TREK

METADATA

USER GUIDE

October 2017

Approved for Public Release; Distribution is Unlimited.

TABLE OF CONTENTS

PARAGRAPH PAGE

1 Welcome 1

1.1 Getting Started 1

2 Technical Support 1

3 Introduction 1

4 Overview of the User Interface 4

4.1 Main Window 4
4.2 Toolbar 5
4.3 Menus 5

5 Quick Start Guides 6

5.1 How to Create a Parameter Collection 6
5.2 How to Create a Database 6
5.3 How to View the List of Packets in a Database 6
5.4 How to Export a Packet from a Database 7
5.5 How to Add a Packet to a Database 7
5.6 How to Create a Limit Alarm 7
5.7 How to Create a Packet 8
5.8 How to Import Data from HOSC Partial Database Download Files Into a TReK Database 8

6 Details 9

6.1 Parameter Collection 9
6.2 Random Packet Collection 12
6.3 Database 15
6.4 Limit Alarm 19
6.5 Expected State Alarm 20
6.6 Polynomial Calibrator 22
6.7 Spline Calibrator 25
6.8 Enumerator 27
6.9 Range Enumerator 29
6.10 Packet 31
6.11 Export 35
6.12 Import 36
6.13 Parameter Details 37
6.14 Packet Attributes 38
6.15 Convert EHS Partial Database Files Into TReK Database 38
6.16 Export Metadata to CDEF 40
6.16.1 General Tab 41
6.16.2 Packet List Tab 42
6.16.3 ISS System Subsets Tab 43
6.17 Import Metadata from CDEF 44
6.17.1 General Tab 45
6.17.2 ISS System Subsets Tab 45
6.18 Command Data Exchange Format Information 46
6.19 User Defined C Structure 47
6.20 Application Messages 50
6.21 Application Configuration File 52

6.22 Application Settings 52

7 FAQ and Troubleshooting 52

FIGURES

FIGURE PAGE

Figure 1 Packets, Parameter Collections, and Parameters 2
Figure 2 A Packet and Its Zones 2
Figure 3 Main Window 4
Figure 4 Main Window in Collection Configuration 9
Figure 5 Parameter Collection Drag and Drop 10
Figure 6 Collection with Parameter 11
Figure 7 Random Packet Collection 13
Figure 8 Random Packet Collection with Packet 14
Figure 9 Database Configuration 16
Figure 10 Database Packets Shown In List 17
Figure 11 Limit Alarm Configuration 19
Figure 12 Limit Alarm Level 20
Figure 13 Expected State Alarm Configuration 21
Figure 14 Polynomial Calibrator Configuration 23
Figure 15 Power Coefficient List Populate when Order is Set to 5 24
Figure 16 Spline Calibrator Configuration 25
Figure 17 Spline Calibrator Information 26
Figure 18 Enumerator Configuration 27
Figure 19 Populated Enumerator 28
Figure 20 Range Enumerator Configuration 29
Figure 21 Populated Range Enumerator 30
Figure 22 Packet Configuration 32
Figure 23 Packet Drop Zones 33
Figure 24 Entire Packet Drop Zone Highlighted 33
Figure 25 Data Zone Drop Zone Highlighted 33
Figure 26 Packet Contents 34
Figure 27 Export Dialog 36
Figure 28 Import Dialog 36
Figure 29 Parameter Details Dialog 37
Figure 30 Packet Attributes Dialog 38
Figure 31 Convert EHS Partial Database Files Into TReK Database Dialog 39
Figure 32 Export Metadata to CDEF Dialog 41
Figure 33 Export Metadata to CDEF (Packet List Tab) 43
Figure 34 Export Metadata to CDEF (ISS System Subsets Tab) 44
Figure 35 Import Metadata from CDEF 45
Figure 36 Import Metadata from CDEF (ISS System Subsets Tab) 46
Figure 37 Messages Dialog 50
Figure 38 Configure Messages Dialog 51
Figure 39 Clear Messages Dialog 51

1 Welcome

The Telescience Resource Kit (TReK) is a suite of software applications and libraries that can be used to monitor and control assets in space or on the ground.

The TReK Metadata application provides the capability to create and manage telemetry and command metadata. This includes support for creating databases and metadata files and translating between different types of metadata formats.

The topics in this user guide require an understanding of the topics covered in the TReK Concepts document. Please be sure you have read the TReK Concepts document before reading this user guide.

1.1 Getting Started

Start with the Introduction which provides an application overview. Next, try the Quick Start Guides for "How Tos" for common functions. For help with details, reference the Details section. See the FAQ and Troubleshooting section for helpful hints and solutions to the common "gotchas".

2 Technical Support

If you are having trouble installing the TReK software or using any of the TReK software, please contact us for technical assistance:

TReK Help Desk E-Mail, Phone & Fax:

E-Mail: trek.help@nasa.gov
Telephone: 256-544-3521 (8:00 a.m. - 4:00 p.m. Central Time)
Fax: 256-544-9353

If you call the TReK Help Desk and you get a recording please leave a message and someone will return your call. E-mail is the preferred contact method for help. The e-mail message is automatically forwarded to the TReK developers and helps cut the response time. The HOSC Help Desk (256-544-5066) can provide assistance as needed and is available 24x7.

3 Introduction

The TReK Metadata application provides the capability to create and manage telemetry and command metadata. This includes support for creating databases and metadata files and translating between different types of metadata formats. For a more detailed introduction to TReK metadata please reference the TReK Concepts document.

Packets are most often the data that are sent from one system to another system. Commands and telemetry are just packets. Command data is packets that tell another system to do something. Telemetry data is packets that supply information about the system sending the packet.

Parameters are the individual data values that contain information about the state of the sending system or actions to be taken by the receiving system. Parameters have a value and are either placed in the outgoing data or pulled from the incoming data. TReK uses the terms "build" to describe placing parameters in a packet and "extract" to describe pulling parameters from a packet.

Parameters are grouped with related parameters into collections named parameter collections. Parameter collections are the basic building blocks of packets which are the data sent from one system to another. Figure 3 has four views of the same packet. The first row shows a packet as a single entity that could be sent between systems. The second row shows that the packet is composed of parameter collections and another packet. The third row shows that eventually a packet will break down into a series of parameter collections. The final row shows that all parameter collections are a series of parameters. Each row is a different view of the same data.

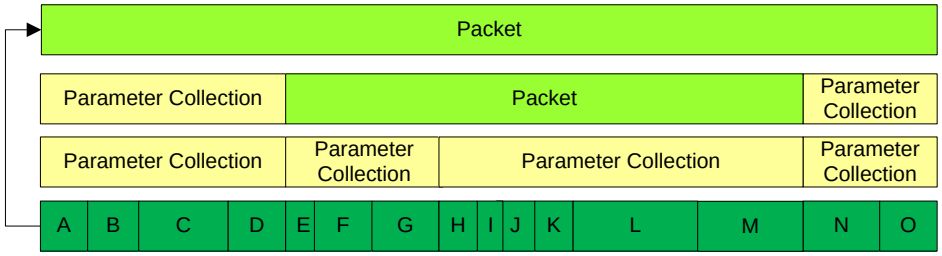

Figure 1 Packets, Parameter Collections, and Parameters

As mentioned earlier, Packets are the data that travel between systems. The packet is the largest aggregation of data in TReK.

Packets are divided into three zones: header, data, and trailer. One or more zones must be defined in a packet for it to be considered valid. Figure 2 shows each of the three zones and their relative locations. Each zone of a packet contains either another packet or a parameter collection.

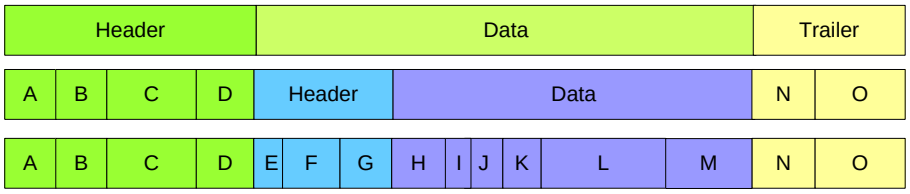

Figure 2 A Packet and Its Zones

The first line in the figure above shows a packet that has all three zones defined. The second level shows that the header and trailer zone are composed of parameter collections which contain one or more parameters. The data zone is composed of another packet which only has the header and data zones defined. The third line shows that the packet in the data zone of the top level packet is composed of two parameter collections and that all of the data in a packet will eventually break down into a series of parameters.

The Metadata application provides the capability to create and modify a telemetry packet or a command packet. The application provides all the support needed to define the details for a Packet, Parameter Collection, and Parameters like those shown in Figure 1 and Figure 2. Once defined these packet definitions can then be used with various TReK applications to monitor and control an asset in space or on the ground.

Since there are many different ways to format metadata, TReK supports a variety of formats as described below.

C Programming Language Header File
Many software developers represent a packet definition using a C structure defined in a C header file. TReK can import the packet definition by reading the C structure defined in a C header file. See Section 6.19 for details.

Common Data Exchange Format (CDEF)
CDEF is an International Space Station (ISS) program format that contains metadata described using XML. It is used to exchange telemetry and command information between data systems. See Section 6.18 for details.

TReK Database
A TReK database is a SQLite database that can contain metadata information for one or more packets.

TReK Metadata File
This is a TReK generated file that contains metadata described using XML (e.g., a Packet definition, a Parameter Collection definition).

The Metadata application can read, write, and translate between these different types of formats.

4 Overview of the User Interface

4.1 Main Window

The main window contains several areas as shown in Figure 3. The Toolbar contains multiple buttons that reconfigure the main window to focus on a specific metadata element such as a Parameter Collection, a Database, a Limit Alarm, a Packet, etc.

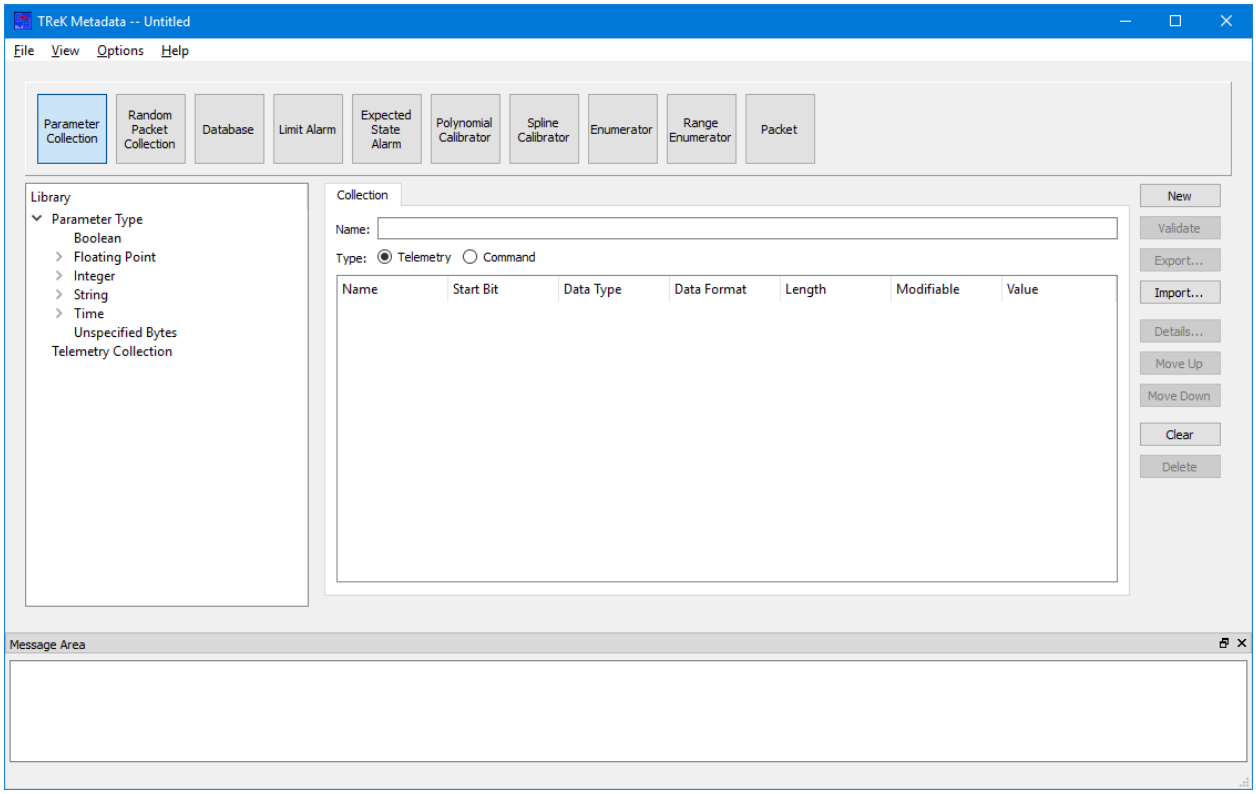

Figure 3 Main Window

Toolbar

The toolbar at the top of the window provides quick access to configure the main window to work on a specific metadata element.

Library Area

The Library Area is context sensitive and will be reconfigured to support the specific area you are working on. For example, if you are creating a Parameter Collection, the Library Area will be configured to show a list of all the types of parameters that you can add to a parameter collection. If you are working on a Database, the Library will be configured to show you a list of databases you can work with.

Work Area

The Work Area is context sensitive and will be reconfigured to support the specific type of element you are working on. If you are working on a Parameter Collection, the Work Area will display the Parameter Collection content. If you are working on a Database, the Work Area will display the list of packets in the Database you have selected. If you

are working on a Limit Alarm, the Work Area will display the content of the Limit Alarm. If you are working on a Packet, the Work Area will display the contents of the packet. The Work Area provides the capability to create or modify the contents of a specific element.

Button Area
The Button Area is context sensitive and will be reconfigured to display functions available for the specific area you are working on. For example, for a Parameter Collection, you would have functions such as creating a New Parameter Collection, Validating a Parameter Collection, Exporting the Parameter Collection (saving it in a file in a specified format), and Importing a Parameter Collection (reading a specified format from a file). There are also functions available to modify the order of parameters in the collection by moving them up and down in the list and deleting a parameter.

Message Area
The Message Area displays important status and error messages. The message area can be cleared using the View menu. The Message Area is a dock window that you can float or dock. To float a dock window, use your left mouse button to click and hold the title area while dragging the window to another area of the screen. To dock, use the title bar to drag the dock window over the main window and drop.

4.2 Toolbar

The Toolbar contains multiple buttons that reconfigure the main window to focus on a specific metadata element such as a Parameter Collection or a Database.

4.3 Menus

The Metadata application menus are: File, View, Options, and Help. Each of these menus is described in more detail below.

File Menu
The File menu provides the capability to manage configurations and exit the application.

View Menu
The View menu provides the capability to clear the main window message area and show and hide different areas in the main window.

Options Menu
The Options menu provides access to the Messages dialog and other special functions such as working with HOSC Partial Database Download files or Common Data Exchange Format files.

Help Menu
The Help menu provides access to on-line help and application version information.

5 Quick Start Guides

This section provides “How Tos” for common functions.

5.1 How to Create a Parameter Collection

The following steps describe the minimum necessary to create a Parameter Collection. For additional information and details please reference section 6.1.

1. Push the Collection button to configure the Work Area to create a Parameter Collection.
2. Enter a name for the Parameter Collection.
3. Select the Telemetry or Command radio button to specify whether the Parameter Collection will be a telemetry collection or a command collection.
4. Add one or more parameters to the Parameter Collection by dragging a Parameter Type from the Library Area and dropping it in the Parameter Collection List.
5. Enter the minimum information necessary to define a Parameter: Name, Start Bit, Data Type, and Length.
6. Validate the Parameter Collection.
7. Export the Parameter Collection to save a copy in a file.

5.2 How to Create a Database

The following steps describe the minimum necessary to create a TReK Database. For additional information and details please reference section 6.3.

1. Push the Database button to configure the Work Area to work with a TReK Database.
2. Select the Telemetry or Command radio button to specify whether the Database will be a telemetry database or a command database.
3. Push the New button and enter the name for the new Database. Once you push the OK button to close the New Database dialog, the Database will be opened and ready for editing in the Work Area.

5.3 How to View the List of Packets in a Database

The following are the minimum necessary steps to view the list of Packets in a TReK Database. For additional information and please reference section 6.3.

1. Push the Database button to configure the Work Area to work with a TReK Database.
2. Select the Telemetry or Command radio button to indicate the type of Database. The Library will reconfigure to list the selected type of databases.
3. Select a Database item in the Library and drag it to the Name field and drop it. The packets in the Database will be displayed in the list.

Note: The Library displays telemetry databases that reside in the TReK workspace telemetry_database folder and command databases that reside in the command_database folder.

5.4 How to Export a Packet from a Database

This section describes how to export a packet definition from a TReK Database. For additional information and details please reference section 6.3.

1. Push the Database button to configure the Work Area to work with a TReK Database.
2. Select the Telemetry or Command radio button to indicate the type of Database. The Library will reconfigure to list the selected type of databases.
3. Select a Database item in the Library and drag it to the Name field and drop it. The packets in the Database will be displayed in the list.
4. Select a Packet in the list and push the Export button.
5. In the Export dialog, select a Format, enter the required information and push OK to export the packet definition.

5.5 How to Add a Packet to a Database

This section describes how to add a packet definition to a TReK Database using the TReK Database ASCII file format. For additional information and details please reference section 6.3.

1. Push the Database button to configure the Work Area to work with a TReK Database.
2. Select the Telemetry or Command radio button to indicate the type of Database. The Library will reconfigure to list the selected type of databases.
3. Select a Database item in the Library and drag it to the Name field and drop it. Any packets in the Database will be displayed in the list.
4. Push the Import button to add a packet to the database.
5. In the Import dialog select the “Packet Definition in ASCII File Format” option and browse to select the file(s) describing the packet and then push the Import button. If the packet was successfully imported it will appear in the list or an error message will be displayed.

5.6 How to Create a Limit Alarm

The following steps describe the minimum necessary to create a Limit Alarm. For additional information and details please reference section 6.4.

1. Push the Limit Alarm button to configure the Work Area to create a Limit Alarm.
2. Enter a name for the Limit Alarm.
3. Enter a type for the Limit Alarm.
4. Enter the information for one or more levels in the Limit Alarm.
5. Validate the Limit Alarm.
6. Export the Limit Alarm to save a copy in a file.

5.7 How to Create a Packet

The following steps describe the minimum necessary to create a Packet. For additional information and details please reference section 6.10.

