FAIRICUBE –
F.A.I.R. INFORMATION CUBES

Project Number: 101059238

WP4 Share

D4.6 Validation of sharing

Deliverable Lead: 4SF
Deliverable due date: 30/06/2024

Version: 1.0
28/06/2024

Document Control Page

Document Control Page	
Title	D4.6 Validation of sharing
Creator	4SF
Description	D4.6 Validation of sharing
Publisher	"FAIRiCUBE – F.A.I.R. information cubes" Consortium
Contributors	4sf, EPSILON, NILU
Date of delivery	30-06-2024
Type	R — Document, report
Language	EN-GB
Rights	Copyright "FAIRICUBE – F.A.I.R. information cubes"
Audience	☒ Public ☐ Confidential ☐ Classified
Status	☐ In Progress ☐ For Review ☒ For Approval ☐ Approved

Revision History			
Version	Date	Modified by	Comments
0.1	27/05/2024	Kathi Schleidt	Setup of document
0.2	04/06/2024	Rob Knapen	Initial review
0.3	18/06/2024	Jaume Targa and Lorena Banyuls	
0.4	20/06/2024	María Colina	Further input on document
0.5	25/06/2024	Jaume Targa	Finalisation of document. Ready for review
1.0	26/06/2024	Maria Ricci	Internal review and format checking

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

Disclaimer

This document is issued within the frame and for the purpose of the FAIRICUBE project. This project has received funding from the European Union’s Horizon research and innovation programme under grant agreement No. 101059238. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

This document and its content are the property of the FAIRICUBE Consortium. All rights relevant to this document are determined by the applicable laws. Access to this document does not grant any right or license on the document or its contents. This document or its contents are not to be used or treated in any manner inconsistent with the rights or interests of the FAIRICUBE Consortium or the Partners detriment and are not to be disclosed externally without prior written consent from the FAIRICUBE Partners. Each FAIRICUBE Partner may use this document in conformity with the FAIRICUBE Consortium Grant Agreement provisions.

Table of Contents

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

List of Tables

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

1 Introduction

This document, "Validation of Sharing", serves as a validation guide especially for deliverance from work package 4 (WP4 "share") where the short working title share refers to the shared infrastructure components of FAIRiCUBE. The validation guide that is presented in the following will therefore highlight the essential components of FAIRiCUBE Hub that require validation. As the Hub encompasses a wide array of services dedicated to data access, processing, and analysis, this document aims to delineate the specific aspects and functionalities of FAIRiCUBE Hub that require validation to meet the requirements of diverse Use Cases.

Key areas for validation within the scope of the "Validation of Sharing" document encompass the core functionality of FAIRiCUBE Hub, ensuring alignment with the FAIR principles of data management. This includes verifying the comprehensive documentation and knowledge base to support users in navigating and use FAIRiCUBE Hub effectively. Additionally, validation will focus on the robustness of data management practices, specifically examining metadata standards and the efficacy of the data catalog. Ensuring seamless accessibility through protocols such as WCS (Web Co
verage Service), S3, Sentinel Hub, and the Rasdaman platform will be critical. Moreover, the document includes the validation criteria on the processing capabilities of FAIRiCUBE Hub, assessing the functionality and integration of tools like the lab environment, GitHub repositories, and the WCPS (Web Coverage Processing Service) for efficient data processing and analysis.

By documenting these validation requirements, stakeholders and users of FAIRiCUBE Hub can gain clarity on how the platform ensures compliance with FAIR principles.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

2 Core

2.1 FAIRiCUBE Hub

The FAIRiCUBE Hub's approach to sharing promotes efficient and effective collaboration among users by ensuring that components are findable, accessible, interoperable, and reusable. This approach maximizes the utility and impact of the shared data, tools, and knowledge within the community.

The validation criteria include:

- Findability of components
- Authentication across components
- Interplay across components
    - Can identified datasets be accessed?
    - Can identified analysis/processing (a/p) resources be accessed?
    - Can linked components (e.g., datasets and a/p resources provided under data provenance) be accessed?
    - Can processing outputs be stored and provided for external download?
    - Can processing outputs be visualized as required?
    - Is use of a/p resources on out-of-distribution data detected and/or blocked?
        - Is monitoring of data and concept drift implemented?
    - Is environmental resource use when running components reported?
    - Is there a mechanism for user support or a helpdesk service in place?

2.1.1 Findability of components

FAIRiCUBE Hub must ensure that users can easily locate all available components within the Hub and ensure that the Hub’s UI has intuitive navigation paths and clearly labelled sections. An up-to-date index of all components should be maintained. The FAIRiCUBE Hub should provide a robust search function that allows users to find components based on keywords, tags, or categories.

