HL7Kit2018

User's Manual, version 3.0

HL7 KIT IS AN INTEGRATION ENGINE SPECIALLY DESIGNED
FOR BUSY HEALTHCARE IT PROFESIONALS.

You are 20 minutes away from HL7 integration!

H.R.Z. Software Services LTD
11/25/2018

Copyright © H.R.Z. Software Services 2013-2018
Copyright © RZ Software Services 2008 - 2013

H.R.Z. Software Services LTD
6 David Yellin St.
Tel-Aviv, Israel

www.hl7kit.com
info@roniza.com

HL7Kit User's Manual

Overview 3
Package Content 3
Features 4
Installation Instructions 5
System Requirements 5
Installing HL7Kit Pro 5
Activation 6
Evaluation 8
Applications 9
HL7Sender 9
HL7Receiver 15
HL7Mapper 19
DICOM Mapper application (processing DICOM files) 43
Worklist Manager 46
HL7Runtime Configuration 49
Maintenance 68
Monitoring and Logging 68
HL7 Version Configuration 68
Message Rules 68
Creating new configurations 70
Examples 70
Annexes 71
Concepts and Vocabulary 71
Upgrade Instructions 72

H.R.Z. Software Services LTD
6 David Yellin St.
Tel-Aviv, Israel

www.hl7kit.com
info@roniza.com

Overview

HL7KitPro is an interface engine and interface design tool for HL7 and DICOM integration. Use HL7KitPro HL7 Mapper to define and implement interfaces. HL7Kit requires no programming. The mapping definition is done by drag-and-drop of message attributes in the HL7 Mapper Application.

The kit comes with pre-configured Service Order to DICOM Modality Worklist mapping and HL7 Results mapping.

Package Content

HL7Kit Pro includes six applications:

Application	Description
HL7 Service	The HL7 runtime does the actual work of message processing according to the definitions made using the HL7 Mapper.
DICOM Server	The DICOM Server is integrated with the HL7 Server and uses the same database. It is a Modality Worklist SCP, Storage SCP and Q/R SCP.
HL7 Runtime Control Panel	The HL7 runtime control panel is used to configure the HL7 and the DICOM Services
HL7 Sender	HL7 message editor and validation tool and TCP/IP network message sender
HL7 Receiver	HL7 TCP/IP listener and Message processor.
HL7 Mapper	The main design tool for HL7 message processing. It is used to define incoming and outgoing message processing definitions.
Worklist Manager	A simple Worklist manager that can be used for searching patients/procedures stored in DB and scheduling studies for the Modality Worklist SCP
Dicom Mapper	A console application to extract info from DICOM files stored by DICOM server and insert it into HL7 DataBase

Features

Feature	HL7Kit Pro
Edit HL7 Messages	Use HL7Sender to view, verify and edit HL7 messages
Convert HL7 to XML/ V2.XML	Save HL7 message as XML using HL7Sender. Convert incoming HL7 messages to XML/ V2.XML and pass it to another application using HL7Receiver
Send HL7 Messages over TCP/IP	Automatically send HL7 messages whenever data is created or modified in your application database. Send messages manually using the HL7Sender Utility.
Configurable Protocol Definition	HL7 protocol versions are dynamic. Easily modify and test site-specific HL7 message structures and variants.
Receive Messages over TCP/IP	Messages received via TCP/IP are automatically processed by the HL7 Service operates in the background and populate your application database.
Retry Mechanism	Number of retries and retry frequency for outgoing messages that failed to send due to network problems.
Launch custom application	HL7Receiver can activate your custom application with the HL7 message as a parameter, either in HL7 or XML format.
Audit Trails, Logs and Monitoring	Logging events at the message level, optionally with complete message content. Set your own event triggers in the system event viewer to monitor critical events using email or text messages.
Database Integration	Incoming HL7 messages are processed and the data is inserted directly to the database tables of your application. Outgoing HL7 messages are sent when your application inserts data into the database tables.
Dynamic Rules Definition	The mapping between your application databases and the HL7 messages is done using a drag and drop mapping application.
GUI Interface Definition Tool	HL7 Mapper helps you design your interfaces and define incoming and outgoing message processing rules.
Hot Folder Integration	HL7Kit can send and receive messages via TCP/IP network or from/to pre-defined folders.
Multiple destinations	One HL7 backend service can interface with many HL7 applications using
Multiple HL7 Protocol Versions	One HL7 backend service can use different HL7 protocol for different HL7 application. For example it can talk to one messages application using version 2.2 and to another using version 2.3
Multiple HL7 Mapping Rules	Every remote application can have different rules so you can have different database tables for different interfaces.
HL7 Queries and Results processing	Send HL7 Query messages (QRY^*** event types) and process query results.
Multiple Databases	We support MSSQL, MySQL and SQLite
DICOM Integration	Option to parse incoming DICOM files and store info in the database tables of your application.

H.R.Z. Software Services LTD
6 David Yellin St.
Tel-Aviv, Israel

www.hl7kit.com
info@roniza.com

Installation Instructions

System Requirements

Operating Systems:

- Windows 2008 Server R2 or later is recommended.
- HL7Kit is tested on Windows 2008 Server R2 (x64).
- HL7Kit is compatible with windows workstation and server operating systems, 32 and 64 bit editions.
- The kit requires .NET Framework 4.0 or later.

Database Integration:

- Built-in Database integration
- Microsoft SQL Server 2008 or newer, Including express editions
- MySQL
- SQLite

Installing HL7Kit Pro

If previous version of HL7Kit was installed on your computer - see UpgradeInstruction.pdf document explaining how to backup HL7 configuration and restore it after installation.

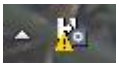
Double Click Setup.exe and follow the instructions.

After installation is complete, SQLite built-in Data Base is set as current DB.

Service Configuration Panel will be opened so you can change connection string manually and start services (see HL7 Runtime Configuration).

After installation, the HL7 Runtime Service is configured to start automatically with your system. You can cancel it by editing your system "Startup" settings.

To start the service without restarting, do one of the following:

1. Using the HL7 Service configuration tray icon 
    a. Double click the tray icon
    b. Click the start button at the top left.

    OR

2. Start the HL7Service from the services control panel
    a. From the start menu select run
    b. Type in services.msc
    c. Find HL7 Runtime Service

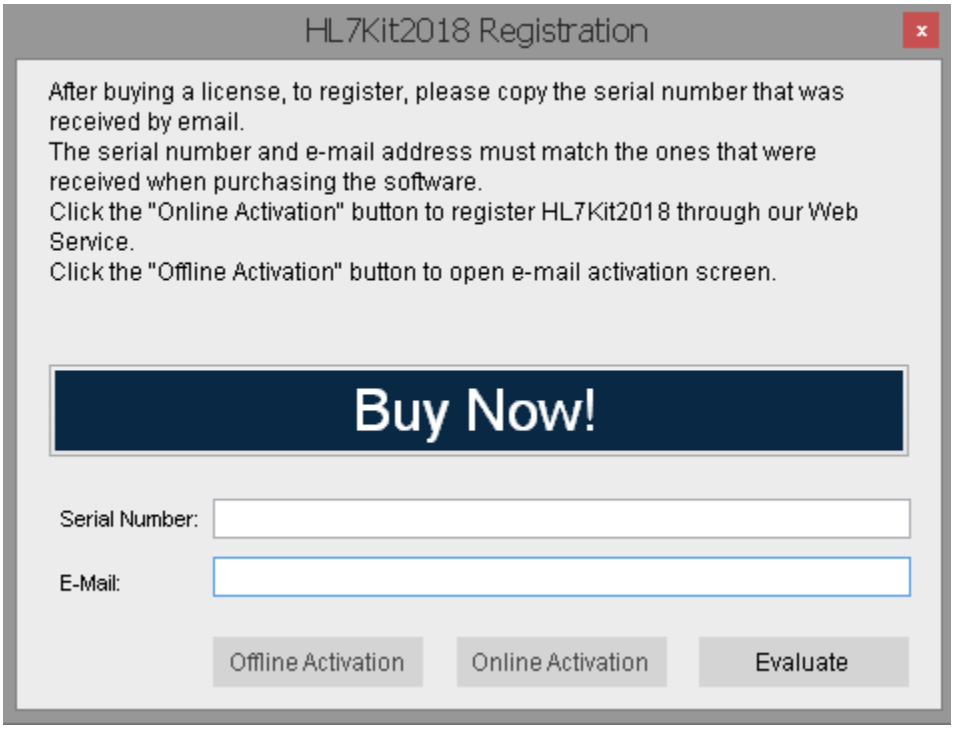
d. Start it

Or

3. Start HL7Service from command line
   a. From the start menu select run
   b. Type cmd
   c. In the command window type:
      net start HL7Service

Activation

Activating the product removes the evaluation copy limitations. A registration form is displayed when the HL7Mapping/ Recevier application or the HL7 Runtime Control Panel are started.

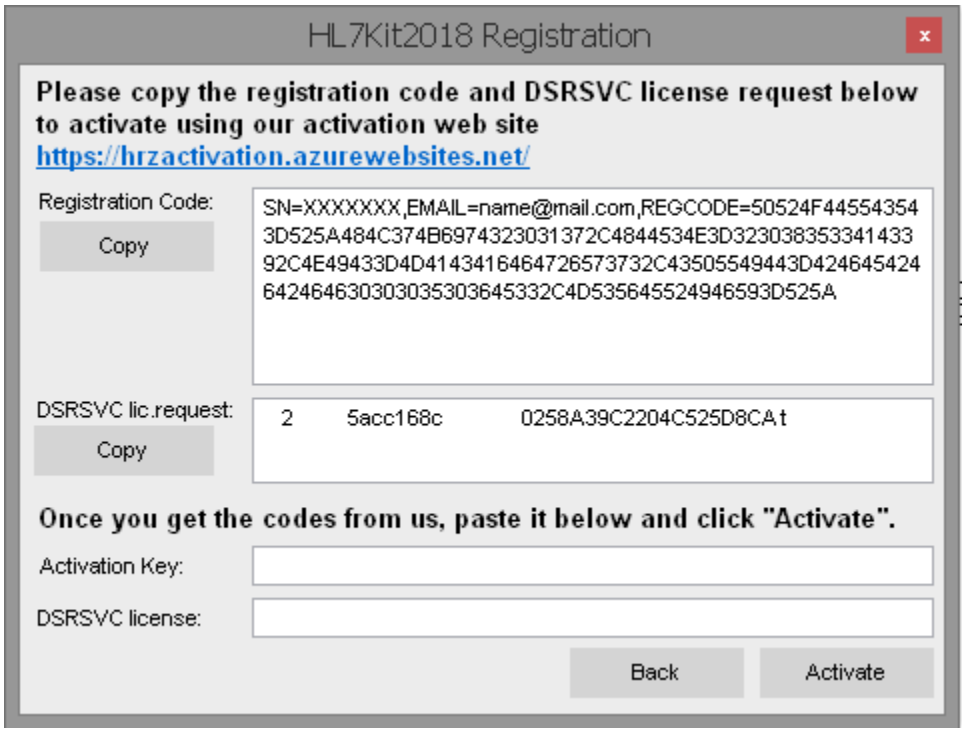


To activate your product online:

1. Key in the serial number (or transaction id) and e-mail address from your HL7Kit license purchase receipt.
2. Click the "Online Activation" button. If your purchase was found in our system - your license will be saved on your PC.

To activate your product offline:

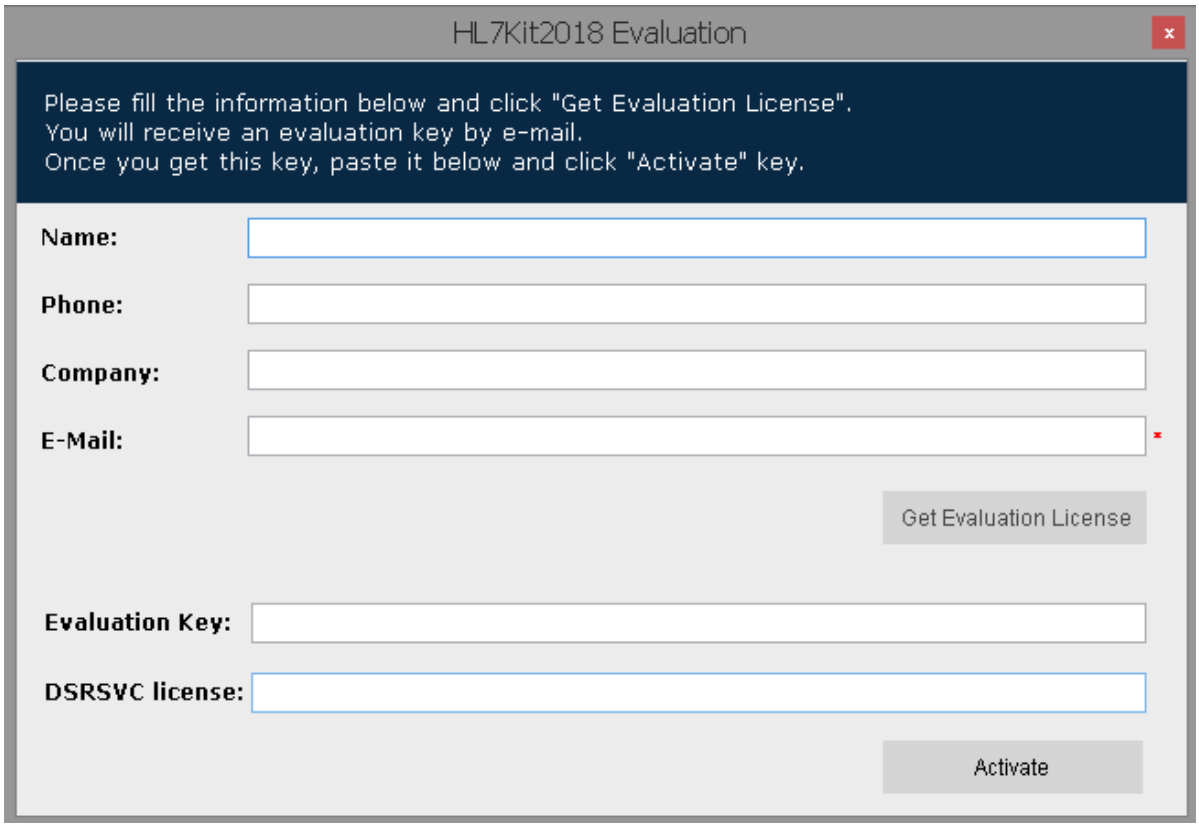
1. Key in the serial number (or transaction id) and e-mail address from your HL7Kit license purchase receipt and click the "Offline Activation" button:



2. Navigate to https://hrzactivation.azurewebsites.net/ and enter there your serial number and email to login to our activation system.
3. Paste the copied registration code and DSRSVC license request into the corresponding fields and get HL7 activation code and DSRSVC license content.
4. Copy activation key and license and paste them into the Activation key/DSRSVC license text boxes.
5. Click Activate.

Evaluation

To evaluate HL7Kit Pro, click the Evaluate button in the registration form, next screen will be opened:



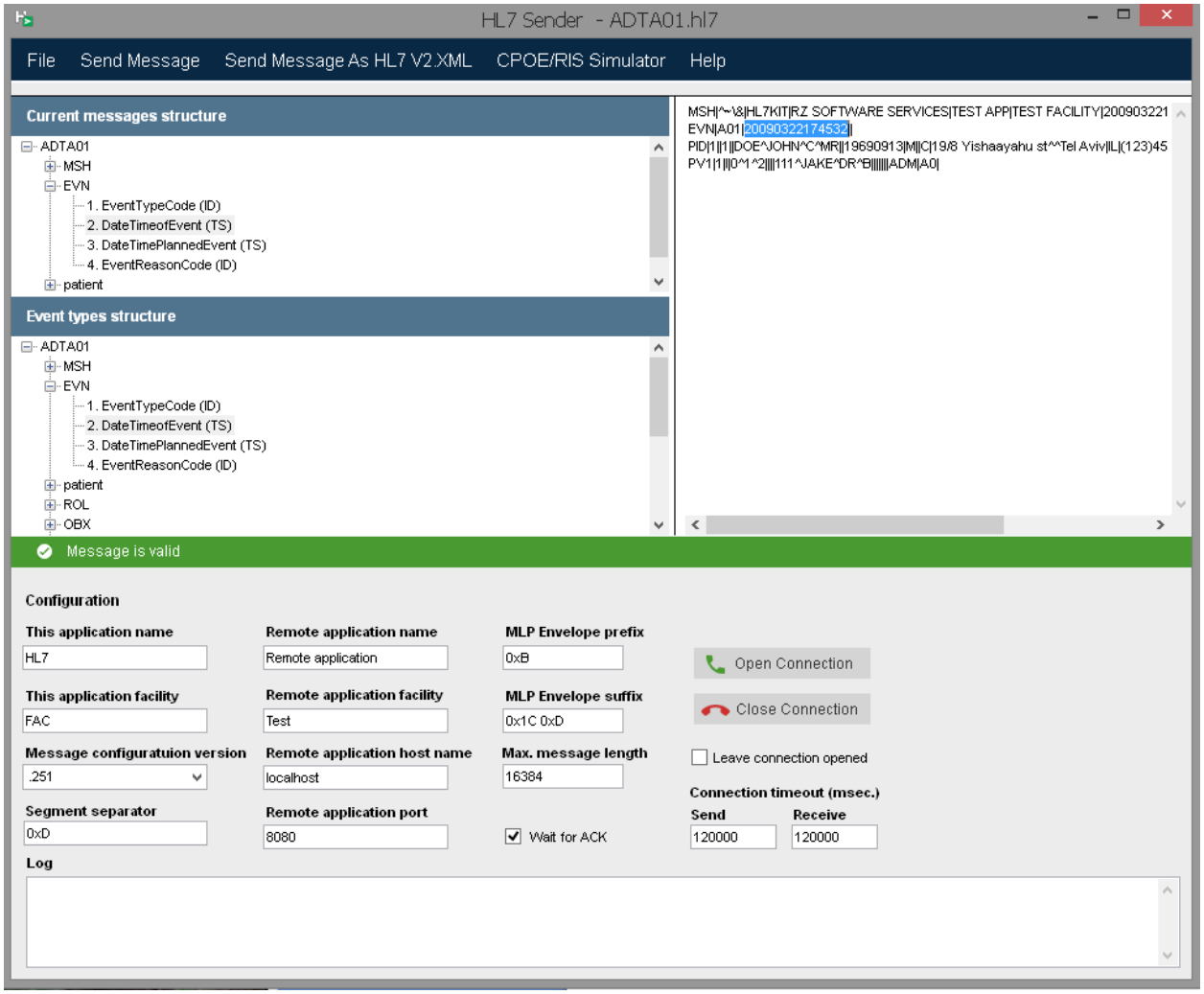
Enter your data (email is required, other fields are optional) and click "Get evaluation license" button.

You will receive email containing HL7 Kit activation key and DICOM server license for one month like:

Dear Customer,
Thank You for evaluating our product.
Please download the latest version of HL7Kit from the following link .
Use this HL7 evaluation key in the activate trial screen:
Evaluation key: **2A25FD77208B:-8586704047463275268:A**
and DICOM Server license file content:
DSRSVC license: **2 DSRSVCEvaluation e6b58beedb7a02ebf2d873d489aa97c7 3 Y 1d417e0f1f40000**
Then click the Activate button.
Thank You,
H.R.Z. Software Services LTD Team

Applications

HL7Sender



HL7Sender is a utility application for sending HL7 messages through the network. It is a single form application that can read, write, validate, convert to XML and send HL7 messages.
The upper right parts of the form shows the message in a rich text control where it can be edited. Every HL7 segment starts in a new line. The best way to edit the message is to cut and paste segments from other messages (you can open multiple instances of the program) or modify the value between two | signs (field separator).

The upper left area is divided into two panels showing the message content in as a tree view (up) and the message definition (down).

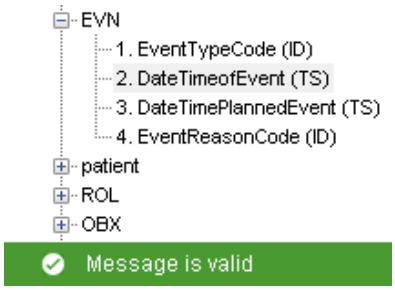
At the lower part of the form there are a few text fields where the user can set the LLP network protocol elements, the TCP/IP protocol attributes (destination host name or IP address, port number and timeouts) and disable or enable ACK's.
The two message structure text configuration files (segDefs and msgRules) are located in the installation folder and can be edited manually if necessary. You can duplicate these files with different suffixes in order to use multiple 'flavors' of the HL7 message structures.

Viewing Message Content

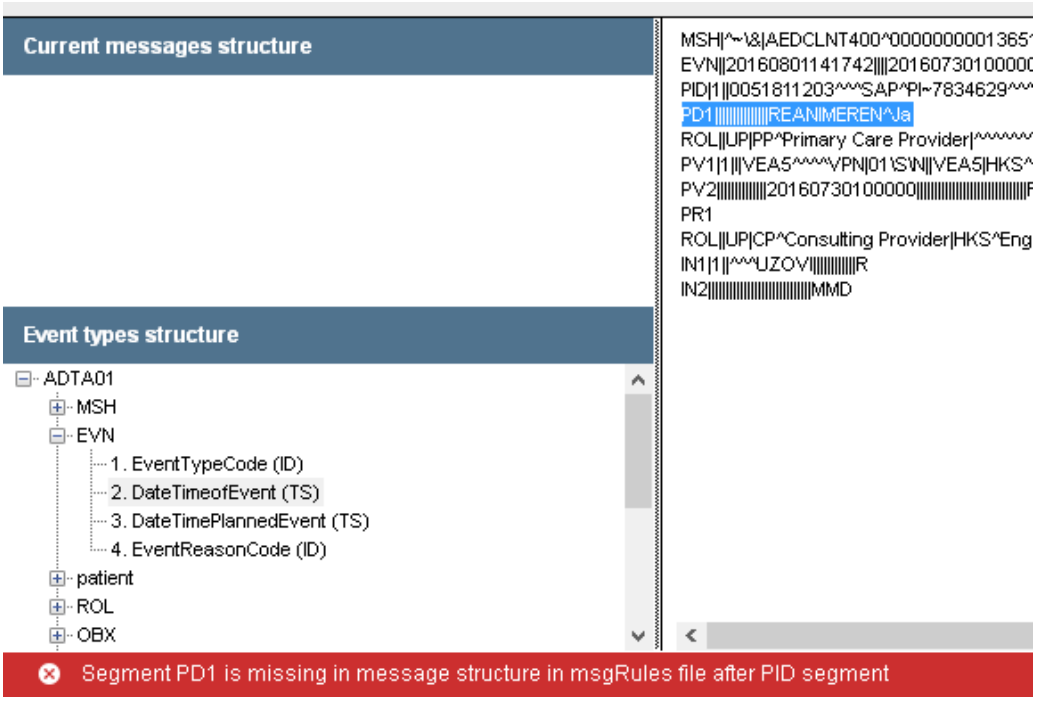
From the File menu select Open HL7 File and select a file to open.

The message content is shown on the right panel.

When the message structure is valid, a green message is shown on the left side of the screen.



If the message is not valid, a red message is shown on the left side of the screen describing the problem.



In the above example, a wrong segment name is present.

To fix that, edit the message content and remove the offending segment.

Sending Messages

Use HL7Sender to send messages stored in HL7 Files. Open a file and click the 'Send Message' menu button. You can send a message formatted as V2.XML by clicking "Send Message as V2.XML"

Editing Messages

Open a file and edit its content in the HL7 Message text box.
Use the message structure left panel to navigate through the message segments and fields.

Modifying Fields' Values

In the message structure left panel click on the field name.
The field text is highlighted in the message text panel.
Use the keyboard to change the field value.

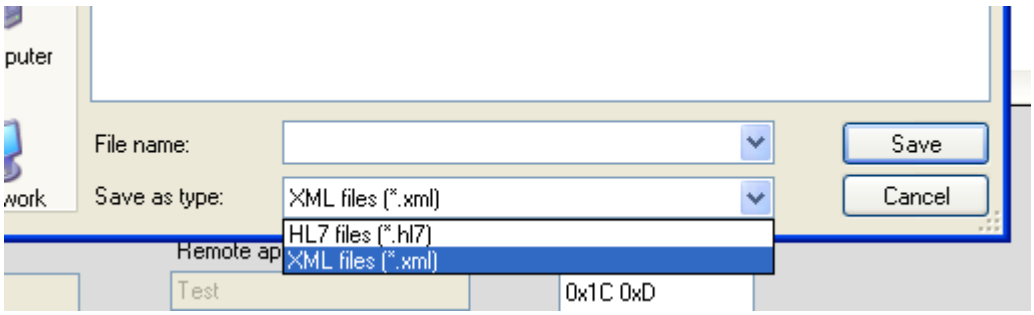
Adding Segments

In the message structure click on the segment you would like to insert a segment before.

Click <Home>.
Click <Enter>.
Key in the 3 letters segment name followed by the | sign.
The new segment is added to the message structure panel.

Convert HL7 to XML

From the File menu select Save As.
Choose XML from the "save as type" combo box.



Convert HL7 to V2.XML format

From the File menu select Save As HL7 V2.XML.

