



Deliverable 1.3 (D1.3)

Systems for mobilizing and managing collection-based data (specimen + DNA-data) fully integrated

M48

Project acronym: EU BON
 Project name: EU BON: Building the European Biodiversity Observation Network
 Call: ENV.2012.6.2-2
 Grant agreement: 308454
 Project duration: 01/12/2012 – 31/05/2017 (54 months)
 Co-ordinator: MfN, Museum für Naturkunde - Leibniz Institute for Research on Evolution and Biodiversity, Germany

Delivery date from Annex I: M48 (November 2016)

Actual delivery date: M48 (November 2016)

Lead beneficiary: NRM (Swedish Museum of Natural History, Sweden)

Authors: Fredrik Ronquist and Markus Skyttner (NRM, Swedish Museum of Natural History, Sweden)

Urmas Kõljalg (UTARTU, University of Tartu, Natural History Museum, Estonia)

Dominik Röppert (BGBM, Botanic Garden and Botanical Museum Berlin-Dahlem, Germany)

Lyubomir Penev and Pavel Stoev (Pensoft, Pensoft Publishers Ltd, Bulgaria)

Israel Peer (GlueCAD, GlueCAD Ltd. – Engineering IT, Israel)

Martin Stein and Isabel Calabuig (UCPH, University of Copenhagen, Natural History Museum of Denmark, Denmark)

Donat Agosti (PLAZI, Plazi GmbH, Switzerland)

Matúš Kempa (IBSAS, Slovak Academy of Sciences, Institute of Botany, Slovakia)

This project is supported by funding from the specific programme 'Cooperation', theme 'Environment (including Climate Change)' under the 7th Research Framework Programme of the European Union

Dissemination Level

PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

This project has received funding from the European Union's Seventh Programme for research, technological development and demonstration under grant agreement No 308454.

All intellectual property rights are owned by the EU BON consortium members and protected by the applicable laws. Except where otherwise specified, all document contents are: "© EU BON project". This document is published in open access and distributed under the terms of the Creative Commons Attribution License 3.0 (CC-BY), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



Executive Summary

Introduction. A large portion of the biodiversity data in natural history collections is still not available digitally. Increasingly, innovative high-throughput methods are being applied to digitize this backlog in bulk, generating large amounts of data. In parallel, natural history museums are becoming increasingly involved in the generation of large amounts of molecular biodiversity data using new massively parallel sequencing platforms. Against this backdrop, the goal of EU BON Task 1.4 has been to support data mobilization efforts targeting collection-based and molecular data, mainly through the development and integration of innovative open-source tools and services.

Progress towards objectives. The activities have involved work within the context of three major projects: i) DINA, an open-source, modular, web-based collection management system for natural history specimen data. ii) JACQ an open-access system for botanical (herbarium) data. iii) PlutoF, a web platform for working with traditional and molecular biodiversity research data. The task has also involved work on a number of other EU BON partner systems and services, as well as integration across internal EU BON and external biodiversity informatics resources. Finally, these systems have been used for targeted data mobilization efforts.

Achievements and current status. Within DINA, the focus has been on supporting the engineering of sophisticated biodiversity information systems through the exploration of tools supporting distributed development and a modular plug-and-play design based on services-oriented architectures. This has involved the testing and adoption of tools like Apiary for the design of Application Programming Interfaces (APIs) and Docker for systems integration and deployment tasks. A Python library for data migration to DINA was also developed and tested. Within JACQ, a number of tools were developed to facilitate deployment and data migration to the system, and the AnnoSys tool for annotation of data has been integrated. Within PlutoF, EU BON efforts focused on the development of a citizen-science module and improved functionality for the mobilization of collection (living) specimen data. A number of innovative tools were developed by Pensoft to help mobilize biodiversity data published in the scientific literature, including semantic mark-up of species conservation papers, direct import of data from a range of biodiversity platforms into manuscripts, and a mechanism for providing stable links from publications to global biodiversity repositories. Plazi implemented an automated workflow mining published scientific papers for taxonomic data, currently mobilizing 25 % of all published new names as they become available. GlueCad developed apps allowing citizen scientists reporting spontaneous observations or systematic inventory data to select target taxa and preferred data mobilization platform. IBSAS and UCPH have focused on national data mobilization efforts targeting Slovakia and Denmark, respectively.

Future developments. Although the development is clearly towards increased integration of biodiversity informatics tools into larger and more sophisticated systems, it is clear that there is no one size that fits all. Nevertheless, the increasingly widespread adoption of community standards, open-source development practises and service-oriented architectures are pushing the capability of current systems forward and facilitating tighter integration across systems. This trend is supported by the appearance of sophisticated tools enabling the design and deployment of complex modular systems. The adoption of the Docker approach is one example of how the biodiversity informatics community may benefit from this.

Table of content

1. Introduction	5
1.1. Background.....	5
1.2. Open Science, Open Data and Open Source.....	6
1.3. Overview of EU BON contributions in Task 1.4.....	8
2. Progress in systems development.....	8
2.1. JACQ	8
2.2. PlutoF– online solution for biological data management and sharing of Open Data	9
2.3. DINA – Digital information system for natural history collections	12
3. Integration of systems	12
3.1. Apiary for API blueprints	13
3.2. System integration and deployment using Vagrant	16
3.3. System integration and deployment using Docker	17
3.4. Taxonomic integration.....	18
3.5. Globally unique and persistent identifiers	20
3.6. Annotations.....	20
4. Data Mobilization Projects.....	21
4.1. Dataflos in Slovakia.....	21
4.2. Mobilizing biodiversity data published in the scientific literature through Plazi	21
4.3. Mobilizing citizen science data through GlueCAD apps.....	23
4.4. Mobilizing Danish biodiversity data.....	25
4.5. Pensoft mobilization efforts.....	25
4.6. JACQ Data Mobilization Efforts at BGBM	29
4.7. DINA data mobilization efforts	29
5. Services Provided to the Community	29
6. Summary – Looking Ahead	32
7. References	33

1. Introduction

1.1. Background

Natural history collections in the world are estimated to contain 1.2–2.1 billion specimens (Ariño 2010), around a third of which are likely to be housed in European institutions¹. These specimens represent some of the most reliable historical biodiversity data we have. Natural history collections are also the best source of information for many groups of organisms that are poorly represented or entirely lacking in more recent biodiversity datasets collected by citizen-science projects or through monitoring efforts. Specimen-based biodiversity data may also be associated with rich sets of metadata, including information about morphology, genetic makeup and other aspects that are rarely available from other types of biodiversity data. Unlike many other sources of information about the environment, collections-based data also allow unexpected or critical records to be further validated at any point in the future by going back to the actual specimens for additional investigations.

Only a small fraction of the specimen-based biodiversity data is currently web-accessible. At the time of writing, GBIF (<http://gbif.org>) serves 120 M specimen records (in GBIF terminology, records where the ‘basis of record’ is ‘specimen’), about half of which are georeferenced and easily available for research and applied environmental analyses. This means that the available data represent less than 5 % of the estimated total. In other words, there is a tremendous challenge ahead of us in digitizing and mobilizing data from natural history specimens. A number of initiatives across the world are now addressing this challenge through the development and application of innovative approaches, such as high-throughput image-based digitization, optical character recognition, artificial intelligence, and crowdsourcing.

Collection data has traditionally been managed in card catalogues if there has been any external system for information management at all. In more recent times, card catalogues have usually been replaced by purpose-built database systems, often developed and maintained by the curators themselves in generic database software like FileMaker and Access. This is still the situation in most natural history collection institutions. The past decades of mobilizing biodiversity data through GBIF has often involved the movement of data from such specialized in-house systems to web servers with standardized data interfaces through more or less complicated processes involving periodic data dumps that have been run through custom-built scripts to clean the data and map the content to standardized formats.

For several reasons, institutions have been looking at the possibility of moving to more centralized collection management systems (CMSs) in the last decade or so. One of the drivers of this change is the rapidly increasing legal and technical requirements on information management in natural history collections, which has made it difficult for staff without substantial informatics expertise to deliver satisfactory solutions. Curators are also getting increasingly accustomed to sophisticated web user interfaces to services they use in their everyday life, like online banking and online travel agencies, which contributes to raising the technical bar for the CMSs they use at work. From a museum management perspective, centralized CMSs are attractive because they promise to be more cost-effective than a diverse flora of small databases maintained by individual curators, and they provide

¹ At the writing of this report, around one third of the available GBIF specimen records are published by European countries. Arguably, this represents a reasonable estimate of the European share of all natural history specimens, including those that have not been digitized yet.

better administrative control. There is also the argument that a centralized CMS can free up time for curators to focus on collections work, where their core expertise lies.

Currently, there are many different centralized CMS options available. They range from proprietary commercial systems, like Axiell Software's EMu (<http://emu.axiell.com>), over open-source systems developed by commercial providers, like CollectiveAccess (<http://collectiveaccess.org>), to community-built open-source solutions, like Specify (<http://specifysoftware.org>). The main advantage of community-built solutions is, arguably, that they tend to better match user needs than other types of systems, and respond faster to changes in those needs. With the transformative changes we are seeing now in the digitization and mobilization of natural history collections data, community-based solutions are particularly appealing to many curators.

Another important factor driving the development towards community-developed centralized information management systems is the revolution in DNA sequencing technology. Massively parallel sequencing (MPS; also known as next-generation sequencing) is increasingly being used for biodiversity analyses of environmental samples, and this creates a need for assembling reference libraries of relevant genetic markers tied to physical collections of DNA extracts and voucher specimens. MPS platforms are also instrumental in genome sequencing, increasingly used in evolutionary and environmental research. Natural history museums are involved in all of these activities, which generate massive amounts of data. The processing and management of these data require appropriate computing resources and informatics infrastructures, which are typically provided by centralized facilities running community-developed software systems. An important reason for this is that the field of genetic and genomic biodiversity data is close to the research front and evolves very rapidly, and community-developed solutions tend to fit the needs better than any off-the-shelf commercial solutions, if such solutions are available at all.

