Ricgraph: A flexible and extensible graph to explore research in context from various systems

Rik D.T. Janssen, Utrecht University, Heidelberglaan 8, 3584 CS Utrecht, the Netherlands, r.d.t.janssen@uu.nl, ORCID 0000-0001-9510-0802

Abstract

Ricgraph, also known as Research in context graph, enables the exploration of researchers, teams, their results, collaborations, skills, projects, and the relations between these items.

Ricgraph can store many types of items into a single graph. These items can be obtained from various systems and from multiple organizations. Ricgraph facilitates reasoning about these items because it infers new relations between items, relations that are not present in any of the separate source systems. Ricgraph is flexible and extensible, and can be adapted to new application areas.

In this article, we illustrate how Ricgraph works by applying it to the application area research information.

Keywords

- data harvesting
- data enrichment
- data linkage
- linked data
- knowledge graph
- metadata

Metadata

Nr	Code metadata description	
C1	Current code version	v1.14
C2	Permanent link to code/repository used for this code version	https://github.com/UtrechtUniversity/ricgraph
C3	Permanent link to reproducible capsule	https://doi.org/10.5281/zenodo.7524314
C4	Legal code license	MIT License
C5	Code versioning system used	Git
C6	Software code languages, tools and services used	Python Neo4j for graph database backend
C7	Compilation requirements, operating environments and dependencies	Python ≥ 3.7, flask, markupsafe, numpy, pandas, py2neo, pyalex, requests, ratelimit, xmltodict, sickle
C8	If available, link to developer documentation/manual	https://github.com/UtrechtUniversity/ricgraph/blob/master/README.md
C9	Support email for questions	https://github.com/UtrechtUniversity/ricgraph/issues

1 Motivation and significance

Consider these use cases:

a. As a journalist, I want to find researchers with a certain skill and their publications, so that I can interview them for a newspaper article.
b. As a librarian, I want to enrich my local research information system with research results that are in other systems but not in ours, so that we have a more complete view of research at our university.
c. As a researcher, I want to find researchers from other universities that have co-authored publications written by the co-authors of my own publications, so that I can read their publications to find out if we share common research interests.

These use cases use different types of information (called “items” in this article): researchers, skills, publications, etc. Most often, these types of information are not stored in one system, so the use cases may be difficult or time-consuming to answer.

In this article, we present Ricgraph, also known as Research in context graph. Ricgraph is software that is about relations between items. These items can be collected from various source systems and from multiple organizations. We explain how Ricgraph works by applying it to the application area *research information*. The use cases above are from this application area. We show the insights that can be obtained by combining information from various source systems, insight arising from new relations that are not present in each separate source system.

Research information is about anything related to research: research results, the persons in a research team, their collaborations, their skills, projects in which they have participated, as well as the relations between these entities. Examples of *research results* are publications, data sets, and software.

Although this article illustrates Ricgraph in the application area research information, the principle “relations between items from various source systems” is general, so Ricgraph can be used in other application areas.

