Data Management: File Organization

Christine Malinowski

January 18, 2017

MITLibraries

IAP 2017

- **Researcher Funder Open Access Requirements from NASA, DOE, and Other Federal Agencies**
  Tue, Jan 24, 11am-12pm, 2-146

- **Data Management: Strategies for Data Sharing and Storage**
  Wed, Jan 25, 1-2pm, 14N-132

- **LaTeX/BibTeX & Citation Management Tools**
  Thu, Jan 26, 4-5pm, 14N-132

- **Manage your PDFs and Citations: Zotero and Mendeley**
  Wed, Jan 25, 10am-12pm, 14N-132 (in person)
  Mon, Jan 30, 2-3pm, WebEx

MIT Libraries IAP classes

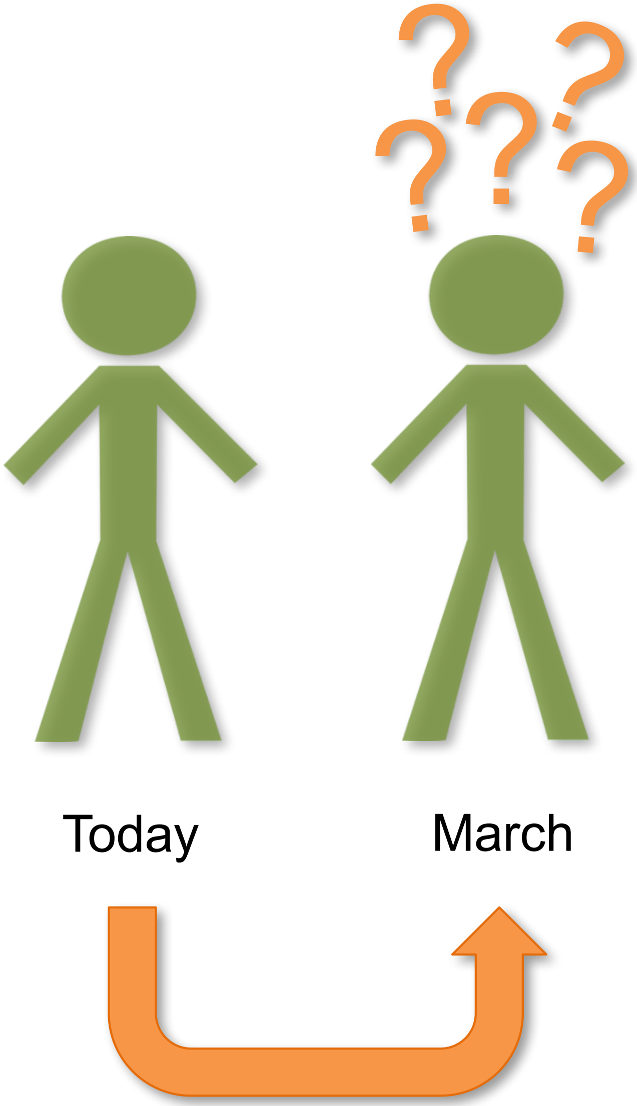
MITLibraries

Data Management Services @ MIT Libraries

- Workshops
- Web guide: http://libraries.mit.edu/data-management
- Individual consultations
  - includes help with creating data management plans

Contact: data-management@mit.edu

Why file organization is important

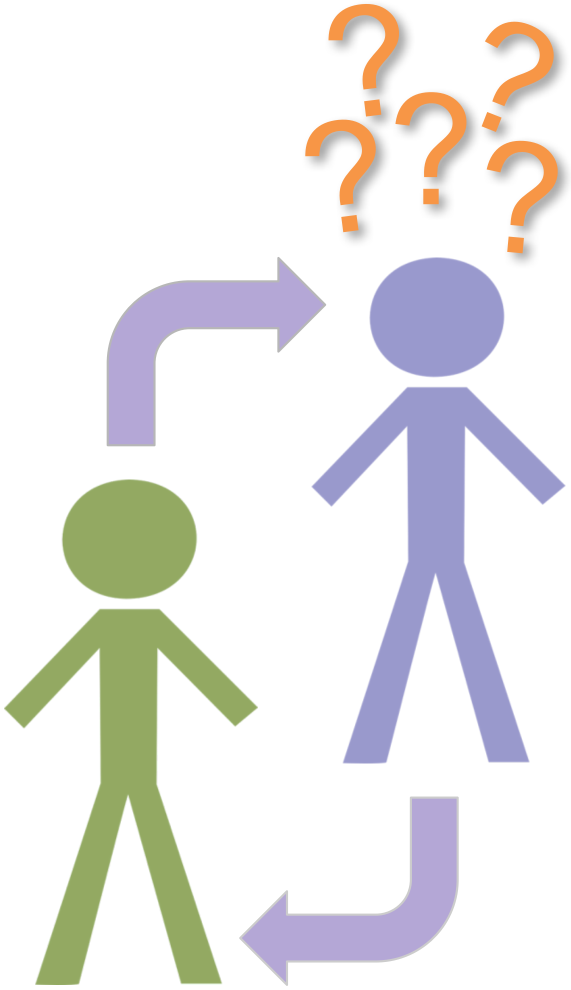


The first person with whom you will share your data is yourself.

