Brown Dog: Making the Digital World a Better Place, a Few Files at a Time

Sandeep Puthanveetil
Satheesan*
NCSA
UIUC
sandeeps@illinois.edu

Jay Alameda*
NCSA
UIUC
alameda@illinois.edu

Shannon Bradley*
NCSA
UIUC
sbrad77@illinois.edu

Michael Dietze
Department of Earth and
Environment
Boston University
dietze@bu.edu

Benjamin Galewsky*
NCSA
UIUC
bengal1@illinois.edu

Gregory Jansen
College of Information
Studies
University of Maryland
jansen@umd.edu

Rob Kooper*
NCSA
UIUC
kooper@illinois.edu

Praveen Kumar*
Department of Civil and
Environmental Engineering
UIUC
kumar1@illinois.edu

Jong Lee*
NCSA
UIUC
jonglee1@illinois.edu

Richard Marciano
College of Information
Studies
University of Maryland
marciano@umd.edu

Luigi Marini*
NCSA
UIUC
lmarini@illinois.edu

Barbara S. Minsker
Department of Civil and
Environmental Engineering
Southern Methodist
University
minsker@smu.edu

Christopher M.
Navarro*
NCSA
UIUC
cmnavarr@illinois.edu

Arthur Schmidt*
Department of Civil and
Environmental Engineering
UIUC
aschmidt@illinois.edu

Marcus Slavenas*
NCSA
UIUC
slavenas@illinois.edu

William C. Sullivan*
Department of Landscape
Architecture
UIUC
wcsulliv@illinois.edu

Bing Zhang*
NCSA
UIUC
bing@illinois.edu

Yan Zhao*
NCSA
UIUC
yanzhao3@illinois.edu

Inna Zharnitsky*
NCSA
UIUC
inna@illinois.edu

Kenton McHenry*
NCSA
UIUC
mchenry@illinois.edu

ABSTRACT

Brown Dog is a data transformation service for auto-curation of long-tail data. In this digital age, we have more data available for analysis than ever and this trend will only increase. According to most estimates, 70-80% of this data is unstructured, and together with unsupported data formats and inaccessible software tools, in essence, this data is not either easily accessible or usable to its owners in a meaningful way. Brown Dog aims at making this data more

accessible and usable by auto-curation and indexing, leveraging existing and novel data transformation tools. In this paper, we discuss the recent major component improvements to Brown Dog including transformation tools called extractors and converters; desktop, web and terminal-based clients which perform data transformations; libraries written in multiple programming languages which integrate with existing software and extend their data curation capabilities; an online tool store for users to contribute, manage and share data transformation tools and receive credit for developing them; cyberinfrastructure for deploying the system on diverse computing platforms leveraging scalability via Docker swarm; workflow management service for creatively integrating existing transformations to generate custom, reproducible workflows which meet research needs, and its data management capabilities. This paper also discusses data transformation tools developed to support some scientific and allied use cases, thereby benefiting researchers in diverse domains. Finally, we briefly discuss our future directions with regard to production deployments as well as how users can access Brown Dog to manage their un-curated unstructured data.

*NCSA is the abbreviation for The National Center for Supercomputing Applications. UIUC is the abbreviation for The University of Illinois at Urbana-Champaign.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

PEARC '18, July 22–26, 2018, Pittsburgh, PA, USA

© 2018 Copyright held by the owner/author(s). Publication rights licensed to the Association for Computing Machinery.

ACM ISBN 978-1-4503-6446-1/18/07...\$15.00

<https://doi.org/10.1145/3219104.3219132>

CCS CONCEPTS

• **Information systems** → **Semi-structured data**; **Extraction, transformation and loading**; *Data provenance*; *RESTful web services*; • **Networks** → **Cloud computing**; • **Software and its engineering** → **Software as a service orchestration system**;

KEYWORDS

API gateway, auto-curation, big data, data conversion, data curation, data cyberinfrastructure, data transformation, data wrangling, metadata extraction, orchestration, provenance, unstructured data

ACM Reference Format:

Sandeep Puthanveetil Satheesan, Jay Alameda, Shannon Bradley, Michael Dietze, Benjamin Galewsky, Gregory Jansen, Rob Kooper, Praveen Kumar, Jong Lee, Richard Marciano, Luigi Marini, Barbara S. Minsker, Christopher M. Navarro, Arthur Schmidt, Marcus Slavenas, William C. Sullivan, Bing Zhang, Yan Zhao, Inna Zharnitsky, and Kenton McHenry. 2018. Brown Dog: Making the Digital World a Better Place, a Few Files at a Time. In *PEARC '18: Practice and Experience in Advanced Research Computing, July 22–26, 2018, Pittsburgh, PA, USA*. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3219104.3219132>

1 INTRODUCTION

Long-tail data or unstructured data represents that particular spectrum in the big data realm, which is either not easily accessible to its owners (e.g. due to unavailability of software to read a specific data format) or has very little metadata available and hence is mostly unusable in any meaningful way. Today we generate more digital data than ever in the history of mankind and this trend is continually increasing. Though some part of this digital data is structured and curated, the most part of it is not. According to most estimates, this is as much as 70–80% of the total digital data in the world, if not more [4, 5, 17].

