

Article

# Resolving Dilemmas Arising during Design and Implementation of Digital Repository of Heterogenic Scientific Resources

Tomasz Kubik <sup>1,\*</sup>  and Agnieszka Kwiecień <sup>2</sup> 

<sup>1</sup> Faculty of Electronics, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland

<sup>2</sup> Wrocław Centre for Networking and Supercomputing, Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland; agnieszka.kwiecien@pwr.edu.pl

\* Correspondence: tomasz.kubik@pwr.edu.pl

**Abstract:** The creation of digital repositories for archiving and disseminating scientific resources faces many challenges. These challenges relate not only to the modelling of the processes of preparing, depositing, sharing, maintaining, and curating resources. They also face the feasibility of the adopted assumptions and final implementation. Such kind of issues become particularly important in the case of processing of resources containing multimedia. The critical factor then becomes a properly designed architecture that supports efficient data processing and universal data presentation. This article aims to answer questions that may arise when approaching various designing and implementation dilemmas, such as how to handle processes in a digital repository, how to use cloud solutions in its construction, how to work with user interfaces, and how to process collected multimedia. The presented study explores their practical context based on the experiences gained during the AZON platform's implementation. This platform stores tens of thousands of scientific resources: books, articles, magazines, teaching materials, presentations, photos, 3D scans, audio and video files, databases, and many more. It serves as a running example for all presented proposals.

**Keywords:** digital repository; multimedia processing; cloud architecture; scientific resources



**Citation:** Kubik, T.; Kwiecień, A. Resolving Dilemmas Arising during Design and Implementation of Digital Repository of Heterogenic Scientific Resources. *Appl. Sci.* **2021**, *11*, 215. <https://doi.org/10.3390/app11010215>

Received: 25 November 2020

Accepted: 23 December 2020

Published: 28 December 2020

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Digital repositories (DRs), in brief, are IT systems offering tools for publishing, storing, retrieving, and sharing data and information by collecting data records along with related metadata. They are exciting subjects for conducting a variety of research. Its broad spectrum covers database management, including inference with incomplete/inconsistent datasets, knowledge modeling and discovery, querying, security issues handling, etc., as described in [1]. In particular, the research might cover models used to index, search and recommend multimedia data as presented in [2,3].

From the users' point of view, digital repositories should offer: (i) a user-friendly graphical interface enabling: easy search for records, classification and sorting of search results, content preview, retrieval of items; (ii) the ability to process data at different stages of the supported processes, for example, to automatically generate metadata when publishing records; (iii) multi-level access control of individual metadata fields to meet the needs of different audiences; (iv) the ability to cooperate with other repositories; (v) handling and providing long-term support for different types of data files; etc.

For the designers, essential parameters of the repositories are: infrastructure and hardware requirements; modularity, extensibility, and scalability of the architecture; a stack of technologies used; security mechanisms implemented; supported protocols and offered APIs (Application Program Interfaces) allowing for adding new functions and integration with other systems; etc.

The various software solution examination in terms of their potential use was only a part of the analyses carried out during the AZON (Atlas of Open Scientific Resources)

platform's construction. This platform enables the collection, processing, and sharing of open science resources in digital form. It was built in the scope of the project *Aktywna Platforma Informacyjna e-scienceplus.pl*, POPC.02.03.01-00-0010/16, aimed at increasing accessibility, improving the quality and reuse facilitating of scientific resources of such institutions as Wrocław University of Science and Technology, University School of Physical Education in Wrocław, Wrocław University of Environmental and Life Sciences, Wrocław Medical University and Systems Research Institute Polish Academy of Sciences. Thus, the AZON platform can be compared, in some sense, to the data management systems like Dryad (<https://datadryad.org/stash>), Open Science Framework (<https://osf.io>), CKAN (<https://ckan.org>), and Dataverse (<https://dataverse.org>) addressed to the community of users, enabling collaboration, resources sharing and discovery. The digital archives, like National Archives Catalog (<https://catalog.archives.gov>), Narodowe Archiwum Cyfrowe (<https://www.nac.gov.pl>), or Europeana (<https://www.europeana.eu/en>) are also comparable.

In the beginning, it was necessary to design an appropriate architecture for the AZON platform, especially that it was supposed to support the publication of 32,792 documents (assuming, for simplicity, that one document = one data record) that would require the dissemination of 124 TB of data. Moreover, according to the assumptions, increasing scientific resources' availability should be automated, with batch processing of data performed in the computing centre. The platform was supposed to support data standardization, records annotation, voice transcription, search, linking, keywords definition, and text analysis. The designed domain metadata models, dictionaries, and thesauri should support the information dissemination and search.

The plan was to design separate subsystems for handling such tasks as data collecting, processing, long-term preservation, presentation, and sharing due to the possibility of concepts separation. Many of these tasks could be addressed by the software providing the digital repository functionality. However, the existing solutions did not meet the expectations. For example, the problem of effective multimedia processing, including automatic conversion, generation of transcripts and preparation for streaming, remained untouched, and there was no support for consistent presentation of heterogeneous resources. Therefore, the design of a new platform started from scratch, and this was a challenge.

During the work, many dilemmas have been raised and solved, both on the side of modeling and implementation. This article focuses on presenting some of them, emphasizing those related to the processes running within the AZON platform, its architecture, and problems associated with the design of interfaces ensuring sufficient deposit and presentation of scientific resources, including resources with multimedia attached.

## 2. Literature Review

The logical architecture of many digital repositories consists of several parts. Usually, there is a part responsible for the business logic (platform or engine). Another one is responsible for data preservation (file systems or databases). The last one is used to interact with users within the web browser window (access application). These parts are loosely coupled and often run as independent web services. The communication between them goes through the endpoints with well-defined application programming interfaces.

In Table 1, some of the existing implementations of digital repositories are listed, including platform only and fully featured repositories. All are open-source software-based. A more detailed comparison of selected repositories is given in [4] (a bit outdated, but still valid) while custom implementations are in the focus of [5,6].

**Table 1.** Selected digital repositories delivered as software packages.

Project Name and Its Homepage	Supported Metadata Standards	Software Type and Search Features
Fedora Commons, <a href="https://duraspace.org/fedora/">https://duraspace.org/fedora/</a>	MODS, Dublin Core, QDC out of the box, arbitrary metadata sets	platform only, with API, with native Linked Data support
Islandora, <a href="https://islandora.ca">https://islandora.ca</a>	MODS, Dublin Core, PREMIS, MARCXML	repository, built on Fedora Commons, uses Apache Solr
Samvera (previously Hydra), <a href="https://samvera.org">https://samvera.org</a>	MODS, DC, DC11, EDM, FOAF, custom metadata set	bespoke solution, build on Fedora Commons, uses Apache Solr
DSpace, <a href="http://duraspace.org/dspace/">http://duraspace.org/dspace/</a>	Dublin Core, arbitrary metadata sets (but flat)	repository, uses Apache Solr
Archivematica, <a href="https://www.archivematica.org/en/">https://www.archivematica.org/en/</a>	METS, PREMIS, Dublin Core, Library of Congress BagIt	repository, based on ISO-OAIS functional model, uses Elastic Search
AtoM, <a href="https://www.accesstomemory.org/en/">https://www.accesstomemory.org/en/</a>	EAD, Dublin Core XML, MODS XML, EAC, SKOS	repository, uses Elastic Search
Invenio, <a href="https://invenio-software.org">https://invenio-software.org</a>	MARC	repository, uses Elastic Search
VIVO, <a href="https://duraspace.org/vivo/">https://duraspace.org/vivo/</a>	VIVO ontology, arbitrary metadata sets (custom ontologies)	repository, uses Apache Solr
RODA, <a href="https://www.roda-community.org">https://www.roda-community.org</a>	EAD, DC, METS, PREMIS, and others (e.g., NISO Z39.87 for digital still images)	repository, uses Apache Solr
EPrints, <a href="https://www.eprints.org">https://www.eprints.org</a>	custom metadata set	repository, uses custom indexer

In scientific or business-related fields, the term digital repositories (DRs) is often associated with institutional repositories (IRs)—understood as sets of services for the management and dissemination of digital materials created by the institution community members, including long-term preservation, access control, and distribution [7]; or the term current research information systems (CRISs)—a family of information systems used for storage and management of data about research conducted at an institution [8]. Their development can be based on a hosted solution [9] or custom implementations. Some insights on a qualitative comparison of IRs and CRISs can be found in [10].

