pISA-tree - a data management framework for life science research projects using a standardised directory tree

Marko Petek*[†], Maja Zagorščak[†], Andrej Blejec[†], Živa Ramšak, Anna Coll, Špela Baebler, Kristina Gruden

Department of Biotechnology and Systems Biology, National Institute of Biology, Večna pot 111, SI-1000 Ljubljana, Slovenia

* corresponding author (email: marko.petek@nib.si)

[†] these authors contributed equally to this work

Abstract

We have developed pISA-tree, a straightforward and flexible data management solution for organisation of life science project-associated research data and metadata. It enables *on-the-fly* creation of enriched directory tree structure (project/Investigation/Study/Assay) via a series of sequential batch files in a standardised manner based on the ISA metadata framework. The system supports reproducible research and is in accordance with the Open Science initiative and FAIR principles. Compared with similar frameworks, it does not require any systems administration and maintenance as it can be run on a personal computer or network drive. It is complemented with two R packages, *pisar* and *seekr*, where the former facilitates integration of the pISA-tree datasets into bioinformatic pipelines and the latter enables synchronisation with the FAIRDOMHub public repository using the SEEK API. Source code and detailed documentation of pISA-tree and its supporting R packages are available from https://github.com/NIB-SI/pISA-tree.

Keywords: data management, FAIR principles, standardisation, reproducible research, Open Science, Open Data, SEEK, ISA

Introduction

Compared to a decade ago, present-day life science projects generate vast amounts of data, requiring a substantial amount of research time and funds dedicated to analysis and adequate storage of experimental results. Both researchers and funders are aware that completed published results often lack important information that would ensure their reproducibility. Raw data supporting researchers' conclusions are frequently either inaccessible, inadequately organised or not sufficiently annotated to allow straightforward reuse (Baker, 2016a, 2016b; Miyakawa, 2020). To alleviate these issues, the FAIR data principles were proposed (Wilkinson et al., 2016) that advocate policies which ensure findability, accessibility, interoperability and reusability of the generated data.

Minimum information metadata standards, providing guidelines for standardised reporting of experimental results, have been developed for a large number of wet-lab assays; e.g. plant phenotyping (MIAPPE; Papoutsoglou et al., 2020), quantitative PCR (MIQE; Bustin et al., 2009), gene expression microarrays (MIAME; Knudsen & Daston, 2005), proteomics (MIAPE; Martínez-Bartolomé et al., 2014); as well as dry-lab assays, e.g. quantitative modeling (MIRIAM; Novère et al., 2005) and simulations in systems biology (MIASE; Waltemath et al., 2011). Standards, databases and policies related to FAIR data sharing are catalogued at FAIRsharing.org (Sansone et al., 2019).

To organise and keep track of experimental metadata, the ISA framework defines metadata files on three hierarchical levels: Investigation, Study and Assay (ISA; Sansone et al., 2012). Each level contains files describing experimental goals, ontologies, design, protocols, and detailed description of experimental conditions. ISA framework developers also provide ISA tools, a software suite which, among others, contains a metadata editor and packages for integration with common programming languages (González-Beltrán et al., 2014; Johnson et al., 2021; Kuhring et al., 2019).

For the duration of a research project, datasets are frequently stored and analysed locally, usually on the researcher's personal computer or institution's network drive. Institutions have various strategies on how to deal with organisation and medium to long-term storage of project data. These strategies may include the use of a Laboratory Information Management System (LIMS), electronic laboratory notebook (ELN), and either centralised or decentralised storage of raw and analysed data. However, for research projects, such software solutions can have some disadvantages - they might be associated with license costs, are prone to rigidity, can become unsupported over the course of a few years, or are unable to handle diverse and large scale datasets (Arend et al., 2016; Kanza et al., 2017). Several solutions for research data management providing long term data and metadata storage exist, such as Zenodo (www.zenodo.org), SEEK/FAIRDOMHub (Wolstencroft et al., 2017), openBIS (Barillari et al., 2016), ISA tools (Rocca-Serra et al., 2010), COPO (Shaw et al., 2020), and e!DAL (Arend et al., 2020), however they require substantial administration if run locally.

Here we describe pISA-tree, an easy-to-use and maintain system for intra-institutional organization and storage of research data, with a special emphasis on the generation of adequate metadata. pISA-tree is compliant with minimal requirements of the ISA framework and FAIR principles, utilises minimum information guidelines for metadata formulation, and encourages the inclusion of metadata at an early stage of research. Its complementary R packages also enable easy transfer to and from public (meta)data repositories and (re)use of data in analytical R pipelines.

Implementation

We developed pISA-tree and its accompanying R packages (*pisar* and *seekr*) to provide a user-friendly system that guides researchers towards organisation of their data without the requirement of any advanced systems maintenance.

pISA-tree functionality. pISA-tree is based on a consecutive set of batch files that generate an enriched directory tree structure for scientific research projects. It uses a familiar directory system with a standardised structure to organise each **p**roject hierarchically into three nested ISA levels: **I**nvestigations, which consist of multiple **S**tudies, where each individual study can contain multiple **A**ssays (Fig. 1).

