

OpenEMR: Achieving DICOM interoperability using Mirth

Technical Guide

Version 1.0

By ViCarePlus Team



Prepared by

ViSolve Inc.,
Contact: 408.666.4320
EMail: services@vicareplus.com
www.vicareplus.com

07th April 2011

Revision History

Version	Date	Author	Reviewed By
1.0	07/04/2011	Devi	Team

Table of Contents

Objective	4
DICOM Images	4
1. What is DICOM? Its Structure?	4
2. Downloading DICOM images	4
DICOM Image Viewers	5
1. Standalone DICOM Viewers.....	5
2. Browser based DICOM Viewers	5
Sending and Receiving DICOM Messages	5
1. dcm4che utility.....	5
2. Mirth Interface Engine.....	6
Configuring Mirth for DICOM support	6
3. Sending & Receiving DICOM Message:	7
4. Mirth Channel: Information Exchange Status	7
Integrating DICOM with OpenEMR	9
1. OpenEMR Files modification	9
2. Accessing the OpenEMR application after customizing for DICOM:	9
Further Work:	11

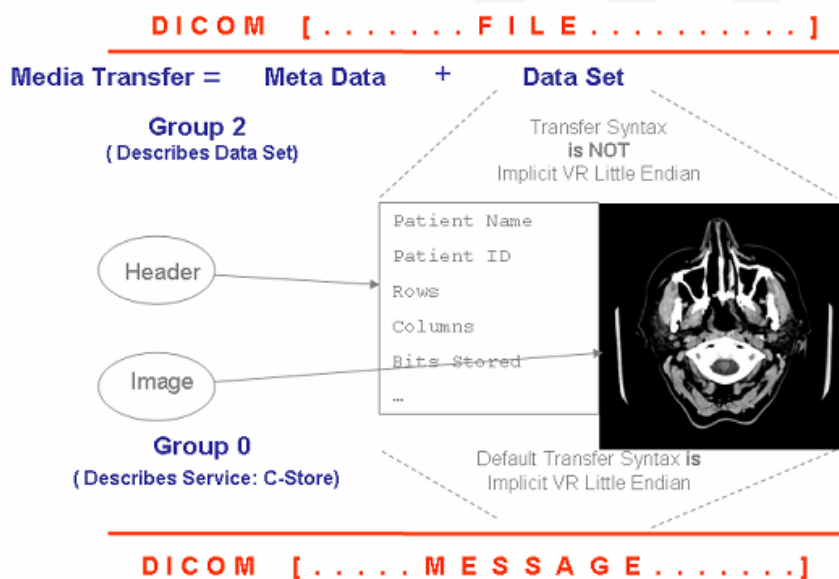
Objective

To gain good knowledge in Radiology imaging & DICOM standards of Healthcare domain and also demonstrating the interoperability of DICOM in OpenEMR using Mirth Interface Engine.

DICOM Images

1. What is DICOM? Its Structure?

Unlike other image formats like bmp, jpg, png etc, DICOM image structure has 2 components (i) Header (metadata) (ii) Actual Radiology Image. The header contains the Metadata of the image such as PatientName, PatientID, Physician details, Modality etc. The following is the detailed representation of Radiology image exchange using DICOM standard.



2. Downloading DICOM images

For the understanding of DICOM images, we need to download it from the freely available links or if we have radiology department database/share access, we can even get it from there.

Note:

- (1) The DICOM standard images cannot be viewed by ordinary image viewers like MSPaint, Photoshop, web browsers etc. But it seems that the latest version of Adobe photoshop has the support for DICOM standard images.
- (2) DICOM standard images can be viewed by 'DICOM viewer' tools, which are available for free too.
- (3) To understand the meta information of the DICOM standard image, just open it with some text editors and try understanding the information

DICOM Image Viewers

To view/read DICOM images, we need DICOM viewer tool or utility. These utilities should have the capability to read the raw DICOM files and parse the header information and the image. There are 2 types of DICOM viewers, which are mostly developed in Java.

1. Standalone DICOM Viewers

These viewers are standalone utility which will run the local machine or in the network and it has its own VIEW part. One such utility experimented is *JiveX DICOM Viewer*. Free DICOM Viewer

2. Browser based DICOM Viewers

There are few DICOM viewers which have the parsing engine with itself and render the VIEW part in a web browser. Since OpenEMR is a web based application, browser based DICOM viewer would get easily sync with the OpenEMR application. One such browser based Open source free utility analyzed and implemented is "Radscaper".

Sending and Receiving DICOM Messages

This is achieved using - dcm4che DICOM utility and Mirth Interface Engine.

