EFCOG SQA Spring 2021 MEETING NOTES
Tuesday April 20, 2021

Safety, Security, and Quality Moments, *Presenter Carol Olijar (ANL):*
- Deficiency: Use of a template for SQA forms – finding because not locked down with version control (e.g., revision number and date).
- OFI: Referenced the DOE 414 Guide currently in Revcom in SQA forms (assumed it would be published/approved within year) – it is not allowed to reference unapproved documents or drafts in official documents or forms.

SQA-Related Issues from the Office of Nuclear Safety Enforcement, *Presenter Jacob Miller (EA-12):*

- Presenter is Director, EA-12 - Office of Nuclear Safety Enforcement, Office of Enterprise Assessments. EA-13 is Information and Security.
- Presentation provides recent cases with links to software for the last 2-3 years. The causal factors found related to software.
- The notes below are based upon the presentation slide deck:
  - ONSE is tasked with enforcing the Codes of Federal Regulation (CFRs) and assessing how well DOE sites are implementing them (see slides for list of Laws/CFRs/Orders assessed). The link to enforcement actions is: https://www.energy.gov/ea/information-center/enforcement-infocenter. The remainder of the slides were recent assessments that had Software (SW) -related issues.
  - Slide 4: biggest concern was that ONSE found the SQA issue rather than the site or DOE oversight group.
  - Slide 5: SW model used was not properly configured – the SW (AERMOD) was used properly, but assumptions made and data input were inadequate for the actual conditions resulting in contamination off-site.
    - The technicians assumed the doses of materials present were lower than they actually were.
    - Good practice to use multiple codes in safety situations to double check answers
    - The age of the site being demolished was also a contributing factor. The available knowledge did not include all aspects of the building.
  - Slide 6: Storage drums exploded:
    - released radiological waste dust into the air
    - dust clogged the alarm system so that the alarm shut off prematurely
    - fire fighters were briefly exposed to waste contamination
    - unexpected reactions within the waste caused the explosion
    - Corrective Action – use SW to analyze radioactivity of materials
    - SW concern – SW not controlled as safety software.
  - Slide 7: data entered into software that manufactured outer plate was incorrect resulting in microscopic flaws in the design; contributed to failure and damage; also, SW used to verify/check outer plate was not calibrated correctly, so identified flaws were dismissed as okay when they were not.
  - Slide 8: Rad Protection Group – worker questioned why he hadn't bene asked for a bio-assay sample; discovered that SW used to track personnel and sample frequency, had a bug that missed certain people:
    - SW not controlled or compliant with the site's SQA Program
    - Algorithm used to calculate sampling was incorrect – off by a factor of 100.
  - Slide 9 (related to slide 8):
    - SQA Program not adequately implemented
    - Personnel didn't receive any SQA training
    - Quality improvements were not implemented (issue had been identified in 2015)
    - SQA records were not generated or maintained.

- Slide 10: Conclusions:
  - SW not properly identified as safety software
  - Sites didn't always follow their own procedures
  - Inadequate training on SQA or the procedures used
  - Quality Improvements were inadequate – causal/root analyses of issues didn't go deep enough, therefore corrective actions were not effective.
- The notes below are based upon what the presenter stated for the events/cases:
  - Case 1: Inadequacies in the implementation of SQA (MSA). Expect contractors to find their own problems and fix them. Significant omissions and inadequacies (procedures, implementation, and training).
  - Case 2: Spread of Contamination at the Plutonium Finishing Plant Demolition Site.
    - Used software to model dispersion of contamination was a factor two times what was modeled. No concern with how the software itself was used. Concerned with the assumptions that went into the software was not completed accurate.
    - Airborne radioactivity was greater than ten times higher than modeled. Material got distributed outside of radiological boundary.
  - Case 3: Drum Over-pressurization Event at Idaho Cleanup Project.
    - Continuous air monitor (CAM) stopped alarming, on onset of event. Implemented a TSR by using software to process and display the thermal imaging information, but did not adequately control. This software was processed per their procedures for safety software. Dust in air, loaded filters of CAMs, poor fit made radiological conditions, not equal to expected conditions. After alarm went off, no longer visual indication of alarming. Workers entered space. Waste unplanned, corrective action for TSR to monitor material prior to put in drum. TSR was for thermal imaging software, software was not controlled as safety software. Issue was with the input to the software not how the software functioned.
    - Airborne Found much larger doses; unplanned issue with waste.
  - Case 4: Fuel Element Failure at High Flux Isotope Reactor (ORNL). Deformation of fuel plates; possible weld failure; software comes in with manufacturing. Using 1940's vintage lathe; in the process of updating machinery; when programming a very difficult to read document was used. No easy method for verifying new method; came up with new strategy (coordinate measurement machine); they didn't realize they still had to have a very specific origin point; were a little off and didn't appear to be within tolerances; ultimately, plates ended up in slightly different alignment. Software being used in manufacturing and verification, which contributed.
  - Case 5: Radiation Protection Program Weaknesses, Portsmouth Site, Ohio. November 2018, expects Bioassay sample, 193 employees not asked to give samples for 3 years. Using Access database, incorrectly radiological environment. Manual change needed to software. Erroneous data, when query ran. May 2019, error spreadsheet, instead of X, data reports off.

