Appendix-C: OBSW Missionization in the SRDB with the RW's SEDS in the SRDB

This appendix aims to illustrate the conclusion that we obtained for RQ1 regarding our discussion on EDS works. With our SimpleSAT example we show a real-life scenario illustrating the use of data defined in SEDS in the context of SRDBs. This data concerns the reaction wheel of SimpleSAT and we aim to use it to override the generic telemetry provided by the OBSW. The overridden data allows the satellite operator to read the RW **mode** measure carried by the OBSW parameter.

1. We consider that our SRDB defines a product tree -composed of definition system elements- as shown in Figure-1.

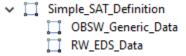


Figure-1: Definition system elements tree

2. The **OBSW_Generic_Data** definition system element contains the details of the OBSW parameter that has been defined by the OBSW to report RW data. The interesting details are shown in Figure-2.

In this figure our OBSW parameter is a 16-bit array containing two other 8-bit parameters. These 8-bit parameters aim to carry some measures related to the RW.



Figure-2: Details of the OBSW generic parameter

At this stage, it is important to highlight that if the onboard value of that parameter is downlinked to the ground, the satellite operator will not be able decode that value as it will be displayed in its raw format. Thus, it is necessary to missionize the generic OBSW parameter.

3. In order to missionize the OBSW parameter (RW_PARAMETER_1), it is first necessary to import the data that will be used for the missionization. The RW_EDS_Data definition system element shown in Figure-1 contains the data imported from the reaction wheel's SEDS discussed in our paper. We map the content of that SEDS (on the right) with the SRDB content (on the left) in Figure-3.