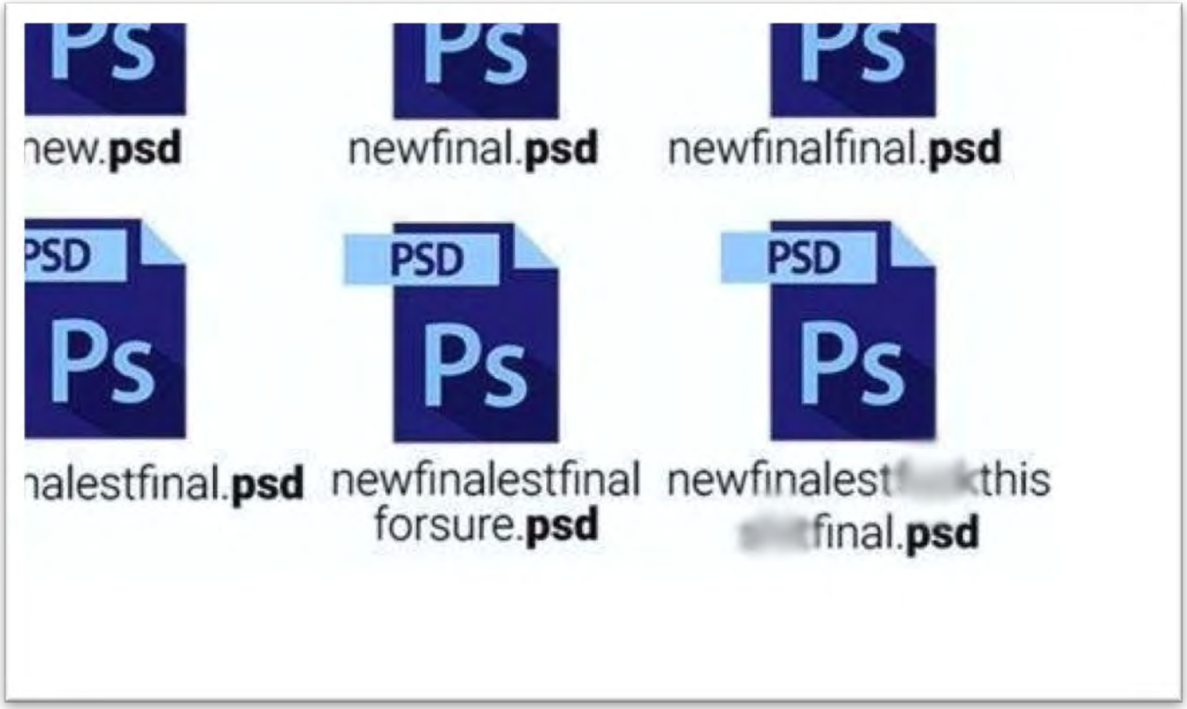
Why file organization is important

Can someone else understand/use your data files?

Now?
Tomorrow?
In 5 years?



Why file organization is important


Ps
new.psd
newfinal.psd
newfinalfinal.psd
PSD Ps
nalestfinal.psd
newfinalestfinal forsure.psd
newfinalest[redacted]this [redacted]final.psd

@AksharPathak
YASH BHARDWAJ & JUGAAD POSTERS

Once your research gets underway, there may be multiple files in various formats, multiple versions, methodologies, etc., all relating to your research.

Key principles of file organization

Spending a little time upfront, can save a lot of time later on.

Be realistic: strike a balance between doing too much and too little.

There’s no single right way to do it; establish a system that works for you.

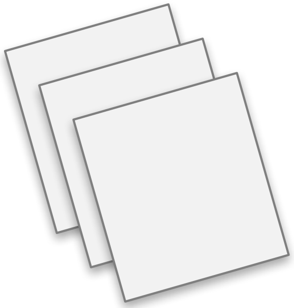
Think about who your system needs to work for: Just you? You and your lab group? Collaborators?

What do we mean by file organization?

File structures

File naming

File versioning



File structures

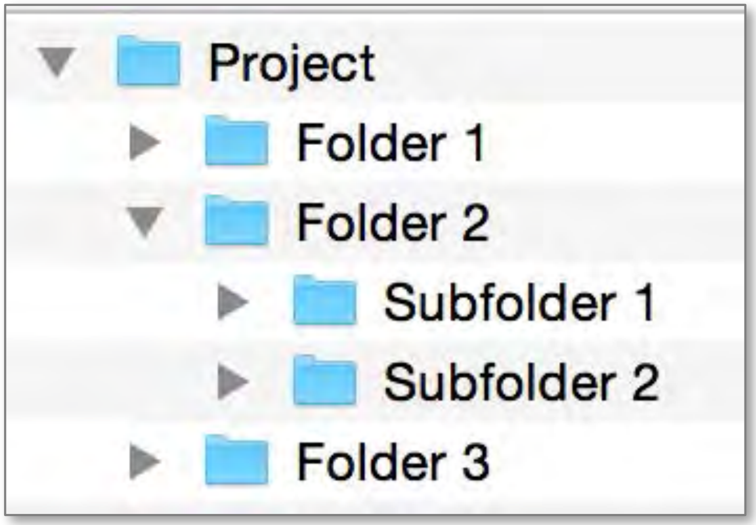
where to put data so you can find it

Method 1: Hierarchical

Items organized in folders and subfolders

Benefits:

- Familiar & widely used
- Good at representing the structure of information
- Similar items are stored together
- Subfolders can function as task lists

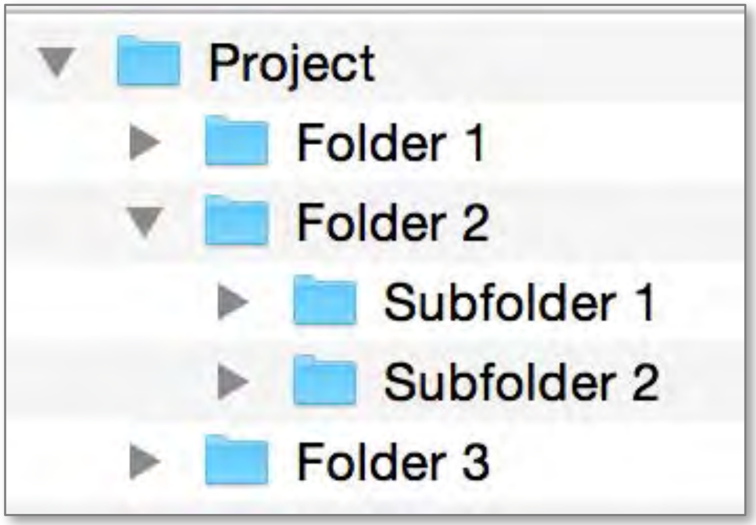


Method 1: Hierarchical

Items organized in folders and subfolders

Drawbacks:

- Surprisingly hard to set up
- Challenging to get the right balance between breadth & depth
- Items can only go in one place
- Time consuming to reorganize if the hierarchy becomes out of date

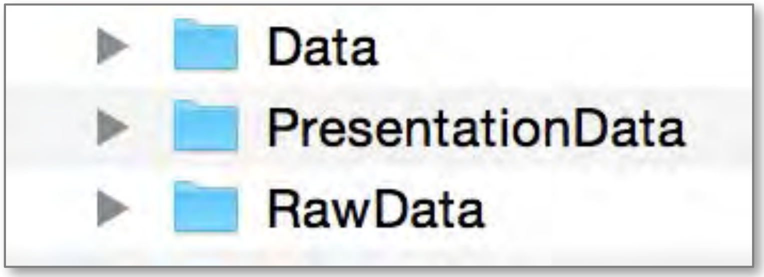


- ▼ Project
  - ▶ Folder 1
  - ▼ Folder 2
    - ▶ Subfolder 1
    - ▶ Subfolder 2
  - ▶ Folder 3