2.1.2 Authentication across components

The FAIRiCUBE Hub must be accessed via a secure authentication process across all components of the Hub. A secure authentication protocols (e.g., OAuth2, OpenID Connect) should be implemented, including a role-based access control to manage permissions across different components.

2.1.3 Interplay across components

FAIRiCUBE Hub must ensure smooth interaction and data flow between different components within the Hub.

Funded by the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

Key aspects to validate are:

- Provide direct access links or APIs for datasets, ensuring they are easily retrievable once identified.
- Ensure availability and accessibility of a/p resources via direct links or APIs.
- Ensure that all linked components, including those under data provenance, are accessible and that relationships between them are clear.
- Implement storage solutions for processing outputs and provide mechanisms for external download, such as secure links or APIs.
- Integrate visualization tools that allow users to view processing outputs directly within the Hub, supporting formats like graphs, maps, or custom visualizations.
- Implement checks to detect and prevent the use of analysis/processing resources on data that falls outside the expected distribution.
- Set up monitoring systems to detect data drift (changes in data distribution) and concept drift (changes in the underlying relationship between inputs and outputs).
- Track and report the usage of computational and environmental resources (e.g., CPU, memory, energy consumption) when components are run, providing users with insights into resource efficiency.

2.1.4 Mechanism for user support/helpdesk in place

The FAIRiCUBE Hub will provide users with adequate support to resolve issues and answer queries. FAIRiCUBE helpdesk should be implemented with ticketing and tracking capabilities.

Related to keeping documentation updated, a comprehensive knowledge base with FAQs, tutorials, and documentation should be maintained. The FAIRiCUBE managers must ensure timely and effective responses to user inquiries, potentially including live chat support.

3 Information

In this chapter, we cover the validation of information which had seen an uneasy split between these components *documentation* vs. *Knowledge Base (KB)* which is recently realigned so that documentation is more static content as we see it in read-the-docs while the KB has the capability to intelligently search over documentation as well as resources, data, etc. ... In a future update of this deliverable, the structure of the chapters will be revised and potentially merged.

3.1 Documentation

Effective documentation is essential for the successful use and understanding of FAIRiCUBE. The validation of documentation involves assessing various criteria to ensure that the provided information meets the users' needs. Here, we will evaluate the documentation of FAIRiCUBE based on five key criteria: findability of required information, correctness of available information, completeness of available information, usefulness of available information, and ease of provision of documentation.

The validation criteria for FAIRiCUBE documentation[1] include:

- Findability of required information
- Correctness of available information
- Completeness of available information
- Usefulness of available information
- Ease of provision of documentation

3.1.1 Findability of Required Information

Findability refers to how easily users can locate the information they need within the documentation. For FAIRiCUBE, this can be assessed by examining how the documentation is organised and structured, the presence of a search function, the applicability of search results and the use of clear headings and subheadings, and an understandable writing style. The documentation must be intuitively structured with a well-organised table of contents.

3.1.2 Correctness of Available Information

Current, accurate and reliable information are key for the usability of the documentation. The correctness of information (including hyperlinks) should be regularly checked. Documentation should be periodically reviewed and updated to reflect the latest features, bug fixes, and improvements.

[1] https://fairicube.readthedocs.io
https://fairicube.readthedocs.io/en/latest/guide/fic_Hub

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

3.1.3 Completeness of Available Information

FAIRiCUBE documentation must cover all necessary aspects including detailed guides, comprehensive code, thorough API references (when applicable), and explanations of key concepts. All documentation should have been reviewed by an external expert to ensure sections are not incomplete and/or missing.

3.1.4 Usefulness of Available Information

All documentation must provide practical value to FAIRiCUBE users. The information should not only be correct and complete but also relevant and actionable. Provided documentation must be clear, should include step-by-step instructions (when feasible) with practical examples. Moreover, when possible real examples from Use Cases should be added.

3.1.5 Ease of Provision of Documentation

All created FAIRiCUBE documentation is to be maintained, and updated, including tools' documentation, internal guidelines, and platforms. It must be easily possible to provide additions and corrections to the existing documentation.

In summary, the validation of FAIRiCUBE documentation involves ensuring that it is easily findable, correct, complete, useful, and easy to provide. By meeting these criteria, FAIRiCUBE can enhance user engagement, and stakeholder reproducibility. Regular reviews and updates, combined with user feedback, will help maintain the high quality and relevance of the documentation over time.

3.2 Knowledge Base

Knowledge base[2] (KB) is FAIRiCUBE centralised repository that stores information, data, and knowledge in the project. The validation criteria include:

- Findability of required information
- Correctness of available information
- Completeness of available information
- Usefulness of available information
- Ease of provision of information for the KB.

