User Manual

DVTk Storage SCU emulator 5.1.0

A DVTk based tool

Document version 1.5
September 13, 2018

Table of Contents

1 Introduction

1.1 Revision History

Version	Date	Description
1.0	October 2, 2007	First version of this document, describing the functionality of DVTk Storage SCU Emulator version R3.0.0.
1.1	March 13,2008	Releasing .NET2.0 version.
1.2	June 16,2009	Releasing 3.2.0 version based on HLI.
1.3	October 08, 2009	Releasing 3.3.0 version for Storage Files functionality.
1.4	May 16, 2017	Releasing version 5.0.0 based on .NET Framework 4.
1.5	September 13, 2018	Added new transfer syntaxes

1.2 General

The DVTk Storage SCU emulator application is used to transfer DICOM part-10 files to a remote DICOM Storage SCP system. It can also be used to ask the SCP node for Storage commitment of the transferred dataset. The waiting time for receiving the storage commitment response is configurable in the SCU emulator.
The selection of data for export can be done by:

- Single/multiple media or DICOM file selection
- Directory selection (data present in all subdirectories will be included as well)
- DICOMDIR selection

1.3 System Requirements

1.3.1 Operating system

The following operating systems are supported:

- Windows XP
- Windows 7
- Windows 8
- Windows 10

1.3.2 Software requirements

The following packages are required for the installation of the software packages:

- Microsoft .NET framework 4

The Microsoft .NET framework software package is included in the installer of the **DVTk Storage SCU emulator** tool.

See: http://www.dvtk.org for new versions and features.

2 Software installation

All the steps of the installation process are controlled by the DVTk Storage SCU emulator installer package. During the installation process, the installer will check if the Microsoft .NET Framework 4 is already installed on the system. If present, this step of the installation process will be skipped.

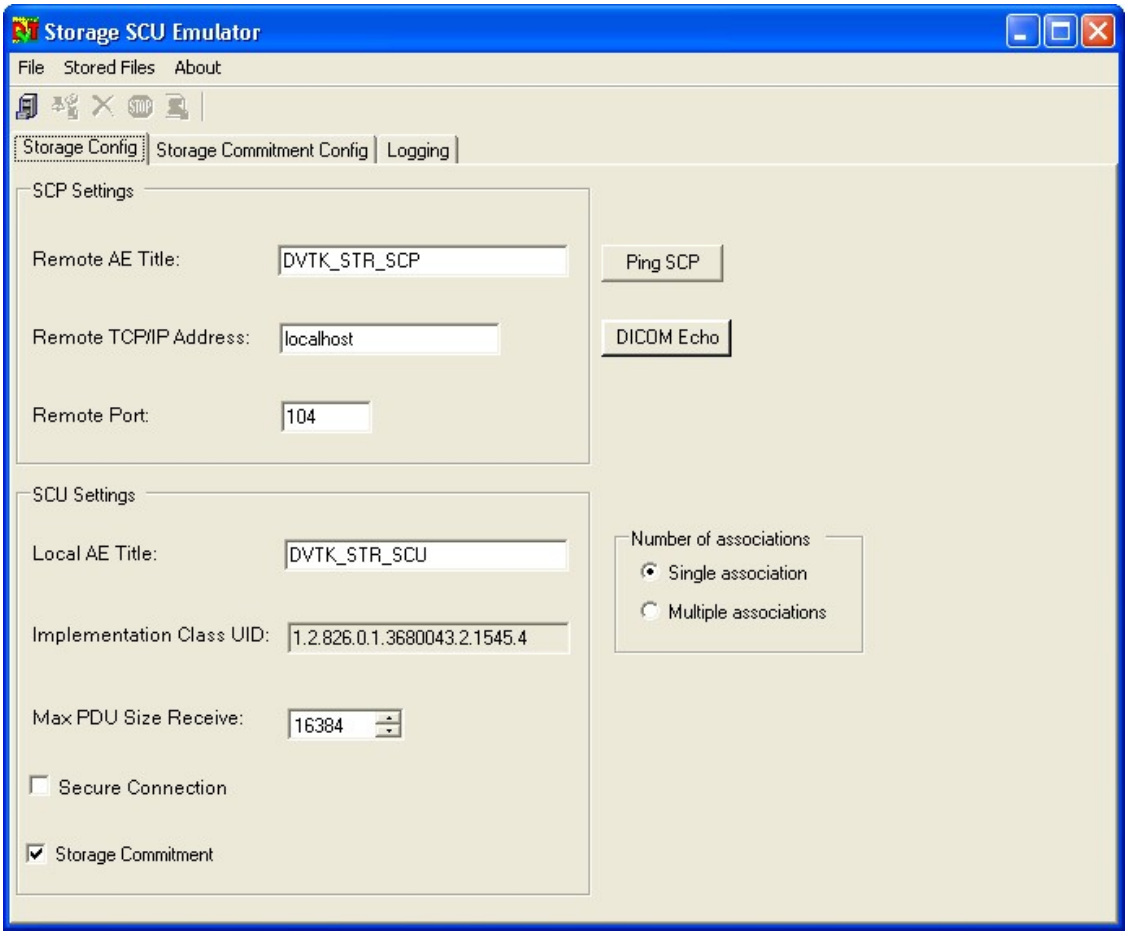
2.1 Installation of DVTk Storage SCU emulator software

