



**DOCUMENT 126-16**

**DDML SCHEMA VALIDATION**

**ABERDEEN TEST CENTER  
DUGWAY PROVING GROUND  
REAGAN TEST SITE  
WHITE SANDS MISSILE RANGE  
YUMA PROVING GROUND**

**NAVAL AIR WARFARE CENTER AIRCRAFT DIVISION  
NAVAL AIR WARFARE CENTER WEAPONS DIVISION  
NAVAL UNDERSEA WARFARE CENTER DIVISION, KEYPORT  
NAVAL UNDERSEA WARFARE CENTER DIVISION, NEWPORT  
PACIFIC MISSILE RANGE FACILITY**

**30TH SPACE WING  
45TH SPACE WING  
96TH TEST WING  
412TH TEST WING  
ARNOLD ENGINEERING DEVELOPMENT COMPLEX**

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION**

**DISTRIBUTION A: APPROVED FOR PUBLIC RELEASE  
DISTRIBUTION IS UNLIMITED**

This page intentionally left blank.

**DOCUMENT 126-16**

**DDML SCHEMA VALIDATION**

**February 2016**

**Prepared by**

**Telemetry Group**

**Published by**

**Secretariat  
Range Commanders Council  
White Sands Missile Range  
New Mexico 88002-5100**

This page intentionally left blank.

## Table of Contents

<b>Preface</b> .....	<b>v</b>
<b>Acronyms</b> .....	<b>vii</b>
<b>1. Introduction</b> .....	<b>1</b>
<b>2. Schema Validation Approach</b> .....	<b>5</b>
2.1 Approach Overview .....	5
2.2 User Story Requirements Capture .....	6
2.3 How to Use These Results .....	6
2.3.1 Using this Document to Validate the Schema .....	6
2.3.2 Maintaining This Document .....	7
2.4 Future Work .....	8
<b>3. User Stories / Scenarios</b> .....	<b>8</b>
3.1 General DDML User Stories .....	8
3.2 Data Display Object Containers.....	10
3.3 Common Data Display Object Elements .....	12
3.3.1 Axis.....	12
3.3.2 Colors.....	15
3.4 Data Display Objects .....	15
3.4.1 Bar Charts .....	15
3.4.2 Buttons .....	16
3.4.3 Custom Display Objects .....	16
3.4.4 Dials .....	17
3.4.5 Frequency Plots.....	17
3.4.6 Frequency Response Plots .....	18
3.4.7 Heads-Up Displays .....	18
3.4.8 Pie Charts .....	18
3.4.9 Radial Charts.....	19
3.4.10 Sliders .....	19
3.4.11 Strip Charts .....	20
3.4.12 Text .....	20
3.4.13 XY Charts .....	21
3.5 Dynamics .....	22
3.6 Data Variables.....	24
3.7 Data Sources .....	25
<b>4. Test Cases</b> .....	<b>26</b>
<b>5. Deficiencies</b> .....	<b>41</b>
<b>Appendix A. Citations</b> .....	<b>A-1</b>

## Table of Figures

Figure 1.	A Sample Data Display System.....	2
Figure 2.	Code Development Effort for Translators Needed for Current System: O(n <sup>2</sup> ).....	3
Figure 3.	Code Development Effort for Translators to and from DDML: O(n). ....	4
Figure 4.	XML Snippet .....	5
Figure 5.	Relationship Between User Stories, Test Cases, and Instance Documents .....	6

## Preface

This document describes the results of the IRIG schema validation task performed under the direction of the RCC TG Data MUX committee. To meet the task objective, a set of user stories, test cases, and instance documents was defined over a subset of intended functionality.

The purpose of this task is to create a method for performing IRIG XML schema validation. As opposed to XML instance document validation that determines if an XML instance document conforms to a schema, this method is used to validate that the XML schema can model the scenarios it was intended to support. This task includes validation of the current schemas and the creation of a framework and methodology through which future versions of the schema can be thoroughly validated prior to release.

The RCC gives special acknowledgement for production of this document to the TG Data Multiplex Committee. Please direct any questions to the committee point of contact or to the RCC Secretariat as shown below.

Telemetry Group Data Multiplex Committee Chairman: Mr. Jon Morgan  
412 TW, Edwards AFB  
Bldg 1408 Room 5  
301 East Yeager  
Edwards AFB, CA 93524  
Phone: DSN 527-8942 Com (661) 277-8942  
Fax: DSN 527-8933 Com (661) 277 8933  
email [jon.morgan.2.ctr@us.af.mil](mailto:jon.morgan.2.ctr@us.af.mil)

Secretariat, Range Commanders Council  
ATTN: TEDT-WS-RCC  
1510 Headquarters Avenue  
White Sands Missile Range, New Mexico 88002-5110  
Phone: DSN 258-1107 Com (575) 678-1107  
Fax: DSN 258-7519 Com (575) 678-7519  
email [usarmy.wsmr.attec.list.rcc@mail.mil](mailto:usarmy.wsmr.attec.list.rcc@mail.mil)

This page intentionally left blank.

## Acronyms

DDML	Data Display Markup Language
HUD	heads-up display
IRIG	Inter-Range Instrumentation Group
RCC	Range Commanders Council
SVG	Scalable Vector Graphics
T&E	test and evaluation
TMATS	Telemetry Attributes Transfer Standard
XML	eXtensible Markup Language

This page intentionally left blank.

## 1. Introduction

This Data Display Markup Language (DDML) Schema Validation document is intended to supplement Chapter 9<sup>1</sup> of Range Commanders Council (RCC) Inter-Range Instrumentation Group (IRIG) 106 Telemetry Standards. This document describes the approach taken to capture schema validation requirements, as well as a general approach to using these results for validation of both current and future versions of these schemas. We conclude with a presentation of the task results in detail.

Data display is a critical component for test and evaluation (T&E) environments in aircraft, space, and energy systems under test. Telemetry functions associated with these systems produce too much data for a single person to comprehend as alphanumeric information. Displays ease the task of interpreting raw measurands faster than the eye can fathom. Moreover, they depict when measurands are within safe and meaningful limits, show relationships between measurands, and spot trends. To assist with these efforts, data display systems provide a wide variety of customizable display objects, including strip charts, bar charts, vertical meters, round gauges, cross plots, tabular displays, orientation displays, and bit maps. Each display type can be tailored with respect to size, foreground and background colors, fonts, grids, and time and data format, to name a few.

Each data display object has peculiarities of its own. Not only is there a wide range of parameters and attributes, but these values are often a function of the state of the data that they display. For example, the attributes of an object can change as the color of a curve or numeric value changes when a measurand approaches a limit or is out of a limit. In addition to processing algorithms that detect changes, large time scales make it easier to visualize trends. Dynamic 3-D models of objects under testing can be used to show orientation, as opposed to interpreting a table of numeric orientation values. Multiple objects can be grouped into a single window to form instrument panels. Windows can be created for a test plan that is used over and over either with the same measurands and processed parameters or with new ones as required. Measurands and parameters can be changed in real time. Similarly, attributes such as data limits can also be changed. Standard drawing and graphics tools may be used in creating process diagrams and embellishing control panels. Snapshots of events can be sent to color printers or saved to disk for inclusion in reports. Features, such as local disk and ring buffers that are associated with video displays and are independent of system archiving, give operators the ability to recreate data leading to an event of interest.

This description illustrates how complex a singular data display system can be. To compound this situation, there is a variety of vendors offering software packages for data acquisition and display with such features, each requiring its own data display specification. [Figure 1](#) shows a snapshot of a data display.

---

<sup>1</sup> Range Commanders Council. "Telemetry Attributes Transfer Standard," in *Telemetry Standards*. IRIG 106-15. June 2015. May be superseded by update. Retrieved 22 July 2015. Available at [http://www.wsmr.army.mil/RCCsite/Documents/106-15\\_Telemetry\\_Standards/Chapter9.pdf](http://www.wsmr.army.mil/RCCsite/Documents/106-15_Telemetry_Standards/Chapter9.pdf).

