

A System to Easily Manage Metadata in Biomedical Research Labs Based on Open-source Software

Manuel A. Castro-Alamancos

Department of Neuroscience, University of Connecticut School of Medicine, Farmington CT 06001, USA

*For correspondence: mcastro@uchc.edu

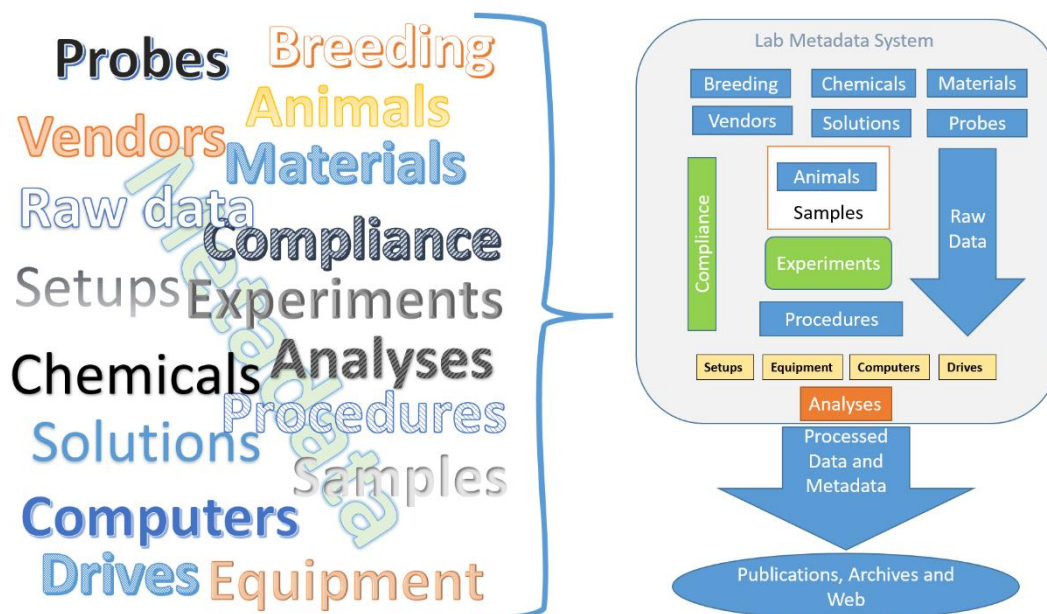
Abstract

In most biomedical labs, researchers gather metadata (*i.e.*, all details about the experimental data) in paper notebooks, spreadsheets, or, sometimes, electronic notebooks. When data analyses occur, the related details usually go into other notebooks or spreadsheets, and more metadata are available. The whole thing rapidly becomes very complex and disjointed, and keeping track of all these things can be daunting. Organizing all the relevant data and related metadata for analysis, publication, sharing, or deposit into archives can be time-consuming, difficult, and prone to errors. By having metadata in a centralized system that contains all details from the start, the process is greatly simplified. While lab management software is available, it can be costly and inflexible. The system described here is based on a popular, freely available, and open-source wiki platform. It provides a simple but powerful way for biomedical research labs to set up a metadata management system linking the whole research process. The system enhances efficiency, transparency, reliability, and rigor, which are key factors to improving reproducibility. The flexibility afforded by the system simplifies implementation of specialized lab requirements and future needs. The protocol presented here describes how to create the system from scratch, how to use it for gathering basic metadata, and provides a fully functional version for perusal by the reader.

Keywords: Metadata, Lab management, Data, Database, Rigor, Reproducibility

This protocol was validated in: J Neurosci (2021), DOI: 10.1523/JNEUROSCI.0027-21.2021

Graphical abstract:



Lab Metadata Management System.

Background

The process of acquiring, analyzing, and sharing research data is complex. As currently implemented by most biomedical research labs, this process is prone to significant errors, leading to problems with scientific rigor and reproducibility. These negative outcomes signify wasted efforts and investments that frustrate both researchers and funding agencies. This problem has prompted NIH to make this a major consideration in peer review. There are now hundreds of academic references directly addressing this problem. A simple PUBMED search (*rigor AND reproducibility*) yields hundreds of hits, most of which are from the last few years (e.g., (Landis *et al.*, 2012; Steward and Balice-Gordon, 2014; Bandrowski and Martone, 2016; France, 2016; Sahoo *et al.*, 2016; Yates, 2016; Baxter and Burwell, 2017; Williams *et al.*, 2017; Borghi and Van Gulick, 2018; Botker *et al.*, 2018; Brown *et al.*, 2018; Dingleline, 2018; Gulinello *et al.*, 2018; Lee and Kitaoka, 2018; Plant *et al.*, 2018; Yosten *et al.*, 2018; Prager *et al.*, 2019; Turner, 2019). Even the popular press has widely disseminated this problem in many high-profile articles (New York Times, <https://www.economist.com/briefing/2013/10/18/trouble-at-the-lab>; The Economist, <https://www.economist.com/briefing/2013/10/18/trouble-at-the-lab>; The Atlantic, <https://www.theatlantic.com/magazine/archive/2015/09/a-scientific-look-at-bad-science/399371/>, etc.). While the causes of this crisis are complex, a main part of the problem is associated with the way basic biomedical research labs are accustomed to managing (gathering, organizing, storing, accessing, and sharing) the details about experiments (metadata). Essentially, there are no established methods for doing this, and labs are basically on their own. While there are several electronic laboratory notebooks in the marketplace, most of these have significant limitations, such as limited scope, focus on a particular subfield, high cost, and inflexibility.