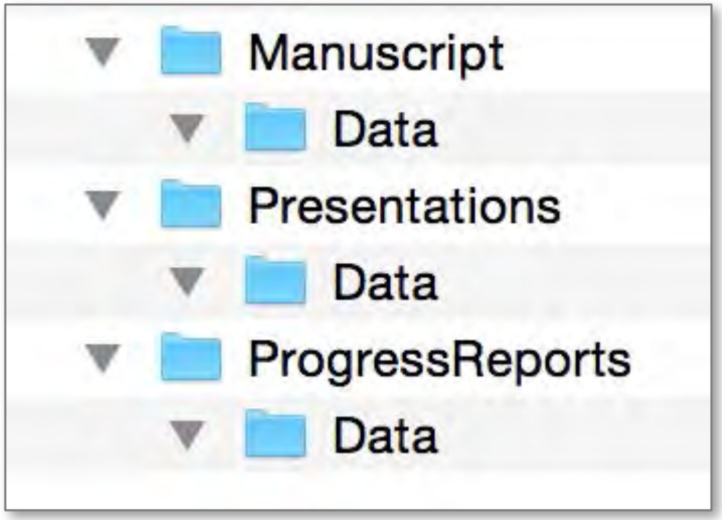
Method 1: Hierarchical

Best practices

- Avoid overlapping categories



- Data
- PresentationData
- RawData

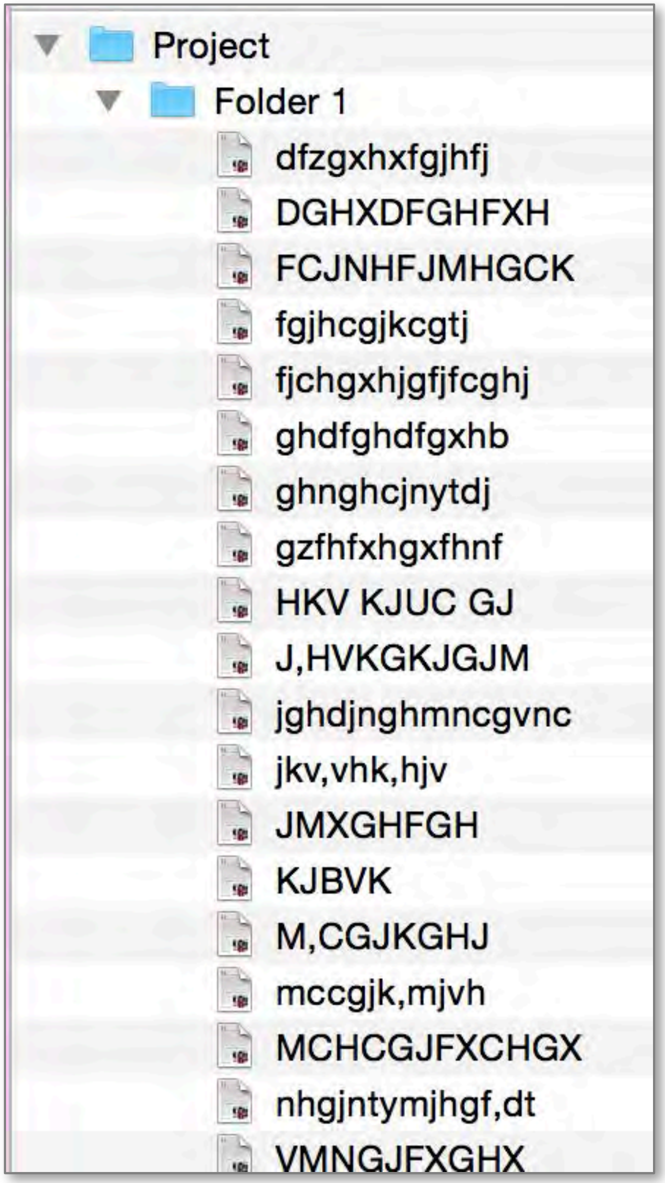


- Manuscript
  - Data
- Presentations
  - Data
- ProgressReports
  - Data

Method 1: Hierarchical

Best practices

- Avoid overlapping categories
- Don’t let your folders get too big

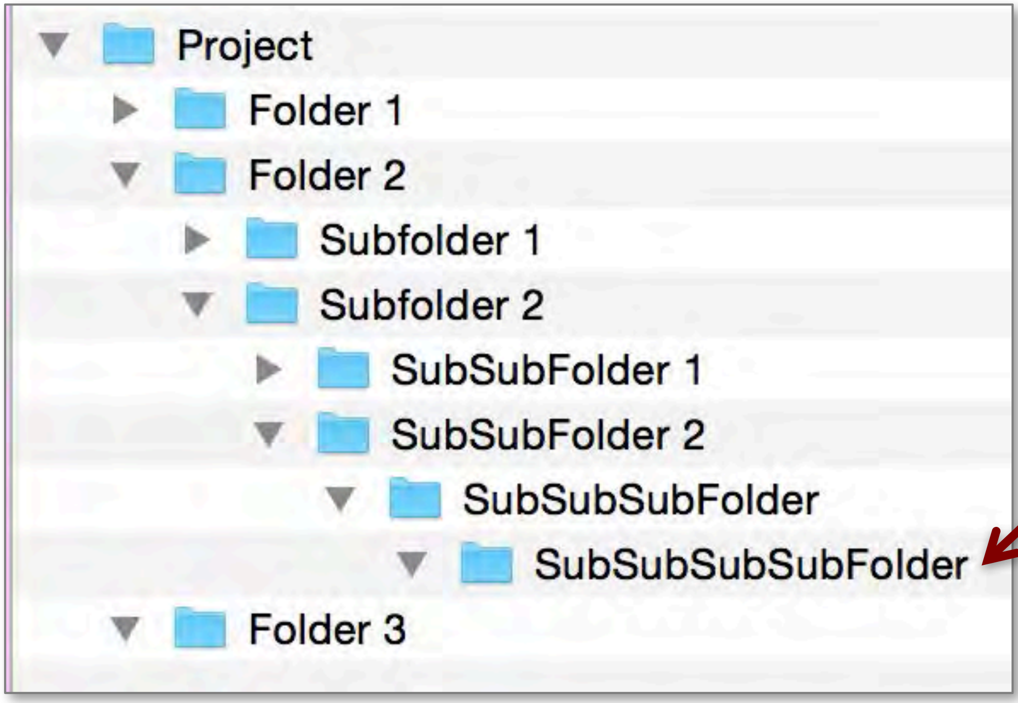


Project
Folder 1
dfzgxhxfgjhfj
DGHXDFGHFXH
FCJNHFJMHGCK
fgjhcgjkcgtj
fjchgxhjgfjfcghj
ghdfghdfgxhb
ghnghcjnytdj
gzfhfxhgxfhnf
HKV KJUC GJ
J,HVKGKJGJM
jghdjnghmncgvnc
jkv,vhk,hjv
JMXGHFGH
KJBVK
M,CGJKGHJ
mccgjk,mjvh
MCHCGJFXCHGX
nhgjntymjhgf,dt
VMNGJFXGHX

Method 1: Hierarchical

Best practices

- Avoid overlapping categories
- Don’t let your folders get too big
- Don’t let your structure get too deep



- Project
  - Folder 1
  - Folder 2
    - Subfolder 1
    - Subfolder 2
      - SubSubFolder 1
      - SubSubFolder 2
        - SubSubSubFolder
          - SubSubSubSubFolder
  - Folder 3

How many clicks does it take to get there?

