

# CWTS review of Research Portal Denmark: Technical infrastructure

*November 2024*



# CWTS review of Research Portal Denmark: Technical infrastructure

## Report for NORA

Mogens Sandfær, *Head of NORA*

E-mail [mosa@dtu.dk](mailto:mosa@dtu.dk)

## Project team at CWTS

Nees Jan van Eck, *Project leader*

E-mail [ecknjpvan@cwts.leidenuniv.nl](mailto:ecknjpvan@cwts.leidenuniv.nl)

Bram van den Boomen

E-mail [b.van.den.boomen@cwts.leidenuniv.nl](mailto:b.van.den.boomen@cwts.leidenuniv.nl)

André Brasil

E-mail [a.l.brasil.varandas.pinto@cwts.leidenuniv.nl](mailto:a.l.brasil.varandas.pinto@cwts.leidenuniv.nl)

CWTS B.V.

P.O. Box 905

2300 AX Leiden, The Netherlands

Tel. +31 71 527 3909

Fax +31 71 527 3911

E-mail [info@cwts.nl](mailto:info@cwts.nl)

# Table of contents

<b>1. Introduction</b> .....	<b>5</b>
<b>2. Data sources</b> .....	<b>7</b>
2.1 Global data sources.....	7
2.2 Local data sources.....	8
<b>3. Data workflow</b> .....	<b>11</b>
3.1 Data harvesting .....	11
3.1.1 Global data sources .....	12
3.1.2 Local data sources .....	13
3.2 Data enhancements and enrichments.....	13
3.3 Data linking.....	15
3.3.1 Linking global data publication records .....	15
3.3.2 Linking local data publication records.....	16
3.3.3 Consolidating global and local publication clusters .....	18
3.3.4 Cleaning and normalization .....	18
3.4 Data presentation.....	19
3.4.1 Local data search module .....	19
3.4.2 Global data search module.....	19
3.4.3 Analytical modules .....	21
<b>4. Software stack</b> .....	<b>23</b>
<b>5. Code management</b> .....	<b>25</b>
<b>6. Server and backup management</b> .....	<b>26</b>
<b>7. Documentation</b> .....	<b>27</b>
<b>8. Data validation</b> .....	<b>28</b>
8.1 LOI issues.....	28
8.2 GOI issues .....	29
8.3 NOI issues .....	29

8.4 Author name and ORCID inconsistencies.....	30
<b>9. Conclusions and key recommendations .....</b>	<b>31</b>
<b>Appendix 1: LOI, GOI, and NOI issues.....</b>	<b>35</b>
<b>Appendix 2: Author name and ORCID inconsistency .....</b>	<b>37</b>

# 1. Introduction

The Research Portal Denmark<sup>1</sup> is an online platform that aims to provide comprehensive access to information about Danish research. By integrating data from various sources, it consolidates information on Danish research outputs and activities. The portal makes this consolidated information openly accessible, providing a centralized data infrastructure that enables users to discover, explore, and analyze Danish research, thereby enhancing the visibility and accessibility of this research. The portal offers a valuable resource for researchers, policy makers, funding bodies, and the general public interested in understanding the landscape of Danish research.

The Research Portal Denmark is being developed by NORA (National Open Research Analytics) commissioned by and in close collaboration with the Danish Agency for Higher Education and Science of the Ministry of Higher Education and Science and with the portal's advisory board<sup>2</sup>. A first version of the Research Portal Denmark was officially launched in March 2024<sup>3</sup>. To inform the development of future versions of the portal, NORA has commissioned the Centre for Science and Technology Studies (CWTS) at Leiden University to perform an in-depth review of the portal's technical infrastructure. This report presents the findings of the review along with recommendations for future improvements to the portal.

To ensure a thorough and proper review of the portal's technical infrastructure, NORA provided CWTS with full access to all relevant data, procedures, source codes, and associated documentation. This comprehensive access enabled us to perform a detailed evaluation of the portal's infrastructure and its various components. The documentation included system architecture diagrams, process flowcharts, user manuals, and technical specifications. We studied all this material in detail.

In addition, we conducted a site visit on May 22 and 23, 2024. During this visit, we held several meetings with the NORA team and the external developers involved in the project. These meetings were instrumental in gaining a deeper understanding of the practical implementation of the portal's infrastructure. We had discussions about various aspects of the technical infrastructure, including the data workflow, software solutions, and system integrations. These conversations allowed us to better grasp the

---

<sup>1</sup> <https://forskningsportal.dk>

<sup>2</sup> <https://forskningsportal.dk/about-the-portal/the-ministry-advisory-board>

<sup>3</sup> <https://forskningsportal.dk/conference>

rationale behind specific technical choices and implementations. The site visit also provided an opportunity to observe the day-to-day operations and collaboration dynamics between the NORA team and the external developers. This was essential for identifying what works well within the current setup and where there may be opportunities for improvements.

During our review we paid special attention to the functioning of key components in the data workflow, such as the harvesting, deduplication, linking, and enrichment of the portal's data. We not only assessed the technical performance of these components, but also evaluated how effectively they integrate with each other to support the overall functionality of the portal. Understanding the data workflow in detail allowed us to pinpoint specific stages where improvements could be made to enhance efficiency and data quality. Additionally, we examined the organizational aspects of the infrastructure, focusing on how the NORA team collaborates with the external developers and how data, source codes, and servers are managed to ensure the continuity and reliability of the portal.

The remainder of this report is structured as follows. We start in Section 2 by discussing the various data sources integrated in the portal. In Section 3, we detail the current data workflow, examining each stage from data acquisition to processing and integration, and highlighting strengths and weaknesses. Section 4 examines the software stack used for the portal, assessing the effectiveness, complementarity, and overlap of the software solutions and tools employed. Section 5 focuses on source code management practices, including storage, access, and maintenance, and suggests enhancements where necessary. Section 6 addresses server and backup management, exploring infrastructure measures for data security and continuity, and recommending improvements. In Section 7, we cover documentation practices with suggestions for improvements. Section 8 presents the results of a data validation exercise we performed to assess the integrity and reliability of the current data workflow. Finally, in Section 9, we summarize our key findings and recommendations, and propose a future data workflow for the Danish Research Portal to enhance its efficiency, reliability, and scalability.

## 2. Data sources

The Research Portal Denmark integrates data from a variety of sources to offer a comprehensive view on Danish research. It distinguishes between global data sources, which include information on research outputs and activities worldwide (within the context of the Research Portal Denmark, however, only outputs and activities related to Danish research institutions are considered), and local data sources, which contain information solely on outputs and activities from Danish research institutions.

None of the global and local data sources cover all Danish research outputs. However, the data sources complement each other and together the data sources aim to ensure that the Research Portal Denmark provides a comprehensive and detailed overview of Danish research outputs.

### 2.1 Global data sources

The portal integrates publication information from three prominent global data sources:

- Dimensions<sup>4</sup> (provided by Digital Science<sup>5</sup>)
- Scopus<sup>6</sup> and SciVal<sup>7</sup> (provided by Elsevier<sup>8</sup>)
- Web of Science<sup>9</sup> and InCites<sup>10</sup> (provided by Clarivate<sup>11</sup>)

These global data sources cover a wide range of peer-reviewed articles, conference papers, patents, and other scholarly works published globally. However, the Research Portal Denmark specifically includes only publications authored by individuals affiliated with Danish organizations.

---

<sup>4</sup> <https://www.dimensions.ai>

<sup>5</sup> <https://www.digital-science.com>

<sup>6</sup> <https://www.elsevier.com/products/scopus>

<sup>7</sup> <https://www.scival.com>

<sup>8</sup> <https://www.elsevier.com>

<sup>9</sup> <https://clarivate.com/products/scientific-and-academic-research/research-discovery-and-workflow-solutions/webofscience-platform>

<sup>10</sup> <https://clarivate.com/academia-government/scientific-and-academic-research/research-funding-analytics/incites-benchmarking-analytics/>

<sup>11</sup> <https://clarivate.com>

The global data sources currently considered are frequently used and well-respected for their data quality. However, a disadvantage of these data sources lies in their proprietary nature. Because these data sources are proprietary, they impose restrictions on how their data can be used. For the Research Portal Denmark this means that some information from these global data sources cannot be presented in the web interface or included in user exports. This significantly limits the value of the data sources. To overcome this limitation, the NORA team plans to integrate OpenAlex<sup>12</sup> as a global data source. OpenAlex, made by the nonprofit OurResearch<sup>13</sup>, is a free and open database of scholarly papers, journals, researchers, and institutions along with their connections.

We strongly support the idea of adding open data sources like OpenAlex to the set of global data sources. OpenAlex offers an excellent API<sup>14</sup> for efficient data retrieval and, more importantly, its data is CC0<sup>15</sup> licensed, meaning it can be used completely free of restrictions. This has a number of advantages. Data collection will be simpler, as there will be no restrictions on the use of data and no need for licensing agreements. For portal users, this translates to complete information visibility within the portal and direct access to source data without requiring any subscriptions. Importantly, there will be no restrictions on data included in exports. This enhancement would empower users to conduct various follow-up analyses based on Research Portal Denmark data, which is currently not possible.