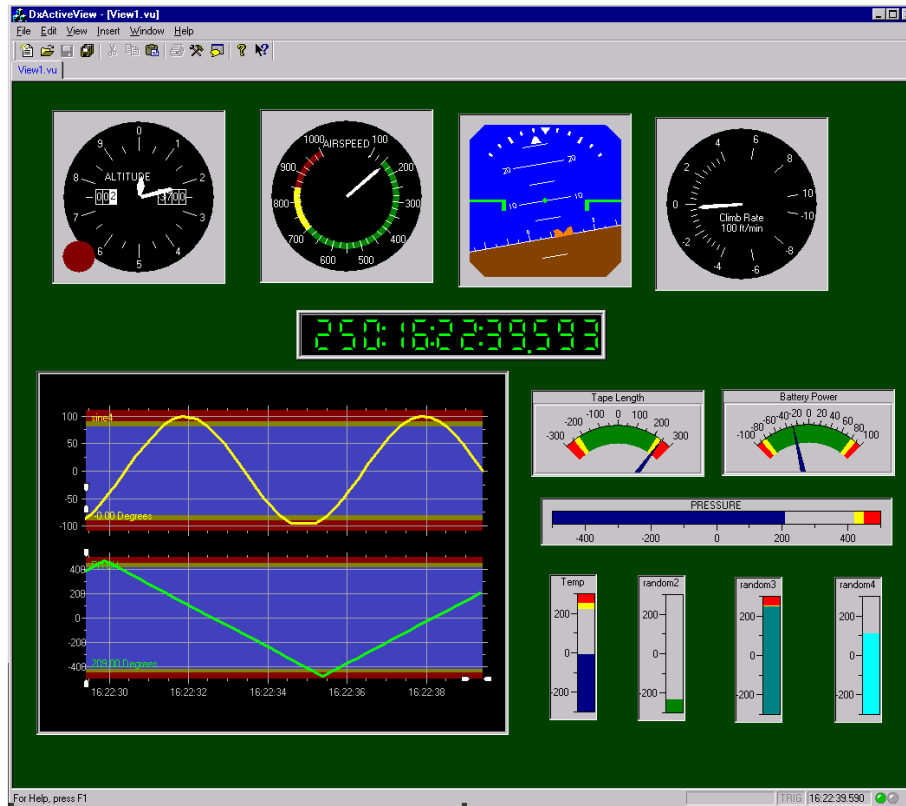


Figure 1. A Sample Data Display System.

Use of Telemetry Attributes Transfer Standard (TMATS) (RCC, July 2015) is an increasingly popular method for transferring files between non-compatible systems. Because each system uses a different internal format, translators are required to convert data to and from the TMATS intermediate format. The purpose of TMATS is to provide a common format for the transfer of information between the user and a test range or between different ranges. This format will minimize the activities unique to stations that are necessary to support any test item. In addition, the format is intended to eliminate the labor-intensive process currently required to reformat the information by providing the information on computer-compatible media, thus reducing errors and requiring less preparation time for test support.

Even though TMATS provides a powerful means for transferring telemetry information, it does not provide any support for capturing display objects and their layout for systems that require common data displays. Moreover, the tendency of T&E is towards a plug-and-play-like data acquisition system that requires standard languages and modules for data displays. Currently, the only way to transfer data displays between display applications is to manually recreate displays using an experienced programmer. Also, absence of a neutral format requires the programmer to manually craft the display transfers between each system pair in the application domain.

For example, a T&E system of six data display systems requires 30 unidirectional data display transfers, as shown in [Figure 2](#). This can be formulated as  $n(n-1)$  transfers, where  $n$  is the number of applications. Also, since there are no automatic translators between display

systems, a small change in one of the systems requires manual changes in the other related transfers.

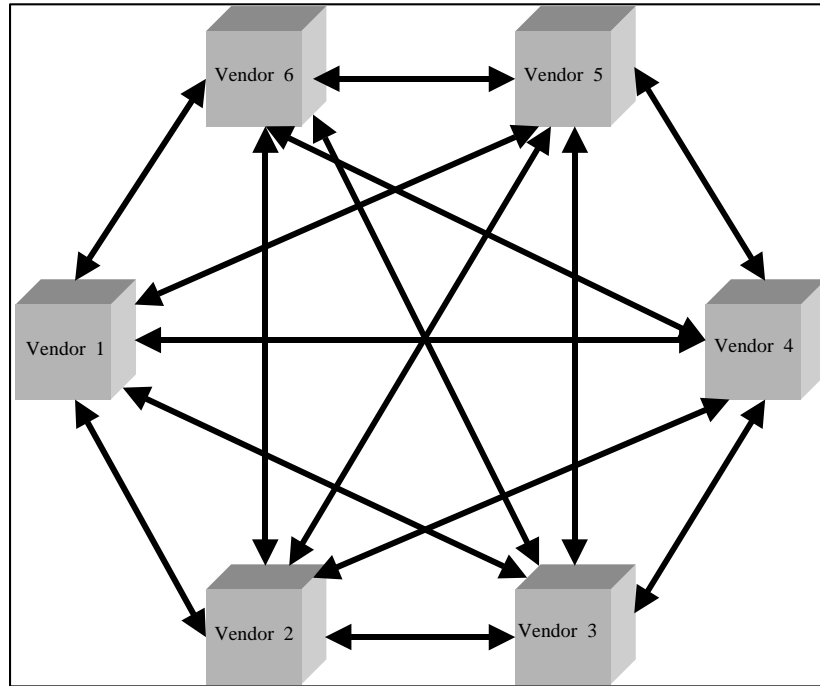


Figure 2. Code Development Effort for Translators Needed for Current System:  $O(n^2)$ .

The DDML standard is a specification of an eXtensible Markup Language (XML)-based neutral format that is intended to be the inter-lingua of data displays. The DDML standard has the requirement of being generic enough to encompass various vendor-specific data display formats and at the same time being unified (not a loose grouping of XML-ized vendor formats). In addition, it is required to support reusable concepts (such as variables and data sources), be robust (e.g., use of cross-references), and support future objects without warranting a change of the DDML format.

Availability of DDML as the inter-lingua drastically reduces the number of unidirectional translators to two per vendor-specific format. Returning to the T&E system of six data display systems, we would require 12 unidirectional display transfers, as shown in [Figure 3](#). In general, for a system of  $n$  formats, the translator development effort is  $O(n)$ . Also, the task of developing translators would be highly simplified because of two reasons. First, because DDML is an XML format, there is ample support by way of free software to parse and generate DDML. Second, there is a high degree of reuse of a number of translator components because of the new hub-and-spoke translator framework. Our proposed solution, therefore, includes the development of highly reusable, customizable, well-documented translator components along with well-documented end-to-end processes for rapid translator development. This translator framework will then be used to develop the bi-directional translators between DDML and the vendor formats. As a result, DDML along with the system of translators is a practical and cost-effective solution to the current manual method of recreating displays in the target format.

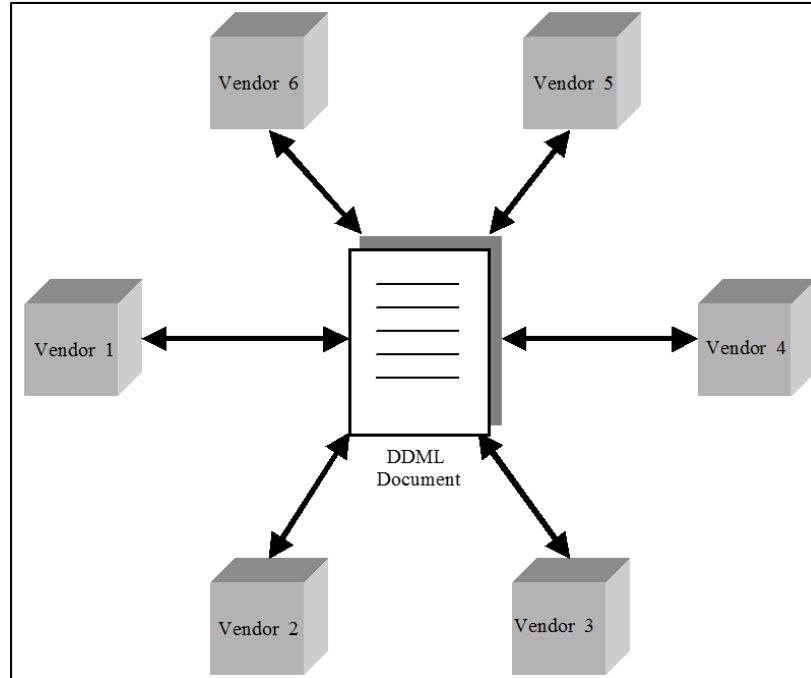


Figure 3. Code Development Effort for Translators to and from DDML:  
 $O(n)$ .

