Training Objectives

- Awareness of EPRI’s Software Development processes
- Focus on changes to process from last year
- Questions & answers
Training Course Outline

- Introduction & Overview

- Process Steps
  - Concept
  - Planning
  - Requirements
  - Design
  - Implementation & Testing
  - Installation & Checkout
  - Release & Distribution
  - Operations & Maintenance
  - Retirement

- Who to call for help?
Introduction – who should be here?

- Project managers
- Technical staff
- Anyone developing, or interested in developing, R&D software, including EPRI contractors

The EPRI Nuclear Quality Assurance (NQA) Program supplements the requirements of this document. If there is a conflict between the requirements of the EPRI R&D Software Development Policy and the NQA Program, then the NQA Program shall take precedence.

EPRI QA site link
Introduction – what is this training?

- EPRI has a new policy on R&D software

- Software is a GREAT tool for EPRI, and we want to continue doing it
  - Technology transfer, computer based training
  - Tools for members to apply EPRI calculations
  - Data collection to support EPRI research

- This training is how to do it WELL, maximize benefits to members and minimize risk to EPRI

- This training is NOT “software for dummies”

- Assure that R&D software produced by EPRI is:
  - Technically correct
  - Not a license risk to EPRI or its members
  - Not a security risk to EPRI or its members.

The Project Manager is responsible to assure adequate performance of the software
Introduction – what changed?

- **Software type changes**
  - Proof of concept software – NEW category
    - Does not have to become production
  - “Simple” spreadsheets defined for eMedia
  - Computer Based Training (CBT) – NEW Category
    - No pre-production required

- **Process changes**
  - Clarity on software types and requirements by type
  - Clarity on what forms are required when & some consolidation
  - NEW recommended list of software tools (pre-approved) & approval process for new tools
  - No deliverable material ID required prior to issuing R&D contracts (but may be required for other business functions)
  - No detailed software documents (Quality Assurance Plan, etc.) required prior to issuing contracts
  - Encourage consultation by internal software experts

- **Software Excellence Network created**
Introduction – how does this process relate to other processes?

- Project Management Guidelines
  - Initiation
  - Planning
  - Execution
  - Closure

- Software Development Process
  - Concept
  - Planning
  - Requirements
  - Design
  - Implementation & Testing
  - Installation & Checkout
  - Release & Distribution
  - Operations & Maintenance
  - Retirement
Introduction – what happens next?

- Final form & website updates for 2015
  - Note that any changes here do not have to impact existing contracts, but should consider

- Potential improvements in 2015 - 2016
  - Form consolidation
  - Standardized web delivery of training modules
  - Improved guidelines for open source software and mobile apps
  - Improved post-release guidance (funding, support, issue tracking)

- Have ideas, need help? Contact your Software Excellence Network representative:
  - PDU: Tina Burnette (Knoxville)
  - Nuclear: Jeff Riley (Palo Alto)
  - Generation: Justin Thibault (Charlotte)
  - Software Quality Assurance: Manuel Morales
  - Nuclear Quality Assurance: Steve Stine
  - Legal: Ellen Brodbine
  - Lead: Rob Austin
Software Processes at EPRI

- Developing a software product is similar to developing a technical report
- But there are some unique characteristics and risks
- The purpose of the process is to help you identify, manage, and communicate the risks to the stakeholders

<table>
<thead>
<tr>
<th>Software Development Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concept</td>
</tr>
<tr>
<td>Planning</td>
</tr>
<tr>
<td>Requirements</td>
</tr>
<tr>
<td>Design</td>
</tr>
<tr>
<td>Implementation &amp; Testing</td>
</tr>
<tr>
<td>Installation &amp; Checkout</td>
</tr>
<tr>
<td>Release &amp; Distribution</td>
</tr>
<tr>
<td>Operations &amp; Maintenance</td>
</tr>
<tr>
<td>Retirement</td>
</tr>
</tbody>
</table>
Concept Phase - Summary

- Key Decisions:
  - Basic Requirements
    - What is the problem you are trying to solve?
    - What does the user need?
    - Why EPRI? What is being developed that’s unique?
  - Software Type
    - What is the technical readiness level (TRL) of the technology?
    - Which type(s) could fill the need?
    - Does EPRI have existing software that is a close analogue to what you want to do?
  - Budgetary Cost and Schedule
    - Establish general funding limits
    - Compare different costs
    - Communicate with stakeholders
Concept Phase

