

AUTOMATIC DEPLOYMENT OF A REPOSITORY OF DIGITAL DIDACTIC DOCUMENTS AT UNIVERSITIES

Sławomir KOCZUBIEJ¹, Marzena NOWAKOWSKA², Paweł STAPÓR^{3*},
Paweł ŚWIETLIK⁴, Damian WALCZYK⁵

¹ Kielce University of Technology, Faculty of Management and Computer Modelling;
slawomir.koczubiej@tu.kielce.pl, ORCID: 0000-0003-0423-6064

² Kielce University of Technology, Faculty of Management and Computer Modelling; spimn@tu.kielce.pl,
ORCID: 0000-0002-6934-523X

³ Kielce University of Technology, Faculty of Management and Computer Modelling; stapor@tu.kielce.pl,
ORCID: 0000-0002-2345-1707

⁴ Kielce University of Technology, IT Centre; pswietlik@tu.kielce.pl, ORCID: 0000-0002-3970-6620

⁵ Kielce University of Technology, IT Centre; walczyk@tu.kielce.pl, ORCID: 0009-0006-2609-9661

* Correspondence author

Purpose: This work aims to establish a template containing a skeletal version of the existing Repository of Didactic Processes Data (RDPD) along with its environment and, by utilizing this system, to develop universal installation scripts enabling the implementation of the repository at any university in Poland. These scripts are used for cloning, meaning the template replication. They are intended to enable the implementation of Repository as a Service (RaaS) launched on demand by the service recipient.

Design/methodology/approach: The RDPD skeletal version is developed as a system offering repository functionality for managing data from didactic processes. It does not contain visual identification elements and data. To operate, the skeletal version needs an environment that requires creating a VM on Proxmox and installing the OS and applications needed for RDPD: PostgreSQL, Apache, Tomcat, Solr, and Angular. The RDPD skeletal version and its environment create a template which is cloned by Ansible scripts running sequentially. Ansible scripts also configure the created clone. The university then only needs to customize the system and feed the repository with data.

Findings: The university (client) provides identification parameters to generate a virtual machine (VM), ultimately containing the repository. In response, the RDPD system is automatically replicated and configured on a clone of the template in line with the parameters provided.

Practical implications: A solution was proposed allowing for the automatic implementation of the RDPD repository at any Polish university. Hence, the provision of a service to generate a ready-to-use version of the RDPD system is significantly convenient for universities.

Originality/value: The developed RDPD system, dedicated strictly to storing and managing data generated by a university, can be automatically implemented and configured per request. There are no such solutions, which is the original element of this work.

Keywords: repository as a service, data management, automatic replication and configuration, batch loading data, DSpace.

Category of the paper: Conceptual paper, Case study.

1. Introduction

Data generated during the educational processes at a university include didactic materials, curricula, test reports, theses, authorities' orders, reports for accreditation bodies, study plans, etc. Some of those are prepared in paper form while others are prepared electronically; some are stored, and others, having fulfilled their function, are disposed. All of them however provide information which can be used to organize work and manage the university in terms of educational needs. Hence, they should be collected in a single institutional repository with a logical and intuitive structure.

The authors have developed such a repository as the Repository of Didactic Processes Data (RDPD) DSpace-based system for the management of data generated in didactic processes for Kielce University of Technology (KUT), Poland (Koczubiej et al., 2022). It is the outcome of a larger project titled "National Data Storage (NDS). Universal infrastructure for sharing, storage and effective processing of large data volumes in HPC, Big Data and Artificial Intelligence models" (National Data Storage..., 2024), Intelligent Development Operational Program, Agreement: POIR.04.02.00-00-D010/20-00. Work on the Repository of Didactic Processes Data system commenced in April 2021. However, the system is structured in a way which allows other universities to use it.

The RDPD repository consists of a set of services (e.g., database, backend and frontend) which need to be properly installed, configured and linked together. Therefore, the RDPD implementation requires expertise in various areas of IT and competence in data management. Hence, it would be a significant convenience for universities wishing to use such a system to provide a service that generates a ready-to-use version of RDPD. Following this idea, the authors have implemented an advanced solution for the automatic deployment of RDPD.

