Cat4KIT: A Cross-institutional Data Catalog Framework for the FAIRification of Environmental Research Data

Mostafa Hadizadeh[1], Christof Lorenz[1], Sabine Barthlott[1], Romy Fösig[1], Uğur Çayoğlu[2], Robert Ulrich[3], Felix Bach[4]

[1]Institute of Meteorology and Climate Research, Karlsruhe Institute of Technology;
[2]Steinbuch Centre for Computing, Karlsruhe Institute of Technology;
[3]KIT Library, Karlsruhe Institute of Technology;
[4]Leibniz Institute for Information Infrastructure, FIZ Karlsruhe

A contemporary and flexible Research Data Management (RDM) framework is required to make environmental research data Findable, Accessible, Interoperable, and Reusable (FAIR) and, thus, provide the foundation for open and reproducible earth system sciences. While datasets accompanying scientific articles are typically published through large data repositories such as Pangaea, Zenodo, or RADAR4KIT, intermediate, day-to-day, or actively used data is often still exchanged through simple cloud storage services and email. However, despite the FAIR principles emphasizing the need for openly findable and accessible data, it is often confined to closed and restricted infrastructures and local file systems.

Therefore, our research project, Cat4KIT, aims to develop a cross-institutional catalog and RDM framework to FAIRify such day-to-day research data. The framework consists of four modules with different tasks: (1) providing access to data on storage systems through well-defined and standardized interfaces, (2) harvesting and transforming (meta)data into consistent and standardized formats, (3) making (meta)data publicly accessible using well-defined and standardized catalog services and interfaces, and (4) enabling users to search, filter, and explore data from decentralized research data infrastructures. Each module is developed and implemented within an inter-institutional consortium comprising scientists, software developers, and potential end-users. This approach ensures that our framework is applicable to a wide range of research data, from multi-dimensional climate model outputs to high-frequency in-situ measurements.

We place emphasis on the application of existing open-source solutions and community standards for data interfaces, (meta)data schemes, and catalog services such as the Spatio-Temporal Assets Catalog (STAC). This approach ensures easy integration of research data

Publiziert in: Vincent Heuveline, Nina Bisheh und Philipp Kling (Hg.): E-Science-Tage 2023. Empower Your Research – Preserve Your Data. Heidelberg: heiBOOKS, 2023. DOI: https://doi.org/10.11588/heibooks.1288.c18072

into the Cat4KIT framework and facilitates straightforward extension to other research data infrastructures.

1 Introduction

Nowadays, numerous real-world applications generate voluminous quantities of precise and ambiguous data from a broad variety of rich data sources at a rapid rate, and utilizing big data heralds the beginning of a new era of rising production (Hampton et al. 2013). This holds particularly true for the environmental sciences, where advances in modeling, *in situ* observation, and remote sensing systems, as well as the rapid growth of applications in the field of citizen science, have led to a massive increase in the number and volume of environmental data (e.g., Buytaert et al. 2012). And collaborative projects with partners spread across the world, as well as the increasing attention to such information from non-scientific communities, require this data to be remotely available and accessible via standardized and well-communicated interfaces. All this is aggravated by the release of the FAIR-principles (Wilkinson et al. 2016), which are becoming more and more mandatory in research projects or publications. In particular, the enrichment of data with consistent and well-defined metadata and unique indexes, as well as the public provision of this information via standardized interfaces, adds another level of complexity to modern research data management (RDM).

