ICEDIG.EU *Innovation and consolidation for large scale digitisation of natural heritage*

Grant Agreement Number: 777483 / **Acronym:** ICEDIG
Call: H2020-INFRADEV-2017-1 / **Type of Action:** RIA
Start Date: 01 Jan 2018 / **Duration:** 28 months

REFERENCES:
Deliverable **D6.3** / R / PU
Work package **6.3.2** / Lead: **Plazi**
Delivery date July 2019

Digitisation infrastructure design for Zenodo

DELIVERABLE D6.3

Date: July 25, 2019

Donat Agosti (Plazi, Bern, Switzerland)
Lars Holm Nielsen (Zenodo, Meyrin, Switzerland)
Mathias Dillen (Meise Botanical Garden, Meise, Belgium)
Quentin Groom (Meise, Botanical Garden, Meise, Belgium)

DOI: 10.5281/zenodo.3346782

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

ICEDIG.EU

This deliverable specifies the requirements for adapting CERN's Zenodo services for long-term storage of large-scale digitised biodiversity data. The report describes the service(s) features, capacities, functions and its suitability. Recommendations of use and possible designs are also included. The report is structured in eight parts:

1. The context section describes the setting of the pilot, including the institutions involved (sources and services providers) and the actors within those institutions, the processes covered, and services provided.
2. The infrastructure section depicts the facilities supporting the integration of service providers and consumers, particularly APIs, programming languages, protocols, and speeds.
3. The data section provides an overview of the data model used for storing digital specimen data requiring long-term preservation, including the minimal data required for making a deposit, as well as data and metadata, which are part of each digital specimen.
4. The design section describes the overall architectural view of the implemented solution.
5. The test case section describes the upload of two data sets, a 2K of herbarium sheets with rich metadata and large digital files and a 281K of herbarium sheets with limited metadata.
6. The recommendation section provides practical recommendations to use the service(s) evaluated in the pilot, the benefits and trade-offs.
7. The references section provides the full references of bibliographic citations.
8. The appendix includes code examples.

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

1. Context

Digital copies of physical specimens stored in natural history museums play a pivotal role to provide access to and document the estimated well over 1.5 billion specimens held in European natural history collections (DiSSCo, 2017; ICEDIG, 2018). Production and usage of the images is optimized with the implementation of automated workflows using highly standardized views of the objects.

To handle the output of potentially hundreds of millions of files requires repositories powerful enough to handle that quantity and allow a highly automated input. They should also provide a data curation facility, search tools and a sustainable business model that can scale up to this dimension.

The production of digital copies is outpacing the capacity to extract metadata about the objects and the scientific names of the objects may change over time. The repositories must therefore include a stepwise process from minimal metadata to increasingly richer metadata that can be updated at any time and respective versioning.

Discovering digital objects in repositories is becoming increasingly difficult as the number of objects grows. At the same time, their value increases for application well beyond biodiversity itself (Bakker et al 2019; Watanabe, 2019; Wäldchen & Mäder, 2018)), and thus the creation of highly self-contained FAIR objects (Wilkinson et al., 2016) is essential. Making objects Findable, Accessible, Interoperable and Reusable increases their discoverability through alternative, not domain specific search tools such as Google Search and does not require a central registry containing information about the deposit.

The repository infrastructure has to be very reliable and sustainable, but at the same time specific enough to satisfy the interoperability needs that exist within different fields. Whilst reliability and sustainability is best served with infrastructure components (both hard- and software) that are widely used, the specificity of the repository depends on domain-specific vocabularies or ontologies.

Reuse of digital objects of physical specimens depends on a machine readable, open licence (RDA-CODATA Legal Interoperability Interest Group. 2016.). Scientific images of physical objects, created to enable comparative analysis, are neither individual nor unique in a legal sense and thus they in fact belong to the public domain (Egloff et al., 2017). Even so, a licence has to be included in the metadata.

To describe and test repositories, the ICEDIG project selected three different types of repositories (see Task T6.3.1, Task T6.3.2, T6.3.3.)

In the present task, T6.3.3, the viability of CERN's Zenodo repository has been tested using two types of corpora with i) low number of (large) files and very rich metadata *(A benchmark dataset of herbarium specimen images with label data* Community at Zenodo); and ii) high number of files with minimal metadata (*Belgium Herbarium of Meise Botanic Garden*).

1. 1 Certification

Zenodo is designed and operated according to the Open Archival Information System (OAIS) reference model. Full details about Zenodo's organisational and technical infrastructure, as well as

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

repository policies can be found on http://about.zenodo.org/infrastructure/ and https://about.zenodo.org/policies/.

Zenodo have not yet applied for Core Trust Seal certification but are likely to do so in 2019 or 2020.

1.2 Business model

Zenodo is offered by CERN as part of its mission to make available the results of its work (CERN Convention, Article II, §1). Zenodo is hosted by CERN, which has existed since 1954 and currently has an experimental programme defined for the next 20+ years. CERN is a memory institution for High Energy Physics and renowned for its pioneering work in Open Access[1].