HL7 Sender main form

Hl7Sender Menus

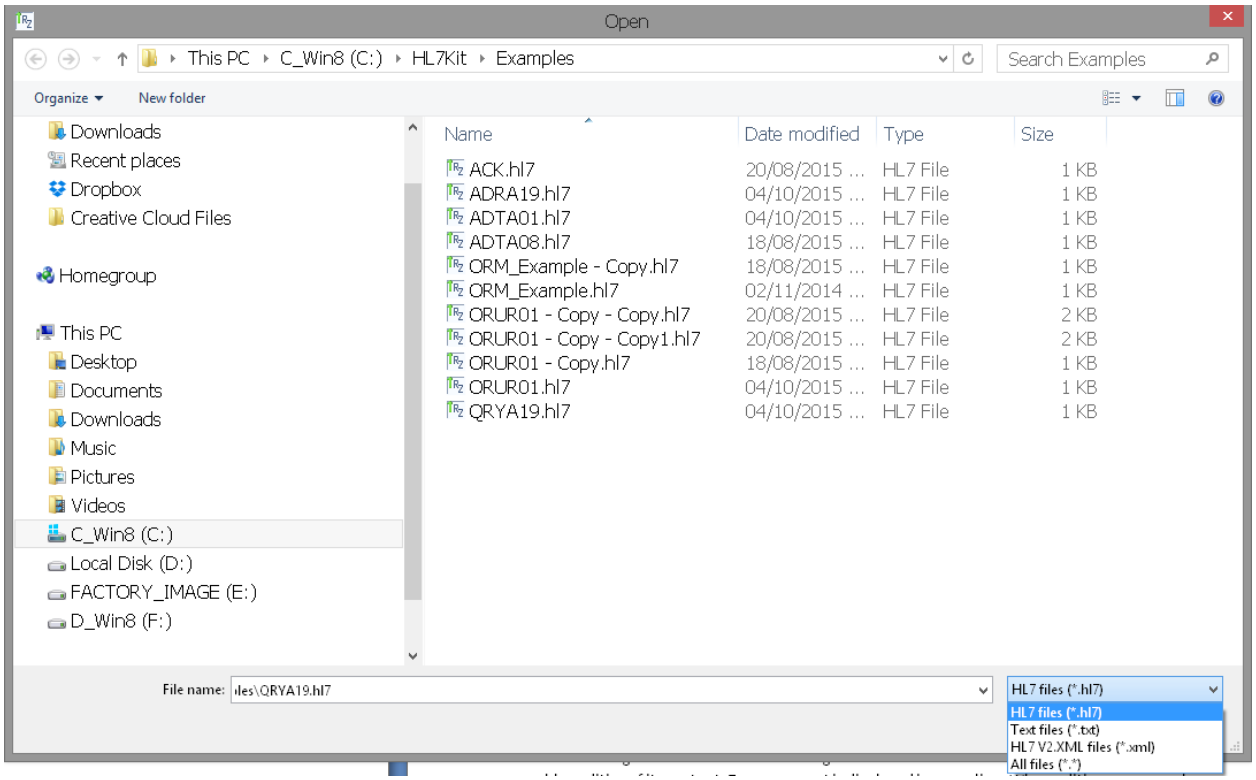
The File Menu

New

HL7 Message structure tree and HL7 Message text box will be cleared.

Open

Select the file type you want to open, browse and select an HL7 file. The file content will be loaded to the HL7 Message text box.



Save

Save current context of the HL7 Message text box to a HL7 file or XML file. If a file was previously opened, its content will be replaced. If you edit a new file you'll be prompted to select file type (HL7 or XML) and destination place.

Save As

Save current context of the HL7 Message text box to a new HL7 file or XML file. You'll be prompted to select file type (HL7 or XML) and destination place.

Save As HL7 V2.XML

Save current context of the HL7 Message text box to a new V2. XML file. You'll be prompted to select destination place.

The Send Message Menu Button

Clicking the Send Message menu button sends the content of the HL7 Message text box through the network to the remote application.

The Send Message As HL7 V2.XML Message Menu Button

Clicking the Send Message As Hl7 V2.XML menu button sends the content of the HL7 Message text box formatted as V2.XML message through the network to the remote application.

The Help Menu

About HL7 Sender

This item displays the about form.

HL7 Message Content Tree

The HL7 Message Content tree control displays the message structure as a tree. Each logical part of the message is represented as node holding a collection of segment nodes. Each segment node is a collection of fields. The node's names are the field name.

When clicking on a message structure node, the corresponding message text is selected in the HL7 Message text box.

HL7 Message Definition Tree

The HL7 Message Definition tree control displays the message definition structure according to the message definition rules. Use this display to compare the actual message content with the message definition.

HL7 Message text box

The HL7 Message text box is the main editing area of the HL7Sender. It displays the HL7 Message and enables editing of its content. Every segment is displayed in a new line. When editing a message, be careful not to break segments.

The Configuration Area

Message Configuration Version

Use this drop down to select the HL7 version to work with. See creating new configurations for explanation how to add HL7 versions to this drop down.

Segment Separator

Use this text box to set the characters combination (Hexadecimal notation) to use as segment separator. By default it's 0xD (LF).

This Application Name

Use this text box to set Sending Application field in MSH segment of message to send.

This Application Facility

Use this text box to set Sending Facility field in MSH segment of message to send.

Remote Application Host Name

Use this text box to set the host name or the IP address of the computer that you want to send the message to.

Remote Application Port

Use this text box to set the port number to the one the destination application listens on.

MLP Envelope Prefix

Use this text box to set the LLP prefix using Hexadecimal notation. See LLP for more information.

MLP Envelope Suffix

Use this text box to set the LLP suffix using Hexadecimal notation. See LLP for more information.

Connection Timeout

Send and receive network timeouts in milliseconds. When HL7 Sender performs network activities and there's no response from the peer for the specified period, the activity is stopped and an error message is displayed.

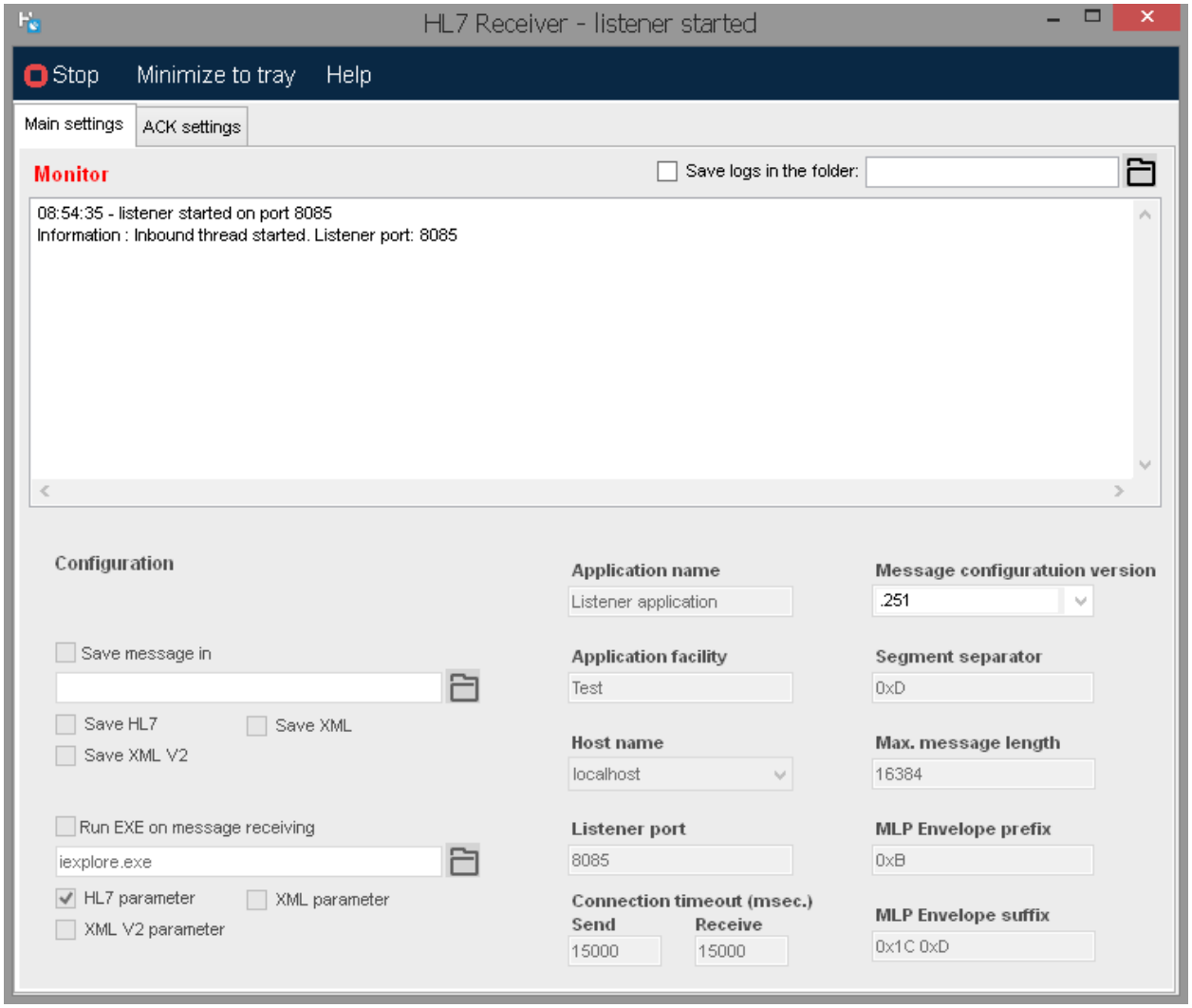
Open/Close Connection Buttons and Leave Connection Open Checkbox

The Open/Close connection buttons allow opening the TCP/IP connection without sending any message yet. When the "Leave Connection Open" checkbox is checked, the Send Message command will not close the connection after sending the message, and allow sending additional messages over the same connection.

Log

The log text area shows diagnosis messages relevant to the message sending and ACK processing. It also shows the response received when sending messages.

HL7Receiver



HL7 Receiver is a utility for receiving HL7 messages through the network (AKA HL7 Listener).
The receiver listens on a network port and waits for incoming connections. Every incoming message is validated and can be saved on the local hard disk. A configurable application or script can be launched after receiving a message. The messages' content can be saved in HL7 and/or in XML format.

HL7Receiver Main Form

HL7 Receiver Menu

Stop/Start

Click this button to change the HL7 network listener state. By default, the listener is active and messages are accepted. Toggle the Stop/Start button after configuration changes to apply the new configuration.

Event Log

HL7Receiver creates an event log that can be used for monitoring and trouble shooting. To view the event log:

1. From the start menu select Control Panels/Administrative Tools/Event Viewer
2. Click on the RZHL7Pro log.

Text Log File

Check the “Save logs in the folder” checkbox to save text log files in a specific folder.

The Configuration Area

To change a configuration you must stop the listener to enable confutation controls. After you finished editing, start the listener again; all changes will applied automatically and stored in the configuration file for next Receiver start.

Save message

You can save every incoming HL7 message in various formats in a destination of your choice.

When you click on "Save message" notice that you are automatically prompted to select destination folder. You can change this folder at any time by clicking "..." button.

When the Save HL7/Save XML/Save XML V2 checkbox is checked, HL7Receiver creates an Hl7/XML/V2.XML file for every valid inbound message. The file structure is determined according to the message rules definition in the configuration files.

Run EXE on message receiving

You can start some external process after receiving HL7 message passing incoming message as process input (message **must be saved** using "Save message" mechanism if you want to use this option).
For example you can start iexplore.exe opening message saved as XML.

To select process file (EXE or other type) click on the "..." button. Check HL7 parameter / XML parameter / XML V2 parameter to pass incoming message in correspondent format to the selected process **if message was saved in this format**. More than one parameter can be used.

Message Configuration Version

Use this drop down to select the HL7 version to work with. See creating new configurations for explanation how to add HL7 versions to this drop down.

Segment Separator

Use this text box to set the characters combination (Hexadecimal notation) to use as segment separator. it's 0xD (LF) by default.

Application Name

Use this text box to set the Sending Application field in MSH segment of ACK message.

Application Facility

Use this text box to set Sending Facility field in MSH segment of ACK message.

Host Name

Use this text box to set the host name or the IP address of the Listener.

Listener Port

Use this text box to set the port number of the Listener.

MLP Envelope Prefix

Use this text box to set the LLP prefix using Hexadecimal notation. See LLP for more information.

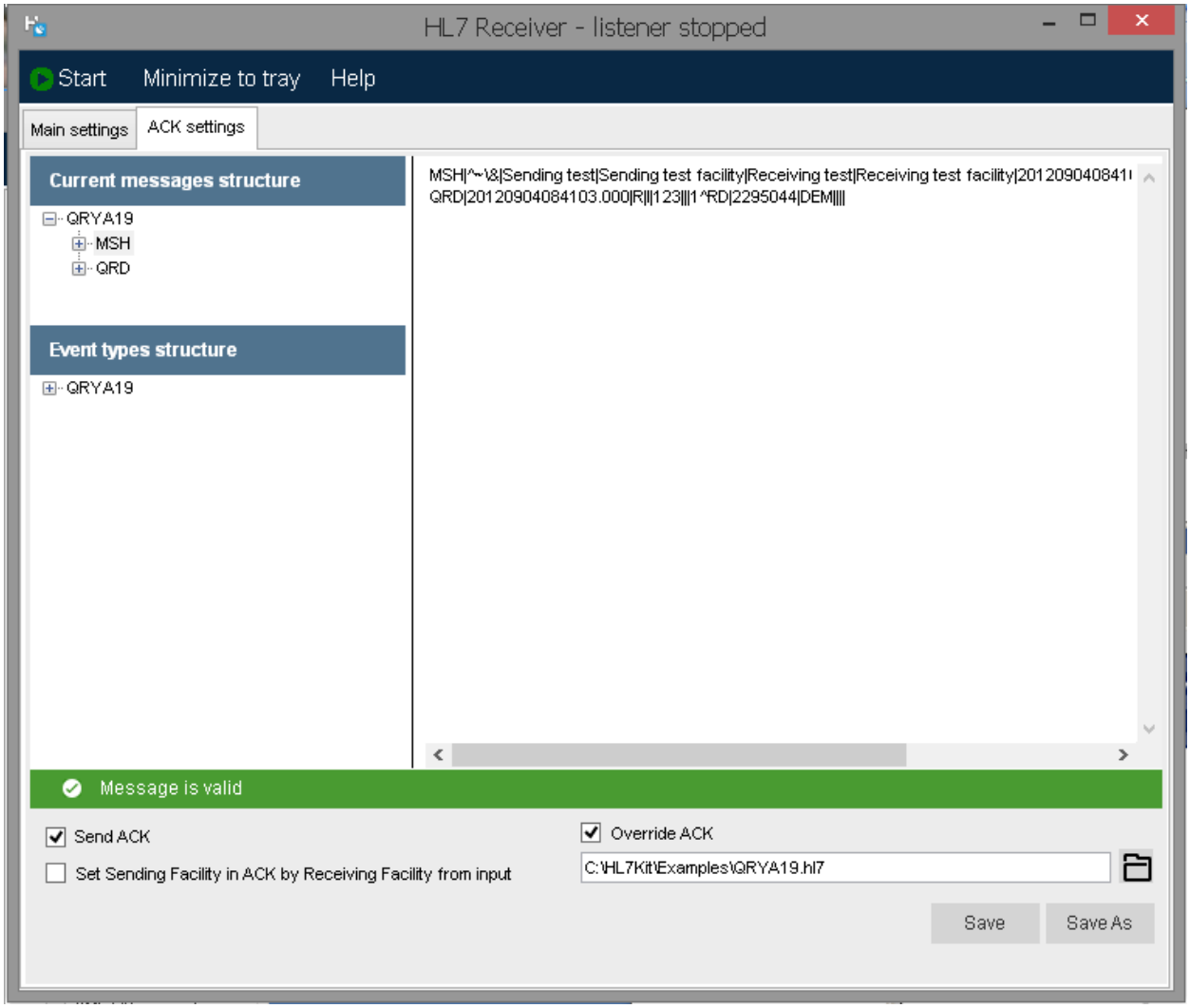
MLP Envelope Suffix

Use this text box to set the LLP suffix using Hexadecimal notation. See LLP for more information.

Connection Timeout

Send and receive network timeouts in milliseconds. When HL7 Receiver performs network activities and there's no response from the peer for the specified period, the activity is stopped and an error message is displayed.

ACK settings tab



Send ACK

Mark this checkbox if you want to send ACK message as response to every received HL7 message.

Override ACK

Use the Override ACK checkbox to reveal the Override ACK panel where a substitute message can be loaded and used instead of ACK message. This is useful for example for testing when you would like to control the response that the HL7Reciever sends. Any valid HL7 message can be loaded to the Override ACK panel.

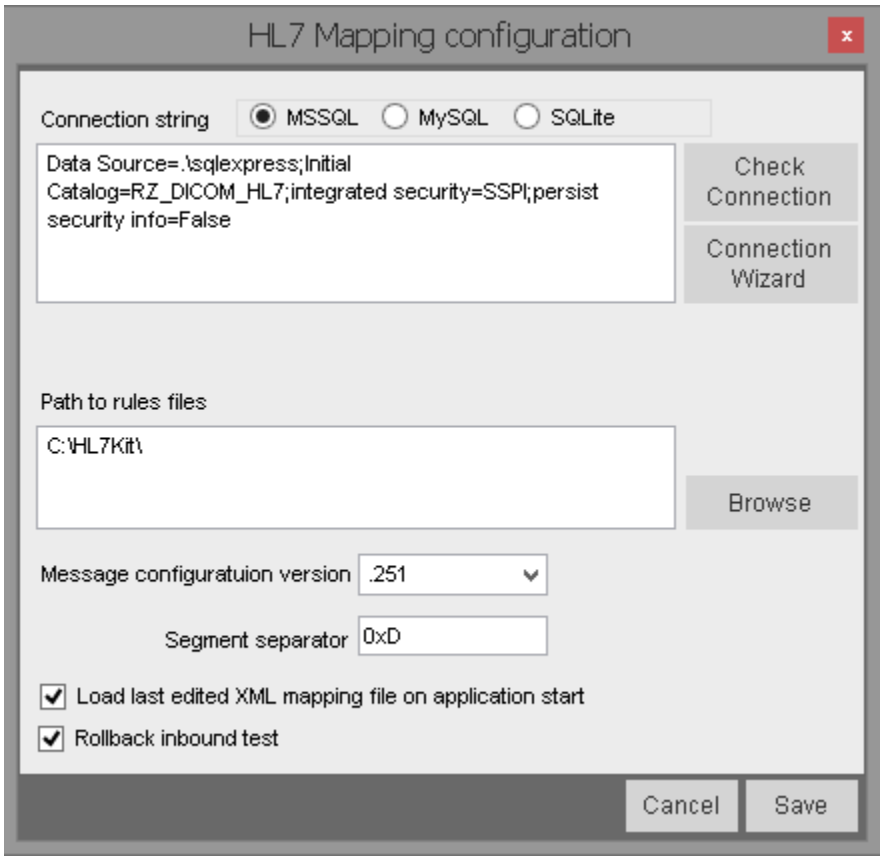
HL7Mapper

Overview

The HL7 Mapping Application is an authoring tool for the HL7Service mapping rules. The rules are stored in an XML file that is used by the HL7Service and defines how to process messages.

Configuration

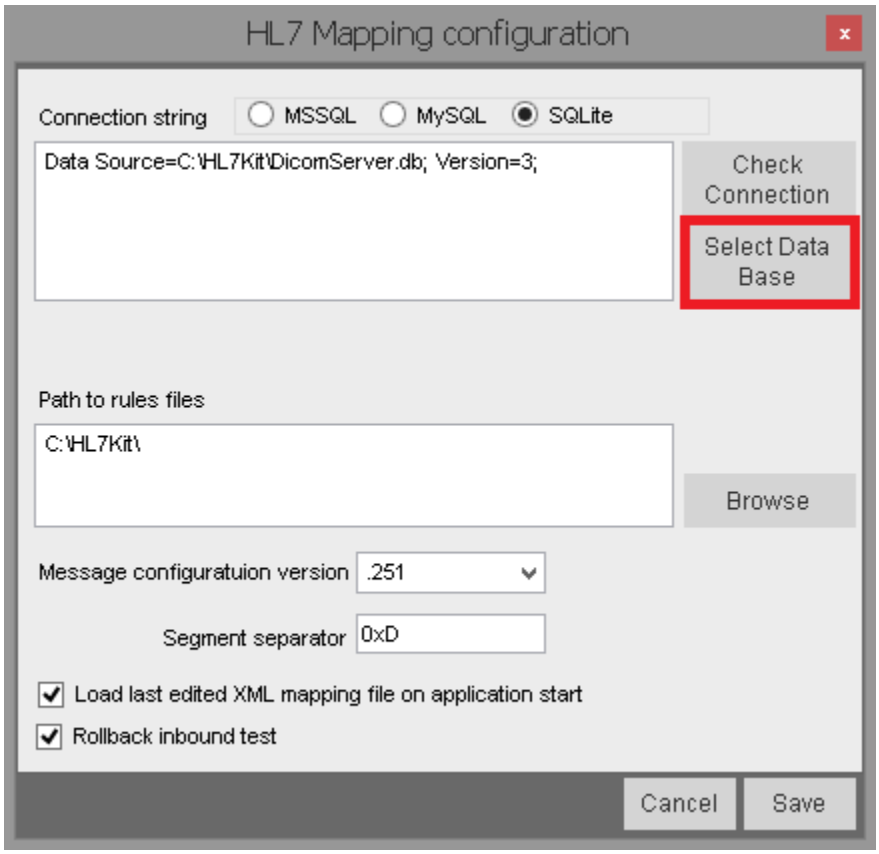
To open a configuration form click on "Configuration" menu item..



Connection to Data Base

Select MSSQL, MySQL or SQLite option and use the connection string text box to configure the database connection or click the connection wizard button to use the UML utility (for MSSQL server only).

If you choose SQLite option - by default built-in data base DicomServer.db will be selected. To set another SQLite DB file - click on "Select Data Base" button and select path to the file.



NOTE: If you are pasting the connection string from another source - don't forget to select appropriate server before. Connection string template will be displayed after selecting each option.

Creating default Data Bases in MSSQL and MySQL servers

There are few scripts allowing you to create default Data Base in MSSQL or MySQL servers manually:

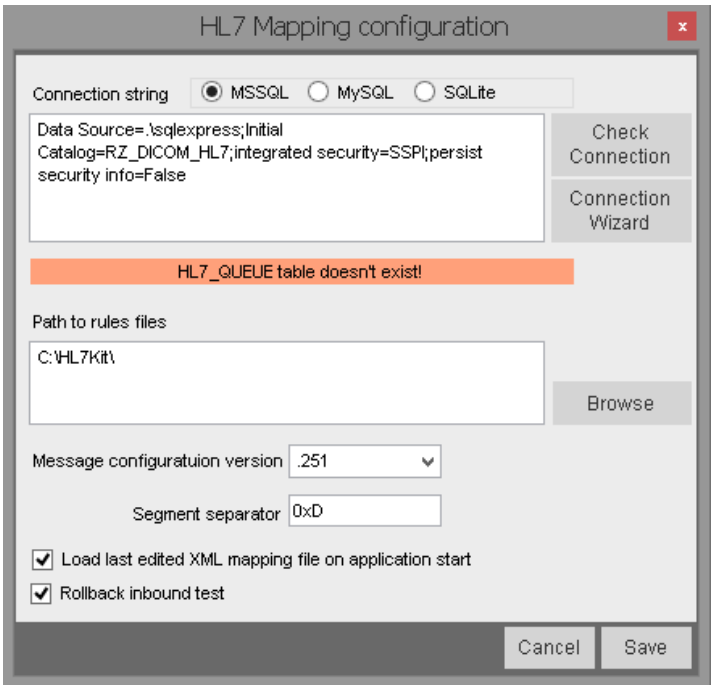
1. MSSQL - run installDB.bat located in the C:\HL7Kit\MSSQLScripts folder or run C:\HL7Kit\MSSQLScripts\ RZ_DICOM_HL7_CreateDB_Script.sql in SQL Server Management Studio
2. MySQL - run C:\HL7Kit\ MySQLScripts\ RZ_DICOM_HL7_CreateDB_Script.sql in MySQL Workbench

In both cases DB by name RZ_DICOM_HL7 will be created, use this name in the correspondent connection string

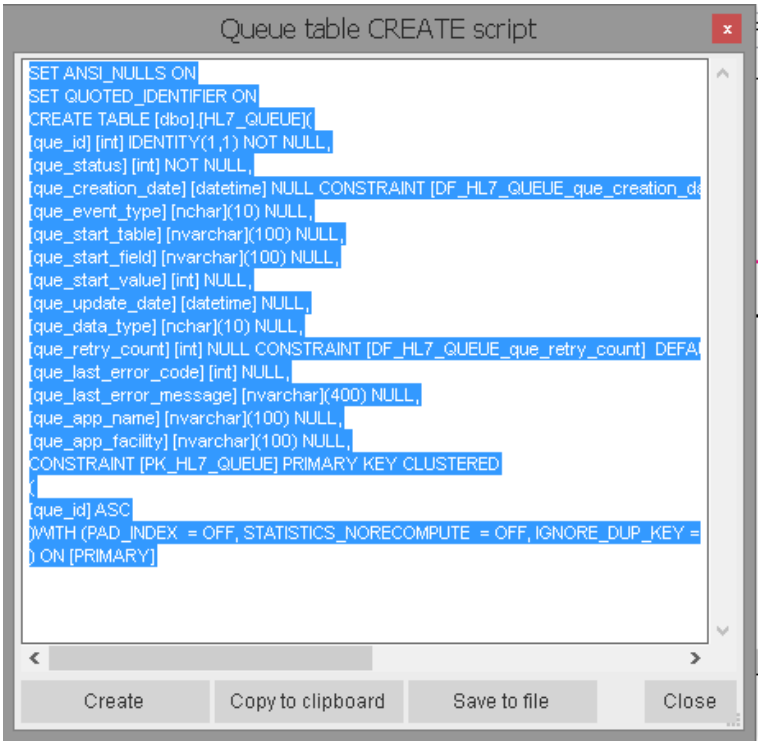
Checking queue table

Once the connection string is set properly, the application will attempt to find the HL7_QUEUE table that is used for sending outbound messages.