3.2.1 Findability of Required Information

Findability refers to how easily users can locate the information, within the KB. Effective search functionality is essential. A well-organised structure is also crucial, with logical categorisation, clear hierarchy, and consistent naming conventions, ensuring users can navigate the knowledge base intuitively. Additionally, comprehensive metadata and tagging improve search accuracy, making relevant information readily accessible.

[2] https://fairicube-kb.dev.epsilon-italia.it/

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

3.2.2 Correctness of Available Information

Correctness ensures that the information within the KB is accurate, reliable, and up to date. Mechanisms for error handling, such as user reporting systems, allow for prompt correction of inaccuracies.

3.2.3 Completeness of Available Information

Completeness measures whether the KB covers all relevant topics comprehensively. It is essential to address all areas in detail, providing thorough explanations, step-by-step guides, and practical examples. Cross-referencing related articles and external resources enhances the context and provides background information.

3.2.4 Usefulness of Available Information

Usefulness evaluates how beneficial the information is for users in solving their problems or gaining insights. Relevance is key, ensuring the content meets the specific needs and queries of the target audience. Regular updates keep the information current and pertinent. Practical examples of UC should be provided. User engagement is encouraged through interactive elements like comments, ratings, and feedback, making the knowledge base more dynamic and user centred.

3.2.5 Ease of Provision of Information for the KB

Ease of provision assesses how simple it is for contributors to add, update, and manage information in the knowledge base. Providing training and ongoing support for contributors ensures they are well-equipped to maintain the knowledge base.

To sum up, evaluating a knowledge base based on findability, correctness, completeness, usefulness, and ease of provision ensures it is not only comprehensive and accurate but also user-friendly and valuable. By focusing on these criteria, organisations can create and maintain a robust and effective knowledge base that enhances decision-making, productivity, and knowledge retention.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

4 Data

4.1 Metadata Editors

FAIRiCUBE should ensure that Metadata Editors are effective tools for sharing and validating metadata, thereby enhancing the overall quality and usability of shared data. By maintaining a robust and user-friendly interface, accurate data persistence, and clear error handling, the project can significantly improve user engagement and data management efficiency.

The validation criteria for FAIRiCUBE metadata[3] include:

- Content of GUI correctly persisted in JSON file, all values entered in GUI available in the JSON
- Alignment metadata schema (as described in the deliverables), Editor and JSON file
- Code lists findable and complete, sorted
- All labels in the GUI described
- Human-readable error messages.

4.1.1 Content of GUI correctly persisted in JSON file, all values entered in GUI available in the JSON

All values entered in the GUI should be accurately saved in a JSON file. This will ensure that any data input or selections made by the user through the GUI are accurately recorded and stored. This involves mapping every field in the GUI to a corresponding key in the JSON file. Automated tests can be created to validate the consistency between the GUI input and the JSON file output.

4.1.2 Alignment metadata schema (as described in the deliverables), Editor and JSON file

The structure and format of metadata in the JSON file should conform to the specified metadata schema, and the GUI editor should support this structure. This will ensure that the data is structured in a way that is consistent with the project's metadata schema, facilitating interoperability and data integration. This requires the GUI editor to enforce the metadata schema rules, such as data types, required fields, and value constraints, when users input data. The JSON file generated should adhere to the same schema. Compare the JSON output against the metadata schema to ensure conformity. This can be automated using schema validation tools to check for discrepancies.

[3] https://catalog-editor.eoxHub.fairicube.eu/
https://fairicube-md.dev.epsilon-italia.it/

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

4.1.3 Code lists findable and complete, sorted

FAIRiCUBE needs to ensure that the code lists (implemented as drop down lists) in the GUI should contain all necessary options, be easily navigable, and sorted in a logical order. This enhances the usability and efficiency of the GUI, ensuring users can easily find and select the appropriate options. Populate code lists with a comprehensive set of options relevant to the context. Sort options alphabetically or by relevance to make them easy to find. Review the code lists to ensure they contain all required options and are logically sorted. User feedback can be valuable in identifying any usability issues.

4.1.4 All labels in the GUI described

Every label and field in the GUI should have a clear and concise description, making the interface intuitive and self-explanatory. This improves user experience by providing clarity on what each field or option in the GUI represents, reducing the likelihood of errors. Add descriptive tooltips, help texts, or documentation that explains each label and field. Use clear and unambiguous language. Ensure that each label in the GUI has an accompanying description. Conduct usability testing to confirm that users understand the purpose of each field based on the provided descriptions.