Zenodo is funded by:

- the European Commission via the OpenAIRE projects through FP7 (OpenAIRE (246686), OpenAIREplus (283595)), Horizon 2020 (OpenAIRE2020 (643410), OpenAIRE-Connect (731011) and OpenAIRE-Advance (777541)).
- CERN
- Alfred P. Sloan Foundation (2 grants)
- Arcadia Fund
- Donations to CERN & Society Foundation

Zenodo is developed and supported as a marginal activity, and hosted on top of existing infrastructure and services at CERN, in order to reduce operational costs and rely on existing efforts for High Energy Physics. CERN has some of the world's top experts in running large scale research data infrastructures and digital repositories that we rely on in order to deliver a trusted digital repository.

Currently, there is no fee or cost for the use of Zenodo. This policy might be reconsidered with upload, metadata and storage requirements beyond the current load.

2. Infrastructure

Zenodo is a repository which is fully hosted in the European Organization for Nuclear Research (CERN) Data Centre. It is built on top of the Invenio digital library framework[2]. Zenodo, Invenio and all infrastructure used to run Zenodo are all licensed under open source licences approved by the Open Source Initiative (OSI). All files uploaded to Zenodo are stored in CERN's EOS[3] storage service in a dedicated cluster. CERN currently stores more than 300PB of both physics and user data in the CERN EOS service over a number of clusters. In one of the CERN EOS clusters, Zenodo currently (July 2019) hosts some 125TB of data (logical) spread over 2.8 million files and data Zenodo is operated on some 30 virtual machines and annually handles approximately 2 million visitors. Zenodo further writes Submission Information Packages (SIPs) in Bagit format to disk in order to ensure the independence of the repository software. Zenodo internally checks file integrity of the 2.8 million

[1] https://dash.harvard.edu/bitstream/handle/1/4724185/suber_timeline.htm
[2] https://inveniosoftware.org/
[3] http://information-technology.web.cern.ch/services/eos-service

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

files on a 14-days cycle. It has the technical capacity to scale up its handling of upload and requests depending on user needs, whereby special user requirements are implemented through joint projects (see e.g. the case of the Biodiversity Literature Repository community funded by Arcadia Fund[4]).

CERN, for its part of operating Zenodo, guarantees that items deposited will be retained for the lifetime of the repository which is currently the lifetime of the host laboratory CERN, i.e. the next 20 years at least. In case of closure of the repository, CERN guarantees best effort to integrate all content into suitable alternative institutional and/or subject based repositories.

A full description of Zenodo's organisational and technical infrastructure can be found on https://about.zenodo.org/infrastructure/ including all security measures.

Zenodo relies on DataCite services for registration of Digital Object Identifiers (DOI) for all uploads into Zenodo. By registering DOIs via DataCite, metadata are integrated into a suite of other discovery services that rely on the DataCite metadata registry. Zenodo, however, also exposes an OAI-PMH API for harvesting as well as integrating metadata (e.g. JSON-LD) in landing pages in order for crawlers like Google Dataset Search and Unpaywall to index Zenodo. Zenodo further has an IIIF Image API in order to preview and zoom on image data in the repository.

Zenodo further allows authentication via ORCID accounts, as well as integration with ORCID via allowing ORCIDs to be registered in Zenodo metadata and relying on the DataCite-ORCID integration to push the metadata records into ORCID.

All uploads to Zenodo, both human or machine deposits, happen via Zenodo's REST APIs. Zenodo's current 1.4 million deposits (as of June 2019) have all been uploaded through this REST API. The REST API is publicly documented at http://developers.zenodo.org, including examples.

3. Data model

An upload in Zenodo consists of a record (metadata) with one or more associated files (data). The key reason for storing the metadata is to make records findable within the repository, but also to allow the records to be integrated in any number of other discovery systems. These discovery systems can both be generic such as Google Dataset Search or domain specific such as the Global Biodiversity Information Facility (GBIF). A key benefit of using a generic repository like Zenodo, is that the metadata are made findable for a larger number of discovery systems, because the domain specific metadata are aligned to a more general metadata model.

3.1 Metadata

The Zenodo record's metadata are based upon DataCite's Metadata Schema v4[5]. Zenodo supports multiple additional export formats including Dublin Core (according to OpenAIRE Guidelines),

[4] https://tinyurl.com/y5lt42l7
[5] https://schema.datacite.org/meta/kernel-4.0/

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

MARC21, DataCite XML, JSON-LD and Citation Style Language JSON. A key benefit of registering metadata according to the DataCite Metadata Schema is that it is a widely understood metadata format that is used across all disciplines and thus makes the metadata records understandable, not only in biodiversity research, but in many other fields. Additionally, the DataCite metadata record has good support for persistent identifiers, both for authors, but also for linking related objects, which is important for reuse and linking of records across repositories. External schemas such as Darwin Core, widely used in biodiversity research, can be embedded and later exported.

See the section 5 test cases for details on specific fields.