Brown Dog is a data transformation service that provides a solution to the unstructured data problem by means of auto-curation and indexing. It uses a combination of existing and novel transformation tools to enable users to meet their data transformation needs. At its core, Brown Dog contains two types of transformation tools called extractors and converters, a data management service (Clowder), a conversion engine (Polyglot), a workflow engine (Data-Wolf), an online tool store (Tools Catalog), cyberinfrastructure for orchestration (Docker Swarm, ClusterMan, and ClusterStats), a variety of client interfaces (BD-CLI, BD Windows Client, BDFiddle), libraries in different programming languages, API Gateway (Fence), and a 90 TB uncurated/unstructured digital data testbed (DRAS-TIC CI-BER).

Many works in the past have studied and tried to address this problem and this is currently an actively researched topic [3, 10, 11]. What makes Brown Dog unique is that it addresses modern-day data-centric research needs in a manner that cut across scientific disciplines and is extensible towards handling the ever-changing data/analysis requirements. It is scalable in terms of supporting ever growing data/computational loads. It currently serves a wide variety of communities such as biology, geoscience, material science, crop science, civil engineering, social science, and the humanities. Apart from these, it also has a foothold in the general public use case with the aim of taking care of everyday data transformation needs. It enables and encourages tools sharing/leveraging previous

work for new research goals so that everyone is not reinventing the wheel and we can build on one another's successes.

The rest of this paper is organized as follows: In Section 2, we briefly introduce the Brown Dog transformation tools and then discuss some of its recent improvements. In Section 3, we discuss the software clients and libraries, with a focus on how they help researchers and developers to integrate Brown Dog services with their new or existing software. In Section 4, we discuss the Brown Dog web and programming interfaces and how these enable both basic and advanced users to interact with Brown Dog. In Section 5, we briefly discuss how Brown Dog supports sharing and reusing of tools through Tools Catalog. In Section 6, we discuss in detail the cyberinfrastructure for deploying the Brown Dog components and services in a cluster. In Section 7, we briefly discuss the workflow and data management modules of Brown Dog and how they enrich Brown Dog's data curation capabilities. In Section 8, we demonstrate how Brown Dog is enabling the research community to advance their research by discussing some of the recent use case updates. In Section 9, we discuss the current deployments of Brown Dog and future plans. In Section 10, we describe the performance tests that were done on Brown Dog and share the recent results.

2 TRANSFORMATION TOOLS AND RECENT IMPROVEMENTS

Brown Dog transformation tools perform transformation of data from one form to another. A transformation can refer to a change in file format or it can refer to change in form. The former process is called conversion and are done by converters and the latter refers to metadata extraction and are done by extractors. These tools are exposed as services in Brown Dog, which make it easier for users to access them via the Cloud without having to install any software at their end [13, 14]. A RabbitMQ messaging bus helps to manage the high volume of transformation requests and forwards them to appropriate converters or extractors based on the file type.

The Brown Dog transformation tools have undergone a few major changes recently. We dockerized most if not all of the transformation tools and are made available via Docker Hub^{1,2}. This helps in easy deployment and hence scaling of the transformation tools, sharing of the data transformation tools, as well as in better dependency management. We also recently embraced JSON-LD³ as the standard for the metadata generated by extractors. This has helped in making the JSON metadata more portable, interlinked and support more advanced semantic queries. Leveraging Linked Data and Semantic Web technologies, the system can accommodate any ontology. Individual extractors are free to adopt the most appropriate existing ontology or create a new one. All of the extractors are undergoing a major refactoring. Now, the extractors use pyClowder (a Python library that handles communication between extractor and Clowder), contain at least one pair of test input and output files and a metadata document that describes the extractor is made a requirement. This extractor metadata file contains information about the extractor such as extractor name, JSON-LD context information, Bitbucket repository URL, docker image name, etc. Also,

¹<https://hub.docker.com/u/clowder/dashboard/>

²<https://hub.docker.com/u/ncsapolyglot/dashboard/>

³<https://json-ld.org/>

Figure 1: Screenshots of the Brown Dog Windows File Manager Extension depicting the conversion of an image from JP2 to PNG format (view clockwise starting from top-left).

continuous integration tests are executed against extractors and converters to ensure that the tools, as well as the entire system, are performing well when new changes are incorporated into the system.

