DIBBs Brown Dog

A Science Driven Data Transformation Service

NCSA BOSTON UNIVERSITY SMU UNIVERSITY OF MARYLAND


Photos
Structural Defects
Image Stitching
Green Infrastructure
Materials Development
Gap Filling
Loss of Organ Function
Tabular Data
Root Tip Tracking
Mapping
Evolution
NLP
Databases
Underwater Photos
Climate Modeling
Video
Hyperspectral
Lidar
Microscopy Images
Historic Maps
Color Correction
Publications
Radar
Flood Plain Analysis
Documents
River Depth Distribution
Hazard Modeling
Cell Tracking
Web Sites
People Detection/Tracking
Bee Colony Behavior
Feedlot Tracking
Stream Detection and Sinuosity
River Maturity
Paleoclimate
Bee Detection/Tracking
Phenomics
Human Preference Modeling
Tissue Classification
Pollen Detection/Classification
Satellite/Aerial Photos
Land Cover/Usage
Disease Detection
Regions in Conflict
Species Detection/Counting
Reef Changes
Coastline Changes
Water Detection (e.g. Lakes, Retaining Ponds)
Sentiment Analysis
Large Dynamic Group Behavior
River Meander
Food Supply
3D Data
Handwritten Documents
Pre-Digital Datasets
Renal Failure
3D Reconstruction
Event Detection
Simulations

New Capabilities

- Facilitate, simplify, and make more reproducible the digital **data wrangling** of modern scientific research activities

Data Wrangling

Extractions: Data transformations that create new, often higher level, data from the contents of data (e.g. metadata and other derived products). Typically not reversible. Towards indexing, searching, cleaning, further analyzing the data.

Data Collection(s)
URL, File System, Paper, …

Byte Representation
File Formats, Micro-formats, Ad Hoc formats, Data Bases, Websites, Documents, …

Convert

Data Contents
Arrays, Strings, Images, Videos, Audio, 3D Models, …

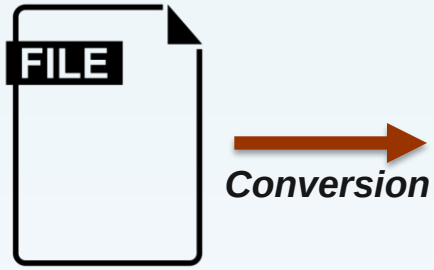
Extract

Derived Data
Metadata, Visualizations, Fused products, Cleaned products, Subsets

Scientific Application
Analyze, Search, View, Relate, …

Conversions: Data transformations that largely preserve the entirety of the data (i.e. potentially reversible). Towards accessing, loading, or viewing the data contents.

Usable Data



Conversion

Extraction


```
[
    {
        "extractor_id": "ncsa.image.exif",
        "Image": "558c3d84e4b00c3a039d5ac5",
        "Format": "JPEG (Joint Photographic Experts Group JFIF format)",
        "Class": "DirectClass",
        "Geometry": "2592x1936+0+0",
        "Resolution": "72x72",
        "Print size": "36x26.8889",
        "Units": "PixelsPerInch",
        "Type": "TrueColor",
        "Endianess": "Undefined",
        "Colorspace": "sRGB",
        "Depth": "8-bit",
        "Channel depth": {
            "red": "8-bit",
            "green": "8-bit",
            "blue": "8-bit"
        },
        "Channel statistics": {
            "Red": {
                "min": "0 (0)",
```

```
{
    "id": "558c3d84e4b00c3a039d5ac5",
    "filename": "IMG_0997.JPG",
    "tags": [
        "Human Face Automatically Detected",
        "Person Automatically Detected",
        "Human Eyes Automatically Detected"
    ]
}
```

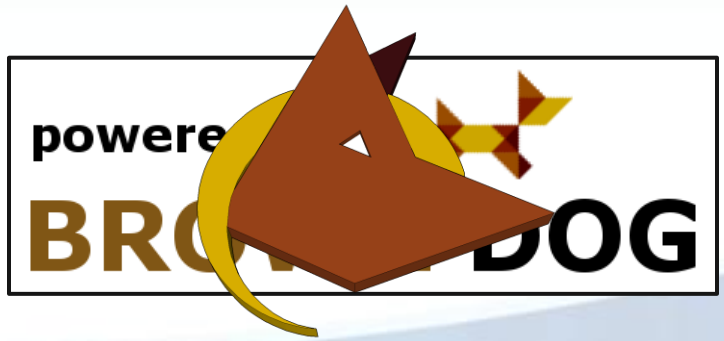
```
{
    "extractor_id": "ncsa.image.ocr",
    "ocr_simple": [
        "EB BROWSER MOSAIC THE FIRST POPULAR BROWSER FOR THE WORLD WIDE
        BY MARC ANDREESSEN BINA THE NATIONAL CENTER COMPUTING APPLICATIONS
        1993 RELEASE TO THE PUBLIC INTERNET USERS EASY ACCESS TO SOURCES OF
        INFORMATION win HAVE TRANSFORMED THE INFORMATION UNIVERSITY OF "
    ]
},
{
    "Human Preference Extractor": {
        "Definitions": {
            "Human Preference": "A Computer Vision model that uses the
            spectral data of an image to get a human preference value ranging from 1 to 5,
            "Green Index": "The green index is the estimated percentage of green pixels wi
        },

n Preference": "4",
n Index": "53.8"

les": [
ts-dev.ncsa.illinois.edu:9000/files/558c3d84e4b00c3a039d5ac5"

tasets": [
ts-dev.ncsa.illinois.edu:9000/datasets/558c3dd6e4b00c3a039d5b77"
```


NCSA

Brown Dog - A Science Driven Data Transformation Service

- Extensibility
  - Easy to add new transformations (i.e. converters and extractors)
  - Encapsulated transformation software & dependencies
- API
  - Supporting other applications/frameworks to build on top of
- Support for diverse usage (i.e. clients, languages, community tools & applications)
- Scalability, Distributed, Data Movement, Provenance with File Validation & Information Loss, Tool Preservation & Publication, **Open Source**

https://en.wikipedia.org/wiki/Mongrel



https://bd-api.ncsa.illinois.edu

swagger Authorize

Brown Dog API

Documentation for the Brown Dog API

Created by Brown Dog Support
See more at https://opensource.ncsa.illinois.edu/confluence/questions/topics/41156618/brown-dog
Contact the developer
University of Illinois/NCSA Open Source License

Authorization

Show/Hide | List Operations | Expand Operations

POST	/keys	Create a new api key
POST	/keys/{key}/tokens	Create a new access token from api key
DELETE	/keys/{key}	Delete api key and all related access tokens
DELETE	/tokens/{token}	Delete token
GET	/tokens/{token}	