This protocol describes a simple method to set up a freely available lab management system for biomedical research labs, that provides an easy way to store, access, peruse, and organize metadata. Lab metadata are all the information required to understand the data generated by the lab. Metadata includes details about subjects, samples, materials, chemicals, methods (protocols), data files, analyses, *etc.* Without proper metadata, the data derived from any experiment are useless or, at a minimum, can lead to misrepresentations and faulty conclusions. With just a few clicks on a browser (e.g., Chrome), the system presented in this protocol allows the user to know, for example: (i) exactly what was done (detailed protocols), who did it, and when it was done, (ii) what samples and/or subjects

were used, (iii) what materials, chemicals, drugs, and equipment were employed, and (iv) where the data files are stored. This information can go back as many years as the system has been used in the lab. The system described has been operational for several years in the author's lab, and the experience has convinced everyone who has used the system that this approach is the best way to enhance efficiency, transparency, reliability, and rigor, and is therefore likely to improve reproducibility.

The system described has several important features. First, it is based on freely available, and open-source software called DokuWiki, which is widely employed for many purposes. It is valued for its simplicity, while providing a high level of security, including access-control list (ACL) permissions. Second, a wiki is the ideal platform for a research lab since, by definition, it is a repository of knowledge. Third, by running on a popular wiki platform, the system can be enhanced through the availability of hundreds of plugins, that allow users to add features required for their specific needs. In addition, specialized plugins can be developed and easily integrated. Fourth, the system can be easily deployed by lab personnel that have only basic computing knowledge. Having lab members manage their experimental metadata, with minimal effort, for themselves, is practical, and provides both flexibility and control. Fifth, the system can be set up on a network-attached server (NAS), personal computer, or a web hosting service. However, a NAS running in the local lab network is most desirable, and is described in the present protocol. Sixth, all the metadata in the system are stored in standard, and universally accessible formats. For example, part of the system includes a relational database (sqlite3), which can be directly visualized through aggregations on the wiki platform, or accessed using SQL commands through connection from third party software commonly used in research labs (*e.g.*, LabVIEW, MATLAB, OriginLab, Python, R, *etc.*).

There are obviously many reasons why having all lab metadata organized in a centralized system would be useful. First, in most cases, individuals gather their metadata in paper notebooks, spreadsheets, or, sometimes, electronic notebooks. Moreover, when data analyses occur, the related details usually go into other notebooks or spreadsheets, and even more metadata becomes available. The whole thing rapidly becomes very complex and disjointed. By having all lab metadata available for perusal in one location, the process becomes simpler. Second, most labs use an open approach by which each lab member independently organizes the metadata they generate. However, "to be organized" means different things to different people. Moreover, having metadata dispersed, and organized following different reasoning and in different formats, is highly inefficient and prone to errors. A lab system that provides a flexible, coherent, and logical structure eliminates guessing about how to be organized. Third, a typical scenario is that the lab may need to repeat a successful procedure exactly as it was done several years ago, by someone no longer present in the lab, or who may not recall. Having to troubleshoot a complex procedure again can lead to wasted efforts, and significant problems. The system presented here provides a central location to track, update, and use media (images, videos, tables, *etc.*) to detail protocols for all lab procedures eliminating uncertainty. Fourth, there are a number of electronic notebook services available in the marketplace, but these can be costly, and may offer little flexibility or control. The system described here is free, highly adaptable, and runs within the lab. As already noted, a wiki is the ideal repository for a lab metadata management system. Fifth, being able to instantly access metadata on a database greatly facilitates not only manual perusal, but also automated data analyses through scripts, eliminating common human errors. Paper notebooks are no longer required, but can be used as support if deemed useful. Sixth, compliance is a required, and ever increasing institutional burden on research labs. The system described here implements standard record keeping, such as animal usage and breeding — related to Institutional Animal Care and Use Committee (IACUC) protocols, and controlled drug usage — related to Drug Enforcement Agency (DEA) licenses. Seventh, the process of organizing metadata for analyses, publication, and output to data archives can be very time-consuming, sometimes difficult, and is prone to a high degree of errors. By having all metadata, from the start, in a centralized system that contains all experimental details, the process is greatly simplified. For example, the process of uploading specific data with its metadata to data archives can be easily automated. Finally, reproducibility is a recognized major problem in biomedical research labs, and in other science fields. The rigor implemented by deploying the system described here should result in significant improvements in reproducibility.

The system is easy to set up and use. Researchers input metadata from the moment the data are gathered using previously defined, and easily selectable terms that refer to defined variables and procedures (method descriptions), which are described in detail within the system, and constantly updated by lab members. The maxim of the system is that any piece of metadata is input only once, and becomes immediately accessible for any purpose. Moreover, many programming software packages commonly used in labs (*e.g.*, LabVIEW, MATLAB, OriginLab, Python, R,

etc.) can directly communicate with the system, which limits the need for human intervention during data acquisition or analysis. This protocol describes how to set up a fully functional system; a completed operational example is included for perusal. Readers should first peruse the example (step A in Procedure), and learn how to use the system (step B). If the reader decides to implement the system in their lab, they should proceed to the instructions on how to host DokuWiki on their local NAS (step H; preferred), or using a hosting service (step J), followed by the instructions on how to create their lab metadata system (step K). The focus of the protocol presented here is on basic experimental metadata. It does not include how to handle metadata generated during analyses, because this is done best in combination with programming software, and will be described elsewhere. The system provides biomedical researchers with an easy way to be rigorous about managing metadata, so their efforts can focus on the complexities of the science.

Materials and Reagents

1. NAS drives used to set up the NAS (WD Red Pro WD6003FFBX 6TB)
2. External drives used for final data storage (5 TB WD Elements, WDBU6Y0050BBK-WESN)

Equipment

1. NAS server (*e.g.*, Synology DS3617xs). If the server can host DokuWiki, it will work. The size of the server depends on the lab data requirements, and how the lab plans to store the data. If files generated by the lab are small, it is possible to store them, and have them readily accessible on the server. If the data files generated are generally large, it is more practical to store the raw data in duplicate external drives (*e.g.*, same data in two drives), which are inexpensive, and to access those drives when needed. The options are flexible depending on the lab requirements. The important consideration is that the system knows the location of the data.