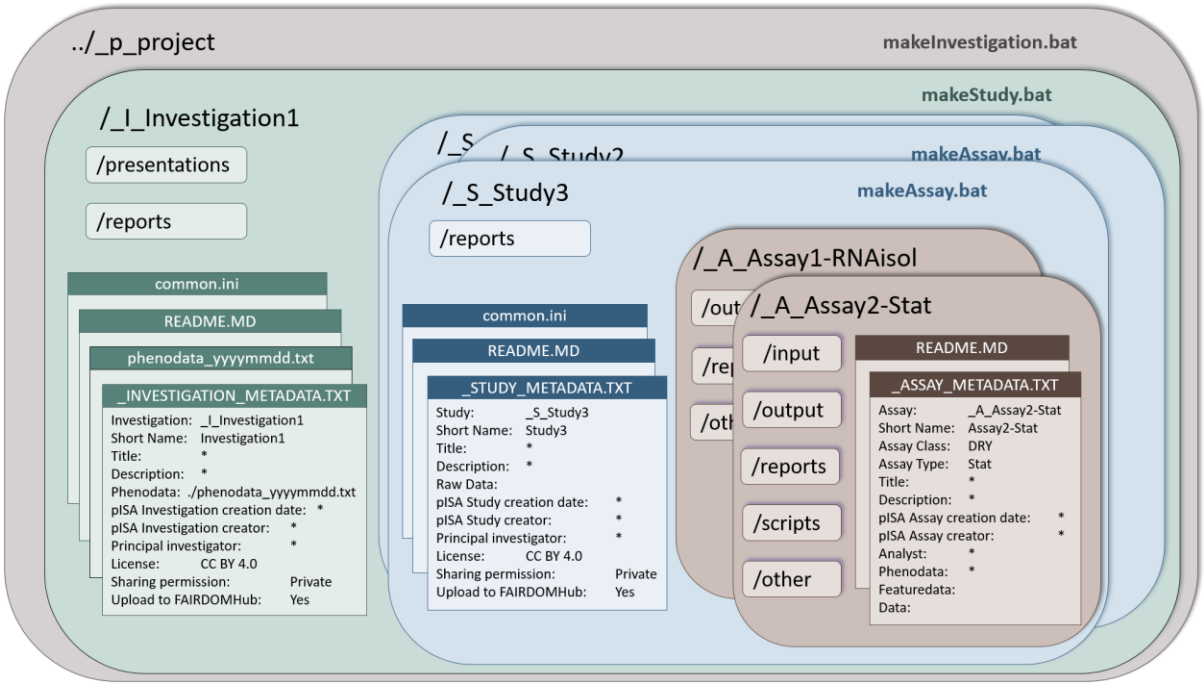

Fig. 1. pISA-tree directory tree structure and corresponding metadata files. Large dark shaded rectangles with rounded corners represent pISA-tree levels and corresponding directories, light shaded rectangles with rounded corners represent supplementary subdirectories as predefined for each level, and rectangles represent pISA-tree related files. Bolded file names in the upper right corner of the levels represent batch files (.bat) that generate new pISA-tree layers.

One of the core components of the system are consistent metadata files, associated with each pISA-tree level entry. Prior to folder creation, the user answers a series of questions (keys) with predefined attributes or free-text answers (values), which are then stored in metadata files as

tab-delimited key-value pairs, that can be further edited with text editing tools. This encourages users to enter the minimal information at an early stage of the research preparation and brings benefits in finding relevant information later. In addition, updated and consistent metadata can be used in the data analysis stage.

While the upper levels, i.e. project, investigation and study, hold a more descriptive role, the assay level is the richest level of the research record. Each data analysis step can be defined as a separate Assay. Moreover, pISA-tree can support two different classes of assays: laboratory experiment (wet) and data analysis (dry). Each assay class contains several pre-prepared assay types, depending on measurement methodology (e.g. RNA isolation, qPCR, GC-MS). The assay class and type define the subdirectories, i.e. standard sublevels such as `output` or `reports`, and other standardised information that is typically required. For example, the wet-lab assays will have the `output\raw` subdirectory for storing raw experimental data, whereas the dry-lab assays will have the `scripts` subdirectory (Table 1). All options and predefined minimal information are kept in text files and can be customised to extend assay classes and types. For each new class or type, the user adjusts directory structure and defines metadata template files.

Additional pISA-tree features for easy metadata creation. Before creating any project sub-directories, key-value pairs that should be present in all sublevel metadata files, e.g. organisation or licensing info, can be added to the `common.ini` file. This way key-value pairs will be automatically assigned to the complete set of directories within the project and one will not need to re-enter them with every new investigation or study. The investigation level specifically holds the master sample description table, i.e. the phenodata file (`phenodata_yyyymmdd.txt`), which should be populated with all relevant information on all the experimental samples that will be collected in the associated studies and further analysed

in corresponding assays. Users are also encouraged to create descriptions of all measured variables within assays and store it in a featuredata file (`featuredata_yyyymmdd.txt`). Each study holds assays conducted on a set of samples from a single coherent experimental setup. More details on how to structure and keep track of the potential changes in the phenodata file can be found in the pISA-tree user manual (Supplementary File 1). Certain assay types additionally generate an analyte table (`analytes.txt`) which serves as a template for input of specific measurements (e.g. sample quality control values), thus helping the wet lab researchers to input the results of experiments into the system. All levels contain three auxiliary batch scripts: `showtree.bat` displays the complete directory tree structure and allows for better overview of the information in the system for complex projects; `showmetadata.bat` displays all recorded metadata thus far; and `xcheckMetadata.bat` displays missing mandatory metadata fields. These auxiliaries are all meant to help the scientists when reviewing their project status in terms of data management completeness, thus promoting data FAIRness.