- Key Elements and Risks to Consider:
  - Software Architecture Drivers
    - Who will use the software? (skill, frequency, motivation)
    - Quantity of data?, is it real-time?
  - Funding
    - Get general estimates on different approaches.
    - What’s the appetite for the stakeholders for not just development; but security, administration, and maintenance
  - Software licensing of the final product
    - Often overlooked; but it’s better to think about it early
    - 3rd party content? Interface to 3rd party products?
    - Several models and all deserve consideration (EPRI deliverable only, licensed for commercialization)
  - Policies
    - Each Software Type falls in different ranges of testing and deployment requirements
    - Other policies could apply: Mobile Apps, Web Delivery, 3rd Party & Open Source Software, etc.
Concept Phase

- Resources Involved:
  - Stakeholders
    - Members
    - Program Staff
    - Approval Chain
    - Development Team (if already identified)

- Who to call for help? EPRI Software Excellence Network
  - Experienced developers and project managers in EPRI who can help

- Forms/Templates:
  - Lifecycle Document
Regarding Software Life Cycle Planning

- Software projects represent an additional commitment due to the ongoing support and evolutionary development requirements of a successful and popular product.
  - What is the problem begin solved?
  - What stage is the software supporting at this time?
  - How might updates be handled in the future?

- The purpose of the Life Cycle Plan is to facilitate the communication of expectations with management and members.

- A Life Cycle Plan should be developed and approved by the responsible Program Manager prior to the project kickoff.
Life Cycle Plan

- These items should be addressed in the Plan:
  - Use case (who will use the proposed product and how it will be used)
  - What is the development strategy and resources
  - Where does this fit in the software process requirements
  - How will the software be licensed to members and other parties
  - What is the expected life of the product. How do we expect it to evolve over time
    - do we expect updates to occur
    - how will these be funded
    - What will ultimately happen to the software?
      - Commercialize, open source, retire, make free to public
  - What will happen when users call for support, or when bugs need to be fixed
  - If it is a web/server app, how will hardware support and obsolescence be handled
Planning Phase

Once the concept has been agreed upon, actual project planning can proceed.

- Key Decisions:
  - Software Type and Production Level (TRL)
    - Identification of EPRI Software Process Requirements and associated risks
  - Budget Cost and Schedule
  - Development Team (internally or externally developed)
## Technology Readiness Levels (TRL) & Software at EPRI

<table>
<thead>
<tr>
<th>TRL Level</th>
<th>Software Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 – 2: Concepts to be proven</td>
<td>Prototype</td>
</tr>
<tr>
<td>3 – 6: Concepts proven, software solution to be proven</td>
<td>Proof-of-concept</td>
</tr>
<tr>
<td>6 – 9: Proven software solution being developed</td>
<td>Production</td>
</tr>
</tbody>
</table>

Refer to [http://swdev.epri.com/softwaretrls.asp](http://swdev.epri.com/softwaretrls.asp) for complete list of TRL’s as used at EPRI
Planning Phase – Software Types

- Special software types:
  - Computer-Based Training (CBT)
  - eMedia - Desktop (includes simple spreadsheets)
  - eMedia - Web

- Common software types:
  - Complicated spreadsheets
  - Desktop
  - Server/Enterprise
  - Software Implemented via a Commercial Software Platform
  - Software Extension
  - Web Application / Web Service
  - Mobile Application

- Exception:
  - Billable Service Agreement (BSA) – requirements are contract-driven
Planning Phase – Software Type: CBT

- Basic characteristics:
  - Delivered as a desktop product
  - Implemented in any commercially available product
  - Does not require a preproduction (i.e. “beta”) review/test

See list of pre-approved software tools. Other tools will need licensing approval.
Planning Phase – Software Type: eMedia

- **Basic characteristics:**
  - Contain only content, content with hyperlinks to local or external sources, or other very basic interactions with the user
    - E.g. Spreadsheet, database, presentation
  - Might be used within a commercially available product
    - Be aware of licensing costs and product compatibility within member environments

- **eMedia Desktop**
  - Delivered as a “desktop” product (e.g. CD or download)

