VIPER: The Visual Project Explorer

Final Report by Open Knowledge Maps

Case Study

The Visual Project Explorer (VIPER) is a unique open science application created by Open Knowledge Maps that provides visual overviews of research projects indexed by OpenAIRE. It enables funders, institutions and researchers to systematically explore a project's output, and to understand its impact in different areas. VIPER exploits a unique property of OpenAIRE data: the link between projects and publications and datasets.

In VIPER, each project is represented as a knowledge map: a visualization showing a topical clustering of a project's publications and datasets. Users can interact with the visualization by zooming into a particular cluster and inspect the underlying outputs in detail. The knowledge map can be scaled according to different metrics, including citation data and altmetrics. Therefore, VIPER enables users to keep track of the reception of a project in a multifaceted and contextualized way that goes beyond simple aggregate numbers and rankings.
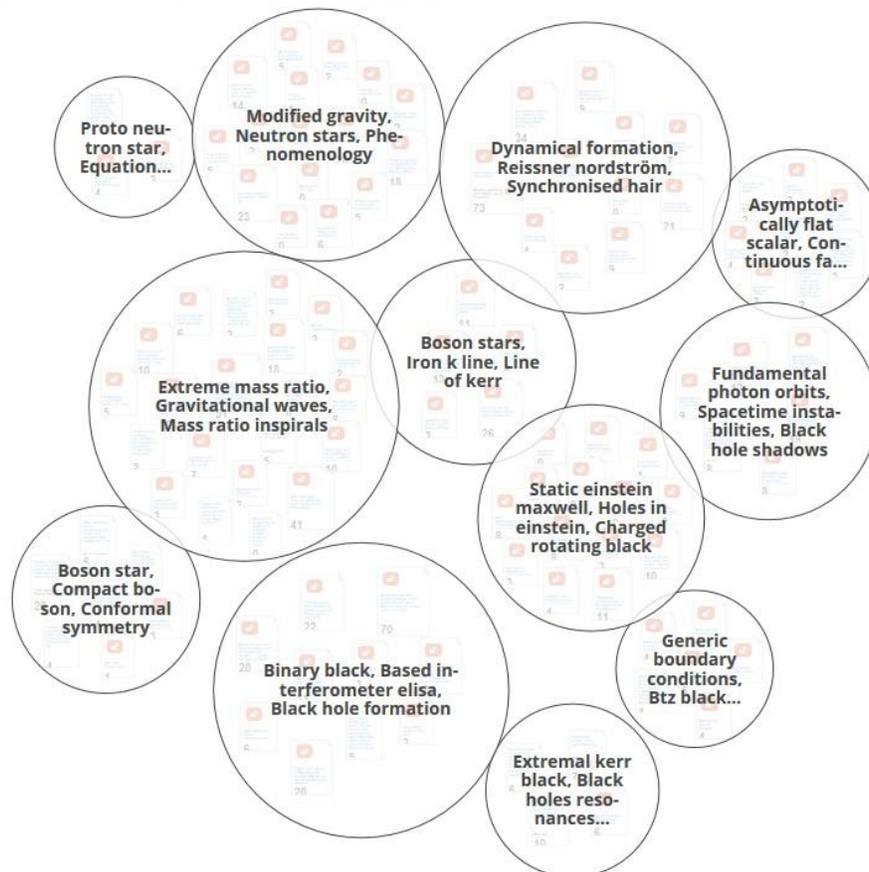
While VIPER can be used on its own, project representations can also be embedded on other websites, e.g. on project websites or within institutional dashboards using a simple HTML snippet.

The user tests conducted in the scope of this project have shown that VIPER is perceived as a visual guide to the world of research projects. It has great potential for providing a better overview of research projects and thus improving the visibility of project results overall. Furthermore, VIPER will possibly also have a positive effect on the OpenAIRE database as it motivates users to keep their project records in OpenAIRE updated.

In the user tests, we also identified four main use cases for VIPER:

1. As a tool for researchers to get overviews of research projects and identify relevant publications and datasets.
2. As an automatically updated dissemination page for projects and a way of keeping track of the reception of their outputs.
3. As a system for organisations to showcase their output and identify resources relevant for research communication.
4. As a way for funders to systematically compare funding input to output.

VIPER is accompanied by comprehensive dissemination efforts. The service is included on the Open Knowledge Maps website, which is visited by more than 30,000 users each month. We have also incorporated VIPER in our online communication efforts and in our offline workshops and talks. In addition, we suggest having joint webinars with OpenAIRE for the aforementioned user groups to introduce them to VIPER.

To realize VIPER, we have adapted and extended our existing open source visualization framework Head Start. New features include:

- A metrics component to enrich publication metadata with metrics from different sources. This is accompanied by a scaling functionality that rescales the map according to a given metric to enable a multi-faceted view on project outputs.
- An automatization component to automatically update project maps in certain intervals or when new resources have been added.
- An embed functionality to enable the inclusion of VIPER on external websites and in dashboards.

We have also updated the R package ropenaire to the latest version of the OpenAIRE API in collaboration with rOpenSci, and we have developed a dedicated search interface for selecting projects for visualization.

The user tests also showed that there is a lot of potential for enhancing VIPER in the future. This includes further integration with OpenAIRE, such as a seamless integration of the linking service as well as the development of a visual project search, enabling the visual exploration of large amounts of research projects on a user-defined topic. In addition, we see great potential for integrating the Matchmaker recommendation service developed in another OpenAIRE Tender project in VIPER. Finally, as pointed out in the user tests, it would be useful to have an editing mode, where project representations can be modified, for example enabling enriching metadata or rearranging map items.

