

OPEN
ARTICLE

A proposed FAIR approach for disseminating geospatial information system maps

P. Travis Thompson¹, Sweta Ojha^{1,2}, Christian D. Powell^{1,3}, Kelly G. Pennell^{1,2} ^{ID} & Hunter N. B. Moseley^{1,4} ^{ID} 


We present a draft Minimum Information About Geospatial Information System (MIAGIS) standard for facilitating public deposition of geospatial information system (GIS) datasets that follows the FAIR (Findable, Accessible, Interoperable and Reusable) principles. The draft MIAGIS standard includes a deposition directory structure and a minimum javascript object notation (JSON) metadata formatted file that is designed to capture critical metadata describing GIS layers and maps as well as their sources of data and methods of generation. The associated miagis Python package facilitates the creation of this MIAGIS metadata file and directly supports metadata extraction from both Esri JSON and GEOJSON GIS data formats plus options for extraction from user-specified JSON formats. We also demonstrate their use in crafting two example depositions of ArcGIS generated maps. We hope this draft MIAGIS standard along with the supporting miagis Python package will assist in establishing a GIS standards group that will develop the draft into a full standard for the wider GIS community as well as a future public repository for GIS datasets.

Introduction

FAIR (Findable, Accessible, Interoperable and Reusable) is a set of guiding principles for managing publicly accessible data that emphasizes making data findable with unique identifiers and search mechanisms, easily accessible, i.e. downloadable, interoperable between computer systems and programs, and reusable for future use-cases^{1,2}. For around a decade, ideas of FAIR-like stewardship have been implemented by several stakeholders³, especially to implement infrastructure that enables machine-actionable curation, conversion, and reuse of data^{1,4}. Moreover, useful data, information, and knowledge derived from research should be shared in a FAIR manner that is transparent and usable by other researchers in related fields⁴. This extension of the FAIR principles moves closer to implementing the concept of open science by creating a research ecosystem that makes all research products FAIR for related researchers and stakeholders⁵.

However, the implementation of open science is a work in progress that requires the development of transparent procedures for depositing all types of research data and associated metadata. These procedures have been developed and codified for certain types of data, for example macromolecular structure in the Protein Data Bank⁶, genome sequence in the Sequence Read Archive, gene expression data in the Gene Expression Omnibus⁷, metabolomics data in Metabolomics Workbench⁸, and biological assay study data in BioStudies, for any biological assay modality⁹. These procedures include minimum reporting standards like MIAME, minimum information about a microarray experiment¹⁰, even though such minimum reporting standards have limitations¹¹. Even with minimum standards, deposition quality, especially in terms of metadata needed to support reusability, is a constant issue that hinders FAIRness^{12–14}. For research data types that have no procedures or minimum standards, these problems are far worse. Therefore, development of FAIR-minded procedures and minimum deposition standards and procedures are the first required steps towards bringing these data types into open science.

A geospatial information system (GIS) is software for managing, organizing, and creating visual representations (maps) of data and information that are associated with specific geographical locations (geolocations). An example GIS map for the state of Kentucky is shown in Fig. 1.

¹University of Kentucky Superfund Research Center (UKSRC), Lexington, KY, USA. ²University of Kentucky, College of Engineering, Department of Civil Engineering, Lexington, KY, USA. ³University of Kentucky, Department of Computer Science (Data Science Program), Lexington, KY, USA. ⁴University of Kentucky, Department of Molecular and Cellular Biochemistry, Lexington, KY, USA.  e-mail: hunter.moseley@uky.edu