1. Push the Packet button to configure the Work Area to create a Packet.
2. Enter a name for the Packet.
3. Select the Telemetry or Command radio button to specify whether the Packet will be a telemetry packet or a command packet.
4. Drag and drop items from the library into the Packet definition area to construct the packet. For example, start the packet definition by using a predefined packet template. Then add or update the packet zones to create a custom packet. [Hint: You can drag item such as collections to the individual drop zones such as the Header drop zone, Data drop zone, or Trailer drop zone, or you can right click on a Header, Data, or Trailer in the packet definition to get a pop-up menu to Add, Replace, or Delete the contents of a zone.
5. Validate the Packet.
6. Export the Packet to save a copy in a file.

5.8 How to Import Data from HOSC Partial Database Download Files Into a TReK Database

The following steps describe the minimum necessary to import data from HOSC Partial Database Download files into a TReK database. The data must be imported into a new TReK database. For additional information and details please reference section 6.10.

1. Go to the Options menu and select 'Convert EHS Database Files to TReK Database'.
2. In the Convert EHS Database Files to TReK Database dialog select Telemetry or Command based on the type of data to import.
3. In the TReK Database Information area, enter the name and directory for the new TReK database that will be created to hold the information from the EHS database files.
4. In the EHS Partial Database File Information, enter the information that identifies where the EHS partial Database files can be found.

   Note: Filenames for EHS Partial Database files with telemetry information contain a prefix number. This number should be entered in the Prefix field.

5. Push the Convert button to execute the import. Messages related to the convert process will be displayed in the Results area. The process can take a while if the EHS partial database files contain a large amount of data. You can continue to work on other tasks in the application while the convert is in work.

6 Details

This section covers various application details.

6.1 Parameter Collection

The Parameter Collection configuration provides the capability to work with a Parameter Collection. Figure 4 shows the Main Window with the Parameter Collection configuration.

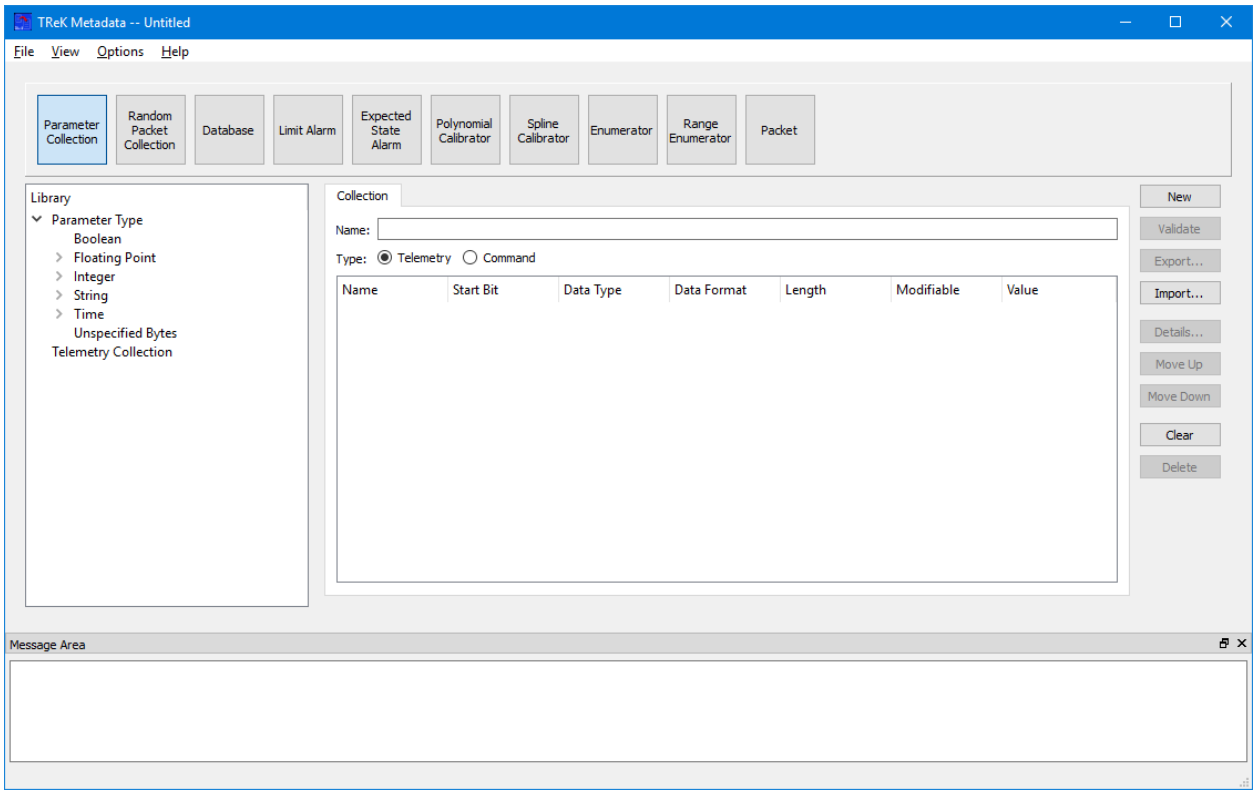

Figure 4 Main Window in Collection Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to items you can use to create a Parameter Collection. The Parameter Type list provides a list of all the parameter types that can be added to a Parameter Collection. The Telemetry Collection or Command Collection list is configured to provide access to existing Parameter Collections stored in the trek_workspace. When you create a Parameter Collection and export it to the trek_workspace, it will automatically be added to the Parameter Collection list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Collection Tab. This is where you will define the details of your Parameter Collection. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Parameter

Collection you are working on (e.g., Validate to validate the Parameter Collection or Export to export the Parameter Collection).

The Work Area provides the capability to define the following properties for the Parameter Collection:

Name
The name field is used to enter the name of the Parameter Collection.

Type
The type selection is used to identify the type of Parameter Collection: Telemetry or Command.

Parameter List
The Parameter List is used to identify the parameters in the Parameter Collection.

The Metadata application uses Drag and Drop to move items from the Library Area into the Work Area.

To add a parameter, use your left mouse button to select a parameter from the Library Parameter Type list, and while holding the left mouse button down, drag the item to the Work Area and drop it as shown by the green arrows in Figure 5.

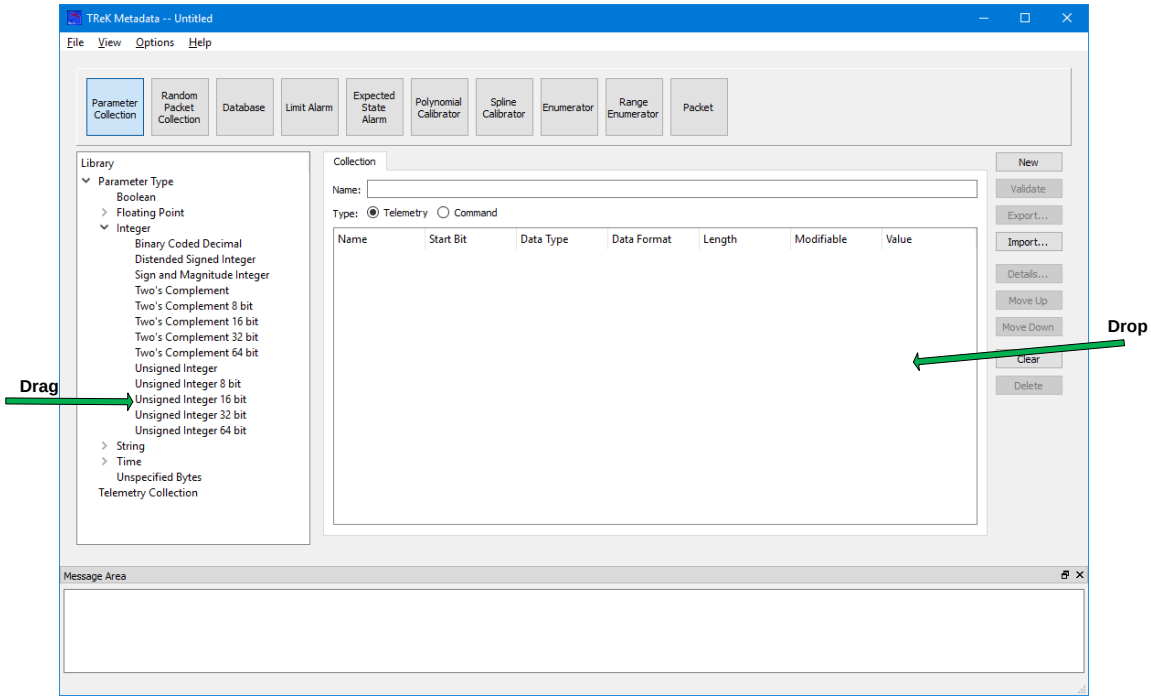

Figure 5 Parameter Collection Drag and Drop

After dropping the parameter, you should see the parameter in the list as shown in Figure 6.

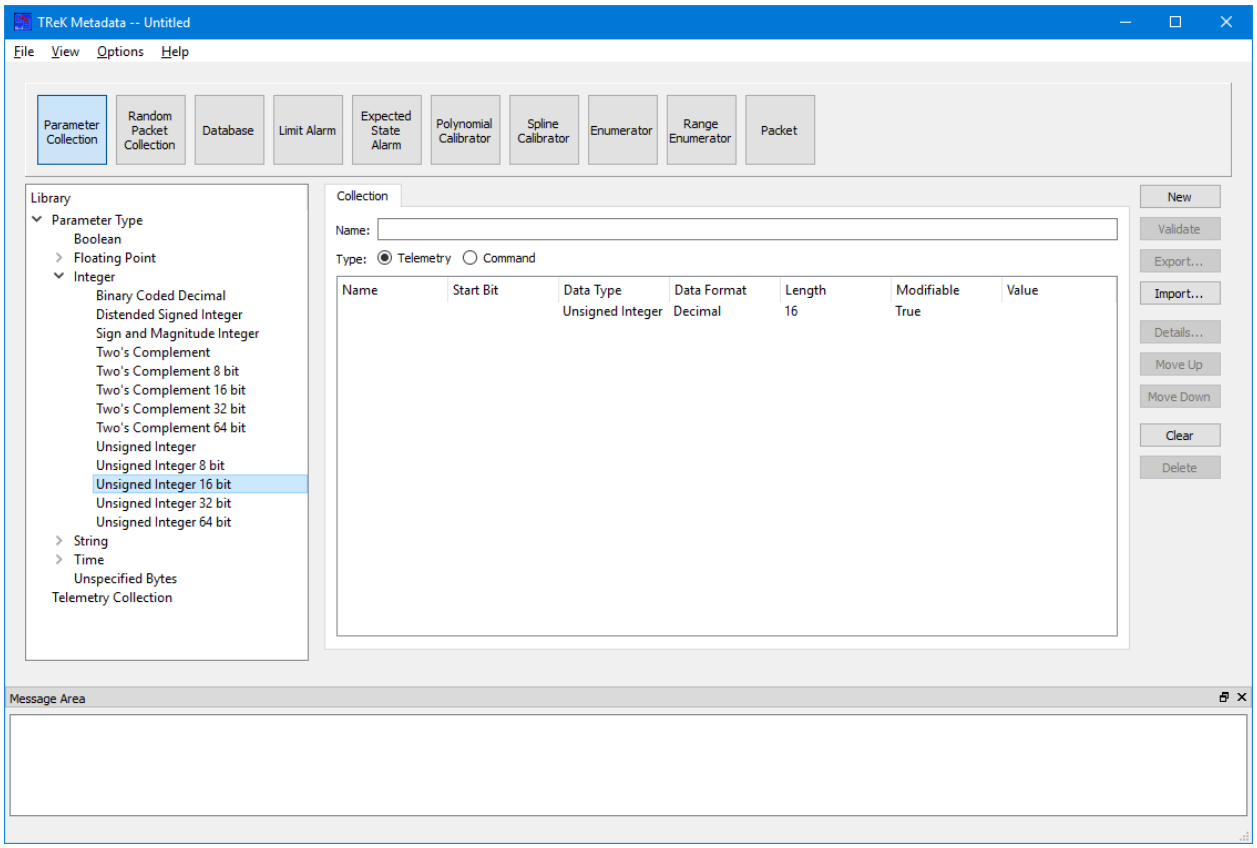

Figure 6 Collection with Parameter

To modify an existing Parameter Collection, you can drag the collection from the Telemetry Collection list or the Command Collection list and drop it in the Name field. This will load the Parameter Collection information into the Work Area.

Collection Drag and Drop Tips and Tricks

- If you drop an item from the Parameter Type list anywhere on the list it will be placed at the end of the list.
- If you drop an item from the Parameter Type list on top of another item, the item will be placed below that item in the list.
- If you drag an item from the Parameter Type list using the right mouse button, a dialog will pop up asking how many of that item you would like to add to the list. [This is only available on Windows].
- If you drag an item from a Telemetry Collection list or Command Collection list and drop it in the Name field, the Parameter Collection information will be loaded into the Work Area.

The following functions are available when working with a Parameter Collection:

New
New provides the capability to clear the existing Parameter Collection information.

Validate
Validate provides the capability to validate the Parameter Collection information.

Export
Export provides the capability to save the Parameter Collection to a file.

Import
Import provides the capability to load a Parameter Collection from a file. When a Parameter Collection is imported the collection information is displayed on the Collection tab. To store any modifications back to the file, the Parameter Collection must be exported.

Details
Details provides the capability to view and modify the properties associated with a selected parameter.

Move Up
Move Up provides the capability to move a selected parameter up in the list.

Move Down
Move Down provides the capability to move a selected parameter down in the list.

Clear
Clear provides the capability to clear the Work Area.

Delete
Delete provides the capability to delete a selected parameter.

6.2 Random Packet Collection

The Random Packet Collection configuration provides the capability to work with a Random Packet Collection. Figure 7 shows the Main Window with the Random Packet Collection configuration. Random Packet Collections can be used to process data that is random such as ISS health and status.

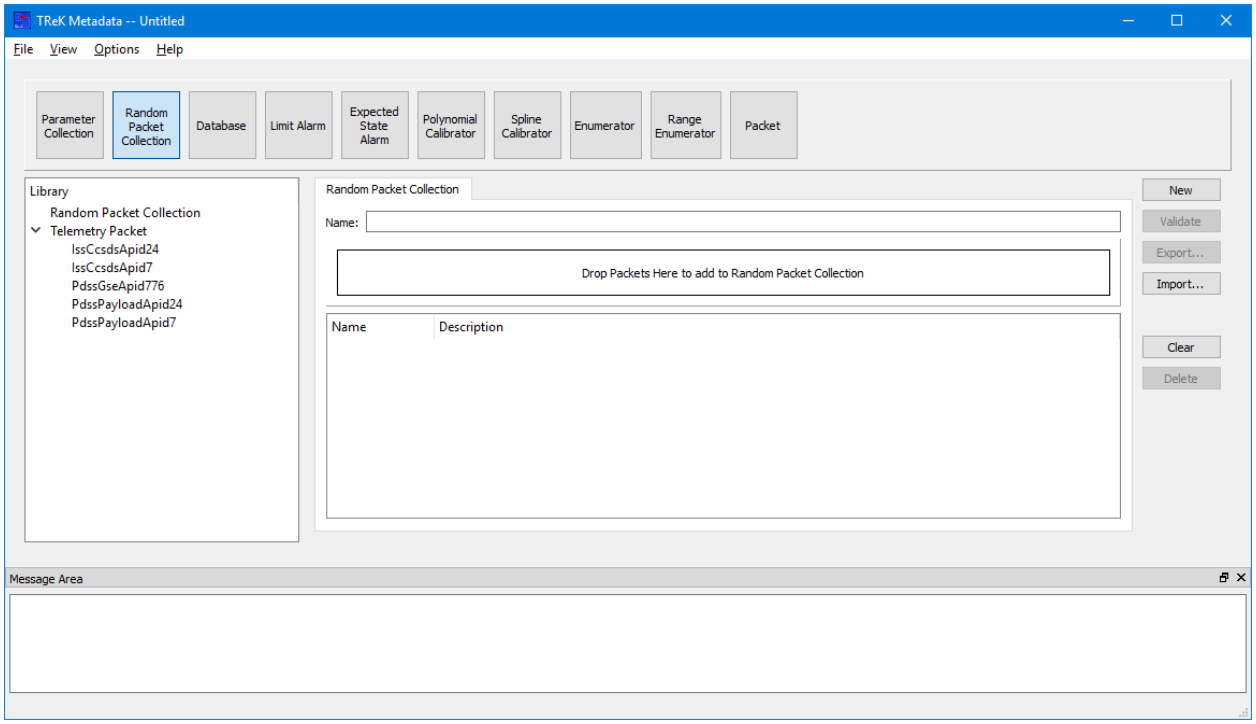

Figure 7 Random Packet Collection

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to items you can use to create a Random Packet Collection. The Telemetry Packet list provides a list of telemetry packets that can be used to select a packet to add to a Random Packet Collection. The Random Packet Collection list is configured to provide access to existing Random Packet Collections stored in the trek_workspace. When you create a Random Packet Collection and export it to the trek_workspace, it will automatically be added to the Random Packet Collection list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Random Packet Collection Tab. This is where you will define the details of your Random Packet Collection. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Random Packet Collection you are working on (e.g., Validate to validate the Random Packet Collection, Export to export the Random Packet Collection, etc.).

The Work Area provides the capability to define the following properties for the Random Packet Collection:

<u>Name</u>
The name field is used to enter the name of the Random Packet Collection.

Packet Drop Zone

The Packet Drop Zone is used to add packets to the Random Packet Collection.

The Metadata application uses Drag and Drop to move items from the Library Area into the Work Area.

To add a packet to the Random Packet Collection, use your left mouse button to select a packet from the Library Telemetry Packet list, and while holding the left mouse button down, drag the item to the Packet Drop Zone and drop it. The packet will be added to the Random Packet Collection as shown in Figure 8.