Table 1. Automatically generated pISA-tree directories and metadata files and recommended allocation of data files. Each level contains level-specific metadata files and directories containing data. '-' directory not created at this level**.**

	pISA-tree levels			
created directories / *files*	**INVESTIGATION**	**STUDY**	**ASSAY**	
			wet-lab ASSAY	**dry-lab ASSAY**
level-specific metadata files	`_INVESTIGATION_METADATA.TXT` `phenodata_yyyymmdd.txt` (master sample table) `featuredata_yyyymmdd.txt` (variable description file)	`_STUDY_METADATA.txt`	`_ASSAY_METADATA.TXT` `analytes.txt` (analyte table)	`_ASSAY_METADATA.TXT` `analytes.txt` (analyte table)
input	-	-	-	input data set(s) or file(s) containing path(s) to input data
scripts	-	-	-	scripts, pipelines, workflows or commands used for the analyses
output	-	-	pre-processed results	final results
output/raw	-	-	raw result files, e.g. data collected from the machine	-
reports	summary reports for project partners and stakeholders	sample handling or any other study related protocols or reports	assay specific protocols, laboratory logs	documented procedures, protocols, log files with tool(s) version(s) used for the analyses
presentations	any presentation related to the project	-	-	-
other	-	-	other relevant files not included elsewhere	other relevant files not included elsewhere

Integration of pISA-tree into web-based data management systems. Because the R environment for statistical computing has a firmly established developer and user community for bioinformatic data analysis, with well-maintained packages, we further complemented pISA-tree with two R packages. The predefined pISA-tree directory structure enables integration into automated data analysis pipelines, while R enables repeatability of analyses, e.g. by including output of the *SessionInfo()* function in the analysis reports. Therefore, we further complemented pISA-tree with *pisar*, an R package that enables import of pISA-tree metadata into R workflows (Fig. 2) and thus repeatability of R analyses within the pISA-tree system. Since information is structured in a standardised way and augmented by metadata, connection to FAIR data management and cloud platforms based on ISA framework can be established. The other part of the system enables seamless upload to the project metadata repository FAIRDOMHub (Wolstencroft et al., 2017) using the R package *seekr* (https://github.com/NIB-SI/seekr) which interacts with the FAIRDOMHub API (Fig. 2).

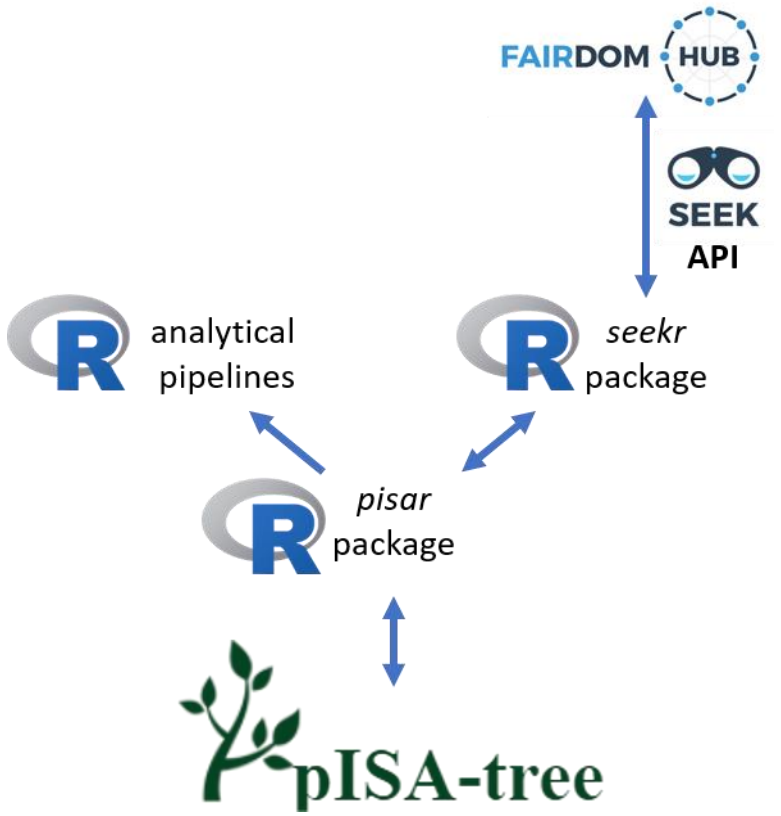

Fig. 2. pISA-tree information flow between public metadata repositories and R analytical pipelines. Data flow between pISA-tree, R environment and the metadata repository FAIRDOMHub is represented with blue arrows. Users can acquire pISA-tree metadata for analysis in R using the *pisar* package.

Results

FAIR paradigms are becoming integral to the way we conduct scientific research. We developed pISA-tree for local project data management to facilitate implementation of these paradigms in day-to-day research. We illustrate the usage of pISA-tree with two example life science projects described below, both including diverse assay types. The seekr package was used for the automatic upload of the complete projects to FAIRDOMHub, thus making these projects data easily accessible and reusable.

