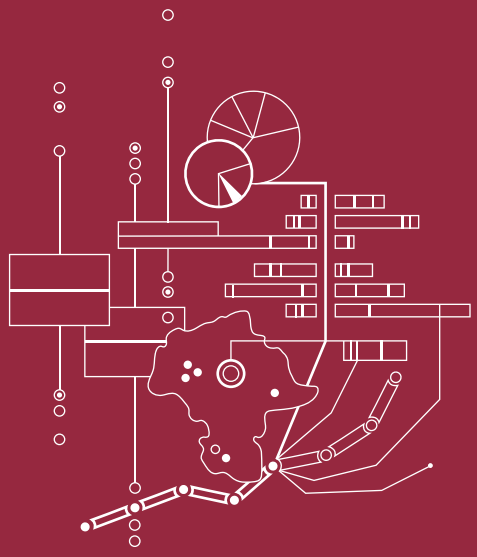
NATIONAL INFORMATION PROCESSING INSTITUTE

RAD-on

Reports
Analyses
Data



Scientific editors
Aldona Tomczyńska
Anna Knapińska
Sylwia Ostrowska

Information technology systems
that support science and higher education

RAD-on:
Reports, analyses, data

Scientific editors:
Dr Aldona Tomczyńska
Dr Anna Knapińska
Sylwia Ostrowska

NATIONAL
INFORMATION
PROCESSING
INSTITUTE

Information technology systems that support science and higher education
RAD-on: Reports, analyses, data

Scientific editors:
Dr Aldona Tomczyńska
Dr Anna Knapińska
Sylwia Ostrowska

Authors:
Marcin Białas: Chapters 2 and 5
Łukasz Błaszczyk: Chapter 4
Dr Sławomir Dadas: Chapter 3
Dr Anna Knapińska: Chapters 2 and 3
Dr Marcin Mirończuk: Chapter 5
Sylwia Ostrowska: Chapters 1, 2, and 4
Emil Podwysocki: Chapter 4
Dr Aldona Tomczyńska: Chapters 1 and 3

Translator:
Michał Tomaszewski

Copyeditor:
Kevin McRobb

Reviewer:
Dr Zenon A. Sosnowski, Associate Professor at the Bialystok University of Technology

Published by:
National Information Processing Institute
al. Niepodległości 188b
00-608 Warsaw
email: opi@opi.org.pl
www.opi.org.pl

ISBN 978-83-63060-26-8

Cover and graphic design:
Karina Maszewska

Typesetting:
Dr Jakub Kierzkowski

Printed by:
Lenis Przemysław Dubrzyński
ul. Kosynierów 27, 04-641 Warsaw

Dear readers,

In the contemporary world, where information plays a pivotal role, the ability to access data is crucial for the advancement of societies. We must ensure that data is up to date and reliable, originates from trusted sources, and is processed responsibly. Operated by the National Information Processing Institute (OPI PIB) on behalf of the Polish Ministry of Education and Science, the RAD on portal serves as a valuable source of such data for the academic community and the wider science sector in Poland.

This monograph delivers a comprehensive overview of the RAD-on portal. It offers compelling information on the concepts of data integration and processing, as well as on national systems of science and higher education. It may be of interest to the scientific community, IT professionals involved in the development of modern information systems, and decision-makers who rely on accurate data to execute informed political decisions.

I highly recommend that you not only peruse this monograph, but also take advantage of the RAD-on portal, which facilitates the examination of higher education trends in Poland, and provides insightful data related to the science and higher education sector. The portal is a cutting-edge knowledge base that is regularly updated with useful features. In addition to raw data, it also offers in-depth reports and analyses.

RAD-on offers comprehensive IT solutions that streamline and accelerate data accessibility. Data can be filtered in various ways, which enables users to customise their search options. RAD-on data can be downloaded and used in applications, databases, programmes, and IT services free of charge. The portal is equipped with REST API, which provides users with unrestricted access to open data.

I extend my congratulations to the experts at OPI PIB for their work in preparing a valuable scientific monograph, and express my gratitude for their commitment to the implementation and continuous modernisation of RAD-on.

Przemysław Czarnek

Minister of Education and Science

Dear readers,

We are living in the fourth industrial revolution, which is also known as the digital revolution. Information has become an extremely valuable resource. It is truly remarkable that so much information is now at our fingertips, accessible to nearly everyone through a small device. In the twenty-first century, accessing the vast amount of information that has accumulated over time is easier than ever before.

To establish an information society that relies on reliable sources of knowledge, initiatives and projects like the RAD-on portal are crucial. Developed as part of the ZSUN II. The Integrated System of Services for Science. Stage II project, this cutting-edge IT tool is the fruit of collaboration between the Polish Ministry of Education and Science (MEiN) and the National Information Processing Institute (OPI PIB). RAD-on provides access to reliable information that is sourced not only from the extensive databases of the POL-on Integrated System of Information on Science and Higher Education, but also from many other central systems that operate in higher education and science. The portal enables the preparation of customised reports and analyses that cater to the specific needs of its users. RAD-on is a prime example of open science policy in action. With 11.5 terabytes of data, it offers one of the richest data resources on higher education and science in Europe.

The RAD-on portal is designed for a wide range of users that incorporates the academic community, students, industry representatives, and political decision-makers. RAD-on is not limited, however, to these groups; it can cater to anyone who seeks detailed, reliable, and up-to-date information on the institutions that contribute to the science and higher education system—including their didactics and scientific activities.

This monograph delivers a comprehensive account of the development and refinement of RAD-on, highlighting its impressive capabilities and potential for future growth. I am confident that it will not only persuade readers to use the dependable knowledge that can be accessed using RAD-on, but also encourage collaboration and the ongoing enhancement of the platform's functionalities.

Łukasz Wawer

Deputy Director of the Centre for Digital Transformation

Ministry of Education and Science

CONTENTS

INTRODUCTION

The RAD-on portal forms an integral component of an IT system that presents reports, analyses, and data on science and higher education in Poland. RAD-on was developed by the National Information Processing Institute (OPI PIB) in collaboration with the Polish Ministry of Education and Science as part of the Integrated Network of Information on Science and Higher Education project.

In terms of the data it collects and shares, RAD-on is currently the largest national IT system in science and higher education in Europe. It stores information on nearly all scientific institutions in Poland, as well as their scientists. This includes academic teachers and other individuals who conduct classes at higher education institutions. RAD-on also offers comprehensive lists of scholarly publications, national and international patents, research projects funded by various sources, and institutional investments. It contains extensive information on students, graduates, and on tertiary education programmes, including fields of study, foreign students, PhD students, and study formats.

RAD-on's key services are:

- a knowledge base that contains up-to-date and official data on scientific institutions, scientific and artistic activities, academic staff, and scientific promotion procedures
- interactive, cross-sectional reports on higher education institutions, students, graduates, academic staff, and research conducted in Poland and internationally
- a repository of reliable studies in science and higher education that contain comprehensive commentaries
- an application programming interface (API), which guarantees free access to RAD-on resources. This allows users to develop their own solutions and applications that require access to data on higher education
- user accounts that are used to browse the data gathered in RAD-on and its source systems. Registered users can update and verify their data to ensure that RAD-on provides the highest quality of information.

Most of the services described above are available to the public through the RAD-on website, with only a select few reserved for verified users. The system evolves continually, which results in steady increases in data volume and ever-more useful services. This monograph presents the functionalities of the system, examines its analytical and IT foundations, and presents examples of its practical applications.

The purpose of this monograph is to showcase RAD-on as a one-of-a-kind solution in Europe, and as a system that aligns with scientific policy trends in Poland and globally.

The first of these trends is the concept of open government data, which is rooted in the belief that public authorities should be transparent in their actions. Providing data to all citizens on an equal basis is one of open government data's strategies for increasing social engagement. Providing data to all citizens on an equal basis is one of open government data's strategies for increasing social engagement. By offering tools that collect, share, and analyse data from official sources, governments can contribute to the advancement of civil society and foster innovation [61, 39].

RAD-on also supports data-driven policy making. Political decision-makers address social and economic issues by creating and enforcing laws, regulations, and guidelines. Monitoring the effectiveness of these policies forms a crucial aspect of the political process. The political cycle involves various stages, including agenda setting, policy formulation, decision-making, implementation, and evaluation. IT systems can be used at each of these stages to collect data, improve its flow, process it into valuable information, and forecast future events [57].

RAD-on contributes to the design of data-sharing systems that prioritise the principles of findable, accessible, interoperable, and reusable (FAIR) data. The FAIR concept is associated primarily with scientific datasets used in research. Although RAD-on is not a repository of this type of data, its resources can be used to analyse higher education and science. In this respect, they adhere to the FAIR principles [27, 62].

All three concepts were considered in the design of RAD-on. They will be discussed in more detail in Chapter 1, which presents Poland's scientific policy regarding the collection of data on the activities of students and scientists employed by scientific institutions. Presenting the intricacies of the systemic conditions will help demonstrate that the development of RAD-on is a natural consequence of a data-driven policy in science. Chapter 1 also compares RAD-on to other IT systems in Europe that share similar functionalities. To ensure that the content of this monograph can be applied practically, we share our design experiences and discuss the challenges we encountered during the project's implementation.

Chapter 2 focuses on RAD-on services that are provided through an open civic portal, which is the main result of the project. The chapter also describes the functionalities of the portal, such as the 'Citizen Data' service, which facilitate access to data and ensure that the data is up-to-date and complete.

Chapter 3 discusses RAD-on tools that can replace commercial business intelligence (BI) systems and support the transformation of data into information and knowledge. The chapter presents our reporting system that generates user-friendly reports. It also demonstrates the dilemmas faced by the data analysts who are responsible for the development of statistics for nonhomogeneous groups of recipients.

Data is the beating heart of RAD-on. Chapter 4 discusses the scale of our data management activities and explores the creation of our data exchange model (DEM). The chapter also introduces our data warehouse, which integrates nearly all of the information systems that were developed by the National Information Processing Institute

during the last decade. The chapter describes the challenges presented by data format standardisation, and outlines our proposed strategies for overcoming them.

Chapter 5 is dedicated to exploring the API—a crucial element of contemporary information systems. It is intended for programmers who work in scientific institutions or other organisations that seek to leverage machine processing for data analysis.

We trust that the subjects addressed in this monograph will appeal to readers who are engaged in the development of software that facilitates the gathering and processing of data on behalf of, or in partnership with, public entities. We firmly believe that the information presented in this monograph will prove valuable in comprehending the complexities of data in higher education and science. It also serves as an invitation to discuss the future of RAD-on. We are confident that our efforts to promote more widespread understanding of RAD-on will inspire stakeholders to collaborate in the portal's development and in the enhancement of its efficacy.

CHAPTER 1

THE CONCEPT AND PURPOSE OF THE RAD-ON PORTAL

Sylwia Ostrowska
Dr Aldona Tomczyńska

1.1. The concept of data integration and sharing

The purpose of this monograph is to showcase the RAD-on system as a one-of-a-kind solution on a European level. RAD-on aligns with three scientific policy trends in Poland and globally, which are discussed in greater detail below.

1.1.1. Data-driven policymaking

Western countries, including the member states of the European Union, strive to maximise their use of data in their decision-making processes. This approach is not novel, and its roots can be traced back to the 1990s [51].

In 1994, *The new production of knowledge* [22], was published, which discussed the interaction between scientists and citizens. The book had a profound impact on the academic community, sparking debate about the changes that affect science as a result of the processes occurring in Western societies. In 2001, *Re-thinking science: Knowledge and the public in an age of uncertainty* [38] explored the concepts introduced in the 1994 book more deeply. The authors analysed the role of society in the knowledge creation process. In their view, the research process involves not only scientists, but also citizens who ask pertinent questions and anticipate coherent answers based on reliable data. As a result, the once well-defined boundaries between science, the economy, and politics are becoming increasingly blurred.

Theoretical considerations regarding the role of science can be linked to discussion on the changes that have resulted from successive technological revolutions. Interest in knowledge that is derived from data is increasing as analytical processes become less time-consuming and costly. The development of data-driven policymaking can, therefore, be linked to four phenomena [45]:

- advancements in the technology, including infrastructure and software, used to process large datasets (big data). The process has been driven primarily by the business sector, which recognises the commercial potential of analysing data from various sources
- the emergence of new methods for data analysis. Advanced data science combines statistics and computer science to enable machine learning, including the processing of natural language, images, and audio and video materials
- increased interest in IT tools among nonexperts and the improvement of digital literacy skills in economically developed societies
- deeper understanding of the importance of scientific research and the benefits of applying scientific knowledge in everyday life.

1.1.2. Open government data

The trends outlined above have led to a gradual increase in interest in tools that provide access to nonprivate and nonconfidential data [48]. An excellent illustration of this phenomenon is the widespread use of Google Trends, a service that offers insights into the most frequently searched phrases on the internet[1]. Google Trends is already being used not only by internet marketing experts as a source of knowledge, but also by researchers, including economists and sociologists—despite justified doubts around the quality of the data it provides [11].

One of the most notable categories of data collected by the business sector is that obtained from official registers and reports, and shared by public bodies. Open government data (OGD) is a reliable source of information that is made available free of charge by public institutions and their subordinate entities. The European Union and the United States have been pursuing the goal of opening official data to the public for over a decade [20]. Numerous countries have recognised the importance of establishing dedicated entities that are responsible for the digitalisation of public services. This has led to the development of increasingly complex and interconnected IT systems. Poland boasts *Otwarte Dane*[2], a website that grants access to a wealth of information sourced primarily from government and local administration entities. Similar services exist in other European Union member states.

As work on OGD progresses, new challenges continue to emerge. Currently, emphasis is placed on ensuring database interoperability. Studies suggest [17] that the use of OGD in decision-making processes is linked inextricably to the manner in which the data is made available. Users of IT systems are reluctant to utilise raw data [60], because cleaning it requires analytical expertise. For that reason, the crucial element in the creation of OGD infrastructure is information processing. Patricia Huston, Victoria L. Edge and Erica Bernier [29] stress that the most considerable advantages can be attained by

[1] trends.google.com (accessed 2 March 2023)
[2] dane.gov.pl (accessed 2 March 2023)

integrating data from multiple sources that form an OGD system; in other words, the optimal system is one that replaces large, raw datasets with access to interconnected and suitably processed information. Only OGD that is prepared in this manner can be properly interpreted and used directly in analytical processes.

1.1.3. FAIR data

To provide users with contextual in-depth data analysis, the FAIR standard's underlying principles must be adhered to. The standard was first presented in the article, *The FAIR Guiding Principles for scientific data management and stewardship* [62], which was published in *Nature* in 2016. Since then, the standard has been developed further as part of various initiatives, including the European Research Area.

According to the standard, data should be findable, accessible, interoperable, and reusable (FAIR). The FAIR concept is a crucial component of the broader open science movement, which emphasises the importance of data repositories for research purposes, including experiments' results. The authors of the concept admit that FAIR can be approached more broadly. In fact, it can be seen to encompass all digitalised research resources—from data to analytical processes. The sharing of information collected at every stage of the research process is crucial in guaranteeing transparency, reproducibility, and the ability to replicate research results. Changes pertaining to data storage should benefit both users and the machines that assist them. For that reason, in FAIR, emphasis is placed on the creation of metadata sets and instructions regarding their processing that can be understood both by humans and by computer programs.

In science and higher education, the FAIR concept is expressed by the Common European Research Information Format (CERIF), which is recommended by the European Union. It is used to store information on scientific activities, and enables the free exchange of data between scientific information systems. CERIF is developed by euroCRIS, the International Organisation for Research Information[3].

1.2. National systems of information on science and higher education

The use of information systems to organise data collected by scientific institutions is becoming increasingly widespread, both in Poland and internationally. Alongside student management systems, research information systems are also gaining in popularity. Some of them are available to the public, such as the DSpace-CRIS[4] repository system, which is developed by the Massachusetts Institute of Technology in collaboration with

[3] eurocris.org (accessed 2 March 2023)
[4] 4science.com/dspace-cris (accessed 2 March 2023)

Hewlett-Packard. In 2012, Warsaw University of Technology initiated the development of Omega-PSIR, a comprehensive repository that delivers a range of features to facilitate the work of university administrators and academic staff. The license and open source code are both available free of charge. In 2022, nearly forty scientific institutions, including higher education and research institutes, utilised the system[5].

Despite such solutions proving popular among individual higher education institutions and their associations, only a few countries have implemented the student management or research information systems centrally. As far as data on students and staff of the research and development sector is concerned, the most popular systems are developed by international organisations, such as the World Bank, the United Nations Educational, Scientific and Cultural Organization (UNESCO), the Organization for Economic Co-operation and Development (OECD), and the European Union (through Eurostat, which serves as the union's statistical office). The data contained in such systems pertains to foundational indicators, such as the number of students or personnel within the research and development sector in individual member states. This data is sourced from the statistical offices of the respective countries. Due to data typically being aggregated nationally, the systems fail to offer detailed information on the higher education sector. In 2014, the European Tertiary Education Register (ETER) was introduced with the aim of providing data on higher education institutions across Europe. The register includes data on students, academic staff, and the finances of higher education institutions. The data is collected through the statistical offices and government ministries of individual countries; IT systems are not used in this process. ETER, at present, lacks a significant amount of data—particularly in regard to less commonly used indicators[6].

More comprehensive data on the situation of particular scientific institutions is provided by the central systems that are developed by individual countries. The Higher Education Statistics Agency (HESA) in the United Kingdom stands out as a noteworthy resource of information on educational institutions and students. The HESA system provides unrestricted access to data on higher education in England, Wales, Scotland, and Northern Ireland, including on students, academic staff, and graduates [10]. It is updated continuously and offers more than data access: it also features business intelligence systems that enable data filtering and visualisation. HESA does not include data on science and machine processing services[7]. Selected information on the higher education sector is also published in this manner by the Austrian Ministry of Education, Science and Research (*Bundesministerium für Bildung, Wissenschaft und Forschung* – BMBWF) and by the Nordic Institute for Studies in Innovation, Research, and Education (*Nordisk institutt for studier av innovasjon, forskning og utdanning* – NIFU) in Norway[8].

[5] omegapsir.io/pl (accessed 2 March 2023)
[6] eter-project.com (accessed 2 March 2023)
[7] hesa.ac.uk (accessed 2 March 2023)
[8] Comprehensive descriptions of the information provided by the systems can be found at: hesa.ac.uk/data-and-analysis (accessed 2 March 2023), unidata.gv.at/Pages/auswertungen.aspx (accessed 2 March 2023), nifu.no/fou-statistiske/fou-statistikk (accessed 2 March 2023)

Many countries are developing centralised scholarly information systems that collect data on scholars and their research, and facilitate the sharing of this information between services. According to EuroCRIS presentations and publications, such systems have been or are being developed by at least thirty countries[9]. One of the most recent projects is Research Portal Denmark, which was launched in 2022. It contains data on science from various sources, including that from higher education institutions that fund scholar projects. system is being developed with the financial support of the Danish Ministry of Higher Education and Science and the Danish Agency for Higher Education and Science. Interestingly, the system offers access to data provided by both state and international sources, including that collected by commercial entities, such as Clarivate, Elsevier, and Digital Science[10]. At the time of writing, the system provides data primarily on publications by Danish scientists, and relies on integration with the bibliographic and abstract databases mentioned above.

Compared to other systems developed in Europe, RAD-on stands out due to the following features:

- a wide range of available data from the higher education and science sectors. RAD-on integrates information on teaching and research conducted by scientific institutions in Poland
- a high degree of data disaggregation and interoperability. Some RAD-on registers, such as the register of academic teachers, provide information on specific individuals employed in the sector
- an extensive portfolio of services, including machine data processing, tools for visualisation and downloading of data that has already been processed, developed by experts and supplemented with appropriate commentaries, and tools for viewing and correcting data by individuals to whom it pertains.

1.3. Science policy and the process of digitalisation of science and higher education in Poland

Research and didactics information systems have the potential to support a wide range of processes, one of the most crucial being the allocation of funds among scientific institutions. To gain a comprehensive understanding of the significance of RAD-on in science policy, we must delve into the process of the evolution of the science and higher education system in Poland. A crucial element of this process is the gradual increase in the significance of scientometric data.

Julita Jabłecka and Benedetto Lepori [31] identified the following distinct phases in the development of the science system in Poland:

[9] dspacecris.eurocris.org (accessed 2 March 2023)
[10] forskningsportal.dk (accessed 2 March 2023)

- 1989–1991: a radical change in a short period. In 1990, following the political transformation and collapse of the socialist regime in Poland, the Western concept of university autonomy and research freedom was reintroduced
- 1991–2000: a period of stabilisation. From 1991, science policy was led by the Committee for Scientific Research (KBN), which was also responsible for the allocation of funds for research
- 2000–2007: systematic changes that led to further restructuring of science policy and the financing system for science. In 2005, the KBN became a part of the then Polish ministry responsible for science. At that time, science policy was shaped by the Ministry of Science and Digitalisation, which was later transformed into the Ministry of Education and Science (MEiN).

The year 2007 marked the beginning of a process that ultimately resulted in the creation of Poland's current science system. The National Centre for Research and Development (NCBR)[11] was the first independent agency in Poland that provided funding for research. It was established based on the principles of similar agencies in the European Union and the United States. NCBR's mission is to foster innovation by promoting collaboration between the science and business communities. In 2010, the Polish National Science Centre (NCN)[12] was established as an agency dedicated to funding basic research. In 2017, the Polish National Agency for Academic Exchange (NAWA)[13], joined forces with other agencies to further the internationalisation of Polish science. NAWA's primary objective is to support and stimulate research collaboration and international academic exchange.

The agencies handle both national and European funds that can be utilised for scientific and innovative endeavours. They distribute the funds through competitive project proposals submitted by individual scientists, and by research teams who work for scientific institutions and innovative enterprises.

As the project-based funding system evolved, the process for allocating funds to ensure the continued operation of scientific institutions was refined and improved. Higher education institutions and other entities within the scientific community have the opportunity to apply for public funds for research, in addition to didactic funds. Research funds are distributed based on the results of institutional evaluation. Unlike project-based funds, institutional funds are not allocated for specific research initiatives. The institutions themselves hold the decision-making power on how to allocate the funds.

The first KBN evaluation of scientific activity, which resulted in the allocation of funds for research conducted by institutions, happened in 1991. The evaluation of research potential was conducted by experts and aimed to classify scientific units, such as university faculties, based on their degree of scientific excellence. Subsequent

[11] ncbr.gov.pl (accessed 2 March 2023)
[12] ncn.gov.pl (accessed 2 March 2023)
[13] nawa.gov.pl (accessed 2 March 2023)

evaluations placed greater emphasis on scientometric indicators, which provided a more quantitative means of measuring research productivity [32]—a concept that describes the degree of intensity with which scientists publish their findings, participate in research projects, and commercialise the results of their scholarly work [43]. The application of scientometrics in evaluation was intended to serve two purposes: first, to enhance the objectivity of the evaluation process; and second, to enable exploration of the potential of the digitalisation of the science and higher education sectors. As funding allocation systems became more complex, the need for extensive data collection grew. This presented a range of legislative, organisational, and technological obstacles.