- Case 6: Radiation Protection Program Weaknesses. Work process SQAP was inadequately implemented.
  - FAP employee questions why bioassay sample - using Access db developed by the dosimetry group, there was a manual adjustment that needed to be made that people weren't aware of; software was not developed or controlled IAW SQA Program.
  - Shared with others; checked another spreadsheet report found error; doses were still within limits but info was being made public
    1. SQA was inadequately implemented
    2. No formal SQA training
    3. Problems were not corrected between 2015-2018 to improve SQA
    4. SQA records were not generated or maintained
- Case 7: Common Areas of Weakness:
  - Identification of Safety Software. Software everywhere, firmware. Software not controlled nor identified.
  - Procedural Compliance. Processes in place, not followed, lack of training. If not trained in how to determine software is safety software.
  - Training. Insufficient.
  - Inadequate corrective actions. Quality Improvement:
    - QA Criterion 10 quality problems.
    - Missing/incomplete causal factors.
    - Inadequate corrective actions.
    - Wrong causal factors, wrong actions. Fixes made quickly, bulked number of issues into 1 problem statement. Rest of corrective action not implemented.
- Questions/Comments:
  - From David Louie - what software was used in Case 2? Whose fault was it the software or analyst? The problem with software was assumptions, missing parameters. No issues with software. Info went in, was inaccurate.
    Answer: Software used was AERMOD. The fault was not the software it was the input entered into the software. Software was being used correctly, the assumptions of the parameters entered into software, didn't take into account certain factors. The problem was with what was put into the software. The software should tell you limitations of it. Sid Ailes noted – the software may have assumed certain source terms. At demolition found larger doses of material than expected. The process knowledge of the facility may be 60 years old, may find something different than what you expected based upon this process knowledge, in the actual demolition. Such as in this instance Cesium capsules, then demolition shut down.

- From David Louie – Can more than one type of software be used?
  Answer: No requirements to use two software. Could have two people performing independent analysis such as for calculations per review, make sure answer is in ball park. Good practice especially, with potential exposure.
  Greg Smith noted – Isn't that the whole definition of validation. Al Zuchero agreed.
- How do we better identify Safety Software? Are there adequate controls being applied to software? Are failure/hazard analyses adequate?
  - Ask developers/designers to explain requirements to see if they really understand them. Make sure they don't put too many requirements into a single module, making it hard to understand/debug and more likely to have errors.
- From Veronica Morris – For procedure, deep dive, more detail steps, anyone seen this done well? How rigorous can we apply requirements in place? Are we adequately controlling the software?
  Answer: Depends upon how implementing the software, may want to use FMEA, for safe failure modes.
- From Stella McKirdy – when reviewing specifications; push them to explain requirements and impacts; asks developers to provide better info. Feels a big part of failure is due to requirements not being specified adequately. Is there a requirement or procedure to do this? Been doing this due to experience.
- From Vicki Pope – For project management, project may be stalled, while I take 2-4 weeks to go over requirements, mapping out decision chain, who is associated with what, and do on white board. When going over this and a prototype, get lots of "ah-ah" moments. Too many assumptions. People supply requirements to development, but they are in different language for development to understand. In another instance, configuration management of software means something different to everyone. Make sure all on same page for path forward. Sometimes team don't use same language; work to ensure terms are used consistently.
- We need to request Enterprise Assessments (EA) for better comments with respect to SQA. Can view details of annual and semi-annual assessments at website – DOE Enforcement Info Center. Link provided in presentation handout.