3.2 Data

For the data files, Zenodo allows any format and size (the default maximum quota per record is 50GB but can be extended upon request). Zenodo guarantees bit-level preservation, but no format migration and thus it is important that the data uploaded to Zenodo is suitable for long-term archiving prior to being uploaded to Zenodo. This is relatively easy for image data, where good archiving formats are relatively well known (Library of Congress, 2019). The reason why Zenodo does not provide format migration is that this is highly complex for anything but standard file formats for text, images and videos. Especially for numerical research data you risk altering the actual data (e.g. due to floating point arithmetic) while doing format migration, and thus in the very end you risk altering the results presented in papers.

4. Design

The overall architecture is rather uncomplicated and relies on the repository exposing a REST API over HTTP and a background uploader client that connects to the REST API in order to reliably deposit a large number of records into Zenodo.

4.1 Background client uploader

The primary purpose of the background uploader is two-fold. First, the background uploader allows the running of the upload process over a longer period of time and keeps track of progress and possible failures. Essentially, it automates the upload into the repository for a large number of records. The second purpose is to parallelize the uploads into the repository. The REST APIs are usually focussed on depositing individual records and thus the overhead of making many HTTP requests can partly be reduced by making many HTTP requests from the client at the same time. This also better utilizes the repository, which in the case of Zenodo supports many hundreds of concurrent connections.

4.2 Repository

The primary purpose of the repository is to act as safe long-term storage and enable the discovery of

the deposited assets. The safe long-term storage is implemented by the repository itself, but essentially involves partly technical and partly organisational measures. Technical measures include e.g. regular file integrity checks (via checksums) and storing metadata/data in bagits independent of the repository software. Zenodo and the underlying Invenio framework is implemented according to the Open Archival Information System (OAIS) reference model. The organisational measures includes the mission and organisational commitment to long-term archiving of the assets which is also the case for Zenodo. Zenodo further strictly adheres to CERN's security standards and is operated according to ITIL Framework[6] which is implemented across all services at CERN.

In addition, to make the assets discoverable as described in the infrastructure section and providing IIIF Image APIs into the images, Zenodo provides COUNTER-compliant (Fenner et al., 2018) research data usage metrics (views, downloads, data volume).

5. Test cases

To evaluate Zenodo, two different datasets were uploaded to Zenodo.

- **Dataset 1:** A small number (~2,000) of herbarium specimens with rich metadata, multiple files and large file sizes (~150MB), total 208 GB, *A benchmark dataset of herbarium specimen images with label data*.
- **Dataset 2:** A large number (~281,000) of imaged specimens with limited metadata, a single file, and small file sizes (~1–5MB), total ca 1 TB, *Belgium Herbarium of Meise Botanic Garden*.

5.1 Dataset 1

Dataset description

The first dataset constituted the publication of a dataset of herbarium specimens (Fig. 1) which was part of a data paper (Dillen et al., 2018). This dataset is intended to be a benchmark for multiple use cases, including training and validating for machine learning algorithms, as well as comparing results from different citizen science transcription platforms. A considerable proportion of the dataset's records also include lossless TIFF images, resulting in a total file volume of about 208 GB. This is a considerable amount for a data paper, rendering Zenodo a compelling solution for making and keeping the dataset publicly accessible.

A full description of the dataset can be found in the data paper (Dillen et al. 2019, Open Access). The dataset comprises 1,800 records, each consisting of a compressed (but high quality) JPEG image and metadata complying with the Darwin Core (DwC) standard (Wieczorek et al. 2012). A subset of 1,400 records also each have a lossless TIFF image. A subset of 250 records have two segmented PNG images. These images indicate the location and nature of the labels on the herbarium specimen. A small number of records have more than one image for the same herbarium specimen.

Test setup

This initial upload did not use the background uploader described in the design section, but instead

[6] https://en.wikipedia.org/wiki/ITIL

employed Python scripts that used the Zenodo REST API. Essentially, the scripts are similar to the background uploader except that they serialize all HTTP requests. Each upload of a record required a minimum of 3 HTTP requests (1 request to initiate/upload the metadata, 1 HTTP request per file, and 1 request to publish the record). The scripts used for this dataset separated the initialization request and metadata upload request, which might slightly increase the processing time.

Several test runs were made against the Zenodo sandbox through its REST API to test the scripts as well as the proper metadata format.