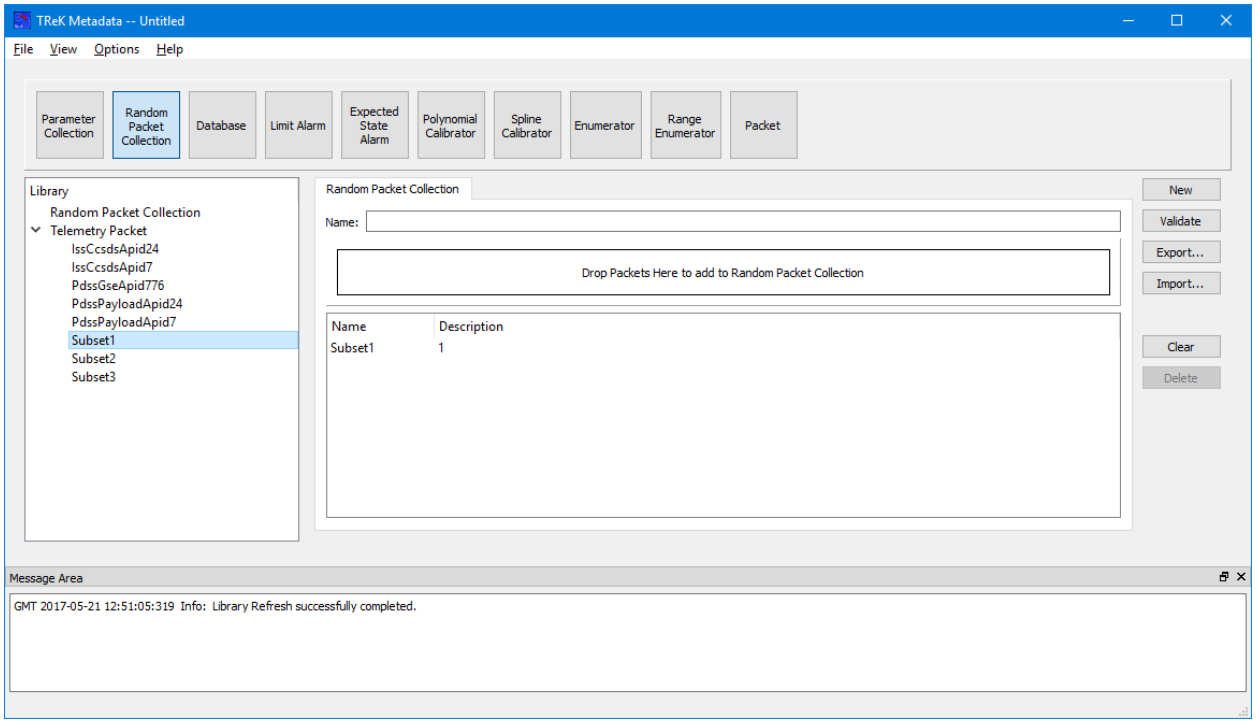

Figure 8 Random Packet Collection with Packet

Random Packet Collections require that all packets within it are of the same type. The first packet placed in the Random Packet Collection sets the type for the packet. If you attempt to place a packet with a different type in the Random Packet Collection, it will be rejected with a friendly message.

The following functions are available when working with a Random Packet Collection:

New

New provides the capability to clear the existing Random Packet Collection information.

Validate

Validate provides the capability to validate the Random Packet Collection information.

Export
Export provides the capability to save the Random Packet Collection to a file.

Import
Import provides the capability to load a Random Packet Collection from a file. When a Random Packet Collection is imported the collection information is displayed on the Random Packet Collection tab. To store any modifications back to the file, the Random Packet Collection must be exported.

Clear
Clear provides the capability to clear the Work Area.

Delete
Delete provides the capability to delete a selected packet in the Random Packet Collection.

6.3 Database

The Database configuration provides the capability to work with a TReK Database. Figure 9 shows the Main Window in the Database configuration. The main capabilities provided include creating a new database, viewing information about packets, collections, and types in the database, adding a packet to the database, deleting a packet from the database, and validating a database. To work with database tables directly, a SQLite tool can be used to open and edit the database contents. There are many free and shareware tools available.

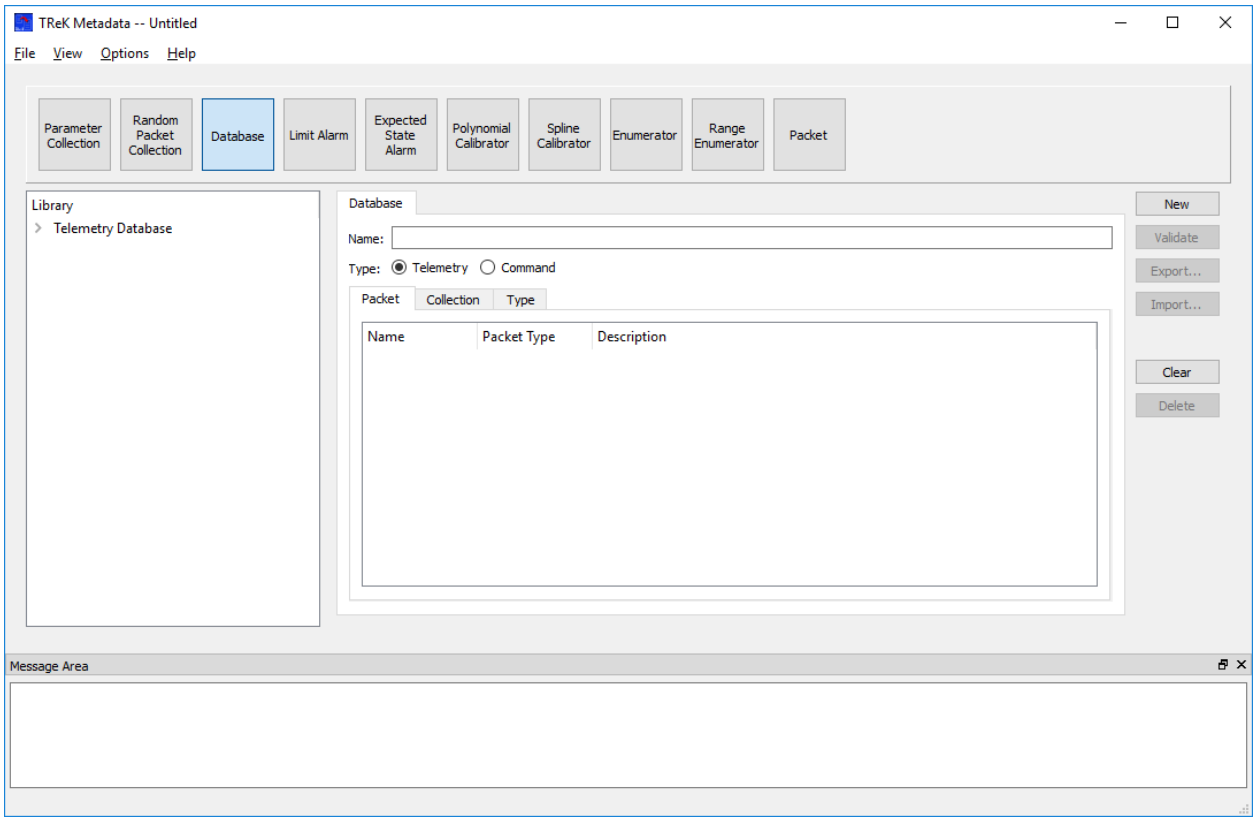

Figure 9 Database Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to show a list of Telemetry Databases or Command Databases that reside in the trek_workspace. When you create a database and export it to the trek_workspace, it will automatically be added to the applicable database list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Database Tab. This is where you will see information stored in the Database. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Database you are working with (e.g., Validate to validate the Database or Export to export a packet from the database).

The Work Area provides the following information:

<u>Name</u>
The name field is used to display the name of the Database.

<u>Type</u>
The type selection is used to identify the type of Database: Telemetry or Command.

Packet Tab
The Packet Tab shows the packets in a Telemetry Database and the Commands in a Command Database.

Collection Tab
The Collection Tab shows the collections in a Telemetry Database and the collections in a Command Database.

Types Tab
The Types Tab shows the types in a Telemetry Database and the types in a Command Database.

The Metadata application uses Drag and Drop to move items from the Library Area into the Work Area.

To open a database, use your left mouse button to select a database from the Library, and while holding the left mouse button down, drag the item to the Name field and drop it. The database will be opened and the list of packets in the database will be displayed in the list as shown in Figure 10.

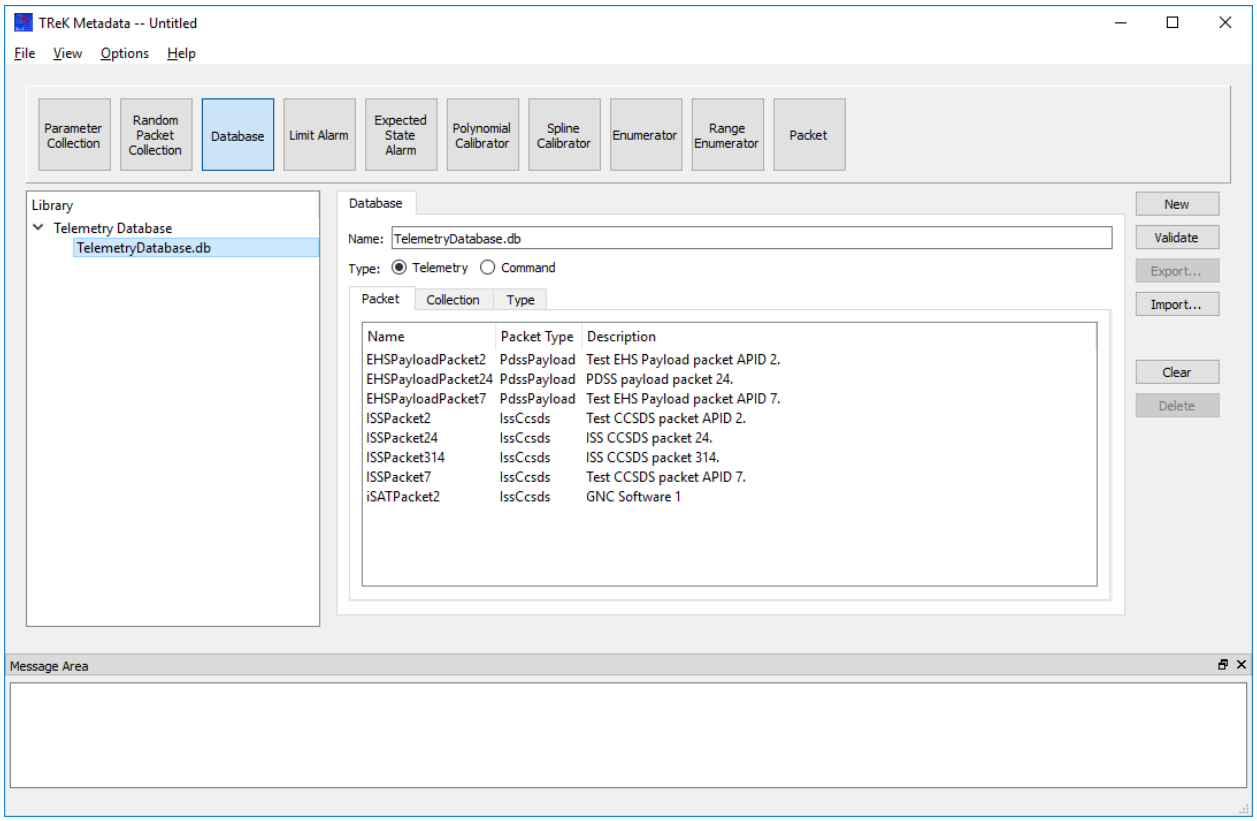

Figure 10 Database Packets Shown In List

The following functions are available when working with a Database:

New
New provides the capability to create a new database.

Validate
Validate provides the capability to validate a database.

Export
Export provides the capability to export a selected packet to a file.

Import
Import provides the capability to import information into a database. For a telemetry database, this includes the following types of imports: a packet definition defined in an ASCII file format as described in the TReK Telemetry Database Definition Document, a collection definition defined in an ASCII file format as described in the TReK Telemetry Database Definition Document, a type definition defined in an ASCII file format as described in the TReK Telemetry Database Definition Document, and a HOSC Ground Support Equipment (GSE) packet definition as described in the Payload to Generic User Interface Definition Document (PGUIDD).

Important Note about Parameter Names after Importing a GSE Packet Definition
When a GSE packet definition is imported the parameter names are changed. In order to keep parameter names unique, characters representing type of processing are added to the name. For example, suppose the name of the parameter is UGZG20RT2004J. If the parameter is unprocessed, the parameter will be named UGZG20RT2004J=UN where =UN was added to indicate that it is an unprocessed parameter. A converted value will have =CO added to the name, and a calibrated value will have =CA added to the name. Additional characters are also added to indicate Overall Status (=OStatus), Number of Samples (=Samples), and Sample Status Values (=Status). The following shows an example for a GSE parameter named UGZG20RT2004J:

UGZG20RT2004J	Original Parameter name in GSE Packet Definition
UGZG20RT2004J=UN	Unprocessed Value Name
UGZG20RT2004J=UNOStatus	Unprocessed Overall Status Name
UGZG20RT2004J=UNSamples	Unprocessed Number of Samples Name
UGZG20RT2004J= UNStatus	Unprocessed Sample Status Values Name

For a command database, this includes the following types of imports: a packet definition defined in an ASCII file format as described in the TReK Command Database Definition Document, a collection definition defined in an ASCII file format as described in the TReK Command Database Definition Document, and a type definition defined in an ASCII file format as described in the TReK Command Database Definition Document

Clear
Clear provides the capability to clear the Work Area.

Delete
Delete provides the capability to delete a selected packet from the database.

6.4 Limit Alarm

The Limit Alarm configuration provides the capability to work with a Limit Alarm. Figure 11 shows the Main Window in the Limit Alarm configuration.

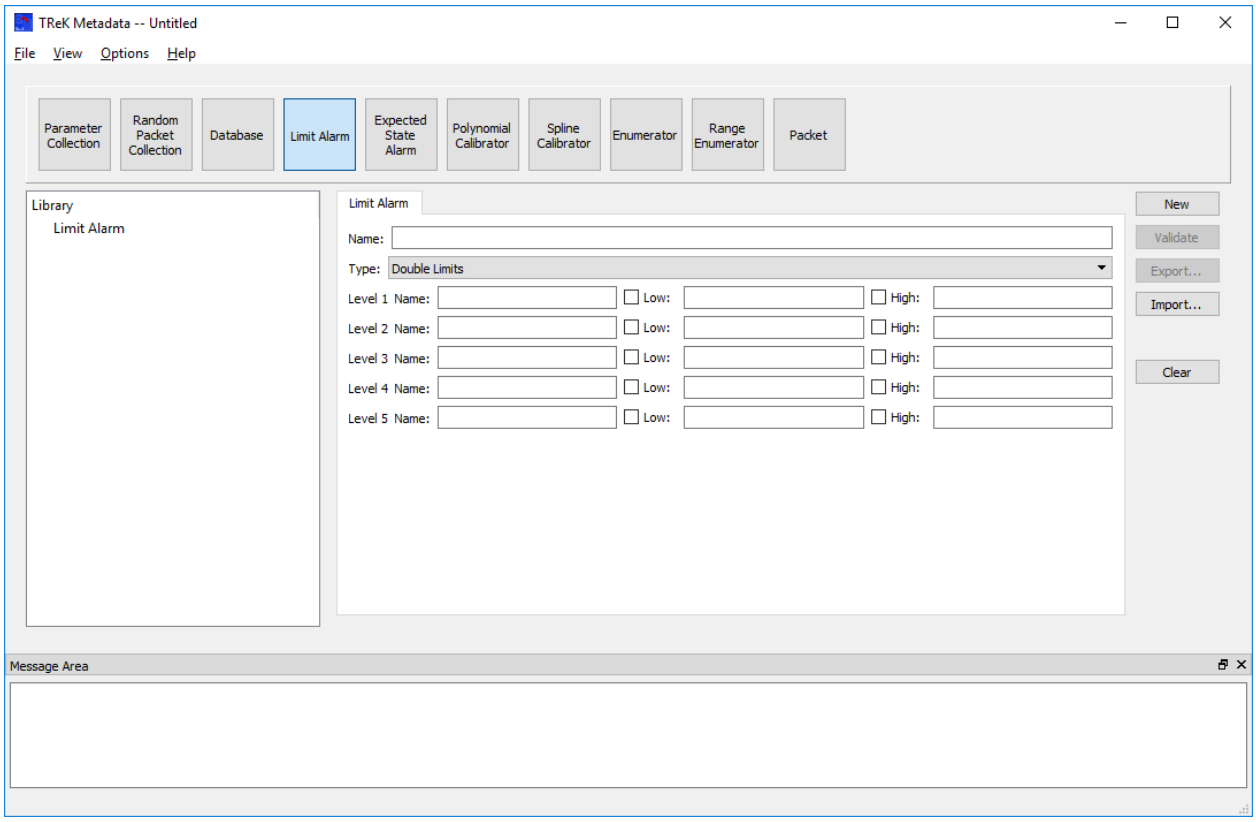

Figure 11 Limit Alarm Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Limit Alarms in the trek_workspace. When you create a Limit Alarm and export it to the trek_workspace, it will be added to the Limit Alarm list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Limit Alarm Tab. This is where you will define the details of the Limit Alarm. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Limit Alarm you are working on (e.g., Validate to validate the Limit Alarm or Export to export the Limit Alarm).

The Work Area provides the following information:

Name
The name field is used to display the name of the Limit Alarm.

Type
The type menu is used to select the type of Limit Alarm.

Limit Alarm Levels
Fields are provided to set details for each Limit Alarm Level as shown in Figure 12. The Low checkbox is used to indicate if the Low for that level is in use. The High checkbox is used to indicate if the High for that level is in use.

Figure 12 Limit Alarm Level

The following functions are available when working with a Limit Alarm:

New
New provides the capability to clear the existing Limit Alarm information.

Validate
Validate provides the capability to validate the Limit Alarm information.

Export
Export provides the capability to save the Limit Alarm to a file.

Import
Import provides the capability to load a Limit Alarm from a file. When a Limit Alarm is imported the information is displayed on the Limit Alarm tab. To store any modifications back to the file, the Limit Alarm must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.5 Expected State Alarm

The Expected State Alarm configuration provides the capability to work with an Expected State Alarm. Figure 13 shows the Main Window in the Expected State Alarm configuration.