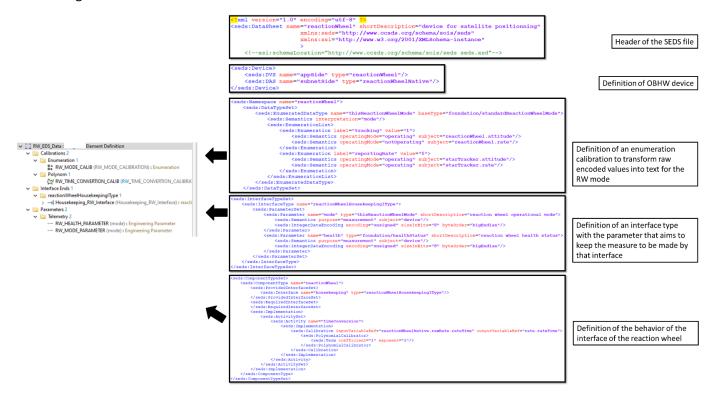


Figure-3: Mapping between SEDS file and its data imported in the SRDB

4. Besides the RW interface definition, if the SAVOIR-EDS that includes the mechanical view is imported into the SRDB, the result will be the comprehensive description of the RW in the SRDB as depicted in Figure-4.

```
xml version="1.0" encoding="utf-8
seds:DataSheet name="reactionWheel" shortDescription="device for satellite positionning
                                   xmlns:seds="http://www.ccsds.org/schema/sois/seds"
                                    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     <!--xsi:schemaLocation="http://www.ccsds.org/schema/sois/seds seds.xsd"-->
     <seds:Device>
             <seds:DVS name="appSide" type="reactionWheel"/>
             <seds:DAS name="subnetSide" type="reactionWheelNative"/>
                                                                                                                                                                            SAVOIR-EDS could include
      </seds:Device>
                                                                                                                                                                             other SEDS files
     <XInclude:include href="file:MechanicalView.eds" xpointer="element(/1/1)"/>
  Mechanical SEDS file
     //seds:Entrylist>
//seds:Entrylist>
//seds:Entrylist>
//seds:Entrylist>
//seds:Entrylist>
//seds:ContainerhataType>
seds:DataTypeSet>
seds:DataTypeSet>
seds:DataTypeSet>
/seds:ContainerhataContainerhataType>
seds:DataTypeSet>
/seds:GenericTypeSet>
/seds:GenericTypeSet>
/seds:GenericTypeSet>
/seds:GenericType shortDescription="The specific type for the Dimension parameter" name="DimensionMeasured" baseType=
"Precision percentTypesty"/>
/seds:GenericType shortDescription="The specific type for the Dimension parameter" name="DimensionMeasured" baseType=
"Dimension materProperty"/>
//seds:GenericTypeSet>
//seds:GenericTypeSet>
//seds:GenericTypeSet>
          runs:reframestroty

Coeds:Parameter shortDescription="The Dimension of the component as designed" name="DimensionMarginP" type="DimensionMarginP" readOnly="true",
Coeds:Parameter shortDescription="The Dimension of the component as designed" name="DimensionMeasuredP" type="DimensionMeasured" readOnly=
"true"/>

"true"/>

<
                            Calibrations 2
                                         RW_MODE_CALIB (RW_MODE_CALIBRATION) : Enumeration

→ Polynom 1
                                         RW_TIME_CONVERTION_CALIB (RW_TIME_CONVERTION_CALIBRATION) = 1 : Poly
                                    reactionWheelHousekeepinglType 1
                                      ♦ Input: Monitoring And Control Aspect Path

→ Parameters 2

▼ [☐ Telemetry 2]

                                         *- RW_HEALTH_PARAMETER (mode) : Engineering Parameter
                                         *- RW_MODE_PARAMETER (mode) : Engineering Parameter

	✓ □ Properties 2

                                    RW_Physical_Category_SAVOIR_EDS
                                         DimensionMargin : Real Property Value
                                         DimensionMeasured: Real Property Value
```

Figure-4: Mapping between SAVOIR-EDS file and its data imported in the SRDB

At this stage, the SRDB contains:

- a. the comprehensive definition of the OBSW generic data
- b. the definition of parameters, interfaces and calibrations contained in the RW EDS.
- 5. Now, we need to override the OBSW generic data with RW data coming from the EDS. This overriding activity consists mainly in:
 - a. Renaming the data following a given naming convention that upholds the constraints of the target data exchange format.
 - b. Replacing the generic OBSW parameters with RW parameters coming from the RW EDS. More specifically, in our example, the overriding consists in replacing the first 8 bits of

the generic parameter OBSW RW_PARAMETER_1 with the *mode* parameter defined in the RW SEDS. The same is done with the second 8 bits.

The outcome of this overriding activity in the SRDB is shown in Figure-5.) The overridden data are now specific to the OBSW of SimpleSAT PRW0001 ✓ 🌠 OBSW_SimpleSAT_Data : Iement Configuration $RW_MODE_and_HEALTH$ Overridden data has PRW0001 (RW_MODE_and_HEALTH) : Aggregate Param 💠 field = PRW0002 : Aggregate Parameter Field been renamed 🏄 field = PRW0003 : Aggregate Parameter Field The two generic **OBSW** parameters have been removed Shows parameters logical structure Structure Offset -- PRW0002 (mode_status) : Engineering Paran · PRW0003 (health): Engineering Parameter PRW0002 is 4 CRW0001 associated with a RW specific RW_MODE_CALIBRATION calibration, defined in parameters, the RW EDS, to Default Text defined in the transform raw values 0 0 1 1 A EDS, have

replaced the

OBSW generic parameters

Tracking

Figure-5: Overriding OBSW generic parameter with RW's SEDS data

into readable text

specific to the RW

With this overriding, when the TM generated by the OBSW is received on the ground, the definition of parameter PRW0002 will be used to decode the raw value of the RW mode. Additionally, thanks to the calibration CRW0001 associated with the parameter PRW0002 as depicted in Figure-5, the incoming telemetry flow will either show **Tracking** or **Reporting_Rate** instead of 1 or 5.

At this stage, the OBSW data in the SRDB is well missionized with hardware data coming from the RW's SEDS. Now it is possible to export that data into the appropriate formats in order to operate the satellite and decode the incoming telemetry data correctly.