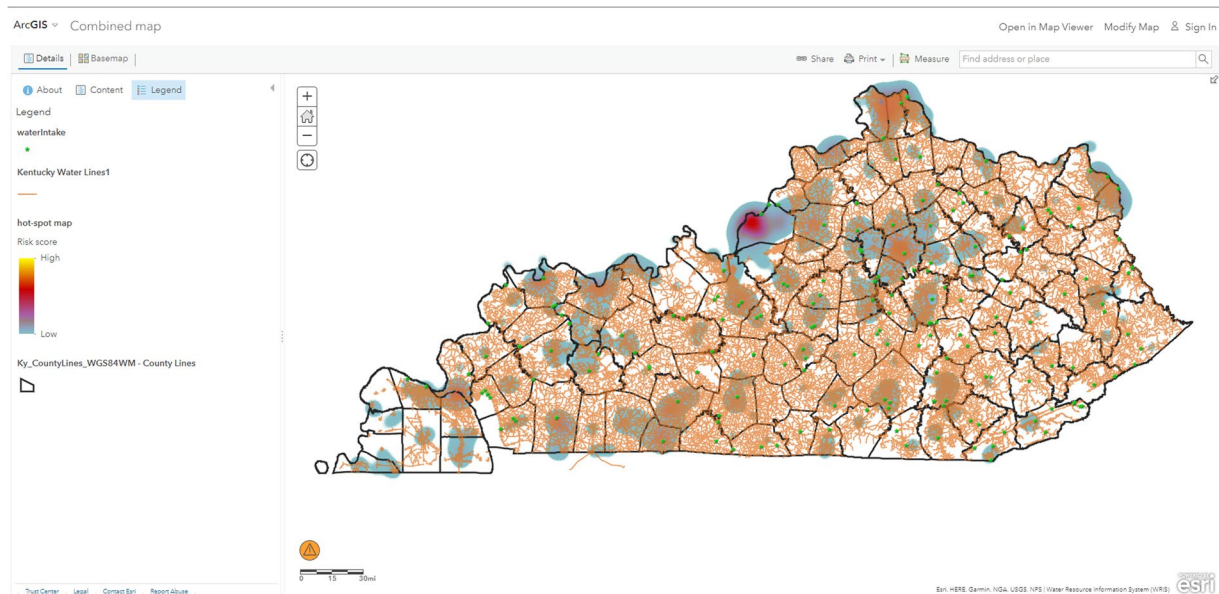


Fig. 1 ArcGIS map example³³.

GIS maps visualize any given type of data or information associated with specific geolocations. Thus, GIS datasets are composed of pairs with an arbitrary datum and its corresponding geolocation, where the first half of the pair is data type agnostic (i.e., supporting any data type). This makes GIS datasets and software very flexible with respect to the type of data being visualized. However, this data agnostic flexibility complicates deposition of GIS datasets into public scientific repositories, especially since no such open-deposit public repository currently exists for depositing GIS datasets in a FAIR manner by anyone in the broader GIS research community.

The ArcGIS Online mapping and analytics software is a very popular web-based GIS platform for generating and sharing GIS layers and maps¹⁵. However, ArcGIS layers and maps can be deleted by the user and the platform is thus not a true data repository providing dataset permanence. In addition, the ArcGIS platform is developed, maintained, and marketed by Esri, a software company headquartered in Redlands, California. An ArcGIS public account is free, but has limited non-commercial usage. Thus, ArcGIS must be licensed for full access. Also, ArcGIS does not provide unique persistent identifiers for datasets, like a Gene Expression Omnibus accession number id. Moreover, the ArcGIS object-relational database formats are proprietary. Fortunately, one of the ArcGIS map formats, Esri JSON, is in the highly computer parsable JavaScript Object Notation format; however, even this JSON format is only interoperable with some mapping and visualization tools, like open source QGIS¹⁶. While robust ecosystems of software tools have been developed to analyse geographical data and information, for example ArcGIS and QGIS, no open-deposit public repository exists for depositing such datasets in a FAIR manner by anyone in the broader GIS community. This is partly due to the development of this research community largely from civic purpose with a variety of government entities at the country, state, county, and even city level. As a government function, many GIS maps are continually maintained by these government entities in specialized GIS repositories^{17,18} without the concerns for permanence and FAIRness for the broader research enterprise. The presence of many closed and minimally FAIR GIS repositories along with the challenges of following FAIR has hindered the development of an open-deposit public GIS repository for the broader GIS community^{17,19}. This lack of an open-deposit public GIS repository will become an acute problem for many researchers trying to comply with new government policies requiring public deposition of GIS datasets generated from federal grant funding, for example the new National Institutes of Health Data Management and Sharing Policy that went into effect January 25, 2023²⁰.

Since no open-deposit public GIS repository exists, we have developed a draft Minimum Information About Geospatial Information System (MIAGIS) with associated deposition procedures for maximizing the FAIRness of ArcGIS analysed datasets and associated GIS maps. We demonstrate the application of MIAGIS and the deposition procedure to the generation of depositions for two separate sets of ArcGIS generated maps. Our goal is to kickstart the development of a FAIR deposition standard for GIS datasets as well as a community-serviceable meta repository¹⁴ for improving the Findability of GIS datasets, which could be trivially adapted for QGIS generated datasets as well. This grass roots approach has reached a level of success for other research communities²¹.

