Objectives

- Internal Use Software and Agile
  - ASC 350-40 Internal Use Software
  - What is Agile?
  - Capital Accounting Challenges
  - Duke’s Accounting Convention

- How We Roll Out to the Business
  - Project Accounting Process - *Then*
  - Microsoft InfoPath and SharePoint
  - Request for Capitalization and Accounting Form
  - Review and Approval Process - *Now*
Internal Use Software and Agile
FERC does not provide specific guidance on capitalization of software. Utilities mainly use GAAP guidance ASC 350-40 for internal-use software.

- Planning
- Training
- Data Conversion/New Data
- Department Overheads
- Capitalization ceases: Testing complete vs. punch list
The software guidance is activity based and generally occurs within planned phases:

**Preliminary Project Stage** - Internal and external costs shall be **expensed** as they are incurred.

**Application Development Stage** - Internal and external costs to develop internal-use computer software shall be **capitalized**.

**Post-implementation-Operation Stage** - Internal and external training costs and maintenance costs shall be **expensed** as incurred.
The **waterfall** model is a sequential (non-iterative) design process, used in **software development** processes, in which progress is seen as flowing steadily downwards (like a **waterfall**) through the phases of conception, initiation, analysis, design, construction, testing, production/implementation and maintenance.

What is Agile?

The term "Agile" was applied to a collection of methodologies of project development in early 2001 to consolidate and standardize new approaches cropping up from around the software industry.

- Close collaboration between the development team and business stakeholders
- Frequent delivery of business value
- Tight, self-organizing teams

The Agile Alliance was formed to encourage practitioners to further explore and share ideas and experiences. It also serves to curate resources.

Source: www.agilealliance.org
We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.
Definitions:

**SCRUM**: Scrum is a subset of Agile. It is a lightweight process framework for agile development, and the most widely-used one. *It is iterative and incremental development of software.*

**SPRINT**: An *iteration of work* during which an increment of product functionality is implemented.

Source: https://www.cprime.com
Agile Scrum Framework

1. Starts with a high level product vision and continuously prioritized product Backlog

2. Each 2 weeks, the top business priorities are moved into the next ‘Sprint’ Backlog

3. 15 minute daily ‘Standup’ meetings help measure Sprint progress, facilitate transparency and escalate roadblocks quickly

4. Each Sprint delivers working software, which is showcased in a ‘Demo’. Business feedback is given within the Sprint and during Demos

5. ‘Retrospectives’ are held at the end of each Sprint to facilitate continuous improvement
Waterfall vs. Agile

**Waterfall**
- Plan-driven
- Change adoption challenging
- Not feasible to quickly adapt to dynamic business needs - assumes all requirements are understood in full at the onset
- Value realized at the end
- Value delivered earlier, incrementally and frequently
- Does not encourage innovation
- Risk mitigation may be weak

**Agile**
- Value-driven
- Welcomes change to support dynamic business needs
- Allows for requirements to naturally reveal themselves over time
- Value delivered earlier, incrementally and frequently
- Continuous improvement through constant feedback drives natural innovation
- Early and frequent risk mitigation leads to high quality
Accounting Issues

- In-Service Date
  - Does the delivery of each sprint trigger in-service?

- Project tracking
  - Should each release be considered a separate chargeable project?
ASC 350-40 Internal Use Software:

- “Capitalization stops when software is substantially complete and ready for its intended use. Computer software is considered ready for its intended use after all “substantial testing” is completed. This should occur no later than the in-service date and is the end of the application development stage.”

FERC CFR Electric Plant Instruc. No. 3:

- “When only part of a plant or project is placed in operation or is completed and ready for service…that part of the cost of the property placed in operation or ready for service shall be treated as Electric plant in Service and AFUDC thereon…shall cease.”
Does the delivery of each sprint trigger in-service?

**ASC 350-40 Internal Use Software:**

- “For each module or component of a software project, amortization shall begin when the computer software is ready for its intended use, regardless of whether the software will be placed in service in planned stages that may extend beyond a reporting period. For purposes of this Subtopic, computer software is ready for its intended use after all substantial testing is completed.”