1. Download or copy the file **Storage SCU.zip** to a temp directory on the PC.
2. Double click with the left mouse button on the Storage SCU.zip file and extract the file "Storage SCU Emulator.exe" to the temp directory.
3. Start the installation procedure by double clicking with the left mouse button on the file Storage SCU Emulator.exe
4. In the window " Microsoft .NET Framework 1.1 setup" select " I agree" to accept the license agreement. Press "install" to start the installation process. At the end of the installation procedure press "OK".The Microsoft .NET Framework R1.1 software is installed.
   After the .NET Framework software is installed, the installer package continues with the installation of the Storage SCU Emulator software
5. Follow the instructions in the installer window and accept the license agreement.
6. After the button "install" is pressed, the installation will start. At the end, press the "Finish "button. The DVTk Storage SCU Emulator application is installed and ready for use.

In windows "All programs" there is an entry created "DVTK". When selecting DVTK, a submenu with all installed DVTK applications will be opened. From this submenu the Storage SCU Emulator tool can be selected and started. There is also a shortcut created on the desktop.

3 Functional description of the SCU emulator

In the screen capture below, the User Interface of the Storage SCU emulator tool is shown:



The Storage SCU emulator start up window contains the following sections:

3.1 Menu bar

From the Menu bar the following file options can be selected to control the Storage SCU emulator:

File:

- Export DICOM data selection
  - ➢ select Source Directory
  - ➢ select DICOM files
  - ➢ select DICOMDIR

- Send storage commit

- Config File
  - Load
  - Save As

- Exit

Stored Files:

- Explore Validation Results…

- Options…

About:

- About Emulator

3.2 Tool bar

The tool bar contains the following short keys to control the Storage SCU emulator process:

Export Dicom data 

Send Storage Commit  (only visible after the start of the storage emulator)

Abort Export

Stop Commit SCP 

Display validation result 

3.3 Information screen selection

The following screens can be selected for display

- Storage configuration

- Storage Commitment configuration

- Log information

- Validation results (this screen is only present in case the toolbar option “Display validation result” is enabled.

3.3.1 DICOM Storage configuration

3.3.1.1 DICOM Storage SCP node configuration

In the SCP setting of the Storage configuration, the parameters for the communication with the SCP node have to be configured. The configuration parameters include:

- AE title of storage SCP node
- IP address op Storage SCP node
- Port number of the Storage SCP node.

After the configuration parameters have been entered, the test buttons "**Ping SCP**" and "**DICOM Echo**" can be used to check if both the network and DICOM communication are working correctly.

3.3.1.2 DICOM Storage SCU node configuration

In the SCU settings of the Storage configuration window, the parameters used by the SCU node (= DVTk Storage Emulator) for the DICOM communication with the SCP node can be configured. The SCU parameters include:

- SCU AE title
- Max PDU size proposed by the SCU to the SCP node
- Secure connection
- Perform Storage commit operation

3.3.2 DICOM Storage commitment configuration

After the transfer of DICOM data to the Storage SCP node is finished, the Storage SCU emulator can request the SCP for commitment of the transferred data.
This storage commitment request is created by the Storage SCU emulator after pressing the "Send Storage Commit  " button in the toolbar or by selecting the "send storage Commit" function from the File menu.

By pressing the tab "Storage Commitment Config " in the Storage SCU emulator window, the configuration screen for the Storage Commitment operation as shown below.

3.3.2.1 DICOM Storage Commitment SCP node configuration

In the SCP setting of the Storage commitment configuration, the parameters for the communication with the SCP node (Archive) have to be configured.