Results

The draft MIAGIS standard is described in terms of a JSON data schema for a GIS metadata deposition file as shown in Fig. 2 along with a deposition directory structure illustrated in Fig. 5. The data schema provides a comprehensive description of the deposition that is computer parsable with a level of human readability. The deposition directory structure provides a consistent organization of GIS layers/maps along with optional source data. The first part of the data schema captures general entry metadata like MIAGIS format version, entry id, and

```

{
  "format_version": "DRAFT_MIAGIS_VERSION_0.1",
  "entry_version": <version>, ## starting with "1"
  "entry_id": <entry_id>,
  "date": <date>,
  "description": <description>,
  "products": [ <id>, ... ],
  "resources": { <id> : [
    "location": <filepath | URL | DOI>,
    "alternate_locations": [ <filepath | URL | DOI>, ... ], #
    "type": <map | layer | statistics | program | other | ...>,
    "creator": [ | #2
      "name": <organization | author>,
      "type": <organization | author> ], ... ]
    "description": "...",
    "fairness": <FAIR>,
    "format": <json | xml | csv | txt | web ...>,
    "schema": <ESRI | undefined | ...>, #
    "fields": [ <fieldname> : { #3
      "name": <fieldname>,
      "type": <ontology term | int | float | str>,
      "identifier": <id>, #4
      "identifier@type": <column_name | column_index | key_name | nested>, #
      ... }, ... ],
    "sources": [ <id>, ... ] #2
  ],
  ... }
}

# - optional.
#2 - optional but strongly encouraged.
#3 - optional for "program" and "other" type, when a schema is provided, or when the
file is not Accessible.
#4 - optional, but should be specified when identifier is different from <fieldname>
or nested in a data structure. Colons are used to separate nested identifiers.

```

Fig. 2 MIAGIS JSON data schema.

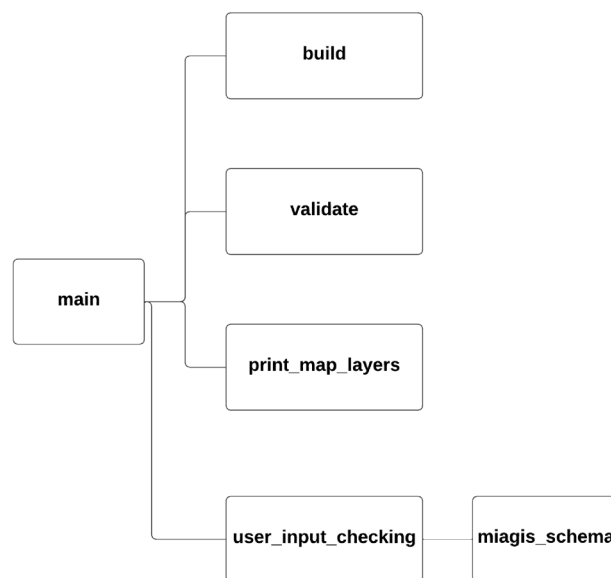


Fig. 3 The miagis Python package module organization.

entry description. The second part of the data schema contains “products” and “resources”. The heart of the data schema is the “resources” section, which captures the bulk of the description of a single data resource associated with a unique identifier along with an optional list of “sources”. The “products” and “sources” are simply lists of id’s, with each id referring back to a specific resource. Taken together, the “resources” descriptions along with the “products” and “sources” references enables data provenance for the deposition.

This data schema allows for a product-specific directed acyclic graph representation (DAG) that describes a product and the sources used to generate the product. This DAG representation can capture complex GIS analyses describing the data and computational resources used to generate a given GIS map. The “fields” subsection of the “resources” data schema was purposely designed to be very flexible with respect to data fields and their types to be applicable to the widest range of GIS datasets possible. Even some data structure nesting is allowed, but additional nesting flexibility may need to be designed into the full MIAGIS standard. In the current draft MIAGIS standard, all the top-level sections are required: “format_version”, “entry_version”, “entry_id”, “date”, “description”, “products”, and “resources”.

Given the obvious burden to generate a MIAGIS-compliant deposition, we designed and implemented the miagis Python package to facilitate the generation of the MIAGIS-compliant JSON metadata file. To reduce confusion, we use lowercase miagis to refer to the Python software package and uppercase MIAGIS to refer to the deposition standard. The package organization is illustrated in Fig. 3 and provides the command line interface (CLI) shown in Fig. 4, which has three subcommands: “build”, “validate”, and “print_map_layers”.

```

Usage:
miagis build [options]
miagis validate <metadata_json>
miagis print_map_layers <metadata_json> [--save_path=<save_path>]

Options:
--help                               Show this help documentation.
--resource_properties=<file_path>   Filepath to a csv, xlsx, or JSON file with file properties.
--exact_name_match                   If used then file name matching will be done exactly instead of fuzzy.
--add_resources                       If used then add resources from resource_properties directly to the metadata.
--overwrite_format                   If used then overwrite the determined format for files with what is in resource_properties.
--overwrite_fairness                 If used then overwrite the determined fairness for files with what is in resource_properties.
--remove_optional_fields             If used then delete optional metadata fields that are empty from files.
--json_schemas=<file_path>         Filepath to a JSON file with schemas for different JSON formats.

Base Metadata Options:
--entry_version=<integer>           Set the entry_version field for the metadata. Should be an integer starting from 1. [default: 1]
--entry_id=<id>                     Set the entry_id field for the metadata.
--description=<description>        Set the description field for the metadata
--base_metadata=<file_path>       Filepath to a JSON file with the base metadata fields to use.

