How to Manage IT Resources in Research Projects? Towards a Collaborative Scientific Integration Environment

Marius Politze, Florian Claus, Bela Brenger, M. Amin Yazdi, Benedikt Heinrichs, Annett Schwarz

RWTH Aachen University, Templergraben 55, 52062 Aachen, Germany {politze, claus, brenger, yazdi, heinrichs, schwarz}@itc.rwth-aachen.de

Keywords
distributed systems, eScience, research data management, service-oriented architecture

1. SUMMARY

With *CoScInE*, the *Co*llaborative *Sc*ientific *In*tegration *E*nvironment, we present a software platform to allocate and manage IT resources in research projects. While the platform itself does not store any research data, it enables integration of multiple storage resources and allows metadata management across these resources. This supports researchers in curating their research data and complying with principles to safeguard good scientific practice such as findability, accessibility and reusability. Additionally, operators of central IT infrastructures can benefit from resource allocation management and usage monitoring while complying with established scientific and economic standards.

2. INTRODUCTION

Researchers face the challenge to obtain, allocate and manage resources to store increasing amounts of research data. While various on-premise and cloud offers promise relief to their users, the necessary combination of different services introduces a new form of management problem for researchers. The freedom to choose multiple services to fit specific needs leads to a fragmentation of research data to a variety of service providers. In turn, this makes the already challenging task of research data management (RDM) even more complex.

2.1. Related Work

In general, there are several offers supporting researchers to index, publish or to retain research data long term. Nevertheless, RDM is a widely unsolved problem for many research groups (Dreyer & Vollmer, 2016). On the other hand specialized projects that target specific workflows of researchers have a wide acceptance in, often narrow, communities as shown by the different platforms created in TR CRC 32 for geographical data (Curdt, 2014), medical study data (Kirsten, Kiel, Wagner, Rühle, & Löffler, 2017) or chemical samples (Politze, Schwarz, Kirchmeyer, Claus, & Müller, 2019). To become widely accepted by the community a data management system hence has to cater towards the actual workflows of the researchers. Closing the gap between these individual workflows and central, scalable IT infrastructures thus becomes a challenge for IT service providers at the universities.

Additionally to these local setups that aim towards rich support of workflows specific to individual research groups, researchers typically employ a broad spectrum of IT service infrastructures for their projects that range from local to centralized, federated and external IT service providers. Central applications like *MetadataManager* (Politze, Bensberg, & Müller, 2019), *Radar* (Kraft, et al., 2016) or *MASi* (Grunzke, et al., 2019) are less specific and address a wider community with more generic RDM workflows. External “clouds” like *Zenodo*, *Figshare* or *Open Science Framework (OSF)* support basic RDM workflows like citation or persistent identification. By far most prominent are generic “clouds” like the *Owncloud*-based *Sciebo* (Vogl, et al., 2015), *Dropbox*, *Google Drive* or *GitLab* to store and manage data, however, these options usually lack in support of RDM workflows or policies.

2.2. State of RDM at RWTH Aachen University

Within the RDM workflows we observed that with multiple services being employed by a research project, distributed research data and cross-institutional projects the complexity of already challenging RDM tasks become a burden especially to senior researchers who are in charge of keeping track of the all resources used within their projects (Yazdi, Valdez, Lichtschlag, Ziefle, & Borchers, 2016).

We see the central service and infrastructure providers of the university like libraries and computing centers as key to support researchers in this transition. However, being central bodies, service providers at universities face challenges arising from the difference of the scientific disciplines. At our university, for example, engineering and natural sciences dominate by numbers of researchers and students. Based on our experiences in the RDM project, however, the needs of researchers severely diverge even within different areas of engineering sciences (Hausen, et al., 2018). It remains in question, how to support these discipline-specific needs with central services.

2.3. Goals and Methodology

To mitigate these challenges, we introduce *CoScInE*, the *Co*llaborative *Sc*ientific *In*tegration *E*nvironment. The platform allows researchers to manage and combine different IT resources they already use in their research projects while consequently ensuring a minimal standard for RDM. *CoScInE* mainly draws from two guidelines, the *FAIR Guiding Principles* (Wilkinson, et al., 2016) and the Code of Conduct of the German Research Foundation (Deutsche Forschungsgemeinschaft (DFG), 2019).

