A repeatable informatics workflow that links people and places to stable identifiers of those entities, including a validation and preservation step

Deliverable D7.3

07 October 2023

Author(s)
Mathias Dillen[1]
Andreas Plank[2]
Quentin Groom[1]

[1]: *Meise Botanic Garden, Meise, Belgium*
[2]: *Botanical Garden and Botanical Museum, Berlin, Germany*

BiCIKL

BIODIVERSITY COMMUNITY INTEGRATED KNOWLEDGE LIBRARY

This project receives funding from the European Union's Horizon 2020 Research and Innovation Action under grant agreement No 101007492.

Start of the project:		May 2021
Duration:		36 months
	Project coordinator:	Prof. Lyubomir Penev Pensoft Publishers
Deliverable title:		A repeatable informatics workflow that links people and places to stable identifiers of those entities, including a validation and preservation step
Deliverable n°:		D7.3
Nature of the deliverable:		Other
Dissemination level:		Public
WP responsible:		WP7
Lead beneficiary:		MeiseBG
Citation:		Dillen, M., Plank, A. & Groom, Q. (2023). *A repeatable informatics workflow that links people and places to stable identifiers of those entities, including a validation and preservation step*. Deliverable D7.3 EU Horizon 2020 BiCIKL Project, Grant Agreement No 101007492.
Due date of deliverable:		Month 30
Actual submission date:		Month n°

Deliverable status:

Version	Status	Date	Author(s)
1.0	Final/Draft	DD Month YYYY	Name Organisation

Table of contents

Summary

This document describes the rationale behind, the operation of and some results of a set of workflows that were developed to automatically enrich concepts of people and geography in the context of natural history collections. These workflows were designed to make use of Wikidata as a resource for these concepts, as it mints stable identifiers for them, includes input from other domains to reduce double-work and allows open contributions and curations by any volunteer. Wikidata is particularly useful as a tool to solve the “roundtripping” issue of preserving data, which commonly occurs when attempting enrichment, as it is not always possible to easily update the data at the source with the new enrichments. Wikidata’s open contribution model and generic scope addresses common obstacles such as a lack of support in domain data standards and the imperviousness of data pipelines to third party contributions.

A previous document (i.e. Milestone 32) described mainly the landscape of enrichment for natural history specimens and an analysis of different resources that may be incorporated into this workflow, in particular with regards to optimising existing methods. Taking those findings into account, the workflows described here attempt to fill gaps identified in the Milestone and provide an easy-to-use implementation to process large lists of name strings and return candidate persistently identified authority pages for those entities. To offer flexibility for different experiences with software and to investigate the results of different computational approaches, two workflows are presented, one using Python (Plank 2023) and one using R (Dillen 2023).

List of abbreviations

API	Application Programming Interface
BELS	Biodiversity Enhanced Location Services
BHL	Biodiversity Heritage Library
DiSSCo	Distributed System of Scientific Collections
EU	European Union
GBIF	Global Biodiversity Information Facility
ICN	International Code of Nomenclature for algae, fungi, and plants
ICZN	International Code of Zoological Nomenclature
IPNI	International Plant Names Index
ISNI	International Standard Name Identifier
JSON	JavaScript Object Notation
LLM AI	Large Language Model, Artificial Intelligence
MS	Mile Stone
openDS	open Digital Specimens
REST API	Representational State Transfer, Application Programming Interface
SPARQL	SPARQL Protocol and RDF Query Language

Preface

People and places are fundamental components within the biodiversity knowledge graph (Page 2016). People play a central role as the creators of scientific knowledge, contributing in

various capacities. They are distinct entities with finite lifespans, serving as concrete anchors for biodiversity data; in contrast to other more dynamic entities, people have generated extensive documentation on people through the publication of biographies (Groom et al. 2020).

Geographic entities, especially political ones, exhibit greater flexibility in their boundaries and definitions, yet they hold intrinsic significance in biogeography. Some geographic entities endure for millennia and may be crucial for policy making, delineating jurisdictions and legislative boundaries. Additionally, geographic features like mountains and forests, even without precise boundaries, can be unmistakably identified through unique identifiers (Marcer et al. 2021).

Interpreting these data presents various opportunities for data refinement and novel research, including biogeography and the history of science. Clarity emerges after the disambiguation process, which links named entities to distinct, unchanging identifiers (Groom et al. 2022). This process not only helps practitioners understand the aliases associated with individuals or locations but also overcomes language barriers that might hinder entity linkages.

Disambiguation can be a labour-intensive task due to the lack of standardisation in the names of people and places on collection objects. With approximately 1.9 billion such objects globally, the undertaking appears formidable. However, as Groom et al. (2020) noted, a power law distribution characterises the prevalence of aliases, particularly for people. By disambiguating the top 3 % of the most prolific collectors, it becomes possible to clarify the collectors of 80 % of specimens.

