Agroecology Partnership: Data Management Guidelines

LifeWatch ERIC

October 30, 2025

Table of contents

Introduction

Welcome to the Data Management Guidelines documentation of the Agroecology Partnership. This online documentation expands on the official Agroecology Partnership Data Management Plan (DMP) and complements it with more practical and technically detailed information.

While the DMP outlines the full strategy for data handling across the partnership and it is updated every 2 years, the present documentation is meant to be updated as we learn from doing. Its objective is to support partners with hands-on guidance, concrete examples, and step-by-step explanations of the data pipeline — from collection to publication — including how to make data FAIR (Findable, Accessible, Interoperable, and Reusable). This space aims to be more to the point than the DMP, helping teams implement the plan more effectively in day-to-day work. Furthermore, being a public document, we aim to offer the wider agronomy community with state-of-the-art data management practices in the Agroecology domain.

For any questions, please write us a line at agroecology-data@lifewatch.eu.

1 Identifying a data asset or research output

Some project outputs are only useful internally, while others are relevant to people outside the project. The rule is:

> If an output could be useful to someone outside the project, it is a research output and must be published — unless there is a valid reason not to.

Outputs that are only useful within the project (e.g. working drafts, internal consultations, or coordination notes) should be stored only on the project's internal cloud.

2 Collect and send the metadata

Metadata are *data about data*. They provide the essential context that allows us to understand, locate, and reuse a dataset or research output. Without metadata, information can easily become unusable or forgotten. With metadata, we ensure that each data asset is described clearly, remains findable, and can be connected to the broader research landscape.

In our partnership, collecting and submitting metadata is a collective responsibility. Whenever you identify a research output, you should first inform your **task leader**. The task leader then notifies the data team at **agroecology-data@lifewatch.eu**, while copying the Work Package lead. This way, every new output is tracked from the beginning.

The task leader is also responsible for collecting the metadata fields listed in Annex I of the DMP and included here below in the Table 1 for convenience. There are two ways to provide them:

- By filling in the Excel metadata form available on our SharePoint.
- By sending the necessary information directly to the data management team, who can complete the form on your behalf.

Once the metadata are collected, the data team will work with you to identify the type of data, select an appropriate standard, and decide on a trusted repository where the asset should be published. This ensures that our outputs are stored securely, remain discoverable, and align with FAIR principles.

---

*Table 1. Metadata fields and definitions. They are largely based on the **DataCite schema**, which is an international standard for describing research data. Each field has a short definition and an example to help you fill it in correctly.*

Section	Field	Definition	Example
Data denomination	**Data set reference name***	A unique reference name starting with the prefix "DA_AGROECOLOGY".	`RO_AGROECOLOGY_stakeholders_surv`

Section	Field	Definition	Example
	Data set title*	A short, descriptive title, easily searchable.	`Stakeholders_Survey_Q2_2023`
	Description	A brief explanation of the data content.	`Survey on technological tools used in agroecology.`
	Keywords	Terms that describe the content and make it discoverable.	`survey, tools, living labs`
	Version number*	Version identifier for tracking changes.	`V 1.0`
Data origin	**Creator(s)***	Names, affiliations, and countries of dataset creators.	`Jane Doe, University of X, Belgium`
	Data source	How or why the dataset was generated or re-used.	`Generated dataset`
	Creation date*	Date when the dataset was produced.	`12.06.2023`
	Quality assurance	Description of quality checks or validation performed.	`Response rate monitoring, data profiling`
Data specifications	**Type***	General type of data.	`Survey data`
	Format*	File format(s).	`.csv`
	Expected size*	Approximate dataset size.	`1 MB`
Data accessibility	**Data location***	Repository where the dataset is stored.	`GBIF`
	Repository submission date*	Date of deposit.	`2023.07.15`
	Persistent identifier (PID)*	DOI or other permanent identifier.	`https://doi.org/10.xxxx/abcd`
	Access status*	Level of access (open, consortium, restricted).	`Consortium`
	Embargo period*	Embargo duration, if applicable.	`No embargo`