Another perspective to these issues comes from IT service providers, like the IT Center of RWTH Aachen University, that allocate central resources like storage and compute infrastructures or services like GitLab. These providers have to meet researchers' requirements, provide needed services and have to assure that resources are allocated according to scientific and economic standards to meet the requirements of funding agencies.

To handle the high user expectations, we have adopted the Scrum software development process. Scrum is an agile software development process for managing and delivering complex and high-quality software solutions through iterative processes of analysis, design, and implementation of functional and non-functional requirements. The Scrum framework enables us to bridge the communication gap between end-users and a development team by identifying, analyzing, documenting, and validating the most critical requirements. This process allows for a timely determination of the most valuable requirements based on interviews, use cases, and brainstormings while running on limited resources and schedules. On the one hand, this user-oriented framework allows for adoptive reprioritizations of requirements. On the other hand, this method provides us with an opportunity to discuss possible designs and technical solutions that fit the development team's competencies and adhere to the needs of users. In our development process, alongside the initial focus on the elicitation of functional requirements, we also use the data science to extract and handle non-functional requirements such as the support needed by researchers for resource allocations, privacy issues, and operational performance analysis (Yazdi, M. A., 2019).

Use cases are used to design and will be used to test the newly developed platform. The design of *CoScInE* was informed by researchers' feedback on previously existing services as well as by several use cases. In those use cases, we worked closely with research groups on finding and providing solutions to specific data management challenges. However, to ensure that the platform meets requirements we identified a group of pilot users with different use cases spanning most of *CoScInE*'s features. In one scenario researchers aim to gain an overview of all their data that is stored on different locally run platforms by curating metadata on *CoScInE* and link their local resources. Another use case focuses on storing large quantity data from computer simulations as well as providing a public interface to search and access those data. In a third use case, experimental data shall be made findable and accessible within a faculty while reserving the possibility to narrow down availability for some data. A final use case focusses on structuring scenario data by annotating it on the levels of individual files as well as collections representing scenarios and linking those to each other.

3. INTEGRATION OF PROCESSES & RESSOURCES

CoScInE acts as an information hub for researchers. It allows for interlinking existing infrastructures at universities (local and centralized) (Schmitz & Politze, 2018) as well as cloud services. *CoScInE* gathers infrastructure components from different environments, abstracts core features in generic adapters and provides an RDM platform for researchers based on these components. Figure 1 shows different dimensions of processes and infrastructure components that are integrated.

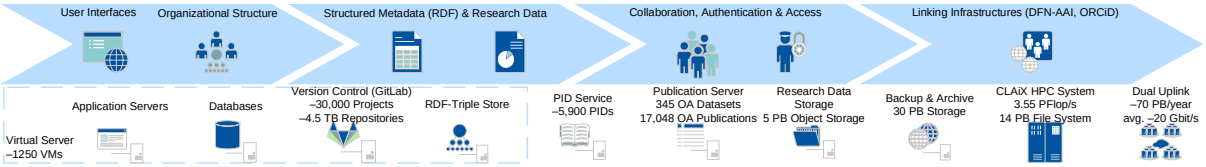

Figure 1: Central IT services at RWTH Aachen University supporting various research processes.

Note that while resource allocation and -access are managed by *CoScInE*, it functions only as an information broker since time-critical and data-intensive workflows still use the interfaces provided directly by the resources. Hence, *CoScInE* aims neither to replace nor to obscure these systems but to integrate them. This becomes vital for workflows that cross system boundaries, (e.g. moving data sets for data publication or archival) which now can be represented as consistent and cohesive functions in the user interface that also are in accordance with RDM best practices.

3.1. RDM Processes and Practices

On a high abstraction level, the German Science foundation defines the handling of research data as the retention and use of research data and information in workflows that lead to the generation of scientific knowledge. The abstract model of the research data life cycle with its different phases shows key activities that are iteratively repeated during research projects. The model assumes research data is passing through different phases of Collection and Analysis to Preservation and Re-Use as depicted by Figure 2. The goal of an RDM system thus is to provide tools for the researchers to allow transitions between phases.