EU BON task 1.4 has been focused on driving the development of community-built solutions for digitization and mobilization of collections-based data, including both traditional specimen data and DNA sequence data. As these systems increase in complexity, they grow beyond the capacity of a single development team at one institution, and it becomes important for different development teams to be able to work effectively together across institutions. Modern tools for collaborative open-source development and the trend towards services-oriented architectures have facilitated distributed development efforts. However, different teams typically adopt different programming languages and other software tools in their development, making systems integration and deployment challenging. Therefore, a major effort in Task 1.4 has been devoted to exploring and implementing technologies that simplify systems integration and the deployment of complex integrated systems. In Task 1.4, we have also used the available resources for targeted pilot digitization and mobilization efforts, and for the development of open services available to the community for such tasks. Most of the work has been in the context of one of three different systems: DINA (Digital information system for natural history data, a CMS), JACQ (a CMS intended primarily for herbaria), and Pluto-F (a general platform for biodiversity data management and research). Each of them will be presented in more detail below. EU BON task 1.4 has also involved work on several additional systems for digitizing and mobilizing biodiversity data, including Plazi, GlueCAD and DataFlos.

1.2. Open Science, Open Data and Open Source

While there is widespread agreement that biodiversity data should be open, there are different opinions on how the principle of openness should be applied to biodiversity information management systems more generally. The position taken by the [open science](#) movement is that all aspects of

scientific research should be accessible to all levels of an inquiring society, amateur or professional. Open science encompasses practices such as publishing [open research](#), campaigning for [open access](#), encouraging scientists to practice [open notebook science](#), and generally making it easier to publish and communicate scientific knowledge.

[Reproducible open research](#) goes one step further in requiring that data analyses, and scientific claims more generally, be published with their data and software code so that others may verify the findings and build upon them. This makes it possible to share and collaborate on all steps in the chain from raw data to scientific knowledge dissemination, and provides the tools needed for anyone to critically analyse and validate published findings.

The need for reproducibility is increasing rapidly as data analyses become more complex, involving larger data sets and more sophisticated computation. Reproducibility allows for people to focus on the actual content of a data analysis, rather than on superficial details reported in a written summary. In addition, reproducibility makes an analysis more useful to others because both the data and code used in the analysis are made available.

Systems for mobilizing and managing biodiversity data from natural history collections can contribute to the development of open science and reproducible open research in several ways. Traditionally, the systems in this space ensure that biodiversity data are shared as [open data](#), that is, data that are freely available to everyone to use and republish as they wish, without restrictions from copyright, patents or other mechanisms of control. Some EU BON systems, like PlutoF, also directly support open-access publication of data sets and hypotheses derived from data.

Whether digitization and mobilization systems should also be open-source software is more contentious even though open-source licenses provide a number of benefits. GBIF is an example of an organization that provides open biodiversity data but also makes its software available under an open-source license. This means that software components developed by GBIF can be reused in other systems, and that further work on these components by external developers can be contributed back to GBIF as potential future improvements. This greatly facilitates international collaboration in systems development. The permission to use, modify and redistribute open-source components is critical in facilitating system integration efforts; if open licenses are not used, systems integration and packaging of components using modern tools such as Docker and Docker Hub (see below) often in practice becomes so difficult and cumbersome that collaborations will be restricted to the traditional model of sharing datasets rather than software components.

Inspired by this vision, the DINA system is designed to maximally exploit services-oriented architecture, mandatory open-source licensing and open code development in facilitating a distributed development effort. This is arguably the best approach for a system that is intended for deployment in multiple instances at separate institutions, like DINA. The JACQ and PlutoF projects are less focused on distributed development; these systems are primarily intended to be run as central instances that offer openly and freely available services to the community. Nevertheless, thanks to EU BON efforts, we can provide one PlutoF component, the taxonomy module, as an easily installable, independent system component under an open-source license, for those interested in using it as a component in larger systems. We also offer the full JACQ system in a similar way.

1.3. Overview of EU BON contributions in Task 1.4

The EU BON contributions in task 1.4 can be summarized as consisting of four different, interdependent activities, that partly ran in parallel (**Fig. 1**). The first activity focused on further development of the DINA, JACQ and PlutoF systems, targeting data mobilization functionality among other things. In the second activity, we explored two different approaches to general systems integration, Vagrant and Docker, and embarked on specific systems integration efforts. In the third activity, the EU BON systems and tools were used for mobilizing specific data sets in pilot projects. In the fourth and final activity, a number of services were released to the community. Each of these activities will be described in more detail below.



Figure 1. Overview of EU BON task 1.4 efforts. Development of the DINA, JACQ, and PlutoF systems (“Systems development”) was followed by integration efforts (“Integration of systems”) and mobilization of specific specimen and DNA data sets (“Mobilization of data”). Finally, a set of EU BON services and tools for systems integration and data mobilization were released to the community (“Services to the community”).

2. Progress in systems development

2.1. JACQ

The open source herbarium collection management system [JACQ](#) is an entirely web-based software platform for managing and publishing herbarium specimen metadata. The JACQ network includes 32 participating institutions managing over 50 collections worldwide, so it is a widely established botanical collection information system. JACQ is based on a shared MySQL database, which is mirrored and fully synchronized by four hosting institutions. The particular strength of the system lies in the re-usability of duplicate specimen-information across participating institutions, as well as its thoroughly maintained pool of high quality scientific names. In addition, JACQ can be used by institutions of any size even if they have no IT facilities. By joining JACQ, participating institutions can make their data and images immediately available to the community and data aggregators like GBIF via the [BioCAsE provider software](#) services.

During the EU BON Task 1.4, efforts were primarily focused on speeding up the workflows for getting field data from scientific projects into JACQ and publishing them via accepted service layers. We implemented a staging area for quick data imports (see section 4.8) and published them in a useable and citable manner by integrating CETAF Stable Identifiers (see section 3.3) and the AnnoSys annotation service (see section 3.4).

For the aim of integrating JACQ with external web services, the current architecture of JACQ - being a closed bundled web application talking directly to a MySQL database - has been identified as an obstacle. For example, there is no API available allowing external web service to consume JACQ data. To improve this situation, the JACQ system will be lifted by basing it on the Yii 2 application framework (<http://www.yiiframework.com/>), which provides excellent support for service integration.

For flexible and smooth deployment of the JACQ system we started to evaluate Docker images, which provide the whole JACQ system or separate components for onsite installations. This is an important step since server infrastructures are increasingly being built on top of this modern virtualization solution.

Sources and instructions related to deploying and using JACQ can be found here:

<http://sourceforge.net/projects/jacq/>

http://jacq.nhm-wien.ac.at/dokuwiki/doku.php?id=export_documentation#install_jacq_system

2.2. PlutoF– online solution for biological data management and sharing of Open Data

The PlutoF platform (<https://plutof.ut.ee>) provides online services to create, manage, analyse and publish biology-related databases and projects. Platform users include natural history collections, international, regional or institutional workgroups developing common databases, individual researchers and students, as well as Citizen Scientists. PlutoF brings together, into a single online workbench, datasets that are usually spread over different solutions and therefore difficult to access or work with. The PlutoF system allows users to manage most types of biology-related data like specimens and other taxon occurrences, DNA sequences, traits, locality, habitat, projects, agents, etc in one place. Sharing, exporting and importing, and publishing your data is easy and logical. There are plenty of options to publish datasets as Open Data – data can be displayed in any portal via an API connection, Digital Object Identifiers can be requested internally, data can be released to GBIF (<http://www.gbif.org/>), etc. There are currently over 2,000 registered users from 75 countries.

The main concept behind PlutoF is to provide services where the entire data life cycle can be managed online and in one workbench (Fig. 2). The very first version of the PlutoF was built in 2001 for the specimen and associated DNA sequence datasets. These first datasets were released publicly by the UNITE community in 2003 as an online DNA sequence key for the fungi (Köljalg et al. 2005; <https://unite.ut.ee>). Since then the system has been expanded and new data types have been added to the platform. Soon after the first version was developed, natural history collections started to exploit PlutoF for their institutional databases and transactions. Early users also included ecologists, taxonomists and Citizen Scientists bringing different datasets into the system. The first online PlutoF workbench was released in 2005 (Abarenkov et al. 2010).

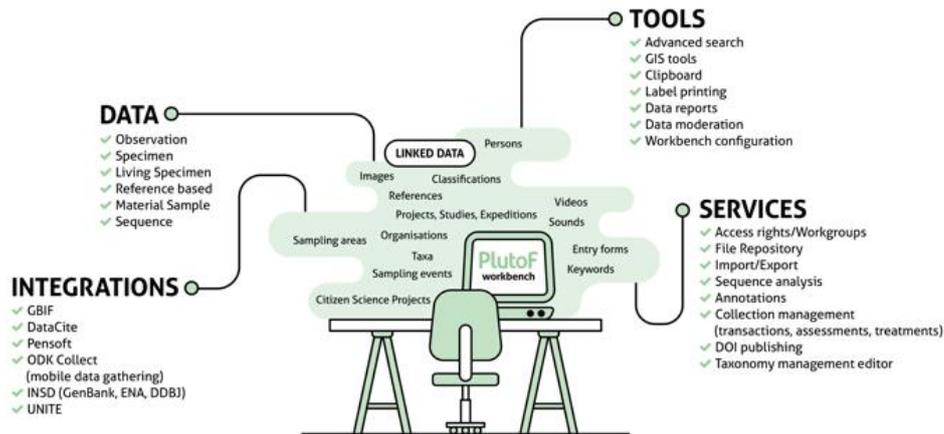


Figure 2. Schema demonstrating linked data, tools and services of PlutoF.

The current version of PlutoF incorporates several modules (**Fig. 2**), which support the creation and management of databases across disciplines. Specimen data in collection databases are available for external annotation with DNA sequences, new identifications, traits, multimedia, references, etc. Data can flow in the other direction when databased specimens of a specific study are lodged in a collection. Only ownership and location of the specimen must be updated. The same applies to environmental samples. The platform supports projects that develop databases of taxon occurrences covering different Kingdoms, their interactions and traits.



Figure 3. Data entities and services of the PlutoF platform.