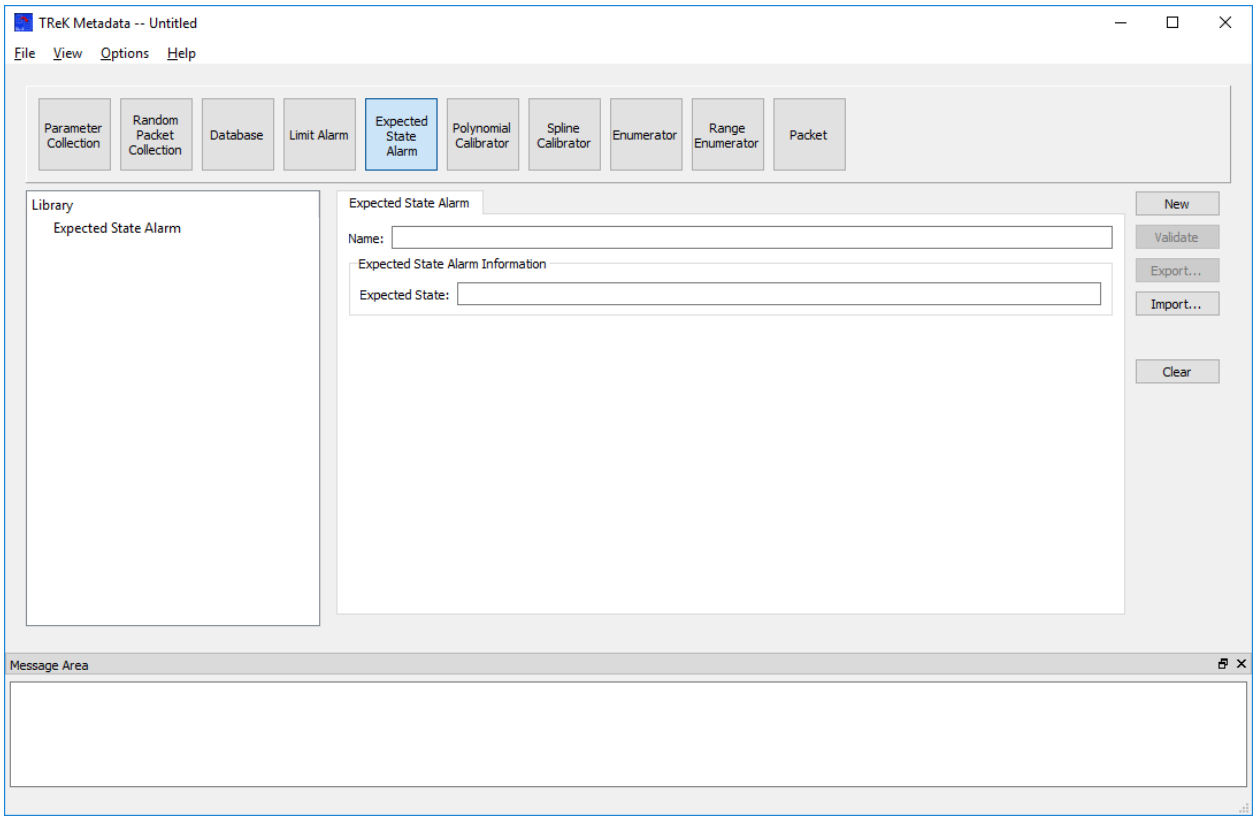

Figure 13 Expected State Alarm Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Expected State Alarms in the trek_workspace. When you create an Expected State Alarm and export it to the trek_workspace, it will be added to the Expected State Alarm list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Expected State Alarm Tab. This is where you will define the details of the Expected State Alarm. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Expected State Alarm you are working on (e.g., Validate to validate the Expected State Alarm or Export to export the Expected State Alarm).

The Work Area provides the following information:

<u>Name</u>
The name field is used to display the name of the Expected State Alarm.

<u>Expected State</u>
The expected state field is used to enter the expected state for the Expected State Alarm.

The following functions are available when working with an Expected State Alarm:

New
New provides the capability to clear the existing Expected State Alarm information.

Validate
Validate provides the capability to validate the Expected State Alarm information.

Export
Export provides the capability to save the Expected State Alarm to a file.

Import
Import provides the capability to load an Expected State Alarm from a file. When an Expected State Alarm is imported the information is displayed on the Expected State Alarm tab. To store any modifications back to the file, the Expected State Alarm must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.6 Polynomial Calibrator

The Polynomial Calibrator configuration provides the capability to work with a Polynomial Calibrator. Figure 14 shows the Main Window in the Polynomial Calibrator configuration.

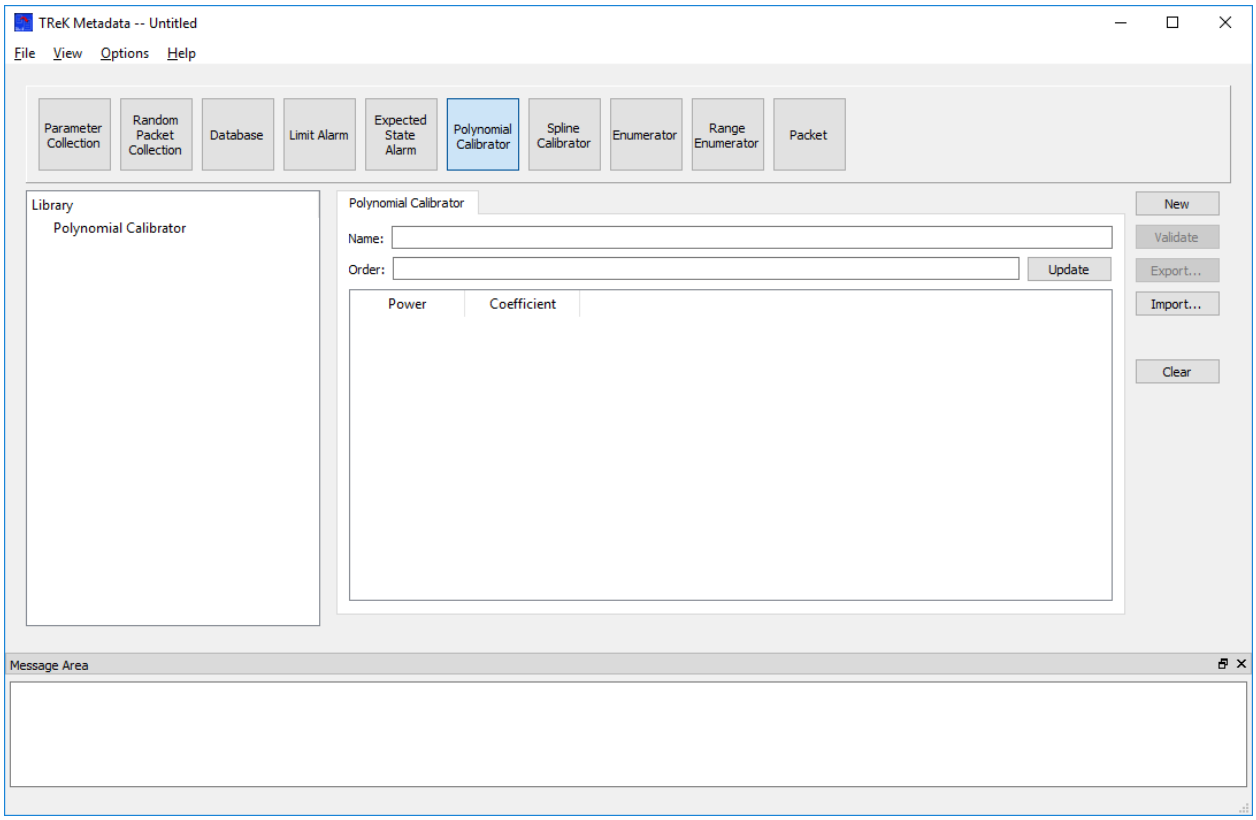

Figure 14 Polynomial Calibrator Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Polynomial Calibrators in the trek_workspace. When you create a Polynomial Calibrator and export it to the trek_workspace, it will be added to the Polynomial Calibrator list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Polynomial Calibrator Tab. This is where you will define the details of the Polynomial Calibrator. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Polynomial Calibrator you are working on (e.g., Validate to validate the Polynomial Calibrator or Export to export the Polynomial Calibrator).

The Work Area provides the following information:

<u>Name</u>
The name field is used to display the name of the Polynomial Calibrator.

<u>Order</u>
The order field is used to set the order of the Polynomial Calibrator. The correct number of rows is added to the Power/Coefficient List when the Update button is pushed.

Power Coefficient List

The Power Coefficient List is used to enter the Power and Coefficent information. The Power Coefficient List is configured when you enter a value for the Order and push the Update button. The results are shown in Figure 15.

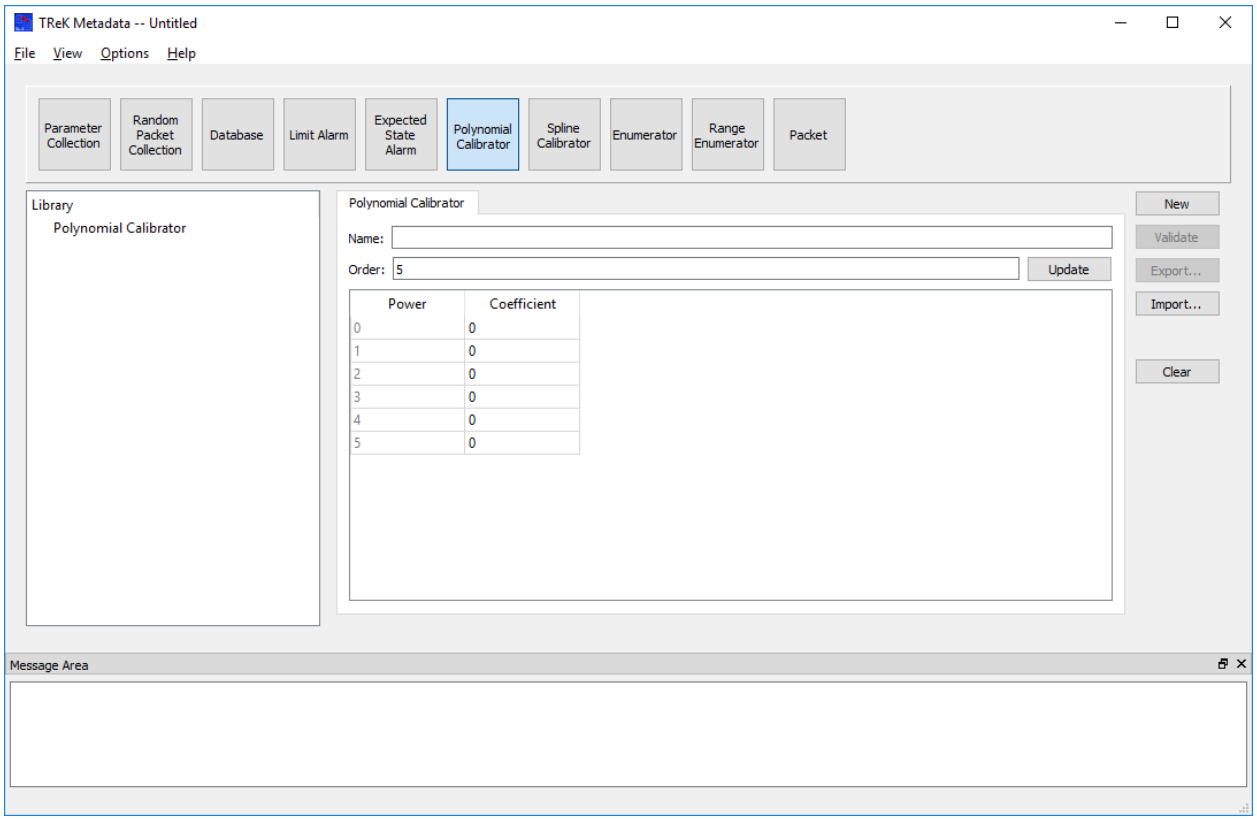

Figure 15 Power Coefficient List Populate when Order is Set to 5

The following functions are available when working with a Polynomial Calibrator:

New

New provides the capability to clear the existing Polynomial Calibrator information.

Validate

Validate provides the capability to validate the Polynomial Calibrator information.

Export

Export provides the capability to save the Polynomial Calibrator to a file.

Import

Import provides the capability to load a Polynomial Calibrator from a file. When a Polynomial Calibrator is imported the information is displayed on the Polynomial Calibrator tab. To store any modifications back to the file, the Polynomial Calibrator must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.7 Spline Calibrator

The Spline Calibrator configuration provides the capability to work with a Spline Calibrator. Figure 16 shows the Main Window in the Spline Calibrator configuration.

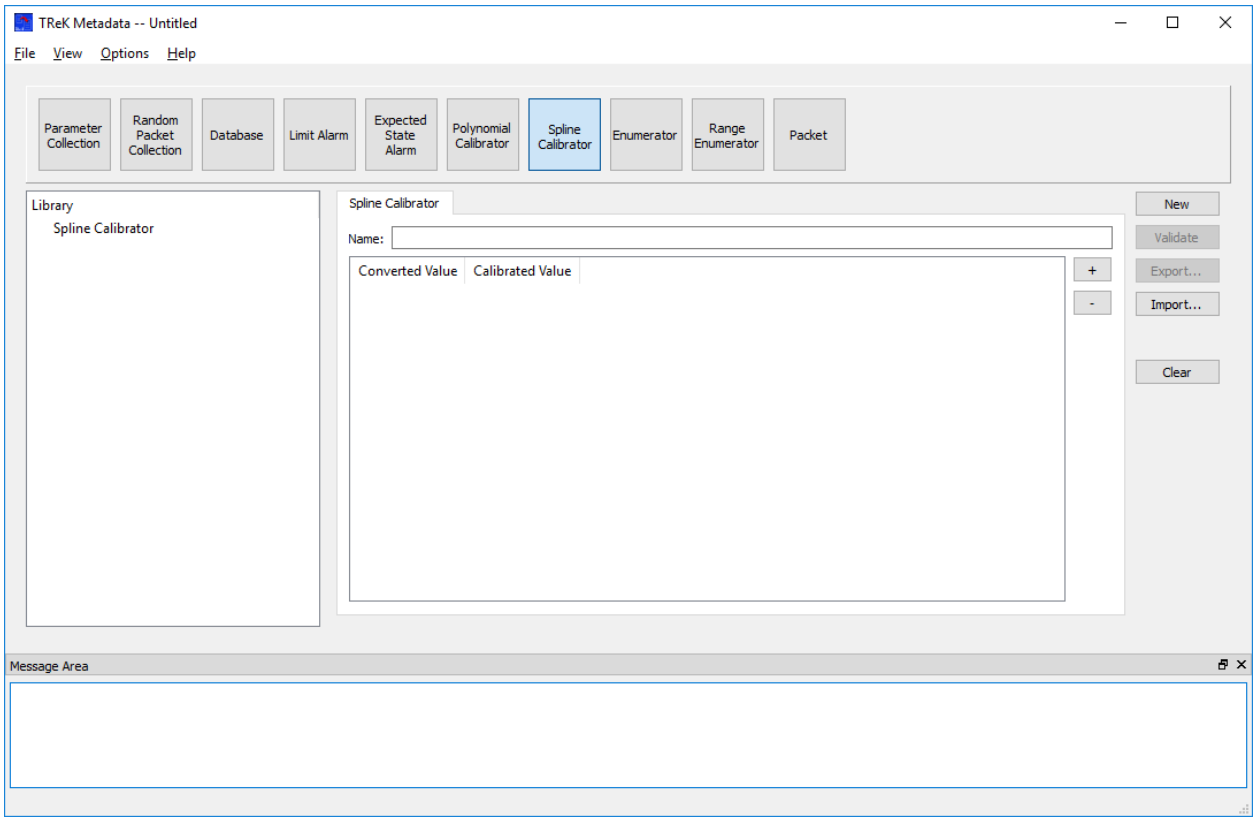

Figure 16 Spline Calibrator Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Spline Calibrators in the trek_workspace. When you create a Spline Calibrator and export it to the trek_workspace, it will be added to the Spline Calibrator list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Spline Calibrator Tab. This is where you will define the details of the Spline Calibrator. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Spline Calibrator you are working on (e.g., Validate to validate the Spline Calibrator or Export to export the Spline Calibrator).

The Work Area provides the following information:

Name
The name field is used to display the name of the Spline Calibrator.

Converted Value Calibrated Value List
The Converted Value Calibrated Value List is used to enter the Converted Value and Calibrated Value information. The + (plus) button is used to add a row to the list and the – (minus) button is used to delete a row from the list. Converted Value and Calibrated Value information can be entered directly into the list as shown in Figure 17.

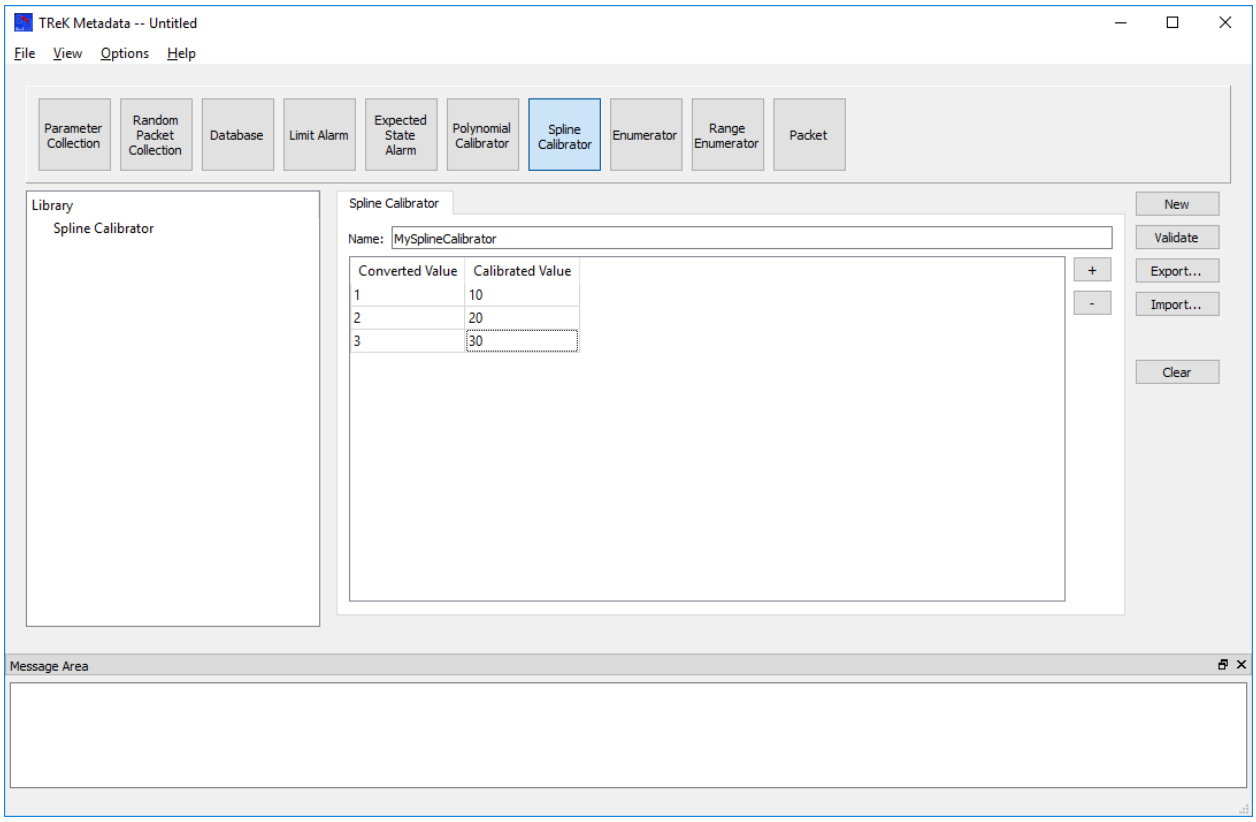

Figure 17 Spline Calibrator Information

The following functions are available when working with a Spline Calibrator:

New
New provides the capability to clear the existing Spline Calibrator information.

