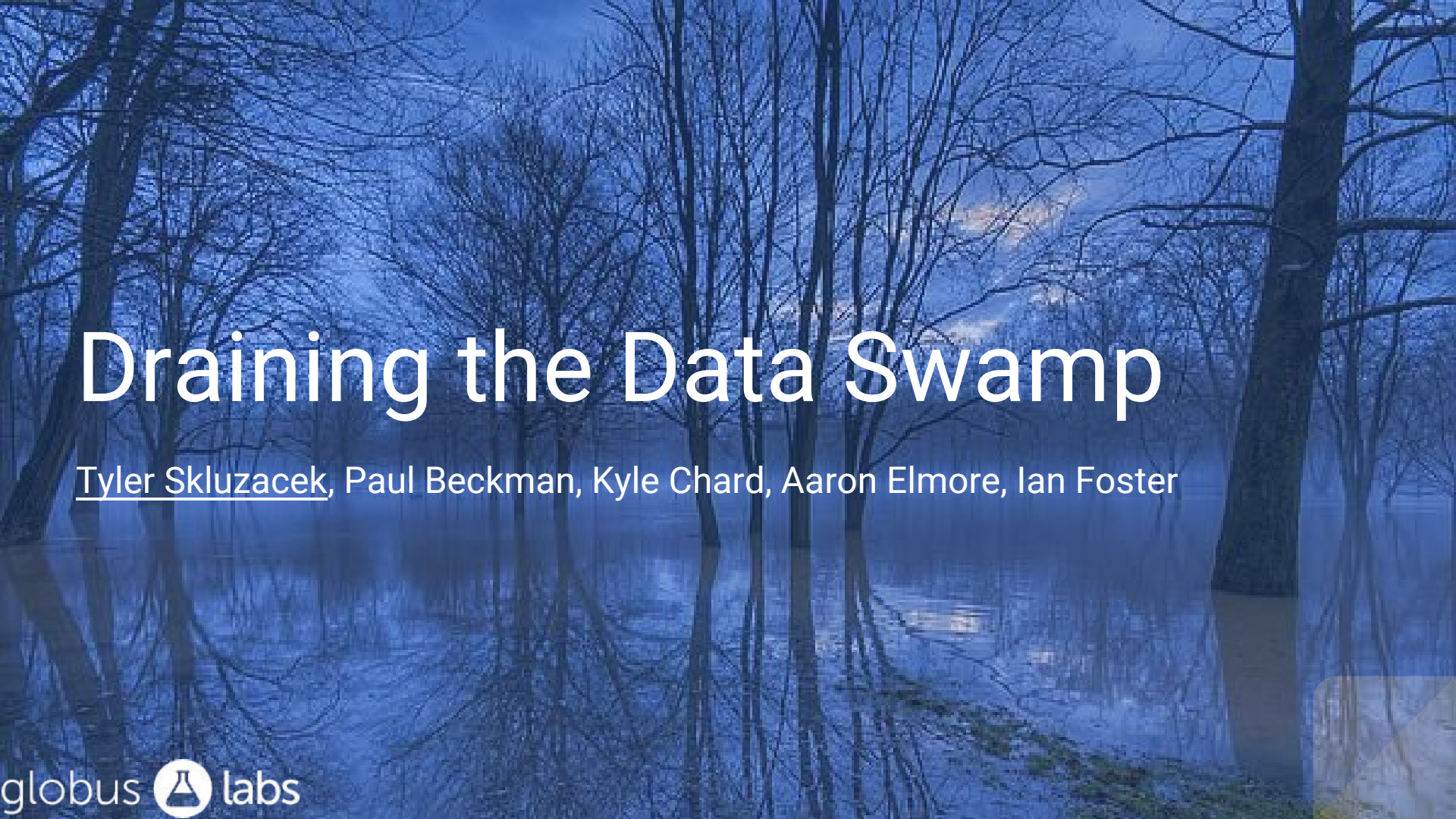
