DCMJS

Cross-compilation of DCMTK to Javascript.

http://dcmjs.org

Jean-Christophe Fillion-Robin[1], Steve Pieper[2]

[1]Kitware Inc
[2]Isomics Inc

19th NA-MIC Summer Project Week @ MIT
2014-06-24

Background

During the 9th CTK Hackfest hosted by Lawrence Tarbox and Dan Marcus from Mallinckrodt Institute of Radiology of the Washington University in St. Louis School of Medicine, a group of international and enthusiastic hackers addressed a large number of topics including DICOM support, CLIs, XNAT and also the CTK build and testing infrastructure.

They created dcmjs - A javascript cross-compilation of dcmtk that can be re-used in a HTML5 compliant browser. It allows the analysis and the process of DICOM images directly within the browser. Two demos have been crafted: (1) dcmjs dump to “View the header of a dicom file” and (2) dcmjs view to "View the pixel data of a dicom file”

For more details, see http://www.kitware.com/blog/home/post/676

Definitions

- DICOM: Digital Imaging and Communications in Medicine
    - The DICOM Standards Committee exists to create and maintain international standards for communication of biomedical diagnostic and therapeutic information in disciplines that use digital images and associated data.
    - The goals of DICOM are to **achieve compatibility** and to **improve workflow efficiency** between imaging systems and other information systems in healthcare environments worldwide. DICOM is a cooperative standard.
    - Fun fact: In 1993, v.3.0 of the standard, and rename from ACR-NEMA -> DICOM

Source: http://medical.nema.org/dicom/geninfo/Strategy.pdf

- DCMTK: DICOM Toolkit
    - DCMTK is a collection of libraries and applications implementing large parts the DICOM standard.
    - DCMTK is is written in a mixture of ANSI C and C++.
    - It comes in complete source code and is made available as "open source" software.
    - 20 years old

Source: http://dicom.offis.de/dcmtk.php.en

- CTP profile: Standard method for de-identification of images

Source: https://wiki.nci.nih.gov/display/CIP/Finalized+CTP+Anonymization+Profile+-+Basic

Motivations

- Client side anonymization (leveraging CTP profile)
- DICOM is a wonderful but tricky standard to support.
- DCMTK is a robust toolkit that embeds hard-earned knowledge in a C++ library.
- DCMTK C++ library needs to be compiled to be useful.
- DICOM is exceptionally complicated and infamous for non-conformant data 'in the wild', it may be a while before new software can be written that has been battle hardened to the extent of DCMTK.

Steve Pieper, CEO Isomics

Caveats

- Natively-written javascript version could be cleaner and more efficient

LLVM

- The LLVM Project is a collection of modular and reusable compiler and toolchain technologies.
- Clang is an "LLVM native" C/C++/Objective-C compiler, which aims to deliver amazingly fast compiles.
- LLVM assembly language. LLVM is a Static Single Assignment (SSA) based representation that provides type safety, low-level operations, flexibility, and the capability of representing ‘all’ high-level languages cleanly. It is the **common code representation used throughout all phases of the LLVM compilation strategy.**
- Code generator: is a framework that provides a suite of reusable components for translating the LLVM internal representation to the machine code for a specified target
- University of Illinois/NCSA Open Source License

Source:

http://llvm.org/
http://llvm.org/docs/LangRef.html#abstract
http://llvm.org/docs/CodeGenerator.html#introduction
http://llvm.org/docs/DeveloperPolicy.html#license

emscripten

- An LLVM-to-JavaScript Compiler
- MIT license and the University of Illinois/NCSA Open Source License
- OpenGL support: direct mapping to WebGL / OpenGL ES 2.0 emulation / OpenGL emulation

Source:

https://github.com/kripken/emscripten/wiki
https://github.com/kripken/emscripten/wiki/FAQ
https://github.com/kripken/emscripten/wiki/OpenGL-support

DCMJS

- CMake project allowing to easily create “dcmjs.js” by cross-compiling DCMTK
    - Configurable subset of DCMTK applications
    - Also take care of building zlib, libpng and libxml

- A website to demo the use of two application
    - DICOM Header Dump (client side dcmdump)
    - DICOM Image Viewer (client side dcm2pnm)

Source:

http://www.cmake.org/Wiki/CMake_Cross_Compiling
https://github.com/commontk/dcmjs
https://github.com/commontk/dcmjs#add--remove-applications
http://dcmjs.org/dump/index.html
http://dcmjs.org/view/index.html

Prerequisites

- Install emscripten: https://github.com/kripken/emscripten/wiki/Emscripten-SDK
- Install CMake: http://www.cmake.org
- Install Git: http://git-scm.com/downloads

Source:
https://github.com/commontk/dcmjs#prerequisites

Building

```
git clone git://github.com/commontk/dcmjs
mkdir dcmjs-build
cd dcmjs-build
cmake -DEMSCRIPTEN_ROOT_PATH:PATH=/path/to/emscripten \
      -DCMAKE_BUILD_TYPE:STRING=Release                \
      ../dcmjs
make -j5
```

Source:

https://github.com/commontk/dcmjs#building

Usage

- From a webpage

```
var reader = new FileReader();
[...]
reader.readAsArrayBuffer(file);
[...]
var fileName = file.name;
var uploadedFilePath = '/uploadedfile.dcm';
var content = new Int8Array(reader.result);
FS.writeFile(uploadedFilePath, content, {encoding: "binary"});

var imageFilePath = 'convertedImage';

Module.callMain(['dcm2pnm',              \
             '--verbose',                \
             '--histogram-window', '2', \
             '--frame', '1',             \
             '--write-raw-pnm', uploadedFilePath, imageFilePath]);

var stat = FS.stat(imageFilePath);
var stream = FS.open(imageFilePath);
var pnmBuffer = new Uint8Array(stat.size);
FS.read(stream, pnmBuffer,0,stat.size);
FS.close(stream);
[...]
```

- With Nodejs
    - To list available application:

```
$ nodejs ./dcmjs-build/bin/dcmjs.js list
dcm2pnm
dcmdump
dcm2xml
dcmftest
```

    - To execute an application:

```
$ nodejs ./dcmjs-build/bin/dcmjs.js dcmdump
$dcmtk: dcmdump v3.6.1 DEV $

dcmdump: Dump DICOM file and data set
usage: dcmdump [options] dcmfile-in...

parameters:
  dcmfile-in                    DICOM input file or directory to be dumped
[...]
```

Source:
https://github.com/commontk/dcmjs#usage

Under the hood

- Checkout the code:
    - Build system: https://github.com/commontk/dcmjs
    - Website: https://github.com/commontk/dcmjs/tree/gh-pages