While there are initiatives that aim at the *FAIRification* of research data (e.g., Jacobsen et al. 2020; Kersloot et al. 2022), the usual way is still to publish data via dedicated data repositories. However, despite the need for such open and freely accessible data as well as the recognition that we urgently need to develop concepts for rewarding the publication of data (e.g., Pierce et al. 2019), this task is still assumed to be an additional (and often cumbersome) step at the end of a study or research project. And even if data is made publicly available, its exploitation is often rather limited: it is described and presented with properties that are relevant for domain experts (data producers) but that are not properly understood and reusable by other scientific communities (Annane et al. 2022) or the metadata is limited to a generic minimum description without crucial information and guidelines for proper usage of the underlying data. Furthermore, most modern domain-specific repositories like, e.g., PANGAEA[1] (Diepenbroek et al. 2002) or the World Data Center for Climate[2] (WDCC) at the German Climate Computing Center (DKRZ) or their generic counterparts like, e.g., ZENODO (European Organization For Nuclear Research and OpenAIRE 2013) or RADAR4KIT[3], only allow for the download of full datasets instead of allowing users to interact with the data, e.g., for subsetting or visualization. On the contrary, currently, it is common practice to transfer intermediate, day-to-day, or regularly accessed data through readily available cloud storage services and email, rather than utilizing a dedicated data portal or metadata service for the purpose

1 `http://www.pangaea.de`
2 `https://www.wdc-climate.de`
3 `https://radar.kit.edu`

of sharing, filtering, and exploring data from the Institute of Meteorology and Climate Research (IMK) at the Karlsruhe Institute of Technology (KIT).

Environmental scientists and data producers are hence facing a severe challenge: while the need for FAIR and collaborative research data is increasing, there are only limited frameworks and tools that help to make particularly intermediate and day-to-day data openly available and accessible according to the FAIR principles.

Within the research project Cat4KIT, we hence want to develop a catalog framework, that allows for a simple and straightforward provision of environmental research data. This is achieved by the development and implementation of four independent but interlinked modules:

- Dedicated data services take research data from existing storage systems at the KIT and make this data remotely accessible via standard interfaces
- A (meta)data harvester that involves systematically scanning data services and collecting metadata attributes in a standardized manner
- A dedicated catalog service allows for the interaction (search, filter, and modify) of the collected metadata
- A portal service aims at the user-friendly presentation of the collected metadata

An overview of these different modules and how they are interlinked is presented in Figure 1.

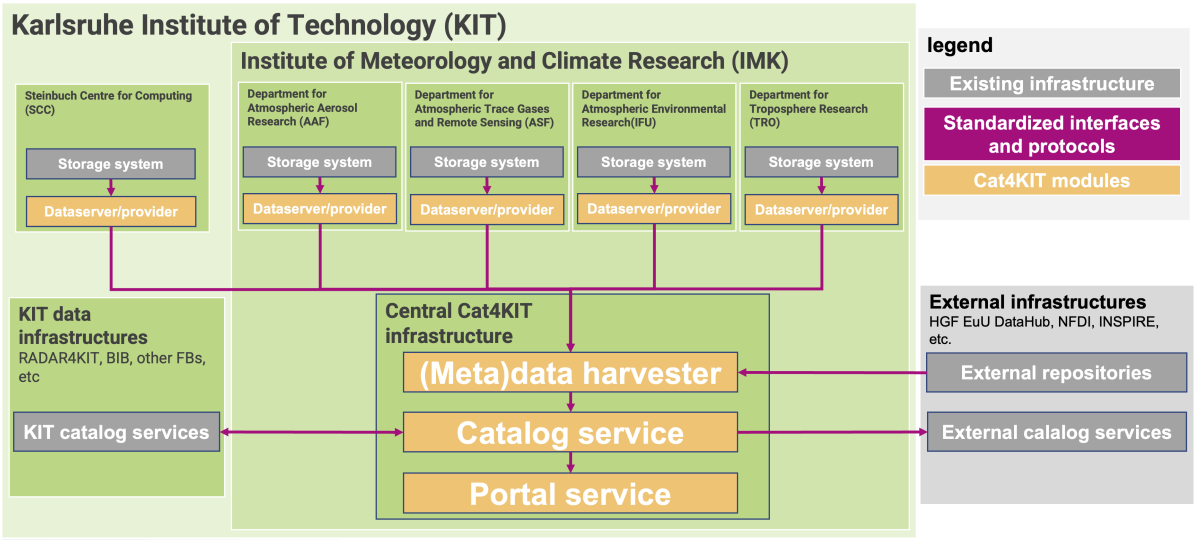

Figure 1: Schematic representation of the components in Cat4KIT project.