Example 1: Insect pesticide development – a multidisciplinary project. The first example is a project with both wet and dry lab assays ("Using RNAi and systems biology approaches for validation of insecticide targets in CPB guts", https://fairdomhub.org/projects/252). The scope of this project was to develop double-stranded RNA (dsRNA) based insecticide against Colorado potato beetle pest. It included identification of Colorado potato beetle gene targets, establishing dsRNA production, and validating the dsRNAs' insecticidal potential in laboratory and field trials (Petek, Coll, et al., 2020).

The directory tree of this project consists of three investigations covering altogether 16 studies with wet-lab and dry-lab assays (Fig. 3). The investigation "_I_01_LabTrials" includes 11 studies, the first three of which contain data on dsRNA design and production, while the rest contain data on feeding trials. The phenodata file in this investigation describes all samples in the underlying assays in detail. The "RNAisol" wet-lab type assays in the feeding trial studies

include analyte files that contain information on isolated RNA quantity and quality parameters measured for samples analysed in the particular trial.

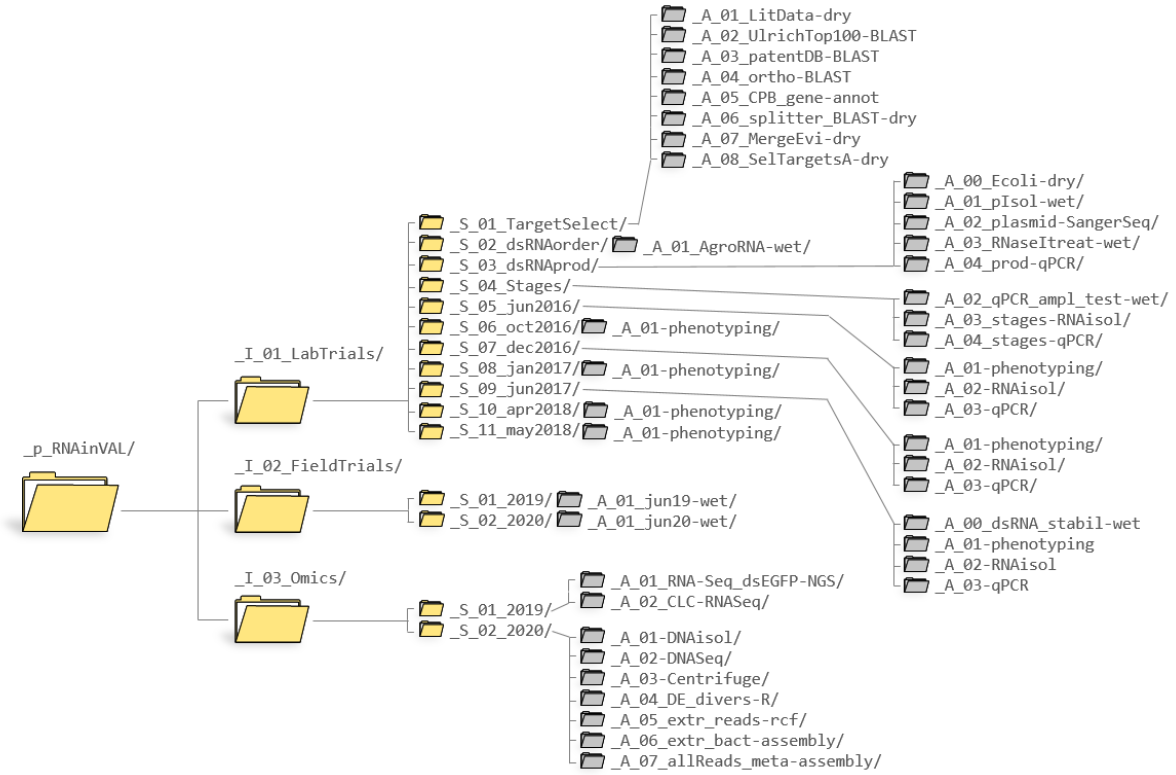

Fig. 3. pISA-tree directory structure of the project from Example 1. Only the pISA-tree level directories are shown for clarity; with project, investigation and study directory icons in yellow and assay level icons in grey.

Example 2: *De novo* potato transcriptome assembly – a large-scale dry-lab project. The second example is an implementation of pISA-tree in a large-scale dry-lab project ("*Solanum tuberosum* Reference Transcriptomes", https://fairdomhub.org/projects/161). This project's aim was to construct a pan-transcriptome from high-volume data accessible from publicly available resources and in-house defined workflows in a collaborative way (Petek, Zagorščak, et al., 2020).

The project consists of a single investigation, four studies and a plethora of data files (Fig. 4), encompassing four consecutive bioinformatics procedures: (1) high-throughput sequencing acquisition and pre-processing, (2) *de novo* assembly, (3) cultivar-specific transcriptomes generation and annotation, and (4) pan-transcriptome generation and annotation. Assays nested within the Studies are structured as predefined for the dry-lab assay class (see Table 1), yet no additional specific assay types were developed within this project, as they were custom for each step of the developed assembly and quality control pipeline.

