Phylotastic: improving access to tree-of-life knowledge with flexible, on-the-fly delivery of trees

Van D. Nguyen[1*], Thanh H. Nguyen[1†], Abu Saleh Md. Tayeen[1‡], H. Dail Laughinghouse IV[2,3§], Luna L. Sánchez-Reyes[4¶], Enrico Pontelli[1‖], Dmitry Mozzherin[5], Brian O'Meara[4] and Arlin Stoltzfus[6,2††‡‡]**

[1]*Department of Computer Science, New Mexico State University, Box 30001, MSC CS, Las Cruces, 88003, New Mexico, USA*

[2]*Institute for Bioscience and Biotechnology Research, 9600 Gudelsky Drive, Rockville, 20850, MD, USA*

[3]*Ft. Lauderdale Research and Education Center, University of Florida/IFAS, 3205 College Avenue, Davie, 33314, FL, USA*

[4]*Department of Ecology and Evolutionary Biology, University of Tennessee, 569 Dabney Hall, Knoxville, 37996, TN, USA*

[5]*Illinois Natural History Survey, Species File Group, University of Illinois, 1816 South Oak St., Champaign, 61820, Illinois, USA*

[6]*Office of Data and Informatics, NIST, 100 Bureau Drive, Gaithersburg, 20899, MD, USA*

Abstract

(1) A comprehensive phylogeny of species, i.e., a tree of life, has potential uses in a variety of contexts in research and education. This potential is limited because accessing the tree of life requires special knowledge, complex software, or long periods of training.

(2) To mitigate these barriers, the Phylotastic project access and provides phylogenetic knowledge through web-service technologies, with the aim to make it as easy to get a phylogeny of species as it is to get online driving directions. In prior work, we designed an open system of operations to validate and manage species names, find phylogeny re-

*E-mail: vnguyen@cs.nmsu.edu
†E-mail: tnguyen@cs.nmsu.edu
‡E-mail: tayeen@nmsu.edu
§E-mail: hlaughinghouse@ufl.edu
¶E-mail: sanchez.reyes.luna@gmail.com
‖E-mail: epontell@cs.nmsu.edu
**E-mail: bomeara@utk.edu
††Corresponding author.
‡‡E-mail: arlin@umd.edu

sources, extract subtrees matching a user's species list, scale them to time, and connect them with images and information from online resources.

(3) Here we report the first implementation of a publicly accessible system for on-the-fly delivery of phylogenetic knowledge, developed with the guide of user feedback on what types of functionality are considered useful by researchers and educators. The implementation currently consists of a web portal to execute a general workflow to obtain species phylogenies (scaled by geologic time and decorated with thumbnail images); more than 30 underlying web services accessible via a common registry; and code toolkits in R and Python so that others can create applications that leverage these services. These resources cover most of the use-cases identified in our analysis of user needs.

(4) The Phylotastic system, accessible via http://www.phylotastic.org, provides a unique resource to access the current state of phylogenetic knowledge, useful for a variety of cases in which a tree extracted quickly from online resources (as distinct from a tree custom-made from character data) is sufficient, as it is for many casual uses of trees identified here.

1 Introduction

As Cracraft, et al. [3] suggest, a tree of life "provides a comparative and predictive framework for all fundamental and applied biology." However, inferring a suitable phylogeny from raw data following best practices is known to be a formidable task. Yet phylogeny experts typically do not employ practices that make their phylogenies discoverable, accessible, and re-useable [4, 11, 23]. Indeed, most trees are generated for a narrow and specific purpose. Nevertheless, literature surveys show that trees are re-used in research, typically by subtree extraction from larger phylogenetic trees [23].

The prospects for disseminating tree-of-life knowledge that is up to date and curated by experts in the field via subtree extraction has increased greatly with the OpenTree project [7], which currently provides a "synthetic tree" with over 2 million species constructed by a supertree method from 819 source trees [20]. OpenTree's database contains these and many more source trees awaiting curation, all publicly available online via a web-service interface. Web services are software that is available over the internet and capable of interacting with other software (by using standardised machine-to-machine communication protocols). Web services are very useful because they facilitate automated and large queries over online databases, and allow flexible integration of steps resulting in workflows that can be tailored to users need on the moment. The downside is that this is possible as long as you have the programming skills necessary to handle them. Web services are the baseline of well known phylogenetic tree querying services such as TreeBASE [18] and Phylomatic [24], and most data bases striving to facilitate access to information, including but not limited to taxonomies [2, 17], occurrence records from GBIF (Global Biodiversity Information Facility) [5] or iNaturalist, and images and other information (size, habitat, etc) from the Encyclopedia of Life (EOL) [16].