After configuring the database connection, create the HL7_QUEUE table if it doesn't exist already. Click the salmon colored ribbon with the label "HL7_QUEUE table doesn't exist!".



The following dialog will show where you can either click the 'Create' button the run the database script or copy the script and run it manually.



Connection String Text Box

The connection string text box shows the database connection string. It can be edited manually or using the connection wizard.

Check Connection Button

The check connection button is used to test the connection string.

Select Data Base Button

The select Data Base button is used to set the path to SQLite database file.

Connection Wizard

The connection wizard button displays a connection utility (UDL) that can be used to configure the database connection (for MSSQL server only).

Path to Rule Files Text Box

The path to rule files text box is used to edit the name of the folder in which the message definition files are stored. These files define the HL7 message parsing.

Browse Button

The browse button is used to set the path to rule files.

Message Configuration Version Combo Box

The message configuration Version Combo Box displays the message definition files suffix. There may be multiple sets of message definition files in the rules folder.

Segment Separator

Use this text box to set the characters combination (Hexadecimal notation) to use as segment separator. By default it's 0xD (LF).

Load Last Edited XML Mapping File …

If this check-box is set than the last edited file will be opened automatically when the application starts.

Rollback inbound text check-box

If this check-box is set, the data that is inserted when performing inbound tests will be rolled-back. Uncheck this check-box to leave the data in the database. This is useful when validating the mapping rules.

Cancel Button

The cancel button closes the configuration form without saving.

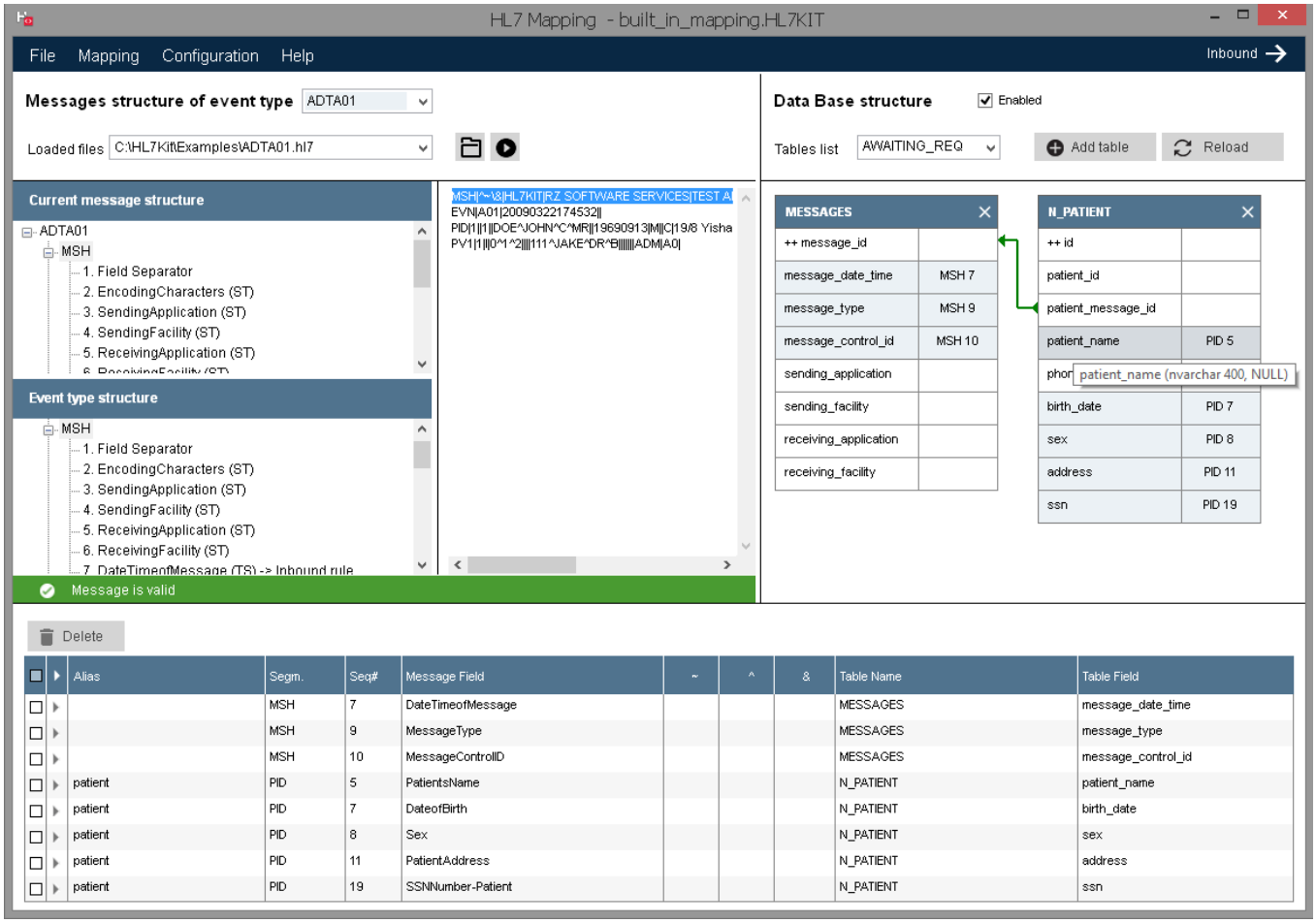
Save Button

The save button saves the configuration and then closes the form.

H.R.Z. Software Services LTD
6 David Yellin St.
Tel-Aviv, Israel

www.hl7kit.com
info@roniza.com

HL7Mapper Main Form



The HL7Mapper Main Form is divided into three areas. The left area displays the hl7 message structure, the right area displays database tables and the bottom area displays the mapping rules between the message and the tables.

Inbound Message Rules

Inbound messages are messages that are sent from an external system to the HL7Kit Pro Runtime Service.

Creating a simple Inbound Rules

1. In the main view, select the event type of the incoming message from the event-type combo-box or open some file containing HL7 message (click button to select file) or paste message text into its panel.
   The message structure is displayed on the left panel.
2. Select a table from the table list combo box and click the 'Add Table' button.
   The table structure is displayed on the right panel.

3. Click a field on the one of the trees in left area (message structure or event type structure) and drag it over a column name in the table.
4. Repeat step 3 for other fields.
5. Hover over a database column name to see its data type.
6. Hover over a HL7 Message field name to see its mapping target.
7. Continue mapping all required fields. Make sure all not-null columns are mapped.
8. When you're done mapping choose 'Inbound Test' from the mapping menu to select file containing HL7 message or click button if message text was loaded.
9. Select a HL7 message with the same event type to test.
10. If the test fails, the SQL error will be displayed.
11. Repeat steps 3 – 9 until the test is successful.

Testing Inbound Rules

In order to test an inbound rule:

1. From the mapping menu select 'Inbound Test'
2. Open a message file with the corresponding event-type
3. Review the test result. if an error occurs the message will display the SQL error.

or

1. Click button
2. Inbound test will be executed using loaded message
3. Review the test result. if an error occurs the message will display the SQL error.

Creating a Parent-Child Mapping Rule Example

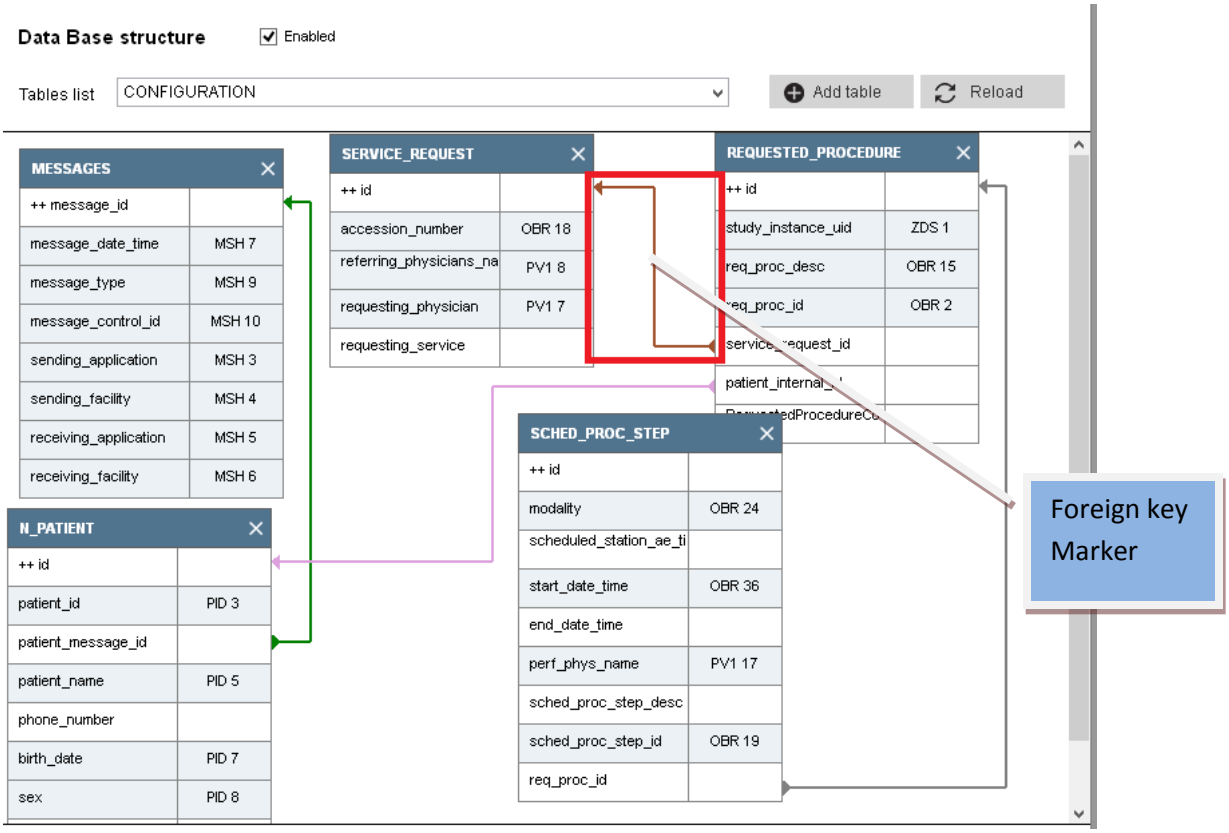
This example explains how to populate database tables with parent-child structure from a HL7 message.

In this example we'll use the ORU^R01 message. This message contains a report with optionally many observations.

We will map the data into two tables with a parent-child relation:

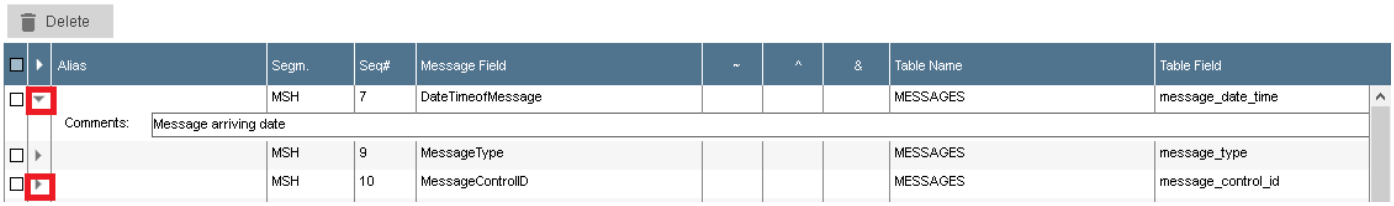
1. REPORTS
2. REPORT_ITEMS

HL7Kit identifies tables' relations using foreign-keys. Matching primary and foreign keys fields are connecting by "from-to" lines.



Rule Comments

Click on "Arrow left"/ "Arrow down" button in the rule line to open/close rule comment. Rules comments are stored in the mapping XML file.

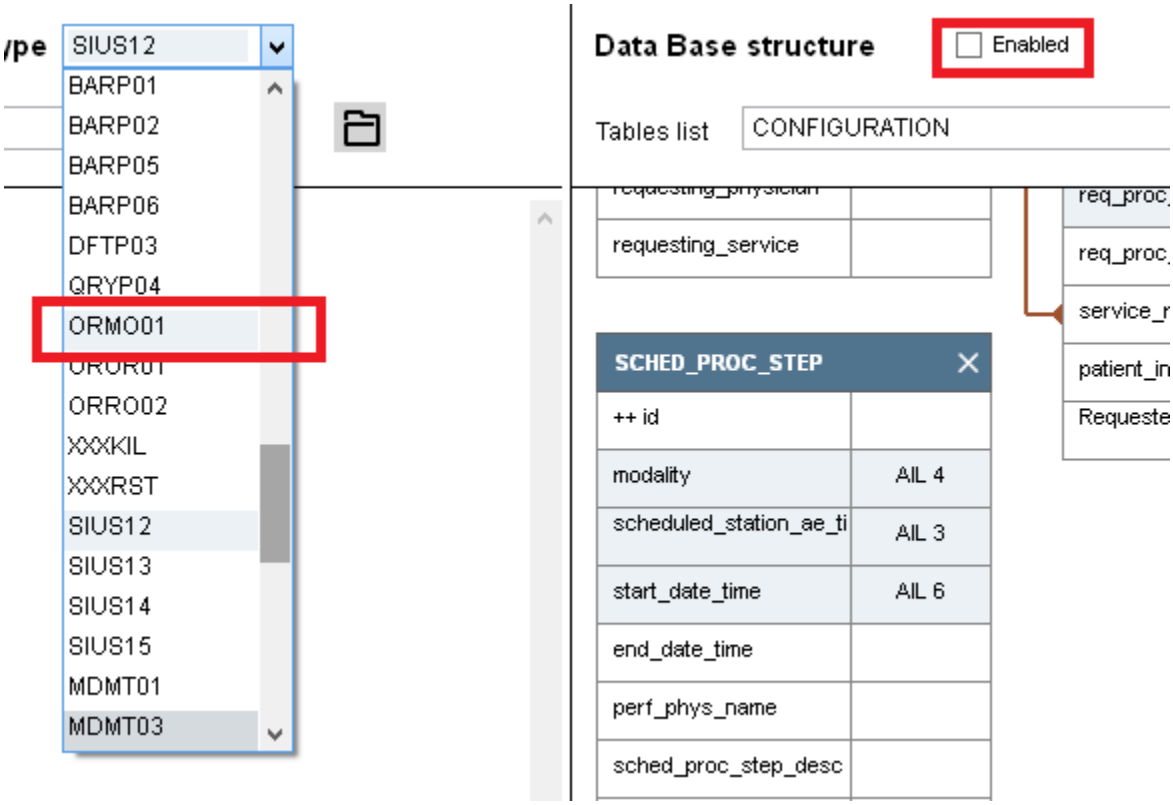


Enabled/Disabled mapping

HL7 Service will not process incoming messages if their mapping is disabled (entry will be added to System event log)

To mark any mapping as "Disable" user has to un-check "Enabled" check box. By default new mapping is enabled.

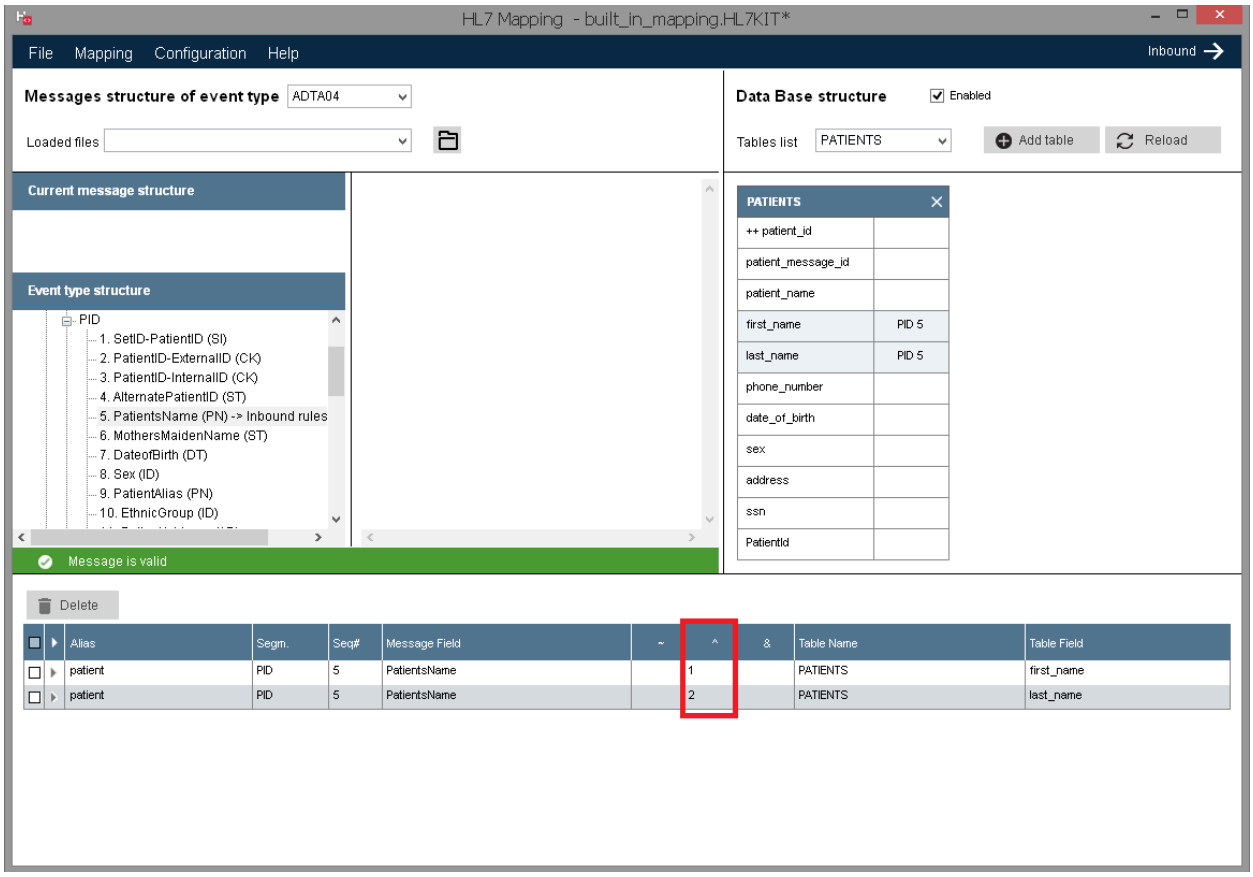
Existing mappings are marked by grey in the event types list. If there is no mapping for event type - its background in the list is white.



Components, Sub-Components and Repetitions

Sometimes, it is required to set part of a field in one column and another in another column. For example first and last name are divided by a ^ sign in the same field.

Follow this example that demonstrates breaking PID 5 into first and last name:



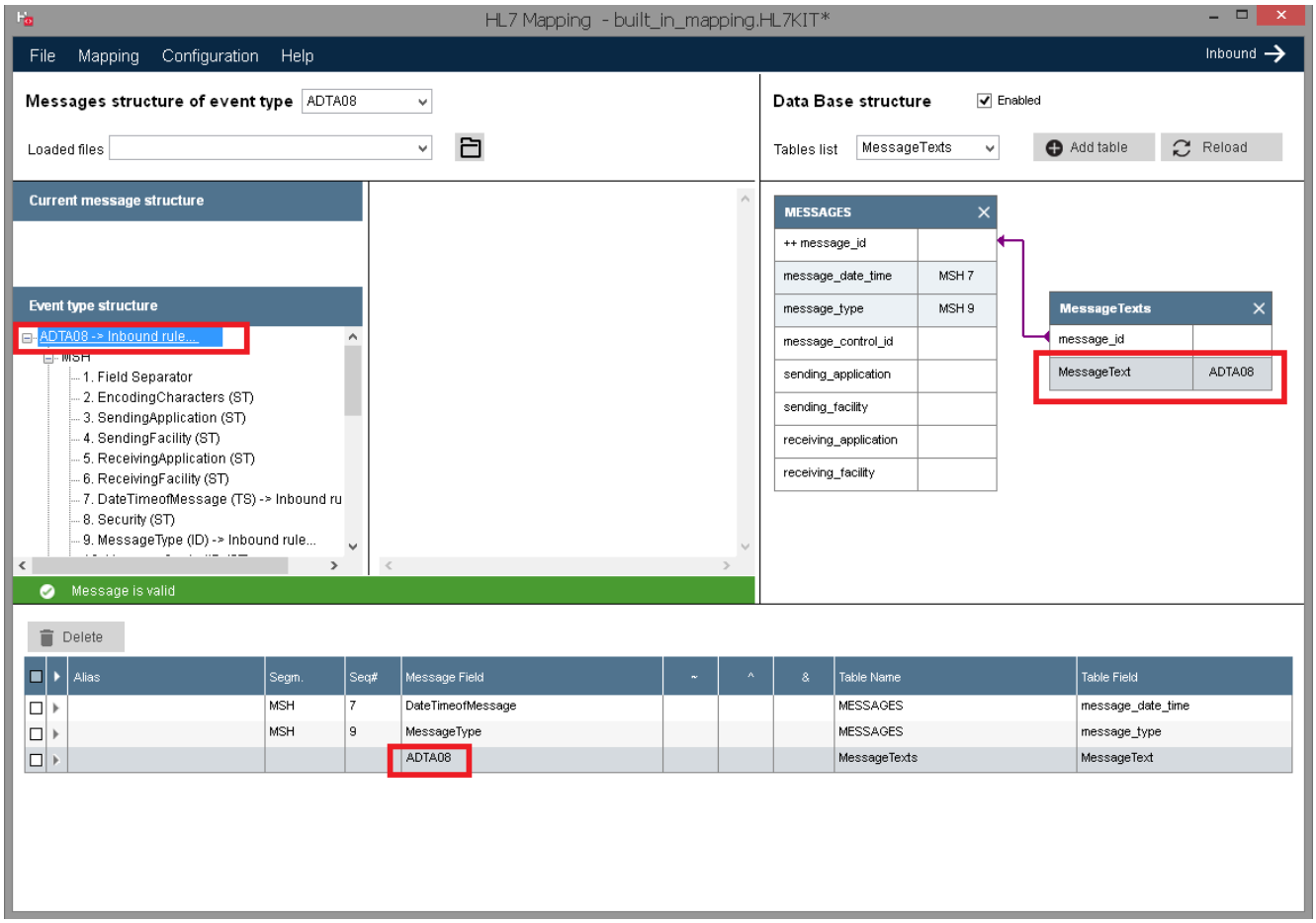
1. Start with unchecking the rollback test data in the configuration form, so you could see the results in the database
2. Select Event Type ADT^A04
3. Add the patients table to the database panel
4. Drag PID 5 on first_name field
5. In the rules panel - edit the rules that map first name and set the component (^) value to 1
6. Exit from "^" text box (click on the any other textbox)
7. Drag PID 5 this time over last_name field
8. In the rules panel - edit the rules that map last name and set the component (^) value to 2
9. Run the inbound test

	ID	PatientID	FirstName	LastName
	1	NULL	KING	MARTIN

The same can be done for sub-components (& separator) if each component between "^" is also composed of the few parts and repetitions (~ separator) for example when multiple phone numbers are present.

Inbound mapping of entire message text

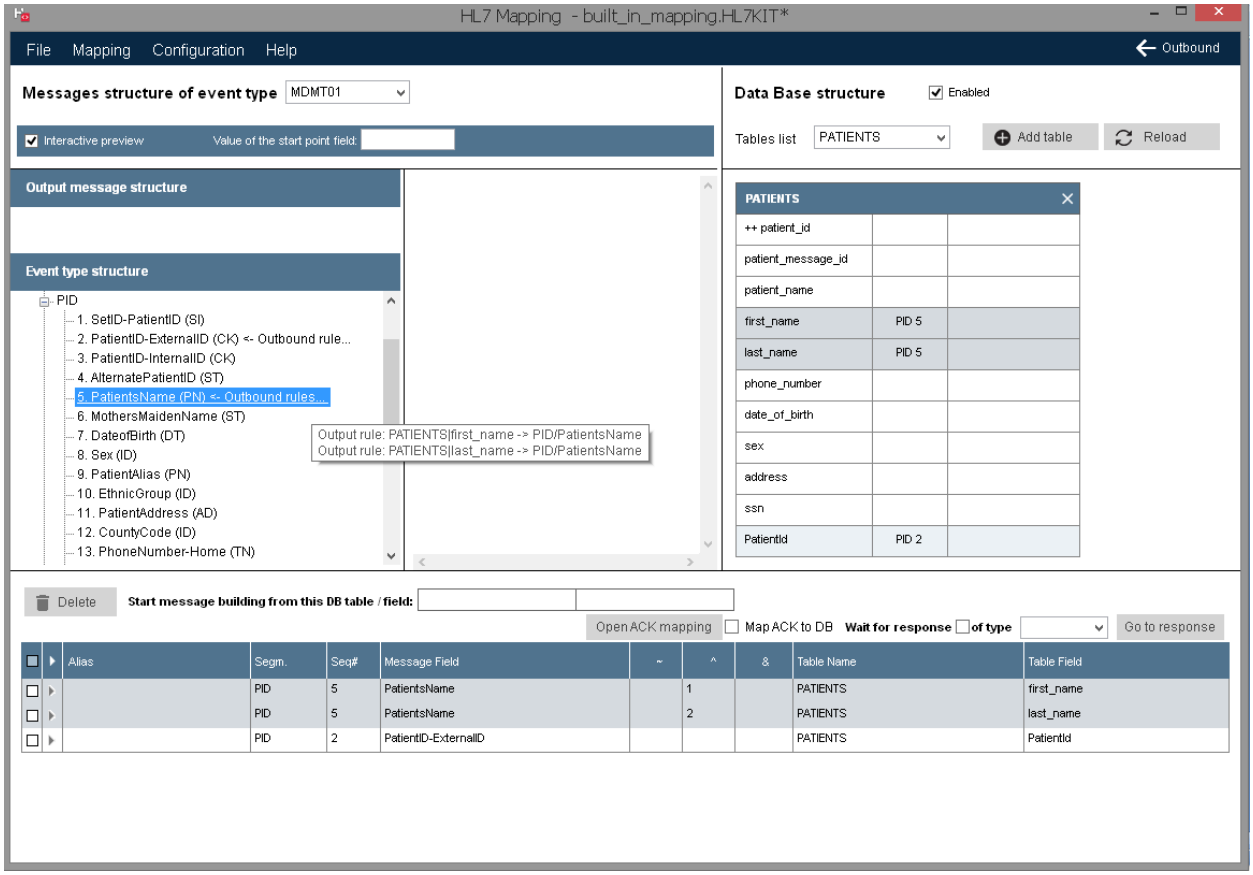
You can insert text of the incoming message into DB as a string. To do this, drag root node of the message tree (message type node) onto required DB field (this field must be big enough to absorb entire message which might be large enough, especially in V2.XML format).