Funded by the *Exzellenzuniversitätsvorhaben Forschungsdatenmanagement* of the Karlsruhe Institute of Technology (KIT), The collaborative endeavor encompasses the cooperation of four KITs departments within the Institute of Meteorology and Climate Research, the Steinbuch Centre for Computing, and the KIT Library. In order to keep the entry

barrier as low as possible, Cat4KIT is hence build upon existing storage systems and infrastructure components at the participating institutes so that data procurers can use our framework without changing their established workflows. We further focus on widely used interfaces and community standards to ensure a simple and straightforward linkage to other catalog infrastructures, both within the KIT and with external repositories and catalog services (see Figure 1).

2 Components of the Cat4KIT framework

Our Cat4KIT-framework is based on a software stack that consists of existing open-source tools as well as complementing in-house developments, particularly for harvesting and ingesting the metadata from the different data sources. Each of these modules will be based on one (or multiple) interlinked Docker-containers which simplify the implementation of (sub)modules of Cat4KIT in other infrastructures. In the following, we will discuss each of the modules and tools in more detail.

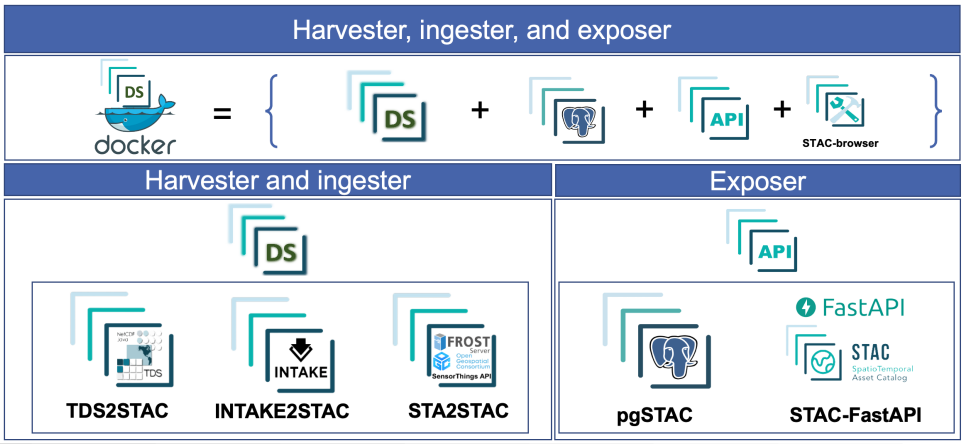

Figure 2: Components of the harvester, ingester, and exposer module.

2.1 Data service/provider

The first task of Cat4KIT is to make data on common and existing storage systems externally available. We currently focus on storage systems that are heavily used by the participating institutes and that is particularly the Large Scale Data Facility (LSDF) at KITs Steinbuch Centre for Computing as well as the BeeGFS (ThinkParQ GmbH 2023) and S3-compliant object storage implementations at Department for Atmospheric Environmental Research at Institute of Meteorology and Climate Research (IMK-IFU). In doing so, we want to ensure that a) we include a wide range of existing data into our Cat4KIT framework and b) users can easily integrate their data without changing their existing workflows.

