Agenda

- Software for Regulated Processes
- TIR “In Scope” Discussion
- Software V&V Basis for TIR
- The Journey to Critical Thinking
- Technical Information Report (TIR) 36
Software for Regulated Processes
Quality System Regulation (QSR)

21CFR820.70(i) Automated Processes

“When computers or automated data processing systems are used as part of production or the quality system, the manufacturer shall validate computer software for its intended use according to an established protocol. All software changes shall be validated before approval and issuance.”
“computers or automated data processing systems”

- A system consisting of
  - one or more *computers*
  - associated *software*
  - associated *peripheral devices*
  - personnel
  - documentation

- Includes items such as:
  - PCs, PDAs, Servers, Networks
  - Software Applications, Operating Systems, Excel Spreadsheets, Access Databases
  - Scanning Devices, Printers
  - Users, Run Teams, Manuals
“production or the quality system”

Software systems that are used to automate and/or execute an activity required by the QSR
TIR “In Scope” Discussion
What Software is “In Scope”?

- TIR provides a tool to help determine the answer to this question.
- Define at a high level the process to be automated and use of the software within that process.
- Perform a regulatory use assessment....
Regulatory Use Assessment

Determine use of software for regulated processes by asking:

- Could the failure or latent flaws of the software affect the safety or quality of medical devices?
- Does the software automate or execute an activity required by regulation (in particular, the requirements of the QSR)?
Regulatory Use Assessment

Determine if there are electronic records by asking:

- Does the software generate or manage data to be used in or support of a regulatory submission?
- Does the software generate or manage records that are required by a regulation (e.g., device master record, device history record, design history file, or clinical trial records) or records that would be accessed in the future to provide evidence of the completion of an activity required by regulation?
Regulatory Use Assessment

Determine if there are e-signatures by asking:

- Is the software used to execute or record an electronic signature required by regulation?
Any “Yes” Answers?

Then software is “In Scope” and must be validated
Basis of SW V&V for the TIR
“validate computer software”

“A conclusion that software is validated is highly dependent upon comprehensive verification activities.. performed at each stage of the software development life cycle” (SDLC).

General Principles of Software Validation: Final Guidance for Industry & FDA Staff; January 11, 2002
(italics added emphasis)
“validate computer software”

“a matter of developing a “level of confidence” that the device meets all requirements and user expectations for the software automated functions and features…”

*General Principles of Software Validation: Final Guidance for Industry & FDA Staff; January 11, 2002 (italics added for emphasis)*
“validate computer software”

“Determination of the correctness of the final program or software produced from a development project with respect to the user needs and requirements. Validation is usually accomplished by verifying each stage of the software development life cycle.” (SDLC)

Glossary of Computerized System and Software Development Terminology
August, 1995
“The level of confidence, and therefore the level of software validation, verification, and testing effort needed, will vary depending upon the safety risk (hazard) posed by the automated functions…”

General Principles of Software Validation: Final Guidance for Industry & FDA Staff; January 11, 2002
(italics added for emphasis)
The reduction in the risk of harm to an acceptable level through the performance of confidence-building activities that support the conclusion that the software is validated
Harm to....

- Persons associated with the commercialized medical device product (patient, operator, bystanders, service personnel)

- May want to include:
  - Persons using the software system
  - Environment
Errors That Can Lead to Harm

**Mfg equipment** software errors that can:

- Produce unacceptable product
- Miss the detection of unacceptable product
- Improperly accepts unacceptable product
- Cause equipment safety features to malfunction
- Cause the release of harmful chemicals into the environment
Errors That Can Lead to Harm

Quality system software errors that can:
- Release unacceptable product
- Miss the ID of product that requires field action
- Improperly trend product safety issues that may require field actions
The Journey to Critical Thinking
The Journey….

- Started 5 years ago with ~12 members from industry and John Murray (FDA)
- Working sessions every 6 months or so
- Found that describing what we “knew” to be appropriate and enough was far more challenging than we thought
The Journey….

- Realized that there’s not always the same right answer..... it depends

- Decided it made more sense to describe an approach to determining the answers
Approach => Critical Thinking

- A process of *analyzing and evaluating* various aspects of the processes being automated, the software and the environment in which it will be used.