The National Information Processing Institute (OPI PIB) was established in 1991 with the aim of providing IT and analytical support to the evolving scientific landscape. OPI PIB has developed efficient IT infrastructure, including the Funding Stream Support System (OSF), which manages the allocation of funds for grants, and the System for Evaluation of Scientific Achievements (SEDN), which handles the allocation of institutional funds as part of evaluation process. The institute has also designed and implemented several database systems that collect information on aspects of research and didactic activities at higher education institutions. In 2010, OPI PIB embarked on the development of POL-on, which is used by higher education institutions and other scientific entities, such as research institutes, the institutes of the Polish Academy of Sciences, international research institutes, and Łukasiewicz Research Network institutes. POL-on serves as a solution for publishing a range of information that is required under applicable laws and regulations. The system collects data on the didactic and scientific activities of such entities and their financial information [36].

OPI PIB has also been involved in the development of other information systems for the Polish government ministries responsible for science and higher education. One notable contribution by the institute was the development of the Polish Scholarly Bibliography (PBN), which is—in simplified terms—the Polish counterpart of commercial bibliographic and abstract databases, such as Scopus. Each of the systems was developed at various stages of OPI PIB's operation, and this resulted in technological and content-related differences. All of the systems will be discussed in the subsection below.

RAD-on was developed in collaboration with MEiN. It is not a system designed to support a specific reporting, financial, or evaluation process; instead, it serves as a centralised hub that provides data reported by scientific institutions to a multitude of source systems developed at OPI PIB.

1.4. Stakeholders' needs

RAD-on was developed with the aim of integrating and sharing the data collected from OPI PIB's IT systems to create a data-based science policy. Experts at OPI PIB and MEiN identified the following problems and intended to address them using the new system:

- distribution of data on science and higher education in Poland across noninterconnected systems
- insufficient technical and methodological data integration
- no central hub capable of managing the data that is processed in individual systems
- a low degree of data openness.

Following an initial analysis, we designed services that were tested against the expectations of the scientific community. A qualitative study of RAD-on's stakeholders' needs was conducted between 31 August and 25 September, 2015 at OPI PIB's Laboratory of Statistical Analysis and Evaluation, under the guidance of Dr Marzena Feldy. The in-depth interview method was applied, which enabled thorough exploration and collection of detailed information. Nineteen meetings were held, and twenty-nine individuals were interviewed. The interviews were conducted using a two-part script (prepared in advance), whose objectives were: 1) to gather information on current and future needs regarding access to information on science and higher education, and determine how to meet them, and 2) to gather opinions on the services and modules that were planned to be implemented as part of the project.

The interviewees included representatives of potential groups of RAD-on stakeholders: scholars, representatives of university authorities, senior and junior staff of the Polish Ministry of Science and Higher Education (currently MEiN) and the Polish Ministry of Administration and Digital Affairs (currently the Ministry of Digital Affairs), members of the Polish Accreditation Committee (PKA), employees of the funding agencies NCN and NCBR, as well as journalists and representatives of nongovernmental organisations in education.

The information obtained during the interviews was supplemented with descriptions of the needs reported to the technical support department of the POL-on system between October 2013 and August 2015.

During the interviews, it came to our attention that the distribution of data on the sector across multiple databases and portals had caused considerable hindrance in the daily operations of our interviewees. The institutions entered data into OPI PIB databases individually, but were unable to feed their internal systems with such data. From their perspective, central databases were nothing more than new reporting requirements. Scientific institutions, research funding agencies, and the PKA expected RAD-on to play a significant role in streamlining bureaucratic processes. They highlighted the remote reporting and automatic feeding of higher education registers—thus, ensuring timely compliance with reporting obligations—as a key component of the new system. The interviewees wanted the system to uniformise data collection and to ensure data stability and quality. The most crucial aspects were ensuring the compatibility of the central system with the institutions' internal databases and preventing duplication of work related to the synchronisation of individual systems.

Individual researchers expressed the need for greater autonomy in managing their personal data that is stored in existing databases. The interviewees highlighted several advantages of integrating distributed systems, such as reduced search times

for scientific information and decreases in the number of access codes that they had to memorise.

The analysis of stakeholders' expectations helped determine the scope of services that are offered by RAD-on. This is described in greater detail below.

1.5. Project objectives

After carefully considering the needs of OPI PIB systems' users, we determined that RAD-on should complement the existing POL-on ecosystem [42]. RAD-on would supplement the array of services offered by OPI PIB systems by providing automatic processing and visualisation of aggregated data, and assist users in making decisions pertaining to higher education, research and development, and innovation. While the majority of OPI PIB systems gathered data, RAD-on would ensure that the data was properly processed and shared.

It was agreed that RAD-on was designed to:

- integrate data on science and higher education in Poland that originates from autonomous databases
- ensure open access to up-to-date and reliable data on research and development, science, and higher education
- reduce bureaucracy by eliminating the requirement for users to input data into databases
- support decision-making processes by providing IT tools that analyse and interpret available data.

1.6. Data shared by RAD-on

The initial step towards achieving the objectives described above involved identifying the various systems with which RAD-on was required to integrate (Figure 1.1.). The following is a comprehensive list of the systems:

- **The Integrated System of Information on Science and Higher Education (POL-on)**[14] gathers data according to the Higher Education Act [56] and the POL-on ordinance [47]. It includes modules that provide data on scientific and higher education institutions, students, academic teachers, doctoral students, scholarly and academic staff, scientific promotion procedures, domains and disciplines of study, investments, scientific achievements, financial reports, reports on recruitment to higher education institutions, and others

[14] polon2.opi.org.pl (accessed 2 March 2023)

- **The Polish Scholarly Bibliography (PBN)**[15] stores data on the publication achievements of Polish scholars, scientific institutions, and journals
- **The Polish National Repository of Theses (ORPPD)** was launched in 2009 and stores copies of the theses and dissertations of Polish graduates;
- **The Polish Graduate Tracking System (ELA)**[16] presents the results of yearly analyses on the circumstances of students, graduates, and PhD students on the labour market. The analyses are based on anonymised data from POL-on and from the Polish Social Insurance Institution (ZUS). The findings are presented as interactive reports and infographics that aid students in deciding what programmes to pursue
- **Inventorum**[17]is designed to foster collaboration between business and science. This active recommendation system provides targeted information on innovation, projects, innovative enterprises, scholars, and scientific institutions;
- **Polish Science**[18] is the oldest database of OPI PIB. Launched in 1999, it continues to provide free-of-charge information online. It contains data on Polish science, including institutions, individuals, scientific work, publications, and conferences. In 2018, it was archived and became part of POL-on
- **The Support System for Selection of Reviewers (SWWR)**[19] is an adaptive knowledge base of potential reviewers who are responsible for reviewing grant proposals, for example. SWWR recommends experts either by specific criteria or by the text included in proposals
- **The System for Evaluation of Scientific Achievements (SEDN)**[20] handles the process of evaluation of entities that operate in science and higher education in Poland, by using the data obtained from POL-on and PBN
- **The Funding Stream Support System/the Integrated System of Services for Science (OSF/ZSUN)**[21] is responsible for registering and managing applications for funding research and development projects. The funds are granted by MEiN, NCN, and NCBR. OSF and POL-on exchange information on entities operating in science and higher education, research projects, and publications (for more information, see [6]).

[15] pbn.nauka.gov.pl (accessed 2 March 2023)
[16] ela.nauka.gov.pl (accessed 2 March 2023)
[17] inventorum.opi.org.pl (accessed 2 March 2023)
[18] nauka-polska.pl (accessed 2 March 2023)
[19] recenzenci.opi.org.pl (accessed 2 March 2023)
[20] sedn.opi.org.pl (accessed 2 March 2023)
[21] osf.opi.org.pl (accessed 2 March 2023)

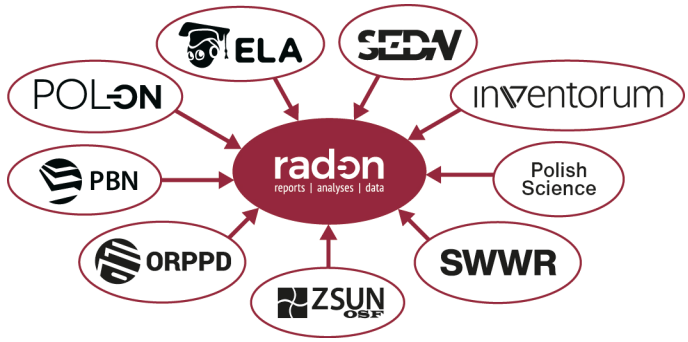


Figure 1.1. The systems integrated by RAD-on.

1.7. Project implementation and usability testing

The official name of the RAD-on project is 'ZSUN II. The Integrated System of Services for Science. Stage II'. The project was granted funding from the Operational Programme, Digital Poland. It was initiated in November 2017 and was successfully completed in January 2021. The total budget was PLN 27.6 million. RAD-on continues to be developed and promoted in the scientific community.

RAD-on was developed using a hybrid approach that combined both agile and waterfall methodologies. Presently, the majority of IT projects are managed using agile solutions. This approach offers a high degree of flexibility in meeting product requirements and delivering services (or their parts incrementally) [13]. However, organisations that finance research using public funds tend to prefer traditional waterfall approaches, which allow for greater control over project budgets and schedules. Both approaches have advantages, but differ fundamentally [55]. To complete the project successfully, it was determined that the development team's work conducted using the SCRUM methodology should be adapted to align with the required management model of the funding institution. Due to the project being cofinanced by the European Union, its budget, schedule, and scope—which had been approved at the stage of applying for funding—could not be altered during its implementation. The combination of agile and waterfall methodologies presented numerous challenges, which do not merit further discussion in this monograph.

We will, however, describe our efforts to ensure that the stakeholders are satisfied with the services delivered by RAD-on. Given the extended timeline of the project, a flexible approach to the final products was essential to accommodate the evolving needs of users.

To evaluate the quality of services provided by RAD-on accurately, the Laboratory of Interactive Technologies at OPI PIB conducted four comprehensive usability tests after each crucial stage of the system's development. The tests were based on in-depth interviews. Distinct groups of system stakeholders, which had been predefined during the initial needs assessment, were identified. To ensure that the system meets the highest

accessibility standards, it was essential to consider the needs of elderly users and users with visual impairments.

The usability studies first focused on the RAD-on system mock-up, and then on selected services and on the graphical layout. Each study was followed by a report whose recommendations modified the scope or form of the services. This resulted in significant alterations to the development timeline, yet ultimately led to higher user satisfaction ratings.

To monitor the ratings, a dedicated survey, which is available on the system website, is used. The results of the survey provide information on: 1) overall satisfaction with the reports, analyses, and integrated access to data; 2) satisfaction with the data's quality, and 3) satisfaction with the machine data processing services.

1.8. Project indicators and RAD-on application examples

To ensure the transparency of our actions, we also decided to monitor the project indicators publicly. The system that collects data on the popularity of individual services is currently available on the RAD-on website[22].

Since the launch of the project, RAD-on has shared 11.45 terabytes of science and higher education data. In 2021, documents were viewed or downloaded over 150 million times, which represents a five-fold increase compared to 2020.

The significant increase in the utilisation of RAD-on was due to the system's expanded quantity of available data, which rose by a quarter between 2020 and 2021. Moreover, as a result of the project's promotion and the dissemination of information regarding the ever-expanding array of services, there has been a notable increase in the number of individual users and entities that regularly incorporate the data into their own systems.

Between December 2021 and November 2022, the most popular API services for downloading official data were:

- scientific institutions (POL-on): 105,004,023 downloads
- publications (PBN): 455,962 downloads
- staff (POL-on): 193,142 downloads
- disciplines and domains of study (POL-on): 106,414 downloads.

To provide readers with a clearer understanding of the practical uses of shared data, we have outlined some examples of how external entities utilise RAD-on's services and integrated systems.

[22] radon.nauka.gov.pl/o-systemie/statystyki (accessed 2 March 2023)

RAD-on is used by the Polish National Agency for Academic Exchange (NAWA), which is responsible for the international promotion of information on the Polish science and higher education system. NAWA downloads information on higher education institutions in Poland and university programmes. The agency's access to official and regularly updated data enables it to organise the international education recognition process more efficiently. NAWA transmits RAD-on data to representatives of foreign higher education institutions, education recognition centres, and student organisations.

Another institution that uses RAD-on data is the Educational Research Institute (IBE)[23], which conducts interdisciplinary research on the functioning and efficiency of the education system in Poland. With access to machine public data sharing (RAD-on API), the IBE can easily and automatically populate its register[24] of qualifications awarded by higher education entities with data on university programmes.

RAD-on API's users also include foreign entities. As part of the Network4Growth programme, UNICO downloads data on scholars in Poland, as well as their publications, patents, and projects. The Network4Growth initiative fosters the technology transfer potential in Poland, Slovakia, Czechia, Hungary, Georgia, and Armenia[25].

RAD-on data is used in the submission of applications for research funds in the Funding Stream Support System (OSF). OSF is integrated with the RAD-on API service, which provides official data on entities that operate in the science and higher education sectors and that are recorded in POL-on. Because of this solution, entities that submit applications need not reenter data that is already available in another public register. RAD-on API is also utilised by the internal systems of large higher education institutions in Poland, usually to obtain data on publications.

[23] ibe.edu.pl (accessed 2 March 2023)
[24] kwalifikacje.gov.pl (accessed 2 March 2023)
[25] v4transfer.com (accessed 2 March 2023)

CHAPTER 2

THE PORTAL

Sylwia Ostrowska
Marcin Białas
Dr Aldona Tomczyńska
Dr Anna Knapińska

2.1. Users and services of the portal

The RAD-on system was designed initially to cater to a diverse audience that includes scientists, students, entrepreneurs, public administration staff, and journalists. Its primary objective was to provide reliable and up-to-date information on institutions that operate in the science and higher education sectors in Poland. Prior to the implementation of RAD-on, pertinent information was sourced from disparate databases and portals. The multitude of data sources in the country's science sector posed a considerable challenge for those seeking quick access to information. An analysis of user needs, described in section 1.4., revealed that some stakeholders of the science system were unaware of the existence of individual databases, which were potential sources of valuable information.

The RAD-on project entailed the creation of the necessary infrastructure for the effective management of data on science and higher education. Since the beginning of the project's implementation, we have recognised that the most crucial outcome was the development of a civic portal that serves as an open, integrated data access point.

This chapter offers a detailed description of the individual functionalities of the portal that illustrate the various ways in which data can be shared. It also demonstrates that our system provides customised services to user groups with varying degrees of analytical skill and knowledge of the science sector in Poland.

RAD-on is available at radon.nauka.gov.pl. The homepage of the portal is presented in Figure 2.1.

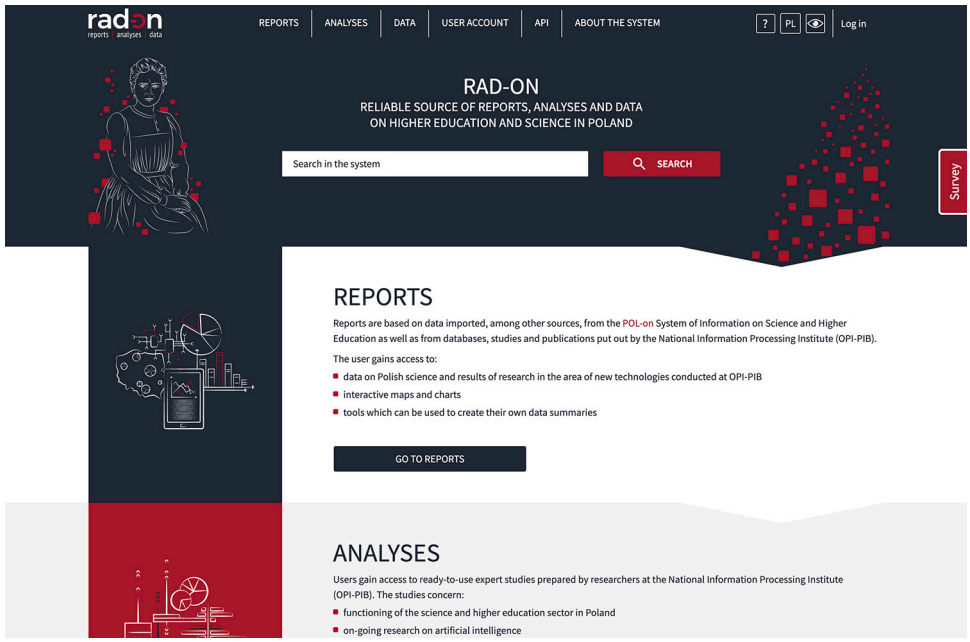


Figure 2.1. The RAD-on homepage.

RAD-on provides a range of services that can be categorised into two main types: those that are available to the public and those that require authentication. Those that are available to the public constitute an integrated knowledge base for the science and higher education sector. They include:

1. **Reports:** interactive data visualisations, including those from the POL-on system, as well as OPI PIB's databases, studies, and publications
2. **Analyses:** ready-to-use studies prepared by OPI PIB experts on the functioning of science and higher education in Poland
3. **Data:** data comparisons on institutions, as well as their scientific and artistic activities, employees, and academic promotion procedures.

A **full-text search engine** enables the resources related to science and higher education that are shared by all of the services described above to be searched.

The catalogue of services that are available to portal users is supplemented by **user accounts**—the only functionality that requires users to log in. The security of the service must be ensured, as the service allows users to access their own data, including personal data and other nonpublic information.

2.1.1. Data

The Data module is a basic service that provides easy and direct access to official data from multiple source systems. Our objective is to present the data in a clear and concise manner that is friendly to all groups of users and does not require advanced analytical skills. The Data module comprises summaries from diverse thematic categories that can be browsed using filters and downloaded in various formats. At present, approximately a dozen of sets of data are available (see Figure 2.2.). They are divided into the following categories:

1. **Institutions and their activities:** institutions of the science and higher education system in Poland, their branches, courses offered, doctoral schools, and bodies that award academic degrees and titles
2. **Scientific and artistic activities:** patents and rights of protection, publications, projects, artistic achievements, investments, and information related to the evaluation of the quality of scientific activity of entities
3. **Staff:** academic teachers, other teaching staff, and individuals who conduct or participate in the conductance of scientific activities
4. **Promotion procedures:** a list of procedures regarding the awarding of the degrees of doctor, habilitated doctor, and professor; the list is fitted with a search engine that enables easy access to files related to promotion procedures (theses, abstracts and descriptions, reviews and opinions, and applications).

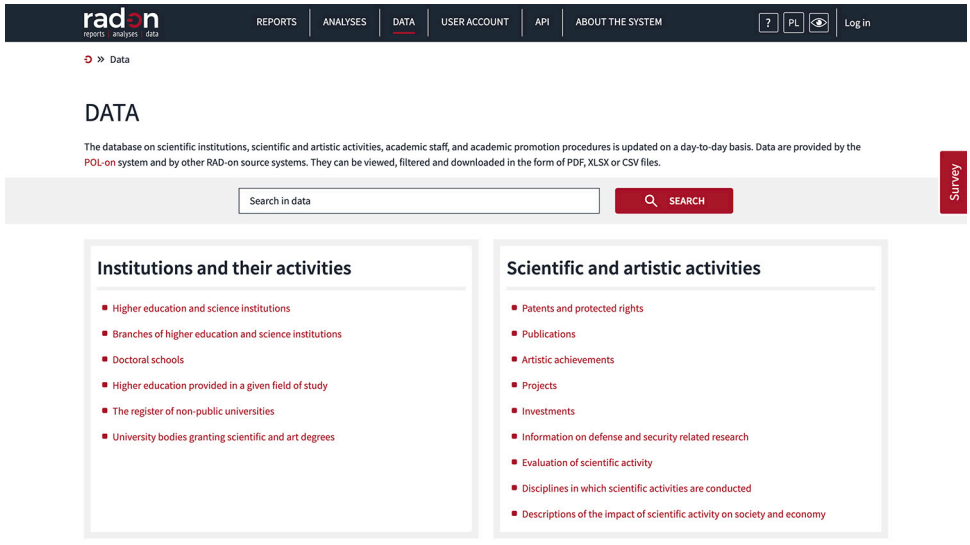


Figure 2.2. The Data module of the RAD-on portal.

The data comes from official sources, such as POL-on and PBN. They adhere to regulations that define the scope of data, as well as the rules for data entry, updating, and

access [56, 47]. Information is entered into the system only by authorised entities (e.g. universities, scientific institutions, or government ministries), which ensures its quality and reliability. The RAD-on portal shares only data that is legally deemed public.

Despite the considerable number of registers available, the list of science and higher education institutions plays a special role. The list encompasses the following entities specified in the Act [56][26]:

- public higher education institutions
- nonpublic higher education institutions
- church institutions
- scholarly institutions
- federations of institutions.

Institutions have individual profiles that contain their basic data (e.g. their names, identification numbers, heads, supervising bodies, and statuses) and histories of changes, which are available for some fields. Users can also browse detailed data using links to subpages that present selected aspects of the activities of particular institutions. The profiles of all institutions include links to all related information, including:

- branches where institutions operate (outside their headquarters)
- study programmes and doctoral schools
- staff
- scientific and artistic activities (publications, artistic achievements, patents, or projects)
- investments related to education and scientific activities, including scientific research equipment and IT infrastructure whose value exceeds PLN 500,000
- ongoing promotion procedures related to the awarding of the degrees of doctor and habilitated doctor
- evaluation of the quality of the scientific activities of institutions (see Chapter 1.).

Regardless of the linking possibilities between resources, users can browse each collection of data separately using the advanced filter section. Additionally, they can download data in the format of their choosing (XLSX, CSV, or PDF) while retaining their filters. The data included in the collections is also available for machine download using the Application Programming Interface (API) services (see Chapter 5.).

In the long term, consolidating data into a centralised location and implementing custom solutions that cater to users' specific needs and degrees of analytical skill can greatly enhance data accessibility, improve its overall quality, and ensure that it remains up to date. With a growing number of users comes improved control of data. For example, academic teachers have the ability to verify the data that institutions have entered into the POL-on source system and subsequently published on the RAD-on portal.

[26] Items 1, 2, and 4 through 8 of Article 7.1

If discrepancies are detected, teachers can notify the relevant institutions of the need to correct the data. Institutions whose data is published on the portal prioritise the data's completeness. This drives them to update their data more frequently in their source systems. Most data collections are updated daily and some registers are updated hourly. This enables users to retrieve accurate and current information efficiently.

2.1.2. Reports

The comprehensive process of creating interactive reports using the system is discussed in greater detail in Chapter 3. It is important to mention at this point that the navigation dashboard that contains the reports is updated constantly and, at present, consists of thirteen categorised folders. The folders contain thematic reports on higher education institutions, student recruitment, students, study programmes, graduates, PhD students, academic teachers, and research on artificial intelligence (Figure 3.2. presents some of the folders).