```

Fig. 4 The miagis Python package command line interface.

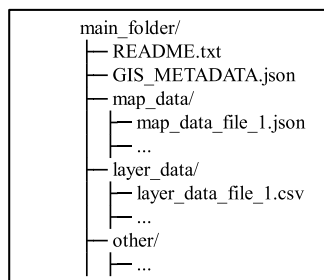


Fig. 5 MIAGIS deposition directory structure.

```

{
  "format_version": "DRAFT_MIAGIS_VERSION_0.1",
  "entry_version": 1,
  "entry_id": "KY SOP Paper",
  "description": "Data used for the paper titled \"Paper Title\" DOI: ",
  "products": [
    "Combined map",
    "Water map",
    "Map Sewer",
    "SWAPP map Ky",
    "Hotspot Map" ]
}

```

Fig. 6 Example base metadata JSON file.

When given the “build” subcommand, the miagis package analyses a MIAGIS deposition directory structure (Fig. 5), parses relevant files, and generates a MIAGIS GIS_METADATA.json file. Several files are parsed including a base metadata file, a resource properties file, an optional JSON schemas file, and any GEOJSON and Esri JSON files identified in the resource properties file or via the deposition directory structure. The base metadata file is provided via the “–base_metadata” option and has the following sections: “format_version”, “entry_version”, “entry_id”, “description”, and “products”, with an example shown in Fig. 6. Unfortunately, much of this information must be filled out by hand because it cannot be determined or inferred programmatically. Therefore, it is best for the depositor/curator to provide this file so the build process can be easily redone with a minimum of repeated human intervention. The resource properties file is either a tabular or JSON file that contains information about resources in the deposition. Both JSON and tabular representations are shown in Fig. 7 and a full JSON example is shown in Supplementary Fig. 1. The JSON schemas file provides the miagis package a JSON schema description for additional JSON formatted files to parse beyond GEOJSON and Esri JSON. Next, the user can use the “validate” subcommand to validate the GIS_METADATA.json file and fix any discrepancies either through updating input files and redoing the build and/or editing the GIS_METADATA.json file directly

A)

```
{
  <resource_name>: {
    'alternate_locations': [<URL_to_file>],
    'creator': [({'name': <creator_name>, 'type': 'organization'}, {'name': <creator_name>, 'type': 'URL'}), ...],
    'description': <description>,
    'geographical_area': <geographical_area>
  }
}
```

B)

resource_name	alternate_locations	creator	creator_type	description	geographical_area
<resource_name>	<URL_to_file>...	<creator_name>...	<organization URL>...	<description>	<geographical_area>

Fig. 7 JSON (A) and tabular (B) representations of the resource properties file.

<p>https://doi.org/10.6084/m9.figshare.16560144</p> <p>dataset/</p> <ul style="list-style-type: none"> GIS_METADATA.json README.txt layer_data/ <ul style="list-style-type: none"> groundwater_not_detected.csv groundwater_not_detected.geojson groundwater_not_detected.json gw_sampled_and_detected.csv gw_sampled_and_detected.geojson gw_sampled_and_detected.json hot-spot_map.csv hot-spot_map.geojson hot-spot_map.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.json Model_says_NO.csv Model_says_NO.geojson ... other/ <ul style="list-style-type: none"> AI LIST REPORTS RAW.csv all_12_mile_water_risk_TRI.csv gis_distance_matrix.py HISTORIC LANDFILLS MAP- 2019 RAW.csv KY_WATER_SYSTEMS.csv Model result for regression .csv PFAS DRINKING WATER REPORT FINAL.pdf PFA_SITES.csv SUPERFUND SITES RAW.xlsx TRI_2019_KY RAW.csv TRI_2019_US RAW.csv 	A	<p>https://doi.org/10.6084/m9.figshare.15218958</p> <p>dataset/</p> <ul style="list-style-type: none"> GIS_METADATA.json README.txt layer_data/ <ul style="list-style-type: none"> hot-spot_map.csv hot-spot_map.geojson hot-spot_map.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_1.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_2.json Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.csv Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.geojson Ky_Water_Resources_Polygons_DOW_SWAPP_Zone_3.json New KDEP PFAS Data_Compiled-2020 with latlong.csv New KDEP PFAS Data_Compiled-2020 with latlong.geojson New KDEP PFAS Data_Compiled-2020 with latlong.json PFAS sampling and purchasing and intake detail.csv PFAS sampling and purchasing and intake detail.json PFAS_sampling_and_purchasing_and_intake_detail.geojson waterIntake.csv waterIntake.geojson waterIntake.json water_district.csv water_district.geojson water_district.json WATER_SYSTEMS_IN_KENTUCKY.csv WATER_SYSTEMS_IN_KENTUCKY.geojson WATER_SYSTEMS_IN_KENTUCKY.json 	B
---	---	---	---