In the cultural heritage (CH) field, digital repositories form the basis for implementing digital galleries, libraries, archives and museums, and archaeological datasets [11–13]. In addition to the possibility of cataloguing and disseminating digital versions of artefacts collected, these software systems offer specialized interfaces supporting multimedia [14]. With their help, browsing through scans of documents, archival audio/video recordings, 360 photos, or 3D models can take place not only on websites [15] but also in artificial or augmented reality. They may appear in the form of platforms running in the cloud or as software packages. Among them quite an important role is being played by collection management systems (CMSs)—software systems used to manage the whole lifecycle of preserved physical and digital materials, coordinate the undergoing tasks, control the responsibilities and access policies, etc. An example is ArchivesSpace, (<https://archivesspace.org>).

Quite close to discussed are content management systems (CMSs). They share the same abbreviation and several functions: support in describing the collected artefacts, usage monitoring, and access management. The examples of open source platforms of that sort are Mukurtu CMS (<https://mukurtu.org>), Omeka S (<https://omeka.org/s/>), and Collective Access (<https://collectiveaccess.org>). The last in this list has two main components: Providence (engine) and Pawtucket2 (front end). It could be listed in Table 1, but it was not because of its narrowed application domain. For the same reason, all the platforms mentioned so far differ from the content management systems used to build web portals for publishing user content, such as Joomla (<https://www.joomla.org>), Drupal (<https://www.drupal.org>) and others.

Another acronym that is often associated with DR is DRMS. DRMS stands for digital resource management system—a system or platform, most often aimed at managing and delivering various media assets to other systems. Offered interfaces enable the integration of DRMSs with existing enterprise architectures. DRMS often run in a

cloud, like ContentDM (<https://www.oclc.org/en/contentdm.html>) or ResourceSpace (<https://www.resourcespace.com>).

The digital repository concept sometimes extends to specialized systems used to store and process files in the cloud, like DuraCloud (<http://duraspace.org/duracloud/>), OpenStack (<https://www.openstack.org>). However, these belong to the class of webdisks, rather than to digital repositories. One might also wonder whether to assign social networking sites or commercial services, which offer media sharing, storage, and streaming functions, to the category of digital repositories. However, these portals and services have different functionality and significantly different objectives than long-term preservation and sharing of scientific resources, data conversion for better accessibility, and other aims addressed by digital repositories.

Digital repositories are also close to digital libraries, which are web sites consecrated to preserve and distribute knowledge. These sites encompass a wide range of materials, from books to representations of three-dimensional artefacts stored in the institutions' libraries. The content is either created digitally or converted from a variety of analogue sources through digitization. [16] Quite often of the materials are republished copies of originals that the end-users can consult and read without purchasing. The standard formats used are pdf and DjVu.

Digital repositories can take registers' character, i.e., systems with more demanding security policies, serving as identifier generators and reference data sources. The management of these registers may follow recommendations and official standards, for example, ISO 14721:2012 [17], ISO 20652:2006 [18], ISO 16363:2012 [19]. ISO 14721:2012 defines a reference model for an open archive information system. ISO 20652:2006 describes the relationships and interactions between the information producer and the repository. It establishes a methodology to systematize activities from the first contact between the information producer and the information repository to the transfer and information validation. ISO 16363:2012 describes the audit and certification of digital repositories.

Many software solutions discussed use external media processing and management systems to prepare assets for deposit or publication. Often such external systems operate in the cloud on a commercial basis. The Limecraft Platform (<https://www.limecraft.com>) is one of them. It is a commercial, cloud-based solution for multimedia management, offering, among the others, transcription generation, streaming, share and search functions. It is used mainly to support the production process for multimedia content providers. Kaltura (<https://corp.kaltura.com>), another commercial, and cloud-based solution, offers similar functions. Not so far is TAKTIK OZONE (<https://ozone.taktik.com/en/>), with its cloud-based media storage, processing and sharing, dynamic transcoding, metadata description, or simple video processing for, e.g., didactic classes. All three platforms are adequate for multimedia management, particularly for handling and streaming large video files. Because of specialized usage, their versatility is limited (in terms of low-level customization and adjustment). Nevertheless, each one can be an excellent element of a larger system that will benefit from their advanced functions of effective video processing.

The core mechanisms for content management in digital repositories rely on the use of metadata. However, metadata plays an essential role not only within repositories but also in their surroundings. They can: (i) improve resource search (metadata describe resources: the nature of data sets, service capabilities or procedures); (ii) help to assess resources before they are used (on their base one can determine whether the searched resources meet requirements); (iii) inform how the resource is used (the provision data and usage statistics are often metadata fields); and (iv) provide a platform for distributed information integration (metadata standards ensure a common understanding of the information exchanged). Thus, a well-developed metadata model and a standardized way of publishing resources (in terms of APIs and protocols used) allow for information processing automation. On such a basis, it is possible to create tools retrieving information from dispersed sources (as Image crawler [20], which allows one to search for resources using the OAI-PMH protocol).

Currently, there are many metadata standards in everyday use. Apart from schemas developed by the DCMI (Dublin Core Metadata Initiative), FGDC (Federal Geographic Data Committee), OGC (Open Geospatial Consortium) and ISO TC 211 (International Organization for Standardization, Technical Committee 211—Geographic information/Geomatics), and Network Development and MARC Standards Office (MODS); many other also exist. Their list includes, but is not limited, to METS (Metadata Encoding and Transmission Standard), MIX (NISO Technical Metadata for Digital Still Images), PREMIS (The PREMIS Data Dictionary for Preservation Metadata), ALTO (Analyzed Layout and Text Object). In addition, no one can neglect the well-known vocabularies, such as schema.org, Dublin-Core, Google Scholar, microformats, and annotation techniques like RDFa. For a more comprehensive comparison of the most popular metadata standards considered in the perspective of modelling, please refer to [21].

### 3. Materials and Methods

The construction of a digital repository is a task combining elements related to the IT project management (ending with a software system implementation) and activities connected with the acquisition and publication of digital resources (daily use of the implemented system). It was similar in the described case. In both areas, generally accepted best practices have been applied. However, the manuscript misses details of adopted methodologies for project management and software development life cycle due to the editorial limitation.

Originally the concept of the AZON platform has been described in various perspectives: information perspective (focus on kinds of information handled), functional perspective (focus on the presentation of the provided functions), implementation perspective (focus on the logical architecture of the system, including its modules, interfaces, etc.), and technology perspective (focus on the technology used). Some parts of these descriptions, mainly related to the functional requirements and the information model, are presented below, followed by details of the implemented infrastructure and processing.

#### 3.1. Domain-Specific Vocabulary

A domain-specific dictionary developed ensured a common understanding of the problems encountered. Some of its essential entries are listed below.

- accessibility—a feature of designed products, devices, services and environments related to their adaptation to the needs of users with diverse abilities, and manifested either directly or through the use of assistive technologies;
- collective data records view—a view in which more than one data record is rendered, which may involve showing a specific portion of representative contents of data records, with the use of filters and pagination;
- data record—a set of metadata, data, and licenses forming a single overall object stored in the repository;
- data record rendering—presenting the contents of the data record on a computer screen;
- data record validation—the process of assessing the accuracy of data records according to non-measurable criteria (compliance with expectations);
- data record verification—the process of evaluating the correctness of data records according to measurable criteria (compliance with schema);
- depositing—the process of delivering data records to a repository (through the available interfaces) and saving those records in the repository, which may be accompanied by verification and validation of these data records;
- metadata—a structured set of named attributes with associated values used to describe data;
- publishing—the process of making data records available on websites and through APIs or GUI;
- search—the process of formulating and submitting queries to the repository and generating responses.

The dictionary helped in discussions with end-users during the analysis, testing, and AZON platform deploying phase.

### 3.2. Requirements Analysis

Below are the general and specific requirements for the AZON platform's functions (bolded parts are directly related to multimedia processing).