4.1.5 Human-readable Error Messages

Error messages should be clear, concise, and provide actionable information to help users resolve issues. Helps users understand and correct errors without requiring technical support, improving the overall usability and user satisfaction. Design error messages that explain the problem in plain language and, if possible, suggest steps to resolve the issue. Avoid technical jargon that may confuse users. Review error messages to ensure they are user-friendly and informative. Perform testing by intentionally causing errors and evaluating if the messages are helpful and clear.

4.2 Data Catalog

FAIRiCUBE should make sure that the Data Catalog[4] is user-friendly and efficient in helping users find and access the datasets they need. By providing robust search functionalities and clear navigation options, the Data Catalog can significantly enhance user experience and data usability. Regular updates and feedback mechanisms can further refine these features to meet evolving user needs.

The validation criteria include:

- Findability of required data based on search criteria, e.g.:
  - Spatiotemporal bounding box
  - Title
  - Keywords

[4] https://catalog.eoxHub.fairicube.eu/
https://fairicube.rasdaman.com/rasdaman/ows#/services

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

- Simple search (Spatiotemporal bounding box, Title, Keywords...) and advanced search (all fields)
- Ability to navigate to the identified dataset (either by link to original data download, when possible; or API endpoint with related API documentation for data access, e.g., endpoint for data in an AWS S3 bucket, or Rasdaman datacube URI, and link to relevant documentation).

4.2.1 Findability of Required Data Based on Search Criteria

Ensure users can efficiently locate datasets using specific search criteria.

- Spatiotemporal bounding box: Users should be able to find datasets based on geographic location and time frame. Provide a search feature where users can input spatial coordinates (latitude, longitude) and temporal ranges (start date, end date) to filter datasets.
- Title: Users should be able to search for datasets by their titles. Include a title search field that allows users to enter keywords or phrases to find datasets with matching titles.
- Keywords: Users should be able to search for datasets using keywords. Provide a keyword search option where users can enter specific terms related to the data they are looking for.

4.2.2 Simple and Advanced Search

Offer both basic and comprehensive search capabilities to cater to different user needs.

- Simple Search: A straightforward search interface allowing users to quickly find datasets using basic criteria like Spatiotemporal Bounding box, Title, and Keywords. Design a user-friendly search bar where users can enter one or more of these basic search criteria to retrieve results.
- Advanced Search: A more detailed search interface that allows users to specify multiple criteria and fine-tune their search. Provide additional fields and options to refine searches, such as filtering by data format, provider, publication date, metadata fields, etc.

4.2.3 Ability to Navigate to the Identified Dataset

Ensure users can easily access the datasets they find through search.

- Link to Original Data Download: Direct users to the source where they can download the original dataset. Include a link in the search results or dataset details page that redirects users to the original data source.
- API Endpoint with Related API Documentation: Provide API endpoints for programmatic access to datasets, along with relevant documentation. List API endpoints in the dataset details and include comprehensive documentation on how to use these endpoints.
- Endpoint for Data in S3 Bucket or Rasdaman Datacube URI: Provide specific endpoints for accessing data stored in cloud services or specialized databases. Detail the endpoints for accessing datasets stored in services like AWS S3 or Rasdaman datacubes, including any necessary access credentials or instructions.

Funded by the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

4.3 Data Access WCS

FAIRiCUBE will ensure that data access via WCS is reliable, accurate, and adheres to relevant specifications and standards. Regular validation and compliance checks are essential to maintain the integrity and usability of the data services provided. The validation criteria include:

- Request returns correct responses in accordance with the WCS specification as well as underlying specifications.
  - Measure types from SWE Common Data Model Standard[5] used according to the original specification
  - XML responses valid
  - Extent/bounding box information aligned with GML
- Categorical data is provided with (access to) descriptive code list, vocabulary, etc.
- Data returned is for the requested subset of the original source data
- Data is returned in the requested format
- Data is returned clearly indicating the property the data represents
- Data is linked to relevant metadata

4.3.1 Request returns correct responses in accordance with the WCS Specification

FAIRiCUBE will verify that all requests to the WCS return responses that adhere to the WCS specification and related underlying specifications. All responses should conform to the standards set by the WCS specification, ensuring interoperability and reliability. Regularly test responses against the WCS specification to ensure they meet the required criteria.