- **eMedia (Content-only) websites:**
  - Include (but not limited to):
    - Traditional HTML sites with no input capability
    - Sites created within commercially available products such as SharePoint (without custom/programmed extensions)
  - Require support from Web & Mobile Solutions (formerly Digital Solutions) and/or IT
Planning Phase – Software Type: eMedia vs. Spreadsheets

- **Simple Spreadsheets:**
  - Treated like eMedia - Desktop software type
    - Functionality must be verified by someone other than the author prior to submission to Tech Pubs
  - No macros, hidden cells, rows, or columns
  - All calculations can be easily viewed and verified on no more than three worksheets

- **Complicated Spreadsheets:**
  - Treated like Desktop software type
    - Functionality must be reviewed and tested by SQA Team

Difference affects process, cost, and schedule
Planning Phase – Budget Cost and Schedule

- **Estimating Costs**
  - Developers should be able to provide estimates for each phase of the software development life cycle
  - Include documentation costs

- **Testing**
  - The budget must account for all testing that might apply
  - Include the SQA Team’s review

- **Schedule**
  - Be sure to allow adequate time for pre-production internal reviews and customer reviews.
  - SQA review may take longer near the end of the year.

- **Assume something will go wrong**
Planning Phase – Development Team

▪ Team Composition
  – Implementation personnel: internal developers, contracted developers, or both
  – If software includes server, web, or mobile components, include plenty of time in schedule for EPRI Server Ops, Web & Mobile Solutions and/or Database Administrators
  – Consider how the software will be tested. In some cases using a third party for testing is appropriate.

▪ Seek assistance from EPRI staff:
  – SQA Team, who have insight to experience of existing contractors
  – Use internal software development staff to review architecture, verify adherence to EPRI’s objectives, preview software
  – EPRI peers who have delivered the same software type
  – Software Excellence Network representative in your sector
Planning Phase – Key Considerations

- Some EPRI Policies and other software requirements have significant impact on the planning – costs, schedule, or feasibility

- Key Policies
  - Deliverables Policy
  - Software Development Policy
  - Third Party and Open Source Software Policy
  - Subscriber Websites Policy
  - Rogue and Homecoming Websites Policy
Planning Phase – Key Considerations

- **Funding**
  - Be aware of your funding source. Most deliverables are **required** to be publically-available.

- **Licensing**
  - Many members obtain software through a master agreement
    - Not compatible with some products (e.g. royalty fees)
  - Always check licensing costs and obligations for any third-party product
    - Some products have been pre-reviewed by Legal
  - Some software licenses are incompatible and might not be able to coexist in the same distribution
  - Software that is open source have significant additional product and process requirements

- **Technology**
  - Technology must support business goals and desired business impact
  - Unsupported products/platforms require early review and appropriate approval(s) from IT (for support options) and/or Legal (for licensing)
Planning Phase – Key Considerations

- **Web-Based Software (except content-only)**
  - Web software is an excellent method of providing value to members, but is more complicated
    - Web-based software requires front-end approvals, whether hosted inside or outside of EPRI
    - Web-based software requires a kick-off meeting with IT
    - Websites and web apps have [specific branding/layout requirements](#)
    - Web apps and web services have significant security requirements that require additional [security testing](#) and [documentation](#)
    - Contractors may need network access to prepare, test, and submit software

- Content-only web sites developed as collaboration sites in SharePoint can be requested via [http://dsrequest/](http://dsrequest/)
Planning Phase – Key Considerations

- **Mobile Apps**
  - If the Member Center will not be used for distribution, mobile apps require approval from Export Counsel and the Program Manager prior to project initiation.
  - Require approval from Web & Mobile Solutions prior to project initiation.
  - iOS apps have UI specifications and deployment controls (iTunes) that are dictated by Apple – currently no good EPRI / iTunes model.
    - iOS apps should be free results postings.
  - Android apps can be released as .apk files on the Member Center.
    - Provided users can install apps that are not distributed via Google Play.
Planning Phase

- **Resources Involved:**
  - Project staff

- **Who to call for help?**
  - SQA for contractor requirements, contract approval
  - Legal for licensing questions
  - Web & IT for web sites
  - EPRI Software Excellence Network representative for planning, technology, and business compatibility questions

- **Forms/Templates:**
  - Software Development Review Form (SDRF) for submitting a contract request
  - Web apps and web services require a [checklist](#) for coordinating with IT
Requirements Phase – Key Decisions