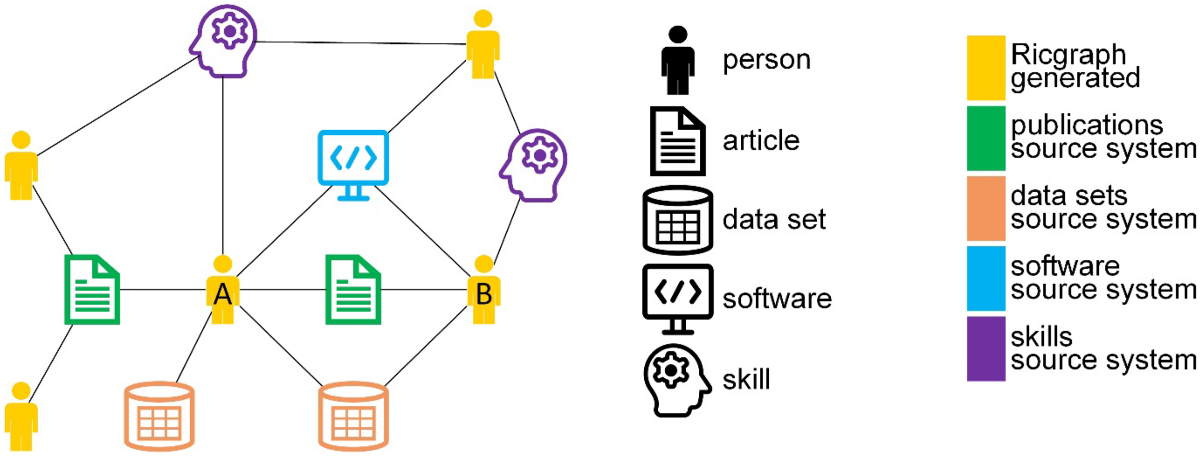
1.1 Main contributions of Ricgraph

1.1.1 Contribution 1: Ricgraph can store many types of items in a single graph

Ricgraph helps users to determine and categorize the important information (items) in a source system, and helps to determine the relevant relations between these items for a certain application area. Ricgraph only needs an identifiable item with a relation to one or more other items. If that is the case, the items and their relations can be added to Ricgraph.

Ricgraph uses a graph to model items (nodes) and their relations (edges). It uses a graph because context is close: in a graph, the items that are directly related are neighbors, only one “step” away from each other.

Ricgraph only stores metadata (information describing an item, such as name, category, value, title, year, link), not the objects they refer to (such as PDF files or data sets). Figure 1 shows an example.



(a)
Example graph

(b)
Items

(c)
Source systems

Figure 1: Example of a graph. The lines between the items represent the relations between those items.

For the application area research information, “items” and “relations” translate to:

- examples of items:
    - persons, their identities, their skills;
    - (sub-)organizations, e.g., teams, units, departments, faculties, universities;
    - research results, e.g., publications, data sets, software;
    - grants, projects;
    - and any other type that is interesting for an application area.
- examples of relations (connections between items, with symbol ↔):
    - person ↔ publication: a person has contributed to a publication;
    - person ↔ skill: a person has a skill;
    - person ↔ person: a person collaborates with someone else;
    - person ↔ (sub-)organization: a person is part of a (sub-)organization.

1.1.2 Contribution 2: Ricgraph harvests multiple source systems into a single graph

Ricgraph obtains items and their relations from a source system in a process called *harvesting*. Harvesting can be done for more than one source system. These source systems may span multiple organizations. Ricgraph will ensure that all items and relations of all harvested source systems will be added to one single graph.

Since every source system has its own harvest script, harvesting can be tailored to accessibility or peculiarities of that source system. So, a harvest script can get metadata from a source system only accessible in an organization, or from a system that does not have a standard interface for harvesting. Since users can create their own harvest scripts (or reuse existing scripts), it is possible to include local or sensitive data in Ricgraph, and combine these with publicly available data.

For the application area research information, Ricgraph includes harvest scripts for five research information systems that are used by many academic organizations in Europe. This makes it easy for organizations or researchers to get started with Ricgraph. In this article we will use four example source systems (Figure 1(c)):

- a publications source system, containing persons, publications, and (sub-)organizations;
- a data sets source system, containing persons and data sets;
- a software source system, containing persons and software;
- a skills source system, containing persons and skills.

1.1.3 Contribution 3: Ricgraph Explorer is the exploration tool for Ricgraph

Ricgraph provides an exploration tool, so users do not need to learn a graph query language. This tool is called Ricgraph Explorer. It can be customized as needed for a certain application area.

For the application area research information, Ricgraph Explorer has several pre-build queries, each with its own button, for example:

- find a person, a (sub-)organization, a skill;
- when a person has been found, find its identities, skills, research results.

1.1.4 Contribution 4: Ricgraph facilitates reasoning about items because it infers new relations between items

Ricgraph infers new relations between items when it adds items and relations from multiple source systems. For example, source system A has *item1* ↔ *item2* and source system B has *item2* ↔ *item3*. Adding these to Ricgraph results in *item1* ↔ *item2* ↔ *item3*, so one can traverse the graph from *item1* to *item2* to *item3*. This means there is a new inferred relation from *item1* to *item3*, which was neither in source system A, nor in source system B. This facilitates reasoning about all items, irrespective of where they originate from. See Figure 1(a): items with different colors (i.e., from different source systems) are connected.