3 CLIENTS AND LIBRARIES

Multiple clients and libraries which interact with Brown Dog are currently available. These were developed with the intent of making Brown Dog easily integratable with other software systems. We hope this will largely reduce the overhead for other software tools to use Brown Dog.

3.1 Command-Line Interface (BD-CLI)

The Brown Dog Command-Line Interface or BD-CLI⁴ comes with a set of features that enable a command-line tool user to use Brown Dog and if need be, integrate it with their own systems. Using BD-CLI, a user can perform the data transformations supported by Brown Dog – extractions and conversions. The current version of BD-CLI is written in Python using the Brown Dog Python Library (BD.py).

Apart from the features that are supported by other Brown Dog clients, BD-CLI also supports moving compute to data. With the `--bigdata` flag, a user can perform a data transformation operation where the data resides (“move the compute to the data”), rather than using Brown Dog’s remote API endpoints. This is really useful when working with very large data sets (typically ~GBs or more) and is difficult to transfer back-and-forth between client and the server. This aligns with the paradigm of data centric computing [6]. BD-CLI accomplishes this by the use of an API endpoint that gets a list of extractors/converters that can process a particular file type

⁴<https://pypi.python.org/pypi/bdcli/>

```
% Converts file, downloads converted file, and returns path to converted file
%
% bdapi          browndog api domain
% input_filename path to input file, can be local or http
% output_extension convert to this format e.g., 'png'
% output_path    location to put converted file
% varargin(1)    wait time for conversion before download
function return_string = convert(bdapi,input_filename,output_extension,output_path,token)
switch nargin
case 5
    wait=60;
otherwise
    warning('Warning: too many input arguments for function convert, trailing arguments ignored');
end

url = strcat(bdapi,'/v1/conversions/',output_extension);

[pathstr,name,ext] = fileparts(input_filename);
filename = strcat(name,ext);

% POST TO DAP
url_converted = char(post_file(url,filename,input_filename, token,'convert'));

% DOWNLOAD CONVERTED FILE
return_string = download_file(url_converted,output_path,token,wait);
end
```

Figure 2: A screenshot of the BD.m convert method.

and then by downloading Docker images of those corresponding extractors and converters. The Docker containers mount the data, perform the transformation and the results are saved along with the input data.

3.2 Windows Desktop Client

We designed and developed a Windows desktop client called the File Manager Extension (FME), which provides Windows users with an easy to use graphical interface to Brown Dog services, perform conversion and extractions, and provides the capability to index and search files within folders based on extracted metadata. FME is developed in the C# programming language using the .Net framework. A C# library (BDLib) has been developed to interact with Brown Dog services. After installation, FME commands are integrated into the Windows Context Menu. Users can browse files and folders via Windows Explorer and simply right-click on the desired file or folder and select FME commands from the pop-up menu as shown in Fig. 1.

3.3 Libraries (BD.py, BD.r, BD.m, and BD.js)

We have also developed libraries that wrap the Brown Dog API endpoints in several commonly used programming languages like Python, R, MATLAB, and JavaScript. These libraries help to easily integrate Brown Dog with new or already existing systems by reducing the development time. The MATLAB Brown Dog library (BD.m) is shown in Fig. 2 as an example.

Four public methods are provided to allow a MATLAB user access to Brown Dog to perform data transformations:

- *token* - gets an access token and a specific API key from Fence that can be used to access Brown Dog services
- *outputs* - gets all possible conversions for a specific file format
- *convert* - uploads a file and performs conversion using Brown Dog
- *extract* - uploads a file and returns extracted metadata in JSON-LD format

Instead of the MATLAB native functions *websave*, *webwrite*, *webread*, and *weboptions*, more advanced HTTP interfaces using Java libraries are employed in the development. In particular, the Sun HttpClient, which is included within MATLAB, and *org.json*, which was added as a jar file to the MATLAB client, were used extensively

Figure 3: A screenshot of BDFiddle showing how to select a file and perform a transformation and the corresponding code snippets generated.

as well as many other Java classes that are included within MATLAB. Furthermore, a custom Java class was developed for posting a file to a REST endpoint and is included in the Java build path for the app. This is necessary because MATLAB has no native object for handling a byte stream buffer. The library has been packaged as a MATLAB project/application.