Section	Field	Definition	Example
	Funding statement	Funding acknowledgement text.	`This dataset was generated within the AGROECOLOGY partnership funded by the EU.`
Data utility	**Significance inside the partnership***	Work Package or task the dataset comes from.	`WP5 - Task 5.1`
	Significance outside the partnership	Who else may benefit from the dataset.	`Policy makers`
Data publications	**Related publications**	References to outputs derived from the dataset.	`Deliverable 3.2, Article XYZ`
	Data citation	A ready-to-use citation for the dataset.	`Doe, J. & Smith, A. (2023). Stakeholders Survey Q2 2023. Zenodo.`

* Mandatory fields

3 Deposit the data in the project's internal repository

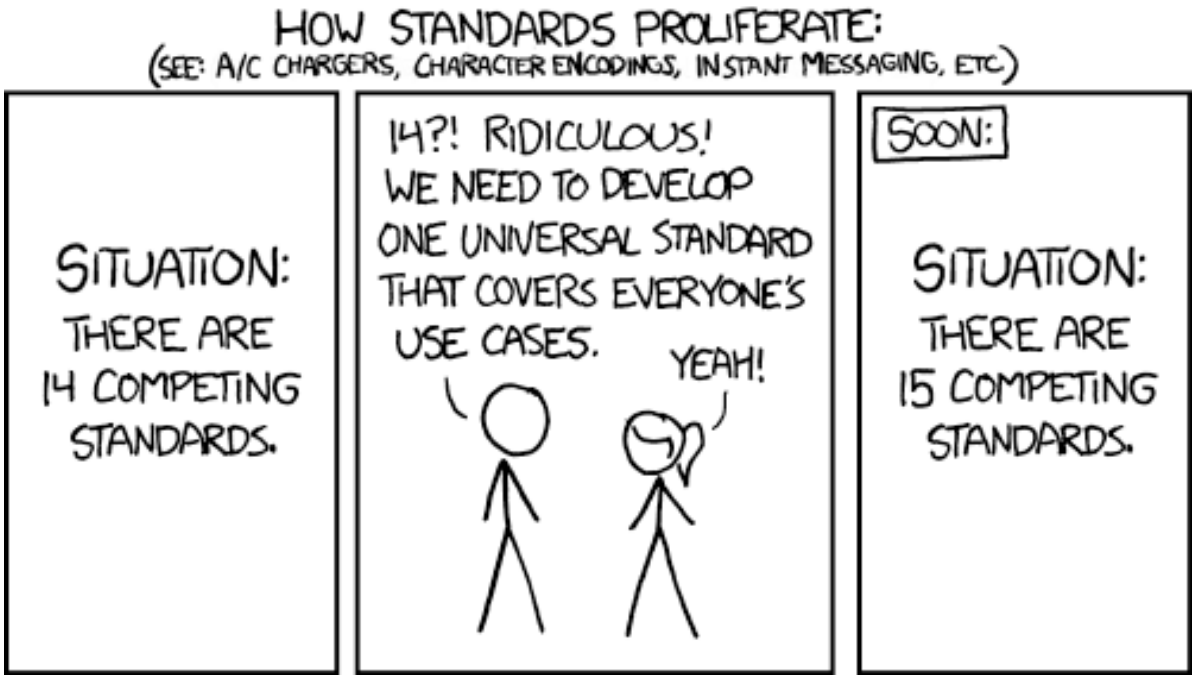
The Agroecology partnership uses a dedicated SharePoint cloud drive as its internal repository. Access is restricted to project partners, and it serves as the central place to store and share data, analysis, and outputs. Whenever possible, you should save your work directly here, in the relevant directory, so that it remains available to the consortium and properly backed up.

The only exception concerns personal data. To comply with GDPR, any files containing personal information must not be uploaded to the shared drive. Instead, they must remain in your institution's own secure storage.

Part I

Data Standarization

Data on Agroecology are multidisciplinary: There are biological data, socio-economic, geographical and many more. The main goal is to standardize data to the preferred community standard. However, sometimes there is not a clear winning standard, or the winning standard may not meet the FAIR principles. This chapter describes the preferred data standards for each data type collected in the Agroecology Parnership. In some cases, when there is not an obvious standard, we suggest evidence-based way of structuring data that fits the FAIR principles.



Source: xkcd

4 Tabular data

Tabular data are often referred as **data structured in rows and columns, where each row contains values for a set of properties and each column represents a specific property of the things described by the rows** (Tennison and Kellogg 2015). Following certain best practices and well-studied structures when creating tabular data make it easier to analyze, visualize and reuse - by others or by your future self.

