CSDR Quantity Data Report

Version 1.0

File Format Specification

March 5, 2019

1 Overview 2
2 Data Model 2
2.1 Data Model Conventions 2
2.2 Tables 4
2.3 Primitive Data Types 9
2.4 Enumerations 10
3 File Format 12
3.1 File Conventions 12
3.2 File Contents 12
3.3 File Type/Version 12
4 Representation in JSON 13
4.1 JSON Conventions 13
4.2 JSON Schema Sample 13
4.3 JSON Data Sample 14
5 References 14

1 Overview

This document is intended as a technical reference for computer programmers implementing software to support the exchange of data composing the CSDR Quantity Data Report.

2 Data Model

Data in a CSDR Quantity Data Report are modeled as a collection of tables related by primary and foreign key constraints.

2.1 Data Model Conventions

2.1.1 Tables

A table has a collection of fields and a collection of records. Each field has a name and a primitive data type. Each record has a collection of field values. Each field value must be consistent with the data type of the corresponding field. Records have an implicit sequence.

2.1.2 Nullable Fields

Records may omit values for fields that are nullable but must include values for fields that are not nullable. Unless otherwise noted, omitted field values may be interpreted as null, undefined, or “not applicable”. All of these interpretations are considered equivalent.

2.1.3 Key Constraints

A primary key constraint defines a collection of fields for a table such that the corresponding field values uniquely identify each record. No two records may have the same collection of corresponding field values.

A foreign key constraint defines a correspondence between a field in one table and a field in another table such that the field value for each record in the former must match the field value for some record in the latter.

2.1.4 Enumerations

Enumerations are implicit lookup tables available for use with foreign key constraints.

2.1.5 Singletons

Singletons are tables with exactly one record.

2.1.6 Strings

Strings are sequences are of text characters defined by the Unicode standard.

The following control characters are prohibited in all string values: (U+0000–U+0008), (U+000B–U+000C), (U+000E–U+001F), (U+007F). String values used as ID's are further limited to a character set representing common printable characters (U+0020–U+007E).

Most string values must have normalized whitespace. A string value with normalized whitespace cannot begin or end with whitespace characters, cannot contain any whitespace characters other than the space character (U+0020), and cannot contain any sequence of two or more contiguous space characters.

Normalized whitespace is not required for string values used for remarks or other expository text.

Comparison of string values for the purpose of verifying key constraints is not case sensitive.

Empty string values are interpreted as equivalent to null values for fields with a string data type. Records must include non-empty string values for string fields that cannot be null.

2.1.7 WBS Hierarchical Structure

The hierarchical structure of the WBS is determined based on the level and sequence of the elements reported. Specifically, elements must be sorted in a manner consistent with a depth-first search of the element hierarchy, such that the parent element of a given element must be the nearest preceding element with a reported level less than that of the given element. Each element that succeeds another must have a level that is no more than 1 greater than the level of the preceding element. The minimum level is 1.