1. dcm4che utility

dcm4che is an implementation of DICOM and IHE actors in Java.

dcm4che is a collection of open source applications and utilities for the healthcare enterprise. These applications have been developed in the Java programming language for performance and portability, supporting deployment on JDK 1.4 and up. dcm4chee is an advancement of dcm4che, with UI

List of dcm4che utilities:

- [dcm2txt](#)- Convert a DICOM object to text
- [dcm2xml](#)- Convert a DICOM object to XML
- [dcmDir](#)- Manipulate a DICOM dir
- [dcmecho](#) - Initiate a C-ECHO command as an SCU
- [dcmgpwl](#) - Query a General Purpose Worklist SCP
- [dcmMWL](#) - Query a Modality Worklist SCP
- [dcmof](#) - Simulate an Order Filler application
- [dcmqr](#) - Perform C-FIND, C-GET and C-MOVE operations as an SCU
- [dcmrcv](#) - DICOM receiver (C-STORE SCP)
- [dcmsnd](#) - Perform C-STORE operations as an SCU
- [dcmups](#) - Unified Worklist and Procedure Step SCU

- [dcmwado](#) - Initiate DICOM WADO requests
- [jpg2dcm](#) - Convert a JPEG image to DICOM
- [logger](#) - Log files to a Syslog destination
- [mkelmdic](#) - Create the serialized dcm4che2 DICOM Dictionary
- [mkuiddic](#) - Create the dcm4che2 UID dictionary
- [mkvrmap](#) - Create the dcm4che2 VR Mappings
- [pdf2dcm](#) - Convert a PDF document to DICOM
- [rgb2ybr](#) - Convert pixel data from YBR to RGB format
- [txt2dcmsr](#) - Convert text to a DICOM Structured Report
- [xml2dcm](#) - Convert XML to DICOM

To exchange DICOM messages, the 2 utilities - dcmsnd and dcmrcv were used.

Step 1: Start the DICOM message Receiver utility and make it to listen to 11112 port. Bind a directory to the DICOM receiver to store the received DICOM images.

Step 2: Now send a DICOM message to 11112 port and verify that it is being stored in attached directory.

When the port is not active, it throws error.

2. Mirth Interface Engine

Mirth Connect has support for DICOM SR (Structured Report). But it does not support DICOM querying.

Configuring Mirth for DICOM support

The following is the step by step instructions on configuring Mirth connect for DICOM communication.

Step 1: Create new channel to accept the Incoming Data DICOM data.

Step 2: Move to source tab and configure it. Set the Connector Type to 'DICOM listener' and configure the listener address and listener port. Then, click "Edit Transformer" to add rules for the incoming DICOM message.

Step 3: In the Source Transformer, input the xml structure of the DICOM messages as the message template and configure it according to the requirements.

Step 4: Now configure the "Channel Destination".

Here, configure the directory to save the received DICOM message (image). Make sure the connector type is "File Writer".

Finally Save the Channel.

3. Sending & Receiving DICOM Message:

Here, it is assumed dcm2che as the radiology department and Destination location in the Mirth as the Hospital.

The Mirth will be listening to the port 11113 port. If there is any message received at this port, then it is parsed and stored in the location specified in the 'destination'. Now the dcm2che is sending a radiology image of DICOM format to 11113 port. Mirth Connect will listen to this port and receives the image. The below screenshot depicts the Mirth dashboard and the status of the channel we created.

Dashboard mirth connect								
Status	Name	Received	Filtered	Queued	Sent	Errored	Alerted	Connection
Started	Radiology_DICOM_Communication1	0	0	0	0	0	0	Waiting

Now send the image to the port 11113

The message is received by the mirth and the status is updated. Refer the screenshot below.

Dashboard mirth connect								
Status	Name	Received	Filtered	Queued	Sent	Errored	Alerted	Connection
Started	Radiology_DICOM_Communication1	1	0	0	1	0	0	Waiting

Server Log \ Dashboard Status Panel				
Timestamp	Channel	Connector Info	Event	Info
2011-04-04 04:04:06.858	Radiolog...	Destination: File Writer - Destination 1	Busy	Result written to: /opt/openemr-4.0.0/sites/defa...

Though Mirth status report shows that 1 message is received and 1 message is sent, it is always good to verify it

4. Mirth Channel: Information Exchange Status

- (i) From the Mirth Dashboard, double click the channel
- (ii) It will show the status, protocol, timestamp etc of both the Channel Source and Channel Destination
- (iii) Just click the latest 'Source' info
- (iv) The tabbed panel (below the listing) will reflect the corresponding information
- (v) Click the 'Raw Message' tab. It will show the 'raw message' of the source connector and below screenshot shows the same.