Outbound Message Rules

Outbound messages are messages that are sent by HL7Kit Pro to an external system.

Creating a Simple Outbound Rule



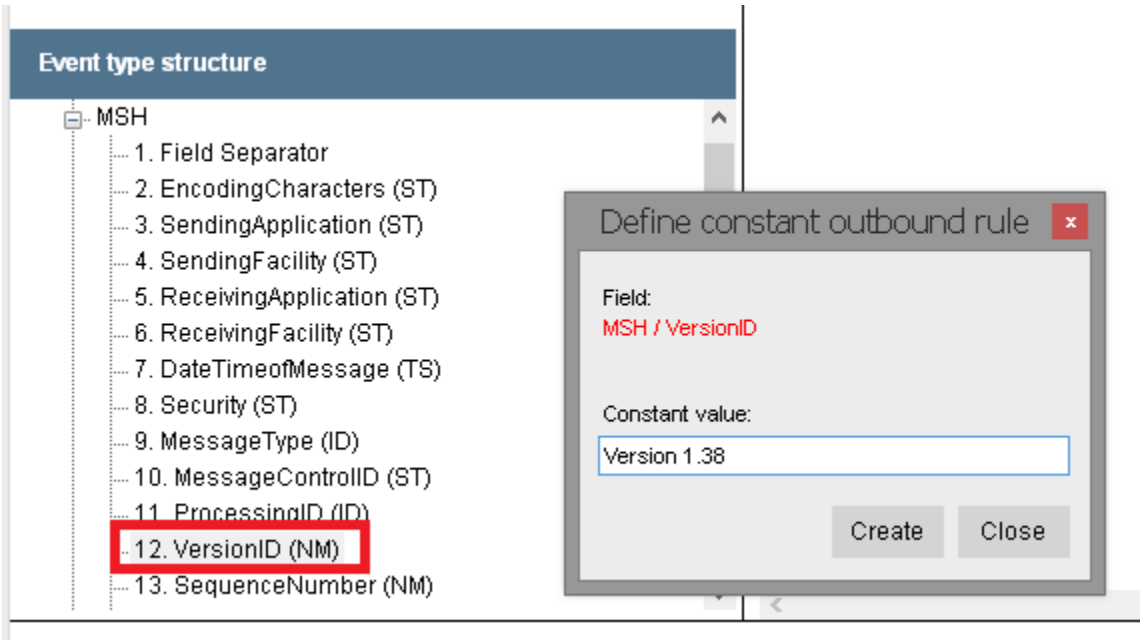
Outbound messages differ from inbound messages because they have to be triggered from the database. For this reason the 'Start message building from DB table/field' is required.

To set the start table/field drag a field from the database panel on this bar.

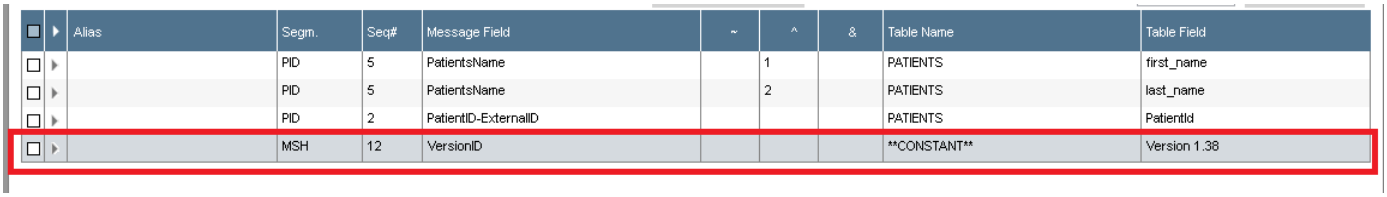
Follow these steps:

1. Drag the patient_id on the start table/field bar.
2. Drag PatientID column on PID2
3. Drag patient first name column on PID5
4. Set the ^ field of the rule to 1 (and click outside the text box)
5. Drag patient last name column on PID5
6. Set the ^ field of the rule to 2

To set a constant value to a field double click the field on the left panel and set the constant value:



Enter "Version 1.38" and click "Create":



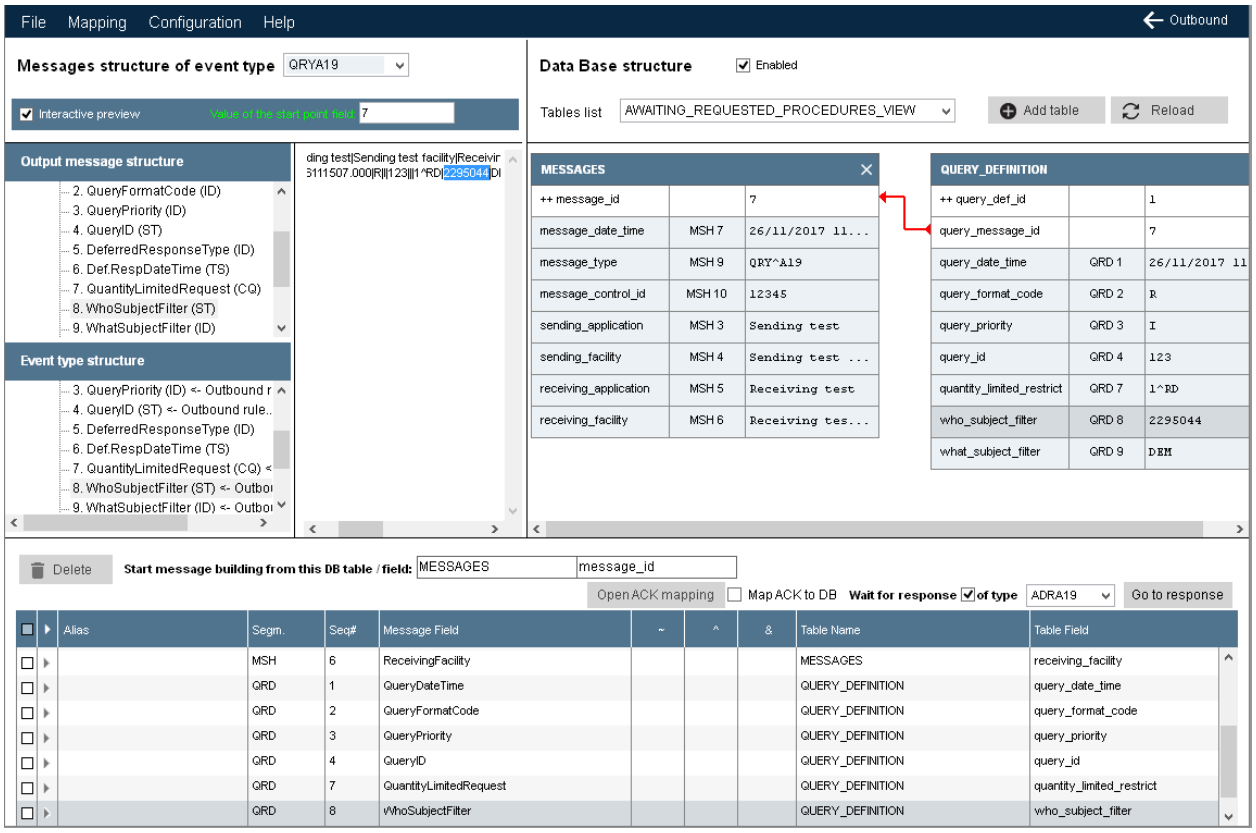
Alias	Segm.	Seq#	Message Field	~	^	&	Table Name	Table Field
	PID	5	PatientsName		1		PATIENTS	first_name
	PID	5	PatientsName		2		PATIENTS	last_name
	PID	2	PatientID-ExternalID				PATIENTS	PatientId
	MSH	12	VersionID				**CONSTANT**	Version 1.38

Testing Outbound Rules

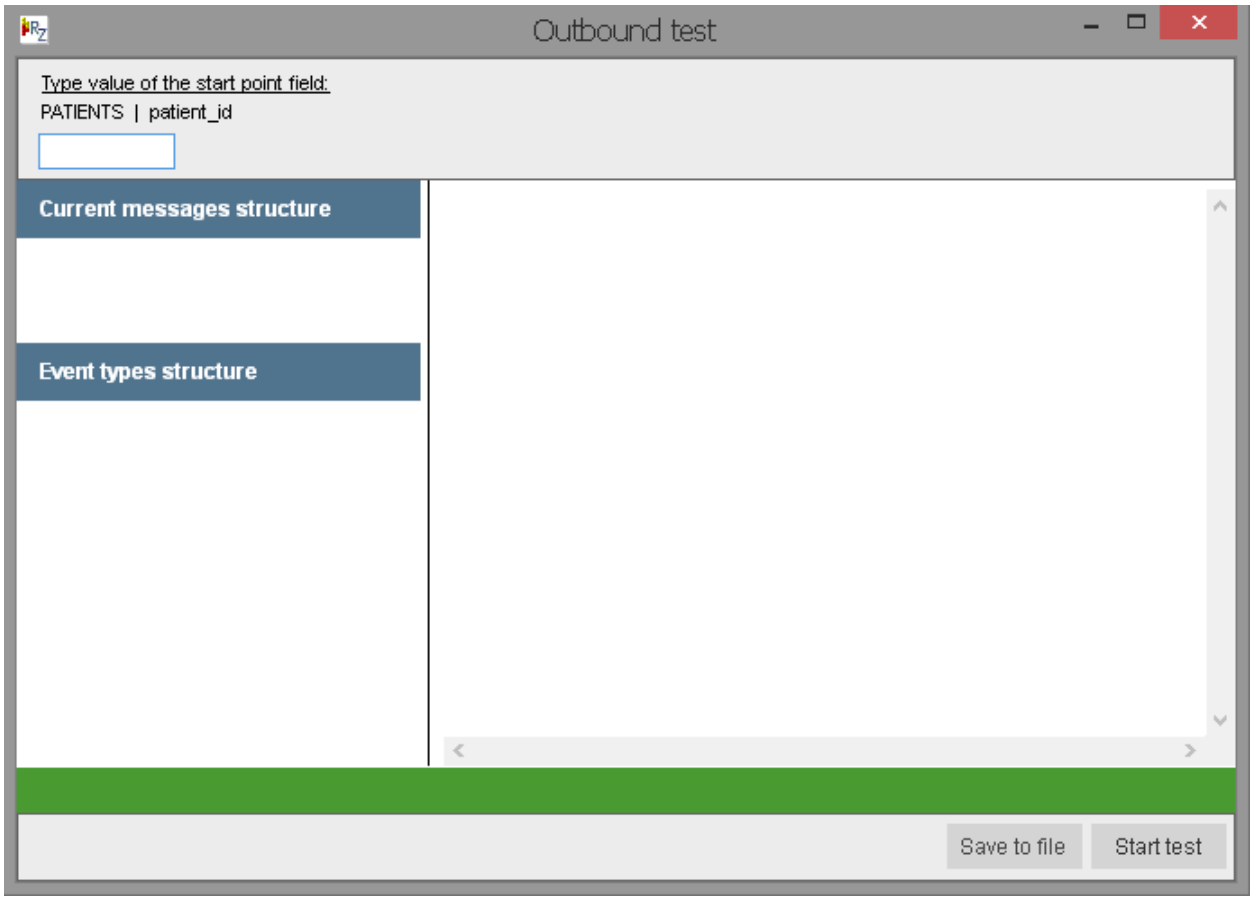
There is an option to preview outbound message while defining it's rules. For that mark "Interactive preview" check box - example of the current outbound message will be built on-the-fly using some data from the DB. Also all values from DB will be displayed as most right column of each table control in the "Data Base structure" area:

H.R.Z. Software Services LTD
6 David Yellin St.
Tel-Aviv, Israel

www.hl7kit.com
info@roniza.com



If you don't want to use it you can build example message in the separate window by selecting 'Outbound Test' from the Mapping menu.



In the start point field type the value of the field in the database that you would like to create a message from. In our example this would be one of the ID's in the Patients table.

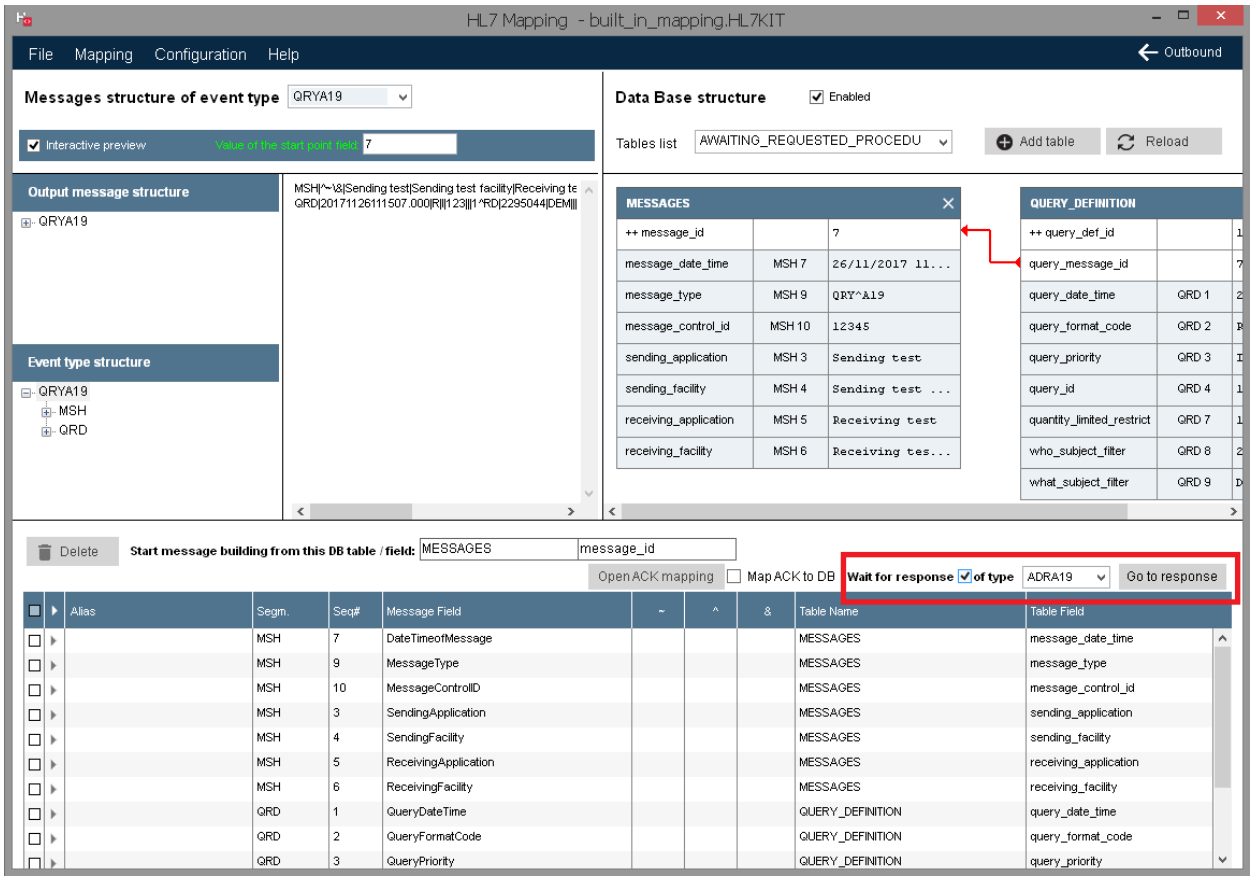
Creating a Parent-Child Outbound Rule

Creating a parent-child outbound message rule is similar to the process of parent-child inbound rule with one exception: you must choose the parent table key that is used to link to the child tables so that HL7Kit can start the processing from the correct place.

Creating HL7 Query Rules

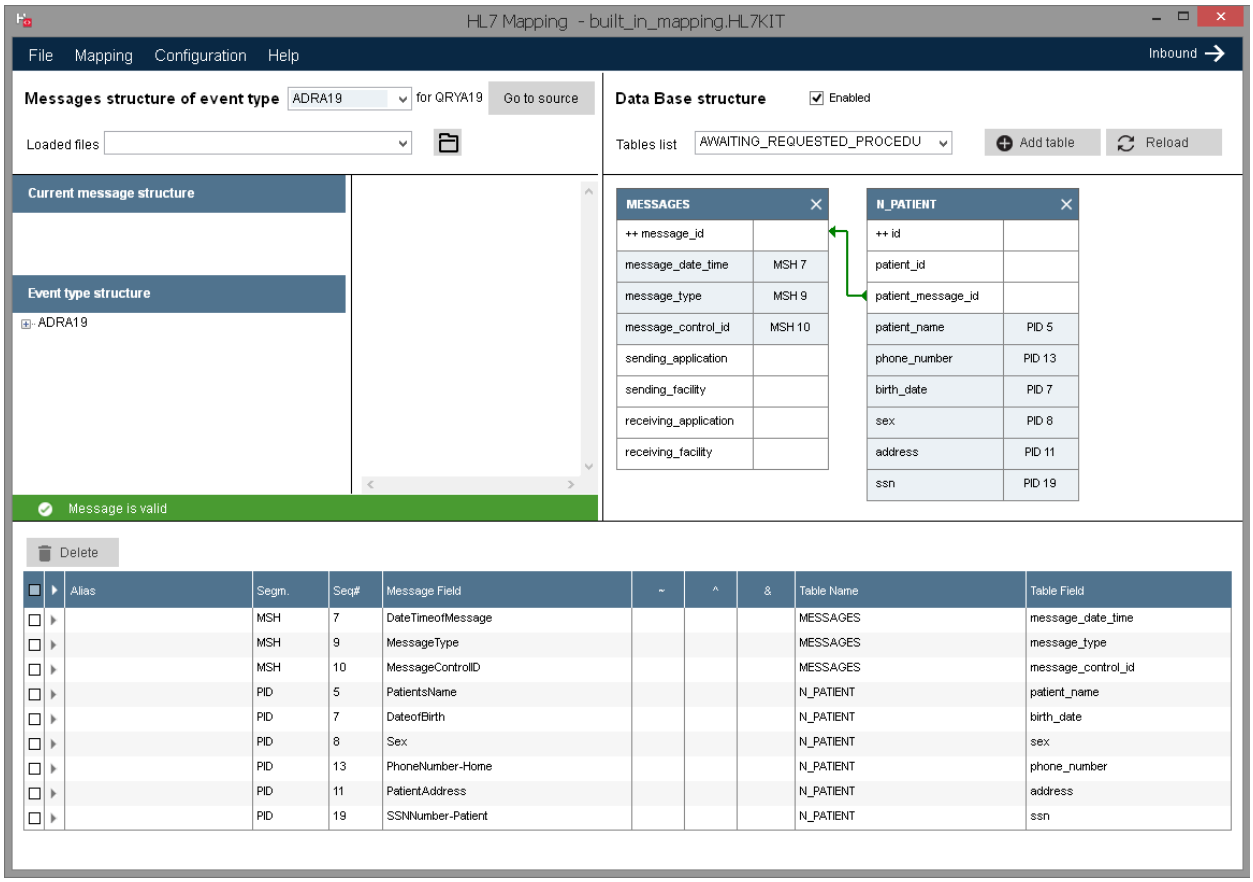
HL7 Queries enable a system to send an HL7 Message representing a query and receive a query result instead of an ACK message.

Creating Query rules is comprised of mapping the outbound query and then mapping the inbound result and linking the two together.



The 'Wait for response of type' checkbox and drop-box are used to notify the kit that instead of waiting for an ACK message, a different response is expected.

The 'Go to response' button switches to the inbound mapping for the response event type.



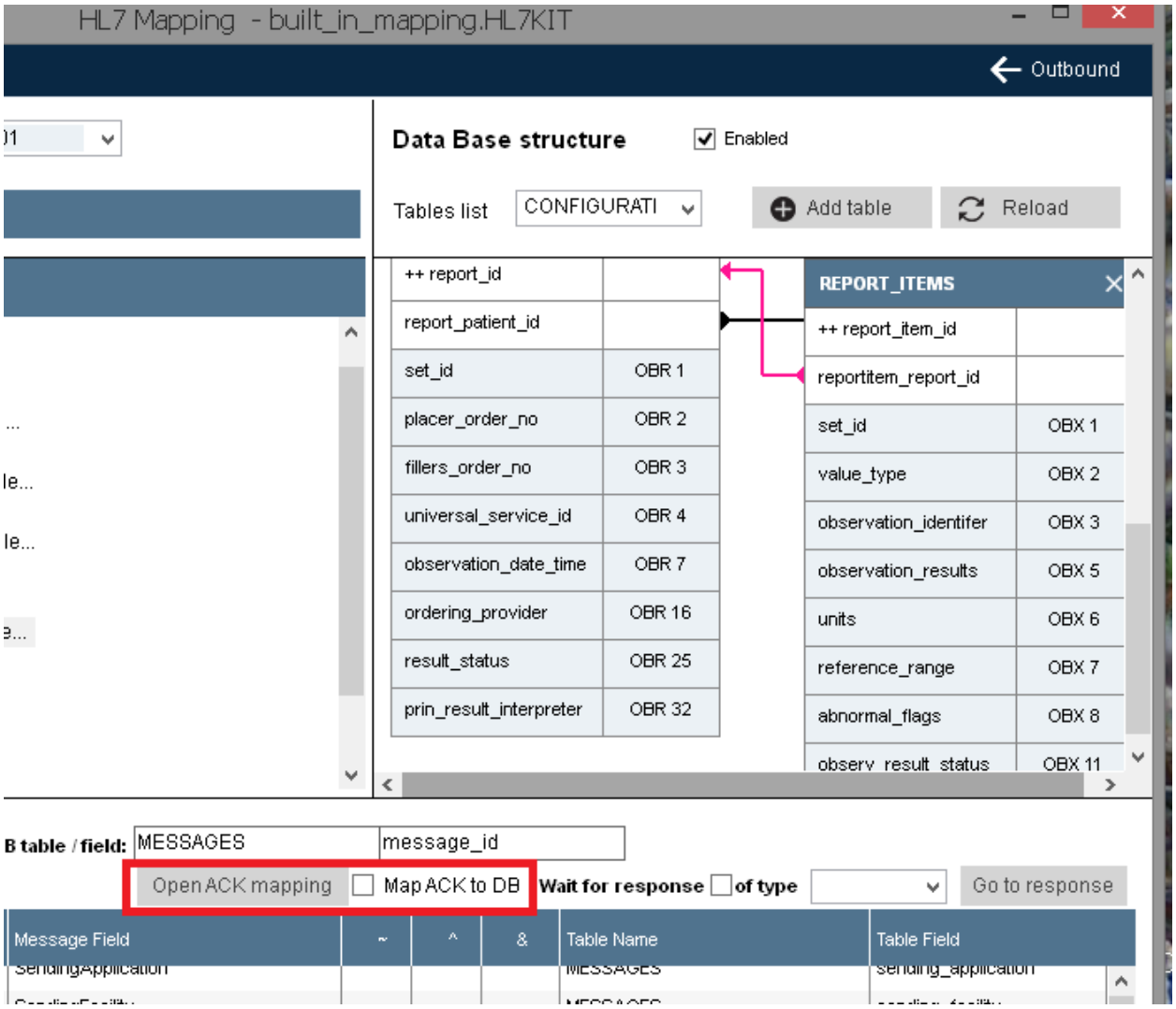
Database Panel Symbols

Table columns in the database table panels have the following symbols:

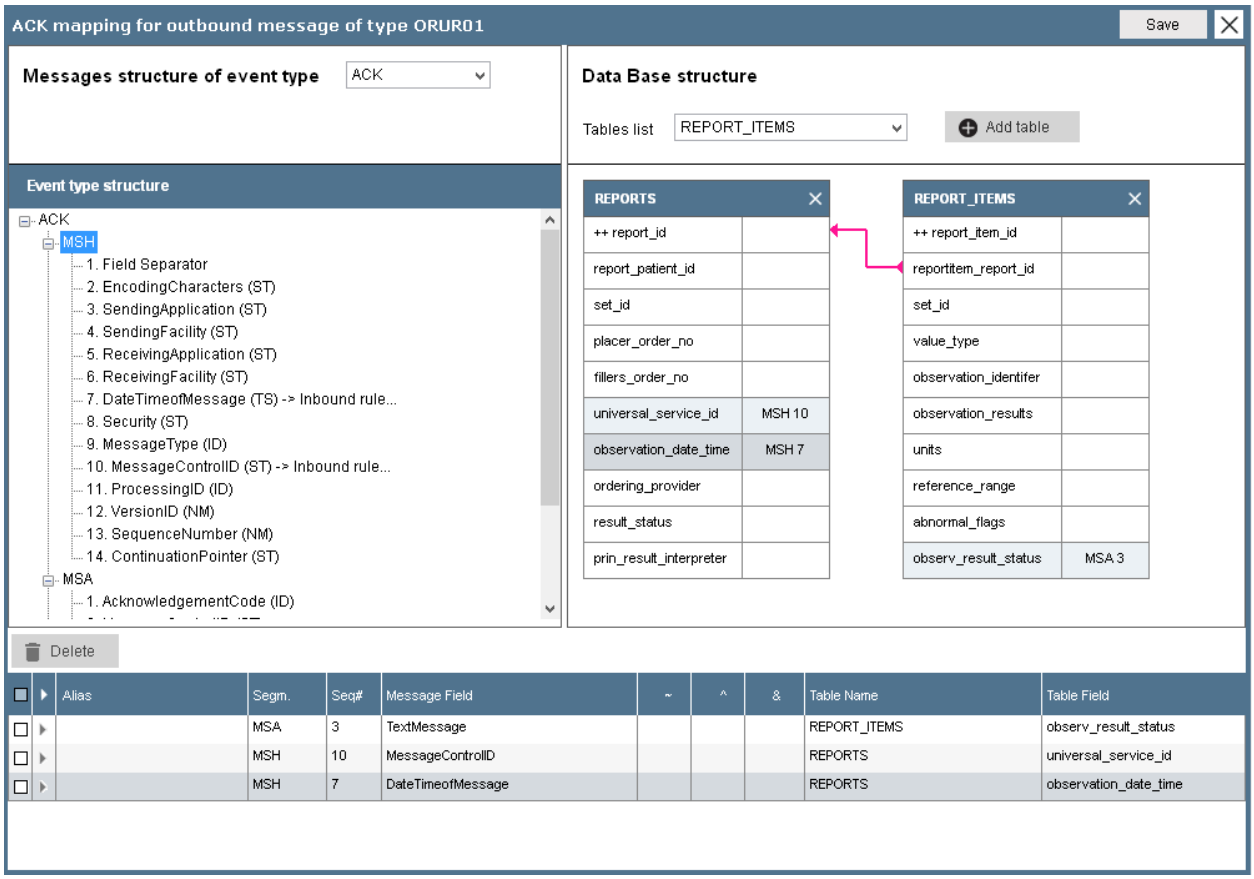
++	Identity Column

Map ACK message to Data Base

There is an option to insert data from ACK message received by HL7 Service into Data Base.
In the bottom part of the HL7 Mapper screen, near "Wait for response" checkbox, there is new checkbox "Map ACK to DB" and button "Open ACK mapping" (enabled when "Map ACK to DB" selected):



On "Open ACK mapping" click - pop-up window will be opened, allowing to define inbound mapping rules for selected ACK type.