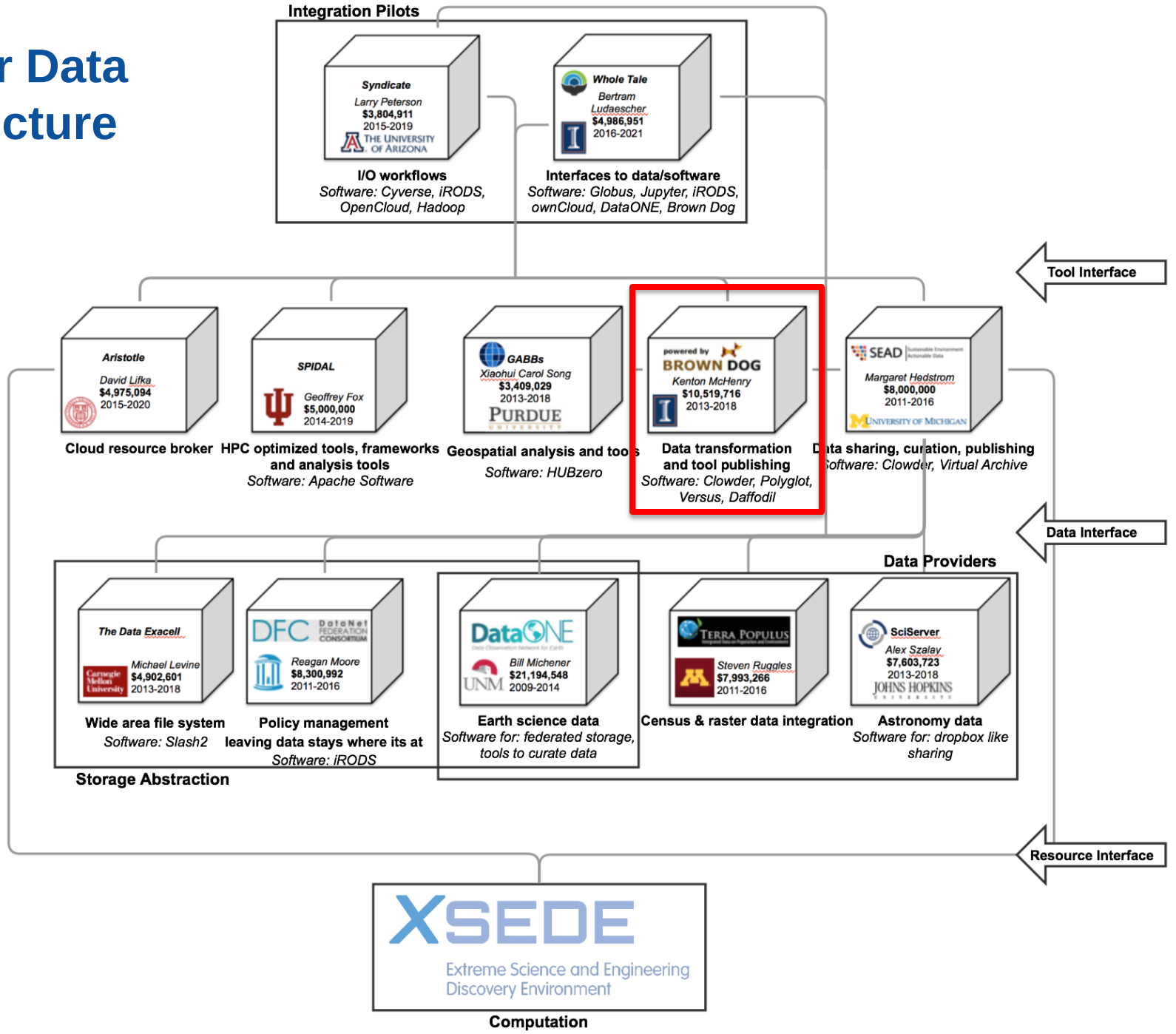
Conversions

Show/Hide | List Operat

GET	/dap/outputs	List all output formats that can be reached

powered by BROWN DOG

Enhance Other Data Cyberinfrastructure Components



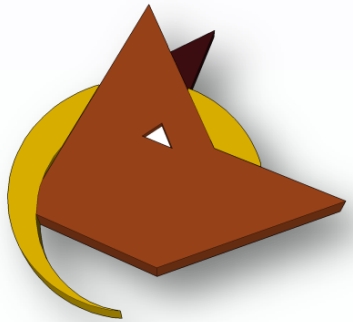

re3data.org
REGISTRY OF RESEARCH DATA REPOSITORIES
Enhanced Higher Data Cyberinfrastructure Components
EZID
Long-term identifiers made easy
Fedora
OpenAIRE
zenodo
HYDROSHARE
AXES
MetaCat
SciDrive
Web Lab
Rappture
Tool Interface
ownCloud
TwoRavens
BisQue
swift
OpenCloud
ckan
SLASH2
Agave
Science API
B2SHARE
EUDAT
EARTHCUBE
pegasus
DataONE
DRYAD
figshare
Girder
RDA
RESEARCH DATA ALLIANCE
iRODS
CYVERSE
yt
CyberGIS
jupyter
Galaxy
globus
The Dataverse Project
hubzero
PLATFORM FOR SCIENTIFIC COLLABORATION
dSPACE
Kepler
SciServer
BitCurator
DFC
DataNet FEDERATION CONSORTIUM
SEAD

Status & Dissemination

SCIENCE GATEWAYS .ORG

RDA RESEARCH DATA ALLIANCE

CZO | CRITICAL ZONE OBSERVATORIES
U.S. NSF NATIONAL PROGRAM



ILLINOIS DATA BANK

WEST BIG DATA INNOVATION HUB

Midwest Big Data Hub

Accelerating the Big Data Innovation Ecosystem

SOUTH BDHUB

NORTHEAST BIG DATA INNOVATION HUB

software carpentry

NATIONAL DATA SERVICE PARTICIPANT

NSF **DIBBs / DataNets**

XSEDE Extreme Science and Engineering Discovery Environment

Research Data Serv ices

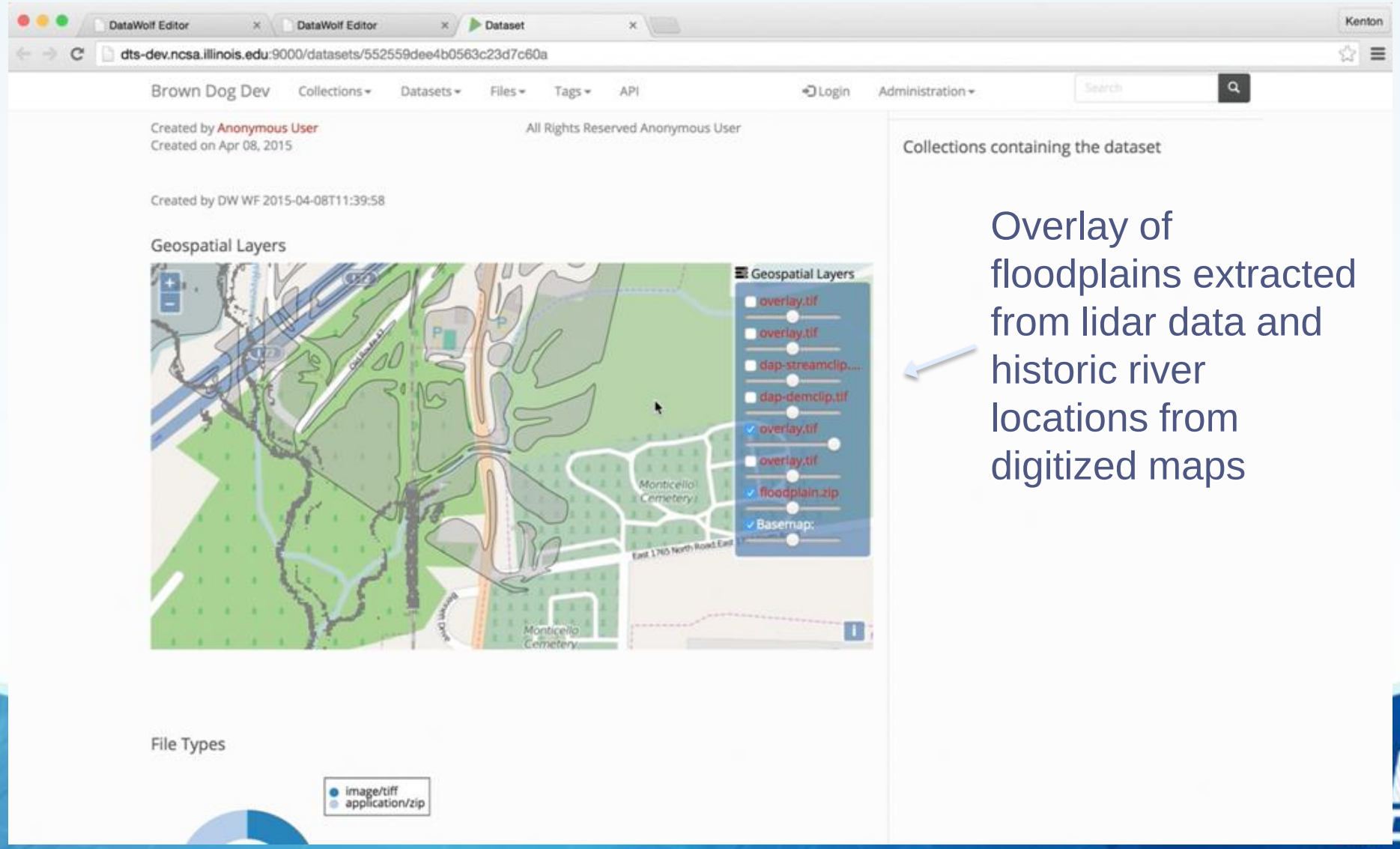
EARTHCUBE

NCSA

SEAD | Sustainable Environment Actionable Data

Vi-SEEM

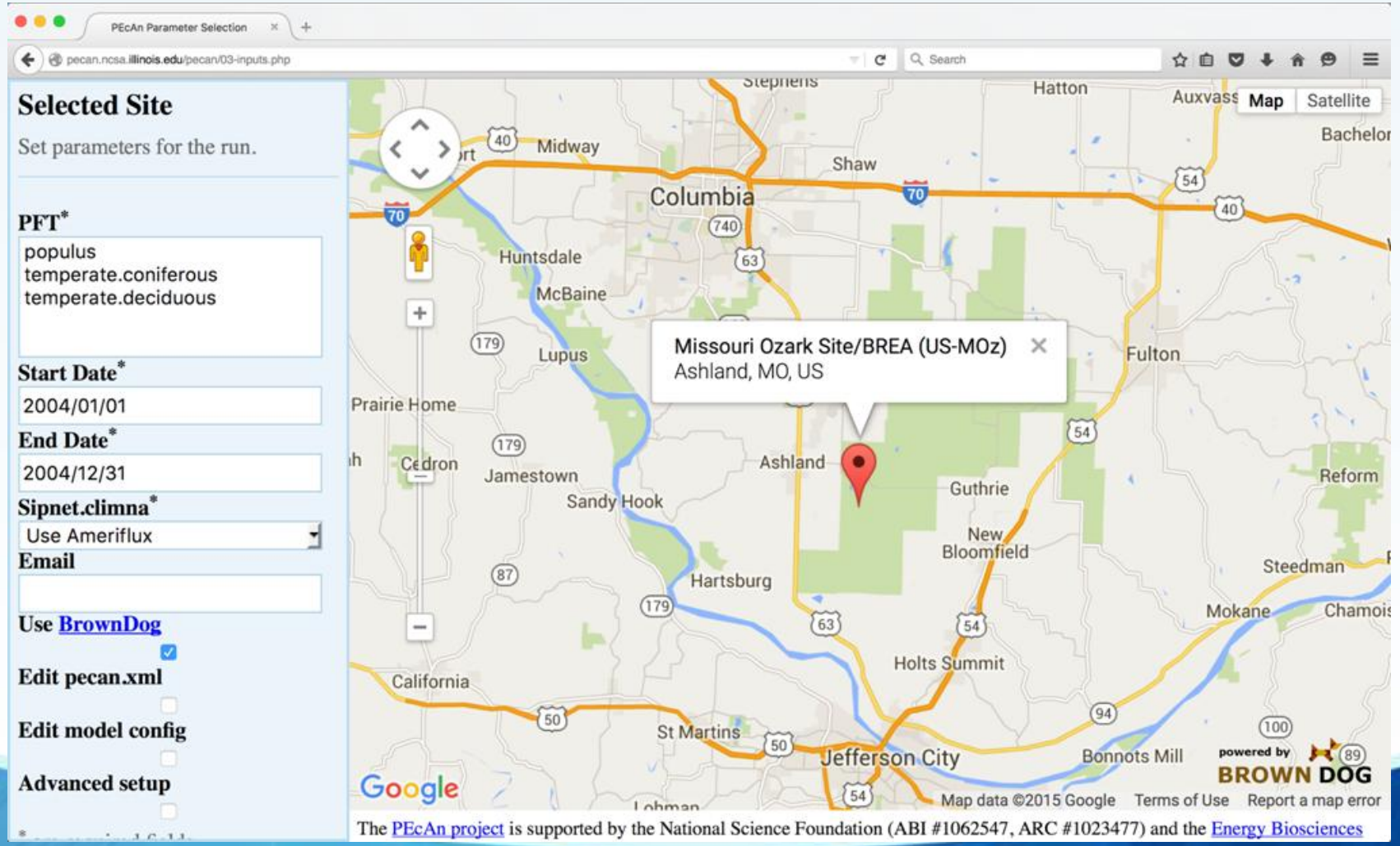
4CEED



Overlay of floodplains extracted from lidar data and historic river locations from digitized maps

