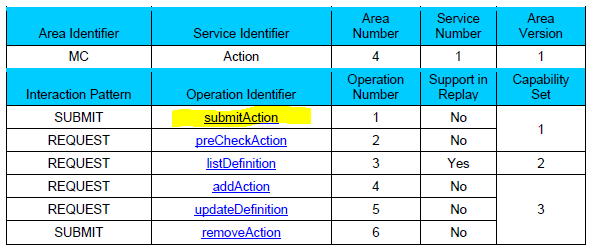


Figure 6 - List of operations available for the Action service as per MO MC book Nov 2016 (CESG approval)

a

Due to issues found in the reference version of MC book, GMV has agreed to provide an implementation of the submitAction instruction which is compatible with the interim version of the MO MC book as of November 2016. This work is still on-going.

Action service operations:

* submitAction is the only operation provided by the OBSW.
* preCheckAction is part of the same capability set, but not implemented.

OBSW ICD currently defines 47 actions. 8 actions return data in form of a non-periodic aggregation value. Actions with return values returned through Aggregation service[[3]](#footnote-5):

* ReadParameter
* PerformHealthCheck (OPSSAT\_PF.AdvancedOBC.performHealthCheck)
* GetGPSStatus
* ReadMemory (OPSSAT\_STD.MemoryManagement.loadMemory)
* CheckMemory (OPSSAT\_STD.MemoryManagement.checkMemory)
* GetTimeMode (OPSSAT\_PF.TimeManagement.getTimeMode)
* GetCurrentTime
* I2CWriteRead

The main impact this approach is that published aggregation values cannot be directly linked to the related action instance[[4]](#footnote-6). However, SCOS would not benefit from this since it will not provide any kind of automated response tracking through TC history (except for verification through MAL ACK and ActivityTracking).

### Parameter service (Area MC)

*Not defined in SGICD. Not implemented in OBSW.*

Instead, there is an onboard datapool of parameters accessible through Aggregation services (by referencing using parameter ID) and Actions:

* ReadParameter (returning the parameter value in an aggregation).
* WriteParameter.

The following parameter service data types are defined in the OBSW ICD:

* Validity enumeration – no changes
* ParameterDefinition, MAL::Composite - no changes
* ParameterValue, MAL::Composite - no changes
* ParameterConversion, MAL::Composite - no changes

### Alert service (Area MC)

Not defined in SGICD. Added in OBSW ICD.

The service and capability set 1 was implemented:

* enableGeneration, no changes

The core service functionality is based on generating Alert events published using COM Event service.

### Check service (Area MC)

Not defined in SGICD. Added in OBSW ICD.

The service and capability sets 1 (partially) and 3 were implemented.

* getSummaryReport, no changes
* enableCheck, no changes

The core service functionality is based on generating CheckTransition events published using COM Event service. It was implemented with no changes regarding the standard.

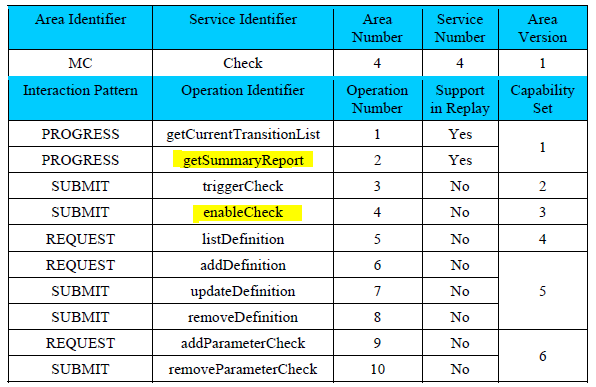


Figure 7 - Check service as is defined in the applicable MO MC book with the OBSW ICD implemented operations highlighted

a

### Statistics service (Area MC)

*Not defined in SGICD. Not implemented in OBSW.*

### Aggregation service (Area MC)

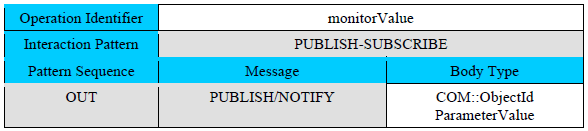


Figure 8 - monitorValue operation as it is defined in the applicable book

a

Implemented in OBSW:

* monitorValue, no changes, although operation description is missing in the OBSW ICD.
* getValue, will be added to the OBSW ICD as recently agreed.
* enableGeneration, no changes
* enableFilter, not implemented
* listDefinition, no changes
* addDefinition, no changes
* updateDefinition, no changes
* removeDefinition, no changes

Notable fact is that Aggregation service is supposed to be used for aggregating parameters only. It is extended in OBSW to return data from some Action instances. This is a non-standard behaviour not considered in M&C book.

### Conversion service (Area MC)

*Not defined in SGICD. Not implemented in OBSW.*

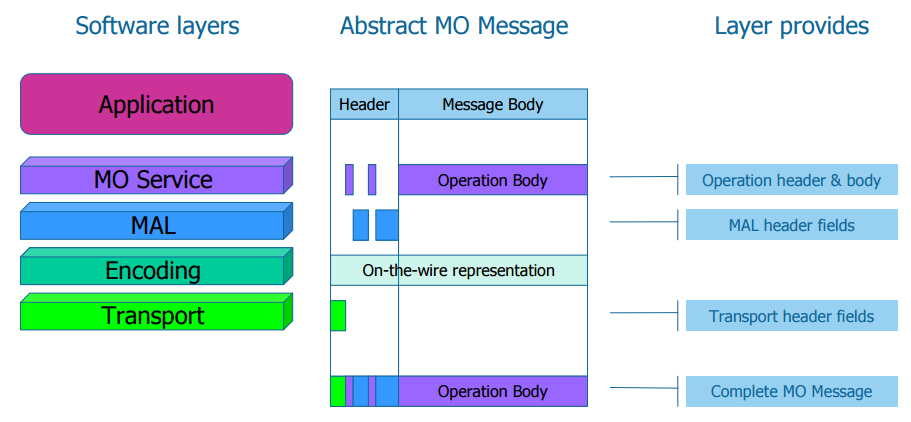
### Group service (Area MC)

*Not defined in SGICD. Not implemented in OBSW.*

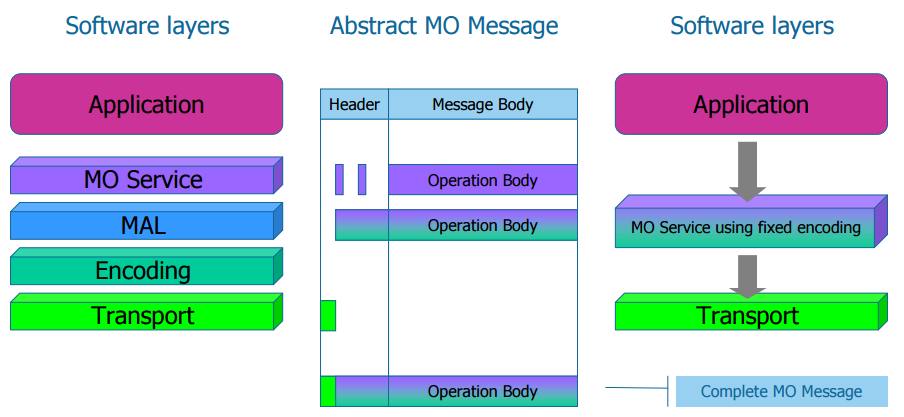
# MO Framework

## MO Messages

The different parts of the MO message are built by different layers. The picture below provides a visual representation of what parts of the MO message are set by their corresponding layer:



For optimizations and efficiency reasons, the OBSW is “flat” (in terms of MO layering) and it was not developed in a pure-MO stratified way:



This is possible to be done (and expected) because the encoding and the transport were fixed.