2.1.3. Analyses

The analyses, developed by scientific experts at OPI PIB and downloadable as PDF files, are dedicated to various aspects of higher education and science. Some of the studies are extensive and involve cross-sectional statistical analysis. Currently, approximately a dozen analyses (see Figure 2.3.) are available. They are divided into the categorised folders 'science in Poland', 'higher education in Poland', and 'scientists in Poland'. Individual analyses supply readers with information on:

- the financing of the research and development sector in Poland
- the scientific centres that are at the forefront of research on artificial intelligence
- the financial circumstances of higher education institutions
- the situation of the humanities in Poland
- the countries of origin of foreign students in Poland
- the number of foreign academic teachers in Poland
- the number of women who pursue technical and IT programmes at Polish higher education institutions.

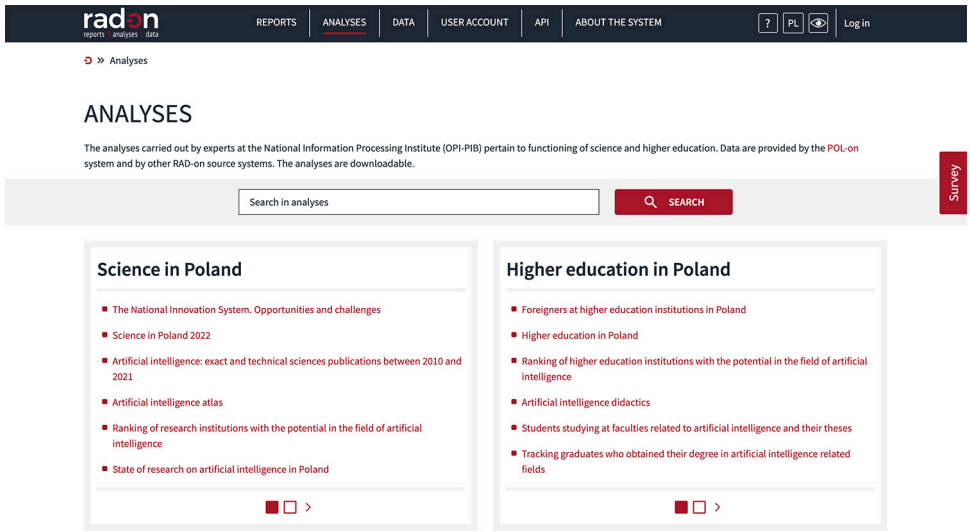


Figure 2.3. The Analyses module of the RAD-on portal.

2.1.4. Search engine

The Elasticsearch-based full-text search engine [3] analyses source databases and facilitates the discovery of semantic correlations between data.

Users can browse search results from the Data, Analyses, and Reports modules, as well as selecting specific categories or subcategories of information in which they are interested (e.g., category: scientific and artistic activities, subcategory: projects—see Figure 2.4.).

The search results provide access to various details, including project profiles, institutions, and research reports that pertain to a specific area. Moreover, every section is equipped with its own search engine, which enables users to search resources within a particular domain with ease.

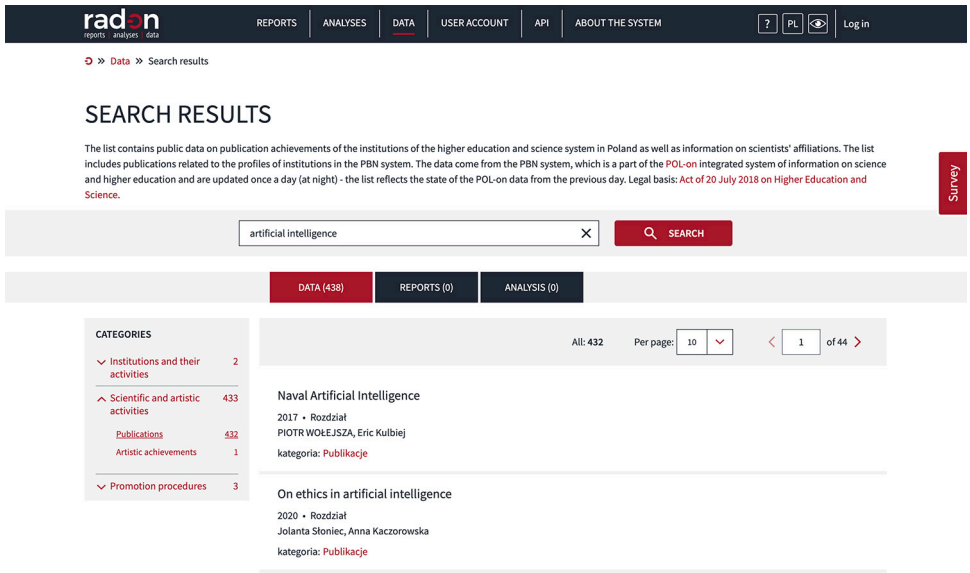


Figure 2.4. Search results on the RAD-on portal.

2.1.5. User account

When opening or sharing data, it is crucial that special measures be taken to protect personal data. To adhere to the regulations set forth by the General Data Protection Regulation (GDPR) [46]), it is imperative that IT infrastructure provides users with the necessary functionalities to access, correct, delete, restrict, and object to the processing of their data, as well as transferring such data to other controllers [8]. To meet these requirements, we developed a RAD-on service that grants users access to their data. Using this service, users can verify their data that has been collected from various source systems in one convenient location.

Due to the scope of data that is processed in the source systems, the data access service is intended primarily for individuals who function in the field of higher education and science, such as students, academic teachers, researchers, experts, and reviewers. Users who are logged in to RAD-on can use their accounts to verify whether their data is being processed in systems and databases related to higher education and science in Poland. These include POL-on, Inventorum, Polish Science, the Support System for Selection of Reviewers, and the Polish Scholarly Bibliography (PBN). The systems are described in greater detail in section 1.6.). Users can then download individual reports that present all of their data that has been processed in each of the systems. The reports provide details on a variety of subjects, including study programmes and scholarships awarded, scientific projects and publications, employment history in higher education and science, and academic degrees and titles awarded.

The data access service relies on two solutions:

- **the data warehouse**, which combines, deduplicates, and aggregates data on science and higher education from all domain-specific systems. Personal data from various transactional systems is sent to the data warehouse and then integrated with individuals' profiles
- **the central logging module (CLM)** grants users access to the service. The CLM is integrated with the National Electronic Identification Node[27] (*Krajowy Węzeł Identyfikacji Elektronicznej* – KWIE, login.gov.pl).

The data access service uses the data that is aggregated in the warehouse, which means it uses the profiles of individuals. The service is available via user accounts. To ensure that all nonpublic information on citizens is properly secured, access to the service requires authentication via the login.gov.pl system. The integration with the KWIE allows the CLM/RAD-on system, acting as a service provider, to identify users at all identity providers that are integrated with login.gov.pl. Identification systems that are connected to the node must adhere to the most stringent security protocols. Users' identities can be authenticated through various electronic identification methods, including trusted profiles, e-ID cards, and banking systems, as chosen by the users. The primary objective of the service access model is to ensure that personal data is accessible only to individuals to whom data pertains (data subjects).

The user identity authentication process is based on the federated identity model[28] of login.gov.pl, which is presented in Figure 2.5.

Figure 2.5. The federated identity model.

The CLM/RAD-on initiates contact with an identity provider (IdP) via login.gov.pl to verify the identity of a user who seeks to use the citizen data access service. The IdP's system releases the user's personal data upon receiving a request from an external system (the service provider) and verifying the user's identity. Based on the personal data obtained from the IdP, the data warehouse verifies which source systems process the user's data. The list of such systems is then presented to the user. The list may vary, depending on whether the logged-in individual is a student, a reviewer, or an author of a publication. Figure 2.6. displays a sample view generated for a user whose data is

[27] gov.pl/web/login (accessed 2 March 2023)
[28] The federated (distributed) identity model is based on multiple electronic identification means that are issued by various public and private entities. Public electronic identification means include trusted profiles and eIDs. Commercial electronic identification means include electronic banking.

stored in POL-on, Polish Science, Polish Scholarly Bibliography (PBN), and the Support System for Selection of Reviewers (SWWR).

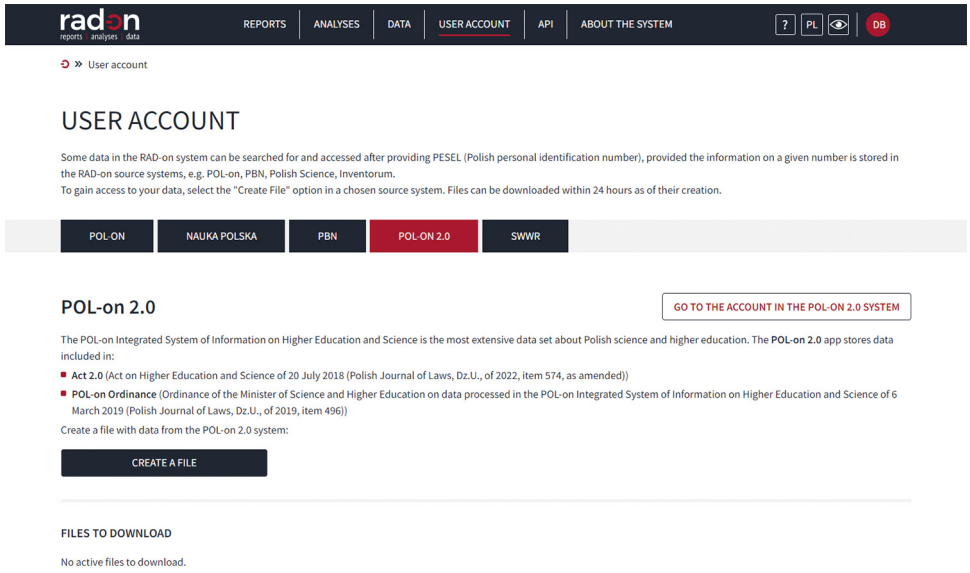


Figure 2.6. The User account module of the RAD-on portal: the view after logging in.

In some cases, none of the systems will be listed, which means that the person's data was not found in any of the RAD-on source systems. can request and download individual reports that contain their data from each system. After a request to create a report with data from a selected system is registered, the service retrieves the data from the warehouse and provides it to the user. Relevant files can be downloaded within twenty-four hours of their creation in two formats: PDF (recommended for well-structured presentation of data) and JSON (a machine-readable format that is ideal for further processing activities, such as data analysis or transmission). Each request to create data from the system regarding a specific individual and each document download is recorded in the data warehouse's history. The simplified architecture of the service is presented in Figure 2.7.

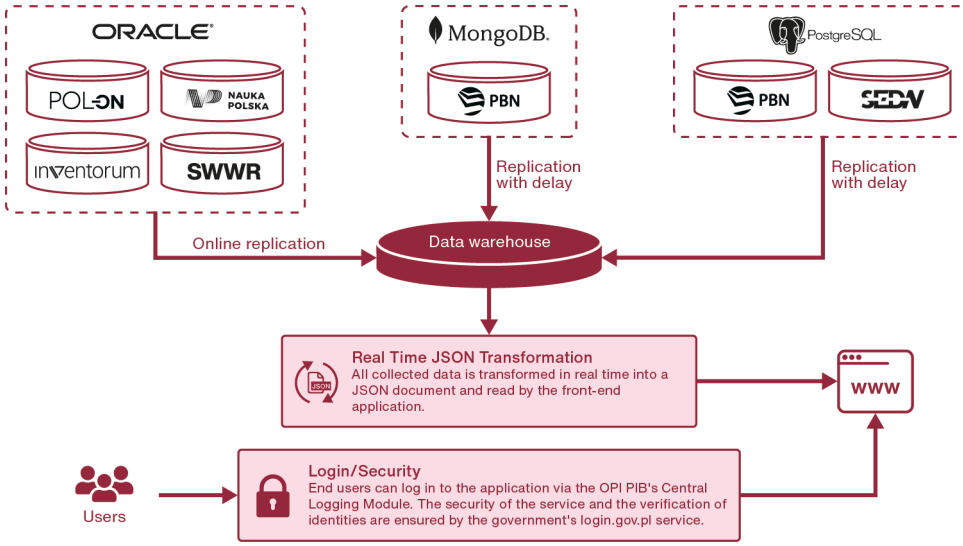


Figure 2.7. The simplified architecture of the citizen data access service.

After downloading data, users can verify whether it is accurate, complete, and up-to-date, and, if necessary, react to any discrepancies. The service is integrated with the ZSUN Helpdesk system[29], handles requests and errors, and through which users submit requests that pertain to their data, such as requests to correct or delete it (exercise their right to be forgotten). All user requests are analysed carefully. If it is determined that a request is relevant, the data is modified in the appropriate domain systems. It should be emphasised, however, that not all requests can be granted. For instance, a condition[30] applies to the POL-on system that precludes users from exercising their right to be forgotten and, in consequence, from having their data deleted.

Due to the implementation of the data exchange model, updated data is fed into the integrated systems and published on the RAD-on portal as part of the services provided. Citizens have the power to impact the accuracy of data related to science and higher education in Poland directly. The service that allows users to oversee the correctness of their data enables us to deliver the most current, reliable, and credible information on science and higher education. Simultaneously, we can meet the obligations that arise from the GDPR and contribute to the further digitalisation of public administration.

[29] lil-helpdesk.opi.org.pl (accessed 2 March 2023)
[30] The condition is specified in letter b) of Article 17.3 of GDPR, according to which the right to be forgotten cannot be exercised if data processing is necessary for compliance with a legal obligation which requires processing by Union or Member State law to which the controller is subject or for the performance of a task carried out in the public interest or in the exercise of official authority vested in the controller.

2.2. The architecture of the portal

The overall architecture of the citizen portal is presented in Figure 2.8. The portal comprises the following elements:

- **citizen portal front end:** Angular JS[31]
- **citizen portal server:** JAVA Spring[32]
- **reporting system:** JAVA Spring
- **S3 analyses:** the S3[33] file server, where files with analyses are stored
- **citizen data server:** JAVA Spring.

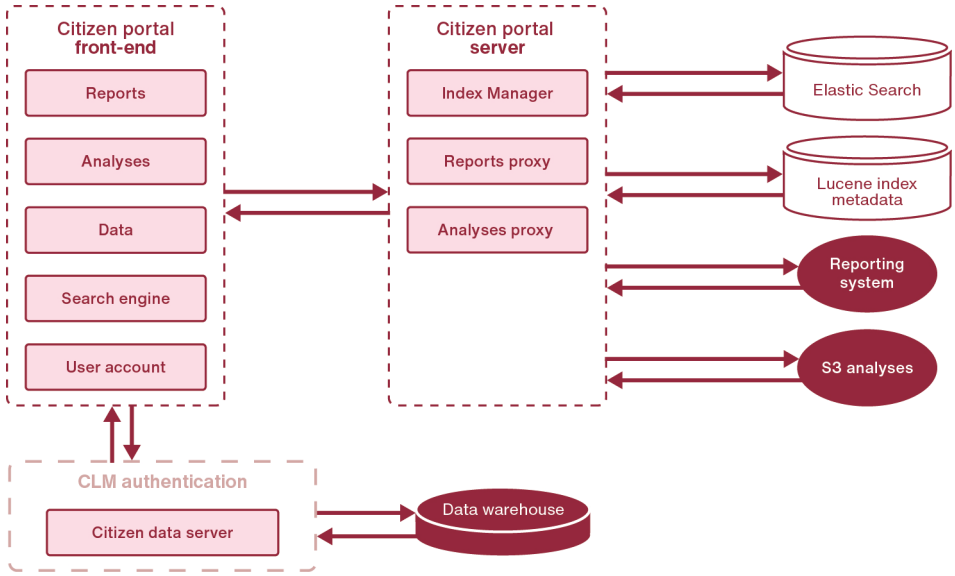


Figure 2.8. The overall architecture of the citizen portal.

Users gain access to a web application (**citizen portal front end**) that consists of five modules: Reports, Analyses, Search engine, Data, and User account. The modules communicate with the server part through an API, which is described in section 5.3.3. We will now discuss how each module communicates with its server-side counterpart.

The Reports module, which presents interactive reports, is fed with data from the **reporting system** (see Chapter 3). The web application communicates with it using a proxy (**the citizen portal server**). Both the metadata and the JSON report data are available

[31] angularjs.org (accessed 2 March 2023)
[32] docs.spring.io (accessed 2 March 2023)
[33] min.io (accessed 2 March 2023)

through the API. After downloading the metadata, the web application displays a list of reports available to the user. If the user decides to view any of the reports, detailed report data is downloaded. Charts are generated based on the downloaded JSON data. The interactive form of the reports is ensured by sets of filters, which can be customised by the user as needed. Filters are also managed using the API.

The citizen portal server also acts as a proxy server for the Analyses module. As with the reports, the API provides metadata on available analyses. If a user attempts to download a file that contains one of the available analyses, the web application sends a query to the API and the file is downloaded. The use of metadata in the Reports and Analyses modules enables easy publication of new resources without the application code having to be modified. In other words, the content in the Reports and Analyses modules can be modified dynamically by analysts without the need to deploy a new code version.

The Data module is fed with the records that are indexed on the Elasticsearch (ES) cluster. Technical details regarding index management by the Index Manager are discussed in sections 5.3.2. and 5.3.3. In addition to displaying data, the system also enables users to download it in PDF, CSV, and XLSX formats. Transformation of records from the source JSON format to CSV and XLSX formats is handled by the server part of the portal. PDF files are generated by the web application.

The search engine enables full-text searching of data available in the Reports, Analyses, and Data modules. In the cases of reports and analyses, an additional Lucene [5] index is created, which is managed by the server part of the citizen portal. Documents in the index are created based on metadata on reports and analyses. Information that is presented in the Data module is indexed on the ES cluster and managed by the Index Manager (for more information, see section 5.3.2.). The user's query sent through the API is delivered to both indices. Results for different modules are not combined. Results that comply with the client's query are presented in separate tabs.

The last of the modules is User account. It presents data that can be viewed only after a user logs in. Authentication is handled by the CLM. After logging in, a user can download their data in PDF and JSON formats through the **citizen data server** application. The data for a user's account is sourced from the data warehouse, with which the server application communicates directly. The server application also generates PDF files, which are created from the source JSON files.

CHAPTER 3

REPORTS

Dr Anna Knapińska
Dr Aldona Tomczyńska
Dr Sławomir Dadas

3.1. Business intelligence tools and analytical platforms

The reporting system discussed in Chapter 2 is the primary analytical tool available on the RAD-on portal.

This chapter demonstrates the reporting system's usefulness in data processing. It also presents examples of interactive reports generated by the portal, which were prepared using the reporting system. However, we must first identify the differences between independent analytical platforms and commercial business intelligence (BI) tools.

The rapid growth in the quantities of stored and processed digital data has increased the popularity of systems that are used to analyse and visualise such data. These sophisticated solutions combine various technologies at all stages of data analysis to satisfy specific business needs. They store, manage, and prepare data for analyses which, once completed, yield valuable information.

An example of such solutions can be found in the BI applications used to analyse data, and to discover new and useful insights, which help businesses develop or maintain their competitive advantage. The term 'business intelligence' was coined in 1989 by Howard Dresner from the Gartner Group [40]. At present, the chief developers of BI software include Microsoft, Tableau, and Qlik [44]. Large enterprises are the primary users of BI software, whose features allow them to analyse their businesses. Therefore, most users of such tools are corporate employees.

If we consider the importance of analytical systems from the perspective of decision-making processes [63], it must be noted that they are key not only for the private sector, although that is where they are primarily used [16]. They also form a crucial component of the public sector, which faces similar challenges in processing, storing, and analysing increasing volumes of data. Sophisticated analytics make decision-making processes more effective, which enhances the efficiency of various areas of public management and specific institutions [66].

The same applies to the science and higher education sector, in which higher education institutions and other research entities compete for students, scientists, and research funds. Having and processing detailed information is crucial for attaining competitive edge, but also gives rise to new challenges, such as data security concerns and higher costs. It is undoubtable, however, that both individual scientific institutions and entities responsible for science policy must use data extensively to make informed decisions and improve the effectiveness of their operations (see [25, 49, 52]). This is yet another dimension of the steadily accelerating digital transformation that has become part of academic culture [67]. Future prospects, including possible scenarios and consequences of potential decisions, are as crucial as current knowledge [18].

BI software offers solutions from descriptive, predictive, and prescriptive analytics. To public institutions, the greatest flaws of such software are its prohibitive licence fee and other nonlicence costs [28]. Tools are usually priced in two ways: per user licence and per device (CPU) licence. Likewise, the costs of BI cloud solutions are high and often unpredictable. Another crucial problem is the inaccessibility of software source codes, which excludes the possibility of tailoring the services to the specific needs of end users.

As a result, access to data is becoming severely limited, which violates the open access policy—one of the most important principles in the Organisation for Economic Cooperation and Development (OECD) and European Union (EU) countries. In Poland, the act on open data and reuse of public sector information of 11 August 2021 lays out the principles of data openness and the principles and methods of sharing and forwarding public sector information for reuse, as well as specifying the entities that share or forward such information. Of particular importance is the systemic change in the sharing of scientific data and knowledge, which improves the quality of research and ensures that it better addresses social needs. That is why the frameworks of Horizon 2020 and Horizon Europe, the largest scientific research and innovation undertakings in the EU's history, require the beneficiaries to manage their research data in a manner that makes it findable, accessible, interoperable, and reusable (FAIR) [53] (for more information, see section 1.1.). Another noteworthy initiative is the European Open Science Cloud, which focuses on advanced tools and resources for data storage, sharing, and processing [9].

The costs of commercial systems on the one hand and the pressure to implement the concept of public data openness on the other have resulted in the public sector developing open analytical platforms. In science and higher education, such platforms serve as support for both policymakers and researchers. They offer more efficient methods for the collection of data and statistics and the facilitation of information-based decision-making. They also ensure that interoperability standards are met to enable comparative studies—such as evaluations of scientific achievements—at the local, national, and international levels. Databases and dashboards that present public statistics pertaining to scientific research, innovation, and higher education are delivered by, for example, Eurostat and the OECD (see Chapter 1). In Poland, open information on higher education and science is provided by RAD-on [36, 41].

The analytical platform offered by OPI PIB is accessible through any internet browser and does not require any licensing fee. This makes it an attractive option for those who

seek alternatives to commercial BI software. In addition to analyses and data, RAD-on offers a dashboard in the form of reports that contain interactive tables and diagrams, which can be used to create evidence-based policies and align with the government's vision of an open access platform that provides the public with credible data [7]. These reports are enhanced constantly with the aid of a special engine, which is described below.