Predictive Ecosystem Analyzer (PEcAn)

Dataset / model conversions



Novel (On the Fly) Data Products

ui.R

```
library(shiny)
library(leaflet)

# Define UI
ui <- shinyUI(fluidPage(
  # Application title
  titlePanel("Generate xml file"),

  sidebarLayout(
    sidebarPanel(
      selectInput("type", "Type", c("AmeriFlux", "FLUXNET", "NARR")),
      uiOutput("agreementUI"),
      textInput("start_date", "Start year", placehold="2001"),
      textInput("end_date", "End year", placehold="2001"),
      textInput("token", "BrownDog Token"),
      conditionalPanel(
        condition = "input.token != ''",
        selectInput("model", "Model", c("ameriflux.zip", "clim", "dal
      )
    ),
    mainPanel(
      fluidRow(
        leafletMap(
          "map", "100%", 400,
          initialTileLayer = "//{s}.tiles.mapbox.com/v3/jcheng.map-5e
          initialTileLayerAttribution = HTML('Maps by <a href="http:/
          options=list(
            center = c(37.45, -93.85),
            zoom = 4,
            maxBounds = list(list(17, -180), list(59, 180)))
        ),
        style='padding:10px;'
      ),
      verbatimTextOutput("xmltext"),
      fluidRow(
        column(3,
               conditionalPanel(
                 condition = "input.agreement == true",
                 downloadButton('downloadXML', 'Download XML')
               )),
        column(9,
               conditionalPanel(
                 condition = "input.token != '' && input.agreement ==
                 downloadButton("downloadData", "Download Data")
               )
        )
      )
    )
  )
```

Line 1, Column 1

Generate xml file

pecan-dev.ncsa.illinois.edu/shiny/BrownDog/

Search

Generate xml file

Type

AmeriFlux

I agree to AmeriFlux license.

Start year

2001

End year

2001

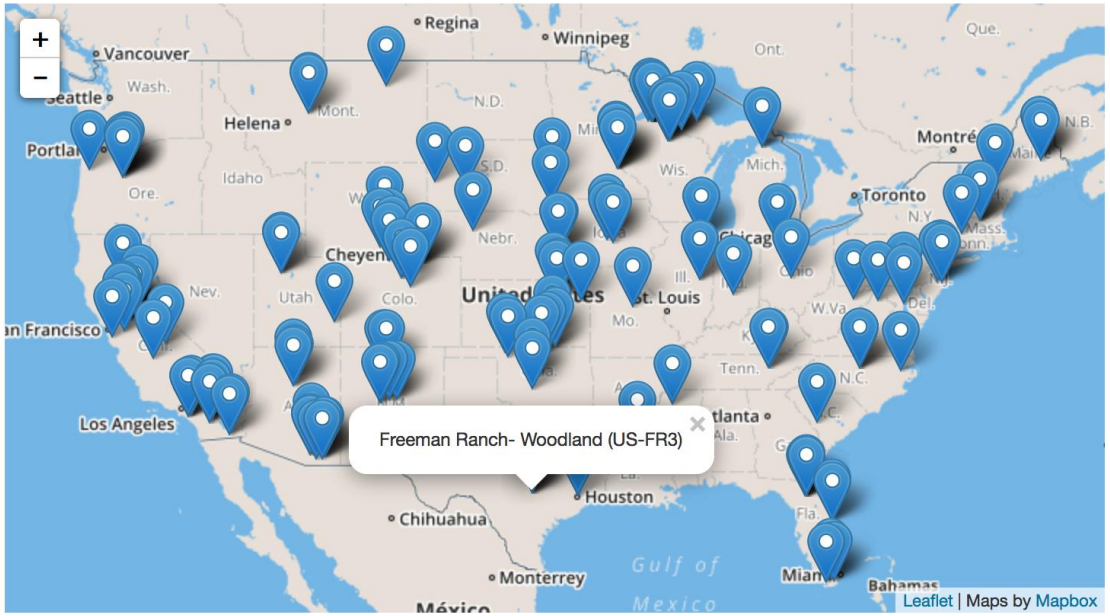
BrownDog Token

c8ac077a-0cd8-4486-8be6-d2bb70c14abb

Model

ed.zip

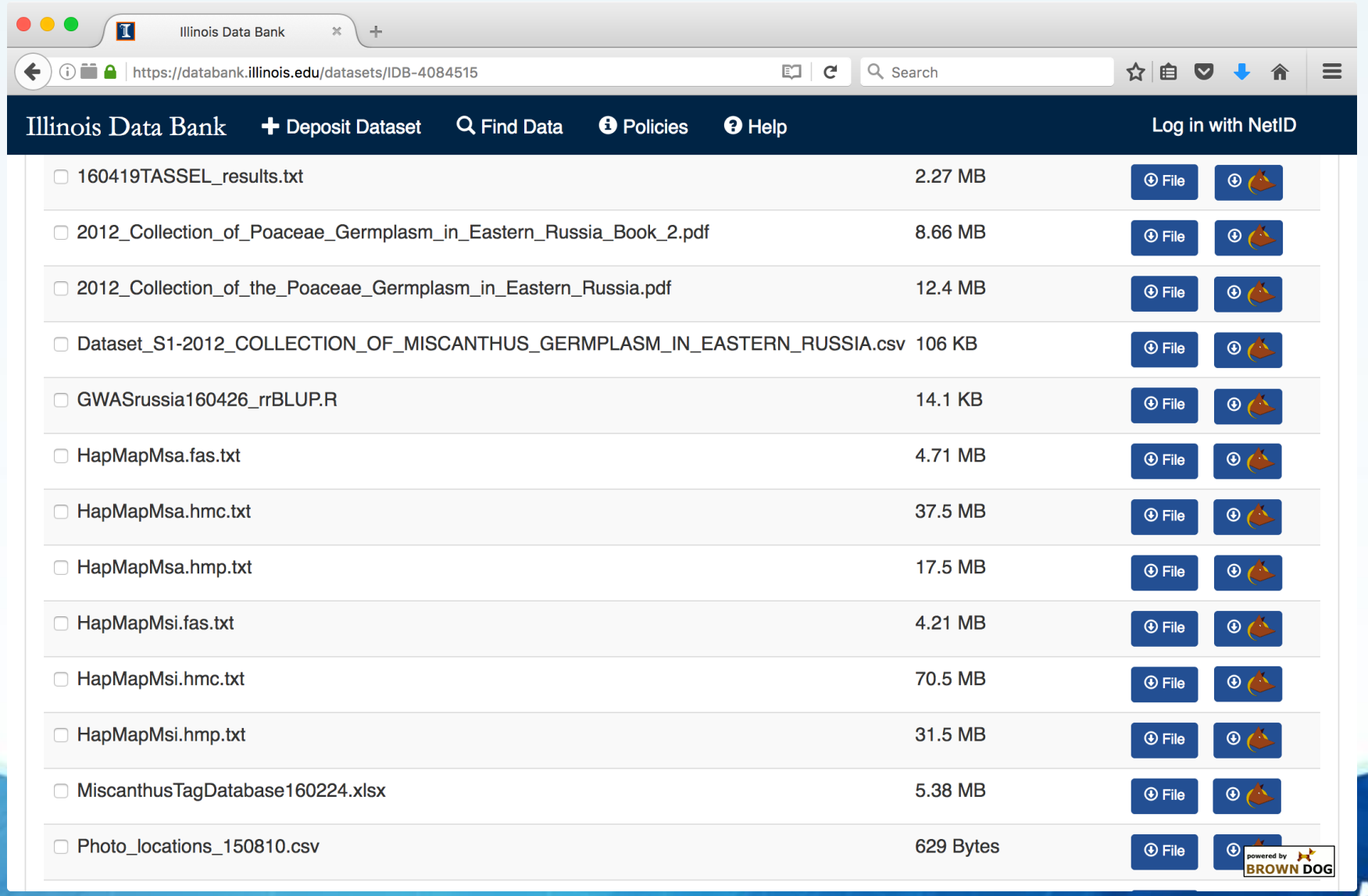
- ameriflux.zip
- clim
- dalec
- ed.zip
- linkages
- pecan.nc
- pecan.zip