4.1 Best practices for Tabular Data

Tabular data can **follow very different structures**. There is a lot of flexibility in how you name your columns or what values you put in every cell. This flexibility is at the same time a **double-sided knife**: follow the right recommendations, and your data will be easy to analyze. Do the wrong decisions, and your same data could become not usable. Best practices go around columns names and consistency. Here we will describe common mistakes and how to solve them:

4.1.1 File formats

While you are free to use proprietary spreadsheet software such as Microsoft Excel or Google Sheets to work on your data, **we strongly recommend storing datasets in `.csv` or tab-delimited formats**. This ensures compatibility across software and, most importantly, protects your work from accidental obsolescence. Imagine you open an old project and find that, after a software update, some formulas no longer work, column types have shifted, or the file won't even open.

Open formats like `.csv` are readable by virtually any tool, now and decades from now. If a problem ever arises, there's a global community of developers who can build solutions to recover your data. Proprietary formats are fine for analysis, but their internal rules can change without notice, and you might only discover it when it's too late.

In short: **use Excel or Sheets to crunch numbers if you like, but keep your raw data in `.csv`** and publish both raw and processed datasets in `.csv` whenever possible. Avoid `.xlsx` as your master copy. And you can always open `.csv` in Excel for analysis without locking your work into a single program's rules.

4.1.2 Handling Empty cells

Most software handling text, tabular files, are able to recognize empty cells. This is, a cell where a value is not added. In Excel, these show as blanks, while in other tools like R or Python they show as "NA" or "NULL". **We recommend to leave empty cells when a value is not known**. A common mistake is to add 0's in a numeric variable when a value is unknown. This could distort further analysis as 0's are considered meaningful values. Common file formats that allow empty values are `.csv`, `.txt` or `.xlsx`.

Certain formats don't allow adding empty values. In this case, it is recommended to add a placeholder such as `-9999` in numeric variables, or `NA` in text variables. This blank value must be documented properly. For instance, the binary format `NetCDF` does not allow empty cells, but you can document empty values by using the attributes `missing_value` or `_FillValue`.

4.1.3 Column names

The names used to describe your variables should be **clear and consistent**. Avoid spaces or special characters, and adopt standard units. Examples:

- `max_temp_celsius` instead of `Max Temp Celsius`
- `soil_ph` instead of `Soil-pH`

We recommend to use lowercase and underscores _ instead of spaces. **This style is known as `snake_case`**. Other well-known style is `camelCase`, where spaces are avoided and words are spitted by using a Capital letter to emphasize the different words.

4.1.4 Adhere to community standards

Think of data like a written language: if everyone makes up their own grammar, nobody can understand each other. In data, this applies not only between humans but also between humans and machines. **We rely on computers to analyze our data, but unlike us, they're terrible at guessing meaning**. To them, data are just zeros and ones. In science, the key is in the context, so **we need to shape our data so both people and machines can read them correctly.**

The easiest way to do that? **Follow community standards**. Chances are, you're not the first person working in your field. Others have faced the same problems and worked hard to solve them. Communities dealing with certain types of data — geospatial, surveys, biodiversity, genetics — have already agreed on rules and formats so that other researchers and software can make sense of the data. Take dates as an example. You could write them in many ways: `DD/MM/YYYY`, `MM/DD/YYYY`, or even with month names. But most software understands them best in the format `YYYY-MM-DD`. This is not arbitrarty: it's an international standard (ISO 8601) designed after much discussion and testing.

In the next chapters, we will look at several community standards relevant to Agroecology, and we will recommend which ones to use for maximum clarity and interoperability.

4.1.5 Separate raw data from calculations

Raw data should always live apart from any data cleaning, wrangling, or analysis. Even if your raw file has typos, strange characters, or awkward formatting — leave them as they are. They're part of the original record.

Do all cleaning and analysis in a separate file or script (R, Python, Julia, Excel formulas…) and share that too. **This will enable anyone can reproduce your process and get the same results**. This is of course, unless you are unsure of your analysis and embarrassed to show to others. In that case you should proofread until you are happy.

Why keep the artifacts? Because it preserves a trustworthy copy of the original, makes your work reproducible, and gives you a safe point to return to if something later goes wrong.