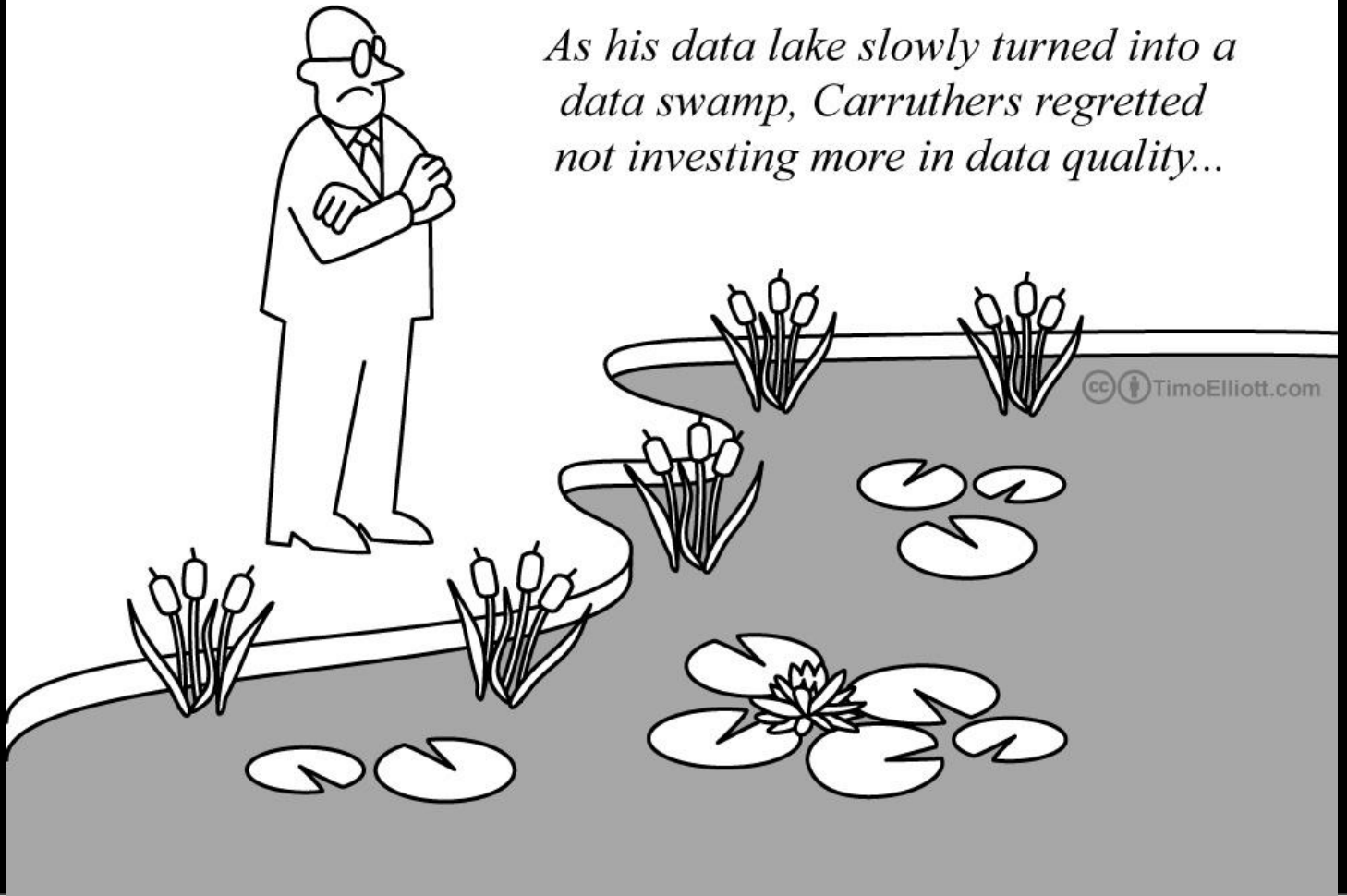