Thus, the aspiration to disambiguate people and places on most collection objects may not be as unattainable as initially perceived, particularly with the potential application of automation.

People

1. Author abbreviations

Of all the people connected to specimens in natural history collections, taxonomic authors are a special kind. They are not necessarily directly related to the specimen, unlike collectors or determiners, and the way their name is rendered should follow certain rules and recommendations inscribed in “the Codes” (ICN and ICZN). For the taxa covered by the Code of Algae, Fungi and Plants there are standardised abbreviations for nomenclatural authors based upon the system of Brummitt and Powell (1992). The Zoological Code recommends using the whole name of at least the first author of a name. However, in Zoology the author of the basionym is generally the only one listed and not the author of subsequent combinations. Curated lists exist of these authors together with some biographical details, connecting the common ways their names are rendered in taxon names to identifiers in dedicated resources such as IPNI or Zoobank. While some issues still remain, such as person homonyms and changes of (the interpretation of) nomenclatural rules in the past, these are quite different in scope than the people we are interested in when linking up specimens.

It is still possible to enter the names of authors (or abbreviations) into the workflows described in the rest of this section, but they were not in focus when designing the automated procedures. In particular, resolving discrepancies and gaps in taxonomic authorships requires a whole other set of resources than disambiguating collectors or determiners, including consulting and interpreting the literature where names and species were described, applying

the rules of “the Codes” and applying taxonomic knowledge to verify either the determination of the specimen, the taxonomy of the scientific name or, worse, both.

2. Subsetting Wikidata

The vast majority of humans in Wikidata (or ever in the world) have nothing to do with the science of biology or natural history collections. This is a problem as human names can be formatted in different ways and homonyms are relatively common. A way of mitigating this problem is by filtering the number of ‘potential’ humans based on criteria that make them more likely to be (or even explicitly) tied to natural history collections. In the Milestone, we assessed the current state of enrichment to help establish a set of SPARQL queries that returns a satisfying subset of Wikidata, in that it encompasses most people we want to find and can find. The 10 SPARQL query filters used can be found in this Github repository and in Table 1. The Wikispecies query uses the connection to this other Wikimedia project rather than a Wikidata property. It’s also the one most likely to time out.

Table 1: SPARQL filters used to find likely collectors/determiners in Wikidata

Property	property id	item id
occupation: botanist	P106/P279*	Q2374149
occupation: zoologist	P106/P279*	Q350979
Bionomia ID	P6944	
Harvard Index of Botanists ID	P6264	
IPNI author ID	P586	
BHL creator ID	P4081	
Entomologists of the World ID	P5370	
Zoobank author ID	P2006	
collection items at Wikispecies	P11146	

Querying a coarse occupational group, such as biologists, was too general for name matching, as it includes many occupations that have little to do with natural history collections. Zoologist and botanist, as well as their subclasses, were more reliable. This approach is still not perfect, as quite a few collectors that have been included in Wikidata have no claim that suggests they have been involved with collecting. To help mitigate this, in particular in future rounds of enrichment, the property collection_items_at (P11146) was populated through all enriched specimen records linked in the Botany Pilot triple store. This was not done for GBIF annotations nor Bionomia, as the former poses problems with regards to both specimen identifiers and institution identifiers, whereas the latter already has its own property that is well-established (Bionomia ID). This property seemed the easiest way in the current Wikidata property model to explicitly connect person records to specimens they have collected, where even a single specimen is sufficient to ‘tag’ someone as a collector.

Sending these 10 SPARQL queries to the Wikidata servers takes only a short bit of time, as only ca. 140.000 unique items will be returned in total. Still, queries are always at risk of timing out, which hampers in particular the reproducibility of the workflows. Hence, unlike previous efforts, the queries were designed to be as minimal as possible, only returning item IDs and item labels, including aliases. To avoid continuous API overhead and potential bottlenecks, in particular when time-outs occur, the workflow allows saving Wikidata results locally. However, it is recommended to only use this option in the short term, as Wikidata is ever changing.

Other item claims in Wikidata may be helpful to deliberate between multiple candidate matches or filter out false positives. In the Milestone, only a few (usable) properties were very common on person items, mainly date of birth and death. To incorporate this refining step into the workflow, we harvested these (and other potentially useful) claims directly from the items using the Wikimedia REST API, rather than the SPARQL endpoint, and only for the items that had at least one candidate match. The workflow also allows saving the results from this harvest locally, but again this is not recommended unless the workflow is rerun regularly in a short time span.

