

CHAPTER 23

Metadata Configuration

23. CHAPTER 23	23-1
23.1 Introduction.....	23-1
23.2 Metadata Description Language	23-1
23.2.1 MDL Schema Concepts	23-1
23.2.1.1 Conditional elements in the MDL Schema Definition File.....	23-2
23.2.1.2 Naming Conventions in the MDL Schema Definition File.....	23-2
23.2.1.3 Indexing Policies	23-2
23.2.1.4 Uniqueness of ID Attributes.....	23-2
23.2.1.5 Extending MDL with Supplementary XML-Based Standards	23-2
23.2.1.6 Usage of Telemetry Attributes Transfer Standard in MDL	23-2
23.2.1.7 Usage of DirtyBit Element	23-3
23.2.1.8 Usage of GenericParameter Element	23-3
23.2.2 MDL Global Element Glossary	23-3
Appendix 23A: MDL Examples.....	23-4
Appendix 23B: MDL Relationships to Chapters	23-5

Distribution Statement A

Approved for public release; distribution unlimited.

23. CHAPTER 23

Metadata Configuration

23.1 Introduction

This chapter describes system configuration data for network-based telemetry systems in a common fashion, and provides a language for describing the configuration of all of the components in a telemetry system, as well as their logical and physical interrelationships. The language is intended to be expressive enough to describe a wide variety of systems: large and small, simple and complex, from the low-level transducer-to-measurement association for an acquisition card on a Data Acquisition Unit (DAU) up to network topology of multiple test mission networks.

This chapter defines a language, the Metadata Description Language (MDL). The MDL syntax defines vocabulary and sentence structure, while the MDL semantics provide meaning. Using the MDL syntax and semantics, *MDL Instance Documents* can be created to describe telemetry systems. The descriptions may be used in various ways: to configure components, to predict the performance of the network, or to capture requirements for the telemetry system. Additionally, the MDL provides a common exchange language that facilitates the interchange of configuration information between telemetry system components.

23.2 Metadata Description Language

The MDL is described only as an Extensible Markup Language (XML) schema. The MDL XML language schema consists of an XML Schema Document (XSD) file that defines the structure of valid *MDL Instance Documents*. The schema is available [here](#). Additionally, an automatic transformation process is applied to the schema in order to create a graphical depiction of the schema in Hyper Text Markup Language (HTML) format. The HTML depiction is available [here](#).

NOTE



The MDL XML based schema is not intended to be read in a plain text fashion. The HTML graphical depiction is provided as an aid for those desiring to read the schema. The HTML files created to depict the MDL schema are very large and may take time to download.

23.2.1 MDL Schema Concepts

The MDL schema defines a syntax, which includes a vocabulary, a set of types, and the rules for how an *MDL Instance Document* shall be structured. The syntax definition is realized using the XML Schema standard, which is maintained by the World Wide Web Consortium (W3C -- www.w3.org).


NOTE



The MDL builds upon the existing schema standard of the World Wide Web consortium. Readers unfamiliar with W3C schemas as a whole are encouraged to study basic concepts from the W3C prior to attempting to understand the MDL.


Constraints as defined by the W3C are a part of the MDL schema. In the MDL schema implementation, these constraints are distributed between statements that are syntax related (encoded and enforced by the schema) and statements that are semantic related (additional rules that are levied and provide meaning). The syntax of the language is enforced using W3C XML Schema constraints.

When possible, XML mechanisms are used to enforce semantic constraints. In cases not supported cleanly by XML, text has been added directly to the MDL schema documentation. In such cases, the text will typically include the keyword “shall”. The phrase “the value of the **Name** element of the **Measurement** element shall be unique” is one such example.

 NOTE	<p>Not all constraints that must be met in order for an MDL file to be valid for configuration can be expressed by W3C constraints. Plain text is used in the schema to describe such cases. Additionally, applications consuming or generating MDL instance documents are expected to assure that the files are valid.</p>
---	---

23.2.1.1 Conditional elements in the MDL Schema Definition File

The MDL schema is a system-level description. Not all components require every element to properly configure. In such cases, these elements are conditional. The documentation specifies when the conditional elements must and shall be present and processed. Components not specifically called out in documentation of conditional elements shall not fail to configure if the particular conditional element is not present.