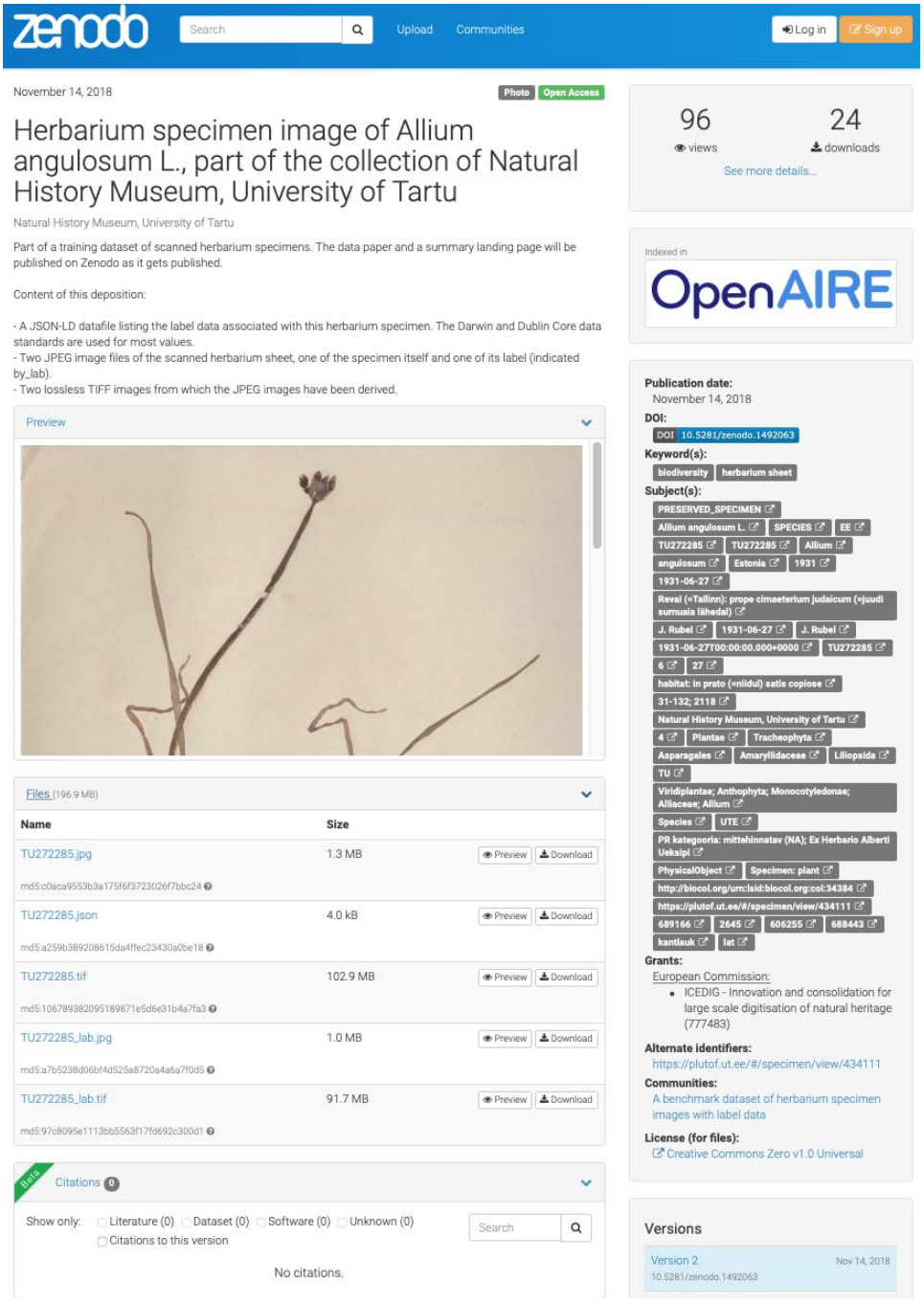

Figure 1. Exemplar page view of a herbarium specimen from the dataset 1. doi: https://doi.org/10.5281/zenodo.1492063

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

Records design

Each specimen in the dataset was uploaded as a separate record, consisting of both images and metadata. The Darwin Core metadata were mapped to the Zenodo metadata format (based on DataCite metadata schema) as much as possible, but also uploaded as a separate file in JSON-LD format. This file can be harvested and interpreted in an automated way if the record identifiers are known, but it cannot be found through Zenodo's API or search engine by its content. It is worth noting that after these tests were concluded, Zenodo added support for adding custom fields to the Zenodo metadata record, including Darwin Core, and thus the concern that the metadata were not searchable and harvestable has been addressed. Whether DataCite will add Darwin Core to its metadata standard depends on the biodiversity community approaching it.

Table 1 shows the mapping of the metadata.

Table 1: Metadata introduced into the Zenodo data model for Pilot 1. Terms in square brackets are derived from the provided specimen metadata or a list of institution names. For the right formatting of these different terms, see the mdata variable in the Python upload script (https://doi.org/10.3897/BDJ.7.e31817.suppl7).

Zenodo metadata term	**Value**
title	Herbarium specimen image of [dwc:scientificName], part of the collection of [institution name]
upload_type	image
image_type	photo

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

description (information on TIFFs and PNGs only if applicable.)	Part of a training dataset of scanned herbarium specimens. The data paper and a summary landing page will be published on Zenodo as it gets published. Content of the deposit: - A JSON-LD datafile listing the label data associated with this herbarium specimen. The Darwin and Dublin Core data standards are used for most values. - A JPEG image file of the scanned herbarium sheet. - A lossless TIFF image from which the JPEG image has been derived. - Two PNG files containing segmented image overlays of the scanned herbarium sheet. The _all extension indicates that all labels, color charts and pieces of text have received a different color against a black background color. The _sel extension indicates that these elements are white if they're barcode labels, yellow if they're color charts and red if they're anything else.
creators	[institution name]
grants	id:777483 (ICEDIG)
language	[dcterms:language]
relatedIdentifiers	identifier:[CETAF persistent identifier], relation:isAlternateIdentifier
keywords	biodiversity, herbarium sheet
subjects	[content of JSON-LD file (see text)]
communities	identifier:icedigtest

Results

The first pilot's execution ran from the 7th to the 19th of November 2018. The dataset was uploaded in batches during six nonconsecutive days, starting from a size of 6 and up to 200. The total time for all uploading was ~32 hours i.e., about 1 minute per record. Publishing all 1,800 took around 40 minutes. For troubleshooting, the publication request step was kept as a separate procedure from all other requests. For a quick overview after each batch, the status codes of each request response were logged into a simple spreadsheet.