Update for the EFCOG SQA Task Group, *Presenter Chris Beaman (SRS) Group Discussion:*

- Been with DOE for about 2 ½ months. Primary work is on QA Policy. Three agenda items.
A. Status of Safety Software Central Registry (SSCR).
  1. A few years ago (2018?), HQ did an MSA of the Central Registry and Toolbox Qualification process -> a number of findings (will ask Christian/Chris if we can get a copy of the report). AU-32 struggling through corrective actions. They don't really understand who, how often, or how users are using the versions of codes on the Toolbox versus more updated versions. Christian/Chris didn't seem to understand why sites didn't just qualify newer versions on their own instead of using the outdated Toolbox versions. Working through these to bring it to good. However, having a hard time articulating what is good, how it is being used (don't know), using guidance. Interested in input from EFCOG SQA. help with understanding who is using; how using; thoughts for improvements.
  2. Dr. David Louie, a MELCOR developer at Sandia said they got lots of requests to get newer versions on the Toolbox and have made requests of this nature to AU-32. They have been waiting for AU-32 to request an update of the code in order to apply for a new qualification. They are currently supporting the Toolbox version (very old) and the newer versions used world-wide. Some questions he raised:
     - How do they request qualification assessments for newer codes
     - Is there a way to qualify the Sandia SQA Program/processes used by the MELCOR development team so version updates can be added to the Toolbox without full qualification assessments
     - Can SQA assessments from other government entities (e.g., NRC) be used to qualify (or help qualify) codes and/or updated versions. (Christian commented that NRC uses MELCOR for different purposes – Non-safety – than DOE, so their assessments will probably be insufficient.)
     - Also asked about using DOE site assessments of a code to qualify it to the Toolbox.
  3. AU-32 really wants to hear from the user groups, (Accident Analysis EFCOG Task Group – our Safety Basis division for the codes currently on the Toolbox), to see which versions of which codes they want to see on the Toolbox. Referred them to the AA toolbox code survey. They would also like a better count/handle on the number of sites and users for each code. That was when Dr. Louie raised the issue of user groups for each code. This could be managed by a site, user community, or EFCOG, but would be best if initiated by the code developer organization. We also briefly discussed sharing our 830/Safety SW Inventories. That would help us discover which sites are using which codes (both those on the Toolbox and other titles/versions).

B. DOE Guide 414.1-4A Revision
   a. Did not have best config mgmt; pushing to have it reviewed through RevCom. Revcom closes 4/26/21. Comment resolution will be conducted using SMEs. AU-32 will consider and address comments via the formal Revcom process.
   o Revcom has been extended until May 10th. Check with your site's approval coordinator to see if/to when your site's deadline has been changed.
   b. A committee of SMEs will address the comments.
   c. Several members of our group reported that they were told they needed to copy/paste their comments on the DOE 414 Guide back into Revcom when the review deadline was extended to April 26th. Chris Beaman spoke with Doxcelerate regarding this issue. Here is a breakdown of what he discovered:
      i. When the original draft Guide was put into Revcom back in March, Doxcelerate accidentally deleted a paragraph from Section 2.2.1.5 (last word of paragraph 1 and the next paragraph explaining Table 2.2-1, Safety Software Grading Levels). – See below for that missing paragraph.
      ii. This was caught due to early comments, so when Chris requested the review extension and correction of the document, Doxcelerate closed the original review, uploaded the complete document and created a brand new review with the due date of April 26th.
      iii. Since the original review was closed (due to the need for a new document upload), comments that had been submitted against the original Revcom record were also closed (i.e., they did not automatically transfer over). Hench the need to copy/paste them into the new Revcom review.
      iv. Since the most recent extension to May 10th did not need a new document upload, in theory, we won't need to copy/paste previously submitted comments. Doxcelerate will be working with the site approval coordinators on what needs to be done. However, if you have already submitted comments, it would be a good idea to confirm that your comments are still showing in the current Revcom record.
   d. Chris also said that if you have additional comments on the Guide and your site's deadline has already passed, you can email those comments directly to him (Christopher E. Beaman christopher.beaman@hq.doe.gov).
   e. Chris has agreed to walk the EFCOG SQA group through the entire Revcom process as part of the May WebEx.