However, as data from environmental sciences is usually highly heterogeneous, we need to apply dedicated services for different data types. In Cat4KIT, we hence distinguish

between multi-dimensional (e.g., from remote sensing or modeling systems) and one-dimensional data (e.g., from environmental sensor systems). For most of our multi-dimensional data, we apply a THREDDS-Data-Server (TDS; Domenico et al. 2002) while selected (high-volume) datasets (e.g., from climate models or high-resolution remote sensing systems) are also made available via so-called Intake-Catalogues[4]. Access to and interaction with one-dimensional data is realized via the Open Geospatial Consortium (OGC) SensorThings API (Liang, Huang, and Khalafbeigi 2016), which is provided by the so-called FROST Dataserver[5].

THREDDS Developed by the unidata community[6], THREDDS is a tailor-made data server for publishing (Network Common Data Format (NetCDF) data via various interfaces. As NetCDF is the quasi-standard many domains of environmental sciences, THREDDS-Server is widely used in the community for providing access to research data. Right now, we include data and catalogs from two TDS-instances at IMK-IFU[7] and Department for Atmospheric Trace Gases and Remote Sensing at Institute of Meteorology and Climate Research(IMK-ASF)[8]. But it is planned to implement further THREDDS servers both within but also outside of KIT.

Within TDS, data is organized in *catalogues*. These catalogs support both the static linking of datasets and their dynamic creation via so-called *DatasetScans*. TDS also features a wide range of interfaces with which users can interact with the data. Here, we only present some of the interfaces that are relevant within the Cat4KIT-framework:

- The Open-source Project for a Network Data Access Protocol (OPeNDAP)[9] allows for simple remote access to the data in the NetCDF-files. There are various libraries for most programming languages and environments that support data access via OpenDAP. Hence, this interface is used more and more to realize, e.g., workflows with decentralized data storage.

- The Web Mapping Service[10] (WMS) from the OGC allows for the server-side visualization of data. Due to its long history and ease of use, it is the quasi-standard for the visualization of geospatial raster data.

- ncISO [11] allows for the construction of ISO 19115[12] conformal metadata from NetCDF-files. Being the metadata standard for various communities, ISO 19115 features a wide range of attributes and information that (usually) has to be collected manually. Via ncISO, however, this information can be extracted automat-

4 https://github.com/intake/intake
5 https://github.com/FraunhoferIOSB/FROST-Server
6 https://www.unidata.ucar.edu
7 https://thredds.imk-ifu.kit.edu/thredds/catalog.html
8 https://thredds.imk-asf.kit.edu
9 https://www.opendap.org
10 https://www.ogc.org/standard/wms
11 https://artifacts.unidata.ucar.edu/service/rest/repository/browse/unidata-all/EDS/nciso
12 https://www.iso.org/standard/53798.html

ically from the NetCDFs which greatly simplifies the construction of standardized metadata.

Despite all these features, TDS currently does not include catalog services like, e.g., OGCs Catalog Service for the Web (CSW) or OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting). Hence, if we want to implement searching or filtering capabilities, we need to harvest the (meta)data from our THREDDS-servers into dedicated catalog frameworks.

Intake Intake is a lightweight Python package that allows to load data from a variety of formats and sources into well-known containers such as Pandas dataframes (The pandas development team 2020) or Xarray DataSets (Hoyer and Hamman 2017), etc. Data is usually collected in catalogs and each item within such a catalog contains all information that is required to interact with the data. It hence removes the need for a potential end-user to know about the exact storage location/system or data format. Moreover, Intake allows for the generation of simple catalogs with data of different formats and across different institutions. It hence can be a crucial element in the development of reproducible workflows with data on decentralized storage systems. In Cat4KIT, we apply such Intake catalogs manly for high-volume model and remote sensing data that typically lies on cloud-optimised storage systems like, e.g., S3-conformal object storage.

