FDA Regulatory Issues for Open Source Software

Richard Chapman, Chief, General Hospital
Devices Branch, Office of Device Evaluation, CDRH

OSEHRA
October 18, 2012
Medical Devices

♦ Submissions to FDA
  ♦ IDE - Investigation Device Exemption
  ♦ 510(k) – Substantial Equivalence
  ♦ PMA - Premarket Application
  ♦ HME – Humanitarian Device Exemption

♦ Safe and Effective

♦ When is a software product a medical device?
  ♦ http://www.fda.gov/cdrh/devadvice
"an instrument, apparatus, implement, machine, contrivance, implant, in vitro reagent, or other similar or related article, including a component part, or accessory which is:

- recognized in the official National Formulary, or the United States Pharmacopoeia, or any supplement to them,
- intended for use in the diagnosis of disease or other conditions, or in the cure, mitigation, treatment, or prevention of disease, in man or other animals, or
- intended to affect the structure or any function of the body of man or other animals, and which does not achieve any of its primary intended purposes through chemical action within or on the body of man or other animals and which is not dependent upon being metabolized for the achievement of any of its primary intended purposes."
Are there any regulatory issues with open-source software?
Short Answer

There are no regulatory issues with open-source software
Long Answer

But,
§ 820.30(a) General.
Each manufacturer of any class III or class II device, and the class I devices listed in paragraph (a) (2) of this section, shall establish and maintain procedures to control the design of the device in order to ensure that specified design requirements are met.

The following class I devices are subject to design controls: (i) Devices automated with computer software; and …
What Are Design Controls?

Design controls are:

- An integrated set of management practices (policies, processes, and procedures), which are
- applied to design activities
- To control the quality of products and services.

Process is important!
How Do Design Controls Work?

- Via mechanisms to provide visibility (i.e., means to measure the controlled variable) throughout the development process
- Via documented procedures to exercise continuous (or at least frequent) control of resources (i.e., feedback mechanisms)
- Via a semantic structure (language, taxonomy) to facilitate communications

Process is important!
What Are The Limitations?

- Design controls do not assure the quality of products and services (but they provide a framework for assessing and documenting quality).

- Design controls do not completely eliminate design errors (but they prevent many errors and facilitate finding and correcting errors earlier in the development process).

- Management still needs the right people and the right tools to do the design work and review the results for adequacy.
Needs & Intended Uses

Review

Verification

Validation

Design Input Process

Requirements

Initial Design Stage

Stage 1 Design Output

Nth Design Stage

Final Design Output

Possible Interim Reviews

Production

Test Articles

Requirements & Intended Uses

Needs & Intended Uses

Design Input Process

Verification

Validation

Stage 1 Design Output

Initial Design Stage

Nth Design Stage

Final Design Output

Possible Interim Reviews

Production

Test Articles
QSR versus Pre-market submissions

♦ Device manufacturers may use the same procedures and records for compliance with quality system and design control requirements, as well as for pre-market submissions to FDA.

♦ Specific safety or effectiveness issues related to software validation
Guidance Documents

♦ General Principles of Software Validation
♦ Guidance for Off-the-Shelf Software Use in Medical Devices
♦ Guidance for Industry - Cybersecurity for Networked Medical Devices Containing Off-the-Shelf (OTS) Software
♦ Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices
Websites


- http://www.fda.gov/cdrh/humanfactors for human factors information
Level of Concern

♦ Choose the appropriate level of concern
  ♦ Minor, Moderate, Major
  ♦ Key Questions
  ♦ Assess the Level of Concern before mitigating any hazard; that is, you should assess your software device against these questions as though you have not implemented hazard mitigations
Level of Concern

- FDA reviewers examine:
  - Device Description from pre-market submission
  - Software Description
  - Hazard Analysis
  - Software Requirements
  - Opinion of Domain and Software Experts
Level of Concern

♦ Drives the documents that you submit to FDA in a pre-market submission.

♦ Ideally documentation should be artifacts from your design control activities.

♦ If the FDA reviewer disagrees with your assessment of level of concern, it should be a simple photocopy exercise to provide the additional documentation requested.
Software Description

♦ A summary overview of the features and software operating environment.
Device Hazard Analysis

- Tabular description of identified hardware and software hazards, including severity assessment and mitigations.
Software Requirements Specification

A triad

- Functions
  - What the device does

- Performance
  - Accuracy, speed, reliability, environmental influences

- Interfaces
  - Input/output, power, data protocols, user interface
Requirements—Guiding Principles

♦ Must specify what is needed, not the solution
  Complete to an engineering level of detail
♦ Requirements are developed by engineers, not by marketing department or users
♦ Adequacy
  ♦ Unambiguous (objectively verifiable)
  ♦ Quantitative limits expressed with a realistic measurement tolerance
  ♦ Self-consistent
  ♦ Environment completely characterized
  ♦ Completeness and relevance of external references
Architecture Design Chart

- Detailed depiction of functional units and software modules. May include state diagrams as well as flow charts.
Software Design Specification

- Software design specification document.
Traceability Analysis

- Traceability among requirements, specifications, identified hazards and mitigations, and Verification and Validation testing.
Software Development Environment Description

- Summary of software life cycle development plan. Annotated list of control documents generated during development process. Include the configuration management and maintenance plan documents.
Description of V&V activities at the unit, integration, and system level. Unit, integration and system level test protocols, including pass/fail criteria, test report, summary, and tests results.
Verification = assessing conformance to requirements (did I do the design right?)

Validation = objective evidence that devices fulfills intended use (did I do the right design?)

I.e., verification is details-oriented and validation is a cumulative summation of all efforts to assess suitability of design.

Validation almost always includes user evaluation.
Software V&V

LEGEND
v = VERIFY
V = VALIDATE
V & V—Guiding Principles

- V & V encompasses many activities: Tests, Inspections, and Analyses on the final version of software.
- V & V overlaps with design review to some extent. Companies may draw the dividing line anywhere reasonable.
- The design records should contain one or more verification and validation reports which summarize V & V activities, explain discrepancies, and document approvals.
Design Reviews

♦ The cycle is:
  ♦ Design
  ♦ Audit (V&V)
  ♦ Review
  ♦ Resolution of review findings
  ♦ Not all “problems” detected by reviewers are real, or need to be corrected.
  ♦ There should be a procedure for tracking concerns and ensuring follow-up.
  ♦ There should be a procedure for resolving differences of opinion.
  ♦ Design review procedures should identify who is in charge.
Revision Level History

♦ Revision history log, including release version number and date.
Unresolved Anomalies

- List of remaining software anomalies, annotated with an explanation of the impact on safety or effectiveness, including operator usage and human factors.
- The software guidance is vague about what “indicate the problem” means. Many sponsors simply list the symptoms of the problem.
Unresolved Anomalies

♦ Determine the root cause, i.e., put your finger on the problem. Point to the problem in the source code.

♦ Search code base for other occurrences of the software pattern, idiom, expression, or other software formulation that resulted in the defect that caused the observed anomaly.

♦ Coupling analysis
Other Topics

- Automated Analysis Tools
- Human Factors Analysis
- Risk Management
- Post-market Issues
- Assurance Cases
- IEC 62304
The bottom line…

- FDA does not prescribe the specific design processes appropriate for software design (or any other technology, for that matter).
- In making judgments about the adequacy of design and development processes, FDA applies generally accepted principles of good design practice, as dictated by the software engineering discipline.