DICOM in Dart (DCMiD)
Damish Shah
Danielle Tinio
Mentor: Dr. James Philbin

Topic and Goal

Our goal is to determine the feasibility of using binary DICOM for building brower-based medical applications. We want to design and implement a DICOM editor that reads and writes binary DICOM and displays it using HTML5, CSS3, and Google's open source programming language, Dart. Our program should be able to parse and display binary DICOM imaging studies in less than 3 seconds.

Relevance and Importance

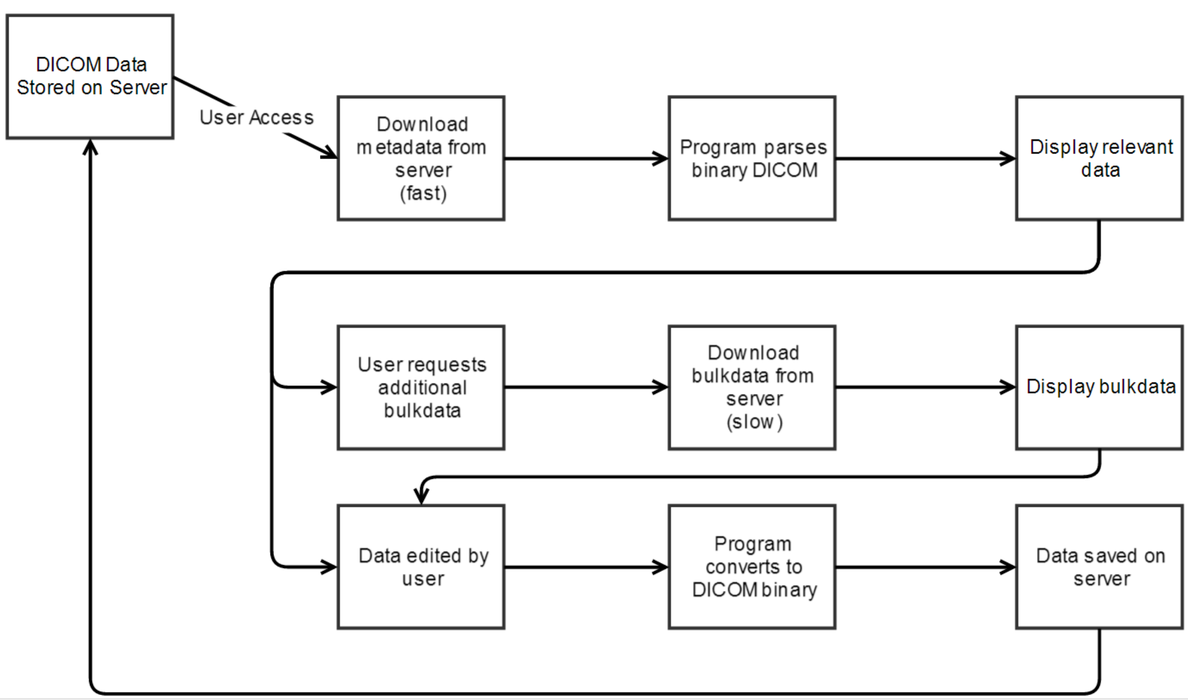
DICOM is the universal standard by which medical imaging information is handled, stored, printed, and transmitted for use by hospitals, clinics, imaging centers, and specialists. Nearly all hospitals use the DICOM standard in their medical equipment and information systems. Since DICOM does not specify any implementation details, it is crucial that information systems be created to allow end users to access and edit DICOM data in a fast, secure, and intuitive manner.

It is crucial that information systems by created to allow end users to access and edit DICOM data in a fast, secure, and intuitive manner. Having a zero-footprint client for parsing binary studies has many advantages over downloaded software applications. HIPPA covered patient data will not be stored on the end point device. Any data that does need to be stored in the data cache can be encrypted. Along with the centralization of data, costs relating to management, implementation, and security can be minimized because applications can be maintained and upgraded without the need for access control on the end point device. Having our program function within a web browser removes most management costs and allows the programmers to handle the technical specifics of the program without any effort or specialized knowledge on the part of the user.