PlutoF provides specific tools for third-party annotations of different datasets from external databases. One such example includes developing regional reference-based taxon checklists where taxon occurrences from published literature can be complemented with diverse geographical and ecological information. The UNITE community is using these tools to annotate and improve the quality of fungal ITS sequences in the International Nucleotide Sequence Databases (INSD: GenBank, ENA, DDBJ). The local PlutoF copy of the INSD dataset is updated on a regular basis. Any third-party annotation added to the INSD dataset (e.g. locality, habitat, source, traits, taxon identifications, and interacting taxa) is made publicly available to the research community through web services and on the UNITE homepage (Nilsson et al. 2016). Other PlutoF communities and users may start their own projects where external datasets are imported and annotated.

There are specific modules on the workbench to help users with importing their data from CSV files, exporting in various formats (e.g. CSV, JSON, PDF for specimen labels, FASTA for DNA sequences), and displaying data on maps.

PlutoF supports Open Data and data publishing in various ways – support for Digital Object Identifiers is provided by a direct link to DataCite (<https://www.datacite.org>), publishing to GBIF can be set up on demand, and publishing in Pensoft journals (<http://www.pensoft.net/>) is made easy through import options in the ARPHA writing tool (<http://arpha.pensoft.net/>) and automated creation of Ecological Metadata Language (EML) formatted metadata for datasets.

The PlutoF platform is built using the Django REST Framework (DRF) and Ember.js. The database management system is based on a PostgreSQL and PostGIS database. The public RESTful web services are provided by DRF. Software packages for the analysis module are written in the Perl and Python programming languages.

During the EU BON project, a Citizen Science module was developed, and improved functionality for the mobilization of collection (living) specimen data were developed (**Fig. 4**).

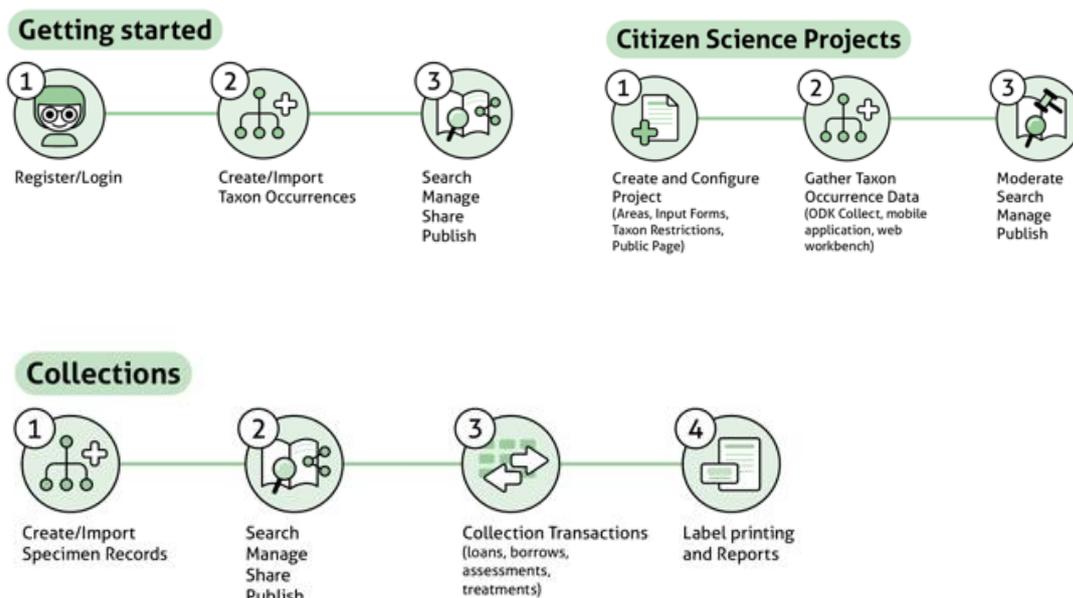


Figure 4. Workflows showing how to: a) start with PlutoF; b) develop Citizen Science Projects and c) manage collection databases.

2.3. DINA – Digital information system for natural history collections

The [DINA project](#) develops an open-source Web-based information management system for natural history data. At the core of the system is support for assembling, managing and sharing data associated with natural history collections and their curation. Target collections include zoological, botanical, geological and paleontological collections, living collections, biodiversity inventories, observation records, and molecular data. DINA is primarily intended for large installations servicing the collection data management needs of a country, a region, or a large institution.

DINA is developed by the DINA consortium, an unincorporated international partnership among organizations and individuals for collaborative open-source development. The DINA consortium was founded in 2014 by six natural history collection institutions in Europe and North America. Many of these institutions are coordinating national or regional consortia of institutions, and the total number of collections and institutions covered by the DINA consortium is substantial. The DINA has its roots in a Swedish initiative to replace a heterogeneous collection of unsustainable in-house databases with a modern, web-based national collection management system. The consortium is open to additional members as detailed on the [DINA project web site](#).

Several DINA consortium partners now have hybrid systems in production, which are based on a combination of Specify 6/7 with components developed within the DINA consortium. These hybrid systems are known as DINA-Specify systems. Public DINA interfaces to collections data currently in production at the Swedish Museum of Natural History (NRM) in Stockholm include [Naturarv](#) ("Natural Heritage" - collection portal), [Swedish DNA key](#) (public DNA barcode portal), and [Naturforskaren](#) ("The Naturalist" - species profile pages [in Swedish]; for an English version of the latter system (with very little content), see [The Naturalist](#). Agriculture and Agri-Food Canada, another DINA partner, uses Specify 6 in combination with a separate system for managing DNA sequence data, SeqDB, which is currently deployed also at NRM.

The next generation of the DINA system, DINA-Web, will be a fully integrated and entirely web-based collection management system that is independent of Specify components. The first complete version of DINA-Web is slated for release in 2018. Various DINA-Web components and APIs will be made available in test versions during the period leading up to the release of the full system. See the [DINA Technical Committee](#) page on DINA project wiki and the [DINA-Web github repository](#) for additional information about DINA-Web development.

Within the DINA context, EU BON Task 1.4 efforts have primarily focused on systems integration and facilitation of deployment (Section 3.1), but we have also developed and tested data mobilization tools targeting the DINA and Specify communities (Section 4.10).

3. Integration of systems

Biodiversity informatics is an active field and various software tools for data management, mobilization and digitization tasks are continuously being developed and contributed by different teams around the world. For many reasons, the technologies used vary considerably across teams, resulting in a mixed bag of software systems written in different programming languages and having different external dependencies. To build sophisticated integrated systems that reach beyond what a single team can accomplish, we need to be able to combine these heterogeneous tools into a system of systems that is easy to deploy and operate, and in which the different components work efficiently together.

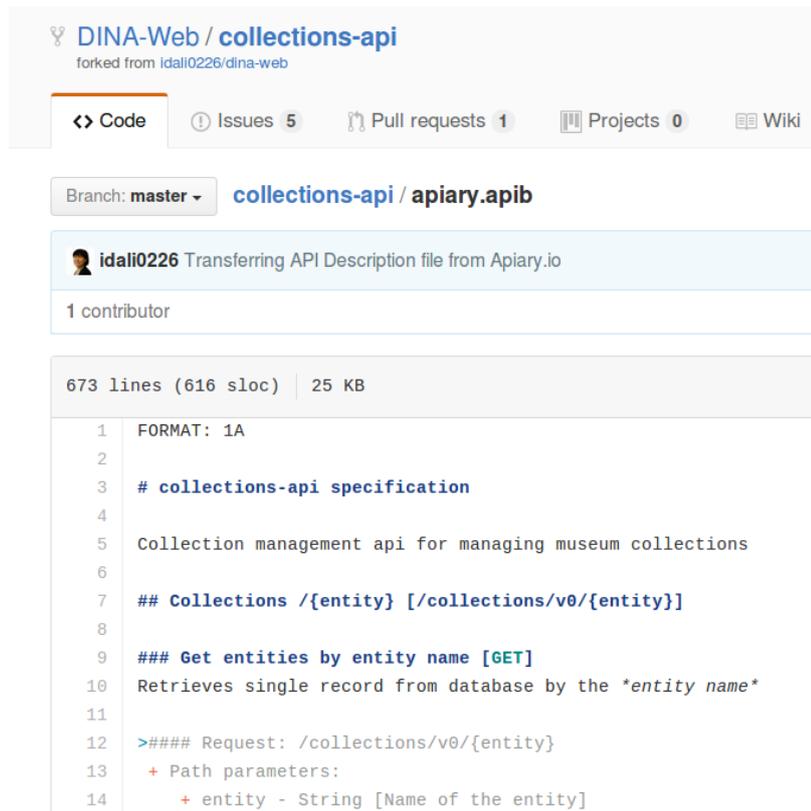
The development of [services oriented architectures](#) represents a crucial step forward in facilitating systems integration. By clearly separating the outward-facing [application programming interface](#) (API) of each component from its inner workings, it is possible for individual programmers or single teams to work independently on the implementation of a module without breaking the contract (API) on which other modules rely in a larger system of systems. Thus, stable, well-designed, and well-documented APIs is one cornerstone in facilitating systems integration. Within EU BON target 1.4, we have explored the [Apiary](#) design stack in particular for facilitating the development and documentation of such APIs.

Another challenge in systems integration is the external dependencies in the development and deployment environments used for different components of a larger system. In EU BON task 1.4, we have focused on exploring both [Vagrant](#) and [Docker](#) as potential solutions to these problems. Taxonomic concepts are essential in linking information content across systems and system components. To facilitate stable taxonomic linking within a larger biodiversity management system, and between such a system and external taxonomic authority systems, we explored two approaches. First, we developed the taxonomy module of PlutoF into a standalone web-service component that could be easily deployed as part of a larger system. Second, we explored the possibility of integrating external taxonomic authorities in large biodiversity information management systems using GBIF software.

We also worked with persistent identifiers, which are important in providing stable links in general across systems or system components. Finally, we explored integration of biodiversity data management systems with external annotation systems, particularly [AnnoSys](#), allowing outside users to provide comments about data shared openly online.