## 2.2 Local data sources

In addition to global data sources, the portal integrates publication information from research information systems and portals managed by various Danish research institutions. These systems, known as CRIS systems (Current Research Information Systems)<sup>16</sup>, are used to register and manage the publication outputs and activities of institutions. The CRIS systems of the following institutions currently serve as local data sources for the Research Portal Denmark:

- Aalborg University (AAU)

---

<sup>12</sup> <https://openalex.org>

<sup>13</sup> <https://ourresearch.org>

<sup>14</sup> <https://docs.openalex.org/how-to-use-the-api/api-overview>

<sup>15</sup> <https://creativecommons.org/public-domain/cc0>

<sup>16</sup> [https://en.wikipedia.org/wiki/Current\\_research\\_information\\_system](https://en.wikipedia.org/wiki/Current_research_information_system)



- Aarhus School of Architecture
- Aarhus University (AU)
- Copenhagen Business School (CBS)
- Danish Institute for International Studies (DIIS)
- Danish School of Media and Journalism (DMJX)
- Design School Kolding
- IT University of Copenhagen (ITU)
- Ministry of Culture Denmark - Archives, Museums, and Royal Library Denmark (KUM)
- Roskilde University (RUC)
- Royal Danish Academy - Architecture, Design, Conservation
- Technical University of Denmark (DTU)
- The Danish Center for Social Science Research (VIVE)
- UCL University College (UCL)
- University College Absalon (AB)
- University College Copenhagen (KP)
- University College of Northern Denmark (UCN)
- University College South Denmark (UC SYD)
- University of Copenhagen (KU)
- University of Southern Denmark (SDU)
- VIA University College (VIA)

The institutional research information systems currently used as local data sources are at the moment all based on PURE<sup>17</sup>, a CRIS system provided by Elsevier. However, efforts are ongoing to also include systems from additional institutions that are based on other CRIS platforms, such as Esploro<sup>18</sup> from Clarivate.

An important feature of PURE is its support for the Danish Research Database Metadata Exchange Format (DDF-MXD)<sup>19</sup>, a format developed in Denmark for exchanging metadata of publications. The Research Portal Denmark uses the DDF-MXD format to retrieve publication information from the local systems of Danish research institutions in a standardized manner. This standardization ensures full interoperability of publication information from different systems, simplifying follow-up processing and reducing errors. Another advantage of the DDF-MXD format is that it is well

---

<sup>17</sup> <https://www.elsevier.com/products/pure>

<sup>18</sup> <https://clarivate.com/products/scientific-and-academic-research/research-funding-and-analytics/esploro/>

<sup>19</sup> <https://forskningsportal.dk/national-exchange-format>

documented and under active development<sup>20</sup>. For instance, support for ROR<sup>21</sup> and OpenAlex organization identifiers has been added recently.

We commend the Danish research community for developing the DDF-MXD format. Equally notable is Elsevier's integration of DDF-MXD support in PURE<sup>22</sup> and Clarivate's steps towards introducing DDF-MXD support in their Esploro CRIS system. By developing the DDF-MXD format and integrating it into systems and infrastructures like PURE and the Research Portal Denmark, Denmark sets an example for other countries in promoting open, accessible, and interoperable research data.

To further improve the coverage of Danish research information in the Research Portal Denmark, we recommend investigating whether other CRIS systems that are in use by Danish research organizations can also support the DDF-MXD format. If this is not immediately possible, developing customized modules that transform data from formats supported by the CRIS systems into the DDF-MXD format could be considered.

---

<sup>20</sup> [https://forskningsportal.dk/wp-content/uploads/2024/04/DDF\\_MXD\\_v1.4.4.pdf](https://forskningsportal.dk/wp-content/uploads/2024/04/DDF_MXD_v1.4.4.pdf)

<sup>21</sup> <https://ror.org>

<sup>22</sup> <https://helpcenter.pure.elsevier.com/ddf-mxd/ddf-mxd-documentation>

## 3. Data workflow

Processing of data from the data sources discussed in Section 2 follows a number of steps. First, the publication data is harvested from these data sources (Section 3.1). The collected publication data is then processed further by cleaning and enriching it (Section 3.2). In addition, the publication records from the different data sources are matched and linked to each other (Section 3.3). Finally, the publication data and analytics on top of the data are exposed in the frontend of the Research Portal Denmark (Section 3.4). In the following subsections, we will discuss each of these steps in more detail.

Two separate pipelines are used to harvest and process data, one for the global data sources and one for the local ones. These pipelines use different software applications. While maintaining separate pipelines for global data and local data can offer certain advantages, it may also present several disadvantages. We will discuss these disadvantages in the report as well.

### 3.1 Data harvesting

As discussed in Section 2, the data for the Research Portal Denmark originates from multiple sources, each implementing their own method of accessing the data. Furthermore, the format in which this data is obtained varies in syntax and schema. Consequently, it is necessary to develop customized approaches for retrieving and storing this data.

Although there is a long list of local data sources, all local data sources considered by the Research Portal Denmark have adopted a uniform data format (DDF-MXD) and data acquisition method. Consequently, a single data harvesting approach suffices for obtaining data from the local data sources. In contrast, each of the three global data sources uses a completely different data format and method for accessing the data. As a result, four different customized approaches are needed to harvest publication information and populate the portal.

Data from the global data sources is stored as JSON in a MongoDB<sup>23</sup> instance. MongoDB is a NoSQL<sup>24</sup> database which excels in the storage, wrangling, and retrieval of JSON

---

<sup>23</sup> <https://www.mongodb.com>

<sup>24</sup> <https://en.wikipedia.org/wiki/NoSQL>

documents. For data from the local data sources a different approach is taken. The data is stored as XML in an SQLite<sup>25</sup> database.

### 3.1.1 Global data sources

All global data sources offer multiple methods for harvesting their data, including downloading a complete data snapshot, downloading a filtered data snapshot tailored to the client's needs, or utilizing an API. For each global data source, the Research Portal Denmark works with the filtered snapshot or the API rather than the complete data snapshot. This is a good choice, as it prevents spending excessive resources to store large amounts of data and filter it locally.

#### 3.1.1.1 Scopus and SciVal

The data from Elsevier is delivered in the form of two filtered snapshots, one for the basic Scopus data and one for the enhancements from SciVal. The filtered snapshots exclusively include the metadata for publications indexed in Scopus that are authored by individuals affiliated with Danish organizations. Initially delivered in XML format, the data is subsequently converted to JSON. The reason for this conversion is that the MongoDB instance used for data storage does not support XML storage. Since JSON and XML are feature-equivalent, the converted data can be considered 'unedited'.

#### 3.1.1.2 Web of Science and InCites

The data from Clarivate was originally obtained by querying the Web of Science and InCites APIs and downloading the metadata in JSON format. Recently, Clarivate changed this to deliveries in the form of two filtered snapshots, one for basic Web of Science data and one for enhancements from InCites. The filtered snapshots include only the metadata of publications associated with the country code of Denmark and are provided in XML format. Before storing the data in the MongoDB instance, it is first converted to JSON, following a process that is similar to that used for the Elsevier data.

#### 3.1.1.3 Dimensions

In contrast, the data from Digital Science is retrieved using their Dimensions API. Specifically, filtered API requests are made to obtain relevant publication information. The resulting JSON is directly stored in the MongoDB instance. However, the Dimensions API imposes certain restrictions and limits, which necessitate splitting the data harvesting on the client side. The API does not allow returning all metadata fields

---

<sup>25</sup> <https://www.sqlite.org>

in a single response. The names of the supported metadata fields are therefore first scraped from the API documentation page. Subsequently, these fields are queried separately and the returned data is merged on the client side.

The harvesting of Dimensions data is somewhat cumbersome, fragile, and prone to errors, and it would be preferable to simplify it. However, since the API restriction is imposed on the side of Dimensions, simplifying the process requires Dimensions to update its API. Despite the technical limitations, the decision has been made to move forward with the API and to implement automated validation of the harvesting process to identify and rectify any errors. We consider this to be the best possible solution given the circumstances.

### 3.1.2 Local data sources

As discussed in Section 2, all local data sources considered by the Research Portal Denmark are currently based on PURE. PURE provides an API using the OAI-PMH (Open Archives Initiative Protocol for Metadata Harvesting)<sup>26</sup> protocol to obtain data in XML format according to the DDF-MXD schema. The OAI-PMH protocol ensures a uniform method for obtaining data, while the DDF-MXD schema provides a consistent format for the data. This standardization means that data harvesting publication information from the different local data sources can be done using the same pipeline, simplifying the process and ensuring consistency.

Complete data dumps from the local data sources are downloaded in parallel, while files from each source are downloaded sequentially. This setup seems to be a good choice. It minimizes delays, as downloading all sources consecutively would risk bottlenecks from slower sources. Additionally, downloading files sequentially for each source is a practical choice, as not all sources can support multiple simultaneous downloads.

Since data from the local data sources is processed separately from the global data, it is not converted to JSON but instead it is stored unedited in an SQLite database.

## 3.2 Data enhancements and enrichments

The first step in enhancing and enriching the collected global data is to create a more uniform dataset. A new dataset containing the global data is created using a schema

---

<sup>26</sup> <https://www.openarchives.org/pmh>

that ensures uniform fields across all records. Some preliminary cleaning and mapping are performed at this stage, but no complex processing is conducted.

Additionally, a separate dataset is created to map unique name variants for Danish organizations, countries, subject classifications, and open access categories found in the data to their corresponding cleaned and standardized names. For each name variant of Danish organizations, the dataset also includes the number of records in which it occurs.