Changes in one of the vendor formats would require re-coding only in the translators between that format and the neutral format. While the effort to accommodate the changes is mitigated by the design of modular components, we can also *automatically generate* key components of the translator code that can be easily compiled, tested, and deployed. This capability will significantly mitigate the effort to keep up with “moving targets” or evolution of source/target vendor languages because now the focus of the effort will be on modeling the data display specification and not on the translator code development. In that sense, it will be similar to using Computer-Assisted Software Engineering tools to develop object-oriented software and automatically generate the target code for compiling.

Typically, software produces and reads DDML files. Automating this process reduces the time needed to prepare a data display configuration and eliminates errors that inevitably result from entering this information manually.

The rules of an XML schema govern DDML instance documents. For information about XML, refer to RCC 125-15, XML Style Guide.<sup>2</sup> [Figure 4](#) provides an XML snippet of a DDML structure.

<sup>2</sup> Range Commanders Council. *XML Style Guide*. 125-15. July 2015. May be superseded by update. Retrieved 28 January 2016. Available at [http://www.wsmr.army.mil/RCCsite/Documents/125-15\\_XML\\_Style\\_Guide/125-15\\_XML\\_Style\\_Guide.pdf](http://www.wsmr.army.mil/RCCsite/Documents/125-15_XML_Style_Guide/125-15_XML_Style_Guide.pdf).

```

<ddmlbarchart:barchart id="BAGC1">
  <ddmlcommon:name>BAGC1</ddmlcommon:name>
  <ddmlcommon:point>
    <ddmlcommon:x>500000</ddmlcommon:x>
    <ddmlcommon:y>0</ddmlcommon:y>
  </ddmlcommon:point>
  <ddmlcommon:point>
    <ddmlcommon:x>500000</ddmlcommon:x>
    <ddmlcommon:y>250000</ddmlcommon:y>
  </ddmlcommon:point>
  <ddmlcommon:point>
    <ddmlcommon:x>750000</ddmlcommon:x>
    <ddmlcommon:y>250000</ddmlcommon:y>
  </ddmlcommon:point>
  <ddmlcommon:point>
    <ddmlcommon:x>750000</ddmlcommon:x>
    <ddmlcommon:y>0</ddmlcommon:y>
  </ddmlcommon:point>
  <ddmlcommon:titleFont>Arial</ddmlcommon:titleFont>
  <ddmlcommon:title>Bar Chart</ddmlcommon:title>
  <ddmlcommon:titleColor> 65280</ddmlcommon:titleColor>
  <ddmlcommon:titleFontSize>24</ddmlcommon:titleFontSize>
  <ddmlcommon:scrollDirection>DOWN</ddmlcommon:scrollDirection>
</ddmlbarchart:barchart>

```

Figure 4. XML Snippet

## 2. Schema Validation Approach

This section describes the approach taken to capture schema validation requirements, as well as a general approach to using these results for validation of both current and future versions of these schemas.

### 2.1 Approach Overview

In the simplest sense, our schema validation approach involves capturing the requirements of the schemas and then testing the schemas' ability to satisfy those requirements by implementing specific cases of each requirement. In order to achieve this, we have documented three main types of information.

- **User Stories:** A user story describes a specific requirement of the schema in the terms of a user's need. An example of a user's need is "I need to configure a data display."
- **Test Cases:** For each user story, we capture one or more test cases. A test case is a description of a very specific example of the need described in the user story. An example of a test case is "Configure a strip chart to display the sampled vibration measurements."
- **Instance Documents:** For each user story, we then create one XML instance document. The instance document is a schema-valid XML file that completely describes the information in the test case in a manner that satisfies the user story. If it is not possible to create an instance document for a given test case, then the schema is not valid and requires modification.

The relationships between user stories, instance documents, and test cases are illustrated in [Figure 5](#).

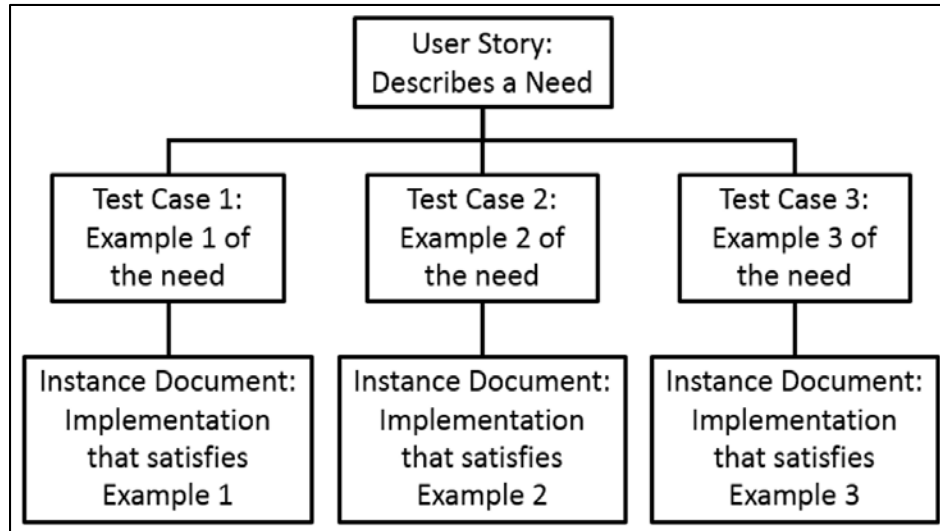


Figure 5. Relationship Between User Stories, Test Cases, and Instance Documents

Under this effort, when a test case could not be satisfied by a schema-valid instance document, we created a description of a schema defect for further investigation by the committee.

## 2.2 User Story Requirements Capture

The general approach taken for this task is to capture schema requirements using user stories, which are used in the agile software development process to capture the function that a system is to perform. Since an IRIG schema is intended to capture the function that a telemetry system is intended to perform, we feel that user stories are a simple and concise way to capture requirements. The general format of a user story is:

As a/an <user>, I want to be able to <task description>, so that <objective to meet>.

## 2.3 How to Use These Results

In this section, we describe how to use this document to perform schema validation, as well as how to maintain the document as an accurate and up-to-date guide to schema requirements.

### 2.3.1 Using this Document to Validate the Schema

With a complete and accurate set of user stories, test cases, and instance documents, schema validation is a 4-step process.

1. **Validate all instance documents against the schema.** This instance document validation checks to ensure that each document is syntactically valid according to the rules laid out in the schema. This step can be performed by a number of automated commercial off-the-shelf and government off-the-shelf tools. If the schema has changed

since the initial development of the instance documents, it is expected that some or all of the documents will fail validation.

2. **Determine the cause of failure for any documents that failed validation.** A document will fail schema validation due to a change in either the syntax or content of the standard. If the change is only syntactic, it will be apparent when comparing the invalid document to the schema because all of the data in the document will still have a place in the new schema, but the location or name of the XML element or attribute may have changed. If the change is to the content of the standard, then there will be information in the instance document that does not have an apparent place under the new schema.
3. **Fix the invalid documents and document schema defects.** Once the causes of a document failing validation have been determined, the syntactic differences can be resolved by simply modifying the instance document, making sure that all data content is preserved in the newly formatted document. If there are content changes resulting in data from the instance document not having a place in the newly formatted document, this must be documented as a potential defect in the schema. The schema must then be considered invalid until all defects are addressed.
4. **Address all schema defects.** Each schema defect can be traced back to an invalid instance document, and from there to a specific test case and user story. The committee must compare the test case description to the new schema and determine if there is any way to support the test case. This is a more thorough version of the analysis described in step 3, in which the schema is looked at as a whole for ways to support the test case. If there is no way to support the test case with the new schema, then the committee must either modify the schema to support the test case, or modify the test case (and possibly the user story) to reflect a change in content requirements.