- Measure Types from SWE Common Data Model Standard Used According to the Original Specification: Ensure that measure types defined in the Sensor Web Enablement (SWE) Common Data Model are used correctly as per their specifications. SWE Common defines standard measure types for various data values (e.g., quantities, counts, time). These should be implemented as specified. Cross-check measure types in the responses against the SWE Common specification to ensure accuracy and consistency.
- XML Responses Valid: Ensure that all XML responses from the WCS are valid according to XML schemas. Use XML validation tools to check that responses conform to the relevant XML schemas. Implement error handling to catch and correct invalid XML structures.
- Extent/Bounding box Information Aligned with GML: Ensure that extent and bounding box information in responses are correctly aligned with Geography Markup Language (GML) standards. GML defines how spatial data and geometries should be represented. Bounding box and extent information in WCS responses should follow these standards. Verify that spatial coordinates and geometries in responses are correctly formatted and align with GML.

[5] https://www.ogc.org/standard/swecommon/

4.3.2 Categorical Data is Provided with (Access to) Descriptive Code List, Vocabulary, etc.

FAIRiCUBE must ensure that categorical data is accompanied by descriptive code lists or vocabularies to provide context and meaning. Provide links or references to code lists and vocabularies that describe categorical data values. Include URIs or embedded definitions in the responses to facilitate understanding of categorical data.

4.3.3 Data Returned is for the Requested Subset of the Original Source Data

FAIRiCUBE must ensure that data returned from a WCS request corresponds precisely to the subset specified in the request. The requested spatial, temporal, and attribute filters should be accurately applied to the data returned. Test requests to confirm that the returned data matches the specified subset criteria.

4.3.4 Data is Returned in the Requested Format

FAIRiCUBE should guarantee that data is returned in the format specified by the user in the request. WCS should support various data formats (e.g., GeoTIFF, NetCDF, HDF) as specified in its capabilities. Verify that the data format in the response matches the requested format.

4.3.5 Data is returned clearly indicating the property the data represents

Ensure that the returned data clearly specifies what property or attribute it represents.
Each dataset should include descriptive labels that specify the property being represented (e.g., temperature, precipitation, vegetation index). The response should include clear information about the attributes, units of measurement, and any relevant property descriptions. Provide accompanying documentation or metadata that details the nature of the property represented by the data.

Ensure that the WCS response structure includes fields or elements that explicitly state the property name and description. Use metadata tags to include detailed descriptions and units for each property.

4.3.6 Data is linked to relevant metadata

FAIRiCUBE will make sure that all data returned by WCS is accompanied by comprehensive metadata, providing context and additional information. Use established metadata standards to ensure consistency and comprehensiveness. Make metadata easily accessible and linked directly from the data response. Metadata should include information about data provenance, creation date, creator, spatial and temporal coverage, and any other relevant details.
Embed metadata references directly in the WCS response. Maintain and provide detailed metadata records that can be accessed through the WCS service.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

4.4 Data Access S3

Validation pf Data Access on AWS S3 buckets (S3 in the following) focuses on pricing transparency, data optimization for cloud retrieval, and authentication requirements. It should ensure that data access via S3 is user-friendly, efficient, and secure.

The validation criteria include:

- Transparent pricing for data access (free, requester pays, quota limits, free trials…)
- Gridded data stored in S3 is expected to be "cloud optimized" for efficient data retrieval (e.g. cloud optimized formats such as zarr, COG; data chunked or tiled when necessary)
- Clear authentication requirements

4.4.1 Transparent Pricing for Data Access

It is important to clearly outline the different pricing models available for data access. This will make sure users clearly understand the costs associated with accessing data stored in S3. This may include free access, requester-pays models, quota limits, and free trials. Ensure that pricing information is straightforward, without hidden fees, allowing users to estimate their costs accurately.

4.4.2 Gridded data in S3 expected to be "cloud optimized" for efficient data retrieval

FAIRiCUBE should guarantee that gridded data stored in S3 is optimized for efficient access and retrieval, particularly for cloud-based processing. This may be done using formats such as Cloud Optimized GeoTIFF (COG) and Zarr that are designed for efficient access in cloud environments. Organise data into chunks or tiles to facilitate efficient retrieval of subsets without needing to download the entire dataset.

4.4.3 Clear authentication requirements

FAIRiCUBE should ensure that users are aware of and understand the authentication requirements for accessing data stored in S3. Clearly define the access policies, including whether access is public, restricted, or requires specific permissions. Specify the methods of authentication required, such as AWS IAM roles, access keys, or temporary security credentials. Provide detailed instructions on how to obtain the necessary credentials and configure their access settings.

4.5 Data Access Sentinel Hub

Sentinel Hub[6] ensures users have a transparent understanding of costs and a clear path to authenticate and access the required data. The validation criteria include:

- Transparent pricing for data access (free, requester pays, quota limits, free trials…)
- Clear authentication requirements

[6] https://www.sentinel-Hub.com/

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

4.5.1 Transparent pricing for data access (free, requester pays, quota limits, free trials…)

FAIRiCUBE should ensure users have clear information about the costs associated with data access. Provide detailed pricing information, including options such as free access, requester-pays models, quota limits, and free trials.