Validate
Validate provides the capability to validate the Spline Calibrator information.

Export
Export provides the capability to save the Spline Calibrator to a file.

Import
Import provides the capability to load a Spline Calibrator from a file. When a Spline Calibrator is imported the information is displayed on the Spline Calibrator tab. To store any modifications back to the file, the Spline Calibrator must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.8 Enumerator

The Enumerator configuration provides the capability to work with an Enumerator. Figure 18 shows the Main Window in the Enumerator configuration.

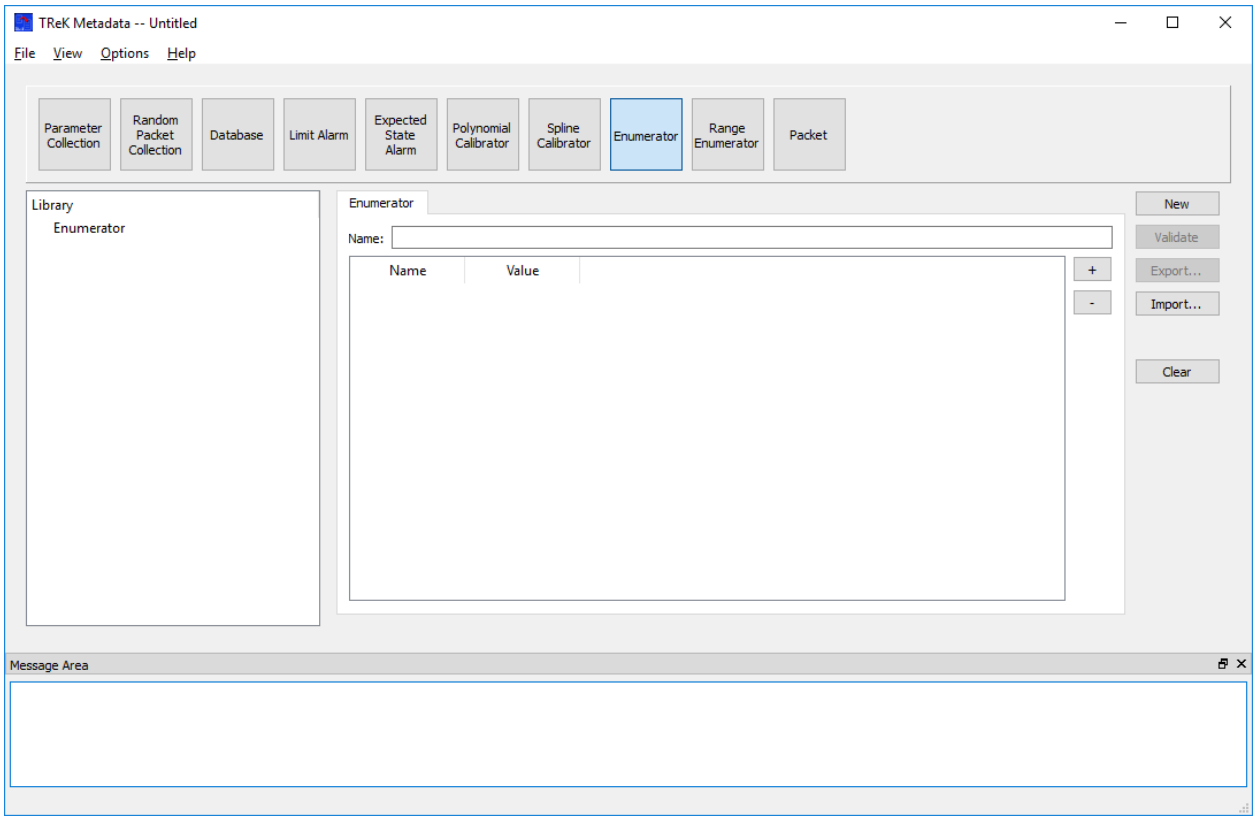

Figure 18 Enumerator Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Enumerators in the trek_workspace. When you create an Enumerator and export it to the trek_workspace, it will be added to the Enumerator list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Enumerator Tab. This is where you will define the details of the Enumerator. On the right hand side of the Main

Window you will see a set of buttons that provide various capabilities applicable to the Enumerator you are working on (e.g., Validate to validate the Enumerator or Export to export the Enumerator).

The Work Area provides the following information:

Name
The name field is used to display the name of the Enumerator.

Name Value List
The Name Value List is used to enter the Name and Value information for the Enumerator. The + (plus) button is used to add a row to the list and the – (minus) button is used to delete a row from the list. Name and Value information can be entered directly into the list as shown in Figure 19.

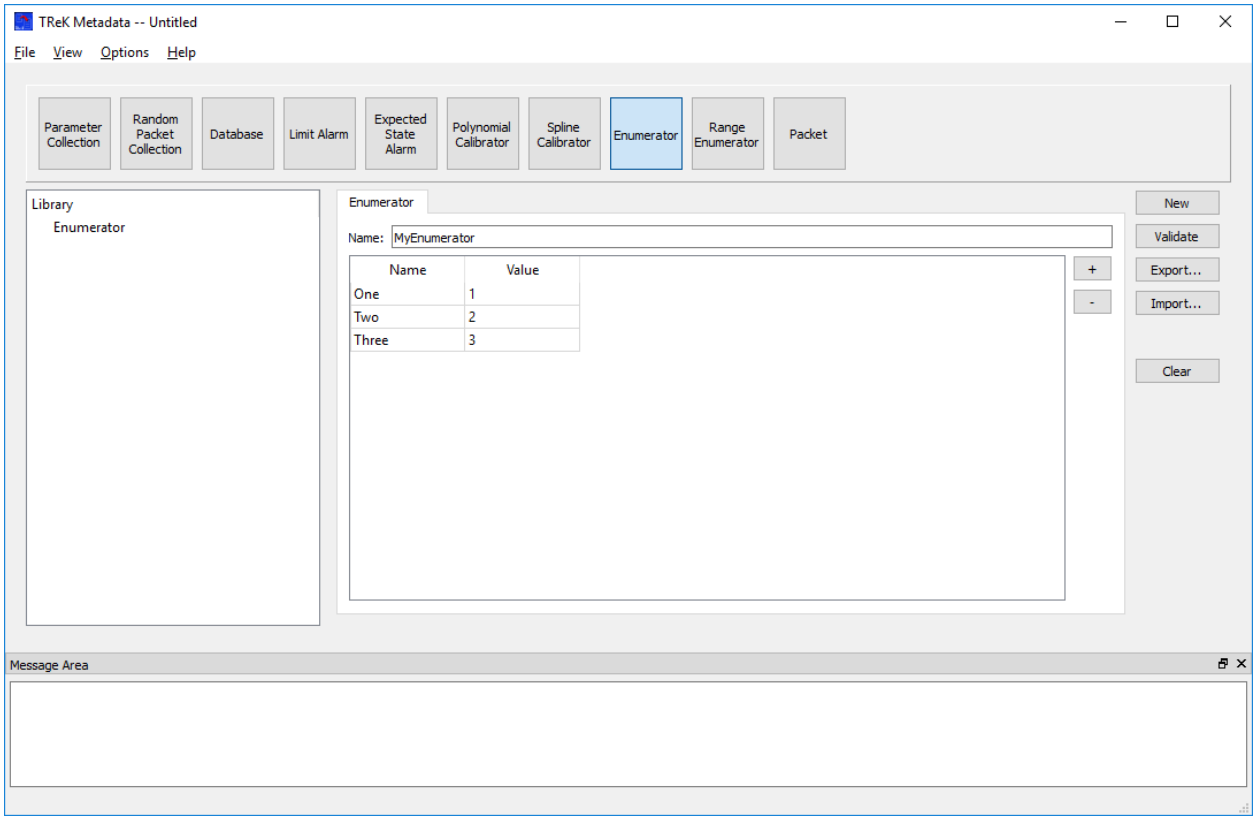

Figure 19 Populated Enumerator

The following functions are available when working with an Enumerator:

New
New provides the capability to clear the existing Enumerator information.

Validate
Validate provides the capability to validate the Enumerator information.

Export
Export provides the capability to save the Enumerator to a file.

Import
Import provides the capability to load an Enumerator from a file. When an Enumerator is imported the information is displayed on the Enumerator tab. To store any modifications back to the file, the Enumerator must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.9 Range Enumerator

The Range Enumerator configuration provides the capability to work with a Range Enumerator. Figure 20 shows the Main Window in the Range Enumerator configuration.

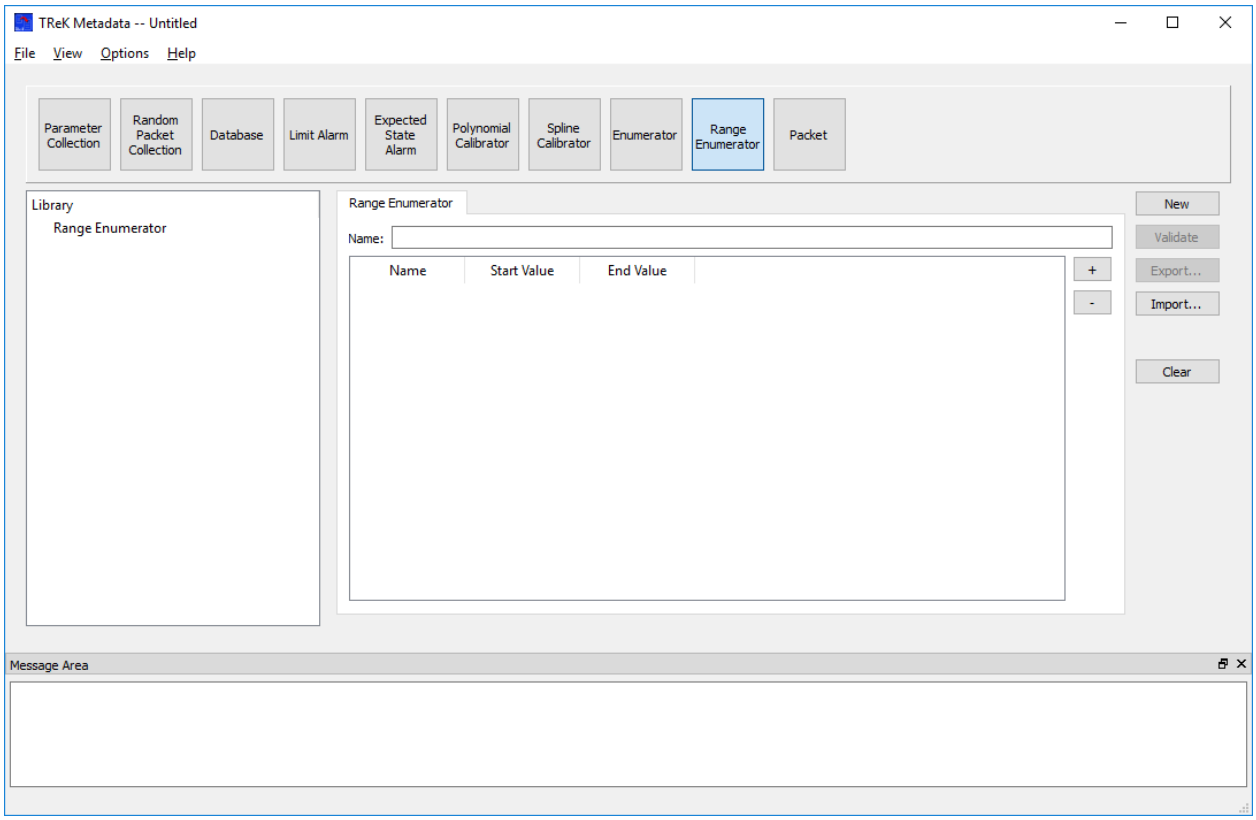

Figure 20 Range Enumerator Configuration

On the left hand side of the Main Window you will see the Library Area. It is configured to provide access to existing Range Enumerators in the trek_workspace. When you create a Range Enumerator and export it to the trek_workspace, it will be added to the

Range Enumerator list. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item. In the middle of the Main Window you will see the Work Area showing the Range Enumerator Tab. This is where you will define the details of the Range Enumerator. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Range Enumerator you are working on (e.g., Validate to validate the Range Enumerator or Export to export the Range Enumerator).

The Work Area provides the following information:

Name

The name field is used to display the name of the Enumerator.

Name Start Value End Value List

The Name Start Value End Value List is used to enter the Name, Start Value, and End Value information for the Range Enumerator. The + (plus) button is used to add a row to the list and the – (minus) button is used to delete a row from the list. Name, Start Value, and End Value information can be entered directly into the list as shown in Figure 21.

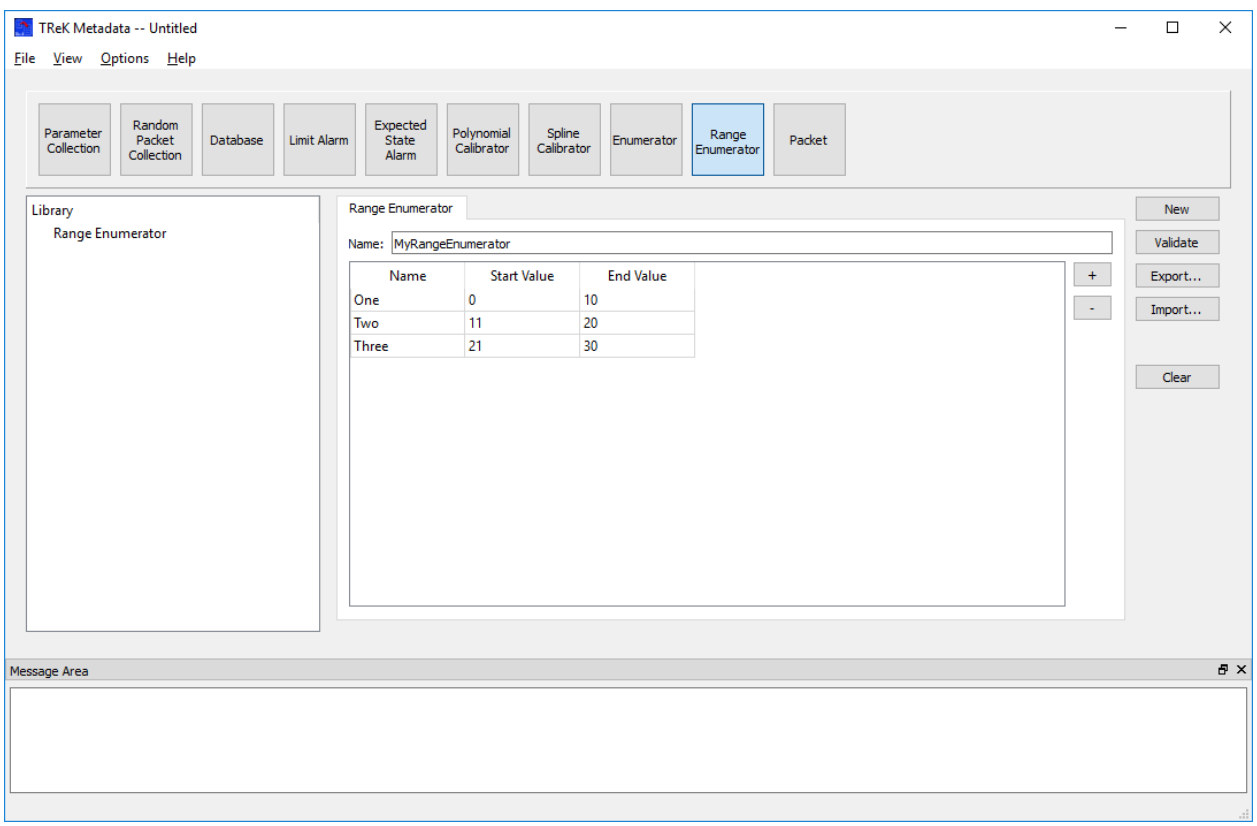

Figure 21 Populated Range Enumerator

The following functions are available when working with a Range Enumerator:

New
New provides the capability to clear the existing Range Enumerator information.

Validate
Validate provides the capability to validate the Range Enumerator information.

Export
Export provides the capability to save the Range Enumerator to a file.

Import
Import provides the capability to load a Range Enumerator from a file. When a Range Enumerator is imported the information is displayed on the Range Enumerator tab. To store any modifications back to the file, the Range Enumerator must be exported.

Clear
Clear provides the capability to clear the Work Area.

6.10 Packet

The Packet configuration provides the capability to work with a Packet. Figure 22 shows the Main Window in the Packet configuration.