C. Organization Excellence Website.
   o For QA, SQA. Post questions, store presentations. OE - community of practice forum. Please register and participate (https://orgex.energy.gov/forums/qa-community-practice)

- Questions/Comments:
  1. From David Louie: SSCR - how do we get more users in EFCOG, sites waiting for DOE to update guidance. SSCR is there a way for EFCOG and DOE to work together to improve? Update to versions, most are old/obsolete. In past, attended EFCOG meetings, keep saying waiting for DOE, is this way for both EFCOG & DOE to come to consensus to improve process? LANL used MELCORE V2.2 assessment, covers all criterion to qualify, and then moved to new version. However, we don't want to do this for each version. Lot of cost. Need to address for international users. Site spends money to go from one version to another. Would need to do every 3 years, the assessment.

     Answer: Suggest participate in Toolbox Alternate SQA sub-group.

     Chris/Christian - sites are always able to qualify software they need.

  2. From Veronica Morris – We were advised by our Engineer reviewers that comments made in the first review round that ended Apr 9 won't be automatically included and reviewers are expected to copy-paste in the this second round or request Doxcelerate to do it for them (support@doxcelerate.com).
     Answer: Garrett/Chris - will look into the process and get back to Vickie.
  3. From Carol Olijar – Any Idea when the Guide will be put out?
     Answer: Chris - no; have already received comments that will require some re-work of sections
  4. From Christian Palay – SSCR - Actual users of the SSCR codes are needed. Have heard from developers, but not hearing from users. Want to make better for users. Don't have users list. Codes can be downloaded by anyone. Developer of code should have list of users when managing their safety software list, for who users are. Each site has safety basis group, perhaps they can tell you which software they use based upon DSAs.
     i. New Task Group to kick off to help with SSCR; coordinating with other EFCOG groups.
        - What sites use what software
        - Want user input -- software is good
        - Potential for LL - if using newer version; it would be good to know
        - It takes too long to get newer versions approved before another version might be out.

     We need user input. Such as, I like the version I use. What version would they like to see? May be able to check with Accident Analysis people to check into this. Need lessons learned for newer versions, how to qualify them. By the time the evaluation is done, new version of code is out. In addition, for NRC, they may use different code than DOE such as for Max. Level of qualifications may be different. These are all details the SQA group and DOE could work out. SSCR is highly valued.

     Suggestion: David Louie: Have user list of MALCOR. Could provide contacts to Christian. Older versions of MALCOR are used, 2004. Suggest having user group for SSCR codes. Form within nuclear safety group. Share info. What versions.

     Comment: Dave Thoman – Could query sites, for users of codes.

Comment: Veronica Morris – She has internal process for SSCR, could provide to DOE, can share.

Comment: Kevin Shaw – not sure what comments have been submitted, but not sure what the purpose of Appendix A and not folded into the body of the document?

Answer: Chris – good question, put in the comment and we'll take a look…good comment.

Comment: Carol Olijar– Whenever a site is going to use a central registry code, they need to do V&V per their lab's SQAP process. They shouldn't use the code without testing it for the way they will use it.

NQA-1 SW Subcommittee Mapping of Agile Development to DOE 10 Work Activities, *Presenter Lance Abbott (SRS):*

- NQA-1 Update, adding Agile Guidance to the Standard.
- Mapping Agile to 10 Work Activities, NQA-1 subgroup took over task from SQA subgroup. NQA-1 Agile Guidance, Task Proposal Notice (TPN). Subcommittee working on better aligning NQA-1 with Agile development and work products.
- Background – Varying ideas about what Agile is – based upon Manifesto and Principles. The fears: No processes and tools, no documentation, and no plan. This is not what Agile is. See slides for additional information.
- The reality is Agile enhances, does not drive development. Documentation should be "right-sized" to the project. Estimates and schedules are communication tools. Goal is working software that meets customer needs. Adjust plans and communicate. Focusing on "right-sizing" SQA -> graded approach.
- Involve stakeholders/users early and often in development process; make them part of development team; and have SQE as part of team.
- Big problem is how to produce the required records without being an administrative burden. Even tools don't always capture all required elements of required records.
- Challenges – there are 7 types of Agile Frameworks. Each institution of a framework is unique.
- Need to reset expectations:
    - Not a mapping of every "shall" statement in NQA-1 Subpart 2.7.
    - Not a matrix of all agile frameworks.
    - Not a tutorial.
    - Complete per Agile definition of "done".
- NQA-1 Standard is intended to be flexible with respect to the software engineering methodology. Answers "what", not "how" and "when". Part I, section 5; subpart 2.7 section 400. Identify key deliverables from section 400 and how they may be achieved using Agile framework.
- Will use an example of a Scrum framework to present compliance to NQA-1 for the key deliverables.