- Control scope creep by clearly identifying the requirements

- All requirements should be:
  - Measureable
  - Testable
  - Unambiguous
  - Complete

- Prioritize requirements — and communicate the prioritization to funders — as a way to manage scope and expectations as the schedule progresses
Requirements Phase – Defining the Requirements

- The format, rigor, complexity, and scope of your requirements should track with the criticality and complexity of your project

- If the contractor is needed to write the Requirements, then a separate task is recommended
  - THIS IS NORMAL IN SOFTWARE - REQUIREMENTS DEFINITION CAN BE HARD

- Prioritize requirements — and communicate the prioritization to funders — as a way to manage scope and expectations as the schedule progresses

- Identify High Risk areas:
  - Conduct prototyping during the detailed design development.
  - First of a Kind? If it’s never been done or if your developer has never done it, be sure to include capability development and proof of concept to ensure that the learning curve and risk are baked into your project.

- The Software Testing approach should be identified as part of the Requirements
Requirements Phase – Detailed Requirements

- EPRI has specific *product requirements* for various aspects of the software implementation. Some key areas:
  - EPRI Corporate Policy requirements
  - Graphical User Interface
  - Installation/Un-Installation
  - Software Documentation
  - Test Cases (or Tutorial)
  - Spreadsheet Application
  - Web Application
  - Internationalization (date/number formatting, SI units)

- Requirements vary depending on software type and TR level

- Be aware that these requirements do not completely address the technical and functional requirements that will be needed.
Requirements Phase

- Resources Involved:
  - Requirements Developer
  - Project Managers
  - End Users and Stakeholders

- Who to call for help?
  - SQA for software requirements
  - Legal for licensing questions
  - IT if it is a web application
  - EPRI Software Excellence Network representative for planning, technology, and business compatibility questions

- Forms/Templates:
  - Requirements Specification
Design Phase

- **Key Decisions:**
  - Define how the software will operate, as opposed to requirement, which is what is will do
  - Scope of the design should be commensurate with the needs of the project
    - Primary objective is to communicate
      - Developers, Project manager, Users
Design Phase

- Key Elements to Consider:
  - Some typical elements
    - Architecture
    - Algorithms
    - Database structures
    - Key user screens
    - Software internals
  - Approach depends on the needs of the project
    - Define use-cases
    - Consider personas
    - Write the user manual
    - Prototyping and demonstrate to stakeholders via webcast
  - Design is usually iterative
    - Include frequent user feedback
Design Phase

▪ Resources Involved:
  – Project Team
  – Stakeholders

▪ Who to call for help?
  – EPRI Software Excellence Network

▪ Forms/Templates:
  – Review the requirements, assure that the design reflects all aspects of
    the requirements. Update if required
Implementation & Testing Phase

- Key Decisions:
  - Implementation and testing need to work together to assure proper progress status
  - End user feedback is essential to successful product
Implementation & Testing Phase – Progress Reviews

- Frequency and formality driven by software type, and project size
- Make frequent software builds to facilitate frequent stakeholder feedback
  - Demonstrate features to end users or a regular basis
  - Use multiple “alpha” or “beta” releases to enable user feedback
  - Ensure vendor contracts reflect multiple builds
- Expect changes to project documentation
  - Manage changes to requirements to remain flexible within cost and schedule
  - Leverage the prioritization. Allow flexibility for requirements to change
  - Capture scope changes in project documentation and be clear on what is current version (e.g., SharePoint, shared folders)
- Policy reminder: Don’t release software on servers that circumvent the EPRI product access mappings
Implementation & Testing Phase – Coding

- Enable your developers
  - Developers should be aware of EPRI requirements, coding and/or UI standards, particularly secure coding standards that apply
  - Experienced developers recognize and avoid gold-plating
  - Developers should use some sort of source code control, whether formal or informal, as appropriate
  - Larger and/or critical projects will benefit from regular source code reviews

- Remember the mythical man-month
  - Progress is not a function of time
  - Too much overtime to accommodate poor estimates or scope creep can affect quality, introduce more bugs, and incur more costs in the long run
  - Adding more developers increases the communication factor of a project more than it improves the progress of project, especially late in the schedule