3.1. Apiary for API blueprints

The [Apiary](#) tool chain allows developers to easily build so-called blueprints documenting the APIS, and facilitating high-level design of a modular, services-based system. In the blueprints, the APIs can be specified using markdown - a simple text format. For an example of an Apiary blueprint, see <https://github.com/DINA-Web/collections-api/blob/master/apiary.apib>, which describes and defines the Collections API of DINA-Web (**Fig. 5**).



```
1  FORMAT: 1A
2
3  # collections-api specification
4
5  Collection management api for managing museum collections
6
7  ## Collections /{entity} [/collections/v0/{entity}]
8
9  ### Get entities by entity name [GET]
10 Retrieves single record from database by the *entity name*
11
12 >#### Request: /collections/v0/{entity}
13 + Path parameters:
14 + entity - String [Name of the entity]
```

Figure 5. API blueprint specification for the DINA-Web Collections API.

Based on the blueprint, various Apiary tools can be used to automatically provide mock-up web services, which provide minimalistic functionality based solely on the information expressed in the blueprint (.apib file). Such mock-up web services allow rapid prototyping of front-end components and other clients that use the API, before the back-end implementation has even started. This greatly facilitates the division of responsibilities among collaborating development teams. As long as the contract specified in the blueprint is honoured, teams working on each side of the interface can proceed independently of each other, using different technology stacks and programming languages, and following their own timetables.

Apiary tools use blueprints to support the rendering of other useful resources, such as portable, standalone HTML reference documentation of the API. An example of such Apiary-generated HTML reference documentation is provided for the DINA-Web Collections API here:

<https://rawgit.com/DINA-Web/collections-api/master/collections-api-reference.html>

(**Fig. 6**). With a well-documented REST API exposing the data services, it is easy for developers to then read and write data to the system using any programming language.

The screenshot displays the Apiary-generated HTML documentation for the DINA-Web Collections API. On the left, a sidebar lists various actions: 'Get entities by entity name', 'Create a new entity', 'Edit an entity', 'Collections /{entity}/{id}', 'Get entity by id', 'Delete an entity', 'Get list of entities from databas...', and 'Get total count of the entity from...'. The main content area is titled 'collections-api specification' and describes the 'Collection management api for managing museum collections'. It features a 'Resource Group' section for 'COLLECTIONS /{ENTITY}'. A GET endpoint is shown for '/collections/v0' with the description 'Retrieves single record from database by the entity name'. The request path is '/collections/v0/{entity}'. Parameters include: Path parameters (entity - String [Name of the entity]), and Query parameters (offset - Paging offset [int Default = 0], limit - The maximum number of records to return [int Default = 50], minid - The minimum id to retrieve [int Default = 0], maxid - The maximum id to retrieve [int], sort - Sort order [ASC or DESC Default = ASC]). An example URI is 'GET /collections/v0' and the response is '200'. A 'Show' button is visible at the bottom right of the endpoint details.

Figure 6. Apiary-generated HTML documentation for the DINA-Web Collections API.

In DINA-Web, a front-end component for collections management on the Web was implemented using EmberJS with source code available at <https://github.com/dina-web/collections-ui> and a reference implementation deployed at <https://beta-cm.dina-web.net/>. This graphical user interface (**Fig. 7**) consumes the DINA-Web Collections API specified in the Apiary blueprint.

In conclusion, Apiary allowed rapid prototyping of components in the DINA-Web system and facilitated both, effective collaboration within and among teams. Apiary blueprints specified in the simple Markdown format also provided a simple but powerful way of generating the documentation of APIs needed for users, developers and system architects. There are other tools for API documentation, such as Swagger, but our conclusion is that Apiary provides one of the most attractive options currently for prototyping and documenting the APIs of large, modular biodiversity information management systems.

DINA Collection manager

Start Collection object ▾

Collection manager beta

Collection manager beta is a test site to evaluate the new collection manager. The site will be continuously updated with new features and fixes based on the internal priority and feedback from user testing.

Information

The site will be updated with the latest available release and you can find a release log in the "Release log" column. To get started click on *Collection object* and choose *Register new*

DINA Samling

Start Samlingsföremål ▾

Lista
Registrera föremål

Current sprint

Main focus in the current sprint will be extra fields for Collecting event and Locality and fixing minor bugs reported in the feedback form.

Feedback

Use the feedback form if you find anything that is not working correctly or if you have suggestions on improvements.

Sign in to leave feedback.

Release log

2016-06-23

New features

- Search for first and last name when searching for determiner and collectors.
- Display birth and death date when searching for determiner and collectors.

Fixes

- Renamed comments to "Internal" and "External".
- **Botany:** Change label for "Object description" and "Habitat/Substrate"

2016-06-03

New features

- Private comments
Possible to add private comments.
- Limit the number of extra fields to one of each kind (Private, Regular, Verbatim)

Figure 7. Collection manager user interface consuming the DINA-Web Collections API.

3.2. System integration and deployment using Vagrant

[Vagrant](#) enables users to create and configure lightweight, reproducible, and portable development environments and deployments. This is one of the most difficult challenges facing a community developing large and complex, modular software systems. Within EU BON, our testing of the Vagrant approach involved packaging the DINA-Web components into a server that could be deployed in a range of different providers, either inside existing local IT infrastructures, using [VirtualBox](#), or in the cloud using third party services like Amazon Elastic Cloud.

A range of component from DINA-Web were included, such as:

- naturarv (<https://www.dina-web.net/naturarv>)
- loan (<https://www.dina-web.net/loan>)
- loan-admin (<https://www.dina-web.net/loan-admin>)
- inventory (<https://www.dina-web.net/inventory>)
- dna-key (<https://www.dina-web.net/dnakey/>)

A number of lessons were learned from using this delivery method. Firstly, good sample datasets were required and not easily available in a format that provided an appropriate amount of data (not too little or too sparse, not too much or too rich) for demonstrating its usage in a suitable way. Time was spent on packaging sample data, using appropriate licenses and this work was made available at <https://github.com/dina-web/datasets>.

Also, the software components needed to be made available through a build process so that they could be packaged into a single system using Vagrant. This work is available here: <https://github.com/dina-web/modules>. Preferably, a delivery method should be set up allowing continuous deliveries of these

build artefacts so that the integrated system can be evolved iteratively, step by step. Early approaches involved using Jenkins and similar tools to achieve this. Currently, cloud services such as Travis CI is the preferred build mechanism due to its simplicity - just add a `.travis.yml` text file in the source code repository to use it. Travis CI is also convenient to use due to the excellent integration with services like GitHub Releases.

One lesson learned in this respect is that GitHub is targeting primarily source code and has limitations when it comes to handling binaries or building artefacts (especially files larger than 50 MB) along with regular source code text files. To solve these problems, GitHub provides Release functionality, but there are alternative ways to work that may be more suitable, such as using Docker and Docker Hub for managing binary images and deploying “containers” (see below).

A second lesson concerned support services. The core components packaged into a Vagrant instance relied on a range of support services, and it involved a fair amount of work to identify the best set of auxiliary components that would support the core functionality in a stable way without introducing too many dependencies on third party services that were associated with license costs or configurations that were difficult to reproduce or bundle into the full package in a straightforward way. Examples of such auxiliary components include email services, traffic analytics, and backup and archive services. In the end, we were able to provide all of these services through open source stacks that allow use, modification and redistribution based on open source licenses.

Finally, in our experience, the process of capturing all configuration settings and then building and rebuilding Vagrant images fast and reproducibly is a lengthy and sometimes finicky process. The end result is a “monolith” in the sense that changing an individual component and then integrating that change in the overall Vagrant instance is sometimes quirky and requires a lengthy rebuild step. This type of work would be greatly facilitated by something akin to a version control system like “git”, which would capture minimal change sets and effectively provide versioning of the entire system based on successive commits of system components. This type of functionality is provided by Docker (see next section).

3.3. System integration and deployment using Docker

[Docker](#) is a system that wraps a piece of software in a complete file system that contains everything needed to run it, resulting in a so-called “container”. A complex system based on micro-service architecture can be packaged into several different containers, providing much better flexibility and avoiding the monolithic nature of Vagrant images. The containers can be changed independently of each other, and they can be combined into larger systems using fast and convenient tools such as Docker Compose. Docker also provides excellent support for building and (re)combining independent units based on what is called “layers” - essentially binary “commits” - that can be recombined and reused, much like using LEGO pieces.

Docker build tools can be provided in the form of source code projects. DINA-Web provides such a repository, which checks out all relevant components in DINA-Web (the modules): <https://github.com/DINA-Web/bob-docker>. This repository is planned to add also building, testing and release of the whole system, something that happens individually for each module already.

The modules in DINA-Web are provided as repositories with a “-docker” suffix. Each of these repositories contains a Makefile that bundles the software components and packages them as portable Docker containers. See the section later in this document about services provided to the community for a listing of the DINA-Web modules provided currently as Docker containers.

These infrastructure developments are now opening up for biodiversity informatics teams around the world to contribute plug-and-play modules to the DINA-Web ecosystem. In order to achieve harmonization across contributed modules, a set of guidelines have been developed to help in the design, implementation, release, quality assurance, and development of API functionality among other things (see <https://github.com/DINA-Web/guidelines>). They also provide instructions for harmonizing the look and feel of user interface components.

Specifically, for a software component to be packaged and integrated into DINA-Web as a compliant module, it needs to provide an open source license, a README file with an introduction explaining usage and giving module-specific information, a change log (or even better a full commit history in GitHub capturing all changes since the start of module development), a docker-compose.yml file specifying the module dependencies, and a Makefile that provides targets to actions for building, running and testing the module. If the module provides an API, there should be an up-to-date API blueprint and appropriate API documentation. Releases should be versioned and available at GitHub Releases and/or Docker Hub.

In terms of files, this means that source code repositories for DINA-Web modules often include (depending on type of module):

- LICENSE # open source license
- README.md # explain usage
- CHANGES.md # latest changes
- docker-compose.yml # composition of micro-services
- Dockerfile # definition of portable images
- Makefile # automation: various VERBs for building, testing, starting/stopping services etc
- .travis.yml # continuous integration, providing delivery of build artefacts to GitHub Releases and Docker Hub
- Apiary.apib # API specification
- Api-documentation.html # rendered apiary blueprint as HTML documentation

3.4. Taxonomic integration

Taxonomic concepts play a key role in organizing and linking biodiversity data within and across systems. This means that all biodiversity information management systems need to handle taxonomic information. Increasingly, authoritative source of taxonomic information are available online. At least in the short term, this does not eliminate the need for maintaining specialized local taxonomies, for instance taxonomies that reflect how a collection is organized. However, it means that such local taxonomies can and should increasingly be maintained by comparing and, where appropriate, copying information from more general external systems. Slowly and surely, this is likely to lead to convergence among taxonomies, but it will be a long time before we reach a global consensus classification of life, if we will ever get there. Recent progress in molecular phylogenetics and

molecular taxonomy is also creating a number of challenges in keeping taxonomic systems relevant, and making sure that they are useful both for molecular and traditional biodiversity data.