4.2 Table schema

A table schema is a data dictionary that standardizes and explains each variable in your dataset. It records key information such as the data type of each column (numeric, date, string, boolean…), expected units, allowed values, or format rules. By clearly describing what each column should contain, the schema ensures that your data is consistent, interpretable, and aligned with project standards.

The schema is stored as a separate file that should accompany your dataset. It is typically named after the dataset it describes, for example: `mydataset.schema.json`, `mydataset.schema.yaml`, or even a text/Word version for non-technical users. Including the schema with the dataset provides a clear reference for anyone using or reviewing the data.

A schema also allows datasets to enforce rules automatically: for instance, specifying which fields are required, the expected data type for each column, the valid range of numeric values, or the expected format for dates and text. These rules make the dataset self-descriptive and easier to integrate with other datasets following the same standards.

4.2.1 Frictionless Table Schema

Within the Agroecology Partnership, we recommend using the Frictionless Table Schema to define and document tabular datasets. Frictionless is an open-source framework designed to reduce common data workflow issues—what they call *friction*. It provides both **standards** and **software**: the standards define how data should be structured, and the software applies these rules to automatically transform, validate, and describe datasets.

In Agroecology, we will use one of the Frictionless standards: The Frictionless Table Schema. This specification allows you to define tabular data—such as field names, data types and formats. While it is primarily designed for text-based tables like CSV, it can be extended to other tabular formats.

For example, consider a minimal dataset with three columns: crop_species, crop_yield_kg_per_ha, and planting_date. The dataset contain inconsistencies such as empty cells, inconsistent numeric formats, or irregular date formats. We will see how to detect this issues automatically in the next section Quality Control.

crop_species	crop_yield_kg_per_ha	planting_date
Maize	5200	2023-03-15
Wheat	4.8 tons	2023-03-22
maiz	4900	
Rice		2023-04-01
Barley	5500	2023/03/18

A table schema describes each column with properties like:

- `name`: the column name
- `type`: expected data type (string, number, date)
- `description`: human-readable explanation of the field
- `format`: expected format for the values
- `rdfType`: optional semantic identifier linking the field to a concept in a controlled vocabulary (See the note below for more information)

Applied to our example:

```yaml
fields:
  - name: crop_species
    type: string
    description: Crop species (scientific name)
    rdfType: http://aims.fao.org/aos/agrovoc/c_1972
  - name: crop_yield_kg_per_ha
    type: number
    description: Crop yield in kilograms per hectare
    rdfType: http://aims.fao.org/aos/agrovoc/c_10176
  - name: planting_date
    type: date
    format: '%Y-%m-%d'
    description: Date when the crop was planted
    rdfType: http://aims.fao.org/aos/agrovoc/c_24065
```

```yaml
primaryKey: crop_species
missingValues:
  - ""
```

All this information is saved into a file. The standard accepts two machine readable formats: `json` and `yaml`. For us however, we will also accept text formats such as MS Word. While this is a not machine readable format and not widely used for this purpose, we understand that is a well-known format by the general public, and can act as an entry point to encourage non-technical users to document their data. We consider that it's better to have documentation in a human-readable format than none at all. Converting a bullet point list in Markdown or Word to JSON/YAML later is a minimal effort compared to the cost of having undocumented data. You can see this example in the three formats below by clicking in each of them

example.schema.json

```json
{
  "fields": [
    {
      "name": "crop_species",
      "type": "string",
      "description": "Crop species (scientific name)",
      "rdfType": "http://aims.fao.org/aos/agrovoc/c_1972"
    },
    {
      "name": "crop_yield_kg_per_ha",
      "type": "number",
      "description": "Crop yield in kilograms per hectare",
      "rdfType": "http://aims.fao.org/aos/agrovoc/c_10176"
    },
    {
      "name": "planting_date",
      "type": "date",
      "description": "Date when the crop was planted (format: yyyy-MM-dd)",
      "format": "yyyy-MM-dd",
      "rdfType": "http://aims.fao.org/aos/agrovoc/c_24065"
    }
  ],
  "primaryKey": "crop_species",
  "missingValues": [""]
}
```