Software

1. Dokuwiki (www.DokuWiki.org)

Procedure

There are several ways to follow this procedure. If you have never seen an example, it is recommended that you first peruse the example and learn how to use the system, before setting up your own. Here are the recommended steps:

1. Peruse the example provided with this protocol: Go to step A.
2. Learn how to use the system: Go to step B.
3. Host DokuWiki and setup a new system from scratch:
 - a With a NAS: Go to step H, and then jump to step K.
 - b Without a NAS:
 - i Using a hosting service: Go to step I, and then jump to step K.
 - ii On a local computer: Go to steps J and K.

A. Peruse the example wiki system

The example provided with this protocol includes fictitious metadata. It is compressed, and must be unzipped before use. Follow these steps:

1. Download the [WikiExample1](#) to the **Downloads** folder of your computer.

2. Use a program to unzip the folder. For example, in Windows 10, select the downloaded zip file, right-click, and select **Extract ALL**. Make sure to select a folder destination that is different from your current (Downloads) location, or change the name of the unzipped folder. For example, select the destination to be your **Desktop**. This will assure that, in the next step, you don't mistakenly select the compressed folder to run the example.
3. On the Desktop, open the **unzipped** folder, and double-click on the **Run** file. This will launch a command prompt window (...cmd.exe), and your default browser. Most standard browsers should work, but these steps have been performed in Chrome. Your firewall may ask for permission, so give it access. The result should look like Figure 1 below. If it does not, make sure to follow the previous steps again, exactly as noted. To stop the wiki system, click on the command prompt, and press any key.

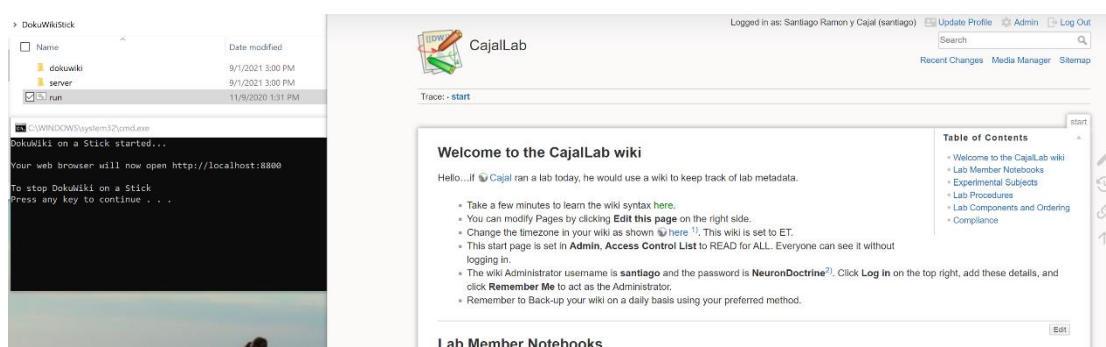


Figure 1. Result of running the example wiki provided.

B. How to use the system

Follow these steps using the example wiki system (step A) provided with this protocol:

1. Clicking on **Run** (located inside the folder) will open the login page or take you directly to the start page (note that the example wiki is titled CajalLab). If this is your first login, add this **username** (Santiago) and **password** (NeuronDoctrine), and select **Remember me**. If this is not your first login, and you had selected Remember me, you will be logged in automatically. Check if you are logged in by looking at the top right area (Logged in as).
2. The start page shows some basic information at the top, and lists all the main wiki pages below. Note the **Admin** button on the top right, and the pencil icon (**Edit this page**) on the right. The Sitemap on the top right will display a directory of the wiki namespaces and pages. The **Media Manager** allows you to see any media uploaded to wiki pages.
3. To modify pages, or add text into pages, it is essential to learn the basic **syntax**. This should take only a few minutes, since it is very simple. Follow the link in the **Welcome** section of the start page. Note the options at the top of the page when you click on **Edit this page**. These buttons automatically set the syntax (e.g., bold, italic, links, etc.).
4. To add an **image** to a page, click on Edit this page, and then drop the image into the wiki page. It will assign a name, load it to the Media Manager, and the link syntax will be added to the page. Place this link where you want it to appear. Use syntax to center it, change size, etc.

C. Examine the start page

Feel free to click on any link, and check out the tables provided in each page. Make sure to click on the links inside the tables. You can always return to the start page by clicking on the CajalLab icon.

1. The **Lab Member Notebooks** section provides each lab member with a namespace, where they can organize their own notebook with direct access to all the metadata in the wiki. In our example, there are 3

lab members. Santiago is the Superuser, and Jake and Jane are members of the labmembers group.