Table 2: Summary of the first pilot results. Listed are the number of specimens per batch (#), the time it took to process the batch, the number of errors, the number of records with large TIFF images in the batch and the rate of specimens per minute. At the bottom are a few summary statistics. Batches of a single specimen were re-uploads due to errors. The extra specimen (1,801 instead of 1,800) is due to an accidental duplicate, which was deleted afterwards.

Upload	#	time (s)	error	tiff	n/min	file size (mb)	mb/s
Batch 1	6	311		6	1,16	564	1,81
Batch 2	6	297		6	1,21	424	1,43
Batch 3	14	1.061		14	0,79	2.981	2,81
Batch 4	25	1.692		25	0,89	3.311	1,96
Batch 5	50	2.362	1	50	1,27	4.361	1,85
Batch 6	100	7.357		100	0,82	13.773	1,87
Batch 7	1	104		1	0,58	84	0,81
Batch 8	100	7.331	1	100	0,82	15.007	2,05
Batch 9	99	7.986		99	0,74	14.831	1,86
Batch 10	100	8.732		100	0,69	18.726	2,14
Batch 11	99	9.480	1	99	0,63	15.278	1,61
Batch 12	1	96		1	0,63	143	1,49
Batch 13	100	9.238		100	0,65	15.764	1,71
Batch 14	100	7.863		100	0,76	14.590	1,86
Batch 15	200	15.298	1	200	0,78	30.402	1,99
Batch 16	100	10.565		100	0,57	20.094	1,90
Batch 17	100	10.505		100	0,57	20.257	1,93
Batch 18	200	910		0	13,19	1.158	1,27
Batch 19	200	3.258		32	3,68	5.852	1,80
Batch 20	100	10.248		100	0,59	15.737	1,54
Batch 21	100	2.176		68	2,76	10.852	4,99
total	**1.801**	**116.870**	**4**	**1.401**		**224.189**	

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

time (h)		**32,5**					

Discussion

The rate of uploading was quite slow at most 2MB/s, with the size of the TIFF file hence being the limiting factor in the workflow (Table 2). This rate would imply that about 1,000 specimens could be pushed to Zenodo in a single (full) day for an average TIFF size of 150MB. Server traffic at Zenodo played a key role during this test, as another user in the same period uploaded 516,000 records to Zenodo. The last batch, during which this other user was blocked, was about twice as fast as the others. This would push the daily amount up to 2,000 specimens.

After the full set was uploaded, the additional four images for four of the specimens were added manually through the Zenodo user interface, as the small number of files made it inefficient to automate this procedure. After the data paper was published in 2019, a summary deposition was added with CSV files containing all links to the individual files in the 1,800 specimen depositions, to enable selective batch downloading.

5.2 Dataset 2

Dataset description

The second dataset (Table 2) constituted a subcollection of the Meise Botanic Garden (MBG) herbarium collection named the “Belgium Herbarium”. This collection consists of 281,372 imaged specimens, but only limited data has been transcribed from their labels (Table 3). The images are available on MBG’s collection portal, but in such a way that they cannot be easily utilized by other platforms, such as Wikispecies or Europeana. When uploaded to Zenodo, they would be available for easier access and re-use.

Test setup

Based on the experiences of the test upload of dataset 1 and initial tests in December 2018, we wrote a background uploader client called zenodo-uploader (https://github.com/lnielsen/zenodo-uploader). It includes automated logging to an SQLite database, and uses the Celery distributed task queue (written in Python and uses a message queue) in order to parallelize the uploads. The uploader tool allows different concurrency levels depending on the number of CPUs on the host system. The background uploader can be distributed over multiple machines. However, in this case we ran it on a single server running Ubuntu 16.04 which had access to the image archive through a NFS share. The number of HTTP requests made to the Zenodo REST API per record was the same as for dataset 1, the only difference being the parallelization and better logging of timing for each different request.

Records design

Table 3: Metadata introduced into the Zenodo data model for Pilot 2. Terms in square brackets are derived from the provided specimen metadata. `imprint_publisher` *was used along with* `creators` *to generate the preferred format of citation. For the right formatting of these different*

terms, see the mdata variable in the Python upload script (https://github.com/lnielsen/zenodo-uploader).