- “If the functionality of a module is entirely dependent on the completion of other modules, amortization of that module should begin when both that module and the other modules upon which it is functionally dependent are ready for their intended use.”
In-Service Date: Does the delivery of each sprint trigger in-service?

**Answer:**

- **Functionally dependent** components should be delivered with **one** in-service date.
  - AFUDC continues until all components are delivered
  - Amortization should not begin until all components are delivered

- **Non-functionally dependent** components should be placed into service upon completion.
  - AFUDC should cease when each component is released to production
  - Amortization should begin when each component is released to production
Project tracking: Should each release be considered a separate chargeable project?

• Answer:
  • *Not unless materiality dictates.*
  • Identify financial statement risk based on Duke’s planned software budget and only charge separate projects when necessary.
Capital Accounting Challenges – Solutions

- Capital Project – Multiple Releases
  - Not Functionally Dependent
    - Low Dollar, Short Duration
      - One In-Service / One Project
    - Medium Dollar
      - Blanket Method – One Project with quarterly in-service
    - High Dollar
      - Case by Case
  - Functionally Dependent
    - One In-Service / One Project
“An accounting convention is a common practice used as a guideline when recording a business transaction. It is used when there is not a definitive guideline in the accounting standards that govern a specific situation. Thus, accounting conventions serve to fill in the gaps not yet addressed by accounting standards.” – www.accountingtools.com

Duke’s convention compares AFUDC and depreciation impacts to materiality threshold quarterly; conventions are reported annually
How We Roll Out to the Business
IT PAC Team

• IT PAC works as an advocate for IT and is the first point of contact for IT when assistance is needed
  • Meet with IT leaders to provide updates on timely topics
  • Train and assist users through process of requesting accounting
  • Review all capital accounting requests before sending to Asset Accounting
  • Provide training to IT on capitalization and the Accounting Request Form

★ IT PAC must have strong partnership with Asset Accounting
IT Accounting Request Process - Then

- **IT Project Team**
  - Prepared Accounting Request Forms

- **IT Management**
  - Reviewed by IT Project Manager
  - Signed approval by appropriate level DOA

- **IT Finance Project Accounting Team**
  - Project accounting set up in Powerplan

- **Asset Accounting**
  - Review and Approval

- **Accounting Code**
  - Provided to IT Project Team
Microsoft InfoPath is a software application for designing, completing and submitting electronic forms

- **Controls** to tailor data requests include text boxes, radio buttons and check boxes
- **Rules** built to customize the form to display specific data fields dependent on prior responses
- **Workflows** designed to move form through creation to completion and prepare for any future updates

Integrates with Microsoft SharePoint

- Enables SharePoint lists and libraries
# Accounting Request Form – General Information

- **Top third of the form is general information**

## General Information

<table>
<thead>
<tr>
<th>Project Description:</th>
<th>Classification:</th>
<th>Project Start Date:</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEST - Project Description</td>
<td>Invest</td>
<td>3/1/2019</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Department:</th>
<th>Project delivery method:</th>
<th>Estimated In-Service Date:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Technology</td>
<td>Agile</td>
<td>6/30/2020</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Project Manager:</th>
<th>PMO Manager:</th>
<th>IT Sponsor:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acker-Donnelly, Kate</td>
<td>Acker-Donnelly, Kate</td>
<td>Acker-Donnelly, Kate</td>
</tr>
</tbody>
</table>

The project scope should be informative and with enough detail to enable reviewers and approvers to understand what the project is and determine the capital versus O&M components.

### Project Type

- [ ] Agile Product Team
- [ ] Communication
- [x] Hardware
- [ ] Office Equipment
- [ ] Security & Surveillance Equipment
- [x] Software

### Purpose for Project

- [x] Addition of new asset
- [ ] Complete replacement of existing asset
- [ ] End of life of existing asset
- [ ] Failure of existing asset
- [ ] Hardware Initiative

### This project has special considerations:

Free form field to include additional instruction for setting up the project.
Accounting Request Form – Software Details Section

- Software Details Section – Let’s get started

### Software Details

This software project consists of the following:
(Select all that apply)

- Upgrade/Enhancement to Existing Duke-Owned Software
- Purchase - Duke Owned (Includes Purchase of Additional Licenses)
- Internally Developed
- Purchase - Vendor Hosted/Leased, Cloud Computing, or SaaS
- O&M Project Only

What is the expected life of the software (years)?  
What is the total Capital Expenditure?