The configuration parameters include:

- AE title of storage SCP node
- IP address op Storage SCP node
- Port number of the Storage SCP node.

After the configuration parameters have been entered, the test buttons "**Ping SCP**" and "**DICOM Echo**" can be used to check if both the network and DICOM communication works correctly.

3.3.2.2 DICOM Storage Commitment SCU node configuration

In the SCU settings of the Storage Commitment configuration window, the parameters used by the SCU node (= DVTk Storage Emulator) for the DICOM communication with the SCP node has to be configured.
The SCU parameters include:

- **Local AE title.**
  This is the AE title used by the Storage SCU emulator to listen for incoming Storage commitment messages.

- **Local TCP/IP address** (= this is an informative field which shows the IP address of the SCU emulator. The contents cannot be changed)

- **Listen port**
  This is the TCP port used by the Storage SCU emulator to listen for incoming Storage commitment messages

- **Commit max reply waiting time (sec)**
  This is the time that the SCU (= Storage SCU Emulator) keeps the Storage Commitment Association open after sending the N-ACTION-RQ message to the SCP. By keeping the association open, the SCP node is able to send the N-EVENT-REPORT message back in the same association (=synchronous mode). It has to be checked with the Archive which time delay has to be configured for synchronous operation
  For synchronous operation, the following waiting time values can be entered:

Waiting time value	**Meaning**
0	waiting time = infinite
1-n	waiting time = n seconds

  For asynchronous mode, the archive opens a new association to send the N-EVENT-REPORT message to the SCU node. In this case the waiting time has to be set to the value "-1" which forces the Storage SCU emulator to

immediately close the association for Storage Commitment after the N-ACTION-RQ message was sent.

Remark: when the cursor is positioned on the green question mark  , a list of suggested values for the parameter "Commit max reply waiting time" is displayed.

- **Max PDU size**
  This is the Max PDU size proposed by the Storage Emulator SCU application to the SCP node. Default value is set to 16k.

  Remark: all configuration settings for Storage and Storage commitment operation can be saved by using the File menu option " Config File, Save As". The system writes the config settings into an xml file in a user defined location. With the "Config File, Load" function, the configuration file can be imported into the Storage SCU emulator again.

4 Example of a Storage SCU emulator operation

In this chapter all required steps for configuring and running the Storage SCU emulator are discussed.
In this example a single DICOM file will be sent to a DICOM SCP node that supports both the Storage and Storage Commitment SOP class.
The SCP node sends the Storage Commitment N-EVENT-REPORT in a separate association to the SCU node.

4.1 Configure the emulator

The SCP node uses the following settings for the DICOM communication:

Storage SCP:
AE title = DVTK_STR_SCP
Port = 104
IP = localhost

Storage Commitment SCP:
AE title = DVTK_STRC_SCP
Port = 105
IP = localhost

The Storage SCU emulator uses the settings below for communication with the SCP:

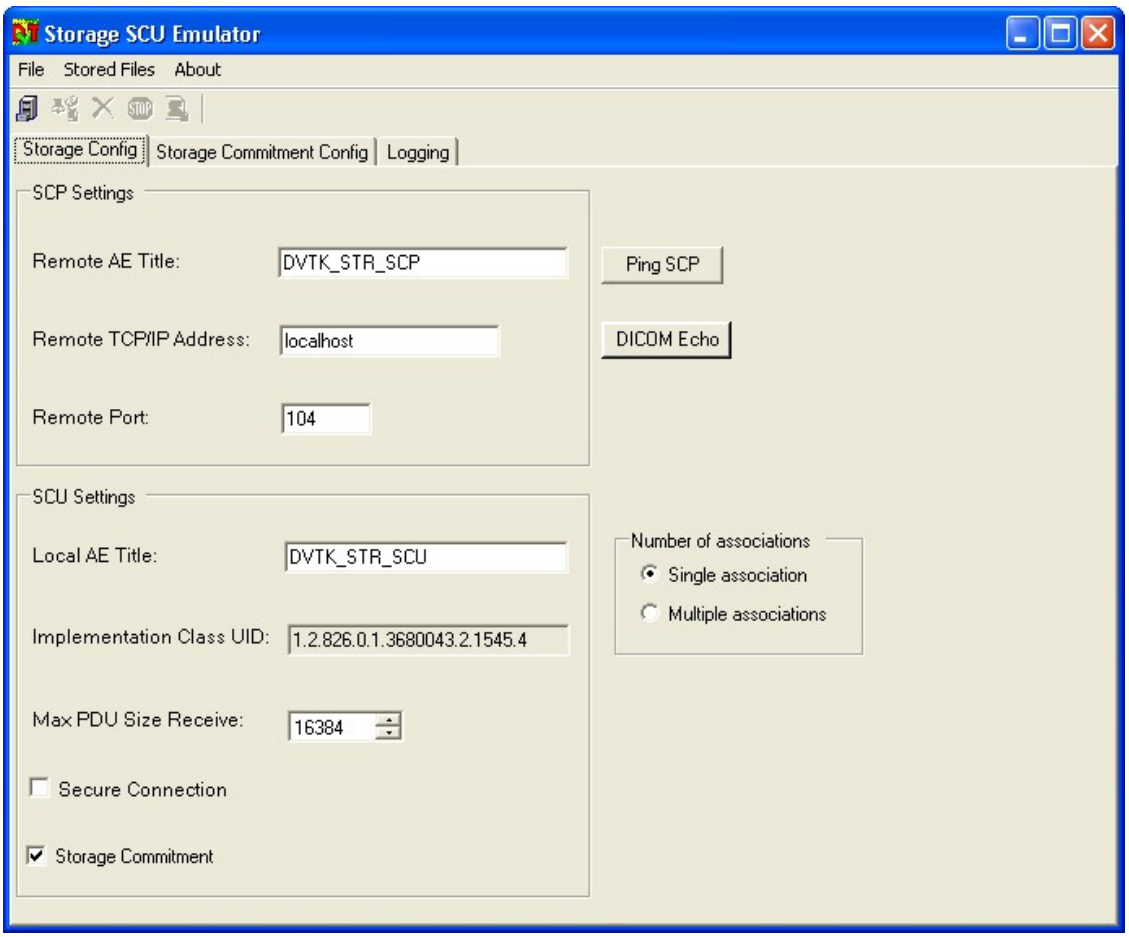
Storage SCU
AE title = DVTK_STR_SCU

Max PDU = 16384

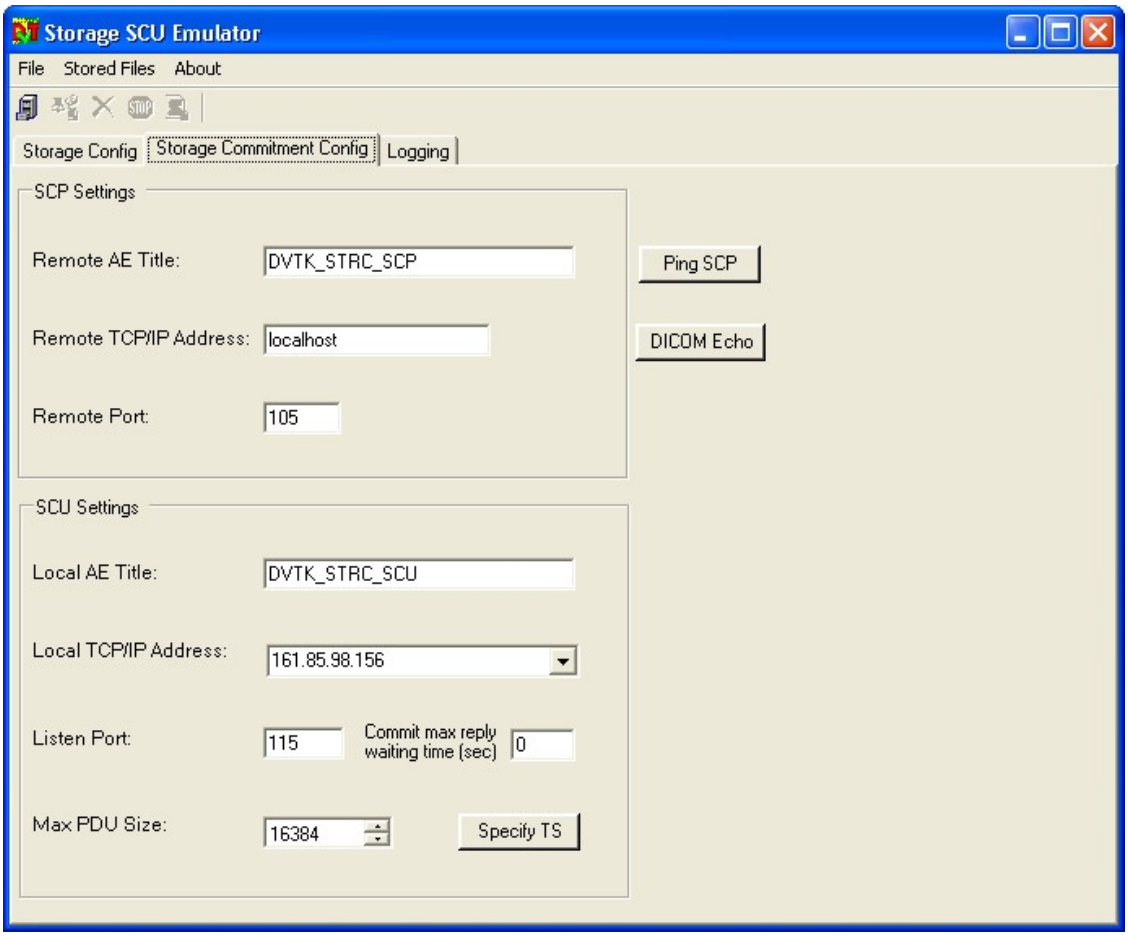
Storage Commitment SCU

AE title = DVTK_STRC_SCU
Max PDU = 16384
Port = 105
IP address = 161.85.98.156

The screen captures below shows how the required configuration data is entered into the Storage SCU emulator tool.



