

# Message Maker User's Guide

Message Maker Version 1.0

**DRAFT 1.0**

Robert Snelick

Len Gebase

Sydney Henrard

National Institute of Standards and Technology

rsnelick@nist.gov

**Please note: This is a first draft—it is not a complete user's guide and may not be current with the Message Maker tool.**

## Preface

The Message Maker Project is a collaboration effort between the National Institute of Standards and Technology (NIST) and the Health Level 7 (HL7) Standards Consortium. NIST is directing their efforts towards the development of a conformance-testing tool that automatically generates test messages for HL7 message specifications. The messages can be used to test systems for conformance to a given profile. Message Maker is public domain software and is freely available. Message Maker is a work-in-progress, as such, not all planned functionality is implemented. This release is intended as a prototype; feedback on its design, feature set, and usefulness is welcomed.

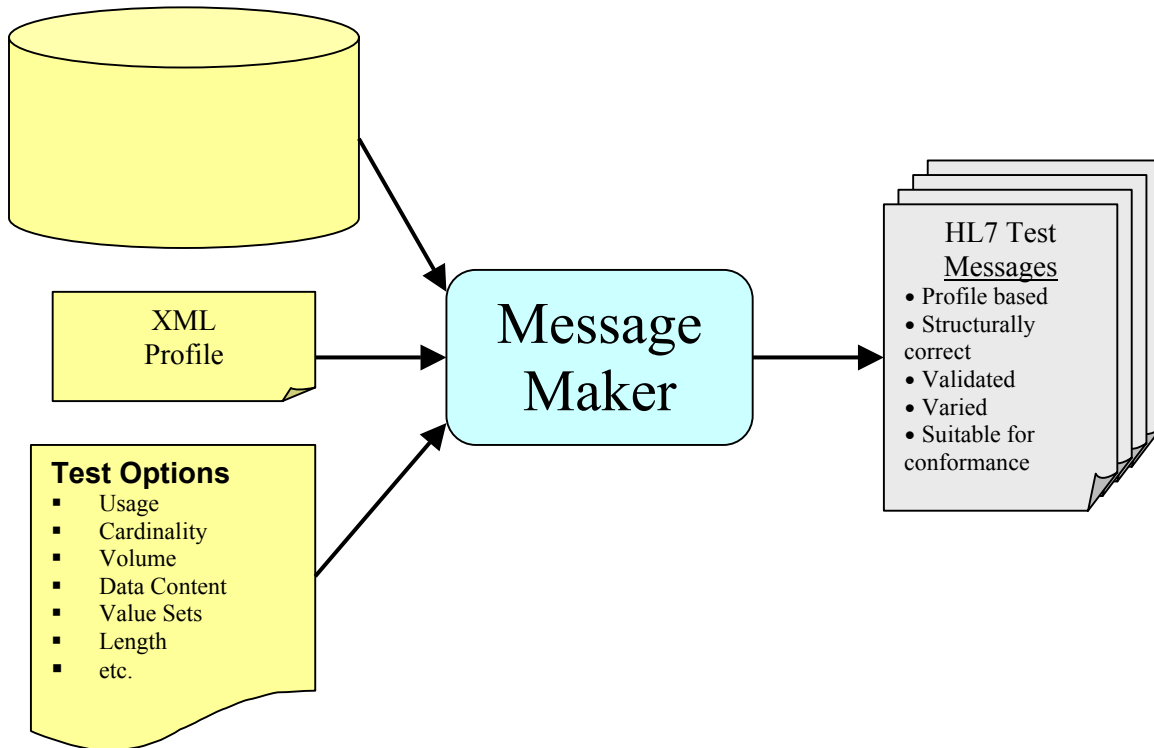
This User's Guide provides an introduction on how to install and use the tool. This version gives an overview of key functionality. It is not yet a comprehensive guide. Please see Appendix A in this guide for a current list of known limitations of Message Maker.

## 1 Background

HL7 is an *application-level messaging standard* for the healthcare industry and defines the interfaces that allow centrally located and/or distributed information systems to communicate. HL7 establishes the rules for building interfaces and provides many optional features to accommodate the disperse needs of the health care industry. However, for interfaces to be implemented, a precise and unambiguous specification must be defined. Message profiles can be employed to constrict the definition of a message in a manner that specifically states the optional constructs and processing rules of a message. Tools, such as the Messaging Workbench (MWB), have been developed to help in the construction of message profiles. Interfaces modeled by a message profile need to be tested to ensure that they have been implemented correctly. Current practice involves meticulous by hand debugging of implementations as problems appear. Test message suites and the tools to create them are needed. Message Maker is one such tool that automatically and dynamically generates test messages.

Message Maker uses the message template provided by the profile definition and adores the primitive elements (fields, components, and sub-components) with data content to

create test messages. The data used to populate the messages is drawn from a number of sources including the NIST developed database of HL7 data items, HL7 tables, user tables, and external tables. Alternatively, data can be extracted from a site-specific database.



**Figure 1. Overview of Message Maker Process**

Message Maker creates a test suite of messages for a given profile. The messages can be valid or invalid and contain variation from message to message. An example of an invalid message is a missing data item for a required field. A number of test parameters can be varied in the construction of a message. These may include segment and field cardinality, the usage of certain primitive fields, value sets, data content, and more. Data content variation is achieved by randomly selecting items from the HL7 items database.

The core engine of Message Maker generates messages in an XML format. These messages can be subsequently transformed into the HL7 ER7 format. Figure 1 depicts a functional overview of Message Maker. Processing is XML based. Profile generation tools, such as MWB, export message specifications as XML documents. Data sources, converted into XML files, along with the profile are used by the XSL transformation generation engine to create the test messages. Descriptive metadata is recorded for each message. This information can be browsed to gain insight on the purpose of the test messages. Later stages of this project will include a testing framework that will utilize

these messages to examine the correctness of an interface implementation to a specification.

## 2 Installation

### 2.1 Download

A copy of Message Maker can be obtained from the following website:  
<http://www.nist.gov/messagemaker>

You can download either a self-installing executable jar or the entire software bundle that includes the source code. Choose the self-installing executable if you just plan on using the tool. The file name of the installer is *maker\_install\_n.n.jar*. Where *n.n* represents the version number.

### 2.2 System Requirements

Message Maker uses Java and XML technologies. The following software is required:

- Java 2 SDK (Version 1.4.2 or later)

Note that the self-installing executable is a jar file, so the Java 2 SDK needs to be installed in order to run it.


Message Maker has been tested on both Windows and Linux platforms. The following operating systems are supported:

- Windows XP
- Windows 2000
- Linux Redhat 9.0

Since Message Maker is Java based, it is likely that the software will run on other systems. However, the software has not been tested on other systems.

### 2.3 Installing Message Maker

Double-click on the installer's icon to run the self-installing executable to unpack and install the Message Maker software bundle. Then follow the instructions the installer provides—they are self-explanatory. Upon successful completion, you should see the *run\_mm.bat* (in the *maker bin* directory under your chosen install directory) and the

desktop short-cut icon  if you requested one. When done with the installation, you can delete the download file to recover disk space.

**Troubleshooting:** Installer won't execute.

The *installer* may not execute if the jar file does not have the proper file type association. One instance where this may occur is when you upgrade your java SDK version. First check to see that the *installer* is an *executable jar file* by examining its properties. If it is

and still doesn't work, ensure that the proper execution instructions are associated with the file type. Below gives the steps for Windows 2000. Similar steps apply for other versions of Windows, including Windows XP.