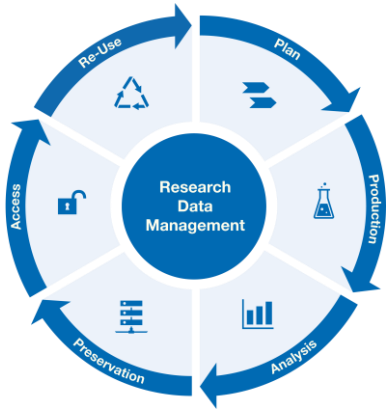

Figure 2: Research Data Management Life Cycle at RWTH Aachen University.

A second dimension is the transition between domains of access: While working with research data, each data set can be processed in different environments that define its visibility and accessibility to researchers and their peers. The resulting domain model formalizes that research data is processed in and transferred between different domains of access from personal via group to persistent (Klar & Enke, 2013).

The FAIR Guiding Principles represent the standard that each RDM platform should support. These principles place requirements towards both, the actual research data and describing metadata to allow researchers and their peers the retrieval and meaningful interpretation of re-used data.

A central integrative platform like *CoScInE* provides the opportunity to frame user interactions according to key workflows of the data life cycle, the domain model, and the requirements of the FAIR principles. A project artefact mirrors the structure of most scientific work. Projects and sub-projects on *CoScInE* allow managing membership and thus access to information and data. Projects also have a

time dimension as they have a start and an end. The end of a project is a natural point of time to trigger the decision what should happen to the accumulated data: it might be transferred to an archive, published, or directly re-used in a follow-up project. The process of archiving data will also be modeled on the platform. It will include steps to select and assemble the data that is worthy of preservation, check and improve documentation, set a projected time for archiving, and define rules for accessing the data. User interactions such as uploading data can also be used to nudge or even force users to provide additional metadata and documentation. Thereby workflows that guarantee certain standards can be implemented directly in the platform. Another process that is implemented from the start is the management of storage resources.

3.2. Resource Management

CoScInE will also be used to manage access to a new storage system for research data that recently has been procured via a grant from the state of North Rhine-Westphalia. Access to this system is managed by an application process that takes the needed storage capacity into account and includes a technical and scientific review process for larger requests.

Furthermore, users can choose between three methods of accessing the storage system. The first option is direct access via *CoScInE*: Here, users assign an application profile to the storage resource and are required to fill in the metadata for each uploaded object. Metadata can be marked as optional and mandatory and prefilled with a suggested or even fixed value. For example, if the storage is used for data produced by a microscope metadata describing the instruments' properties can be saved as fixed values and information in calibrations that rarely change, can be prefilled with a suggested value. Those options aim to reduce the effort required for documentation with metadata as much as possible. Still, data and metadata can also be uploaded via an API if researchers have already digitalized these process steps. Since by this method the proper documentation of all data is ensured, applying for storage with this access method is very easy and can, up to a certain capacity, be done automatically and entirely digital without needing human evaluation.

Alternatively, the storage can be used as a fileserver via SMB/CIFS or directly via the S3 protocol. However, since compliance with RDM standards cannot be ensured in these cases, users are required to provide a data management plan (DMP) in the application process. Additionally, beyond a certain amount of storage space applicants must demonstrate the scientific viability by referring a grant ID that proofs that their endeavor has cleared a scientific review process.

Once the application process has been successful, the provision of the storage resources proceeds automatically and applicants can start using it.

Though the process is implemented for a particular storage system at RWTH Aachen University, it is sufficiently generic to adapt it for the management of other storage systems. Those might be the storage systems of a partner university or even commercial offers such as Amazon Web Services or Microsoft Azure. However, at German universities the usage of such commercial offers is primarily a legal and administrative challenge.

4. SOLUTION ARCHITECTURE

As the implementation of CoScInE arose from the need to bringing together scientific projects and allocation of IT resources for these projects, the solution architecture follows a highly generic approach. In general, there are two dimensions need to be considered: the supported IT systems and the supported, generic RDM processes and best practices.

4.1. Supporting IT Systems

CoScInE makes use of several existing high-level components for authentication, data storage and metadata. Identities are collected and managed in a way that allows collaboration and identification across organizations, highlighting the increasing importance of authentication and authorization infrastructures as operated by *eduGAIN* and researcher identifiers like *ORCID* (Haak, Fenner, Paglione, Pentz, & Ratner, 2012). *CoScInE* makes use of the results of the *AARC* project (Liampotis, 2019) to allow combining academic and "cloud" identity providers.