Creating a systematic file folder structure

Steps for defining your system:

1. Define the types of data and file formats
2. Include important contextual information
3. Organize folders by meaningful categories

    primary/secondary/tertiary

    subject/collection method/time

4. Choose a directory naming convention

Be Clear, Concise, Consistent, Correct, Conformant

1. Define the types of data and file formats

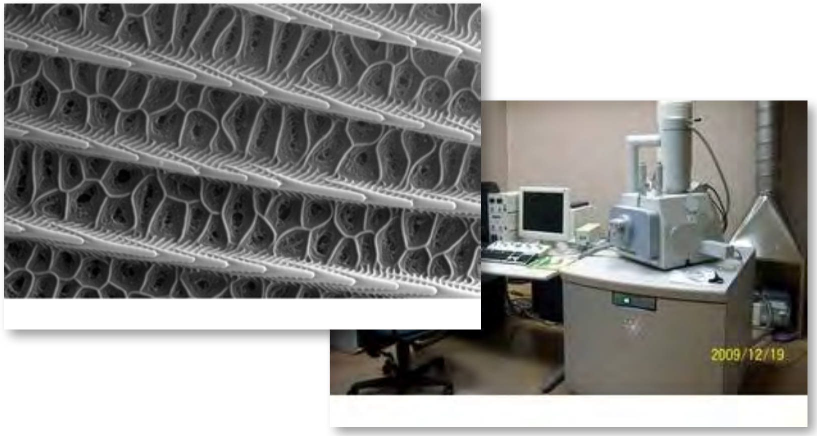
Images from the field (.jpeg)

Progress reports & presentations (.docx, .pptx & .pdf)

Field observations (.xlsx → .csv)

NOAA climate data (.csv & .txt)

Analysis files & graphics (.xlsx & .R)



Microscopy images
(proprietary format & .tiff)

Literature (.pdf)

2. Include important contextual information

When you* are looking for *X file*, how do you think about it?

- As part of X study/location?
- By its type (e.g., presentation figures, report, raw data, analyzed data)?

Example information:

- Date
- Collection method
- Collector
- …

* Also consider how others in your team think about these files

3. Organize folders by meaningful categories

Primary / Secondary / Tertiary

[Project] / [Sub-project] / [Experiment] / [Instrument] / [Date]

[Research area] / [Project] / [Data or documentation] / [Date]

[Project] / [Type of file] / [Data collector name] / [Date]

/ butterfly / images / cmalin / 20170117
/ butterfly / tabular / cmalin / 20170117
/ butterfly / projectDocs /
/ butterfly / literature /

A quick word on organizing/storing articles

Would I really want to store my literature files simply in a directory?

Maybe, but...
…consider using citation management tools

MENDELEY

zotero

http://libguides.mit.edu/references
personal-content@mit.edu

Method 2: Tag-based

Each item assigned one or more tags

Benefits:

- Items can go in more than one category
- Can be quicker/easier to set up
- When collaborating, it can be easier to combine than hierarchical systems

File Organization

org

file org

versioning

File Names

Method 2: Tag-based

Each item assigned one or more tags

Drawbacks:

- Not how operating systems store files
- If item isn’t tagged properly when first acquired, it can be hard to find
- Increased risk of inconsistency
- Less good at representing the structure of information

File Organization

org

file org

versioning

File Names

Tag-based system examples

Social media platforms (e.g., Twitter, Instagram)
#TagsEverywhere

Journal Article keywords

Citation Management tools (e.g., Zotero, Mendeley)

Notetaking tools (e.g., Evernote)

Gmail labels

Method 2: Tag-based

Creating a tag-based system:

1. Determine the contextual information by which you want to discover your files

2. Create a consistent naming convention for these contextual categories

3. Tag your files!

   In OS: Add searchable keywords/tags to file information

See our guide to Tagging and Finding Your Files: http://libguides.mit.edu/metadataTools/

File naming

what to call data so you know what it is

File naming conventions

Naming conventions make life easier!

Naming conventions should be:

- **Descriptive**
- Consistent

Consider including:

- Unique identifier (ie. Project Name or Grant # in folder name)
- Project or research data name
- Conditions (Lab instrument, Solvent, Temperature, etc.)
- Run of experiment (sequential)
- Date (in file properties too)
- Version #

File naming conventions

Naming conventions make life easier!

Naming conventions should be:

- Descriptive
- **Consistent**

YYYYMMDD
MMDDYYYY
YYMMDD
MMDDYY
MMDD
DDMM

Maintain order

TimeDate
DateProjectID
TimeProjectID

Sample001234
Sample01234
Sample1234

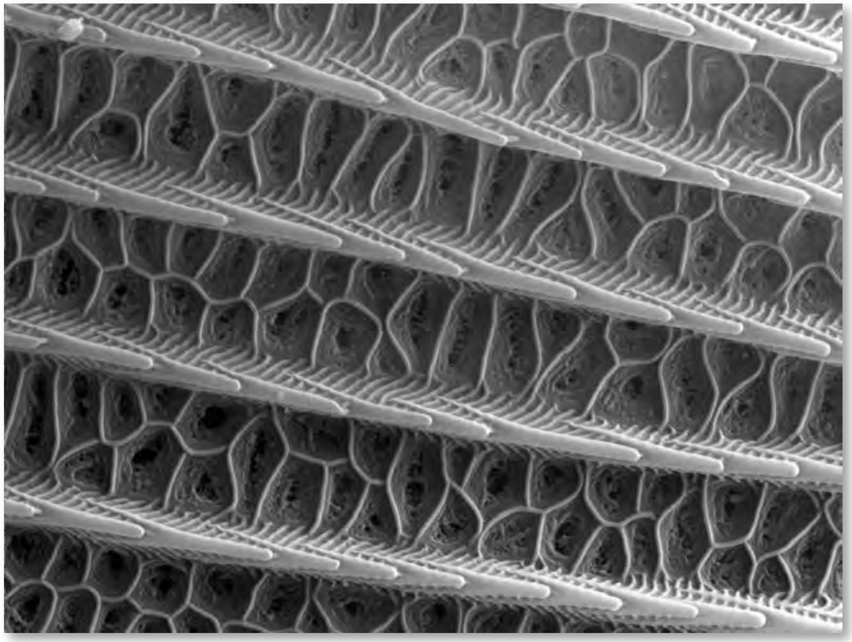
Include the same information

File naming conventions

Best Practice	Example
Limit the file name to 32 characters (preferably less!)	32CharactersLooksExactlyLikeThis.csv
When using sequential numbering, **use leading zeros** to allow for multi-digit versions For a sequence of 1-10: 01-10 For a sequence of 1-100: 001-010-100	**NO** ProjID_1.csv ProjID_12.csv **YES** ProjID_01.csv ProjID_12.csv
Don't use special characters & , * % # ; * ( ) ! @$ ^ ~ ' { } [ ] ? < > -	**NO** name&date@location.doc
Use only one period and use it before the file extension	**NO** name.date.doc **NO** name_date..doc **YES** name_date.doc
Avoid using generic data file names that may conflict when moved from one location to another	**NO** MyData.csv **YES** ProjID_date.csv