If user clicks on "Save" - ACK mapping will be stored as XML node inside original outbound rule node like:

<ORUR01 StartTable="MESSAGES" StartField="message_id" ACKMappingEvent="ACKXXX" Response="">

<ACKXXX>

<Rule MsgAlias="" MsgSegment="MSH" MsgField="SendingApplication" SeqNum="2" Component="-1" Repetition="-1" SubComponent="-1" TblName="ACKParentTable" TblField="ACKParentField" />

.....

</ACKXXX>

<Rule MsgAlias="" MsgSegment="MSH" .../>

.....

</ORUR01>

When outbound thread of the HL7 Service receives ACK it checks in the current rule for ACK mapping existence and if exists and received ACK is of defined type - inserts ACK content to DB

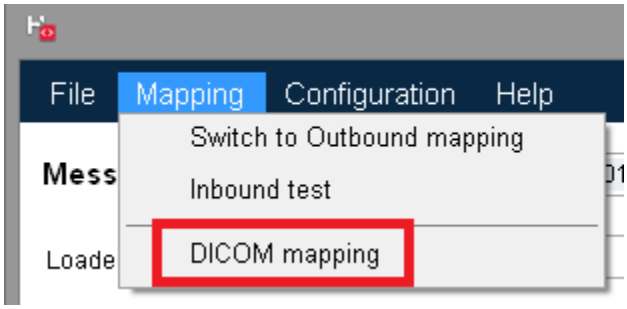
Inbound DICOM mapping

HL7 Kit allows to map incoming DICOM files which DICOM server (installed as a part of HL7 Kit) stores on the disk.

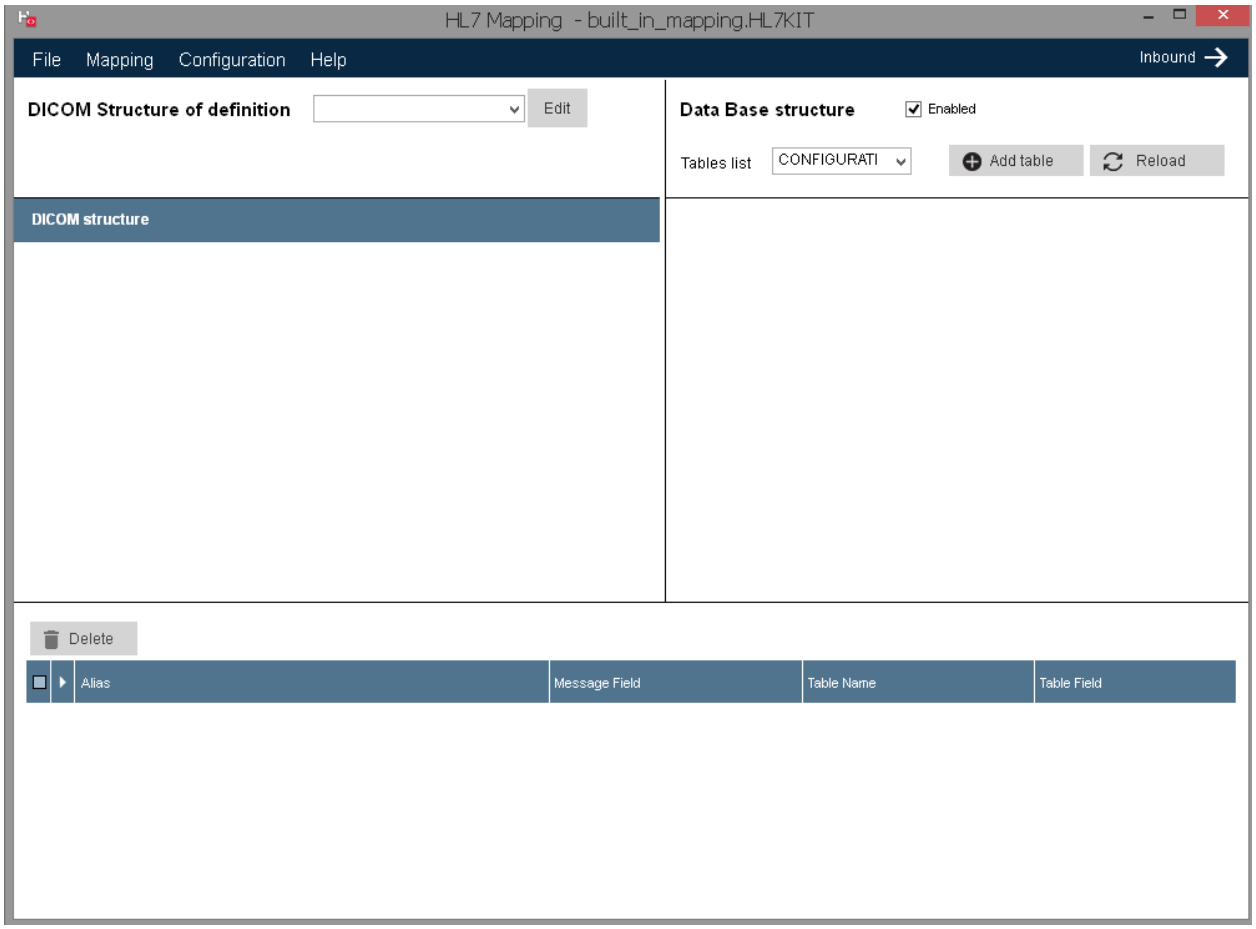
Basic info about each DICOM file (patient info, study info etc.) is stored by a DICOM service in the DICOM_FILES table. If you want to extract additional data from a DICOM file and save it in the DB - you can do it using our DICOM mapping mechanism.

How to create DICOM mapping in the HL7 Mapper application

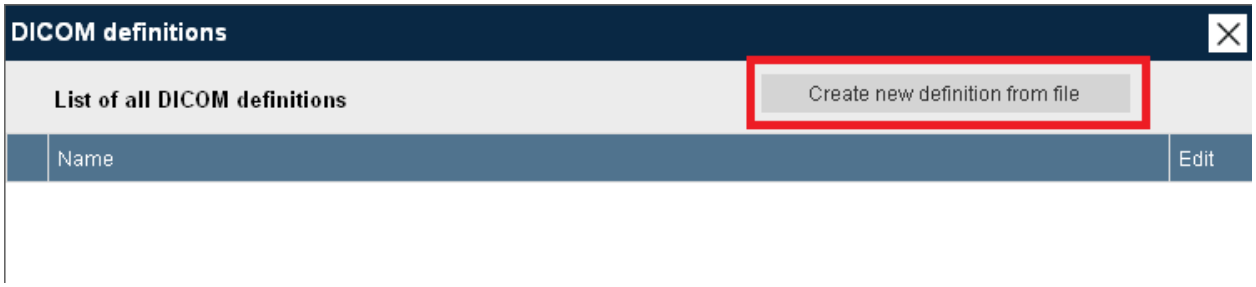
To switch from HL7 mapping to DICOM, click "DICOM mapping" menu:



Next screen will be opened



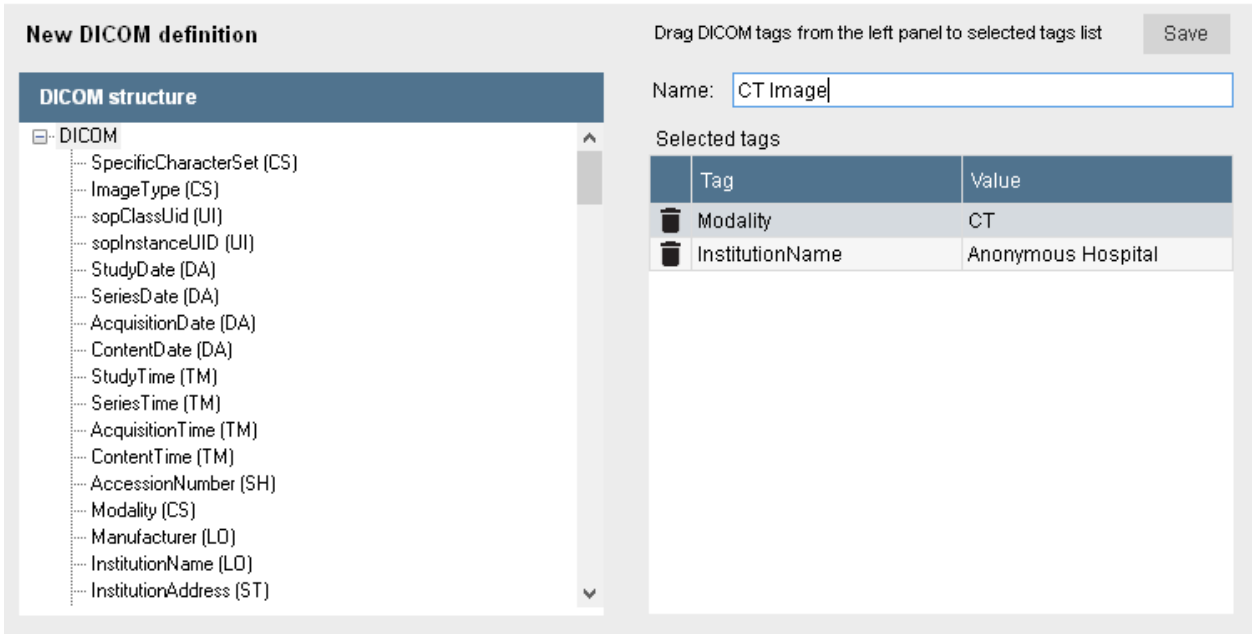
Click "Edit" to start DICOM definitions editing (screen allowing to define by what tags incoming DICOM file will be identified by mapping mechanism, similar to event type of HL7 message):



Click "Create new definition from file" to add new DICOM definition and select template DICOM file (one containing tags you want to use to identify all incoming DICOM files of one type, for example CT images or Structured Reports).

All tags will be displayed in the left panel. Drag-drop required tags to the right panel to add them to the list.
Enter some meaningful name to display in the list of DICOM definitions on the main screen. Click "Save" to add it to the list of all definitions or just close the form.

NOTE: You can save definition without any DICOM tag selected, in that case **ALL** incoming DICOM file will be processed therefore this definition has to be only one in the mapping XML.



Saved definition will appear in the list. You can edit any existing definition (change name, add/remove fields) by clicking "Edit" in the grid row:

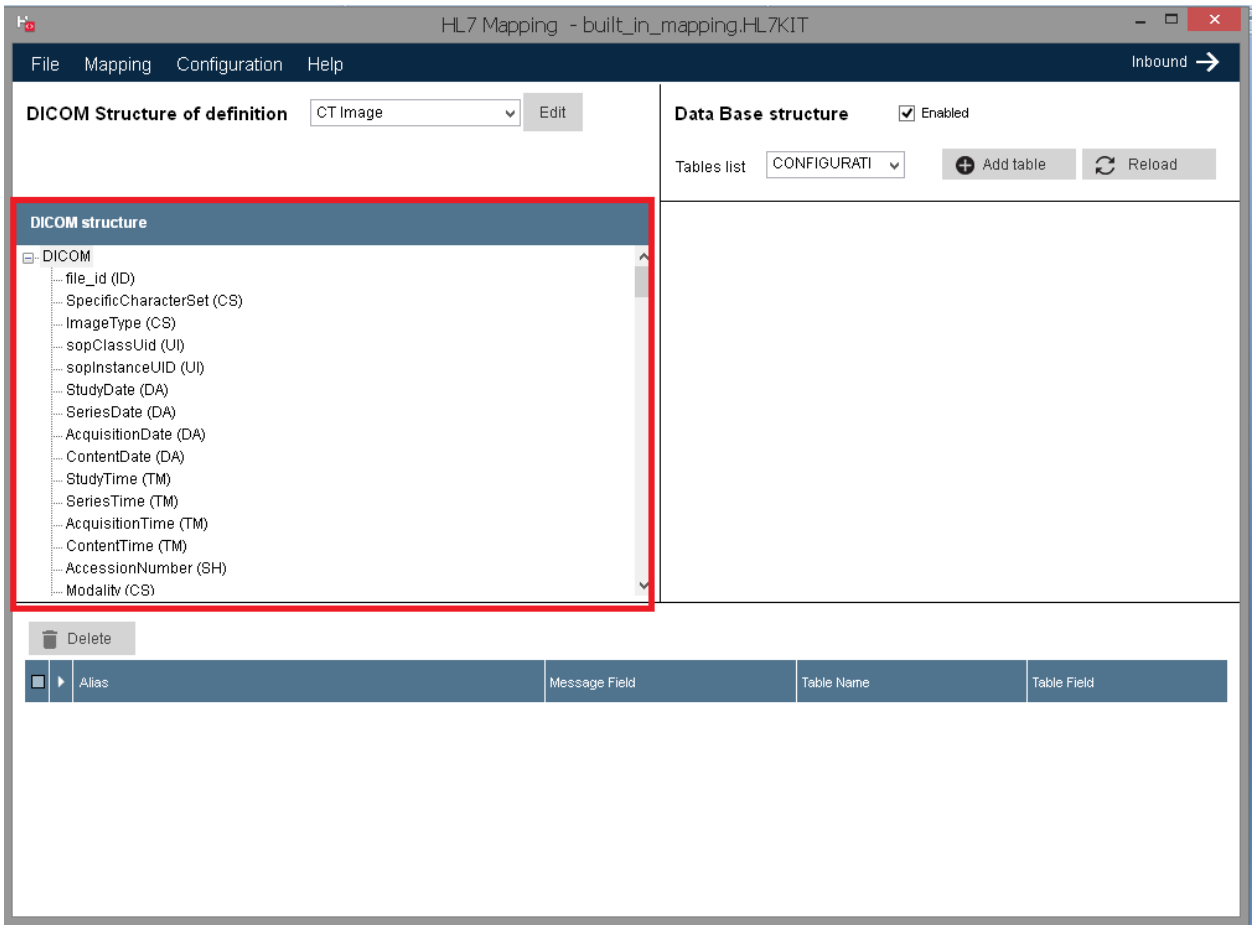
Template DICOM file will be copied into C:\HL7Kit\DICOMTemplates folder with name as "DICOMDefinition_1.dcm"

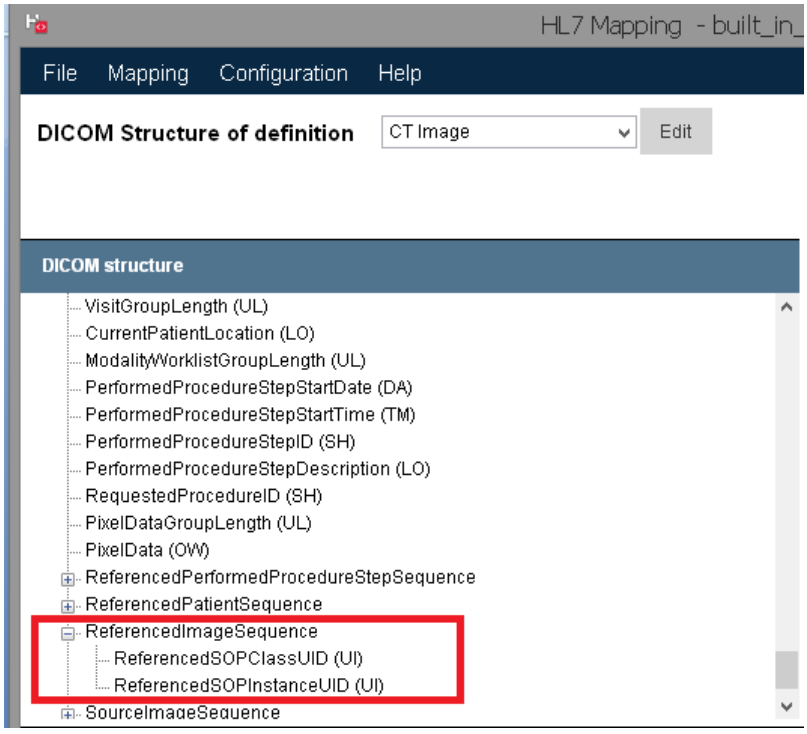
All DICOM definitions are stored in the file C:\HL7Kit\DICOMDefinitions.xml :

```
<DICOMDefinitions>
	<Definition>
		<ID>DICOMDefinition_1</ID>
		 <Name>CT Image</Name>
		<PathToSource>C:\HL7Kit\DICOMTemplates\DICOMDefinition_1.dcm</PathToSource>
		<Fields>
			<Field>
				<Tag>Modality</Tag>
				<Value>CT</Value>
			 </Field>
			<Field>
				<Tag>InstitutionName</Tag>
				<Value>Anonymous Hospital</Value>
			</Field>
		</Fields>
	</Definition>
</DICOMDefinitions>
```

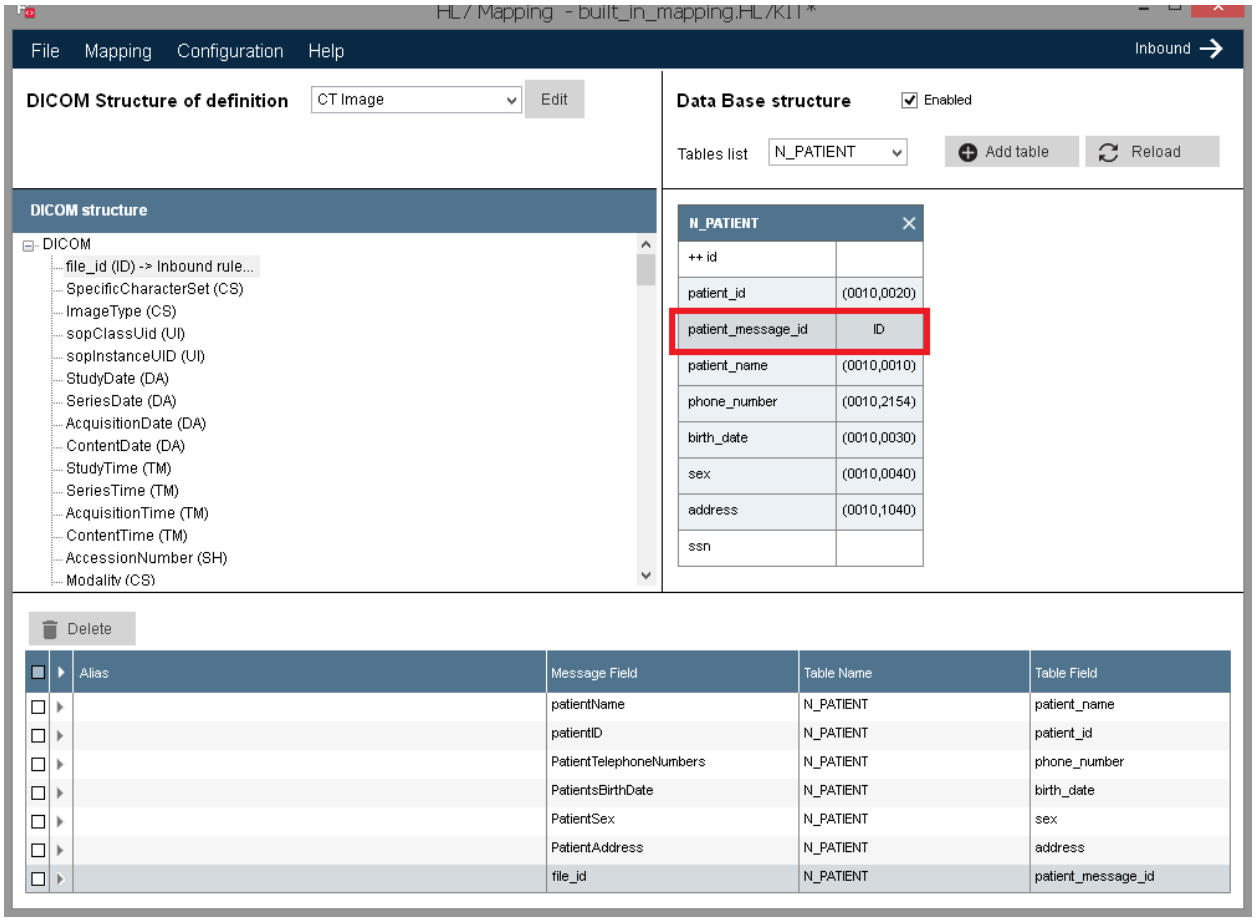
To delete definition - click "X" button in the grid row.

Click on "Close" to return to the main screen, new definition will appear in the list and all tags from the template DICOM file will be loaded to "DICOM structure" tree view (including sequences which are used as aliases of the HL7 messages)


HL7 Mapping - built_in_mapping.HL7KIT
File
Mapping
Configuration
Help
Inbound
DICOM Structure of definition
CT Image
Edit
Data Base structure
Enabled
Tables list
CONFIGURATI
Add table
Reload
DICOM structure
DICOM
file_id (ID)
SpecificCharacterSet (CS)
ImageType (CS)
sopClassUid (UI)
sopInstanceUID (UI)
StudyDate (DA)
SeriesDate (DA)
AcquisitionDate (DA)
ContentDate (DA)
StudyTime (TM)
SeriesTime (TM)
AcquisitionTime (TM)
ContentTime (TM)
AccessionNumber (SH)
Modality (CS)
Delete
Alias
Message Field
Table Name
Table Field


HL7 Mapping - built_in_
File
Mapping
Configuration
Help
DICOM Structure of definition
CT Image
Edit
DICOM structure
VisitGroupLength (UL)
CurrentPatientLocation (LO)
ModalityWorklistGroupLength (UL)
PerformedProcedureStepStartDate (DA)
PerformedProcedureStepStartTime (TM)
PerformedProcedureStepID (SH)
PerformedProcedureStepDescription (LO)
RequestedProcedureID (SH)
PixelDataGroupLength (UL)
PixelData (OW)
ReferencedPerformedProcedureStepSequence
ReferencedPatientSequence
ReferencedImageSequence
ReferencedSOPClassUID (UI)
ReferencedSOPInstanceUID (UI)
SourceImageSequence

Select required table and drag-drop DICOM tags to a table fields (exactly as for HL7 message):



NOTE: "file_id" of the DICOM structure is not real DICOM tag but value of the "file_id" field from the row of DICOM_FILES table (or id field of MPPS table and so on) .

If you want to connect your custom table to the DICOM_FILES table filled by DICOM server - map first node of the DICOM tree (file_id which is values of the PK field of the DICOM_FILES) into corresponding column of your table.