2.2 Tables

2.2.1 ReportMetadata

Table	ReportMetadata		
Entity	ReportMetadata		
Fields	Name	Data Type	Nullable
	SecurityClassification	String	No
	ProprietaryStatement	String	Yes
	ProgramName	String	Yes
	PhaseOrMilestoneID	StringID	Yes
	PrimeMissionProduct	String	Yes
	CommodityType	String	Yes
	ReportingOrganization_OrganizationName	String	Yes
	ReportingOrganization_DivisionName	String	Yes
	ReportingOrganization_CageCode	String	Yes
	ReportingOrganization_Location_Street	Text	Yes
	ReportingOrganization_Location_City	String	Yes
	ReportingOrganization_Location_State	String	Yes
	ReportingOrganization_Location_ZipCode	String	Yes
	ReportingOrganization_Location_Country	String	Yes
	ApprovedPlanNumber	String	Yes
	ApprovedPlanRevisionNumber	String	Yes
	CustomerName	String	Yes
	ContractTypeID	StringID	Yes
	ContractPrice	Decimal	Yes
	ContractCeiling	Decimal	Yes
	ContractNumber	String	Yes
	PeriodOfPerformance_StartDate	Date	Yes
	PeriodOfPerformance_EndDate	Date	Yes
	ReportCycleID	StringID	Yes
	SubmissionEvent_Number	Integer	Yes
	SubmissionEvent_Name	String	Yes
	SubmissionEvent_IsWildcard	Boolean	Yes
	ResubmissionNumber	Integer	Yes
	ReportAsOf	Date	Yes
	PointOfContact_Name	String	Yes
	PointOfContact_Department	String	Yes
	PointOfContact_TelephoneNumber	String	Yes
	PointOfContact_EmailAddress	String	Yes
	DatePrepared	Date	Yes
Primary Key	[N/A]		
Foreign Keys	PhaseOrMilestoneID: PhaseOrMilestoneEnum(ID) ContractTypeID: ContractTypeEnum(ID) ReportCycleID: ReportCycleEnum(ID)		
Use Constraints	ReportMetadata is a singleton.		

2.2.2 OrdersOrLots

Table	OrdersOrLots		
Entity	OrderOrLot		
Fields	Name	Data Type	Nullable
	ID	StringID	No
	Name	String	No
	PhaseOrMilestoneID	StringID	Yes
	CustomerName	String	Yes
	ContractTypeID	StringID	Yes
	ContractPrice	Decimal	Yes
	ContractCeiling	Decimal	Yes
	PeriodOfPerformance_StartDate	Date	Yes
	PeriodOfPerformance_EndDate	Date	Yes
	AppropriationTypeID	StringID	Yes
Primary Key	ID		
Foreign Keys	PhaseOrMilestoneID: PhaseOrMilestoneEnum(ID) ContractTypeID: ContractTypeEnum(ID) AppropriationTypeID: AppropriationTypeEnum(ID)		
Use Constraints			

2.2.3 EndItems

Table	EndItems		
Entity	EndItem		
Fields	Name	Data Type	Nullable
	ID	StringID	No
	Name	String	No
Primary Key	ID		
Foreign Keys	[N/A]		
Use Constraints			

2.2.4 WBS

Table	WBS		
Entity	WBSElement		
Fields	Name	Data Type	Nullable
	Level	Integer	No
	ID	StringID	No
	Name	String	No
	ParentID	StringID	Conditional
Primary Key	ID		
Foreign Keys	ParentID: WBSElement(ID)		
Use Constraints	Order of records is significant. In particular, hierarchical structure is determined based on level and sequence of records. See above. The first record must have Level equal to 1. All other records must have Level greater than 1. The first record must have a null ParentID. All other records must have a ParentID equal to the ID of the parent record determined by the hierarchical structure.		

2.2.5 QuantitiesAtCompletion

Table	QuantitiesAtCompletion		
Entity	QuantitiesAtCompletionRecord		
Fields	Name	Data Type	Nullable
	OrderOrLotID	StringID	No
	EndItemID	StringID	No
	WBSElementID	StringID	No
	DeliveredQuantityAtCompletion	Decimal	No
	InternalQuantityAtCompletion	Decimal	No
	CoproductionOrConcurrentQuantityAtCompletion	Decimal	No
	GFEQuantityAtCompletion	Decimal	No
Primary Key	OrderOrLotID, EndItemID, WBSElementID		
Foreign Keys	OrderOrLotID: OrderOrLot(ID) EndItemID: EndItem(ID) WBSElementID: WBSElement(ID)		
Use Constraints			

2.2.6 QuantitiesToDate

Table	QuantitiesToDate		
Entity	QuantitiesToDateRecord		
Fields	Name	Data Type	Nullable
	OrderOrLotID	StringID	No
	WBSElementID	StringID	No
	CompletedQuantityToDate	Decimal	No
	InProcessQuantity	Decimal	No
Primary Key	OrderOrLotID, WBSElementID		
Foreign Keys	OrderOrLotID: OrderOrLot(ID) WBSElementID: WBSElement(ID)		
Use Constraints			