3.2. Architecture of the RAD-on analytical platform

The reports engine[34] is a service that supports the generation of HTML pages and output files that contain graphical representations of data. The data is edited by data analysts to create specific reports, which can be published on public and private websites. The data presented in a single report can derive from multiple queries that refer to various relational databases and flat files, such as CSV and XLSX. To manage these dependencies, the architecture of the analytical platform is divided into four layers of abstraction: connectors, cache, queries, and reports (cf. Figure 3.1.).

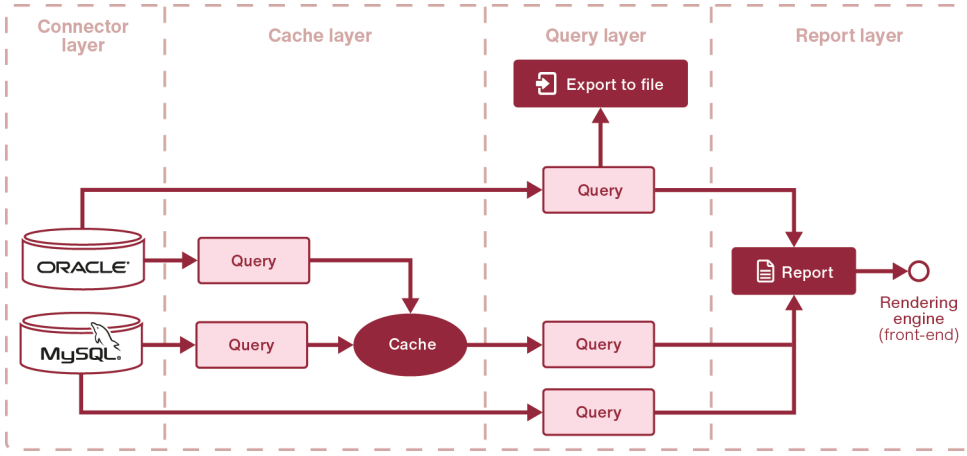


Figure 3.1. The layers of the reports engine.

[34] The reports engine was also used to create the GENDERACTION Data Dashboard (genderaction-data-dashboard.opi.org.pl (accessed 2 March 2023)) as part of the project, GENDer equality in the ERA Community To Innovate policy implementatiON. Coordination and Support Action, funded under the Horizon 2020 framework programme. The dashboard displays statistics that pertain to gender equality in Europe.

3.2.1. Layers of the system architecture

Each layer is responsible for managing one type of object. The responsibilities of each layer are as follows:

1. The **connector layer** is responsible for defining and managing connections to external data sources. WIn the case of a relational database, a connector is a pool of connections to any database, such as Oracle or MySQL.
2. The **cache layer** enables faster access to data. The analytical platform enables the creation of reports and the direct execution of queries using source databases. A local H2 database, which can store copies of data from external sources, eliminates the need for continuous querying of the source servers. Technically speaking, this process is executed in the following way: after receiving a request to create a cache, the system creates a database table with a temporary, random name (e.g. TMP_-36847234764). Next, the data returned by the source query is copied into this table. Due to the query's result containing no information about the source tables' metadata, the cache will hold no primary or foreign keys, nor indices. As soon as all data is copied correctly, the table's name is changed to the correct cache identifier. In the case of caches that are subject to cyclic refreshing, the process is repeated. As a result, when data is refreshed, the cache is unavailable for a short period while the old version of the table is deleted and the temporary table's name is changed to the target name. Before the cache is refreshed fully, it remains possible to use its previous version. If the refresh procedure fails for any reason, the cache version from the last successful refresh remains active. It is worth noting that from the reports engine's perspective, the cache layer is no more than another type of connector; one that bears the 'cache' identifier. Queries that are addressed to it are defined in the same way as those to any other database connector are.
3. The **query layer** enables the definition and execution of queries. A query object consists of a reference to a connector object, query content (SQL), and an optional set of specific parameters. If a query is parameterised, the values of the defined parameters should be transferred during its execution. This enables the creation of filters that can be used in unique reports. Query execution requests should also contain one or more definitions of the so-called outputs that instruct the system how to process the query result. The most frequently used outputs include returning the data in an HTTP response, saving it to a file, and returning the query statistics. Outputs also allow for direct interaction with the query layer. The system uses internal data transfer to the next layer, i.e. to the report that utilises the query results.
4. The **report layer** contains report definitions. Report specification is a complex object for which a special set is defined. This set contains a list of queries to be used in the report, a report parameters list that matches the parameters of previous queries, and a report section list. During the report generation process, specification objects are transformed into result objects that contain data to be presented to end users. The engine (the front end) transforms the result objects into ready-to-use pages that contain reports with data, which can be presented to users.

During the report creation process, data analysts are typically responsible for delivering SQL queries or XLSX files that contain the appropriate data. Based on this, the system creates the proper cache. The next query from the buffered data transforms the simple data table into a dynamic report with filters.

3.2.2. The report layer

A report is an object that integrates elements defined by the application's previous layers. Interdependencies exist between these objects and those defined inside a report, such as parameters or sections. To understand how the report layer operates, these interdependencies and their implications must be understood: changes in report elements often require changes in other dependent elements. Report specification comprises the following elements:

1. The **query list** is downloaded from a predefined data source. The query list includes all queries used by a report that originate from the query layer. The result of executing a single query may be used by multiple elements of the report simultaneously. For example, it can be used to draw diagrams and tables, as well as supplementing dictionaries for the report's parameters. If multiple elements use the same query, it will be executed only once.
2. The **parameter list** is the sum of all parameters from the queries declared. For example, if a report contains two queries, Q1 and Q2—where Q1 contains parameters A, B, and C, and query Q2 contains parameters B, C, and D—the report should contain the definitions of at least four parameters: A, B, C, and D. The set of data required to supplement the report's parameters is broader than in the cases of the queries themselves. The elements to be supplemented include, but are not limited to, the parameter's label, the default value displayed when the report is launched for the first time, and the list of dictionary values that individual parameters may assume.
3. The **list of sections** comprises elements of the report that are visible to end users. Each section has a dedicated specification (which describes how it is to be presented and what data it is supposed to use) and a resulting representation (a generated set of information that enables specific sections to be presented to users). For example, the specification of a diagram section may contain information stating that the diagram consists of two series, and that the data for them should be downloaded from query A and query B. The resulting representation returns a list of series already populated with data, but without information about the queries from which this data originated—they are irrelevant to the end user. Many types of section exist, and they will be discussed in detail in subsequent sections of this chapter.

This layer is furnished with various features, such as filters, data visualisations, and text, which facilitate the creation of ready-to-publish reports. The sections that can be used to present data are described below.

3.2.3. Section types

The structure of each report browsed by end users is determined by data analysts. Given that different section types have different functions, each section contains a unique set of fields that control its behaviour and manner of presentation. The RAD-on reports engine contains the following sections:

1. The **text section (TEXT)** is used to display the panel with text description (e.g. introduction). This section accepts HTML code, which enables text formatting.
2. The **filter section (FILTERS)** is used to interact with reports by filtering their parameters. This section's specification contains information on what types of element should be displayed (e.g. multiple choice, text, selective filtering, autocomplete) and with which report parameters they are associated.
3. The **chart section (CHARTS)** is used to display various visualisations (column, surface, plot, pie, and radar charts, and maps) that contain data from queries declared in the query list. A visualisation may illustrate one or more data series, and each series may originate from a different query. A data series is defined as a list of (x, y) points, where x is the point's label, and y is its value. Some charts support an additional data dimension (x, y, z). In such situations, value z may be visualised, for example, by the bubble's size on a bubble chart. Maps are a specific type of chart. They contain numeric values that are indexed by names or codes of geographic areas (countries, provinces, or districts).
4. The **table section (TABLE and CHART_TABLE)** is used to present data that originates from tabular queries. In the case of TABLE, the data displayed is identical to that returned by the query; in the case of CHART_TABLE, it is downloaded directly from the chart. In the latter case, the table's columns will assume the names from the list of labels of the given chart and the values from the individual data series. Optionally, an additional column can be displayed that aggregates the values from all series.
5. The **summary section (SUMMARY)** is used to display numeric values that summarise the report and that originate from the relevant columns in the queries declared in the query list. As summary tiles may contain fluctuations of specific values as a function of time, it is good practice to assign queries to them that return a single record with an aggregated value.

To recapitulate, the report creation process is relatively simple. Analysts prepare datasets using SQL queries or simple CSV files. The most important task is parameterizing the queries through which the reports' end users may interact with data. By using layers and sections, analysts can create a multitude of various reports and upload a ready-to-use online dashboard. In the following sections, we will discuss the reports generated by RAD-on, which serve as an example of an efficient dashboard.

3.3. RAD-on reports as an information dissemination and decision-making tool

A dashboard can be defined as a visualisation of the crucial information required to achieve specific goals, consolidated and distributed in a manner that enables data monitoring at first glance ([19], p. 26) or as a set of visual resources that enable recipients to understand the data displayed and/or analyse it [59]. Dashboards' key strengths include quick access to diverse data, drill-to-detail, analysis of trends as a function of time, and assistance in institutional and strategic decision-making processes [35].

With these strengths in mind, we will now describe two sample reports that were designed using the reports engine and were implemented in RAD-on: Students: Comparison of Higher Education Institutions, and Students: Comparison of Groups of Higher Education Institutions, both of which can be found in the 'Students' folder (see Figure 3.2.). Although the reports will be presented from the user perspective, we will also discuss what actions must be performed by analysts to achieve the final result: a dynamic report.

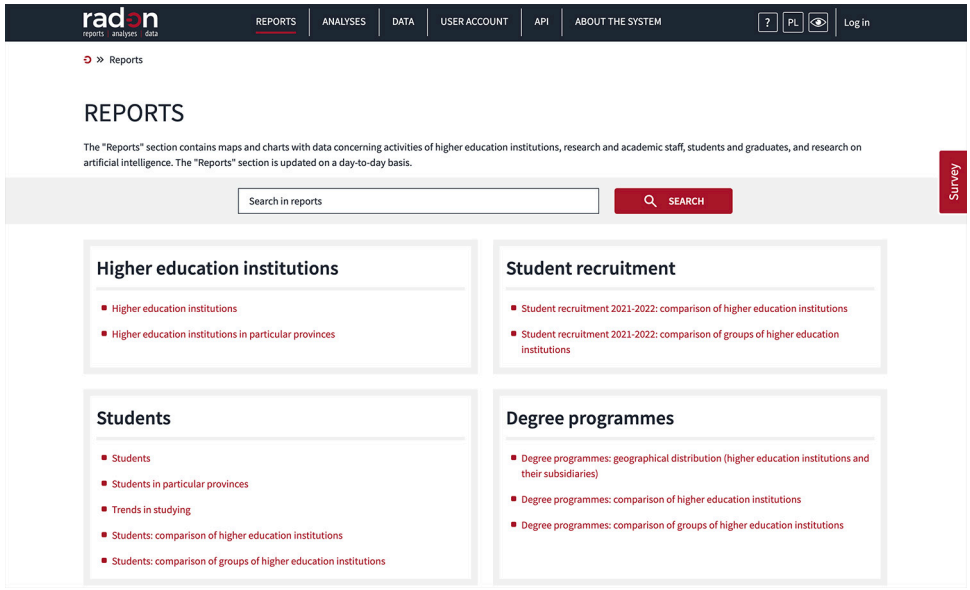


Figure 3.2. The 'Reports' module of the RAD-on portal.

The guiding principle behind both reports is that they should enable comparison either of individual higher education institutions or of groups of higher education institutions, relative to student characteristics. Users select the information that interests them in a number of steps. First, the users must specify the institutions for which data will be visualised. The following filters are available: 1) institution name; 2) city where the institution is located; 3) province; 4) city size, measured by the number of inhabitants; 5) institution size, measured by the number of students; 6) institution establishment date

(before 1939; 1940–1989; 1990–2004; after 2005); 7) institution type (public, nonpublic, church institution); and 8) institution profile (university or nonuniversity institution). The filters are interdependent, which means that by selecting a particular institution and clicking the 'Apply' button, the user will receive filtered information about that institution immediately. For example, if the user is interested in the Jacob of Paradies University (cf. Figure 3.3.), they will learn that it is a public nonuniversity institution founded between 1990 and 2004, attended by more than 1,000 but fewer than 5,000 students, and is located in Gorzów Wielkopolski, a city in the Lubuskie province with between 100,000 and 250,000 inhabitants. By analogy, should the user select only nonpublic higher education institutions in the 'Institution type' filter, the report will return all information about those institutions.

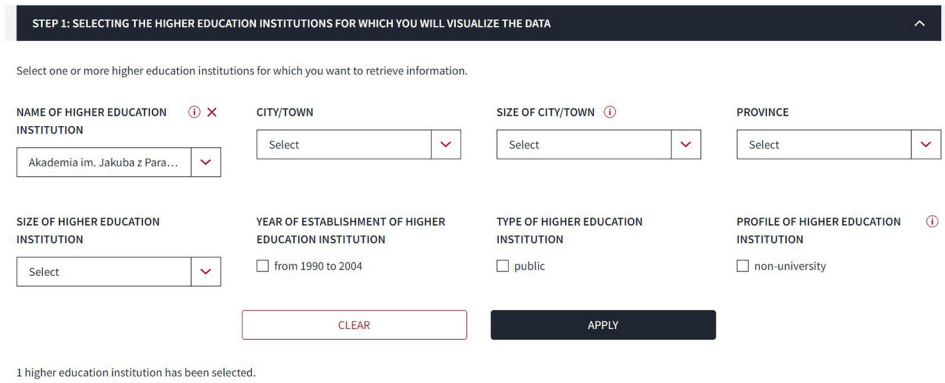


Figure 3.3. Example of interdependent filters.

When browsing information on the Jacob of Paradies University, if the user filters the data to display only first-year students, they will see eight data visualisations in the form of diagrams. The visualisations present the total number of students by sex, place of residence, origin, age, mode of study, study cycle, and study profile. In the case of reports used to compare groups of higher education institutions, the primary difference lies in the index values being calculated for the whole group, including the average value for groups of institutions. When hovering the mouse over the individual columns on diagrams that that present shares, the report will display numeric values that correspond to those shares. By clicking the legend entries, the data can be drilled, i.e. the user can toggle the display of selected data on the diagram on and off to widen or narrow the scope of information displayed.

Analytical tasks consisted chiefly in using the engine to define all essential text sections, or panels that make it possible to isolate the views with specific types of information (in Figure 3.3., it is the topic bar that describes the first step, but reports also contain introduction, filter, and methodological explanation panels). It was also specified which panels should be expanded in the main view and whether they should be expandable. Next, all of the necessary filters were set and tooltips were added for some of them (in Figure 3.3. there is a tooltip located next to the institution profile filter; it describes the

features of university and nonuniversity institutions assigned to that filter). The third analytical step involved creating visualisations, which included selecting the query from which the data for the diagram would be downloaded, selecting the type of visualisation (horizontal), and selecting the manner of data aggregation (sum, mean). The analysts decided whether to hide values on the diagram and whether the categories should be displayed in reverse order. They also assigned names to tabs and diagrams; defined what data should be assigned to the X and Y axes, as well as entering the names of the axis labels, and the minimum and maximum values to be marked on the axes; specified whether the values should be stacked, and whether scrollbars should be placed next to the diagram axes; and assigned colours to the diagram columns.

The RAD-on Reports section is improved on an ongoing basis. At present, it offers a selection of reports on higher education institutions, student enrolment, the students themselves, study trends, graduates, doctoral students, and academic teachers. All reports are available in the Polish and English languages. The data that is used for visualisation can be downloaded in XLSX and CSV formats, while the charts themselves can be downloaded as PNG image files. Individual reports are supplemented with expert commentaries that include abridged data interpretations and references to external documents and analyses that pertain to specific subjects. Report components help users to create custom data breakdowns and presentations, which can be used to make decisions at various levels.

3.3.1. Challenges in the report creation

It is undeniable that navigation dashboards offer numerous advantages; designing and operating them, however, is a complex process. If the emerging risks are not addressed, the platform loses the trust of its users. Dashboards that lack credibility are rendered useless. The examples of specific RAD-on reports presented below illustrate the key challenges [35] and the solutions necessary to overcome them.

RAD-on's chief advantage is the high quality of its data. In most cases, the reports are based on data that is uploaded to the POL-on system by higher education institutions for the purpose of reporting to Statistics Poland (*Główny Urząd Statystyczny* – GUS), which means that such data is part of public statistics in Poland. It is widely acknowledged that the quality of data and indices has a significant impact on the decisions made by users when selecting the appropriate tool to utilise [12, 23, 50]. The fact that most of the data is owned by the Polish Ministry of Education and Science and OPI PIB, as its administrator, eliminates the risk of distributed responsibility for the quality of the information [35].

Ensuring a high quality of output data is crucial, but presents a challenge in the subsequent stage of data analysis: the data must be properly cleaned and prepared [14]. In the case of RAD-on reports, this is ensured by the Data Science Team at OPI PIB's Laboratory of Databases and Business Analytics, which comprises analysts, statisticians, software engineers, and scientists whose research falls within the field of science and

higher education. The Reporting and Analysis Team from the same laboratory helps to ensure the highest quality of the data provided. Combining technical with expert knowledge on science and higher education results in well-constructed database queries that yield credible reports.

Another challenge lies in providing users with reports that can be used seamlessly by individual users from various groups, including science policymakers, administrators of universities and other scientific institutions, journalists, and prospective students (user-centred design [37]). Whether it is possible to create one-size-fits-all dashboards [54]) is debatable, but the 'customise, personalise, adapt' principle [59] remains prevalent and translates to a comprehensive approach.In the case of RAD-on reports, this means, first of all, that the indices and terms used must be explained (e.g. the differences between university and nonuniversity higher education institutions are obvious to ministerial employees, but might be confusing to other users). Although reports are consulted with UX specialists, there remain no guidelines for designing dashboards that display data from the public sector [1]. For example, the interdependent filters that we use in our project (see Figure 3.3.) might be unintuitive to users who are not accustomed to online shops or hotel booking systems. In our opinion, however, the filters constitute an additional source of information, and discontinuing them would be a mistake.

The problem of incorrect interpretation of data [35] can also be overcome by introducing additional tips for users. The Data Science Team at OPI PIB designs detailed methodological explanations that facilitate the comprehension of the reports (e.g. users are informed that students enrolled in multiple programmes are counted multiple times) and references to external sources (e.g. the help pages of the POL-on system). Data analysis can be made easier with the help of expert commentaries and the presentation of information in various formats (e.g. visualisations and tables), as appropriate). Moreover, user suggestions submitted after the implementation of the reports are analysed and acted upon on an ongoing basis.

Another type of risk relates to dashboard updates [4] and the limited labour and time available to control the process. Preparing reports for production implementation—particularly when their formats differ from existing reports—is a time-consuming analytical process that can last months. Errors must be fixed rapidly and updates must be deployed regularly, or users will perceive the dashboard as noncredible and useless. RAD-on reports are updated only once a year, and present information as at 31 December (after higher education institutions submit their reports to Statistics Poland and after the Polish Ministry of Education and Science verifies them). Despite the process being automated to a degree, adapting it to changing legislation remains a challenge (e.g. when the minister of education decides to add new disciplines to the existing classification of scholarly disciplines and domains, analysts have to 'copy' the previous disciplines). Ongoing changes raise new questions that might require additional data, which increases the workload and extends the time required to complete the tasks.

The reports engine also requires maintenance and updates. The team of software developers is tasked with enhancing the engine's functionality to streamline the analysts' work and to provide users with improved information. The latter includes new dis-

trict maps, which make it possible to study the specifics of higher education in Poland in detail greater than that provided by the province map. For example, in addition to the well-known fact that the greatest number of higher education institutions can be found in the Mazowieckie province, users can see what institutions—including their secondary branches—are located in each district of the province, which provides a more comprehensive view of the sector. However, when implementing improvements, developers must remember that users become accustomed to interface layouts and dislike frequent feature changes [65].

The last of the key risks is limited utilisation of dashboards by the users for whom they are designed [35]. RAD-on reports are promoted heavily by OPI PIB in the mass media, on social media, and during scientific and industry-specific conferences. Regardless, the key task of the Data Science Team is to provide users with reports that are well-designed, well-described, and easy to use.

This chapter presents an alternative to standard BI applications that may be developed and used by organisations of various types. Their importance continues to grow—although in some cases, such as those described below, analytical platforms developed by IT teams may be the better choice.

Analytical platforms are preferable in situations in which the potential number of end users is large and unpredictable—for example, when an open data dashboard is planned for release on a publicly-accessible website. With unlimited numbers of end users, licensed commercial BI tools are much more expensive.

Analytical platforms prove their worth when it becomes necessary to adapt to the changing needs of end users. Each type of BI software offers a unique set of features, which is usually fixed. Analytical platforms are flexible and expandable, which makes them more adaptable to changes.

Analytical platforms are much more useful when the dashboard's graphic design is key—for example, when it must match the appearance of another website. Most BI tools offer limited graphic layouts, while analytical platforms are easier to integrate with existing designs.

All of the considerations above apply to the RAD-on reports discussed in this chapter.

CHAPTER 4

THE DATA WAREHOUSE AND THE DATA EXCHANGE MODEL

Łukasz Błaszczyk
Sylwia Ostrowska
Emil Podwysocki

4.1. The data exchange model

One of the key goals of the project was to improve access to scientific and higher education data in accordance with the open government data principle. When data from the given sector is distributed (i.e. there are multiple discrete systems, databases, and microservices that process data for various purposes), actual improvement in data access requires not only that users be provided greater quantities of data from official sources, but also that they are provided such data in one place. Pursuant to the FAIR principle (see section 1.1.), data should be findable, accessible, interoperable, and reusable; therefore, data must be integrated and uniformed. In the RAD-on system, the integration tier (in addition to the data warehouse) is ensured by the data exchange model (DEM)—the central element of the system, whose design and implementation were key to the project's success. This chapter presents the process of selecting the right technology for the DEM, as well as its framework and architecture. This is supplemented with examples of practical implementations.

4.1.1. Technology

The DEM is responsible for the integration of source systems and for the provision of data for machine services (such as the RAD-on public API, which ensures access to data from source systems), for services that are available through the RAD-on portal (such as public data comparisons), and for other systems that use the data. DEM participates indirectly in the implementation of all system services. That is why selecting the right technology was a key technical decision from the outset of the project. To identify the best solution, we analysed three technologies that could potentially be used to develop DEMs:

- Mule Enterprise Service Bus (ESB)[35];
- Spring Integration[36];
- Apache Kafka[37].

Each technology was reviewed in terms of its stability, scalability, programming limitations, ease of development, ease of implementation, documentation, asynchrony, integration with databases, integration with various web services, integration with Apache Hadoop[38], monitoring (graphic design tools), global logging and monitoring of queries, and monitoring of service availability. The review was supported by the implementation of practical solutions, which enabled evaluation of each technology's applicability in this specific project. After conducting a thorough analysis, we departed from the conventional service-oriented architecture (SOA) approach, instead opting to use the Apache Kafka message queue system to address our communication needs.