With each update of the global and local data sources, unmapped name variants of Danish organizations are collected and manually assigned with cleaned and standardized names. Google Sheets is used as a user interface for this process. First, unmapped Danish organization name variants are automatically pushed to a Google Sheets spreadsheet. Where possible, an automated suggestion for a mapping is included for human review. Next, a NORA team member manually assigns a standardized name in the spreadsheet whenever possible for each unmapped name variant. Name variants occurring only in a few records are not taken into account and are automatically mapped to an “Other” category. In the case of organizations, the standardized names are also accompanied by a ROR ID (if a ROR could be uniquely identified for this standardized name) and an organization type. In the case of countries, the standardized names are accompanied by a region. The spreadsheet has been carefully designed. In future versions of the spreadsheet, we recommend adding a column to record the name of the team member who last changed the mapping and the date on which the mapping was last updated. This will help to trace who made which changes.

The use of Google Sheets is an excellent approach to centralize the unification process and prevent NORA members from directly modifying the global and local data. It leverages the collaborative features of Google Sheets. For instance, the team member who assigns the standardized name has the possibility to add comments or remarks. This is useful for collaboration and sharing information with other team members, especially when dealing with difficult cases. It also enables quality assurance checks by NORA analysts on every update, such as checking the number of mapping candidates. Additionally, Google Sheets offers a basic backup strategy and revision history, allowing changes to be rolled back if necessary. We suggest keeping a copy of the spreadsheet for each update of the global and local data to facilitate a hard rollback to a specific version of the portal.

The mappings of name variants to standardized names are automatically pulled from the Google Sheets spreadsheet into the MongoDB database. The mappings are then made available through a REST<sup>27</sup> API, which is implemented in Python using the FastAPI<sup>28</sup> framework and documented using Swagger<sup>29</sup>. These are widely used tools for constructing robust and easily extendable REST APIs.

Finally, the mappings are applied in the global and local data pipelines to enhance and enrich the publication records.

### 3.3 Data linking

Because different records of the same publication can appear in multiple data sources or even in the same data source, it is essential to cluster and link these records into groups that represent actual publications. Since the type of data that is available in the local and global data sources is very different and of varying quality, different clustering approaches are used for each dataset. This includes clustering the global and local data separately and subsequently linking the resulting global and local publication clusters.

#### 3.3.1 Linking global data publication records

For the matching and clustering of publication records in the global dataset, a rule-based matching and clustering algorithm<sup>30</sup> is used. An example of a matching rule is "The DOI and title of the records match and the ISSN and ISBN are not in conflict". If two records can be linked to each other based on a matching rule, they are grouped into the same cluster. Matching rules are applied sequentially until a match is found. Records that do not satisfy any matching rule are not linked to other records.

We consider the approach used for matching and clustering publication records in the global dataset to be well-suited for this type of data. Data from the global data sources has typically already been professionally cleaned and enhanced by the data providers, and therefore the publication records are usually of high quality and relatively complete. For instance, author names, publication years, and page numbers are typically available in a clean format, and source names are standardized. As a result,

---

<sup>27</sup> <https://en.wikipedia.org/wiki/REST>

<sup>28</sup> <https://fastapi.tiangolo.com>

<sup>29</sup> <https://swagger.io>

<sup>30</sup> For an example of a rule-based matching and clustering algorithm in the context of citation matching, see <https://doi.org/10.1002/asi.23590>.

relatively simple matching rules suffice. These simple rules ensure that the reasons for publication records to be clustered together are transparent and traceable. In addition, a record can be kept of the specific matching rule that led to the clustering of a publication record. This enables quality checks and validation at later stages in the process.

The matching and clustering of publication records in the global dataset result in the assignment of a Global Object Identifier (GOI) to each publication record. This process is performed using a Neo4j<sup>31</sup> instance. Neo4j is a graph database. Graph databases are well-suited for clustering operations. Therefore, the use of Neo4j in this context is an appropriate choice.

Finally, after each run of the matching and clustering algorithm, reports are generated to compare the results of the matching and clustering against previous iterations. In particular, the number of clusters per matching rule, which is included in one of the reports, offers a good indicator for quality assurance. The reports can be analyzed in each update to ensure no unexpected errors have occurred. It should even be possible to build automated alerts for specific cases, for instance when certain counts show a large increase or decrease.

### 3.3.2 Linking local data publication records

In contrast to most data from the global data sources, data from the local data sources can be much harder to process. This data has typically been entered and submitted by many different people, leading to more errors, typos, and omissions. Moreover, records for the same publication, originating from different CRIS systems, may contain conflicting information. The local data also includes a greater variation in publication types than the current global data sources, including conference and communication contributions that often have less metadata. Given the varying quality and completeness of the local data, the use of a more sophisticated blocking and score-based matching and clustering algorithm<sup>32</sup> instead of using a simple rule-based one is appropriate. The algorithm assigns a matching score to corresponding fields of pairs of publication records. These scores are weighed by a factor taken from a manually constructed table. The factor that is applied depends on whether the score is

---

<sup>31</sup> <https://neo4j.com>

<sup>32</sup> For examples of blocking and score-based matching and clustering algorithms in the context of author name disambiguation, see <https://doi.org/10.1007/s11192-020-03410-y> and [https://doi.org/10.1162/qss\\_a\\_00081](https://doi.org/10.1162/qss_a_00081).



considered a match, a non-match, or whether the field is missing in either record. This approach ensures that certain fields, such as the DOI, have more weight in the matching process than other fields, such as page numbers. If two records have a cumulative score above a specific threshold, they are considered the same publication and placed in the same cluster. This process is repeated until no more changes to the clusters are made. The cluster algorithm has been implemented from scratch in Perl. The amount of data that needs to be processed is relatively small, and therefore a well-performing Python program should suffice.

While this approach introduces more complexity, making provenance of the clustering and quality assurance more difficult, we consider the approach to be very sophisticated and well-suited given the local data's variability. However, the scoring system seems somewhat complicated and hard to understand. We suggest exploring ways to simplify the scoring system in order to enhance transparency in the matching process and facilitate quality assurance.

The matching and clustering of publication records in the local dataset result in the assignment of a Local Object Identifier (LOI) to each publication record. This identifier is partly semantic, composed of a unique ID along with cluster-specific information such as the number of records in the cluster and the matching percentage. Including this information in a LOI may make quality assurance a little simpler because of the easy availability of some information about a cluster. However, it also means that LOIs can change when any information in the cluster changes. We consider this an important disadvantage. Having identifiers for clusters that are persistent between updates is likely to be of significant help in (automated) quality assurance, even though it may sometimes be difficult to determine whether a cluster is still the same when records are added, removed, or modified between updates. We recommend to explore the possibility of having persistent identifiers between updates. The need for persistent identifiers also applies to the GOIs.

After matching and clustering, the information in each cluster is consolidated to obtain a record of the cluster that contains the 'best' value of each metadata field. This is done using a heuristic approach with distinct rules tailored to each metadata field. These rules are easy to understand and seem very sensible.

Similar to the global matching and clustering process, automated reports are generated for quality assurance after each run of the local matching and clustering algorithm. Since the matching and clustering are more challenging for the local data than for the global data, quality assurance inevitably is also more difficult for the local

data. Nevertheless, the quality assurance reports offer an excellent solution for ensuring high-quality results and detecting any issues at an early stage.

### 3.3.3 Consolidating global and local publication clusters

The next step is to connect the data from the global and local data sources. The matching and clustering process used in this step assigns a NORA Object Identifier (NOI) to global and local publication clusters. In this way, global publication clusters and local publication clusters that represent the same publications should be linked to each other. The process is very similar to the process of producing global publication clusters and uses mostly the same clustering rules. Given the cleaned and consolidated state of the local data at this stage, there is no need for a more complex approach, and applying the same rule-based approach therefore seems appropriate.

However, it may be worth exploring if the NOI clustering can be omitted entirely by including the consolidated local publication clusters directly in the global data matching and clustering process. This approach seems feasible given the quality of the data resulting from the local publication clustering and the associated consolidation process. It would have the advantage of having to run only two matching and cluster algorithms instead of three, and the need to maintain only a single set of matching rules. Additionally, it could prevent some discrepancies that may result from clustering the same data twice, for example having multiple GOI clusters in a single NOI cluster (see Section 8).

### 3.3.4 Cleaning and normalization

In the steps outlined above, various metadata fields in the global and local datasets need to be cleaned and normalized. Currently, these operations are somewhat intertwined with the matching and clustering processes. We recommend performing cleaning and normalization before the matching and clustering processes, using a single repository of functions. This approach ensures that cleaning and normalization are performed in a uniform and consistent way for all data. For instance, cleaning a DOI string should be done in the same way in all processes, so the function that is used for this should be shared across all processes. Performing DOI cleaning immediately after storing the raw data ensures that there are no discrepancies between DOIs anywhere else in the pipelines.

## 3.4 Data presentation

To present the results of the processed data and to enable users to interact with these results, multiple modules are available in the frontend of the Research Portal Denmark. First of all, two separate publication search modules are available. One for the global data and one for the local data. While the look and feel of these two search modules is very similar, the backend of the associated web applications are very different. In addition to the publication search modules, there is a search module available for Danish patents. Finally, two analytical modules are available: An analytical module for the Danish Open Access Indicator, monitoring the realization of the Danish national open access strategy of 2018, and a prototype dashboard monitoring the Danish green research strategy.