4 WEB AND PROGRAMMING INTERFACES

4.1 BDFiddle

BDFiddle⁵ is a web interface to Brown Dog to introduce users to its capabilities, play around with its features, and obtain simple

⁵<https://browndog.ncsa.illinois.edu/bdfiddle/>

code snippets for transformations in different languages. Users will need to sign up for a Brown Dog account before they can get access to BDFiddle. Like other “fiddle” environments like JSFiddle, ScalaFiddle, and PythonFiddle, BDFiddle is for the users to get easily accustomed with Brown Dog without having to install anything on their end.

When using BDFiddle, as the first step, the user selects a file or a file URL. Next, they click on a button to obtain a list of Brown Dog supported transformations. There are couple of ways to select a transformation. In the automatic mode, a user can select a particular file format (in case of conversion) or submit the file for extraction using all the available extractors. In the manual (or more advanced) mode, a user has the option to choose a specific application to perform a particular conversion or select a specific extractor for metadata extraction. Once the file is submitted for a specific transformation, the results are shown in an output text area. In case of conversion, a URL of the converted file is provided, while in case of extraction, the extracted metadata is displayed. Code snippets in multiple languages that perform the same selected transformation are also displayed in the BDFiddle interface. This is provided to those users who would like to programmatically interact with Brown Dog. Some of the features of BDFiddle that are described above are depicted in Fig. 3.

4.2 API Gateway (Fence)

Brown Dog services are available to users through an API Gateway called Fence⁶. Previously, clients connected directly to the individual Brown Dog services. Fence normalizes the backend services and provides a central implementation for authentication, authorization, activity logging, and rate limiting the entire Brown Dog service. Fence is developed in Scala using the Twitter Finagle RPC system. For the most part, Fence works as a passthrough proxy enforcing authentication and authorization, but in the case of some endpoints, Fence modifies the request or response to hide some of the complexity or private knowledge of the backend services. For example, in the case of a Polyglot conversion, Fence substitutes the URL of the converted file inside the Polyglot server with a Brown Dog API external URL.

Software clients can request API keys by providing a username and password. Fence currently supports authentication against LDAP, Atlassian Crowd and credentials stored in a configuration file to support local development and testing. API keys can then be used to request access tokens, which in turn can be used to make requests to specific service endpoints. Requests for transformation are rate limited by user using a combination of leaky bucket rate limiting and total requests. API keys, access tokens, and rate limiting information are stored in Redis.

Fence logs every request for a transformation by user, including Polyglot and Clowder identifiers, external file URLs, data and time in a MongoDB database. This database is used for debugging, accounting and enforcing access control on transformations results by authenticated users. The Brown Dog Web service API is documented using the Swagger framework. Developers can use the

⁶<https://bd-api.ncsa.illinois.edu/>

Figure 4: A screenshot of the Brown Dog online tools repository showing some of the currently available transformation tools.

Swagger definition to create client libraries for a variety of programming languages. Users can use tools such as the Swagger UI to browse and test the API. The Swagger UI is shown in Fig. 5.

5 SHARING DATA TRANSFORMATION TOOLS

Brown Dog leverages off of existing and novel transformation tools to come up with a large set of transformation services for curating unstructured data. In research domains, we commonly see that custom tools get created for a project, but they are not maintained after the project concludes. In order to encourage sharing and reuse

Figure 5: A screenshot of the Brown Dog API documentation page showing the authorization endpoints. Please note that the other endpoints are collapsed for display purposes.

of software tools, Brown Dog comes with an online tool repository, currently known as Tools Catalog, where tool creators can publish their tools, the home page of which is shown in Fig. 4. Developers of extractors and converters can, in turn, use existing tools to create new transformation services. Additional information regarding this tool repository is provided here [13].

6 CYBERINFRASTRUCTURES FOR ORCHESTRATION AND SCALING

The Brown Dog infrastructure is deployed as a Docker Swarm cluster. We selected Docker Swarm for this implementation due to its low resource overhead and simple deployment as compared to other containerized orchestration environments.

6.1 Orchestration

The individual extractors and converters run as Docker services downloaded as images from Docker Hub. These services are scaled by increasing the number of replicas. A general-purpose service, named *ClusterStats*, has been developed to simplify interactions with the swarm. It provides a REST interface for deploying, scaling (up or down), restarting, and deleting services.

The API accepts memory and CPU constraints that are used in creating the service. Extractors and converters deployed through this API take advantage of the arbitrary labels provided for swarm services to associate some domain specific information with each deployed service. These labels are used by other Brown Dog tools to carry out their duties in the swarm.