Apache Kafka is based on a public subscription model that mediates the exchange of messages between applications (systems) and enables their propagation in real time. Originally, the system was designed for the LinkedIn[39] website to act as a centralised platform for event piping in online data integration tasks [58]. Currently, it is available as open-source software. As Kafka can operate on any number of servers, irrespective of their locations, it is also malfunction resistant. Integration based on Kafka ensures reliable and secure communication between modules and systems. The software comes with embedded mechanisms that replicate and protect messages against loss in production environments (c.f. [26, 33, 64]). In the context of RAD-on, Kafka's scalability and impressive catalogue of practical applications are unquestionable advantages.

4.1.2. Key concepts of the data exchange model

The DEM was built using the Apache Kafka open-source software, which enables the publication of and subscription to data using queues of ordered messages. The application receives data records and stores them for any system that is interested in downloading them. It creates queues of messages by arranging the records received in the order they were sent by the source system. Each data record, called a message, contains a key, a value, and a timestamp, which enable the systems that download the data to decide whether the message must be read and processed. A single message sent by the source system is a set of data (byte table) that is important to the recipients. The recipients may store or process that message to perform their tasks. Kafka mediates in this exchange of messages between applications (systems) and enables

[35] mulesoft.com/resources/esb/what-mule-esb (accessed 2 March 2023)
[36] spring.io/projects/spring-integration (accessed 2 March 2023)
[37] kafka.apache.org (accessed 2 March 2023)
[38] hadoop.apache.org (accessed 2 March 2023)
[39] about.linkedin.com/pl-pl (accessed 2 March 2023)

their propagation in real time. Figure 4.1. presents the process flow diagram of a Kafka-based solution, including all components that participate in the exchange of messages.

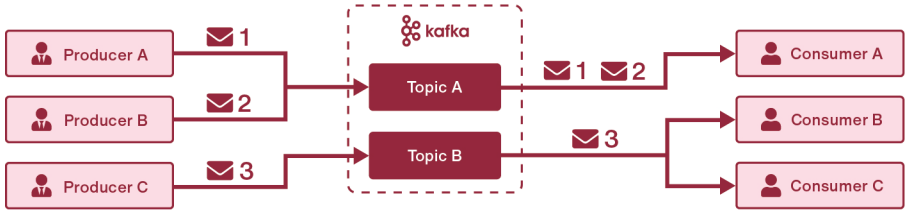


Figure 4.1. A general diagram of the DEM based on Apache Kafka.

Apache Kafka combines two data exchange models between distributed systems. First, data is made available in the form of an ordered message queue, which enables all records to be read by any number of recipients. Second, Kafka integrates the queue-based communication model with a subscription one. The subscription model assumes that each system that is interested in receiving data from the source system may subscribe to messages shared by Kafka. Subscription enables data to be forwarded only to the systems that use or process it. Moreover, Kafka allows the same message to be sent to multiple recipients (consumer groups), and provides mechanisms that enable parallel processing of the same message by multiple recipients. The main terms used in the context of the technology's application are illustrated in Figure 4.2.

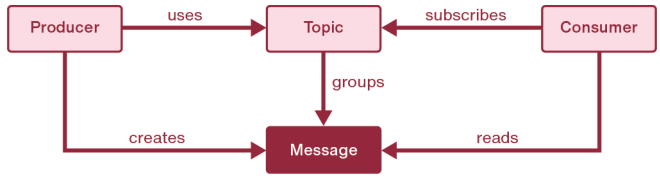


Figure 4.2. Primary object classes utilised by the DEM based on Apache Kafka.

The application that reads messages is called a recipient or a consumer, while the one that creates messages is called a sender or a producer. A key concept of the Kafka-based architecture is that messages have no predefined recipients that would be interested in them. The sender's chief responsibility is to classify messages by assigning them to specific topics and sending them to Kafka. Senders and recipients are linked to each other through topics. These are used to group together messages that, from the perspective of their recipients, are associated with one another or pertain to similar domain issues. A single topic may be used by multiple recipients, and this is one of the advantages of integration using Kafka. If a topic is created as part of system integration, each interested system may subscribe to it and receive related messages.

An architecture that utilises Apache Kafka is based on a central communication hub on which messages that are available to all interested systems are published. This so-

lution prevents the mutual querying of the systems and enables the systems to react dynamically to changes that occur in the reference system. Messages can contain data records that enable data updates in dependent systems that do not need to download full datasets recurrently and rely on time-consuming analyses of differences. Another significant advantage of Kafka is its scalability: it has proved to be a reliable solution for large distributed systems that are available as open-source software. The technology is malfunction-resistant because it may be launched on any number of servers installed at different locations. Kafka-based integration provides a mechanism of reliable and secure communication between modules and systems. Moreover, it comes with embedded mechanisms that replicate and protect messages against loss in production environments. Figure 4.3. presents the flow of data between communication participants using Kafka.

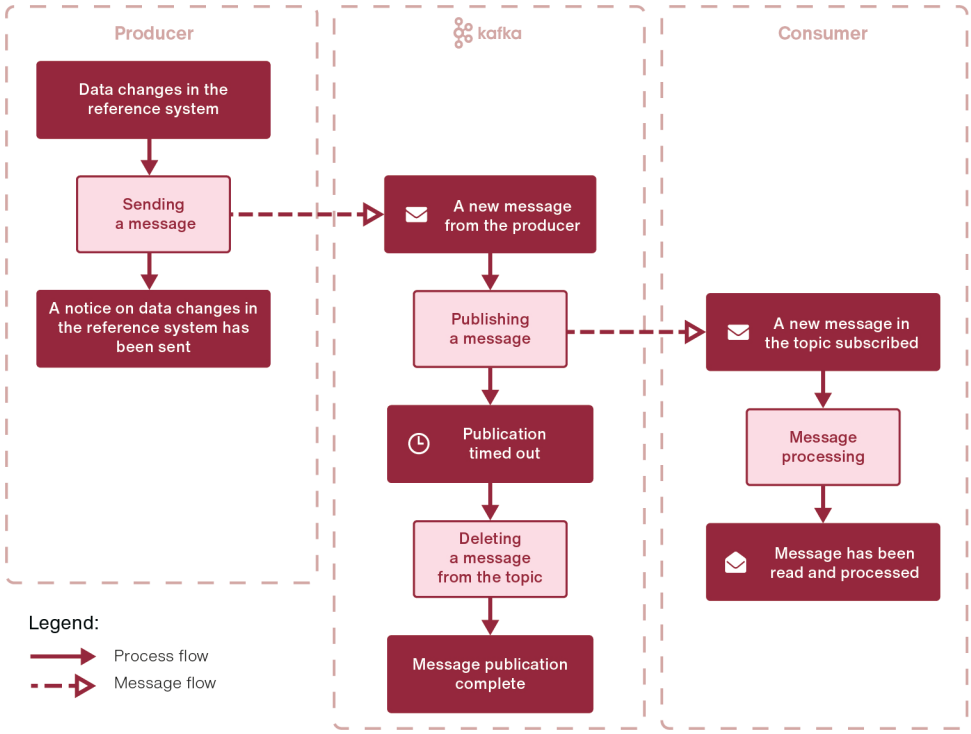


Figure 4.3. General diagram of data flow in the Kafka-based DEM.

4.1.3. Application of the data exchange model

One of the advantages of a DEM based on Apache Kafka technology is its broad scope of practical applications. By implementing a solution built upon the framework and architecture described above, we can handle multiple processes in various areas.

To illustrate the universal character of the DEM, this section discusses three examples of its application:

1. Sharing public data (on the RAD-on portal and through an open API)
2. Sharing data with domain-specific systems
3. Using the DEM in the process of handling admin messages

USING THE DATA EXCHANGE MODEL TO SHARE PUBLIC SCIENTIFIC AND HIGHER EDUCATION RESOURCES

The DEM plays a crucial role in sharing scientific and higher education resources on RAD-on. Depending on their needs and analytical skills, RAD-on's users may take advantage of various data sharing formats and services. The portal offers data breakdowns that feature extensive filtering capabilities, the option to download CSV and XLSX files, and full-text searches. Data is also available for machine download via public API services (the portal's architecture and its scope of services are described in Chapter 2.). Both the browser and the web services associated with it are based on the DEM. The source data that is shared publicly is aggregated by a data warehouse and subsequently converted for the purposes of the DEM, which then feeds data to the individual services offered by the portal. A detailed description of this process, as well as the solution's architecture, are discussed in more detail in Chapter 5. Figure 4.4. presents a general diagram of data flow from domain-specific systems to the data warehouse and to the knowledge base of RAD-on.

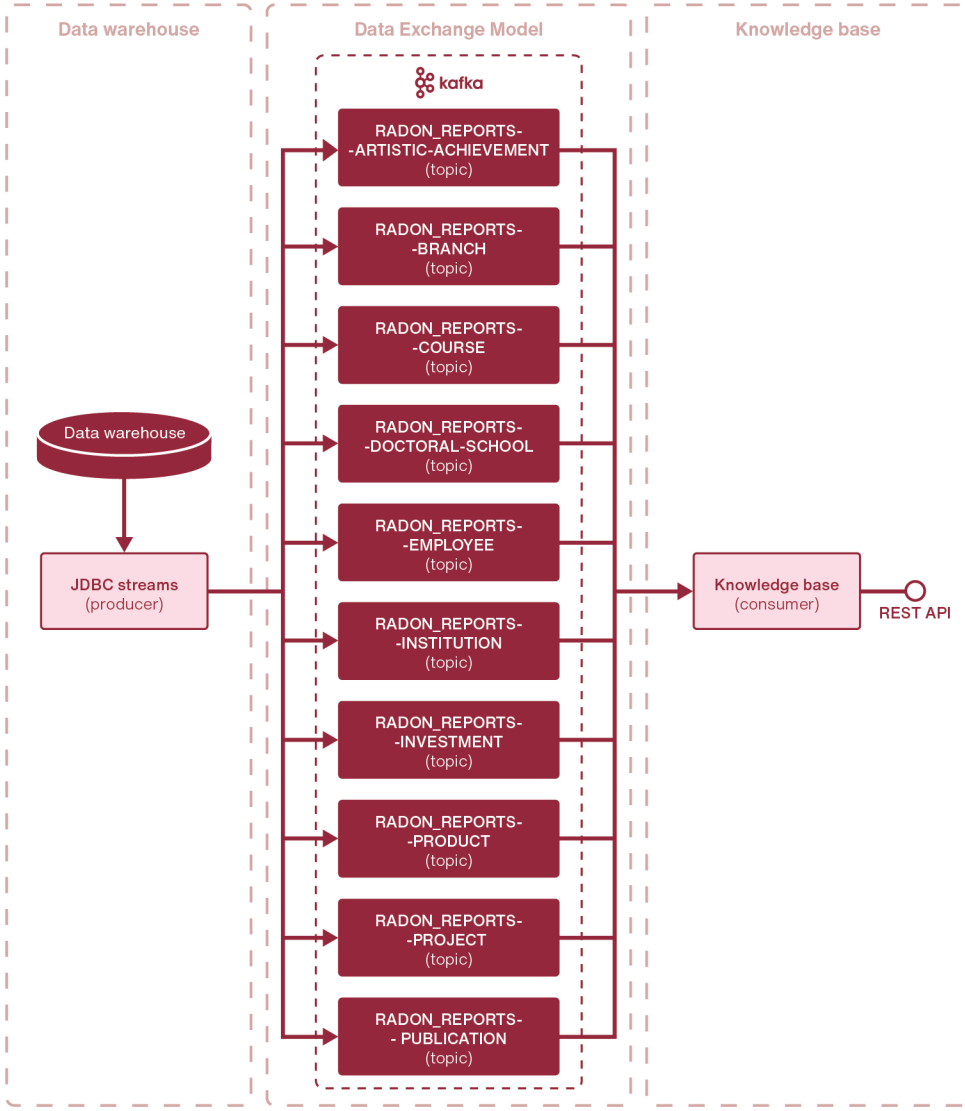


Figure 4.4. A diagram of data flow between the data warehouse and RAD-on's knowledge base, using the DEM.

USING THE DATA EXCHANGE MODEL TO REGISTER INFORMATION ABOUT SCIENTIFIC ACHIEVEMENTS ON PBN

The Polish Scholarly Bibliography[40] (*Polska Bibliografia Naukowa* – PBN) is a portal of the Polish Ministry of Education and Science (MEiN) that collects information on Polish scientists' publications, works published by scientific institutions, and Polish and foreign journals. PBN is a part of the Integrated System of Information on Science and Higher Education (POL-on)[41]. Publications that are gathered on PBN are associated with higher education and scientific institutions (including, but not limited to universities and research centres) and their employees. Data on the institutions and their employees is held in separate modules of POL-on.

To ensure PBN's access to up-to-date and correct data during registration of information on scientists' publications and institutions' achievements, a data-sharing mechanism that utilises the DEM was implemented. One advantage of this solution is that the individual POL-on microservices need not be queried for data. Source data pertaining to employees and institutions is aggregated by the data warehouse. Next, the warehouse converts the aggregated data for the purposes of the DEM using the JDBC Streams solution (for more information, see section 5.3.1.). The data that has been converted and published is then read and used by PBN to associate publications with members of staff and institutions that are registered on POL-on. The data flow from domain-specific systems to the data warehouse and to the PBN system is illustrated in Figure 4.5.

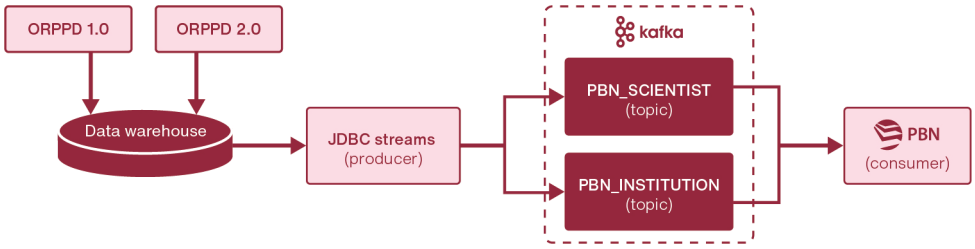


Figure 4.5. Message flow between domain-specific systems and the PBN 2.0 system supported by the DEM.

USING THE DATA EXCHANGE MODEL FOR EXCHANGING ADMIN MESSAGES

The Message Handling System (MHS) is an application that enables the sending of admin messages to systems that are integrated into the RAD-on infrastructure. Admin messages are important notices to users of domain-specific systems. Such messages should be displayed in a visible place, so that each user can read them. The messages often provide information on planned implementations and system downtime caused by maintenance. MHS comes with a graphical user interface that enables users to create

[40] pbn.nauka.gov.pl (accessed 2 March 2023)
[41] polon2.opi.org.pl/siec-polon (accessed 2 March 2023)

and modify admin messages. MHS also has its own dedicated, independent database in which all published administrative messages are stored. MHS was integrated with the DEM, which is responsible for the distribution of information on changes to the system. Two of its key functions are its classification of messages by their importance (warnings, notices, or errors) and its definition of the timeframe for the messages' publication in domain-specific systems. Aside from their text, admin messages also contain message identifiers, identifiers of the systems to which they pertain, and information on whether the given messages remain active. The purpose of the messages that are available in Apache Kafka is to act as notices on the creation or modification of admin messages from the graphical user interface level.

The integration of systems with MHS provides for two scenarios of admin message flow. The first is used when a system that is integrated with MHS has been started and performs without issue. In this scenario, messages created or modified by MHS users are automatically assigned to the appropriate topic in Kafka, and subsequently published. The process of how messages are created, managed, and shared is illustrated in Figure 4.6.

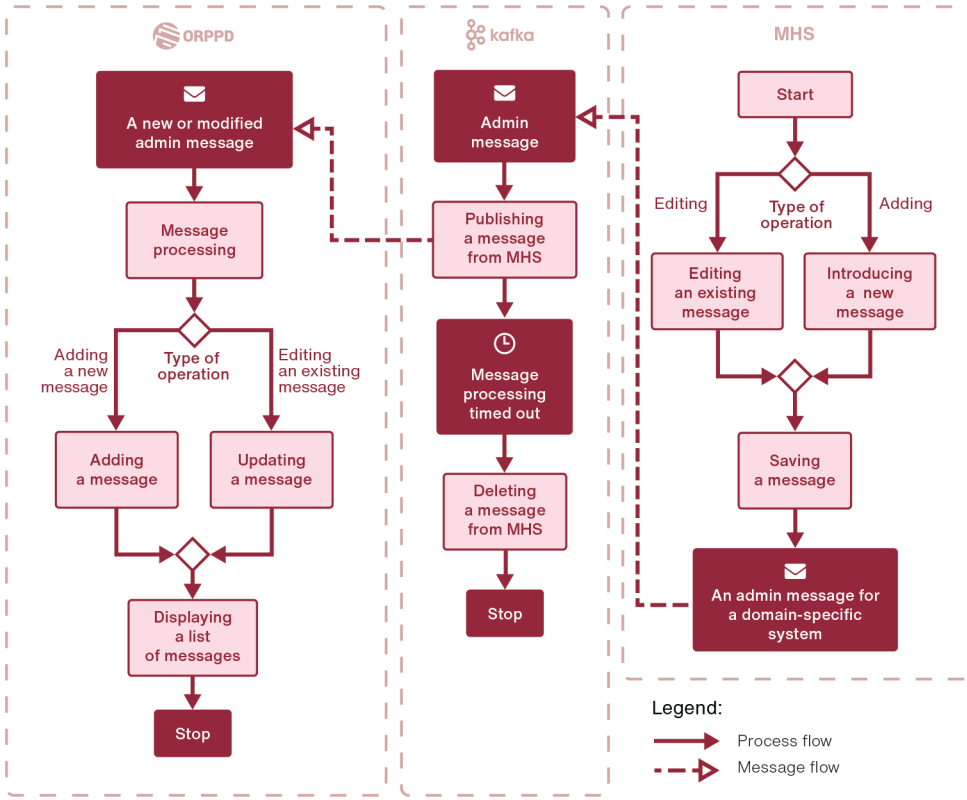


Figure 4.6. The process of creating admin messages using the DEM—the ORPPD system.

Systems that subscribe to the topic in which a new or a modified message was created download the published notice and display it in the designated place. A series of topics that correspond to admin messages for the individual domain-specific systems has been created. The topics that are used for sharing admin messages with individual domain-specific systems are presented in Figure 4.7.

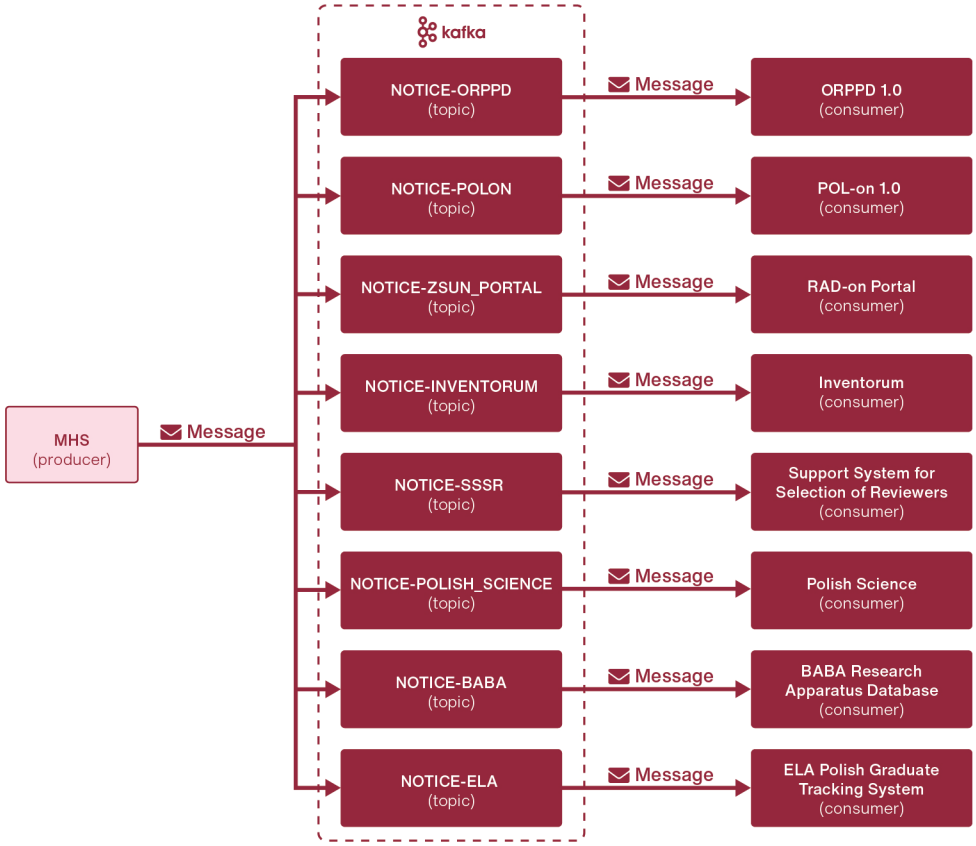


Figure 4.7. Integration of domain-specific systems with MHS using the DEM.

The second scenario of MHS's integration with domain-specific systems handles situations in which a domain-specific system has been shut down and restarted. The process of downloading admin messages after a system restart is presented in Figure 4.8.

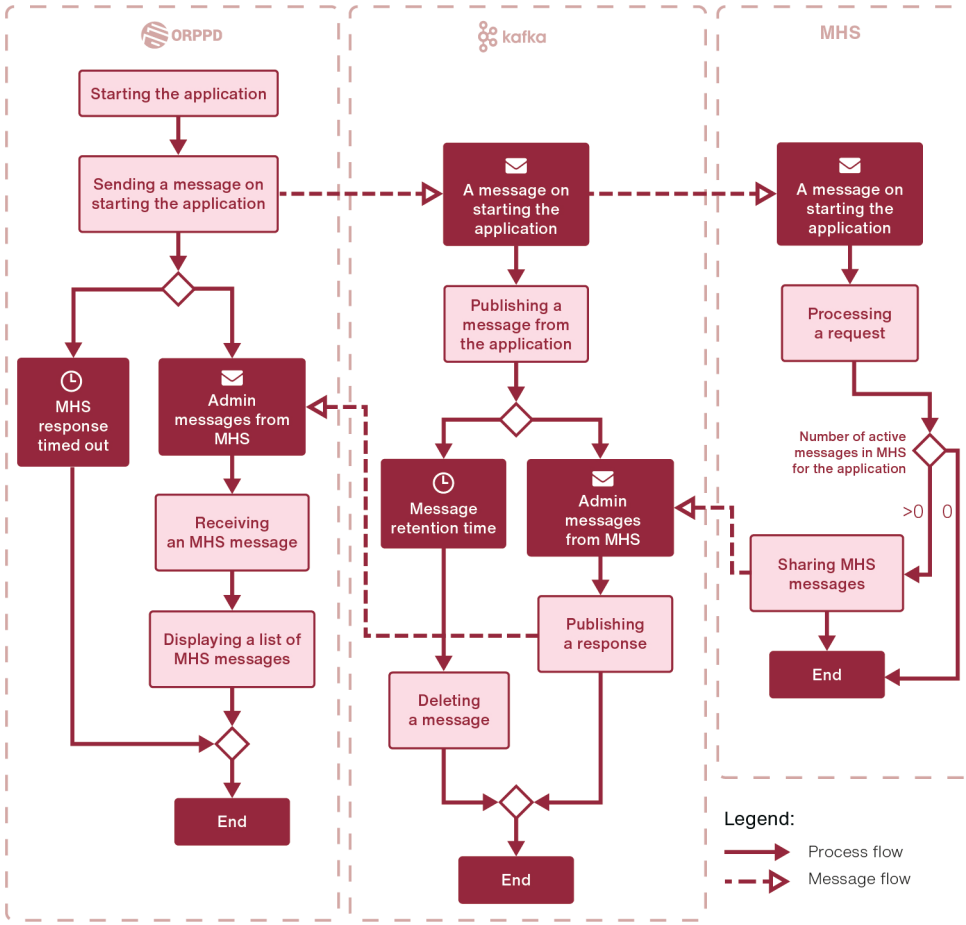


Figure 4.8. The process of downloading admin messages after a domain-specific application's restart using the DEM—the ORPPD system.