3. Name string processing

From here, the two workflows might differ in some respects, so these aspects will be covered separately.

3.1. dwc_agent

For the name strings in the source data, e.g. the names of specimen collectors, the names were presented according to various different syntaxes, sometimes in groups of multiple people (e.g. collector teams), and with many historical and/or cultural idiosyncrasies. The Bionomia platform has ample experience with such issues, and makes use of a name parser that attempts to address many of these subtleties and harmonise the output, including breaking up teams into their members. This dwc_agent Ruby Gem was therefore incorporated in both workflows to harmonise messy name strings. A wrapper was written (for both workflows) that calls a Ruby script to apply the dwc_agent parsing. The numerous different syntaxes noted for the names in the source data also helped to improve and refine the parsing performed by the Ruby Gem, by providing feedback through Github issues.

3.2. Python approach

People's names found through the Wikidata queries (resource data) were recompiled to achieve the highest possible similarity check, since in scientific literature and databases the family name is generally written first, followed by a comma, and then the given names follow, often abbreviated to just the initials. Starting from the complete, written-out name labels in Wikidata, (1) an abbreviated name and (2) a full name were created each as a so-called "canonical string" (e.g. "Hans Hermann Behr" ➔ "Behr, Hans Hermann" and "Behr, H.H."). These two canonical name strings were subsequently used in the name matching as the resource data, to which the name strings to be linked will be matched.

3.3. R approach

Names from the source data can currently only be taken from Simple Darwin Core occurrence TSV files, or from the similar Occurrence Core file from a Darwin Core Archive. As the workflow is set up in a modular way, additional formats can be added relatively easily. JSON exports produced by the current DiSSCo Sandbox for specimen data can also be ingested and processed, but this needs to still be developed further, as the openDS data model is not fully operational yet and no comparable test data were available.

Source names are parsed by the dwc_agent parser as explained in 3.1.

After harmonising name strings from the source data with dwc_agent, both the names from Wikidata and from the source are processed to guess for each unique string a (first) given name, a family name, the name initials and simplified name initials (excluding middle names). This is done on names on both sides of the matching, and is applied to all aliases of the Wikidata item labels as well.

4. Automated matching and post-processing

4.1. Python approach

In the name clustering approach programmed in Python, we followed the example of Klazenga (2020), where decomposed author names and their ordered name parts (ngrams) are analysed using a distance measure, and applied 2 methods: k-neighbour distance and cosine similarity. After calculation, the numerical value of distance or similarity is a measure of the name match, and this is the basis for deciding whether there is a name match or not. The full program code of this is documented on GitHub with detailed examples for botanical collector names (see Plank 2023).

4.2. R approach

Matching is done for each source name (e.g. a parsed collector name). Exact matching is assessed for the different name derivatives, i.e. first name, family name and initials. Initially, a fuzzy matching implementation (using Levenshtein string distance) was also included, but this was eventually dropped as it did not improve the matching results and added a lot of additional computational overhead.

Processing each name separately increases computational needs for this process, but the processing can be parallelized if the hardware supports it. The workflow is parallelized for multiple CPU cores for this reason, but as a result it becomes important for the user to carefully set how many logical cores they have available. This value (and some other parameters) can be set by modifying the config.ini file, the contents of which are documented on Github. It's advisable to monitor system resources during a run to establish the optimal number of cores to use, and to minimise any other processing done simultaneously.

5. Validation

5.1. Python approach

A calculated name distance of 0 or similarity equal to 1 can be evaluated as an exact match; in these cases of exact match, only the evaluation of possible different persons who have the same name remains. Automatic evaluation is more difficult if the match lies in intermediate ranges. Figure 1 shows a classic example with matches on two different resource names, in the intermediate ranges there are many real matches but also false ones.