Configuration of the DICOM Storage process

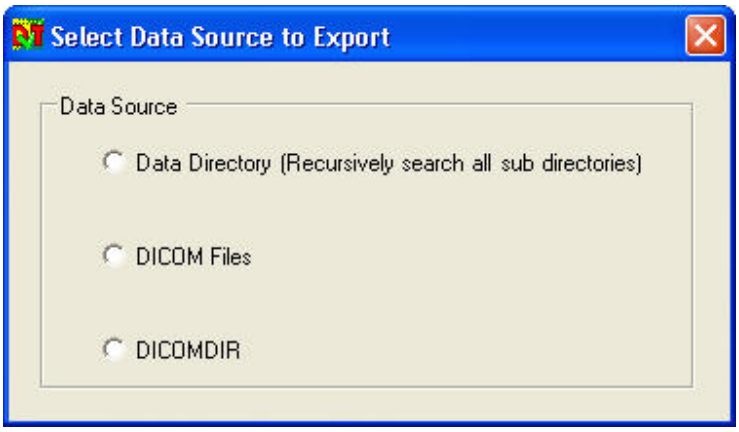


Configuration of the DICOM Storage commitment process

Because the remote DICOM SCP system opens always a new association for sending the storage commitment N-EVENT-REPORT message, the value for “Commit max reply waiting time (sec)” is set to -1.