Mapping will be saved in the file (for example default built_in_mapping.HL7KIT) as:

```
<DICOMDefinition_1 Response="">
    <Rule MsgAlias="" MsgSegment="" MsgField="file_id" SeqNum="-1001" Component="-1"
Repetition="-1" SubComponent="-1" TblName="CTImages" TblField="file_id" />
    <Rule MsgAlias="" MsgSegment="" MsgField="sopInstanceUID" SeqNum="524312" Component="-1"
Repetition="-1" SubComponent="-1" TblName="CTImages" TblField="sopInstanceUID" />
    <Rule MsgAlias="" MsgSegment="" MsgField="InstanceNumber" SeqNum="2097171" Component="-
1" Repetition="-1" SubComponent="-1" TblName="CTImages" TblField="InstanceNumber" />
```

```
    <Rule MsgAlias="" MsgSegment="" MsgField="ImagePositionPatient" SeqNum="2097202"
Component="-1" Repetition="-1" SubComponent="-1" TblName="CTImages"
TblField="ImagePositionPatient" />
    <Rule MsgAlias="" MsgSegment="" MsgField="ImageOrientationPatient" SeqNum="2097207"
Component="-1" Repetition="-1" SubComponent="-1" TblName="CTImages"
TblField="ImageOrientationPatient" />
</DICOMDefinition_1>
```

where name of the node (DICOMDefinition_1) is ID from the DICOMDefinitions.xml

DICOM Mapper application (**process**ing **DICOM files**)

If you intend to use existing DataBase (default RZ_DICOM_HL7 for example) and it doesn't contain new columns required for DICOM mapping you need to run SQL script which adds new "Mapping_state" column to DICOM_FILES and MPPS tables and inserts new value to CONFIGURATION table:

MaxDICOMFilesToMap	100	How many DICOM files to insert into DB on one pass
port	104	Listener port
retry_seconds	180	The number of seconds to wait before retrying a failed task
root_dir	\DSRSVC	Working file system directory
scm_max_retry	3	The maximum number of retries for sending back commit responses

Script for MSSQL is like:

```
USE [RZ_DICOM_HL7]
GO
ALTER TABLE [dbo].[MPPS] ADD [mapping_state] [int] NULL CONSTRAINT [DF_MPPS_mapping_state]
DEFAULT ((0))
GO
UPDATE [dbo].[MPPS] SET [mapping_state] = 0
GO
ALTER TABLE [dbo].[DICOM_FILES] ADD [mapping_state] [int] NULL CONSTRAINT
[DF_DICOM_FILES_mapping_state]  DEFAULT ((0))
GO
UPDATE [dbo].[DICOM_FILES] SET [mapping_state] = 0
GO
INSERT INTO [dbo].[CONFIGURATION] ([Key],[Value],[Description]) VALUES ('MaxDICOMFilesToMap',
'100', 'How many DICOM files to insert into DB on one pass');
GO
```

There is console DICOMMapper application in the HL7 Kit installation folder. Its config file contains connection string to DB and path to mapping file which will be updated automatically when you define correspondent properties on HL7 Runtime Configuration panel. But they also can be changed manually if

for example you want to use separate mapping files for HL7 service and DICOM files processing. Just open DICOMMapper.exe.config in the Notepad and change path to mapping file in the:

```
 <appSettings>
        <add key="MappingXMLName" value="built_in_mapping.HL7KIT"/>
 </appSettings>
```

Config file also contains list of all tables containing info about DICOM files which will be proccessed by DICOMMapper. By default this list includes DICOM_FILES table as "active" and MPPS table as "not active":

```
 <TablesToScan>
   <tables>
    <add TableName="DICOM_FILES" IDField="file_id" StatusField="mapping_state"
RelPathField="rel_path" />
    <!--<add TableName="MPPS" IDField="id" StatusField="mapping_state" RelPathField="rel_path" />--
>
   </tables>
 </TablesToScan>
```

If you want to proccess MPPS table too - just remove "<!--" and "-->"

You can add any other table which has unique ID field, integer field for storing mapping state (initial value must be 0) and field containing relative path to a DICOM file inside folder defined in CONFIGURATION table as "root_dir"

To process all new rows in all tables - run DICOMMapper.exe (recommended way to do it - add DICOMMapper to the system scheduler). All rows from each table with mapping_state value 0 (ready for mapping) will be processed: DICOM files will be opened and checked against all DICOM definitions defined in Mapper. If DICOM file contains all tags of the definitions and values of these tags are the same - application will search for found definition in the selected mapping file. If mapping is found - all tags from DICOM files defined in the mapping rules will be inserted to the correspondent tables.

After all new rows are processed corresponding value in the CONFIGURATION table will be updated so next time only rows added after last processed file will be fetched:

EVGENIA\SQ...NFIGURATION

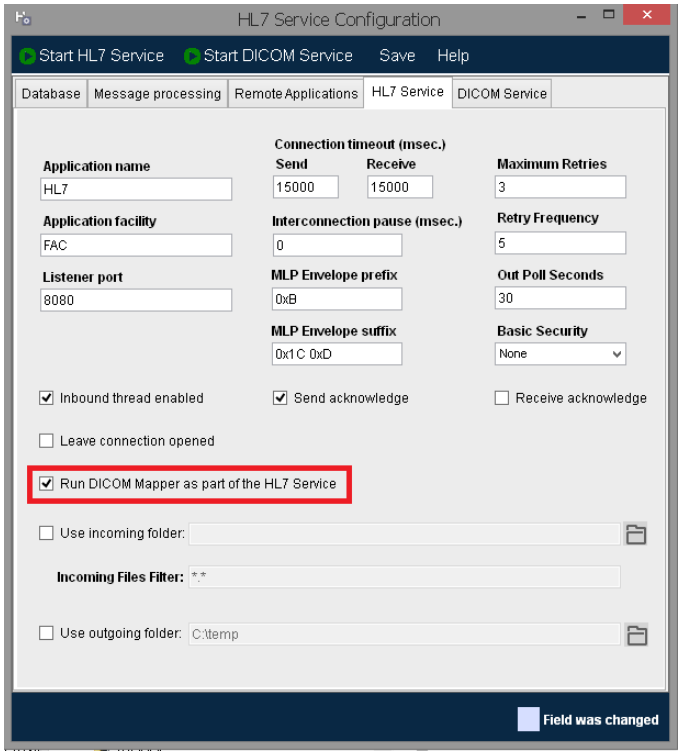
Key	Value	Description
ae_title	DSRSVC	Local ae title
dicom_log	0	When this value is 1, detailed logging is on
DM_DICOM_FILES_last_id	367	Last ID proccessed by DICOM mapper
DM_MPPS_last_id	8	Last ID proccessed by DICOM mapper
MaxDICOMFilesToMap	100	How many DICOM files to insert into DB on one pass
port	104	Listener port
retry_seconds	180	The number of seconds to wait before retrying a failed task
root_dir	\DSRSVC	Working file system directory
scm_max_retry	3	The maximum number of retries for sending back commit responses

Value of the mapping_state field of each processed row will be updated accordingly to mapping result: 1- mapped OK, 2 - mapping failed, 3- DICOM definition or mapping rule was not found. If result is not OK - message will saved in system EventLog ("RZHL7Pro" log)

EVGENIA\SQL...DICOM_FILES | EVGENIA\SQ...NFIGURATION

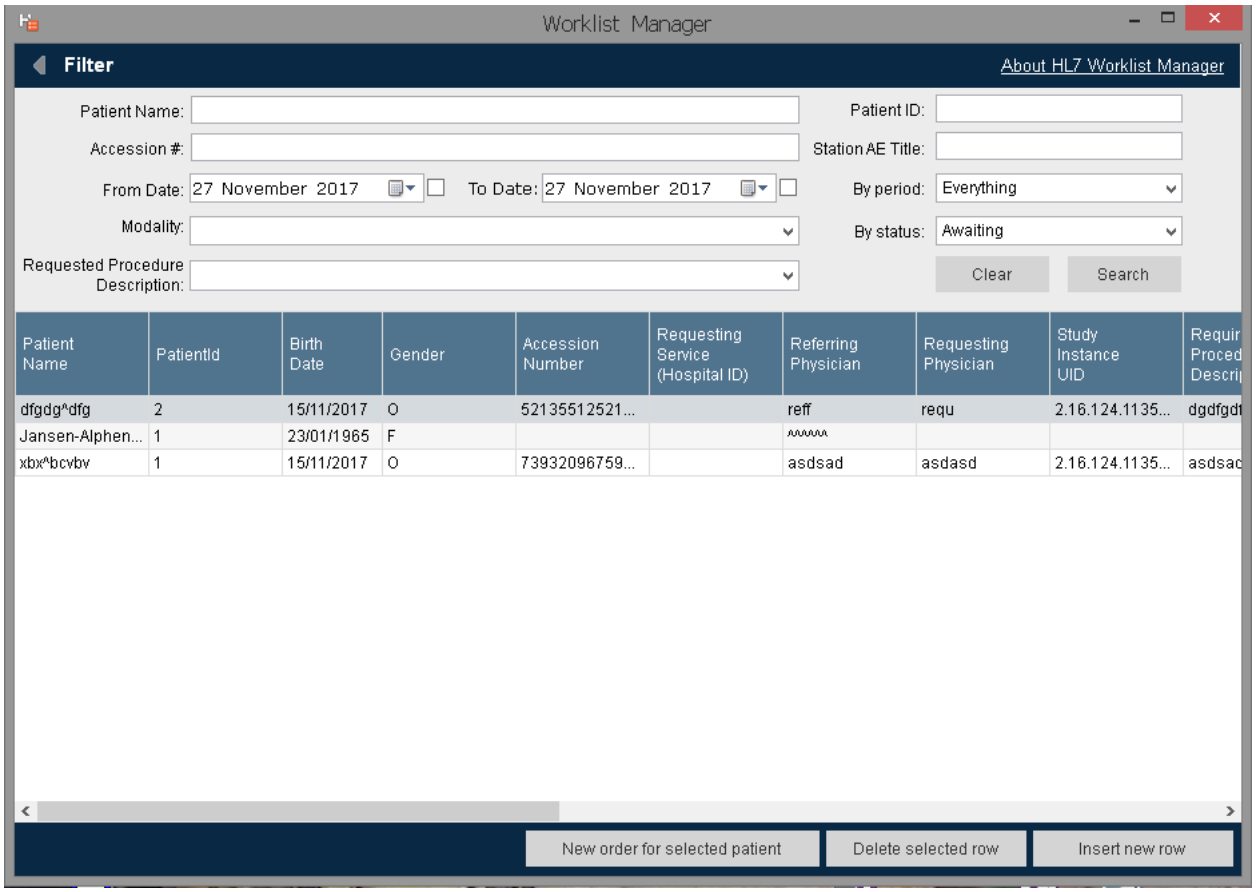
ty	series_nu...	series_ins...	instance_...	sop_class...	sop_insta...	source_a...	rel_path	task_id	mapping...
	3	2.16.124....	1	1.2.840.1...	2.16.124....	RZDCX	2016/06/0...	NULL	1
	1	2.16.124....	1	1.2.840.1...	2.16.124....	RZDCX	2016/06/0...	NULL	1
	1	2.16.124....	NULL	1.2.840.1...	2.16.124....	RZDCX	2016/06/0...	NULL	1
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

NOTE: There is also an option to execute DICOM mapping as part of the HL7 service (you can select it using HL7 Runtime Configuration application)



Worklist Manager

A utility for managing MWL related tables in RZ_DICOM_HL7 Data Base which you can create during installation process or by running 'RZ_DICOM_HL7_CreateDB_Script.sql' script from 'MSSQLScripts' sub-folder located in your HL7Kit installation folder.



In main screen you can fetch list of studies from DB filtered by parameters you choose, edit values of the existing study, create new study for existing patient, create new study, entering new patient data and delete existing study.

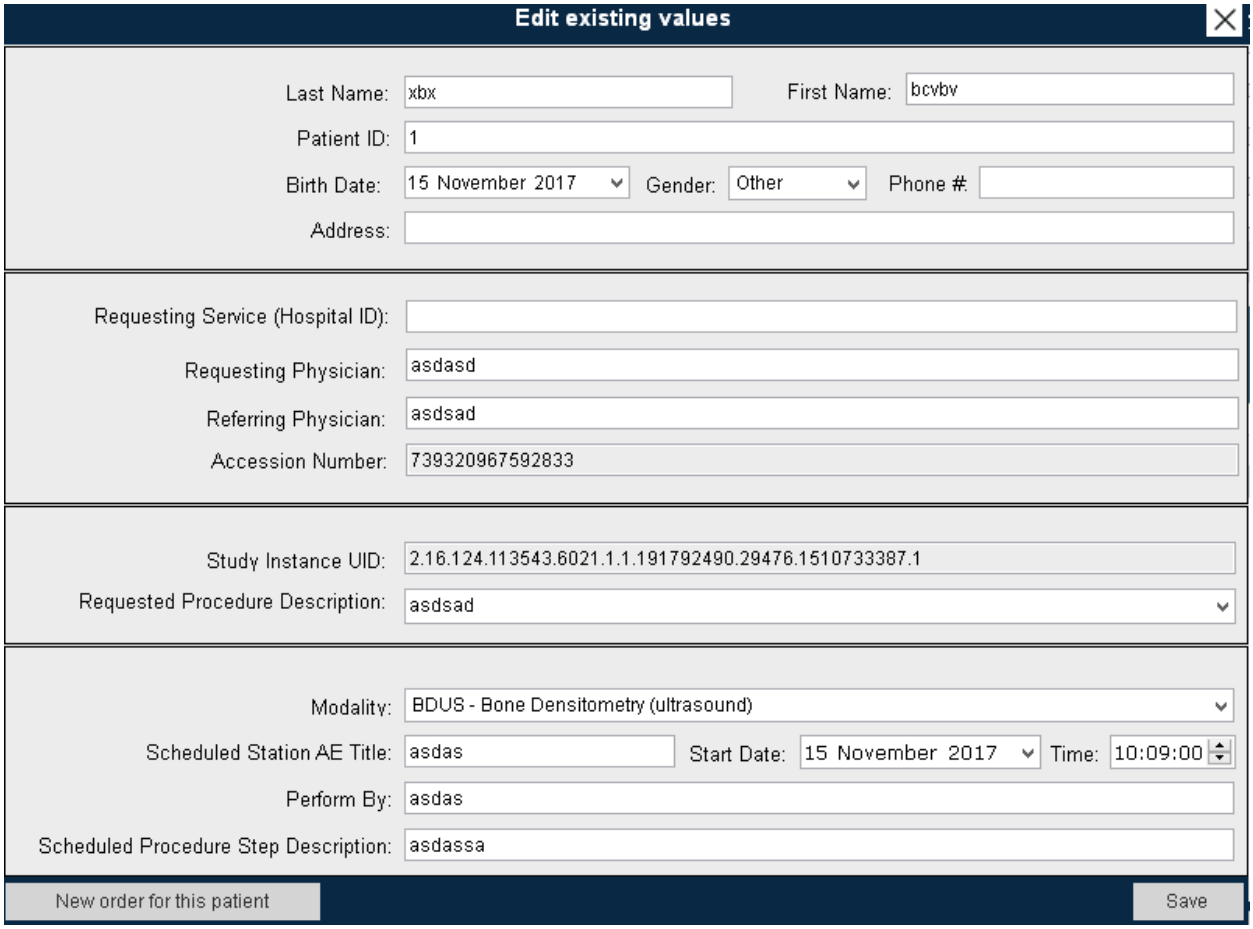
On application Launching, today's studies will be loaded. To get another set of studies - set required filters and click "Search". Clicking “Search” with no search parameter set will list all the existing studies. To reset all filters - click "Clear".

"By period" filter value is selected from the list containing 'Everything' (meaning that study date will not be checked in query), Today, This Week, This Month and This Year.

Modality filter value is selected from a pre-defined list which is stored in ModalityCodes.txt file located in your HL7 Kit installation folder. You can open this file in Notepad and add new modality as new line like " SM *** Slide Microscopy" where "SM" is modality code for DB query and "Slide Microscopy" is description to display in the list.

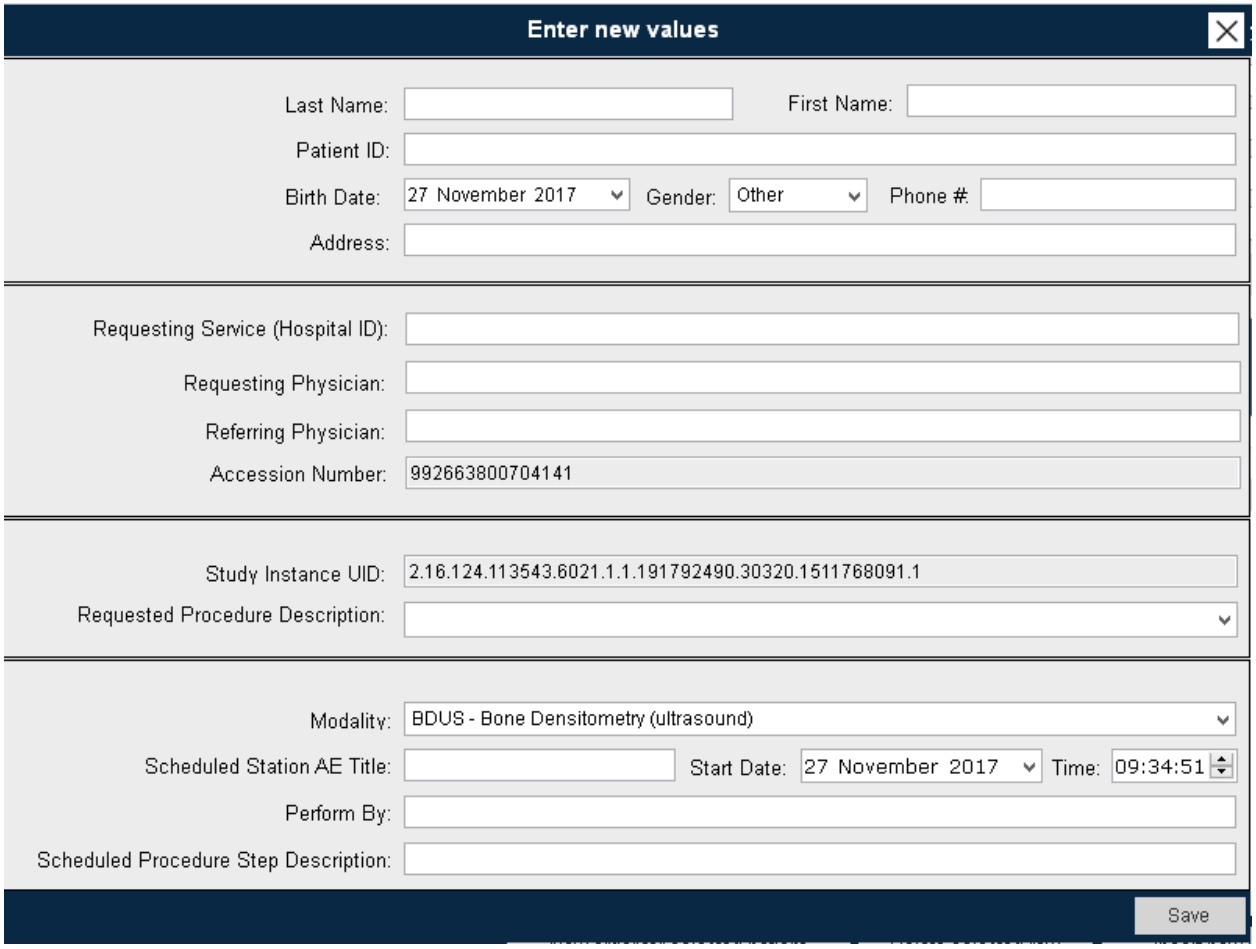
Requested Procedure Description filter value is selected from pre-defined list which is stored in ProcedureDescription.txt file located in your HL7 Kit installation folder. You can open this file in Notepad and add new description. You can also type description text.

To edit values of the existing study - double-click on its row, change required fields in the pop-up window and click "Save":



You can create a new study for a current patient by clicking "New order for this patient" - all patient data will be disabled, new accession number and study instance UID will be generated. Change other study data as you wish and click "Save" to create a new study. You can create new study for existing patient from main screen too by selecting study row and clicking "New order for selected patient".

When you click "Insert new row" on the main screen - pop-up window for entering patient/study data will be opened with automatically generated Accession number & Study Instance UID:

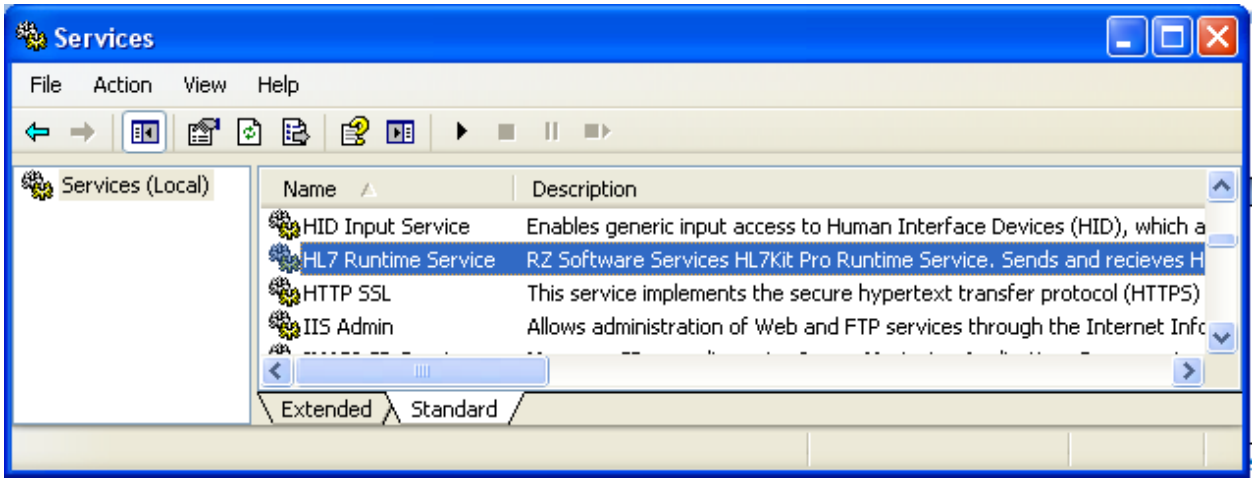


Enter all required data and click "Save"

To delete existing study - select row in the grid on main screen and click "Delete selected row".

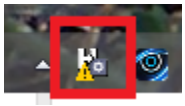
HL7Runtime Configuration

The HL7 Service is a background process that starts and stops automatically with the operating system. The service is responsible for receiving inbound messages and for sending outbound messages according to the defined mapping rules.

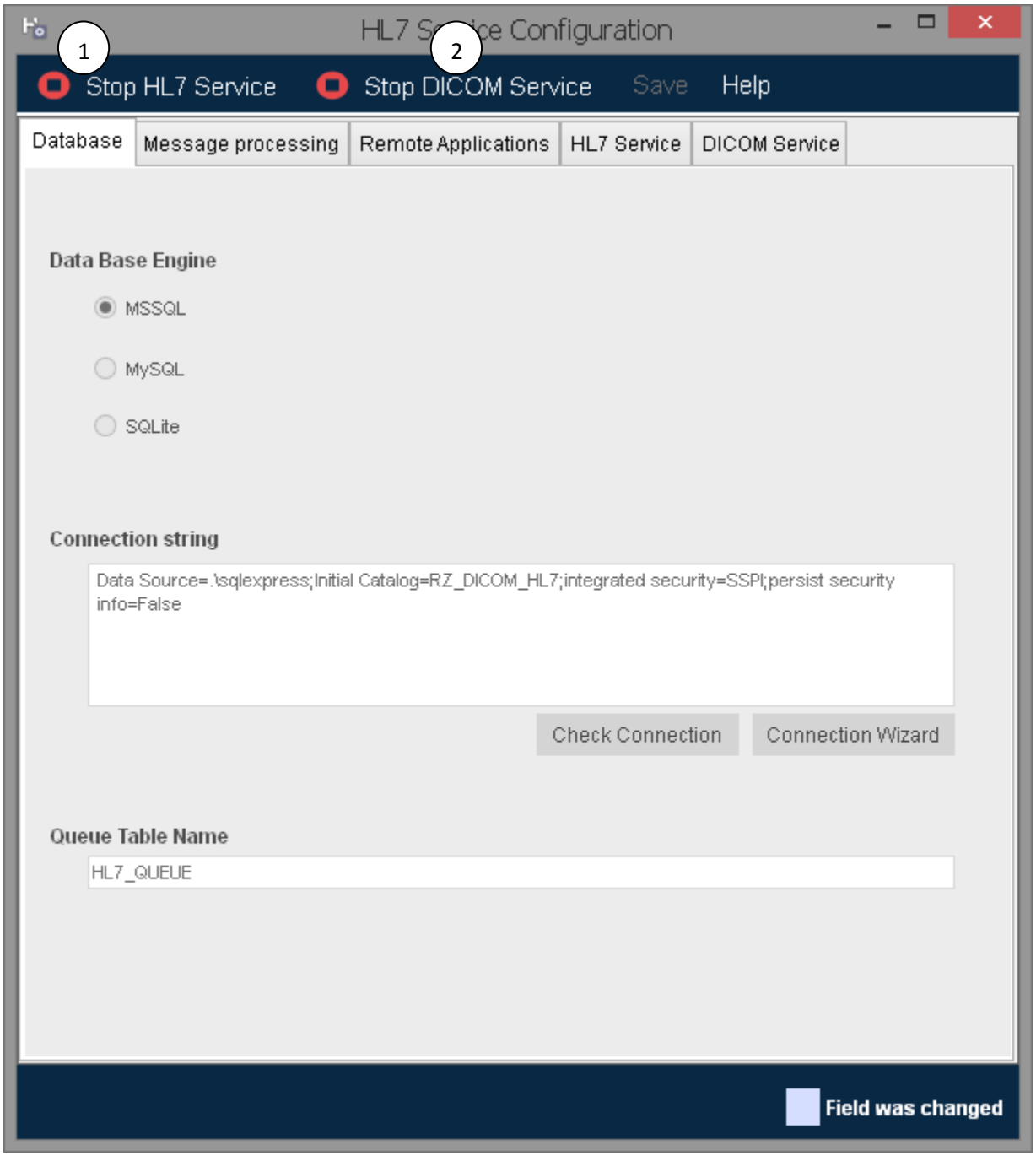


HL7 Runtime Configuration

The HL7 Runtime Configuration Control panel is installed together with the HL7 Runtime Service. When logging in, the control panel is started and minimized to the system tray area.