### 2.3.2 Maintaining This Document

For proper validation of future versions of the schemas, this document must be maintained to accurately reflect the requirements of the standards. As the committee considers schema changes, it must determine whether the need for these changes has resulted from a requirement being added or removed, from a previously unforeseen example of an existing requirement, or from a need to update the style or syntax of the standard.

When new requirements are identified, a new user story must be added or an existing user story must be re-worded to include the new requirement. Then, a set of test cases should be developed for validating the schema's ability to support the new requirement. Finally, instance documents must be created that implement each test case.

If a requirement is being dropped from the standard (a historical example would be the dropping of support for Pulse Amplitude Modulation from TMATS-XML), then the associated user stories, test cases, and instance documents must be deleted or modified to reflect the change in requirements. The instance documents can be used as a guide to which portions of the schema are associated with the deleted requirement and may aid in determining which schema constructs to remove.

When previously unforeseen examples are discovered, the appropriate user story should be identified and new test cases associated with this user story should be added that represent the new usage example. As before, an instance document should be created for each new user story.

Finally, if the changes are only the result of a need for different syntax, this document will not require any changes; however, the associated instance documents must be updated to reflect the new syntax.

In all cases, the new schema must be validated according to the steps in Subsection [2.3.1](#).

## 2.4 Future Work

The efforts of this RCC Telemetry Group task have produced a methodology and a framework for performing schema validation; however, there is still work to be done to complete the data needed for full schema validation. Currently, the top-level user stories and most sub-user-stories have been fully fleshed out according to the current understanding of the two standards; however, only a representative set of test cases has been created. The next steps following the completion of the current task are:

1. Address all of the schema deficiencies documented in Section [5](#);
2. Create test cases and instance documents for all user stories in Section [3](#) with “TBD” currently listed under “Test Cases”;
3. Review the user stories list and add new stories to Section [3](#) as new requirements come to light.

## 3. User Stories / Scenarios

This section describes the high-level DDML XML schema validation user stories.

### 3.1 General DDML User Stories

US-1. As a data analyst, I want to be able to use graphic resources to display data so that the data can be visualized and analyzed.

Source reference(s)

- Test case(s): [TC-5](#), [TC-6](#), [TC-7](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-2. As a data analyst, I want to retrieve data from multiple data sources so that data is not limited to a single source and more data can be displayed.

Source reference(s)

- Test case(s): [TC-5](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-3. As a data analyst, I want to store data in data variables so that data can be utilized in software.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-4. As a data analyst, I want to derive new data from existing data sets so that I can gain new insights into the data.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

US-5. As a data analyst, I want to dynamically update data displays so that I can show real-time data.

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-6. As a data analyst, I want to use multiple colors so that users can easily decipher different elements.

Source reference(s)

- Test case(s): [TC-5](#), [TC-6](#), [TC-7](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-7. As a data analyst, I want to position graphics so that they can adequately present data.

Source reference(s)

- Test case(s): [TC-5](#), [TC-6](#), [TC-7](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-8. As a data analyst, I want to combine low-level graphic elements (lines, rectangles, etc.) so that I can build complex graphics.

Source reference(s)

- Test case(s): [TC-7](#)
- Deficiencies: N/A

US-9. As a data analyst, I want to use Scalable Vector Graphics (SVG) so that I can support interactivity and animations.

Source reference(s)

- Test case(s): [TC-7](#), [TC-12](#)
- Deficiencies: N/A

US-10. As a data analyst, I want to use a project structure to store data display information so that various configurations and displays can be created.

Source reference(s)

- Test case(s): [TC-1](#)
- Deficiencies: N/A

US-11. As a data analyst, I want storage of data history so that previously collected data can be shown using data displays.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

### 3.2 Data Display Object Containers

US-12. As a data analyst, I want graphs to specify data display configurations so that the data can be interpreted and analyzed.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-13. As a data analyst, I want graph titles to display a general statement of what the graph is displaying so that context can be given to the graph data.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-14. As a data analyst, I want to set graph title color, font, and size so that the graph title is visually appealing.

Source reference(s)

- Test case(s): [TC-8](#)
- Deficiencies: N/A

US-15. As a data analyst, I want to set background and data area colors for graphs so that I can differentiate different data areas.

Source reference(s)

- Test case(s): [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-16. As a data analyst, I want grids to display a group of display objects in a tabular layout so that display objects can be evenly spaced.

Source reference(s)

- Test case(s): [TC-14](#)
- Deficiencies: N/A

US-17. As a data analyst, I want to specify grid row and column for display objects so that objects can be placed in specific cells of a grid.

Source reference(s)

- Test case(s): [TC-14](#)
- Deficiencies: N/A

US-18. As a data analyst, I want to show or hide gridlines so that the grid cells can be outlined or not outlined.

Source reference(s)

- Test case(s): N/A
- Deficiencies: [D-3](#)

US-19. As a data analyst, I want color assignments for gridlines so that they can indicate different connected elements of the grid.

Source reference(s)

- Test case(s): N/A
- Deficiencies: [D-4](#)

US-20. As a data analyst, I want models to store graphic resource configurations and processes so that they can be stored and reused.

Source reference(s)

- Test case(s): [TC-4](#)
- Deficiencies: N/A

US-21. As a data analyst, I want to specify whether a model needs to be displayed or referenced so that only the necessary data displays will be shown.

Source reference(s)

- Test case(s): [TC-4](#)

- Deficiencies: N/A

US-22. As a data analyst, I want to set  $x$  and  $y$  directions so that model orientation can be configured.

Source reference(s)

- Test case(s): [TC-4](#)
- Deficiencies: N/A

US-23. As a data analyst, I want to reference sub-models so that existing models can be used within new container models.

Source reference(s)

- Test case(s): [TC-12](#)
- Deficiencies: N/A

US-24. As a data analyst, I want maps to display geographical information so that geospatial information can be understood.

Source reference(s)

- Test case(s): [TC-6](#)
- Deficiencies: N/A

US-25. As a data analyst, I want to choose a minimum and maximum value for longitude and latitude so that a specific geospatial area can be shown.

Source reference(s)

- Test case(s): [TC-6](#)
- Deficiencies: [D-2](#)

US-26. As a data analyst, I want the option to choose a map image file so that a map image can be used as the map.

Source reference(s)

- Test case(s): [TC-6](#)
- Deficiencies: N/A

### 3.3 Common Data Display Object Elements

#### 3.3.1 Axis

US-27. As a data analyst, I want axes to display values or times for plots so that plot data can be interpreted.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-28. As a data analyst, I want color assignments for axes so that multiple axes can be differentiated.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-29. As a data analyst, I want a label for each axis so that the data values can be interpreted.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-30. As a data analyst, I want to set label font size and style so that font can easily be read.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

US-31. As a data analyst, I want to specify minimum and maximum axis values so that the axis can be bound.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-32. As a data analyst, I want to set the axis unit so that context can be given to the displayed values.

Source reference(s)

- Test case(s): [TC-9](#)
- Deficiencies: N/A

US-33. As a data analyst, I want axis grids so that data values on plots can be accurately estimated.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#)
- Deficiencies: N/A

US-34. As a data analyst, I want axis grid intervals so that data granularity can be increased or decreased.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#)
- Deficiencies: N/A

US-35. As a data analyst, I want the ability to show or hide axis ticks so that precise data values can be accurately estimated.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

US-36. As a data analyst, I want to specify tick color so that emphasis can be given to the ticks.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#)
- Deficiencies: N/A

US-37. As a data analyst, I want tick labels so that the value of each tick is shown.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#)
- Deficiencies: N/A

US-38. As a data analyst, I want to specify tick label font size and style so that the labels can be read easily.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

US-39. As a data analyst, I want to set tick label intervals so that labels are only shown on the specific tick intervals.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#)
- Deficiencies: N/A

US-40. As a data analyst, I want to set the axis type to either VALUE or TIME so that context can be given to the axis.