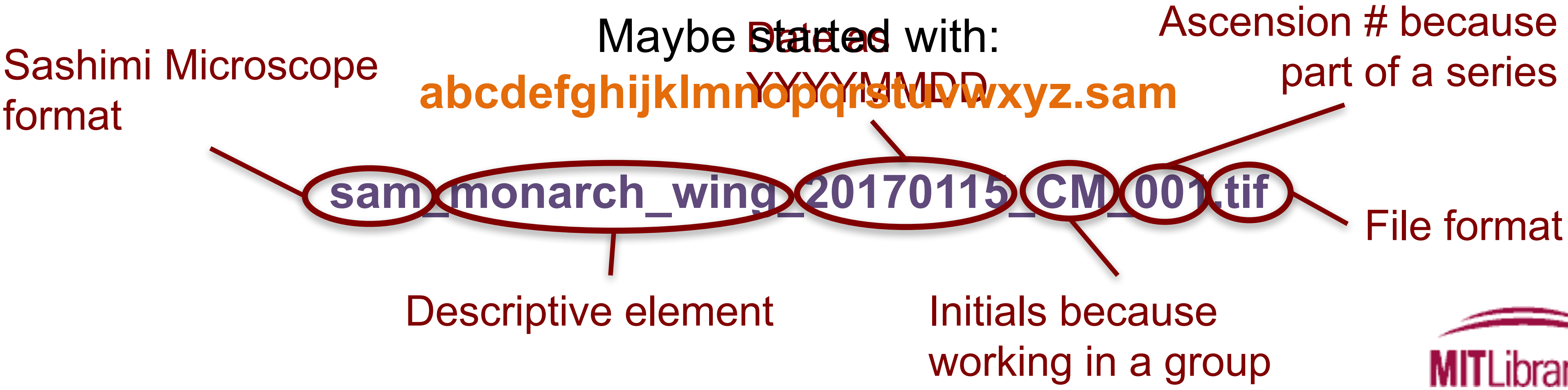
MITLibraries

For example…



Maybe started with:

abcdefghijklmnopqrstuvwxyz.sam



Date

YYYYMMDD

Sashimi Microscope format

Ascension # because part of a series

sam_monarch_wing_20170115_CM_001.tif

File format

Descriptive element

Initials because working in a group

For example…

sam_monarch_wing_20160115_CM_001.tif
[instrument]_[item]_[date]_[collector]_[ascension#].ext

FileOrgSlides_20170118.pptx
[class][material]_[date].ext

SevilletaLTER_NM_2001_NPP.csv
[project name]_[state]_[year]_[dataset].ext
SevilletaLTER_NM_2001_NPP_20170117.csv
[project name]_[state]_[year]_[dataset]_[analysisID].ext

Use abbreviations and acronyms consistently!

File naming & discipline standards

Check for established file naming conventions in your discipline

Some examples:

- DOE's Atmospheric Radiation Measurement (ARM) program
- GIS datasets from Massachusetts
- The Open Biological and Biomedical Ontologies

File naming & instruments

Check to see if your instrument, software, or other equipment that outputs your data files can be set with a file naming system

Less work than retrospectively changing filenames

But if you still have to change many file names downstream…

File naming & batch/bulk renaming

Can use tools that retrospectively align file/folder names with naming conventions

Caveats:

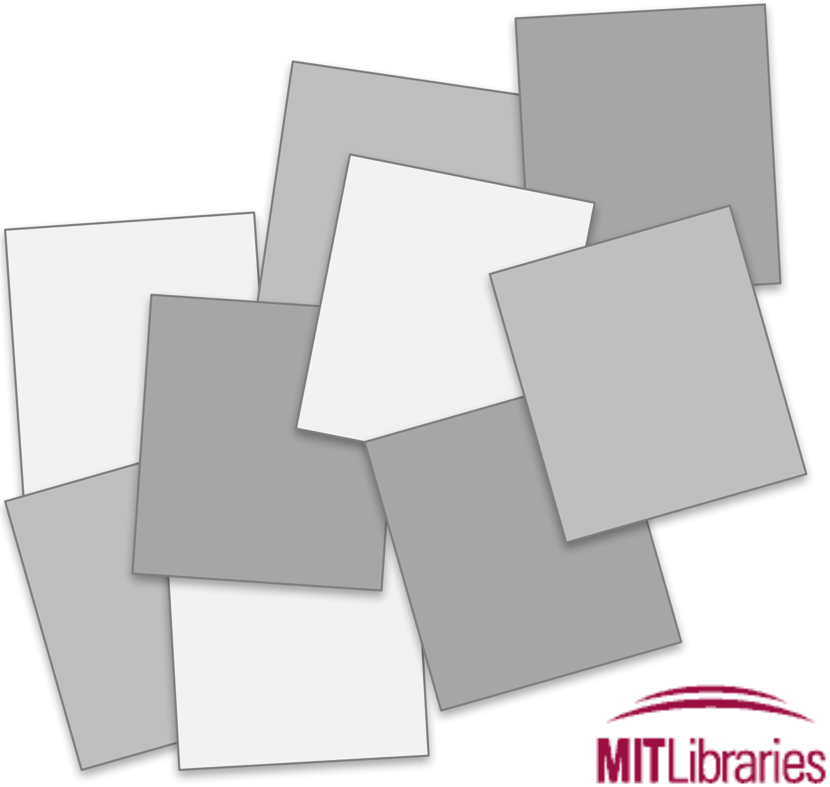
- Ideally you want to be able to map the original to new names
- Make sure it doesn't change the file extension

Some File Renaming Tools:

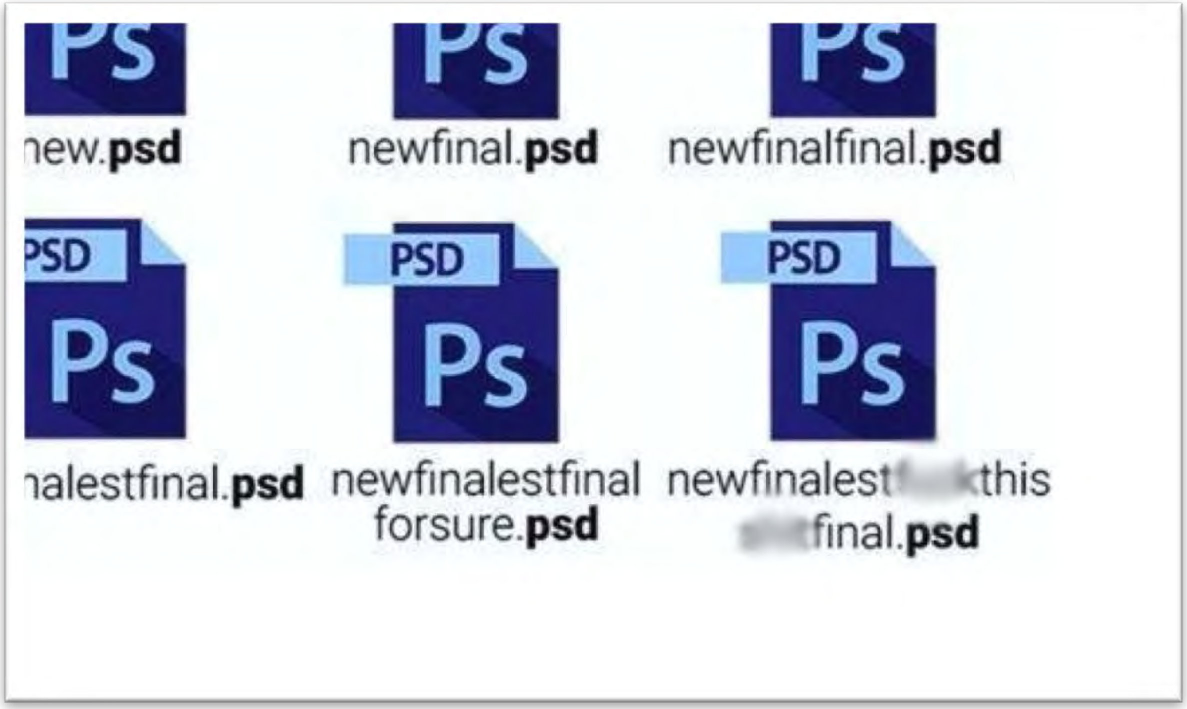
Bulk Rename Utility
Renamer
PSRenamer
WildRename

File versioning

keeping track of data



Versioning: *the why*



ew.psd

newfinal.psd

newfinalfinal.psd

alestfinal.psd

newfinalestfinal forsure.psd

newfinalest[redacted]this [redacted]final.psd

@AksharPathak
YASH BHARDWAJ & JUGAAD POSTERS

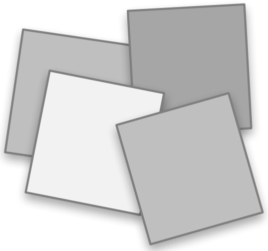
Versioning: *the when*

Depending upon practices in your field, version either:

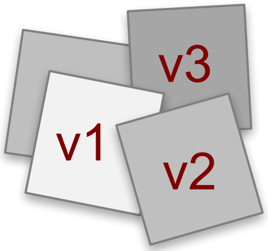
- Analysis/program/script files
- Data files themselves

Also important for project documentation and files

Versioning: *the how*



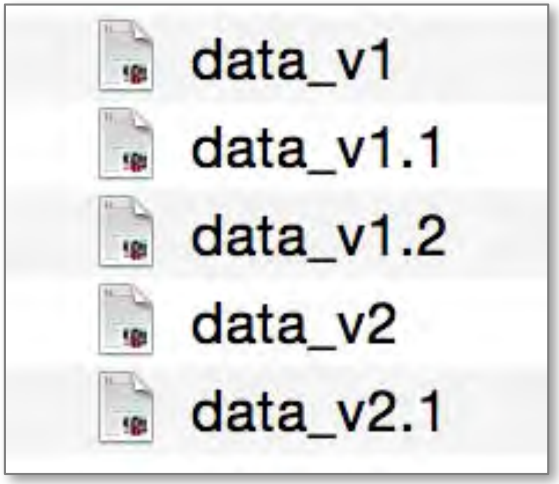
Save new versions



Establish a consistent convention

Versioning: *the how*

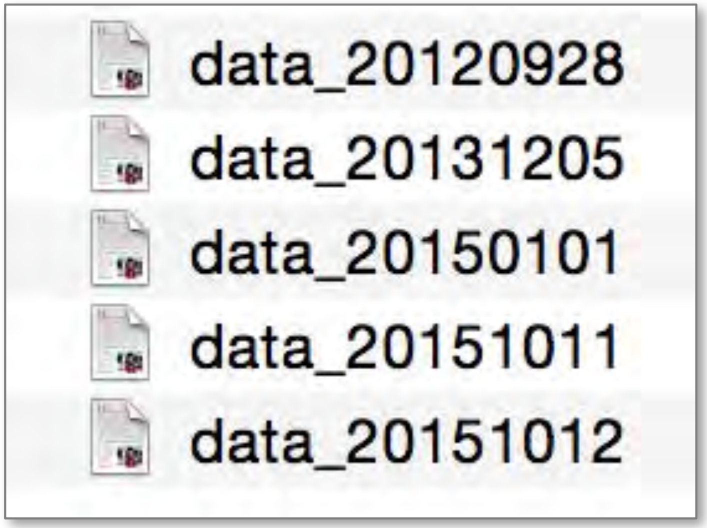
Use ordinal numbers (1,2,3,etc) for major version changes and a decimal for minor changes