FROST The Fraunhofer Open Source SensorThings API Server (FROST) is the reference implementation of OGCs SensorThings API (STA). With its roots in the Internet of Things(IoT), STA is tailor-made for one-dimensional data from typical sensor systems. At its core, the STA consists of an interface (a set of commands to interact with the (meta)data) and a dedicated datamodel. This datamodel is constructed around *Things*, which could be, according to STA, "anything in the physical or information world that can be uniquely identified and integrated into communication networks". It further provides entities for *Location* (directly attached to a *Thing*), *Sensor*, and *ObservedProperty*. When combining a *Thing* with a *Sensor* and an *Observed Property*, we obtain a so-called *DataStream*, which defines the container into which data is written. In Cat4KIT, we apply FROST-Servers mainly for providing access to one-dimenisonal data from various observation and monitoring systems that are operated at the IMKs. But as the implementation of FROST-Servers and the SensorThings API has just started, we only include data from the instance at IMK-IFU[13], that, so far, only holds some preliminary data.

2.2 Metadata catalog framework

As our catalog framework, we apply the SpatioTemporal Assets Catalog (STAC)[14], that has its origin in the collaboration of satellite imagery providers. In the last years, the user base and community around STAC has increased substantially and it is now applied by a wide range of data providers as their main catalog framework. There has also grown a

13 `https://sensorthings.imk-ifu.kit.edu`
14 `https://stacspec.org`

large ecosystem[15] around STAC with extensions and plugins for all kinds of applications ranging from visualization solutions for geospatial data over dedicated databases for STAC to client libraries for major programming languages. Today, STAC describes itself as "common language to describe geospatial information, so it can more easily be worked with, indexed, and discovered".[16]. In that sense, it is a tailormade framework for realizing a catalog and data exploration infrastructure with a minimal set of mandatory metadata. Because at its core, STAC is build around Items that are simple GeoJSONs with all necessary information about a dataset. In general, a STAC Item only requires a title, some information about the bounding box as well as some temporal information. And this information can easily be retrieved from all data servers within Cat4KIT (see section 2.1) so that we can easily generate a consistent set of STAC Items. Multiple STAC Items, that share properties and metadata, can be combined in STAC Collections. And finally, a STAC Catalog is constructed of one or multiple Collections and (Sub)Catalogs.

As many of the datasets included in Cat4KIT are typical raster datasets with multiple variables, we further make use of the DataCube[17] extension. This extension allows to automatically retrieve and add information about variables and dimensions within a dataset. This, again, reduces the need for manual metadata curation of the final STAC Items.

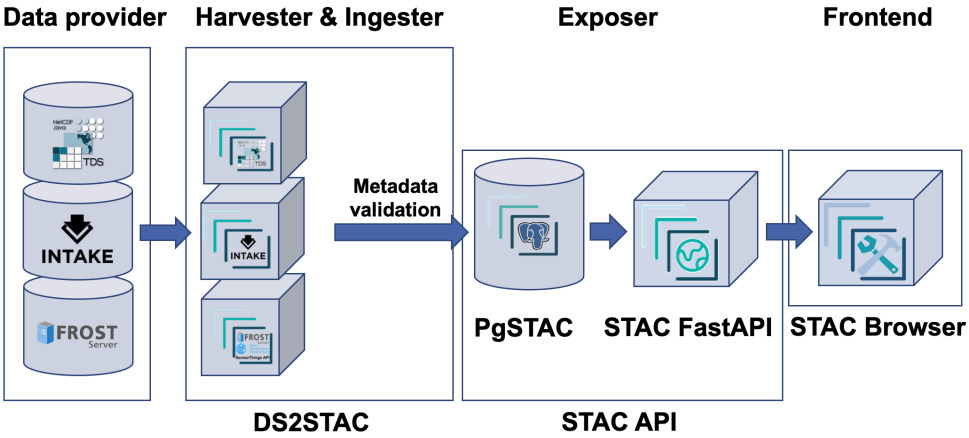

Figure 3: Workflow of the Cat4KIT system.

2.3 Harvester, ingester and exposer

Once the data is available via different data server (see section 2.1, the next step is to retrieve or *harvest* the respective metadata and ingest it into a consistent STAC database. This step is done by the *harvester/ingester* while the *exposer* is responsible for making the catalog data accessible from the outside (see Figure 2).