### 3.4.1 Local data search module

To present the data from the local data sources, a tailor-made frontend has been developed. The results from the publication matching and clustering, and consolidation are enriched with the NORA cleaned and standardized entity names, and stored in an Elasticsearch<sup>33</sup> engine to facilitate the necessary performance.

As Elasticsearch is a state-of-the-art search and indexing engine that natively uses JSON as a data format, this seems a good choice as the basis for a frontend that is primarily concerned with search and filter operations. The frontend itself is a custom-built website using plain HTML and JavaScript. In our experience, it is beneficial to use a framework such as Angular<sup>34</sup>, React<sup>35</sup>, or Vue<sup>36</sup> to build these types of frontends when the scale increases and multiple developers are involved. This helps manage some of the inherent complexity. However, for the current situation, the existing approach suffices and avoids introducing any unnecessary dependencies.

### 3.4.2 Global data search module

The final processing steps and presentation of the data from the global data sources, are done using VIVO<sup>37</sup>. VIVO is an open source software platform designed for creating

---

<sup>33</sup> <https://www.elastic.co/elasticsearch>

<sup>34</sup> <https://angular.dev>

<sup>35</sup> <https://react.dev>

<sup>36</sup> <https://vuejs.org>

<sup>37</sup> <https://vivo.lyrasis.org>

and maintaining a web-based system that can display the research activities of an institution. It is typically used by universities, research organizations, and other institutions to showcase the expertise, publications, research projects, and other academic activities of their faculty and researchers. Since VIVO is an integrated solution that uses semantic web technologies at its core, the data is first obtained in its processed JSON format (Section 3.1.1), then combined with the NORA enhancements (Section 3.2) and mapped to the VIVO ontology<sup>38</sup> in the RDF data format. This ensures the data is structured in a way that aligns with VIVO's semantic framework, facilitating efficient data presentation and retrieval.

Data is stored separately for each of the three global data sources, with each data source having its own Apache Solr<sup>39</sup> index to increase performance. A fourth index is built to store the connections between records from the different global data sources. This setup ensures efficient retrieval and management of data across the global data sources. Furthermore, processing is undertaken to consolidate conflicting metadata fields found in records from different sources. These consolidations are integrated into the RDF version of the data.

We recommend that the additional processing performed to apply the NORA enhancements (Section 3.2) and to consolidate the data can best take place before the mapping to the VIVO ontology. In this way, the enhancement of the data and the process of exposing the data to users are separated and the results of the enhancements can be stored as a single source of truth. It also ensures that the enhancements are applied in a uniform way to the global and local data, minimizing inconsistencies and errors.

The use of ontologies and semantic web technologies in VIVO is well-suited for unifying similar data from different sources. However, introducing such technologies at this stage of the process seems redundant, as most data processing has already been completed. VIVO currently seems to be used mainly as a frontend application for the global data, but it requires several adjustments and workarounds to provide the desired features and performance. We recommend either to introduce VIVO at an earlier stage in the process, using its software stack and technologies also for storage, processing, and exposing of the local data, or to replace VIVO with a frontend application that uses the current local data pipeline stack and that is therefore easier

---

<sup>38</sup> [https://doi.org/10.1007/978-3-031-79435-3\\_2](https://doi.org/10.1007/978-3-031-79435-3_2)

<sup>39</sup> <https://solr.apache.org>

to maintain. This also eliminates the need to keep the two frontend applications in sync, visually and functionally, when new features are added.

### 3.4.3 Analytical modules

While the NORA team plans to offer more analytical modules in the future, currently there are two modules available: A module for Open Access and a module for Green Research. The Open Access module focuses on the Danish Open Access Indicator<sup>40</sup>. It provides insights into the realization of the Danish Open Access Strategy 2018-2025<sup>41</sup>. It is a tailor-made web application offering a few dashboards as well as access to the raw data in a few formats. The Green Research module offers a dashboard for data related to the Danish Green Research Strategy<sup>42</sup>. It is built in Looker<sup>43</sup>, a business intelligence dashboarding tool provided by Google, and offers a dashboard with some limited interactivity.

A valuable feature of the Green Research module is its integration with the search interface, enabling users to directly access publications underlying specific figures. Implementing this functionality in the Open Access module could provide similar benefits.

Currently, both analytical modules are based on a snapshot of the data in the Research Portal Denmark. Unlike the search modules, they do not use the latest version of the data. This is intentional, as certain indicators such as the Danish Open Access Indicator are updated annually and accommodate publication delays due to embargoes. However, using different data versions could create discrepancies across modules when viewed in the frontend of the portal.

Also, because the modules are implemented in different frameworks, they have a distinctly different look and feel. While this has the benefit of flexibility during development, the user experience of the platform may be improved by providing an interface for the analytical modules that has the same look and feel as the interface

---

<sup>40</sup> <https://ufm.dk/en/research-and-innovation/cooperation-between-research-and-innovation/open-access/Publications/open-access-barometer>

<sup>41</sup> <https://ufm.dk/forskning-og-innovation/samspil-mellem-viden-og-innovation/open-access/artikler/danmarks-nationale-strategi-for-open-access/danmarks-nationale-strategi-for-open-access-1>

<sup>42</sup> <https://ufm.dk/publikationer/2020/fremtidens-gronne-losninger-strategi-for-investeringer-i-gron-forskning-teknologi-og-innovation/gron-forskningsstrategi>

<sup>43</sup> <https://cloud.google.com/looker>

for the search modules. We recommend using the same web framework for the analytical modules and the search modules to maintain uniformity and to minimize time spent on maintenance. As mentioned before, our suggestion is to make use of standardized web frameworks such as Angular, React, or Vue (see also Section 3.4.1).

## 4. Software stack

The number of steps involved in processing all the data and presenting it (Section 3) is quite large, as there is a need to deal with multiple data sources, multiple data formats, people doing manual work, and some complex data processing. On top of that, different parts of the two pipelines and the different user interfaces to the portal are being developed and maintained by different contracted developers. To accommodate this complexity, the decision was made to approach this in a flexible and modular way, selecting software applications that are specialized for specific tasks and integrating them, instead of trying to use a single integrated solution. This approach ensures that developers can work with applications with which they have experience and that problems can be solved efficiently by leveraging specialized software. However, this approach can also lead to fragmentation and additional overhead, like multiple servers and software packages, and it increases the time needed to integrate and maintain the software. Additionally, this can make it more difficult for a single individual to have a solid grasp on the entirety of the project and to maintain the various applications over a longer period of time.

The following software applications make up the main stack of the Research Portal Denmark:

**Python:** Orchestration of the global data workflow is mainly done in the Python programming language.

**MongoDB:** Database engine used for the storage of the data from the global data sources, the NORA enhancements, and the results of the global matching and clustering.

**Neo4j:** Graph database engine used to ingest the global data and perform the global matching and clustering.

**FastAPI:** Python framework used to provide an internal and public endpoint for data stored in the MongoDB instance.

**VIVO:** Application framework used to build a user interface for the global data portal. VIVO also uses its own software stack to store, process and serve data.

**Java:** Mapping, processing, and orchestration of data within VIVO is done in the Java programming language.

**Apache Solr** (integrated in VIVO): Search indexing engine to provide efficient search for the global data portal.

**Perl:** Orchestration of the local data workflow is done in the Perl programming language.

**SQLite:** Database engine used for the storage of the data from the local data sources and to store the results of the local matching and clustering, consolidation, and enhancements.

**Elasticsearch:** Search indexing engine to provide efficient search for the local data portal.

**HTML+JS:** Used to build the user interface for the local data portal.

For most of the components of the technical stack there is a clear justification, but there also seem to be redundancies. There are three database engines in use, two search indexing engines, and two methods of frontend development. This is mostly due to the two separate pipelines that process the different datasets. We recommend to evaluate whether there are parts of the stack that can be handled by components already present elsewhere in the stack. This helps to ensure that there is a smaller dependency footprint, that the stack is simpler, and that the stack is easier to maintain by a smaller team.



## 5. Code management

Because different parts of the Research Portal Denmark are developed by different teams and individuals, the source code for these parts are managed in separate repositories. All code is tracked by Git<sup>44</sup> and stored on GitHub<sup>45</sup>. This standard practice facilitates collaboration, maintains a complete development history, allows reverting the code to specific points in time, and ensures a reliable backup of the code.

We recommend to have a uniform structure for names, code and documentation across the different repositories. This makes it easier for external parties to scrutinize the source codes. It also facilitates collaboration. We also recommend to use more of the features that Git and GitHub provide for quality control. For example, features could be developed in separate branches and be pulled into the main branch only after a code review by another developer. This would not only improve the quality of the code, but would also make more members of the team familiar with the general outline and functioning of the codebase. The use of GitHub Issues<sup>46</sup> can also greatly enhance collaboration and transparency, as everyone involved can stay updated on specific problems and progress. We recommend to use the issues feature of the GitHub repositories in which the source codes live, making it easy to link issues to specific code commits, pull requests, and branches. This integration ensures a seamless workflow from identifying a problem to implementing a fix and tracking its solution. For project management, GitHub Projects<sup>47</sup> offers features to manage work across the multiple repositories. By using GitHub Projects, the NORA team could visualize tasks, organize issues, and track development progress, enabling an efficient coordination. Lastly, GitHub Actions<sup>48</sup> can be leveraged for automated testing or deployment. This could minimize the chance of erroneous code being pushed to production servers.