It is particularly difficult to find publications covering issues similar to those presented in this work. Related literature was sought out (2024/06/12) in the three largest bibliographic databases: Scopus (www.scopus.com), Web of Science (www.webofscience.com), and Dimensions (app.dimensions.ai/discover/publication) using the following keywords: *repository replication, repository as a service, repository automatic deployment*. A single work was found that presents the concept of automatic generation of a Repository as a Service (RaaS) system: "Digital repository as a service: automatic deployment of an Invenio-based repository using TOSCA orchestration and Apache Mesos" by Antonacci et al. (2019). The authors used containerization for this automation and employed an orchestrator to configure and manage distributed cloud resources. They developed a demonstrator for the Arts and Humanities Research domain. In contrast, in this work, the automatization of the RDPD implementation was prepared with the use of virtualization. In practical terms, the developed RDPD is a working system which can serve any university.

2. RDPD implementation process

The RDPD implementation process is presented in Figure 1. The RDPD skeletal version is created at the outset, serving as a source for the RDPD template – this is a one-time action. Any university can request the deployment of RDPD by completing the order form. Following positive verification, the replication of the RDPD system (its clone) is automatically generated based on the RDPD template, with the use of a set of Ansible scripts (Ansible Documentation, 2024). In the next step, the university adapts the clone to its specific requirements. Finally, the system can be fed with data. Once the implementation is complete, the repository endpoint is provided to the users. They can browse, search, upload and download data from the didactic processes contained in the repository via the dashboard.

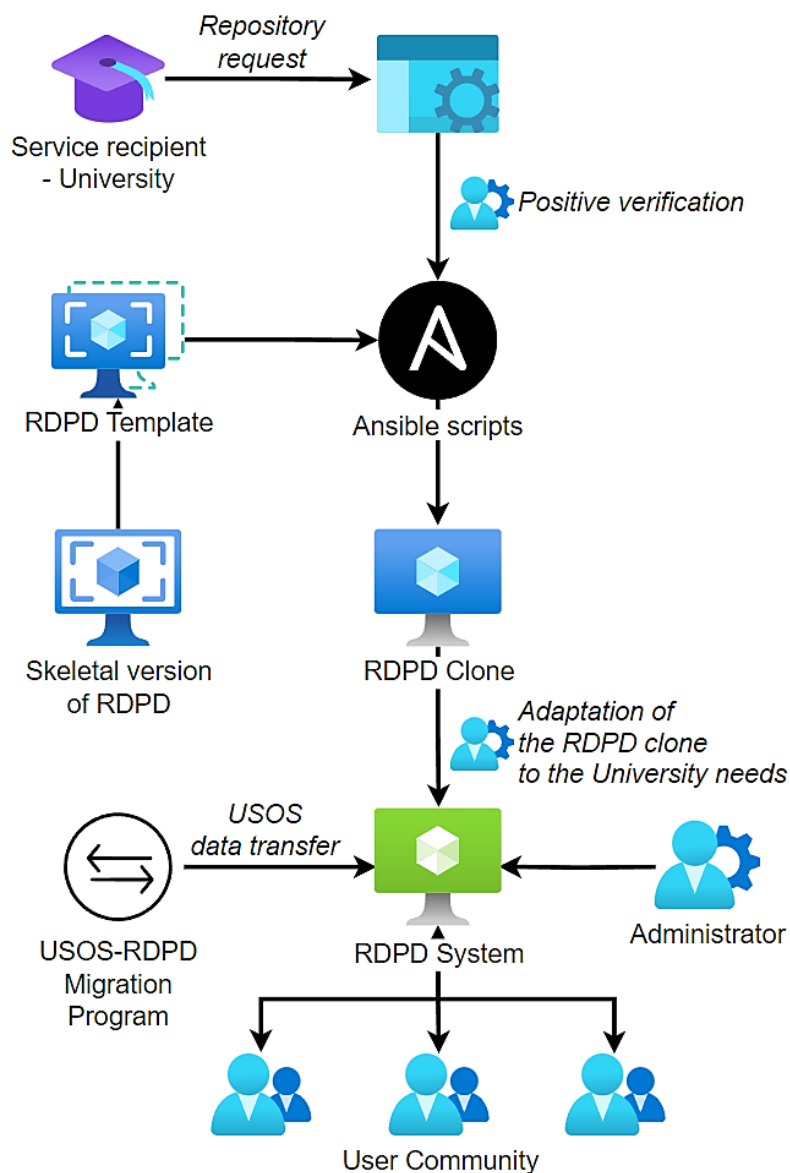


Figure 1. The RDPD system implementation workflow.