15 `https://stacindex.org/ecosystem`
16 `https://stacspec.org/en`
17 `https://github.com/stac-extensions/datacube`

For the harvester and ingester, we have developed the so-called *DS2STAC*[18] (Data Servers/-Services to STAC metadata catalog) package. DS2STAC includes three tailored subpackages for scanning and harvesting datasets. Each of these packages retrieves the geospatial and temporal information that is necessary for creating STAC items:

- TDS2STAC[19] (harvester for THREDDS Server): STAC Items via THREDDS datasets, coordinates, variables and temporal information via ncISO and WMS (depending on availability)
- INTAKE2STAC[20] (harvesrer for Intake catalogs): STAC Items via Intake catalog entries, coordinates, variables and temporal information via STAC DataCube extension and Xarray
- STA2STAC[21] (harvester for OGC SensorThings API): STAC Items via *Things*, coordinates via *Location*-entities, temporal information via *pheonomenonTime* in the *Datastream*-entities

In its current implementation, DS2STAC is run manually but it is planned to add a scheduler so that we can harvest from each data source, e.g., at pre-defined times or when new data has been added or modified. Once new metadata is retrieved, it is first passed through a *Validator*[22] that checks the data for the availability of all mandatory elements (temporal and spatial coordinates, general description, description of available data services). If this validation is passed, the new entry is moved forward to the *Exposer*-module (see below). If not, it is envisaged that the data provider receives a notification with details about missing or erroneous elements.

For TDS2STAC, the generation of STAC Collections and Catalogs (see section 2.2) is triggered automatically and resembles the structure of the underlying THREDDS catalogues. For STA2STAC and INTAKE2STAC, this is still a manual process that needs some further refinements and hierarchies directly at the data source.

In the next step, we are going to develop tailored harvesters for other data sources and repositories. Furthermore, the harvester for the SensorThings API is still work in progress, as we are currently developing specific STA metadata profiles and data models for environmental sciences. As these profiles will also feature new entities particularly for a more detailed grouping of STA-*Things* (e.g., via projects or networks), they will allow for, e.g, the direct harvesting and mapping into STAC Collections and Catalogs.

The *exposer* is based on the STAC API[23], which is implemented via the STAC FastAPI[24] with the *pgSTAC*[25] storage backend. We further plan to implement data within our

18 https://codebase.helmholtz.cloud/cat4kit/ds2stac
19 https://tds2stac.readthedocs.io
20 https://intake2stac.readthedocs.io
21 https://sta2stac.readthedocs.io
22 https://stac-validator.readthedocs.io
23 https://github.com/radiantearth/stac-api-spec
24 https://github.com/stac-utils/stac-fastapi
25 https://github.com/stac-utils/pgstac

Cat4KIT infrastructure into other higher-level repositories and portals. Hence, as a demonstrator, we currently develop a dedicated STAC exposer for the so-called *Earth Data Portal*[26] of the Helmholtz Research Field Earth and Environment[27]. The STAC API furhter allows, due to the growing ecosystem and user basis of STAC, to seamlessly integrate our Cat4KIT catalogues into other third-party software like QGIS[28].

2.4 Data portal and frontend

Finally, once metadata is available via the *exposer*, it is presented in a data portal and graphical user interface that based on the STAC-Browser[29]. Here, a common site for a STAC Item allows to present information about the geographic location on a map, the temporal span as well as further available metadata of the underlying dataset. As the latest version of the STAC-Browser is fully consistent with the STAC API, it further allows for a direct searching and filtering of the catalogs.