While currently available services make great strides toward making the "hard won

results of phylogeny generation available to everyone" [22], the lack of a convenient delivery system remains a significant barrier. One way to overcome this barrier is to the "Phylotastic" design suggested previously [22], which leverages an open system of web services to support various workflows to discover, modify, and add value to trees. Here we describe the first full implementation of this design, featuring a set of more than 30 web services, a web-services registry, libraries in R and Python, and a web portal all allowing various inputs such as documents, web sites, user lists, and higher-taxon searches. We explain the functionality of these tools relative to a set of use-cases based on interviews with prospective users, and compare them to other tools currently available.

Analysis, design and implementation

Use cases

The concept for a Phylotastic system [22] grew from the research interests of scientists with expertise in phylogenetics and informatics. To ensure broader value to the community, we developed a prototype Phylotastic web portal and used it to obtain feedback (via correspondence as well as in-person interviews) from a broader range of potential users, emphasizing on researchers and educators at multiple levels who were not experts in informatics or phylogenetics. Based on this information, we prioritized the development of tools and workflows that allowed the following use-cases.

Generate a tree from a specified list of taxa. Provide a tree for a user-supplied list of species or higher taxa.

Generate a tree of N species sampled from a named taxon. Given a named taxon and a number N, provide a tree with N species chosen in some way, e.g., at random, by popularity, or by maximal diversity (taxonomic or phylogenetic).

Generate a phylo-guide from an electronic resource. Create a tree with images and links to species information from a web page or document that includes taxonomic names, such as a a document listing the species found in a park (or zoo, region, or collection), a scientific paper, or an online encyclopedia (e.g., EOL, Wikispecies).

Contextualize phylogenetic relationships. Place a given list of species in a larger tree that shows phylogenetic relationships in a broader context. Or, given a small set of taxa, generate a tree using representative species that illustrate the relationship, possibly including species from other (unspecified) taxa for context.

Integrate data or metadata with phylogeny. Given the set of species implicated by any method described above, return a tree and an associated data table integrating information or resources of interest, including images, links to information (EOL or wikipedia), or data on features such as toxicity, pathogenicity, availability of fossil data, medicinal value, conservation status, size, biogeography, or habitat.

Currently, the system we implemented covers most of these uses (see Examples and Discussion). Some desirable types of data are not available systematically (e.g., medicinal

value, pathogenicity), and some types of operations are not yet implemented in available algorithms (e.g., choosing a set of species based on both popularity and diversity).

Implemented operations and services

The types of operations identified previously for the implementation of a Phylotastic system [22] include (1) taxonomic name resolution: rectifying possible misspellings in input scientific names by matching with authorative taxonomic data bases; (2) tree retrieval: finding available trees with coverage of user-identified taxa and extracting subtrees; (3) tree scaling: assigning branch lengths to subtrees, e.g., fossil scaling; (4) tree comparison: compare subtrees; (5) taxon information and images: getting and adding data or metadata from species or higher taxa; (6) rendering a tree graphically. The expanded set of use cases identified above implicates a slightly larger set of operations, including (7) scraping names: extracting taxonomic names embedded in text and media, (8) taxon sampling: sampling members of a taxonomic group by some criteria, and (9) converting common names to scientific names. Finally, (10) list management: save, publish, access, remove or update a list of names from a user account.

These operations were translated to actual tools (services) that can be accessed and manipulated by users, that are implemented and executed via web services. In general, Phylotastic web services are designed to operate synchronously. This means that workflows are carried out on real time. One exception is the set of services to manage persistent lists, so that a list created by a client in a session may be accessed in a later session, or by a different client, enhancing the potential for reusability.

Currently we have made available more than 30 phylotastic services that fall into the categories above and are further described in Table 1 and at https://github.com/phylotastic/phylo_services_docs/tree/master/ServiceDescription. Some of these services are thin wrappers around external services, while others were developed for this project. The source code for all web services is available at WHERE??