Questions/Comments:
1. From Christian Palay - Is that all what is required? Will that provide the adequate assurance? Is there more beyond the requirements? Should there be?
2. From Abhijit Sengupta - Which edition 2021 or 2023? And which part number this will be in part 3 or 4?

    Answer: More likely 2023. There are more than just requirements, design, developing, testing, etc. See parts 3 & 4.

3. From Vicki Pope: Had external assessment of team using a hybrid; found several findings/improvements. They felt that templates were too "waterfall" based process. Higher risk; more detail. Less risk; less detail. Including a workflow diagram that covers that they have to do within the process. Can change during meeting, print/laminate, verify current version, then follow process. Developers and SQA work together; embedded in project to help with whole process. Lance - requirements are never right at beginning; user thinks they've given you all requirements when in reality they only gave you 10%.
   2 challenges -
   - What are requirements, how to satisfy, where happen in SCRUM-like process.
   - Records == user stories/requirements - value in getting detail of requirements in just-in-time nature….can create records challenges too.
   - Sometimes tools will create/manage records; working with teams to understand what it buys us with Records; not just creating documents to create documents to check a box.

4. From Veronica Morris - Electronic records. Real value is in native tool. Documentation for version of software tool. Get tools blessed. SQA recommendation. Or we could have electronic records. Use native tool for traceability; getting them blessed as records repository would add value.
   Answer: We are required to generate pdf output of documents, as objective evidence. We use configuration management for source code. Have data available only for certain amount of time.

5. From Pat Auer - tools provide actual evidence but does not always translate into .pdf. Amount of time creating records should not outweigh time to create/develop, etc.

6. From Stella McKirdy - have a process in which you identify your records repository. Then have retention requirements. Has been working well. Struggling with ***. Tools are helping tremendously.

Applying SQA Requirements to R&D Software, *Presenter Jennifer Turgeon (SNL):*
- Jeni is part of Sandia's Advanced Simulation Computing (ASC) Program, and V&V Assessment Group. She has over 16 years of experience, attended EFCOG meetings on/off.
- Sandia follows a formal SQE program, which utilizes an appraisal methodology program. This program ensures codes are appraised throughout the life cycle.
- The notes below are based upon Jeni's slide presentation deck:
  - Slide 3: way to read the "wheel":
    - Taking "Project Management" (blue wedge toward center of wheel) as an example, there are 6 categories of practices
    - For the Measurements and Analysis category, there are 3 practices
    - For practice 5, there are 3 artifacts produced.
  - Training is for development team training, not user training.
  - The program was built from CMMI methodology.
  - Appraisals: Jeni has a team of 15 appraisers, all have received official training:
    - Appraises ASC codes. Those of "high value" are reviewed every three years.
    - Maintain independence of appraisers (appraisers have other duties).
    - ASC codes are NOT identified or developed as safety software; some users are using them as safety software and must do necessary qualification activities before using them as such.
    - Sandia SQA requirements are very similar between those for Safety and Non-safety software.
  - Slide 6: production code levels are well defined; other types of codes are a work in progress:
    - Work of Others/COTS – someone else develops the code.
    - Maintenance – no development being done, just keeping it working.
    - Sm Production Codes – this category will probably be eliminated; very limited funding so no codes are currently in this category.
    - R&D – higher formality targets for codes they suspect will become production codes:
      - Light requirements allow flexibility to experiment in early stages.
      - If development seems fruitful, R&D codes move into Production Ready category with higher SQA targets.
      - Once solid as Production Ready, will move to QER Codes with even higher SQA targets.
      - As codes move up in these categories, more SQA practices are added and/or the formality levels of the practices are increased.
  - Slides 7-9 are blow ups of Slide 6 for the R&D Code categories.
  - Not all R&D codes move to Production Ready; most stay in R&D.
  - Working to put together a list of artifacts, documents, and records required for SCRUM development.
  - Slide 10: considering getting rid of low-level formality or coordinating these with R&D targets.