Source: authors' elaboration.

3. Skeletal version of the RDPD system

3.1. System specification

The skeletal version of RDPD was developed based on the existing open-source repository system DSpace (DSpace 7.x Documentation, 2024; DSpace, 2024). Logical metadata layers were developed, followed by implementation, adopting a customized solution in line with the requirements of the authors' metadata schema. RDPD reflects the typology of documents from the university's didactic processes and not its organizational structure (departments, chairs, institutes). Six communities are at the top of the hierarchy structure: *Students' academic performance*, *Fields of study*, *Orders of the university authorities*, *Teaching activity organization*, *Students' accomplishments*, and *Various others*. Communities contain collections, which are groups of thematically related content; for example, the *Students' academic performance* community contains the collections: *Thesis* and *Course credit protocols*. Each collection consists of items which constitute the basic archive components of the Repository. For instance, the *Theses* collection covers engineering, bachelor's and master's theses of all university students. User groups are also created in RDPD. The groups have permissions defined according to their function in the didactic processes:

- *Anonymous* refers to any Internet user, in particular a student.
- *Academic teachers*, refers to all teachers at the university.
- *University authorities* refer to the rector, vice-rectors, deans, deputy deans, and other heads of teaching units.
- *Administrator* refers to those responsible for managing the system.

The RDPD system has Polish and English localizations, with the English localization being part of the main DSpace distribution, while the Polish localization was implemented according to the authors' concepts.

3.2. Polish localization

3.2.1. Back-end language support

Activating the Polish-language version of the RDPD system required configuring two parameters in the file `[dspace-source]/dspace/config/local.cfg`:

- `default.locale`, the value for RDPD: `default.locale = pl`
- `webui.supported.locales`, the value for RDPD: `webui.supported.locales = en, pl`

The language settings in the `local.cfg` file overwrite the settings defined in the `config.cfg` file. The `default.locale` parameter defines the default language for communicating with the users when the system is opened (loaded). The `webui.supported.locales` parameter lists the language codes and allows for:

- user's selection of preferred language as part of their profile,
- switching the session language,

- sending e-mails by the system in the language selected for the session,
- editing the file containing messages in the language selected for the session (*dspace-admin Edit News* command).

Depending on the languages to be supported, all language version files must be obligatorily prepared according to i18n (W3C. Internationalization (i18n) Activity, 2024). In the case of RDPD, this applies to Polish and English.

Preparations of back-end resources to support the language include the following elements:

- List of messages. The list of messages in English is part of the main DSpace distribution. It is stored in the file:

[dspace-source]/dspace-api/src/main/resources/Messages.properties

A list of Polish equivalents of these messages should be prepared in the *Messages_pl.properties* file, which must be saved in the folder:

[dspace-source]/dspace/modules/server/src/main/resources/

Setting the parameter *default.locale* determines which file with a list of messages is used.

- Contents of e-mails. The contents of the e-mails sent automatically by the DSpace system are stored in files located in the folder *[dspace]/config/emails/*. Each file has a unique name providing information about its content (e.g., the *welcome* file contains feedback about the successful registration of the user to the repository). For the RDPD system, the authors created a set of such files with appropriate content in Polish, with the file names extended by adding the *_pl* ending (e.g., the equivalent of the *welcome* file is the *welcome_pl* file).
- Metadata language. In RDPD (according to the DSpace concept), the user can connect each metadata field value with a language code at the stage of defining this field.
- Submission forms. Due to the variety of metadata structures in the RDPD system, the authors developed a separate submission form in English and Polish for each of its collections. The definitions of the English version of the forms are stored in the *submission-forms.xml* file, while the definitions of the Polish version are stored in the *submission-forms_pl.xml* file. Both files must be saved in the folder */opt/dspace/config*. For each different language version, a new file must be created, in the name of which, instead of *pl*, the code of the corresponding language must be specified in accordance with the ISO 639-1 list (ISO 639 - Language codes, 2022).