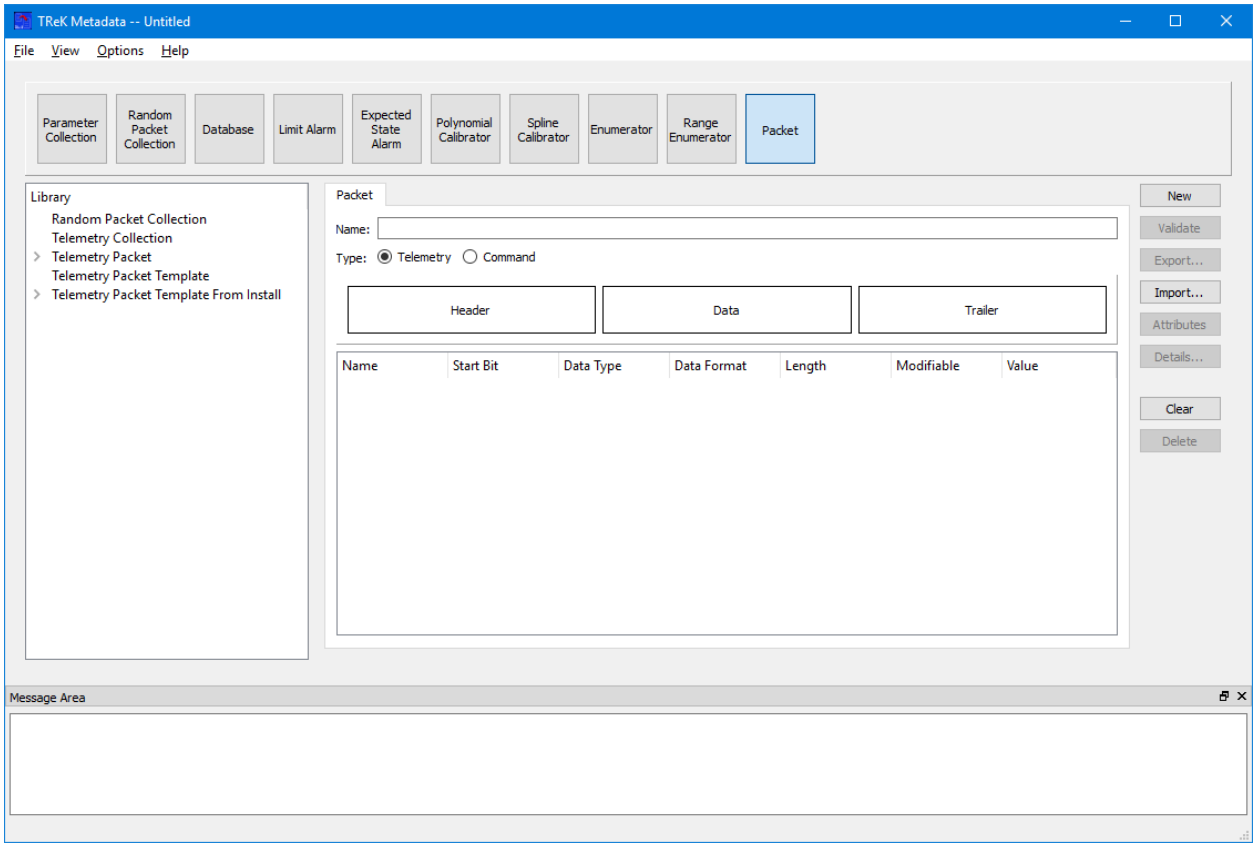

Figure 22 Packet Configuration

The Main Window Library Area is configured to provide access to items you can use to create or modify a packet. On the left hand side of the Main Window you will see the Library Area. In the middle of the Main Window, the Work Area provides a place to define the details of your packet. On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Packet you are working on. Each is described below.

<u>Library Area</u>

The Library Area is configured to provide access to items you can use to create or modify a Packet. The Telemetry Packet Template From Install list provides access to packet templates that are installed with the TReK software. You can create a packet from scratch. However, it is much easier to use a packet template and save yourself some work. There is also a Command Packet Template From Install. A packet template is a packet that includes pre-populated zones and packet attributes corresponding to the specific type of packet. For example, a CCSDS packet template contains a CCSDS Header and the following packet attributes: a Counter parameter, an APID Identifier, a Length parameter, and a Timestamp Parameter. By using a template, you can start your packet definition with all that information already defined. The Telemetry Collection or Command Collection list is configured to provide access to existing Parameter Collections stored in the trek_workspace. The Telemetry Packet or Command Packet list is configured to provide access to existing Packets stored in the trek_workspace. When

you create a Parameter Collection or Packet and export it to the trek_workspace, it will automatically be added to the Parameter Collection list or Packet list. The Random Packet Collection can be used to add a Random Packet Collection to a telemetry packet. The Telemetry Packet Template and Command Packet Template lists display user defined packet templates that are stored in the trek_workspace. The Library Area only displays items stored in the trek_workspace. If you need to work on an item outside of the trek_workspace, you will need to import the item using the appropriate configuration area.

Work Area

The Work Area provides a place to define the details of your packet. Notice the rectangle that contains three smaller rectangles with one for each zone of the Packet (Header, Data, and Trailer) as shown in Figure 23.

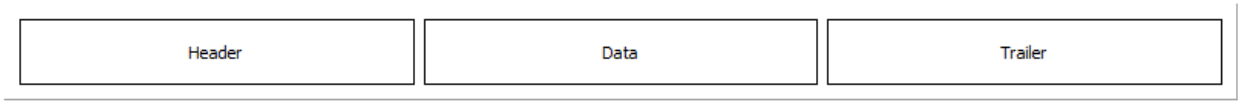

Figure 23 Packet Drop Zones

These rectangles are "Drop" zones. The large rectangle is a drop zone for the Entire Packet. If you drop a Packet Template or a Packet on the large rectangle, this will populate the entire Packet. If you drop a Packet or a Parameter Collection on one of the individual packet zones, that zone of the Packet will be updated. The drop zones will highlight so you can verify you are dropping the item in the correct location. Figure 24 shows what the "Entire Packet" drop zone looks like when it is highlighted. Figure 25 shows what the "Data Zone" drop zone looks like when it is highlighted.

Figure 24 Entire Packet Drop Zone Highlighted

Figure 25 Data Zone Drop Zone Highlighted

Button Area

On the right hand side of the Main Window you will see a set of buttons that provide various capabilities applicable to the Packet you are working on (e.g., Validate to validate the Packet or Export to export the Packet).

The Work Area provides the capability to define the following properties for the Packet:

Name
The name field is used to enter the name of the Packet.

Type
The type selection is used to identify the type of Packet: Telemetry or Command.

Packet Contents List
The Packet Content list is used to show the contents of the Packet. Figure 26 shows an example of a Packet displayed in the Packet Contents List.

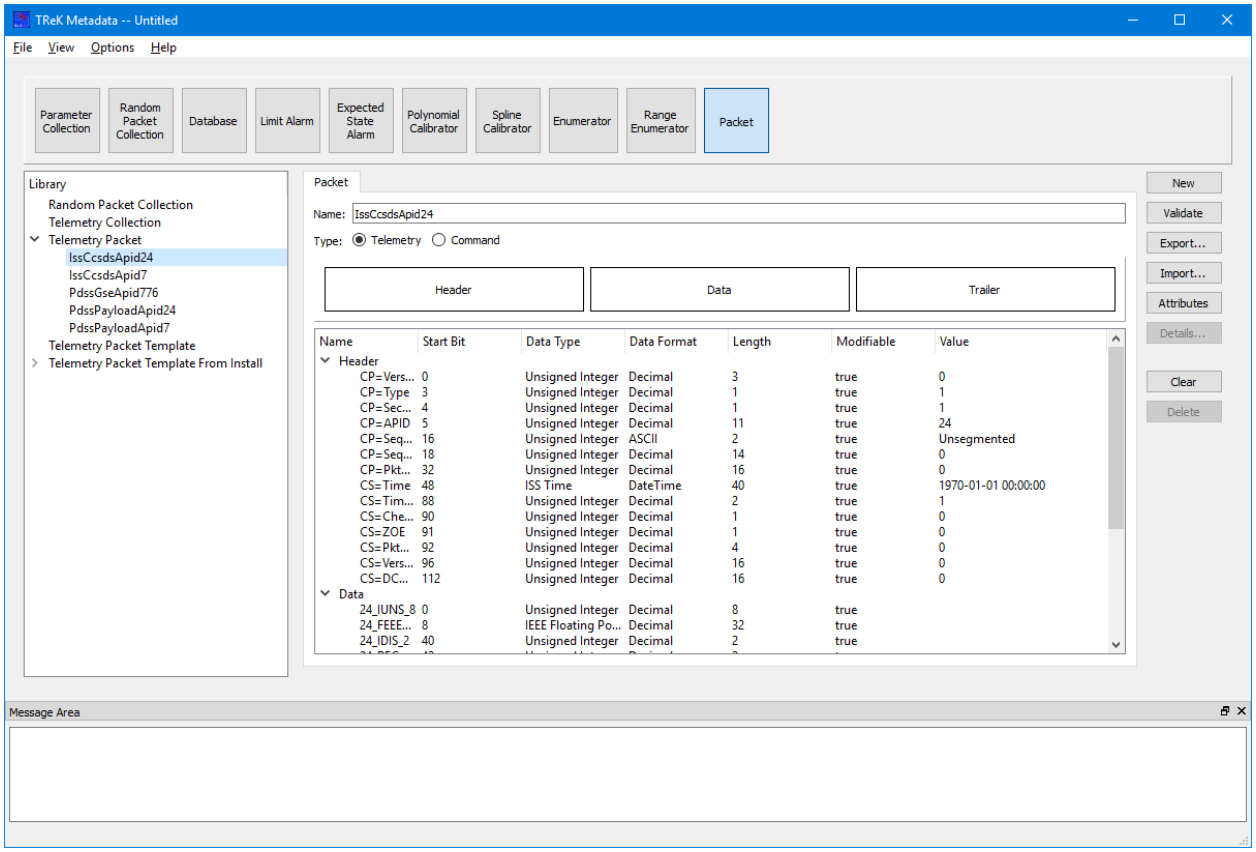

Figure 26 Packet Contents

The Metadata application uses Drag and Drop to move items from the Library Area into the Work Area.

To create or modify an existing Packet, you can drag items from the Library Area and drop them into one of the drop zones as described above. This will update the Packet information in the Work Area.

Packet Tips and Tricks

- If you drop a Packet Template or a Packet on the large rectangle, this will populate the entire Packet.

- If you drop a Packet or a Parameter Collection on one of the individual packet zones, that zone of the Packet will be updated.
- If you use your right mouse button you can access a context sensitive pop-up menu when selecting a Header Zone, Data Zone, or Trailer Zone in the list. This menu will provide capabilities such as Add Header, Add Data, Add Trailer, Replace, and Delete. This provides the capability to add, modify, or delete child zones.

The following functions are available when working with a Packet:

New
New provides the capability to clear the existing Packet information.

Validate
Validate provides the capability to validate the Packet information.

Export
Export provides the capability to save the Packet to a file.

Import
Import provides the capability to load a Packet from a file. When a Packet is imported the packet information is displayed on the Packet tab. To store any modifications back to the file, the Packet must be exported.

Attributes
Attributes provides the capability to define Packet Attributes. Please reference section 6.14 for details.

Details
Provides details about the selected item.

Clear
Clear provides the capability to clear the Work Area.

Delete
Delete provides the capability to delete a selected zone.

6.11 Export

The Export dialog is shown in Figure 27. This dialog is context sensitive and will show the Export Options available based on the configuration of the application.

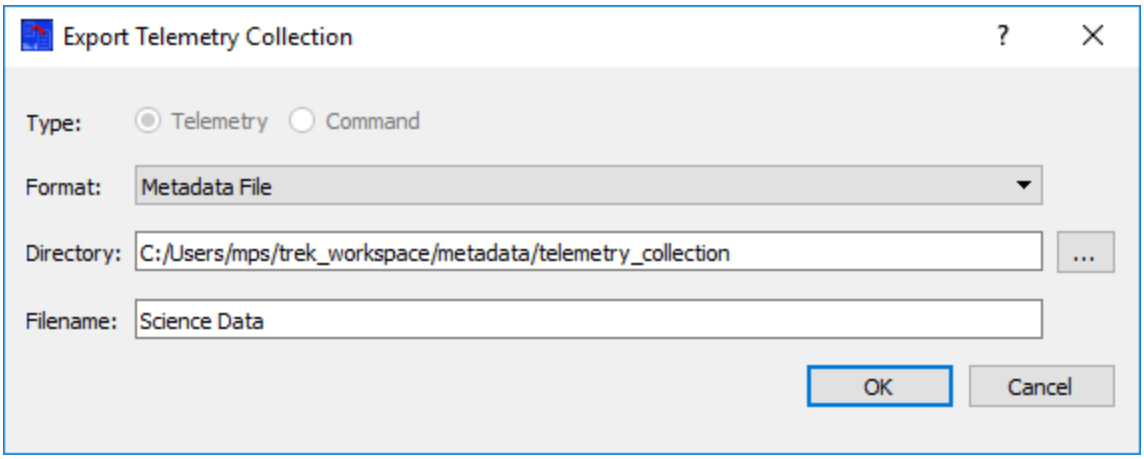

Figure 27 Export Dialog

Each field is described below:

<u>Format</u>
The Format menu provides the capability to select the format of the file.

<u>Directory</u>
The Directory field provides the capability to enter the Directory location for the exported element. It defaults to the appropriate directory in the trek_workspace.

<u>Filename</u>
The Filename field provides the capability to enter the filename for the exported item. It will default to the text entered into the Name field in the Work Area.

6.12 Import

The Import dialog is shown in Figure 28. This dialog is context sensitive and will show the Import Options available based on the configuration of the application.

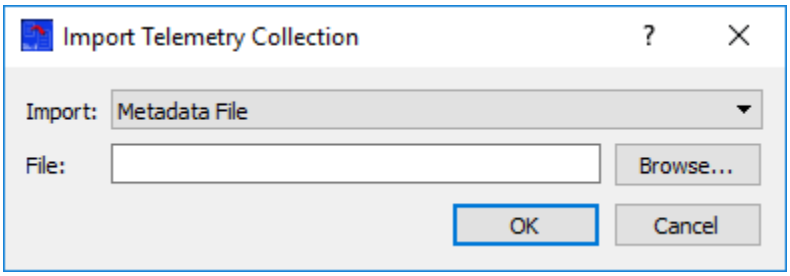

Figure 28 Import Dialog

Each field is described below:

Format
The Format menu provides the capability to identify the format of the file. If the file does not match the format identified, the import will fail.

File
The File field provides the capability to enter the absolute path for the item to import.

6.13 Parameter Details

The Parameter Details dialog is shown in Figure 29. The Parameter Details dialog provides access to the properties of the selected parameter. For information about the items in this dialog please reference the TReK Data library parameter class definition.

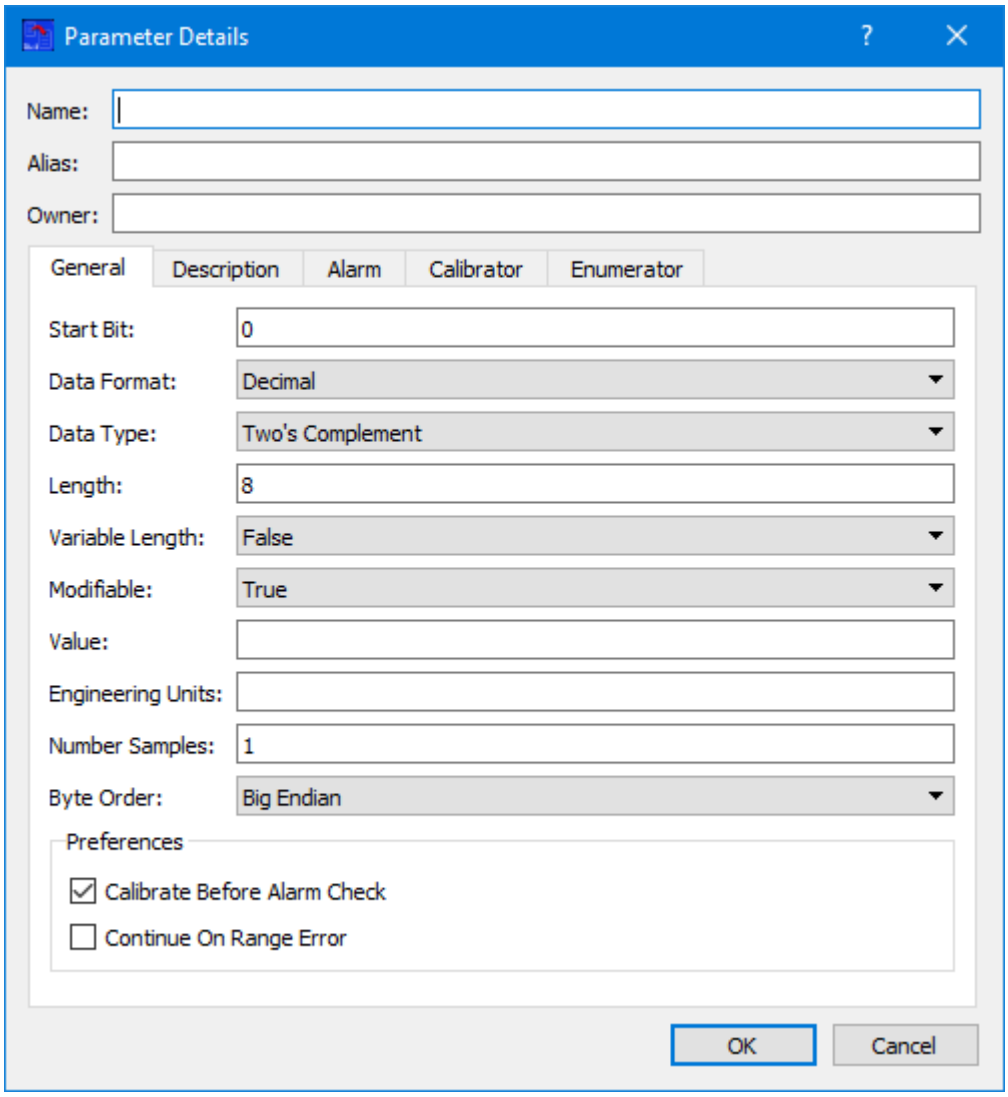

Figure 29 Parameter Details Dialog

6.14 Packet Attributes

The Packet Attributes dialog is shown in Figure 30. The Packet Attributes dialog provides the capability to define one or more attributes in a packet. For details on Packet Attributes please reference the TReK Concepts document. Each tab contains a checkbox to indicate if the Packet Attribute should be set in the Packet. If the checkbox is set, all the data entered on the tab will be validated and the attribute will be added to the Packet.