- To *identify the most meaningful set of confidence-building, value-added activities* to be applied during the SDLC.
Approach Includes: Drivers, Tools & Decisions

Decision Drivers
- Results of analyzing and evaluating various aspects of software, the process and the environment in which it will be used

Tools
- Confidence-building, value-added activities (e.g. code reviews, testing, trace analyses)

Decisions
- Identifying the tools that, when applied, support the conclusion that the software is validated
Decision Drivers

- Process Risk and Software Risk
- Other drivers that can influence decisions:
  - Complexity of the software and/or the process
  - Type of software (firmware, PLC, OTS)
  - Software pedigree (understanding of the quality/robustness of the software)
  - Ability to influence life cycle controls (vendor supplied or “in-house” built)
Tools in the Toolbox

- Are recognized good software engineering practices (e.g. requirements review, traceability, code reviews)
- The tools may change
  - As technology changes
  - As newly proven practices come into use
- A company’s tool may change:
  - As a company embraces new technology
  - As personnel experience and knowledge changes
Tools in the Toolbox
(Examples)

- Formal software requirements review
- Risk Management Planning
- Identification of risk control measures within the manufacturing/business process
- Software architecture documentation & review
- Design Specification
- Development/Design Review
- Identification of risk control measures within the software design
- Code Review/Code Verification
- Traceability matrix
- Vendor audit
- Test Planning
- Unit test
- Integration test Interface Test
- Regression Test
- Vendor Supplied Regression Test Suite
- Software system test
- Normal Case Test
- Robustness Test (Stress Testing)
- Output Forcing Test
- Combination of Inputs Test
- Beta testing
- Performance testing
- User Procedure Review (Includes procedures that are related to the use of the software)
- Internal Training for the application
- Installation Qualification
- Operational and Performance Qualification (when Process Validation is performed)
- Final Acceptance Tests (when not performing OQ or PQ)
- Operator Certification

From current draft of TIR
Decisions

- Level of effort applied to and scrutiny of documentation/deliverables
- Extent of content in the documentation/deliverables
- Selection of “tools” from the “toolbox” and methods for applying the tools
- Level of effort in applying the tools.
Critical Thinking Key Elements

- One must think
- There's no single checklist
- Not a “one-size-fits all”, “cookie cutter” approach
- Allows for validations to vary from software to software
- Allows for different validation solutions for same software utilized for different “intended uses”
- Considers all key stakeholders and their needs.
TIR 36
Validation of SW for Regulated Processes
TIR Purpose & Intent

- Provide guidance on what to think about when determining the appropriate content and size of a validation effort.

- Provide guidance on the application of critical thinking to reach the appropriate depth and rigor of activities.

- Describe a process flow model that enables the generation of the driver information prior to when it’s used to drive decisions.

- Provide examples of applying critical thinking in a variety of circumstances.
Validation work stream activities during the development of a software system

Red dashed lines indicate information from drivers to blocks where decisions are made

** : the application of selected tools
Process risk analysis

- Intended to identify the harm that could occur as a result of process failure.
- Identified risks can be considered when selecting a software solution.

It should be noted that the act of moving a process from manual control to an automated process based on software can inherently affect the risks associated with the process.
Risk Analysis Work Stream Activities

- Process risk analysis
  - Risks apply whether the software solution is internally developed or obtained externally
  - When risks of harm are low, there is less concern about spending a lot of time, energy and money on more robust software solutions

- Drives:
  - Level of effort applied to and scrutiny of documentation/deliverables
  - Extent of content in the documentation/deliverables
  - Tools from the toolbox
Critical Thinking Approach

Analysis/Evaluation ➔ Drives ➔ Decisions

- Process Risk Analysis
- Level of Effort and Rigor
- Tools from Toolbox
Risk Analysis Work Stream Activities

- **Software risk analysis**
  - Determine the inherent risks associated with software failure
  - Devise risk control measures

- **Drives:**
  - The choice of tools from the toolbox
  - May drive the re-evaluation of process risk if the introduction of software affects those risks
Critical Thinking Approach

- Analysis/Evaluation
  - Software Risk Analyses (FMEA etc)
  - Drives
  - Tools from Toolbox
  - Decisions
The Validation Plan
- Documents the application of critical thinking including the selection of the confidence-building, value-added activities (ie the tools) and the information that drives the selection

The Validation Report
- Cites the completion of the confidence-building, value-added activities that support the conclusion that the software is validated
TIR Examples

- Are neither authoritative nor prescriptive
- Layout is not intended for use as a validation template, nor does it contain all the depth and detail that would be expected for actual validation documentation.
TIR Examples

- Assume that the prerequisite processes
- Meant to provide information about the decisions and drivers of decisions used in the critical thinking process and do not necessarily represent the comprehensive validation of the software discussed.
TIR Examples

- PLC for manufacturing equipment
- Automated welding system
- Automated welding process control system
- C/C++ language compiler
- Automated Software Test System
- A simple spreadsheet
- A (not so) simple spreadsheet
- Parametric sterilizer
- Nonconforming material reporting system—Total system upgrade
- Software for scheduling nonconforming material report review board meetings
- Approved vendor list system
- Calibration management software
- Automated vision system
- Pick and place system
Critical Thinking

- Results in a validation solution that establishes compliance for a manufacturer
- Ensures that the software is safe for use
- Results in appropriate and adequate documented evidence
- Results in the use of least burdensome and value-added validation activities
TIR 36
The Application of Critical Thinking to the Validation of 21CFR820.70(i) Regulated Software
Thank You!