For the application area research information, an example of such an inferred relation is:

- from the skills source system: skill ↔ person;
- from the publication source system: person ↔ publication;
- the inferred relation is: a person with a skill has written a publication.

1.1.5 Contribution 5: Ricgraph can be tailored for an application area

Every application area can be different. Ricgraph can be tailored to that application area by changing the harvesting or exploration part. Since Ricgraph is written in Python, someone who can program in Python can do that.

1.2 How to use Ricgraph

To use Ricgraph, users first needs to decide which source systems to harvest. Then, for each system, determine the relevant items and relations. For some source systems, Ricgraph provides harvest scripts. For others, new harvest scripts have to be created. Then this person runs the harvest scripts for those systems, and data will be imported in Ricgraph and will be combined automatically with items which are already there.

Next, Ricgraph can be explored using Ricgraph Explorer, a web-based tool. If the application area is research information, it can be used out of the box. Otherwise, it might be necessary to change parts of it.

1.3 Related work: other graphs with research in context

Other graphs with research in context include:

- OpenAIRE graph: harvests research information from thousands of sources. Their graph is quite different to Ricgraph because it is much larger. Ricgraph is extensible and flexible and runs on a small computer.
- Freya PID graph: a graph to contain identities. It was a proof of concept around 2019, development has stopped. It is less extensible and flexible than Ricgraph.
- EOSC research discovery graph and EOSC PID graph: development has not yet started.

2 Software description

2.1 Ricgraph architecture

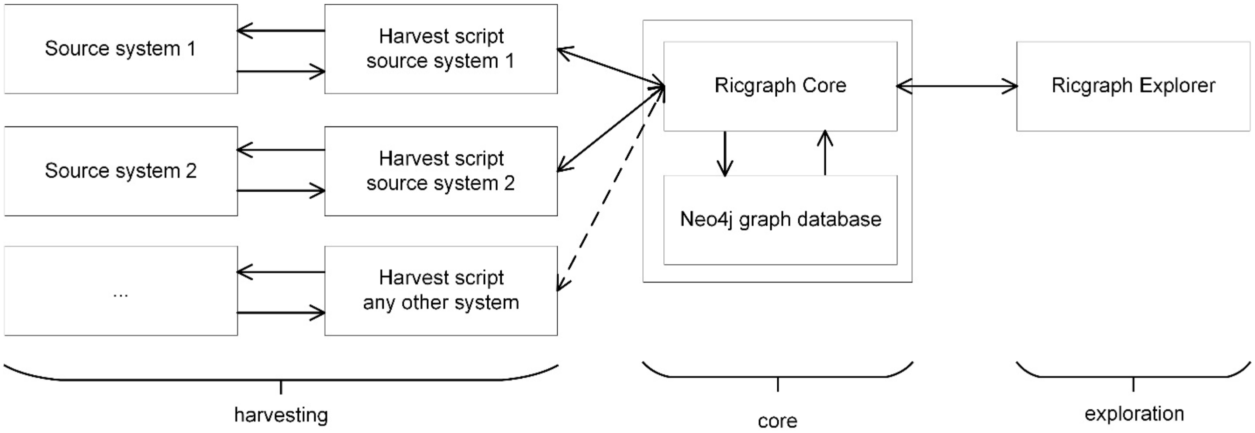


Figure 2: Ricgraph architecture.

Figure 2 shows the architecture of Ricgraph. It consists of three parts, to be explained in the next section:

- harvesting: connects source systems to the core;
- core: offers library calls to do the harvesting, and calls for exploration. It also connects to the Neo4j graph database.
- exploration: allows to explore the graph.

2.2 Ricgraph software functionalities

2.2.1 Ricgraph Core

Ricgraph uses the Neo4j graph database as backend to store nodes and edges. Neo4j is well known and offers a free to use version. Ricgraph Core is written in Python and consists of various function calls that allow to create, read, update, and delete nodes and edges in the Neo4j graph database. These calls are also used to explore the graph.

Items and their relations are inserted by specifying a list of two items that have a relation to each other. For every *item1* ↔ *item2*, the call to Ricgraph Core is to insert the pair [*item1*, *item2*]. If one or both of *item1* or *item2* are a person identifier, Ricgraph uses a special “in between” node called *person-root* node. This node “represents” a person. For more details, see section 3.2.