- **Depositing**—upon deposit, quality control of data records should be concerned. Thus, the platform should support the editing and review of data records by different users. Moreover, depositing should be possible through GUI and API, in the context of the depositor or his representative. Large data records that cannot be loaded into the repository with the HTTP protocol should also be supported.
- **Publishing**—published data records should appear on a website in collective and individual views, in both human and machine-readable form. The rendering should be consistent for all supported types of data records while ensuring their high availability. The data centre should activate batch processing of data (including, among others, standardization, annotation, generation of voice transcription, indexing for search purposes, linking, and determination of keywords). Records' use should be monitored, and their correction and curation should be possible.
- **Searching and browsing**—published data records should be searchable in various ways (simple, advanced, faceted, by browsing, and with filtering). This requirement involves indexing their attributes and supporting the full-text search. The parameterization of search queries should be supported by utilizing dictionaries. Dictionary management (creating, adding, deleting, modifying terms) and accessing them (e.g., finding related terms in advanced search) should be possible through GUI and API.
- **Securing**—three pillars of security must be provided by the infrastructure: authentication, authorization, and monitoring. Moreover, the functions offered should correspond to the adopted security policies and formal procedures (including audits, issues handling, etc.).
- **Maintaining**—the system should ensure secure preservation of deposited resources and their long-term maintenance. The former function includes checking for consistency and physical integrity, backing up, storing several copies in different locations, etc. The latter involves checking for up-to-date links and migrations triggered by introducing new data types or upgrades of system hardware and software.

The original list of requirements was much longer and detailed. It also included the set of technical requirements for audio/video processing that can be shortened as follows:

- **Audio/video handling**—uploading large video or audio should be supported, together with the assignment of unique IDs and publicly accessible links (so audio/video players can use that).
- **Transcripts/subtitles handling**—uploading multiple transcription text files, subtitles and associate them with the corresponding video should be possible.
- **Streaming**—video streaming and rendering with subtitling in the player on zasoby-nauki.com is expected.
- **Video processing**—adapting the stream's quality to the quality of the connection (or choosing the quality from offered options in the player) is required.
- **Video type supporting**—the solution must assure support for HLS (and/or MPEG-DASH).

### 3.3. Information Model

The design of the information model for the data records required the consideration of many factors. The model had to take into account the needs of the users while meeting the assumptions concerning the construction of the metadata system (based on the relational-graphic model, taking into account the requirements of the concept of linked open data, LOD) and the limitations related to the preparation and provision of associated data. By assumption, every data record consists of:

- metadata—a set of mandatory (e.g., name, description, author, keywords) and optional elements (e.g., links to related resources). Some of these elements are of a simple type (can be represented by literals), while others are references to existing objects using URIs. Such URIs should make it possible to create links to reference registers (internal or external).
- data—an optional part, which might be represented by one or many files.
- license—information about license defined at a record level.

Additionally, every data record must have a unique identifier and a publicly accessible URL once published.

The work undertaken was divided into stages. In the beginning, based on users' interviews, the types of data records were identified and assigned into the groups as follows (groups' names are italicized, the types requiring multimedia processing are bolded):

- *Data*: Catalogue; Chemical analysis; Dataset, database; **Dendrological collection**; Flow (NetFlow) from a firewall device; **Herbarium sheet**; **Histological preparation—human**; **Histopathological preparation—animal**; Source code; Threat log;
- *Documents*: Article, chapter; Book; Journal; Legislation; Legislation collection; Map; Map collection, atlas; Synopsis; Thesis;
- *Materials*: Archival material; Educational material; Manuscript; Note; Other documents;
- *Multimedia*: **3D**, **foto360**; **Artistic, architectural work**; **Audio**; **Photo**; **Presentation**; **Video**;
- *Portfolio*: Expert; Research equipment; Research laboratory; Research offer.

Every data type was then characterized by metadata attributes and possible types of data constituting its payload. Such descriptions were mapped into the relational schema and graph model compliant with specially developed ontology. The mapping to other standards (as schema.org and Dublin Core) was done as well. Finally, the implementation of tools for depositing records with attached data has begun.

#### 4. Results

During the construction of the AZON platform, the available infrastructure was used and expanded. The launch of the platform made it possible to deposit and publish a significant number of data records. However, to achieve this, many problems had to be resolved, like: how to supply the repository with data records, how to process these records, and how to present the data records' content. The following sections will discuss them in detail.

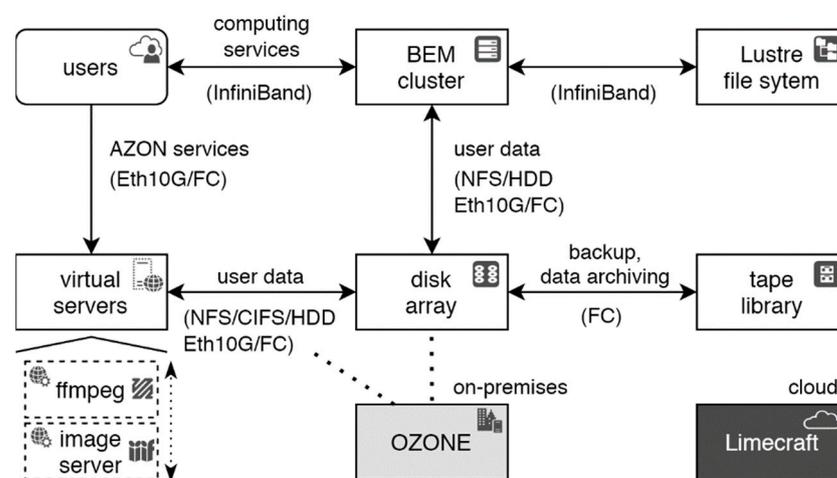
##### 4.1. Infrastructure

AZON platform uses multimedia infrastructure offered by the Wrocław University of Science and Technology, Wrocław Centre for Networking and Supercomputing (which is a base for local services implementation). It runs on-premise services along with services offered by an external cloud. It can offer a presentation layer built on the OTT (over-the-top) platform features, enabling media content delivery to users equipped with any device running a compatible Internet browser.

In Figure 1 the used infrastructure is outlined. The meaning of the depicted blocks is as follows:

- *virtual servers*—a virtual servers farm managed by VMware virtualizer;
  - *ffmpeg*—web service deployed on one of the virtual servers, used mainly for video transcoding, build on the base of ffmpeg tool;
  - *image server*—web service deployed on one of the virtual servers, used for user-friendly multimedia presentation, build on iipsrv (IIPImage—High Resolution Streaming Image Server);
- *disk array*—a system built from multiple disk drives and a cache memory offering storage capacity for numerous hosts;

- *tape library*—a high-capacity storage system, also called a tape robot, preserving data on many tape cartridges;
- *Lustre file system*—an efficient file system, optimized for access (write, read) to temporary data generated during calculations on the cluster, with a dedicated communication network—InfiniBand;
- *BEM cluster*—a supercomputer, with 22 thousand computing cores and a total output of 860 TFLOPS. The cluster has 720 24-core computing nodes (Intel Xeon E5-2670 v3 2.3 GHz, Haswell) and 192 28-core computing nodes (Intel Xeon E5-2697 v3 2.6 GHz, Haswell). It has 74.6 TB of RAM (64, 128, or 512 GB per node). The exchange of data files between BEM and its clients takes place through the enabled disk shares.
- *OZONE*—an on-premises platform for multimedia processing (deployed in the existing infrastructure);
- *Limecraft*—a cloud-based platform for multimedia processing (deployed in the external infrastructure).



**Figure 1.** An outline of the infrastructure on which the AZON platform has been deployed. The Wrocław University of Science and Technology, Wrocław Centre for Networking and Supercomputing maintains the core infrastructure characterized by calculation power: 860 TFlops, operating memory: 74.75 TB, disk space: 2.7 PB, tape memory: 5.6 PB. AZON platform uses part of it.

The blocks with a white fill represent managed parts of the infrastructure. The greyed and darkened blocks represent unmanaged parts of the infrastructure serving as bases for external services deployment. Such external services work, respectively, on-premise or on an external cloud. Some of the blocks have dashed contours (*ffmpeg*, *image server*). Such blocks are not really infrastructure's parts but represent services deployed on virtual servers.

The *virtual servers* farm, annotated with an icon representing “virtual web servers”, acts as an accessible infrastructure. Such a solution enables the migration of virtual machines from the original environment to other locations (to different physical devices or blade shelves) ensuring redundancy.