Code toolkits: We also developed R and python packages to allow users to access the Phylotastic system with their own software and computers, using functions and methods written in the native language. Both toolkits provide access that provide access to nearly all of the categories of services described in Table 1. The rphylotastic package (https://github.com/phylotastic/rphylotastic) wraps phylotastic's web services using the R packages jsonlite [14] and httr, designed for working with URLs and HTTP calls. The package structure is standard, including function documentation, a manual and a vignette. Function names follow ropensci's style (https://github.com/ropensci/onboarding/blob/master/packaging_guide.md). Documentation and manual were generated with roxygen2 R package [19]. To evaluate rphylotastic's package performance and robustness, a set of unit tests were designed and implemented using the R package testthat [19]. Around 58% of the package code is currently covered by this test suite (https://codecov.io/gh/phylotastic/rphylotastic/tree/master/R).

The phylotastic_py package (https://github.com/phylotastic/phylotastic_py) has a main module, *phylotastic_services* composed of sub-modules implementing the different phylotastic's services. The package documentation was generated with Sphinx

http://www.sphinx-doc.org/en/1.5/index.html python documentation generator. To test the functional correctness of phylotastic_py sub-modules, a set of unit tests were implemented using Python Unit Testing Framework and deployed in Travis-CI (https://travis-ci.org/).

Web portal and server: The portal (http://www.phylotastic.org) provides a user-friendly interface to access the general phylotastic workflow (Fig. 1). Without the need for extensive time and training in phylogenetic inference methods, or access to specialized software and resources, a user may for example extract names from a web page that may contain text other than scientific names; sample some species by popularity from list of extracted names; and obtain a graphical representation of their phylogenetic relationships. The graph may be customized, and either the tree (in newick format) or the graph (in png format) can be downloaded. These and other phylotastic operations can be accessed by any user that finds the portal, meaning that no login is required to access these workflows, and that features are accessible anonymously in a session-dependent manner. A user can choose to log in using a gmail address, creating an account where lists and trees associated with a previous session will be migrated to and that will be accessible from other devices. The web portal is written in Ruby using Rails, a model-view-controller architecture for rapid design and development of robust web applications. The portal takes advantage of PostgreSQL for database management; Paperclip for managing file attachments; TwitterBootstrap, JQuery, and FontAwesome for front-end development; Devise for authentication management; Wicked PDF for PDF generation; Capybara and Minitest for automated testing; Docker and Kubernetes for containerization and deployment. The test suite covers model tests, controller tests, and interactive tests (simulated in Poltergeist, which mimics user interactions).

Web services registry and support for automated planning

The concept of an open system that can be maintained as a community resource induces additional requirements, beyond the operations necessary to address the use-cases above. Because web services provide access to data and operations from any point on the internet via standard protocols, they are an obvious choice for a decentralized community system. In many cases, there is a one-to-many correspondence between operations above and web services that exist or could exist. For example, there are multiple services for taxonomic name resolution that take names and match them against a taxonomy [2, 15].

Given a rich set of web services and a design that allows for a dynamic combination of these services, the integration of a great diversity of applications is possible. For this purpose, we implemented a *web services registry* (WSR) as a software application to register, store, discover, and execute available web services, acting as a detailed source of information on what each service does, and how to access it. External service providers and developers can *register* and *store* their web services by providing services information in a machine-readable way using special language known as WSDL (*Web Service Description Language*. An example of this can be accesed HERE. Description of the service and its components (e.g., inputs and outputs) follows a dedicated vocabulary—the Phylotastic Ontology [12] that allows web services to be *discoverable*—both for inclusion in the

pre-defined workflows used by the Phylotastic portal and for use by the workflow composition system. The WSR supports the portal and the workflow execution engine [12], producing a dynamic and responsive ecosystem of operations that allows the discovery of phylogenetic knowledge to be automated and tolerant to fault. The use and importance of the WSR in automated planning has been thoroughly described elsewhere [12, 13].

Examples

In this section we exemplify the utility of Phylotastic to facilitate workflows that yield useful phylogenies.

