Autocuration Cyberinfrastructure For Scientific Discovery and Preservation

Smruti Padhy, Edgar Black, Betsy Cowdery, Liana Diesendruck, Michal Dietze, Greg Jansen, Rob Kooper, Praveen Kumar, Jong Lee, Rui Lui, Richard Marciano, Luigi Marini, Dave Mattson, Barbara Minsker, Chris Navarro, Ankit Rai, Marcus Slavenas, William Sullivan, Jason Votava, Qina Yan, Inna Zharnistsky, Kenton McHenry

National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign
E-mail: {spadhy, mchenry}@illinois.com

DIBBs Brown Dog [1] is a recent cyberinfrastructure effort which aims to create two new services to aid users in the searching, accessing, and usage of digital data and provide these services in a manner that is as broadly and easily accessible as possible. At its lowest level, the Data Access Proxy (DAP) providing file format conversion capabilities and the Data Tilling Service (DTS) providing content-based extractions will be accessible via a deliberately compact REST interface (TABLE I and TABLE II). On top of this a number of client libraries and applications can and are being constructed to even further reduce the overhead of accessing the provided functionality (e.g. libraries in Javascript, Python, R, MATLAB and interfaces such as bookmarklets, browser extensions, and other standalone applications). At the heart of the two services, however, is their extensibility, allowing one to potentially incorporate any library or tool as a conversion or extraction component within the services towards both the leveraging of the functionality they provide as well as the preservation of the tools themselves (Fig.1 and Fig. 2). Both cases support a variety of scripting languages to include wrap tools for inclusion in the Brown Dog services (e.g. Python, MATLAB, R, bash, AutoHotKey, etc.)

```
extractors.connect_message_bus(extractorName=extractorName,
                               messageType=messageType,
                               rabbitmqURL=rabbitmqURL,
                               rabbitmqExchange=rabbitmqExchange,
                               processFileFunction=process_file,
                               checkMessageFunction=check_message)
                                                                  Connect

def process_file(parameters):
  global extractorName
  inputfile=parameters['inputfile']

  # call actual program
  result = subprocess.check_output(['wc', inputfile], stderr=subprocess.STDOUT)
  (lines, words, characters, filename) = result.split()           Work on File

extractors.upload_file_metadata(mdata=metadata,
parameters=parameters)                                            Return Metadata
```

Figure 1. pyClowder library calls needed to include a new Python tool as an extractor service within the DTS.

Built on top of a number of software components such as NCSA Clowder [2, 3] to manage and carry out extractions, Polyglot [4, 5] to find and chain format conversions, Versus [6] for content based indexing and retrieval, and Daffodil, an open source implementation of the Data Format Description Language (DFDL) language [7] to capture file format specifications in a machine readable manner, Brown Dog aims to support data conversion and extraction/analysis needs from a broad range of communities. In Fig. 4 we show examples of a number of these use cases, e.g. carrying data conversions from various data sources [8, 9] to use as inputs to a broad range of ecological models [10, 11] via PEcAn [12, 13], extracting images along routes and from these extracting a measure of the greenness along that path, extracting river locations from historical maps, and extracting flood plains from Lidar data [14]. Through the extensibility of the two services and a Tools Catalog akin to the Apple App Store or Galaxy Tool Shed [15] we aim to expand the number of communities and use cases we support over time. An architecture (Fig. 3) focused on the distribution of jobs across heterogeneous resources and tools allows us to be as flexible as possible in terms of the tools we can include. Further an elastic scalability component allows these heterogeneous collections of tools to be replicated as needed in order to efficiently handle large numbers of requests to the two services.

```
#Application name (Version)
#File types supported (e.g. document, depth, image, ...)
#Comma separated list of supported input formats
#Comma separated list of supported output formats
                                                                  Describe

#Call external application and/or carry out conversion
...                                                               Convert File
```

Figure 2. Comments needed to include a new tools as a converter service within the DAP.