The monitoring of *virtual servers* farm performs at the following levels: power supply and air-conditioning in the server room, status and parameters of the blade shelves, status and parameters of individual blade servers, status and parameters of the operating system of the virtual machine host, status and parameters of the operating system inside the virtual machines (in particular the status and parameters of AZON platform services).

*Disk array* serves mainly as a data storage system. Its shares are made available through various protocols on the cluster and, through the virtualizer, to virtual machines performing multiple AZON platforms' functions. In this way, the number of copies can be minimized, and the data transfer over the network can be limited. From there, archiving takes place, following established policy, using a dedicated FibreChannel (FC) network.

Operating systems of virtual machines thus work with the use of disks shared from the disk virtualizer. It allows more flexible data storage space management than in ordinary block matrix use cases. All data files are made available to virtual machines from the NAS head using depicted file protocols. Any type of data can be backed up using a backup system. It is also possible to archive data rarely accessed. The backup and archival take place in a *tape library*. Such a solution allows significant savings in hardware purchase and operation compared to others.

*Lustre file system* is mainly used to perform computational tasks (e.g., language analysis of the deposited resource, data processing on GPU, Matlab scripts execution, conducting in silico experiments with runtime scripts, input files, and results deposited in the AZON platform).

The designed architecture proved to be efficient in audio/video materials storage and sharing and adjusting transmission parameters. The AZON platform communicates with the Media Asset Management components through a dedicated SOAP API. However, shortly after the first successful tests, some problems appeared affecting the approach to media handling. These concentrated mainly on fixed and relatively high costs of using external cloud-based services, which offered broader functionalities than required. Looking for more economical alternatives, operating within the AZON platform, a service boxing ffmpeg tool for video transcoding has been developed and implemented.

#### 4.2. Logical Architecture

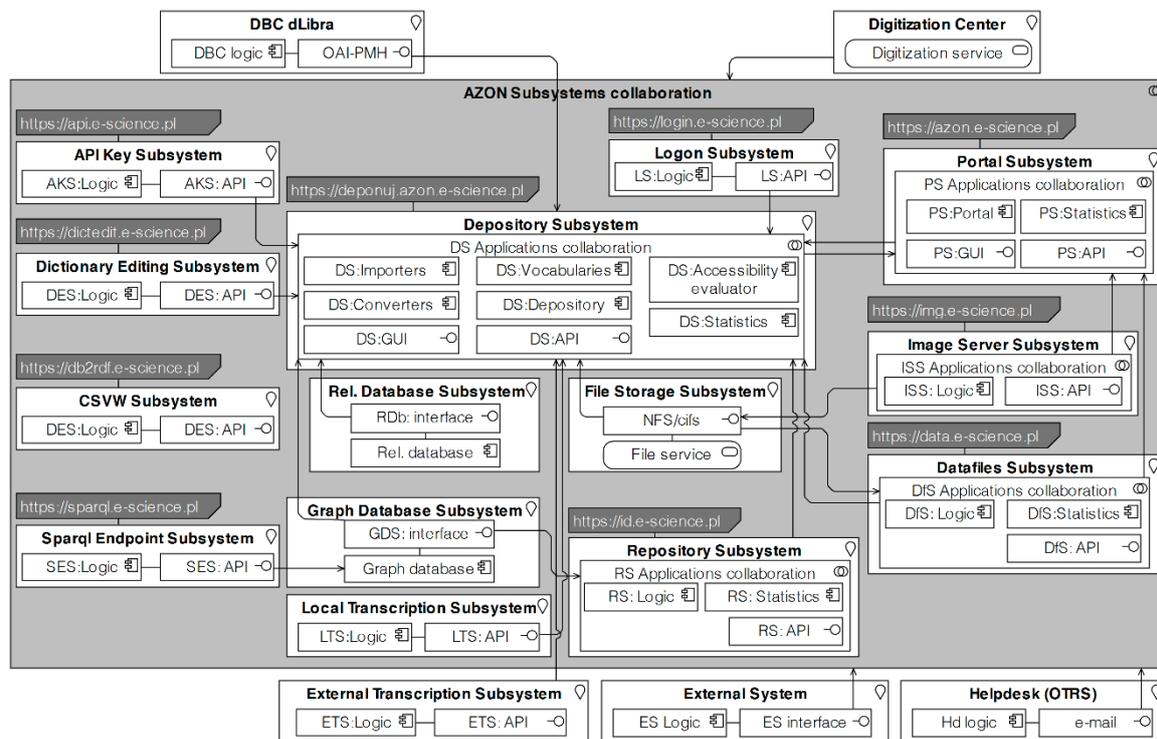
Figure 2 shows the structure of the AZON platform. The following elements maintained in the same infrastructure appear in the *AZON Subsystems Collaboration* block:

- *Depository subsystem* (digital repository engine),
- *Portal subsystem* (digital repository front-end),
- *Database subsystem, Graph Database Subsystem, File Storage Subsystem* (parts of the AZON platform used for data storage implementation: relational database, graphs database, file system)
- *Local Transcription Subsystem* (part of the AZON platform responsible for multimedia processing),
- *Image Server Subsystem, Datafiles Subsystem* (web services accountable for content delivering to *Portal Subsystem* or to be downloaded by the users)
- *Repository Subsystem, Sparql Endpoint Subsystem* (web services used to publish metadata in a semantic form and according to LOD principles)
- *Logon Subsystem, API Key Subsystem* (web services responsible for handling user credentials and assure system security). Logging into the AZON platform and accessing its services works with the SAML mechanism, realized through CAS. The CAS server retrieves authentication data from indicated LDAP and Active Directory servers. For depositors from external universities or institutions with no authentication source (LDAP, AC), registration in the platform is obligatory. The *Logon Subsystem* is responsible for handling such tasks. However, access for reading is open for everyone with some exceptions enforced by security rules. To reach API offered by AZON components, the users must apply individually for access tokens through *API Key Subsystem*. This subsystem incorporates API Umbrella (<https://apiumbrella.io>) working in the background.
- *Dictionary Editing Subsystem* (web services used to view and modify dictionaries used in *Depository Subsystem* or manage custom dictionaries).
- *CSVW Subsystem* (web service enabling conversion of tabular data in CSV format to RDF format using the CSVW standard (<https://www.w3.org/TR/tabular-data-model/>)).

The elements maintained in the external infrastructures (working on an external cloud) appear outside the *AZON Subsystems Collaboration* block:

- *External Transcription Subsystem* (service responsible for multimedia processing, like OZONE system used for transcription).

- *DBC dLibra* (element used to integrate AZON depository with Lower Silesian Digital Library system).
- *Helpdesk (OTRS)* (element responsible for handling users' help requests and comments).
- *External System* (any external system that uses AZON API).
- *Digitization Center* (services running within institutions involved in the project, aimed at resources digitization).



**Figure 2.** An outline of the logical architecture of the AZON platform.

One may wonder about associations depicted in the diagram or question the subsystems placements. There should be more dependencies among subsystems. Unfortunately, it was not possible to show them all. The diagram only presents an “architectural outline” and does not fully reflect it. However, the URLs associated with the main components operating in several locations are real URLs.

#### 4.3. Business Processes

The AZON platform is a set of cooperating elements, performing tasks according to the modelled processes. One of the implemented is the process of depositing and publishing data records (DPDR). Figure 3 highlights the states a data record may reach in it.

An important aspect to consider when building a digital repository is the quality assurance of the resources deposited. In the case of resources already published, their evaluation can be implemented in various ways, ranging from providing mechanisms for ranking resources through the like/dislike options made available to users, to allowing for broader pros and cons commentary. There are also formal methodologies that involve evaluating resources using a validated rubric by trained teachers [22]. However, the quality of the resources deposited in the AZON platform is ensured by the editorial/review tasks run within the DPDR process. Thanks to them, records reviewed by specialists are subject to publication. Moreover, there is an opportunity to comment on already published records (*Helpdesk* handles such comments).

A user depositing a data record may attach any number of data files to it. Two possible options are: (i) direct upload, using the HTTPS protocol, and (ii) indirect upload, using the webdisk service, operating within the same infrastructure, supporting CIFS, SFTP, HTTPS protocols. The existing web servers' limitations enforced the distinction between these two cases. HTTP specification itself does not impose any specific size limit for posts. However, the receivers of the request (webservers) might not be ready to process large payloads. The limit is usually set at 2 GB level, which is not enough for transferring large movies.