Table of Contents

1 Introduction to VIPER

[The Visual Project Explorer (VIPER)](#) is a unique open science application that enables funders, institutions and researchers to systematically explore a research project's output. Each project is represented as a knowledge map: a visualization showing a topical clustering of a project's resources. The knowledge map can be scaled according to different metrics, including citation data and altmetrics. Therefore, VIPER enables users to keep track of the reception of their project in a multifaceted and contextualized way.

VIPER is based on the OpenAIRE infrastructure (Manghi et al. 2012, Manghi et al. 2014). The application exploits a unique property of OpenAIRE data: the link between projects and publications and datasets. We use the [OpenAIRE API](#) to retrieve publications and datasets related to a project and visualize them with the award-winning open source mapping software [Head Start](#) (Kraker et al. 2016) The resulting maps are automatically updated by a processing component that queries the OpenAIRE API at regular intervals.

VIPER received funding by OpenAIRE under the [OpenAIRE Tender Calls for services](#), Lot II (Value-added services). VIPER addresses the following tender topics of the call:

- **Services to enhance scholarly communication:** VIPER enables and improves use cases and usage scenarios for funders, institutions, projects and researchers. See section 3 for an overview of usage scenarios.
- **Links to other infrastructures:** In VIPER, OpenAIRE data is being enriched with open citation data and altmetrics data. In addition, VIPER has been integrated in the Open Knowledge Maps website and infrastructure.
- **Application of text and data mining (TDM) methods:** VIPER builds on the award-winning open source knowledge mapping software Head Start, which applies TDM methods to metadata in order to create topical overviews of research outputs.
- **Improvement of the OpenAIRE scholarly communication graph:** VIPER includes functionality that sends users to the OpenAIRE service for linking research results to projects.
- **Visualization:** Project maps in VIPER are realized as interactive, web-based knowledge maps, following Shneiderman's principle of "overview first, zoom and filter, then details-on-demand".

1.1 User Interface

The user interface on https://openknowledgemaps.org/viper/ consists of three basic screens. On the first screen (see Figure 1), users can search for projects of different funding agencies using project metadata, such as keywords, grant ids, or project acronyms. After a click on Go, they are presented with a paginated list of projects related to their search. The projects in the search results list are enriched with the number of resources related each project.
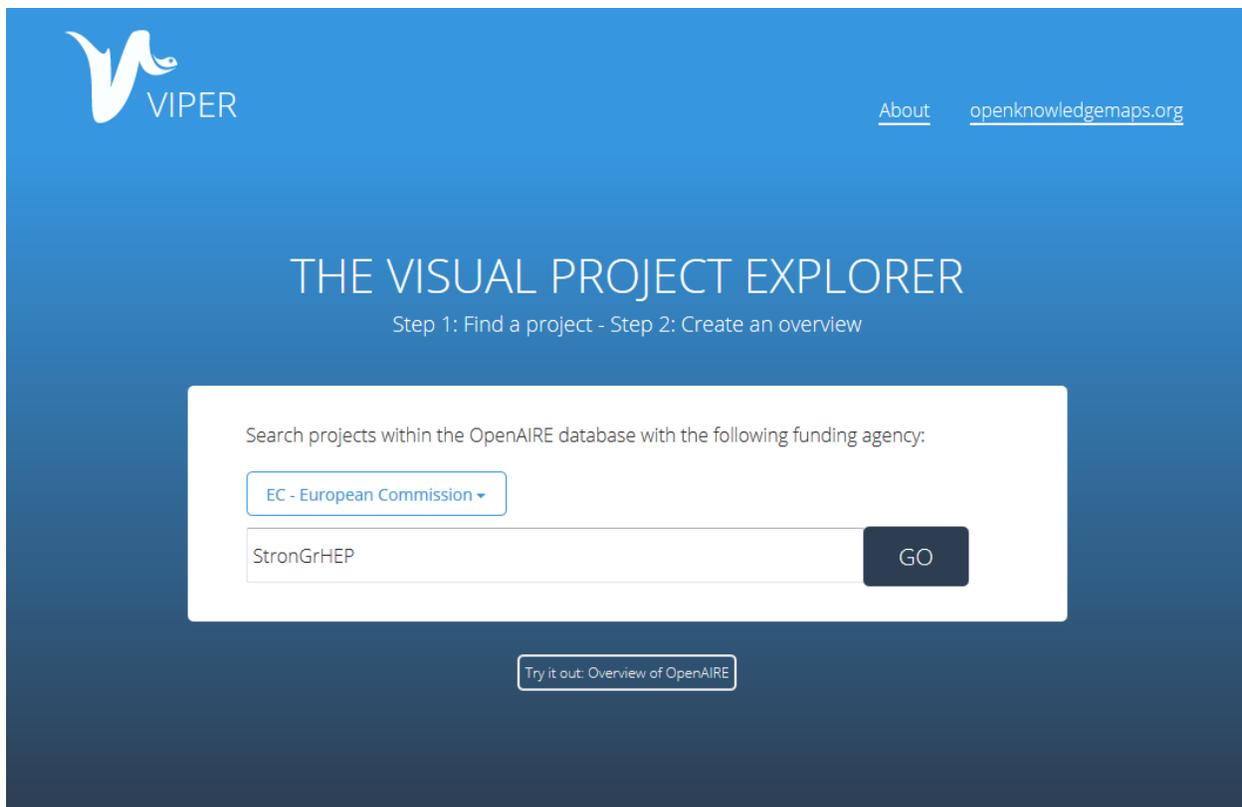
Figure 1: VIPER project search with search results list for the project acronym *StronGrHEP*

On selecting a relevant project, the user is redirected to a waiting screen while the knowledge map is created on the server. If it does already exist, the user is immediately redirected to the corresponding knowledge map.