ClusterStats also offers a web-based status screen to show metrics related to the cluster like number of nodes in the cluster, number of services, total number of containers, current CPU core usage, current memory usage, and current disk usage. It also highlights the top services based on total number of replicas, CPU cores used, memory used, and disc space used. Finally, the tool generates charts to show trends in overall cluster utilization. The admin users can

interact with the application to view details on individual nodes in the cluster as well as deployed services.

6.2 Scaling

The deployed tools receive work from message queues administered by an instance of RabbitMQ. When Brown Dog is launched, it is configured with information to connect to a Clowder and a Polyglot instance. These services are represented in the RabbitMQ environment as *exchanges*. Requests to transform data are written to these exchanges.

Extractors and converters listen on specific RabbitMQ queues. As part of deployment to a cluster, these queues are bound to exchanges by specifying selector keys. Messages that match the selector key will be posted to the corresponding extractor or converter's queue. Each tool can be scaled up or down by simply changing the number of replicas.

To meet variable demand placed on the cluster, a service called *ClusterScale* is deployed to the swarm. Its job is to periodically snapshot the backlog of messages in each queue. When it judges that an extractor or converter is falling behind current demand, it will attempt to scale it up, subject to the constraints on replicas set by the administrator when the service was deployed. Similarly, when a queue remains persistently empty, *ClusterScale* will reduce the number of replicas while obeying the minimum replica constraint established by the administrator.

6.3 ClusterMan Application

The deployments and configuration settings can all be managed via command line interfaces and the RabbitMQ proprietary administration web page. Nevertheless, these tools can be difficult to learn, and deploying production tools in this manner tends to be error prone and brittle. We are developing an end-user application that will simplify these routine procedures. This new tool, named *ClusterMan* is a Play application that ties together connections to Brown Dog Tools Catalog, ClusterStats, and RabbitMQ API.

In *ClusterMan*, a *cluster* is defined as a ClusterStats instance, combined with one RabbitMQ URI and virtual host for extractors, and a separate URI and virtual host for converters. A single cluster could support multiple exchanges and services. Additionally, in *ClusterMan*, a *project* represents a deployment of Clowder and/or Polyglot. In practice, projects correspond to exchanges in RabbitMQ.

ClusterMan has two main functions, basically 1) deploying extractors or converters to a cluster and 2) binding extractors to a project. We will describe these operations in more detail.

6.3.1 Deploying Extractors or Converters to a Cluster. The user is required to select a cluster from one that is defined in *ClusterMan*'s configuration file. They indicate their wish to deploy a tool to that cluster by clicking on the "Deploy" button. They are redirected to a window where they choose whether to deploy an extractor or a converter. Based on this election, they are presented with a drop-down list of all tools of the indicated type from the global tools catalog service. The user selects the tool from this list and clicks "next."

Finally, the user is prompted to enter parameters related to the tool deployment. This list of parameters is kept as small as possible to keep the operation easy to learn. When the user accepts the

Figure 6: A screenshot of the lightweight workflow graph viewer depicting the conversion transformation applied on an example input image file.

parameters, *ClusterMan* will submit a request to *ClusterStats* to handle the actual deployment.

6.3.2 Binding Extractors to a Project. Extractors will only process messages for which there is a matching binding to the exchange that generates the message. A binding screen is included in *ClusterMan* where users can view all of the projects associated with the selected project. The user may click on a bindings action to display a page that shows all of the services deployed to the cluster and reports the current bindings to the selected project.

The administrator may change the binding settings by selecting a new value for one or more services and clicking "OK". *ClusterMan* will adjust the bindings to match the user selections.

7 WORKFLOW AND DATA MANAGEMENT

While it is important to have the capability to perform different kinds of data transformations, it is equally or more important to know what kind of transformations were applied to transform data from one form to another. *DataWolf*⁷, the workflow management module of Brown Dog, keeps track of the provenance of transformations. We added an API endpoint to Fence, which creates a workflow graph that represents the conversions/extractions that were applied to a data file. This is displayed using a lightweight web viewer. Now a user can visualize the steps in a data transformation and can store a permanent URL that can be accessed in the future as well, as shown in Fig. 6.

Clowder [9] is the data management module of Brown Dog. It is a cloud-based open source scalable data management system for research that supports curation of unstructured data and metadata across multiple research domains and diverse data types. Apart from other prominent features of Clowder, which includes support for large amounts of data, custom data visualizations, social annotation etc., Brown Dog extensively uses its extraction bus and REST APIs for data curation and management needs. In turn, the extractors developed in collaboration with Brown Dog's use cases are contributed back to this extraction bus for consumption by the community. When a file is submitted to Brown Dog for extraction, it gets first stored in an underlying Clowder instance, which then sends a message to all extractors listening to that specific file type through its extraction bus. The extractors then process the file, extract specific metadata, which is then stored back in Clowder.