example.schema.yaml

```yaml
fields:
  - name: crop_species
    type: string
    description: Crop species (scientific name)
    rdfType: http://aims.fao.org/aos/agrovoc/c_1972  # crops
  - name: crop_yield_kg_per_ha
    type: number
    description: Crop yield in kilograms per hectare
    rdfType: http://aims.fao.org/aos/agrovoc/c_10176  # crop yield
  - name: planting_date
    type: date
    format: '%Y-%m-%d'
    description: Date when the crop was planted
    rdfType: http://aims.fao.org/aos/agrovoc/c_24065  # planting date
primaryKey: crop_species
missingValues:
  - ""
```

example.schema.docx

> **Note**
>
> The property `rdfType` is an optional property we use to link each field to a concept in a controlled vocabulary or ontology. It helps machines (and humans) understand the meaning of a field beyond its name and link to many other resources in the web, and plays an important role in building AI-ready, machine-readable datasets and knowledge graphs.
> For example, the AGROVOC multilingual thesaurus by FAO has a controlled vocabulary for the word "crop yield", linked to the same term in many languages and linkes to other concepts. It can be found at the URI below:
>
> `http://aims.fao.org/aos/agrovoc/c_10176`
>
> Other ontologies with controlled vocabularies that are useful in Agroecology are the Ecoportal, the Agroportal or the Survey Ontology.

4.3 Tidy data for seamless analysis

We recommend to follow a wide tabular data structure for archiving and publishing tabular data, as described before. This is because the table schema allows to explain in detail the data, and quality control can be automatized.

However, when analyzing these data, following the principles of tidy data can make tabular datasets easier to work with (Wickham 2014). **In tidy data, each row represents a single observation, each column a variable, and each table a distinct type of observation**. For example, consider this table with grape yield and wine production across three European regions . A non-tidy table might look like this:

farm	grape_yield_kg_h	wine_production_hl_ha
Bavaria (DE)	7000	60
La Rioja (ES)	5000	40
Peloponnese (GR)	6000	50

Here, different variables are spread across columns, and units are written inside the values. While this structure is easy to understand for most of us, it is harder to analyze for machines:

- **Variable names as column headers:** Each variable requires a separate column, so adding new variables (e.g., sugar content, harvest date) would require adding new columns, breaking automation.
- **Inconsistent row structure:** If some observations are missing a value, it's harder to handle missing data systematically.
- **Limited scalability:** Combining multiple datasets or reshaping data for plotting, statistical analysis, or modeling requires extra preprocessing steps.
- **Filtering and grouping complexity:** Aggregating or comparing values across variables requires multiple steps instead of simple column-based operations.

A tidy version of the same data represents each measurement as a separate row with explicit columns for the variable type, value, and unit, so that **each row contains a single observation of a single variable**.

Farm	Variable	Value	Unit
Bavaria (DE)	Grape Yield	7000	kg/h
Bavaria (DE)	Wine Production	60	hl/ha
La Rioja (ES)	Grape Yield	5000	kg/h
La Rioja (ES)	Wine Production	40	hl/ha
Peloponnese (GR)	Grape Yield	6000	kg/h
Peloponnese (GR)	Wine Production	50	hl/ha

This tidy format makes it easy to filter, group, or plot by crop type, farm, or yield, and it aligns with modern data analysis workflows. Following these principles helps ensure that tabular data are both **clear** and **practical** for downstream use.

5 Survey data

Typically, surveys are created using an online tool. The link to the questionnaire is later passed to the participants, and the responses are collected in the platform. This method is both convenient and more effective compared to surveys done via text documents or email. However, there are certain steps to follow to ensure that the survey design and the results are fully interoperable and they are properly preserved in the long term, when software changes or gets deprecated. We will propose the use of three common survey software: We recommend the use of KoboToolbox as the first choice, since it is compatible with the XLSForm format. Alternatively, LimeSurvey can also be used. EUSurveys can be used only if there is need to adhere to very specific EU requirements that the other platforms cannot meet, as it has no tools for exporting the survey in interoperable formats.

5.1 XLSForm and KoboToolbox

We recommend to adopt the XLSForm format for survey authoring. This format allows to write from scratch a survey following certain standard rules. The advantage is that there is a large community that has adopted this format and many tools have been developed, not only to create surveys and deploy online automatically, but also to automate reports, analysis and more. The format propose the use of a spreadsheet such as MS Excel, OpenOffice calc or Google Sheets. Three sheets shall be created: the first one "survey" contains the questions and their type, including free text answers, multiple or single choice and many more.