Draining the Data Swamp

Tyler Skluzacek, Paul Beckman, Kyle Chard, Aaron Elmore, Ian Foster

globus labs


As his data lake slowly turned into a data swamp, Carruthers regretted not investing more in data quality...
TimoElliott.com

Dealing with Data Pollution

College	Alaska	N	11249	37
Fairbanks	Alaska	N	30843	108
Anchorage	Alaska	N	226338	827
Juneau	Alaska	Y	26751	99
Bellingham	Washington	N	52179	211
Havre	Montana	N	10201	40
Anacortes	Washington	N	11451	46
Mount Vernon	Washington	N	17647	68
Oak Harbor	Washington	N	17176	59
Minot	North Dakota	N	34544	139
Kalispell	Montana	N	11917	52
Williston	North Dakota	N	13131	51
Port Angeles	Washington	N	17710	73
North Marysville	Washington	N	18711	61
Marysville	Washington	N		
West Lake Stevens	Washington	N		
Everett	Washington	N		

Null = -999


```
**I Transfer logging started at Thu Mar 22 15:18:52 IST 2007 I**
**I Transfer process 1 of 2 completed with status = 0 I**
file:/C:/DOCUME~1/VSREER~1.ST-/LOCALS~1/Temp/transfer/bridges/null-nullMy_Metadata_Transfer1
org.xml.sax.SAXException: file:/C:/DOCUME~1/VSREER~1.ST-/LOCALS~1/Temp/transfer/bridges/null-n
    at oracle.cwm.bridge.parse.XMIParseHandler.fatalError(XMIParseHandler.java:753)
    at oracle.xml.parser.v2.XMLError.flushErrorHandler(XMLError.java:425)
    at oracle.xml.parser.v2.XMLError.flushErrors1(XMLError.java:284)
    at oracle.xml.parser.v2.NonValidatingParser.parseRootElement(NonValidatingParser.java:329)
    at oracle.xml.parser.v2.NonValidatingParser.parseDocument(NonValidatingParser.java:291)
    at oracle.xml.parser.v2.XMLParser.parse(XMLParser.java:229)
    at oracle.cwm.bridge.parse.XMIParse.XMIParseInit(XMIParse.java:253)
    at oracle.cwm.bridge.parse.XMIParse.parse(XMIParse.java:155)
    at oracle.cwm.vek.xmi.XMLInputStream.readPackage(XMLInputStream.java:121)
    at oracle.cwm.vek.mapping.XMIPhysMappingEngine.decode(XMIPhysMappingEngine.java:116)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
    at java.lang.reflect.Method.invoke(Method.java:324)
    at oracle.cwm.vek.mapping.TMPS.loadModel(TMPS.java:188)
    at oracle.cwm.OM.toowb.toOWB.runBridge(toOWB.java:827)
    at oracle.cwm.tools.bridge.BridgeWrapper.run(BridgeWrapper.java:509)
```

Index of ftp://cdiac.ornl.gov/

Up to higher level directory

Name	Size	Last Modified
README	1 KB	02/09/2000 12:00:00 AM
incoming		03/10/2017 11:32:00 AM
key		02/19/1992 12:00:00 AM
pub		04/03/2017 03:29:00 PM
pub10		07/14/2014 12:00:00 AM
pub11		11/02/2011 12:00:00 AM
pub12		11/14/2013 12:00:00 AM
pub2		01/31/2017 12:43:00 PM
pub3		10/14/2003 12:00:00 AM
pub4		10/02/2008 12:00:00 AM
pub5		10/14/2003 12:00:00 AM
pub6		10/13/2005 12:00:00 AM
pub7		10/14/2003 12:00:00 AM
pub8		12/04/2013 12:00:00 AM
pub9		10/13/2005 12:00:00 AM

Context

Carbon Dioxide Information Analysis Center

The Carbon Dioxide Information Analysis Center (CDIAC), located at the U.S. Department of Energy's (DOE) Oak Ridge National Laboratory (ORNL), is the primary climate change data and information analysis center for DOE. CDIAC is supported by DOE's Climate and Environmental Sciences Division within the Office of Biological and Environmental Research (BER).