Users are now presented with a knowledge map of the project (see Figure 2): a visual representation of projects' outputs (publications and datasets). The map on the left hand side shows the main topics of the project as bubbles. Project resources have been assigned to sub areas. The closeness of the sub-areas indicates subject similarity, grouping those bubbles together that have a similar vocabulary.

The list on the right hand side includes the same information in sequential order. The list can be sorted by different metadata items and the metrics included in the map. Furthermore, users are able to filter outputs according to a free text of their choosing, or by selecting a dedicated filter for open access resources, publications or datasets. These filters are also reflected in the map, meaning that it shows only the resources that meet the current filter criteria.

VIPER offers additional information on the project and the map, including project runtime, number of resources, and grant id, both in the title and the "more info" popup next to it.
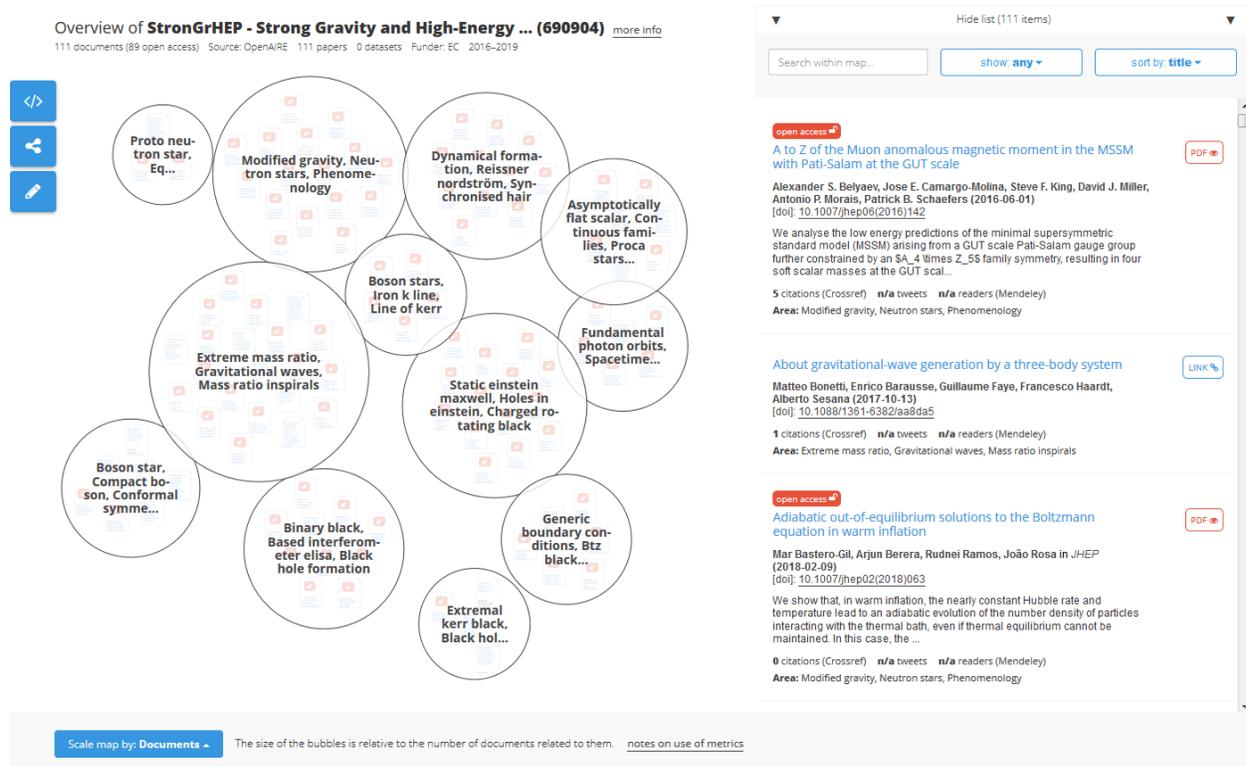
Figure 2: Knowledge map for the EC-funded project *StronGrHEP*

Once users click on a bubble, they are presented with the outputs related to this sub-area in detail (Figure 3). Publications and datasets associated to the topic are shown in both the bubble and the list. Bubbles and papers can be scaled according to several metrics, providing an indication as to which outputs have gathered more attention/impact in different services. Users can choose a metric in the toolbar at the bottom of the list, which also provides additional information on the metrics and their use. Available metrics are citations (data source: Crossref) as well as tweets and Mendeley readers (data source: Altmetric).

By clicking on a resource in the map or the list, users can inspect the full metadata of that item. In addition, users can view open access publications within the same interface as they are retrieved and shown in an overlay. Alternatively, users can follow the link to the landing page of the output.