3.2.2. Front-end language support

By default, DSpace checks the language of the user's browser. If a file with interface translations in that specific language is available in the system, the page content is rendered accordingly. Otherwise, English or another language configured in DSpace is used by default. Language package files contain texts displayed in widgets (components) of the application interface, such as button descriptions, messages, and menu items.

English language package file *en.json5* is part of the main DSpace distribution. An additional, authors' file of the Polish language package *pl.json5* was defined for RDPD. Both files must be saved in the folder */src/assets/i18n/*. In the *config.yml* file, in the */opt/dspace/ui/config* folder, one can configure the default language, and the languages supported in the user interface. The default language of the RDPD system is Polish, and the supported languages are Polish and English.

4. Creation of the RDPD template for cloning

A template is a resource that enables the generation of a ready-to-use version of RDPD at the user's request. This is a significantly convenient function for the university, as it does not require specialist knowledge in various IT areas or competences in data management.

The template is created following the steps described below.

- Creation of Virtual Machine (VM) on Proxmox. Proxmox VM is a virtualization environment for managing servers running Windows and Linux operating systems (Proxmox VM, 2024). It supports dynamic scaling of compute infrastructure and storage as the data center needs to grow. Proxmox VM uses the QEMU emulator (QEMU (Quick Emulator), 2024), which, in combination with the KVM (Kernel-based Virtual Machine), 2024) hypervisor integrated with the Linux kernel, offers hardware-assisted virtualization with performance close to a real computer system.
- Completing the template resources on the virtual machine. This step includes the following actions:
 - installing and configuring the operating system,
 - installing applications required for DSpace: PostgreSQL (PostgreSQL, 2024), Apache (Apache, 2024), Tomcat (Tomcat, 2024), Solr (Solr, 2024), Angular (Angular, 2024),
 - installing the RDPD skeletal version,
 - configuring the connection to the Ansible server using an SSH key,
 - disabling the administrator password for later use of Ansible scripts; this is an operation required to automatically execute the scripts (without the administrator intervention),
 - stopping VM.
- Converting (transforming) the template resources into a form ready for cloning. This form defines the template.

5. RDPD cloning

A portal for servicing the *pracelab* projects (Cooperation in Advanced Computing in Europe – *pracelab*, 2024) and the NDS project (NDS – National Data Storage, 2024) is available. Universities can place orders for the RDPD system through the portal. The form with the necessary data has to be filled out, as shown in Figure 2. The *Mail user* and *Password* in the *Mail server data for the RDPD system* panel are used for automatic e-mail communication with the system users.

Figure 2. RDPD skeletal system order form - English equivalent.

Source: authors' elaboration.

After placing the order and verifying the data, a copy (clone) is automatically created from the prepared template. Next, the clone is configured. These operations are performed by running an original shell script, containing calls to Ansible scripts (also the original ones) in a specific order (the names of the authors' scripts are provided in parentheses):

- Script 1 (*automate.yml*) creates (clones) a new VM with RDPD and the associated software from the template. The script uses information from the *hosts* file: the IP of the server where the clone is created and the data used to generate *hosts* files for subsequent Ansible scripts.
- Script 2 (*reconfigip.yml*) configures the clone's network settings: in the *netplan* configuration file, changes the IP address of the master server to the IP address of the clone server, creates an authenticated connection between the clone and the Ansible server using an SSH key, and triggers a re-read of the *netplan* configuration file.
- Script 3 (*configrdpd.yml*) configures the network parameters of internal communication of RDPD modules: sets the IP for the database server, front-end, and back-end using the same IP address, configures the mail account for e-mail correspondence sent by the RDPD system. At the end of the entire cloning operation, the script revokes the user's rights to execute shell commands on the clone server as *sudo* without entering a password (see Section 2).