3 Working with Cat4KIT - a first concept

One of the key objectives of our Cat4KIT framework is to make the integration of data as simple as possible. Thus, data producers should be able to stick to their established workflows while taking advantage from the external accessibility and interactivity. A typical algorithm using Cat4KIT, hence, should look like

1. User/data provider stores data on an integrated storage system (raster data as CF-conformal NetCDF, one-dimensional data in database with STA interface)
2. Only for THREDDS and Intake: Cat4KIT-admin adds dataset to THREDDS- or Intake-catalogues
3. Automatic harvesting of metadata (esp. spatial and temporal coordinates as well as data services) from newly created entries
4. Automatic generation of STAC items and integration into STAC database
5. Automatic exposing of newly generated STAC items via STAC API
6. Presentation of harvested information, dimensions and variables as well as available data services in STAC Browser

Users can now find/search/explore the newly integrated data in the STAC Browser. The respective description also includes links to available data services like WMS, OpeNDAP, or STA and hence allow for direct and tailored access to the data, e.g., for visualization, extraction or analysis.

26 `https://earth-data.de`
27 `https://www.helmholtz.de/en/research/research-fields/earth-and-environment`
28 `https://stac-utils.github.io/qgis-stac-plugin`
29 `https://github.com/radiantearth/stac-browser`

4 Conclusions and Outlook

In this paper, we present our current concept for a catalog framework, that should help researchers and data providers to make their data FAIR with a focus on findability and accessibility. Once fully operational, it will allow for an easy integration of (meta)data by automatic harvesting from several data sources. By that, there is no need to change existing and established workflows; instead, data providers will benefit from the added accessibility and interactivity via standard interfaces and data services as well as the straightforward integration into higher-level data and catalog infrastructures via the STAC API. This is further supported by the heavy usage of open-source tools and interfaces that are widely applied in the scientific community. In particular, if data is available via one of the (currently) three supported data sources (THREDDS, Intake, SensorThings API), an integration is straightforward and does not need any further manual configuration.

One key aspect of Cat4KIT is the integration of (meta)data from highly decentralized storage and infrastructure systems within the four departments of the Institute of Meteorology and Climate Research at the Karlsruhe Institute of Technology, as well as the Steinbuch Centre for Computing and the KIT Library. This should also allow for a simple and straightforward integration of other internal and external repositories and data sources. In order to timely consider this integration, we are already in contact with interested parties and potential users from external institutions.

Currently, we are in the testing-phase of the DS2STAC-module, which contains tailored harvesters for each of the three data sources. While the harvesting, in general, is working as supposed, particularly the division into STAC Catalogs and Collections still requires some further refinements and conventions (like, e.g., the concrete meaning of Collections and Catalogs for different use cases). It is important to acknowledge that following the launch of the initial version, the users' perspectives on infrastructure improvements will be gathered through the feedback system in Cat4KIT infrastructure. Subsequently, these perspectives will be reviewed and incorporated into future developments.

Overall, it is planned that a first running version of the Cat4KIT-framework is available from Q4 2023. During the coming months, we will hence focus on the linkage of the different modules, the implementation of further data sources as well as continuously enhancing the number of integrated datasets.

After its launch, Cat4KIT will be the central entry point for searching, filtering and exploring data from the Institute of Meteorology and Climate Research at the KIT. It will hence provide a substantial contribution towards the FAIRification of research data and further foster an open research from environmental sciences.

Acknowledgements

The Cat4KIT project has been funded by the *Exzellenzuniversitäts-Vorhaben Research Data Management* of the Karlsruhe Institute of Technology, Germany. We further thank

our colleagues Dr. Philipp Sebastian Sommer and Linda Baldewein from the Helmholtz-Zentrum Hereon for many inspiring and fruitful discussions about our framework. Additionally, we would like to express our appreciation to the anonymous reviewers for their constructive feedback, which has greatly improved the quality of this paper.

References

Annane, Amina, Mouna Kamel, Cassia Trojahn, Nathalie Aussenac-Gilles, Catherine Comparot, and Christophe Baehr. 2022. "Towards the FAIRification of Meteorological Data: A Meteorological Semantic Model", 81–93. DOI: https://doi.org/10.1007/978-3-030-98876-0_7.