As a management UI, *CoScInE* uses a heavily customized instance of Microsoft SharePoint. The portal features of SharePoint form an ideal base as they already take care of basic technical workflows like sign-on, user and or rights management. Applications implementing research data workflows can then at least partially rely on this base infrastructure but still have to mirror some functionality to allow for an independent API implementation. The *CoScInE API* forms the central hub to coordinated "manual" actions from the management UI as well as from "scripted" applications. These allow researchers to implement individual scenarios as compared to the centralized RDM workflows of the provided applications.

Resource Adapters abstract specific interfaces of storage (and potentially other) services and define an interface with common operations. Currently, *CoScInE* uses the *Waterbutler API* (Center for Open Science, 2020) from the *OSF* project to connect 15 cloud storage providers. *CoScInE* assigns a persistent identifier that allows global identification to each resource. Sub-identifiers make each data set uniquely identifiable. Discipline-specific metadata across the resources makes research data discoverable in the system. *CoScInE* utilizes the *W3C* standards *RDF* (Cyganiak, Wood, & Lanthaler, 2014) and *SHACL* (Knublauch & Kontokostas, 2014) as the internal metadata model and for validation. This allows researchers to enrich their data independently of its actual storage location. See Figure 3 for a representation of the composition and interaction of high-level components.

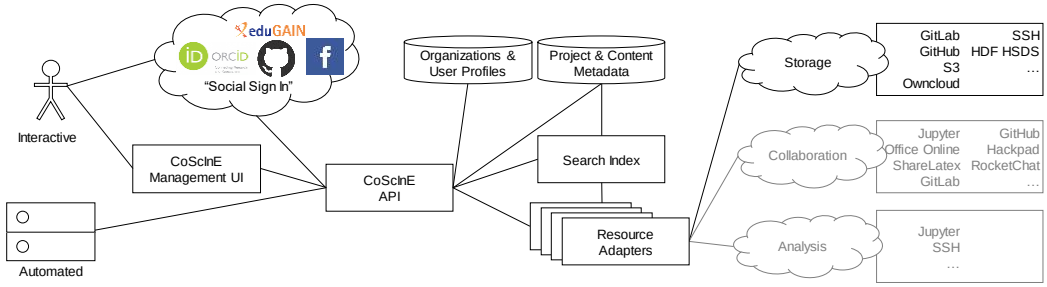

Figure 3: Overview of the system architecture for resource management of *CoScInE*.

In accordance with the agile methodology, within the first case studies, only some of the features are fully accessible to the users within the case studies. More specifically, *CoScInE* allows the integration of GitLab projects and object store buckets using the *S3* interface. Both services are offered by the IT Center of RWTH Aachen University. While both services, in general, were accessible by researchers without *CoScInE*, consolidation within the data management environment allows association with scientific projects and enables advances provisioning workflows and user management within the central interface.

4.2. Supporting RDM Practices

By integrating different resources into the management environment, *CoScInE* aims to introduce support for *FAIR* Guiding Principles for these resources. In the aforementioned example of storage providers, the *Handle* based *ePIC* PID system allows to identify storage location and contained files or, more generically, data objects uniquely on a global scale. Instead of assigning PIDs to individual objects, *CoScInE* uses sub-identifiers to deeply link to resources' contents. The *ePIC* PID system allows using fragment or template identifiers. As such *CoScInE* appends an ID derived from the internal structure of the resource to the PID. As a base assumption for most storage providers, the file path relative to the root folder of the resource is quite suitable. More advanced storage providers however allow for additional identifying information like version hashes computed by git repositories. With the extended PID this allows *CoScInE* to identify a certain file version without the overhead of minting PIDs for every single file in the resource.

The PID generated by these rules is then used by the metadata management component of *CoScInE* to identify individual data objects within the resource. With the PID resolver as a base URL, the extended PIDs are suitable to become nodes in a linked data knowledge graph that can store meta information. To form this knowledge graph *CoScInE* uses *RDF* triples to express the metadata information using triples of the form subject-predicate-object where the subject is the data object represented by the generated PID, the predicate is the metadata property and the object is the metadata value. Each resulting partial graph for a single research object is then stored in a *Virtuoso* RDF database for persistence and retrieval.