Zenodo metadata term	**Value**
title	[dwc:scientificName] (dwc:catalogNumber)
upload_type	image
image_type	photo
description (blue and green parts only if applicable.)	Belgium Herbarium image of Meise Botanic Garden.
creators	Meise Botanic Garden
access_right	open
grants	id:777483
language	[dcterms:language]
relatedIdentifiers	identifier:[CETAF persistent identifier], relation:isAlternateIdentifier
keywords	Biodiversity, Taxonomy, Terrestrial, Herbarium, [dwc:family]
imprint_publisher	Meise Botanic Garden Herbarium
communities	identifier:belgiumherbarium

Results

The uploading was launched in December 2018, but aborted due to very slow progress caused by unprecedented and unexpected busy traffic at Zenodo. The zenodo-uploader tool was developed in January 2019. The tool was first tested in February, but ran into unexpected slowness after which a troubleshooting day was held on May 8th at Meise Botanic Garden with the Zenodo developer lead. After this, from May 8th to May 27th, the whole dataset was uploaded to Zenodo (Table 4). The whole task took about 12 days of full-time uploading.

As with dataset 1, the specimens were uploaded in several batches. Results from the different batches can be found in the table below.

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

Table 4: Results for the second pilot. n is the amount of specimens in the batch, c is the concurrency setting (if any), t is time in seconds. Also listed are the numbers of errors logged and the mismatch between batch size and effectively added successful publications ('unsuccess').

ID	n	C	method	t (s)	t/n	errors	unsuccess	notes
older-ones	6,597		various		8+			Various scripts used
Batch0805	2,000	2	classic	11094	5,55	33	49	
Batch0905-1	2,000	2	classic	11069	5,53	1	2	
Batch0905-2	2,000	8	updated	12073	6,04	1	2	
Batch0905-3	7,408	8	updated	47143	6,36	120	104	Fixing of backlog before 0805
Batch1005-1	8,000	8	updated	42064	5,26	345	299	
Batch1005-2	8,000	8	updated	45698	5,71	467	327	
Batch1105	12,000	8	updated	72512	6,04	571	403	
Batch1205	12,000	5	updated	70542	5,88	18	14	
Batch1305	12,000	5	updated	76649	6,39	560	546	
Batch1405	12,000	5	updated	84734	7,06	466	441	
Batch1505-1	435	10	updated	8630	19,84	344	0	Aborted due to high error rate
Batch1505-2	11,565	6	updated	69085	5,97	2128	2,096	Rest of previous batch

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

Batch1605	12,000	6	updated	30009	4,17	5804	5,795	Zenodo went down
Batch1705	16,000	6	updated	37272	2,33	0	0	
Batch1805-1	20,000	6	updated	46210	2,31	0	0	
Batch1805-2	24,000	6	updated	56767	2,37	3	3	
Batch1905	24,000	6	updated	58026	2,42	3	3	
Batch2005	23,999	6	updated	67077	2,79	5	10	Missing image from archive
Batch2105	24,000	8	updated	59270	2,47	29	20	
Batch2205	20,000	8	updated	50527	2,53	25	14	
Batch2305	20,000	8	updated	50347	2,52	609	608	
Batch2705	1,368	6	updated	3714	2,71	0	0	
TOTAL	281,372		t (d):	11,696		11532	10,736	

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

Table 5: Status codes of the errors in the second pilot. All the 404 incidents occurred on May 16 (i.e. Batch1505-2 and Batch1605).

code	n	%	response
404	5,759	50	not found
504	2,033	18	gateway timeout
502	1,997	17	bad gateway
500	1,725	15	internal server error
503	16	0	service unavailable
408	1	0	request timeout
413	1	0	request entity too large

5. 3 Discussion

There were 10,736 (4 %) specimens which failed to be successfully uploaded and published. There were 11,532 errors, of which the status codes are listed in Table 5. One other specimen was not uploaded because it was not found on the image archive due to a naming bug. A major cause for the errors (74%) was a Zenodo database incident and subsequent downtime on May 16. A standard procedure migration of the Zenodo database (performed by CERN database team operating some 800 databases) to a different host caused corruption to the database transaction logs, which prevented fast point-in-time recovery. The corruption was caused by a bug in the database management infrastructure which booted two database instances on the same database files (see http://blog.zenodo.org/2019/05/17/2019-05-17-database-incident/).

The database incident, however, occurred during a migration to a much more powerful bare-metal host, which resulted in more efficient indexing and had a significant impact on the API response time. This resulted in improving from 6 seconds to 2.5 seconds per record.

Nevertheless, it appeared very difficult to fix these failures. The logs generated by the worker were imperfect, containing both false positives and false negatives. There were 701 more successful depositions to Zenodo done than there were logged. By a few instances of trial and error, specimens were also identified which were logged as unsuccessful depositions but were found to be successfully published to the repository. These problems occur because time-outs can occur at the Zenodo server side, with requests to the API failing to be processed in time, but they can also occur when the response JSON the API sends back times out. Requesting a list of all records through the REST or OAI-PMH API runs into the hourly request limits, so this would take about 35 hours to complete and could also include errors. A modification of the tool seems necessary, which does periodic queries through the API to list what effectively failed to get successfully published. Another

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

method would be an improvement to the community system, where reports or lists can be generated which indicates the content of the community in a basic fashion (e.g. all DOIs and deposition titles or all filenames).