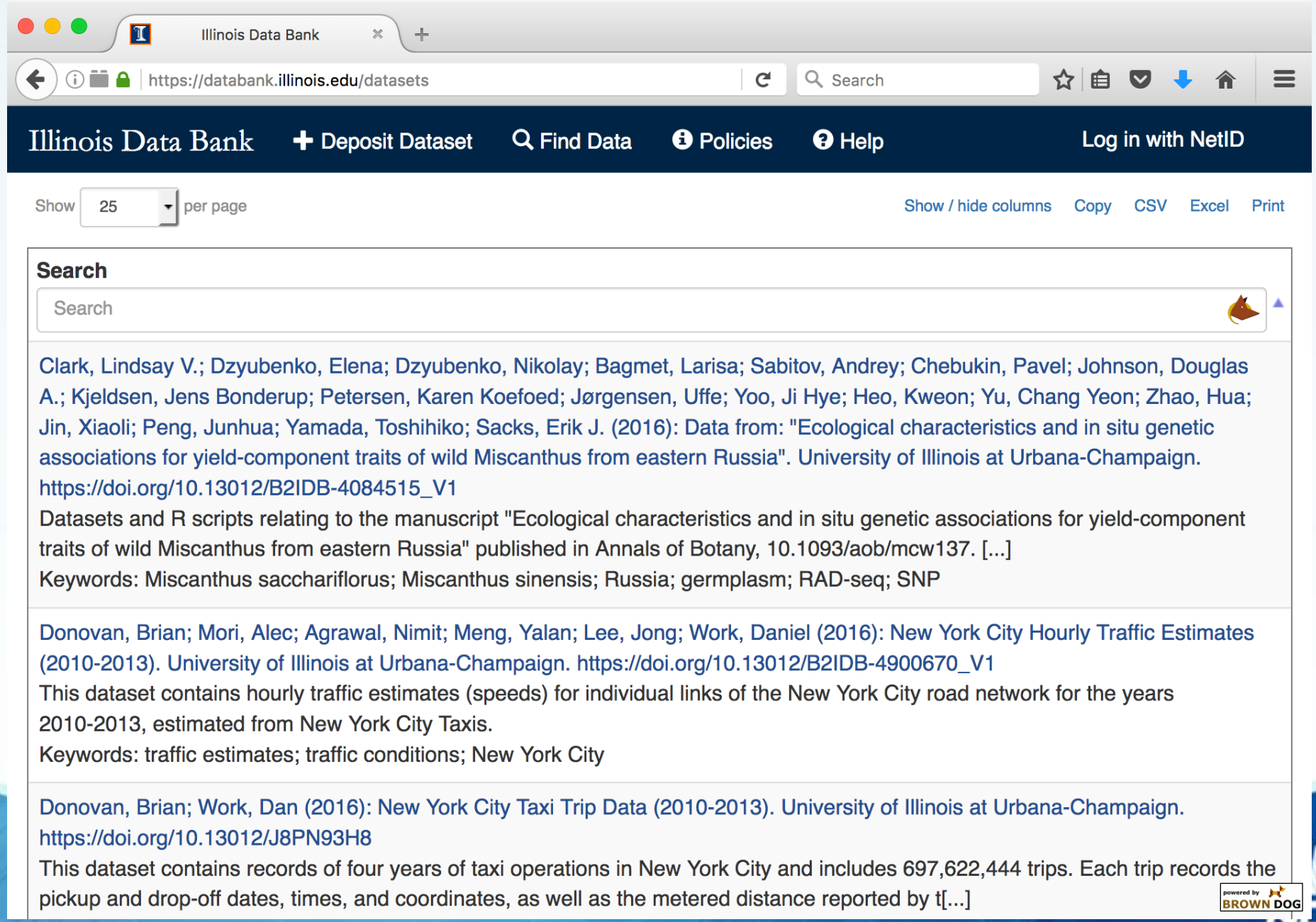

```
<input>
  <type>AmeriFlux</type>
  <site>Freeman Ranch- Woodland (US-FR3)</site>
  <lat>29.94</lat>
  <lon>-97.99</lon>
  <start_date>2001-01-01 00:00:00</start_date>
  <end_date>2001-12-31 23:59:59</end_date>
</input>
```