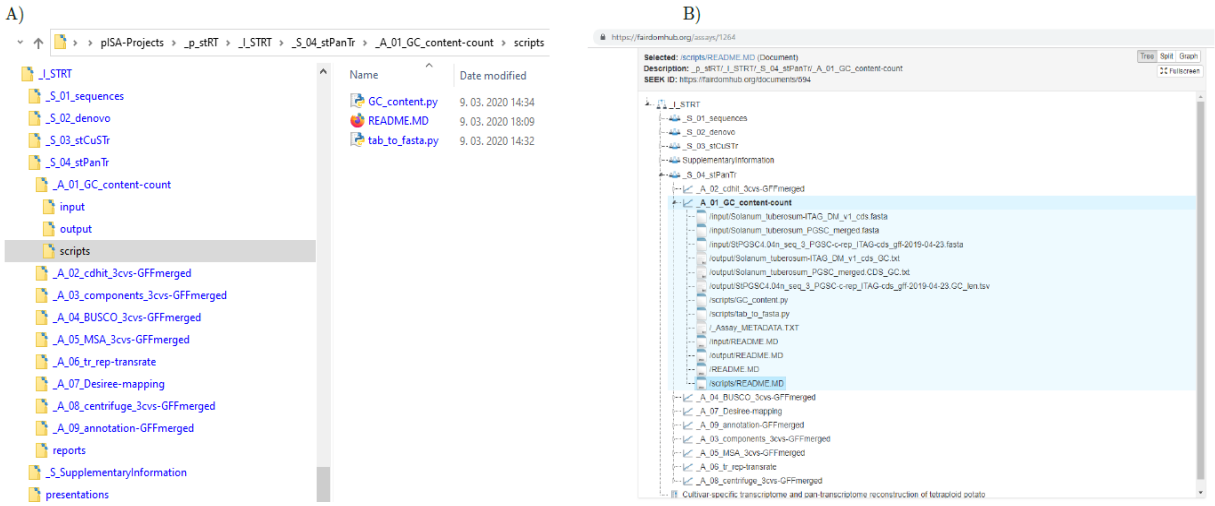

Fig. 4. Local pISA-tree directory structure and the corresponding FAIRDOMHub structure of the investigation from Example 2. A) Hierarchical structure of investigation "_I_STRT" from the project of Example 2 as stored on the local network drive, with one of the assays shown in detail. B) Same pISA-tree structure as shown in A stored at FAIRDOMHub and an example of all file types within one of the assays shown in detail. Slash characters (/) in file names are used to denote the subdirectory paths of the listed files in the assay.

Discussion

To promote the adoption of FAIR principles, many funding organisations and journals are committed to FAIR and open data principles (Bloemers & Montesanti, 2020; McQuilton et al.,

2020). Proposals submitted for evaluation are expected to contain a reasonably detailed data management plan. This plan often has to incorporate policies for data collection, organisation and exchange during the project, during the publication process and after the project's closure, i.e. the plan should take into account local storage and deposition to publicly accessible repositories. The life science community has at large adopted the core FAIR values. Bottlenecks lie in the implementation of said management plans, partially because the data management systems in place are not designed for researchers that typically collect, annotate and organise the data. As these systems are usually organised on an institutional level, their practicality and usefulness for individual researchers or research groups are often not taken into account (Leonelli et al., 2017). In that respect, pISA-tree is distinct, initiated by end user requirements and implemented by developers. The development went through several rounds of refinements to accommodate most of the involved life scientists' needs. It is essential that scientists internalise the benefits of using a standardised data management system, before the data collection starts. Only then will the data and metadata be seamlessly and systematically collected and organised, which in return encourages reusability. This is highly endorsed in pISA-tree. For example, the phenodata and featuredata files, as well as all considered experimental protocols are expected to be prepared before performing the assays as a part of the experimental design. Considering the impracticality of strict requirements for immediate metadata input, which could discourage end users (Leonelli et al., 2017), pISA-tree allows users to input only some of the relevant metadata, to be completed at later stages.

Compared to other life science project data management systems, the advantage of pISA-tree is in its simplicity and flexibility. It relies on file system directories, inherent to all operating systems and can be used by scientists with basic computer skills following a short practical course or careful reading of the user manual (Supplementary File 1). It is also not limited to specific laboratory or computational technologies, as the generic wet and dry metadata

templates can be applied and adjusted to any new technology. Metadata are stored in plain text files that can be edited with little effort. Other advantages include open source code, independence from web browsers and straightforward access control. Since manually organizing metadata into ISA-Tab format is a quite complex task for life scientists, pISA-tree is a step towards the adoption of the FAIR paradigm. We believe that a simple system will be more easily adopted by a wide range of scientists, thus increasing compliance with FAIR principles. For the same reason, pISA-tree does not require professional technical support. Nevertheless, we recommend that a data steward be nominated to take care of maintaining order, access control, updates, uploading to public repositories, and data backup. Therefore, pISA-tree is particularly practical for smaller institutions or research groups with limited resources and staff.

Due to its design, pISA-tree users and data stewards have to be aware of some of its limitations. One is the limitation of the maximum path length to 247 characters in Windows operating systems, which can only be circumvented by changing the Windows registry key. To avoid this issue, the usage of short directory and file names is recommended. In addition, no automatic file and directory version control is enabled, except by using subsidiary software. In the user manual (Supplementary File 1), we advise the users on how to manage phenodata and featuredata file versioning, however for all other files a different versioning logic can be applied. The lack of version control can also be solved by uploading to FAIRDOMHub promptly after each significant update to the project. Taking into account the accessibility premise from the FAIR data principles, integration with FAIRDOMHub is also an elegant solution for sharing data with collaborators, stakeholders or the public.