data_v1
data_v1.1
data_v1.2
data_v2
data_v2.1

Versioning: *the how*

Use dates to distinguish between successive versions

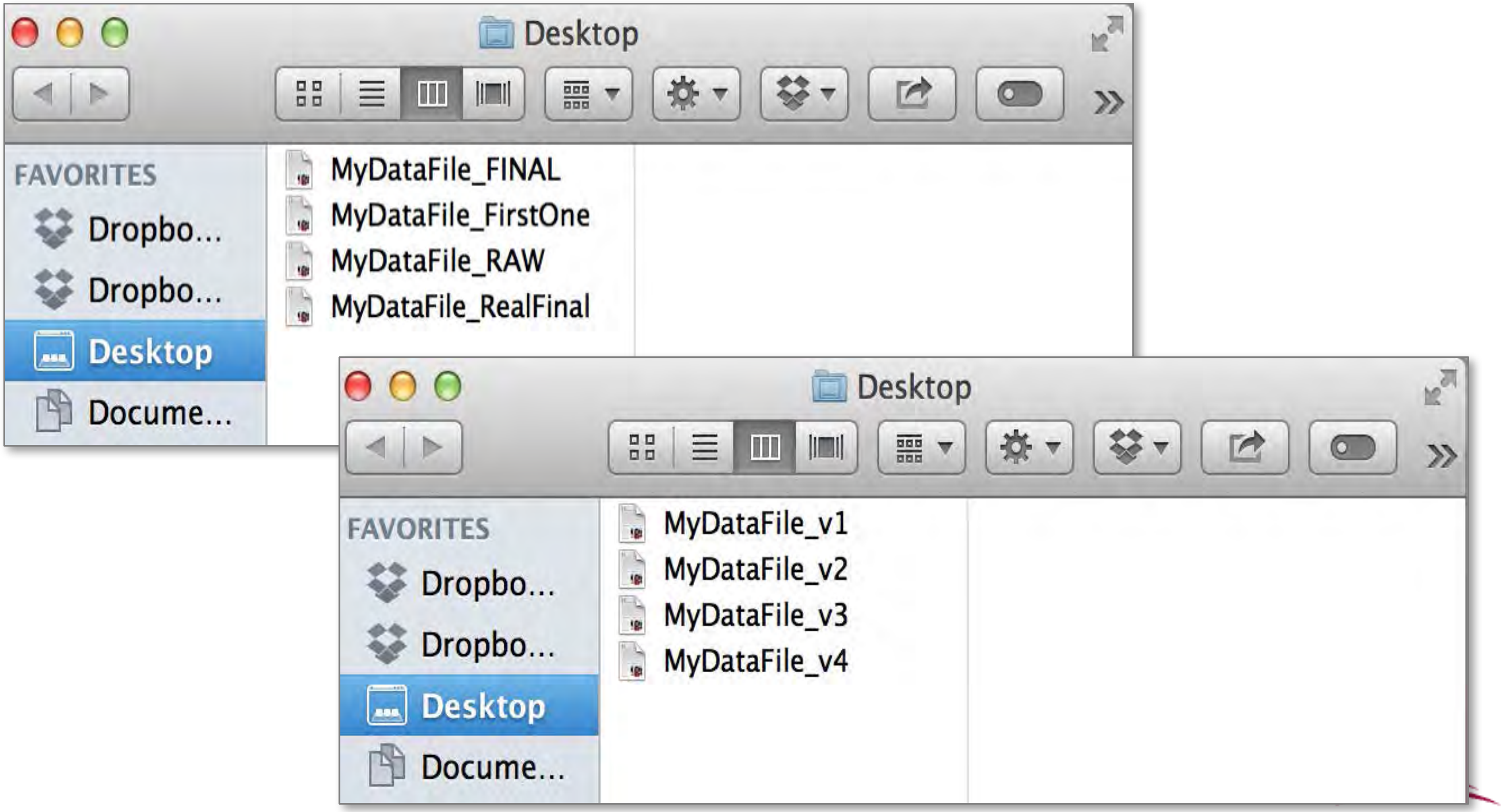


data_20120928
data_20131205
data_20150101
data_20151011
data_20151012

Not ideal when you can potentially have multiple versions in a day.

Versioning: *the how*

Avoid imprecise “final” labels

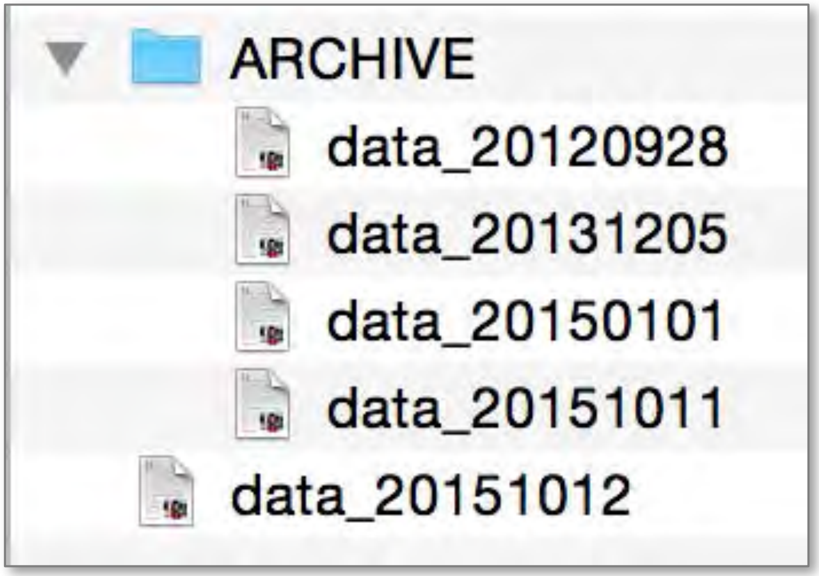


MyDataFile_FINAL
MyDataFile_FirstOne
MyDataFile_RAW
MyDataFile_RealFinal

MyDataFile_v1
MyDataFile_v2
MyDataFile_v3
MyDataFile_v4

Versioning: *the how*

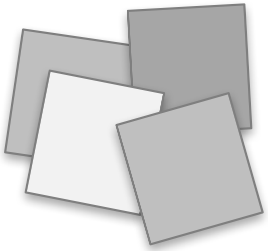
Tip: Put older versions in a separate folder



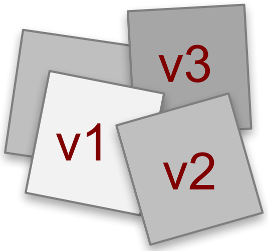
- ARCHIVE
  - data_20120928
  - data_20131205
  - data_20150101
  - data_20151011
- data_20151012

Do you really need to keep obsolete versions?

Versioning: *the how*



Save new versions



Establish a consistent convention

git

CHANGELOG.txt

Document your convention

Versioning: *document it!*

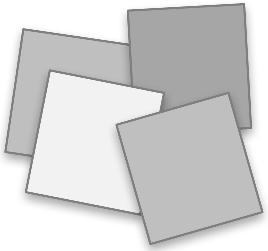
Some options:

- Create a version table or file history w/in or alongside your data files

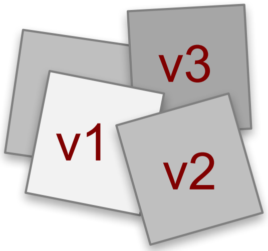
- Use built-in capabilities of software (when available)
  - Wikis, Google docs, etc. that track changes
  - Platforms that allow for checking in/out files
  - Setting permissions

- Use version control software
  - Git, GNU RCS, Mercurial (Hg), etc.