2. The **Experimental Subjects** section provides access to lists of the animals in the lab, including those that are alive, or all the animals that have existed in the lab. There is also a link to create Animals, which is a required step when Animals arrive in the lab (via purchase or breeding). The Strains is a list of the strains used in the lab. Every time a new strain is used, it must be defined in the Strains page. The Breeding cages allow to track breeding. There is only one breeding cage in the example (Cage1). This cage consists of a Mom (Vg7) and Dad (Vg6) that have so far produced 10 new animals (Vg8–17). Keeping track of the breeding simply requires adding an entry in the Cage1 page every time mice are born, and subsequently these animals are added to the wiki normally (see Create New Animal below) at the weaned date. Subgroups are groupings of animals, usually as part of experiments. There are 3 Subgroups specified.
3. The **Lab Procedures** section provides access to the Procedures, Solutions, and Probes used in the lab. Procedures are the methods defined as detailed protocols. The wiki example does not include any detailed protocols inside the Procedures namespace, but this is a main part of a normal system (*i.e.*, the inclusion of detailed protocols describing the procedures). The procedures pages should include images, videos, tables, web links, *etc.* with all the details needed to repeat the procedure. There are two ways to keep track of changes to protocols: i) Each time a page is modified, a revision is made, and this can be accessed and compared using the clock icon on the right side. ii) A new procedure can be created referring to the previous procedure, except for the changes detailed in the new procedure.
4. The **Lab Components and Ordering** section includes all the things that are needed to run a lab. It is only necessary to keep track of items that are essential for procedures (*e.g.*, standard paper towels, pencils, *etc.* do not need to be included). Chemicals and Materials keep track of these essential items purchased for experiments, while Solutions and Probes are items generated within the lab using Chemicals and Materials. Setups consist of Equipment, Computers, and Drives where data are acquired. This allows one to easily determine where data are stored.
5. The **Compliance** section includes IACUC protocols and DEA controlled drugs. Clicking on these links shows the currently active IACUC protocols, including the animals associated with them, as well as the DEA controlled drug bottles in the lab, including current drug usage.

D. Create a new Animal

As practice, you will add a new animal purchased from Jackson Labs (Jax).

1. On the start page, click Create New Animals. Read the section titled **Create a new ANIMAL Page**. In the field, type cf18, and press **Add page**. Note that you are using cf because this is the code for the strain of the animals, and 18 is the next available animal number in the table shown below the field.
2. Once the new page opens, add **Cf18** to the page title (====Cf18=====).
3. Add 18 to the **Number** field.
4. Select **cf** in the **strain** field (you need to add 2 letters to search the options available for selection, then select the correct option). If the text appears red, you have not selected a valid option.
5. Add the **Source** by typing “ja” or “ve”, which will show the options starting with “ja”, or all the vendors. Select Jax.
6. Source and **Mom/Dad** are mutually exclusive. If a source is not provided, the animals were bred in the lab, and will have a specified Mom and Dad.
7. Fill the other options. **Arrival** is the day mice arrived from the vendor, or were weaned.
8. Mice are available for experiments if they don’t have a specified Subgroup.
9. Click Save. Note that all the details you added are included in the top of the page.
10. If you go back to the **create** page (now listed on the top in Trace), you will see the new animal Cf18 in the table.

E. Add Procedures done to Animals

To specify what was done to Animals, the Procedures must be defined in the wiki. The example wiki includes

several procedures. To create new Procedures, follow the same steps used to create new Animals. Make sure to read the specifics in the Procedures page (accessed from the start page). If the animal is part of an existing experiment, the Subgroup should already exist. Otherwise, create the new Subgroup in the Subgroups page. Note that there are rules on how to name the pageids for the new pages, but you can decide on the Title to use within the first page heading. This title can be changed any time, and will be automatically changed throughout the wiki. So, don't worry about the Subgroup name structure if you don't have an established format. Follow these steps to add procedures done to animals:

1. Click on the page you created for Cfl8. Notice the sections below the Animal details.
2. Let's assume the first thing done to this animal is a surgery in which you injected an adeno-associated virus (AAV) into the brain. Go to the **Surgeries** section on the page, and expand **Create New Entry**. Notice that the fields for this serial schema appear.
3. In the procedure field, type `proc`. This will list the procedures in the wiki. Select **injAAV**. If you did this correctly, the namespace:pageid for this procedure should appear.
4. Select the **Date**, add the **weight**. Note the information icon on the left of the field box. Hover over it to see what you should add.
5. **Anesthesia** allows you to search for chemicals and solutions. Type "iso" and select isoflurane. Add the amount. In this case, the % used (1.1).
6. In **Surgeon**, select who is doing the surgery, by searching for their name.
7. **Implanted** allows you to search for Materials and Probes in the wiki. You could select a probe (electrode) to be chronically implanted.
8. **Injected** allows you to search for Chemicals and Solutions in the wiki. You should select the AAV that is being injected, by typing AAV and selecting the correct one. In this case, select AAV1.
9. Add the **Coordinates** used for the injection. This is a basic text field, but could be linked to a page schema with commonly used coordinates.
10. Select the side where the AAV was injected.
11. In **Comments** you can add details about the injection, such as the volume. Alternatively, a new field could be created in the schema, to include this information as a defined variable.
12. Press **Save**. At this point, all this information has been aggregated to all the tables in the wiki where it is relevant. In this case, we did not use DEA controlled drugs, but, if we did, those would be added by clicking on the link (here) located below the **Surgeries** heading.
13. The **Post-surgery recovery** section allows the surgeon to detail the recovery process, as required by IACUC. Each entry (clicking **Save**) defined by a date/time lists drugs injected into the recovering animals, or visual checks of their state.

F. Sessions section

The Sessions section allows users to indicate procedures that led to the generation of data files, while the animals were alive. The following fields are particularly relevant for this purpose:

1. The **Session** field defines a way to specify things done repetitively to animals, such as daily drug injections, behavioral training sessions, *etc.* An ascending letter or number can be used for this purpose.
2. The **Filenum** field defines any files generated within the session. Each ascending number indicates the file order, and establishes the file names.
3. The **Code** field is useful to set codes for procedures indicating particulars of things done during the session. This is particularly useful for automated analysis of the data.
4. The **Setup** field defines where the data were generated, which establishes the drive in which the files are stored. Every time an external Drive is added to a computer, the drive is added to the wiki in the Drives page. The **start** date and the **filled** date keep track of where the files created in Sessions are located. After generating 10's or 100's of drives over the years, it is simple to find where the files are located with one click. Moreover, by accessing the wiki, programming scripts can determine the file locations, and request that a specified Drive be connected to a USB port during automated analyses of data.

G. Histology section

The Histology section defines the tissues that were obtained for histological processing. It is typical to extract tissues from animals for processing, particularly at death. This section allows specification of the treatment of these tissues.