The local server's specifications were at no point limiting to the tool's performance.

The two test uploads of dataset 1 and 2 have proven that it is possible to upload larger numbers of records to Zenodo in reasonable amounts of time. We have identified several areas for improvement of the uploader tool, primarily related to ensuring better error handling and automatic recovery of errors to ensure less burden on the person managing the upload. Also, we identified parts of the repository where performance can be improved relatively easily. However, to achieve significantly higher transfer speeds, a change of strategy is required from uploading individual records via the REST API to bulk transfers, i.e. packaging up records and files in batches of 10,000 records on the client side and supporting bulk upload APIs on the repository side. This will allow to remove the overhead of HTTP requests, and will only be limited by the available network bandwidth. The risk of network connectivity issues causing failed uploads increases with larger file sizes, but that can be mitigated with allowing resumable uploads (via chunked multipart uploads which is already supported by Zenodo).

6. Recommendation

The following section provides practical recommendations based on uploading digitised herbarium images to Zenodo. These are applicable also for other digital copies of specimens in the scope of DiSSCo.

6.1 How to

Uploading a dataset of digitised images to Zenodo involves:

1. Preparation of metadata
2. Preparation of data files
3. Preparation of the background uploader.
4. Running the uploader
5. Validation and documentation

Preparation of metadata
To allow Zenodo to mint a DataCite DOI for an upload (deposit), a minimal metadata set defined by DataCite requirements for an image type is required (publication date, title, author, description, access right and licence). However, we strongly recommend that as much metadata as possible is added to Zenodo. The Zenodo API documentation[7] provides a full overview of all fields available. The following fields are of particular interest:

- Licence: Preferably use a Creative Commons licence that's legally well-understood.

[7] http://developers.zenodo.org/#representation

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

- Keywords: Free-form tagging of records.
- Subjects: Tagging according to vocabularies.
- Related identifiers: Using this field, you can add persistent links to other records. Using the relationship type `isAlternateIdentifier`, you can, for example, link the Zenodo record to a version of the record in your own portal.
- Communities: You can create your own community, which essentially makes all images browsable as their own collection.

The metadata are very important in order to make your dataset discoverable by other discovery systems.

In addition to the above fields, Zenodo will support new fields in September 2019 which include all, such as:

- Geographical coordinates
- Temporal metadata (e.g. date of collection)
- Darwin Core fields (e.g. collectionCode)

Preparation of data files
Long-term archiving should be considered when selecting the files to be uploaded as well as overall storage requirements. First, you should have your images in a preservation friendly format like JPEG2000 (smaller size), lossless TIFF (large size) and/or PNG since Zenodo does not do format migration but only bit-level preservation.

In addition, you may consider to add further files to each record, which could, for example be an extra metadata file in Darwin Core XML or JSON-LD as in pilot 1. Zenodo will also from September 2019 onwards be able to support an updateable extra metadata file.

It is important to note that because DOI's are assigned to each individual record, it is not possible to change the files once a record has been published other than by publishing a new version of the record. Each new version will receive a new DOI, thus previous versions remain accessible. . Thus proper preparation of the source dataset is important prior to uploading to Zenodo.

Preparation of client
Depending on the number of images that has to be uploaded, we recommend the use of the Zenodo-Uploader[8]. The linked documentation contains installation and running instructions. The tool is written in Python and customizable by a developer, in case you need to get data from local systems. This allows you to, for example, auto-generate metadata, if possible, by extraction from the image.

Running the uploader
In general, it is recommended to contact the Zenodo-team prior to performing larger automated uploads to Zenodo. You do this via https://zenodo.org/support where you normally get a reply within 1 business day. First of all, Zenodo team can help you with questions related to metadata/data preparation, but it also prevents the upload from being blocked.

[8] https://github.com/lnielsen/zenodo-uploader

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

It's advisable to first test out a smaller sample of your upload against the Zenodo Sandbox, which is running on https://sandbox.zenodo.org, but otherwise works the same as Zenodo. This allows you to see how your data will appear on Zenodo.

Finally, we recommend for the benefit of the uploader that you divide your dataset into batches of 10,000-50,000 records if it is very large.

6.2 Benefits

Zenodo is a general purpose repository and thus using Zenodo exposes your images to a larger number of discovery systems such as Wikispecies or Europeana. Moreover, by fitting your data in a general purpose data repository, you make your dataset more understandable to other disciplines, and the generic metadata allows linking between both biodiversity content and non-biodiversity content. While Zenodo is a general purpose data repository, one of its biggest communities is the Biodiversity Literature Repository which is very active and contributes with extensive domain knowledge to Zenodo.

The other key benefit of Zenodo is that it is hosted by CERN, which is already a memory institution for high-energy physics and has the size, and scale to be resilient. In addition, CERN is already operating an existing big data infrastructure as well as many other large-scale digital repositories and thus has significant operational expertise in hosting and managing research data. Essentially by uploading your data to Zenodo, the data are hosted in the CERN Data Centers, and you have a very knowledge and skilled team operating the infrastructure.