Fig. 8 Two examples of draft MIAGIS-compliant depositions.

with a text editor. The “print_map_layers” subcommand prints out the layered structure of the deposition as described in the GIS_METADATA.json file (e.g., see Supplementary Fig. 2).

MIAGIS, the deposition directory structure, and the miagis Python package were specifically designed to accommodate maps and layers generated by ArcGIS using the detailed deposition generation procedures in the Methods section. These procedures describe: (i) publishing ArcGIS layers, (ii) organizing the dataset under a consistent directory structure, (iii) creation of adequate metadata and description in draft MIAGIS JSON format, (iv) uploading the dataset to a general open data repository like Figshare²² or Zenodo²³, (v) generating a Digital Object Identifier (DOI), and (vi) and cross-linking the public ArcGIS dataset with the public open repository as a stopgap measure for making the deposition Findable. With the exception of the first major step describing the generation of ArcGIS maps, these procedures can be directly adapted to handle GIS maps and layers generated by other GIS software, especially GIS maps and layers utilizing the GEOJSON format.

We have used these deposition generation procedures to craft two separate depositions publicly available on Figshare. Figure 8A shows the DOI and directory structure of a draft MIAGIS-compliant deposition for ArcGIS generated layers illustrating likely public water systems within the State of Kentucky where per- and polyfluoroalkyl substances (PFAS) are likely to be detected²⁴. Figure 8B likewise shows the DOI and directory structure for a similarly constructed deposition for ArcGIS generated layers illustrating PFAS contamination in relation to potential PFAS users within the State of Kentucky²⁵.

Discussion

We have developed MIAGIS as a draft minimum deposition standard for GIS datasets with the associated miagis Python package for creating MIAGIS-compliant metadata in a JSONized deposition format. Furthermore, we have developed general procedures for converting ArcGIS layers/maps into a MIAGIS-compliant deposition. As demonstrated in our prior efforts^{12,13,26}, the overall goal is for this combination of draft standard, tool, and

procedures to both promote Open Science, specifically in the GIS community, as well as help reduce the burden of compliance to funding-source-specific research requirements like the new NIH Data Management and Sharing Policy. We have applied these procedures to two separate GIS dataset depositions that have utility to PFAS researchers and stakeholders within the State of Kentucky. Even with the description of MIAGIS, the deposition procedures, and the miagis Python package, it will take time for the members of the GIS community to become proficient enough to integrate these new procedures into their general workflow. We recommend that these procedures and the draft MIAGIS description be read carefully and any terms or steps that are unfamiliar should be looked up and understood. In the beginning, go slow and make sure every step is executed correctly before moving to the next step.

We hope the work presented here provides a starting point for the establishment of a GIS standards working group to further develop MIAGIS into a full standard for the GIS community. We expect an already established GIS standards organization like the Open Geospatial Consortium^{27,28} to create this MIAGIS standards working group. A natural first step, for this working group, is to incorporate relevant parts of the Content Standard For Digital Geospatial Metadata (CSDGM)²⁹ developed by the Metadata Ad Hoc Working Group of the Federal Geographic Data Committee (FGDC) and further developed into the ISO 19115 Geographic Information Metadata Standard³⁰, which is a minimum metadata content standard required for all federally generated GIS datasets. With this draft MIAGIS standard and a deposition SOP, we demonstrate the FAIRest deposition possible for publicly available maps generated within ArcGIS Online. But not all layers are necessarily FAIR. Some layers are generally Findable, but only interoperable and reusable within the ArcGIS ecosystem and not accessible. We label these layers as “FIR” within the GIS_METADATA.json file. This is an important distinction for data within restricted software ecosystems. While full FAIRness cannot be achieved in these situations, FAIRness can still be maximized. Technically, MIAGIS only improves: (i) Accessibility by providing the raw data of the map, which is not generally available through ArcGIS Online; (ii) Interoperability by providing descriptive metadata supporting data provenance in a machine readable JSON format, map data in machine readable formats, and a consistent directory organization of the deposition files; and (iii) Reusability by providing adequate data and metadata in machine readable formats. However, MIAGIS alone does not directly support Findability, which requires not only a persistent unique identifier (e.g. DOI) for a deposition, but also the systematic indexing of these identifiers in a manner that supports public searching. But we hope that the development of MIAGIS will enable the establishment of a community-serviceable meta repository for improving the Findability of GIS datasets that are Accessible from general public repositories like Figshare and Zenodo. So, either directly or indirectly, we see MIAGIS as a mechanism of improving GIS dataset FAIRness. Furthermore, we believe the associated deposition generation procedures can be trivially adapted for QGIS generated datasets as well.