⁷<https://datawolf.ncsa.illinois.edu/>

8 USE CASES' UPDATES

Brown Dog currently supports many scientific and allied use cases [12, 14, 15]. The success of Brown Dog will be based not only on our initial use cases which helped us as we built and tested the system, but also the ongoing inclusion of transformation tools from Brown Dog and contribution of new tools as part of innovative scientific community research and business case implementations. Here we highlight some recent use case updates and how they have been able to benefit from Brown Dog and vice versa.

Brown Dog is being leveraged to model soil organic carbon dynamics across the whole watershed to understand the mechanism and dynamics of Soil Organic Matter (SOM) both on the surface and below ground over decades of time. They are able to take data on activities such as tillage, crop rotation, tile drain installation and determine how they impact the dynamics of SOM. This helps researchers understand 1) how soil erosion and landscape evolution impact organic C dynamics; 2) how tillage influences soil organic matter dynamics; 3) whether an erosional landscape is a carbon sink or source [19].

One of Brown Dog's initial use cases, the PEcAn benchmarking system, has been able to move on to emphasis on scientific analysis, rather than software development because of the implementation of Brown Dog. Some of their recent works include [2], which focused on benchmarking land models at the Duke and ORNL FACE experiments and [18], which focused on benchmarking models at a semiarid eddy-covariance site.

Brown Dog has supported some allied use cases in the digital humanities domain in the past [7, 8, 16]. The transformation tools that were developed during these efforts are now being integrated with Brown Dog for future use.

9 DEPLOYMENTS AND FUTURE PLANS

The initial instance of Brown Dog is currently available at NCSA⁸. However, to advance the accessibility of the Brown Dog we are partnering with other institutions for individual deployments of Brown Dog. University of Maryland has their own instance of Brown Dog. The UMD iSchool team has assembled "a unique data training and testing facility called the Cyber Infrastructure for Billions of Electronic Records (CI-BER) testbed. The CI-BER testbed currently represents 72 million records and 57 terabytes of data from decades of legacy records across 135 government agencies. The team's goal is to enable new forms of archival analytics at scale" [1].

Clowder is currently being leveraged for in-classroom education, supported by new extractors through Brown Dog, at Southern Methodist University for civil engineering informatics.

The Brown Dog beta version has been released and we are currently in a "friendly-user" mode. New users are welcome to sign up for an account from the Brown Dog home page and their accounts will be activated in batches. Users can try out the features once their account is activated.

Forthcoming plans for Brown Dog development include 1) upgrading and dockerizing previously submitted transformation tools, 2) incorporating new research transformations in cooperation with our use case partners, 3) finalizing requirements for the version 1.0

⁸<https://browndog.ncsa.illinois.edu/>

release, and 4) transitioning from a software research and development to a software support and development paradigm.

There are several fundamental needs to consider when transitioning an academic, research-oriented project into a self-sustaining, ongoing service. While funding agencies can support the research and development phase of a project a more public approach is needed to target sustainability and growth. When funding ends, the code developed for the project is likely to become obsolete due to non-trivial maintenance and improvement costs. Hence, as we move past our Beta rollout we are evaluating options and implementing plans for a more concentrated marketing style approach to promote Brown Dog.

Our future vision for Brown Dog is for it to serve as a low-level infrastructure to data enabling a new era of science and applications which makes use of and relies on uncured data sources. The goal is to serve both the scientific community and the general public as a service for data manipulation, moving towards an era where a user's access to data is not limited by a file's format or what software created it or whether a collection has been fully curated or not. The future plan is to follow a path for ensuring the stability of the Brown Dog service and providing a sustainable funding source for both maintenance, development and personnel continuity.

This year the team plans to implement a Gap Analysis to obtain a baseline for performance expectations for Brown Dog as a product and use this over the course of the next four years to determine what was achieved and find the gaps between projected and actual results.

Goals for this first year of implementing a marketing plan for Brown Dog include 1) determining ideal customers, fleshing out the remarkable uniqueness of Brown Dog, and 3) creating a core strategy for branding, defining an appropriate initial market, what exemplifies market penetration in these markets, define initial product and service offerings, marketing materials, web and social media presence plan.

Our milestones will include regular newsletters/blogs/updates on social media, gathering feedback from our initial set of users, public speaking, completing training and educational materials, finalizing a support plan and completing marketing materials.