VIPER also offers a number of complementary features via the toolbar on the left hand side. The button at the top leads to a popup that shows an embed code, for reuse of the map on websites and in dashboards. The button in the middle opens Add2Any, which facilitates sharing in a large number of social media services. Finally, the bottom button opens a popup explaining how project resources can be added using the respective OpenAIRE service.
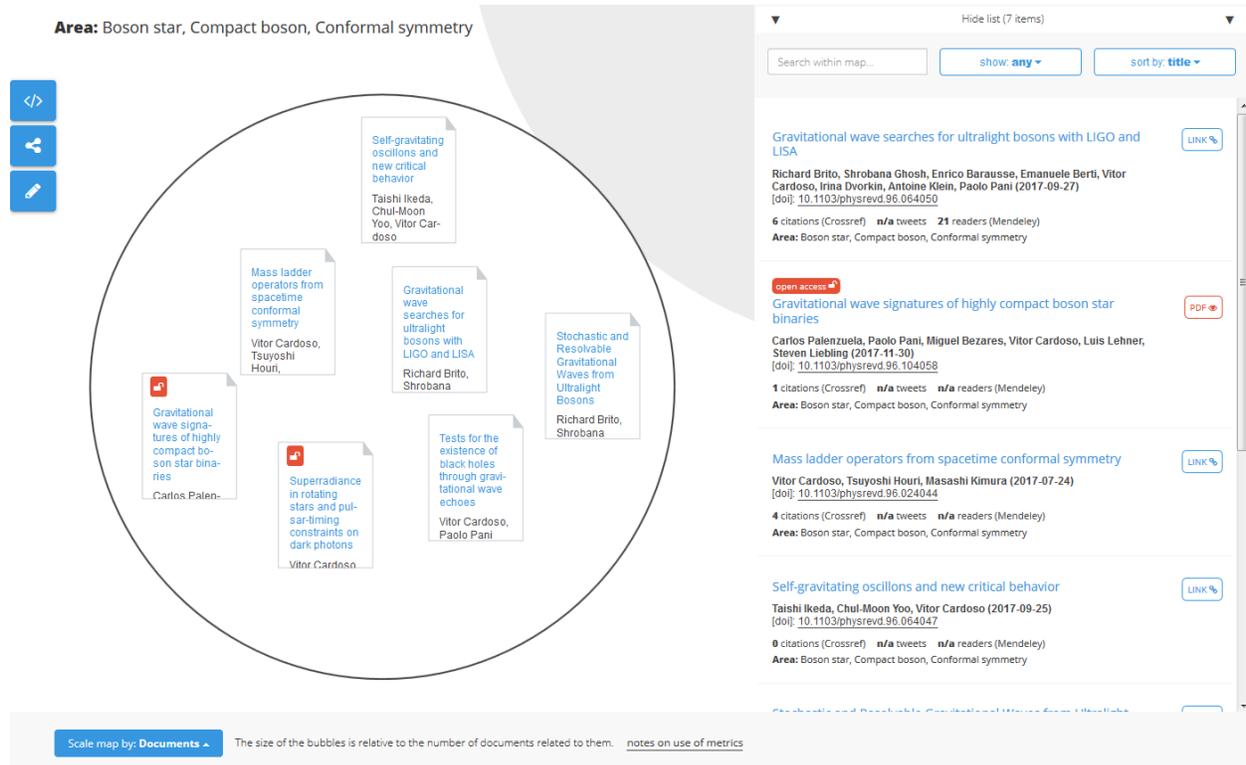
Figure 3: Detailed view of the sub-area "Boson star, Compact boson, Conformal symmetry" in the knowledge map for the EC-funded project *StronGrHEP*

1.2 Metrics

When the idea of altmetrics was conceived, there was the notion that they would be "tomorrow's filters" (Priem et al. 2010), enabling a multi-faceted and contextualized way of discovering relevant literature. These ideas have indeed been picked up, but mostly when it comes to evaluation. ImpactStory Profiles, for example, provides a multi-faceted view on the reception of a single researcher's output. Altmetric Badges have a similar concept for single research outputs. When it comes to exploration and discovery, however, most interfaces still do not provide much beyond ranked lists.

Ranked lists have a big cognitive problem: they are not able to convey ratios between the ranked items. Therefore, an output with 100 citations is easily perceived as superior to an output with 99 citations, even though this difference (a) negligible in reality and (b) may even be an artefact of the way data is collected and normalized by the respective system. In addition, most discovery system rely on a single metric (mostly citations) and therefore provide a one-dimensional view on research outputs. When systems go beyond a single metric, they often use compound scores, such as the Altmetric score, even though these are usually irreproducible and use an arbitrary weighting system that significantly impacts the final score (Kraker et al. 2015).

VIPER takes a different approach to metrics to mitigate the issues mentioned above and support discovery in a more meaningful way. While it still provides a list that can be ranked according to a single metric, it provides much more context to this ranking in the visualization in that papers and bubbles are scaled according to the number of mentions. They are therefore guiding researchers, which sub-areas and which outputs have received the most attention - but as the user is not able to tell the difference between the size of an output with 100 citations and 99 citations, it does so in a more balanced way. In addition, the difference in size between an output with 100 citations and one with 0 citations is not as dramatic as in a ranked list (see Figure 4).

Area: Dynamical formation, Reissner nordström, Synchronised hair

Explosion and final state of an unstable Reissner-Nordstrom black hole

Nicolas Sanchis-Gual ,Juan Car-
34 citations

Violations of the Kerr and Reissner-Nord-
1 citations

Inside black holes with synchronized hair

Yves Brihaye, Carlos Herdeiro, Eu-
21 citations

Dynamical formation of a Reissner-Nordström black hole
9 citations

KerrNewman black holes with scalar hair

Jorge F.M. Delgado, Carlos
16 citations

Remarks on the Taub-NUT solution in
0 citations

Kerr black holes with Proca hair

Carlos Herdeiro, Eugen Radu, Helgi Rúnarsson

in (2016-06-29)

73 citations

Probing the universality of synchronised hair
1 citations

Numerical evolutions of spherical Proca stars

Nicolas San-
7 citations

Dynamical formation of a hairy black hole
2 citations

Kerr black holes with synchronised hair: an analytic model
**9** citations

Figure 4: Example of scaling by citations in the sub-area "Dynamical formation, Reissner nordström, Synchronised hair" of the knowledge map for the EC-funded project *StronGrHEP*

9

In addition, the sub-areas provide additional topical context. In the bubbles, outputs, which have received a lot of attention, are put together with related outputs that may also be interesting to the user. VIPER thus mitigates the effect that users may conclude after a quick scan of a ranked list, that a given output is the only relevant piece of literature for their current information need.