Download XML Download Data

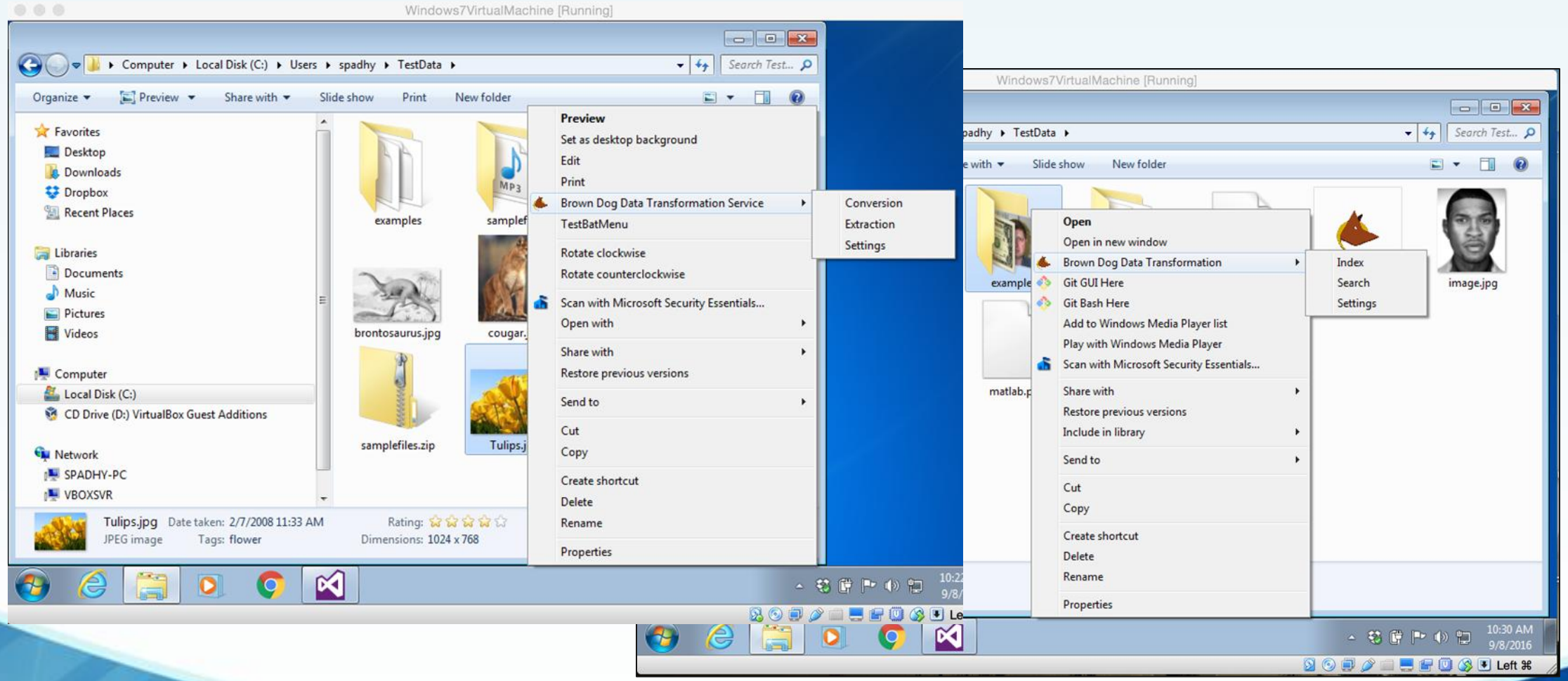
Research Data Services



NCSA

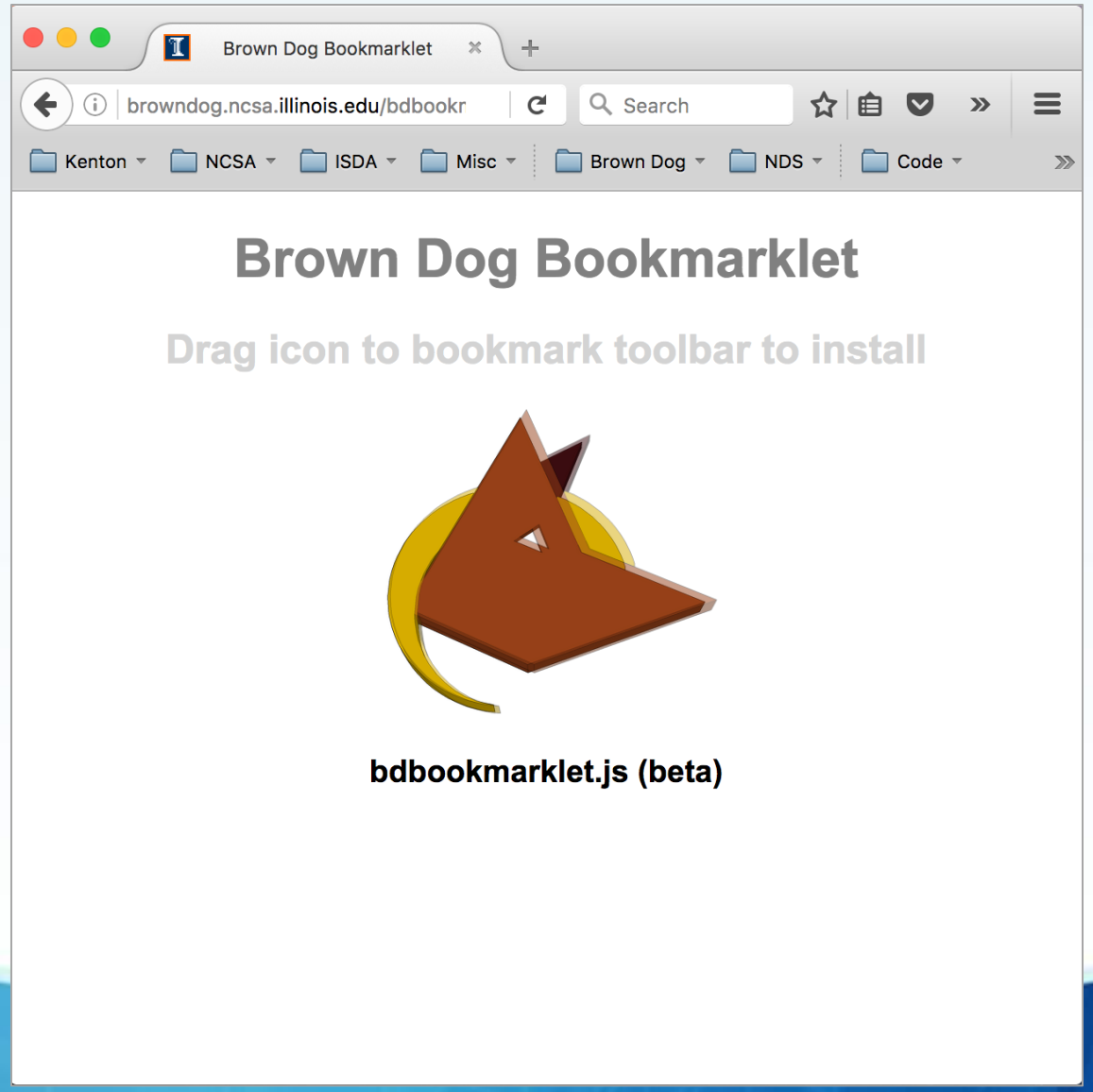

Illinois Data Bank
https://databank.illinois.edu/datasets
Search
Illinois Data Bank
Deposit Dataset
Find Data
Policies
Help
Log in with NetID
Show 25 per page
Show / hide columns
Copy
CSV
Excel
Print
Search
Search
Clark, Lindsay V.; Dzyubenko, Elena; Dzyubenko, Nikolay; Bagmet, Larisa; Sabitov, Andrey; Chebukin, Pavel; Johnson, Douglas A.; Kjeldsen, Jens Bonderup; Petersen, Karen Koefoed; Jørgensen, Uffe; Yoo, Ji Hye; Heo, Kweon; Yu, Chang Yeon; Zhao, Hua; Jin, Xiaoli; Peng, Junhua; Yamada, Toshihiko; Sacks, Erik J. (2016): Data from: "Ecological characteristics and in situ genetic associations for yield-component traits of wild Miscanthus from eastern Russia". University of Illinois at Urbana-Champaign. https://doi.org/10.13012/B2IDB-4084515_V1
Datasets and R scripts relating to the manuscript "Ecological characteristics and in situ genetic associations for yield-component traits of wild Miscanthus from eastern Russia" published in Annals of Botany, 10.1093/aob/mcw137. [...]
Keywords: Miscanthus sacchariflorus; Miscanthus sinensis; Russia; germplasm; RAD-seq; SNP
Donovan, Brian; Mori, Alec; Agrawal, Nimit; Meng, Yalan; Lee, Jong; Work, Daniel (2016): New York City Hourly Traffic Estimates (2010-2013). University of Illinois at Urbana-Champaign. https://doi.org/10.13012/B2IDB-4900670_V1
This dataset contains hourly traffic estimates (speeds) for individual links of the New York City road network for the years 2010-2013, estimated from New York City Taxis.
Keywords: traffic estimates; traffic conditions; New York City
Donovan, Brian; Work, Dan (2016): New York City Taxi Trip Data (2010-2013). University of Illinois at Urbana-Champaign. https://doi.org/10.13012/J8PN93H8
This dataset contains records of four years of taxi operations in New York City and includes 697,622,444 trips. Each trip records the pickup and drop-off dates, times, and coordinates, as well as the metered distance reported by t[...]
powered by BROWN DOG

Windows & Command Line Clients




```
kmchenry@kgm-d3:~/git/bdcli$ bd -o png examples/photo_cd/IMG0001.PCD | bd -v
```

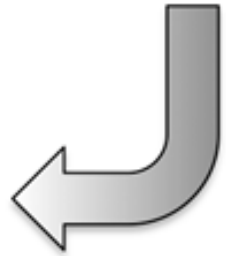
BD Bookmarklet



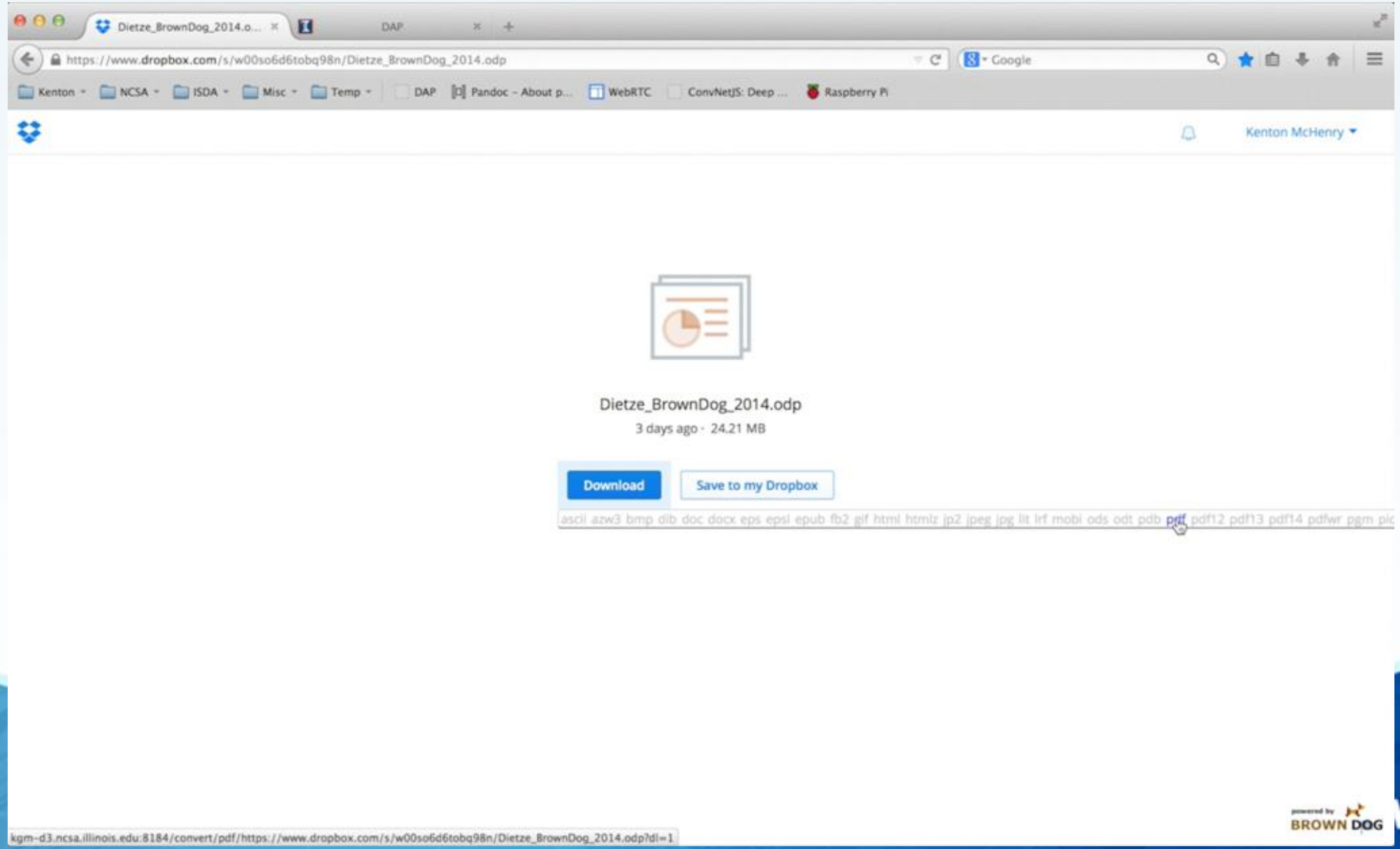
BD Bookmarklet

Name	Last modified	Size	Description
Parent Directory		-	
0005.jp2	2014-07-03 13:49	633K	
0094.jp2	2014-07-03 15:15	7.6M	
105_1.sid	2014-07-03 13:47	3.7M	
aac aif aiff ascii asf avi azw3 bmp clim dalec dib doc docx ed.zip eps epsi epub			
City_Limits.7z	2014-10-01 12:43	763	
Human__knownGene_chr22_1-51304566.bed	2015-04-14 13:57	260K	
IMG_0997.jpg	2016-07-03 23:22	2.1M	
US-Dk3-2001-2003.xml	2015-07-08 22:31	191	
alice.txt	2014-12-27 23:06	141K	