Methods

MIAGIS enables a generalized procedure for making geospatial data FAIRer. The major steps of this procedure are:

1. Make your map available to the public.
2. Create a metadata file for the underlying data.
3. Make the underlying data available on a general public repository.
4. Cross link the map and underlying data.

A detailed description for performing these steps is presented below.

Procedure to publish layer publicly and share to group for an ArcGIS Online organizational account only.

- I. The layer is created and saved with proper name and tag.
- II. After layer is saved, select Details > Content and click the ellipsis (three dots) that bring up **Show item details**.
- III. The layer detail will be shown and select **Publish** that is situated on the right side of the details page. Proper name and tag should be selected to publish the layer.
- IV. When the layer is published, ‘Share’, ‘Update’ and ‘Export’ options will be seen on the right side of the layer details page.
- V. Click on **Share** and select ‘Everyone (public)’ to share the layer to public.
- VI. Click **Edit group sharing** from the bottom of the sharing option page and select groups to allow their members to export and download the data of the layer.
- VII. The last option is to select ‘Settings’ option from top right side of the layer details page and select ‘Allows others to export to different formats’ that is situated on the bottom of the Settings page.

Note: The owner can only publish and share the layer if the owner created the layer by himself or herself, making the layer FAIR, but with no permanence. The group members to whom the published layers were shared can extract or download the data in different forms, but only group members. However, if the owner extracted the layer from another source, it can be shared but its data cannot be downloaded or extracted from the layer, making it generally Findable but only interoperable and reusable within the ArcGIS ecosystem and not accessible (downloadable). In other words, the layer is “FIR”, where lowercase indicates the restriction to the closed software ecosystem. This inability to allow the public to download layer data is one reason why a public repository must be used and linked to the layer to achieve a better level of FAIRness. A user can get the layer data from the repository, and the draft MIAGIS standard describes where the data is located for each layer.

Procedures for creating a draft MIAGIS-compliant deposition and making it publicly available on Figshare. ArcGIS Online lacks the ability to create Digital Object Identifiers (DOIs), so this feature is outsourced to Figshare. But first, you will need to make an account with Figshare. Then a Figshare deposition item can be created as follows:

- a. On your computer create a folder to hold all the data, including maps, layers, and any other data.
- b. Within that folder, create the following subfolders as needed: 'map_data/', 'layer_data/', and 'other/'. Note that the forward slash indicates a subfolder on Mac and Linux operating systems, while a back slash indicates a subfolder on Windows.
- c. Place whole maps (integrated collection of layers) into the 'map_data/' subfolder.
- d. Place the processed layer data (data used to create feature layers) in the 'layer_data/' subfolder.
- e. Place any other data or files in the 'other/' subfolder.
- f. Create a README.txt (or README.md) file to be included in the main folder which provides a description of the datasets included and a copy of any relevant license agreements.
 - Add a URL to the dataset on ArcGIS Online into this README file.
 - README.md is a Markdown text file that will allow formatted visualization in a browser.
- g. Create a MIAGIS-compliant GIS_METADATA.json file that will contain metadata and a description of all files and data sources.
- h. The file directory should now look like this:

```
main_folder/
├── README.txt
├── GIS_METADATA.json
├── map_data/
│   ├── map_data_file_1.json
│   └── ...
├── layer_data/
│   ├── layer_data_file_1.csv
│   └── ...
├── other/
└── ...
```

- i. Compress the main folder to create an archive (i.e., ZIP, TAR.GZ, etc.).
- j. In the 'My data' menu, click '+Create a new item'.
- k. In the new item window, drag and drop the compressed archive or click browse to upload the datasets to the Figshare item.
 - l. Fill out the remainder of the fields in the new item window.
- m. In the new item window, click 'Reserve Digital Object Identifier'. This will create a single DOI that can be used for the datasets.
- n. In the new item window, click 'Save changes'.
- o. The item can be edited by reopening the new item window by clicking on the item in the "My data" menu. After editing the data, follow the process below:
 - i. Reopen the added item window by clicking on the item in the "My data" menu.
 - ii. In the item window, make sure Publish is selected.
 - iii. In the item window, click 'Save changes'.
 - iv. The data of the Figshare item can be updated and all updated versions can be seen in the Figshare item; for example, the Figshare item for Ojha *et al.*²⁴, with the following <https://doi.org/10.6084/m9.figshare.16560144>³¹. Multiple version updates can be seen in this Figshare item.