VIPER uses two metrics APIs: the [CrossRef REST API](), which exposes open citation information, and the [Altmetric API](), which exposes a number of altmetrics for use in research projects and by non-profits under the condition that metrics are not re-distributed in bulk. From the set of available metrics, VIPER currently offers three different metrics:

1. Citations as the most established indicator of scholarly reuse of scientific knowledge.
2. Mendeley readership as an indicator of scholarly reception that provides large coverage and is usually available much earlier than citations (Zahedi et al. 2017).
3. Tweets as an indicator that may also capture attention beyond the academic sphere (Bornmann 2015).

VIPER exposes the total count of an indicator, clearly differentiating between cases where the data provider reported zero occurrences of a particular metric and cases, where no record could be found (n/a). All indicators are reported without any weighting. The reason for this is that relevant weighting factors (e.g. for disciplines or for geographic regions) are not readily available via open APIs and could thus not be incorporated. The only available weighting factor would have been time; however, we decided against using it, as it may have given the impression that the reported numbers are ready for reuse in an evaluation. Instead, we have added the following note for users interested in evaluation:

"Metrics are provided for indicative purposes only and cannot be used as-is in an evaluation. If you intend to use the metrics provided for evaluative purposes, consider relevant guidelines such as the [Leiden Manifesto for Research Metrics]()."

With this statement, we want to encourage stakeholders in the scholarly communication system to conform to the latest bibliometrics standards and recommendations when conducting an evaluation - in short: the responsible use of metrics, which is also highlighted by the [Report of the European Commission Expert Group on Altmetrics]() and the [Metric Tide]().

2 User tests

In the context of this project, we also conducted a user test on the first prototype of VIPER. The objectives of the user test were (1) to get feedback for the second iteration of system design and development and (2) to better understand use cases for and the potential of VIPER.

2.1 Setup

The test took place from April 16th to 18th and involved three participants, each one from a relevant stakeholder group: (1) research funding, (2) research management, and (3) research. The

user tests were conducted in the form of interviews via Skype or Google Hangout. Sessions were recorded after a consent form was signed by the participants.

Each session started with a couple of warm-up questions about participants' roles at their institution and how these roles relate to research projects. Then we proceeded with a short overview of VIPER using screen sharing functionality. This was followed by a use case covering all relevant functionalities of VIPER. For this part, we asked participants to share their screen and complete a series of tasks given to them. Tasks included searching for and creating a specific map, locating information and entities on the map and scaling the map with a different metric. We followed up this use case with evaluation questions on the usability of VIPER and its application to different usage scenarios.

2.2 Results

The following gives a condensed version of the results; a more extended version of the interview plan and results are available upon request.

In the user test, VIPER was generally very well received. Test participants repeatedly mentioned that the system was a good starting point to get an overview of a research project. They cited the following reasons for this assessment: (1) VIPER combines all data on a research project in a single interface and even enriches this data with metrics information and (2) VIPER enables users to see all the main project topics at a glance, which is usually hard to determine using a project's website or reports alone. The scaling feature was seen as very useful as it enables users to compare different metrics in one interface and helps finding relevant outputs more quickly. Furthermore, the feature to easily identify open access resources and the possibility of displaying them within the same interface was seen as a massive time saver. Finally, test users mentioned that VIPER will motivate them to link resources to projects they are working in. On a more systematic level, VIPER may mitigate the issue that with the end of a research project the data stream usually also ends, even though research outputs will be published after that. Having a tool like VIPER would be a great incentive to provide OpenAIRE with more data and more frequent updates.

Test participants also named further use cases for VIPER beyond getting an overview of individual research projects:
- For projects: showcasing a project's output on the project website, effectively using it as an automatically updated dissemination page.
- For research organisations: showcasing an institution's or department's research output in a concise and contextualized form by displaying related research projects' overviews on the website.
- For funders: using VIPER as an internal tool that connects funding input to output, which is continuously updated. In addition, it would be interesting for individual funding agencies to include a subset of VIPER on their own websites to enable stakeholders to research all projects funded by that agency.

- In general, VIPER was named a useful for research communication. In particular, it was seen as an efficient way to determine project outputs that have received attention in the research community and beyond and may thus be especially interesting in communication measures.

In terms of feedback, we received a lot of useful information on how to improve the user interface and system. This included missing or ambiguous context information and the placement and design of certain elements of the map. For example, test users had trouble identifying datasets in the map and list. This problem was mitigated by distinguishing them more clearly from publications and by reducing the amount of design details for publications. Another issue was that the scaling toolbar was not easy to find as it was displayed too far below the map and was was displayed in neutral colors. This was mitigated by making sure that the toolbar would always appear on the same screen as the map and by highlighting the dropdown with a signal color. In addition, we added more information on the metrics and their provenance and use, which was of particular interest to the test users.

Furthermore, the test users also mentioned a number of features that they would like to see in VIPER. For example, they thought that an additional filter allowing them to display only resources with a certain property, e.g. open access outputs, would be very beneficial. As a result, we designed and implemented this feature in the second iteration of VIPER development. Test users also mentioned a number of features and enhancements, which were beyond the scope of this project:

- Providing additional types of clustering, e.g. clustering by discipline.
- Having knowledge maps for the project search as well, i.e. showing the results of the project search as a clustered overview of project metadata.
- Including VIPER overviews within the OpenAIRE interface on every project page.
- Providing a knowledge map over all outputs of an organisation to identify their overall impact.
- Enabling editing of metadata to improve data quality.
- Enabling editing of maps, both adding resources directly to the map and changing configuration and labelling of the map.
- Including further metrics for scaling.