In the *Window's File Explorer*, select the "**Folders Option ...**" item from the "**Tools**" menu. Then select the "**File Types**" tab. Check to see if there is a "*JAR*" extension with an "*Executable Jar File*" File Type in this list.

If it is not present:

- Select "**New**" and enter "*Jar*" for the file extension.
- Select the "**Advanced**" button and enter "*Executable Jar File*" in the text field at the top of the dialog
- Select "**New**". Enter "*Open*" under Action: and enter "*C:\j2sdk1.4.2\_06\bin\javaw.exe -jar "%1" %\**" (or the location of the *javaw* executable on your system) under "**Application used to perform action**"
  - check the "**Use DDE**" box and enter the following in the text fields:  
**DDE Message:**  
**Application:** *javaw*  
**DDE Application:**  
**Topic:** *System*

If it is present:

- Confirm that it "Open withs" *javaw*. If not, select the **Change** button to launch the "**Open With**" dialog. Choose "**Other**" and navigate to the location of your Java installation (e.g., *C:\j2sdk1.4.2\_06\bin*) and then select "*javaw*".
- Select "**Advanced**" and enter the string "*Executable Jar File*" in the text field at the top of the dialog.
- Under the "**Actions**" box, enter "*Open*" and make sure the "*Confirm open after download box*" is checked.
- Select "**Edit**" and enter the for these fields:

**Action:** Open

**Application used to perform action:** *C:\j2sdk1.4.2\_06\bin\javaw.exe -jar "%1" %\**

**Use DDE:** (confirm this box is checked)

**DDE Message:**

**Application:** *javaw*

**DDE Application:**

**Topic:** *System*

Note that in some cases it is easier to remove an existing Jar file type association and start with a new one.

## 2.4 Running Message Maker

To run Message Maker double-click on the  desktop short-cut icon or run the `run_mm.bat` file in the directory where Message Maker was installed.

## 3 Overview of Message Creation

Below the basic steps to create messages for a given profile are listed. The user can control to various degrees the type of messages created and the data content of the messages. Alternatively the user can choose to let Message Maker control these parameters automatically.

### Message Maker Process:

**Initialization-** In this step the user selects the *Profile* in which the messages will be based upon. A *session name* attaches a name to the current message generation setup. It can be used to later recall settings. The *message path* indicates the location where messages are to be stored.

**Setting Options-** The user can choose from a number of options when generating messages. By default Message Maker will create a set of valid message. The user can keep the default settings or customize the process. The *volume* parameter sets the number of messages that will be created.

**Data Initialization-** The user can set the values in the configuration file and the element database as needed. Values set in the configuration file become the default value for a particular data element. A typical element that might be set in the configuration file is the Sending Application (MSH-3.1). Also, values in the element database can be modified. Various utilities are provided by Message Maker to customize the data files.

**Message Generation-** After initialization is complete messages are generated automatically. Message Maker provides a utility that monitors the progress of the generation process. Messages are automatically validated against the profile to ensure correctness. A trace of the process is provided in a log file.

**View/Modify Messages-** After successful completion of message generation, the test messages can be viewed and/or edited. Message Maker provides three views: 1) XML, ER7, and Enhanced ER7. Messages in the Enhanced ER7 view are presented in a tree structure where element values can be modified and saved.

**Message Management-** Message Management provides a utility that allows the user to browse current and previously created messages. It also provides utilities to help organization message test suites (not implemented yet).

## 4 Initialization

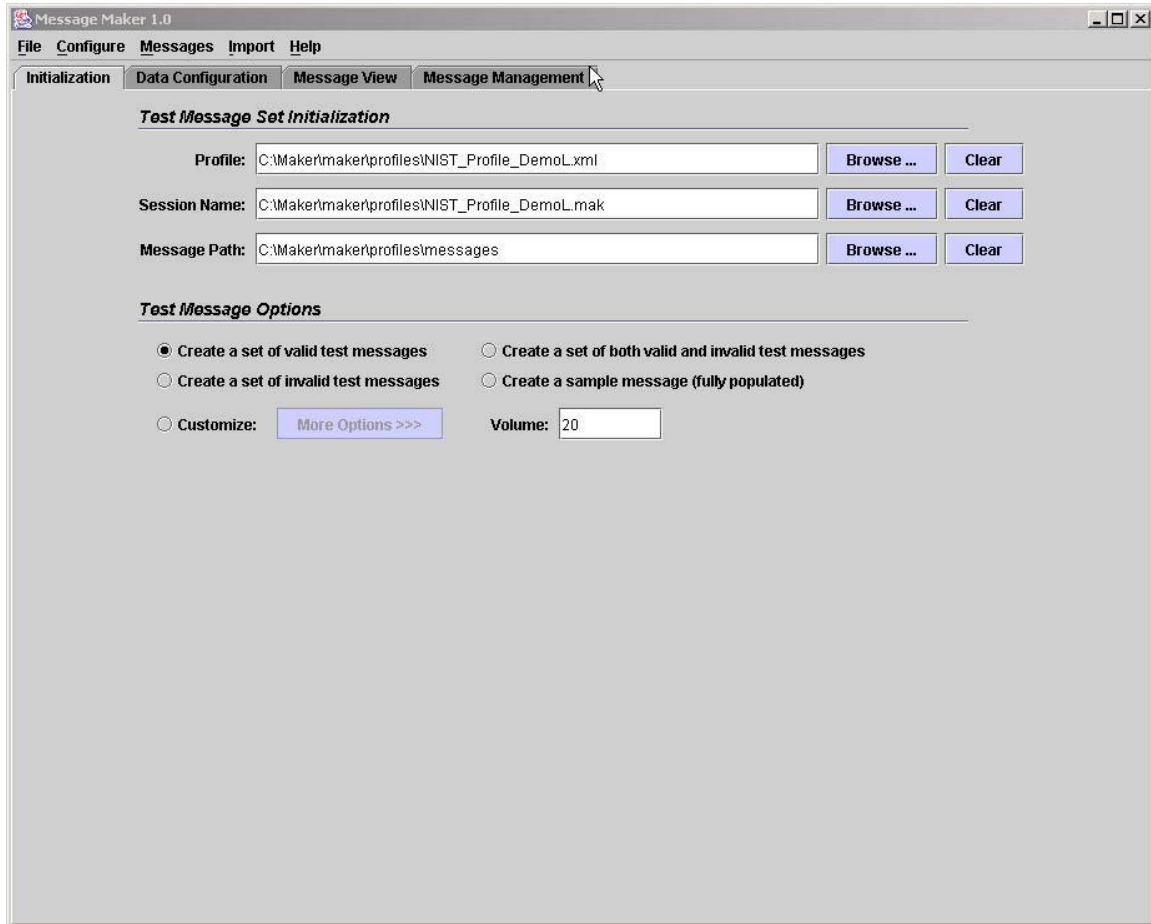
### 4.1 Selecting a Profile

Message generation in Message Maker is driven from profiles. The profile provides the *template* that guides message creation. So to begin the process choose the profile in which you want to create messages for. The profile can be selected from the Initialization Panel (the default panel upon startup, Figure 2). Click on the “Browse” button to locate a

particular profile or enter the path and file name directly in the text field. Once a profile is selected the *session name* and *message path* directory are automatically named. These values are provided for convenience and are suggested values that can be modified. The *session* provides persistence that saves the settings for the message generation process. This allows for repeating of message generation and partial setup. The *message path* indicates the location where the message files are created and stored.