- Test & document product as you go
Implementation & Testing Phase – Testing

- The Project Manager is responsible to assure adequate performance of the software
  - The SQA Team only tests for compliance with EPRI requirements

- Testing should be an on-going activity throughout the development

- Testing includes:
  - Unit testing on individual software functions of modules
  - Functional testing of the entire software package
  - Integration testing with other software tools
  - Regression testing when updates are made to the software
  - Installation testing on target platforms
Implementation & Testing Phase – Testing

- The nature of the testing should be identified in the project plan.
  - Testing needs to demonstrate the completion of the requirements
  - Testing should involve searching for possible errors in the software, not simply verification that a few cases happen to work.
  - To the extent possible, the testing should include testing by 3rd parties not directly involved in the development of the software

- Test cases or tutorial must be provided as part of the project
  - However, in most cases this is not sufficient, and more testing should be performed

- Developers should
  - be aware of how the SQA Team’s standards for usability testing affects their efforts
  - should make sure that any data displayed is correct/complete and verify the results of algorithms/calculations

- Testing should be documented to be repeatable

- Again, the Project Manager is responsible
Implementation & Testing Phase – Documentation

- Externally distributed software requires some form of user documentation that is appropriate for the product
  - Spreadsheets might have a Help worksheet
  - Web apps might have a Help menu that redirects to one or more pages of help
  - A script file might have a readme

- Formal software manuals must use the [EPRI Software Manual Template](#)

- Make sure that documentation, especially screenshots, do not leak confidential information
Implementation & Testing Phase

- **Resources Involved:**
  - Project Team
  - Stakeholders

- **Who to call for help?**
  - SQA for requirements, test expectations, pre-production releases
  - EPRI Software Excellence Network for advice on coding standards, testing scope, and related technical issues.

- **Forms/Templates:**
  - Review the requirements, update if required
  - User documentation templates
Installation & Checkout Phase

This phase represents the acceptance testing prior to release of the software to members (final or pre-production).
In the case of web or server applications, this would include installation on the target environment.

- **Key Decisions:**
  - Verify adherence to EPRI requirements (installation, security, documentation, UI ready for end-uses (based upon type & TRL) )

- **The SQA Team does not perform:**
  - Verification and Validation (V&V) testing
  - Tests with or validation with real world data
  - Exhaustive testing or “white box” (source code) testing

- **Key Risks and Elements to Consider:**
  - Software requirements
  - Software policies
Installation & Checkout Phase

- Preproduction (e.g., Beta) Testing:
  - Not required for all software types, see Software Type Matrix for requirements
  - Beta testing is important to ensure you have met user expectations before completing the project

- Two preproduction testing options available:
  - Option 1: Standard
    - The SQA Team thoroughly reviews applicable EPRI product requirements
  - Option 2: Limited
    - The SQA Team partially reviews applicable EPRI product requirements. Takes shorter time to complete and provide feedback. The EPRI Project Manager accepts the risk that the release may contain anomalies that can result in a Final Acceptance Test failure.
    - Can release with Option 2 and then get fuller review for Option 1
Installation & Checkout Phase

- Final Acceptance Testing:
  - Required for all software types, see Software Type Matrix for specific requirements.

- The software is ready to submit for the Final Acceptance test when:
  - All preproduction test results have been addressed
  - Developer testing has been completed and no further content changes will be made
  - All known errors are dispositioned
  - The test cases run exactly as documented in the software documentation

- Note that security issues are non-negotiable
## Installation & Checkout Phase

### Testing submittals package

<table>
<thead>
<tr>
<th>Form</th>
<th>Required for Preproduction</th>
<th>Required for Final</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software Acceptance Form (SAF)</strong></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td><strong>Software Product Description &amp; Export Coding Form</strong></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Requirements Specifications</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Software Executable/Installation package</td>
<td>X (with pre-production splash screen)</td>
<td>X (remove pre-production splash screen)</td>
</tr>
<tr>
<td>User Documentation (e.g., Software Manual)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Software Source Code* with <strong>Source Code Build Instructions</strong></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Developer response to previous SQA Team report (i.e., Preproduction, &quot;Not Pass&quot; Final)</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>
Installation & Checkout Phase

- **Resources Involved:**
  - SQA Team
  - Project Manager
  - IT if it is a web application