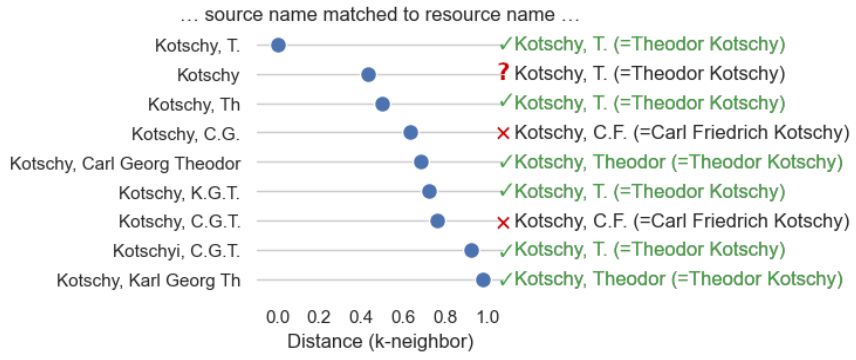
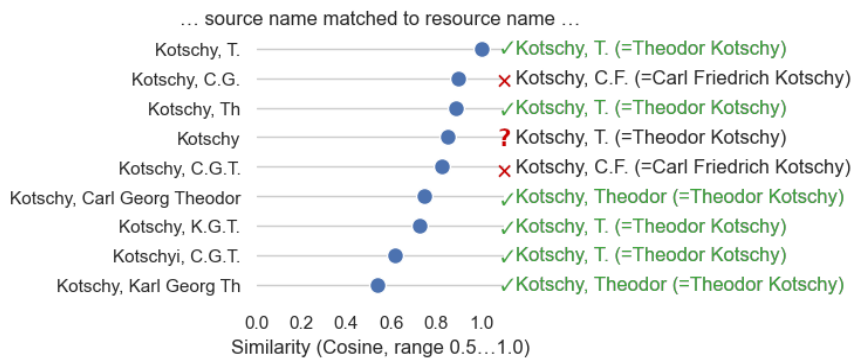


Fig 1: *Comparison of the source collector name results using the example of* Kotschy*—there are 2 available names: Karl Georg Theodor Kotschy (=Theodor Kotschy (1813-1866), Q113299) and Carl Friedrich Kotschy (1789-1856), Q86842); a name distance of 0 or a similarity of 1 is an exact match; green ✓ are true matches, red ✕ are*



false ones, red ? is ambiguous.

At present, the evaluation is still manual, but additional properties can be added to improve the accuracy of the matches, e.g. the lifetime data in combination with the collection date. It would also be conceivable to additionally evaluate alias names from Wikidata: the aim here would be to find only a 100 % match within the alias names, which would increase or confirm the certainty of finding the possibly correct name.

5.2. R approach

The workflow includes a few automated validation steps. Different name match types are given different scores, which are used to drop spurious matches (e.g. only the initials are the same). Then, only a few combinations of matches are kept as trustworthy. These constitute:

- Both name strings match exactly in full.
- Both the first and the family name match exactly.
- The family name matches and either the initials or the simplified initials match exactly.

If specimen collection dates are available, these will be leveraged to filter for lifespan of the Wikidata person to fit these collection dates. To do this, as described earlier, claims are downloaded from Wikidata for all candidate Wikidata records using the Wikimedia API. Values for date of birth (P569) and date of death (P570) are taken from those records, rounded to the year and compared with specimen date ranges (if any). Suspect specimen date ranges (i.e. longer than 120 years) are ignored. Other properties from Wikidata could be leveraged similarly, but based on the findings from Milestone 32, the two dates were the most reliable properties by far.

Multiple potential matches can still occur, as well as false positives. These could be addressed by loading the exported results (see section 6) into a curation interface such as OpenRefine. Alternatively, a simplified workflow mode can be set and the “best” result be taken for each name string – keeping in mind that source names can correctly be linked to multiple Wikidata records in the case of teams or Wikidata unmerged duplicate records.

6. Roundtripping

6.1. Roundtripping

The term “roundtripping” has many meanings in different contexts. When considering semantic enrichment by attaching persistent identifiers to otherwise ambiguous entity labels, we understand roundtripping as the process that facilitates preservation of the links established or estimated through workflows such as the ones described in earlier sections. Specifically, roundtripping is the process of somehow incorporating links into the system the original source data was managed, or otherwise ensuring the links are discoverable, non-ambiguous and persistent, for instance through a third-party system like Wikidata.

As outlined in Milestone 32, roundtripping links is not straightforward for specimen data, as the standards, the systems implementing them and the data workflows underlying the systems are not yet capable of steadily ingesting annotated links. To mitigate this issue, we implement multiple export possibilities compatible as much as possible with current implementations, even if they are still in a stage of draft or prototype.

6.2. Python approach

A draft for data output to Darwin Core (draft attribution extension, with some suggested adjustments) is currently being worked on to facilitate “roundtripping” of personal name data – this data could be reworked in OpenRefine, for example, so that subsequent feeds of the extracted data into WikiData or custom data portals would be easier.

6.3. R approach

In MS32, a script was described that maps frictionless data for attributions made in Bionomia to the proposed Darwin Core Agent Attribution extension. Similarly, the workflow can export its attributions into this extension. It is possible to also restrict this export to only the ambiguous matches (i.e. multiple candidates for a single parsed name string), so they can be later imported into OpenRefine or another manual validation interface. Two modifications were made to improve the comprehensiveness of this model in the context of the workflow. Given that this approach takes name strings, harmonises them and then matches them to labelled items in Wikidata, we have at least three name strings that can all be different:

- The original string as provided.
- The parsed string, which may be a single member of a team.
- A label (or one or more alias labels) from the matching Wikidata item.