---

<sup>44</sup> <https://git-scm.com>

<sup>45</sup> <https://github.com>

<sup>46</sup> <https://docs.github.com/en/issues>

<sup>47</sup> <https://docs.github.com/en/issues/planning-and-tracking-with-projects/learning-about-projects/about-projects>

<sup>48</sup> <https://docs.github.com/en/actions>

## 6. Server and backup management

Execution of the different processes and components of the Research Portal Denmark is carried out on several virtual servers, all of which are running on-premises on machines managed by the Department of IT Service of DTU. The use of different servers for different tasks is beneficial as long as the data transfer and communication between servers is relatively easy. Moreover, it is essential to maintain a clear distinction between development and production servers. Specifically, we recommend separating servers dedicated to data processing from those responsible for hosting the web modules of the portal. This separation ensures that data processing does not impact the performance of the web modules and the experience of end users. Finally, having the servers managed by a single internal service provider seems a good choice from the viewpoint of maintenance, reliability and security.

For backing up the pipeline codes, the built-in capabilities of Git and GitHub are used. This ensures that rolling back code to an earlier version is straightforward and that, in addition to a local copy, all code is also securely stored on GitHub servers. For backups (and rollback) of the Google Sheets spreadsheet that includes the manual mappings, the built-in capabilities of Google Sheets are used. There is currently no strategy for backups for the processed data itself. In case of an emergency, the processed data can in principle be recreated from the raw data and the pipeline codes. Nevertheless, we recommend creating a periodical backup of the raw data and processed data tied to a specific version (or Git commit hash) of the pipeline codes. This for example enables making comparisons between different versions of the data in case of suspicious patterns in the data. It also helps to meet data management requirements.

## 7. Documentation

The technical documentation of the Research Portal Denmark is generally comprehensive and well-written. However, due to the fragmentation of the various technical components, gaining an overview of the entire project and all its dependencies can be a daunting task. Despite this, the documentation effectively describes most parts of the project. A weakness of the documentation is its lack of uniformity. Some elements of the data workflow are described in multiple places in different ways, and occasionally there is conflicting information. These discrepancies are likely due to updates in the project that are not yet fully reflected in the documentation. Differences in documentation styles among the various developers may play a role as well.

At the moment, Google Docs is used to produce documentation. To simplify creating and managing documentation, we recommend using a modern documentation platform like GitBook<sup>49</sup>, Read the Docs<sup>50</sup>, or MkDocs<sup>51, 52</sup>. These platforms provide user-friendly interfaces, real-time editing, and integration with version control systems. They enhance accessibility and customization, ensuring up-to-date, professional, and easily navigable documentation.

---

<sup>49</sup> <https://www.gitbook.com>

<sup>50</sup> <https://about.readthedocs.com>

<sup>51</sup> <https://www.mkdocs.org>

<sup>52</sup> <https://squidfunk.github.io/mkdocs-material>

## 8. Data validation

To validate the data workflow described in Section 3, we examined multiple random samples of the processed data from the Research Portal Denmark. By thoroughly checking the sampled data, we aim to validate whether the data workflow delivers accurate results and to provide insights for further improvements. We focused in particular on the clustering and linking of publication records (see Section 3.3), as this is a crucial and challenging step in the data workflow. Additionally, we also paid special attention to the data consolidation and enrichment processes to ensure their accuracy and consistency.

Our validation is based on the following samples obtained from the May 2024 version of the portal:

- 50 LOI publication clusters: 25 containing only one publication record and 25 containing multiple publication records
- 50 GOI publication clusters: 25 containing only one publication record and 25 containing multiple publication records
- 50 NOI publication clusters: 25 containing only one publication cluster and 25 containing multiple publication clusters

We looked at publication clusters that contain only a single publication (or cluster) to determine whether there are any other publications (or clusters) that should belong to that publication cluster (false negatives). We also looked at publication clusters that contain multiple publications (or clusters) to determine whether any of the clustered publications (or clusters) should not belong to that publication cluster (false positives).

In the following subsections, we describe our observations. A list of issues we encountered can be found in Appendix 1.

### 8.1 LOI issues

In the sampled LOI publication clusters, we encountered one false positive and two false negatives. The false positive involved two distinct conference papers (with the same title and authors, but presented at two different editions of the same conference) that were incorrectly clustered together. The false negatives involved two publication clusters that should have included additional versions of the publications. These versions were not matched, presumably because of slightly differing titles and journal information.

As there were only three errors in a sample of 50, with at least one being an edge case, we consider the clustering of publication records from the local data sources to perform well. This is a notable accomplishment given the varying quality and completeness of these publication records (see Section 3.3.2).

## 8.2 GOI issues

In the sampled GOI publication clusters, we encountered no false positives, but we did find nine false negatives. However, these false negatives can all be explained by incorrect or missing information in the source records rather than errors in the matching and clustering algorithm. Most of the false negatives were caused by publication records from Dimensions for which affiliation information was missing. Since the Danish affiliations were missing for these publication records, these records were not even harvested from Dimensions, as the harvesting step relies on country information in affiliations (see Section 3.1.3). This indicates that missing affiliation information in Dimensions is a significant issue that needs to be taken into account. Since this is a Dimensions issue, it cannot be fixed by the NORA team directly but must be addressed by Digital Science. Apart from this, the clustering of publication records from the global data sources performs well.

## 8.3 NOI issues

In the sampled NOI publication clusters, we found two false negatives where a local publication cluster was not matched to a global publication cluster and two false negatives where a global publication cluster was not matched to a local publication cluster. The main reason for these mismatches seems to be missing information in the local publication records, resulting in insufficient data to support accurate clustering.

Additionally, we identified one GOI publication cluster within a NOI publication cluster that should have contained an additional publication record from Dimensions. Most likely, this publication record was not assigned to the GOI publication cluster by the global matching and clustering algorithm due to an anomalous publication year in Dimensions.

In the sampled NOI publication clusters, we found no false positives.

Finally, we found a few cases where there were more than two publication clusters within a single NOI publication cluster. This is surprising, as it suggests that either multiple GOI publication clusters or multiple LOI publication clusters were clustered together within the NOI publication cluster, indicating that some publication records

were not properly clustered by the global or local matching and clustering algorithm. This inconsistency suggests that there is an error in either the GOI/LOI publication clustering or in the NOI publication clustering. Our advice is to further investigate this issue. To maintain consistency across the matching and clustering approaches, we recommend limiting each NOI publication cluster to a single GOI publication cluster and/or a single LOI publication cluster. Alternatively, to avoid such consistency issues altogether, it may be worth exploring if the NOI clustering can be omitted entirely by including the consolidated local publication clusters directly in the global data matching and clustering process (see Section 3.3.3).

#### **8.4 Author name and ORCID inconsistencies**

During our data validation, we found several inconsistencies in author names and ORCID identifiers in the processed global data. In Appendix 2, we report an example of such an inconsistency.

## 9. Conclusions and key recommendations

We commend the NORA team for their excellent work on the Research Portal Denmark. In general, we consider the technical infrastructure to be robust and well designed. The most crucial and challenging algorithms for harvesting, deduplicating, linking, and enriching publication records from a variety of data sources have been properly designed and implemented. Regular manual quality checks performed by the analysts on every update cycle further ensure high quality of the consolidated data. Most of the issues that we encountered in our review are relatively minor and seem to be related to fragmentation and the inheritance of older processes and software, which is to be expected in any project of this size and scope. Different programming languages and software applications are used for similar types of operations. This is a natural consequence of the modular setup and of working with developers with diverse experience and expertise. There is room for improvement here. For instance, a more uniform approach could be taken for the presentation of results. Adopting a single web framework and a consistent set of software applications could simplify code sharing and collaboration among developers. This consolidation would also simplify maintenance of software and servers. Furthermore, certain processes in the data workflow could be streamlined by performing steps like data cleaning and normalization at earlier points in the data workflow. These improvements would likely help to reduce inconsistencies in the results.

Throughout this report we have provided various more detailed recommendations. In the remainder of this section, we summarize the key recommendations. To contextualize our key recommendations, the diagram in Figure 1 illustrates an adjusted data workflow that can be considered for the Danish Research Portal. This data workflow builds on the current data workflow but incorporates some structural changes that will make the data workflow more robust, easier to maintain, and less complex. To structure the proposed data workflow, we follow to some extent the Medallion Architecture<sup>53, 54</sup>, a data design pattern proposed by Databricks. In the Medallion Architecture, three datasets are constructed throughout a data workflow: A

---

<sup>53</sup> <https://www.databricks.com/glossary/medallion-architecture>

<sup>54</sup> <https://medium.com/@junshan0/medallion-architecture-what-why-and-how-ce07421ef06f>

*Bronze* dataset containing the unaltered raw data, a *Silver* dataset containing the filtered, cleaned, and transformed data, and a *Gold* dataset containing the final version of the data including enrichments and aggregates needed for applications and user interfaces.

Harvesting of raw data is currently well-managed and can remain largely unchanged. The only adjustment we propose is to store the local data in the same database engine as the rest of the data. This eliminates the need for multiple pipelines at the earliest possible moment. We do not foresee issues with converting between data formats, such as JSON and XML, as the content remains unchanged. The harvesting step results in the *Raw dataset* containing all 'unprocessed' data.