CDIAC's data holdings include estimates of carbon dioxide emissions from fossil-fuel consumption and land-use changes; records of atmospheric concentrations of carbon dioxide and other radiatively active trace gases; carbon cycle and terrestrial carbon management datasets and analyses; and global/regional climate data and time series.

CDIAC provides scientific and data management support for numerous projects including large-scale DOE ecosystem

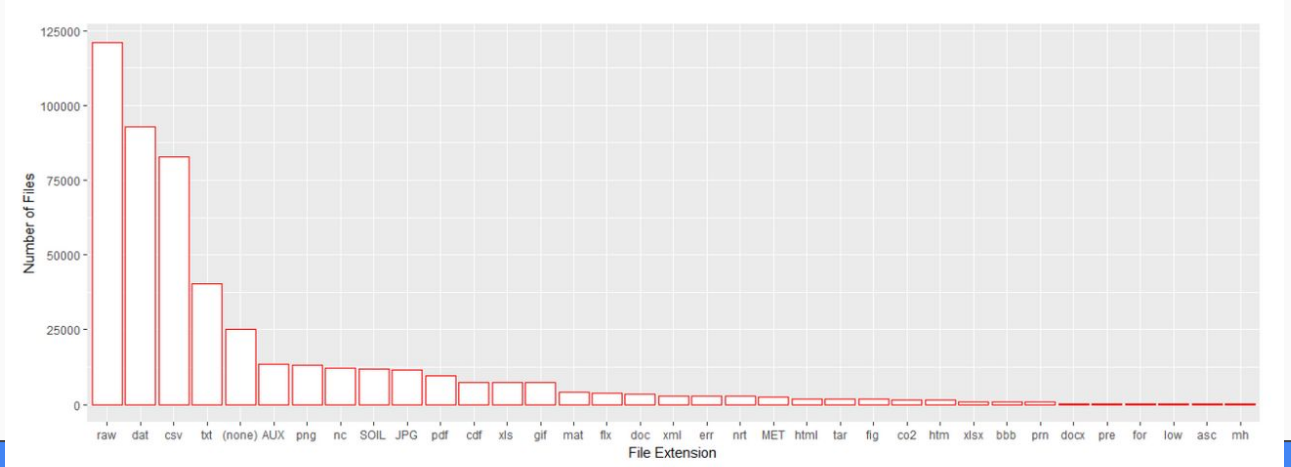
Since its inception in 1968, the Climate Monitoring and Diagnostics Laboratory (CMDL) [known before 1989 as the Geophysical Monitoring for Climati Change (GMCC) group] of the National Oceanic and Atmospheric Administration (NOAA) has developed a network of flask sampling sites for the analysis of atmospheric CO2 (Komhyr et al. 1985). Beginning on an experimental basis in April 1983, NOAA/CMDL expanded its flask sample analysis to include methane as well as CO2 (Lang et al. 1990a). The sampling network now includes 37 fixed sites, ranging in latitude from 82 degrees N to 90 degrees S (Lang et al. 1990b). Collection sites are typically located in remote areas to ensure that samples are representative of a large, well-mixed volume of the atmosphere (Steele et al. 1987). In 1986, the NOAA/CMDL cooperative air sampling network was expanded to include a program of shipboard measurements (Lang et al. 1992) Currently, methane data from shipboard sampling are available for 5 degree latitude intervals in the Pacific Ocean from two cruise vessels [Southland Sta (PAC) and Wellington Star (PAW)] traveling between North America and New Zealand. Shipboard data are also available for 3 degree latitude intervals in the South China Sea (SCS) from two cruise vessels (Carla A. Hills and Great Promise) traveling between Singapore and Hong Kong.