To open configuration screen - right click on the tray icon and select "Open HL7 Configuration panel". Also you can start/stop HL7 service and DICOM service and exit configuration application (it doesn't stop the service) by right-click on the tray icon.

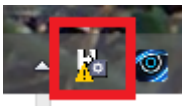


Services Start/Stop Buttons (1,2)

These buttons send a stop or start message to the runtime service.

Services Status Tray Indicator

This indicator shows the services status:

1. HL7 icon with exclamation sign : HL7 service or/and DICOM service stopped 
2. HL7 icon: both HL7 service and DICOM service are running

Save Menu

Saves the current configuration.

Changes Indicator

When edits are made, the modified controls and the save menu are highlighted in yellow.

How to connect to Data Base

See Connection to Data Base.

Connection String Text Box

See Connection String Text Box.

Check Connection Button

See Check Connection Button.

Connection Wizard Button

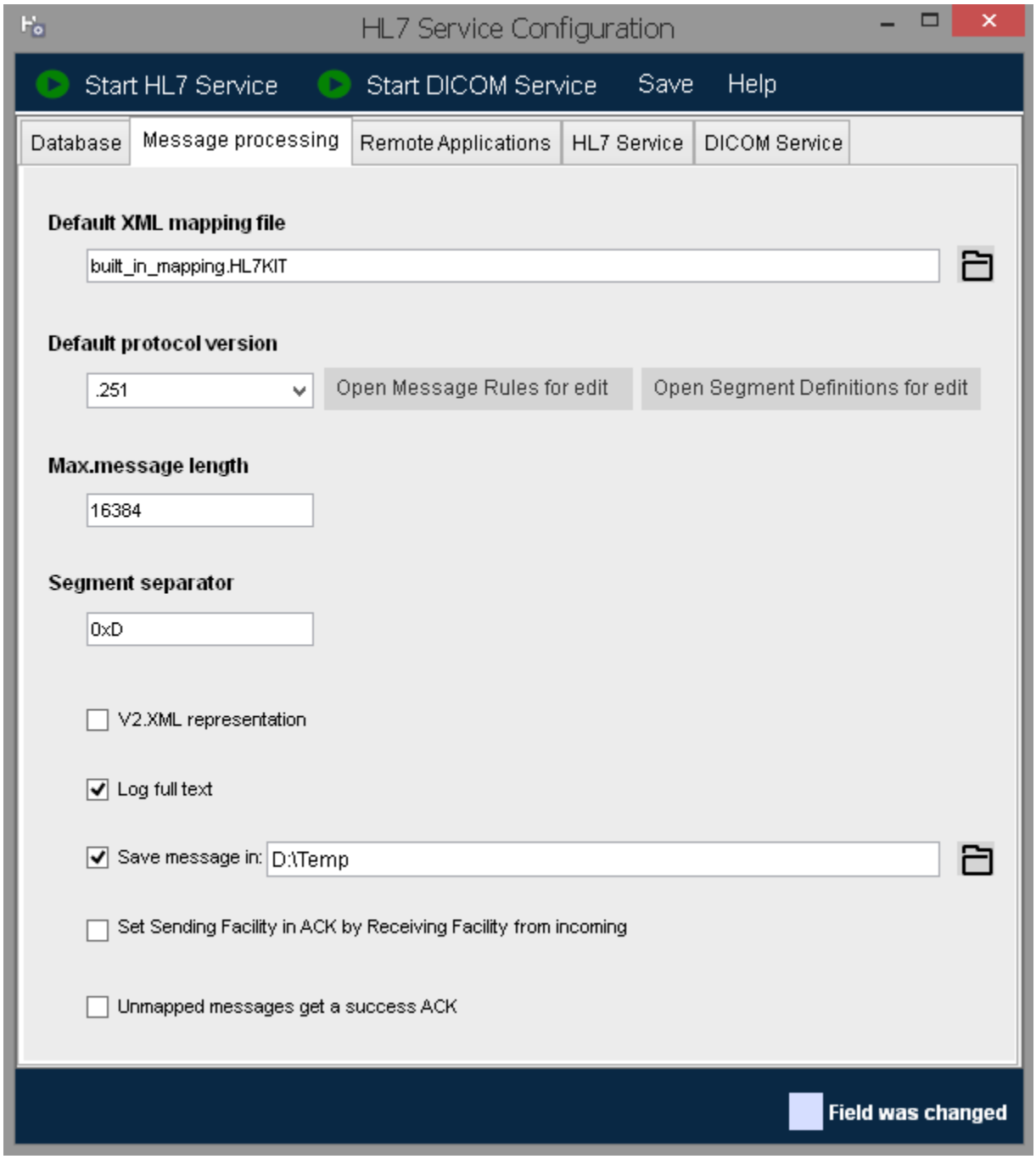
See Connection Wizard.

Select Data Base Button

See Select Data Base Button.

Queue Table Name

You may use a different name for the HL7_QUEUE table. This may be useful when running multiple instances of the HL7 Runtime or if by coincidence your database already has a table with that name. The HL7_QUEUE table structure must however stay the same.



XML Mapping File Text Box and Browse Button (Default Mapping File)

This is the default mapping file that the runtime service uses for processing messages. Use the browse button to select a file. The mapping file is created using the HL7Mapper application. You may configure a different mapping file for every remote application.

Message Configuration Version (Default)

See Message Configuration Version. You may configure a different message configuration version for every remote application.

Max. Message length

Maximum length (in bytes) of the buffer to send message through network. Increase this value if you intend to send large messages.

Segment Separator

Use this text box to set the characters combination (Hexadecimal notation) to use as segment separator. By default it's 0xD (LF).

V2.XML representation Checkbox

When checked, send/receive all messages formatted as HL7 V2 XML. This property might be overridden for each remote application.

Log Full Text Check-Box

When checked, the complete message body of every inbound and outbound message is logged in the event log. See Monitoring and Logging.

Save Message Controls

When check-box is checked, the complete message body of every inbound message is saved in the selected folder. **NOTE:** This option works only if "Log full text" is selected.

Set Sending Facility in ACK by Receiving Facility from incoming message Check-Box

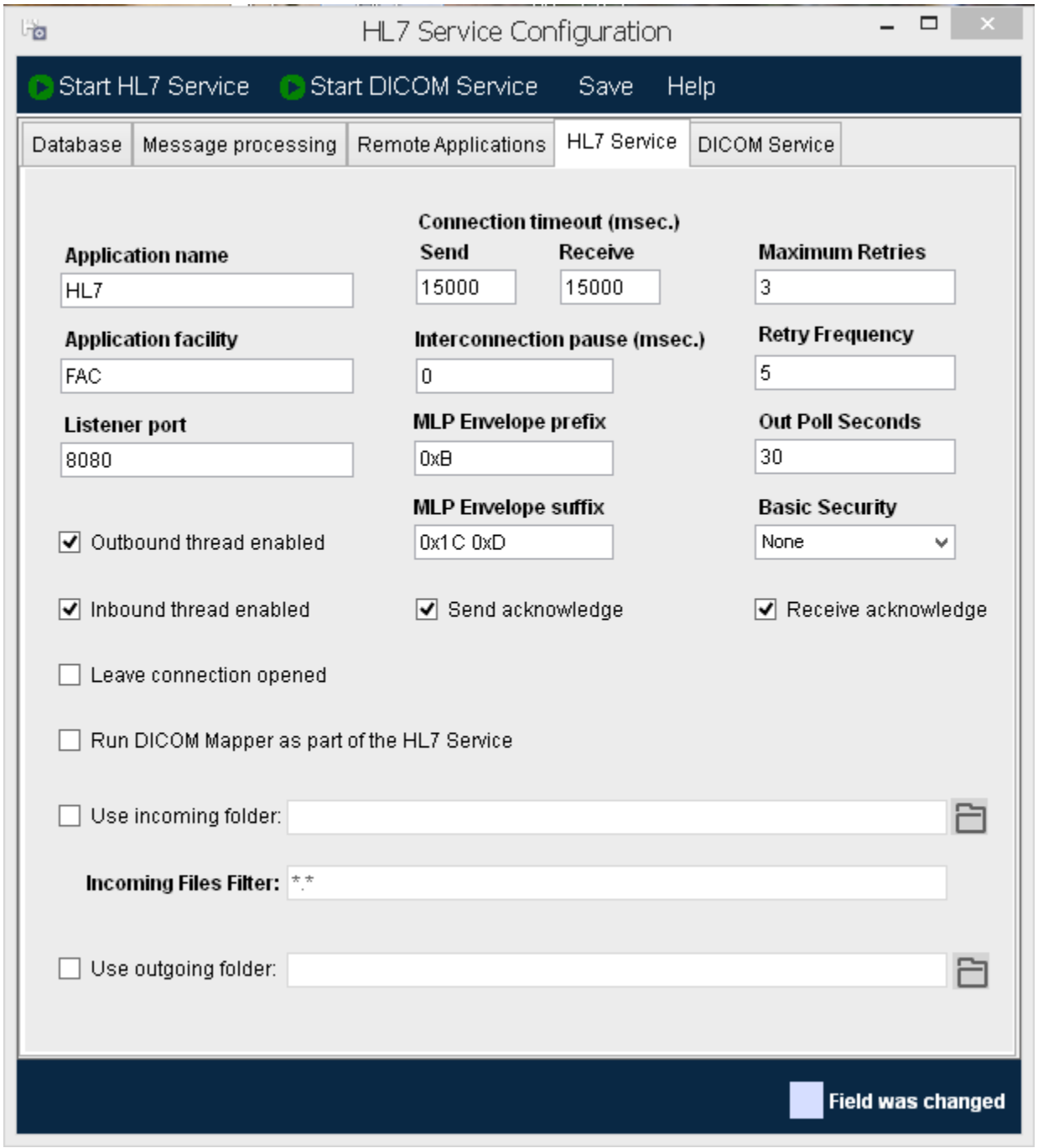
If this flag is selected - value of "Receiving Facility" in MSH segment of incoming message will be used as "Sending Facility" in MSH segment of the ACK message, If not selected - "Application Facility" value will be inserted.

Set Sending Facility in ACK by Receiving Facility from incoming message Check-Box

When checked, direct the kit to use MSH6 field from the inbound message as the value for MSH4 field of the Acknowledge message.

Unmapped messages get a success ACK Check-Box

When checked, expect a HL7 Acknowledge message for every outbound message.



Application Name Text Box

This is the application name of the runtime service (your application). This value is set to MSH 2 (Sending Application) of every outbound messages and expected in MSH 4 (Receiving Application) of inbound messages. This value is used for verification of the messages source and target when Basic Security is set.

Application Facility Text Box

This is the facility name of the runtime service (your application). This value is sent in MSH 3 (Sending Facility) of outbound messages and expected in MSH 5 (Receiving Facility) of inbound messages. This value is used for verification of the messages source and target when Basic Security is set.

Listener Port

This is the network port that the runtime service is using for inbound messages. Make sure that this port is not used by other applications and that it is not blocked by any firewall.

Outbound thread enabled Check-Box

When un-checked, the outgoing messages processing of the HL7 service will not be started.

Inbound thread enabled Check-Box

When un-checked, the inbound messages processing of the HL7 service will not be started.

Run DICOM Mapper as part of the HL7 Service Check-Box

When checked, incoming DICOM files will be processed using defined mapping rules.

Multiple Destinations Checkbox

One HL7 runtime service can integrate with many remote HL7 applications. When checked, configuration of remote applications is enabled. See Multiple Destinations.

Leave connection opened Checkbox

When not checked (default), new connection will be opened for each outgoing message in batch sending. If checked - all messages of the packet will be sent using single connection.

Connection Timeout Text Boxes

These are the connection timeouts for inbound and outbound connections in milliseconds.

Interconnection pause

Period in milliseconds after network connection was closed before check for another connection. Default value is 0 which means no waiting between connections.

MLP Envelope Prefix

This is the hex notation of the MLP envelope prefix characters that the runtime service uses for inbound and outbound messages.

MLP Envelope Suffix

This is the hex notation of the MLP envelope suffix characters that the runtime service uses for inbound and outbound messages.

Send Acknowledge Check-Box

When checked, send HL7 Acknowledge message for every inbound message. This property might be overridden for each remote application.

Receive Acknowledge Check-Box

When checked, HL7 "Success" Acknowledge message will be send even if no mapping was defined for incoming message. This property might be overridden for each remote application.

Maximum Retries

The runtime service will try sending a message this number of times before marking it as an flawed message. Retries will be performed only if the message is syntactically valid but the remote application didn't receive it.

Retry Frequency

This is the number of seconds that the runtime service waits before attempting another retry.

Out Poll Seconds

The outbound messages queue poll period. The runtime service polls the HL7-QUEUE table every this number of seconds and looks for new messages to send.

Basic Security

This flag defines if and how the values in MSH 2, MSH 3, MSH 4, and MSH 5 are used for validating Application Name, Facility Name, Remote Application Name and Remote Facility:

- None – No validation, accept all messages regardless of the values in MSH 2-5
- Sending – Validate Sending – For inbound messages, check that sending application (MSH2) and facility (MSH3) match one of the predefined remote applications.
- Receiving – For inbound messages, check that application name (MSH4) and facility (MSH5) match the predefined local application name and application facility.
- Both – For inbound messages, check both sending and receiving application and facility.

Incoming Folder

This text box specifies the directory that is monitored for incoming messages. The HL7 runtime service will pick every file (or files filtered by incoming file filter) and process it. The service deletes files after reading them. This property might be overridden for each remote application.

Incoming Files Filter Textbox & Checkbox

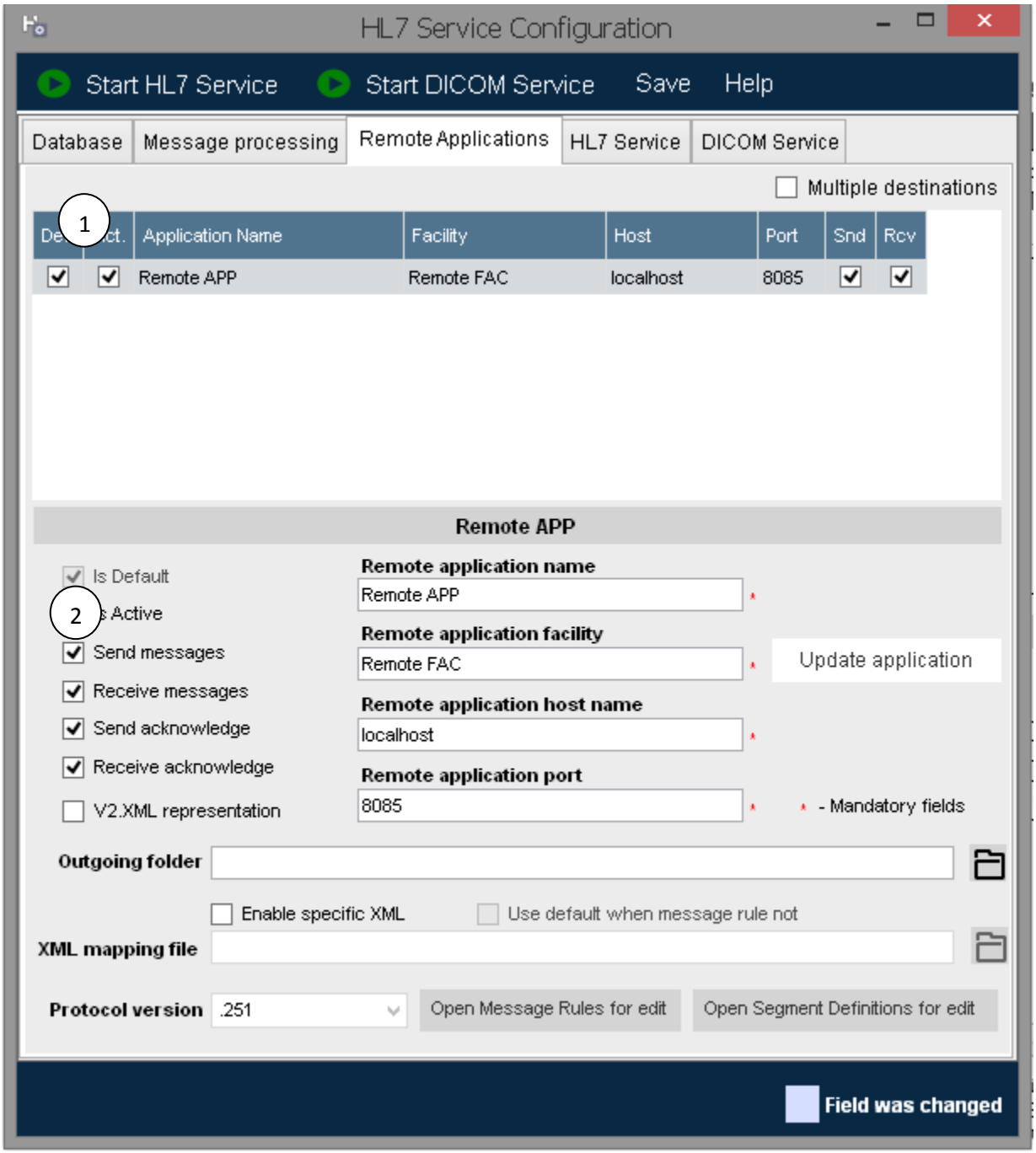
If checkbox is checked - only files suitable for the mask typed in textbox will be proceeded as incoming files.

Outgoing Folder

This text box specifies the directory where the service saves outgoing messages. Message files are named with timestamp and "hl7" suffix.

Remote Application List (1)

The remote application list displays all the applications that the kit is configured to integrate with. Click an application line to highlight it and edit the application information is the edit area.



The Edit Area (2)

The controls in the edit area show the information of the highlighted application in the remote applications list.

Is Active Checkbox

When this checkbox is marked, messages will be sent to and received from the selected application.

By default, every outbound message is sent to all active remote applications, unless the integrator explicitly sets the destination in the queue table entry.

Send Messages Checkbox

When this checkbox is set, messages will be sent to the selected application.

Receive Messages Checkbox

When this checkbox is set, messages received from the selected application will be processed.

Send Acknowledge Check-Box

When checked, send HL7 Acknowledge message for every inbound message.

Receive Acknowledge Check-Box

When checked, HL7 "Success" Acknowledge message will be send even if no mapping was defined for incoming message.

V2.XML representation Checkbox

When checked, send/receive all messages formatted as HL7 V2 XML.

Remote Application Name Text Box

This is the application name of the selected remote application. This value is set in MSH 4 (Sending Application) of outbound messages and expected in MSH 2 (Receiving Application) of inbound messages. This value is used for verification of the messages source and target when Basic Security is set.

Remote Application Facility Text Box

This is the facility name of the selected remote application. This value is set in MSH 5 (Sending facility) of outbound messages and expected in MSH 3 (Receiving facility) of inbound messages. This value is used for verification of the messages source and target when Basic Security is set.

Remote Application Host Name (or IP Address)

This is the host name or IP Address of the selected remote application. The runtime service sends outbound messages intended to the selected application to this network address.

Remote Application Port

This is the network port that the selected remote application uses. The runtime service sends outbound messages to this port of the selected remote application host.

Enable specific XML

Check this checkbox to enable the message configuration version and XML mapping file controls for the currently selected remote application.

Every remote application can have a different flavor of the HL7 protocol message definitions and different mapping rules.

Message configuration version Combo Box

Select an application specific HL7 version that used when parsing and composing HL7 messages. This setting overrides the settings in the Main Values pane.

XML mapping file text box and browse button

An application specific mapping rules to use when parsing and composing HL7 messages. This setting overrides the settings in the Main Values pane.

Use default when message rule not defined

When this checkbox is checked and the application's specific mapping file doesn't include a rule for the event type of a message (received or sent) the HL7 runtime service will look for a rule in the default mapping rules file and use it if found.

Start New Application Button

This button clears the edit area so the user can configure a new remote application.

Save Application Button

This button saves the new application configuration or the changes to an existing application configuration.

DICOM Service

This pane allows you set parameters of DICOM server installed as a part of HL7 Kit (stored in DB In CONFIGURATION table). Also you can add/edit/delete DICOM applications associated with the server.

Stop DICOM server before changing any value (if it was started), edit all required values, click "Update" and start server to apply changes.

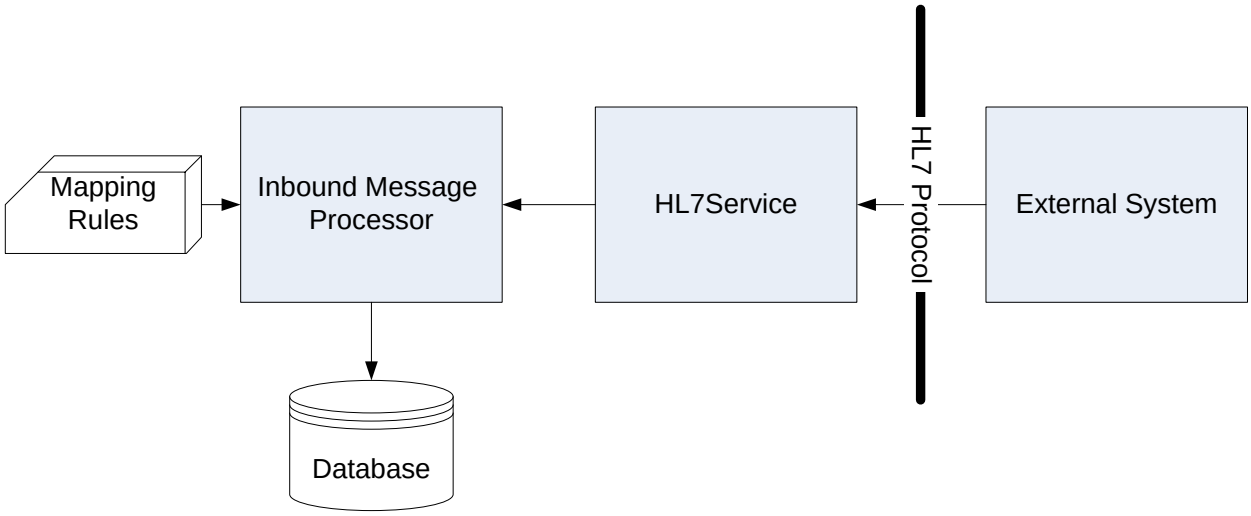
To add new DICOM application - enter its parameter into empty fields (if instead of "Add" there is "Update" button - it means you selected existing application earlier. Click "Clear" to return to new application) and click "Add".

To edit existing DICOM application - just select the row in the grid (application parameters will be loaded into text boxes and second button caption will be set to "Update"), change parameters and click "Update".
Click "Echo" to test application settings.

The same mechanism is relevant to editing of the configuration values.

WARNING: Please be very careful changing configuration!!!

Inbound Messages Processing

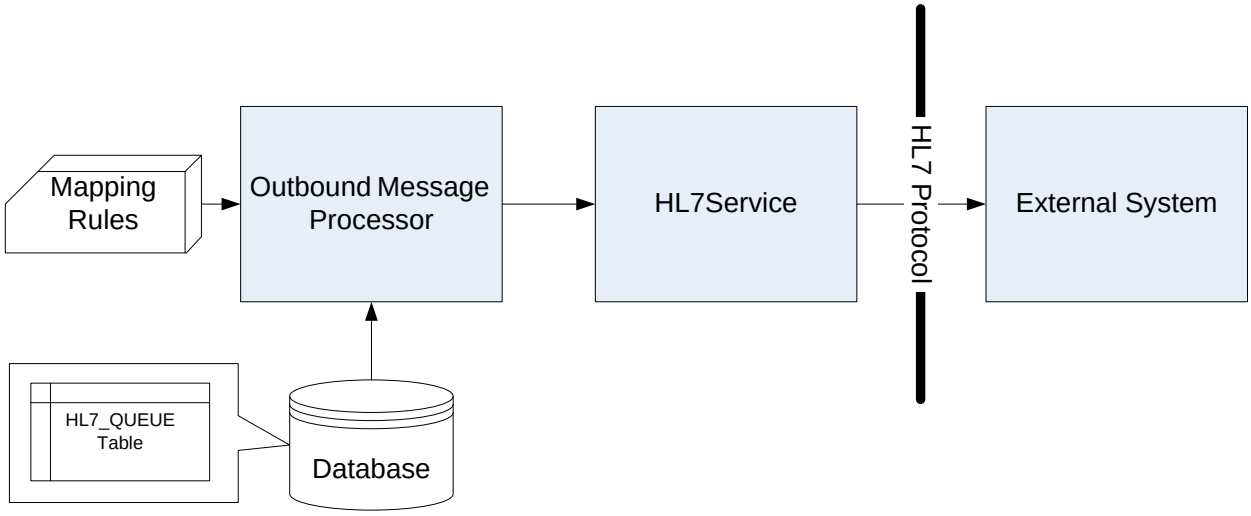


When an external application sends a HL7 message to the HL7 Service, the message is processed according to the inbound mapping rules defined for its event type and records are inserted into the appropriate database tables.

The processing of the message is done based on the message structure definition and the foreign keys (relationships) between the database tables.

The hierarchy of the HL7 message structure should match the hierarchy of the database tables.

Outbound Message Processing



Outbound message processing is triggered by inserting or updating a record in the HL7_QUEUE database table. The HL7 service polls the HL7_QUEUE according to the configured Out Poll Seconds. If it finds record with status ready, it activates the outbound mapping rules and sends the message to the remote system.