Birds from Yellowstone

In this first hypothetical user case, a visitor to Yellowstone National Park or a biology student on a field trip that registered the common names of birds they saw during any day can find the phylogenetic context of those birds in a matter of seconds in three simple steps: (1) translate the common names to scientific names, (2) get all scientific names of bird species that are found in Yellowstone from a public list such as the one available online (https://www.nps.gov/yell/learn/nature/upload/BirdChecklist2014.pdf), (3) get a dated phylogeny for those species, summarized from expertly curated, peer-reviewed data, without the need of genetic markers, calibrations, or previous knowledge or expertise on software used to generate and date phylogenetic trees. Finally, users can plot the tree and mark the observed species as shown in Figure 2, or perform biological analyses of interest such as computing phylogenetic diversity of the species observed, etc. The following code shows implementation of the three steps described above, using Phylotastic's R packages:

```
library("rphylotastic", "datelife")
birds_I_saw <- taxa_common_to_scientific(c("Osprey",
  "House sparrow", "Mallard duck", "American Robin",
  "Song Sparrow", "Mourning Dove", "House Wren"))
yellowstone_birds <- url_get_scientific_names(
    URL="https://www.nps.gov/yell/learn/nature/upload/
    BirdChecklist2014.pdf")
yellowstone_bird_tree <- datelife::datelife_search(
    taxa_get_otol_tree(yellowstone_birds), summary_format
    = "phylo_median")
```

Aquatic mammals

Discussion

Support for use-cases

The Phylotastic system as implemented here supports all use cases identified in our analysis and design. Its graphical interface, the Phylotastic web portal, illustrates the potential of the Phylotastic system, and is designed to serve as a useful resource for scientists, educators and students. From the portal, most use cases can be performed by following the general workflow (use cases 1, 2 and 3) . Other use cases need to be performed in stages (use case 4).

The portal can integrate some kinds of data and metadata with a phylogeny (use case 5), specifically EOL links and thumbnail images. One way to integrate other data is to input a data matrix combining relevant features and species names to the portal. Then names can be scraped out of the matrix to obtain a tree, and the original matrix and tree can be combined using a web tool to generate visualizations of trees with data, such as EvolView [6] or IToL [9].

Visual representation of use case 4, in particular contextualizing species in a larger tree is not well supported by the portal. However this can be accomplished with other graphical tools, as shown in the yellowstone example.

Addressing Previous Limitations

From the protoype implementation of Phylotastic's design a number of limitations were identified and discussed in [22]. In addition to creating a user-friendly interface (the Phylotastic web portal) that encourages dynamic community expansion (via the Web Services Registry), the current Phylotastic implementation addresses previous limitations in the following ways. 1) Rather than employing the limited resources of the MapReduce pruner sourcing the OpenTree project [7] creating easy access to this extensive data base. 2) Wrapping Global Names Recognition and Delivery services http://gnrd.globalnames.org/ into the ecosystem to resolve taxonomic names and allow users to search by and receive output as scientific or common names. 3) Eliminating the need for a user-defined source tree, thus mitigating barriers to scientists and members of the public without extensive phylogenetic training. 4) Designed and implemented the Datelife service, an open source for scaling trees. 4) Provided extensive metadata for each tree produced, crediting those individuals who produced the trees, encouraging their attribution, and providing a clear path for further investigation by researchers.

Comparison with other resources

The resources of this project can be considered broadly as a way of making tree-of-life knowledge accessible, dynamic, and responsive through both a set of tools to support the development of software (client applications including scripts), and a multi-purpose interactive tool, the Phylotastic web portal. Alternative resources currently available to

satisfy a user's interest in phylogenetic knowledge include (1) computing a tree from comparative data (e.g., sequence alignments); (2) discovery and retrieval of pre-computed trees from data bases (e.g., from TreeBASE); (3) interactive browsing of a tree of life (e.g., Open Tree of Life web portal); or (4) subtree extraction from supertrees and other large trees (e.g., Phylomatic).

Currently no resources are fully comparable to the Phylotastic project in the breadth of support for the use-cases listed above. Note that the original proof-of-concept software described previously [22] only addressed the first use-case, was not based primarily on web services, had no fault-tolerance features, and is no longer operational.

Many tools provide support for obtaining gene or protein sequences, aligning them, and inferring a phylogeny (e.g., SUPERSMART [1]). The most convenient resources provide online services with simplified interfaces (e.g., CIPRES, Phylogeny.fr). However, all of these tools require training to understand the methods of phylogenetics along with a significant investment of time. ToLWeb [10], OneZoom [21], IToL [9] and the OpenTree web portal all allow interactive browsing of a tree of life, but they do not support extracting relationships for a user-defined subset of species (except via a web-services API, requiring specialized skills, in the case of OpenTree). These resources do not provide flexible tools to satisfy the non-expert wishing to access and extract phylogeny information for the purposes indicated in the use-cases above.