Conversion menus added to links

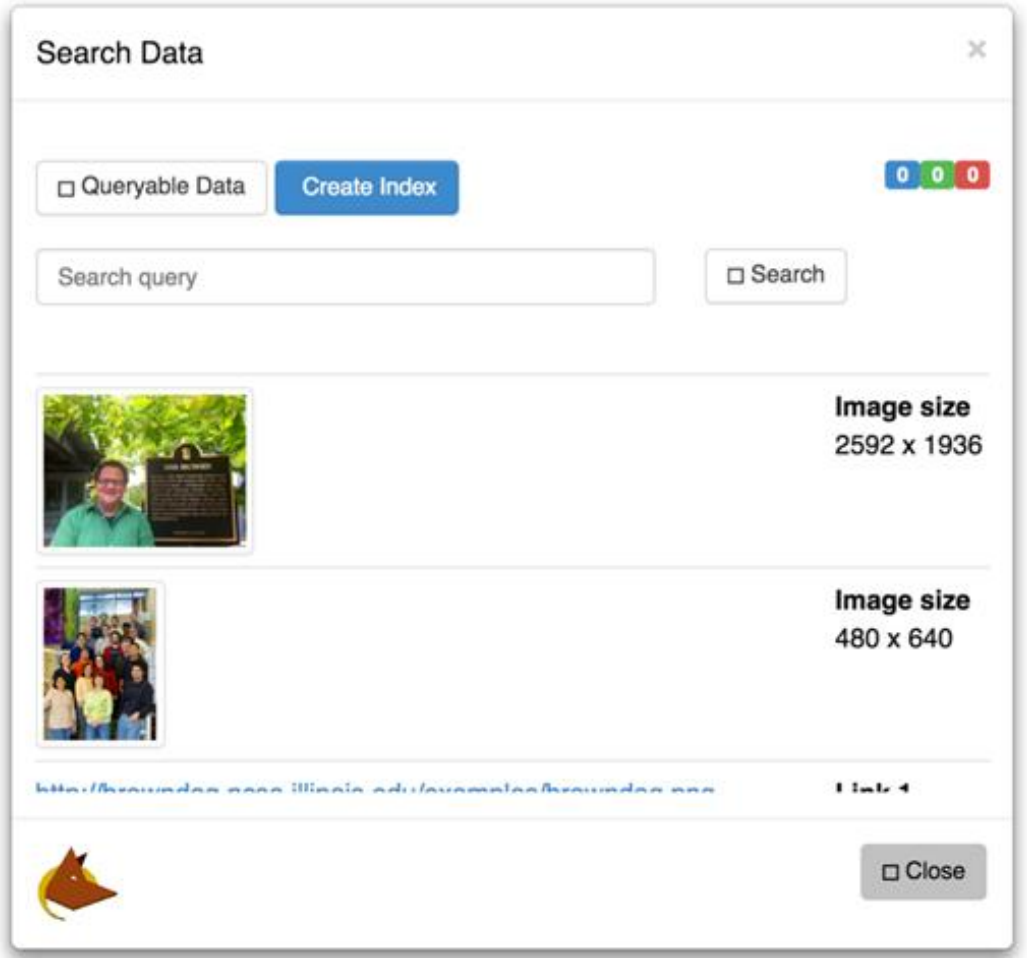


BD Bookmarklet

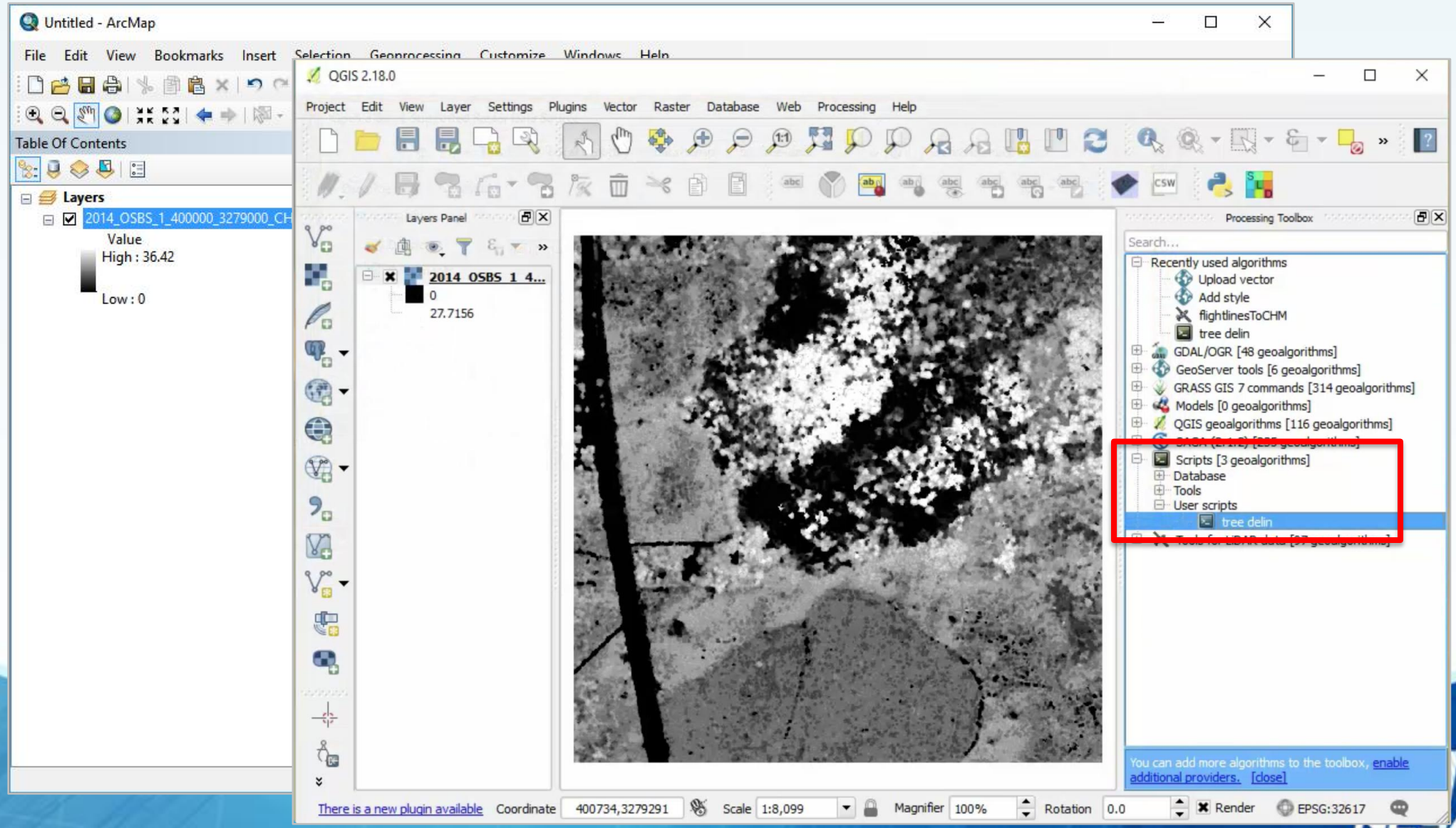


BD Bookmarklet

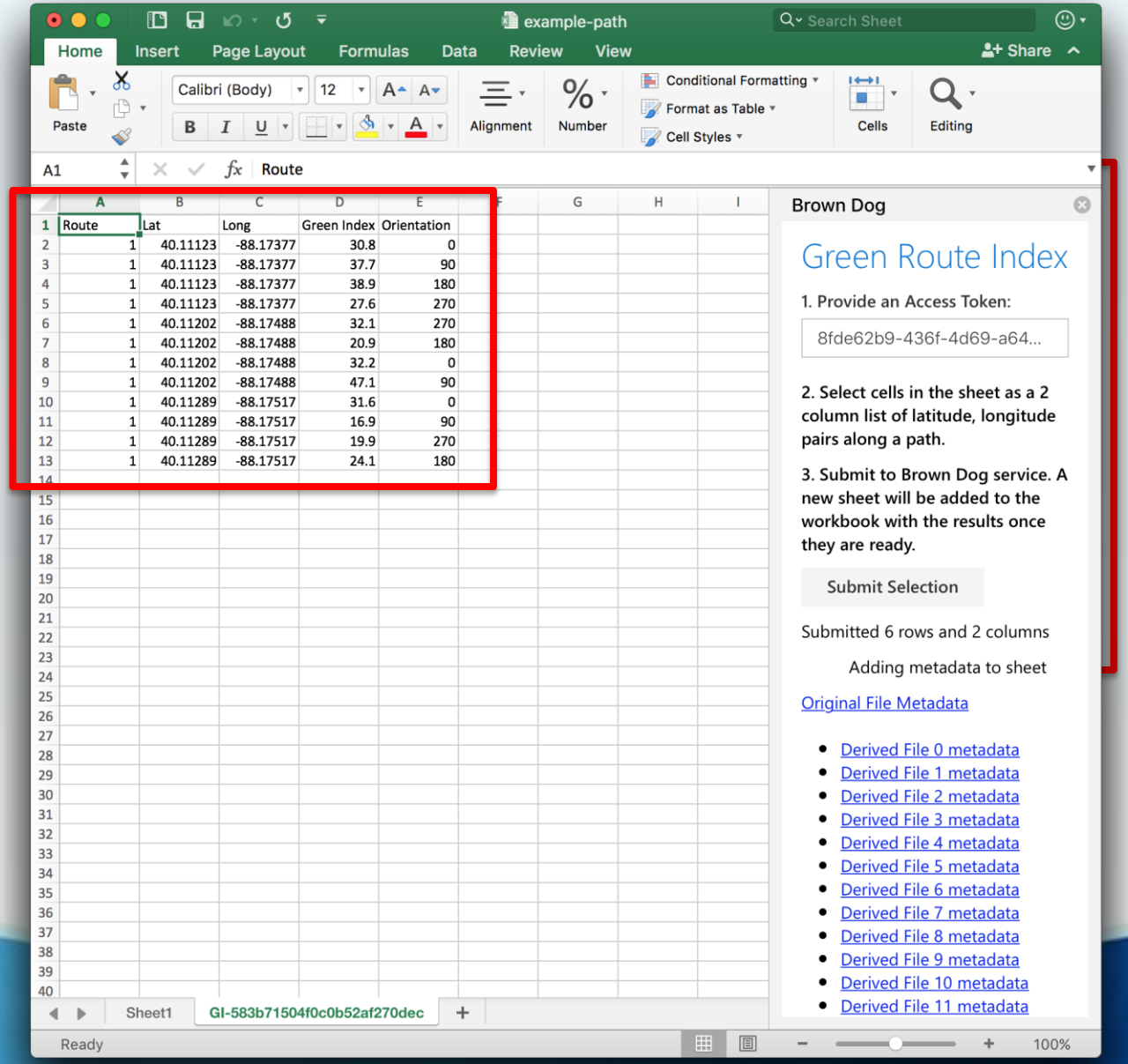
Press Ctrl+F to index and search data on page