Although pISA-tree was designed for small to medium sized projects, it has proven to be flexible enough to be adapted for larger projects with multiple independent research partners, which we showed in two EraCoBiotech projects SUSPHIRE (http://susphire.info/) and INDIE

(https://indie.cebitec.uni-bielefeld.de/), both including five partners from three different countries.

Methods

Batch scripts. pISA-tree scripts and template files are organised in the main directory and the Templates subdirectory. The `Templates` subdirectory is further subdivided into `x.lib`, `DRY` and `WET`, which contain templates for step-by-step creation of bat files, and dry-lab and wet-lab assay classes. The batch script named `makeProject.bat` is stored in the main directory, and is the one to be used at the start of the pISA-tree structure creation.

The main Windows executable batch file, `pISA.cmd`, is stored in the `\Templates\x.lib` directory. Depending on the user's progress in building the pISA-tree structure for their project, this script generates additional `makeInvestigation.bat`, `makeStudy.bat` and `makeAssay.bat` at the appropriate levels within the project and initialises template metadata files. The directory names are automatically given a prefix defining the level i.e. "`_p_`" for project, "`_I_`" for Investigation, "`_S_`" for Study and "`_A_`" for Assay, which facilitates visual inspection and general usage of the structure. In addition, Assay directory names are automatically given an "`-<AssayType>`" suffix, that are defined in the `Templates` directory. The level-specific metadata files are text files containing a collection of tab-delimited key-value pairs in the form "`<key:>\t<value>`" and can be extended to suit the user's needs via any text editor.

Another set of scripts can be used to inspect project directories and metadata files. The `showTree.bat` draws the pISA-tree directory tree structure to the `TREE.TXT` file, while the `showMetadata.bat` outputs a file containing all metadata in either plain text or markdown format, depending on the user's position within the tree. The `xcheckMetadata.bat`

highlights the missing values in the metadata files required by the minimum reporting standards, and kindly reminds the user to fill them in.

Setting up the pISA-tree. To set up the pISA-tree, download the GitHub repository (https://github.com/NIB-SI/pISA-tree) into a local or network directory with read/write/execute permissions. This "pISA-tree root directory" ideally contains all pISA-tree projects. The setup does not require any administrative installation privileges of the user and the entire directory tree can be moved freely, if only relative paths are used in the metadata files, which is one of the recommendations given in the manual (the current version available as Supplementary File 1).

Modifying and extending assay types using metadata templates. The metadata associated with specific assay types is defined in template files in the `\Templates\DRY` and `\Templates\WET` class directories. Users can add new classes of assays, extend the existing types, or develop their own. To predefine the questions (and potentially multiple choice answers) that the user will be asked when creating a new assay entry, one adds rows of "`<key:>\t<value>`" pairs into the `<AssayType>_Template.txt` file. Here, keys define the questions and '/' separated values provide menu choices. Users can also enter their own answers not provided in the template by selecting the option "`other`".

Integration of pISA-tree with R and FAIRDOMHub. R package *pisar* enables repeatability of R-based analyses within the pISA-tree system. The users can retrieve recorded metadata information and the complete directory structure of the pISA-tree to reuse the dry-lab analysis protocols. The *pisa()* function retrieves the relative paths of the pISA-tree directory structure up to the project level whereas the *getMeta()* function retrieves specific fields (key values) from

the metadata files. Thus, one can rerun R scripts without changing the code even if the metadata files are changed. As good practice, we recommend listing all metadata used at the end of the report.

The *seekr R* package was developed for communication of the pISA-tree with the SEEK API used by the FAIRDOMHub public repository and enables bulk upload of pISA-tree investigations. Currently, before uploading, the project page has to be created on the FAIRDOMHub web page to define user roles and permissions. The user can specify to exclude certain directories or files within the pISA-tree project directory tree by editing the "`seekignore.txt`" file. Further, uploading individual levels can be controlled by changing "`Upload to FAIRDOMHub`" key's value in the level metadata file, which is, by default, set to "`Yes`". In addition, the "`Sharing permission`" key sets the sharing of an individual level at FAIRDOMHub as either "`Private`" or "`Public`", while the key "`License`" determines the license under which the data is shared to the public.

Bulk investigation upload to FAIRDOMHub using *seekr* is implemented with the R functions *skFilesToUpload()* and *skUploadFiles()*. The first function creates a list of files to be uploaded and the second function uses this list to prepare a JSON object for the specified investigation and all its accompanying files. It then and communicates with the SEEK API to create the corresponding FAIRDOMHub objects and then uploads the selected files. Package *seekr* uses the package *pisar* to obtain the local pISA-tree directory structure and metadata.

Data availability

Example project data are available on FAIRDOMHub data repository under the project SEEK IDs 161 (https://fairdomhub.org/projects/161) and 252 (https://fairdomhub.org/projects/252).

Code availability

The code and documentation for pISA-tree is available at http://github.com/NIB-SI/pISA-tree. R packages and documentation are available from https://github.com/NIB-SI/pisar and https://github.com/NIB-SI/seekr. The code is openly available under MIT license terms.