2.2.2 Ricgraph harvest scripts

Every harvest script is structured as follows:

- extract data from a source system;
- optional: process or combine (transform) the harvested data, e.g., combine a field with a first name and a field with a last name to one field representing a full name;
- transform the data to node pairs, load in Ricgraph.

Ricgraph includes scripts to harvest research information from five source systems. It is straightforward to create scripts for new sources.

2.2.3 Ricgraph Explorer

Ricgraph Explorer is a Python Flask application. It is used to explore the graph obtained by harvesting the source systems. Ricgraph Explorer has been built in such a way that it can be adapted to a specific application area.

2.2.4 Ricgraph can run on most modern computers

The development of Ricgraph has been done on a reasonable sized (in memory and disc space) modern laptop. Harvesting and exploration can be done on a laptop. Also, a large infrastructure such as SURF Research Cloud has been used.

3 Illustrative example

There are several challenges in combining research information from various sources. This section elaborates on one example and illustrates how Ricgraph solves it.

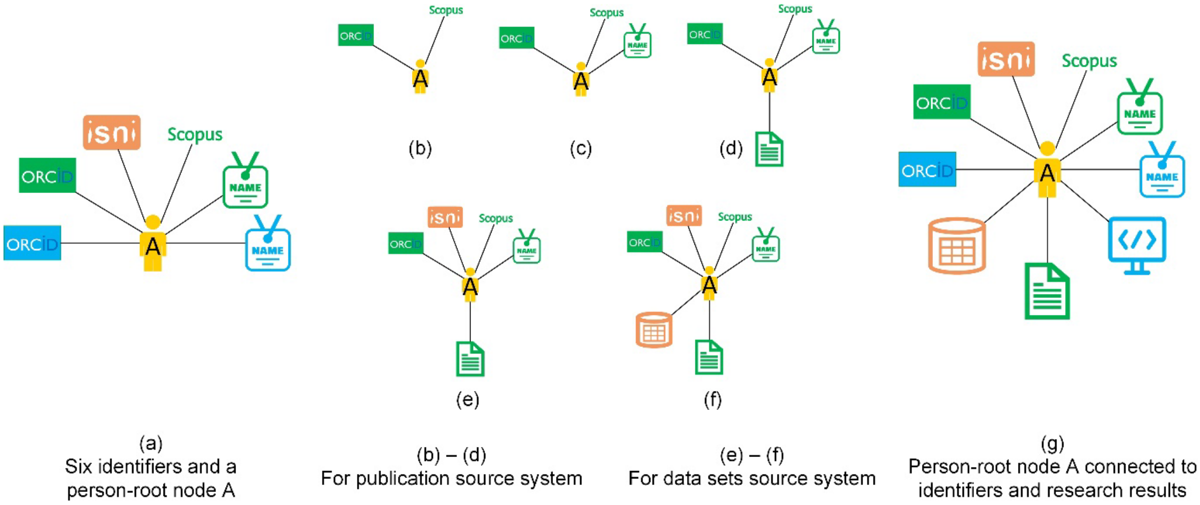
3.1 Items may have several identifiers, and some vary over time

To be able to connect items from various source systems, it is necessary to have *identifiers*, and preferably *persistent identifiers*. Persistent identifiers are long-lasting references to persons or research results. Examples are identifiers for objects (including research results, DOI) and for organizations (ROR). For persons, there are numerous identifiers:

- persistent identifiers: e.g., ORCID, ISNI;
- identifiers assigned by an organization: e.g., email address, employee ID;
- identifiers assigned by a publisher: e.g., Scopus ID;
- names: a person can use different spellings for their name.

Some of these identifiers vary over time (such as organization email address), or persons have more than one identifier of the same type (such as Scopus ID).

3.2 Ricgraph connects identifiers for the same person using a *person-root* node



(a)
Six identifiers and a
person-root node A

(b) – (d)
For publication source system

(e) – (f)
For data sets source system

(g)
Person-root node A connected to
identifiers and research results

Figure 3: Connecting identifiers for the same person using the person-root node.