4.5.2 Clear authentication requirements

FAIRiCUBE will define and communicate the authentication process required for accessing data. Clearly outline the steps and necessary credentials for user authentication, ensuring secure and straightforward access to data.

4.6 Data Access Rasdaman

Rasdaman ensures transparent pricing, efficient data retrieval, and secure, clear access to the required data. The validation criteria include:

- Transparent pricing for data access (free, requester pays, quota limits, free trials…)
- Support for Cloud-Optimized Data
- Clear authentication requirements

4.6.1 Transparent pricing for data access

Ensure users understand the costs associated with accessing data. Clearly display pricing details, including options such as free access, requester-pays models, quota limits, and free trials.

4.6.2 Support for Cloud-Optimized Data

Facilitate efficient data retrieval by supporting cloud-optimized formats. Ensure that gridded data in Rasdaman is stored in cloud-optimized formats such as Zarr and COG and is chunked or tiled when necessary for efficient access.

4.6.3 Clear authentication requirements

Provide a straightforward and secure process for user authentication. Clearly define the authentication steps and required credentials, ensuring users can easily and securely access the data.

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

5 Processing

5.1 Lab

The validation criteria for executing scripts and Jupyter notebooks[7] include:

- Scripts and Jupyter notebooks can be executed without system error
- Required libraries are available
    - If libraries are missing, mechanism in place to add
    - Environments are in sync between EOX and Rasdaman Jupyter Labs
- Processing resources can be scaled to the requirements of a specific task
    - Headless execution of code is supported
        - Logging is possible and accessible (for long running jobs)
- ML experiment tracking is available (e.g. to support reproducibility and record model accuracy)
- Backup mechanism in place
- Requested services activated and correctly set up (e.g. environment variables configured)

Regular reviews and updates, combined with user feedback, will help maintain the high quality and relevance of the documentation over time

5.1.1 Scripts and Jupyter Notebooks can be executed without system error

Ensure that all scripts and notebooks run smoothly without errors. Regularly test scripts and notebooks in the environment to identify and fix potential issues.

5.1.2 Required libraries are available

Ensure that all necessary libraries are pre-installed and available for use. Maintain a list of required libraries and pre-install them in the environment.

If libraries are missing, mechanism should be in place to add them (i.e., allow users to easily install additional libraries as needed). This should include instructions or tools, to do so.

Moreover, environments are in sync between EOX and Rasdaman Jupyter Hubs. Need to ensure consistent environments across different Jupyter Hubs. Regularly synchronisation of environment configurations and libraries between EOX and Rasdaman Hubs is required.

[7] https://fairicube.rasdaman.com/
https://eoxHub.fairicube.eu/

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

5.1.3 Processing resources can be scaled to the requirements of a specific task

Allow scaling of processing resources to match the demands of specific tasks. Implement scalable resource allocation, such as adjustable CPU, memory, and GPU resources.

- Headless execution of code is supported: Enable code execution without a graphical interface. Support command-line execution and batch processing for headless operations. Scripts should be able to run in the background or via command line interfaces without requiring user interaction.
- Logging is possible and accessible (for Long Running Jobs): Provide logging capabilities for monitoring and troubleshooting long-running jobs. Implement a logging system that captures and stores logs, making them accessible to users.

5.1.4 ML experiment tracking is available (e.g., to support reproducibility and record model accuracy)

Facilitate the tracking of machine learning experiments to ensure reproducibility. Integrate tools like MLflow to track experiments, including parameters, metrics, and outputs. Ensure that experiments can be reproduced with the same parameters and configurations. Document the process for tracking experiments and reviewing results.

5.1.5 Backup mechanism in place

Ensure data and code are regularly backed up to prevent loss. Implement automated backup procedures for user data, notebooks, and environment configurations.

5.1.6 Requested services activated and correctly set up (e.g. environment variables configured)

Ensure all necessary services and configurations are properly activated and set up. Provide scripts or procedures to activate services and configure environment variables as needed.

5.2 GitHub

The validation criteria for FAIRiCUBE GitHub[8] include:

- Repository content and structure overview (README)
- License
- Findability of content

[8] https://gitHub.com/FAIRiCUBE

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

5.2.1 Repository content and structure overview (README)

Provide users with a clear understanding of the repository's contents and structure. Include a detailed README file that outlines the purpose of the repository, its main components, folder structure, and usage instructions.