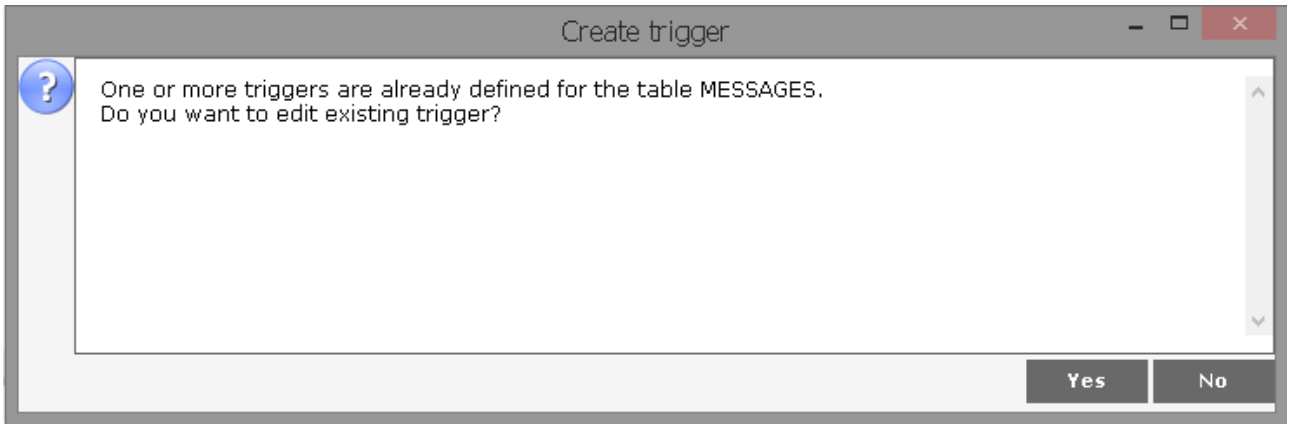
The outbound messages queue

The HL7_QUEUE Table

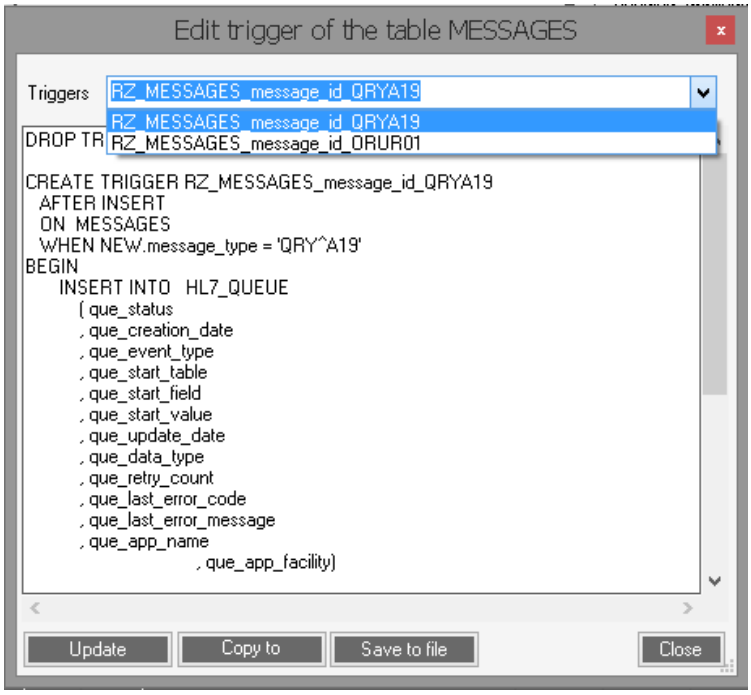
HL7Kit runtime service uses the outbound queue for sending outgoing messages.

When defining an outbound rule, the mapping application can create a trigger on the database tables participating in the rule so that when new data is inserted, a new queue record is created.

Click on "Create trigger" row under "Mapping" menu - application checks whether any trigger exists for start table already. If exists - follow warning will displayed:

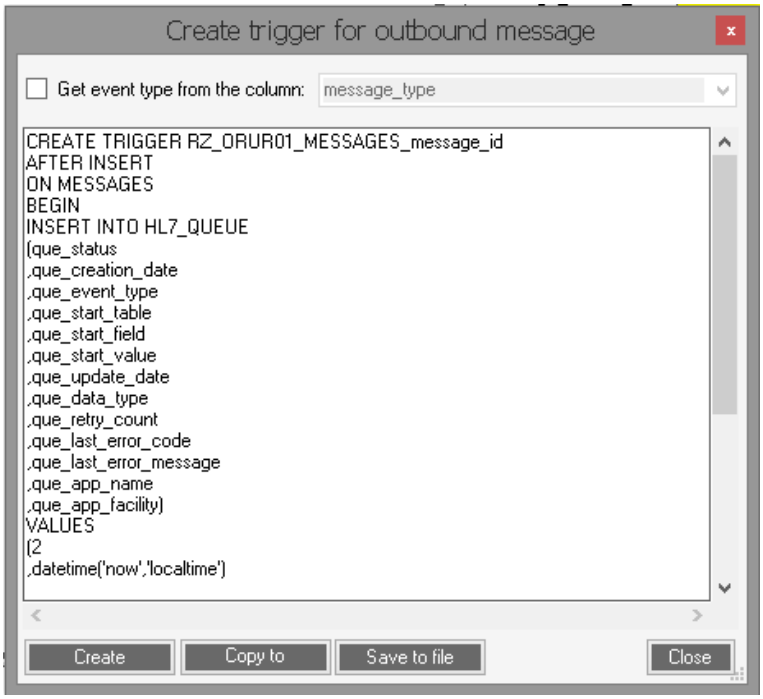


If "Yes" selected - "Edit trigger" screen will be opened:



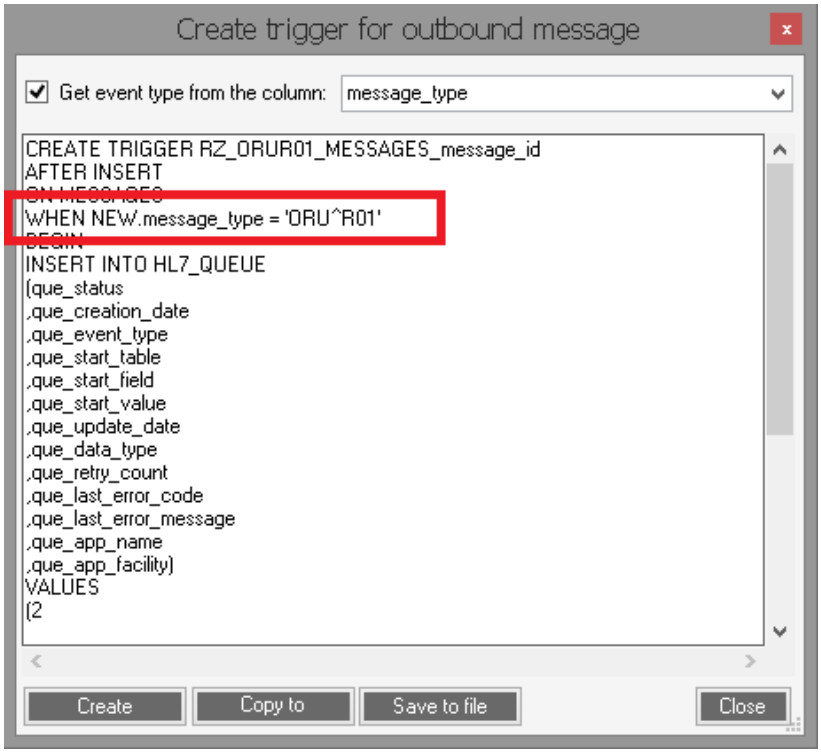
Select trigger from the list, change it's "CREATE" clause and update trigger in DB.

If user selects "No" or table doesn't have any trigger yet - "Create trigger" screen will be opened:

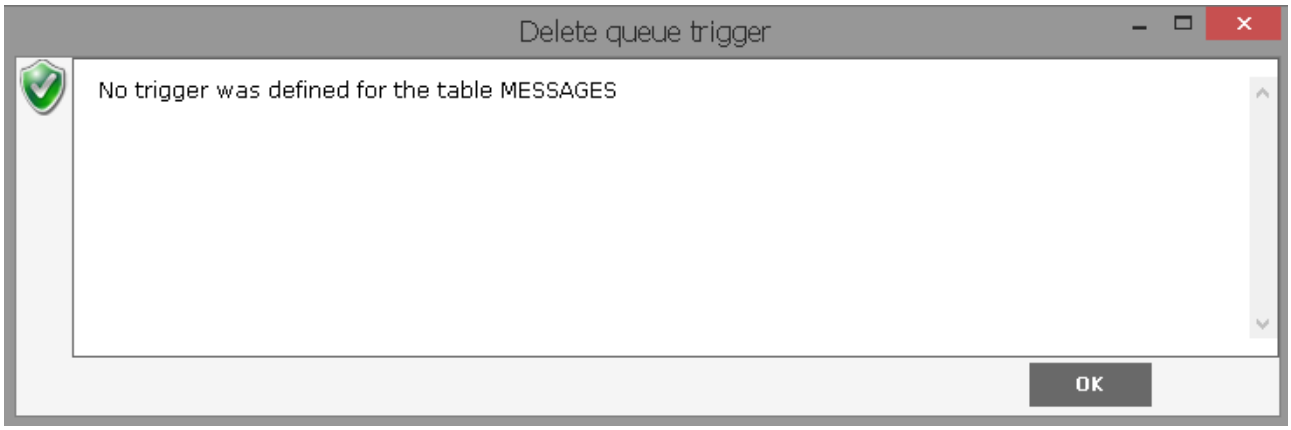


By default application tries to create trigger which will fire on each inserted row, but there is option to build trigger that checks inserted message type.

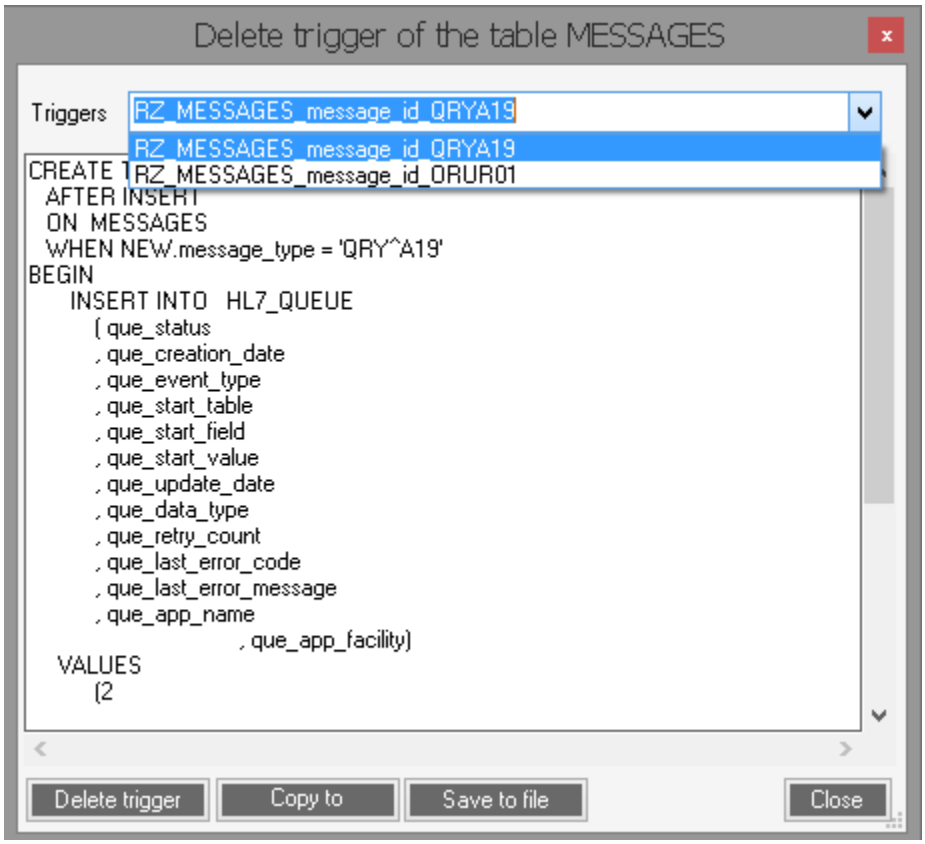
For that check "Get event type from the column" box and select required DB column (for example if messages of several types might be inserted to start table but only part of them must be start point of the queue):



To delete trigger from DB click on "Delete trigger" row under "Mapping" menu - application checks whether any trigger exists for start table already. If not - follow message will displayed:



If one or more triggers exist - "Delete trigger" screen will be opened:



Select trigger from the list and delete it in DB.

NOTE: Built-in Data Base already contains trigger on MESSAGES tables for event types `QRY^A19` and `ORU^R01`

Alternatively, the queue record can be inserted programmatically by your application.

Column Name	Data Type	Allow Nulls	Description
que_id	int	No	The id of this queue entry. Referenced in log records.
que_status	int	No	The status of the queue entry: Inserted = 1, Ready = 2, Error = 3, Processing = 4, Done = 5, Failed = 6
que_creation_date	datetime	Yes	The date and time the record was

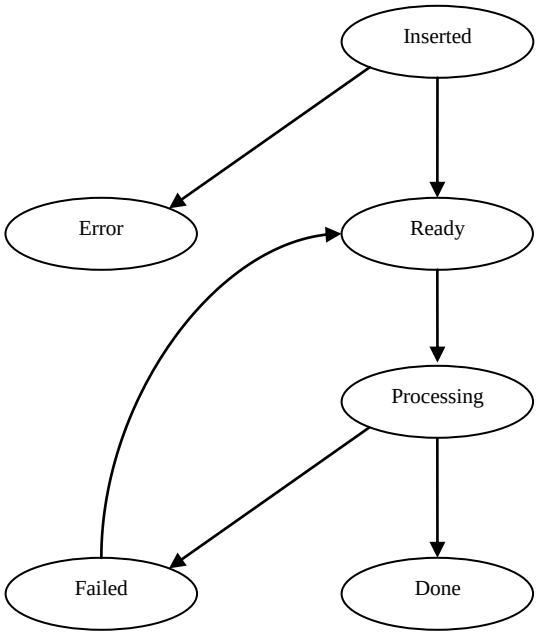
			created on.
que_event_type	nchar(10)	Yes	The HL7 Event type (e.g. ADTA01) that should be sent. This value must match one of the rules in the mapping definition file.
que_start_table	nvarchar(100)	Yes	The table name that is the root table of the outbound message rule.
que_start_field	nvarchar(100)	Yes	The primary key or ID field that identifies the message in the start table.
que_start_value	int	Yes	The id value of the record in the start table that holds the message data to be sent.
que_update_date	datetime	Yes	The last time the queue record was updated. Automatically updated by the runtime service.
que_data_type	nchar(10)	Yes	Not in use.
que_retry_count	int	Yes	Send retries counter.
que_last_error_code	int	Yes	The last retry error code.
que_last_error_message	nvarchar(MAX)	Yes	The last retry error message
que_app_name	nvarchar(100)	Yes	The remote application Name the message will be sent to. If inserted with NULL value, the runtime service will duplicate the record for all active remote application and manage one queue record for each destination.
que_app_facility	nvarchar(100)	Yes	The remote application facility. See que_app_name.

Queue Statuses

The HL7_QUEUE status column can have the following values:

Status	Name	Meaning
1	Inserted	This is the initial status, when a new record is created.
2	Ready	The message is ready for sending. The runtime service picks up records with status ready and sends them.
3	Error	A final state for messages that could not be processed or sent.
4	Processing	Intermittent state while the message is being sent.
5	Done	The message has been sent successfully.
6	Failed	Failed to send message. The service will retry sending failed messages until the retry count is exhausted.

Queue State machine



Sending messages to multiple destinations

If the destination application in the HL7_QUEUE record is NULL, the runtime service duplicates the record for every active remote application that it is configured to send messages to. If the destination application is not NULL, the message is sent to the specified application.

Queries

HL7 Queries are not supported in this version of HL7Kit.

Multiple Destinations

One HL7Kit Pro runtime service can integrate with many remote HL7 Applications. The remote applications are configured using the Remote Application List (1).

Every remote application can be configured for sending, receiving or sending and receiving messages.

When a new message is generated, the runtime service sends it to every active application in the list that is configured to send messages to.

When a new message is recived and the basic security checkbox is set, the runtime service validates the source application name with the active applications that are configured to receive application from.

Multiple Sources

If an application's specific mapping rules and HL7 version are configured, the HL7 runtime service will first parse the MSH segment and use the rules and version that were configured to that application.

Maintenance

Monitoring and Logging

HL7Kit uses the built in windows event log for logging. This makes it very easy to use advanced monitoring tools built into your windows operating system.

Both HL7Receiver and the HL7 Runtime Service write into the RZHL7Pro log file.[1]

If HL7 Service receives invalid input, for example XML document which is not of XMLV2 version, incoming buffer will be stored as XML file in the InvalidInput folder located in your installation folder (C:\HL7Kit by default). Name of the stored file is like "20170718054120159.xml".

NOTE: HL7Kit's evaluation copy does not write into the event log.

HL7 Version Configuration

Message Rules

The message and segment rule files are only slightly different than the HL7 demonstration implementation that is used for example in the IHE test tools. The major difference is the ability to define hierarchical structure using aliases and thus making the definition for XML structure as well.

Editing the message rules

```
ACK:MSH MSA [ ERR ]
QRYQ01:MSH QRD [QRF] [DSC]
DSRQ03:MSH QRD [QRF]{DSP}[DSC]
ADTA01:MSH EVN \ PID <NK1> PV1 [PV2] / patient
<OBX><AL1><DG1><PR1><GT1> <IN1 [IN2] [IN3]> [ACC] [UB1] [UB2]
ADTA03:ADTA01
```

Figure 1: Message Rules Definition File. A hierarchy group alias is highlighted.

The message rules configuration file msgRules.251 is using the common HL7 message definition notation. This notation is very similar but not identical to BNF form.

[1] Starting in version 1.4 the source for the log file is RZHL7Pro. Versions 1.3 and older used the source RZHL7.

Every line in the file represents one message definition.

The first element, preceding the colon sign: is the event type without the ^ separator.

When the event type is followed by a semicolon and another event type, it means that the structure of the message defined by the event type on the left, is identical to the one with event type on the right. In the example above - ADT^A03 is defined to have an identical structure as ADT^A01.

When the event type is followed by a colon, a list of segment names and special grouping characters follow.

The special grouping characters are:

Open group	Close group	Meaning	Mandatory	Repeating
{	}	1 or more occurrences	Yes	Yes
(	)	Exactly one of	Yes	No
\	/	Exactly one	Yes	No
[	]	0 or 1 occurrences	No	No
<	>	0 or more occurrences	No	Yes

Aliases

Elements that are not special characters or a valid segment names are regarded as aliases. Aliases are used to group segments into entities. For example, in the ADT^A01 an alias called 'patient' is defined. Note that the alias name comes AFTER the segments group.

It is recommended to use aliases because the common notation that is used for HL7 v2.x messages uses tabulation to represent nesting - which may be very confusing. A good practice is to add a meaningful alias for every nesting level in the message structure. The exactly one grouping characters \ and / are unique to HL7Kit Pro and were added to enable aliasing for the top most level of the message.

NOTE: If there is group of segments which is present in the message more than one time - you have to create alias for this group otherwise message cannot be processed properly. For example:

ORMO01:MSH <NTE> [PID <NTE> [PV1] [AL1]] { ORC [OBR <NTE> <DG1> <OBX<NTE>> Observation][BLG] [ZDS]} Order

There might be few groups of <OBX<NTE>> in the order so it must enclosed by alias :

<OBX<NTE>> Observation

Editing the segment definitions

The segment definitions can be edited using a simple text editor like notepad.

Every segment is defined as a list of fields.

Field Definition

```
2 ID REQ NO_RPT AcknowledgementCode     /* 1  Acknowledgement Code */
20 ST REQ NO_RPT MessageControlID         /* 2  Message Control ID */
80 ST OPT NO_RPT TextMessage         /* 3  Text Message */
15 NM OPT NO_RPT ExpectedSequenceNumber /* 4 Expected Sequence Number */
1 ID OPT NO RPT DelayedAckType       /* 5  Delayed Ack Type */
```

A field is defined using the following attributes:

1	Length	A number stating the maximum data length of the field
2	Data type	A two-character field type code that defines the data type of the field. Valid field types are: `ST, TX, FT, NM, DT, TM, TS, PN, TN, AD, ID, SI, CM, CK, CN, CQ, CE`
3	Mandatory	A flag that defines if this field is mandatory or can be left blank: • OPT – Optional • REQ - Required
4	Repetition	A flag that define the repetition of the field • NO_RPT – One value • NO_MAX – Multiple values
5	Field name	A string that defines the field name for XML representation. The field name must not include any spaces. CaMeL notation is common.
6	Comment	Free text for documentation enclosed by /* */

Creating new configurations

Configuration files are stored in the application folder. The configuration is stored in two files called:

- msgRules.251
- segDefs.251

To create a new configuration simply copy an existing configuration and change its suffix.

Examples

This section includes three detailed example projects, one for processing incoming ADT^A01 (Patient Admition) messages, one for sending outgoing ORU^R01 (Basic Report) messages and one for creating and sending a QRY^A19 (Query(message.

Every example contains:

1. Readme file with detailed step by step instructions
2. MSSQL/MySQL/SQLite Scripts for creating the databases and inserting data into the tables
3. HL7Kit Mapping Definition file

4. HL7 message file to use in the example

The example projects are stored as HL7_COMBINED_EXAMPLES.zip file in C:\HL7Kit\Examples folder .

C:\HL7Kit\Examples folder also contains few example HL7 message as separate files (ADRA19.hl7, ADTA01.hl7, ORM_Example.hl7, ORUR01.hl7, QRYA19.hl7 and SIUS12.hl7)

Annexes

Concepts and Vocabulary

HL7 File

HL7 File is a file that stores a HL7 message data. HL7 File is not a text file. Every segment in the file is separated by a segment separator character that is usually 0xD. **Do not edit HL7 Files using a text editor** such as Notepad as it will replace the segment separators with newlines.

LLP

The Lower Layer Protocol (LLP), sometimes referred to as the Minimal Lower Layer Protocol (MLLP), is the standard for transmitting HL7 messages via TCP/IP. Since TCP/IP is a continuous stream of bytes. The wrapping protocol (i.e. headers and trailers) is required for communications code to recognize the start and the end of each message. The Lower Layer Protocol is the most common mechanism for sending unencrypted HL7 via TCP/IP over a local area network, such as those found in a hospital.

An HL7 message must be wrapped using a header and trailer (also called footer) to signify the beginning and the end of a message. These headers and footers are typically non-printable characters that would not be shown in the actual content of an HL7 message.

MLLP

MLLP is an acronym for Minimal Lower Level Protocol. See LLP.

ACK

The HL7 mechanism used for acknowledgment of delivery and/or processing. When a system receives a message, it is expected to respond with the matching ACK message. The ACK message carries a status code.

The handling of ACKs can be configured - expect or prevent ACK responses, and send ACKs or prevent sending them.

MWL

MWL is an acronym for Modality Worklist

MPPS

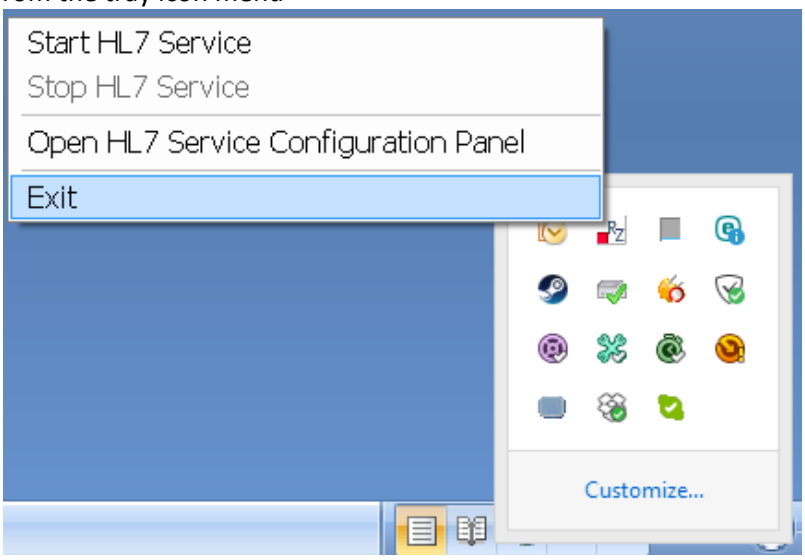
MPPS is an acronym for Modality Performed Procedure Step.

Upgrade Instructions

A. Before starting setup.exe:

1. Stop HL7 Service and DICOM Server from the HL7 Service Configuration Panel

2. Stop all running HL7 applications including HL7 Service Configuration Panel (if minimized - exit from the tray icon menu



)

3. Create some backup folder on your hard disk

4. Copy next files from HL7Kit folder (default is C:\HL7) to this backup folder
   a. Old config files:
      - HL7Mapper.exe.config,
      - HL7Receiver.exe.config,
      - HL7Sender.exe.config,
      - HL7Service.exe.config,
      - ModalityWorklistSCU.exe.config,
      - WorklistManager.exe.config,
      - DICOMMapper.exe.config if exists,
      - DicomServer.ini
   b. Definitions files:
      - msgRules.*,
      - segDefs.*,
      - hl7types.xml,
      - ModalityCodes.txt,
      - ProcedureDescription.txt
   c. built_in_mapping.HL7KIT
   d. DICOMDefinitions.xml
   e. DicomServer.db

C. Important: Un-install HL7 Kit from the Control Panel

D. Run setup.exe and follow installation instructions (if you didn't use built-in DB in previous version you might not connect to any DB after installation complete)

E. Stop HL7 Service and DICOM Server from the HL7 Service Configuration Panel if they are started after installation. Exit HL7 Service Configuration Panel

F. Copy all files mentioned in A.4 from the installation folder (default is C:\HL7) to some reserve place (other than previously created backup folder!) in case you need clean configuration.

G. Copy all files from backup folder to installation folder

H. Open HL7 Service Configuration Panel, start HL7 Service and DICOM Server