Source reference(s)

- Test case(s): [TC-8](#), [TC-9](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

### 3.3.2 Colors

US-41. As a data analyst, I want colors for all data display objects so that data displays can be visually appealing.

Source reference(s)

- Test case(s): [TC-1](#), [TC-2](#), [TC-5](#), [TC-7](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-42. As a data analyst, I want base-10 integer encoding for colors so that a common storage mechanism can be used for all color assignments.

Source reference(s)

- Test case(s): [TC-1](#), [TC-2](#), [TC-5](#), [TC-7](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

## 3.4 **Data Display Objects**

### 3.4.1 Bar Charts

US-43. As a data analyst, I want bar charts to display one or more variables as vertical or horizontal bars whose lengths correspond to the values so that data can be analyzed and compared.

Source reference(s)

- Test case(s): [TC-8](#)
- Deficiencies: N/A

US-44. As a data analyst, I want color assignments for bar charts so that they can indicate different data sets being shown in the bar chart.

Source reference(s)

- Test case(s): [TC-6](#)
- Deficiencies: N/A

US-45. As a data analyst, I want to specify bar chart scroll direction so that the bars will be oriented and extend toward the direction.

Source reference(s)

- Test case(s): [TC-6](#)
- Deficiencies: N/A

### 3.4.2 Buttons

US-46. As a data analyst, I want buttons to perform variable assignments on a user click so that users can dynamically change data displays.

Source reference(s)

- Test case(s): [TC-10](#)
- Deficiencies: N/A

US-47. As a data analyst, I want color assignments for buttons so that button functions can be emphasized.

Source reference(s)

- Test case(s): [TC-10](#)
- Deficiencies: [D-6](#)

US-48. As a data analyst, I want to set a label for a button so that meaning can be given to the button.

Source reference(s)

- Test case(s): [TC-10](#)
- Deficiencies: N/A

US-49. As a data analyst, I want color assignments for button labels so that emphasis can be given to the button.

Source reference(s)

- Test case(s): [TC-10](#)
- Deficiencies: N/A

US-50. As a data analyst, I want to set button label font and size so that the button can be customized.

Source reference(s)

- Test case(s): [TC-10](#)
- Deficiencies: N/A

### 3.4.3 Custom Display Objects

US-51. As a data analyst, I want to build non-standardized display objects so that unique data displays can be created.

Source reference(s)

- Test case(s): [TC-7](#)
- Deficiencies: N/A

- US-52. As a data analyst, I want to utilize one or more existing display objects, as well as other graphic resources, in a custom data display object so that complex custom data displays can be designed using existing data displays.

Source reference(s)

- Test case(s): [TC-7](#)
- Deficiencies: N/A

#### 3.4.4 Dials

- US-53. As a data analyst, I want dials to display circular axis values so that I can create a gauge or compass.

Source reference(s)

- Test case(s): [TC-11](#)
- Deficiencies: N/A

- US-54. As a data analyst, I want to assign colors to dials so that emphasis can be given to the dials.

Source reference(s)

- Test case(s): [TC-11](#)
- Deficiencies: N/A

- US-55. As a data analyst, I want to set minimum and maximum dial angles so that the dial range can be restricted.

Source reference(s)

- Test case(s): [TC-11](#)
- Deficiencies: N/A

#### 3.4.5 Frequency Plots

- US-56. As a data analyst, I want frequency plots to display frequency vs. magnitude so that frequencies can be analyzed.

Source reference(s)

- Test case(s): [TC-9](#)
- Deficiencies: N/A

- US-57. As a data analyst, I want frequency plotted on the *x*-axis and magnitude plotted on the *y*-axis so that the data can be visualized.

Source reference(s)

- Test case(s): [TC-9](#)
- Deficiencies: N/A

US-58. As a data analyst, I want color assignments for frequency plots so that emphasis can be given to the frequency plots.

Source reference(s)

- Test case(s): [TC-9](#)
- Deficiencies: N/A

### 3.4.6 Frequency Response Plots

US-59. As a data analyst, I want frequency response plots to display magnitude and phase on the same frequency axis so that the signal can be analyzed.

Source reference(s)

- Test case(s): [TC-18](#)
- Deficiencies: N/A

US-60. As a data analyst, I want color assignments for frequency response plots so that emphasis can be given to the frequency response plots.

Source reference(s)

- Test case(s): [TC-18](#)
- Deficiencies: N/A

### 3.4.7 Heads-Up Displays

US-61. As a data analyst, I want heads-up displays (HUDs) to project display objects onto transparent surfaces so that data can be displayed without requiring users to look away.

Source reference(s)

- Test case(s): [TC-12](#)
- Deficiencies: N/A

US-62. As a data analyst, I want to specify text color, size, and font for HUDs so that textual information can be readable when overlaid on a video feed.

Source reference(s)

- Test case(s): [TC-12](#)
- Deficiencies: N/A

### 3.4.8 Pie Charts

US-63. As a data analyst, I want pie charts to display multiple percentage variables in a circular form so that percentages can be adequately visualized.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

US-64. As a data analyst, I want color assignments for pie charts so that data sets can be differentiated.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

### 3.4.9 Radial Charts

US-65. As a data analyst, I want radial charts to display objects that represent values as distances outward from a central point so that I can better understand the data.

Source reference(s)

- Test case(s): [TC-17](#)
- Deficiencies: N/A

US-66. As a data analyst, I want color assignments for radial charts so that emphasis can be given to the charts.

Source reference(s)

- Test case(s): [TC-17](#)
- Deficiencies: N/A

### 3.4.10 Sliders

US-67. As a data analyst, I want sliders to allow users to dynamically change the data being displayed so that users can identify the specific data they would like to view.

Source reference(s)

- Test case(s): [TC-14](#)
- Deficiencies: N/A

US-68. As a data analyst, I want color assignments for sliders so that emphasis can be given.

Source reference(s)

- Test case(s): [TC-14](#)
- Deficiencies: N/A

US-69. As a data analyst, I want to set slider orientation so that the slider can be either vertical or horizontal.

Source reference(s)

- Test case(s): [TC-14](#)
- Deficiencies: N/A

### 3.4.11 Strip Charts

US-70. As a data analyst, I want strip charts to display plots on a scrolling grid so that data can be dynamically displayed.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

US-71. As a data analyst, I want color assignments for strip charts so that they can indicate different data sets displayed by the strip chart.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

US-72. As a data analyst, I want to specify strip chart scroll direction so that the strip chart will be oriented and extend towards the scroll direction.

Source reference(s)

- Test case(s): [TC-15](#)
- Deficiencies: N/A

### 3.4.12 Text

US-73. As a data analyst, I want to display text objects so that information can be passed textually to a user.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-74. As a data analyst, I want to display both static and dynamic text so that context can be given to various graphical elements.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-75. As a data analyst, I want colored text so that significance/context can be given to text.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-76. As a data analyst, I want a selection of text fonts so that the appearance of the text can be changed.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-77. As a data analyst, I want to set text size so that emphasis can be given to text.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-78. As a data analyst, I want to set text alignment so that the position of the text can be automatically calculated.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-79. As a data analyst, I want to set text position so that I can set the exact location of the text.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

US-80. As a data analyst, I want to specify the background color of text objects so that contrast can be given to the text color.

Source reference(s)

- Test case(s): [TC-5](#)
- Deficiencies: N/A

### 3.4.13 XY Charts

US-81. As a data analyst, I want *xy* charts to display a line or scatter plot on an *x* and *y* axis so that I can visualize a set of *xy* pairs.

Source reference(s)

- Test case(s): [TC-16](#)
- Deficiencies: N/A

US-82. As a data analyst, I want axis objects to have a type of either VALUE or TIME so that I can better understand the use for the axis.

Source reference(s)

- Test case(s): [TC-16](#)
- Deficiencies: N/A

US-83. As a data analyst, I want color assignments for *xy* charts so that they can be differentiated.