Storing metadata in a "schema-less" *RDF* allows flexibly reacting to changes in researchers demand for recording information with their data, independently of its actual storage location. However, this leads to the problem that recorded metadata can be quite unstructured. The *SHACL* application profiles associated with each resource, however, allow introducing a structural component to (partial) *RDF* graphs. Defined and consistent metadata predicates additionally allow triggering RDM workflows like archival, publication or retention of individual data objects or entire resources.

In connection with the *FAIR* Guiding Principles, *CoScInE* hence ensures that data objects and the corresponding metadata, linked by the fragment PID, are findable and accessible through the *CoScInE* API independently of the actual storage provider. Using *RDF* triples for the representation of metadata additionally widely covers the demands on interoperability and re-usability for the stored metadata.

For organizations concerning to comply with good research practices like the "Code of Conduct" by the German Research Foundation, this approach may even allow defining RDM workflows across the resource providers integrated into the system. For these integrated resources, *CoScInE* could enforce retention or archival periods after a research project ended (as demanded by Guideline 17). In the same manner, *CoScInE* may allow documenting the work process leading to data-driven research results (as demanded by Guidelines 12 and 13).

5. OUTLOOK

CoScInE just started in a pilot phase for users at RWTH Aachen University. The development team will continuously improve and extend the existing functionalities based on implicit and explicit user feedback. The focus will be on the support of more sophisticated RDM workflows like data publication, archival and automatic extraction of metadata from contents of managed resources.

All software components are developed as open-source (RWTH Aachen University, 2020). The microservice architecture allows reusing of individual components in other systems. Consequently, other universities or research groups can adopt standardized interfaces to their local environments.

6. REFERENCES

Center for Open Science. (2020, 01 20). *Waterbutler*. Retrieved from https://github.com/CenterForOpenScience/waterbutler/

Curdt, C. (2014). *Design and Implementation of a Research Data Management System: The CRC/TR32 Project Database (TR32DB)*. Cologne, Germany: Universität zu Köln.

Cyganiak, R., Wood, D., & Lanthaler, M. (2014). *RDF 1.1 Concepts and Abstract Syntax*. Retrieved from https://www.w3.org/TR/rdf11-concepts/

Deutsche Forschungsgemeinschaft (DFG). (2019). Guidelines for Safeguarding Good Research Practice. Bonn, Germany.

Dreyer, M., & Vollmer, A. (2016). An Integral Approach to Support Research Data Management at the Humboldt-Universität zu Berlin. In Y. Salmatzidis. Thessaloniki, Greece.

Grunzke, R., Hartmann, V., Jejkal, T., Kollai, H., Prabhune, A., Herold, H., . . . Nagel, W. E. (2019). The MASi repository service. *Future Generation Computer Systems, 94*, pp. 879–894.

Haak, L. L., Fenner, M., Paglione, L., Pentz, E., & Ratner, H. (2012). ORCID: a system to uniquely identify researchers. *Learned Publishing, 25*(4), pp. 259–264.

Hausen, D. A., Eich, U., Brenger, B., Claus, F., Magrean, B., Müller, M. S., . . . Wluka, A.-K. (2018). Introducing Coordinated Research Data Management at RWTH Aachen University. A Brief Project Report.

Kirsten, T., Kiel, A., Wagner, J., Rühle, M., & Löffler, M. (2017). Selecting, Packaging, and Granting Access for Sharing Study Data. In M. Eibl, & M. Gaedke, *INFORMATIK 2017: Digitale Kulturen* (pp. 1381–1392). Bonn, Germany: Köllen.

Klar, J., & Enke, H. (2013). Projekt RADIESCHEN. *Rahmenbedingungen einer disziplinübergreifenden Forschungsdateninfrastruktur*.

Knublauch, H., & Kontokostas, D. (2014). *Shapes Constraint Language (SHACL)*. Retrieved from https://www.w3.org/TR/shacl/

Kraft, A., Razum, M., Potthoff, J., Porzel, A., Engel, T., Lange, F., . . . Furtado, F. (2016). The RADAR Project – A Service for Research Data Archival and Publication. *ISPRS International Journal of Geo-Information, 5*(3), p. 28.

Liampotis, N. (2019). AARC Blueprint Architecture 2019.

Politze, M., Bensberg, S., & Müller, M. S. (2019). Managing Discipline-Specific Metadata Within an Integrated Research Data Management System. In J. Filipe, M. Smialek, A. Brodsky, & S. Hammoudi. SCITEPRESS - Science and Technology Publications.