EU BON efforts in this area have focused on three different tasks. First, we have worked to provide the PlutoF taxonomy module as a standalone open-source component that could be incorporated in other biodiversity information management systems. The taxonomy module has a rich and well-documented API, and can handle multiple taxonomies behind the API. We isolated the taxonomy module web service and the corresponding back end code from the rest of the PlutoF system and packaged it in a Docker container, which is available to the community from the DINA-Web github repository (see list of services provided to the community below).

The current PlutoF taxonomy module has some limitations. The front-end used in PlutoF for taxonomy management is tightly integrated with the user interface for other parts of the system, so we have not been able to provide a separate front-end client for the module. Thus, any user of this component will have to develop their own user interface to make use of the taxonomy module web service. The source code for the standalone PlutoF taxonomy module is maintained in a separate branch of the PlutoF code repository, which is maintained separately from the main development branch. At the time of this writing, it is unclear whether other back end or front end components of the PlutoF system will be released as open-source modules for incorporation in other biodiversity information management systems.

A second effort has focused on the GBIF ChecklistBank system. ChecklistBank is the main aggregator of biological taxonomies; it currently contains more than 16,000 checklists, the information in which is provided through a uniform, stable and well-documented API. Among the checklists provided by GBIF, we find virtually all of the major and widely used external reference taxonomies, such as Catalogue of Life, World Register of Marine Species (WORMS), and the National Center for Biotechnology Information (NCBI) taxonomy. ChecklistBank also contains tools for cleaning checklists and for automatically constructing a single consensus classification from them, a taxonomic backbone. Clearly, this is a feature set that would be useful in managing also local taxonomies in any biodiversity information management system.

The primary aim of the EU BON effort has been to work together with GBIF in providing ChecklistBank as a standalone Docker component that could be incorporated into relevant larger systems. At the time of this writing, we have made good progress in this direction, and we expect that we will be able to provide a first version of a standalone ChecklistBank web service component as a Docker container from the DINA-Web GitHub repository in the spring of 2018.

EU BON resources have also been used to develop and implement automated mechanisms for harvesting new taxonomic information from the scientific literature, to help keep taxonomic reference systems up to date. An estimated 17,000 new species descriptions are published each year; in addition, many species are synonymized or affected by other actions that result in name changes. Plazi is providing a system that continuously mines scientific publications for text and data tied to new species descriptions and other taxonomic actions. Among other things, the extracted information is used to keep the EU BON taxonomic backbone (D1.4) updated. Names are also provided to GBIF, where Plazi is now one of the main name providers. Plazi's coverage is at the moment around 25 % of all new taxonomic names published in the scientific literature.

3.5. Globally unique and persistent identifiers

An important challenge in integrating biodiversity information management systems is to ensure that objects can be unambiguously identified through globally unique, persistent identifiers. Such identifiers will ideally allow users to find images, websites, and other metadata of particular biodiversity data objects of interest. Importantly, it also allows the construction of large Linked Open Data (LOD) clouds, which can be used for information mining, data analysis, and many other tasks.

In recent years, the Consortium of European Taxonomic Facilities (CETAF) has proposed the use of Universal Resource Identifiers (URIs) as persistent identifiers in the biodiversity informatics community. To date, 13 CETAF institutions have joined the initiative and provide LOD-compliant identifiers for individual specimens (<http://cetaf.org/cetaf-stable-identifiers> see also http://wiki.pro-ibiosphere.eu/wiki/Best_practices_for_stable_URIs). This initiative provides the community with mechanisms for consistently referencing individual specimens, as well as the means for redirecting information requests to human-readable webpages and machine-readable (preferably RDF) metadata records, as appropriate.

Within EU BON target 1.4, the [CETAF Stable Identifiers](#) have been implemented throughout the JACQ network. This means that JACQ records can now be presented as LOD with persistent identifiers, which can be used as a mechanism for citing specimens and resources in semantics-aware biodiversity informatics infrastructures.

To equip JACQ with CETAF-identifiers, a PHP module for handling the service communication between JACQ and a web service for revolving the URI was developed. In a first prototype the BGBM started to implement this feature for the Herbarium Berolinense (BGBM), because at BGBM a resolver web service for BGBM URIs was already in place. The BGBM added the necessary fields for creating the URI on the fly following the guidelines provided by the CETAF identifier initiative. In addition, BGBM supports institutions within JACQ who want to use stable URIs by providing the BGBM resolver web service as long as they do not have their own instance in place. Both source code and example documents for the stable identifiers development are accessible from <https://sourceforge.net/projects/stablecollectionidentifiers/>.

Plazi is minting globally unique identifiers for taxonomic treatments using a UUID embedded in a http URL. The UUID is synchronized with those used for nomenclatural acts in ZooBank. They are included in the Darwin Core Archives submitted to GBIF from Plazi, making it possible for end users to cite the source taxonomic treatment for each of the Plazi occurrence records in GBIF. For extracted illustrations and articles, for which no DOI exists, DOIs are minted by uploading the illustrations and articles to the Biodiversity Literature Repository, a collaboration between Zenodo/CERN, Plazi and Pensoft.

3.6. Annotations

Annotations are comments provided as metadata about data. It is not uncommon in biodiversity informatics to have annotation systems be implemented as separate entities, providing a mechanism for the community to annotate information in other systems. Provided that annotations are shared across systems, this provides a good mechanism for adding value to existing data, for instance by pointing out or correcting errors, or by adding information that was not captured in the original system.

[AnnoSys](#) is such a web-based system for suggesting corrections and enriching biodiversity data in publicly available biodiversity data portals. The annotation system, which is shared across multiple data sources and portals, offers the opportunity for the community as a whole to pool expertise and benefit from the contributions of all parties. The current release of AnnoSys manages a repository for annotations and annotated records and establishes a number of workflows that enable online annotations of specimen and observation data. It is integrated into several biodiversity data portals, including the GBIF and BioCAsE portals.

Within task 1.4, we agreed that a shared annotation infrastructure for the different collection data systems – DINA, PlutoF and JACQ – should be implemented. As a first step towards such an infrastructure, AnnoSys was integrated into JACQ, and can now be extended to PlutoF or DINA.

The only prerequisite needed to add the annotation feature for the data is a standardized data endpoint. For JACQ, the BioCAsE provider software is already in place to publish data to data aggregators like GBIF, but it is also possible to use Darwin Core Archive files for annotations.

Specifically, a separate class for handling the service calls between JACQ, BioCAsE and the AnnoSys system running at BGBM was developed and integrated into the web site. While opening a specimen detail page, the AnnoSys web service is called to see if there already is an annotation for this specimen. The result is shown in the detail page. In addition, a link to add a new annotation was implemented. The link contains the AnnoSys and BioCAsE parameters for accessing the data.

As soon as an annotation is added for example from the JACQ portal, it will show up on any portal that has the AnnoSys system in place, for example GBIF. With this development, it is now possible for any institution being part of the JACQ system to add the annotation-feature just by changing the JACQ configuration table.

4. Data Mobilization Projects

4.1. Dataflos in Slovakia

The Institute of Botany of the Slovak Academy of Sciences holds a database - Dataflos - containing complete information on herbarium specimens, published and unpublished observation and occurrence records of vascular and non-vascular plants - angiosperms, algae, cyanobacteria, lichenized fungi (lichens) and non-lichenized fungi - in Slovakia, and herbarium specimen data on foreign species if they are stored in herbaria in Slovakia. The database also contains items of the František Nábělek herbarium (<http://www.nabelek.sav.sk/>). Currently, the database contains more than 120,000 records. The database is intended to allow universities and natural museums throughout Slovakia to include also their collections. For this purpose, we have used EU BON funds to develop our own desktop administration application, which will be distributed to other institutions to help them migrate their data to the system.

4.2. Mobilizing biodiversity data published in the scientific literature through Plazi

In the published scientific literature on biodiversity, especially taxonomic literature, occurrence records are the base data used to describe or annotate the knowledge of species. This data is listed in various degrees of granularity, in the ideal case including all the data known of the specimen and including a unique identifier. This data is in almost all the cases listed as strings, in a few cases in the

form of semantically enhanced data using Darwin Core as the reference vocabulary (i.e. Biodiversity Data Journal, see Section 4.7).

As part of EU BON Task 1.4, Plazi developed a conversion workflow (**Fig. 8**), which mines and extracts data from taxonomic papers that are born digital and makes them accessible in TreatmentBank, the Plazi repository of taxonomic treatments. Two import workflows exist. For semantically enhanced publications based on the Journal and Archival Tag Suit extension Taxpub (Catapano, 2010), the data is extracted by an XSLT conversion. For articles with no semantic enhancements, the workflow will process a PDF and extract all the data using GoldenGate Imagine (GGI). The algorithm and the entire workflow in GGI can be highly customized, allowing fully automatic extraction of data, including the discovery of occurrence records (Fig. 9). Occurrence records are automatically broken down into its elements, such as location, geo-coordinates, collector name, collecting date and specimen code / catalogue number.

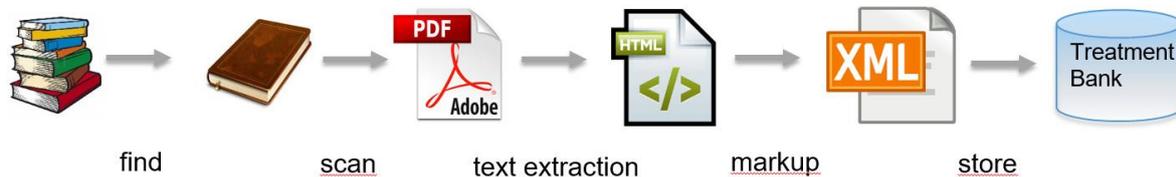


Figure 8. Plazi workflow to find and extract data from scientific articles.