Buytaert, Wouter, Selene Baez, Macarena Bustamante, and Art Dewulf. 2012. "Web-Based Environmental Simulation: Bridging the Gap between Scientific Modeling and Decision-Making". *Environmental Science & Technology* 46 (4): 1971–1976. ISSN: 0013-936X. DOI: https://doi.org/10.1021/es2031278.

Diepenbroek, Michael, Hannes Grobe, Manfred Reinke, Uwe Schindler, Reiner Schlitzer, Rainer Sieger, and Gerold Wefer. 2002. "PANGAEA—an information system for environmental sciences". *Computers & Geosciences* 28 (10): 1201–1210. DOI: https://doi.org/10.1016/S0098-3004(02)00039-0.

Domenico, Ben, John Caron, Ethan Davis, Robb Kambic, and Stefano Nativi. 2002. "Thematic real-time environmental distributed data services (THREDDS): Incorporating interactive analysis tools into NSDL". Accessed: May 19, 2023. https://docs.unidata.ucar.edu/tds/5.4/userguide/index.html.

European Organization For Nuclear Research and OpenAIRE. 2013. *Zenodo.* DOI: https://doi.org/10.25495/7GXK-RD71.

Hampton, Stephanie E., Carly A. Strasser, Joshua J. Tewksbury, Wendy K. Gram, Amber E. Budden, Archer L. Batcheller, Clifford S. Duke, and John H. Porter. 2013. "Big data and the future of ecology". *Frontiers in Ecology and the Environment* 11 (3): 156–162. ISSN: 1540-9295. DOI: https://doi.org/10.1890/120103.

Hoyer, Stephan, and Joe Hamman. 2017. "xarray: N-D labeled arrays and datasets in Python". *Journal of Open Research Software* 5 (1). DOI: https://doi.org/10.5334/jors.148.

Jacobsen, Annika, Rajaram Kaliyaperumal, Luiz Olavo Bonino da Silva Santos, Barend Mons, Erik Schultes, Marco Roos, and Mark Thompson. 2020. "A Generic Workflow for the Data FAIRification Process". *Data Intelligence* 2 (1-2): 56–65. ISSN: 2641-435X. DOI: https://doi.org/10.1162/dint_a_00028.

Kersloot, Martijn G., Ameen Abu-Hanna, Ronald Cornet, and Derk L. Arts. 2022. "Perceptions and behavior of clinical researchers and research support staff regarding data FAIRification". *Scientific Data* 9 (1): 241. ISSN: 2052-4463. DOI: https://doi.org/10.1038/s41597-022-01325-2.

Liang, Steve, Chih-Yuan Huang, and Tania Khalafbeigi. 2016. "OGC SensorThings API Part 1: Sensing, Version 1.0". Accessed: May 19, 2023. https://docs.ogc.org/is/18-088/18-088.html.

Pierce, Heather H., Anurupa Dev, Emily Statham, and Barbara E. Bierer. 2019. "Credit data generators for data reuse". *Nature* 570 (7759): 30–32. ISSN: 0028-0836. DOI: https://doi.org/10.1038/d41586-019-01715-4.

The pandas development team. 2020. *pandas-dev/pandas: Pandas.* Version 2.1.0. DOI: https://doi.org/10.5281/zenodo.3509134.

ThinkParQ GmbH. 2023. "BeeGFS". Visited on August 23, 2023. https://www.beegfs.io/c/.

Wilkinson, Mark D., Michel Dumontier, IJsbrand Jan Aalbersbergand, Gabrielle Appleton, Myles Axtonand, Arie Baakand, Niklas Blombergand, et al. 2016. "The FAIR Guiding Principles for scientific data management and stewardship". *Scientific data* 3 (1): 1–9. DOI: https://doi.org/10.1038/sdata.2016.18.