Both the template and the clones are stored on the hardware resources of the Kielce University of Technology IT infrastructure created as part of the NDS project (service servers, CEPH servers, network infrastructure).

6. Adaptation of the RDPD system to the needs of a university

The administrator can customize the RDPD system to suit the needs of their respective institution in terms of:

- visual identification of the university,
- drop-down lists in the submission forms,
- federated logins.

These operations require modifications of specific configuration files of the DSpace, Tomcat and Apache systems, which may result in the necessity to recompile and restart the services.

6.1. Visual identification of the university

The university's visual identification includes the editing and arrangement of the resources as follows (see Figure 3):

- terms of service provided in the file */opt/dspace-ui/src/app/info/end-user-agreement/end-user-agreement-content/end-user-agreement-content.component.html*
- privacy policy provided in the file */opt/dspace-ui/src/app/info/privacy/privacy-content/privacy-content.component.html*
- university-approved typeface and color scheme configured in the Sass (Syntactically Awesome Style Sheets) cascading style sheet format (the file: */opt/dspace-ui/src/themes/rdpd/styles/_theme_sass_variable_overrides.scss*)
- header logo configured in the file */opt/dspace-ui/src/themes/rdpd/app/header/header.component.html*
- menu logo configured in the file */opt/dspace-ui/src/themes/rdpd/app/navbar/navbar.component.html*
- background and text on the homepage configured in the file */opt/dspace-ui/src/themes/rdpd/app/home-page/home-news/home-news.component.html*
- favicon image files along with the manifest stored in the directory */opt/dspace-ui/src/themes/rdpd/assets/favicons*

Logo and background graphic files are located in the application's resources directory */opt/dspace-ui/src/themes/rdpd/assets/images/*.

The screenshot shows the homepage of the Repository of Didactic Process Data (RDPD) at Kielce University of Technology. The header features a logo and navigation links. The main content area includes a search bar, a list of community categories, and a section for recent announcements. Callout boxes point to specific design features: the logo and menu, the background image and text, the university-approved typography and color scheme, and the terms of service and privacy policy link.

Logo in the header and menu

Background and text on homepage

University-approved typeface and color scheme

Terms of service and privacy policy

Figure 3. RDPD visual identification for Kielce University of Technology.

Source: authors' elaboration.

6.2. Drop-down lists in submission forms

Submission forms (Polish and English versions) embedded in the template contain drop-down lists for the following metadata fields – see the work by Koczubiej et al. (2022):

- *rdpd.dyscyplinyNau* – discipline of science (e.g., computer science, mathematics, management and quality studies),
- *rdpd.formaZaj* – a form of teaching (e.g., lecture, laboratory, project),
- *rdpd.klasyfikacja* – students’ accomplishment classification (scientific, sports, artistic, other),
- *rdpd.poziomKsz* – education cycle (e.g., first-cycle studies, second-cycle studies),
- *rdpd.rodzaj* – type of action to promote the university (e.g., workshop, lecture, interview),
- *rdpd.rokAka* – academic year (e.g., 2023/2024, 2024/2025),
- *rdpd.semestr.nazwa* – semester name (summer, winter),
- *rdpd.semestr.numer* – semester number (e.g., 1, 2, 3, 4),
- *rdpd.studia.profil* – study profile (practical, general academic),
- *rdpd.studia.tryb* – study mode (full-time, part-time),
- *rdpd.wydzial* – faculty name and its Polish acronym (e.g., Faculty of Management and Computer Modelling – WZiMK, Faculty of Mechatronics and Mechanical Engineering, – WMiBM).

The drop-down lists can be customized for a particular university by modifying the contents of the files (DSpace 7.x Documentation): *submission-forms.xml* and *submission-forms_pl.xml*.

6.3. Federated logins

Regardless of the standard one, it is possible to log in to the RDPD system in a federated mode, without the need for explicit registration. Such a solution is made possible due to the fact that the DSpace system cooperates with Shibboleth. Shibboleth is a distributed system for secure user authentication and transfer of user attributes from one or more identity providers (Shaping the future of Shibboleth Software, 2024). To use the Shibboleth system, an external identity provider (IdP) is required that supports Shibboleth technology – a university system. DSpace will then act as a Service Provider (SP), which receives authentication information and then provides the service to the user based on it. In this case, DSpace requires that the Apache server has the appropriate Shibboleth module installed to act as a proxy for all HTTP requests to the application container (Tomcat). DSpace receives user authentication information from this module via HTTP headers.