Moreover, focusing on an architecture which with strong separation of concerns allows you to focus on the science, metadata and uses cases, while Zenodo as a repository takes care of the storage and infrastructure operation.

6.3 Trade-offs

As a general data repository it can sometimes be difficult to fit domain specific use cases into Zenodo because you are forced into generally applicable use cases. On the other hand, that usually means that your images can be more easily reused by discovery systems from other disciplines.

As identified in the tests, there were some issues in obtaining very high upload speeds, though it was demonstrated that almost 300,000 images could be uploaded in 11 days. Zenodo is committed to improving performance and are happy to participate in collaboration to improve the service like has been done in this pilot.

Last but not least, Zenodo/CERN is expert in operating digital repositories. However, they are not domain experts and thus need collaborations with external partners like this case in order to bring in domain knowledge.

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

7. References

Bakker FT, Antonelli A, Clarke J, Cook JA, Edwards SV, Ericson PGP, Faurby S, Ferrand N, Gelang M, Gillespie RG, Irestedt M, Lundin K, Larsson E, Matos-Maraví P, Müller J, von Proschwitz T, Roderick GK, Schliep A, Wahlberg N, Wiedenhoeft J, Källersjö M 2019. The Global Museum: natural history collections and the future of evolutionary biology and public education. PeerJ Preprints 7: e27666v27661. doi: 10.7287/peerj.preprints.27666v1

Dillen M, Groom Q, Chagnoux S, Güntsch A, Hardisty A, Haston E, Livermore L, Runnel V, Schulman L, Willemse L, Wu Z, Phillips S 2019. A benchmark dataset of herbarium specimen images with label data. Biodiversity Data Journal 7: e31817. 10.3897/BDJ.7.e31817

DiSSCo 2017. DiSSCo research infrastructure outline. version 1702.4

Egloff W, Agosti D, Kishor P, Patterson D, Miller JA 2016. Copyright and the Use of Images as Biodiversity Data. Research Ideas and Outcomes 3: e12502. doi: 10.3897/rio.3.e12502

Fenner M, Lowenberg D, Jones M, Needham P, Vieglais D, Abrams S, Cruse P, Chodacki J 2018. The COUNTER Code of Practice for Research Data. doi: 10.5281/zenodo.3340591

ICEDIG 2018. Innovation and consolidation for large scale digitisation of natural heritage. Horizon 2020 grant 777483.

Library of Congress 2019. Recommended Formats Statement. doi: 10.5281/zenodo.3340635

RDA-CODATA Legal Interoperability Interest Group. 2016. Legal Interoperability of Research Data: Principles and Implementation Guidelines. Zenodo. doi: 10.5281/zenodo.162241

Wäldchen J, Mäder P 2018. Machine learning for image based species identification. Methods in Ecology and Evolution 9(11): 2216-2225. doi: 10.1111/2041-210X.13075

Watanabe ME 2019. The Evolution of Natural History Collections: New research tools move specimens, data to center stage. BioScience 69(3): 163-169. 10.1093/biosci/biy163

Wieczorek, J. et al., 2012. Darwin core: An evolving community-developed biodiversity data standard. PLoS ONE, 7(1). Available at: 10.1371/journal.pone.0029715

Wilkinson MD et al 2016. The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data 3: 160018. doi:10.1038/sdata.2016.18.

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483

8. Appendix

A1: Additional documentation on how to use zenodo-uploader (on Ubuntu 16.04)

This script is a test script to prove that Zenodo works. It is not a final program that everybody can use to upload data to Zenodo.

Activating the virtual environment

In the zenodo-uploader dir:

```
source uploader/bin/activate
```

Setting up the Daemon

1) Put zenodo-uploader.service file with following content (dos2unix warning):

```
[Unit]
Description=Zenodo Uploader Celery Service
After=network.target

[Service]
User=[username]
Group=[groupname]
PIDFile=[dir]/zenodo-uploader/uploader/var/celery.pid
Restart=always
WorkingDirectory=[dir]/zenodo-uploader/
ExecStart=[dir]/zenodo-uploader/uploader/bin/celery worker -A app --concurrency 5
--pidfile=[dir]/zenodo-uploader/uploader/var/celery.pid
--logfile=[dir]/zenodo-uploader/uploader/var/log/celery.log --loglevel=INFO

[Install]
WantedBy=multi-user.target
```

Into `etc/systemd/system` (sudo). Replace [username], [groupname] and [dir]. Concurrency can also be set here.

2) To edit it: `sudo nano zenodo-uploader.service`

3) To restart (after edit) : `sudo systemctl restart zenodo-uploader.service`

4) To stop: `systemctl stop zenodo-uploader.service`

5) To purge a faulty queue: `celery -A app purge` and answer `yes`

Modifying the metadata schema

Edit the `metadata` list in utils2.py, function `make_upload`.

Image paths

Edit the function `get_br13_archive_path` in `utils2.py` to have it construct the right filepath.

Funded by the Horizon 2020 Framework of the European Union
H2020-INFRADEV-2016-2017
Grant Agreement No 777483