- **Who to call for help?**
  - SQA

- **Forms/Templates:**
  - Software Acceptance Form (SAF)
  - EPSC Software Product Description and Export Coding Form
  - Cryptography and Encryption Functions Checklist and Definitions
Release & Distribution Phase

- **Key Decisions:**
  - Release software for others to use

- **Key Risks and Elements to Consider:**
  - Be aware of 3rd party IP

- **Resources Involved:**
  - SQA Team
  - Electric Power Software Center (EPSC)

- **Who to call for help?**
  - EPSC

- **Forms/Templates:**
  - N/A
Release & Distribution Phase

- EPRI distribution methods:
  - Preproduction Software (e.g., Beta):
    - EPRI Member Center download as a self-extracting zip file (.exe)
  - Production Software (e.g., Final Acceptance):
    - EPRI Member Center download as a self-extracting zip file (.exe)
    - CD-ROM (or DVD-ROM) from the Electric Power Software Center (EPSC)

- Notes:
  - Web application deliverables are accessed via EPRI.com; no physical media sent to EPRI members
  - Production software source code stored by EPSC permanently

- Project manager should download / order and validate release from epri.com
Operations and Maintenance Phase

After completion of the product, operations support needs to be provided, and maintenance/support may need to be provided to end users.

- **Key Decisions:**
  - Support Strategy
  - Operational Strategy (internally used or hosted Software)

- **Resources Involved:**
  - Project Manager
  - End Users and Stakeholders
  - EPRI IT (for internal or hosted products)

- **Who to call for help?**
  - EPRI Software Excellence Network

- **Forms/Templates:**
  - N/A
Operations and Maintenance Phase – Post Delivery Support

- Operations (web or server software)
  - Depending on the product, and ongoing operations plan will be necessary to address
    - User account creation
    - Database updates and backups
    - Server maintenance
    - Hardware obsolescence

- Maintenance
  - Users should be provided with a suitable contact for questions
  - The project responsible for the software product will need to have a plan for
    - Support for answering questions on using the software
    - Method for dispositioning software problems
Retirement Phase – Key Considerations

**Basic Requirements**
- How should the software be dispositioned?
  - Withdraw or replace?
  - Transfer to a commercial entity? e.g., software that is still useful but not a part of EPRI’s core values
  - Open source?
- Retirement plans should be a part of the Software Life Cycle plan

**Considerations**
- Review who has downloaded the software and who has access to the software
  - Know the impact on current users, if any
  - Formally communicate software retirement plans (several years in advance when possible)
- Determine if the SW is connected to any other business process or software
  - Be sure the impact of software retirement has been formally assessed and approved by the respective business
- Contracted resources
  - Look at access rights to EPRI servers and resources, discontinue as appropriate
Retirement Phase – Archiving Process

- Stop the distribution of SW by notifying Electric Power Software Center (EPSC)
  - Deactivated in SAP
  - Removed from the web
  - Software, source code, and project documentation are currently maintained permanently

- SW developed under Billable Service Agreements (BSA) require consideration of the contract requirements
  - Be sure that all obligations are being met (e.g., Third Party Licensing Agreements) before archiving EPRI Software
Retirement Phase

- **Resources Involved:**
  - Users
  - Business Impact Stakeholder(s)

- **Who to call for help?**
  - EPSC (Colette Handy)
  - Sector Product Access administrator

- **Forms/Templates:**
  - N/A
Takeaways!

- Software is great tool to deliver our research.

- The Project Manager is responsible to assure adequate performance of the software.

- Process are in place to help you do a good job, very similar to what you would do for a technical report. See the software development web site at: [http://swdev.epri.com](http://swdev.epri.com)

- Communicate up, down & sideways – the key to success.

- The Software Excellence Network is here to help.

- We are doing research here – not everything will fit easily into the process.
Who to call for help?

- **Contact your Software Excellence Network representative:**
  - PDU: Tina Burnette *(Knoxville)*
  - Nuclear: Jeff Riley *(Palo Alto)*
  - Generation: Justin Thibault *(Charlotte)*
  - Software Quality Assurance: Manuel Morales
  - Nuclear Quality Assurance: Steve Stine
  - Legal: Ellen Brodbine
  - Lead: Rob Austin

- Please complete your feedback form

- Thank you for attending!
Together…Shaping the Future of Electricity
Software Development Website Demonstration

At:  http://swdev.epri.com/