Ricgraph connects identifiers for the same person by using a special node called *person-root* node. This person-root node "represents" that person. In Figure 3(a), A is the person-root node. A has six identifiers from three source systems (the colors): two ORCIDs, one ISNI, one Scopus ID, and two NAMEs (with different spellings). Research results from a person will be connected to this person-root node.

3.3 Example: insert a publication, a data set, and software from person A in Ricgraph

This example inserts a publication, a data set, and software from three source systems in Ricgraph. Each source system has one or more identifiers for person A. Since some identifiers occur in more than one source system, it is possible to connect everything to the same person-root node.

For the publication source system, green color: insert a publication from person A in Ricgraph. Person A has three identifiers: an ORCID1, a SCOPUS_ID, and a NAME1. Nodes are inserted in pairs.

- Insert [ORCID1, SCOPUS_ID]. → *Effect: person-root node A created; ORCID1 node and SCOPUS_ID node created and connected via person-root node A. See Figure 3(b).*
- Insert [ORCID1, NAME1]. → *Effect: NAME1 node created and connected to already existing ORCID1 node via already existing person-root node A, as in Figure 3(c).*
- Insert [ORCID1, publication]. → *Effect: publication node created and connected to ORCID1 node via the person-root node, Figure 3(d).*
- Done.

For the data sets source system, orange color: insert a data set from person A in Ricgraph. This source system has three identifiers for this person: an ORCID1 (as above), an ISNI (new) and a NAME1 (as above).

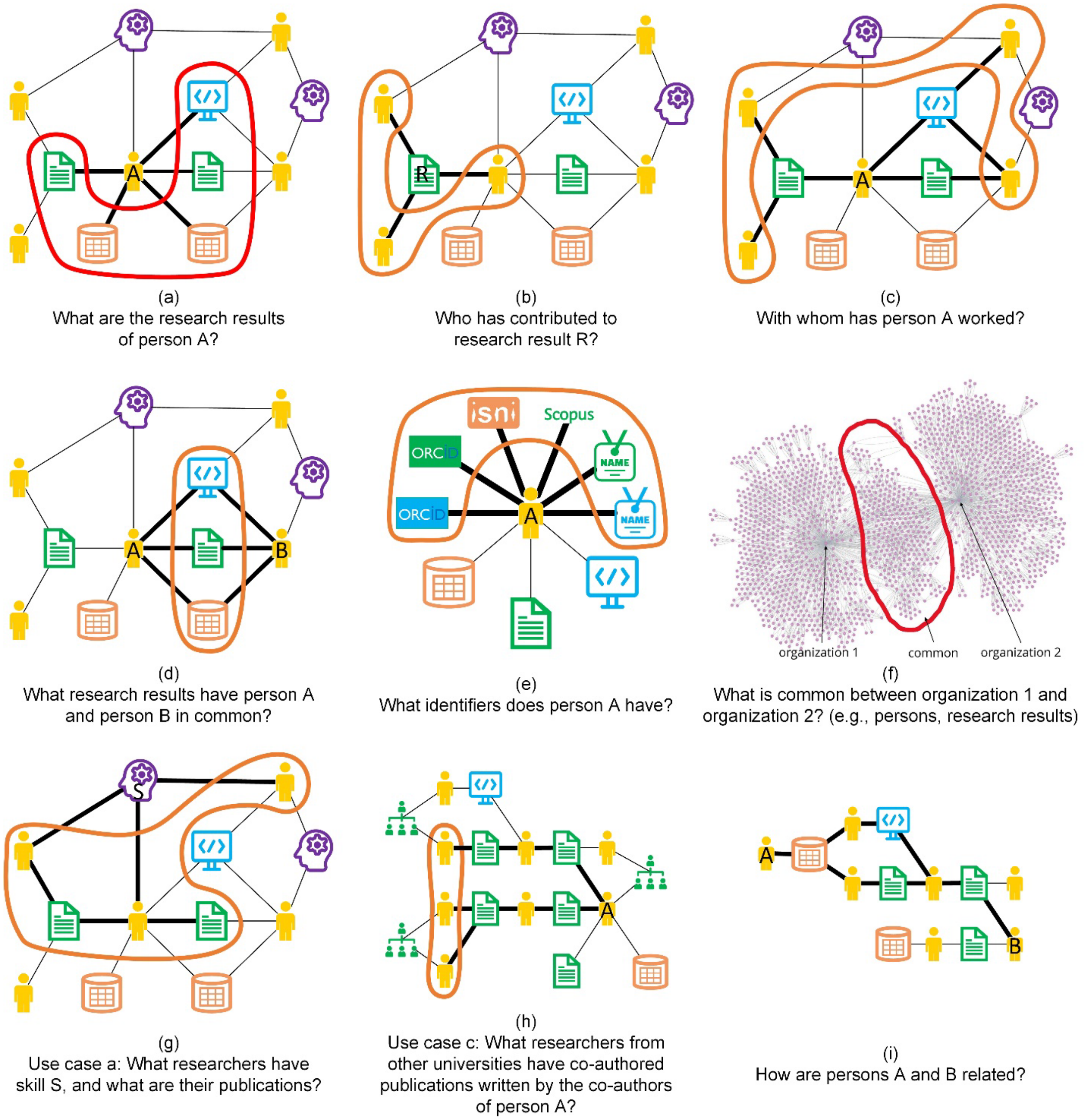
- Insert [ORCID1, ISNI]. → *ISNI node created and connected to already existing ORCID1 node, Figure 3(e).*
- Insert [ORCID1, NAME1]. → *no action, ORCID1 and NAME1 already exist.*
- Insert [ORCID1, data set]. → *data set node created and connected, Figure 3(f).*
- Done.