 NOTE	<p>Use of “conditional” over “optional” is an intentional style chosen for the MDL grammar. Conditional is preferred in order remove ambiguity concerning grammar elements that must be considered by devices exchanging MDL files that are to be used for configuration.</p>
---	---

23.2.1.2 Naming Conventions in the MDL Schema Definition File

In the MDL schema definition file MDL elements and attributes appear as instances of defined **xsd:complexType** and **xsd:simpleType** elements. Each declaration of these MDL-specific **xsd:complexType** and **xsd:simpleType** elements will specify a **name** attribute that is assigned a string that contains the name of the MDL element being described followed by a string suffix of “Type” or “Enum”. For example, the top-level element of the MDL schema is the **MDLRoot** element. In the MDL schema definition file, this element appears as an instance of an **xsd:complexType** element with a **name** attribute of “MDLRootType”. These **name** attribute strings that correspond to the defined MDL elements only appear in the MDL schema definition file.

23.2.1.3 Indexing Policies

Numerical indices present in an *MDL Instance Document* are recommended to count starting at 1 and to increment by one (i.e., 1, 2, 3, 4,...).

23.2.1.4 Uniqueness of ID Attributes

Values of **id** attributes of any element in an *MDL Instance Document* shall be unique within an *MDL Instance Document*. The **id** attributes are used to implement references.

23.2.1.5 Extending MDL with Supplementary XML-Based Standards

The use of other XML-based standards (i.e., other schemas) in conjunction with the MDL schema is permitted. Potentially, the use of these external standards through their accompanying schemas leverages existing knowledge to describe concepts and domains beyond those in the MDL. The MDL does not explicitly constrain the available mechanisms to use external standards; however, the linking of external schemas to the MDL schema shall not result in the modification of the MDL schema.

23.2.1.6 Usage of Telemetry Attributes Transfer Standard in MDL

The MDL schema requires the **tmatsP:PCMFormatAttributesType** to describe Pulse Code Modulation (PCM) measurements, and it is imported directly from the Telemetry Attributes Transfer Standard (TMATS) schema. The TMATS schema files are included with the MDL schema for convenience, and are also available here.

23.2.1.7 Usage of DirtyBit Element

The **DirtyBit** element is used by a component to indicate that one or more values in the reported MDL configuration differ from the *MDL Instance Document* with which the component was configured. Differences may be introduced through System Management interfaces, or other vendor specific interfaces. A value of “true” in a reported *MDL Instance Document* indicates that configuration changes have been made that cause the MDL representation to differ from the last configuration via MDL.

23.2.1.8 Usage of GenericParameter Element

The **GenericParameter** element allows the description of additional information outside the scope of the MDL, and may also be used to document decisions made to arrive at a vendor-specific configuration. **GenericParameter** shall be used to extend the MDL grammar when the MDL grammar cannot support those required concepts natively, or as a key so that vendor tools can arrive at the same configuration as in a previous run.

23.2.2 MDL Global Element Glossary

The MDL schema contains a large number of **xsd:documentation** elements that describe the intent, purpose, or usage of the elements in MDL. These embedded annotations collectively form the global element glossary for the MDL schema. The glossary can be viewed here, and is automatically produced from the schema file by way of an Extensible Stylesheet Language (XSLT) stylesheet which renders the documentation as HTML.

Appendix 23A: MDL Examples

Example MDL files (*XML Instance Documents*), and associated documentation, are here. As with most grammars, it is expected that the examples will be very useful in clarifying the expected use of MDL. However, the documentation of the schema takes precedence over concepts or details that may be inferred through the examples.

DRAFT

Appendix 23B: MDL Relationships to Chapters

MDL is used to describe system configuration data for network-based telemetry systems; therefore, it includes elements to describe the concepts presented across Chapters 22 through 28. Table 23B-1 provides a mapping of the relevant top-level MDL elements and their relationships to these chapters.

Table 23B-1.

MDLRoot Element	Relationship	IRIG 106 Chapter
TestMissions	Data Protocol	22: Network-Based Protocol Suite
	Management	26: TmNSDataMessage Transfer Protocol
MeasurementDomains	Message Internals	24: Message Formats
NetworkDomains	Network Shape	22: Network-Based Protocol Suite
	Network Use	25: Management Resources
	Network Management	26: TmNSDataMessage Transfer Protocol
RANConfigs	Radio Links	27: RF Network Access Layer
	Radio Management	28: RF Network Management
DSCPTable	Quality of Service	22: Network-Based Protocol Suite