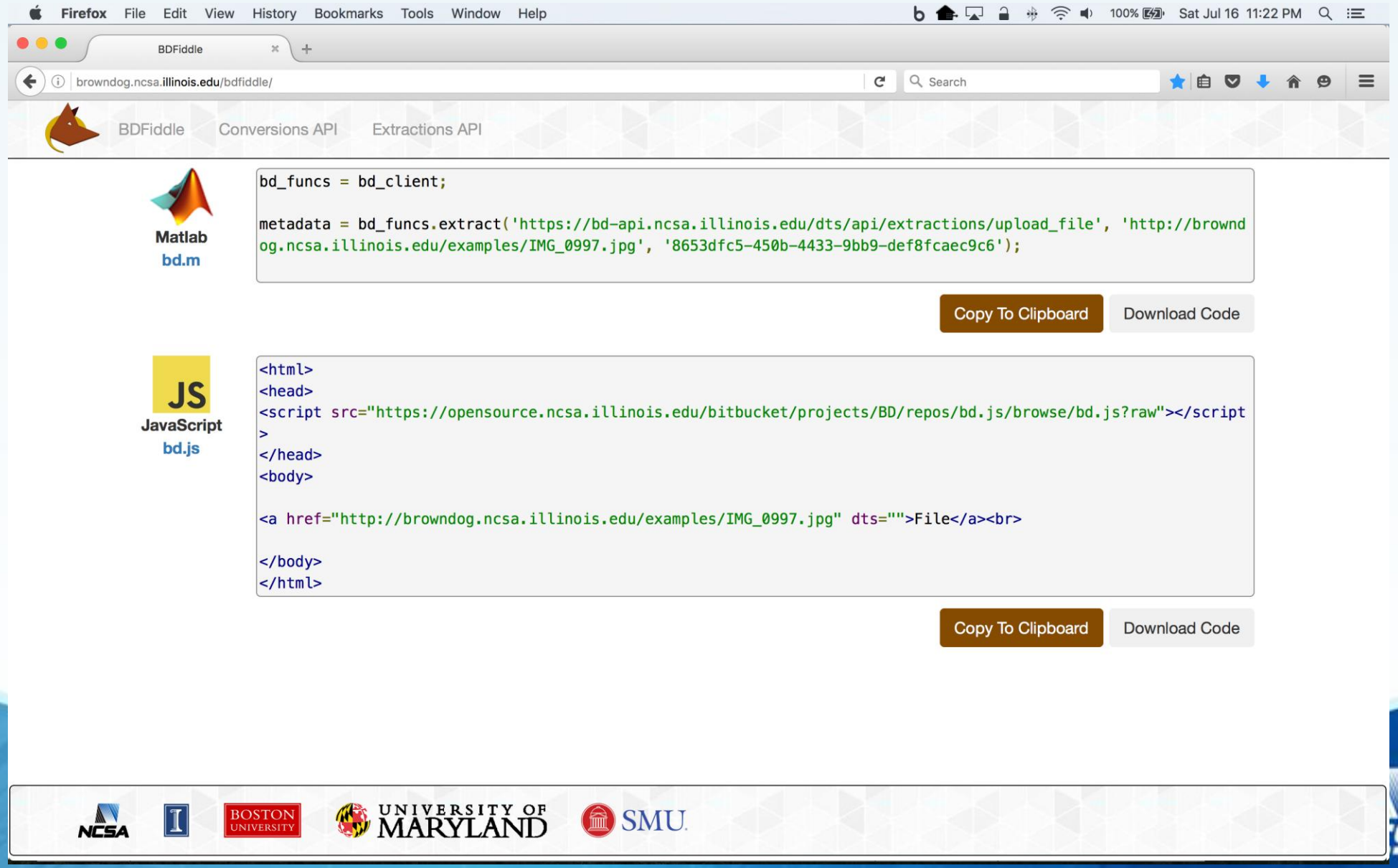
Community Software



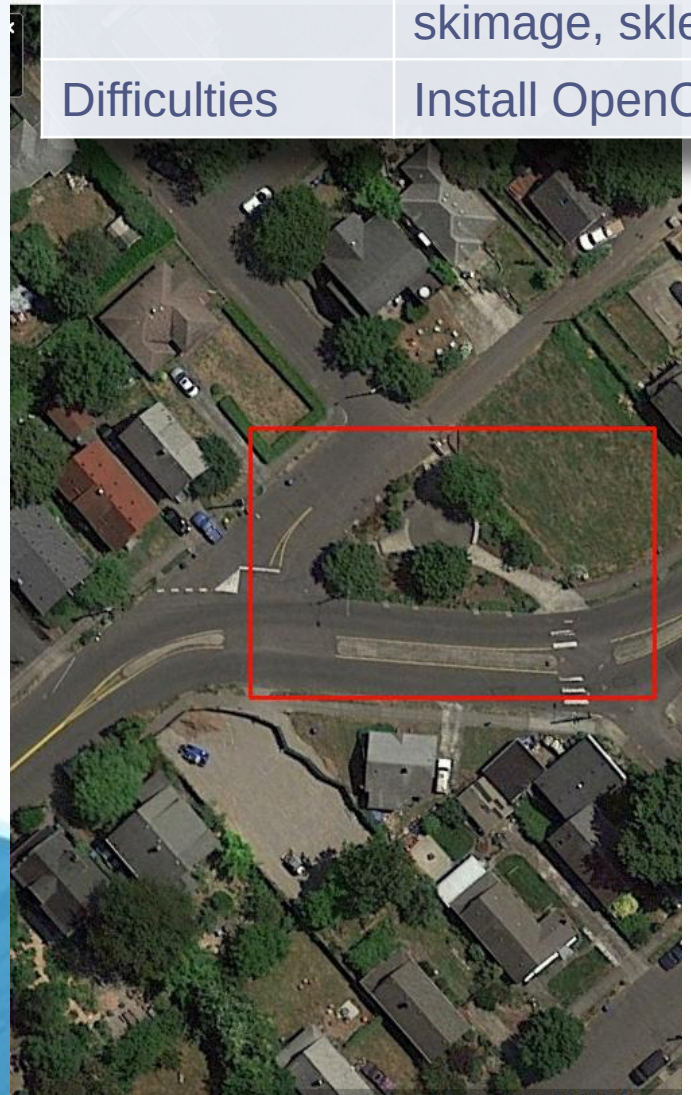
Community Software



BD Fiddle


Firefox File Edit View History Bookmarks Tools Window Help
100% Sat Jul 16 11:22 PM
BDFiddle
browndog.ncsa.illinois.edu/bdfiddle/
Search
BDFiddle
Conversions API
Extractions API
Matlab
bd.m
bd_funcs = bd_client;
metadata = bd_funcs.extract('https://bd-api.ncsa.illinois.edu/dts/api/extractions/upload_file', 'http://browndog.ncsa.illinois.edu/examples/IMG_0997.jpg', '8653dfc5-450b-4433-9bb9-def8fcaec9c6');
Copy To Clipboard
Download Code
JS
JavaScript
bd.js
<html>
<head>
<script src="https://opensource.ncsa.illinois.edu/bitbucket/projects/BD/repos/bd.js/browse/bd.js?raw"></script>
</head>
<body>
File
</body>
</html>
Copy To Clipboard
Download Code
NCSA
BOSTON UNIVERSITY
UNIVERSITY OF MARYLAND
SMU

Lines of Code	273
Other files	Model Image Sample
Dependencies	numpy, argparse, glob, cv2, cPickle, random, h5py, skimage, sklearn, scipy
Difficulties	Install OpenCV (cv2)



```
exposure

and parse the arguments

", required=True,

ize",
    help="Flag indicating whether or not to visualize each iteration")
ap.add_argument("-p", "--path",
    help="path to gi_detector")
args = vars(ap.parse_args())

path_1 = '/Users/ankitrai/Dropbox/ppao_VM/gi_detector/'
def pyramid(image, scale=1.5, minSize=(55,55)):
    # yield the original image
    yield image

    # keep looping over the pyramid
    while True:
        # compute the new dimensions of the image and resize it
        w = int(image.shape[1] / scale)
        image = imutils.resize(image, width=w)

        # if the resized image does not meet the supplied minimum
        # size, then stop constructing the pyramid
        if image.shape[0] < minSize[1] or image.shape[1] < minSize[0]:
            break

        # yield the next image in the pyramid
        yield image
def sliding_window(image, stepSize, windowSize):
    # slide a window across the image
    for y in xrange(0, image.shape[0], stepSize):
        for x in xrange(0, image.shape[1], stepSize):
            # yield the current window
            yield (x, y, image[y:y + windowSize[1], x:x + wi
def detect(image, winDim, winStep=4, pyramidScale=1.5, minPr
        # initialize the list of bounding boxes and associat


        # loop over the image pyramid
    pyramid_layers = pyramid(image,scale=1.5,minSize=(100,10
    for layer in pyramid_layers:
        # determine the current scale of the pyramid
        scale = image.shape[0] / float(layer.shape[0])
        # loop over the sliding windows for the current pyramid layer
        for (x, y, window) in sliding_window(layer, winStep, winDim):
            (winH, winW) = window.shape[:2]
            if winH == winDim[1] and winW == winDim[0]:
            # extract HOG features from the current window and classifiy
```