Acknowledgements

The authors would like to thank Tjaša Lukan for testing pISA-tree, Valentina Levak and Katja Stare for helping with assay type template construction, Wolfgang Müller for helpful advice and discussions, and Carissa Bleker for critical reading and language revision. This work was funded by the Slovenian Research Agency grants (P4-0165, J4-4165, J4-7636, J4-8228, J4-9302 and Z7-1888), BioPharm.Si project (OP20.00363), funded by Ministry of Education, Science and Sport of Republic of Slovenia and European regional fund, European Union's Horizon 2020 research and innovation programme projects ADAPT (grant agreement No. 862858) and EOSC-Life (grant agreement No. 824087), and European Research Area Cofund Action 'ERACoBioTech' (grant agreement No. 722361) projects SUSPHIRE and INDIE, which received funding from the Slovenian Ministry of Education, Science and Sport. For Fig. 2, logos of R, SEEK and FAIRDOMHub were retrieved from https://www.r-project.org/logo/ and https://fair-dom.org/about-fairdom.

Authors contributions

M.P. and M.Z. wrote the first draft and coordinated manuscript editing.

A.B. wrote the batch scripts and both R packages.

A.B., A.C., K.G., Š.B. and Ž.R. conceptualised the idea.

A.C., M.P., M.Z., K.G, Š.B. and Ž.R. wrote documentation and exhaustively tested the pISA-tree system.

M.P., M.Z., K.G, Š.B, A.C and Ž.R. contributed to the usability and improvement of the system.

K.G. secured funding, supervised and managed the project.

All authors read, reviewed and edited the manuscript draft and approved the final submission.

All authors also contributed to the writing of a detailed user manual.

Competing interest statements

The authors declare no competing interests.

References

Arend, D., Junker, A., Scholz, U., Schüler, D., Wylie, J., & Lange, M. (2016). PGP repository: a plant phenomics and genomics data publication infrastructure. *Database*, *2016*. https://doi.org/10.1093/database/baw033

Arend, D., König, P., Junker, A., Scholz, U., & Lange, M. (2020). The on-premise data sharing infrastructure e!DAL: Foster FAIR data for faster data acquisition. *GigaScience*, *9*(10). https://doi.org/10.1093/gigascience/giaa107

Baker, M. (2016a). How quality control could save your science. *Nature*, *529*(7587). https://doi.org/10.1038/529456a

Baker, M. (2016b). 1,500 scientists lift the lid on reproducibility. *Nature*, *533*(7604). https://doi.org/10.1038/533452a

Barillari, C., Ottoz, D. S. M., Fuentes-Serna, J. M., Ramakrishnan, C., Rinn, B., & Rudolf, F. (2016). openBIS ELN-LIMS: an open-source database for academic laboratories. *Bioinformatics*, *32*(4). https://doi.org/10.1093/bioinformatics/btv606

Bloemers, M., & Montesanti, A. (2020). The FAIR Funding Model: Providing a Framework for Research Funders to Drive the Transition toward FAIR Data Management and Stewardship Practices. *Data Intelligence*, *2*(1–2). https://doi.org/10.1162/dint_a_00039

Bustin, S. A., Benes, V., Garson, J. A., Hellemans, J., Huggett, J., Kubista, M., Mueller, R., Nolan, T., Pfaffl, M. W., Shipley, G. L., Vandesompele, J., & Wittwer, C. T. (2009). The MIQE Guidelines: Minimum Information for Publication of Quantitative Real-Time PCR Experiments. *Clinical Chemistry*, *55*(4). https://doi.org/10.1373/clinchem.2008.112797

González-Beltrán, A., Neumann, S., Maguire, E., Sansone, S.-A., & Rocca-Serra, P. (2014). The Risa R/Bioconductor package: integrative data analysis from experimental metadata and back again. *BMC Bioinformatics*, *15*(S1). https://doi.org/10.1186/1471-2105-15-S1-S11

Johnson, D., Batista, D., Cochrane, K., Davey, R. P., Etuk, A., Gonzalez-Beltran, A., Haug, K., Izzo, M., Larralde, M., Lawson, T. N., Minotto, A., Moreno, P., Nainala, V. C., O'Donovan, C., Pireddu, L., Roger, P., Shaw, F., Steinbeck, C., Weber, R. J. M., … Rocca-Serra, P. (2021). ISA API: An open platform for interoperable life science experimental metadata. *GigaScience*, *10*(9). https://doi.org/10.1093/gigascience/giab060

Kanza, S., Willoughby, C., Gibbins, N., Whitby, R., Frey, J. G., Erjavec, J., Zupančič, K., Hren, M., & Kovač, K. (2017). Electronic lab notebooks: can they replace paper? *Journal of Cheminformatics*, *9*(1). https://doi.org/10.1186/s13321-017-0221-3

Knudsen, T. B., & Daston, G. P. (2005). MIAME guidelines. *Reproductive Toxicology*, *19*(3). https://doi.org/10.1016/j.reprotox.2004.10.004

Kuhring, M., Nieminen, M., Kirwan, J., Beule, D., & Holtgrewe, M. (2019). AltamISA: a Python API for ISA-Tab files. *Journal of Open Source Software*, *4*(40). https://doi.org/10.21105/joss.01610

Leonelli, S., Davey, R. P., Arnaud, E., Parry, G., & Bastow, R. (2017). Data management and best practice for plant science. *Nature Plants*, *3*(6). https://doi.org/10.1038/nplants.2017.86

Martínez-Bartolomé, S., Binz, P.-A., & Albar, J. P. (2014). *The Minimal Information About a Proteomics Experiment (MIAPE) from the Proteomics Standards Initiative*. https://doi.org/10.1007/978-1-62703-631-3_53