10 PERFORMANCE TESTS

The Brown Dog testbed at UMD is configured to test Brown Dog services, simulating common user interactions with the service at web-scale performance loads. The performance tests are done using Gatling⁹, which leverages the parallelism of Scala to spawn hundreds of simultaneous web clients according to a predefined load scenario. In the testbed, various client-side events are logged to files and then loaded into Elasticsearch after the test, for analysis using the Grafana¹⁰ visualization tool. Future test plans include the launching of Gatling tests on multiple disposable Docker Swarm containers, thereby multiplying the load scenario many times over, with data captured to the same Elasticsearch instance. Results from the recent performance tests done at UMD are shown in Table 1.

⁹<https://gatling.io/>

¹⁰<https://grafana.com/>

Table 1: Performance Test Results

Test Scenario	Number of Simulated Users	Number of Served Requests	Peak Rate (requests per minute)	Average Request Time (milliseconds)
Conversions ^a	2098	6175	1287	334
Extractions ^b	1956	3977	1310	291

^a Conversion of unique documents in office formats (DOC, DOCX, ODF, RTF, WPD, WP, LWP, and WSD) to PDF.

^b Extraction on files belonging to random file formats.

11 CONCLUSIONS

We discussed Brown Dog, which is a data transformation service that proposes a solution to the unstructured data problem by leveraging existing and novel data transformation tools by means of auto-curation and indexing. We discussed how Brown Dog is unique in that it provides a flexible architecture that is designed to address the modern-day data-centric research needs in a manner that cut across scientific disciplines and is extensible and scalable in terms of handling the ever-changing data, analysis, and computational needs. We described various components of Brown Dog and their capabilities. We also shared some recent updates of the scientific and allied use cases that are benefiting from Brown Dog and how they are contributing to Brown Dog through the creation of transformation tools. Finally, we discussed our current and future plans for Brown Dog.

In today's digital world, properly handling and curating unstructured data is definitely the key to gain insights across data-centric research domains. Even outside of the research fields, the data transformation needs of the general public are huge. To tackle this problem, the research community needs to come together to create, share, and reuse data curation tools. We hope that Brown Dog can provide a platform for this as it continues in its endeavor in making the digital world a better place, a few files at a time!

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. ACI-1261582.