To include all info, we mapped these respectively as `verbatimName`, `name` and `alternateName`. The matching process also comes with a score, based on the fulfilled criteria. Both score and a label for these criteria were concatenated in an `attributionRemarks` custom property. This field is not part of the latest draft of this extension, but for minimal provenance reasons it could be included.

A second method makes use of the `fst` package in R. This implements a fast read/write option for relatively big datasets in a binary format. This makes it easier to save results for later re-use with minimal overhead and (re-)conversion needed.

A third method generates a tab-separated file fit for use with the Quick Statements tool to import into Wikidata. This method is the most immediately usable for roundtripping, but not scalable to enrich all specimens, only for the people in Wikidata, to explicitly label them as a collector. There are also currently no properties (supported) for other actions than collecting. Quickstatements can be loaded through the interface at or using its API endpoint. It is advisable to run the process “in the background” as this avoids the most errors. Statements may still get stuck with status “Run”, typically because they fail to edit items that have since been merged with others. For those, an additional script needs to be run that uses the SPARQL endpoint to find the ID of the item merged into and constructs new Quickstatements for these.

The QuickStatements tool also requires authentication with an autoconfirmed Wikidata account, which is achievable most easily by producing a sufficient number of contributions to the platform (see the linked project page for more info). There are other Wikidata API endpoints that support bulk editing and implementations such as PyWikiBot, but those are generally more difficult to use.

Finally, export to the current draft of the DiSSCo annotation model is prototyped. The workflow can ingest an array of JSON objects exported from the current DiSSCo prototype and, based on the currently implemented data model, extract the needed properties for matching. Export to the annotation model produces an array of JSON annotation objects. The workflow does not interface with DiSSCo's handle API, so a GUID is minted instead as a placeholder PID for the annotation. The score is added based on normalisation with the maximum score of the current run, ensuring a number between 0 and 1, but potentially less reliable. An example output for two DiSSCo sandbox records (listed below) can be seen in Figure 2. There were three parsed collector strings (one team) and one string was matched. The big difference in score between both matches occurs due to an exact match to an alias of one Wikidata item.

https://hdl.handle.net/SANDBOX/8K9-6LP-44M
https://hdl.handle.net/SANDBOX/6CZ-M0F-38E

Figure 2: Example DiSSCo annotation model output


```
[
  {
    "data": {
      "id": "ca1457a8-2669-4dc9-bb68-89a277ba9011",
      "type": "Annotation",
      "attribution": {
        "id": "ca1457a8-2669-4dc9-bb68-89a277ba9011",
        "version": 1,
        "type": "Annotation",
        "motivation": "linking",
        "target": {
          "id": "https://hdl.handle.net/SANDBOX/6CZ-M0F-38E",
          "type": "digital_specimen",
          "indvProp": "ods:collector"
        },
        "body": {
          "type": "ods:collector",
          "value": "http://www.wikidata.org/entity/Q2491043",
          "description": "wikidata: exact_match|surname_match|initials_match|initials_outer_match||surname_match|initials_match|initials_outer_match||
          "score": 1
        }
      }
    }
  },
  {
    "data": {
      "id": "458da77f-e87a-476f-83a2-ee750d7edbbf",
      "type": "Annotation",
      "attribution": {
        "id": "458da77f-e87a-476f-83a2-ee750d7edbbf",
        "version": 1,
        "type": "Annotation",
        "motivation": "linking",
        "target": {
          "id": "https://hdl.handle.net/SANDBOX/6CZ-M0F-38E",
          "type": "digital_specimen",
          "indvProp": "ods:collector"
        },
        "body": {
          "type": "ods:collector",
          "value": "http://www.wikidata.org/entity/Q28554532",
          "description": "wikidata: surname_match|initials_match|initials_outer_match||surname_match|initials_match|initials_outer_match",
          "score": 0.0934
        }
      }
    }
  }
]
```


6.4. Results

To test the workflow, we used three pilot datasets from the collections of task contributors. These include the botany collections of Naturalis (Bijmoer et al. 2023), Meise Botanic Garden (2023) and the Virtual Herbarium Germany network (see all collections in the References section).