3 Dissemination

VIPER is accompanied by comprehensive dissemination efforts. These efforts include:

1. Integration of the Visual Project Explorer on the Open Knowledge Maps website, which is visited by more than 30,000 users each month. VIPER has been added as an additional option on the main page: https://openknowledgemaps.org/viper

2. Dissemination within the international Open Knowledge Maps network and community, involving our multidisciplinary advisory board, our partners & supporters, as well as na-

tional & international networks. This includes dissemination of VIPER via the Open Knowledge Maps social media channels and our newsletter. This involves support of the dissemination efforts of OpenAIRE. Concrete actions are:

- 15 February: Provide information for the OpenAIRE page and blog
- 28 February: OpenAIRE announces tender call results (Link) - OKMaps retweets (Link)
- 20 March: OKMaps sends out newsletter with basic information on VIPER and OpenAIRE (Link)
- 20 March: OKMaps adds the information from the newsletter to the News page (Link) and releases Twitter and Facebook posts (Twitter, Facebook)
- 26 April: Teaser for the upcoming VIPER release sent out to the OKMaps advisory board.
- 15 May: Release of VIPER with a dedicated newsletter and supported by posts on social media and to the whole OKMaps community
- 20 May: Dedicated announcement of the new version of ropenaire with our partner rOpenSci.

3. Inclusion into our offline sessions: Open Knowledge Maps team and community members regularly hold workshops and talks. VIPER has already been included in presentations at the Open Science Barcamp 2018 in Berlin and in the context of our Enthusiasts Program. Upcoming events include Wikimania 2018 in Cape Town, the Annual Meeting of AG Universitätsverlage in Vienna and the Hierarchy of Hypotheses workshop in Hannover.

In addition, we suggest joint webinars with OpenAIRE to present VIPER to interested stakeholders.

4 Technical Documentation

4.1 Further development of Head Start

To realize VIPER, we adapted and extended the existing open source framework Head Start (Kraker et al. 2018). Head Start is a high-performance and high-availability software stack that is capable of automatically producing knowledge maps from a variety of data, including text, metadata and references using unsupervised machine learning techniques. The web-based frontend enables easy overviews of research topics, the identification of relevant concepts, and accessing open content. Head Start is the framework behind the popular search service on the Open Knowledge Maps website, which is visited by more than 30,000 users each month, who leverage the search service for their research, work, and studies. In addition, Head Start is used in several other applications and websites, including the OpenUP Hub, ZBW Lending History, and Conference Navigator 3 (CN3).

4.1.1 Backend

On the backend, our first step was to create a connector from Head Start's backend pipeline to the OpenAIRE API. As described in the next section, we first updated the existing R package ropenaire to the newest version of the OpenAIRE API and used it as a data client. Then, we created an interface from our backend pipeline to ropenaire.

VIPER therefore retrieves metadata of publications and datasets associated with a specific project via the ropenaire-package (see section 4.2). This data is then enriched with missing metadata and additional metrics. When DOIs are missing from the metadata of a publication, VIPER attempts to retrieve the publications' DOIs by querying crossref for matching titles. A missing DOI is accepted when a title does not surpass a Levenshtein distance of 0.06. For this step, we also improved the existing package rcrossref in collaboration with [rOpenSci](#) (see section 4.2). The work included speeding up requests to Crossref to make user experience faster, both for collecting Crossref metadata in general and for collecting citation counts. In the next step, the clustering and cluster naming components of Head Start are executed and the input data for a knowledge map is generated. In a last step, for each available DOI altmetric data (number of Tweeters and number of Mendeley readers) are retrieved from Altmetric, as well as the number of citations from Crossref.

We then conducted an evaluation of the visualization pipeline on a sample of 225 projects for the time range of 2008 to 2017. The sample was generated from a seed list of well-known funders and research institutions. We used the sample to (1) qualitatively inspect knowledge maps for the user tests (see section 2) and to (2) conduct quantitative analyses of results. The evaluation showed that a significant number of projects (around 50%) have a maximum number of 25 publications and datasets associated to it, and a vast majority of around 90% of projects have a maximum number of 50 publications and data sets linked to them. Accordingly, the clustering and cluster naming algorithms have been adapted to perform well in this range.

With the goal of maps being always up-to-date, VIPER includes an automatization component. In regular intervals VIPER refreshes existing maps directly from source data, to ensure that VIPER utilizes data that is not older than 24 hours. Once VIPER detects possible user-made changes to the data, the updates occur with a higher frequency. Maps are marked as likely having new resources, when a user opens the "How to add project resources" popup and clicks on the link to OpenAIRE next. Maps that are marked as such are updated every 10 minutes.

Additionally, the automatization component enables the creation of maps in bulk, according to specific parameters. This allows for rapid generation of project overviews, for example when a research institution or funder wants to gain insight into its associated publication activities in a given timeframe. VIPER can then automatically create maps for all projects associated with a funder or institution.

4.1.2 Frontend

The frontend of Head Start was adapted and extended for VIPER. A scaling component was devised as an additional toolbar, which rescales the map according to a given metric. The size of the bubbles and resources is thereby relative to the number of occurrences of the selected metric. In addition, we adapted the visual representation to the OpenAIRE data, including a suitable representation of datasets and an adaptation of context information. New modals were developed, providing further information about the displayed project, enabling the possibility to embed visualizations on other websites, as well as providing information on how to link further project resources using OpenAIRE services. Furthermore, the list representation was over-hauled to include an additional filter enabling users to show only resources relevant to their current information need, e.g. only open access papers or only datasets.

In addition to the changes to the frontend, we also developed a project search component, which lets users search for projects using keywords, grant id, acronyms and other relevant pieces of information. Users are then presented with a paginated list of projects matching the search criteria and can select projects for visualization. The search was created using the projects endpoint of the OpenAIRE HTTP API. Project information was extracted and enriched with the number of related publications and datasets using the publications and datasets endpoints of the OpenAIRE HTTP API. The project search component is included as a Head Start as an example.

4.2 Updating the R package ropenaire

In the scope of this project, we updated the existing [R package ropenaire](#) (Chamberlain & Jahn 2018) to the current version of the OpenAIRE RESTful HTTP API. As with the extension of rcrossref (see section 4.1.1), this work was carried out in collaboration with [rOpenSci](#).