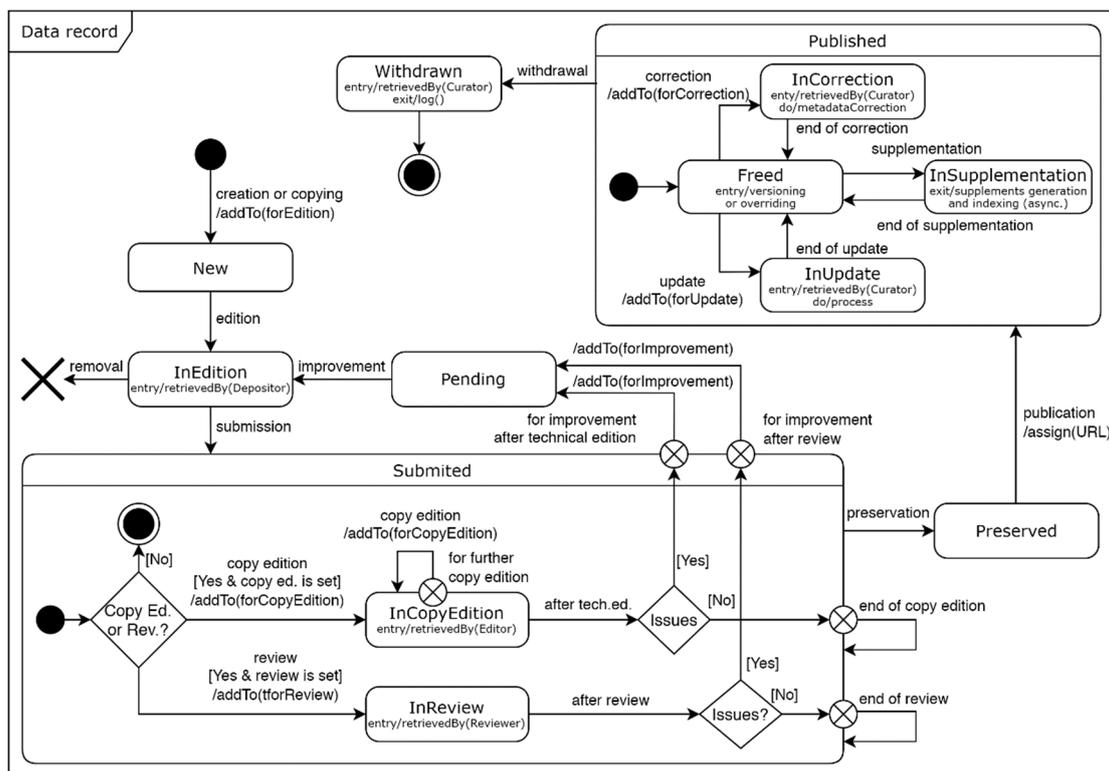
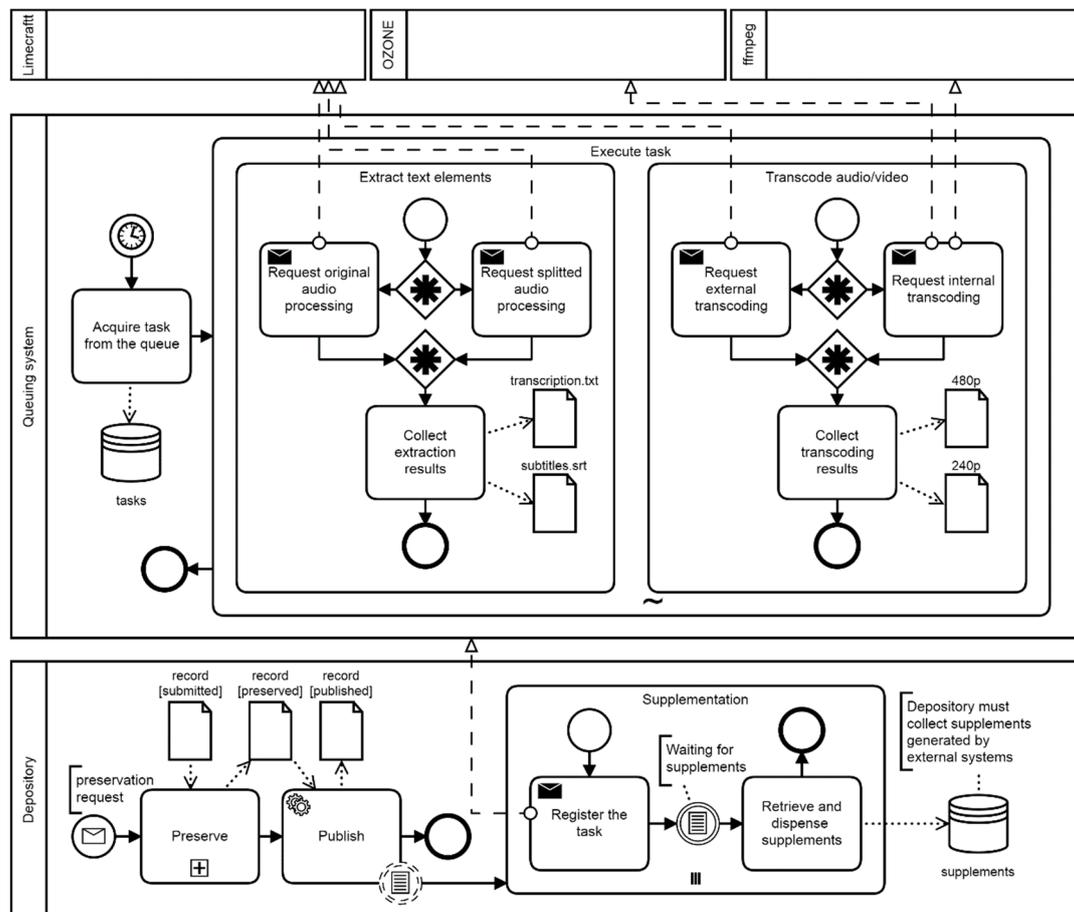


Figure 3. Diagram of states of data records deposited and published in AZON platform.

The automatic multimedia processing (processing of attached files to the data record) occurs when the data record reaches the *Published* state and its *InSupplementation* substate. The *Supplementation* block in Figure 4 represents it. However, the necessity of multimedia processing may appear at other states as well (data files have to be recorded, named, encoded and uploaded, converted to long supported formats, optionally anonymized, prepared for presentation and preview, etc.). These steps are intentionally not shown in the figure to preserve its clarity.

As depicted in Figure 4, several instances of *Supplementation* might be executed in parallel by the *Depository Subsystem* (AZON is a multi-access platform, and several users can submit their data records for publication simultaneously). According to the *Supplementation* block's internal control flow, processing tasks are registered in the *Queuing system*, and results are retrieved and dispensed once specific business rules are met. The *Queuing system* does not process any multimedia by itself during queued task execution. Instead, it delegates processing to the external systems by sending proper requests (see *Execute task* block). Then it collects results.



**Figure 4.** Parts of the processing flow related to the generation of supplements in the AZON platform during depositing and publication of a data record containing multimedia.

The ad-hoc subtasks *Extract text elements* and *Transcode audio/video* require some explanations. The *Extract text elements* subtask relies on the generation of transcript and subtitles in an appropriate format (STR, VTT, STL) for selected audio/video input through API calls to the Limecraft cloud service. Two scenarios are possible:

- simple, by sending original audio to the *Limecraft* cloud for processing, and downloading processing results. The processing flow starts with a POST request parameterized with file name and size. In the response, *Limecraft* returns URL API for file upload. After receiving the information about the upload end (send through API), *Limecraft* starts the transcript generation process and makes their results (transcript, subtitles) available in selected formats.
- complex, by splitting audio from the original audio/video input and then sending that audio to the *Limecraft* cloud to process and download the results. This scenario brings benefits in terms of reducing the amount of data stored in *Limecraft* cloud (this reduces costs) and accelerating data transfer to/from the cloud (this reduces total processing time).

The generated transcripts and subtitles can be further processed (which is no longer shown in the process diagram). On their basis, extended subtitles are created. These are subtitles supplemented with commentary for deaf persons, usually including descriptions of sounds other than speech (e.g., the audience claps, the dog barks in the background, birds singing). The project team members edit these comments. In most cases, their work is limited to making minor corrections to automatically generated texts.