**Demonstration:** To familiarize yourself with Message Maker’s functionality it is suggested that you select the “NIST\_Profile\_DemoL” example provided in the profiles directory as you read through this user’s guide.

## 4.2 Test Options



**Figure 2. Initialization Panel**

Message Maker allows for various testing options. Test options include the number of messages that are created, the message validity, constraints on usage, segment and field cardinality, and more.

### 4.2.1 Predetermine Message Generation Sets

Message Maker by default provides testing variation. When selecting one of the three general test options message variation is provided. You can choose to create messages that are valid, invalid, or a mixture of both. By default all messages are valid. There is also an option to choose a fully populated sample message. By choosing one of these options, you are allowing Message Maker to control which test options are varied.

### 4.2.2 Select the Number of Messages to Create

The Volume field allows you to select the number of messages to create. The default number is 20.

### 4.2.3 Customize Message Creation

You can have finer control of the testing options by selecting **Customize** and the **More Options** button. This will expand the Initialization Panel (Figure 3). This group of testing options allows you to select the type of messages you want to create. Appendix B gives a list of the options and their meaning.

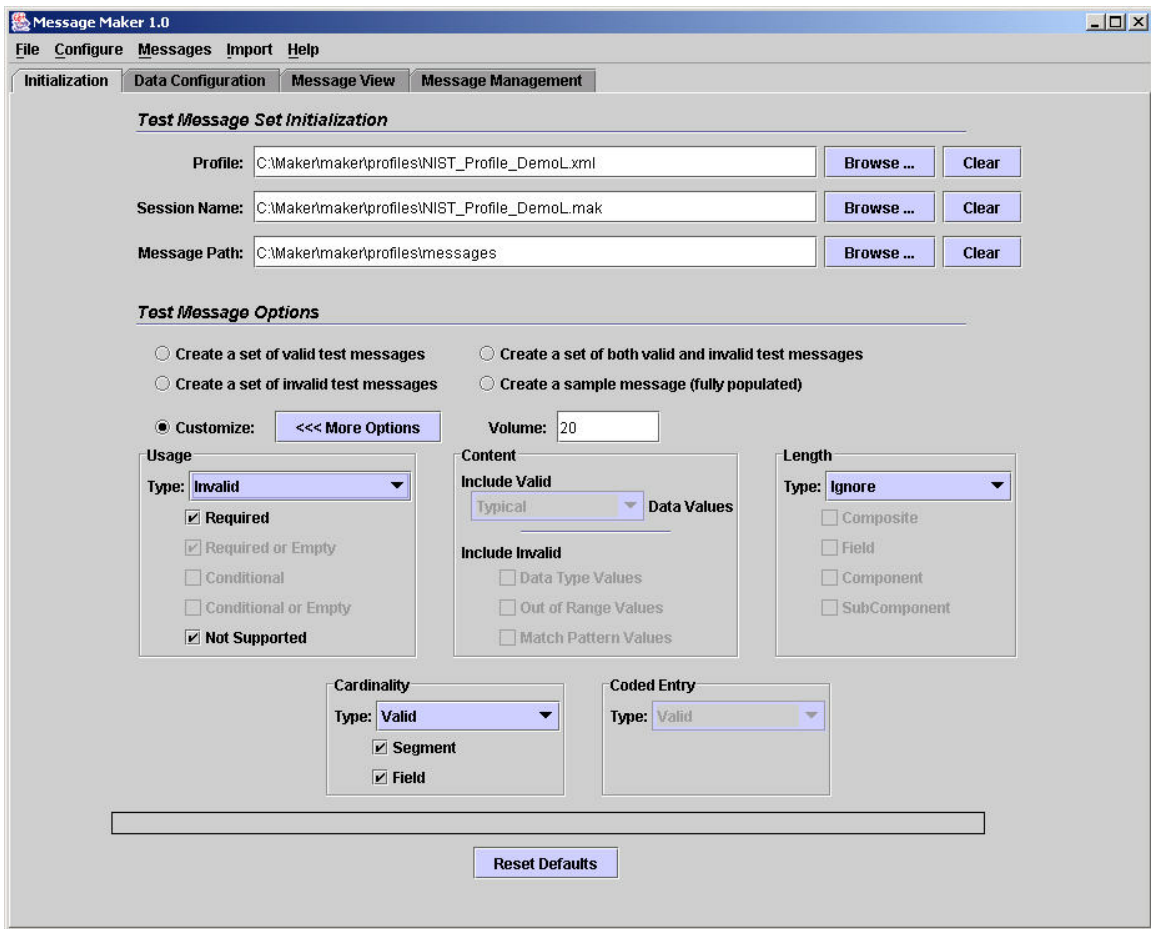


Figure 3 Expanded Initialization Panel

The options set by the user in combination with the number of messages to be generated, determine the types (variation) of messages that are generated. The types of messages generated are divided roughly evenly based on which options are selected. For example, if 10 messages are to be generated and *Field Cardinality* and *R Usage* are selected, then 5 messages for testing field cardinality and 5 messages for testing usage will be generated. The composition of these messages will depend on the setting of the Validity option. Validity can be set so that all messages are valid, all are invalid, or a combination of valid and invalid messages are generated. If only valid messages are being generated, then the cardinality messages that would be generated in this example would include a mix of Fields that repeat the minimum number of times allowed by the profile's cardinality setting, the maximum number of times allowed by the setting, and a number of times that lies between the two end point values when the difference between these values is greater than one.

In the case where the Validity option is set so that all messages generated are invalid, then each cardinality message generated in our example will include a Field that violates the cardinality constraint imposed on the Field by the profile. The violations will be distributed roughly equally between messages containing a single Field that repeats more than allowed by the maximum cardinality setting and messages that contain a single Field that repeats less than allowed by the minimum cardinality setting. In the unusual case where no minimum cardinality Field can be produced (Min=0 for all Fields), then only maximum cardinality violations will be produced.

In the case where the Validity option has been set to allow for both valid and invalid messages, more valid than invalid messages will be generated. Currently, values are set so that roughly one message in four will be invalid.

The variations in the data used for populating the messages is largely a function of the size of the set of data values available for each primitive element, Field, Component, and Subcomponent. These data values are maintained in a set of XML files that are accessed by the message generation software. The software uses a random number generator to determine the location it will use as its starting point for selecting values from the files. The frequency with which the same location will be selected will then largely be determined by the number of possible values. Subsequent values are selected from the files using a scheme that will increment the starting location by a number that depends on the hierarchical structure of the profile. This scheme should result in relatively little repetition of data values, provided a sufficient number of possible data values are available.

*Cardinality* and *Usage* options result in messages being generated in an analogous manner. The length options work somewhat differently. Length constraints can either be enforced or ignored, but the option's setting doesn't otherwise impact the messages that are generated. When length constraints are being enforced for primitive elements, repeated attempts will be made to find suitable values satisfying the constraint; generally if such a value is present in the XML data files, it will be used. When the composite length option is set, its only impact is to produce a warning in those cases where a



composite element might violate its length constraint, even though none of its component elements violates any length constraint.

