

Creating DOIs for DigitalScholarship@UNLV

Scope/Purpose: Provides instructions for Scholarly Communications staff on how to create DOIs for current collections in DigitalScholarship@UNLV.

Note: This documentation should only be used to supplement in-person training on how to create DOIs using Fabrica. Please schedule a training meeting with the metadata librarian before attempting to create DOIs.

Before getting started (DataCite)

Documentation for DataCite can be found [here](#). Access to DataCite DOI Fabrica can be found [here](#). Scholarly Communication Initiatives' DOI Projects folder can be found [here](#).

URL TO ACCESS DATACITE: <https://doi.datacite.org/>

For our purposes, you will need to log into our DataCite Repository account. This repository account was created by the metadata librarian for all DigitalScholarship@UNLV material, and is managed by the main "UNLV" DataCite account. The main UNLV DataCite account is currently run by the metadata librarian, please contact the metadata librarian if you have an issue accessing the Digital Scholarship DataCite Repository, or if there are other DataCite issues.

What are DOIs?

DOIs are Digital Object Identifiers that provide persistent links for research data and other research outputs (such as articles, etc.). DOIs are not case sensitive (e.g. "10.1234/WXYZ" and "10.1234/wxyz" are the same DOI). More information about DOIs can be found [here](#).

DataCite has two different platforms that allow you to create DOIs. There is a **DataCite DOI Fabrica Test** site (doi.test.datacite.org) and **DataCite DOI Fabrica** (doi.datacite.org). For our purposes you will only have an account for DataCite DOI Fabrica (doi.datacite.org).

In DataCite, there are three stages a DOI can be in:

- **Draft:** This is the **only** state where a DOI is able to be deleted, it is not a registered DOI at this stage. It only exists in DataCite DOI Fabrica.
- **Registered:** This DOI is now registered to your prefix. The DOI cannot be used for another title or deleted. It is not published in the DataCite index at this time.

- **Findable** (aka “Published”): This DOI is now searchable across the web. It cannot be deleted, but it can be edited.

Creating DOIs (one at a time)

Please use this if you have not been trained on or do not feel 100% comfortable using the batch process

Creating DOIs individually is a fairly straight-forward task. You can either choose to create a DOI [via File upload](#) or [via Form](#).

The Form option is the most intuitive to use (vs. uploading a file of xml metadata per each DOI you want to create) and is the preferred method to create DOIs.

[DataCite's documentation](#) will contain the most up-to-date instructions on how to create individual DOIs. If you have any questions, please contact the [metadata librarian](#).

Creating DOIs (in batch)

Before getting started (Jupyter Notebook)

In order to create DOIs for DigitalScholarship@UNLV, you will need to make sure you have Jupyter Notebook installed on your computer. Jupyter Notebook requires python to be installed in order to run.

If you already have python installed on your computer, verify the version (Python 3.3 or greater, or Python 2.7 is required) and install Jupyter Notebook via [their instructions](#). **If you do not have python on your computer, follow the [instructions](#) to install Jupyter by installing Anaconda. Instructions for how to install Anaconda on Windows can be found [here](#), and for MAC installation instructions, click [here](#).**

If you have any trouble installing python or Jupyter Notebook on your computer, please contact the metadata librarian.

Our process

DataCite uses a REST API that allows users to upload metadata for DOIs via command line. Normally this metadata must be uploaded one file at a time--essentially mimicking the single file process above--or requires use of homegrown Ruby on Rails script in order to batch upload records.

Scotty Carlson created a python script that would allow users to put all metadata in a csv file and run the python code in Jupyter Notebook in order to batch upload metadata for multiple records simultaneously. That code has been edited to suit the metadata found in DigitalScholarship@UNLV, is available in our Google Drive [here](#), and should be downloaded to your computer.

Some things to be aware of: The script **publishes** the DOIs immediately, they are immediately Findable and cannot be deleted. **You must test a subsection of your records (start with 5 MAX) before you run the script on the entire set.** Follow the directions below with your first set of records.