After logging into the RDPD via the Shibboleth system, the user is automatically assigned to the *Academics* group representing all the university’s teachers. If he had a previously registered account in the RDPD system, the user’s rights will be the sum of the rights of the groups they are assigned to and the *Academics* group.

7. Batch data transfer to RDPD

Apart from uploading data into the system via screen forms, RDPD enables batch data loading using the DSpace Simple Archive Format. This is the functionality of DSpace, which is particularly useful for loading data from other university systems. The process can be automated by creating a dedicated computer program. Such applications were developed for the RDPD system at Kielce University of Technology and support the following collections: *Thesis*, *Course credit protocols*, and *Full-time study timetables*. An example diagram of the process of transferring data from university systems to the *Thesis* collection using these programs is shown in Figure 4. The Polish abbreviations USOS and APD stand for: University Study Service System and Diploma Theses Archive, respectively.

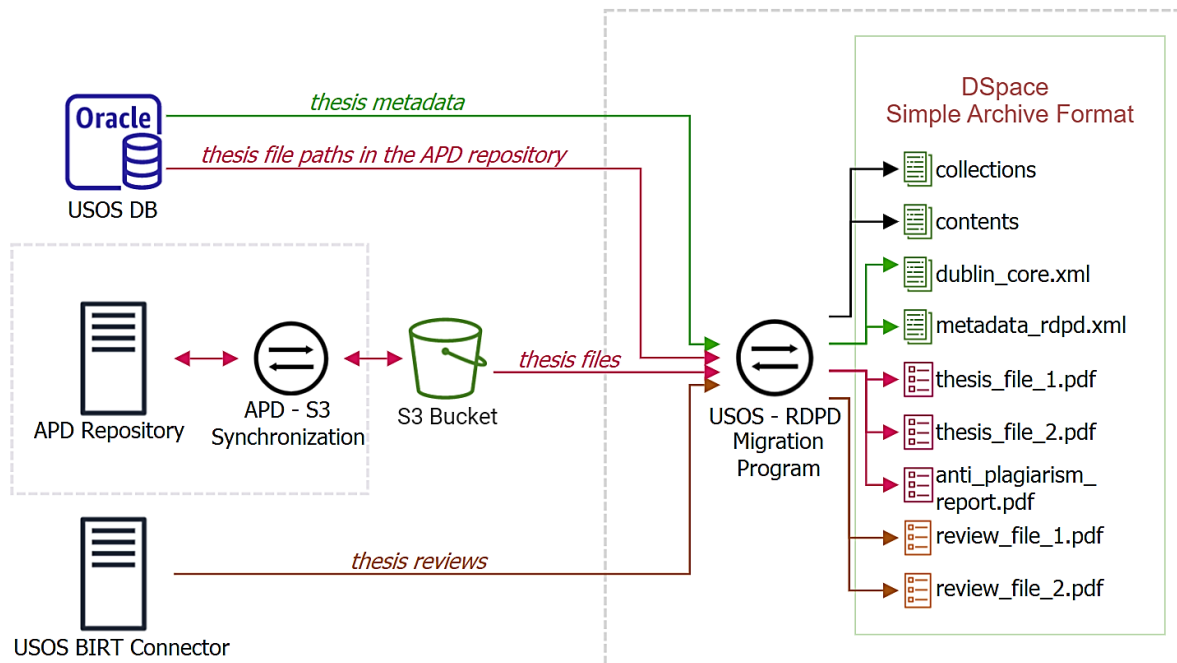


Figure 4. Diagram of data transfer from university systems to RDPD.

Source: authors' elaboration.

8. Summary and conclusions

The RDPD system was prepared for Kielce University of Technology, Poland, within the project titled “National Data Storage (NDS). Universal infrastructure for sharing, storage and effective processing of large data volumes in HPC, Big Data and Artificial Intelligence models”, Intelligent Development Operational Program, Agreement: POIR.04.02.00-00-D010/20-00. The system is a production version for a specific university.