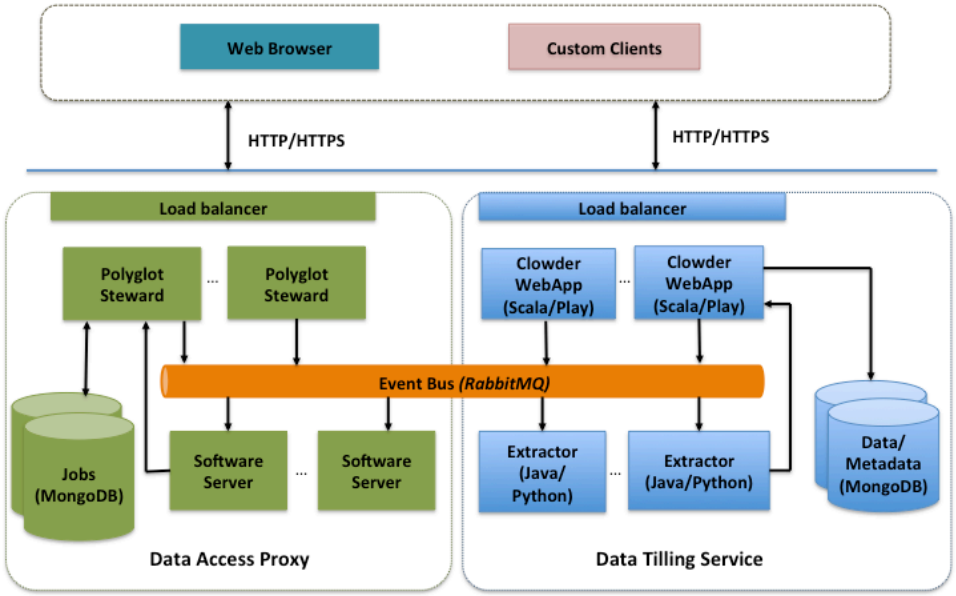

Figure 3. Architecture of the Brown Dog Services (DAP and the DTS). Both converters and extractors connect to a distributed RabbitMQ messaging bus specific to a type of data or software where a head node process (e.g. Clowder for the DTS or Polyglot for the DAP) submits and track jobs. In the case of the DAP a Software Server [5] is used to add arbitrary software as a conversion tool. Both head nodes are stateless, with jobs being stored in a shared MongoDB database, so that they can be replicated behind a load balancer. An elasticity component monitors these queues and awakens or spawns additional VMs or processes of extractors/converters as queue lengths grow.

TABLE I. The DTS REST API for extractions

GET	/api/extractions/inputs	Lists the input file format supported by currently running extractors
POST	/api/extractions/upload	Uploads a file for extraction of meta- data and returns file id
GET	/api/extractions/upload	Uploads a file for extraction using the file's URL
GET	/api/extractions/{id}/status	Checks for the status of all extractors processing the file with id
GET	/api/files/{id}/metadata	Gets tags, technical metadata, and content based signatures extracted for the specified file
GET	/api/extractions/servers	Lists servers IPs running the extractors
GET	/api/extractions/extractors	Lists the currently running extractors
GET	/api/extractions/extractors/ details	Lists the currently details running extractors

TABLE II. The DAP REST API for conversions

GET	/api/conversions/outputs	Lists all output formats that can be reached
GET	/api/conversions/inputs	Lists all input formats that can be accepted
GET	/api/conversions/inputs/ input format	Lists all output formats that can reach the specified input format
GET	/api/conversions/outputs/ output format	Lists all input formats that can reach the specified output format
GET	/api/conversions/convert/ output format/file URL	Converts the specified file to the requested output format
POST	/api/conversions/convert/ output format	Converts the uploaded file to the requested output format
GET	/api/conversions/software	Lists all available conversion software
GET	/api/conversions/servers	Lists all currently available Software Servers