Source reference(s)

- Test case(s): [TC-16](#)
- Deficiencies: N/A

### 3.5 Dynamics

US-84. As a data analyst, I want to use greater-than, less-than, greater-than-or-equal, less-than-or-equal, equal, and not equal comparison operators so that two objects can be compared.

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-85. As a data analyst, I want comparisons between two objects so that I can perform further operations based on the results.

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-86. As a data analyst, I want built-in display object dynamic behaviors so that display object functionality can be easily implemented.

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-87. As a data analyst, I want display object scaling so that display objects can resize during run-time.

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)

- Deficiencies: N/A

US-88. As a data analyst, I want if, then, and else rule types so that object manipulations can be made based on object value(s) or state(s).

Source reference(s)

- Test case(s): [TC-5](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-89. As a data analyst, I want functions so that data variables can be manipulated or created based on a given set of inputs.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

US-90. As a data analyst, I want to be able to rotate the object so that the user can gain a better perspective.

Source reference(s)

- Test case(s): TBD
- Deficiencies: N/A

US-91. As a data analyst, I want to move display objects in the  $x$  or  $y$  planes so that the objects can be placed in desired locations.

Source reference(s)

- Test case(s): [TC-5](#), [TC-6](#), [TC-8](#), [TC-9](#), [TC-10](#), [TC-11](#), [TC-12](#), [TC-13](#), [TC-14](#), [TC-15](#), [TC-16](#), [TC-17](#)
- Deficiencies: N/A

US-92. As a data analyst, I want to change the color of the display object based on an event so that events can be visualized or meaning can be given to an event.

Source reference(s)

- Test case(s): [TC-7](#)
- Deficiencies: N/A

US-93. As a data analyst, I want display object visibility to change so that objects can be shown or hidden.

Source reference(s)

- Test case(s): N/A
- Deficiencies: [D-5](#)

US-94. As a data analyst, I want logical operators (AND, OR, or XOR) so that data variable manipulations can occur when certain conditions are met.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

US-95. As a data analyst, I want to use mathematical operators so that data can be manipulated.

Source reference(s)

- Test case(s): [TC-13](#)
- Deficiencies: N/A

### 3.6 Data Variables

US-96. As a data analyst, I want data variables so that objects and data sources can be linked.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-97. As a data analyst, I want data variable types so that objects can store different sets of information.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-98. As a data analyst, I want a data variable pool so that all variables are enumerated and can be accessed.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-99. As a data analyst, I want to store incoming data in data variables so that I can utilize the data in software.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-100. As a data analyst, I want to store a set of related information so that I can retrieve the information when necessary.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-101. As a data analyst, I want data variable instances so that I can use the variable.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

US-102. As a data analyst, I want static value data variables so that static information can be stored and accessed.

Source reference(s)

- Test case(s): [TC-3](#)
- Deficiencies: N/A

### 3.7 Data Sources

US-103. As a data analyst, I want data sources to store or transmit data so that the data can be stored in data variables.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

US-104. As a data analyst, I want a data source pool so that data sources can be selected and used.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

US-105. As a data analyst, I want an unbounded parameter to specify pertinent data source information so that unique data source information can be enumerated.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

US-106. As a data analyst, I want to utilize various data source types so that data can be retrieved from multiple data source types.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

US-107. As a data analyst, I want to specify the data source type so that data source features can be inferred based on type.

Source reference(s)

- Test case(s): [TC-2](#)
- Deficiencies: N/A

## 4. Test Cases

TC-1. Simple DDML Definition

Details: Define a DDML instance document with minimal data

- Define a Project
  - Name: Schema Validation Task
  - ID: PROJ1
- Define a Model
  - Name: Schema Validation Model
  - ID: MOD1
  - Minx:0, Miny:0, Maxx:1000, Maxy:1000
  - xDirection: RIGHT, yDirection: UP
  - BackgroundColor: 65280 (Green)
  - Display Object
    - Name: Schema Validation Display Object
    - Point1: 0, 0
    - Point2: 500, 1000
    - Point3: 1000, 0
- Define Data Variable in Pool
  - Name: Schema Validation Variable
  - ID: V1
  - Data Source Reference: S1
- Define Data Source in Pool
  - Name: Schema Validation Data Source
  - ID:S1

Instance Document: TC-1\_SimpleDDMLDefinition.ddml

Relevant User Stories: [US-10](#), [US-41](#), [US-42](#)

Deficiencies: N/A

TC-2. Define Data Source pool  
Details: Define an extensive data source pool

- Data Source 1:
  - Name: S1SRP
  - ID: S1SRP
  - Parameters (Name, Value):
    - Type, FIXED-REF
    - Save, 0
    - Stream, 0
    - Symbol, SQUARE
    - Track Color, 65280
    - Scale, 0.5
    - Track Type, TSPI
    - Site, 3
    - Hide, True
  
- Date Source 2:
  - Name: SHT1A
  - ID: SHT1A
  - Parameters (Name, Value):
    - Type, RADAR
    - Save, 1
    - Stream, 0
    - Symbol, AIRPLANE
    - Track Color, 65280
    - Scale, 0.5
    - Track Type, TSPI
    - Site, 3
  
- Data Source 3:
  - Name: EAU01
  - ID: EAU01
  - Parameters (Name, Value):
    - Type, EATS
    - Save, 1
    - Stream, 0
    - Symbol, AIRPLANE
    - Track Color, 65280
    - Scale, 0.5
    - Track Type, TSPI
    - Site, 3
  
- Data Source 4:
  - Name: S1
  - ID: S1
  - Parameters (Name, Value):

- Type, SELECT
  - Save, 1
  - Stream, 0
  - Symbol, AIRPLANE
  - Track Color, 65280
  - Scale, 0.5
  - Track Type, TSPI
  - Site, 3
- Data Source 5:
    - Name: IIP01
    - ID: IIP01
    - Parameters (Name, Value):
      - Type, IIP01
      - Save, 1
      - Stream, 0
      - Symbol, TRIANGLE
      - Track Color, 65280
      - Scale, 0.5
      - Track Type, TSPI
      - Site, 3
  - Data Source 6:
    - Name: TM
    - ID: TM
    - Parameters (Name, Value):
      - Type, TPS
      - Save, 1
      - Stream, 0
      - Symbol, AIRPLANE
      - Track Color, 16777215
      - Scale, 0.5
      - Track Type, TSPI
      - Site, 3

Instance Document: TC-2\_DefineDataSourcePool.ddml

Relevant User Stories: [US-11](#), [US-41](#), [US-42](#), [US-103](#), [US-104](#), [US-105](#), [US-106](#), [US-107](#)

Deficiencies: [D-1](#)

TC-3. Define Data Variable Pool  
Details: Define a data variable pool

- Data Variables (ID-Name):
  - Var0-IRIG\_DAYS, Var1-IRIG\_HOURS, Var2-IRIG\_MIN, Var3-IRIG\_SEC, Var4-IRIG\_MSEC, Var5-IRIG\_PTIN, Var6-IRIG\_COURSE-M, Var7-VEL-KTS, Var8-ALT-KFT, Var9-SR-NMI, Var10-ANGLE-

ELV, Var11-ANGLES-OFF, Var12-BEARING-M, Var13-TRK-ANGLE, Var14-CVEL-KTS, etc.