For Naturalis botany, there were 58.815 parsed names for 4.991.356 specimens with 100.857 unique strings. Of these parsed names, 14.286 were matched to a single Wikidata item (after automated validation) and 7.286 to more than one. 37.243 names could not be (sufficiently reliably) matched to Wikidata items. This may occur for various reasons (which are not all exclusive):

- The person has no item in Wikidata.
- The person has an item in Wikidata, but it was not found through the SPARQL queries listed in section 2.
- The person's name is incorrectly parsed by dwc_agent. Some of these cases were noted by issues on the Gem's Github repository.
- The person's name is formatted in such a way that dwc_agent cannot realistically parse it correctly. This includes typos and transliteration differences.
- The person's name is formatted in such a way that the automated validation will have a low chance of retaining a match. For example, single names (that are probably single surnames) can only be retained if an exact match is found with a Wikidata label or alias.
- The person is, in fact, not a person. For example, it is an expedition or a group of students who collected as an educational assignment.

Running the workflow on a machine with an i7-13700HX CPU with 16 cores and 32GB RAM, the matching process took ca. 10 minutes, including setting up the cluster. The validation process took about 16 minutes, with most of that time (10 minutes) spent on reading cached Wikidata item claims from disk. This is time-consuming as the claims are saved as JSON and need to be converted back into R list objects. This process is not yet parallelized and should benefit from this optimization method. If these claims (still) are to be downloaded from the Wikimedia REST API, the process might take longer depending on the response speed of that API and the size and distribution of item claims in the local cache.

For Quick Statements, this generates claims for collection_items_at for 12.479 person items, of which 10.444 did not have such a property claim yet (and ergo were found in Wikidata by other SPARQL queries from section 2 with other, less explicit claims). Hence, implementing this workflow in multiple collections will increasingly improve the reliability of workflows such as these as an ever greater number of people will get explicitly connected to specimens, dubbing them as "specimen collectors" and connecting them to both one or more institutions and one or more physical specimens. As an example of this approach, we imported claims for collection_items_at based on the Wikidata items identified from the Botany Pilot in Milestone 32 in bulk through QuickStatements. As a result, there are now more than 10.000 items in Wikidata with this property claim.

One downside of this QuickStatements approach is the requirement of an institution or collection ID. To do the bulk import from the results of Milestone 32, a "hack" had to be introduced to deal with the multiple collections that are part of the JACQ consortium and a lookup table had to be manually created. Similarly, for GBIF this approach was not feasible as it is not trivial to identify the holding institution or collection from a GBIF dataset, as this information may be in several different places and formats. The QuickStatements export service therefore currently only supports an export for a single collection with a provided Wikidata item ID.

Table 2: Results of the matching workflow for the three pilot datasets. For the meaning of each number see the breakdown for Naturalis above.

	Naturalis Botany	Virtual Herbarium Germany	Meise Botanic Garden Herbarium
parsed names	58.815	27.613	67.696
specimens	4.991.356	1.098.627	2.779.376

unique strings	100.857	64.440	85.774
single match	14.286	9.943	13.462
multi match	7.286	3.929	6.350
no match	37.243	12.041	43.855
quick statements	12.479	11.829	9.865
new claims	10.444	10.962	7.947

Geography

1. Geonames

Geonames is a somewhat similar resource to Wikidata, although it has a more specific scope for geographic entities. It is an open system that can be curated by volunteers and contains more than 12 million records. Almost 4 million of those are also in Wikidata and directly linked. Mapping exercises between both resources have been attempted but proved quite difficult and have since been abandoned. The exact stats of discrepancy are difficult to calculate as both data structures are rather messy. What makes Geonames a more interesting resource is that it is by its scope restricted to locations, and that its dumps are manageable without access to greater computational resources, unlike for instance OpenStreetMap.

2. Subsetting

For Wikidata on the other hand, geographic entities are difficult to comprehensively group. Unlike people, where `instance of human` is very commonly followed, locations have a myriad of possible claims to identify them and only a union of queries for links to various external resources might work.

However, there are WAY more localities in Wikidata than (context-relevant) people. Unlike people, localities are more difficult to subset, as specimens can be collected (or other actions performed on them) in virtually any of the localities listed. Subsets of Wikidata are hence overkill and will not work through SPARQL due to time outs. Working with raw dumps is a more feasible solution. These will be more easy to acquire from a resource such as Geonames than from Wikidata.

Subsetting can be done once a dump is obtained to make matching more computationally efficient and more reliable. Country is a very helpful filter that is commonly available. This will cause some locations to be matched incorrectly, if the tagged country is incorrect. This is particularly problematic in border cases where the geographic feature crosses one or more borders.

Still, even with a country filter, false positives are much more likely in a matching workflow similar to the ones designed for people. This is particularly true in countries or regions with many homonymic locality names.

3. Geocoding

Geocoding is the process of converting textual descriptions of locations into quantitative geographic coordinates. Several resources exist that allow for Geocoding and these are more and more driven by Machine Learning language algorithms (LLMs) to interpret the text. As a

result, these are also computationally intensive and hence not available for free or without major usage limits. Examples are commercial such as Google Maps and its Geocoding API, but also OpenStreetMap and its Geocoding implementations like Nominatim. These are probably best consulted in a post hoc step for unmatched and dubious match results, to minimise the costs and loads.