The most obviously comparable resources to the web portal are Phylomatic [24] and TimeTree [8], both of which helped to inspire the Phylotastic project. Both are interactive web tools that support extraction of subtrees. The main focus of TimeTree is on providing a time-scale for evolutionary divergence based on published chronograms (time-calibrated trees) and a synthetic chronogram they have put together using those source chronograms. The Phylotastic web portal offers a greater variety of ways to specify a set of taxa, responds to the query by searching multiple services and returns the tree with the best coverage. It can also generate a dated tree using the open software DateLife.

Development priorities

Some of the features repeatedly requested by users are (1) supporting the use of common names, (2) integrating character data, (3) integrating species data that are of broad interest (e.g., medicinal value, pathogenicity, endangered status), and (4) sampling species from a taxon by popularity. Some of these features are currently being tested in the web portal, including the display of common names, and sampling by popularity (based on functionality provided by OneZoom). Development priorities also include containerization and other efforts to ensure that the portal and web services can be deployed more easily.

Availability

Code, applications and services may be discovered and accessed via the project's web home at http://www.phylotastic.org. The web portal is accessible via http://portal.phylotastic.org, and the registry is at http://registry.phylotastic.org. Source code is available under an open-source license from the GitHub Phylotastic organization (http://

www.github.com/phylotastic). At present, all resources are accessible without restriction; usage restrictions on web services may be imposed in the future if necessary to ensure that the services are broadly usable. The stable version of rphylotastic can be installed from R directly from the CRAN repository (https://cran.r-project.org/package=rphylotastic) using *install.packages(pkgs = "rphylotastic")*. Development versions are available from GitHub repository (https://github.com/phylotastic/rphylotastic) and can be installed using *devtools::install_github("phylotastic/rphylotastic")*. The Python library can be installed following the instructions at https://github.com/phylotastic/phylotastic_py.

Author contributions

VDN and ASMD implemented the portal, and contributed to design and testing, along with AS and HDL; AS and HDL analyzed requirements for the portal and web services; LLSR and BO designed, implemented and tested the DateLife web portal, DateLife web services, and the rphylotastic package; ASMD and THN implemented web services, and contributed to design and testing along with AS; ASMD designed and implemented the phylotastic_py package; DM designed and implemented improvements to taxonomic name resolution services; AS, BO and EP conceived of the project and oversaw all aspects of design and testing; AS drafted the manuscript, which was completed with the help of the other authors.

Acknowledgements

This work was supported by funding from the US National Science Foundation (award 1458572, "Collaborative Research: ABI Development: An open infrastructure to disseminate phylogenetic knowledge"). The identification of any specific commercial products is for the purpose of specifying a protocol, and does not imply a recommendation or endorsement by the National Institute of Standards and Technology. We thank Yan Wong, James Rosindell, Karen Cranston, Jonathan Rees, Jaime Huerta-Cepas, Brian Sidlauskas, Jim Allman, Ramona Walls, Mark Holder, Daisie Huang, Cyndy Parr, Katja Schulz, Jodie Wiggins, and Son Tran, for useful comments, discussions, and technical advice. We thank many others who contributed indirectly to this project through their participation in two NESCent hackathons with a Phylotastic theme.

References

[1] Antonelli, A., Hettling, H., Condamine, F.L., Vos, K., Nilsson, R.H., Sanderson, M.J., Sauquet, H., Scharn, R., Silvestro, D., Topel, M., Bacon, C.D., Oxelman, B. & Vos, R.A. (2017) Toward a Self-Updating Platform for Estimating Rates of Speciation and Migration, Ages, and Relationships of Taxa. *Syst Biol*, **66**, 152–166.

[2] Boyle, B., Hopkins, N., Lu, Z., Garay, J.A.R., Mozzherin, D., Rees, T., Matasci, N., Narro, M.L., Piel, W.H., Mckay, S.J., Lowry, S., Freeland, C., Peet, R.K. & Enquist,

B.J. (2013) The Taxonomic Name Resolution Service: an online tool for automated standardization of plant names. *BMC Bioinformatics*, **14**.

[3] Cracraft, J., Donoghue, M., Dragoo, J., Hillis, D. & Yates, T. (2002) Assembling the tree of life: harnessing life's history to benefit science and society. Technical report, U.C. Berkeley.