For the software source system, blue color: insert software from person A in Ricgraph. This source system has three identifiers: an ORCID2 (new, different than above), an ISNI (as above) and a NAME2 (new, spelled different than above).

- Insert [ORCID2, ISNI]. → *ORCID2 node created and connected.*
- Insert [ORCID2, NAME2]. → *NAME2 node created and connected.*
- Insert [ORCID2, software]. → *software node created and connected.*
- Done. See Figure 3(g) for the resulting graph.

4 Impact

4.1 Research questions that can be pursued using Ricgraph



(a)
What are the research results of person A?

(b)
Who has contributed to research result R?

(c)
With whom has person A worked?

(d)
What research results have person A and person B in common?

(e)
What identifiers does person A have?

(f)
What is common between organization 1 and organization 2? (e.g., persons, research results)

(g)
Use case a: What researchers have skill S, and what are their publications?

(h)
Use case c: What researchers from other universities have co-authored publications written by the co-authors of person A?

(i)
How are persons A and B related?

Figure 4: Examples of research questions that can be answered using Ricgraph, including two use cases from section 1. For symbols and colors see Figure 1.

Figure 4 shows several research questions that can be answered using Ricgraph, including two use cases from section 1. The red line shows the answer to the question in the caption of the sub figure. These answers seem very straightforward, however, they are only so because Ricgraph is using a graph. A graph is easy to understand and facilitates reasoning, making it a convenient tool for discussing use cases and research questions.

A future extension could be to collect the keywords of research results (most publications, data sets and software have keywords). After mapping those keywords on a standardized subject list, these subjects can be added to Ricgraph and connected to their research results. By searching for one or more subjects, one can find research results and their contributors that are related. This is another method for finding researchers with common research interests than in use case c from Figure 4(h).