1. Once the animal is dead, the death date is added in any table where this field is visible, by double clicking on the table-cell, or in the Animal page, by modifying the page.
2. Select the tissue that was obtained from the animal, and the Procedure used to treat the tissue.
3. Add an entry for each new procedure performed on the tissue, with as many details as required. Additional fields can be defined. For example, if subsamples are obtained, an identifier field can be added for each tissue subsample.

H. Setup a NAS to host DokuWiki

The preferred option is that you host Dokuwiki on a NAS running in your lab network. The procedure will vary depending on the server available. The only requirement is that it can host DokuWiki. For practical purposes, this protocol assumes that you are using a Synology server running the latest version of Diskstation Manager (DSM 6.x), which automatically hosts Dokuwiki as a third-party application; nevertheless, this is not required. Note that setting up, and routinely using a Synology server running DSM is rather simple. It will also provide the lab with many other capabilities: for example, chat for lab communication (like Slack or Teams), a cloud for file access anywhere (like OneDrive or Google drive), and many other useful utilities are provided. If you don't have access to a NAS, skip to I. Several things must be setup on the NAS:

1. Add users. The simplest approach is to create one username for all members of the lab, if they should all have equal access rights. We will call this general lab member, "LabUser". On the DSM, go to Control Panel, and select Users. Add LabUser as the common username for all lab members, and include a strong password. Note that having a common username is simpler, but less secure, and all lab members will have equal access to the server. This username/password should also exist, as a User, on the lab computers that will access the system.
2. Add a Labmember group. On DSM Control Panel, select Groups, and add a group called "Labmember". Include in this group LabUser, or the individual lab member user accounts you defined above. This group will allow you to define common permissions to shared folders defined below.
3. Create one or more data Shared folders for primary processed data. The folders you create will vary depending on your lab needs. These folders can hold files that contain initially processed data, which you would like to have readily accessible for analyses, or other purposes. In Control Panel, go to SharedFolders, and make DATA a shared folder.
4. Include ACL permissions for each of these folders. The simplest option is to give the LabMember group full access. Alternatively, you may give the LabMember group full permissions, except Delete.
5. Install other utilities included with your NAS. For example, install Drive, which will allow you to run a cloud, and Chat, which will allow seamless lab communication.
6. Logon to DSM, click on Package Center, and scroll down to the Third Party section. Find DokuWiki, and click Install. This will guide you through the process of setting up your Dokuwiki.
7. If you selected this hosting option, jump to step K.

I. Host DokuWiki on a hosting website

Contract a hosting website that includes Dokuwiki in its offerings, or have your institutional IT host DokuWiki for you on their servers. This is acceptable, but may be impractical. If you selected this option, jump to step K.

J. Host Dokuwiki on a local computer

This option provides a way to set up the system for evaluation, but is only a short-term solution.

1. Before you start this step, make sure to **turn off** the example wiki, by selecting the command prompt window, and pressing any key. The example and the Apache server should not be running.
2. The steps provided in this protocol are optimized for PCs running Windows. Mac users should additionally follow the steps described in: <https://www.dokuwiki.org/install:macosx> and https://www.dokuwiki.org/php_build-in_webserver.
3. Go to <https://download.dokuwiki.org/>, select the stable version, and the option for “Include Web-server”.
4. Unzip the downloaded file to a particular folder on your computer or USB drive (Documents or Desktop). Open the DokuWikiStick folder, and click Run. Make sure you are opening the unzipped folder (not the downloaded zipped folder). The first time you run this, you should allow firewall access, if prompted.
5. The DokuWiki installer will appear in your browser (use Chrome preferably).

K. Create your lab metadata system

The previous steps (H, I, or J) should lead to the DokuWiki installer page open in your browser. Once you complete this and the following steps, the wiki system will look identical to the example provided (see step A), but without metadata.

1. Give your wiki a name in the Wiki Name field (*e.g.*, MyLab, where My is the PI’s surname).
2. Make sure that “enable ACL” is selected.
3. Add yourself as the SuperUser by filling in your details, including a password.
4. For security, select the ACL policy to be a Closed wiki. Then press Save.
5. The next page will show a Login: add your username and password to enter your wiki.

L. Install required plugins on DokuWiki

1. On the Dokuwiki Start page, select Admin and Extension Manager. You will find that many plugins are installed by default. Install the following plugins if they are not already installed: Add New Page, Bureaucracy, Imppaste, Smt, Sqlite, Struct, and Templatepagename. Installation is simple, search for the plugin in the Search and Install tab, and once the plugin appears (make sure the plugin name matches exactly), click Install on the right side.
2. Optional: If you are interested in sending emails directly from the wiki, make sure to set up the SMTP configuration. Go to the Admin, click on Configuration Settings, and select SMTP. Setup the server you want to use to relay your emails. If using your institution email, it may be required to contact IT to enable email relay from your server IP. Otherwise use some settings that you know work, such as commonly used Microsoft or Goggle SMTPs.

M. Decide about your lab filenames and subject unique identifier codes

In our example, we will setup a typical biomedical research lab that works with subjects (mice). However, subjects can be anything you work with, but it should be the highest-level identifiable entity from which other samples (*e.g.*, organs, cells, *etc.*) are generated. Lower-level samples that originate from identified subjects will share the subject code. It is important to select a logical and simple rule to serve as a unique identifier within your lab. This identifier will be used to label all data related to a subject. The scheme in the example shown below works well, but you are free to select your own code definitions.

1. In our example, the subjects (mice) are named using a **unique identifier** code structure, consisting of 2 letters that identify the strain, and a consecutive number that increases by an integer for every new subject in the lab. For instance, vg1234 is the 1234th mouse used in the lab, and it is from a Vgat-Cre strain, since vg was defined in the system as Vgat-Cre strain (as described below). If you plan on using more than 325 strains, you should employ 3 letters instead of 2. You are free to choose your own scheme, but it must have an ascending unique number. For example, you can simply use the ascending integer without the strain code.