- The notes below are from the presenter comments during the presentation:
  - 30 practices fall within SQA activity, for 12 Practice Numbers (PR). Artifacts (AR) are mapped to the practices. The Artifacts are examples of documents.
  - Sandia follows the ASC SQE Quality plan, using a graded approach. It grades the practices based upon likelihood of product failure – Very High to Negligible; and the consequences of the failure from Catastrophic to Negligible. Then the level of effort (LOF) from this is determined which results in formality for a deliverable.
  - The Sandia ASC Appraisal Method. Appraise code every 3 years. ASC funded and selected by management based upon which codes are used heavily. Code can be decommissioned. Appraisal team rates codes.
  - R&D codes transition as follows:
    - R& D – new, heavily developed.
    - Production Ready.
    - Production.
    - Qualified Engineering Release (QER).
  - Need to determine the minimum practices to be followed for codes. Developers of R&D don't want to do anymore; these users want to use the codes. The users are analysts. Instead of tested code, use in production setting. Codes break. R&D have to rewrite code. R&D use case following minimum requirements, not excessive enough to transition to production ready without a lot of work.

Questions/Comments:

1. From Carol Olijar: Are there any safety software codes?
   Answer: No. in PR2 determine if code is safety software through independent team. Codes are not developed for use as safety software. Doesn't matter, process is as rigorous as for safety software. Onerous to ensure used appropriately, up to end-user. This process is used for general safety software, follows similar process to SQAP of DOE safety software per 10 CFR 830. Codes can be used in both safety and non-safety areas. Encumbrance is on user to do extra testing.

2. From Vicki Pope: LLNL has similar process.

3. From Al Zuckero: Can you summarize data easily for assigned grade level?
   Answer: The templates do go thru a graded approach consequence level for formality.

4. From David Louie: Are all ASC codes compliant to 414.1D?
   Answer: Yes, LANL doing same thing, LANL ASC codes are not safety software either. Sandia ASC Use Code Types have Target Ratings, set for each practice based upon use case type; and working with management. Can get waiver for not meeting graded level formality. Test new user codes out, and adjust target rating.

5. From David Louie: Modify code often, when you have R&D, modify to go to production ready, test, go to production. Bugs, go back to R&D. Repeat process for each version? What is timeline for each release?
   Answer: Not necessarily. Cycle keeps going. For example for Sierra, did pilot. Look at end-use, develop under QER Use Case. For each release 4 steps. For Sierra, it took 2 years. Depends upon scope of ASC. Sierra has been around for long time, since 2007.

6. What does a SCRUM artifact look like?
   Answer: Struggling with. Such as requirements. Story evolves as captured in tool. Path forward – low-level formality different for R&D. Small use case type need. Target Ratings for all use cases. Made in sync.

Artificial Intelligence Software Development, *Presenter Sid Ailes (Aitkins Global):*

- Artificial Intelligence (AI) – some think of as robots running amuck. Smart phone has AI, so do security systems, and Alexa. Challenge is how do you use these effectively and control.
- The notes below are based upon Sid's slide presentation deck:
  - Slides 4-5: how AI is currently used for commercial purposes.
  - Slides 6-7: how AI is currently used in DOE.
  - Slide 8: examining things like underground storage tanks take approximately 10 hours of drone footage. A human(s) must review those videos to assess possible damage. This could take over 100 hours. Looking into AI helping with these examinations.
  - Slide 13: ML = Mechanical Learning