The *Transcode audio/video* subtask involves converting original video files to a format that can be streamed by a video sharing system. Such conversion can be done dynamically (on the fly, without generating any new data) or statically (asynchronously, with the generation of additional data). This kind of processing can be realized in two ways:

- by using external, on-premise service, which means uploading the data to *OZONE* platform configured for automatic storage and streaming of video, audio, subtitles as required (handled video resolutions: 240p/360p/480p/720p/1080p);
- by using internal, developed service, which means uploading data to a dedicated server, starting the processing through adequately configured and used *ffmpeg* tool, obtaining results of required resolution and storing and streaming video, audio, subtitles as required (handled video resolutions: 480p/720p/1080p);

The consequence of using the *ffmpeg* tool is generating a significant number of files. On average, 183 files were generated in processing tests conducted on the 7 min film. Such cost has to be incurred to simplify the procedure of video streaming (its implementation boils down to loading output files prepared in advance). Transcoding takes place up to 3 resolutions 480p/720p/1080p, with the resolution of 240p/360p omitted (support for resolutions smaller than offered by the old SDTV standard was considered unreasonable). When tuning the appropriate coding parameters, the processing on the 16-core server takes about 2/3 of the source material's duration. So, for a 7 min film, this time is about 4:40 min.

Tests with the original 1.6 GB video showed that the total volume of generated files is about 490 MB. What is essential, the implemented security procedures do not impose the need to back up just generated files. If necessary, they can always be generated anew from the original resource stored in the repository.

The presented solution is less flexible than the one based on the *OZONE* platform. For example, the *ffmpeg* tool produces appropriate *m3u* playlists assisting generated files, consumed by video players. If there is a change in the sharing of video streams, then all playlists must be updated. However, the advantage of this solution is greater stability and faster configuration. Furthermore, no attempt has been made to build any new processing scheme as in [23] nor propose or implement specialized video processing algorithms as those described in [24]. Instead, the focus was on making the best use of existing tools and achieving project goals.

The processed media on the *AZON* platform also include images deposited in various formats (jpg, png, tiff, gif). Their processing consists mainly of conversions conforming to the requirements of *iipsrv* (<https://github.com/ruven/iipsrv>) based service used and deployed in the existing infrastructure as *Image Server Subsystem*. The service supports *IIIF* Image API (see <https://iiif.io/api/index.html>) and offers specially prepared images through the endpoints consumable by compatible viewers. Additionally, the thumbnails of images are generated. Due to editorial constraints, the presentation of the underlying processes in diagrams was omitted.

Scans of text documents were also processed to make them more accessible. This enforced supplements' generation in the form of pdf files containing the text layer. Most of the processing took place automatically. It consisted of character recognition using commercial software. In some cases, the documents were also processed manually, in the *Tyfloo* Computer Lab (Laboratorium Tyfloinformatyki), by adding structure information.

The list of supported media also includes 3D scans. They were loaded to the repository indirectly, through folders on the web disk intended for the files with point clouds generated directly from the scanner (files in *.wrp*, *.stl*, and *.obj* format) or copied by users (files in *.pts*, *.ptx*, *.txt* format). The original files were supplemented with other files necessary for efficient rendering of their content. Thus, the thumbnails and simplified versions of objects were generated (while avoiding excessive deformations). This task was done with the aid of, respectively, *Blender* (<https://www.blender.org>) and *Meshlab* (<https://www.meshlab.net>) tools. The choice of *Blender* for generating thumbnails resulted from the better results obtained (under the assumption that the size of the output file size is not limited and large RAM resources are available). *MeshLab*, on the other hand, gave better results when

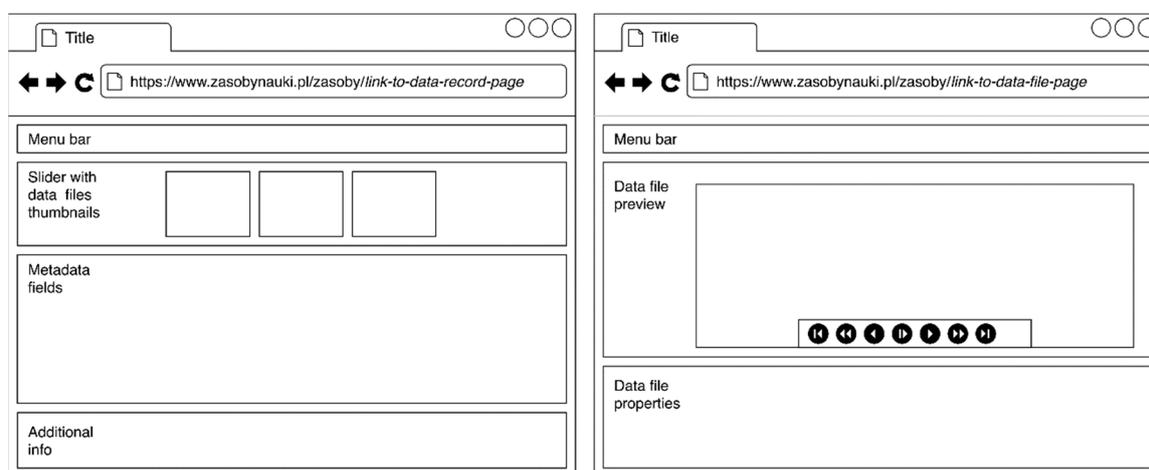
reducing the model for viewing purposes and allowed to process larger files thanks to the lower requirements for system resources.

For both tools, the generation of a new, reduced model depends mainly on the input file's size, the reduction method used, and its parameters. For example, when using Mesh-Lab, once the processing exceeded 1.5h timeout, the process ends without any generated supplements (the user only had to download the original file).

The general processing flow in the case of 360 images (.jpg, .psd) was similar. If a file was not processed and its size did not allow displaying in a web browser window, the user was left with the option to download the original version of the file. The processing methods used are comparable with those described in [25].

#### 4.4. Online Presentation

All the deposited and published data records are made available through a portal running at <https://zasobynauki.pl> (or its alias <https://azon.e-science.pl>). The portal offers unified views to present various data records and related data files (see Figure 5).



**Figure 5.** Mockups of user interfaces offered in the Portal subsystem: data records view (on the left), and data files view (on the right).

The image server's client aids the task of rendering large images in data files view. That client offers basic operation as zoom or scale. The assignment of URLs to viewed images followed a specific strategy. According to the IIIF Image API any URL for requesting an image must conform to the following template:  $\{\text{scheme}\}://\{\text{server}\}/\{\text{prefix}\}/\{\text{identifier}\}/\{\text{region}\}/\{\text{size}\}/\{\text{rotation}\}/\{\text{quality}\}.\{\text{format}\}$ .

However, every published data record in the AZON platform gets its own URL: [https://img.e-science.pl/entries/{record\\_id}/](https://img.e-science.pl/entries/{record_id}/).

Following SEO (*Search Engine Optimization*) best practices this is handled with redirects to URL embedding words coming from the data record's title: [https://www.zasobynauki.pl/zasoby/{words\\_of\\_title\\_record\\_id}/](https://www.zasobynauki.pl/zasoby/{words_of_title_record_id}/).

Furthermore, every data file (in particular—image file) attached to the record is accessible its original form under the URL: [https://img.e-science.pl/entries/{record\\_id}/file/{file\\_id}/](https://img.e-science.pl/entries/{record_id}/file/{file_id}/).

Therefore images served by image server running within the AZON platform are accessible under the URLs matching the following scheme: [https://img.e-science.pl/iiif/{record\\_id}/{file\\_id}/region/{size}/rotation/{quality}.format](https://img.e-science.pl/iiif/{record_id}/{file_id}/region/{size}/rotation/{quality}.format).

Where *record\_id*—is the identifier of the published data record to which the image was attached in its original form (during the deposit), and *file\_id*—is the file identifier of the mentioned image file. The remaining parts or URL matches IIIF Image API specification. Consequently, information about the properties of a given image, e.g., tiles, possible

enlargements, resolution, can be obtained as a metadata file in JSON format from the address of the form: [https://img.e-science.pl/iiif/{record\\_id}/{image\\_file\\_id}/info.json](https://img.e-science.pl/iiif/{record_id}/{image_file_id}/info.json).