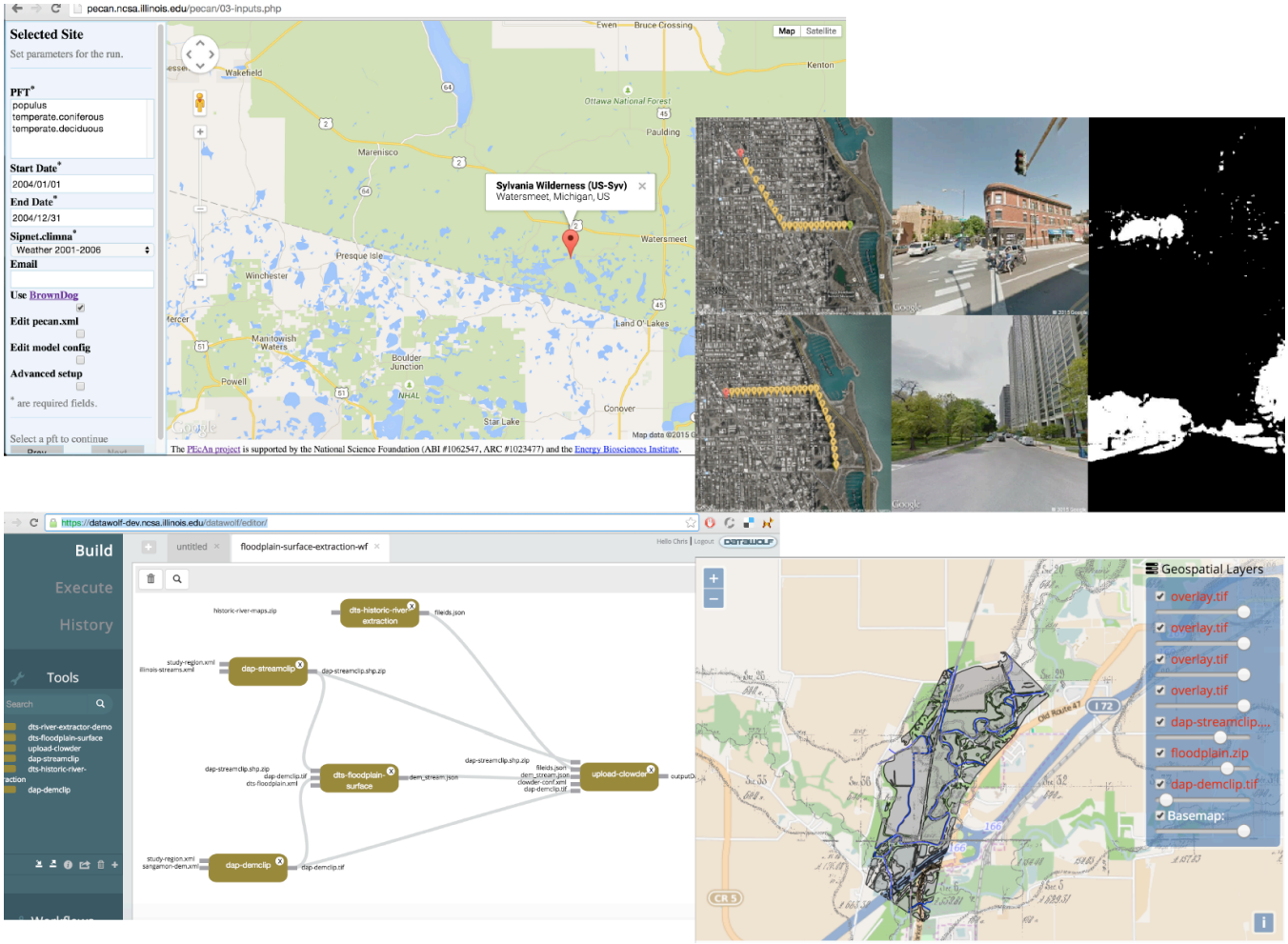

Figure 4. Top: PEcAn using Brown Dog to carry out data conversion for ecological model inputs. Middle: The Clowder semantic content management system ingesting a route and extracting derived data based on images along the route, e.g., the Green Index. Bottom: The DataWolf workflow system chaining a number of DAP and DTS calls to extractor flood basins and overlay historical river locations from geo-referenced digitized maps.