The next step involves consolidating the raw data into a unified schema to ensure that subsequent processing can be performed in a consistent way. This schema should be sufficiently flexible to accommodate upstream changes or the inclusion of new data sources. To apply this schema, transformation, cleaning, and normalization are required. Some of these operations are specific to one data source, while others may apply to all data sources. We strongly recommend applying automated cleaning, normalization, and transformation to all data at this stage. This makes the data as uniform as possible and reduces the need for similar operations at later points in the pipeline. This step results in the *Parsed dataset*, containing a clean and uniform version of the data that is easy to use in more complex subsequent steps.

At this stage, we deviate slightly from the Medallion Architecture. Our suggestion is to perform manual processing on the cleaned and transformed data, as this data can be processed in efficient ways. This aligns with the current data workflow. The results of the manual processing can either be integrated into the *Parsed dataset* or be used to construct a new dataset. In any case, the result is the *Enhanced dataset*, containing a uniform version of the data including enrichments such as standardized organization names, country names, field classifications, and open access information.

The final processing step involves the matching, clustering, and linking of publication records that represent the same publication to consolidate information from different sources and eliminate duplicates. We recommend to first cluster and link the publication records from the local data sources and consolidate their information. Subsequently, a second clustering and linking process can be performed consisting of the consolidated local publication records (including references to the original local publication records) and the publication records from the global data sources. We advise to flag clusters containing multiple consolidated local publication records for



review, as these signify discrepancies between the two clustering and linking processes. The results of the linking and consolidation can be stored in a separate dataset, be integrated into the Enhanced dataset, or be used to construct a new dataset that also includes the enhanced data. Additionally, aggregated data or precalculated indicators or metrics can be added to this dataset. The resulting *Enhanced & consolidated dataset* should be structured in such a way that the web modules and other user analyses can use the data directly without further processing. This approach ensures consistency across web modules and user analyses, as all data originates from the same source. It also allows, for instance, the integration of the separate search modules currently available for global and local data into a single module.

To provide an easy way for users to access and analyze the data, we recommend setting up different APIs, such as a REST API and a query language (e.g., SQL) interface. This has the additional advantage that it is easier to control data access. The REST API would facilitate easy data exchange with users and downstream data sources or services. The query language interface would provide a good starting point for data analysts interested in performing large-scale analyses based on the enhanced data from the portal, and there would be no need for these analysts to be familiar with the specific data storage application used for the Enhanced & consolidated dataset. Depending on how the Enhanced & consolidated dataset is set up, it might also be beneficial to connect the applications that serve the portal, like the search and analytical modules, to the REST API.

Finally, we recommend exploring the possibility of implementing a feedback loop for the global and local data sources. For instance, if a publication record in a global data source has missing or incorrect institutional affiliation information for Danish research organizations, the data from the portal can be used to enrich or correct the publication record in the global data source. This would be particularly valuable when integrating open data sources such as OpenAlex. Providing feedback to upstream open data sources does not only benefit the users of these data sources. It also improves the inclusion of Danish publications in bibliometric analyses performed based on these open data sources and in downstream infrastructures ingesting data from these open data sources. This contributes to the bigger goal of the Danish Research Portal of enhancing the global visibility of Danish research.

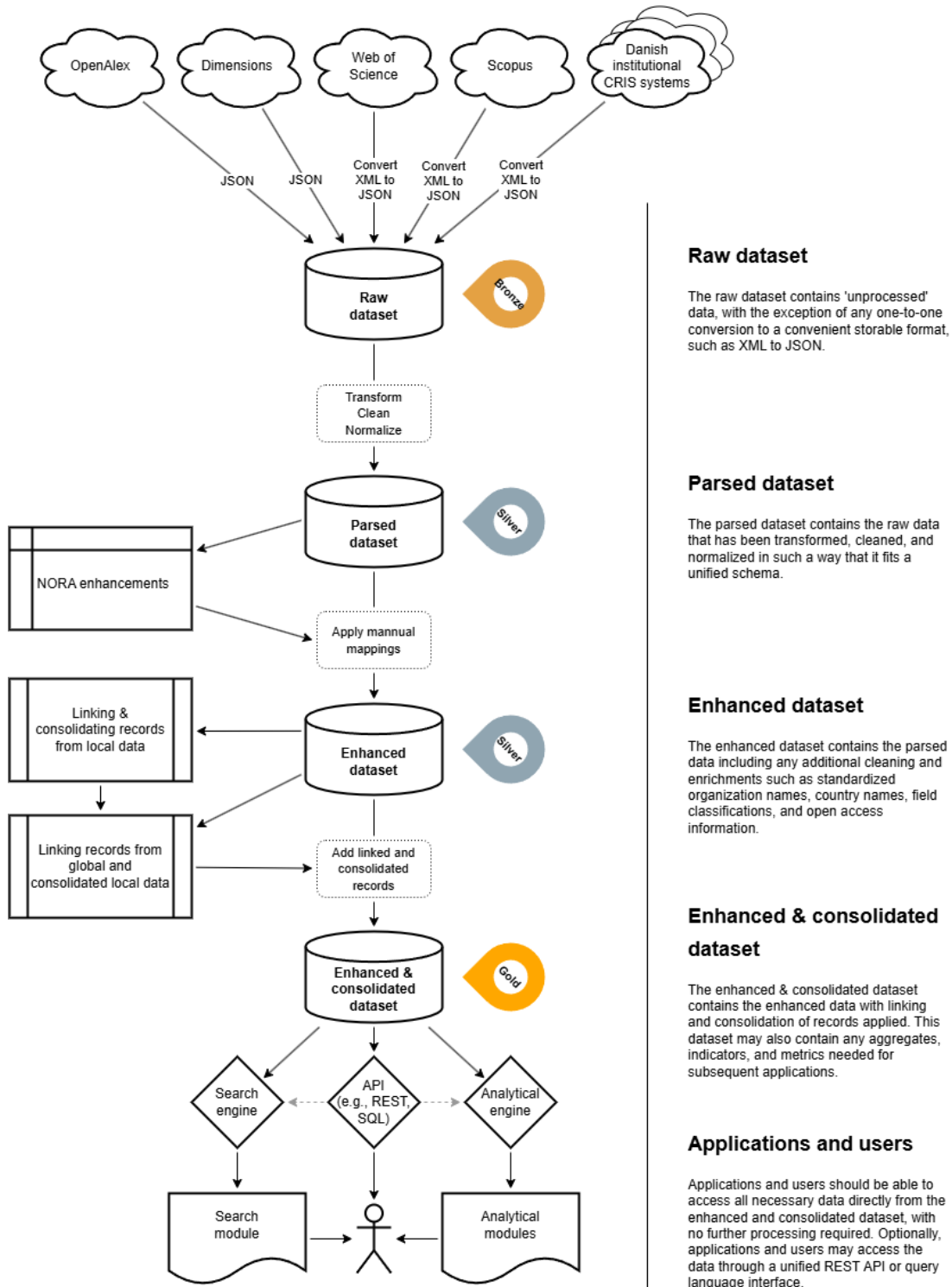


Figure 1: Adjusted data workflow for the Danish Research Portal.

## Appendix 1: LOI, GOI, and NOI issues

In the data validation (see Section 8), we encountered several problematic publication clusters. The following table provides detailed information on these publication clusters.

Cluster Type	Cluster ID	Publication ID	Data Source	Error Type	Notes
LOI	S01-000.00-517480	ku-cf9f0334-1159-4d81-be89-3694a2b3198a	Local data source	False negative	LOI should have included S01-000.00-131531. Slightly different publication and source titles. Same DOI, publication year, volume, and ISSN.
LOI	S01-000.00-572671	ruc-f7c17497-8263-44a0-b6b1-a8015338399f	Local data source	False negative	LOI should have included S01-000.00-502513. Slightly different publication and source titles. Same publication year and page numbers.
LOI	S02-089.39-032676	dtu-cc2039d3-9606-4e44-9c61-b46cce9a4bbc	Local data source	False positive	Local publication records should not have been clustered. Publications with the same title but presented at different editions of the same conference.
LOI	S02-089.39-032676	dtu-04c88384-5db0-47a3-93c9-8d3182b872f9	Local data source	False positive	Local publication records should not have been clustered. Publications with the same title but presented at different editions of the same conference.
GOI	7227980	WOS:000515121900042	Global data source, Web of Science	False negative	Not a Danish publication. Web of Science affiliation data is incorrect.
GOI	7235778	WOS:000773543202075	Global data source, Web of Science	False negative	GOI should have included Dimensions pub.1141857940. Affiliation is missing in Dimensions.
GOI	7418989	pub.1157680542	Global data source, Dimensions	False negative	GOI should have included Web of Science WOS:000995814708485. Dimensions data is not clean.
GOI	7449348	2-s2.0-85088341474	Global data source, Scopus	False negative	GOI should have included Dimensions pub.1098953049. Affiliation is missing in Dimensions.
GOI	7569823	2-s2.0-85141144526	Global data source, Scopus	False negative	GOI should have included Dimensions pub.1086629830. Affiliation is missing in Dimensions.
GOI	7608804	2-s2.0-85118270875	Global data source, Scopus	False negative	GOI should have included Dimensions pub.1140487757. Affiliation is missing in Dimensions.
GOI	7691535	WOS:000300400600016	Global data source, Web of Science	False negative	GOI should have included Dimensions pub.1058849067. Affiliation is missing in Dimensions.
GOI	7586260	2-s2.0-85115895268	Global data source, Scopus	False negative	GOI should have included Dimensions pub.1086840059. Affiliation is missing in Dimensions.
GOI	7586260	WOS:000722640900007	Global data source, Web of Science	False negative	GOI should have included Dimensions pub.1086840059. Affiliation is missing in Dimensions.
GOI	7723922	2-s2.0-85011906876	Global data source, Scopus	False negative	GOI should have included Dimensions pub.1083653247. Affiliation is missing in Dimensions.
GOI	7723922	WOS:000397109700006	Global data source, Web of Science	False negative	GOI should have included Dimensions pub.1083653247. Affiliation is missing in Dimensions.
NOI	Global-1-1-9268452	2-s2.0-85025693067	Global data source, Scopus	False negative	NOI should have included LOI S01-000.00-085604. Missing source information in local publication record.