2.2.7 ProductionSequence

Table	ProductionSequence		
Entity	ProductionSequenceSegment		
Fields	Name	Data Type	Nullable
	EndItemID	StringID	No
	FirstUnitNumber	Integer	No
	LastUnitNumber	Integer	No
	OrderOrLotID	StringID	No
	IsInternal	Boolean	No
Primary Key	EndItemID, FirstUnitNumber		
Foreign Keys	EndItemID: EndItem(ID) OrderOrLotID: OrderOrLot(ID)		
Use Constraints			

2.2.8 SummaryRemarks

Table	SummaryRemarks		
Entity	SummaryRemark		
Fields	Name	Data Type	Nullable
	OrderOrLotID	StringID	No
	Text	Text	Yes
Primary Key	OrderOrLotID		
Foreign Keys	OrderOrLotID: OrderOrLot(ID)		
Use Constraints			

2.2.9 WBSElementRemarks

Table	WBSElementRemarks		
Entity	WBSElementRemark		
Fields	Name	Data Type	Nullable
	OrderOrLotID	StringID	No
	WBSElementID	StringID	No
	Text	Text	Yes
Primary Key	OrderOrLotID, WBSElementID		
Foreign Keys	OrderOrLotID: OrderOrLot(ID) WBSElementID: WBSElement(ID)		
Use Constraints			

2.3 Primitive Data Types

Primitive Data Types	
Boolean	Values of two-valued logic (i.e. “true” and “false”).
Date	Year, month, and day, without reference to the time of day or a specific time zone.
Decimal	Number that can be represented with decimal digits, with possible integral and/or fractional component.
Integer	Number that can be represented with decimal digits, with no fractional component.
String	A sequence of Unicode characters, with normalized whitespace.
StringID	A sequence of Unicode characters, with normalized whitespace and limited character set.
Text	A sequence of Unicode characters intended for remarks or other expository text.

2.4 Enumerations

2.4.1 PhaseOrMilestoneEnum

Enumeration	PhaseOrMilestoneEnum	
Values	ID	Name
	PRE_A	Pre-A
	A	A
	B	B
	C_LRIP	C-LRIP
	C_FRP	C-FRP
	O_AND_S	O&S
	MULTIPLE	Multiple
Use Constraints		

2.4.2 ContractTypeEnum

Enumeration	ContractTypeEnum	
Values	ID	Name
	CS	Cost Sharing
	CPAF	Cost Plus Award Fee
	CPFF	Cost Plus Fixed Fee
	CPIF	Cost Plus Incentive Fee
	CPIF_PI	Cost Plus Incentive Fee (with Performance Incentives)
	FFP	Firm Fixed Price
	FPIF	Fixed Price Incentive, Firm Target
	FPIST	Fixed Price Incentive, Successive Targets
	FPIST_PI	Fixed Price Incentive, Successive Targets (with Performance Incentives)
	FPIFT_PI	Fixed Price Incentive, Firm Targets (with Performance Incentives)
	FPAF	Fixed Price Award Fee
	FP_EPA	Fixed Price with Economic Price Adjustment
	FP_PPR	Fixed Price with Prospective Price Redetermination
	FCP_RPR	Fixed Ceiling Price with Retroactive Price Redetermination
	FFP_LOET	Firm Fixed Price, Level of Effort Term
	IDIQ	Indefinite Delivery Indefinite Quantity
	LC	Letter Contract and Undefinitized Contractual Action (UCA)
	TM	Time and Materials
	OTHER	Other
	MULTIPLE	Multiple
Use Constraints		

2.4.3 AppropriationTypeEnum

Enumeration	AppropriationTypeEnum	
Values	ID	Name
	RDTE	RDT&E
	PROCUREMENT	Procurement
	O_AND_M	O&M
Use Constraints		

2.4.4 ReportCycleEnum

Enumeration	ReportCycleEnum	
Values	ID	Name
	INITIAL	Initial
	INTERIM	Interim
	FINAL	Final
Use Constraints		

3 File Format

The file format for a CSDR Quantity Data Report is a ZIP file containing multiple text file entries. One text file entry conveys type and version information. All other text file entries convey data represented in JSON. Each JSON file entry corresponds to a single data table.