- The notes below are from Sid's comments of the presentation:
  - Two (2) subsets of AI:
    1. Machine Learning. Data, continue to develop.
    2. Deep Learning. Code development, no human interface.
  - Commercial Examples of AI:
    1. Network Rail system. Classify risk of ground movement, such as electrical lines. Improve software based upon highest risks.
    2. Substantial Damage Estimation (SDE). Following hurricanes, model estimate of residential damage patterns. Reduced inspections.
  - DOE Example. Hanford Tank Farms, 177 waste storage tanks, 149 single shells. Found leaks in single shell, transfer to double shell. Double shell from 1968, can have leaks too due to surface erosion. Need inspections. Collect data since 1990. The challenge is DST annual inspection takes 6-10 hours of video recordings. There is enormous volumes of data to review. This would be a good example to use for AI solution.
  - Machine Learning is based upon:
    1. Data. Need for right data, need much data for data points.
    2. Infrastructure. Programming language.
    3. Algorithms. The Algorithms is the engine to use data to draw conditions. Patterns for images.
    4. Visualization. Hallmarks of AI, helps users understand what data represents, graph based networks.
  - Can apply AI to Safety Software.
  - Machine Learning Process:
    1. As the Question – Problem.
    2. Understand the Data.
    3. Clean Data.
    4. Feature Engineering. Controls.
    5. Choose the Model.
    6. Tune & Evaluate.
    7. Use Model.
    8. Feedback.

- AI Failure:
  1. Tesla X, S, 3 fatality. Autopilot rely on.
  2. User Fatality.
  3. Robot crashes, got lost.
  *** Need to understand what controls are necessary.
- Common Pitfalls:
  1. Cold Starts.
  2. Expecting the AI to do all the work.
  3. Not allowing for tuning.
  4. Trap users into bubbles.
- Nuclear QA Control: Subpart 2.7:

	AI Process
Software Engineering	Ask the Question
Baseline Doc	Understand the Data
Design Reqs	Clean Data (or processing time longer)
Reviews	Data Validation
Acceptance Testing	Turning and Evaluating, Fault Testing, Ethical Check.
Config Mgmt	Model / Modifications. Data Libraries. MISIM. Automatic maintenance.
Problem Tracking	Feedback
Procurement	Procurement
Operations Maintenance	Using the Model, Feedback, Error Reporting
Standards & Conventions	Development Environment
Support Software	
Nuclear QA Control: Part II:	AI Process
Design	Feature Engineering
Code	Develop Model.

- Data Acquisition:
  1. Uber autopilot using camera.
  2. Cooling system needed for all electronics – infrastructure.
- Feature Engineering:
  1. Ended autopilot for Uber, fatal crash into j-walking pedestrian. Not account for failure modes.
  2. Validation didn't register, visual didn't expect. No message for pedestrian in path.
  3. Off shelf systems turned off security.
- Develop Model:
  1. Concept.
  2. Inception.
  3. Iteration / Construction.
  4. Requirements.
  5. Development.
  6. Test.
  7. Delivery.
  8. Feedback.
  9. Release.

- Existing code, run faster automatically. Biggest challenge: how do you maintain configuration control.
- Machine Inferred Code Similarity (MISIM). Figures out what program is to do, compares with other programs, makes efficient. Trains on huge code already publicly available.
- Summary – focus on end goal, imperial process, evolve not like waterfall, engineer features based on data, perform functions, acceptance testing, train users to use model, iterative process – improve predictions and efficiency.

Questions/Comments:

1. From Marylou Apodaca: Were the issues with Tesla corrected?

   Answer: No, fatality occurred last week or two. Drivers falling asleep, autopilot is so good they rely on it too much. Example, crash barrier was deformed, not in shape anticipated. NTSB looking into Tesla autopilot, 14 incidents. Data not comprehensive. Features do not account for failure modes.

Software Gone Wrong, *Presenters Vicki Pope, Greg Pope (LLNL) Group Discussion:*

1. Airline Data Breach. SITA system, customer name, gold status. No personal info exposed. If have frequent flier number, good idea to change password every 3-6 months.
2. Boeing 787 Reboot Requirement
   a. Embedded 32-bit clock that gets filled in about 51 days
   b. Once clock data filled, airplane stops recording new data – instead begins displaying data from day one (51 days old). This means things like plane altitude, speed, directions, flight plan, pitch, etc. is wrong
   c. Several warnings are also disabled with the bug
   d. Lesson Learned – do stress and endurance testing of your software to see if long run times could adversely effect your code.
3. Other software – Sid, last week Saturday, Tesla killed passenger, no driver in front seat.

Daily Wrap up, *Presenter Vicki Pope (LLNL):*

- Like to go over what we have done, what we are going to do.