[4] Drew, B.T., Gazis, R., Cabezas, P., Swithers, K.S., Deng, J., Rodriguez, R., Katz, L.A., Crandall, K.A., Hibbett, D.S. & Soltis, D.E. (2013) Lost branches on the tree of life. *PLoS Biol*, **11**, e1001636.

[5] GBIF: The Global Biodiversity Information Facility (2018) What is GBIF?

[6] He, Z., Zhang, H., Gao, S., Lercher, M.J., Chen, W.H. & Hu, S. (2016) Evolview v2: an online visualization and management tool for customized and annotated phylogenetic trees. *Nucleic Acids Res*, **44**, W236–41.

[7] Hinchliff, C.E., Smith, S.A., Allman, J.F., Burleigh, J.G., Chaudhary, R., Coghill, L.M., Crandall, K.A., Deng, J., Drew, B.T., Gazis, R., Gude, K., Hibbett, D.S., Katz, L.A., Laughinghouse, H.D.t., McTavish, E.J., Midford, P.E., Owen, C.L., Ree, R.H., Rees, J.A., Soltis, D.E., Williams, T. & Cranston, K.A. (2015) Synthesis of phylogeny and taxonomy into a comprehensive tree of life. *Proc Natl Acad Sci U S A*, **112**, 12764–9.

[8] Kumar, S., Stecher, G., Suleski, M. & Hedges, S.B. (2017) TimeTree: A Resource for Timelines, Timetrees, and Divergence Times. *Mol Biol Evol*, **34**, 1812–1819.

[9] Letunic, I. & Bork, P. (2016) Interactive tree of life (iTOL) v3: an online tool for the display and annotation of phylogenetic and other trees. *Nucleic Acids Res*, **44**, W242–5.

[10] Maddison, D., Schulz, K.S. & Maddison, W. (2007) The Tree of Life Web Project. *Zootaxa*, pp. 19–40.

[11] McTavish, E.J., Drew, B.T., Redelings, B. & Cranston, K.A. (2017) How and Why to Build a Unified Tree of Life. *Bioessays*, **39**.

[12] Nguyen, T.H., Son, T.C. & Pontelli, E. (????) Automatic Web Services Composition for Phylotastic. F. Calimeri, K. Hamlen & N. Leone, eds., *Practical Aspects of Declarative Languages*, pp. 186–202. Springer International Publishing.

[13] Nguyen, T.H., Son, T.C. & Pontelli, E. (2018) Phylotastic: An Experiment in Creating, Manipulating, and Evolving Phylogenetic Biology Workflows Using Logic Programming. *Theory and Practice of Logic Programming*, **18**.

[14] Ooms, J. (2014) The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects. *ArXiv e-prints*, **arXiv:1403.2805 [stat.CO]**.

[15] Page, R.D. (2005) A Taxonomic Search Engine: federating taxonomic databases using web services. *BMC bioinformatics*, **6**, 48.

[16] Parr, C.S., Wilson, N., Leary, P., Schulz, K.S., Lans, K., Walley, L., Hammock, J.A., Goddard, A., Rice, J., Studer, M., Holmes, J.T. & Corrigan, R. J., J. (2014) The Encyclopedia of Life v2: Providing Global Access to Knowledge About Life on Earth. *Biodivers Data J*, p. e1079.

[17] Patterson, D.J., Cooper, J., Kirk, P.M., Pyle, R.L. & Remsen, D.P. (2010) Names are Key to the Big New Biology. *Trends Ecol Evol*, **25**, 686–91.

[18] Piel, W., Chan, L., Dominus, M., Ruan, J., Vos, R. & Tannen, V. (????) Treebase v. 2: A database of phylogenetic knowledge. *e-BioSphere 2009*.

[19] R Core Team (2018) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[20] Redelings, B.D. & Holder, M.T. (2017) A supertree pipeline for summarizing phylogenetic and taxonomic information for millions of species. *PeerJ*, **5**, e3058.

[21] Rosindell, J. & Harmon, L.J. (2012) OneZoom: a fractal explorer for the tree of life. *PLoS Biol*, **10**, e1001406.