Politze, M., Schwarz, A., Kirchmeyer, S., Claus, F., & Müller, M. S. (2019). Kollaborative Forschungsunterstützung: Ein Integriertes Probenmanagement.

RWTH Aachen University. (2020, 01 20). *CoScInE*. Retrieved from https://git.rwth-aachen.de/coscine/

Schmitz, D., & Politze, M. (2018). Forschungsdaten managen – Bausteine für eine dezentrale, forschungsnahe Unterstützung. *o-bib. Das offene Bibliotheksjournal, 5*(3), pp. 76–91.

Vogl, R., Angenent, H., Rudolph, D., Thoring, A., Schild, C., Stiegliz, S., & Meske, C. (2015). "sciebo — theCampuscloud" for NRW. In M. Turpie. Dundee, Scotland.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., . . . Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data, 3*.

Yazdi, M. A. (2019). Enabling Operational Support in the Research Data Life Cycle. *Proceedings of the 1st International Conference on Process Mining - Doctoral Consortium* (pp. 1-10). CEUR.

Yazdi, M. A., Valdez, A. C., Lichtschlag, L., Ziefle, M., & Borchers, J. (2016, July). Visualizing opportunities of collaboration in large research organizations. *Proceedings of the International Conference on HCI in Business, Government, and Organizations* (pp. 350-361). Springer.

7. AUTHORS' BIOGRAPHIES

Dr. Marius Politze is head of the group "Process and Application Development for Research" at the IT Center of RWTH Aachen University. His research focusses on service-oriented architectures supporting university processes. He received his doctorate in the area of distributed process supporting IT-systems in 2019. In 2012, he finished his M.Sc. studies at Maastricht University in the area of in Artificial Intelligence and, in 2011, his B.Sc. studies in Scientific Programming at FH Aachen University of Applied Sciences. Since 2008, he held various posts at the IT Center as a software developer, software architect and as a teacher for scripting and programming languages.

Florian Claus is head of the group "Process and Application Consulting for Research" at the IT Center of RWTH Aachen University. He studied Political Science, Economics, and Linguistics and received his Master of Arts from RWTH Aachen University in 2010. After working as a research assistant in the Institute for Political Science he joined the IT Center in 2011. Here his tasks included support and training, documentation, knowledge management, and research data management. Since 2015 he works in RDM full time and is a member of the university's RDM team, with a focus on consultancy and requirement engineering.

Bela Brenger is a member of the "Process and Application Consulting for Research" group at the IT Center of RWTH Aachen University. As the product owner, he is responsible for requirement engineering, prioritization, and evaluation to ensure the quality and functionality of the integration platform. In 2015, he graduated from RWTH Aachen University with a degree in Technical-Communication. Since 2017 he held positions at the IT Center as product owner, RDM consultant and project manager.

M. Amin Yazdi is a Ph.D. candidate in the field of data mining and process mining. His research focus is on enhancing User Experience (UX) and discovering implicit user requirements with the help of data science techniques. Since 2015, his professional industrial expertise lies in enabling and maintaining agile project management (Scrum), executing usability analytics and leading the team to go from conceptualization to realization of IT projects. He received his M.Sc. in Media Informatics at RWTH Aachen University with a major in Human-Computer-Interaction (HCI).

Benedikt Heinrichs is a research associate and lead developer of the group “Process and Application Development for Research” at the IT Center RWTH Aachen University since 2018. His research focuses on data provenance, metadata extraction and similarity detection. He received his M.Sc. in Artificial Intelligence from Maastricht University in 2018. In 2016, he finished his B.Sc. studies in Scientific Programming at the FH Aachen University of Applied Sciences. From 2013 until 2018, he worked at the IT Center as a software developer.

Dr. Annett Schwarz is the RDM coordinator of RWTH Aachen University. She coordinates the research data management team comprising experts from the Department Research and Career, the IT Center, and the University Library. She received her doctorate in the area of electron structure quantum monte carlo in 2011. She studied chemistry (diploma in 2007) at RWTH Aachen University and Business Administration and Economics at FernUniversität in Hagen (Bachelor of Science in 2019). After working as a research assistant at the institute of physical chemistry and the chair of technical thermodynamics at RWTH Aachen University, she joint the IT Center of RWTH Aachen University in 2018.