Domain-specific systems do not store messages from MHS indefinitely, because the messages are downloaded in real time and stored in cache memory. If an application is closed and restarted, all messages must be downloaded, so that the domain-specific system can restore its status from before the shutdown.

Messages published using Apache Kafka have a defined retention time in the queue system. PAfter the publication time defined in the server's configuration elapses, the messages available in Kafka are removed from the queue system; however, they may remain in the domain-specific application's cache memory. When the application is closed, all messages in its cache are deleted. A restarted application requires a complete list of the messages created in MHS. To restore the status from before the shutdown of a domain-specific system, a mechanism has been implemented that notifies MHS about the application's restart. Once launched, the system sends a message to Kafka on

what domain-specific system has been restarted. MHS reads the message and provides Kafka with the full list of admin messages. In this scenario, all of the systems that have been integrated through Kafka act as both recipients and as senders. The stages of restoring the list of messages in a domain-specific system and the components that participate in the communication are illustrated in Figure 4.9., using ORPPD 1.0 and POL-on 1.0 as examples.

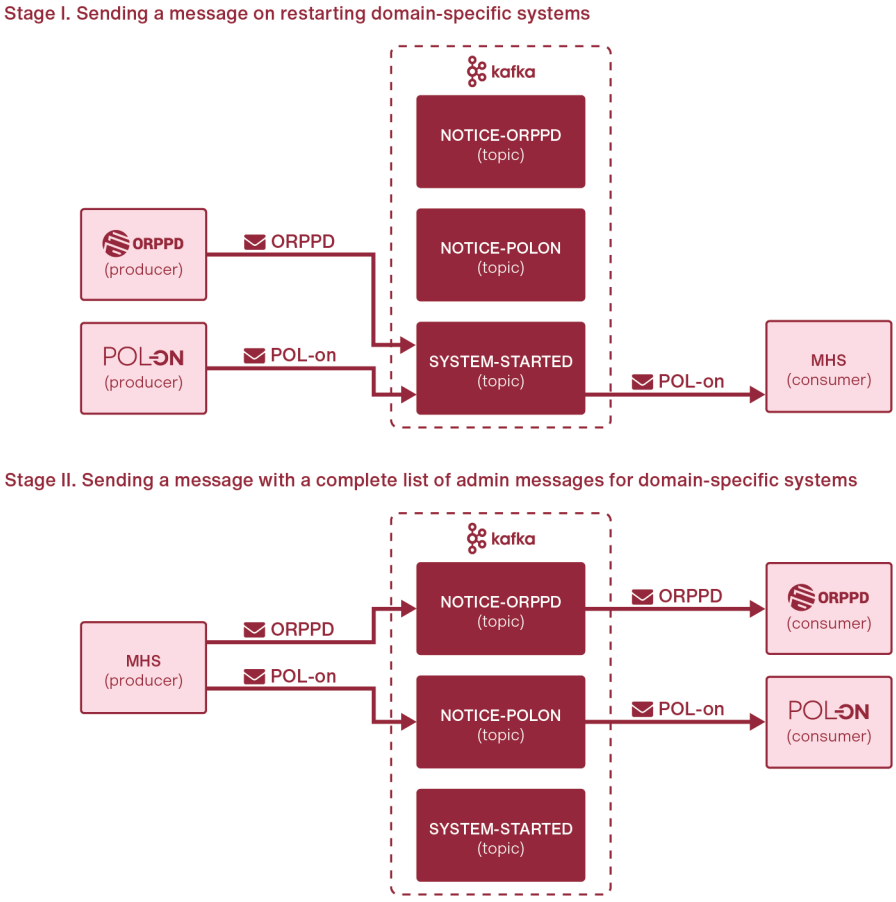


Figure 4.9. Readout of MHS messages involving the DEM after a domain-specific system's restart—ORPPD and POL-on.

MHS is subscribed to the SYSTEM-STARTED topic in Kafka, which enables any system to report that it has been restarted. MHS receives this message and subsequently publishes a complete list of admin messages that pertain to the given topic.

4.2. Data warehouse

The RAD-on IT ecosystem lacked a crucial component that could integrate and organise the data generated by individual systems effectively. Due to the absence of a central data source, reporting had to be done directly by production systems. This resulted in unstandardised processes of exchanging and integrating data between systems, which, in turn, often led to unnecessary workloads for the production systems. Complex analyses that required cross-sectional reviews of data from multiple systems were costly, because the data from the domain-specific systems had to be manually integrated, cleaned, and organised each time. The rising costs of drafting periodical reports, user-requested cross-sectional analyses, and replies to requests for access to public information resulted in the decision to implement a central data warehouse and a business intelligence (BI) system. While developing RAD-on, we also designed and implemented a data warehouse, which became the natural source of complete and credible data presented on RAD-on (the services and data published on the portal are described in Chapter 2.).

4.2.1. Data warehouse architecture

OPI PIB's IT systems utilise various data storage technologies; chief among them are the relational databases that are used frequently to store JSON or XML documents. The data warehouse integrated the following data sources:

- Oracle relational databases
- PostgreSQL relational databases
- MySQL relational databases
- MSSQL relational databases
- MongoDB object databases
- Apache Kafka message broker
- XML documents
- REST API
- static files, such as XLXS, CSV, and TXT.

The data sources are illustrated in Figure 4.10. (the data source layer), which presents a simplified diagram of the architecture of the data warehouse and the information flow between domain-specific systems, the data warehouse, and, ultimately, the RAD-on system. Due to the dominant position of Oracle's technology (which is also used by OPI PIB's largest systems, including POL-on and OSF; see more in section 1.6.), we decided to use the firm's database as the data warehouse's engine.

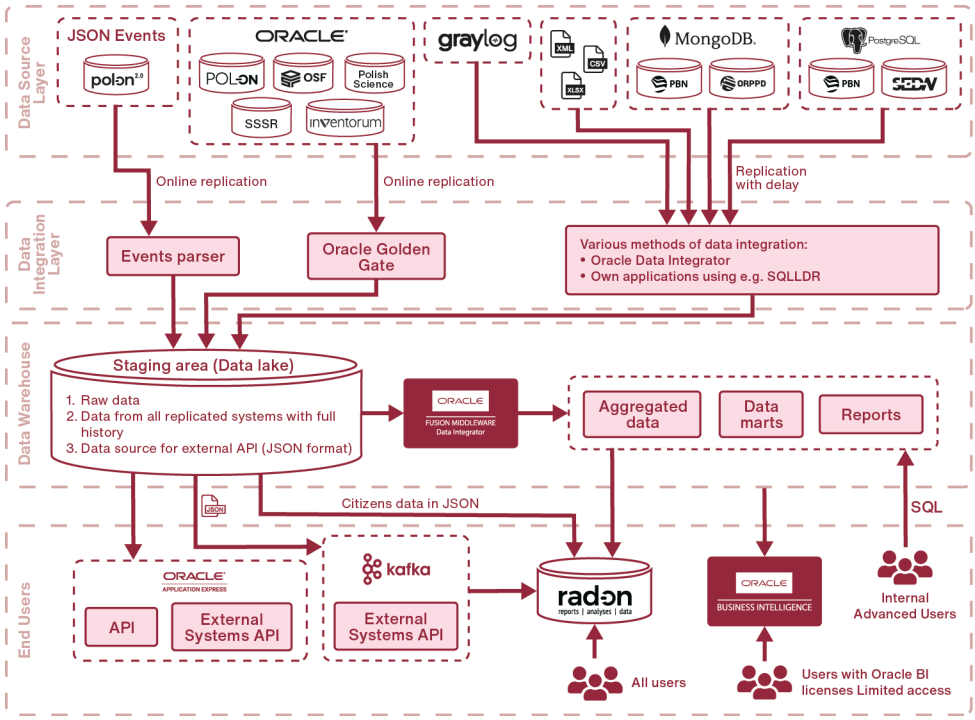


Figure 4.10. The simplified architecture of the data warehouse.

The integration of multiple nonheterogeneous data sources required the preparation of a comprehensive data processing procedure: Extract, Transform, Load (ETL). To achieve this, we used various data integration tools and methods. Data from Oracle databases is replicated to the data warehouse in real time using Oracle Golden Gate[42]. This allows us to transfer changes in data that are made in production systems efficiently and without overloading the systems. The data replication delay is minimal and does not exceed one second. For other data sources, we use Oracle Data Integrator (ODI)[43] or custom solutions in Python[44], that are then executed using ODI. In this case, the data replication delay depends on business requirements. In critical business processes, data may be replicated every few minutes; in other processes, this can be done several times a day. This departs from the recent trend of data warehouses being refreshed once a night. We separated the critical processes that require the most up-to-date data from the standard and cyclical reporting ones.

[42] oracle.com/pl/integration/goldengate (accessed 2 March 2023)
[43] oracle.com/pl/middleware/technologies/data-integrator.html (accessed 2 March 2023)
[44] python.org (accessd 2 March 2023)

The product of the ETL process is data marts (DM): clean, organised, and thematically grouped sets of data. Data structures that form data marts may contain integrated data from multiple systems or areas. One example of such a model is the set of data that pertains to institutions, which integrates data from three systems: POL-on, *Nauka Polska*, and Inventorum (for more information, see section 1.6.). The 'golden record' [15], which is prepared as part of the data's processing, is utilised fully by RAD-on, and serves as a reference for other OPI PIB and external systems, such as those used by higher education institutions, that wish to be machine-integrated with the systems in the domain of the Polish Ministry of Education and Science.

The data warehouse was designed using two methods: 1) the classic star schema, which is utilised as a data source for reports published in the BI system, and 2) a data vault, which is applied to accommodate frequent changes in data structures and to work with documents—including those in JSON format.

For the purposes of this project, we implemented twenty data areas to support the processes performed by the RAD-on, Apache Kafka, and BI systems. In early 2020, a new component was added to the data warehouse architecture: Oracle APEX, a low-code rapid application development (RAD) platform, which simplified the integration of OPI PIB systems and the data warehouse using fast and easy-to-implement REST API services. The benefits of RAD technology expanded the possibilities of using the data warehouse in communication with users. New, user-friendly interfaces have enabled users to send their data to the warehouse, eliminating the need for a team of programmers.

4.2.2. Implementation of the business intelligence tool

To facilitate simplified and intuitive communication of end users with the data warehouse, we implemented Oracle Analytics Server (OAS, formerly Oracle Business Intelligence EE OBIEE). The project's framework postulated that the primary recipients of the data that is made available in the form of interactive dashboards would be experts at the Polish Ministry of Education and Science. The old process required that reports be generated manually on demand by data analysts. That did not allow for a data warehouse or other mechanisms that automate reporting. The advantage of the old process was that each analysis was handled individually. The most significant disadvantage was the cost: each order for a report had to be analysed and handled independently of any other. The long waiting times to receive custom analyses also constituted a significant flaw. Implementation of the BI tool has enabled routine reporting processes to be automated. After analysing the orders for reports, we created dedicated dashboards that provide answers to frequently asked questions and enable users to analyse data by themselves while applying various cross-sections. Below are the key statistics that pertain to the use of the BI tool and the data warehouse:

- **24** thematic dashboards
- **107** unique users
- **10** unique users every day
- over **61 thousand** reports generated in 2022
- over **255 million** lines of text in reports downloaded in 2022
- **eight** public institutions that use the BI tool
- **82** unique ETL processes that power the data warehouses
- **2.2 terabytes** of data stored in the data warehouse
- **191** REST API interfaces shared.

4.2.3. Data management

Having access to large quantities of data does not mean that a user is capable of taking full advantage of the data's potential. OPI PIB uses its systems to process tremendous quantities of diverse data. Due to the nature of the systems, domain-specific knowledge about data is accumulated within the teams that develop the systems, and among the teams and individuals that create analytical reports. The wide distribution of information in the systems makes it difficult to analyse data and develop insights, and mining for cross-sectional data requires many people. In response, in 2021, OPI PIB decided to launch a data governance programme. With consideration for the complexities of data management and the challenges associated with implementing data governance in organisations [2], our primary goals were to raise awareness of the potential of the data that OPI PIB manages and to designate data owners. Within a year, we had managed to draw a comprehensive list of all our datasets, and to implement the matrix organisational structures responsible for the programme's success. We also initiated the process of creating a corporate data model. To date, we have identified fifteen large business areas and fifty-eight smaller ones (see Figure 4.11.) that are managed by twelve data owners. Given that data is the basis of RAD-on's implementation, a crucial element of the project was ensuring that the data was of high quality, and that it was reliable and accessible. Implementation of data management policies has made RAD-on an effective and credible supplier of public information on science and higher education in Poland.

Figure 4.12. illustrates the business data model for the 'Promotion procedure' area.

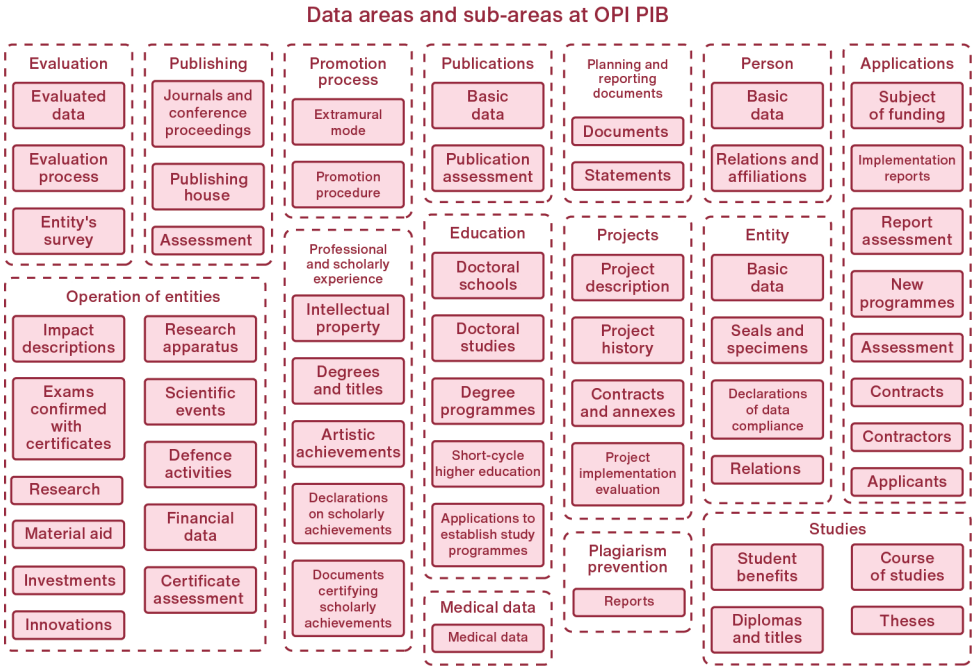


Figure 4.11. Business data areas.

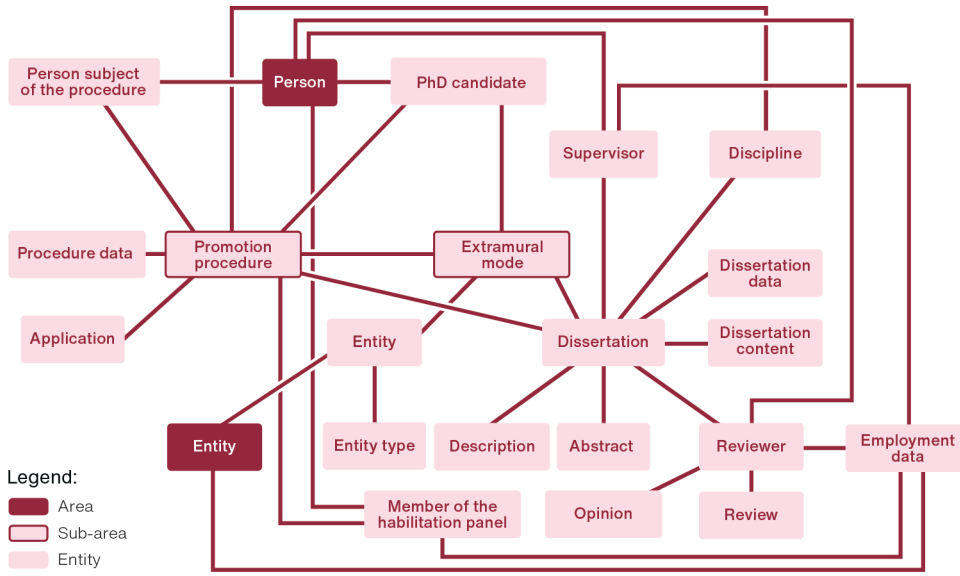


Figure 4.12. The business data model for the 'Promotion procedure' area.

The implementation of the data governance programme at OPI PIB allowed us to understand the importance of correct data management for organisations. Improving awareness and comprehension of an organisation's data can enhance daily operations, bolster security measures, and expedite the delivery of superior products.

4.2.4. The role of the data warehouse in OPI PIB's IT architecture

In the current architecture of IT systems developed by OPI PIB, the data warehouse frequently acts as the data integrator. By definition, all IT systems should be integrated with the warehouse that stores copies of the integrated systems' data. When a system requires credible, converted, and cleaned data, the data warehouse will be the source of it. At OPI PIB, this approach has significantly reduced the time required to implement features in business systems, because software engineers can focus on developing specific features without worrying about the integration of data from multiple sources. For example, the data warehouse was used as an integrator in the process of evaluation of institutions' scientific activity (this is described in detail in section 1.3.). During the evaluation process, the warehouse collected, integrated, cleaned, and converted data from the POL-on, PBN, and Web of Science systems to feed the System for Evaluation of Scientific Achievements (SEDN). That process is presented in Figure 4.13.

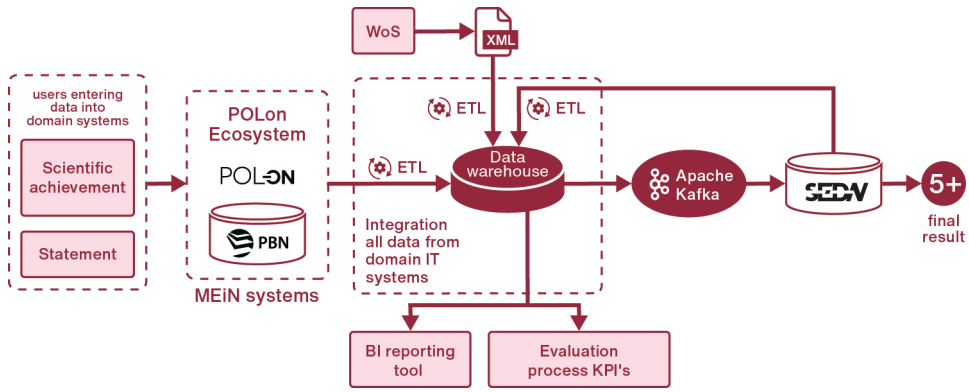


Figure 4.13. The use of the data warehouse in evaluating the quality of scientific activity of Polish science and higher education entities.

CHAPTER 5

APIs

Marcin Białas
Dr Marcin Mirończuk

5.1. Origins of APIs

An application programming interface (API) is a set of rules and conventions that define the manner in which computer programmes may communicate with each other, what information they can exchange and what its structure must be, and how programmers should perform joint tasks [30]. APIs are used in various areas, including information technology, telecommunications, and electronics. Their purpose is to facilitate the development of programmes and enable them to work together. APIs may also utilise components of graphical user interfaces (GUI) [34]. An effective API simplifies software development, reducing it to the joining of blocks according to an agreed convention.

One of the goals of the project was to share standardised source data from repositories of science and higher education through a series of APIs that provide services to end users: humans and machines from the public and private sectors.

The correct data structure, communication, and exchange of information between the individual elements of the RAD-on system, as well as between the users and the system in the form of a web application are ensured by representational state transfer (REST) API. REST API enables the seamless operation of RAD-on services and modules, and communication between them if necessary. REST API is based on the client–server architecture and the namespace in the form of uniform resource identifiers (URIs). REST API uses HTTP for client–server communication and is designed to share resources that can be read, modified, added, or deleted through standard HTTP methods, such as GET, POST, PUT, or DELETE. APIs are also used frequently in internet and mobile applications because they facilitate integration of systems and services.

Currently, RAD-on uses REST API and GUI APIs to share the standard categories of data that constitute the basis of RAD-on's modules and services. Each category corresponds to one of the resources associated with higher education and science (e.g. institutions, publications, or university staff). Most of this data is modified frequently—scientific and higher education institutions change their names, new publications appear, and staff members change their jobs. Furthermore, changes to the data structure or the data itself are necessary whenever new requirements for shared information or

new business models in the source systems emerge. It is also possible that new data categories appear. All of these scenarios must be considered in the process of data integration and sharing, which means that the system must satisfy a multitude of requirements. To manage this dynamically-changing data, we have developed and implemented an innovative strategy, architecture, and system. This has enabled us to respond promptly and efficiently to the evolving needs and requirements of our stakeholders.

The API described in this chapter is linked closely to the services of machine-sharing of resources of higher education and science, as well as metadata. Opinions regarding the need for the services were collected during the analysis described in section 1.4.

The service of machine-sharing of resources in higher education and science has been adapted to the needs of multiple groups of institutional recipients, including higher education institutions, scientific institutions, research-funding agencies (NCN, NCBR, and NAWA), and the Polish Accreditation Committee. From the perspective of administrative bodies of higher education institutions and scientific institutions, the service meets needs related to automatic and machine downloads of different types of data compilation that pertain to science and higher education.