Instance Document: TC-3\_DefineDataVariablePool.ddml

Relevant User Stories: [US-3](#), [US-96](#), [US-97](#), [US-98](#), [US-99](#), [US-100](#), [US-101](#), [US-102](#)

Deficiencies: N/A

TC-4. Define Model

Details: Define a model with more information

- Model Name: SELMAP
- Minx: 0
- Maxx: 1300000
- Miny: 0
- Maxy: 1000000
- X Direction: RIGHT
- Y Direction: DOWN
- BackgroundColor: 255 (red)
- Display Object
  - Name: Schema Validation Display Object
  - Point1: 0, 0
  - Point2: 500, 1000
  - Point3: 1000, 0
- Parameter:
  - Name: Screens
  - Value: GRIDS
- ShowInProject: false

Instance Document: TC-4\_DefineModel.ddml

Relevant User Stories: [US-20](#), [US-21](#), [US-22](#)

Deficiencies: [D-1](#)

TC-5. Define Textual Display Object

Details: Define a textual display object within a model

- Display Object Name: SELMAPTOD
- ID: SELMAPTOD
- Point1: 1000000, 0
- Point2: 1000000, 40000
- Point3: 1300000, 40000
- Point4: 1300000, 0
- Orientation: HORIZONTAL
- Value Position: CENTER
- Value Format: %3d:%2d:%2d:%2d.%3d
- Custom Parameters (Name, Value):
  - RVsubType, TOD

- Framed, False
- Dynamics:
  - Type: text
  - Variable Uses (ID, Name, Pool Ref, Color)
    - Vu0, IRIG\_DAYS, var0, 16777215
    - Vu1, IRIG\_HOURS, var1, 16777215
    - Vu2, IRIG\_MIN, var2, 16777215
    - Vu3, IRIG\_SEC, var3, 16777215
    - Vu4, IRIG\_MSEC, var4, 16777215
  - Rules:
    - If < 0, Value = 0

Instance Document: TC-5\_DefineTextualDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-2](#), [US-5](#), [US-6](#), [US-7](#), [US-41](#), [US-42](#), [US-73](#), [US-74](#), [US-75](#), [US-76](#), [US-77](#), [US-78](#), [US-79](#), [US-80](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: [D-1](#)

TC-6. Define Map Display Object

Details: Define a map display object within a model

- Display Object Name: SELECT-MAP
- ID: SELECT-MAP
- Point1: 0, 0
- Point2: 0, 1000000
- Point3: 1000000, 1000000
- Point4: 1000000, 0
- Custom Parameters (Name-Value):
  - Trace-S1, Scale-OFF, NTDS-ON, Topography-NAWC, Legend-ON, Size-NMI, Cursor-ON, Updates-0.0
- Min Latitude: 30.0
- Max Latitude: 37.0
- Min Longitude: -125.1
- Max Longitude: -118.1
- Background Images (ID, Filename):
  - SELECT-MAP\_world, maps/world.map
  - SELECT-MAP\_socal, Maps/R-2508/SoCal-2508.map

Instance Document: TC-6\_DefineMapDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-24](#), [US-25](#), [US-26](#), [US-41](#), [US-42](#), [US-91](#)

Deficiencies: [D-1](#), [D-2](#)

TC-7. Define Custom Display Object

Details: Define a generic display object within a model

- Display Object Name: background\_ALT
- ID: background\_ALT
- Custom Parameter (Name-Value):
  - gridsSubTypes-background, Points-40.0, Graphics-LINKED, Marker-POINT, Color-RED, Linetype-SOLID, Label-, Font-0, Orient-RIGHT, Justify-LEFT, Offsets-0,0, Smoothness-0.0, Center-(0.0, 0.0), Radius-0.0, xAxis-0.0, yAxis-0.0, Azimuth-0.0, Arcbegin-0.0, Arclength-0.0, Name-ALT
- SVG
  - ID: SVG1
  - ImageRendering: optimizeSpeed
  - ColorRendering: optimzeSpeed
  - OnMouseDown: event
  - Stroke-Opacity: 80
  - Height: 1000
  - FontSize: 14
  - FontStyle: Normal
  - EnableBackground: false
- SVG Description
  - ID: D1
  - Space: default
  - Language: en-us
  - Base: <http://www.altova.com>
  - Class: Custom
  - Style: Normal

Instance Document: [TC-7\\_DefineCustomDisplayObject.ddml](#)

Relevant User Stories: [US-1](#), [US-6](#), [US-7](#), [US-8](#), [US-9](#), [US-41](#), [US-42](#), [US-51](#), [US-52](#)

Deficiencies: [D-1](#)

TC-8. Define Bar Chart Display Object

Details: Define a bar chart display object with axes within a model

- Display Object Name: BAGC1
- ID: BAGC1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Bar Chart
- Title Font: Arial
- Title Font Size: 24
- Title Color: 65280
- Dynamics:
  - Type: built-in

- Variable Use
  - ID: vu0
  - Name: TM1F001F
  - Pool Ref: var19
  - Color: 65280
- Rules
  - If < 0, Value = 0
- X Axis:
  - Color: 0
  - Type: TIME
- Y Axis:
  - Type: VALUE
  - Min: 0
  - Max: 5
  - Label Foreground Color: 0
  - Axis Grid 1:
    - Grid Interval: 2.5
    - Grid Color: 16711935
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 2:
    - Grid Color: 58302
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 3:
    - Tick Label Color: 0
    - Tick Label Format: %4.2f
- Scroll Direction: DOWN

Instance Document: TC-8\_DefineBarChartDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-14](#), [US-27](#), [US-28](#), [US-29](#), [US-31](#), [US-33](#), [US-34](#), [US-36](#), [US-37](#), [US-39](#), [US-40](#), [US-41](#), [US-42](#), [US-43](#), [US-44](#), [US-45](#), [US-53](#), [US-54](#), [US-55](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-9. Define Frequency Plot Display Object

Details: Define a frequency plot display object with axes within a model

- Display Object Name: FREQ1
- ID: FREQ1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000

- Point4: 750000, 0
- Title: Frequency Plot
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0
- Frequency Axis:
  - Color: 0
  - Type: TIME
  - Unit: Radians
- Magnitude Axis:
  - Type: VALUE
  - Min: 0
  - Max: 100
  - Unit: Volts
  - Label Foreground Color: 0
  - Axis Grid 1:
    - Grid Interval: 2.5
    - Grid Color: 16711935
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 2:
    - Grid Color: 58302
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 3:
    - Tick Label Color: 0
    - Tick Label Format: %4.2f

Instance Document: TC-9\_DefineFrequencyPlotDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-27](#), [US-28](#), [US-29](#), [US-31](#), [US-32](#), [US-33](#), [US-34](#), [US-36](#), [US-37](#), [US-39](#), [US-40](#), [US-41](#), [US-42](#), [US-56](#), [US-57](#), [US-58](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-10. Define Button Display Object  
Details: Define a button display object within a model

- Display Object Name: BTN1

- ID: BTN1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Button
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0
- Label: Button

Instance Document: TC-10\_DefineButtonDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-41](#), [US-42](#), [US-46](#), [US-47](#), [US-48](#), [US-49](#), [US-50](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: [D-6](#)

TC-11. Define Dial Display Object

Details: Define a dial display object within a model

- Display Object Name: DIAL1
- ID: DIAL1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Dial
- Title Color: 16711680
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0
- Axis Type: VALUE
- Minimum Angle: 30 degrees

- Maximum Angle: 120 degrees

Instance Document: TC-11\_DefineDialDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-41](#), [US-42](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-12. Define HUD Display Object  
Details: Define HUD display object within a model

- Display Object Name: HUD1
- ID: HUD1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Heads-Up Display
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0
- Sub-Model Use
  - ID: MOD2
  - Point1: 0, 0
  - Point2: 0, 1000
  - Point3: 1000, 1000

Instance Document: TC-12\_DefineHUDDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-4](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-23](#), [US-41](#), [US-42](#), [US-61](#), [US-62](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-13. Define Pie Chart Display Object  
Details: Define a pie chart display object within a model

- Display Object Name: PIE1
- ID: PIE1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0

- Title: Pie Chart
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If >1 AND <100 continue
    - Else Value = 0
  - MathML
    - Divide by 100

Instance Document: TC-13\_DefinePieChartDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-41](#), [US-42](#), [US-63](#), [US-64](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-89](#), [US-91](#), [US-94](#), [US-95](#)

Deficiencies: N/A

TC-14. Define Slider Display Object

Details: Define a slider object within a model inside a grid

- Grid ID: GRID1
- Grid Name: GRID1
- Grid Rows: 2
- Grid Columns: 2
- Grid Point1: 0, 0
- Grid Point2: 10000, 0
- Grid Point3: 10000, 10000
- Display Object Name: SLID1
- Slider ID: SLID1
- Title: Slider
- Range: 0 to 100 mm
- Slider Grid Row: 1
- Slider Grid Column: 1
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0