If an image file has been deposited and no tiles have been generated for the image server yet, only a small thumbnail will be visible in the data record view. However, in the attachment view, an attempt will be made to display this image based on its origin. In long-lasting image loading, the user has to download the image manually. By definition, huge image files are not rendered at all.

The page for video rendering consists of two main parts: a movie player and a transcript viewer. The player receives a link to the m3u playlist including information about available files and their parameters. Thus, it may access and deliver video files with the appropriate resolution, automatic subtitles, extended subtitles. The viewer displays automatically generated transcript next to the movie, with the possibility of jumping from the indicated text to the video fragment. Both OZONE and ffmpeg generate m3u lists. The lists might include several media sources. The player takes one that is available in the order given. The logic hidden behind video player covers three cases:

1. OZONE platform provides video (in HLS or MPEG-DASH format);
2. AZON streaming service (HLS format) provides video when such video is not present in the OZONE platform or OZONE service is unavailable (because of failure or other reasons);
3. if the streaming service cannot deliver video, the original data files are fed from the file system via the data.e-science.pl website (in MP4, MOV, or another format). For example, this may happen if the request for a video is issued before the supplementation phase.

The third case fulfils the prerequisites set for the digital repository—maintaining the availability of the original file for download regardless of the availability of its streaming function is required.

The AZON platform uses Three.js (3D library with a default WebGL renderer, <https://github.com/mrdoob/three.js>) and JSModeler (framework to create and visualize 3D models, <https://github.com/kovacsv/JSModeler>) to render 3D scans in a data file view. The Photo 360 presentation was implemented using Marzipano (360° media viewer for the modern web, <https://www.marzipano.net>).

#### 4.5. AZON in Numbers

The adopted solutions allowed the dissemination of 44,278 data records with 391,926 original, unprocessed data files. These data files occupy 172.29 TB, and 11,197 of them represent multimedia (not including a large number of scanned documents: big size, high-quality tiff images). The size of serialized metadata for all deposited data records is 221.70 MB. These numbers were valid on 18 November 2020 (and continuously increase). The statistic below gives more information (again the types of data records holding multimedia by default are bolded, the numbers in parentheses indicate the number of data records).

- *Data* (total: 19,381): Catalogue (0); Chemical analysis (763); Dataset, database (13,582); **Dendrological collection (356)**; Flow (NetFlow) from a firewall device (886); **Herbarium sheet (1018)**; **Histological preparation—human (1277)**; **Histopathological preparation—animal**; Source code (65); Threat log (1224);
- *Documents* (total: 14,619): Article, chapter (2682); Book (1933); Journal (9475); Legislation (1); Legislation collection (0); Map (1); Map collection, atlas (0); Synopsis (154); Thesis;
- *Materials*: Archival material (0); Educational material (408); Manuscript (0); Note (1); Other document (965);
- *Multimedia* (total 8327): **3D, foto360 (1953)**; **Artistic, architectural work (245)**; **Audio (418)**; **Photo (1956)**; **Presentation (107)**; **Video (3648)**;
- *Portfolio* (577): Expert (176); Research equipment (241); Research laboratory (45); Research offer (115);

The details are as follows. Thumbnails of resources generated automatically occupy around 7.83 GB. Audio/video uploaded and processed in OZONE take 1.27 TB, and processed with ffmpeg: 70.67 GB. Automatically generated subtitles (VTT, SRT): 38.47 MB and transcripts (MD): 15.51 MB (the transcription process took 656 h). The additional volume of images processed for better rendering: 913.13 GB; and foto 360: 71.00 GB. The books conversion results into mobi format: 12.00 GB, and into epub format: 7.20 GB. The total number of downloads of files and metadata views reached 4,577,301 level. Finally, 100% of the deposited and published data records can be rated with 4 stars and 99% with 5 stars according to the 5-star rating system (<https://5stardata.info/en/>).

## 5. Discussion

When estimating the resources required and investing in infrastructure, designers of digital repositories must anticipate all data processing scenarios and their effects. However, this estimation can sometimes be challenging. For example, in our case, it turned out that the information model identified during the preliminary analysis needs some extensions. Every processed multimedia file had to be associated with the results of this processing. The files generated for rendering purposes (images in different resolutions required by image server, thumbnails used as previews of data files, audio/video files needed for streaming), and for increasing availability of the deposited resources (transcripts, subtitles, extended subtitles for video, pdf documents with text layer resulting from running OCR tools on scans) require proper handling.

It was not easy to decide whether to use existing digital repositories that were already available and used by others or to start a new implementation. The number of factors to consider included a price, a type of license, a possibility and easiness of modifying the code, the size and reliability of the community, the data and metadata model provided, offered integration paths with other software, supported authentication and authorization methods, the performance (including handling of big files and a significant number of files), enhanced search functions, a possibility and way of presentation of different types of data, semantic web functions provided and graph database support, open formats supported, Polish law requirements and recommendations for open data repositories, WCAG compliance and other accessibility aspects.

An essential step for the project members was to check the reusability of infrastructure and software in terms of money and knowledge already invested. The main discovery of this stage was that there is a set of excellent software products on the market, but most of them are tailored to specific types of data or use cases, and none provides all the desired features out of the box. The biggest obstacles were a lack of possibility to implement the required data model and difficulties with software solutions extensibility. It was challenging to find an implementation that fully supports expected semantic web enablement (SPARQL endpoint, dereferencing through content negotiation, etc.). From various candidates, Vivo offered the closest metadata model to the required one. The ontology used in it contains many classes and properties relevant to the academic community. However, due to deposited resources' heterogeneity and specific presentation requirements, it had to be abandoned. The further experiments also showed that mapping a custom metadata model using schema.org vocabulary brings many benefits in resource discovery (google search engine can index published records more effectively). These were the primary reasons to start a new software project. The other reasons came from the various technical obligations (for example, caused by the need for integration with the services built in the scope of e-science.pl project).

Side effects of the project are the experiences gained during implementation. This manuscript hides them in-between lines. For example, the statistics collected should help estimate the requirements for similar projects and the following recommendations developed during the project implementation might be helpful.

For safety reasons, an excess of hardware and software is recommended. With correctly designed architecture it may protect from Single Point of Failure, SPoF. In solving

similar problems, disk arrays, redundant SAN (Storage Area Network), FC (Fibre Channel) and redundant network equipment (switches, hubs, etc.) will be useful. However, the task of infrastructure maintenance relied not only on hardware related issues. Software issues, in particular related to the dependencies arising from the security mechanisms implemented, are also important. They enforce the need to obtain and update certificates or maintain domain names. Thus, the associated costs and procedures should find their place in the long-term implementation plan.

Another important aspect is the costs of the licensing of the platform components (cloud services or purchased software). Building multimedia processing infrastructure based on commercial clouds is not cheap and requires careful customization. In the described approach, the generation of transcripts is commissioned by Limecraft or carried out by ffmpeg. Next Google Speech to Text caught attention in the search for alternatives that would give better results. An attempt was made to test its API (applying for Speech-to-Text On-Prem model is also possible, but must be negotiated individually with the service provider). The use of Google Speech to Text services can be similar to the described Limecraft use case. One may send for processing only audio tracks extracted from a video. However, such processing enforces the most expensive licensing model. Furthermore, the lack of consent to the use of the uploaded recordings for teaching google models blocks a possible discount. Since every 15 s is billed separately for each audio track, converting multitrack audio to mono before sending is considerable. However, the transcription of recordings longer than a few minutes requires audio files uploading to storage in google cloud because the unique filehandles offered by this cloud are needed. The result is a JSON file with a transcript in which a timestamp and certainty accompany each word. Such output still needs to be converted to the target format (MD, SRT, VTT), using some logic of combining words into subtitles. That logic can be implemented based on open-source libraries supporting selected subtitle formats. However, using cloud-based storage also must be done according to the offered business plan.

## 6. Conclusions

Creation of the AZON platform provided institutions with a tool and procedures to make multimedia and other resources accessible online. The platform offers several ways to access and explore data records, including online watching, listening or reading the content derived from its original form. The AZON platform offers: the presentation front-end (where users can view, search and retrieve various records, <https://zasobynauki.pl>), the management front-end (where users can manage records according to their privileges and involvement in implemented workflows, <https://deponuj.azon.e-science.pl>), the semantic web interface (manifested by the direct access to the semantic records through the content negotiation, and by the SPARQL query tool, with built-in query editor offering predefined query samples, and the SPARQL endpoint, <https://sparql.e-science.pl>).