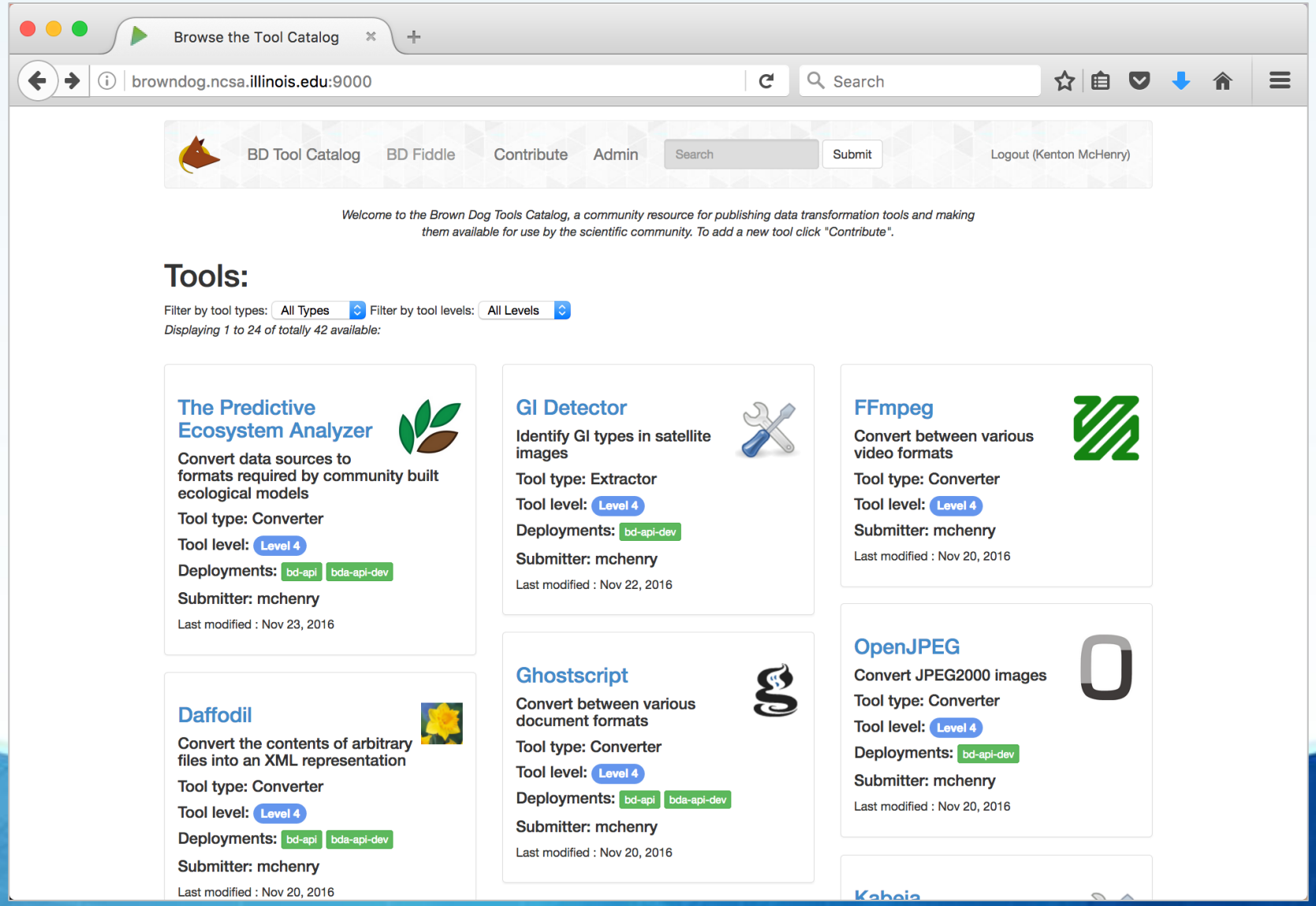
Total Code from 2 Files

Lines of Code	47
Other files	None
Dependencies	bd, requests, os, glob, argparse, time, json, PIL
Difficulties	

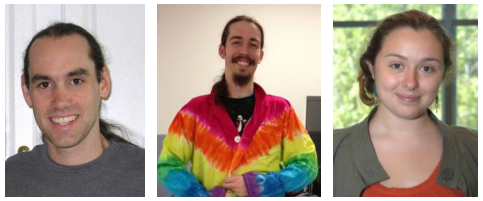

```
# import the necessary packages
import requests
import os
import glob
import argparse
import time
import json
from PIL import Image, ImageDraw
from bd import extract

# construct the argument parser and parse th
ap = argparse.ArgumentParser()
ap.add_argument("-t", "--token", required=Tr
ap.add_argument("-i", "--images", help="Path
ap.add_argument("-b", "--bdapi", help="Brown
args = vars(ap.parse_args())

# define the arguments
token = args["token"]
if args["images"] != None:
    image_path = args["images"]
else:
    image_path = "./images"
if args["bdapi"] != None:
    bdapi = args["bdapi"]
    if bdapi[-1] != "/":
        bdapi = bdapi + "/"
else:
    bdapi = "https://bd-api.ncsa.illinois.edu/"

# Loop through images in image directory
for image_path in glob.glob(image_path + "/*.jpg"):
    metadata = extract(bdapi,image_path,token)['metadata.jsonld']

    for mdata in metadata:
        if "y0" in mdata["content"]:
            x0 =  mdata["content"]["x0"]
            x1 =  mdata["content"]["x1"]
            y0 =  mdata["content"]["y0"]
            y1 =  mdata["content"]["y1"]
    print "coordinates for bioswale bounding box in" +  image_path
    print "[(x0,y0),(x1,y1)] = [(" + str(x0) + "," + str(y0) + "),("+ str(
        x1) + "," + str(y1) + ")]"

    img = Image.open(image_path)
    draw = ImageDraw.Draw(img)
    draw.rectangle([(x0, y0), (x1, y1)])
    img.show()
```

Total Code from 1 File

Allow Researchers to Better Share/Publish Data Transformation Tools

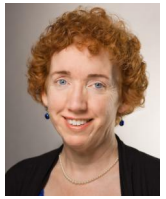


Project Supported Transformations



Domain	Tool	Type	Description
Biology	**netcdf**	Converter	Convert from binary netcdf to text
	PEcAn (Ameriflux)	Converter	Convert Ameriflux data to PEcAn's netcdf CF format
	PEcAn (DALEC)	Converter	Convert PEcAn's netcdf CF format to the format required by the DALEC model
	PEcAn (ED2)	Converter	Convert PEcAn's netcdf CF format to the format required by the ED model
	PEcAn (LINKAGES)	Converter	Convert PEcAn's netcdf CF format to the format required by the LINKAGES model
	PEcAn (Sipnet)	Converter	Convert PEcAn's netcdf CF format to the format required by the Sipnet model
	PEcAn 2.0	Converter	Convert AmerifluxBNL, FLUXNET2015 data to PEcAn's netcdf CF format
	PEcAn 2.0	Converter	Convert PEcAn's netcdf CF format to the format required by the FACE, PALEON, NLDAS, CURNCEP, GLDAS, GFDL, BIOCRO, CLM, GDAY, JULES, LPJ-GUESS, MAAT, MAESPA, PRELES models
	PlantCV	Extractor	Extract plant height, area, and color distribution from photographs.

Project Supported Transformations



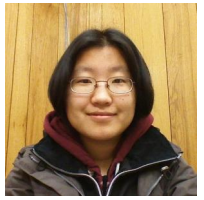

Domain	Tool	Type	Description
Green Infrastructure	**Body of Water Detector**	Extractor	Land coverage, extract locations of bodies of water from **satellite** data
	GI Identification	Extractor	
	Human Preference Score	Extractor	Assign a model derived human preference score to a given image of an urban environment
	Route Greenness	Extractor	Derive the green index of a city route
	Social Media GI Preferences	Extractor	
	Stanford CoreNLP - Sentiment	Extractor	Assign a sentiment score to a piece of text
	TUV Triaxus	Extractor	

Project Supported Transformations

Domain	Tool	Type	Description
Hydrology	**Advection Diffusion**	Extractor	Solve a general advection-dispersion equation
	Chemical Mean Age	Extractor	Determine the mean age of chemical constituents with inputs of chemical dynamics
	Document Tables Extractor	Extractor	Extract tables from documents
	GDAL	Extractor	
	Historical River Extractor	Extractor	Extract the river networks from the ancient hand-drawing maps and compare them with current river networks
	Normalized Difference Vegetation Index	Extractor	
	River Chi Index	Extractor	Identify the river dynamics in a river basin and evaluate human activities' influences through Chi index in the streams
	River Sinuosity	Extractor	Study the maturity and equilibrium conditions of a stream through the sinuosity index.
	Soil Moisture Change	Extractor	Determine role of hydraulic redistribution in AZ (riparian site / upland site) by studying soil moisture change throughout different seasons
	Species Classifier	Extractor	SAM based Species Classification from **Hyperspectral** data, Hyperspectral Indices, NDVI, SAVI, MSAVI, etc.

Project Supported Transformations



Domain	Tool	Type	Description
Hydrology	**TerEx**	Extractor	Identify the flat polygons and the heights inside a river valley.
	Topographic Depressions	Extractor	Identify topographic depressions (TDs) and their distribution on landscape (Number, location, area, volume of TDs).
	Tree Delineation	Converter	Tree-wise voxelization of waveform data for **lidar** metrics that describes canopy structure (max intensity, height, etc...). Individual tree delineation, tree leaf area density to describe vertical leaf distributio.
	Valley Safety Zones	Extractor	Estimate submerging areas and water depths under extreme floods and map the safety zones in a river valley.
	Vegetation Indices	Extractor	Calculating vegetation indices such as NDVI and Surface Temperature from Landsat 7 and 8 **satellite** data.

Project Supported Transformations

Domain	Tool	Type	Description
General	**Tika**	Extractor	Document extractions such as language identification, ...
	txt2html	Converter	Convert text documents to HTML
	Versus – Color Distribution	Extractor	Generate a distribution of color values within an image to be used for comparing how similar two images are
	VLFeat	Extractor	Classify images as to whether they contain objects from the Caltech101 dataset (e.g. people, airplanes, motorcycles, cougars, ...)
	Zip	Converter	Unzip zip archives
	Siegfried	Extractor	Extract information about a given file relevant to identifying its type and validating its format
	Stanford CoreNLP	Extractor	Natural Language Process extractions such as parts of speech, named entities, langauge, etc.
	Tesseract	Extractor	Object Character Recognition (OCR) to extract text from images containing text

82 transformations to be part of beta …

http://browndog.ncsa.illinois.edu

@NCSABrownDog