2. **Filenames** are named starting always with the subject's unique identifier, followed by a few parameters that identify the file. This may include a session code, a file number within the session, and a procedure code. For example, vgl234a2.dat refers to file number 2, in session a. The parameter code logic in your files is up to you, but must contain the unique identifier.

N. Define your lab's basic schemas

Schemas are tables inside a database that will contain the metadata which is amenable to a database. Additional metadata can go within the wiki pages. Several basic (essential) schemas are required for our example, but many other schemas can be created, depending on your needs. Our example contains the following basic schemas:

1. **Animals** are the subjects of the lab experiments (mice, in our example).
2. **Strains** are the strains of animals used in the lab.
3. **Subgroups** are used to group Animals that belong to the same experiment.
4. **Procedures** define the methods (protocols) performed in the lab.
5. **Surgeries** define the details of surgical methods performed on animals.
6. **Surgrecovery** serves to track post-surgery recoveries.
7. **Sessions** define the details of file recording methods performed on animals.
8. **Histology** defines the details of histological samples obtained from animals.
9. **Chemicals** are the chemicals used in the lab.
10. **Solutions** are combinations of chemicals.
11. **Solutionmake** are the steps to make solutions.
12. **Materials** are all the materials used in the lab.
13. **Probes** are devices created with materials that are used to measure biological variables.
14. **Electrodes** are a type of probe.
15. **Computers** are the computers used in the lab for data acquisition.
16. **Drives** are the external hard drives connected to computers, and used for data storage.
17. **Equipment** are all the equipment used in the lab.
18. **Setups** are combinations of equipment, computers, and drives where data are generated.
19. **Vendors** are the vendors from where materials, chemicals, equipment, *etc.* are purchased.
20. **Orders** track all orders from vendors.
21. **IACUC** are the lab IACUC protocols.
22. **DEA** are bottles of DEA-controlled drugs.
23. **DEAuse** is the use of DEA drugs.
24. **Breeders** are pairs (or other combinations) of animals used for breeding.
25. **Born** are the animals born from the breeders.

O. Define additional schema examples

You can make as many schemas as are required to account for your needs. Anything that is relevant to your experiments, and that can be listed and defined should be a schema, or part of a schema. Here are some cases (not included in the example):

1. **Groups** is useful to combine subgroups into coherent studies or papers.
2. **Cages** is useful to define where the animals are housed. This may be required by your institution to maintain a census. It is easy to set up a way to notify your animal facility via email, directly from the system, once a cage is empty (using the SMTP plugin).
3. **Programs** is useful to list specialized programs employed for data acquisition in your lab. For example, if you do behavioral experiments, you may use software definitions to run those experiments. A schema would store details about these programs, including code definitions that are useful to run specific analyses. By including these in the wiki database, they can be accessed directly by scripts for execution and analyses.
4. **Scripts** is useful to list and track your programming language scripts (Matlab, Originlab, Python, R, *etc.*).

P. Add each schema to the wiki database

The details that need to be added for each basic schema are listed in [Table S1](#). These are the steps to add each schema to the wiki:

1. In the Admin page, click on Struct Schema Editor in the Additional Plugins section.
2. Add the schema name in the schema field, and click Save. The schema name, fields, type, and the changes to be made from default are listed in Table S1. In general, you should use contiguous lowercase letters and integers only to name your schemas. Fields can have more complex text but, when making your own, keep it simple.
3. Begin filling each schema field (*i.e.*, table column) one by one, using the values in Table S1. You will need to click Save to add each field. Note that sort value (10's) sets the order of the columns in your aggregations, but not in the database (this allows insertion of other fields later, if desired, without changing all sort values). The SQLite database incorporates the field columns in the order you add them.
4. Once you are finished adding all the fields and schemas, you can try exporting one of them to see its format. Click on the Import/Export tab, select Page, and click Export. If you have Excel installed, or some other CSV program, open the exported CSV file to see the table. Note that the table is an empty sheet with your field headings per column. Once there are values, they will appear in the CSV (use the completed example to see values). Pid refers to the Page ID for Page schemas (fixed name of a page in the wiki). The Page ID is different from the Page Title. The Page Title is the first heading in a page, and can be easily changed (if a page has no headings, the Page ID becomes the Page Title). Rid refers to the row ID for Global and Serial schemas. Each row of a Page schema corresponds to a page in the wiki. Serial schemas have multiple rows that are all associated with specific pages in the wiki (multiple different rows can be associated with each page). Global schemas have no association with any page on the wiki.

Q. Associate page schemas with namespaces

If the schema is a Page schema, this step is required to associate the schema with a namespace in the wiki, whereby a namespace is a folder within the wiki that has the format **namespace:pageid**. Pages allow extensive content (text, images, *etc.*) to be associated with the values defined in the Page schema (*e.g.*, include details about a protocol with extensive images, video, tables, *etc.* in the Procedures pages). To associate the Page schemas with their namespaces, follow these steps:

1. In the Admin page, click on Struct Schema Assignments, and navigate to the bottom of the page to add the contents listed in each row of [Table S2](#), one at a time.
2. Select and copy the full content of the first cell under Page/Namespace in Table S2. Paste the content into the open field at the bottom of the wiki page (under Page/Namespace column), and change the Schema selection column to the value in the Schema column in Table S2. Check that the values align with the row in Table S2, and press **Add**.
3. Repeat the previous step for each row in Table S2.
4. At the end, there should be 16 rows, indicating 16 associations between schemas and namespaces, as shown in **Figure 2** below. The Regular Expressions used in these lines exclude certain pages inside the namespace from being included in the schema.