Creating MIAGIS-compliant metadata JSON file. Figure 2 illustrates the draft MIAGIS JSON data schema. The helper miagis Python package is available to generate most of this metadata file, which can then be edited by the user. The miagis package can also validate the final JSON metadata file before inclusion in the Figshare item. The miagis Python package is written in Python 3.7, is pip installable, and has comprehensive end-user documentation with installation instructions for Linux, Mac, and Windows operating systems. The package has both a command line interface (CLI) and application programming interface (API). Therefore, data curators can directly use miagis via its CLI as well as incorporate miagis into deposition software pipelines via its API.

Updating ArcGIS content to include a DOI. The description pages data files, feature layers, and story maps created from the datasets, which were uploaded to the Figshare item, should be updated to include the generated DOI.

- a. Sign into ArcGIS Online.
- b. From the main menu bar, click 'Content' to show content items.
- c. Click on the desired data file, feature layer, or story map to update.
- d. In the content's description, add the DOI.
 - E.G.: "https://doi.org/10.3390/metabo11030163"
- e. Ensure that each data file, feature layer, and story map is published, and the sharing setting is set to public.

Data availability

A Figshare item titled "A Geospatial and Binomial Logistic Regression Model to Prioritize Sampling for Per- and Polyfluorinated Alkyl Substances (PFAS) in Public Water Systems-Dataset" which includes GIS layers derived from the results of a regression model. <https://doi.org/10.6084/m9.figshare.16560144>³¹.

A Figshare item titled "A FAIR approach to detect and share PFAS hot-spot areas and water systems in Kentucky", which includes GIS layers identifying PFAS hot-spot areas and water systems in Kentucky. <https://doi.org/10.6084/m9.figshare.15218958>³².

Code availability

The miagis Python package is available on GitHub and the Python Package Index with comprehensive user documentation on GitHub.io:

<https://github.com/MoseleyBioinformaticsLab/miagis>
<https://pypi.org/project/miagis/>
<https://moseleybioinformaticslab.github.io/miagis/>

Received: 4 January 2023; Accepted: 31 May 2023;