- Axis Type: VALUE
- Orientation: HORIZONTAL

Instance Document: TC-14\_DefineSliderDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-16](#), [US-17](#), [US-41](#), [US-42](#), [US-67](#), [US-68](#), [US-69](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: [D-3](#), [D-4](#)

TC-15. Define Strip Chart Display Object

Details: Define a strip chart display object within a model

- Display Object Name: STPC1
- ID: STPC1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Strip Chart
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name: TM1F001F
    - Pool Ref: var19
    - Color: 65280
  - Rules
    - If < 0, Value = 0
- X Axis:
  - Color: 0
  - Type: TIME
- Y Axis:
  - Type: VALUE
  - Min: 0
  - Max: 5
  - Label Foreground Color: 0
  - Label Font Size: 10
  - Label Font: Arial
  - Axis Grid 1:
    - Grid Interval: 2.5
    - Grid Color: 16711935
    - Tick Color: 0
    - Tick Label Format: %4.2f
    - Show Ticks: false
    - Tick Label Font: Arial

- Tick Label Size: 8
- Axis Grid 2:
  - Grid Color: 58302
  - Tick Color: 0
  - Tick Label Format: %4.2f
- Axis Grid 3:
  - Tick Label Color: 0
  - Tick Label Format: %4.2f
- Scroll Direction: DOWN
- Frame Scroll Duration: 10

Instance Document: TC-15\_DefineStripChartDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-27](#), [US-28](#), [US-29](#), [US-30](#), [US-31](#), [US-33](#), [US-34](#), [US-36](#), [US-37](#), [US-39](#), [US-40](#), [US-41](#), [US-42](#), [US-70](#), [US-71](#), [US-72](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-16. Define XY Chart Display Object

Details: Define *xy* chart display object within a model

- Display Object Name: XY1
- ID: XY1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: XY Chart
- Background Color: 0
- Data Area Color: 16777215
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name:vu0
    - Pool Ref: var19
    - Color: 65280
  - Variable Use
    - ID: vu1
    - Name: vu1
    - Pool Ref: var14
    - Color: 255
  - Rules
    - If < 0, Value = 0
- X Axis:

- Color: 0
- Type: TIME
- Y Axis:
  - Type: VALUE
  - Min: 0
  - Max: 1000
  - Label Foreground Color: 0
  - Axis Grid 1:
    - Grid Interval: 2.5
    - Grid Color: 16711935
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 2:
    - Grid Color: 58302
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 3:
    - Tick Label Color: 0
    - Tick Label Format: %4.2f
- Plot Type: LINES
- X Variable: TMBUF100
- Y Variable: TMBUF103

Instance Document: TC-16\_DefineXYChartDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-2](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-15](#), [US-27](#), [US-28](#), [US-29](#), [US-31](#), [US-33](#), [US-34](#), [US-35](#), [US-36](#), [US-37](#), [US-38](#), [US-39](#), [US-40](#), [US-41](#), [US-42](#), [US-81](#), [US-82](#), [US-83](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-17. Define Radial Chart Display Object

Details: Define radial chart display object within a model

- Display Object Name: RAD1
- ID: RAD1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Radial Chart
- Background Color: 0
- Data Area Color: 16777215
- Dynamics:
  - Type: built-in
  - Variable Use

- ID: vu0
    - Name:vu0
    - Pool Ref: var19
    - Color: 65280
  - Variable Use
    - ID: vu1
    - Name: vu1
    - Pool Ref: var14
    - Color: 255
  - Rules
    - If < 0, Value = 0
- Circular Axis:
  - Color: 0
  - Type: TIME
- Radial Axis:
  - Type: VALUE
  - Min: 0
  - Max: 1000
  - Label Foreground Color: 0
- Plot Type: LINES
- X Variable: TMBUF100
- Y Variable: TMBUF103

Instance Document: TC-17\_DefineRadialChartDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-2](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-15](#), [US-27](#), [US-28](#), [US-29](#), [US-31](#), [US-40](#), [US-41](#), [US-42](#), [US-65](#), [US-66](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

TC-18. Define Frequency Response Plot Display Object

Details: Define a frequency response plot display object with axes within a model

- Display Object Name: FREQR1
- ID: FREQR1
- Point1: 500000, 0
- Point2: 500000, 250000
- Point3: 750000, 250000
- Point4: 750000, 0
- Title: Frequency Response Plot
- Dynamics:
  - Type: built-in
  - Variable Use
    - ID: vu0
    - Name:TM1F001F

- Pool Ref: var19
  - Color: 65280
  - Rules
    - If < 0, Value = 0
- Frequency Axis:
  - Color: 0
  - Type: TIME
  - Unit: Radians
- Magnitude Axis:
  - Type: VALUE
  - Min: 0
  - Max: 100
  - Unit: Volts
  - Label Foreground Color: 0
  - Axis Grid 1:
    - Grid Interval: 2.5
    - Grid Color: 16711935
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 2:
    - Grid Color: 58302
    - Tick Color: 0
    - Tick Label Format: %4.2f
  - Axis Grid 3:
    - Tick Label Color: 0
    - Tick Label Format: %4.2f
- Phase Axis:
  - Type: VALUE

Instance Document: TC-18\_DefineFrequencyResponsePlotDisplayObject.ddml

Relevant User Stories: [US-1](#), [US-5](#), [US-6](#), [US-7](#), [US-12](#), [US-13](#), [US-27](#), [US-28](#), [US-29](#), [US-31](#), [US-32](#), [US-33](#), [US-34](#), [US-36](#), [US-37](#), [US-39](#), [US-40](#), [US-41](#), [US-42](#), [US-84](#), [US-85](#), [US-86](#), [US-87](#), [US-88](#), [US-91](#)

Deficiencies: N/A

## 5. Deficiencies

- D-1. The param element requires a param sub-element that causes an infinite loop, making it so instance documents that use the element cannot be validated.  
Relevant Test Cases: [TC-2](#), [TC-4](#), [TC-5](#), [TC-6](#), [TC-7](#)  
Resolution: Change the schema so that a param sub-element is not required for every param element.
- D-2. Map bounds can only be stored as latitude and longitude values.

Relevant Test Cases: [T-6](#)

Resolution: Add various formats for storing boundaries.

D-3. If display objects are arranged in a grid, gridlines cannot be shown.

Relevant Test Cases: [TC-14](#)

Resolution: Add a show gridlines element to the schema. Alternatively, add a border display object that can be placed around other display objects.

D-4. If display objects are arranged in a grid, gridlines color cannot be assigned.

Relevant Test Cases: [TC-14](#)

Resolution: Add a gridline color element. Alternatively, add a border display object that can be placed around other display objects with color selection.

D-5. Display objects cannot be hidden.

Relevant Test Cases: All

Resolution: Add a show display object element to the schema.

D-6. Custom functions cannot be stored.

Relevant Test Cases: N/A

Resolution: Add a new schema for storing function information.

This page intentionally left blank.

## APPENDIX A

### Citations

Range Commanders Council. “Telemetry Attributes Transfer Standard,” in *Telemetry Standards*. IRIG 106-15. June 2015. May be superseded by update. Retrieved 22 July 2015. Available at [http://www.wsmr.army.mil/RCCsite/Documents/106-15\\_Telemetry\\_Standards/Chapter9.pdf](http://www.wsmr.army.mil/RCCsite/Documents/106-15_Telemetry_Standards/Chapter9.pdf).

———. *XML Style Guide*. 125-15. July 2015. May be superseded by update. Retrieved 28 January 2016. Available at [http://www.wsmr.army.mil/RCCsite/Documents/125-15\\_XML\\_Style\\_Guide/125-15\\_XML\\_Style\\_Guide.pdf](http://www.wsmr.army.mil/RCCsite/Documents/125-15_XML_Style_Guide/125-15_XML_Style_Guide.pdf).

**\*\*\* END OF DOCUMENT \*\*\***