As attendees enter, hand each one a card w/ pain points listed on one side and a community dialog starter on the other

Short Link: http://bit.do/bpe2019
Session overview

1. Value of Documentation
2. Approaching Documentation
3. Tools for Creating and Maintaining Documentation
4. Workflow Portals and Resources
5. Activity
Documentation can be difficult. It involves people and process and arriving at a shared understanding. That can take time.

Working together to achieve that shared understanding brings team cohesion, accountability, and lays a foundation for growth.

Creating sustainable documentation is an investment of time and energy; there are direct and opportunity costs; but, investments have payoffs.

Investing in usable, iterative, and understandable documentation can help you make the pivot from digital preservation projects into an ongoing program of managed activities.

Good documentation makes it easier to collaborate with new colleagues, transition functions to new roles, transfer knowledge, and onboard new employees.

Good documentation can’t solve all problems, but without it, you are limiting your potential.

The value of documentation is directly related to program growth and sustainability.
How you build out your documentation can affect adoption. And adoption will make or break everything.

Pick a format/style that fits the best for the people using it and is conducive to the work.

It needs to be something people will want to use, this isn’t necessarily the best time to introduce a new tool.

- It can work, depending on the person and their style/aptitude. But for others, it can make them feel alienated and be counter productive.

Remember what hasn’t worked before and more importantly, why it didn’t work. (You’ll dig into this a bit later.)

- You’ll need an exit strategy, because, life happens and things will change.

Whatever you choose, it should be:

- Simultaneously accessible by multiple people
- Easy to update/iterate
- Easy to reference (page, sections, lines)

It may be beneficially to have some roles/permissions or other ability to restrict access. This may sound counter-intuitive, but think about systems security and procedural documentation.

Don’t fall into old habits and practices, leverage the features inherit to the tool you are using.

But, remember your exit strategy, so only use the inherit features to enhance documentation.
Think about how your overall documentation fits with your training material. Are they the same? Are they different? I don’t know the answer, but try not to repeat your own material. There’s probably a way to weave them.
Tools for Creating and Maintaining Documentation

- Version Control Systems
  - Github
    - Archivists’ Guide to Kyroflux
    - UC Guidelines for Born-Digital Archival Description
  - Gitlab
    - DigiPres Team @ PSU
  - Bitbucket

- “Git is a distributed version-control system for tracking changes in source code during software development.” It’s very good at keeping a history of files and changes made over time, who made those changes, and why.
- As a distributed system, there can be many copies of a git repository. Maintainers can keep local copies on their workstation and push copies to a central (web-hosted) git repository, like Github or Gitlab. Or, people can make changes directly in Github or Gitlab. (There are also alternatives, like Bitbucket.)
  - Bitbucket is from the company that owns JIRA and Trello, might be easier to integrate if you use those tools.
- Github and Gitlab are similar but differ in their features, both are geared towards software, but their features are useful for other applications. Both are also open source and many organization have local instances. These local instances make it easier to restrict access to organizational users.
  - Issues may be helpful if you don’t already use some form of a project management tool.
  - Projects -- organize issues into project with a Kanban board
  - Github Pages allow you to create a more web-page like presentation of content in a github repository
- Wikis are a feature of Github and Gitlab, as I’ll show. But a brief thought on stand-alone wikis. They do have some similarities to version control systems. They sometimes require learning a specific markup syntax; but even more concerning, the most current one I could find when looking for an example for
this presentation was from 2017… In general, Wikis rode the Web 2.0 wave and people just don’t really like using them.

- **Archivists Guide to Kyroflux**
  - Repository
    - README -- to render content on the web
    - Markdown
  - Commits -- people saving their local changes to the central git repository (Github)
  - Contributors
  - Pull requests

- **UC Guidelines for Born-Digital Archival Description**
  - Branches
  - License
  - Fork

- **DigiPres Team @ PSU**
  - More of a work-in-progress, but that shouldn’t stop you from getting started. You can figure it out as you go if everyone is onboard.
    - Accessible only to PSU users for now, hope to open up when settled a bit more.
    - Still not sure what things can’t be public.
  - Repository is for more formal documentation like policy.
    - General and specific guidelines
  - Integrated Wiki is for more informal or drafty documentation.
    - Wiki still has a history and actually has its own git repository, just isn’t exposed in the GUI
    - Links to superceded documentation
    - Draft or working material that may eventually be moved to repository if formalized.
  - Easy to share with other library colleagues and promote shared standards across a 24-library system
  - Separate repository to capture commands, scripts, and snippets that are useful one-offs we might want to use again
    - [https://git.psu.edu/digipres/scripts](https://git.psu.edu/digipres/scripts)
Tools for Creating and Maintaining Documentation