The experience gained while working on the RDPD production version allowed for the development of a skeletal version in a manner which allows to implement it in other universities. This version is universal without elements of visual identity and federated login.

The skeletal version serves as the basis of the RaaS – Repository as a Service. For the service to work, an automatic installation process was developed at the request of a university customer. This required the use of advanced information technology including the creation of the RDPD template and the cloning of the template. Cloning is defined in the form of a list of operations to be performed to obtain a ready-to-use RDPD system. Thus, the system is made available to other universities in Poland. The university can customize the system. End users can access the repository, designed to store and manage data generated by the university's didactic processes, enter their metadata and data or download data.

Acknowledgements

This work was financially supported by the KMD (NDS) project funded at the Smart Growth Operational Programme 2014-2020, Measure 4.2 “Development of modern research infrastructure of the science sector” under the grant agreement no. POIR.04.02.00-00-D010/20.

References

1. *Angular*. Retrieved from: <https://angular.dev/>, 1.07.2024.
2. *Ansible documentation*. Retrieved from: <https://docs.ansible.com/>, 23.05.2024.
3. Antonacci, M., Brigandì, A., Caballer, M., Cetinic, E., Davidovic, D., Donvito, G., Moltó, G., Salomoni, S. (2019). Digital repository as a service: automatic deployment of an Invenio-based repository using TOSCA orchestration and Apache Mesos. *EPJ Web Conf. - 23rd International Conference on Computing in High Energy and Nuclear Physics (CHEP 2018), Vol. 204, Article Number 07023*, doi: <https://doi.org/10.1051/epjconf/201921407023>.
4. *Apache*. Retrieved from: <https://httpd.apache.org/>, 1.07.2024.
5. *Cooperation in Advanced Computing in Europe – prancelab* (in Polish: *Współpraca w zakresie zaawansowanych obliczeń w Europie - prancelab*). Retrieved from: <https://prace-lab.pl/>, 23.05.2024.
6. *DSpace 7.x Documentation*. Retrieved from: <https://wiki.lyrasis.org/display/DSDOC7x>, 23.05.2024.
7. *DSpace*. Retrieved from: <https://dspace.lyrasis.org/>, 23.05.2024.

8. *ISO 639 – Language codes*. Retrieved from: <https://www.iso.org/iso-639-language-codes.html>, 24.06.2022.
9. Kocubiej, S., Nowakowska, M., Stąpór, P., Świetlik, P. (2022). Dedicated data management system for university didactic processes. *Scientific Papers of Silesian University of Technology – Organization and Management Series, Vol. 162*, pp. 315-337, doi: 10.29119/1641-3466.2022.162.18.
10. *KVM (Kernel-based Virtual Machine)*. Retrieved from: https://linux-kvm.org/page/Main_Page, 1.07.2024.
11. *National Data Storage: delivery of the long-term storage system elements*. Retrieved from: <https://www.psnk.pl/national-data-storage-delivery-of-the-long-term-storage-system-elements/>, 23.05.2024.
12. *NDS – National Data Storage* (in Polish: *KMD – Krajowy Magazyn Danych*). Retrieved from: <https://kmd.pionier.net.pl/>, 23.05.2024.
13. *PostgreSQL*. Retrieved from: <https://www.postgresql.org/pl/>, 1.07.2024.
14. *Proxmox VM*. Retrieved from: <https://www.proxmox.com/en/proxmox-virtual-environment/overview>, 1.07.2024.
15. *QEMU (Quick Emulator)*. Retrieved from: <https://www.qemu.org/>, 1.07.2024.
16. *Shaping the future of Shibboleth Software*. Retrieved from: <https://www.shibboleth.net/>, 23.05.2024.
17. *Solr*. Retrieved from: <https://solr.apache.org/>, 1.07.2024.
18. *Tomcat*. Retrieved from: <https://tomcat.apache.org/index.html>, 1.07.2024.
19. *W3C. Internationalization (i18n) Activity*. Retrieved from: <http://www.w3.org/International/>, 23.05.2024.