[22] Stoltzfus, A., Lapp, H., Matasci, N., Deus, H., Sidlauskas, B., Zmasek, C.M., Vaidya, G., Pontelli, E., Cranston, K., Vos, R., Webb, C.O., Harmon, L.J., Pirrung, M., O'Meara, B., Pennell, M.W., Mirarab, S., Rosenberg, M.S., Balhoff, J.P., Bik, H.M., Heath, T.A., Midford, P.E., Brown, J.W., McTavish, E.J., Sukumaran, J., Westneat, M., Alfaro, M.E., Steele, A. & Jordan, G. (2013) Phylotastic! Making tree-of-life knowledge accessible, reusable and convenient. *BMC Bioinformatics*, **14**, 158.

[23] Stoltzfus, A., O'Meara, B., Whitacre, J., Mounce, R., Gillespie, E.L., Kumar, S., Rosauer, D.F. & Vos, R.A. (2012) Sharing and re-use of phylogenetic trees (and associated data) to facilitate synthesis. *BMC Research Notes*, **5**, 574.

[24] Webb, C. & Donoghue, M.J. (2005) Phylomatic: tree assembly for applied phylogenetics. *Molecular Ecology Notes*, **5**, 181–183.

TABLE 1
Main Phylotastic services description

Web Service	Description
Common Names to Scientific Names	Get the scientific name of a species from its common name
NCBI_common_name	following the NCBI database
EBI_common_name	following EBI services
ITIS_common_name	following ITIS services
TROPICOS_common_name	following TROPICOS services
EOL_common_name	following EOL services
Scientific Name Extraction	Scrape scientific names from a URL, text or any tipe of file
GNRD_wrapper_URL; GNRD_wrapper_text; GNRD_wrapper_file	using Global Names Recognition and Discovery (GNRD) services
TaxonFinder_wrapper_URL; TaxonFinder_wrapper_text	using Taxon Finder
Taxonomic Name Resolution	Match scientific names to authorative taxonomies and resolve mismatches
OToL_TNRS_wrapper	using the Open Tree of Life taxonomy
GNR_TNRS_wrapper	using the Global Names Resolver tool (several taxonomies)
iPlant_TNRS_wrapper	using iPlant collaborative services
Taxon Sampling	Get all scientific names of species that:
Taxon_all_species	belong to a given higher taxon name
Taxon_country_species	and are found in a given country (using iNaturalist database)
Taxon_genome_species	and have a genome sequence (deposited in NCBI)
Taxon_popular_species	and match the most popular species within the taxon using OneZoom tool
Taxon Information and Images	Get various information of a species such as
Image_url_species	image urls and corresponding license information using EOL
Info_url_species	information urls from EOL
ECOS_Conservation	conservation status from ECO services
Tree Retrieval	Get phylogenetic trees from a list of taxa
OToL_wrapper_Tree;	from Open Tree of Life synthetic tree
OToL_supported_studies	and all supporting studies
Phylomatic_wrapper_Tree	from Phylomatic
Treebase_Tree	from TreeBase
Supersmart_wrapper_Tree	using supersmart
Tree Scaling	Scale branch lengths of a tree relative to time
Datelife_scale_tree	using the DateLife service
OToL_scale_tree	using OToLs unoficial scaling service
Tree Comparison	Compare two phylogenetic trees symmetrically
Compare_trees	
List Management	
Add_new_list; Get_list; Replace_species_list; Update_metadata_list; Remove_list;	Save, publish, access, remove or update lists of names.