- Websites, Blogs, and Content Management Systems
  - Intranets
  - Digital Preservation Coalition Knowledge Base
  - DigitizeUC
- Shared/Cloud Storage and Word Processing/Office Productivity Tools
  - Google Drive / Office 365
  - Box / Dropbox / Network Folders

- Content Management Systems generally have version control built in and can have fine-tuned roles/permissions. But intranets in particular may not expose all the CMS features to users and may require assistance from web developers.
  - CMS usability and accessibility varies. (Adobe CQ anyone?)
  - Funny things can happen when copy/pasting. If you end up having to author documents in somewhere else, this can lead to problems. This option works best if you can author documents in the CMS.
  - Intranets can also make it difficult if you later decide you want to share your documentation externally.

- DPC Knowledge Base
  - WordPress has many themes and plugins for creating documentation, knowledge bases, and similar. They can be attractive, most have some free option, and can include nice options like PDF downloads.
    - But, might not be easily portable.
  - Like most CMS, can be very flexible if you have developers or the technical savvy.

- DigitizeUC
  - Internal audience, public URL, but unlisted and blocked from crawlers.
  - Easy to remember URL, easy to share.
  - Only editable but a small group of people, was maintained by one person. That person left, hasn’t been updated since.
    - Points to multiple issues. Adoption is key. Get more people
involved. Use a collaborative approach and tool.

- Google Drive / Office 365 /
  - If using these types of tools, leverage modern features like collaborative authoring/editing, versioning, and sharing.
    - Don’t use these tools the same way you have for the past 20 years.
    - Don’t make separate copies for new versions.
    - Don’t make separate copies! Make one copy, share/link it.
      - In the old model of word docs and network folders, copies proliferated, hard to tell what was authoritative.
  - Don’t print unless you need to! Who has a documentation binder? How often do you reference/use it vs add new/replace pages?
Workflow Portals and Resources

- COW: Community Owned Workflows
- Library Workflow Exchange
- open-workflows
- OSSArcFlow

Beg, borrow, and steal!

General tips (aimed at software developers, but good stuff)
What nobody tells you about documentation?
https://www.divio.com/blog/documentation/
Writing Documentation When You Aren’t a Technical Writer: Part One
https://stoplight.io/blog/writing-documentation-when-you-arent-a-technical-writer-part-one-ef08a09870d1/
Writing Documentation When You Aren’t a Technical Writer: Part Two
https://stoplight.io/blog/writing-documentation-when-you-arent-a-technical-writer-part-two-59997587cc2a/
Small group discussions

Transition slide to discussion groups
Establish that time / lack of resources is always a “pain point”

Identify areas for each discussion group

Give people time to move to areas....

Choose an issue you would like to discuss
1. Documentation format and approaches for versioning
2. Getting other staff involved in documentation work or activities
3. Determining the granularity or what level of detail is needed for documentation
Divide into groups of three people or fewer.

Arrange participants into groups of three within each discussion topic.
Carly, Sam, Nathan circle through groups and assist in discussion if needed

Carly keep a timer
Final discussion with whole group

Keep conversation going
- session 30 on wednesday will include a specific example of organizational documentation from the Rockefeller Archive Center
- Fill out Community Dialogue portion of form and hand it back into us. We will record best advice and share via twitter or email for those who request.
Participants responded to the question: What was the best piece of advice you heard today?

- **Staff participation in preservation documentation**
  - Stressing importance of leaving bread crumbs for others to follow in the future and getting buy-in through that idea.
  - You just need to get buy-in for part of it and then you can continue to build on it.
  - Get better buy-in through clearer communication between the various partners engaging in digitization and resulting preservation needs.
  - Remember that documentation can be political!

- **Granularity of preservation documentation**
  - Think about staff learning styles when designing documentation.
  - Each person sees a workflow differently, so you will need multiple people to collaborate to create documentation that is detailed enough to last throughout turnover.
  - Set benchmarks throughout the project to get small wins that add up