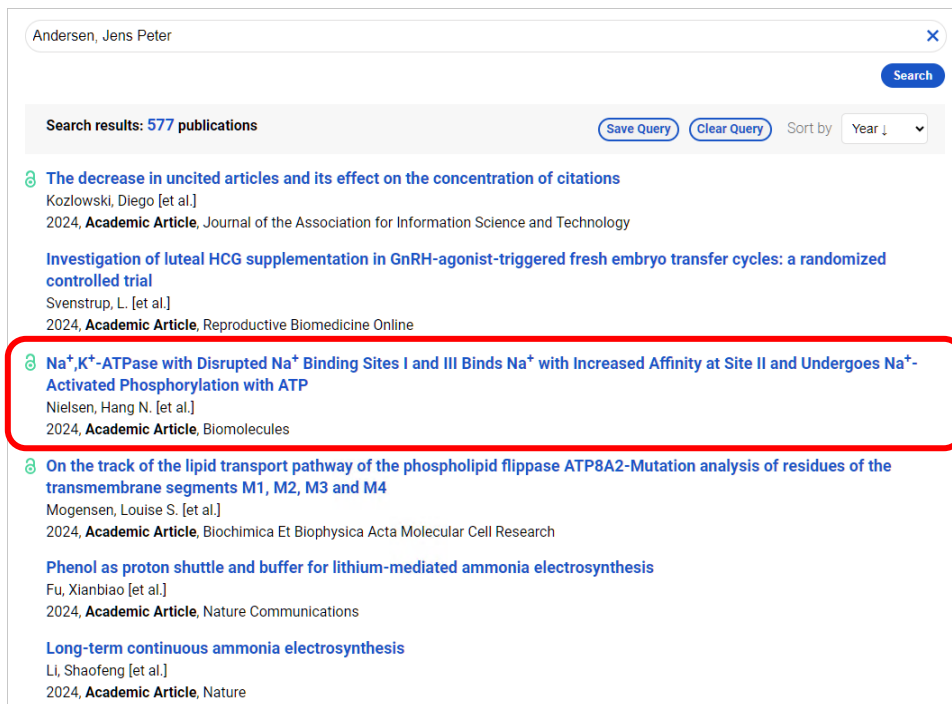
NOI	Global-1-2-9314991	2-s2.0-84960812863	Global data source, Scopus	False negative	NOI should have included LOI S01-000.00-517708. Missing source information in local publication record.
NOI	Local-1-1-8460140	aa-u-e74e1564-82bd-445a-b509-1580d435a0bc	Local data source	False negative	NOI should have included GOIs 7498122 and 7686790. GOIs 7498122 and 7686790 should have been merged.
NOI	Local-1-1-8698039	cbs-88854f14-8b41-45e1-8e8c-f0d6178bf453	Local data source	False negative	NOI should have included GOI 7687979. Missing authors in local publication record.
NOI	Global-Local-2-3-8905460	2-s2.0-85072080615	Global data source, Scopus	False negative	Dimensions pub.1091479830 should have been assigned to the same GOI. Different publication year. Slightly different publication titles and author names.
NOI	Global-Local-2-3-8905460	WOS:000493547500001	Global data source, Web of Science	False negative	Dimensions pub.1091479830 should have been assigned to the same GOI. Different publication year. Slightly different publication titles and author names.
NOI	Global-Local-2-3-8905460	ku-8626a8a4-48eb-4d38-bcc9-6b44ccb9c997	Local data source	False negative	
NOI	Global-Local-3-3-8984765	sdu-be0ecad5-e679-41ec-8836-bfbd9d60e76	Local data source	Multiple LOIs	LOIs should have merged. One of the local publication records contains erroneous information.
NOI	Global-Local-3-3-8984765	sdu-f268153d-b3fc-4d69-b787-ae15e4b1202a	Local data source	Multiple LOIs	LOIs should have merged. One of the local publication records contains erroneous information.
NOI	Global-Local-3-3-8984765	2-s2.0-84976496499	Global data source, Scopus		
NOI	Global-Local-3-7-8987806	2-s2.0-84911049553	Global data source, Scopus	False positive	Multiple letters with the same DOI, should not have clustered in the same GOI and NOI.
NOI	Global-Local-3-7-8987806	2-s2.0-84911059434	Global data source, Scopus	False positive	Multiple letters with the same DOI, should not have clustered in the same GOI and NOI.
NOI	Global-Local-3-7-8987806	WOS:000345347600022	Global data source, Web of Science	Multiple GOIs	Web of Science WOS:000345347600023 contains erroneous information, so should not be present in this cluster (and clustered together in the same GOI as WOS:000345347600022).
NOI	Global-Local-3-7-8987806	WOS:000345347600023	Global data source, Web of Science	Multiple GOIs	Web of Science WOS:000345347600023 contains erroneous information, so should not be present in this cluster (and clustered together in the same GOI as WOS:000345347600022).
NOI	Global-Local-3-7-8987806	pub.1026023131	Global data source, Dimensions		
NOI	Global-Local-3-7-8987806	au-bf08686a-0a2b-473f-8441-1fe2ee00bd2a	Local data source		
NOI	Global-Local-3-7-8987806	sdu-269d03d0-eed5-4441-9f64-7660ce02c92e	Local data source		

Table A1.1: LOI, GOI, and NOI issues encountered in the data validation.

## Appendix 2: Author name and ORCID inconsistency

During our data validation (see Section 8), we found several inconsistencies in author names and ORCID identifiers in the processed global data of the Research Portal Denmark. In this appendix, we report an example of such an inconsistency. In this example, publications of Jens Peter Andersen are attributed to Jacob Palsgaard Andersen in the processed global data.

Using the global search module, a search for either "Andersen, Jens Peter" (Figure A2.1) or "Andersen, Jacob Palsgaard" (Figure A2.2) returns the same article from 2024 in the journal *Biomolecules*. When selecting the article, the author name in the global search module is "Andersen, Jacob Palsgaard" (Figure A2.3). However, in the original Web of Science data, the author is listed as "Jens Peter Andersen" (Figure A2.4). In addition, the linked ORCID identifier in the Research Portal Denmark points to the ORCID profile of Jens Peter Andersen (Figure A2.5). While it seems likely that the identical surname and initials of the two authors caused the inconsistency, it is not entirely clear from which step in the data processing the inconsistency originates. Since we found these inconsistencies only in the global data, they seem to be introduced somewhere in the VIVO processing.



Andersen, Jens Peter

Search

Search results: 577 publications

Save Query Clear Query Sort by Year ↓

**The decrease in uncited articles and its effect on the concentration of citations**  
Kozlowski, Diego [et al.]  
2024, **Academic Article**, Journal of the Association for Information Science and Technology

**Investigation of luteal HCG supplementation in GnRH-agonist-triggered fresh embryo transfer cycles: a randomized controlled trial**  
Svenstrup, L. [et al.]  
2024, **Academic Article**, Reproductive Biomedicine Online

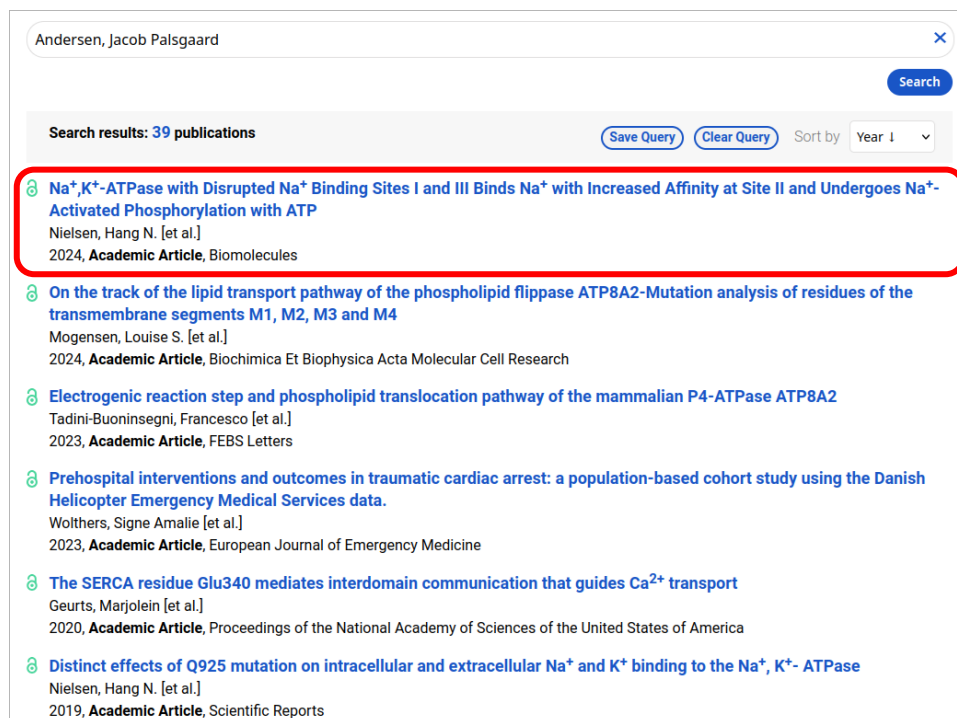
**Na<sup>+</sup>, K<sup>+</sup>-ATPase with Disrupted Na<sup>+</sup> Binding Sites I and III Binds Na<sup>+</sup> with Increased Affinity at Site II and Undergoes Na<sup>+</sup>-Activated Phosphorylation with ATP**  
Nielsen, Hang N. [et al.]  
2024, **Academic Article**, Biomolecules