When an option for generating a length error is set, a single element in the message will be instantiated with a value that violates the length constraint for some numeric (NM) or string (ST) type only.

#### 4.2.4 Message Generation Characteristics

Each generated message is different in terms of both structure and content. Below is a list of message characteristics that should be considered when choosing options for message generation:

- data content is randomly selected from a database
- at most one error is introduced for a given message
- at most one variation is introduced for a given message

### 4.3 Data File and Configuration

Message Maker provides various ways to input and modify the data that is used in message generation. Configuration, Table, and Reference data can be defined and modified. Configuration data are values that will always be used in a primitive data element. Any item in the reference database can be modified with the use of the “Data Configuration” panel (including MSH segment and table data). MSH configuration data can be set with the MSH Segment convenience dialog. Table data can be set or modified with the Table Viewer dialog.

#### 4.3.1 Data Files

Message Maker maintains a number of XML data files for use in message generation. A brief description for each is given below:

**ConfigData:** This is a special file that contains data items, that if set is always used for an element. Typical *configuration* data might include the HL7 version number or the name of the sending application in the MSH segment. If an item is set in this file, the message generation software will use this value when generating messages.

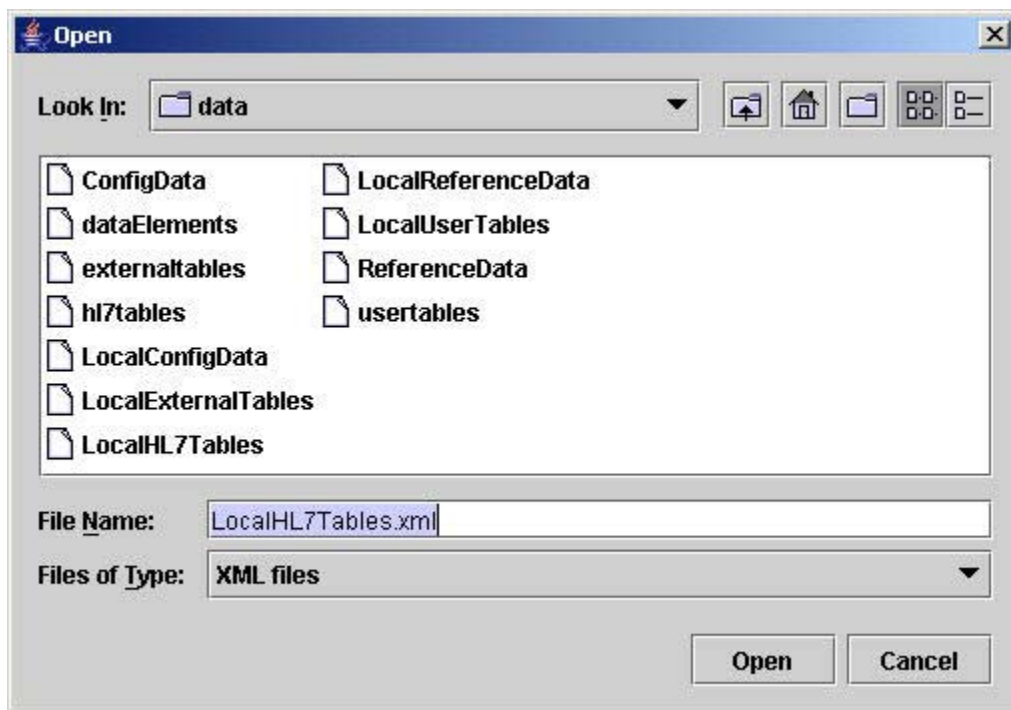
**hl7tables:** This file contains the list of tables defined by HL7. These include HL7 and HL7User table types. The file contains the table id, name, type, and code/value pair. The initial values for the tables are those defined or suggested by the HL7 standard.

**usertables:** This file contains the list of tables defined by HL7 to be of type “User”. There are no suggested values in the standard—the values are site specific (e.g., a hospital room number). These values are typically unique to a particular installation and are defined locally. NIST provide values for these tables (i.e., NIST creates sample values). The NIST created values may be sufficient for many testing scenarios. In such cases where they are not, the tables can be modified with the *Table Viewer*.

**externaltables:** This file contains the list of tables defined by HL7 to be of type “User” with suggested values coming from an external source such as ISO.

**ReferenceData:** The NIST HL7 Reference Database contains example data values for HL7 data elements that are not found in other data sources such as HL7 tables. The database contains items for the HL7 primitive data elements (i.e., field, components, and sub-components). The data items are contained in an Access database and are transformed into an XML file for processing by Message Maker.

Together, these data sources are used in the message generation process. Each of the data source can be modified. When values from a data source are modified, Message Maker will request a name of the file to save (Figure 4) the new data source. By default Message Maker suggests the “Local” prefix. Message Maker will maintain any number of *local* versions of the file.



**Figure 4. Saving Data Files**

The user can select which data files they want to use when generating messages. This can be accomplished with the “**Data Files**” dialog under the “**Configure**” menu (See Figure 5). Once set, the *Local* data files can be used for any message generation session.

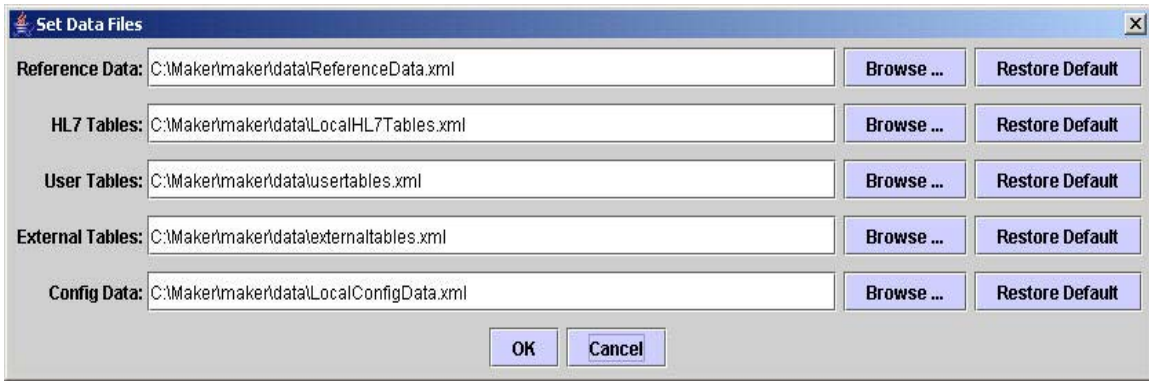


Figure 5. Set Data Files

### 4.3.2 MSH Segment Data Initialization

Configuration data in the MSH segment can be set by using the “MSH Segment” dialog under the “Configure” menu. This dialog presents the data in a tree structure from the profile point of view. Data items colored in yellow can be modified. Items deemed as not supported in the profile are colored in gray and contain the text “Not supported by Profile”. Items colored in green, cannot be modified (e.g., the message type MSH.9).