Table 1. Survey sheet. Note in the column "type", the text just after "select_one" and "select_multiple" are unique identifiers that are used in the next sheet "choices" to link the questions to the possible answers. the column "name" is a unique identifier of the question. "label" has the text of the question that is shown to participants. It is possible to write the labels in several languages (via suffix ":en" for English, ":fr" for French, etc) to tackle participants that speak different languages.

type	name	label::en
text	Q1_name	What is the name of the platform?
text	Q2_website	What is the website of the platform?
text	Q3_contact_name	Who is the primary contact for this survey?

type	name	label::en
text	Q4_contact_email	What is the email address of the primary contact?
select_one list_5_agroecology	Q5_agroecology	Is the platform designed for the Agroecology community?
select_multiple list_6_focus	Q6_focus	Which thematic areas best describe the focus or services of your platform?

The second sheet is "choices", containing the possible answers of questions with single or multiple choices. They are linked through unique identifiers.

Table 2. Choices sheet. The column "list_name" contains the unique identifier that was given in the column "type" of the survey sheet. The column "name" has unique identifiers. The column "label" has the text of the answer that is shown to participants .It is possible to write the labels in several languages (via suffix ":en" for English, ":fr" for French, etc) to tackle participants that speak different languages.

list_name	name	label::en
list_5_agroecology	yes	Yes
list_5_agroecology	no	No
list_6_focus	agroecological_practices	Agroecological practices
list_6_focus	data_and_monitoring_tools	Data and monitoring tools
list_6_focus	training_and_knowledge_exchange	Training and knowledge exchange
list_6_focus	policy_and_governance_support	Policy and governance support
list_6_focus	collaboration_and_stakeholder_engagement	Collaboration and stakeholder engagement
list_6_focus	other	Other

Lastly, there is a "settings" sheet that allows to write metadata, rules and more.

Table 3. Settings sheet. In this case, it only contains the name and version of the survey.

form_title	form_id	version
Agroecology Platform Survey	agroecology_platform_survey	v1.0

5.1.1 Using KoboToolbox for survey design and deployment

KoboToolbox is an online platform that supports the XLSForm standard and allows researchers to design, deploy, and manage surveys for free. It was originally created for humanitarian organizations but is now widely used in research and development contexts. KoboToolbox offers

an intuitive web interface, enabling all users to create complex forms that include conditional logic, geolocation, multimedia inputs, and more.

While KoboToolbox offers optional paid features for large organizations or high data volumes, its free tier is sufficient for most research projects. The platform is open to collaboration with academic institutions and complies with the EU General Data Protection Regulation (GDPR). A Data Processing Agreement (DPA) can be signed online to ensure full compliance with data protection requirements.

5.1.1.1 Steps to create and archive a survey using KoboToolbox

1. **Register** for a free account at https://www.kobotoolbox.org.
2. **Design your survey** directly in the platform using the form builder, or import an existing XLSForm spreadsheet.
3. **Export the survey structure**:
   - Inside your project, go to **Form → (three dots in the top-right corner) → Download XLS**.
   - This export follows the XLSForm standard.
4. **Export each sheet to `.csv`** for long-term archival.
   This ensures maximum interoperability and readability across software and time.
5. **Deploy the survey** to collect responses by sharing the generated link with participants.
6. **Download the responses** at **Data → Downloads → CSV** once data collection is complete.
7. **Archive together** the survey design (XLSForm and CSV versions) and the responses (CSV).

This workflow ensures the entire lifecycle of the survey—from authoring to data collection and preservation—follows open, well-documented standards that can be reused or audited in the future.

5.1.1.2 Additional resources

- **Free documentation**: https://support.kobotoolbox.org/getting_started_xlsform.html
- **XLSForm templates**: Google Sheets template

5.1.2 Alternative tools

Other tools are compatible with the XLSForm standard. The most notable is Open Data Kit (ODK), an open-source software ecosystem that supports field data collection, including offline mobile apps.
Although ODK is fully open-source, its cloud hosting service has costs. Researchers could deploy their own ODK instance if resources allow. However, given the free hosting provided by KoboToolbox (with limitations), this is the most practical option for most research groups.