4. Adapting the workflow

With the issues mentioned in the previous sections in mind, it should be possible to adapt the workflows devised for people to also work for geographical entities. Instead of Wikidata, Geonames could be consulted as a resource. This does not mean that Wikidata will be ignored: if a successful match is made and the corresponding Geonames record is not in Wikidata, this constitutes a valid reason to add it to Wikidata, as there is now a need for its existence.

Changes will have to be made to some parts of the workflows. This work is not yet complete, but its current stage can be found here. Address parsers exist, but are typically designed with certain address standards in mind, something that is hard to rely on in biodiversity data. The BELS toolbox includes some parsing methods, but not as extensive as dwc_agent for people.

The volume of data can be expected to be much greater. Not only are there more location entities to look through, there is also a greater number of locality descriptions. Because of this, computations will take much longer and more false positives are to be expected. Even when adding a country filter for both lists the computations will be intensive and memory may become a bottleneck.

Roundtripping is also much less straightforward for geographic entities. The most common way those are enriched are through geocoding into quantitative descriptions, such as shapes with proper Coordinate Reference System, Datum and uncertainty. PIDs for geographic entities themselves are not often published, in no small part because they are less straightforward to use for research and analysis, as they require an extra resolution step and understanding of the underlying data model of a gazetteer such as Geonames.

Darwin Core has a property suitable for a geographic entity PID in `dwc:locationID`, but this is only commonly used for repeated observations and ecological data, rather than collection events. While it is available in GBIF exports, it is not an indexed field and hence not searchable.

Wikidata is also not the most obvious place to store links between location items and specimens collected there. This leaves annotations through the DiSSCo data model as the main feasible approach at the moment.

Acknowledgements

We thank David Shorthouse for his prompt feedback and code improvements, and deepl.com for translation assistance.

References

Bijmoer R, Scherrenberg M, Creuwels J (2023). Naturalis Biodiversity Center (NL) - Botany. Naturalis Biodiversity Center. Occurrence dataset https://doi.org/10.15468/ib5ypt accessed via GBIF.org on 2023-04-14.

Brummitt, R. K. and C. E. Powell. 1992. Authors of Plant Names Standard. International Working Group on Taxonomic Databases (TDWG) http://www.tdwg.org/standards/101

Dillen, M. (2023). collector_matching: An R workflow. https://github.com/AgentschapPlantentuinMeise/collector_matching

Groom, Q., Dillen, M., Hardy, H., Phillips, S., Willemse, L., & Wu, Z. (2019). Improved standardization of transcribed digital specimen data. Database, 2019, baz129. https://doi.org/10.1093/database/baz129

Groom, Q., Güntsch, A., Huybrechts, P., Kearney, N., Leachman, S., Nicolson, N., Page, RDM., Shorthouse, DP., Thessen, A.E., Haston, E. (2020) People are essential to linking biodiversity data, Database, 2020, baaa072, https://doi.org/10.1093/database/baaa072

Groom, Q., Bräuchler, C., Cubey, R., Dillen, M., Huybrechts, P., Kearney, N., ... Haston, E. (2022). The disambiguation of people names in biological collections. Biodiversity Data Journal. 10: e86089. doi:10.3897/BDJ.10.e86089

Klazenga, Niels. (2020). 'AVH Collectors (Australasian Virtual Herbarium)'. Jupyter Notebook. https://github.com/nielsklazenga/avh-collectors

Marcer, A., Haston, E., Groom, Q., et al. Quality issues in georeferencing: From physical collections to digital data repositories for ecological research. Divers Distrib. 2021; 27: 564–567. https://doi.org/10.1111/ddi.13208

Meise Botanic Garden (2023). Meise Botanic Garden Herbarium (BR). Version 1.29. Meise Botanic Garden. Occurrence dataset https://doi.org/10.15468/wrthhx accessed via GBIF.org on 2023-04-14.

Page, R. (2016) Towards a biodiversity knowledge graph. Res. Ideas Outcomes, 2, e8767. doi: 10.3897/rio.2.e8767

Plank, A. (2023). 'Matching of Collector Names to Other Resources'. Python. https://github.com/infinite-dao/collector-matching/

Virtual Herbarium Germany

https://doi.org/10.15468/dl.tued2e

Biological Institute Helgoland (BAH) in the Alfred Wegener Institute, Helmholtz Centre for Polar and Marine Research. AWI-Herbarium Marine Macroalgae. Occurrence dataset https://doi.org/10.15468/uyo5za accessed via GBIF.org on 2023-04-26.