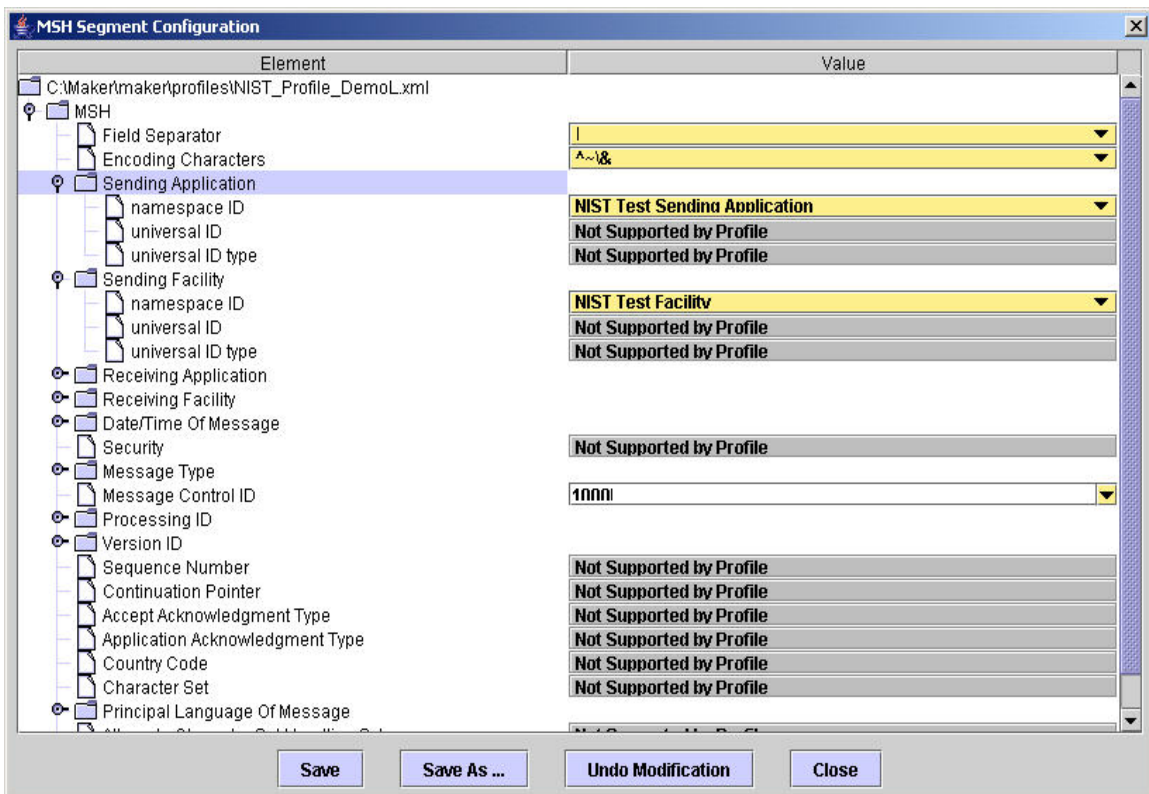
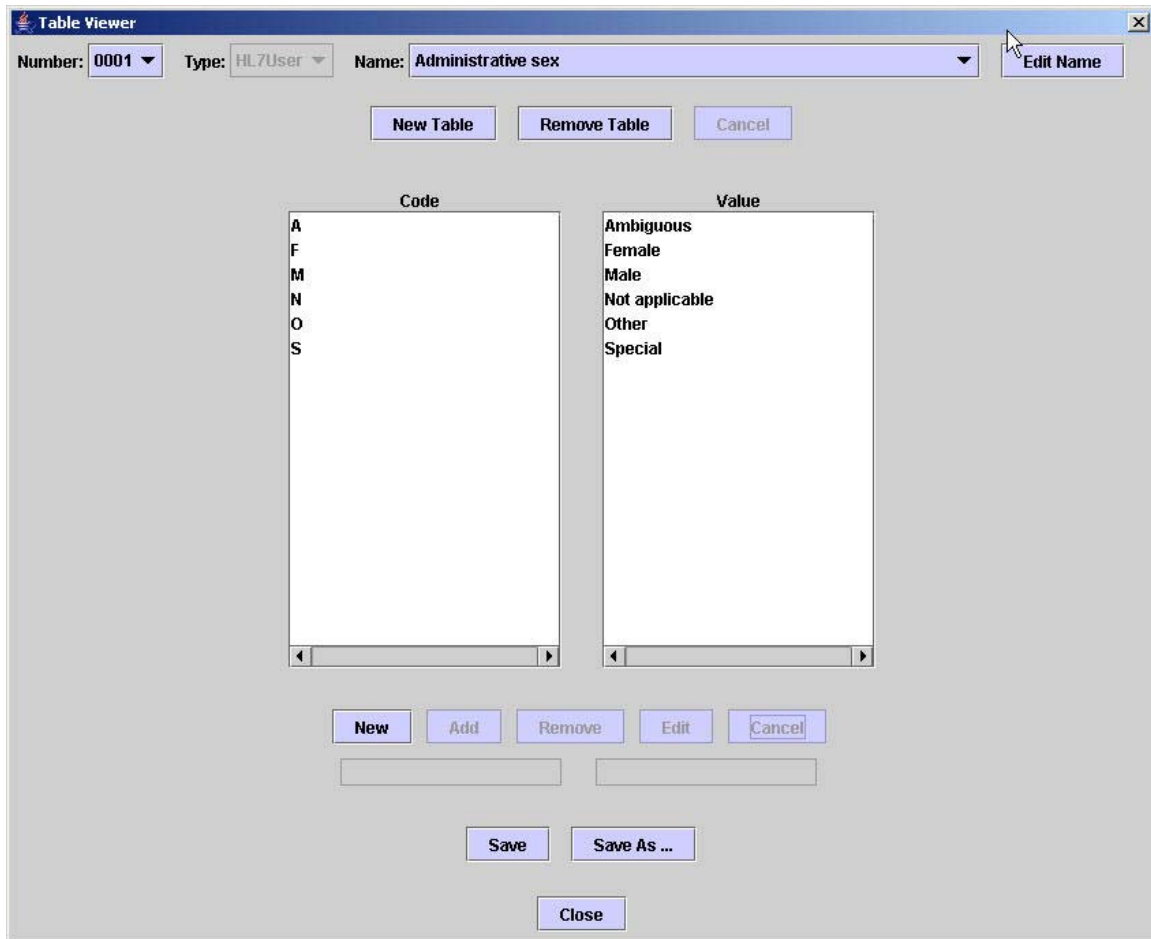


Figure 6. MSH Segment Initialization

Items set on this dialog are written to the ConfigData file and as such will be used as the default value in message creation. Change the “Message Control ID” (MSH.10) element to set the beginning sequence number for identifying messages.

### 4.3.3 Table Data Initialization

Tables can be modified, created, or removed using the “**Table Viewer**” dialog (See Figure 7). To create a new table select the “New” button and then provide the Name, Number, Type, and code/value pairs. To remove a table, select the table and click-on the “Remove” button.



**Figure 7. Table Viewer**

When a table is selected its code/value pairs are shown. Using the utilities provided the table code/value pairs could be modified. To modify an entry, select the code/value pair of interest. This will copy the entry to the text field provided below. Make the modifications and select the edit button to make the changes permanent. New items can be added to the table by using the “New” and “Add” buttons. Select “New”, enter the code/value pair in the text fields, and select “Add” to include the item into the table. To remove an item, select an entry and click-on the “Remove” button.