Page/namespace	Schema	
/animals:(?!c_template)(?!animals)(?!create)/	animals	Delete
/breeders:(?!c_template)(?!breeders)/	breeders	Delete
/chemicals:(?!c_template)(?!chemicals)(?!inventory)/	chemicals	Delete
/computers:(?!c_template)(?!computers)/	computers	Delete
/dea:(?!c_template)(?!dea)/	dea	Delete
/drives:(?!c_template)(?!drives)/	drives	Delete
/equipment:(?!c_template)(?!equipment)/	equipment	Delete
/iacuc:(?!c_template)(?!iacuc)/	iacuc	Delete
/materials:(?!c_template)(?!materials)/	materials	Delete
/probes:(?!c_template)(?!probes)/	probes	Delete
/procedures:(?!c_template)(?!procedures)/	procedures	Delete
/setups:(?!c_template)(?!setups)/	setups	Delete
/solutions:(?!c_template)(?!solutions)/	solutions	Delete
/strains:(?!c_template)(?!strains)/	strains	Delete
/subgroups:(?!c_template)(?!subgroups)/	subgroups	Delete
/vendors:(?!c_template)(?!vendors)/	vendors	Delete
<input type="text"/>	animals <input type="button" value="Add"/>	

Figure 2. Aspect of the Struct Schema Assignments page, after making the associations.

R. Create namespaces and pages for page schemas

After the schema assignments have been made, the namespaces and a few pages need to be created inside each namespace. This involves creating the pages, and adding the required syntax into them. In this step, we will create the namespaces, the main pages that are used within each namespace to create other pages, and the template pages that are called upon each time a new page is created by a Page schema. Follow these steps for each namespace listed in [Table S3](#):

1. In the Search field located in the top right, search for the namespace:pageid listed in Table S3 (e.g., **animals:animals**), and press Enter. This will show a message indicating that the page does not exist.
2. Create it by clicking on the red text. When the page appears, click on **Edit this page** (pencil symbol on the right hand side).
3. Paste the syntax text for this page located in Table 3, and press Save. At this point, you have created your first page in the wiki. Since there is no data in the schema, it will report “**Nothing found**”. Note that Table S3 is provided also as a .txt file, which makes copying the syntax simpler.
4. For each namespace, Table S3 lists both the main pages and the **c_template** page. The c_template page provides the syntax to be added to pages created by the assigned schema.
5. Repeat these steps for all rows listed in Table S3. Note that the last row in Table S3 includes the **start** page.

S. Add lab members to the wiki

So far, the Superuser is the only one with access to the wiki. Lab members should be added to a Labmember group following these steps:

1. Go to the **Admin** page, and select **User Manager**.
2. In the section **Add user**, fill the fields for each user with their password, and include the user in a new Labmember group.
3. Click **Add** for each addition. If successful, each lab member will appear in the list above.

T. Set access control management

Set pages that are accessible to the Labmembers group. By default, the Superuser has access to the whole wiki, and has the ability to delete. There are 6 levels of permissions for each namespace or page: None, Read, Edit,

Create, Upload, and Delete.

1. Go to the Admin page, and select **Access Control List Management**. Select **Group** in **Permissions for**, and type **labmembers**. This will display the permissions for this group regarding the namespace or page selected on the left window.
2. On the left window, select each namespace starting with **Animals**. Change **Add new Entry** to **Upload**, and click Save. Repeat this for all the existing namespaces, except Wiki. This will allow lab members to upload images and other content to these namespaces. If there is a reason to exclude this access level for any namespace, set it only with permission to Edit or Read.
3. However, existing pages (created above), such as the **c_template** and main pages (e.g., **animals**, **create**, **breeders**, **chemicals**, *etc.*) inside each namespace, should probably not be modifiable by lab members, since there is no reason for those not involved in managing the wiki to change any of these existing pages (before new pages are added). To avoid this problem, click on each of the existing pages inside each namespace, and set the currently existing pages to **Read** for the labmembers group. This step is not necessary, but will eliminate the possibility that someone inadvertently changes the syntax on one of these essential pages. This step was not done in the example provided.
4. Once the above steps have been performed, the **Current ACL Rules** for each namespace will be visible in the table at the bottom of the page.

U. Gather your old lab metadata and add it to the schemas, or start as of today

At this point, the wiki is ready for use. There are two ways to start using it. One is to begin using it by manually adding metadata. Alternatively, you can upload your existing old metadata to the schemas. If you select the latter option, it is best to look at the format of the schema tables by exporting the schemas from the example provided with this protocol. This will show the format necessary to upload your metadata in the correct format. For example, the **date** field must follow the wiki format, which is set by default to **yyyy-mm-dd** (as per Excel date nomenclature). So be sure to adjust your dates in the CSV prior to import, or it will not work (yyyy-mm-dd hh:mm for Date/Time fields). When exporting or importing into schemas, it is important to specify if your schemas are Page or Serial schemas. The following are **Serial schemas**: born, deause, electrodes, orders, sessions, solutionmake, surgeries, and surgerecovery. All the others are Page schemas. To export the schemas from the example provided, follow these steps:

1. Go to Admin page, and select **Struct Schema Editor** at the bottom.
2. Select a schema by clicking on it on the right-side list. For example, click on Animals.
3. Select the Import/Export tab.
4. Navigate down to the section **Export raw data to CSV file**. Since Animals is a Page schema, select **page** and press **Export**. If Excel is installed, and Chrome includes an extension to “enable local file links”, the schema table will be opened in Excel. Note the format for each field. Pages are referred by their **namespace:pageid**. Dates will be displayed in Excel default format, but should be uploaded as indicated above.
5. If you decide to upload your existing metadata, it is best to begin uploading the schemas that do not depend on other schemas, and then incorporate the other schemas. For example, Vendors, Strains, Subgroups, IACUC, *etc.* should be imported before Animals, since Animals includes the other schemas.