5.1.3 FAIRness and long-term preservation

The XLSForm format, being based on spreadsheet structures (`.xlsx` or `.csv`), guarantees long-term accessibility and interoperability. Even if the tools used today (such as KoboToolbox) were to become unavailable, the survey structure and data would remain readable and reusable. This makes XLSForm a strong choice for compliance with FAIR principles (Findable, Accessible, Interoperable, and Reusable).

To ensure full FAIRness: - Always archive both the survey structure (`.xlsx` and `.csv`) and the responses (`.csv`). - Store these files in an open data repository or institutional archive. - Include a short README file documenting the purpose, date, authorship, and data collection context.

Regarding the survey structure and responses, we recommend using text-delimited files, such as csv, which are widely adopted in data science and research workflows. These files offer several advantages: they are machine-readable, open, and not tied to any proprietary software, ensuring that data remains accessible and understandable in the long term.

To promote consistency and interoperability, we propose the following specifications for csv files:

- Use semicolons `;` as column separators.
- Use double quotes `"` to delimit text strings.
- Use UTF-8 as the character encoding.

We avoid commas `,` as separators because they are often used in free-text responses and could disrupt the file structure. Similarly, we recommend restricting the use of semicolons `;` and double quotes `"` to their intended purposes only (as column separators and string delimiters, respectively). Whenever possible, configure the survey platform to block or escape these characters in user-provided text. If such restrictions are not feasible, respondents should be instructed not to use these characters when completing the survey.

6 Quality Control of Data

Validating the data is about making sure data makes sense before sharing it. For tabular data, this means checking that required fields are not empty, that numbers are really numbers, and that dates or text follow a consistent format. Verification looks outward: values should also be plausible, like coordinates pointing to the right country or measurements staying within expected ranges.

Doing this by hand is tedious and error-prone, which is why we rely on tools. A good way to start is to define an schema: a file that declares rules about the columns in your table. The schema establishes if the columns have numeric, character or date values. It sets expectations such as minimum and maximum values or mandatory fields.

Once the schema is in place, there are that tools can automatically check your dataset against it and flag any problems: missing values, numbers in the wrong format, typos in categories, or values outside the valid limits. This shifts the burden from manual inspection to automated validation, making the process faster, more consistent, and easier to repeat every time the dataset is updated.

6.1 Frictionless Data Packages and the Frictionless Table Schema

Within the Agroecology Partnership, we recommend the use of the Frictionless Table Schema to validate and verify datasets automatically. In the section about standardizing tabular data, we explained how this standard allow to define the columns of your data table. Furthermore, the standard allows to write rules to automatically validate the data: what are the data types, expected values, maximum and minimum values, handle missing values, and many more.

6.1.1 Validating Data with the Open Data Editor

The Open Data Editor is a user-friendly tool that leverages the Frictionless Table Schema to perform automated checks on datasets. The technical documentation of the Open Data Editor explains in detail how to install the app, and how to highlight errors.

7 Data Publishing

Once the data is correctly formatted, data should be published in a trusted repository, ideally before 6 months since its creation and always before the end of the Agroecology partnership. You should archive your data in a trusted repository even if you are writing scientific publication based on them, but archiving does not mean the data will be public yet. Most repositories allow to submit data and keep it private until you decide. A list of trusted repositories is published by (Lazzeri 2024).

Once the data are in the trusted repository, you should get a DOI for your data. You can include the DOI in the spreadsheet in sharepoint, or mail agroecology-data@lifewatch.eu with the DOI and we will do it for you. If the repository does not provide a DOI, let us know.

References

Lazzeri, E. 2024. "Update of the Study on the Readiness of Research Data and Literature Repositories to Facilitate Compliance with the Open Science Horizon Europe MGA Requirements (1.0)." Zenodo. https://doi.org/10.5281/zenodo.13919643.

Tennison, Jeni, and Gregg Kellogg. 2015. "Model for Tabular Data and Metadata on the Web." World Wide Web Consortium; W3C Recommendation. https://www.w3.org/TR/2015/REC-tabular-data-model-20151217/.

Wickham, Hadley. 2014. "Tidy Data." *Journal of Statistical Software* 59 (10): 1–23. https://doi.org/10.18637/jss.v059.i10.