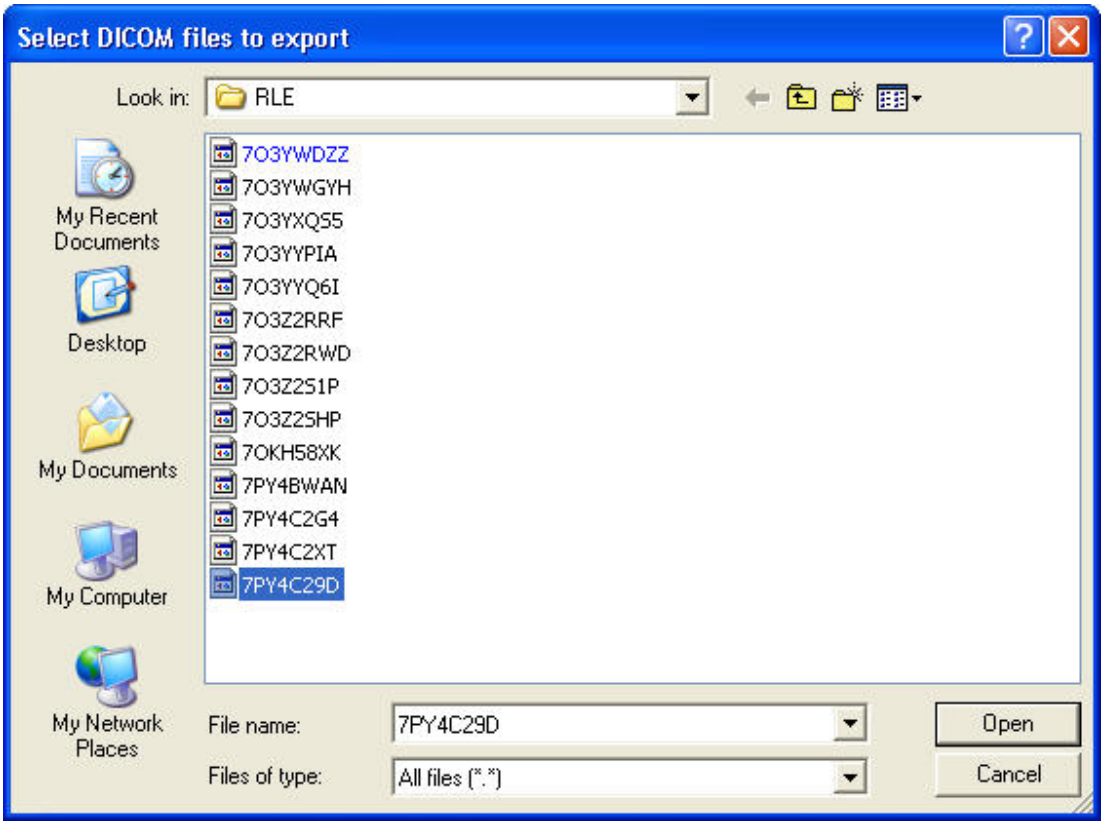
4.2 Selection of data for export

After successful configuration, the DICOM data for export has to be selected by pressing the “Export Dicom data  ” button.
The following window is displayed.



From this window, the source of the DICOM data can be selected.
In our example, we want to export a single DICOM file. Therefore the option “DICOM Files” has to be used.

From the window “Select DICOM Files to export” the source directory of the DICOM file has to be selected. Next step is to select 1 file from this directory.