### 4.3.4 Data Configuration Panel

The Data Configuration panel is a general-purpose utility that allows the users to modify the data elements in the database, including MSH segment and table data. The Data Configuration panel allows the user to set various site-specific values that they want to include in messages or allows them to provide customize variation of the data values for particular elements. The view that is presented is a reflection of the profile. All data elements can be modified including table values. The user may add, edit, or delete values from the database. Depending on the type of data value an appropriate data modification panel on the right side on the Data Configuration panel will appear. To view and/or modify a particular data element browse the tree structure and select the desired element.

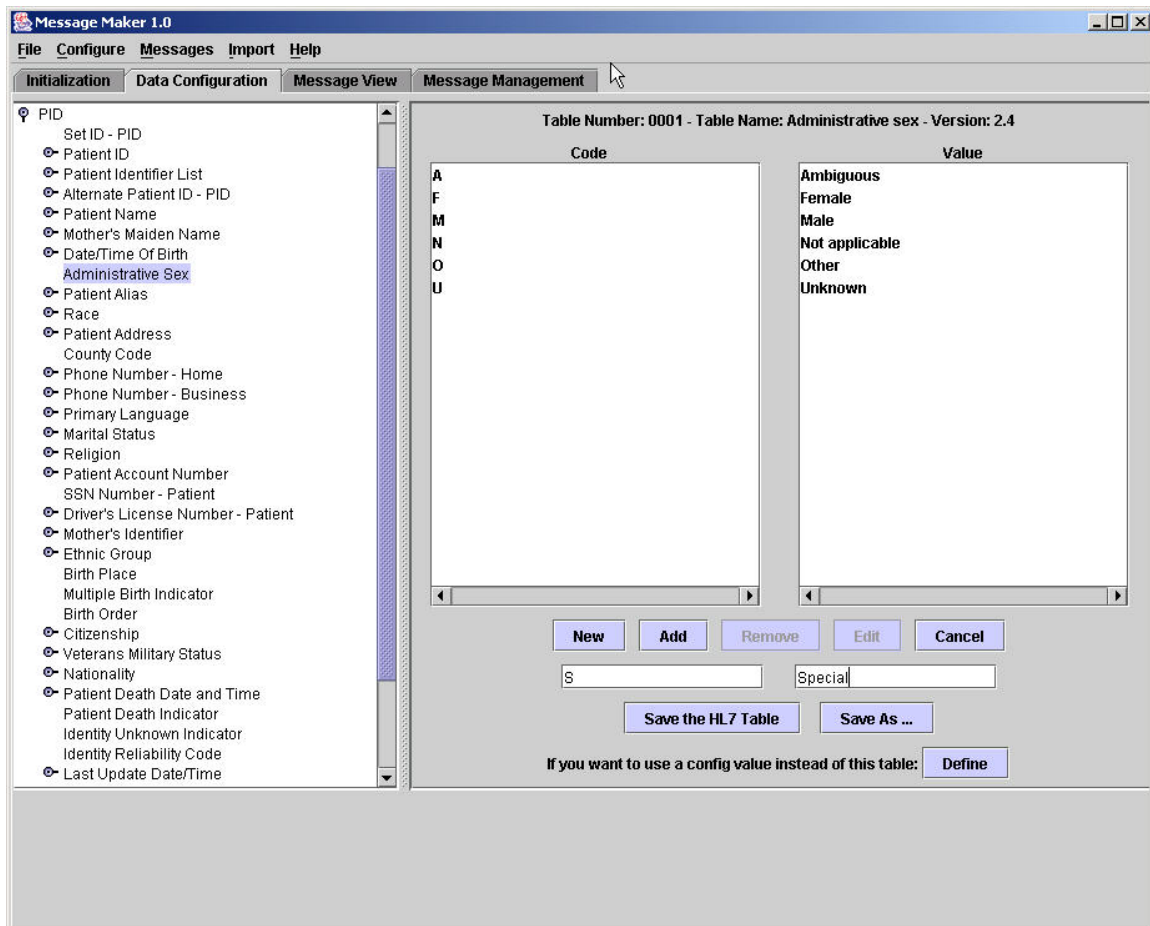
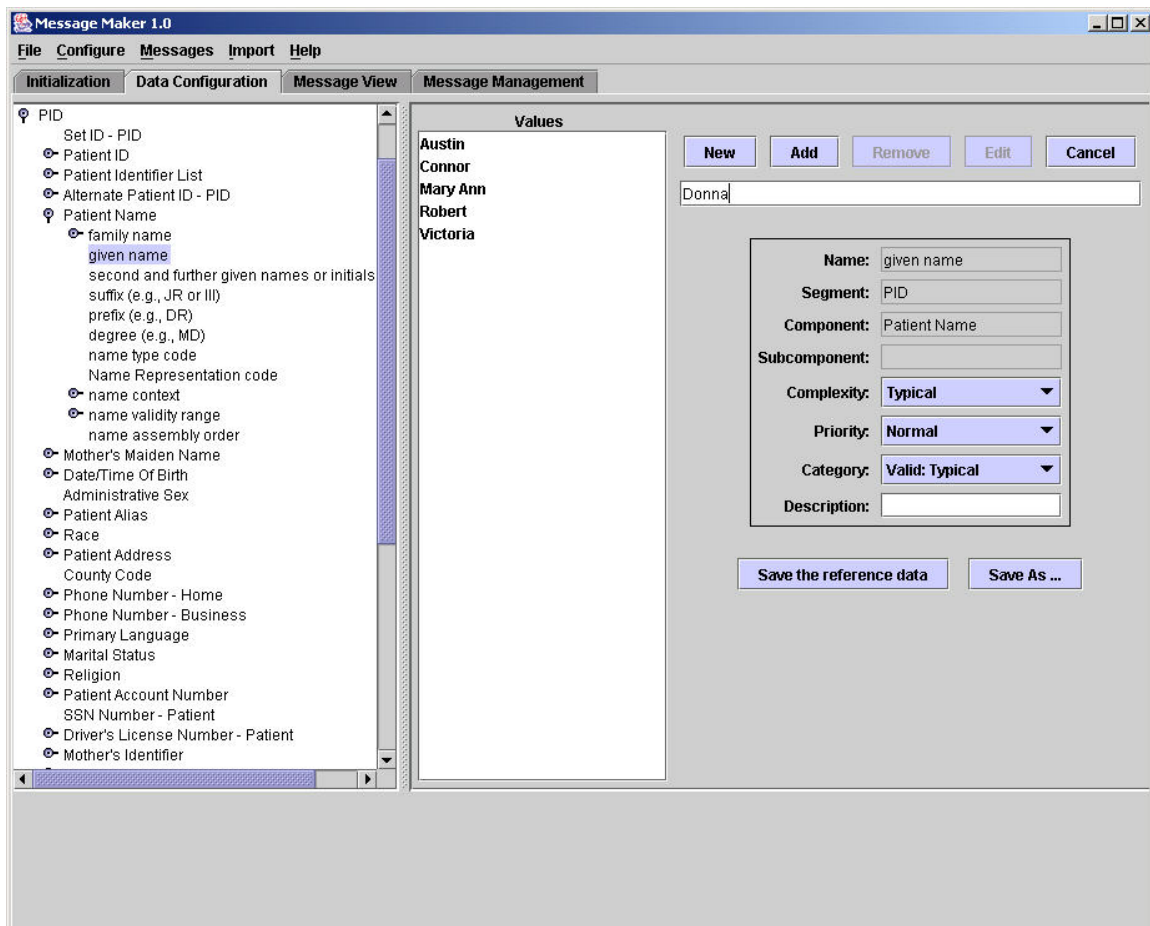


Figure 8. Data Configuration Panel (Table Data Modifications)

Figure 8 shows modification panel for table data (HL7 Table 0001 Administrative Sex). To add an entry into the table select the New and Add buttons. Select **New**, this activates the two text fields where the *Code* and *Value* pair can be entered. Select the **Add** button to commit the values to the table. To delete an entry from the table, select (highlight) *Code* and *Value* pair and select the **Remove** button. To modify an existing entry select (highlight) the *Code* and *Value* pair. This entry will now appear in the *Code* and *Value*

pair text fields below. Perform the modifications and select the **Edit** button to commit the changes to the table.