**On the track of the lipid transport pathway of the phospholipid flippase ATP8A2-Mutation analysis of residues of the transmembrane segments M1, M2, M3 and M4**  
Mogensen, Louise S. [et al.]  
2024, **Academic Article**, Biochimica Et Biophysica Acta Molecular Cell Research

**Phenol as proton shuttle and buffer for lithium-mediated ammonia electrosynthesis**  
Fu, Xianbiao [et al.]  
2024, **Academic Article**, Nature Communications

**Long-term continuous ammonia electrosynthesis**  
Li, Shaofeng [et al.]  
2024, **Academic Article**, Nature

Figure A2.1: [Search for "Andersen, Jens Peter"](#) in global data search module.



Andersen, Jacob Palsgaard

Search

Search results: 39 publications

Save Query Clear Query Sort by Year ↓

**Na<sup>+</sup>, K<sup>+</sup>-ATPase with Disrupted Na<sup>+</sup> Binding Sites I and III Binds Na<sup>+</sup> with Increased Affinity at Site II and Undergoes Na<sup>+</sup>-Activated Phosphorylation with ATP**  
Nielsen, Hang N. [et al.]  
2024, **Academic Article**, Biomolecules

**On the track of the lipid transport pathway of the phospholipid flippase ATP8A2-Mutation analysis of residues of the transmembrane segments M1, M2, M3 and M4**  
Mogensen, Louise S. [et al.]  
2024, **Academic Article**, Biochimica Et Biophysica Acta Molecular Cell Research

**Electrogenic reaction step and phospholipid translocation pathway of the mammalian P4-ATPase ATP8A2**  
Tadini-Buoninsegni, Francesco [et al.]  
2023, **Academic Article**, FEBS Letters

**Prehospital interventions and outcomes in traumatic cardiac arrest: a population-based cohort study using the Danish Helicopter Emergency Medical Services data.**  
Wolthers, Signe Amalie [et al.]  
2023, **Academic Article**, European Journal of Emergency Medicine

**The SERCA residue Glu340 mediates interdomain communication that guides Ca<sup>2+</sup> transport**  
Geurts, Marjolein [et al.]  
2020, **Academic Article**, Proceedings of the National Academy of Sciences of the United States of America

**Distinct effects of Q925 mutation on intracellular and extracellular Na<sup>+</sup> and K<sup>+</sup> binding to the Na<sup>+</sup>, K<sup>+</sup>-ATPase**  
Nielsen, Hang N. [et al.]  
2019, **Academic Article**, Scientific Reports



Figure A2.2: [Search for "Andersen, Jacob Palsgaard"](#) in global data search module.

3 versions available: [Clarivate](#) [Elsevier](#) [Digital Science](#)

Article, 2024

**Na<sup>+</sup>,K<sup>+</sup>-ATPase with Disrupted Na<sup>+</sup> Binding Sites I and III Binds Na<sup>+</sup> with Increased Affinity at Site II and Undergoes Na<sup>+</sup>-Activated Phosphorylation with ATP**

BIOMOLECULES, ISSN 2218-273X, 2218-273X, Volume 14, 1, 10.3390/biom14010135

**Contributors**  
 Nielsen, Hang N. [1]; Holm, Rikke [1]; Sweazey, Ryan [2] [3]; Andersen, Jacob Palsgaard  0000-0003-0654-4300 [1]; Artigas, Pablo [2] [3]; Vilsen, Bente  0000-0002-4727-9382 (Corresponding author) [1]

**Affiliations**  
 [1] Aarhus Univ, Dept BioMed, DK-8000 Aarhus, Denmark [NORA names: AU Aarhus University; University; Denmark; Europe, EU; Nordic; OECD]; [2] Texas Tech Univ, Ctr Membrane Prot Res, Dept Cell Physiol & Mol Biophys, Hlth Sci Ctr, Lubbock, TX 79430 USA [NORA names: United States; America, North; OECD]; [3] Texas Tech Univ, Ctr Membrane Prot Res, Dept Cell Physiol & Mol Biophys, Hlth Sci Ctr, Lubbock, TX 79430 USA [NORA names: United States; America, North; OECD]

Figure A2.3: [Clarivate record in the Research Portal Denmark web interface](#) shows "Andersen, Jacob Palsgaard" (0000-0003-0654-4300) as author.

**Na<sup>+</sup>,K<sup>+</sup>-ATPase with Disrupted Na<sup>+</sup> Binding Sites I and III Binds Na<sup>+</sup> with Increased Affinity at Site II and Undergoes Na<sup>+</sup>-Activated Phosphorylation with ATP**

By [Nielsen, HN \(Nielsen, Hang N.\) \[1\]](#); [Holm, R \(Holm, Rikke\) \[1\]](#); [Sweazey, R \(Sweazey, Ryan\) \[2\]](#); [Andersen, JP \(Andersen, Jens Peter\) \[1\]](#); [Artigas, P \(Artigas, Pablo\) \[2\]](#); [Vilsen, B \(Vilsen, Bente\) \[1\]](#)

[View Web of Science ResearcherID and ORCID](#) (provided by Clarivate)

Source: BIOMOLECULES  
 Volume: 14 Issue: 1  
 DOI: 10.3390/biom14010135

Article Number: 135

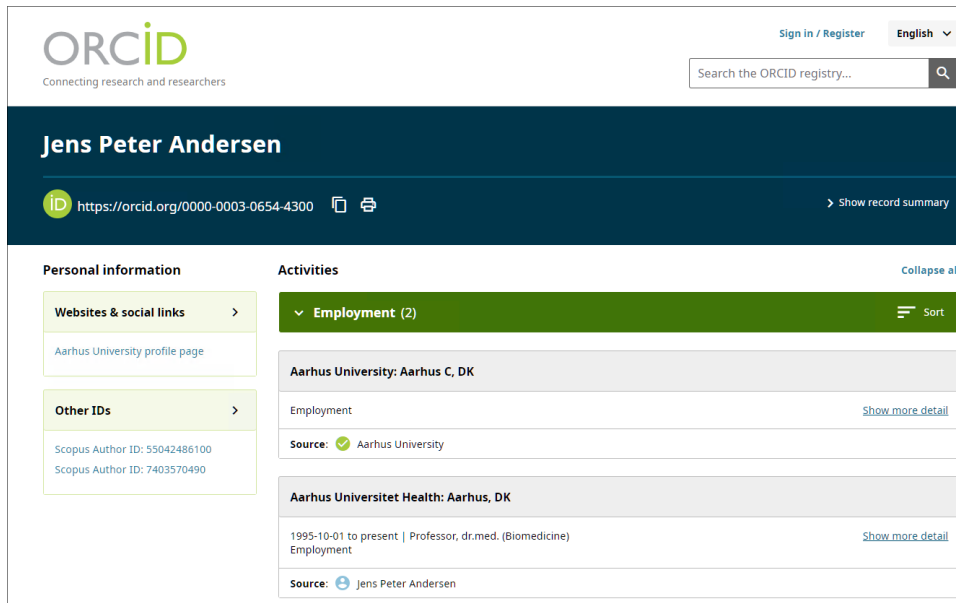
Published: JAN 2024

Indexed: 2024-02-06

Document Type: Article

Jump to: [↓ Enriched Cited References](#)

Figure A2.4: [Clarivate record in the Web of Science web interface](#) shows "Andersen, Jens Peter" as author.



ORCID  
Connecting research and researchers

Sign in / Register English

Search the ORCID registry...

## Jens Peter Andersen

<https://orcid.org/0000-0003-0654-4300> Show record summary

**Personal information**

**Websites & social links**

[Aarhus University profile page](#)

**Other IDs**

Scopus Author ID: 55042486100  
Scopus Author ID: 7403570490

**Activities** Collapse all

**Employment (2)** Sort

**Aarhus University: Aarhus C, DK**

Employment [Show more detail](#)

Source: Aarhus University

**Aarhus Universitet Health: Aarhus, DK**

1995-10-01 to present | Professor, dr.med. (Biomedicine) [Show more detail](#)

Employment

Source: Jens Peter Andersen

Figure A2.5: [0000-0003-0654-4300](https://orcid.org/0000-0003-0654-4300) in the ORCID web interface shows "Andersen, Jens Peter".