4.2 Using Ricgraph changes the practice of users

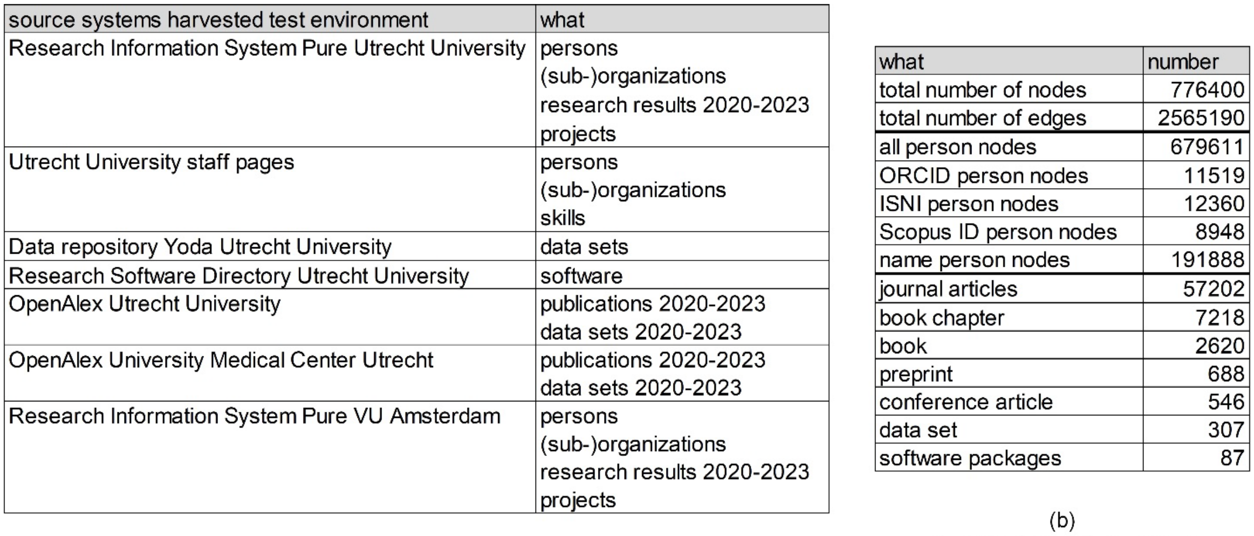
We have noticed that by being able to pose questions as in the previous section, and subsequently by being able to traverse the graph from an answer obtained, our users have gained insight in the research information landscape at our university. The use cases at the start of section 1 and in the previous section are a result of these insights.

Ricgraph can also help in organizing support. For example, suppose an organization has an open data policy. Using Ricgraph, users may observe that in some parts of that organization only a few data sets are shared compared to other parts. This might give rise to asking persons from the first organization if they would like to have help in sharing their data sets.

4.3 Widespread use of Ricgraph

The use of Ricgraph in a short timeline:

- December 2022: Ricgraph development started. In December 2023 there were 200 commits in GitHub, indicating active development.
- March – August 2023: Ricgraph has been used in the NWO PID graph pilot project, with SURF and six Dutch universities.
- June, July 2023: Ricgraph has been used to test the NWO NWOpen-API, the Elsevier Data Monitor and the Elsevier Grant Award API.
- October 2023: Ricgraph has been presented at the Pure International Conference in Dubrovnik, Croatia [1]. This has led to international interest.
- December 2023: See Figure 5 for statistics how we use Ricgraph at Utrecht University in our test environment.



source systems harvested test environment	what
Research Information System Pure Utrecht University	persons (sub-)organizations research results 2020-2023 projects
Utrecht University staff pages	persons (sub-)organizations skills
Data repository Yoda Utrecht University	data sets
Research Software Directory Utrecht University	software
OpenAlex Utrecht University	publications 2020-2023 data sets 2020-2023
OpenAlex University Medical Center Utrecht	publications 2020-2023 data sets 2020-2023
Research Information System Pure VU Amsterdam	persons (sub-)organizations research results 2020-2023 projects

(a)
Source systems harvested

what	number
total number of nodes	776400
total number of edges	2565190
all person nodes	679611
ORCID person nodes	11519
ISNI person nodes	12360
Scopus ID person nodes	8948
name person nodes	191888
journal articles	57202
book chapter	7218
book	2620
preprint	688
conference article	546
data set	307
software packages	87

(b)
Number of nodes of various categories. Note that these do not add up because not all categories are included.

Figure 5: Ricgraph statistics as of December 2023.

5 Conclusions

With Ricgraph, it is possible to create a single graph from research information that is stored in various source systems. This can be done for multiple organizations. Ricgraph allows users to explore this graph and discover previously unknown relations. This gives a lot of insight to our users.

Some of the lessons learned are:

- a graph is a very useful data structure to explore research information;
- it is very convenient to have software containing research information that can be adapted easily to someone’s need;
- identifiers are a prerequisite, and they should be resolvable to each other.

References

[1] Rik D.T. Janssen & Arjan Sieverink. (2023). Ricgraph: showcasing research in context using Pure and other sources. Pure International Conference 2023, Dubrovnik, Croatia, https://doi.org/10.5281/zenodo.10057997.