In the case of the metadata sharing service, whose purpose is to improve the visibility and availability of Polish scientific resources, we planned to make the solution useful to a broad group of recipients, including individual scientists and researchers, representatives of the government agencies responsible for Polish science policy, research and development institutions, journalists, and citizens interested in information on science and higher education. The needs of these stakeholders have been addressed by enabling metadata pertaining to the shared data to be sent to the Central Repository of Public Information (*Centralne Repozytorium Informacji Publicznej* – CRIP) and to be updated in the case of any changes. Due to the integration with CRIP, we have improved the visibility of resources on Polish science.

The next three sections of this chapter describe the architecture of the proposed solution, including the key elements of the system and the data flow. The description of the system's architecture had to be separated into multiple sections due to the degree of detail in which they had to be discussed. Section 5.2.) presents the general framework and the key components of RAD-on's architecture. Section 5.3. discusses the technical aspects of the strategy. Section 5.4. focuses the technologies that are utilised.

5.2. An overview of the system architecture

The system architecture comprises interfaces that are responsible for direct communication between internal and external systems, and between the system's users. The general architecture is presented in Figure 5.1.

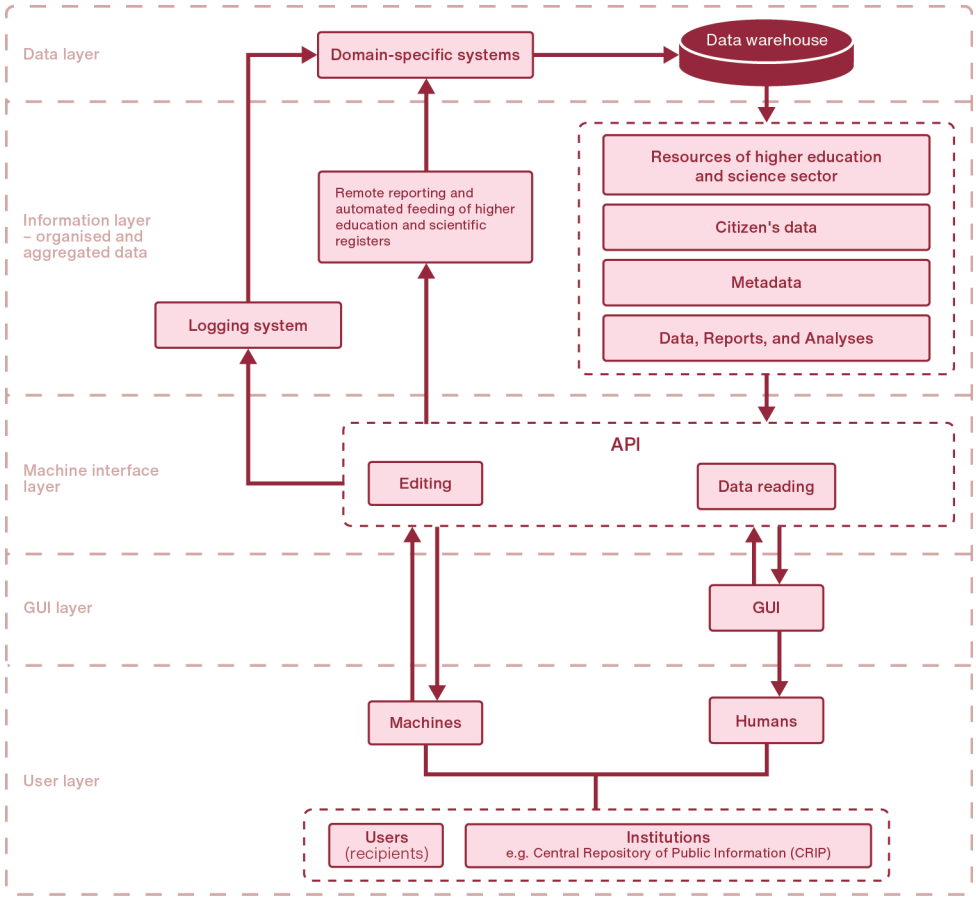


Figure 5.1. The general architecture of the system.

The system in Figure 5.1. comprises the following layers:

1. **The data layer** is responsible for the integration of information from domain-specific systems and for the delivery of standardised source data to services via the data exchange model (DEM).
2. **The information layer** stores the data that has been organised and aggregated into reports and analyses.
3. **The machine interface layer** is used to input and edit data (also known as the layer of services that feed the information registries).
4. **The machine interfaces for data reading layer** (services for access to data, analyses, and reports) provides a standard API for downloading uniform information from the information layer.
5. **The graphical user interface (GUI) layer** (a public portal) is presented the form of an internet portal that provides users with access to information on higher education and science.

6. **The user layer** comprises people (institutions and individuals), and software (machines) developed by users, or at the request of businesses and institutions.

The system architecture is associated with the DEM (which is discussed in detail in Chapter 4., an abstract element that combines all of the layers described above. Its purpose is to integrate the systems in the domain of science and higher education, as well as enabling cooperation with other models, such as national-level information models. The DEM combines internal and external systems, and provides data from source systems via machine services and the portal.

The components of the **data layer** are:

- domain-specific (source) information systems: various information systems and their data registries maintained by OPI PIB and external entities
- the data warehouse, which converts source data into useful information, such as analyses, statistics, or static and dynamic reports by loading, transforming, and exploring such data and by applying business analysis tools. This information is then made available in the form of reports through the portal's knowledge base and through business intelligence (BI) tools for advanced users.

The components of the **Information layer** are:

- the knowledge base (reports, analyses, data): a semantic index for searching. A full-text and semantic search engine is designed to search through the resources of all domain-specific systems and to link data semantically
- resources of higher education and science: unified registries of data and information located in domain-specific systems that are responsible for business processes related to science and higher education
- citizen's data: a registry that contains unified information from various domain-specific systems, including personal data and associated information
- metadata: a registry that contains information on services provided
- event logs: a registry that stores all data saving and reading events through the API provided.

The **machine interface layer**, which is used to input and edit data, comprises a service used to perform tasks associated with remote reporting and automated feeding of higher education and scientific registries from external systems that have access to machine services and that enable data sharing and editing.

The components of the **machine interfaces for data reading layer** are:

- a component that is responsible for sharing the knowledge base's resources through an API
- a service that shares the resources of the higher education system: a set of web services that enable external systems to download data These services are integrated with the source data systems through the DEM, which enables external systems to be separated from data sources and for a standard communication model to be used

- a service that enables access to citizens' data on the portal. The portal displays all data that pertains to specific users because the DEM queries the resources of all source systems and provides a complete set of data
- a service that shares metadata: the DEM contains information regarding the data and services to which it enables access. Users may learn what services and data are available through the portal. External systems may gain the same information through machine services.

The components of the **GUI layer** are:

- reports, analyses, data, and a search engine: the elements available on the internet portal that provide access to information from the knowledge base, citizen's data, metadata, and higher education resources. The portal connects to the information layer using standard APIs. The search engine enables data to be searched using natural language in all domain-specific systems connected semantically in the knowledge base
- citizen's data: the component that allows users to check their data collected in domain-specific systems, and to edit such data by, for example, reporting the need for the data to be corrected.

The components of the **user layer** are:

- external systems: any IT systems that may exchange data with the system through data access services. Such systems may download data from and forward it to source systems
- cooperating systems: external systems that work together with the system by exchanging data and services. The key cooperating systems include CRIP and the Polish Electronic Identification Hub (*Krajowy Węzeł Identyfikacji Elektronicznej* – KWIE)
- users: individuals who can be located anywhere and who can use the system through an internet browser on any device.

5.3. A detailed description of the system architecture

This section describes the elements of the system that participate in data sharing. Before data can reach the end user, it must pass through multiple important system components. A general list of these components can be found below. The outline of the data flow and the system architecture itself is presented in Figure 5.2.

- **data warehouse:** Oracle database
- **JDBC streams:** the JAVA Spring application, which handles data streaming between the warehouse and Apache Kafka
- **Apache Kafka [21]:** a platform for processing data streams
- **Index Manager (IM) and API:** the JAVA Spring application
- **Elasticsearch (ES) [24]:** a full-text search engine
- **metadata repository:** a GIT repository used to store metadata about services.

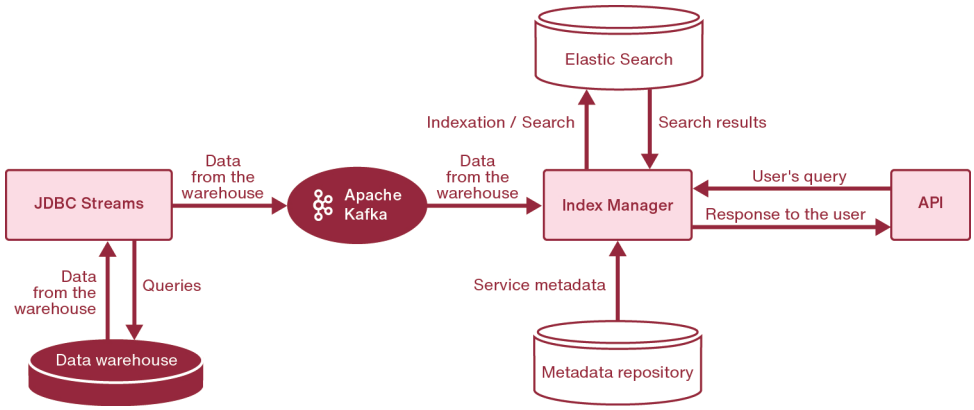


Figure 5.2. An outline of the system's architecture, including data flow.

To present the system's operation intuitively, let us trace the path that data located in the warehouse must traverse to reach the end user. The first component that participates in data transportation is the **JDBC Streams**, application, which acts as a bridge between **Apache Kafka** and the data warehouse. Its basic function comprises routine, pre-planned querying of the data warehouse, and transportation of the data to Apache Kafka. Additionally, it manages message queues in Apache Kafka. From this point onwards, message queues will be referred to as 'topics'. When a new data packet reaches Apache Kafka, it is read by the **Index Manager (IM)**, the central system component, which occupies multiple key roles—one of which is the downloading of data from Apache Kafka. When the data is downloaded, the IM, pursuant to preset instructions, converts the data into documents that are consistent with the **Elasticsearch (ES)** specification. The instructions that define the structure of ES documents depend on the type of indexed data and, specifically, on the types of query that may be of greatest interest to RAD-on's users. Instructions for different types of data are stored in the metadata repository, from which they are downloaded by the IM. Processed documents are then indexed in the ES cluster. Another task undertaken by the IM involves handling incoming client queries from the **API** layer. User queries are delivered to the IM, where they are converted into a format that is consistent with the ES specification, before the IM initiates the search. Next, the search results are forwarded to clients. The principles of API use are made available to clients in the form of interactive **Swagger**[45] documentation. The complete documentation, alongside the indexing instructions, is stored in the **metadata repository**, which communicates with the IM. The individual elements of the system are discussed in the subsections below.

[45] swagger.io (accessed 2 March 2023)

5.3.1. Streaming management

JDBC Streams are system components that are used to send data from relational databases to Apache Kafka. The application's basic functionality enables it to define streams. A stream is defined as a cyclical database query, the response to which is sent to the Apache Kafka topic whose name matches that of the stream. Each stream handles one entity version, and the stream's name combines those two pieces of information. For example, in the case of an entity called 'institutions in version 1.3', both the stream and the topic will bear the name INSTITUTIONS-1.3. Using this naming convention allows various elements of the system to be linked logically. This is discussed in section 5.3.2. of this chapter. The data flow is illustrated in Figure 5.3. The figure presents three different tables in the warehouse, which contain data for the entities 'institutions', 'publications', and 'employees'.

Figure 5.3. Data flow between the data warehouse and Apache Kafka.

The data transferred is not stored by the application, because the application's chief function is data transport. Depending on how a query is worded, the data stream may download all contents of the table or only the new records. The stream also defines how frequently the application should query the database. The most suitable strategy depends on the dynamics of changes in the data and is optimised separately for each entity. In the case of data that is modified frequently, the query interval may be as short as a few seconds or minutes; data that is rarely modified may be refreshed once a day. When defining the frequency, the database's performance must also be considered: too many queries executed simultaneously may overload the system or cause malfunctions. If a different data download pattern is defined for each stream, the system will continue to operate stably and provide users with up-to-date data.

The application's current status can be monitored from the administration panel. Each stream is described by two statuses: the functional status and the status pertaining to the last action performed. The functional status specifies which action the stream performs and takes the following values:

- *WAITING:* the stream is waiting for its next initialisation
- *PROCESSING:* the stream is downloading data from the database and transmitting it to Apache Kafka

- *STOPPED:* the stream has been interrupted due to an excessive number of errors.

The status pertaining to the last action performed takes the following values:

- *STABLE:* the last query has been completed correctly
- *UNSTABLE:* the last query has been completed, but the records processed contain errors
- *ERROR:* the query has been interrupted due to a critical error or too many erroneous records
- *RESET:* the stream's status has been reset by the administrator.

Current stream statuses allow data transfer to be monitored and errors to be corrected as they occur.

Data is transferred by the application in the JSON format and is stored as separate records within an Apache Kafka topic. While transferring data, JDBC Streams can also verify the compliance of the records transmitted with a pre-defined JSON schema for the given entity, which enables erroneous data to be excluded from further processing. Depending on the needs, the number of allowed errors can be defined for each stream, which, when exceeded, will halt further queries and switch the stream into the *ERROR* status.

5.3.2. Index manager

Management of different versions of data categories, hereinafter referred to as entities, is ensured by the Index Manager (IM). Entity versioning is necessary and results directly from ongoing data modifications. For each new entity version, a new API documentation and new indexing rules are created, if required by the new API filters. Each time the data structure or the query schema changes, a new version of the metadata is introduced, which contains the current Swagger documentation and, if necessary, new indexing instructions. Entity metadata is stored in a repository with a GIT version control system. When changes mandate that the API query schema be modified, minor modifications are also required in the application code that handles client queries. The IM provides a GUI for administrators that allows them to add or remove an entity, or to change its version to another one that is available in the repository.

Let us examine the process of adding a new entity by entering a new version, 1.3, of an institution. Each entity version corresponds to a topic in Apache Kafka, which has been fed with data. In this example, the topic will be called INSTITUTIONS-1.3. For the entity in question to be available in the IM interface, its metadata must be available in the repository. The process of adding an entity for this example is presented in Figure 5.4.

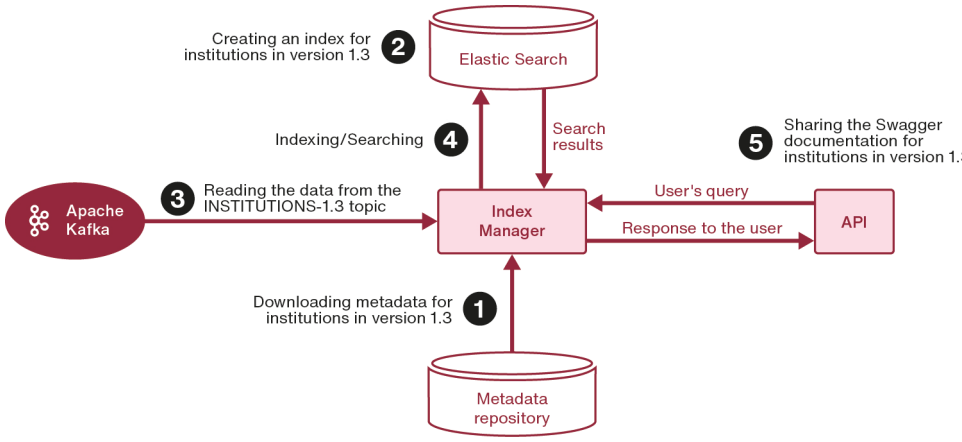


Figure 5.4. The creation of a new entity.

Operations performed by the IM to share the entity's new version are marked in red and numbered. We will now discuss them in order. **Operation one** involves metadata being downloaded from the repository when the IM's user interface is initiated. Information about all entities available, including the one in the example above, is downloaded. The metadata contains indexing instructions, as well as the most up-to-date Swagger documentation. In **operation two**, the IM creates a new ES index, which will store documents that contain the institution's data in version 1.3. When this index is created, the IM subscribes to topic INSTITUTIONS-1.3 **(operation three)** and begins a streaming download of the data. Based on the instructions contained in the metadata, the records downloaded from Apache Kafka are indexed in the ES cluster **(operation four)**. The process ends with **operation five**, when the IM begins to share the new documentation with clients. All of the operations are performed rapidly, and within a few seconds of the new version being made available, all clients may download the new data.

After subscribing to topics in Apache Kafka, the IM downloads records on an ongoing basis, whenever the application is running. This enables effective modification of the indexed documents and keeps the data up to date. Each record in Apache Kafka bears a unique universal identifier (UUID), which is fixed for a given business object. Multiple records may bear the same UUID within a single topic. One example is that of an institution whose name has changed multiple times and, as a result, a new record has been added after each name change. During the indexing procedure, the UUID is used as a document identifier in the ES index. If a document with the given identifier is not present in the index, a new document is added; otherwise, the existing document is replaced with a new one, which facilitates the maintenance of the current data in the index without user intervention. The latest record with the given UUID is always indexed. When a document is deleted, Apache Kafka receives a record with an additional field, which states that an element must be deleted from the index. When such a field appears, the IM searches each record and the appropriate document in the RAD-on system is deleted from the index.

Data flow management through the IM is supported by Topic Reader (TR) objects, which link entity sets together and are responsible for ensuring communication between all system components. A TR may take one of the following statuses:

- *CREATED:* a TR has been created, but has not yet started reading data from Kafka
- *STOPPED:* a TR's data reading from Kafka has been stopped
- *WORKING:* a TR is actively reading data from Kafka
- *SUSPENDED:* following a critical error, data reading has been suspended, but the TR will soon make another attempt to read the data.

When creating a new TR, the user must select the entities to be managed by the TR. For each entity selected, a separate index is created in the ES cluster. The *START* action is performed to initiate data download from Apache Kafka, which causes the TR to subscribe to the topics with the same entity version and download the data for each of them parallelly. This data is then indexed in the ES cluster, and the TR switches to the *WORKING* status. If the record collection downloaded from Kafka is indexed in the ES cluster without errors, the TR will download another set of unread records and commence indexing them. If an error occurs during any of the data processing stages, the TR will switch to the *SUSPENDED* status; after three minutes, it will make another attempt to download the same data. This mechanism prevents data leaks, e.g. in the case of a temporary ES cluster failure or another error that could prevent correct indexation of the records downloaded during any processing stage. Each error that has forced the TR to restart is saved in the database. Once per hour, the IM checks the table that contains information about errors. If new records are detected, the IM sends an email about the malfunction. Thanks to this, the administrator may begin repairs to remedy the problem in a relatively short period. The *STOP* action is used to abort the data flow. If it is executed, the TR switches to the *STOPPED* status and the process of indexing the RAD-on website is interrupted. When the TR is *STOPPED*, it is possible to modify the entity sets managed by it by, for example, adding new entities or removing existing ones. If it becomes necessary to modify entity versions, the *DELETE* action is used to remove the given data category, including its index in the ES cluster, and then it is re-added (the *ADD* action) in the new version required. Maintaining separate entity indices enables quick version changes. Maintaining a single index shared by all entities would require the indexing of all data from the beginning, which would be time-consuming and force unnecessary load on the ES cluster. The solution adopted enables versions of a single entity to be modified without having to rebuild the other entities that are managed by the TR.

If the entities that comprise a TR are also available in the full-text search engine of the RAD-on website, an additional index with a simplified structure is created for that specific purpose. Documents in this index contain only one text field that enables searching. This field may combine various pieces of text information selected individually for each entity, in accordance with users' needs. This special index is created parallelly to all of the indices listed above, and it contains all data categories that are made available to the RAD-on website's search engine. Creating a full-text index enables search results to be optimised across all categories, based on the search term used. When an entity is

removed from the TR, the documents that belong to that entity are also removed from the full-text index and disappear from search results.

5.3.3. Handling queries from APIs

This subsection discusses communication between the APIs and the ES engine. The system provides three different APIs based on the ES search engine: two of them are used by the civic portal, while the third is used for direct machine downloads. The flow of queries is controlled by the IM through an active TR, as presented in Figure 5.5.

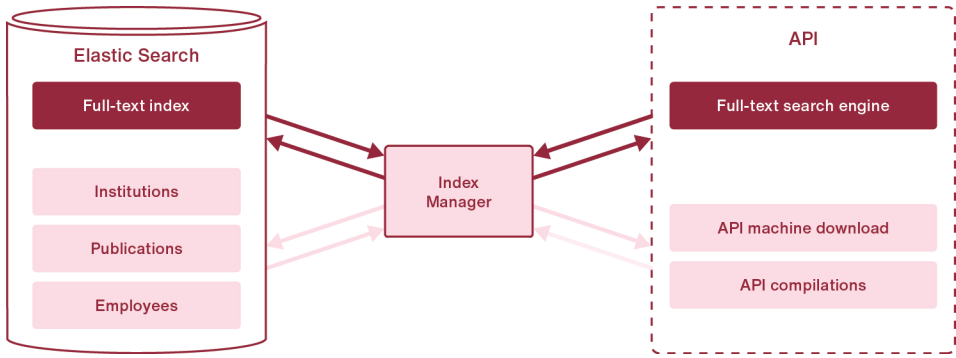


Figure 5.5. Query flow between the APIs and the Elasticsearch engine.

In the example above, we see the ES cluster, which holds three separate indices for each entity handled by the TR ('institutions', 'publications', and 'employees'), as well as one full-text index that combines all of the records. Queries from the full-text search engine are routed to the full-text index, which returns the best matches for the search phrase from all categories. Using only one full-text index enables the search results to be optimised, pursuant to the algorithm implemented in the ES. An alternative solution would require the downloading of the results from all indices separately and the implementation of an additional algorithm to optimise the results. However, such a solution would significantly complicate query handling and increase response times.

Separate indices are used by the users who perform machine downloads of data and by the data breakdowns that are displayed on the citizen's portal. Each breakdown presents data from one entity and is fed with data from one of the separate indices. The relatively small sizes of the indices shorten the querying time, which improves system response times. Using the same index for both machine processing and data breakdowns available on portals ensures data consistency.

In addition to the data included in the breakdowns, RAD-on provides integrated data that is available only through APIs. For this data, a separate TR is created, which lacks the full-text index that links entities. Consequently, the application always has two active

TRs—the first handling data queries for breakdowns, and the other sharing integrated data.

5.3.4. Event logging system

The event logging system is used to collect all API requests and responses, which allows traffic and events on the API services to be recorded. This logging system is an important component of the API architecture because it enables the collection of quantitative information on how the API services are used and their load.