Technical Summary

Our first step is to learn the DICOM standard to understand how medical imaging data is stored. From this knowledge, we will design a data structure for DICOM studies. After creating a project plan which includes packages and classes for our viewer and editor, we will use Dart syntax to translate our plan into code while simultaneously unit testing each class. Once we have functional code, we will work to iteratively optimize it to achieve our goal speed.

The flowchart below is a visualization of the information pipeline that our program implements:



Deliverables

- Minimum deliverables
  - Read and display DICOM in a browser and then write it
  - Build a test program that compares input and output to validate correctness
  - Create unit tests for each class
- Expected deliverables
  - Display a work list of studies of n patients
  - Display patient as collapse/expand tree for study information model
- Maximum deliverables
  - Edit metadata
  - Display images and add overlay information
  - Encrypt and decrypt studies using AES (GCM) using an encryption framework created at Hopkins Security Institute.

Key Dates

- **February 20:** Have project proposal finished and all of the programming planned and reviewed by Dr. Philbin
- **March 6:** Read input (parse)
- **March 20:** Write and validate output
- **April 3:** HTML5/CSS3 display metadata
- **April 17:** Edit data
- **May 1:** Display/Edit images
- **May 9:** Final Poster Presentation

We have arranged a checkpoint for every Wednesday, with the intention of finishing by the first week of April at the latest

Assigned Responsibilities

Both team members will be responsible for understanding DICOM data structures and implementing our plan in Dart. For the initial study class, both partners will write and test code together to ensure mutual understanding of the initial data structure. While one partner writes the code for each remaining class, the other partner will work on debugging code and will write the tests to ensure quality. The team members will alternate responsibilities for the remaining classes.

Dependencies and Plans for Resolving

- **Access to our mentor –** Our mentor has agreed to weekly meetings on Wednesdays at 9:30 a. m. and is open to setting up additional meetings when needed.
- **Computer to write code –** Both team members have computers to write code on. If our computers face any unforeseen disasters, we have access to computers on campus.
- **Bitbucket to share code –** As of February 13, 2014, we have access to the Bitbucket repository for this project.
- **Dart & DICOM Reference Information –** Our mentor has provided us with the DICOM documents that we need. Dart reference information can easily be accessed online.

- **Access to DICOM Test Data –** Our mentor has provided us with five anonymous test studies, with access to more if the need arises.

Management Plan

- Meet twice a week to work on code as a team.
    - Mondays & Fridays at 10:00AM
- Weekly meetings with Dr. Philbin
    - Wednesdays at 9:30 a. m.
    - Code reviews as classes and modules develop
- More meetings can be scheduled as required.

Reading List

- Part 5 of DICOM by the National Electrical Manufacturers Association
    - explicitly describes DICOM data structures and encoding
- Dart documentation
- MINT Toolkit source code and documentation (provided by Dr. Philbin)
- Mahmoud Ismail and James Philbin, Multi-series DICOM: an Extension of DICOM That Stores a Whole Study in a Single Object. Journal of Digital Imaging, August 2013; 26(4):691-697
- **James Philbin**, Tim Culp, Tim Dawson, Jonathan Whitby. RESTful Web Services in DICOM. *The DICOM 2013 International Conference*, Bangalore, India. March 2013.
- **James Philbin**, Mahmoud Ismail. Fast, Storage Efficient De-identification of Medical Studies. *The DICOM 2013 International Conference*, Bangalore, India. March 2013.

- Mahmoud Ismail, Yu Ning, **James Philbin.** Transmission of DICOM Studies using Multi-Series DICOM Objects. *Proceedings SPIE 8674, Medical Imaging 2013: Advanced PACS-based Imaging Informatics and Therapeutic Applications*. April 8, 2013.
- Mahmoud Ismail, Yu Ning, and **James Philbin**, Separation of metadata and pixel data to speed DICOM tag morphing. *SPIE Medical Imaging 2014: PACS and Imaging Informatics: Next Generation and Innovations*, Forthcoming.