5.2.2 License

Clearly define the terms under which the repository's content can be used, modified, and shared. Add a LICENSE file to the repository specifying the chosen open-source license (e.g., MIT, Apache 2.0).

5.2.3 Findability of content

Ensure the repository is easily discoverable by users searching for related content. Use relevant keywords, tags, and topics in the repository's metadata. Optimize the repository description and README with searchable terms related to its content.

5.3 WCPS

The validation criteria for WCPS include:

- Request provides a correct response
- Data returned is for the requested subset of the original source data
- Processing steps numerically accurate
- Data is returned in the requested format
- Code can be integrated as UDF

5.3.1 Request provides a correct response

Ensure that WCPS requests return accurate and valid responses. Ensure that WCPS requests return accurate and valid responses by carrying out regular tests on queries. Implement robust error handling to catch and provide meaningful messages for incorrect requests.

5.3.2 Data returned is for the requested subset of the original source data

Ensure that the data returned matches the exact subset criteria specified in the request. Verify that spatial, temporal, and attribute filters are correctly applied to return the specified subset. Ensure that no extraneous data is included, and no requested data is omitted.

5.3.3 Processing steps numerically accurate

Ensure that any processing performed on the data is numerically correct. Validate processing algorithms and compare the results with known correct outputs to ensure accuracy. Maintain high precision in numerical calculations to avoid errors and inaccuracies.

5.3.4 Data is returned in the requested format

Ensure that the data is returned in the format specified by the user. Support multiple output formats and validate that the data is correctly converted and returned in the requested format (e.g., GeoTIFF, NetCDF).

5.3.5 Code can be integrated as UDF

Allow users to integrate custom code for processing as user-defined functions. Provide an interface or framework for users to upload and execute UDFs within the WCPS environment, ensuring compatibility and security.

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

6 Conclusion

To sum up, the "Validation of Sharing" document provides a comprehensive overview of the essential components within FAIRiCUBE Hub that have been validated to uphold the FAIR principles—Findable, Accessible, Interoperable, and Reusable. Through meticulous examination, this document has highlighted key areas such as core functionality, information management, data handling practices, and processing capabilities.

Validation efforts have ensured that FAIRiCUBE Hub aligns effectively with these principles, from maintaining robust documentation and knowledge bases to implementing standardized metadata and accessible data access protocols like WCS, S3, Sentinel Hub, and Rasdaman platform. Additionally, the validation criteria underscored the platform's capacity for efficient data processing and analysis through integrated tools such as the lab environment, GitHub repositories, and WCPS.

By documenting these validation processes, stakeholders and users of FAIRiCUBE Hub can confidently engage with the platform, knowing it meets stringent standards for data integrity, accessibility, and usability across diverse Use Cases. Moving forward, ongoing validation will continue to uphold these principles, ensuring FAIRiCUBE Hub remains a trusted resource in facilitating collaborative and impactful research environments.

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

7 Appendix: Validation checklist

The following Table summarises the validation steps described in this deliverable.

Table 1 : Validation of sharing checklist.

Component	Process	Check type	Characteristic	Description
Core	Hub	Findability	Findability of components	FiC Hub UI has clear navigation structure and section names; index of all components is up to date; a search function exists.
		Authentication	Authentication across components	Common, secure authentication method available across all components; standard authentication protocol implemented; user roles defined and documented.
		Interplay	Accessibility of datasets	Data resources can be retrieved for use from their metadata records; access modalities clearly described (e.g. local download, cloud APIs…)
			Accessibility of analysis/processing resources	A/p resources can be retrieved for use from their metadata records; access modalities clearly described (e.g. Jupyther notebook, Python library)
			Accessibility of linked datasets and a/p resources	Provenance data or a/p resources or derived data resources are described and accessible
			Storage and accessibility of processing results	Processing results are securely stored and readily accessible for inspection and reuse by authenticated as well as unauthenticated users
			Visualization of processing results	Visualization software for gridded and non-gridded data is an integral component of the Hub and available to users
		User support	User support/helpdesk	Users know how to ask for support in case of questions or technical problems; user support is conducted in a structured manner (e.g. ticketing system); enquiries are timely answered.
Information	Documentation and Knowledge Base	Findability	Findability of required information	Information is organised and structured; consistent labeling of sections; search function is implemented
		Correctness	Correctness of available information	Periodic reviews of the content are conducted; information is kept up to date; broken hyperlinks are replaced

Funded by
the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.