Data analysis

Not applicable. The lab metadata system presented in this protocol serves to manage all lab metadata. It allows direct manual perusal of the lab’s metadata. Importantly, the metadata can also be directly accessed from various programming languages for any purpose, including automated data analyses.

Notes

A mature wiki system that has been used for several years will have thousands of animals, hundreds of subgroups, tens of procedures, *etc.* All this information can be accessed in an instant, by browsing the system. However, a major consideration is that having this information organized within the system allows it to be directly accessed for automated analyses using programming languages. The approach will vary depending on the programming language used in each lab, but will involve those packages and functions that allow access to sqlite3 databases. In addition, the metadata included within wiki pages can be directly accessed by these systems. This may be useful for exportation of full procedures (protocols) to archives. Obviously, all this can be done manually, by simply browsing the system. For instance, the metadata gathered over the years located in the sqlite3 database is exportable as simple CSV files. The details located in pages are exportable in multiple formats, including PDF, or simple text. Users can explore the many available plugins available to enhance their wikis with useful and desirable features.

Acknowledgments

I thank members of my lab, especially S. Hormigo and J. Zhou, for using the wiki system and providing feedback. I thank C. Bonin and G. Hargis for testing this protocol. Supported by grants from NIH. More info at <https://castro-lab.org>.

Competing interests

The author has no relationship with any of the companies mentioned.

References

- Bandrowski, A. E. and Martone, M. E. (2016). [RRIDs: A Simple Step toward Improving Reproducibility through Rigor and Transparency of Experimental Methods](#). *Neuron* 90(3): 434-436.
- Baxter, M. G. and Burwell, R. D. (2017). [Promoting transparency and reproducibility in Behavioral Neuroscience: Publishing replications, registered reports, and null results](#). *Behav Neurosci* 131(4): 275-276.
- Borghi, J. A. and Van Gulick, A. E. (2018). [Data management and sharing in neuroimaging: Practices and perceptions of MRI researchers](#). *PLoS One* 13(7): e0200562.
- Botker, H. E., Hausenloy, D., Andreadou, I., Antonucci, S., Boengler, K., Davidson, S. M., Deshwal, S., Devaux, Y., Di Lisa, F., Di Sante, M., *et al.* (2018). [Practical guidelines for rigor and reproducibility in preclinical and clinical studies on cardioprotection](#). *Basic Res Cardiol* 113(5): 39.
- Brown, A. W., Kaiser, K. A. and Allison, D. B. (2018). [Issues with data and analyses: Errors, underlying themes, and potential solutions](#). *Proc Natl Acad Sci U S A* 115(11): 2563-2570.
- Dingledine, R. (2018). [Why Is It so Hard to Do Good Science?](#) *eNeuro* 5(5).
- France, C. R. (2016). [Promoting experimental rigor in the conduct of conditioned pain modulation studies: the importance of reliability](#). *Pain* 157(11): 2397-2398.
- Gulinello, M., Mitchell, H. A., Chang, Q., Timothy O'Brien, W., Zhou, Z., Abel, T., Wang, L., Corbin, J. G., Veeraragavan, S., Samaco, R. C., *et al.* (2019). [Rigor and reproducibility in rodent behavioral research](#). *Neurobiol Learn Mem* 165: 106780.
- Landis, S. C., Amara, S. G., Asadullah, K., Austin, C. P., Blumenstein, R., Bradley, E. W., Crystal, R. G., Darnell, R. B., Ferrante, R. J., Fillit, H., *et al.* (2012). [A call for transparent reporting to optimize the predictive value of preclinical research](#). *Nature* 490(7419): 187-191.

- Lee, J. Y. and Kitaoka, M. (2018). [A beginner's guide to rigor and reproducibility in fluorescence imaging experiments](#). *Mol Biol Cell* 29(13): 1519-1525.
- Plant, A. L., Becker, C. A., Hanisch, R. J., Boisvert, R. F., Possolo, A. M. and Elliott, J. T. (2018). [How measurement science can improve confidence in research results](#). *PLoS Biol* 16(4): e2004299.
- Prager, E. M., Chambers, K. E., Plotkin, J. L., McArthur, D. L., Bandrowski, A. E., Bansal, N., Martone, M. E., Bergstrom, H. C., Bernalov, A. and Graf, C. (2019). [Improving transparency and scientific rigor in academic publishing](#). *J Neurosci Res* 97(4): 377-390.
- Steward, O. and Balice-Gordon, R. (2014). [Rigor or mortis: best practices for preclinical research in neuroscience](#). *Neuron* 84(3): 572-581.
- Sahoo, S. S., Valdez, J. and Rueschman, M. (2016). [Scientific Reproducibility in Biomedical Research: Provenance Metadata Ontology for Semantic Annotation of Study Description](#). *AMIA Annu Symp Proc* 2016: 1070-1079.
- Turner, J. R. (2019). [Rigor, Reproducibility, and Responsibility: A Quantum of Solace](#). *Cell Mol Gastroenterol Hepatol* 7(4): 869-871.
- Williams, M., Bagwell, J. and Nahm Zozus, M. (2017). [Data management plans: the missing perspective](#). *J Biomed Inform* 71: 130-142.
- Yates, B. J. (2016). [Strategies to Increase Rigor and Reproducibility of Data in Manuscripts: Reply to Heroux](#). *J Neurophysiol* 116(3): 1538.
- Yosten, G. L. C., Adams, J. C., Bennett, C. N., Bunnett, N. W., Scheman, R., Sigmund, C. D., Yates, B. J., Zucker, I. H. and Samson, W. K. (2018). [Revised guidelines to enhance the rigor and reproducibility of research published in American Physiological Society journals](#). *Am J Physiol Regul Integr Comp Physiol* 315(6): R1251-R1253.