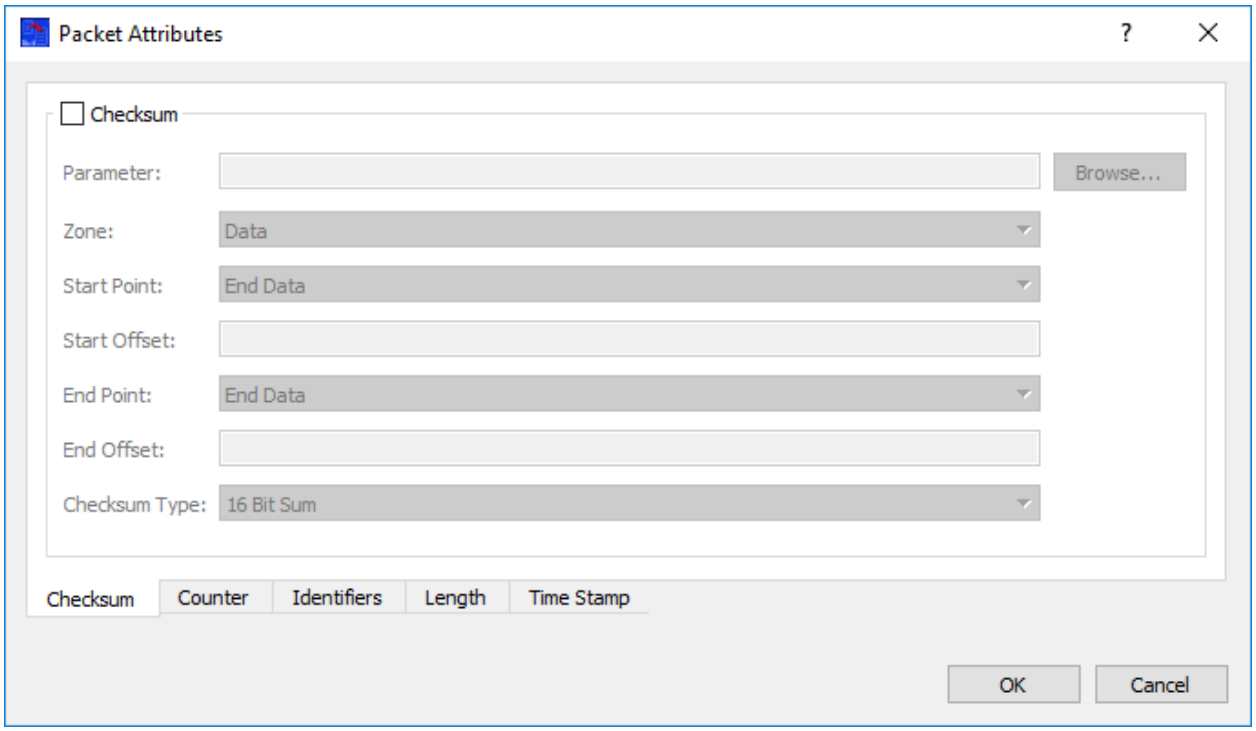

Figure 30 Packet Attributes Dialog

6.15 Convert EHS Partial Database Files Into TReK Database

The Convert EHS Partial Database Files Into TReK Database dialog is shown in Figure 31. This capability is available from the Options menu and is used to import data in EHS partial database files into a new TReK database. The new TReK database is created during the convert. The convert process can take a while if the EHS partial database files contain a large amount of data. You can continue to work on other tasks in the application while the convert is in work.

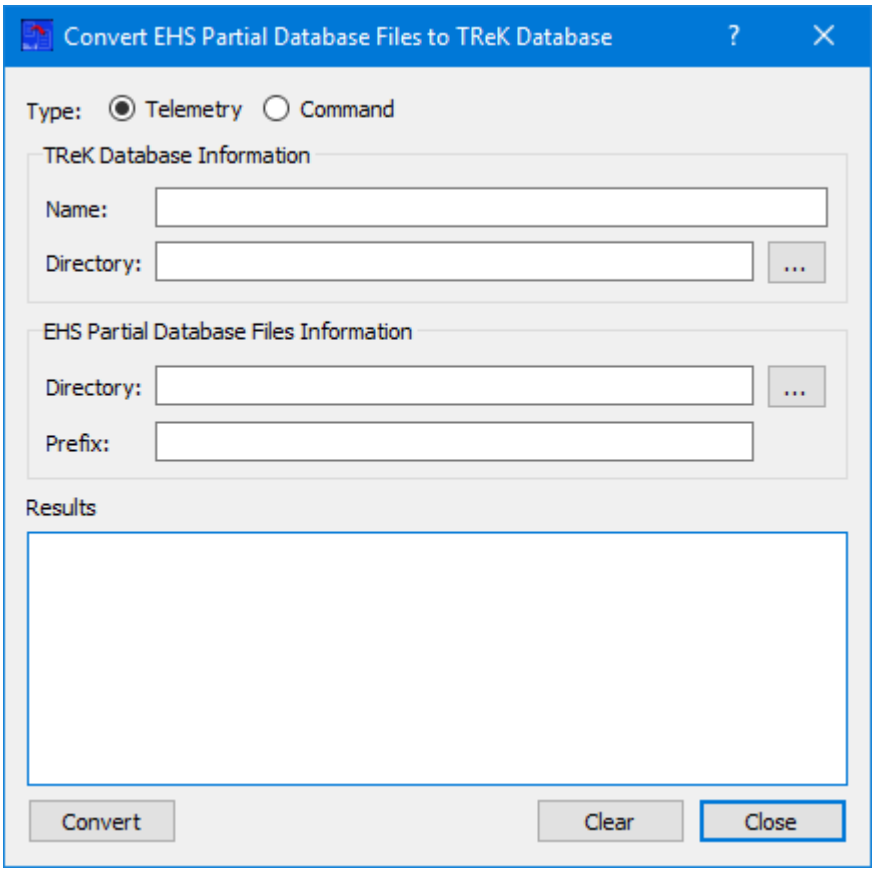

Figure 31 Convert EHS Partial Database Files Into TReK Database Dialog

Each field is described below:

<u>Type</u>
The Type is used to identify whether Telemetry information or Command information is to be imported.

<u>Name</u>
The Name field is used to enter the name of the new TReK database that will be created. If the Telemetry radio button is selected, a new TReK Telemetry database will be created. If the Command radio button is selected, a new TReK Command database will be created.

<u>Directory</u>
The Directory field is used to enter the directory where the EHS partial database download files reside. All the files for the convert (import) should be located in one directory.

<u>Prefix</u>

Filenames for EHS Partial Database files with telemetry information contain a prefix number. This number should be entered in the Prefix field. This field is not applicable for command information and will be grayed out when the Command radio button is selected.

Results
The Results area is used to display messages about the results of converting the data from the EHS partial database files to import it into the new TReK database. Any information, warnings, or errors will be displayed here.

Convert
The Convert button is used to execute the convert (import). Messages related to the convert process will be displayed in the Results area. The process can take a while if the EHS partial database files contain a large amount of data. You can continue to work on other tasks in the application while the convert is in work.

Clear
The Clear button clears the Results area.

Close
The Close button closes the dialog.

6.16 Export Metadata to CDEF

At the time of this release the CDEF standard was still under development. This capability will be provided in a later release. The following material is a preview of the current iteration of the CDEF standard.

The Export Metadata to CDEF dialog is shown in Figure 32. This capability is available from the Options menu and is used to export one or more telemetry or command packets into a Common Data Exchange Format (CDEF) file.

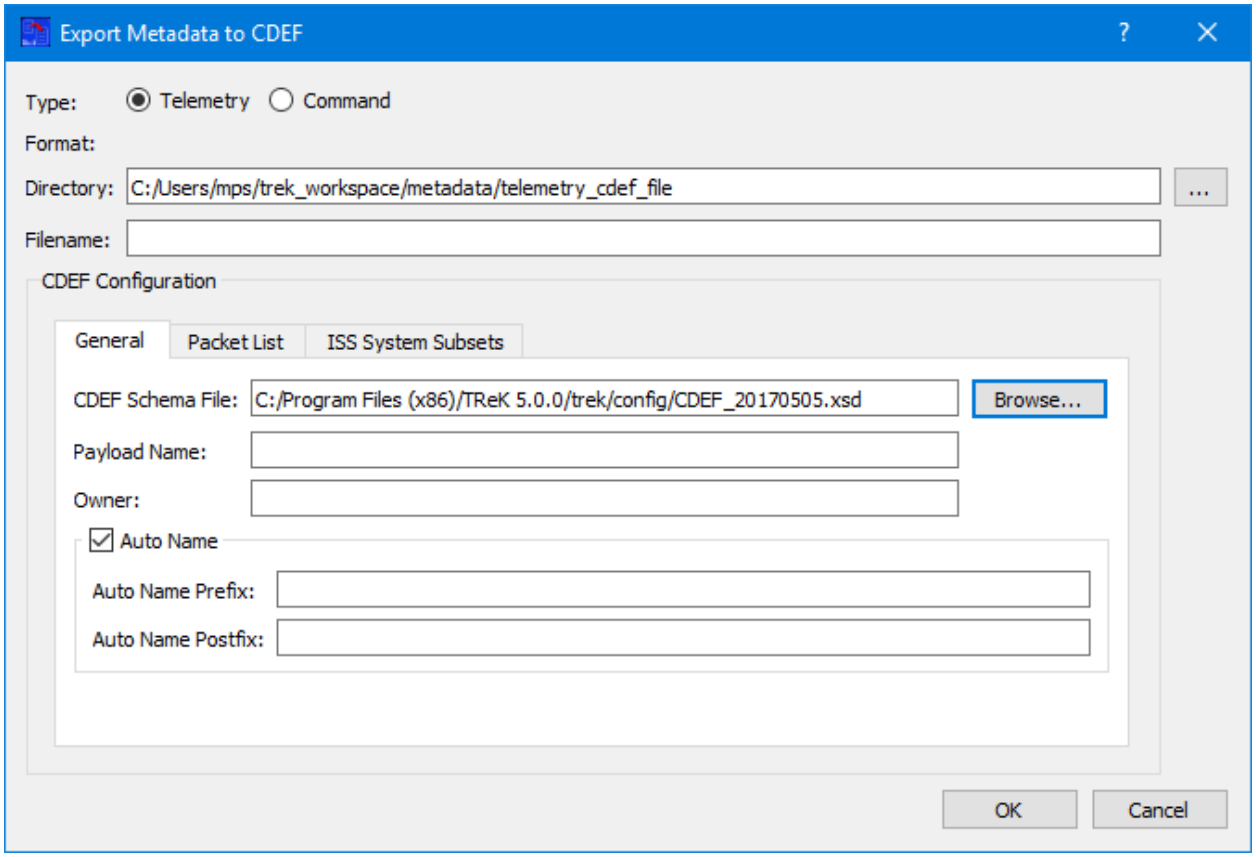

Figure 32 Export Metadata to CDEF Dialog

Each field is described below.

<u>Format</u>
The Format menu is set to Common Data Exchange Format.

<u>Directory</u>
The Directory field provides the capability to enter the Directory location for the exported element. It defaults to the appropriate directory in the trek_workspace.

<u>Filename</u>
The Filename field provides the capability to enter the filename for the exported item.

The CDEF Configuration area provides the capability to enter CDEF specific information. Most of the information collected is to allow you to easily enter information that is required in the CDEF file, but is not necessarily required in the internal TReK files and database. Each field is described below.

6.16.1 General Tab

The General tab is used to enter general information about the CDEF export.

Payload Name
This field is used to identify the Payload Name assigned by the ISS Program.

Owner
This field is used to identify the Owner for the telemetry data. This is only used when an owner is not available for an individual item (e.g., parameter). If one already exists, it will not be overwritten.

CDEF Schema Location
This field is used to identify the location of the CDEF Schema. The Browse button can be used to browse the local file system for the file location.

Auto Name Checkbox
Check the Auto Name Checkbox if auto naming should be used. If this is checked, the Auto Name Prefix and Auto Name Postfix are required. The ISS Program assigns an eight character name to payloads. If you did not use that name when creating you metadata for TReK, you can use this feature to have those names automatically generated. The names you provided for the parameters will be set to the 'alias' for the parameter to preserve more meaningful names.

Auto Name Prefix
This field is used to enter the Auto Name Prefix information.

Auto Name Postfix
This field is used to enter the Auto Name Postfix information. The postfix is a single character (typically 'U') that is added as the last character of the parameter name.

6.16.2 Packet List Tab

The Packet List tab is used to identify one or more packets to be included in the CDEF export. It is shown in Figure 33. The Browse Button is used to browse a TReK Database to select and add a packet to the list. The Browse Disk button is used to browse the local file system to select and add a packet using a TReK metadata file definition. The Delete button is used to delete an item from the list. If the export is for telemetry, all payload telemetry metadata information to be shared across systems (ISS, POIC, TReK etc.) must be included in the CDEF file. Many payload teams will only need to include health and status metadata. If the export is for command, all payload command metadata information to be shared across systems (ISS, POIC, TReK etc.) must be included in the CDEF file.

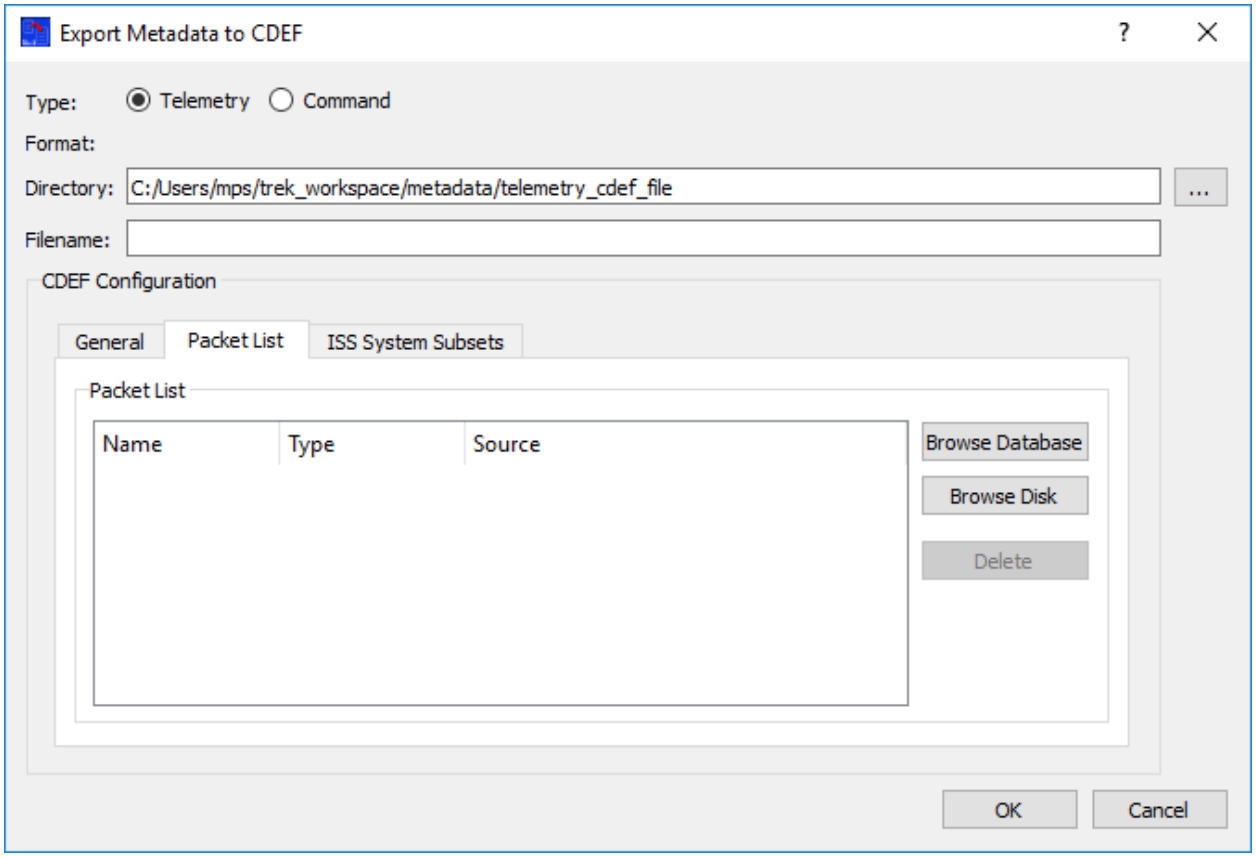

Figure 33 Export Metadata to CDEF (Packet List Tab)

6.16.3 ISS System Subsets Tab

When exporting to CDEF, the ISS System Subsets tab is used to filter out ISS system subsets. It is shown in Figure 34. The + (plus) button is used to add a row to the list. The – (minus) button is used to delete a row from the list. The Subset information can be entered directly into the list. The list should default to the ISS system subsets needed, but the lengths may change if the ISS system software is updated. The default lengths are those known at the time of this TReK release.

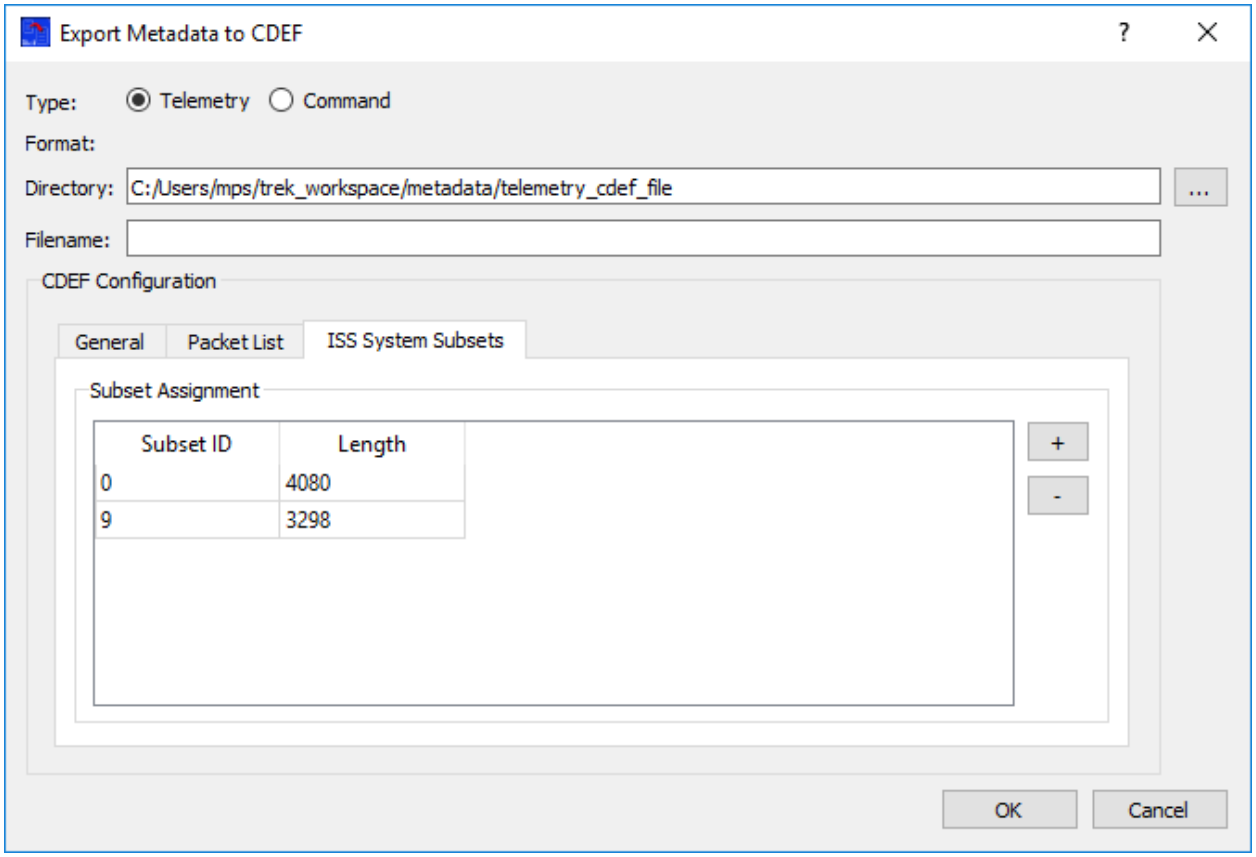

Figure 34 Export Metadata to CDEF (ISS System Subsets Tab)

6.17 Import Metadata from CDEF

At the time of this release the CDEF standard was still under development. This capability will be provided in a later release. The following material is a preview of the current iteration of the CDEF standard.