REFERENCES

- [1] 2015. iSchool's "Brown Dog" Project Develops Cloud Services for Uncurated Big Data. (2015). Retrieved March 21, 2018 from <https://ischool.umd.edu/news/ischools-brown-dog-project-develops-cloud-services-un-curated-big-data>
- [2] Elizabeth Cowdery and Michael Dietze. 2017. Looking Past Primary Productivity: Benchmarking System Processes that Drive Ecosystem Level Responses in Models. *AGU Fall Meeting Abstracts* (Dec. 2017).
- [3] Yuri Demchenko, Paola Grosso, Cees De Laat, and Peter Membrey. 2013. Addressing big data issues in scientific data infrastructure. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on*. IEEE, 48–55.
- [4] Gerard George, Martine R Haas, and Alex Pentland. 2014. Big data and management. *Academy of management Journal* 57, 2 (2014), 321–326.
- [5] Seth Grimes. 2008. Unstructured data and the 80 percent rule. *Carabridge Bridgepoints* (2008), 10.
- [6] IBM. 2014. It's Time to Shift to Data Centric Computing. (2014). Retrieved March 21, 2018 from <http://research.ibm.com/articles/datacentricdesign/>
- [7] Virginia Kuhn, Alan Craig, Michael Simeone, Sandeep Puthanveetil Satheesan, and Luigi Marini. 2015. The VAT: enhanced video analysis. In *Proceedings of the 2015 XSEDE Conference: Scientific Advancements Enabled by Enhanced Cyberinfrastructure*. ACM, 11.
- [8] Alison Langmead, Paul Rodriguez, Sandeep Puthanveetil Satheesan, and Alan Craig. 2017. Extracting Meaningful Data from Decomposing Bodies. In *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact (PEARC17)*. ACM, New York, NY, USA, Article 41, 8 pages. <https://doi.org/10.1145/3093338.3093368>
- [9] Luigi Marini, Indira Gutierrez-Polo, Rob Kooper, Sandeep Puthanveetil Satheesan, Maxwell Burnett, Jong Lee, Todd Nicholson, Yan Zhao, and Kenton McHenry. 2018. Clowder: Open Source Data Management for Long Tail Data. In *PEARC'18: Practice and Experience in Advanced Research Computing*. ACM, New York, NY, USA. <https://doi.org/10.1145/3219104.3219159> To appear.
- [10] William K Michener, Suzie Allard, Amber Budden, Robert B Cook, Kimberly Douglass, Mike Frame, Steve Kelling, Rebecca Koskela, Carol Tenopir, and David A Viegals. 2012. Participatory design of DataONE-enabling cyberinfrastructure for the biological and environmental sciences. *Ecological Informatics* 11 (2012), 5–15.
- [11] James Myers, Margaret Hedstrom, Dharma Akmon, Sandy Payette, Beth A. Plale, Inna Kouper, Scott McCaulay, Robert McDonald, Isuru Suriarachchi, Aravindh Varadharaju, Praveen Kumar, Mostafa Elag, Jong Lee, Rob Kooper, and Luigi Marini. 2015. Towards Sustainable Curation and Preservation: The SEAD Project's Data Services Approach. In *Proceedings of the 2015 IEEE 11th International Conference on e-Science (E-SCIENCE '15)*. IEEE Computer Society, Washington, DC, USA, 485–494. <https://doi.org/10.1109/eScience.2015.56>
- [12] Smruti Padhy, Jay Alameda, Rob Kooper, Rui Liu, Sandeep Puthanveetil Satheesan, Inna Zharnitsky, Gregory Jansen, Michael C. Dietze, Praveen Kumar, Jong Lee, Richard Marciano, Luigi Marini, Barbara Minsker, Chris Navarro, Marcus Slavenas, William Sullivan, and Kenton McHenry. 2016. An Architecture for Automatic Deployment of Brown Dog Services at Scale into Diverse Computing Infrastructures. In *Proceedings of the XSEDE16 Conference on Diversity, Big Data, and Science at Scale (XSEDE16)*. ACM, New York, NY, USA, Article 33, 8 pages. <https://doi.org/10.1145/2949550.2949647>
- [13] Smruti Padhy, Edgar Black, Betsy Cowdery, Liana Diesendruck, Michael Dietze, Greg Jansen, Rob Kooper, Praveen Kumar, Jong Lee, Rui Liu, Richard Marciano, Luigi Marini, Dave Mattson, Barbara Minsker, Chris Navarro, Ankit Rai, Marcus Slavenas, William Sullivan, Jason Votava, Qina Yan, Inna Zharnitsky, and Kenton McHenry. 2015. Autocuration Cyberinfrastructure for Scientific Discovery and Preservation. In *Proceedings of the 2015 IEEE 11th International Conference on e-Science (E-SCIENCE '15)*. IEEE Computer Society, Washington, DC, USA, 264–267. <https://doi.org/10.1109/eScience.2015.60>
- [14] Smruti Padhy, Greg Jansen, Jay Alameda, Edgar Black, Liana Diesendruck, Mike Dietze, Praveen Kumar, Rob Kooper, Jong Lee, Rui Liu, Richard Marciano, Luigi Marini, Dave Mattson, Barbara Minsker, Chris Navarro, Marcus Slavenas, William Sullivan, Jason Votava, Inna Zharnitsky, and Kenton McHenry. 2015. Brown Dog: Leveraging Everything Towards Autocuration. In *Proceedings of the 2015 IEEE International Conference on Big Data (Big Data) (BIG DATA '15)*. IEEE Computer Society, Washington, DC, USA, 493–500. <https://doi.org/10.1109/BigData.2015.7363791>
- [15] Sandeep Puthanveetil Satheesan, Jay Alameda, Shannon Bradley, Michael Dietze, Gregory Jansen, Rob Kooper, Praveen Kumar, Jong Lee, Richard Marciano, Luigi Marini, Barbara S. Minsker, Chris Navarro, Eugene Roeder, Arthur Schmidt, Marcus Slavenas, William Sullivan, Bing Zhang, Yan Zhao, Inna Zharnitsky, and Kenton McHenry. 2017. Brown Dog: A Data Transformation Ecosystem for Research - Advancing from Beta to 1.0. *AGU Fall Meeting Abstracts* IN21B-0038 (Dec. 2017).
- [16] Paul Rodriguez, Sandeep Puthanveetil, Jeffrey Will, Elizabeth Wuerffel, and Alan Craig. 2017. Extracting, Assimilating, and Sharing the Results of Image Analysis on the FSA/OWI Photography Collection. In *Proceedings of the Practice and Experience in Advanced Research Computing 2017 on Sustainability, Success and Impact*. ACM, 42.
- [17] Christopher C Shilakes and Julie Tylman. 1998. Enterprise Information Portals, Merrill Lynch. Inc., New York, NY (1998).
- [18] Anne Thomas, Michael Dietze, Russell L. Scott, and Joel A. Biederman. 2017. Assessing Ecosystem Model Performance in Semiarid Systems. *AGU Fall Meeting Abstracts* (Dec. 2017).
- [19] Dong Kook Woo. 2017. Dynamics of inorganic soil-nitrogen age in intensively managed landscape. Ph.D. Dissertation. University of Illinois at Urbana-Champaign.