ACKNOWLEDGMENT

This research and development has been funded through National Science Foundation Cooperative Agreement ACI-1261582.

REFERENCES

[1] K. McHenry, J. Lee, M. Dietze, P. Kumar, B. Minsker, R. Marciano, *et al.*, "DIBBs Brown Dog, PaaS for SaaS for PaaS," *XSEDE Reproducibility Workshop,* 2014.

[2] L. Marini, R. Kooper, J. Futrelle, J. Plutchak, A. Craig, T. McLaren, *et al.*, "Medici: A Scalable Multimedia Environment for Research," *Microsoft eScience Workshop,* 2010.

[3] J. Myers, M. Hedstrom, D. Akmon, S. Payette, B. Plale, I. Kouper, *et al.*, "Towards Sustainable Curation and Preservation: The SEAD Project's Data Services Approach," *Interoperable Infrastructures for Interdisciplinary Big Data Sciences Workshop, IEEE eScience,* 2015

[4] K. McHenry, R. Kooper, and P. Bajcsy, "Towards a Universal, Quantifiable, and Scalable File Format Converter," *The IEEE International Conference on eScience,* 2009.

[5] K. McHenry, R. Kooper, M. Ondrejcek, L. Marini, and P. Bajcsy, "A Mosaic of Software," *The IEEE International Conference on eScience,* 2011.

[6] L. Marini, P. Bajcsy, S. Padhy, D. Bonnie, A. Vandercreme, R. Kooper, B. Long, *et al.*, "Versus: A Framework for General Content-Based Comparisons", *IEEE eScience,* 2012.

[7] M. Beckerle and S. Hanson, "Data Format Description Language (DFDL) v1.0 Specification," *Open Grid Forum,* 2014.

[8] F. Mesinger, G. Dimego, E. Kalnay, K. Mitchell, and P. Sahfran, "North American Regional Reanalysis," *Bulletin of the American Meteorological Society,* 2006.

[9] *AmeriFlux*, http://ameriflux.lbl.gov/

[10] B. Braswell, W. Sacks, E. Linder, and D. Schimel, "Estimating diurnal to annual ecosystem parameters by synthesis of a carbon flux model with eddy covariance net ecosystem exchange observations," *Global Change Biology,* 2005.

[11] P. Moorecroft, G. Hurtt, and S. Pacala, "A Method for Scaling Vegetation Dynamics: the Ecosystem Demography Model (ED)," *Ecological Monographs,* 2001.

[12] R. Kooper, K. McHenry, M. Dietze, D. LeBauer, S. Serbin, and A. Desai, "Ecological Cyberinfrastructure and HPC Towards More Accurately Predicting Future Levels of Greenhouse Gases," *XSEDE,* 2013.

[13] M. Dietze, S. Serbin, C. Davidson, A. Desai, X. Feng, R. Kelly, *et al.*, "A Quantitative Assessment of a Terrestrial Biosphere Model's Data Needs Across North American Biomes," *Journal of Geophysical Research - Biogeosciences,* 2014.

[14] J. Stout and P. Belmont, "TerEx Toolbox for semi-automated selection of fluvial terrace and floodplain features from lidar," *Earth Surface Processes and Landforms,* 2013.

[15] J. Goecks, A. Nekrutenko, and J. Taylor, "Galaxy: A comprehensive approach for supporting accessible, reproducble, and transparent computation research in the life sciences," *Genome Biology,* 2010.