The event logging architecture comprises the following key elements:

- **artefact** provides a standardised interface that describes what attributes and their values are recorded. It defines the template of the API log and what data is found therein. Currently, logging includes information such as the client's IP and geotargeting data, e.g. the city from which the API was queried. The recorded information also includes the duration of query processing, the name of the service and the method called, as well as responses from API services, including possible exceptions
- **API services** implement the interface that is provided by the logging artefact. They complete the specified data template with appropriate data, which means that they assign values to the attributes provided by the artefact
- **Apache Kafka** transports logs from the distributed environment of various API services to designated recipients
- **Graylog**[46]**:** is the chief recipient of logs from Apache Kafka, which stores them in a full-text index to facilitate quick preliminary analyses of information that comes from domain-specific API systems, and the creation of draft analyses and breakdowns concerning service load or errors;
- **the data warehouse** downloads data from Graylog and, depending on business needs and requirements, analyses and aggregates the data according to predefined timeframes for the individual services;
- **the backend RAD-on GUI service** is responsible for downloading processed and standardised information from the data warehouse and displaying it in the GUI.

[46] graylog.org (accessed 2 March 2023)

5.3.5. Data presentation system

The data presentation system comprises a catalogue of available API services, including technical documentation and service manuals.

Documentation is created with the aid of the **Swagger** library, which facilitates the design and documentation of APIs. RAD-on contains a catalogue of available APIs, which is divided into two primary sub-catalogues that contain data sharing and editing services[47]. Currently[48] thirty-four services are available, which allow users to download data of interest through publicly available APIs. In the case of the data editing catalogue, services enable access to data from domain-specific systems, such as POL on and PBN, through which authorised users may make changes in data registries.

RAD-on system modules, including reports[49], analyses[50] and data[51], are also based on the services and data that they acquire. These modules are discussed in detail in Chapter 2. of this monograph.

RAD-on also provides usage statistics, which are updated daily, for selected APIs[52]. These statistics include, but are not limited to the quantity of data shared by the system, the number of downloads of individual services, and the number of downloads of documents that contain information from the public sector.

It is worth noting that the information on the data published in RAD-on can be found in CRIP at dane.gov.pl, which is discussed in Chapter 1. of this monograph.

5.4. Technologies

The system described above was created using ten primary technologies: Java Enterprise Edition, JDK8, Spring, Hibernate, hurtownia Oracle, ES, Apache Kafka, Tomcat application server, Graylog, and Swagger. Changes in the use of and access to new technologies are monitored on an ongoing basis. Updates are deployed whenever both technical (software development tools) and business (new use cases, new features) needs arise.

API components, and REST API services in particular, are scalable due to their containerisation and use of the Kubernetes environment. Kubernetes (also known as K8s) is an open source system used in cloud computing management, which enables automation of the deployment, scaling, and optimisation of applications that operate in

[47] radon.nauka.gov.pl/api/katalog-udostepniania-danych (accessed 2 March 2023)
[48] As of 21 December 2022
[49] radon.nauka.gov.pl/raporty (accessed 2 March 2023)
[50] radon.nauka.gov.pl/analizy (accessed 2 March 2023)
[51] radon.nauka.gov.pl/dane (accessed 2 March 2023)
[52] radon.nauka.gov.pl/o-systemie/statystyki (accessed 2 March 2023)

environments that contain multiple virtual or physical machines[53]. It is one of the most popular tools for managing applications in cloud computing. Kubernetes is used by large technology companies, including Nokia, Spotify, and Yahoo[54]. It enables software engineers to focus on application development instead of infrastructure management, which streamlines application and service deployment, as well as enabling flexible reactions in changing conditions.

[53] kubernetes.io (accessed 2 March 2023)
[54] kubernetes.io/case-studies (accessed 2 March 2023)

CONCLUSION

To the best of our knowledge, in terms of the data it collects and shares, RAD-on is currently the largest national IT system in science and higher education in Europe. The system can be considered an analytical platform that presents official statistics, adheres to the concepts of open government data and FAIR data, and contributes to the establishment of data-driven political and managerial processes. RAD-on supplements the IT ecosystem in science and higher education in Poland, which has been developed by OPI PIB for over ten years. RAD-on can inspire the creation of analogous systems in other regions and countries.

The successful attainment of the individual objectives of the RAD-on project was facilitated by the gradual implementation of management processes and the establishment of an extensive IT infrastructure.

Poland's data on science and higher education, sourced from independent databases, has been consolidated through the implementation of a data exchange model.

Open access to the most up-to-date and reliable data on research and development, science, and higher education has been ensured by a citizen portal. The portal offers the general public access to comprehensive data compilations that can be conveniently filtered, downloaded in CSV and XLSX formats, and searched using a full-text search engine for swift information retrieval. Professionals in the science sector, including political decision-makers, can browse a plethora of ready-to-use reports complete with tables and charts. Journalists and researchers are granted access to comprehensive analyses, which enable them to conduct in-depth interpretations of data. Portal users have the convenience of verifying their personal data from multiple systems in one centralised location, and reporting corrections when necessary. Utilising the login.gov.pl authentication service for identity verification ensures the utmost security for service recipients.

Although the portal is undoubtedly the most significant outcome of the project, it is by no means the only one. Bureaucracy has been greatly reduced, which is best exemplified by users being relieved from their obligation to enter data into databases manually. This has been achieved through the implementation of an API, the enhanced capabilities of machine processing, and the reuse of existing data in innovative applications and services.

RAD-on is a remarkable solution that serves as an alternative to traditional business intelligence applications. It facilitates decision-making processes by providing a suite of IT tools that enable the analysis and interpretation of shared data. These tools visu-

alise and download data that has already been processed and compiled by experts, and contains relevant commentaries.

The system evolves continually, which results in steady increases in data volume and ever-more useful services. Planned works include the creation of scientometric reports and analyses, collaboration with entities that finance science in the field of sharing data on research grants in Poland, promotion of the API's use by universities' internal systems and researchers, and further development of analytical tools for data visualisation.

The most considerable challenge we must face is the efficient adaptation of our services to the ever-evolving legal landscape. During the development of the project, the science and higher education system underwent significant transformations. Further changes regarding data collection and data processing are expected to be introduced.

Other challenges include some that pertain to the UX layer of the analytical platform. Based on the user satisfaction survey, we estimate that one in five users has problems finding the data they seek. In many instances, this is due to the inability to share some of the data. In other cases, the problem lies with the time and effort consuming opening of previously processed data. The highest-rated aspects of the system are its aesthetic layer (64% positive and 29% neutral opinions), the presentation of data, and the usefulness of the services. The lowest-rated is the ease of finding information (50% positive, 28% neutral, and 22% negative opinions) Users highlight the unavailability of certain features in the mobile version of the system, but are satisfied with the growing selection of reports.

RAD-on is designed to benefit all of its users, regardless of their digital literacy or familiarity with the higher education and research landscape. We are confronted with the challenge of striking a balance between providing accurate data descriptions and ensuring that private individuals and journalists can comprehend them. Despite consultations with UX specialists, we continue to lack guidelines on how to design public sector data dashboards. We must remain aware that users become accustomed to interface layouts and do not appreciate frequent changes to functionalities. We believe that this issue merits further exploration and investigation.

We are confident that sharing our experience in designing the RAD-on system architecture, and discussing the key benefits and challenges associated with the development of such a large-scale system, will be helpful to all who are interested in designing and using analytical platforms as decision-making support tools in the science and higher education sector.

ILLUSTRATIONS

REFERENCES

[1] A. Abduldaem and A. Gravell. Principles for the design and development of dashboards: Literature review. *Proceedings of INTCESS 2019 6th International Conference on Education and Social Science*, pages 1307–1316, 2019. ocerints.org/intcess19_e-publication/papers-412.pdf (accessed 2 March 2023).

[2] I. Alhassan, D. Sammon, and M. Daly. Critical success factors for data governance: A theory building approach. *Information Systems Management*, 36(2):98–110, 2019. DOI: 10.1080/10580530.2019.1589670.

[3] A. Andhavarapu. *Learning Elasticsearch*. Packt Publishing, 2017.

[4] B. Ballou, D. L. Heitger, and L. Donnell. Creating effective dashboards: How companies can improve executive decision making and board oversight. *Strategic Finance*, 91(9):27–33, 2010. go.gale.com/ps/i.do?id=GALE%7CA221186517&issn=1524833X&p=AONE&it=r&v=2.1&linkaccess=abs (accessed 2 March 2023).

[5] A. Białecki, R. Muir, and G. Ingersoll. Apache lucene 4. In *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, pages 17–24, 2012.

[6] A. Bochenek editor. *Systemy informatyczne wspierające naukę i szkolnictwo wyższe. OSF: Obsługa Strumieni Finansowania*. Ośrodek Przetwarzania Informacji – Państwowy Instytut Badawczy, 2021. opi.org.pl/wp-content/uploads/2021/12/OSF-ebook.pdf (accessed 2 March 2023).

[7] A. Brown, J. Fishenden, M. Thompson, and W. Venters. Appraising the impact and role of platform models and Government as a Platform (GaaP) in UK government public service reform: Towards a Platform Assessment Framework (PAF). *Government Information Quarterly*, 34(2):167–182, 2017. DOI: 10.1016/j.giq.2017.03.003.

[8] W. Buchanan, A. Gobeo, and C. Fowler. *GDPR and Cyber Security for Business Information Systems*. River Publishers, 09 2022. DOI: 10.1201/9781003338253.

[9] P. Budroni, J. Claude-Burgelman, and M. Schouppe. Architectures of knowledge: The European open science cloud. *ABI Technik*, 39(2):130–141, July 2019. DOI: 10.1515/abitech-2019-2006.

[10] S. Burke, R. MacIntyre, and G. Stone. Library data labs: Using an agile approach to develop library analytics in UK higher education. *Information and Learning Science*, 119(1/2):5–15, 2018. DOI: 10.1108/ILS-05-2017-0035.

[11] E. Cebrián and J. Domenech. Is Google Trends a quality data source? *Applied Economics Letters*, 30(6):811–815, 2023. DOI: 10.1080/13504851.2021.2023088.

[12] I. Chengalur-Smith, D. Ballou, and H. Pazer. The impact of data quality information on decision making: an exploratory analysis. *IEEE Transactions on Knowledge and Data Engineering*, 11(6):853–864, 1999. DOI: 10.1109/69.824597.

[13] M. Christopher. The agile supply chain: Competing in volatile markets. *Industrial Marketing Management*, 29(1):37–44, 2000. DOI: 10.1016/S0019-8501(99)00110-8.

[14] X. Chu, I. F. Ilyas, S. Krishnan, and J. Wang. Data cleaning: Overview and emerging challenges. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD '16, pages 2201–2206, New York, NY, USA, 2016. Association for Computing Machinery. DOI: 10.1145/2882903.2912574.

[15] D. Deng, W. Tao, Z. Abedjan, A. K. Elmagarmid, I. F. Ilyas, S. Madden, M. Ouzzani, M. Stonebraker, and N. Tang. Entity consolidation: The golden record problem. *ArXiv e-prints*, 1709.10436v2, 12 2017. DOI: 10.48550/arXiv.1709.10436.

[16] N. Denwattana and A. Saengsai. A framework of thailand higher education dashboard system. In *2016 International Computer Science and Engineering Conference (ICSEC)*, pages 1–6, 2016. DOI: 10.1109/ICSEC.2016.7859883.

[17] U. Dombrowski, T. Mielke, and C. Engel. Knowledge management in lean production systems. *Procedia CIRP*, 3:436–441, 2012. DOI: 10.1016/j.procir.2012.07.075.

[18] C. El Morr and H. Ali-Hassan. *Descriptive, Predictive, and Prescriptive Analytics*, chapter 3, pages 31–55. Springer International Publishing, Cham, 2019. DOI: 10.1007/978-3-030-04506-7_3.

[19] S. Few. *Information Dashboard Design: The Effective Visual Communication of Data*. O'Reilly, Sebastopol, CA, 2006.

[20] Y. Gao, M. Janssen, and C. Zhang. Understanding the evolution of open government data research: Towards open data sustainability and smartness. *International Review of Administrative Sciences*, 2021. DOI: 10.1177/00208523211009955.

[21] N. Garg. *Apache Kafka*. Packt Publishing, 2013.

[22] M. Gibbons, C. Limoges, H. Nowotny, S. Schwartzman, P. Scott, and M. Trow. *The New Production of Knowledge: The Dynamics of Science and Research in Contemporary Societies*. SAGE Publications Ltd, London, 2010. DOI: 10.4135/9781446221853.

[23] R. Gitzel, S. Turring, and S. Maczey. A data quality dashboard for reliability data. In *2015 IEEE 17th Conference on Business Informatics*, volume 1, pages 90–97, 2015. DOI: 10.1109/CBI.2015.24.

[24] C. Gormley and Z. Tong. *Elasticsearch: The definitive guide*. O'Reilly, 1 edition, 2015.

[25] I. Guitart and J. Conesa. Adoption of business strategies to provide analytical systems for teachers in the context of universities. *International Journal of Emerging Technologies in Learning (iJET)*, 11(7):34–40, 2016. DOI: 10.3991/ijet.v11i07.5887.

[26] B. R. Hiraman, C. Viresh M., and K. Abhijeet C. A study of apache kafka in big data stream processing. In *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*, pages 1–3, 2018. DOI: 10.1109/ICICET.2018.8533771.

[27] How to make your data FAIR. openaire.eu/how-to-make-your-data-fair (accessed 2 March 2023).

[28] P. Howard. Total cost of ownership for business intelligence. crozdesk.com/software-research/total-cost-of-ownership-for-business-intelligence (accessed 2 March 2023).

[29] P. Huston, V. L. Edge, and E. Bernier. Reaping the benefits of open data in public health. *Canada Communicable Disease Report*, 45(10):252–256, 2019. DOI: 10.14745/ccdr.v45i10a01.

[30] D. Ince. *A Dictionary of the Internet*. Oxford University Press, Oxford, 4 edition, 2019. DOI: 10.1093/acref/9780191884276.001.0001.

[31] J. Jabłecka and B. Lepori. Between historical heritage and policy learning: the reform of public research funding systems in poland, 1989–2007. *Science and Public Policy*, 36(9):697–708, 2009. DOI: 10.3152/030234209X475263.

[32] E. Kulczycki. Assessing publications through a bibliometric indicator: The case of comprehensive evaluation of scientific units in poland. *Research Evaluation*, 26(1):41–52, 2017. DOI: 10.1093/reseval/rvw023.

[33] P. Le Noac'h, A. Costan, and L. Bougé. A performance evaluation of apache kafka in support of big data streaming applications. In *2017 IEEE International Conference on Big Data*, pages 4803–4806, 2017. DOI: 10.1109/BigData.2017.8258548.

[34] W. L. Martinez. Graphical user interfaces. *WIREs Computational Statistics*, 3(2):119–133, 2011. DOI: 10.1002/wics.150.

[35] R. Matheus, M. Janssen, and D. Maheshwari. Data science empowering the public: Data-driven dashboards for transparent and accountable decision-making in smart cities. *Government Information Quarterly*, 37(3):101284, 2020. DOI: 10.1016/j.giq.2018.01.006.

[36] M. Michajłowicz, M. Niemczyk, J. Protasiewicz, and K. Mroczkowska. Pol-on: The information system of science and higher education in poland. *European Journal of Higher Education IT 2018-1*, 2018. eunis.org/download/2018/EUNIS_2018_paper_70.pdf (accessed 2 March 2023).

[37] D. A. Norman and S. W. Draper, editors. *User Centered System Design: New Perspectives on Human-computer Interaction*. Lawrence Erlbaum Associates, 1 edition, 1986.

[38] H. Nowotny, P. Scott, and M. Gibbons. *Re-Thinking Science: Knowledge and the Public in an Age of Uncertainty*. Wiley, 2001.

[39] OECD. Open, useful and re-usable data (ourdata) index: 2019. *OECD Public Governance Policy Papers*, 01, 2020. DOI: 10.1787/45f6de2d-en.

[40] D. Power. A brief history of decision support systems. version 4.0, 2007, DSSresources.COM/history/dsshistory.html (accessed 2 March 2023).

[41] J. Protasiewicz, E. Podwysocki, S. Ostrowska, and A. Tomczyńska. Open access to data on higher education and science: A case study of the RAD-on platform in poland. In S. Bolis, J.-F. Desnos, L. Merakos, and R. Vogl, editors, *Proceedings of the European University Information Systems Conference 2021*, volume 78 of *EPiC Series in Computing*, pages 9–21. EasyChair, 2021. DOI: 10.29007/gz8q.

[42] J. Protasiewicz, S. Rosiak, I. Kucharska, E. Podwysocki, M. Niemczyk, and M. Michajłowicz. RAD-on: An integrated system of services for science – online elections for the council of scientific excellence in poland. *European Journal of Higher Education IT 2019-1*, 2019. eunis.org/download/2019/EUNIS_2019_paper_20.pdf (accessed 2 March 2023).

[43] A. Ramesh Babu and Y. Singh. Determinants of research productivity. *Scientometrics*, 43:309–329, 1998. DOI: 10.1007/BF02457402.

[44] J. Richardson, R. Sallam, K. Schlegel, A. Kronz, and J. Sun. Magic quadrant for analytics and business intelligence platforms. 2020, Gartner ID G00386610.

[45] G. Rieder and J. Simon. Datatrust: Or, the political quest for numerical evidence and the epistemologies of Big Data. *Big Data & Society*, 3(1), 2016. DOI: 10.1177/2053951716649398.

[46] Rozporządzenie Parlamentu Europejskiego i Rady (UE) 2016/679 z dnia 27 kwietnia 2016 r. w sprawie ochrony osób fizycznych w związku z przetwarzaniem danych osobowych i w sprawie swobodnego przepływu takich danych oraz uchylenia dyrektywy 95/46/we. Dz. Urz. UE. L Nr 119, 4.5.2016.

[47] Rozporządzenie ministra nauki i szkolnictwa wyższego z dnia 6 marca 2019 r. w sprawie danych przetwarzanych w zintegrowanym systemie informacji o szkolnictwie wyższym i nauce pol-on, 2019. Dz.U. 2019 poz. 496.

[48] E. Ruijer, F. Détienne, M. Baker, J. Groff, and A. J. Meijer. The politics of open government data: Understanding organizational responses to pressure for more transparency. *The American Review of Public Administration*, 50(3):260–274, 2020. DOI: 10.1177/0275074019888065.

[49] B. Scholtz, A. Calitz, and R. Haupt. A business intelligence framework for sustainability information management in higher education. *International Journal of Sustainability in Higher Education*, 19(2):266–290, 2018. DOI: 10.1108/IJSHE-06-2016-0118.

[50] B. A. Schwendimann, M. J. Rodríguez-Triana, A. Vozniuk, L. P. Prieto, M. S. Boroujeni, A. Holzer, D. Gillet, and P. Dillenbourg. Perceiving learning at a glance: A systematic literature review of learning dashboard research. *IEEE Transactions on Learning Technologies*, 10(1):30–41, 2017. DOI: 10.1109/TLT.2016.2599522.

[51] W. Solesbury. The ascendancy of evidence. *Planning Theory & Practice*, 3(1):90–96, 2002. DOI: 10.1080/14649350220117834.

[52] A. Sorour, A. S. Atkins, C. F. Stanier, and F. D. Alharbi. The role of business intelligence and analytics in higher education quality: A proposed architecture. In *2019 International Conference on Advances in the Emerging Computing Technologies (AECT)*, pages 1–6, 2020. DOI: 10.1109/AECT47998.2020.9194157.

[53] D. Spichtinger and J. Siren. *1. The Development of Research Data Management Policies in Horizon 2020*, pages 11–24. De Gruyter Saur, Berlin, Boston, 2018. DOI: 10.1515/9783110365634-002.

[54] S. D. Teasley. Student facing dashboards: One size fits all? *Technology, Knowledge and Learning*, 22(3):377–384, 2017. DOI: 10.1007/s10758-017-9314-3.

[55] B. V. Thummadi, O. Shiv, and K. Lyytinen. Enacted routines in agile and waterfall processes. In *2011 Agile Conference*, pages 67–76, 2011. DOI: 10.1109/AGILE.2011.29.

[56] Ustawa z dnia 3 marca 2018 r. – prawo o szkolnictwie wyższym i nauce, 2022. Dz.U. 2022 poz. 574 z późn. zm.

[57] A. F. van Veenstra and B. Kotterink. Data-driven policy making: The policy lab approach. In P. Parycek, Y. Charalabidis, A. V. Chugunov, P. Panagiotopoulos, T. A. Pardo, O. Sæbø, and E. Tambouris, editors, *Electronic Participation*, Lecture Notes in Computer Science, pages 100–111, Cham, 2017. Springer International Publishing. DOI: 10.1007/978-3-319-64322-9_9.

[58] D. Vohra. Apache kafka. In *Practical Hadoop Ecosystem: A Definitive Guide to Hadoop-Related Frameworks and Tools*, pages 339–347, Berkeley, CA, 2016. Apress. DOI: 10.1007/978-1-4842-2199-0_9.

[59] A. Vázquez-Ingelmo, F. J. Garcia-Peñalvo, and R. Therón. Information dashboards and tailoring capabilities – a systematic literature review. *IEEE Access*, 7:109673–109688, 2019. DOI: 10.1109/ACCESS.2019.2933472.

[60] V. Weerakkody, Z. Irani, K. Kapoor, U. Sivarajah, and Y. K. Dwivedi. Open data and its usability: An empirical view from the citizen's perspective. *Information Systems Frontiers*, 19(2):285–300, 2017. DOI: 10.1007/s10796-016-9679-1.

[61] What is open data. data.europa.eu/trening/what-open-data (accessed 2 March 2023).

[62] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C. T. Evelo, R. Finkers, A. Gonzalez-Beltran, A. J. G. Gray, P. Groth, C. Goble, J. S. Grethe, J. Heringa, P. A. C. 't Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S. J. Lusher, M. E. Martone, A. Mons, A. L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M. A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, and B. Mons. The fair guiding principles for scientific data management and stewardship. *Scientific Data*, 3:160018, 2016. DOI: 10.1038/sdata.2016.18.

[63] B. Williamson. The hidden architecture of higher education: building a big data infrastructure for the 'smarter university'. *International Journal of Educational Technology in Higher Education*, 15(1):12, 2018. DOI: 10.1186/s41239-018-0094-1.

[64] H. Wu, Z. Shang, and K. Wolter. Performance prediction for the apache kafka messaging system. In *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 154–161, 2019. DOI: 10.1109/HPCC/SmartCity/DSS.2019.00036.

[65] M. Zavyiboroda. Why user interface changes annoy people and how to handle it, 2018. speckyboy.com/annoying-ui-changes (accessed 2 March 2023).

[66] N. H. Zulkifli Abai, J. Yahaya, A. Deraman, A. R. Hamdan, Z. Mansor, and Y. Yah Jusoh. Integrating business intelligence and analytics in managing public sector performance: An empirical study. *International Journal on Advanced Science, Engineering and Information Technology*, 9(1):172–180, 2019. DOI: 10.18517/ijaseit.9.1.6694.

[67] L. Šereš, V. Pavlićević, and P. Tumbas. Digital transformation of higher education: Competing on analytics. In *INTED2018 Proceedings*, 12th International Technology, Education and Development Conference, pages 9491–9497. IATED, 2018. DOI: 10.21125/inted.2018.2348.