The Import Metadata to CDEF dialog is shown in Figure 35 Import Metadata from CDEF. This capability is available from the Options menu and is used to import one or more telemetry or command packets from a Common Data Exchange Format (CDEF) file.

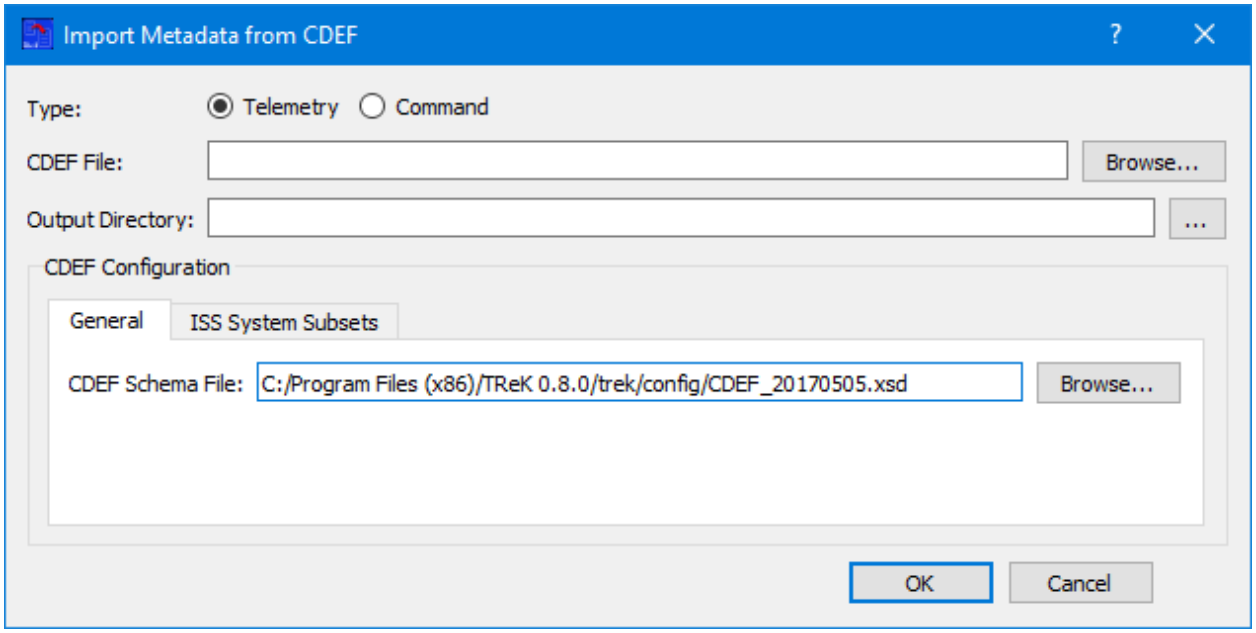

Figure 35 Import Metadata from CDEF

Each field is described below.

<u>CDEF File</u>
The CDEF File to import.

<u>Output Directory</u>
The Directory field provides the capability to enter the Directory location for the imported packets. It defaults to the appropriate directory in the trek_workspace.

The CDEF Configuration area provides the capability to enter CDEF configuration information. Most of the information collected is to allow you to easily enter information that is required to perform the CDEF Import. Each field is described below.

6.17.1 General Tab

The General tab is used to enter general information about the CDEF import.

<u>CDEF Schema Location</u>
This field is used to identify the location of the CDEF Schema. The Browse button can be used to browse the local file system for the file location.

6.17.2 ISS System Subsets Tab

The ISS System Subsets tab is used to assign ISS system subsets to APIDs in the CDEF Import to allow processing of health and status data during stand alone testing. It is shown in Figure 34. The + (plus) button is used to add a row to the list. The – (minus) button is used to delete a row from the list. The Subset information can be entered

directly into the list. The list should default to the ISS system subsets needed, but the lengths may change if the ISS system software is updated. The default lengths are those known at the time of this TReK release.

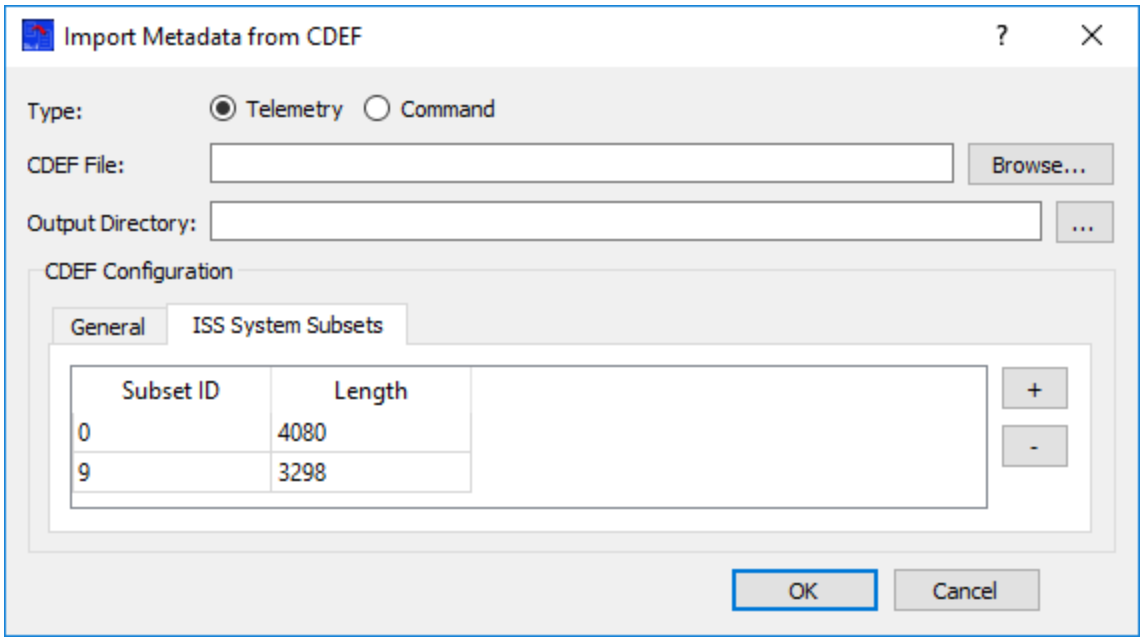

Figure 36 Import Metadata from CDEF (ISS System Subsets Tab)

6.18 Command Data Exchange Format Information

At the time of this release the CDEF standard was still under development. This capability will be provided in a later release. The following material is a preview of the current iteration of the CDEF standard.

This section will be updated prior to the final release. The Common Data Exchange Format is not completely defined at this time. The following types of information will be provided.

Rules for export (all data must be specified)

Note what capabilities are different in TReK and CDEF and that errors/warnings are generated.

Rules for import.

Note what capabilities are different in TReK and CDEF and that errors/warnings are generated.

6.19 User Defined C Structure

If a C structure is used to define telemetry or command data in payload code, the C structure can be imported into TReK to define the metadata. A C structure is read into a Parameter Collection which can then be inserted into any Packet template to complete the definition. You can edit the Parameter Collection to add additional information as needed prior to inserting it into the Packet template.

Allowed Types

The allowed types of data within the structure are more restrictive than the types allowed by the TReK. Additional types can be supported upon request. Currently all data is assumed to be byte aligned (packed). The allowed data types and their corresponding data types and lengths in TReK are shown in the table below. Arrays are supported for all data types. If the translation to the TReK data type differs for an array, the array type is listed separately.

C Type	TReK Data Type	Length (bits)
char	DT_TWOS_COMPLEMENT	8
char [n]	DT_NULL_TERMINATED_STRING	8*n
double	DT_IEEE_FLOATING_POINT	64
float	DT_IEEE_FLOATING_POINT	32
int	DT_TWOS_COMPLEMENT	32
int8_t	DT_TWOS_COMPLEMENT	8
int16_t	DT_TWOS_COMPLEMENT	16
int32_t	DT_TWOS_COMPLEMENT	32
int64_t	DT_TWOS_COMPLEMENT	64
long	DT_TWOS_COMPLEMENT	32
long long	DT_TWOS_COMPLEMENT	64
short	DT_TWOS_COMPLEMENT	16
uint8_t	DT_UNSIGNED_INTEGER	8
uint16_t	DT_UNSIGNED_INTEGER	16
uint32_t	DT_UNSIGNED_INTEGER	32
uint64_t	DT_UNSIGNED_INTEGER	64
unsigned char	DT_UNSIGNED_INTEGER	8
unsigned char [n]	DT_UNDEFINED_BYTES	8*n
unsinged int	DT_UNSIGNED_INTEGER	32
unsigned long	DT_UNSIGNED_INTEGER	32
unsigned long long	DT_UNSIGNED_INTEGER	64
unsigned short	DT_UNSIGNED_INTEGER	16

Byte Order

When importing a C structure into TReK you must specify how byte order of the data types within the structure should be interpreted. The byte order is for the system either sending the data to TReK or processing the data from TReK. The allowed byte order is big endian or little endian.

How a C Structure is Interpreted

The easiest way to show how a C structure is interpreted is with a simple example. Consider the structure below:

```
struct first {
        int8_t v1;
        uint16_t v2;
        float v3;
};
```

The structure will be written to a Parameter Collection XML file based on the name of the structure. In this case the file will be "first.xml". The Parameter Collection will consist of three parameters. Each parameter name will be the variable name (v1, v2, v3). A simple table shows some of the important attributes that are set for each parameter:

Name	Data Type	Length	Start Bit
v1	DT_TWOS_COMPLEMENT	8	0
v2	DT_UNSIGNED_INTEGER	16	8
v3	DT_IEEE_FLOATING_POINT	32	24

Enumerations

You can use enumerated types in your structures. All enumerations are interpreted as 32-bit unsigned integers. Enumerations can be defined with or without values. The enumerations shown below are all valid:

```
enum my_enum{ one = 1, two, three=777 };

enum my_enum2 { four=4, five };

enum my_enum3
{
        six,
        seven
};

typedef enum {
        eight,
        nine
} my_enum4;

typedef enum{ten,eleven,twelve}my_enum5;
```

Nesting of Structures

It is possible to nest structures. Nesting of structures produces names that include the nesting variables. Again a simple example is probably the best way to understand:

```
struct position
{
        double x;
        double y;
```

```
        double z;
};

struct locations
{
        struct position start;
        struct position end;
};
```

Two separate files will be produced in this example (position.xml and locations.xml). It is likely that the only XML file you really need is the top level structure (locations.xml). TReK produces files for all structures found within a file.

Name	Data Type	Length	Start Bit
start.x	DT_IEEE_FLOATING_POINT	64	0
start.y	DT_IEEE_FLOATING_POINT	64	64
start.z	DT_IEEE_FLOATING_POINT	64	128
end.x	DT_IEEE_FLOATING_POINT	64	192
end.y	DT_IEEE_FLOATING_POINT	64	256
end.z	DT_IEEE_FLOATING_POINT	64	320

The resulting Parameter Collection no longer has the nesting of data. Each parameter will be placed at the same level and contain a concatenation of names based on the variables from each structure.

Special Tags

Special tags can be used in structure comments to override information or provide additional input for the resulting Parameter Collection. Each tag takes the form of:

TAG[data]

TAG is the special tag name
data is the value for the tag

The allowed tags are shown in the table below. Additional tags may be provided in a future release.

Tag Name	Description
NAME	Overrides the variable name and uses the specified value as the name of the parameter
ALIAS	Sets the alias of the parameter to the value specified
DESC	Sets the short description of the parameter to the value specified.
FIXED	Sets the value for the parameter as specified and sets the modifiable flag to false.
MOD	Sets the value for the parameter as specified and sets the modifiable flag to true.
LOW	Sets a low range value as specified.
HIGH	Sets a high range value as specified.

A short example of a structure will show how to use the special tags. The tags must appear on the same line as the variable.

```
struct start_health_and_status
{
        uint16_t cmd_id;     // NAME[CommandId] FIXED[2]
        uint16_t rate;       // DESC[Send data every n seconds] LOW[1] MOD[1]
        char ip_addr[16];    // NAME[DestIpAddress] MOD[192.168.1.6]
        uint16_t dest_port;  // MOD[6200]
        uint16_t id;         // ALIAS[DownlinkId] MOD[3]
};
```

Other Rules to Know

There are some other things to consider when using this feature.

- The translator from the C structure to Parameter Collection is not a full C parser. It only knows a limited amount of information. It does not have the capability of looking in 'include' files so all of the information about enumerations and structures must be defined in the same file.
- The translator currently does not support any of the standard C structures such as struct tm. These may be added in future releases as time permits or upon user request.

6.20 Application Messages

Various types of application messages are generated including information, progress, warning, error, and debug messages. Information, warning, and error messages will be displayed in the main window message area. All application messages are sent to the Messages dialog shown in Figure 37. The Messages dialog can be configured to display specific types of messages. By default, the Messages dialog will display information, progress, warning, and error messages. Columns in the Messages dialog can be sorted by clicking on the column header. The Messages dialog is available from the Options menu.

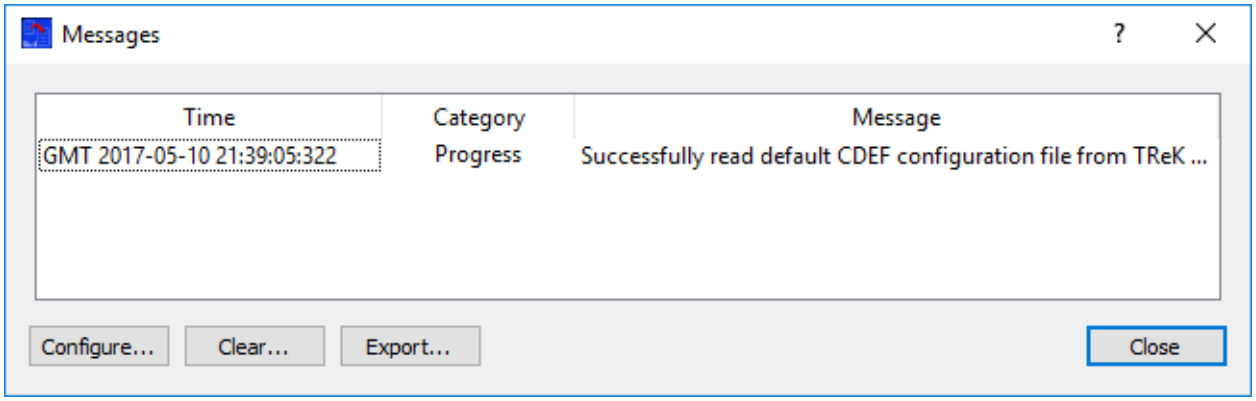

Figure 37 Messages Dialog

Configure

The Configure button provides access to the Configure Messages dialog shown in Figure 38. This dialog is used to filter the types of messages (category) displayed in the Messages dialog.

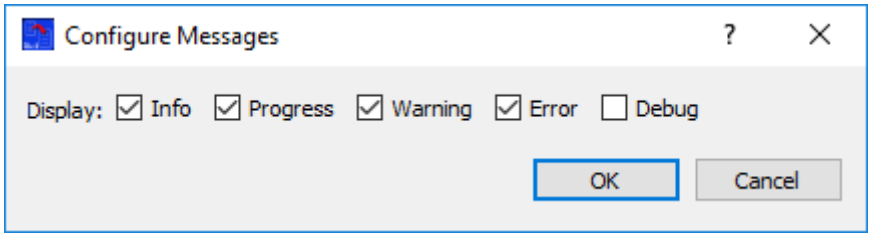

Figure 38 Configure Messages Dialog

Clear

The Clear button provides access to the Clear Messages dialog shown in Figure 39. This dialog provides two ways to clear messages in the Messages dialog. You can clear all the messages or clear selected messages. Once you clear messages, the messages are permanently deleted.

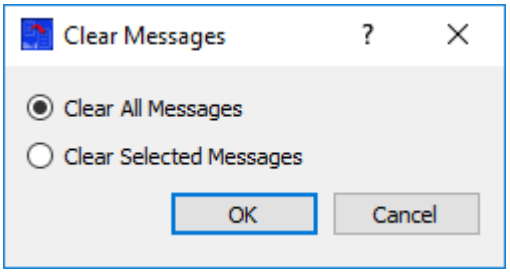

Figure 39 Clear Messages Dialog

Export

The Export button provides the capability to save all the application messages to a file. When you push the Export button you will be prompted for a directory and filename. Export will save all messages, not just the messages currently displayed in the Messages dialog (i.e. the 'Configure' filter is not applied). The name you provide for the file will be modified with a time tag appended to the filename. The time tag indicates the time the file was closed. For example:

Filename Input: messages.txt
Filename Output: messages_2017-05-07_13~03~28.txt

6.21 Application Configuration File

The Metadata application saves the information entered for each work area.

6.22 Application Settings

The Metadata application saves some settings as application settings each time you exit the application. The next time you run the application, the application will initialize with the previous application settings. The following application settings are saved:

- Application Window Size

7 FAQ and Troubleshooting

This section addresses Frequently Asked Questions and provides tips for troubleshooting common gotchas.

No FAQs Yet.