Data values for primitive elements can also be modified. Figure 9 depicts such a panel. Again use the tree structure to navigate to the element you want to modify. In this case we selected the *PID* segment, the *Patient Name* field and the component *given name*. The component *given name* is highlighted. On the right side of the panel the values in the database for this element are given. You can add, modify, or delete values from this list. Also, for each value you can attributes (e.g., Complexity). Currently, Message Maker does not yet make use of these attributes; hence changing the attribute values will have no affect on message construction.

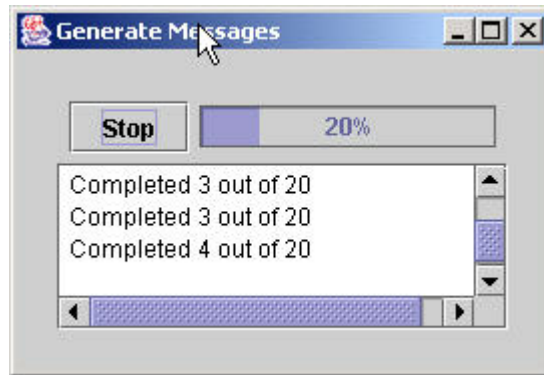


**Figure 9. Data Configuration Panel (Reference Data Modification)**

To add an entry into the element list select the **New** and **Add** buttons. Select the **New** button; this activates the text field where a *Value* can be entered. Select the **Add** button to commit the value to the list. To delete an entry from the list, select (highlight) a *Value* and click-on the **Remove** button. To modify an existing entry select (highlight) the *Value*. This entry will now appear in the *Value* text field. Perform the modifications and select the **Edit** button to commit the changes to the list.

## 5 Message Generation

Once initialization is complete messages can be generated. The minimum initialization requirement is providing a valid message profile. To start the message generator select the “*Generate...*” item under the “*Messages*” menu. This will launch the Generate Messages Dialog, See Figure 10. Click-on the **Start** button to begin the process.



**Figure 10. Generate Message Dialog**

Upon completion messages are saved in the “Message Path” directory specified during initialization. The messages are stored as XML files. Internally, Message Maker processes messages as XML representations. Messages can be saved in XML or ER7 format. To save messages select the Save item from the Messages menu. If you wish to subsequently edit messages, the XML version must be saved. Messages that are not explicitly saved will be removed when Message Maker exits. Messages can be deleted by selecting the Delete item from the Messages menu.



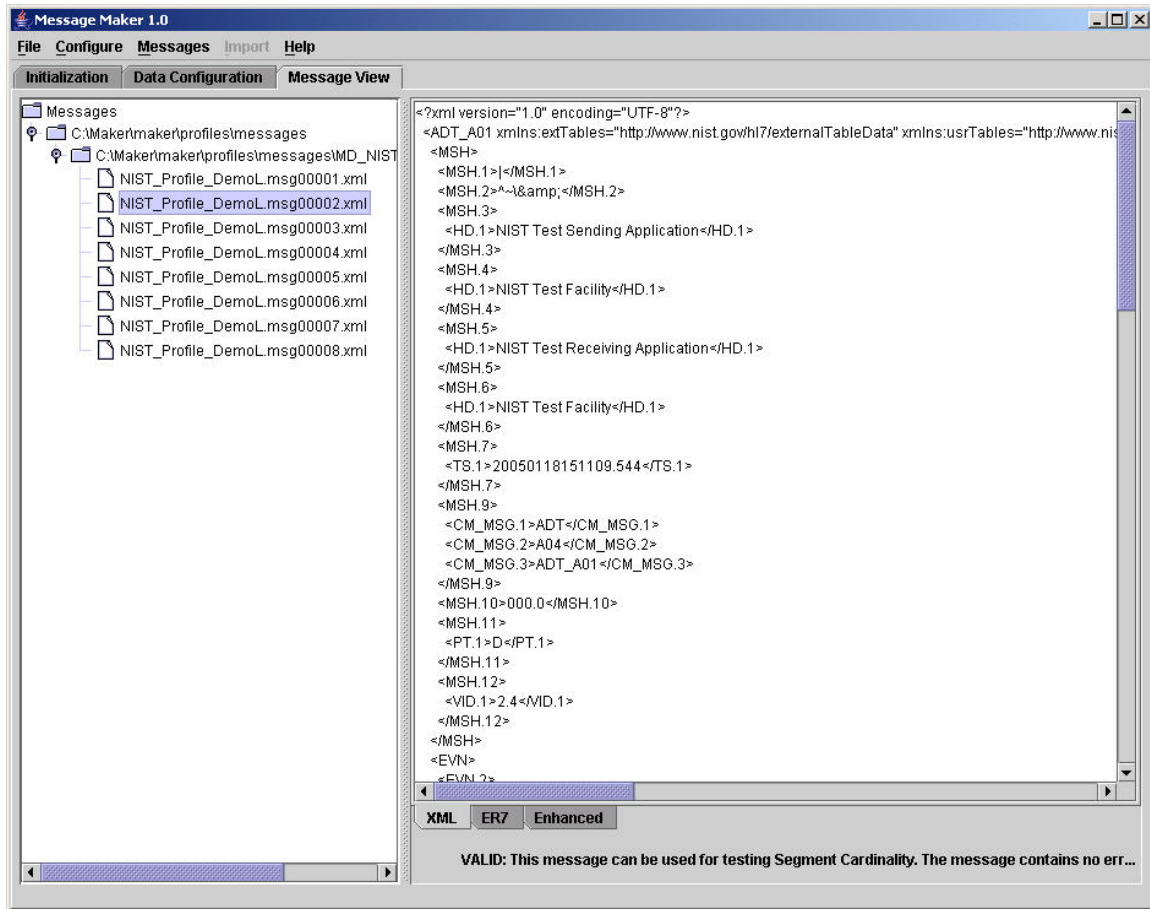


Figure 11. Message View Panel (XML View)

## 6 Message Browsing and Editing

Select the Message View tab to review and edit the messages. The Message Path directory selected during initialization is automatically mounted. Open this folder and select the metadata file associated with your message test suite. By default the name of this file is the name of the profile with a “MD\_” prefix. Selecting this file gives a summary of the messages in the display panel on the right.