Channel Messages - Radiology_DICOM_Communication1 mirthconnect

Search

Start Time:

End Time:

Quick Search:

Status:

Results 1 - 4
Page Size:

Date	Connector	Type	Source	Status	Protocol
2011-04-04 16:07:45:070	Destination 1	dicom		SENT	XML
2011-04-04 16:07:44:147	Source	DICOM	dicom	TRANSFORMED	DICOM
2011-04-04 16:04:06:859	Destination 1	dicom		SENT	XML
2011-04-04 16:04:06:210	Source	DICOM	dicom	TRANSFORMED	DICOM

Raw Message \ Transformed Message \ Encoded Message \ Mappings \ Errors \ Attachments \

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<dicom>
  <tag00020000 len="4" tag="00020000" vr="UL">188</tag00020000>
  <tag00020001 len="2" tag="00020001" vr="OB">00\01</tag00020001>
  <tag00020002 len="26" tag="00020002" vr="UI">1.2.840.10008.5.1.4.1.1.4</tag00020002>
  <tag00020003 len="62" tag="00020003" vr="UI">1.3.46.670589.11.0.0.11.4.2.0.5526.5.3448.2006032317243901942</tag00020003>
  <tag00020010 len="18" tag="00020010" vr="UI">1.2.840.10008.1.2</tag00020010>
  <tag00020012 len="16" tag="00020012" vr="UI">1.2.40.0.13.1.1</tag00020012>
  <tag00020013 len="12" tag="00020013" vr="SH">dcm4che-2.0</tag00020013>
  <tag00080005 len="10" tag="00080005" vr="CS">ISO_IR 100</tag00080005>
  <tag00080008 len="28" tag="00080008" vr="CS">ORIGINAL\PRIMARY\M_FFE\M\FFE</tag00080008>
  <tag00080012 len="8" tag="00080012" vr="DA">20060323</tag00080012>
  <tag00080013 len="6" tag="00080013" vr="TM">173510</tag00080013>
  <tag00080014 len="24" tag="00080014" vr="UI">1.3.46.670589.11.5526.5</tag00080014>
  <tag00080016 len="26" tag="00080016" vr="UI">1.2.840.10008.5.1.4.1.1.4</tag00080016>
  <tag00080018 len="48" tag="00080018" vr="UI">1.2.276.0.50.192168001092.11156604.14547392.220</tag00080018>
  <tag00080020 len="8" tag="00080020" vr="DA">20010101</tag00080020>
  <tag00080021 len="8" tag="00080021" vr="DA">20070514</tag00080021>
</dicom>
```

- (vi) Click the 'Transformed Message' tab. It will show the message that is transformed in the source connector and below screenshot shows the same. To your notice, Mirth transforms the DICOM message to 'XML' and then it performs the exchange.

Raw Message \ **Transformed Message** \ Encoded Message \ Mappings \ Errors \ Attachments \

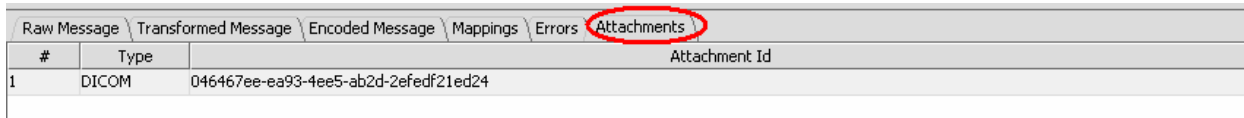
```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<dicom>
  <tag00020000 len="4" tag="00020000" vr="UL">188</tag00020000>
  <tag00020001 len="2" tag="00020001" vr="OB">00\01</tag00020001>
  <tag00020002 len="26" tag="00020002" vr="UI">1.2.840.10008.5.1.4.1.1.4</tag00020002>
  <tag00020003 len="62" tag="00020003" vr="UI">1.3.46.670589.11.0.0.11.4.2.0.5526.5.3448.2006032317243901942</tag00020003>
  <tag00020010 len="18" tag="00020010" vr="UI">1.2.840.10008.1.2</tag00020010>
  <tag00020012 len="16" tag="00020012" vr="UI">1.2.40.0.13.1.1</tag00020012>
  <tag00020013 len="12" tag="00020013" vr="SH">dcm4che-2.0</tag00020013>
  <tag00080005 len="10" tag="00080005" vr="CS">ISO_IR 100</tag00080005>
  <tag00080008 len="28" tag="00080008" vr="CS">ORIGINAL\PRIMARY\M_FFE\M\FFE</tag00080008>
  <tag00080012 len="8" tag="00080012" vr="DA">20060323</tag00080012>
  <tag00080013 len="6" tag="00080013" vr="TM">173510</tag00080013>
  <tag00080014 len="24" tag="00080014" vr="UI">1.3.46.670589.11.5526.5</tag00080014>
  <tag00080016 len="26" tag="00080016" vr="UI">1.2.840.10008.5.1.4.1.1.4</tag00080016>
  <tag00080018 len="48" tag="00080018" vr="UI">1.2.276.0.50.192168001092.11156604.14547392.220</tag00080018>
  <tag00080020 len="8" tag="00080020" vr="DA">20010101</tag00080020>
  <tag00080021 len="8" tag="00080021" vr="DA">20070514</tag00080021>
</dicom>
```

- (vii) Click the 'Mappings' tab. It will show the mapping that the source connector uses. Here there are 2 mappings – one for Modality and another for PatientID. To your notice, the mirth parses the DICOM message and extracts the corresponding value. The below screenshot shows the mapping variable and its corresponding DICOM extracted value.