The occurrence data in Plazi are automatically pushed to GBIF whenever new data have been extracted or changed. Darwin Core Archives including the treatments of one article is used for submission.

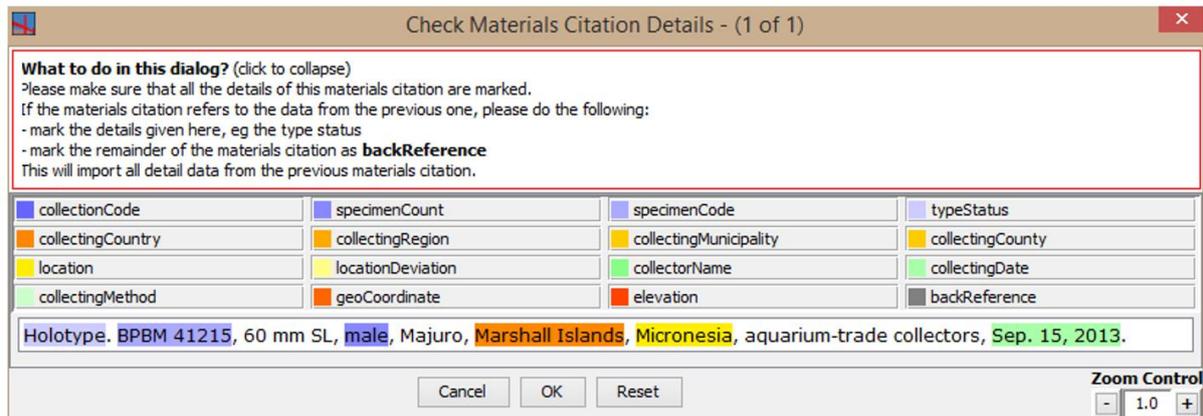


Figure 9. Occurrence extraction interface. The marked elements are suggestions and discovered automatically. They can be confirmed, removed or changed.

The extracted data is contributing to the long tail of species for which little data is available. Increasingly, the additions to TreatmentBank are taxonomic names for recently published species. Of the estimated 17,000 species described annually, TreatmentBank currently discovers and extracts information about one fourth (25 %) automatically by mining articles published in major taxonomic journals using the workflows described above. This is complemented by manual addition of articles that are published in other outlets. An important part of the digitization effort is the extraction of

illustrations of taxonomic relevance from the articles. The illustrations are linked to the taxonomic treatments and submitted to the Biodiversity Literature Repository at Zenodo, where a DOI is issued for them so that they can be cited individually. The community can participate in enhancing the markup through an online tool for occurrence records (Fig. 10) and for taxonomic names. Advanced users can edit the documents using GoldenGate Imagine (D3.3).

The screenshot shows the 'TreatmentBank' online editing interface. On the left, there is a list of treatments under the heading 'Treatment'. One treatment is highlighted: *Scinax rossaferesae* sp. nov. (Figs. 1–3). Below it, another treatment is shown: *Scinax* sp. gr. *ruber* — Crivellari et al. (2014: fig. 25). The main part of the interface is a detailed editing window for a specific material citation. The citation text is: 'Holotype.CFBH 21027, adult male, from Parque Estadual do Cerrado (24°14'47.16" S; 49°45'7.39" W), municipality of Jaguariaiva, state of Paraná, Brazil, collected on October 21, 2008 by B.V.M. Berneck, R. Recoder, M.V. Segalla, and P.H. Valdujo.' Below the citation is a table of attributes with checkboxes for each:

<input checked="" type="checkbox"/>	collectingDate	2008-10-21
<input checked="" type="checkbox"/>	collectionCode	CFBH
<input checked="" type="checkbox"/>	collectorName	B. V. M. Berneck & R. Recoder & M. V. Segalla & P. H. Valdujo
<input checked="" type="checkbox"/>	country	Brazil
<input checked="" type="checkbox"/>	latitude	-24.246433
<input checked="" type="checkbox"/>	location	Parque Estadual do Cerrado
<input checked="" type="checkbox"/>	longitude	-49.752052
<input checked="" type="checkbox"/>	longLatPrecision	1
<input checked="" type="checkbox"/>	municipality	Jaguariaiva
<input checked="" type="checkbox"/>	pageId	1
<input checked="" type="checkbox"/>	pageNumber	246
<input checked="" type="checkbox"/>	specimenCode	CFBH 21027

Below the table are input fields for '<Enter Attribute Name>' and '<Enter Attribute Value>', and a button 'Add / Set Attribute'. At the bottom are buttons for 'OK', 'Cancel', and 'Reset'.

Figure 10. Online editing interface at TreatmentBank. Occurrence records (Materials Citations) can be marked up, the system analyzes the text string and provides suggestions that can be accepted, rejected and changed, as well as elements not assigned a particular meaning added.

Legal aspects pertaining to the Plazi workflow have been discussed by Agosti and Egloff (2008). Patterson et al. (2015) discuss the legal aspects of making the data types linked to taxonomic names openly available, and Egloff et al. (subm.) cover illustrations from the same perspective.

4.3. Mobilizing citizen science data through GlueCAD apps

Since 2012, EU BON partner GlueCAD has provided system management, data maintenance and development of the [Israel Butterflies Observations Portal](#), which facilitates the collection of butterfly observation reports from volunteers. The portal includes one database for sporadic observations and one for the systematic inventory data originating from the Butterfly Monitoring Scheme (BMS-IL). GlueCAD manages and coordinates BMS-IL, with prime efforts focused on recruiting and training communities of volunteers, enhancing data generation and mobilization, and creating long-term sustainability for the [Israeli Systematic Butterfly Monitoring Scheme \(BMS-IL\)](#).

As part of the EU BON effort in Task 1.4, GlueCAD worked with GBIF to publish the data from the BMS-IL using the GBIF Internet Publishing Toolkit (IPT) while developing and testing the new Darwin Core (DwC) standard for monitoring data and the support for it in IPT. The process of

mapping the BMS-IL data to the sample-based DwC standard involved intense discussion and development to ensure that appropriate metadata characterizing the monitoring program could be captured and stored in a format allowing interoperability with other data sources. The work, which involved cooperation between GlueCAD (Israel Pe'er) and the UFZ (Guy Pe'er, a scientific advisor of BMS-IL), allowed the GBIF team (Kyle Braak) to improve the capacity of GBIF to accommodate, handle and map the metadata from systematic monitoring efforts. This facilitates the discovery of the best monitoring data for a particular scientific analysis according to the specific needs of that analysis, including information on the sampling methods and sampling efforts used in different monitoring schemes. The BMS-IL data are now publicly available via the [EU BON IPT](#) and through the EU BON portal using the new DwC format.

Aiming to facilitate the use of mobile devices for the collection of reliable information from citizen scientists and volunteers, GlueCAD also developed and provided two different mobile phone apps as part of EU BON Task 1.4: *'I Saw a Butterfly'* for opportunistic butterfly sighting reports (**Fig. 11**) and *'BMSapp'* for systematic monitoring data (**Fig. 12**). *'I Saw a Butterfly'*, now available freely on Google Play, is based on the design concept of obtaining the maximum amount of data with minimum typing, thus allowing volunteers to focus on observing rather than typing.



Figure 11. The “I Saw a Butterfly” app for collecting spontaneous observation records.

In cooperation with the PlutoF development team in Tartu, the “I Saw a Butterfly” app was modified to communicate with the PlutoF-API so that it could push observations to PlutoF, GBIF and EU BON. Based on the PlutoF and GBIF taxonomies, the app was enhanced so that users can report observations using either the list of butterflies from Europe, Estonia or Israel.

The BMSapp, started for butterflies, was extended to support any transect-based systematic monitoring for any taxa. It is currently being used not only by the Israeli BMS-IL but it is also being tested by the EU BON partner INPA-[PPBio](#) in Brazil for monitoring Western Amazonian frogs using both photos and voice-calls for identification.



Figure 12. The BMSapp for collecting monitoring data.

4.4. Mobilizing Danish biodiversity data

The UCPH efforts in EU BON Task 1.4 focused on implementation of Specify as the national standard CMS for all Danish natural history collections, including customization of the system and its user guides, and training of curators at each of the participating collections. The institutions involved in the effort include: Natural History Museum of Denmark (Copenhagen), Naturhistorisk Museum (Aarhus), Naturama (Svendborg), Museum Sønderjylland (Gram), MUSE®UM (Skive), Museum Mors (Nykøbing, Mors), Østsjællands Museum (Faxe), Fiskeri og Søfarts Museum (Esbjerg).

The main use of EU BON funds has been in the mobilization of collections data from in-house legacy systems by migrating them to Specify. Data normalization has been a major component of the migration work given that the legacy systems were not data-standards compliant. A major goal has been reached in that all migrated data have been made available for direct publication to GBIF, and through GBIF to EU BON.

4.5. Pensoft mobilization efforts

Pensoft mobilization efforts under EU BON Task 1.4 have focused on three different activities. First, tools have been developed to ensure that data can be imported from some common biodiversity data platforms directly into taxonomic manuscripts referring to such data. Second, mechanisms have been implemented to ensure that specimens that are referenced in biodiversity publications can be linked backed to the repositories holding them. Finally, tools have been developed for convenient, semantically enhanced publication of species conservation information.

Import of occurrence records directly from databases into manuscripts. Repositories and data indexing platforms, such as GBIF, BOLD systems, iDigBio, or PlutoF, hold, among other types of data, specimen or occurrence records. Thanks to EU BON support, it is now possible to directly import specimen or occurrence records into Pensoft's ARPHA publishing platform and create taxonomic manuscripts from these platforms (see **Fig. 13**).



Figure 13. Workflow for directly importing occurrence records into a taxonomic manuscript.

Until now, when users of the ARPHA writing tool wanted to include occurrence records as materials in a manuscript, they would have had to format the occurrences as an Excel sheet that is uploaded to the Biodiversity Data Journal, or enter the data manually. While the “upload from Excel” approach significantly simplifies the process of importing materials, it still requires a transposition step – the data which is stored in a database needs to be reformatted to the specific Excel format. With the introduction of the new import feature, occurrence data stored in the GBIF, BOLD systems, iDigBio, or PlutoF platforms can be directly inserted into the manuscript by simply entering a relevant record identifier.