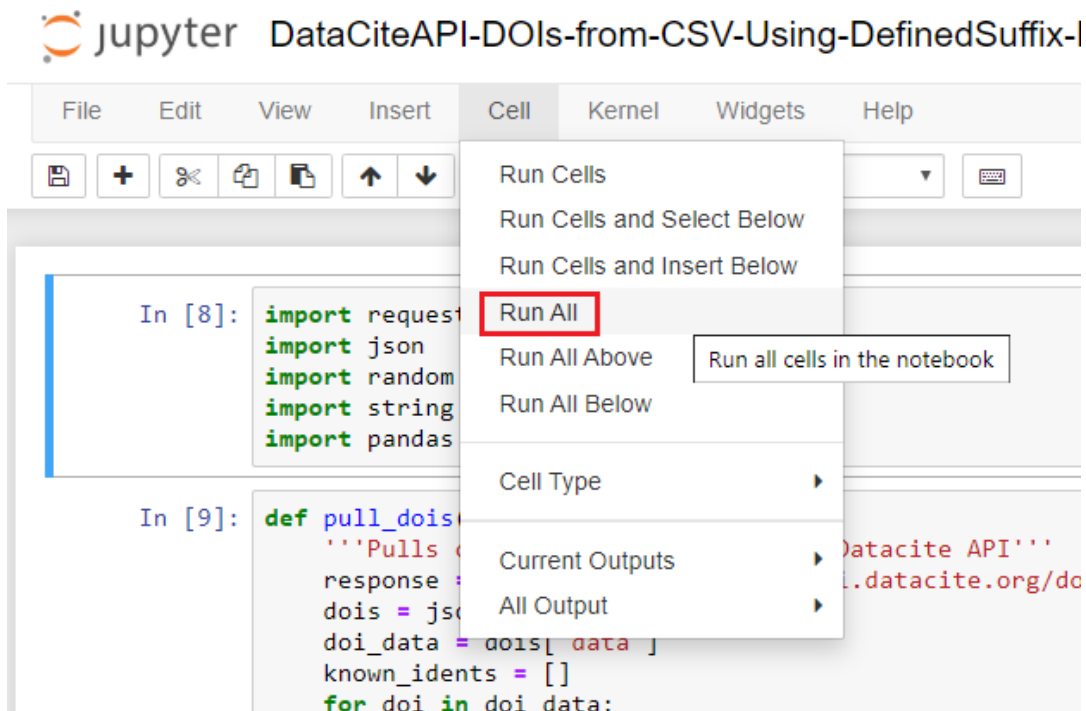
Note: If you have metadata containing **multiple creators or a corporate creator**, please **do not** proceed with batch uploading as the python script would need to be adjusted to accommodate them. At this point, DataCite's API will not allow repeat fields (cannot accommodate batch creating multiple authors).

- For **single, corporate creators** SKIP that row of the metadata and create the DOI via the Form option discussed in the previous section.
- For **multiple authors**, FLAG that row of metadata. You can create the DOI in batch using one of the many author names, but will need to go in manually edit the form in DataCite to add the additional authors. If you are already doing a batch spreadsheet upload with many DOIs, this is likely a faster method than setting aside all of the titles that have multiple authors to create DOIs one-by-one via the form. If you are not sure you feel comfortable with this method, play it safe and just use the form.

Step-by-step guide to batch uploading DOIs

1. Download your metadata from DigitalScholarship@UNLV via a *Batch revise Excel* sheet.
2. Open an additional, new Excel spreadsheet.
3. In order to upload the metadata to DataCite, you must have the following columns in your new spreadsheet with the fields populated by data from the batch revise sheet:
 - a. **“context_key”** : This column contains the context_key column from the *Batch revise Excel* spreadsheet. It is a unique number assigned by BePress to our digital objects. This number will be pulled in as a suffix for our DOIs.
 - b. **“creator”**: This column contains value *“Author Surname, Author Rest of Name”*. If you have any questions on how to compile this format from BePress provided metadata, please contact the metadata librarian.
 - c. **“author1_fname”** : This column contains the author’s first name or all of the author’s name which is not a surname (including middle name, initials, etc.).
 - i. Be aware that you will need to create a "new" *author1_fname* column that equals the old *author1_fname* + any value in *author1_mname*. **For instructions for how to combine columns in Excel, see [this tutorial](#).**
 - d. **“author1_lname”** : This column contains the author’s surname.
 - e. **“date”** : This column should contain only the year that the work was published. If you have any questions on how to convert the date provided by BePress to year only, please contact the metadata librarian.
 - f. **“source”** : This column contains the calc_URL that is provided by BePress. It will link to the landing page in the IR, not to the pdf of the digital object itself.
 - g. **“title”** : Contains the title of the work.
 - h. **“type”** : Contains the document_type of the work. E.g. “thesis”
 - i. **“description”** : Contains the description of the work. Note: If the description of your work is not an abstract, that will need to be edited in the code where it says ‘descriptionType’: ‘Abstract’ in Jupyter Notebooks. OR edited via the Form in DataCite once the DOI has been minted.
 - i. Make sure to do a Find & Replace to remove any HTML paragraph codes that may be in the abstract <p> and </p>
 - j. **“publisher”** : The publisher field only exists in the Book Gallery and Series publication types. For ETDs, you will use the *author1_institution* field. For other publication types, please consult the metadata librarian.
4. Once you have compiled your Excel spreadsheet, *Save As* a .csv file. Be sure to name your file something clear and related to the project (e.g. rtd-doi-metadata-test1.csv) and make a note of the file path to get to your file (e.g. C:\Users\KGeorge\Desktop\)
5. We will now switch over to Jupyter Notebook. Note: if you installed Jupyter Notebook through Anaconda, you will need to launch Anaconda in order to open Jupyter Notebook.