Raw Message \ Transformed Message \ Encoded Message \ **Mappings** \ Errors \ Attachments \

Scope	Variable	Value
Channel	Modality	MR
Channel	PatientID	11791306742903

- (viii) Click the 'Attachments' tab. It shows one attachment. As we discussed, the DICOM message is communicated in 2 parts. One is the Metadata of the radiology/DICOM image and the other one is the 'Image'. Mirth parses the metadata and sends the Image as attachment. The below screenshot shows the 'Image as attachment' by the mirth.



#	Type	Attachment Id
1	DICOM	046467ee-ea93-4ee5-ab2d-2efedf21ed24

Integrating DICOM with OpenEMR

OpenEMR is a web-based application, whose 40% of the code is PHP. So customizing the application to integrate DICOM functionality is done with minimal effort.

1. OpenEMR Files modification

To demonstrate interoperability of DICOM using Mirth in OpenEMR, I have edited the left navigation tree to add a new item "DICOM" under "Patient Client"

And create a new file `/interface/patient_file/dicom/load_form.php` which lists all the DICOM image files from the stored location (This is the location where Mirth Channel Destination is configured; which is the placeholder for DICOM images which are transferred via Mirth) .

Now, DICOM Image viewer file. As said, the 'radscaper' DICOM image viewer is in different port. Modify the radscaper index file such that, it will accept the filename and pass it to the Applet, so as to view in the browser.

2. Accessing the OpenEMR application after customizing for DICOM:

Now access the OpenEMR -> Patient/Client -> DICOM, which lists all the DICOM images for the specified location. Below screenshot shows the same. I have extracted the (customized) filename and shown separately. To your notice, I have given a link to view the DICOM image and have highlighted the same in the below screenshot.

OpenEMR - Mozilla Firefox

http://172.16.1.40:400/interface/main/main_screen.php?auth=login&site=default

Patient: **James William (1)**
 DOB: 1988-03-28 Age: 23

Encounter: 2011-03-28 (2)

Home | Manual | Logged in: Administrator (Default)

Radiology DICOM Images

Patient ID	Image ID	Modality	View DICOM Image
11791306742903	MR	93bd97ab-5ea7-11e0-98a5-cc4956882ee4	11791306742903

When you click the ID, the DICOM image will be opened with the help of a DICOM viewer(Radscaper)
 Refer the status

2011-03-28 Encounter for James William

Blood Pressure: 344/344 Weight: 23.00 lb (10.43 kg)
 Height: 34.00 in (86.36 cm) BMI: 14 kg/m²
 BMI Status: Underweight

http://172.16.1.40/radscaper/index.php?image=11791306742903_MR_93bd97ab-5ea7-11e0-98a5-cc4956882ee4.dcm

Now click the link (highlighted in the above screenshot), which will open the DICOM image in the radscaper DICOM Image viewer.

http://172.16.1.40/radscaper/index.php?image=11791306742903_MR_0357453d-5eb4-11e0-98a5-cc4956882ee4

Radscaper

http://1...2ee4.dcm

(359,137) = 63

Fall 3
 O 11791306742903
 MRT Oberbauch

2007-05-14 10:22:20
 Anonymized Hospital
 11791306743104

Further Work:

This work can be enhanced further in the following avenues:

DICOM:

Analysing on the following items:

1. Radiology Video files and DICOM standards
2. DICOM viewers for Radiology video support
3. DICOM viewers for rendering Multiple images in a single snapshot
4. Highly downloaded OpenSource PACS servers & DICOM Viewers

Mirth Scripts:

1. Configuring Mirth Engine to integrate with Multiple radiology labs and multiple EMR/EHR's and developing Mirth scripts to as to route radiology images from the labs to the respective EMRs
2. Configuring Mirth to transform and parse the Source DICOM information and Storing the images based on the Modality and in the respective Patient Documents Folder

OpenEMR:

1. Customizing Patient's Document Tree to list Radiology images in the respective patient Document Tree