The functionality shows up when one creates a new “Taxon treatment” in a taxonomic manuscript in the ARPHA Writing Tool. To import records, the author needs to:

- Locate an occurrence record or records in one of the supported data portals;
- Note the ID(s) of the records that ought to be imported into the manuscript (see Tips and Tricks for screenshots);
- Enter the ID(s) of the occurrence record(s) in a form that is to be seen in the “Materials” section of the species treatment;
- Select a particular database from a list, and then simply clicks ‘Add’ to import the occurrence directly into the manuscript.

In the case of BOLD Systems, the author may also select a given Barcode Identification Number (BIN; for a treatment of BINs, see below), which then pulls all occurrences in the corresponding BIN.

This workflow is illustrated by creating a fictitious treatment of the red moss, *Sphagnum capillifolium*, in a test manuscript (**Fig. 14**).

Figure 14. Direct import of occurrence records into a fictitious taxonomic treatment.

This workflow can be used for a number of purposes. An interesting future application is the rapid re-description of species, but even more exciting is the description of new species from BINs. BINs (Barcode Identification Numbers) delimit Operational Taxonomic Units (OTUs), created algorithmically at BOLD Systems. If a taxonomist decides that an OTU is indeed a new species, then he/she can import all the type information associated with that OTU for the purposes of describing it as a new species.

Not having to retype or copy/paste species occurrence records, the authors save a lot of efforts. Moreover, they automatically import them in a structured DwC format, which can easily be downloaded from the article text into structured data by anyone who needs the data for reuse. Another important aspect of the workflow is that it will serve as a platform for peer-review, publication and curation of raw data that is of unpublished individual data records coming from collections or observations stored at GBIF, BOLD, iDigBio and PlutoF. Taxonomists are used to publish only records of specimens they or their co-authors have personally studied. In a sense, the workflow will serve as a “cleaning filter” for portions of data that are passed through the publishing process. Thereafter, the published records can be used to curate raw data at collections, e.g. put correct identifications, and assign newly described species names to specimens belonging to the respective BIN and so on.

Mobilization of collection data. The publication of Mesibov (2015) made ZooKeys the first journal to use institutionCodes as a regulated, standardized data element to improve access to specimens referred to in articles. Specifically, this article was the first to use institutionCodes to link the specimens cited in the article to a record in the Global Registry of Biodiversity Repositories (GRBio) for the repository in which each referenced specimens is preserved (Fig. 15). Weijola et al. (2016) provide a more extensive example of this feature.



Figure 15. Use of institutionCodes to link specimens mentioned in publications to the repositories where they are held.

Schindel et al. (2016) introduce GRBio as an online metadata resource for biodiversity collections, the institutions that contain them, and associated staff members. The registry provides contact and address information, characteristics of the institutions and collections using controlled vocabularies and free-text descriptions, links to related websites, unique identifiers for each institution and collection record, text fields for loan and use policies, and a variety of other descriptors. Each institution record includes an institutionCode that must be unique, and each collection record must have a collectionCode that is unique within that institution. The registry is populated with records imported from the largest similar registries and more can be harmonized and added. Doing so will require community input and curation and would produce a truly comprehensive and unifying information resource.

Species Conservation Profiles. Species Conservation Profiles (SCP) are concise treatments of species based on an IUCN-approved template and controlled vocabularies for some of the species characteristics. The Biodiversity Data Journal in collaboration with IUCN SSC members created a workflow that allows for user-friendly authoring, peer-review and publication of SCP via a specially designed template in its authoring platform, the ARPHA Writing Tool (AWT). Apart from the rich editing interface, the tool provides additional functionalities including commenting, replying to comments, importing data from online resources (for example, literature references from CrossRef, PubMed, Mendeley, and occurrence records in Darwin Core format from GBIF, BOLD and iDigBio), versioning, reviewing by external parties during the authoring process, linguistic and copy-editing, building image plates and multimedia, automated technical checking, and others. At the end, the author can submit the profile to the Biodiversity Data Journal just with a click of a button, pass peer-review, and publish it as an open access citable scholarly article within days after acceptance. The publication is available in semantically enhanced HTML, PDF and machine-readable XML. Each field in the template is therefore marked as a particular kind of data, making it possible to export each species assessment directly to the IUCN Species Information Service (SIS) and avoiding duplicate work. Basically, each species assessment published in the journal is fed into the IUCN database and eventually published in the Red List with little extra work. The workflow is expected to play a

significant role in experts' engagement and creates additional incentives for researchers to contribute to the IUCN Red List by publishing new, or updating existing species profiles that can be cited and re-used as any other scholarly article.

4.6. JACQ Data Mobilization Efforts at BGBM

In the last three years, 15 herbaria joined the JACQ system and started managing their collections through this platform. As part of EU BON Task 1.4, the BGBM supported five herbaria preparing their data imports from other data sources or databases. A “staging area” was established to store botanical specimen data from external sources and prepare them for integration into the botanical objects data pool of the JACQ-System. During the import, several consistency and plausibility checks are performed: scientific taxon names, names of collectors and teams, as well as country names and their Adm1 Units (FIPS). After finishing these preparatory steps, data are ready for import into the staging area of the JACQ-System. Records in the staging area are validated by editorial (e.g. scientific names and identification) and/or technical personnel (e.g. Herbarium-specific information). After finishing validation, records can be finally imported into the JACQ specimen table. The staging area allows the import of unrevised specimen data into the JACQ system. The data entry form can be downloaded from: http://wiki.bgbm.org/dnabankwiki/index.php/Collection_Data_Form. The code is available at <https://sourceforge.net/p/jacq/legacy/>.

4.7. DINA data mobilization efforts

At NRM, mobilized datasets include collections at the Swedish Museum of Natural History in Stockholm and the Gothenburg Natural History Museum. The data include several traditional collections, which are now accessible through GBIF and EU BON, as well as through the national DINA collection web portal [Naturarv](#) (“Natural Heritage”). They also include a national DNA barcode reference library, accessible through the public DINA portal [Svenska DNA-nyckeln](#) (“Swedish DNA key”), as well as through the international BOLD and GenBank (International Nucleotide Sequence Database Collaboration) databases. Other DINA data mobilization efforts during the EU BON project period include migrations to the Specify system at UCPH (see above) and at MfN.

EU BON funds have been used to help develop infrastructure and services supporting these mobilization efforts. In particular, we developed a Python library for import of a wide range of data types into the Specify data model. For more details, see the DINA CollectionBatch Tool described in the next section.

5. Services Provided to the Community

Here, we briefly summarize some of the main services provided to the community through EU BON Task 1.4 activities. They range from complex system components of interest for system developers and engineers to simple but powerful services for end users. See also the previous section for specific data mobilization tools and services beyond the ones listed here.

Dockerized system components. As discussed above, Docker containers support a powerful way for system developers to compose complex systems from simpler building blocks. Through EU BON Task 1.4 efforts, a number of interesting and powerful biodiversity information system components are now available as Docker containers for engineers constructing large integrated biodiversity

information management systems. The current list includes the following components (check the [DINA-Web GitHub repository](#) for an up-to-date version, as the development of Docker containers is very active at the moment):

- [inselect-docker](#). Inselect is an innovative tool for image-based digitization of collection specimen data (see <http://www.nhm.ac.uk/>). Originally implemented as a Java desktop application, Inselect is now available as a Docker container, running in the browser.
- [media-docker](#). This is a general media and attachment server that supports storage and metadata tagging of 2D and 3D images, video clips, sounds, and documents.
- [naturalist-docker](#). This is an integration project that builds a number of DINA-Web components, including The Naturalist, The Media Server, Loan, Inventory, DNA Key and Naturarv.
- [collections-data-model-docker](#). Integration project for the complex collection objects data model proof of concept (next generation DINA-Web collections data model). The project uses liquibase and provides support for several database back ends.
- [proxy-docker](#). A reverse proxy based on the NGINX Reverse Proxy.
- [uptime-docker](#). A module for monitoring system status.
- [classifications-docker](#). Integration project for the PlutoF taxonomy module (see below), providing a Docker application with a set of containers, including tools for loading data into the module.
- [mediadrop-docker](#). System integration project providing containers for publishing samba shares and for S3 synchronization.
- [dbdiff-docker](#). This is a tool for generating “diffs” (difference reports) between database schemas, useful in evolving data models.
- [seqdb-docker](#). This is a container running SeqDB, a proto-DINA-Web module for managing molecular biodiversity data.
- [collections-api-docker](#). This is an integration project that bootstraps a database server with the DINA-Web Collections REST API.
- [bob-docker](#). Bob builds default components in DINA-web from scratch, resulting in an integrated CMS (as of this writing, still incomplete).
- [search-docker](#). SolR search for the DINA-Web collection manager.
- [chat-docker](#). An informal slack-like web-based chat-type collaboration space.
- [collections-ui-docker](#). Integration project to provide a collections management user interface on the Web that works against the DINA-Web Collections REST API.
- [biocase-docker](#). System integration project to provide DwC Archive and ABCD exports from DINA-Web.
- [cli-tools-docker](#). Integration project to enable data wrangling using various command line tools for DINA-Web, such as the CollectionsBatch tool (see below).
- [mail-docker](#). An integration project providing an email server based on "dmail" (short for Docker Email).

DINA CollectionBatch tool. A command-line tool (Python library) for importing, exporting, and updating batches of collection data in the DINA system. The intended audience is advanced users such as data managers, migration specialists, and system administrators. It is built on top of the Python libraries peewee and pandas, and it is optimized for handling large datasets. Requires no prior knowledge of SQL and little knowledge of Python. More information and source code at <https://github.com/jmenglund/CollectionBatchTool>.

Mirroreum. A web-enabled platform for reproducible open research. Mirroreum is a platform for Reproducible Open Research and includes various products and software tools produced in the EU BON project. It provides a platform for authoring and sharing reproducible open research (data, tools and results) on the web. More information at <https://github.com/raquamaps/mirroreum>.

PlutoF specimen data mobilization tool. Easy to use import tool for the mobilisation of specimen data. Data can be imported using custom template files (<https://plutof.ut.ee/#/import>), and are fully manageable through the PlutoF cloud after upload. Import tool is tested for the management of medium-sized and private collection data. Uploaded specimen data can be automatically released to GBIF, published with DOI, or sent to the Pensoft journal manuscript editing tool.