In VIPER, we use ropenaire to query publications and datasets related to a project. The package, however, is a general purpose data client, which gives access to the OpenAIRE RESTful API with R. It can be used in a variety of R-based analysis of metadata provided by OpenAIRE. It mainly exposes three functions, which are described below:

roa_projects: searches the OpenAIRE project database and returns project metadata records related to specified criteria. These criteria include keywords, start year and/or end year, participating organisation, funder or call identifier. It uses the Among other things, the search results can be sorted by various criteria (such as project duration) and be returned in the following formats: json, xml, csv, tsv. The function wraps the projects endpoint at http://api.openaire.eu/search/projects.

Examples:
```
roa_projects(keywords = "open science", country = "GR")
roa_projects(call_id =  "FP7-PEOPLE-2010-IRSES",  org="POLITECNICO  DI
TORINO")
```

roa_pubs: searches the OpenAIRE publications database and returns publication metadata records related to specified criteria. These criteria include keywords, publication date, grant ID, funder or funding stream. It uses the Among other things, the search results can be sorted by various criteria (such as date of collection) and be returned in the following formats: json, xml, csv, tsv. The function wraps the publications endpoint at http://api.openaire.eu/search/publications.

Examples:
```
roa_pubs(fp7 = "283595")
roa_pubs(doi = "10.1051/0004-6361/201220935")
```

roa_datasets: searches the OpenAIRE datasets database and returns dataset metadata records related to specified criteria. These criteria include keywords, publication date, grant ID, funder or funding stream. It uses the Among other things, the search results can be sorted by various criteria (such as date of collection) and be returned in the following formats: json, xml, csv, tsv. The function wraps the publications endpoint at http://api.openaire.eu/search/datasets.

Examples:
```
roa_datasets(title = "methane")
roa_datasets(grant_id = "283595", sort_by = "dateofcollection")
```

5 Future Work

The user tests conducted in the scope of this project (see section 2) have shown that VIPER is perceived as a visual guide to the world of research projects. It has great potential for providing a better overview of research projects and thus improving the visibility of project results overall. In addition, we received a lot of positive feedback for the potential of using VIPER maps as automatically updated dissemination pages, on institutional websites and in dashboards. Furthermore, VIPER would also have a systemic effect as it motivates users to keep their project records in OpenAIRE updated.

The user tests have also show that there is a lot of potential for future work. This includes a further integration with OpenAIRE, such as a seamless integration of the linking service as well as displaying knowledge maps on each OpenAIRE project page to complement the existing project representation with a visual component. In addition, VIPER could be used to enrich publication metadata in OpenAIRE by offering to add/edit information via the visualization interface. Another potential area of collaboration could be the development of a visual project search, enabling the visual exploration of large amounts of research projects on a topic by e.g. discipline or subtopic.

In addition, we see great potential for integrating recommendations in VIPER, in collaboration with the Matchmaker service developed in another OpenAIRE Tender project. On the one hand, VIPER could be used to visualize and thus contextualize the recommendations by Matchmaker.

On the other hand, recommendations capabilities could be added to VIPER to provide context for recommendations within a bubble.

Finally, as pointed out in the user tests, it would be useful to have an editing mode, where project representations can be modified, for example enabling changing bubble titles or rearranging map items. In addition, it would be interesting to explore further metrics for integration in VIPER and to investigate how to best normalize metric counts to disciplines and/or geographic regions.

6 References

Bornmann, L. (2015). Alternative metrics in scientometrics: A meta-analysis of research into three altmetrics. *Scientometrics*, 103(3), 1123-1144. https://doi.org/10.1007/s11192-015-1565-y

Chamberlain, S. & Jahn, N. (2018). ropenaire: an R client for the OpenAIRE API (Version v.0.2). Zenodo. http://doi.org/10.5281/zenodo.1247949

Kraker, P., & Lex, E. (2015). A Critical Look at the ResearchGate Score as a Measure of Scientific Reputation. In *ASCW'15 Workshop at Web Science 2015*. http://doi.org/10.5281/zenodo.35401

Kraker, P., Kittel, C., & Enkhbayar, A. (2016). Open Knowledge Maps: Creating a Visual Interface to the World's Scientific Knowledge Based on Natural Language Processing. 027.7 Journal for Library Culture, 4(2), 98–103. http://doi.org/10.12685/027.7-4-2-157

Kraker, P. Schramm, M., Kittel, C., Chamberlain, S., & Arrow, T. (2018). VIPER - The Visual Project Explorer 1.0. Zenodo. http://doi.org/10.5281/zenodo.1247473

Manghi, P., Bolikowski, L., Manola, N., Schirrwagen, J., & Smith, T. (2012). OpenAIREplus: the European Scholarly Communication Data Infrastructure. *D-Lib Magazine*, 18(9-10). http://doi.org/10.1045/september2012-manghi

Manghi, P., Artini, M., Atzori, C., Bardi, A., Mannocci, A., La Bruzzo, S., Candela, L., Castelli, D., & Pagano, P. (2014). The D-NET software toolkit: A framework for the realization, maintenance, and operation of aggregative infrastructures, *Program*, 48(4), 322-354. https://doi.org/10.1108/PROG-08-2013-0045

Priem, J., Taraborelli, D., Groth, P., & Neylon, C. (2010). *Alt-metrics: a manifesto*. Retrieved from http://altmetrics.org/manifesto/

Zahedi, Z., Costas, R., & Wouters, P. (2017). Mendeley readership as a filtering tool to identify highly cited publications. *J. Assoc. Inf. Sci. Technol.*, 68(10), 2511-2521. https://doi.org/10.1002/asi.23883

Appendix A: Deployment

A.1 Installing and deploying the client

To get started, clone the [Head Start repository on Github](). Next, duplicate the file config.example.js in the root folder and rename it to config.js.