Published online: 16 June 2023

References

1. Wilkinson, M. D. *et al.* The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data* **3**, 160018 (2016).
2. Boeckhout, M. & Zielhuis, G. A. & Bredenoord, A. L. The FAIR guiding principles for data stewardship: fair enough? *European journal of human genetics* **26**, 931–936 (2018).
3. Bishop, B. W. & Hank, C. Measuring FAIR principles to inform fitness for use. (2018).
4. Jacobsen, A. *et al.* *FAIR principles: interpretations and implementation considerations*. MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info ... **2**, 10–29 (2020).
5. National Academies of Sciences, E.; Medicine. Open science by design: Realizing a vision for 21st century research (2018).
6. Protein Data Bank. the single global archive for 3D macromolecular structure data. *Nucleic acids research* **47**, D520–D528 (2019).
7. Clough, E. & Barrett, T. The gene expression omnibus database. *Statistical genomics* (pp. 93–110. Humana Press, New York, NY, 2016). DOI: DOI 10, 978–971.
8. Sud, M. *et al.* Metabolomics Workbench: An international repository for metabolomics data and metadata, metabolite standards, protocols, tutorials and training, and analysis tools. *Nucleic acids research* **44**, D463–D470 (2016).
9. Sarkans, U. *et al.* From arrayexpress to biostudies. *Nucleic Acids Research* **49**, D1502–D1506 (2021).
10. Brazma, A. *et al.* Minimum information about a microarray experiment (MIAME)—toward standards for microarray data. *Nature genetics* **29**, 365–371 (2001).
11. Edgar, R. & Barrett, T. NCBI GEO standards and services for microarray data. *Nature biotechnology* **24**, 1471–1472 (2006).
12. Smelter, A. & Moseley, H. N. A Python library for FAIRer access and deposition to the Metabolomics Workbench Data Repository. *Metabolomics* **14**, 1–8 (2018).
13. Powell, C. D. & Moseley, H. N. The mwtab Python Library for RESTful Access and Enhanced Quality Control, Deposition, and Curation of the Metabolomics Workbench Data Repository. *Metabolites* **11**, 163 (2021).
14. Powell, C. D. & Moseley, H. N. The Metabolomics Workbench File Status Website: A Metadata Repository Promoting FAIR Principles of Metabolomics Data. *bioRxiv* (2022).
15. ArcGIS Online. Available online: <https://arcgis.com/> (accessed on)
16. QGIS.org, QGIS Geographic Information System. QGIS Association. <http://www.qgis.org>, accessed on 04/27/2022 (2022).
17. Kinkade, D. & Shepherd, A. Geoscience data publication: Practices and perspectives on enabling the FAIR guiding principles. *Geoscience Data Journal* **9**, 177–186 (2022).
18. Pampel, H. *et al.* Making research data repositories visible: the re3data. org registry. *PloS one* **8**, e78080 (2013).
19. Degbelo, A. FAIR geovisualizations: definitions, challenges, and the road ahead. *International Journal of Geographical Information Science* **36**, 1059–1099 (2022).
20. Health, N. I. O. Final NIH policy for data management and sharing. *NOT-OD-21-013. Vol NOT-OD-21-013. NIH Grants & Funding. Bethesda, MD: Office of The Director, National Institutes of Health* (2020).
21. Sicilia, M.-A., García-Barriocanal, E. & Sánchez-Alonso, S. Community curation in open dataset repositories: insights from Zenodo. *Procedia Computer Science* **106**, 54–60 (2017).
22. Thelwall, M. & Kousha, K. Figshare: a universal repository for academic resource sharing? *Online Information Review* (2016).
23. Zenodo. <https://zenodo.org>. Assessed on 04/27/2022. Available online: (accessed on)
24. Ojha, S. *et al.* A geospatial and binomial logistic regression model to prioritize sampling for per- and polyfluorinated alkyl substances in public water systems. *Integrated environmental assessment and management* **19**, 163–174 (2023).
25. Ojha, S., Thompson, P. T., Powell, C. D., Moseley, H. N. & Pennell, K. G. Identifying and sharing per- and polyfluoroalkyl substances hot-spot areas and exposures in drinking water. *Scientific Data* **Accepted**.
26. Thompson, P. T., Powell, C. D. & Moseley, H. N. Academic Tracker: Software for tracking and reporting publications associated with authors and grants. *PloS one* **17**, e0277834 (2022).
27. Reed, C. N. The open geospatial consortium and web services standards. In *Geospatial Web Services: Advances in Information Interoperability*, IGI Global: pp. 1–16 (2011).
28. Reichardt, M. Open geospatial consortium standards. *International Encyclopedia of Geography: People, the Earth, Environment and Technology: People, the Earth, Environment and Technology* 1–8 (2016).
29. Committee, F. G. D. Content standard for digital geospatial metadata (CSDGM). http://www.fgdc.gov/standards/status/sub2_1.html (2000).

30. Karschnick, O. *et al.* The UDK and ISO 19115 Standard. In Proceedings of EnviroInfo; pp. 475–481.
31. Ojha, S. *et al.* A Geospatial and Binomial Logistic Regression Model to Prioritize Sampling for Per- and Polyfluorinated Alkyl Substances (PFAS) in Public Water Systems-Dataset. *Figshare* <https://doi.org/10.6084/m9.figshare.16560144> (2021).
32. Ojha, S., Moseley, H. & Powell, C. D., Pennell, K. G. & Thompson, P. T. A FAIR approach to detect and share PFAS hot-spot areas and water systems in Kentucky. *Figshare* <https://doi.org/10.6084/m9.figshare.15218958> (2022).
33. Ojha, S. ArcGIS map of Kentucky Water Lines. Available online: <https://www.arcgis.com/home/webmap/viewer.html?webmap=06829286d1274d94b0d4d4be911502f1&extent=-91.8196,35.4102,-81.053,39.7525> (accessed on)

Acknowledgements

The project described is supported by University of Kentucky Superfund Research Program Grant P42 ES007380 (Pennell) and NSF 2020026 (Moseley). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institute of Environmental Health Sciences nor the National Science Foundation.

Author contributions

H.N.B.M. conceptualized the GIS deposition process. P.T.T., C.D.P. and H.N.B.M. designed MIAGIS and the associated JSON data schema, and the deposition directory structure. P.T.T. and H.N.B.M. designed the miagis Python package and P.T.T. implemented it. K.G.P. obtained access to data sources and reviewed map creation. S.O. gathered data sources and created the ArcGIS maps. S.O. created an early draft of the manuscript with heavy revisions from P.T.T., minor revisions from C.D.P. and K.G.P. and a major rewrite by H.N.B.M.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41597-023-02281-1>.

Correspondence and requests for materials should be addressed to H.N.B.M.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023