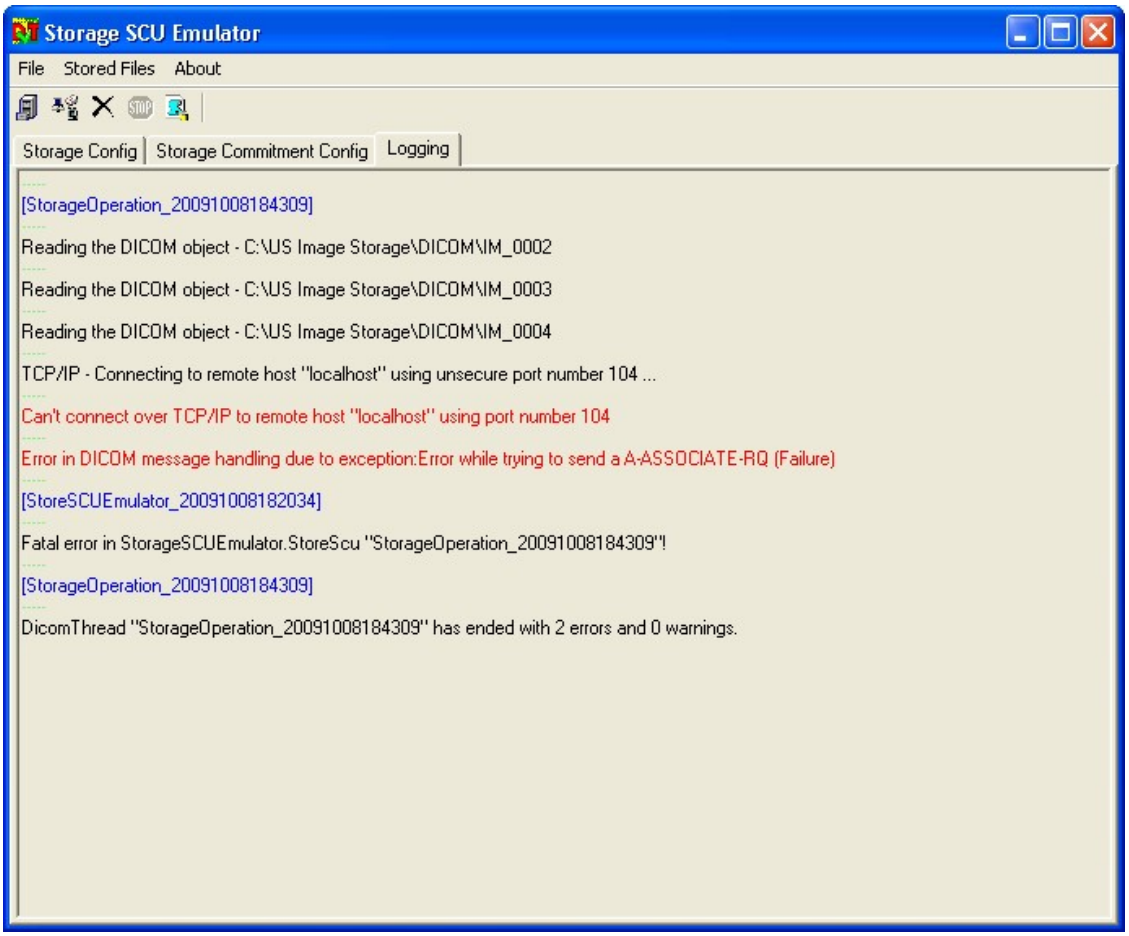


In case results information is present in the Storage SCU emulator from a previous session, the system comes with the question if a backup has to be made for this result data. (Results data is automatically saved by the emulator in the configured directory)

4.3 Sending the selected data

After this, the Storage SCU emulator starts to read the contents of the selected DICOM file. From the file, the information about the DICOM SOP class UID and the transfer syntax is extracted. In our example, the selected DICOM file is an Ultrasound Single Frame image (with SOP class UID 1.2.840.10008.5.1.4.1.1.6.1) with RLE lossless transfer syntax (1.2.840.10008.1.2.5)
Remark: the transfer syntax is not configurable in the Storage SCU emulator. Conversion of transfer syntaxes is not supported by the emulator tool.

With the DICOM SOP class UID and the transfer syntax read from the DICOM file, the emulator opens a DICOM association to the Storage SCP node and proposes the combination of DICOM SOP class and transfer syntax. When supported by the SCP, the Storage SCU emulator transfers the selected DICOM file to the SCP and closes the association afterwards. (See screen capture below)



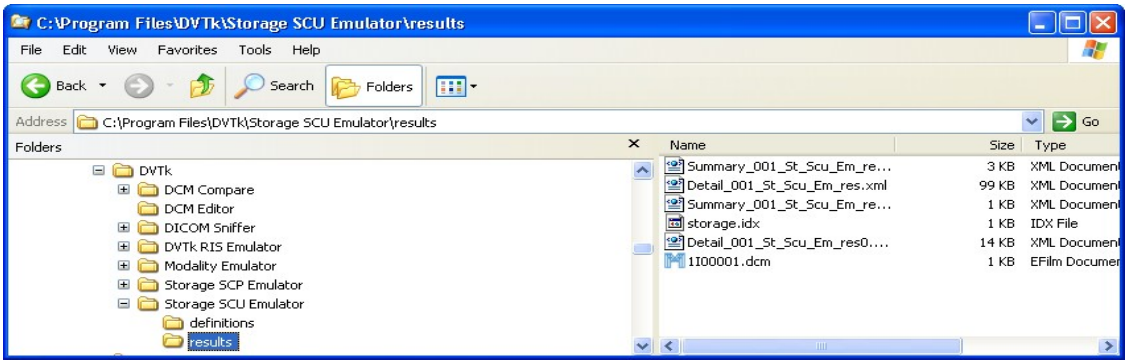
4.4 Sending storage commitment request

For sending a storage commitment request to the SCP node, the button “Send Storage Commit  ” has to be pressed (or selected from the File menu)
This will cause the emulator to open a new association to the SCP node for sending the storage commitment N-ACTION_RQ message. Because the SCP node was configured for asynchronous storage commitment operation in the emulator, the storage commitment association is immediately closed by the Storage SCU emulator after the N-ACTION-RSP message was received from the SCP.
After this association was closed, the Storage Emulator starts listening on port 8104 for incoming storage commitment responses.

The Storage SCP system opens an association to the Emulator on port 8104 and sends the storage commitment N-EVENT-REPORT message. After successful receipt of this message, the association is closed by the SCP node.
The listening process in the Storage SCU emulator can be stopped by pressing the "Stop button" in the tool bar.

4.5 Analysis of results and logging

During the storage and storage commitment operation, the emulator saves results and summary files into the directory c:\Program Files\DVTk\Storage SCU Emulator\results. These files will be automatically deleted when the emulator application is closed.



The validation results will be displayed also in the validation results tab in case the validation option is enabled.

5 Supported DICOM SOP classes

The DVTK Storage SCU emulator supports all DICOM SOP storage classes (standard, specialized and private) for export.