		Completeness	Completeness of available information	All components/topics mapped to respective documentation/articles; external expert review mechanism is implemented
		Usefulness	Usefulness of available information	Periodic reviews of the content are conducted; structured system for collecting user feedback is in place (e.g. comments, rating)
		Ease of Provision	Ease of provision of documentation	Regular maintenance of the system is scheduled; contribution methods are intuitive and documented
Data	Metadata Editors	Persistence	User input persisted in static file	Ensure all values entered in the GUI are accurately saved in a JSON file.
		Alignment	Aligned metadata schema across project deliverables, Editor and metadata records	The Editor and the metadata records conform to the metadata schema documented in the project deliverable.
		Findability	Code lists findable and complete, sorted	Code lists (implemented as dropdown lists) contain all necessary options alphabetically sorted; a dictionary is provided
		Labeling	All labels in the GUI described	Every input field has a label and a clear description.
		Errors	Human-readable error messages	Error messages are clear and provide actionable information.
	Data Catalog	Findability	Findability of required data based on search criteria	Users can efficiently locate datasets using specific search criteria.
		Simple Search	Simple search	Basic search interface for quick dataset retrieval implemented.
		Advanced Search	Advanced search	Detailed search interface to search across all metadata fields implemented.
		Navigation	Ability to navigate to the identified dataset (either by link to original data download, when possible; or API endpoint with related API document for data access, e.g. endpoint for data in S3 bucket)	Data resources can be retrieved for use from their metadata records; access modalities clearly described (e.g. local download, cloud APIs…)
Data Access	WCS	Correct Response	Request returns correct responses in accordance with the WCS specification	Ensure all WCS requests return valid responses according to the specification.
		Measure Types	Measure types from SWE Common used according to the original specification	Ensure measure types are used correctly as per the SWE Common specification.
		XML Validity	XML responses valid	Validate all XML responses against the relevant schemas.
		Extent/Bbox	Extent/bbox information aligned with GML	Ensure extent and bounding box information follow GML standards.
		Subset Accuracy	Data returned is for the requested subset of the original source data	Ensure data returned matches the specified subset criteria.

		Format	Data is returned in the requested format	Ensure data is returned in the user-specified format.
	S3	Pricing	Transparent pricing scheme for data access	The different pricing schemes for data access are clearly and transparently described
		Optimisation	Data formats are optimized for cloud storage	Only cloud-optimized data formats (e.g. COG, zarr) are allowed on S3
		Authentication	Clear authentication requirements	Authentication options (e.g. external authentication) and requirements (e.g. subscription) are clearly and transparently described.
	Sentinel Hub	Pricing	Transparent pricing scheme for data access	The different pricing schemes for data access are clearly and transparently described
		Authentication	Clear authentication requirements	Authentication options (e.g. external authentication) and requirements (e.g. subscription) are clearly and transparently described.
	GitHub	Repository Content Overview	Provide users with a clear understanding of the repository's contents and structure.	Include a detailed README file that outlines the purpose, components, structure, and usage instructions of the repository.
		License	Clearly define the terms under which the repository's content can be used, modified, and shared.	Add a LICENSE file specifying the chosen open-source license (e.g., MIT, Apache 2.0).
		Findability	Ensure the repository is easily discoverable by users searching for related content.	Use relevant keywords, tags, and topics in the repository's metadata to optimize discoverability.
Processing	Lab	Execution	Scripts and Jupyter NB can be executed without error	Jupyter Lab and Python is up to date; essential libraries are installed and up to date (e.g. ipykernel)
		Libraries	Required libraries are available	Commonly used libraries for EO and ML are installed as part of the initial set up
		Library Addition	If libraries are missing, mechanism in place to add	Structured mechanism to request new libraries is implemented (e.g. through helpdesk)
		Scaling	Processing resources can be scaled to the requirements of a specific task	Ensure processing resources can be scaled as needed.
		Backup	Backup mechanism in place	Data and configuration of the Lab is regularly backup; the backup mechanism is known to the user; users can request restoring older backups
		Activation	Requested services activated and correctly set up	Strucutred way to request activation of additional services implemented; activated services are tested before being made available

Funded by
the European Union

This project has received funding from the Horizon Europe research
and innovation programme under grant agreement No. 101059238.

WCPS		Response	Request provides a correct response	Ensure the request provides a correct response.
		Subset Accuracy	Data returned is for the requested subset of the original source data	Ensure data returned matches the requested subset.
		Accuracy	Processing steps numerically accurate	Ensure numerical accuracy of processing steps.
		Format	Data is returned in the requested format	Ensure data is returned in the requested format.
		Integration	Code can be integrated as UDF	Ensure code can be integrated as User-Defined Functions (UDFs).

Funded by the European Union

This project has received funding from the Horizon Europe research and innovation programme under grant agreement No. 101059238.