Thanks to the AZON ontology, metadata and tools, it is possible to explore the content semantically and search through the different data records. The designed architecture supports efficient data processing and online presentation and makes it easy to adapt to new or evolving processing tools and software solutions.

**Author Contributions:** Conceptualisation, T.K.; methodology, T.K.; writing—original draft preparation, T.K.; writing—review and editing, A.K.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was partly funded by Digital Poland Project Centre (CPPC) grant number POPC.02.03.01-00-0010/16-00. The APC was funded by Wroclaw University of Science and Technology.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** This study discusses issues related to the development and deployment of the AZON platform. It does not concern the contents of data records deposited by the platform's users. However, all these data records are made available in the spirit of the idea of open data. The platform includes both resources that can be used without restrictions (public domain), and those

that are shared under certain conditions (e.g. author's attribution, no modification). They can be searched, viewed, and downloaded from the portal running at <https://zasobynauki.pl>. They can be printed, cited, and redistributed for the most part, as long as the use complies with the licenses' terms.

**Acknowledgments:** The authors wish to thank: Wrocław University of Science and Technology for funding statutory activity, and for funding acquisition and conducting of AZON projects; Wrocław Centre for Networking and Supercomputing (WCSS) for providing resources; WCSS Development Team: Mariusz Uchroński, Adam Włodarczyk, Marek Rybak, Dobrosław Kowalski, Bartosz Karpiński, Miłosz Białczak, Maciej Olejnik, Stefan Piróg, Marcin Stajno, Monika Długosz and Mateusz Bolek for participation in design and development of the AZON platform.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Flesca, S.; Greco, S.; Masciari, E.; Saccà, D. (Eds.) *A Comprehensive Guide through the Italian Database Research Over the Last 25 Years*; Studies in Big Data; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; Volume 31, Studies in Big Data.
2. Baran, R.; Dziech, A.; Zeja, A. A capable multimedia content discovery platform based on visual content analysis and intelligent data enrichment. *Multimed. Tools Appl.* **2018**, *77*, 14077–14091. [[CrossRef](#)]
3. Moscato, V.; Picariello, A. Multimedia Data Modeling and Management. In *A Comprehensive Guide through the Italian Database Research over the Last 25 Years*, 1st ed.; Flesca, S., Greco, S., Masciari, E., Sacc, D., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; Volume 31, Chapter 16; pp. 269–284.
4. Bankier, J.G.; Gleason, K. *Institutional Repository Software Comparison. Programme and Meeting Document CI/KSD/2014/PI/H/*; UNESCO: Paris, France, 2014.
5. Antonacci, M.; Brigandì, A.; Caballer, M.; Cetinicé, E.; Davidovic, D.; Donvito, G.; Moltó, G.; Salomoni, D.; Forti, A.; Betev, L.; et al. Digital repository as a service: Automatic deployment of an Invenio-based repository using TOSCA orchestration and Apache Mesos. *EPJ Web Conf.* **2019**, *214*, 1–8. [[CrossRef](#)]
6. Verdugo, P.; Astudillo-Rodríguez, C.; Verdugo, J.; Lima, J.F.; Cedillo, S. *Documentation and Scientific Archiving: Digital Repository. Advances in Creativity, Innovation, Entrepreneurship and Communication of Design*; Markopoulos, E., Goonetilleke, R.S., Ho, A.G., Luximon, Y., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 296–302.
7. Lynch, C.A. Institutional Repositories: Essential Infrastructure for Scholarship in the Digital Age. *Portal Libraries Acad.* **2003**, *3*, 327–336. [[CrossRef](#)]
8. Joint, N. Current research information systems, open access repositories and libraries: ANTAEUS. *Libr. Rev.* **2008**, *57*, 570–575. [[CrossRef](#)]
9. Brush, D.A.; Jiras, J. Developing an institutional repository using Digital Commons. *Digital Libr. Perspect.* **2019**, *35*, 31–40. [[CrossRef](#)]
10. Schöpfel, J.; Azeroual, O. 2-Current research information systems and institutional repositories: From data ingestion to convergence and merger. In *Future Directions in Digital Information*; Baker, D., Ellis, L., Eds.; Chandos Digital Information Review; Chandos Publishing: Kidlington, UK, 2021; pp. 19–37.
11. Castiglione, A.; Colace, F.; Moscato, V.; Palmieri, F. CHIS: A big data infrastructure to manage digital cultural items. *Future Gener. Comput. Syst.* **2018**, *86*, 1134–1145. [[CrossRef](#)]
12. Carvajal, D.A.L.; Morita, M.M.; Bilmes, G.M. Virtual museums. Captured reality and 3D modeling. *J. Cult. Herit.* **2020**, *45*, 234–239. [[CrossRef](#)]
13. Fanini, B.; Pescarin, S.; Palombini, A. A cloud-based architecture for processing and dissemination of 3D landscapes online. *Digit. Appl. Archaeol. Cult. Herit.* **2019**, *14*. [[CrossRef](#)]
14. Heftberger, A. Chapter 4-Exploring the Moving Image: The Role of Audiovisual Archives as Partners for Digital Humanities and Cultural Heritage Institutions. In *Digital Humanities, Libraries, and Partnerships*; Kear, R., Joranson, K., Eds.; Chandos Publishing: Kidlington, UK, 2018; pp. 45–57.
15. Nishanbaev, I. A web repository for geo-located 3D digital cultural heritage models. *Digit. Appl. Archaeol. Cult. Herit.* **2020**, *16*, e00139. [[CrossRef](#)]
16. Xie, I.; Matusiak, K. *Discover Digital Libraries: Theory and Practice*, 1st ed.; Elsevier Science Publishers B. V.: Amsterdam, the Netherlands, 2016.
17. *Space Data and Information Transfer Systems—Open Archival Information System (OAIS)—Reference Model*; Standard ISO 14721:2012; International Organization for Standardization: Geneva, Switzerland, 2012.
18. *Space Data and Information Transfer Systems—Producer-Archive Interface—Methodology Abstract standard*; Standard ISO 20652:2006; International Organization for Standardization: Geneva, Switzerland, 2006.
19. *Space Data and Information Transfer Systems—Audit and Certification of Trustworthy Digital Repositories*; Standard ISO 16363:2012; International Organization for Standardization: Geneva, Switzerland, 2012.
20. Sharma, S.; Gupta, P.; Nagpal, C.K. *A Novel Architecture to Crawl Images Using OAI-PMH. Sensors and Image Processing*; Urooj, S., Virmani, J., Eds.; Springer: Singapore, 2018; pp. 37–48.

21. Martin-Rodilla, P.; Gonzalez-Perez, C. Metainformation scenarios in Digital Humanities: Characterisation and conceptual modelling strategies. *Inf. Syst.* **2019**, *84*, 29–48. [[CrossRef](#)]
22. Xie, K.; Di Tosto, G.; Chen, S.B.; Vongkulluksn, V.W. A systematic review of design and technology components of educational digital resources. *Comput. Educ.* **2018**, *127*, 90–106. [[CrossRef](#)]
23. Jadoon, R.N.; Zhou, W.; Haq, F.U.; Shafi, J.; Khan, I.A.; Jadoon, W. A Reliable Scheme for Synchronising Multimedia Data Streams under Multicasting Environment. *Appl. Sci.* **2020**, *8*, 556. [[CrossRef](#)]
24. Tanseer, I.; Kanwal, N.; Asghar, M.N.; Iqbal, A.; Tanseer, F.; Fleury, M. Real-Time, Content-Based Communication Load Reduction in the Internet of Multimedia Things. *Appl. Sci.* **2020**, *10*, 1152. [[CrossRef](#)]
25. Adão, T.; Pádua, L.; Fonseca, M.; Agrellos, L.; Sousa, J.J.; Magalhães, L.; Peres, E. A rapid prototyping tool to produce 360° video-based immersive experiences enhanced with virtual/multimedia elements. *Procedia Comput. Sci.* **2018**, *138*, 441–453. [[CrossRef](#)]