3.1 File Conventions

Text file entries must be encoded in UTF-8. Compressed file entries must be compressed using the DEFLATE compression method. File entries must not be encrypted.

The file entry for a table may be omitted if the table has no records. The file entry for a singleton may be omitted if all its fields are null. The file entry for type/version information must always be included.

3.2 File Contents

ZIP File Entries		
	Name	Table
	FileType.txt	[N/A]
	ReportMetadata.json	ReportMetadata
	OrdersOrLots.json	OrdersOrLots
	EndItems.json	EndItems
	WBS.json	WBS
	QuantitiesAtCompletion.json	QuantitiesAtCompletion
	QuantitiesToDate.json	QuantitiesToDate
	ProductionSequence.json	ProductionSequence
	SummaryRemarks.json	SummaryRemarks
	WBSElementRemarks.json	WBSElementRemarks

3.3 File Type/Version

The 'FileType.txt' file entry specifies the type and version of the CSDR Quantity Data Report file. This file entry must contain the following exact text string (excluding quotation marks):
"CSDR_QUANTITY_REPORT/1.0".

4 Representation in JSON

4.1 JSON Conventions

Each table is represented in JSON as an array of objects. Singletons are an exception. They are represented directly as a single object. Objects correspond to records, and the sequence of objects in JSON represents the implicit sequence of records in the table.

Each record is represented in JSON as an object with name/value pairs corresponding to field values. The name of each pair must exactly match the name of the corresponding field, and the value of each pair must follow the conventions below for representing the corresponding primitive data type in JSON. Names must be unique within the scope of each object and each name must correspond to a field defined for the table.

Objects must include name/value pairs for fields that are not null, and these pairs must not have a JSON value of null or an empty JSON string value. Conversely, objects may or may not include name/value pairs for fields that are null. If included, these pairs must have a JSON value of null or an empty JSON string value. Only pairs for fields with a primitive data type of String, StringID, or Text may have an empty JSON string value.

Primitive data types are represented as follows:

Representation of Primitive Data Types	
Boolean	JSON value of true or false.
Date	JSON string encoding a valid date, without time component or time zone, formatted as follows: “yyyy-mm-dd” (e.g. “2016-01-31”).
Decimal	JSON number.
Integer	JSON number with fractional component equal to zero.
String	JSON string, with normalized whitespace.
StringID	JSON string, with normalized whitespace and limited character set.
Text	JSON string.

4.2 JSON Schema Sample

Schema	WBS.json

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "Level":    {"type": "number"},
            "ID":       {"type": "string"},
            "Name":     {"type": "string"},
            "ParentID": {"type": ["string", "null"]}
        },
        "required": ["Level", "ID", "Name"]
    }
}
```

4.3 JSON Data Sample

Data	WBS.json

```
[
    {
        "Level": 1,
        "ID": "1.0",
        "Name": "Total"
    },
    {
        "Level": 2,
        "ID": "1.1",
        "Name": "Subsystem 1.1",
        "ParentID": "1.0"
    },
    {
        "Level": 2,
        "ID": "1.2",
        "Name": "Subsystem 1.2",
        "ParentID": "1.0"
    }
]
```

5 References

JSON - The JSON Data Interchange Format, ECMA-404. 2013.
JSON Schema - json-schema.org
Unicode - The Unicode Standard, Version 9.0. 2016.
UTF-8 - “UTF-8 encoding scheme,” The Unicode Standard, Version 9.0, §3.10 D95. 2016.
ZIP File Format - .ZIP File Format Specification, Version 6.3.4. 2014.