Bodensee-Naturmuseum Konstanz. Leiner-Herbar Konstanz. Occurrence dataset https://doi.org/10.15468/zprnhi accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin (2020). Field Museum of Natural History (Botany) - Historical Photographs of Type Specimens from Berlin (B). Occurrence dataset https://doi.org/10.15468/c4krdu accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin (2016). Flora exsiccata Bavarica. Occurrence dataset https://doi.org/10.15468/dnmpiw accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin (2017). Herbarium Berolinense, Berlin (B). Occurrence dataset https://doi.org/10.15468/dlwwhz accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin (2021). Herbarium Bridel at Herbarium Berolinense, Berlin (B). Occurrence dataset https://doi.org/10.15468/mtnt5m accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin. Herbarium Willing at Herbarium Berolinense, Berlin (B). Occurrence dataset https://doi.org/10.15468/abcz8i accessed via GBIF.org on 2023-04-26.

Botanic Garden and Botanical Museum Berlin (2016). Lichens at Herbarium Berolinense, Berlin (B). Occurrence dataset https://doi.org/10.15468/gmyyyu accessed via GBIF.org on 2023-04-26.

Botanischer Garten, TU Dresden. Herbarium Dresdense. Occurrence dataset https://doi.org/10.15468/qwjw7w accessed via GBIF.org on 2023-04-26.
Friedrich-Alexander University of Erlangen-Nürnberg. Herbarium Erlangense. Occurrence dataset https://doi.org/10.15468/i7some accessed via GBIF.org on 2023-04-26.
Georg-August-Universität Göttingen, Albrecht-von-Haller-Institut für Pflanzenwissenschaften, Abteilung Systematische Botanik. Forster herbarium, Göttingen (GOET). Occurrence dataset https://doi.org/10.15468/ywglaz accessed via GBIF.org on 2023-04-26.
Georg-August-Universität Göttingen, Albrecht-von-Haller-Institut für Pflanzenwissenschaften, Abteilung Systematische Botanik. Type herbarium, Göttingen (GOET). Occurrence dataset https://doi.org/10.15468/cft0di accessed via GBIF.org on 2023-04-26.
Georg-August-Universität Göttingen, Albrecht-von-Haller-Institut für Pflanzenwissenschaften, Abteilung Systematische Botanik. Bryophyte herbarium, Göttingen (GOET). Occurrence dataset https://doi.org/10.15468/ppw8gu accessed via GBIF.org on 2023-04-26.
Herbarium Hamburgense. HBGBryophyta - Herbarium Hamburgense. Occurrence dataset https://doi.org/10.15468/m9xcfk accessed via GBIF.org on 2023-04-26.
Herbarium Haussknecht. University of Jena, Herbarium Haussknecht - Herbarium JE. Occurrence dataset https://doi.org/10.15468/8arhjc accessed via GBIF.org on 2023-04-26.
Naturhistorisches Museum Mainz. Naturhistorisches Museum Mainz, Herbarium Oesau. Occurrence dataset https://doi.org/10.15468/sciejp accessed via GBIF.org on 2023-04-26.
Naturhistorisches Museum Mainz. Naturhistorisches Museum Mainz, Botanical Collection. Occurrence dataset https://doi.org/10.15468/l0wmu8 accessed via GBIF.org on 2023-04-26.
Senckenberg. Herbarium Senckenbergianum (FR). Occurrence dataset https://doi.org/10.15468/ucmdjy accessed via GBIF.org on 2023-04-26.
Senckenberg. Herbarium Senckenbergianum (GLM) - Plantae. Occurrence dataset https://doi.org/10.15468/1cvts9 accessed via GBIF.org on 2023-04-26.
Staatliche Naturwissenschaftliche Sammlungen Bayerns. The Vascular Plant Collection at the Botanische Staatssammlung München. Occurrence dataset https://doi.org/10.15468/vgr4kl accessed via GBIF.org on 2023-04-26.
Staatliche Naturwissenschaftliche Sammlungen Bayerns. The Vascular Plant Collection at the Herbarium MSB, Universität München. Occurrence dataset https://doi.org/10.15468/trfgn8 accessed via GBIF.org on 2023-04-26.
Staatliche Naturwissenschaftliche Sammlungen Bayerns. The Vascular Plant Collection at the Herbarium Tubingense. Occurrence dataset https://doi.org/10.15468/2wbxxh accessed via GBIF.org on 2023-04-26.
Staatliche Naturwissenschaftliche Sammlungen Bayerns. The Vascular Plants Collection of the Regensburgische Botanische Gesellschaft. Occurrence dataset https://doi.org/10.15468/rv6kzy accessed via GBIF.org on 2023-04-26.