Opening the metadata folder displays the list of messages. By default the name of the message files are the profile name with a “.msg00000.xml”, where the “00000” represents the message sequence number. By default it starts as 0. Select a message to view its contents. Messages can be rendered in XML, ER7, or Enhanced ER7 formats. By default messages are displayed in XML (Figure 11). Select the ER7 tab at the bottom to display the message in the ER7 format. Select Enhanced ER7 to view and navigate messages in a tree-structure. The Enhanced ER7 view also allows the user to edit the content of the data values in the message. To change the value of a data element, select the value, make the changes, and hit enter (Figure 12).



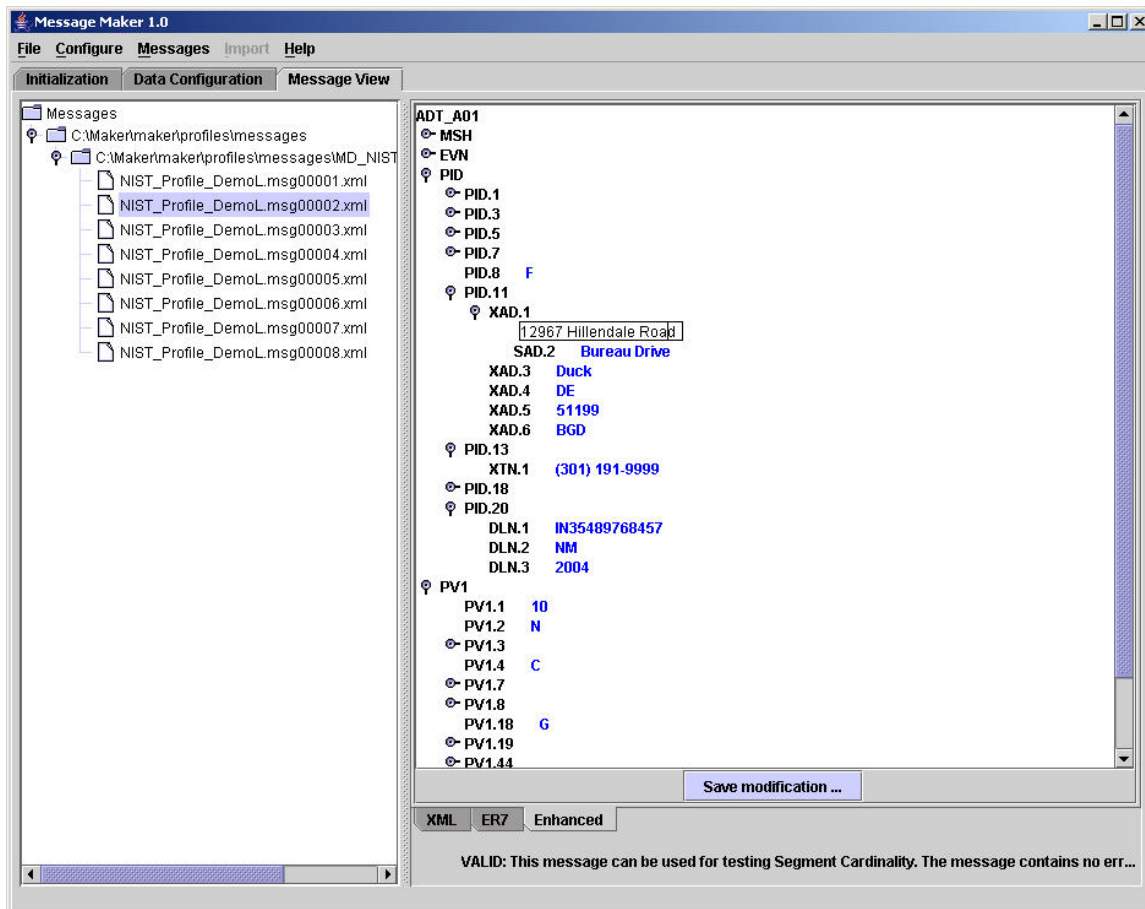


Figure 12. Message View Panel (Enhanced ER7 View-Edit)

Modifications made to messages can be saved. Messages saved with the same name will still be associated with the originating metadata file. Messages saved under a different name will not be associated with a metadata file.

## 6.1 Viewing Previously Generated Messages

Messages that were previously generated can be view by mounting the directory where the messages were saved. Choose the Mount Directory dialog from the Messages menu.

## 7 Appendix A: Message Maker Version 1.0 Limitations

The following list gives the current known limitations of this version of Message Maker. Future versions of Message Maker will include this functionality.

1. The reference data is limited to version 2.4 elements.
2. Localization (Z-segments) is not handled. Although, the Message Generator should process the segments correctly, there is no support to populate their data elements them with values (i.e., values don't exist).
3. Only ADT messages are supported. Although, any message type can be process, the reference data is populated mostly for ADT data elements.

4. The reference data is not yet fully populated. It is partially populated for ADT message types.
5. Message Maker is has limited facilities for importing site-specific data.
6. Message Maker has limited validation and reporting functionality. Messages can be invalid.

## 8 Appendix B: Test Options Descriptions

Usage	Valid	Meaning
Required	checked (Gray)	Required elements are populated.
Required or Empty	checked (Gray)	Elements are sometimes populated.
Conditional	not-checked (Gray)	This option is not yet available.
Conditional or Empty	not-checked (Gray)	This option is not yet available.
Not Supported	checked (Gray)	Not Supported elements are empty.

Usage	Invalid	Meaning
Required	checked	Some required elements are not populated.
Required	not-checked	Required elements will be populated.
Required or Empty	checked (Gray)	Elements are sometimes populated.
Conditional	not-checked (Gray)	This option is not yet available.
Conditional or Empty	not-checked (Gray)	This option is not yet available.
Not Supported	checked	Some Not Supported elements are populated.
Not Supported	not-checked	Not Supported elements are empty.

Cardinality	Valid	Meaning
Segment	checked	Messages will contain segment instances that are within the range defined by the cardinality attribute.
Segment	not-checked	Messages will contain the minimum number of segments.
Field	checked	Messages will contain field instances that are within the range defined by the cardinality attribute.
Field	not-checked	Message will contain the minimum number of fields.

Cardinality	Invalid	Meaning
Segment	checked (Gray)	Some messages will contain segment instances that are not within the range defined by the cardinality attribute.
Field	checked (Gray)	Some messages will contain field instances that are not within the range defined by the cardinality attribute.

Coded Entry	Validity	Meaning
	Valid	All coded entry elements will be populated with values from a valid value set.
	Invalid	Some coded entry elements will be populated with values not contained in a valid value set.

<b>Length</b>	<b>Validity</b>	<b>Meaning</b>
Composite	Ignore (Gray)	Length of data may exceed specified element length.
Field	Ignore (Gray)	Length of data may exceed specified element length.
Component	Ignore (Gray)	Length of data may exceed specified element length.
Subcomponent	Ignore (Gray)	Length of data may exceed specified element length.
Composite	Valid-checked	Length of composite data items adheres to the specified element length.
Field	Valid-checked	Length of data adheres to the length specified for the primitive field element.
Component	Valid-checked	Length of data adheres to the length specified for the primitive field element.
Subcomponent	Valid-checked	Length of data adheres to the length specified for the primitive subcomponent element.
Composite	Valid-not checked	Length of data may exceed specified element length.
Field	Valid-not checked	Length of data may exceed specified element length.
Component	Valid-not checked	Length of data may exceed specified element length.
Subcomponent	Valid-not checked	Length of data may exceed specified element length.