6 Supported transfer syntaxes

In the table below, all transfer syntaxes are listed that are supported by the SCU emulator.

UID Value	UID Name
1.2.840.10008.1.2	**Implicit VR Little Endian**
1.2.840.10008.1.2.1	**Explicit VR Little Endian**
1.2.840.10008.1.2.2	**Explicit VR Big Endian**
1.2.840.10008.1.2.4.50	**JPEG Baseline (Process 1)**
1.2.840.10008.1.2.4.51	JPEG Extended (Process 2 & 4)

1.2.840.10008.1.2.4.52	*JPEG Extended (Process 3 & 5)*
1.2.840.10008.1.2.4.53	*JPEG Spectral Selection, Non-Hierarchical (Process 6 & 8)*
1.2.840.10008.1.2.4.54	*JPEG Spectral Selection, Non-Hierarchical (Process 7 & 9)*
1.2.840.10008.1.2.4.55	*JPEG Full Progression, Non-Hierarchical (Process 10 & 12)*
1.2.840.10008.1.2.4.56	*JPEG Full Progression, Non-Hierarchical (Process 11 & 13)*
1.2.840.10008.1.2.4.57	JPEG Lossless, Non-Hierarchical (Process 14)
1.2.840.10008.1.2.4.58	*JPEG Lossless, Non-Hierarchical (Process 15)*
1.2.840.10008.1.2.4.59	*JPEG Extended, Hierarchical (Process 16 & 18)*
1.2.840.10008.1.2.4.60	*JPEG Extended, Hierarchical (Process 17 & 19)*
1.2.840.10008.1.2.4.61	*JPEG Spectral Selection, Hierarchical (Process 20 & 22)*
1.2.840.10008.1.2.4.62	*JPEG Spectral Selection, Hierarchical (Process 21 & 23)*
1.2.840.10008.1.2.4.63	*JPEG Full Progression, Hierarchical (Process 24 & 26)*
1.2.840.10008.1.2.4.64	*JPEG Full Progression, Hierarchical (Process 25 & 27)*
1.2.840.10008.1.2.4.65	*JPEG Lossless, Hierarchical (Process 28)*
1.2.840.10008.1.2.4.66	*JPEG Lossless, Hierarchical (Process 29)*
1.2.840.10008.1.2.4.70	**JPEG Lossless, Non-Hierarchical, First-Order Prediction (Process 14)**
1.2.840.10008.1.2.4.80	JPEG-LS Lossless Image Compression
1.2.840.10008.1.2.4.81	JPEG-LS Lossy (Near-Lossless) Image Compression
1.2.840.10008.1.2.4.90	JPEG 2000 Image Compression (Lossless Only)
1.2.840.10008.1.2.4.91	JPEG 2000 Image Compression
1.2.840.10008.1.2.4.92	JPEG 2000 Part 2 Multi-component Image Compression
1.2.840.10008.1.2.4.93	JPEG 2000 Part 2 Multi-component Image Compression
1.2.840.10008.1.2.4.94	JPIP Referenced
1.2.840.10008.1.2.4.95	JPIP Referenced Deflate

1.2.840.10008.1.2.4.100	MPEG2 Main Profile
1.2.840.10008.1.2.4.101	MPEG2 Main Profile / High Level
1.2.840.10008.1.2.4.102	MPEG-4 AVC/H.264 High Profile / Level 4.1
1.2.840.10008.1.2.4.103	MPEG-4 AVC/H.264 BD-compatible High Profile / Level 4.1
1.2.840.10008.1.2.4.104	MPEG-4 AVC/H.264 High Profile / Level 4.2 For 2D Video
1.2.840.10008.1.2.4.105	MPEG-4 AVC/H.264 High Profile / Level 4.2 For 3D Video
1.2.840.10008.1.2.4.106	MPEG-4 AVC/H.264 Stereo High Profile / Level 4.2
1.2.840.10008.1.2.4.107	HEVC/H.265 Main Profile / Level 5.1
1.2.840.10008.1.2.4.108	HEVC/H.265 Main 10 Profile / Level 5.1
1.2.840.10008.1.2.6.1	RFC 2557 MIME encapsulation
1.2.840.10008.1.2.6.2	*XML Encoding*
1.2.840.10008.1.2.5	**RLE Lossless**
1.2.840.10008.1.2.1.99	Deflated Explicit VR Little Endian

Table 6-1: Supported transfer syntaxes