<table>
<thead>
<tr>
<th>Less than $150,000</th>
<th>$150,000 - $</th>
<th>$ - $</th>
<th>Greater than $</th>
</tr>
</thead>
</table>

- **O&M**
- **Low $$** One Project One In-Service
- **Medium $$** One Project Qtrly In-Service
- **High $$** Case by Case
### Accounting Request Form – Software Details Section

- **Tell me more**

<table>
<thead>
<tr>
<th>Question</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Will you purchase maintenance contracts/agreements?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Will you have data conversion activities?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Are you developing conversion software?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Are you running data conversion programs?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Are you cleaning up bad data?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Will you have training costs?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For technical training on how to develop/install the product?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>For training the trainer?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>For training the end user on how to use the application?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Will you be creating system documentation materials?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Is there an associated retirement of existing software?</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Are you replacing any existing interfaces?</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
**Accounting Request Form – Software Details Section**

- When and what are the planned releases?
- What is the new functionality that is being built? How is the new software different from the currently existing software?

<table>
<thead>
<tr>
<th>Release #</th>
<th>Release Date</th>
<th>Capabilities</th>
<th>Capital Cost of release</th>
<th>O&amp;M Cost of release</th>
<th>Comments</th>
</tr>
</thead>
</table>

- Add another release
- New functionality provided by this Upgrade/Enhancement to existing Duke owned software
  - Benefit provided
  - Current State
  - Future State

- Add a feature
### Accounting Request Form – Estimated Costs Sections

#### Material Details
- **Division**: [Dropdown]
- **Resp Ctrl ID**: [Dropdown]
- **Resp Ctrl Name**: [Dropdown]
- **Cost**: [Dropdown]
- **Qty**: [Dropdown]
- **UoP Details**: [Dropdown]
- **UoP Description**: [Dropdown]
- **Unit Cost**: [Dropdown]
- **Total**: [Dropdown]

#### Labor / Task Details (Not related to Software)
- **Division**: [Dropdown]
- **Resp Ctrl ID**: [Dropdown]
- **Resp Center Name**: [Dropdown]
- **Work Task**: [Dropdown]
- **Charge Type**: [Dropdown]
- **Estimated Cost**: [Dropdown]

#### Software related Labor / Task Details
- **Division**: [Dropdown]
- **Resp Ctrl ID**: [Dropdown]
- **Resp Center Name**: [Dropdown]
- **Work Task**: [Dropdown]
- **Charge Type**: [Dropdown]
- **Estimated Cost**: [Dropdown]

- **Add another Unit of Property UoP**
- **Add another asset location**
- **Show Warranty Attestation**

- **Add New Task**
- **Add another Software Labor Task**
- **Add another Software Project Type**

Section Valid: false
<table>
<thead>
<tr>
<th>Name</th>
<th>Title</th>
<th>DOA Authority Amount</th>
<th>Date/Time approved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>4/15/2019 15:44:07</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4/9/2019 1:15:52 PM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4/9/2019 17:07:26</td>
</tr>
<tr>
<td>Project Accounting Cycle</td>
<td>Approved - Standard Replacement ...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asset Accounting Research</td>
<td>Approved - Standard Replacement ...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
▪ Workflows are built to move and track the Accounting Request Form through all its phases
Next Challenge – Agile “Projects” versus Durable Teams
Agile “Projects” versus Agile Durable Teams

Agile Projects
- Defined start and end dates
- Scope of project is established at beginning
- Capital estimated and approved at the beginning of the project

Agile Durable Teams
- Long-standing team
- Lack of clarity of product work up front
- Capital estimate and approval is an ongoing process