The earliest methane data from the NOAA/CMDL cooperative air sampling network are from January 1983, and come from three of the remote sites: Amsterdam Island, Halley Bay, and Palmer Station. (Collection began at these sites first, in anticipation of the long delay between sample collection and analysis in Boulder, Colorado.) Over the entire period 1983-1993, air samples were collected at 44 fixed sites, 37 of which were still active at the end of 1993. Seventeen sites contributed samples for one or more months of each year during 1983-1993. Twenty-three other sites began sampling at some time after 1983; of these, four were discontinued before 1993. Four other sites began sampling in 1983 but also were later discontinued. Detailed descriptions of sample collection, storage, and analysis methods are given in Steele et al. (1987, 1992), Lang et al. (1990a, 1990b, 1992, 1994), and Dlugokencky et al. (1994b). Brief summaries of these methods are given below.

A variety of flask types and sample collection methods have been used in

Solution: Drain the Data Swamp

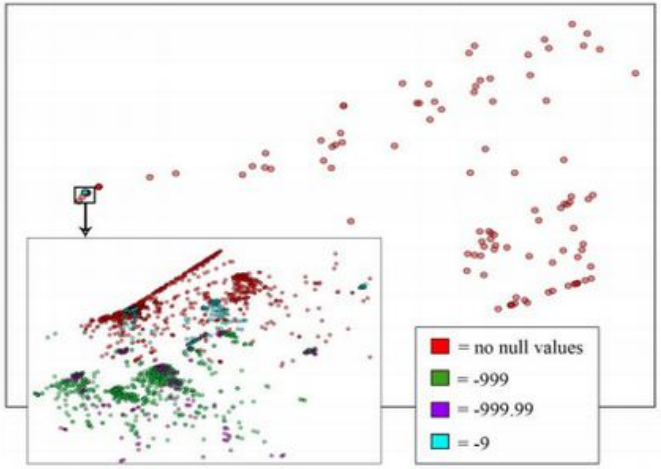
- Data Wrangling / Ad Hoc Metadata Gymnastics
- Content vs. Context



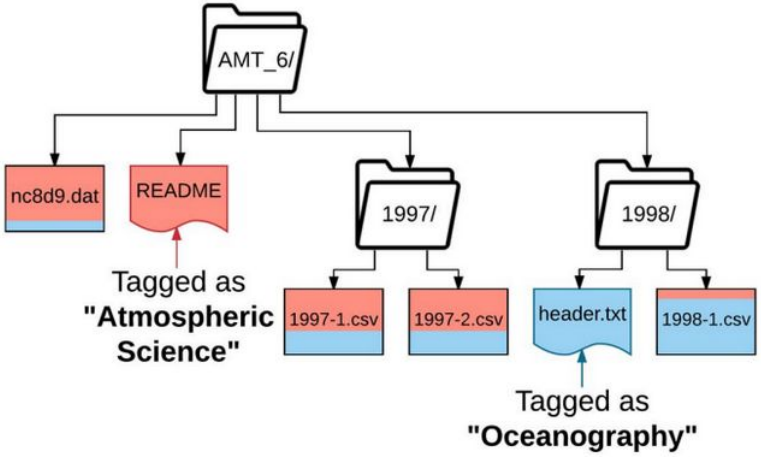
Histogram of top files on a scientific Data Lake

Skluma… and beyond!

- Metadata and Schema Extraction
  - Need to be both fast and accurate (i.e., efficient)
- Defeat nulls and headerless ‘nonsense’
- Crawl data for contextual relationships
- Fixed use case: CDIAC
  - 500,000+ files, 150+ file types



PCA Analysis to detect implied nulls



The Future

- Extraction of image features for topic assignment
- Interactive metadata enrichment
  - “People are lazy, metadata is hard”
  - Creating conversations between chatbots and scientists
- Expand our use case to… wherever is biggest.

Conclusion

- The Data Swamp is growing --- it needs to be drained.
- Building scientific repo enhancers.
  - Remove the trash
  - Supplement the good stuff
- Move on to bigger and better (worse) data.