Make sure to have npm version 3.10.10 installed (it comes with Node.js 6.12.0, you can [download installers here]()) and run the following two commands to build the Head Start client:
```
npm install
npm run dev
```

[webpack](), which is used to build the client-side application, is now started in *watch mode* which means that changes to files are tracked and the created headstart.js is automatically updated.

Now you can run a local dev server:
```
npm start
```

Point your browser to the following address:
```
http://localhost:8080/examples/local_files/index.html
```

To run Head Start on a different server (e.g. Apache), you need to set the publicPath in config.js to the URL of the dist directory:
- Dev: specify the full path including protocol, e.g. http://localhost/headstart/dist
- Production: specify the full path excluding protocol, e.g. //example.org/headstart/dist

If everything has worked out, you should see a visualization of the field of educational technology.

To create a production build of Head Start, run the following command:
```
npm run prod
```

Now, copy the folder *examples/viper* and the folder *dist* to the relevant location on your webserver. In a next step, define in *config.php* the relative path of the dist-folder to the index.php.

A.2 Installing and configuring the server

A.2.1 Requirements

Make sure you have the following packages installed:

- PHP 5.3+ with the following extensions:

- ○ curl
- ○ pdo_sqlite
- ○ mbstring
- ○ fileinfo
- ○ Xml

- R 3.3+ (https://www.r-project.org/) with current updates, with the following libraries. **Make sure you install these packages for all users, so that Apache can load them.**
    - ○ GMD
    - ○ MASS
    - ○ ecodist
    - ○ jsonlite
    - ○ tm (Version 0.6 - **do not use 0.7**)
    - ○ proxy
    - ○ SnowballC
    - ○ rplos
    - ○ parfossil
    - ○ doParallel
    - ○ rentrez
    - ○ curl
    - ○ tibble
    - ○ data.table
    - ○ stringi
    - ○ stringdist
    - ○ xml2 **(Version 1.0 or higher)**
    - ○ ropenaire (Currently, a Github repository only: https://github.com/njahn82/ropenaire. Install with devtools.)
    - ○ readr (for ropenair/VIPER)
    - ○ rcrossref (for VIPER) with the async dev version: install with re-motes::install_github("ropensci/rcrossref@async")
    - ○ rAltmetric
    - ○ plyr (for better data munging)

- phantomjs 2.1+ (http://phantomjs.org/), if you want to use the snapshot feature.

A.2.2 Configuration

Set the following variables in your php.ini:

- max_execution_time = 240
- max_input_time = 480
- memory_limit = 512M
- upload_max_filesize = 100M
- post_max_size = 100M

Copy the server folder in the Head Start repo into the same directory as the dist folder.

Next, duplicate config.ini in server/preprocessing/conf/ and rename it to config_local.ini. Adapt the following variables:
- general->preprocessing_dir: Full path to the preprocessing directory
- general->host: Host of the client visualization
- general->path: Relative path to the client visualization. Needs to be in the public_html/www directory.
- general->images_path: Full path to the images directory for the client. Needs to be in the public_html/www directory. **Make sure that your webserver has write access to this directory.**
- general->services_path: Relative path to the client REST services. Needs to be in the public_html/www directory.
- calculation->binary: Path to RScript binary
- connection->sqlite_db: Full path to the sqlite datatabase file. For development purposes, duplicate headstart.sqlite in server/storage/ and rename it to a filename of your choice. Enter the path to this file here. **Make sure that your webserver has write access to this file and the containing directory.**
- snapshot->snapshot_enabled: Set to 1 to enable snapshot feature, 0 to disable
- snapshot->phantomjs_path: Absolute path to phantomjs binary
- snapshot->getsvg_path: Absolute path to getChartSVG.js
- snapshot->storage_path: Absolute path to the directory, where the snapshots are stored. **Make sure that your webserver has write access to this file and the containing directory.**
- snapshot->snapshot_php: PHP File responsible for rendering the bubble in a way to be snapshotted. Relative path to general->host
- snapshot->snapshot_width: Snapshot width

After you have completed these steps, you should be able to run VIPER on your web server by pointing your browser to the path of the VIPER directory.

A.2.3 Troubleshooting

If you repeatedly receive the message "Sorry! Something went wrong....", then you will need to check your setup. Make sure you have completed all configuration steps in A.2.2. Permissions are the number one issue why the search repos example does not work. If you run Apache on Ubuntu, make sure that www-data has the necessary permission for the SQLite database file and the images_path.

Another issue could be that your web server cannot run the R scripts properly because of missing permissions, or because you didn't install the necessary packages for all users. Under Ubuntu, run the R script with the www-data user from the following folder: server/preprocessing/other-scripts/test

```
sudo   -H   -u   www-data   bash   -c   'Rscript   path/to/text_similarity.R
"path/to/other-scripts"   "OpenAIRE"   "openaire"   "params_openaire.json"'
```

Replace path/to with the actual full paths. This should give you an idea whether the R scripts are producing the results that they should.

A.2.4 Automatisation

To automatically update the maps created by VIPER, we suggest running cron jobs on your server. The basic setup consists of two cron jobs:

1. Description: a cron job, which updates all maps created so far.
   Suggested frequency: daily.
   Command: `php [PATH_TO_VIPER]/server/services/updateViper.php --vis_changed false`
2. Description: a cron job, which updates maps, where users have likely added new resources. Maps are marked as likely having new resources, when a users opens the "How to add project resources" popup and clicks on the link to OpenAIRE next. The flag is automatically removed after 24 hours.
   Suggested frequency: every 10-20 minutes.
   Command: `php [PATH_TO_VIPER]/server/services/updateViper.php --vis_changed true`