DNA-based species hypotheses. Datasets for the identification of eukaryote species from any biological samples based on rDNA ITS sequences. The sequences can come from Sanger sequencing as well as from High Throughput Sequencing (HTS) projects. Datasets are available through HTS pipelines like QIIME, mothur, CREST, UCHIME, etc. They can also be downloaded for off-line analyses. The service is available at <https://unite.ut.ee/repository.php>.

PlutoF/Pensoft automated workflow. The tools supporting this workflow provides direct connection between PlutoF databases and Pensoft's ARPHA writing tool. It allows users to import their data from PlutoF databases directly into online Pensoft journal articles in a dynamic and seamless way. The services are available at <https://plutof.ut.ee> and <http://arpha.pensoft.net/>.

PlutoF taxonomy module. The PlutoF taxonomy module provides an online workbench for managing multiple biological classifications in the same system. The taxonomy module is fully implemented in the PlutoF platform. Taxon occurrences may be identified and linked to taxon names in several classifications. Additional functionalities include taxon name search, RESTful API, and importing taxon names from GBIF. The taxonomy module is an online service provided by the PlutoF platform. A stand-alone version with base support (database and web services) is also available as a separate package at <https://github.com/TU-NHM/plutof-taxonomy-module> and as a dockerized module (see above).

AnnoSys integration. BGBM provides support for EU BON partners to integrate the Annosys system into their data portals or making it possible to annotate their data via the GBIF website. The Annosys system hosted and maintained at BGBM can be used as a central repository for annotations in EU BON. More information available at <https://annosys.bgbm.fu-berlin.de/>.

JACQ. BGBM provides support for EU BON partners who want to start using JACQ. This includes the support of data preparation for imports. BGBM also provides support for EU BON partners, who want to install their own JACQ instance. For testing, the JACQ system has been packaged in Open Virtualization Format (OVF), which runs both in VirtualBox and VMWare. The image contains a Debian system, with Apache, MySql and php. In addition, the catalogue and lookup tables are filled and sample specimen records are available for testing.

Sources and instructions for deploying and using JACQ can be found here:

<http://sourceforge.net/projects/jacq/>

http://jacq.nhm-wien.ac.at/dokuwiki/doku.php?id=export_documentation#install_jacq_system

JACQ mobilisation of specimen data. A little helper for standardized data recording in the field, lab or office. The Collection Data Form (CDF) is an Excel file containing several macros for data handling and label printing. It is an easy way to import standardized data into JACQ. The data form is available online at http://wiki.bgbm.org/dnabankwiki/index.php/Collection_Data_Form.

6. Summary - Looking Ahead

Natural history collection institutions are facing formidable challenges in mobilizing biodiversity data. A large portion of the specimen data is still not available digitally. At more and more institutions, high-throughput approaches are being used to digitize this backlog, generating large quantities of data. At the same time, natural history museums are becoming increasingly involved in the generation of large amounts of molecular biodiversity data using new massively parallel sequencing platforms, both through research projects and participation in environmental monitoring programs. Against this backdrop, the goal of EU BON Task 1.4 has been to support data mobilization efforts targeting collection-based and molecular data, mainly through the development and integration of innovative open-source tools and services.

The activities have involved work within the context of three major projects: i) DINA, an open-source, modular, web-based collection management system for natural history specimen data. ii) JACQ a web-based open-access system for management of botanical (herbarium) data. iii) PlutoF, a web platform for working with traditional and molecular biodiversity research data. The task has also involved work on a number of other EU BON partner systems and services, as well as integration across internal EU BON and external biodiversity informatics resources. Finally, these systems have been used for targeted data mobilization efforts.

Within DINA, the focus has been on supporting the engineering of sophisticated biodiversity information systems through the exploration of tools supporting distributed development and a modular plug-and-play design based on services-oriented architectures. This has involved the testing and adoption of tools like Apiary for the design of Application Programming Interfaces (APIs) and Docker for systems integration and deployment tasks. A Python library for data migration to DINA was also developed and tested. Within JACQ, a number of tools were developed to facilitate deployment and data migration to the system, and the AnnoSys tool for annotation of data has been integrated. Within PlutoF, EU BON efforts focused on the development of a citizen-science module and improved functionality for the mobilization of collection (living) specimen data. A number of innovative tools were developed by Pensoft to help mobilize biodiversity data published in the scientific literature, including semantic mark-up of species conservation papers, direct import of data from a range of biodiversity platforms into manuscripts, and a mechanism for providing stable links from publications to global biodiversity repositories. Plazi implemented an automated workflow mining published scientific papers for taxonomic data, currently mobilizing 25 % of all published new names as they become available. GlueCad developed apps allowing citizen scientists reporting spontaneous observations or systematic inventory data to select target taxa and preferred data mobilization platform. IBSAS and UCPH have focused on national data mobilization efforts targeting Slovakia and Denmark, respectively.

Although the development is clearly towards increased integration of biodiversity informatics tools into larger and more sophisticated systems, it is clear that there is no one size that fits all. Nevertheless, the increasingly widespread adoption of community standards, open-source

development practises and service-oriented architectures are pushing the capability of current systems forward and facilitating tighter integration across systems. This trend is supported by the appearance of sophisticated tools enabling developers and system administrators to design and deploy complex modular systems. The adoption of the Docker approach and the increasing availability of biodiversity information system components as Docker containers is one example of how the biodiversity informatics community may benefit from this in building increasingly sophisticated biodiversity platforms in the near future.

7. References

- Abarenkov K, Tedersoo L, Nilsson RH, Vellak K, Saar I, Veldre V, Parmasto E, Proust M, Aan A, Ots M, Kurina O, Ostonen I, Jõgeva J, Halapuu S, Põldmaa K, Toots M, Truu J, Larsson K-H, Kõljalg U. 2010. PlutoF - a Web Based Workbench for Ecological and Taxonomic Research, with an Online Implementation for Fungal ITS Sequences. *Evolutionary Bioinformatics* 6:189-196.
- Agosti D, Egloff W 2009. Taxonomic information exchange and copyright: the Plazi approach. *BMC Research Notes* 2009, [2:53]. <https://doi.org/10.1186/1756-0500-2-53>.
- Ariño AH. 2010. Approaches to estimating the universe of natural history collections data. *Biodiversity Informatics* 7: 81–92. <https://doi.org/10.17161/bi.v7i2.3991>.
- Borges P, Crespo L, Cardoso P (2016) Species conservation profile of the cave spider *Turinyphia cavernicola* (Araneae, Linyphiidae) from Terceira Island, Azores, Portugal. *Biodiversity Data Journal* 4: e10274. <https://doi.org/10.3897/BDJ.4.e10274>.
- Cardoso P, Stoev P, Georgiev T, Senderov V, Penev L (2016) Species Conservation Profiles compliant with the IUCN Red List of Threatened Species. *Biodiversity Data Journal* 4: e10356. <https://doi.org/10.3897/BDJ.4.e10356>.
- Catapano T 2010. TaxPub: An extension of the NLM/NCBI journal publishing DTD for taxonomic descriptions. Proceedings of the Journal Article Tag Suite Conference 2010 <http://www.ncbi.nlm.nih.gov/books/NBK47081/>
- Egloff W, Agosti D, Kishor P, Patterson D, Miller JA (subm). Copyright and the use of images as biodiversity data. bioRxiv doi: 10.1101/087015 (preprint). Submitted to PLoS Biology.
- Kõljalg U, Nilsson RH, Abarenkov K, Tedersoo L, Taylor AFS, Bahram M, Bates ST, Bruns TD, Bengtsson-Palme J, Callaghan TM, Douglas B, Drenkhan T, Eberhardt U, Dueñas M, Grebenc T, Griffith GW, Hartmann M, Kirk PM, Kohout P, Larsson E, Lindahl BD, Lücking R, Martín MP, Matheny PB, Nguyen NH, Niskanen T, Oja J, Peay KG, Peintner U, Peterson M, Põldmaa K, Saag L, Saar I, Schüßler A, Scott JA, Senés C, Smith ME, Suija A, Taylor DL, Telleria MT, Weiß M, Larsson K-H. 2013. Towards a unified paradigm for sequence-based identification of Fungi. *Molecular Ecology* 22: 5271-5277.
- Kõljalg U, Larsson K-H, Abarenkov K, Nilsson RH, Alexander IJ, Eberhardt U, Erland S, Høiland K, Kjøller R, Larsson E, Pennanen T, Sen R, Taylor AFS, Tedersoo L, Vrålstad T, Ursing BM. 2005. UNITE: a database providing web-based methods for the molecular identification of ectomycorrhizal fungi. *New Phytologist* 166: 1063-1068.
- Mesibov R (2015) A new genus and species of dalodesmid millipede from New South Wales, Australia (Diplopoda, Polydesmida, Dalodesmidae). *ZooKeys* 517: 141–8. <https://doi.org/10.3897/zookeys.517.10187>.
- Nilsson RH, Wurzbacher C, Bahram M, Coimbra VRM, Larsson E, Tedersoo L, Eriksson J, Ritter CD, Svantesson S, Sánchez-Garzía M, Ryberg M, Kristiansson E, Abarenkov K. 2016. Top 50 most wanted fungi. *MycoKeys* 12: 29-40.

- Patterson DJ, Egloff W, Agosti D, Eades D, Franz N, Hagedorn G, Rees J, Remsen DP. 2014. Scientific names of organisms: attribution, rights, and licensing. *BMC Research Notes* 2014 7:79. <https://doi.org/10.1186/1756-0500-7-79>.
- Schindel D, Miller S, Trizna M, Graham E, Crane A (2016) The Global Registry of Biodiversity Repositories: A Call for Community Curation. *Biodiversity Data Journal* 4: e10293. <https://doi.org/10.3897/BDJ.4.e10293>.
- Weijola V, Donnellan S, Lindqvist C (2016) A new blue-tailed Monitor lizard (Reptilia, Squamata, Varanus) of the *Varanus indicus* group from Mussau Island, Papua New Guinea. *ZooKeys* 568: 129–154. <https://doi.org/10.3897/zookeys.568.6872>.