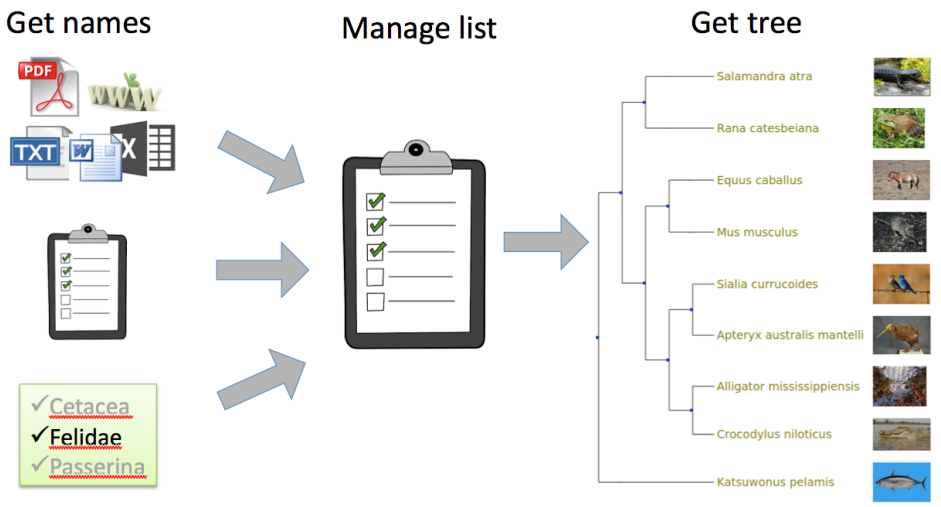

FIGURE 1: Phylotastic portal general workflow: (1) A list of species can be initiated by: selecting a public list already available in the portal; uploading your own list, as a text file with one name per line, or as a Darwin Core Archive (DwC-A) file; uploading any text or image file (formats supported include PDF, doc, txt, xls, png) and scraping names from it; scraping names from a a resource URL (HTML or other resource); get all species names from a named higher taxon, or a random sample of N species, only species with known genomes (via NCBI services), and species with occurrence records in a given location (via iNaturalist). (2) Manage the list of species: choose a subsample of species and process misspellings and misidentifications. (3) Get a phylogenetic tree of your list of species, without branch lengths or scaled to geological time. Add organism images to the tips of the tree and save the graphical render displayed in the portal, or download the tree in newick format and process it with your favorite tool to generate tree images, or use it for downstream analyses in various areas of research and education.

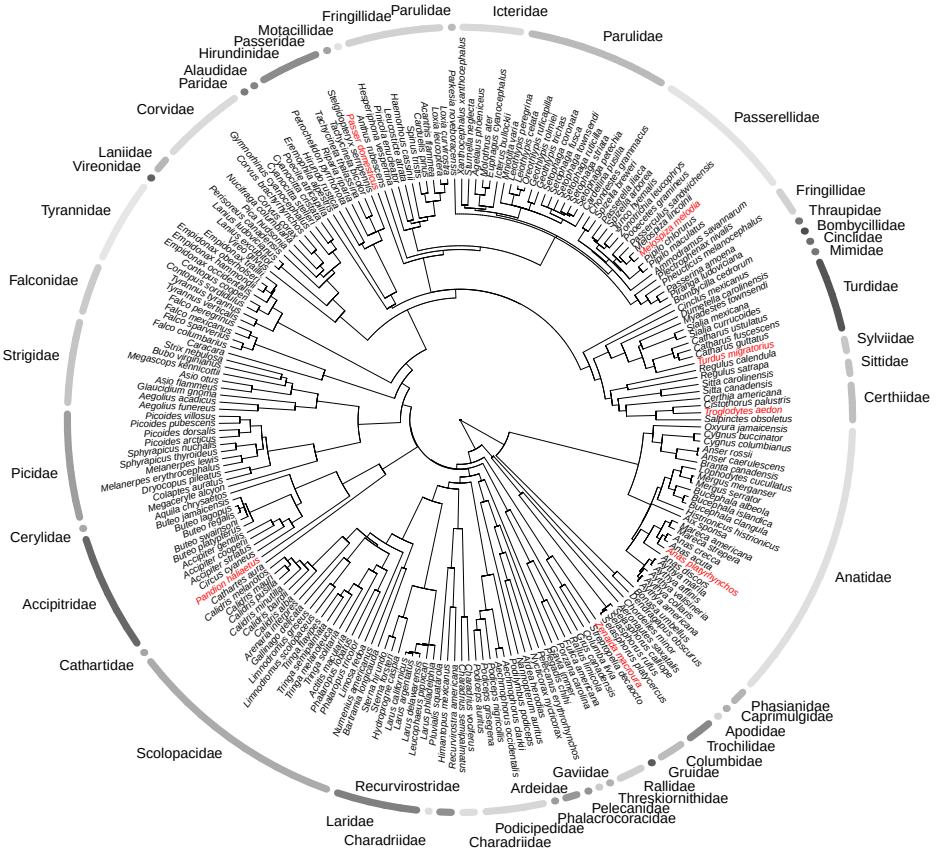

FIGURE 2: Dated phylogenetic tree of birds from Yellowstone National Park obtained with a Phylotastic workflow. Species registered by an observer are marked in red. Families are delimited by gray arcs. This figure was generated with functions from rphylotastic, datelife, rott, and ape R packages. Code has been made fully reproducible by implementing a plan with the drake R package and it is available at https://github.com/phylotastic/rphylotastic/blob/master/data-raw/rphylotastic_examples.R