McQuilton, P., Batista, D., Beyan, O., Granell, R., Coles, S., Izzo, M., Lister, A. L., Pergl, R., Rocca-Serra, P., Schaap, B., Shanahan, H., Thurston, M., & Sansone, S.-A. (2020). Helping the Consumers and Producers of Standards, Repositories and Policies to Enable FAIR Data. *Data Intelligence*, *2*(1–2). https://doi.org/10.1162/dint_a_00037

Miyakawa, T. (2020). No raw data, no science: another possible source of the reproducibility crisis. *Molecular Brain*, *13*(1). https://doi.org/10.1186/s13041-020-0552-2

Novère, N. le, Finney, A., Hucka, M., Bhalla, U. S., Campagne, F., Collado-Vides, J., Crampin, E. J., Halstead, M., Klipp, E., Mendes, P., Nielsen, P., Sauro, H., Shapiro, B., Snoep, J. L., Spence, H. D., & Wanner, B. L. (2005). Minimum information requested in the annotation of biochemical models (MIRIAM). *Nature Biotechnology*, *23*(12). https://doi.org/10.1038/nbt1156

Papoutsoglou, E. A., Faria, D., Arend, D., Arnaud, E., Athanasiadis, I. N., Chaves, I., Coppens, F., Cornut, G., Costa, B. v., Ćwiek-Kupczyńska, H., Droesbeke, B., Finkers, R., Gruden, K., Junker, A., King, G. J., Krajewski, P., Lange, M., Laporte, M.,

Michotey, C., … Pommier, C. (2020). Enabling reusability of plant phenomic datasets with MIAPPE 1.1. *New Phytologist*, *227*(1). https://doi.org/10.1111/nph.16544

Petek, M., Coll, A., Ferenc, R., Razinger, J., & Gruden, K. (2020). Validating the Potential of Double-Stranded RNA Targeting Colorado Potato Beetle Mesh Gene in Laboratory and Field Trials. *Frontiers in Plant Science*, *11*. https://doi.org/10.3389/fpls.2020.01250

Petek, M., Zagorščak, M., Ramšak, Ž., Sanders, S., Tomaž, Š., Tseng, E., Zouine, M., Coll, A., & Gruden, K. (2020). Cultivar-specific transcriptome and pan-transcriptome reconstruction of tetraploid potato. *Scientific Data*, *7*(1). https://doi.org/10.1038/s41597-020-00581-4

Rocca-Serra, P., Brandizi, M., Maguire, E., Sklyar, N., Taylor, C., Begley, K., Field, D., Harris, S., Hide, W., Hofmann, O., Neumann, S., Sterk, P., Tong, W., & Sansone, S.-A. (2010). ISA software suite: supporting standards-compliant experimental annotation and enabling curation at the community level. *Bioinformatics*, *26*(18). https://doi.org/10.1093/bioinformatics/btq415

Sansone, S.-A., McQuilton, P., Rocca-Serra, P., Gonzalez-Beltran, A., Izzo, M., Lister, A. L., & Thurston, M. (2019). FAIRsharing as a community approach to standards, repositories and policies. *Nature Biotechnology*, *37*(4). https://doi.org/10.1038/s41587-019-0080-8

Sansone, S.-A., Rocca-Serra, P., Field, D., Maguire, E., Taylor, C., Hofmann, O., Fang, H., Neumann, S., Tong, W., Amaral-Zettler, L., Begley, K., Booth, T., Bougueleret, L., Burns, G., Chapman, B., Clark, T., Coleman, L.-A., Copeland, J., Das, S., … Hide, W. (2012). Toward interoperable bioscience data. *Nature Genetics*, *44*(2). https://doi.org/10.1038/ng.1054

Shaw, F., Etuk, A., Minotto, A., Gonzalez-Beltran, A., Johnson, D., Rocca-Serra, P., Laporte, M.-A., Arnaud, E., Devare, M., Kersey, P., Sansone, S.-A., & Davey, R. P. (2020). COPO: a metadata platform for brokering FAIR data in the life sciences. *F1000Research*, *9*. https://doi.org/10.12688/f1000research.23889.1

Waltemath, D., Adams, R., Beard, D. A., Bergmann, F. T., Bhalla, U. S., Britten, R., Chelliah, V., Cooling, M. T., Cooper, J., Crampin, E. J., Garny, A., Hoops, S., Hucka, M., Hunter, P., Klipp, E., Laibe, C., Miller, A. K., Moraru, I., Nickerson, D., … le Novère, N. (2011). Minimum Information About a Simulation Experiment (MIASE). *PLoS Computational Biology*, *7*(4). https://doi.org/10.1371/journal.pcbi.1001122

Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., … Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, *3*(1). https://doi.org/10.1038/sdata.2016.18

Wolstencroft, K., Krebs, O., Snoep, J. L., Stanford, N. J., Bacall, F., Golebiewski, M., Kuzyakiv, R., Nguyen, Q., Owen, S., Soiland-Reyes, S., Straszewski, J., van Niekerk, D. D., Williams, A. R., Malmström, L., Rinn, B., Müller, W., & Goble, C. (2017). FAIRDOMHub: a repository and collaboration environment for sharing systems biology research. *Nucleic Acids Research*, *45*(D1). https://doi.org/10.1093/nar/gkw1032