6. Note: You will need to download the Jupyter Notebook file “DataCiteAPI-DOIs-from-CSV-Using-DefinedSuffix-DigitalScholarship” from [this folder](#). Open the Jupyter Notebook file.
7. Now, go to the **Cell** menu and select **Run All**. As cells are running, there will be an * in the brackets “ [*] ”.



8. You will be prompted to enter the file path with the file name in the second to last box. It will be something like C:\User\Kelsey\Desktop\FolderName\FileName.csv
9. Be patient as they are being created, for large sets it will take at least several minutes. DO NOT REFRESH THE PAGE.
10. Successful DOI creation will result in a **response: 201** such as:


```
10.34870/1342878 processed, response: 201
10.34870/1342888 processed, response: 201
```
11. For all other response codes, or should you encounter any issues/errors, please contact the metadata librarian.

Review the DOIs in DataCite

Do not skip this step. Return to doi.datacite.org and click to open the DOIs you just created. Did all of the fields populate correctly? If not, contact the metadata librarian.

Check the abstract to make sure that any mathematical formulas are displaying correctly after going through the upload process, you may have to copy and paste the formula from the original abstract directly into the description field in DataCite.

i.e. " $V(x, z) = -|g|\delta(z) + \lambda\delta(z)\cos(2\pi x/a)$ "

Note: The Description (Abstract) field may appear to be cut off in the preview; DataCite seems to cut off the preview at a certain character limit (which ends up being around 8 lines of text). Going into the Update DOI (Form) screen will allow you to check the description field and verify nothing has been cut off.

Preview:

URL

<https://digitalscholarship.unlv.edu/thesesdissertations/16>

Metadata

Summary View

Theoretical and computational study of time dependent scattering on a 2D surface Text

Michael Sohn

Thesis published 2010 via University of Nevada, Las Vegas

The quantum mechanical treatment of the elastic scattering of atoms from a crystal surface provides valuable information, such as surface properties and gas-surface interaction potentials. However, since it is based on the stationary state solution, it does not provide the details of the scattering process in the neighborhood of the surface, especially when atoms are physically adsorbed. In this thesis, the time evolution of the scattering process is treated in 2D with a model potential, $V(x, z) = -|g|\delta(z) + \lambda\delta(z)\cos(2\pi x/a)$, using the Gaussian wave packet approach. The focus is on the case where the Gaussian wave packet makes a transition into a selective adsorption state because it can provide information on the probability density of selectively adsorbed particles as well as the details of the scattering process in the neighborhood of the surface. The obtained Gaussian wave packet solution shows a transition into a selective adsorption state. However,

<https://doi.org/10.34917/1348560>

Update DOI (Form) view:

Description(s) (optional)

surface properties and gas-surface interaction potentials. However, since it is based on the stationary state solution, it does not provide the details of the scattering process in the neighborhood of the surface, especially when atoms are physically adsorbed. In this thesis, the time evolution of the scattering process is treated in 2D with a model potential, $V(x, z) = -|g|\delta(z) + \lambda\delta(z)\cos(2\pi x/a)$, using the Gaussian wave packet approach. The focus is on the case where the Gaussian wave packet makes a transition into a selective adsorption state because it can provide information on the probability density of selectively adsorbed particles as well as the details of the scattering process in the neighborhood of the surface. The obtained Gaussian wave packet solution shows a transition into a selective adsorption state. However, the probability density of selectively adsorbed particles cannot be accurately determined because the Gaussian wave packet constructed from the Born approximate time-independent wave function does not conserve the total probability density.

All additional information that does not fit in any of the other categories.

If you just tested a set, go back and complete the DOI creation for the rest of the records. Be careful not to create multiple DOIs for the records you used in the test.

Adding DOIs back into the metadata

At this point we want to add our newly minted DOIs back into the Batch Revise spreadsheet we downloaded earlier. One easy way to do this is to go back to our Excel we created to upload metadata to DataCite and create a new column before our *context_key* column. This column should have **10.34917/**. Create a new column combining the prefix column and the suffix column (context_key).

Copy and paste the values only into the Batch Revise spreadsheet in the DOI column.