## Publish-Subscribe interaction pattern

The OBSW as it is implemented assumes that the ground segment *and only the ground segment* is capable of using Publish-Subscribe interaction patterns.

The OBSW does not include a “true” broker for the handling of registrations and distribution of messages from a provider to the consumers. There is a single “hard-coded” registration on the software that publishes messages to Ground.

This behaviour has been considered valid for Aggregation service as per CCSDS 522.1-B-0, November 2016 (see 9.2). In OBSW it was extended to all PUB-SUB operations in all services.

# Others

Information such as APIDs or URIs were not included in the SGICD.

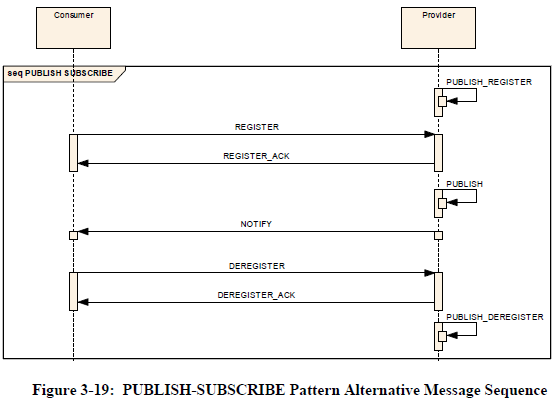
# Additional Considerations

The interfaces for the M&C services use the same version as the original M&C however the services are not exactly the same as the CCSDS standardized ones. This means that there might exist collisions when using the standardized ones in parallel with the tailored version. It is strongly encouraged to change the area version of the tailored version, as soon as possible, to a different number as this might cause considerate disruption and entanglement in later stages when both need to run in parallel.

# Appendix

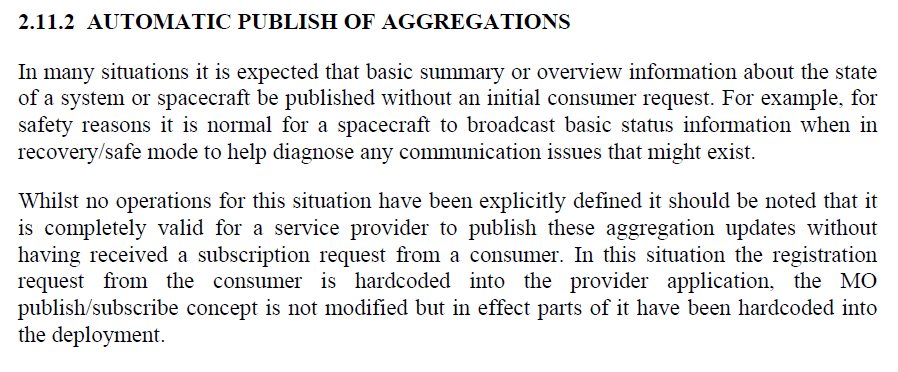
## PUB-SUB Pattern Alternative Message Sequence

See: *CCSDS 521.0-B-2, March 2013, Page 3-54*



## Automatic publish of aggregations

See: *CCSDS 522.1-B-0, November 2016, Page 2-11*



1. OBSW ICD: *“The checkMemory operation allows a consumer to compute the CRC checksum of an area of the target memory device using an absolute addresses or a file identifier. The CRC is reported in one response message.”* [↑](#footnote-ref-1)
2. Because of SCOS issues with processing MAL::Attribute, ESA has proposed to have this replaced by a status value or a composite. It will be replaced in further ICD version. [↑](#footnote-ref-2)
3. PerformHealthCheck, ReadMemory, CheckMemory and GetTimeMode provide the same functionality as equivalent platform or standard service operations (given in parenthesis as Area.Service.Operation). [↑](#footnote-ref-5)
4. An approach more in line with the MO philosophy would be removing this kind of actions altogether and implementing an equivalent dedicated service operations instead. [↑](#footnote-ref-6)