Versioning: *the how*



Save new versions



Establish a consistent convention

CHANGELOG.txt

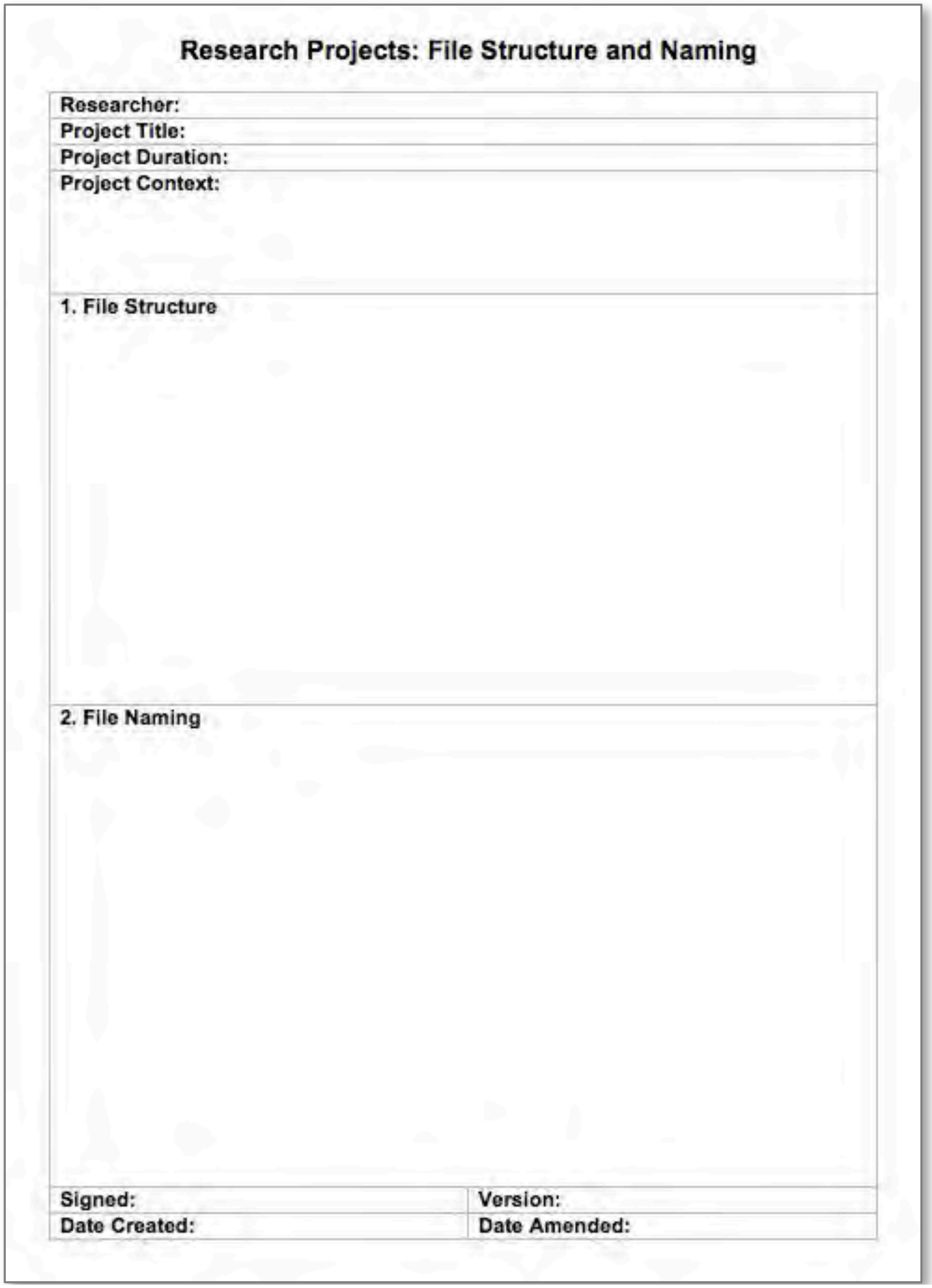
Document your convention

Consider your version control needs

Version control: *general tip*

Be careful when syncing across platforms & simultaneously editing!

Your turn!



Research Projects: File Structure and Naming

Researcher:

Project Title:

Project Duration:

Project Context:

1. File Structure

2. File Naming

Signed:	**Version:**
Date Created:	**Date Amended:**

- Understanding the structure of your own data.
- Allows others to understand your data.
- Establishes good practice early by helping form working habits.
- Print out and stick on the wall above your desk!

Questions? Comments? Tips?

Check out our web site:
http://libraries.mit.edu/data-management

Contact: data-management@mit.edu

Appendix: detailed tips

Tip 1: Embedding metadata

- If feasible, try to enter basic information about the data file within its contents (e.g., author, date created/modified, project, grant, version)
  - May be able to <comment> information in a file
  - May help to identify files using your system's full-text searching capabilities
- Embed metadata in header
- May also be able to assign this information as tags (external to your files); see our guide to Tagging and Finding Your Files: http://libguides.mit.edu/metadataTools/
  - Caveat: some programs strip tags during file transfer or transformation, so don't rely solely upon these

Tip 2: adding searchable keywords to files in Windows

- Open up the Windows folder view and highlight (don’t click to open) your file of interest
- In the pane at the bottom of the folder window, you’ll see metadata about your file
- Click the property that you want to change/add (you’ll see the box for tags all the way on the right), type the new property, and then click Save.
- To add >1 tag, separate each with a semicolon.
- Terms entered here will be found by the Windows search function

Tip 3: Adding tags on a Mac

- When you save a file, from the document menu, or in Finder
- Spotlight Comments (and use Spotlight to search)
- http://support.apple.com/kb/HT5839
- http://www.maclife.com/article/howtos/mavericks_howto_organizing_files_and_folders_tags
- http://computers.tutsplus.com/tutorials/how-to-tag-files-and-create-spotlight-comments-on-a-mac--mac-46431

Tip 4: Shortcuts in Windows

- Shortcuts allow you to open a file from multiple places
- Functions to place a file in >1 category
- Use for frequently accessed items
- Use to create project folders

Tip 5: Shortcuts on a Mac

- On OS X you can create "symbolic links" using the terminal and the 'ln -s' command
- Use Automator (http://support.apple.com/kb/ht2488), alone or in conjunction with AppleScript (http://www.macosxautomation.com/applescript/)

Appendix 2: Batch renaming tools

- Adobe Bridge (via any Creative Cloud products): (Windows or Mac)
- Ant Renamer (Windows)
- Bulk Rename Utility (Windows)
- ImageMagick (Windows, Mac, or Linux)
- GNOME Commander (Linux)
- GPRename (Linux)
- Name Changer (Mac)
- Name Mangler (Mac)
- PSRenamer (Windows, Mac, or Linux)
- RenameIT (Windows)
- Renamer4Mac (Mac)
- WildRename (Windows)

In **Unix**: Use the **grep** command to search for regular expressions