

SatRday
Chicago
2019

MISADVENTURES *IN* BIODIVERSITY DATA



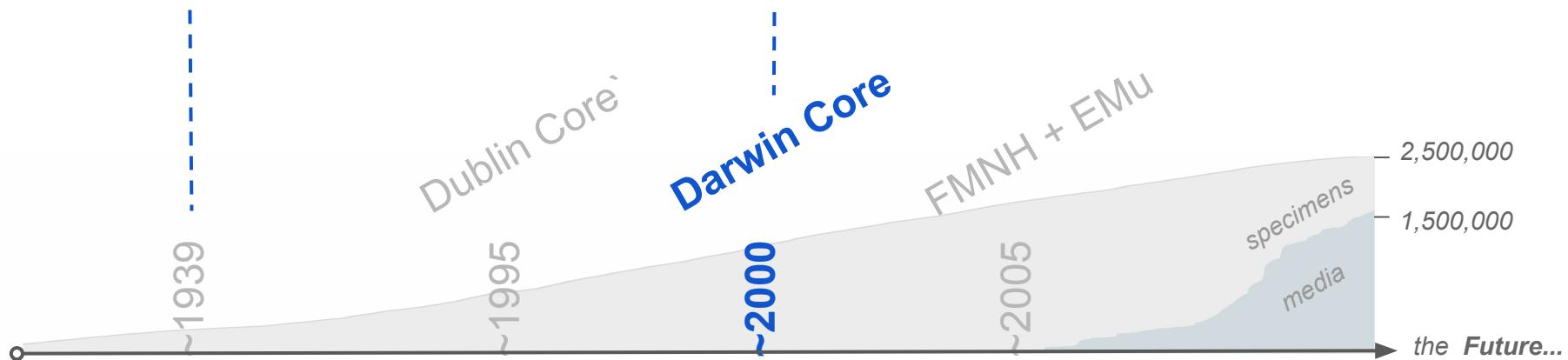
Kate Webbink
Technology Department
Information Systems Specialist

Biodiversity Data

...Standards

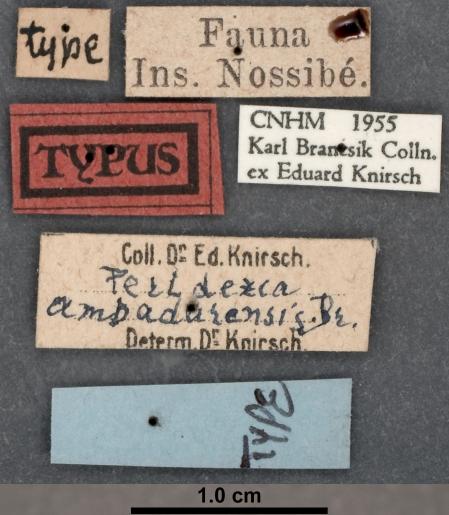


```
recordedBy: "J. W. Woodhead",
country: "Australia",
decimalLatitude: "-42",
decimalLongitude: "147",
scientificName: "Vombatus ursinus hirsutus",
basisOfRecord: "PreservedSpecimen",
occurrenceID: "8864ef77-15ff-4ea8-bd97-f8bc396d7c8f",
catalogNumber: "49085",
collectionID: "Mammals",
year: "1939"
```



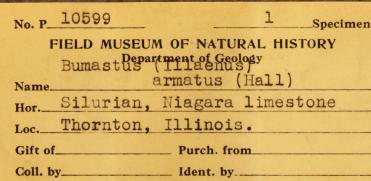
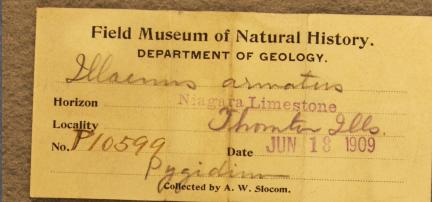
ZOOLOGY

OccurrenceID:
823b4c38-ed49-4444-ad72-2fb2c8bb60de



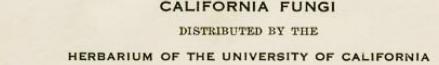
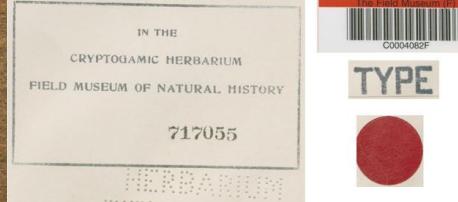
GEOLOGY

OccurrenceID:
793d4505-0e14-4d52-a2cf-6f780b50cf3a



BOTANY

OccurrenceID:
4bf6f930-105f-4749-895e-56478f3e9573

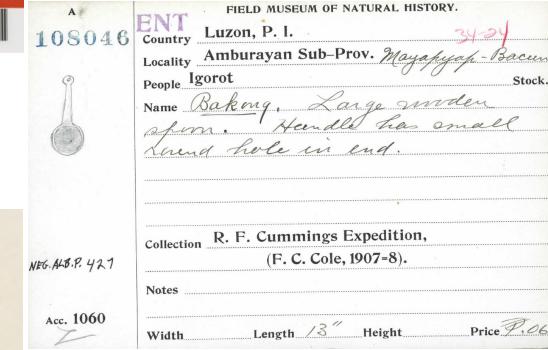


Host, *Alsophila quadripinnata* C. Chr.
Cerro del Boqueron, Mexico,
September, 1913. Coll. C. A. Purpus.
Type collection.



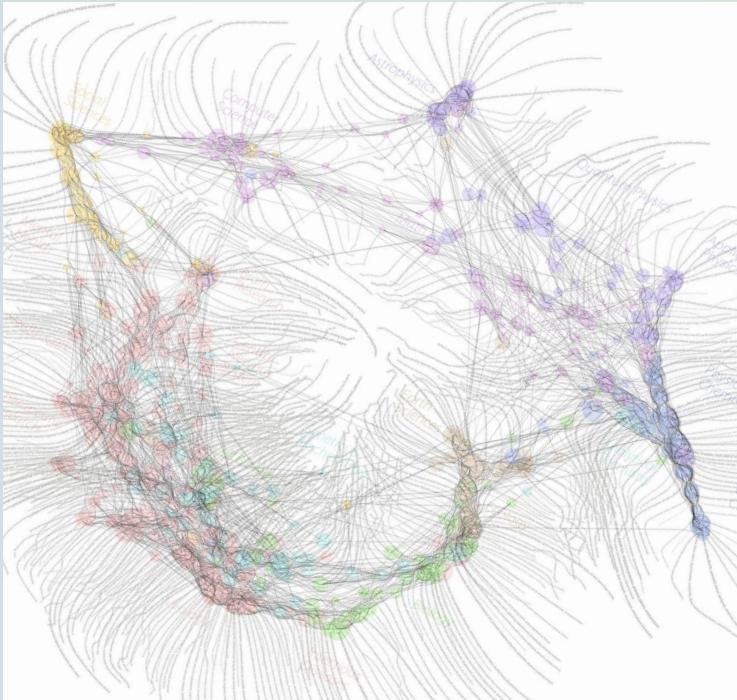
ANTHRO.

GUID:
d8f1d60c-701c-4430-8f04-227260a0f842



Diversity...

...is great!



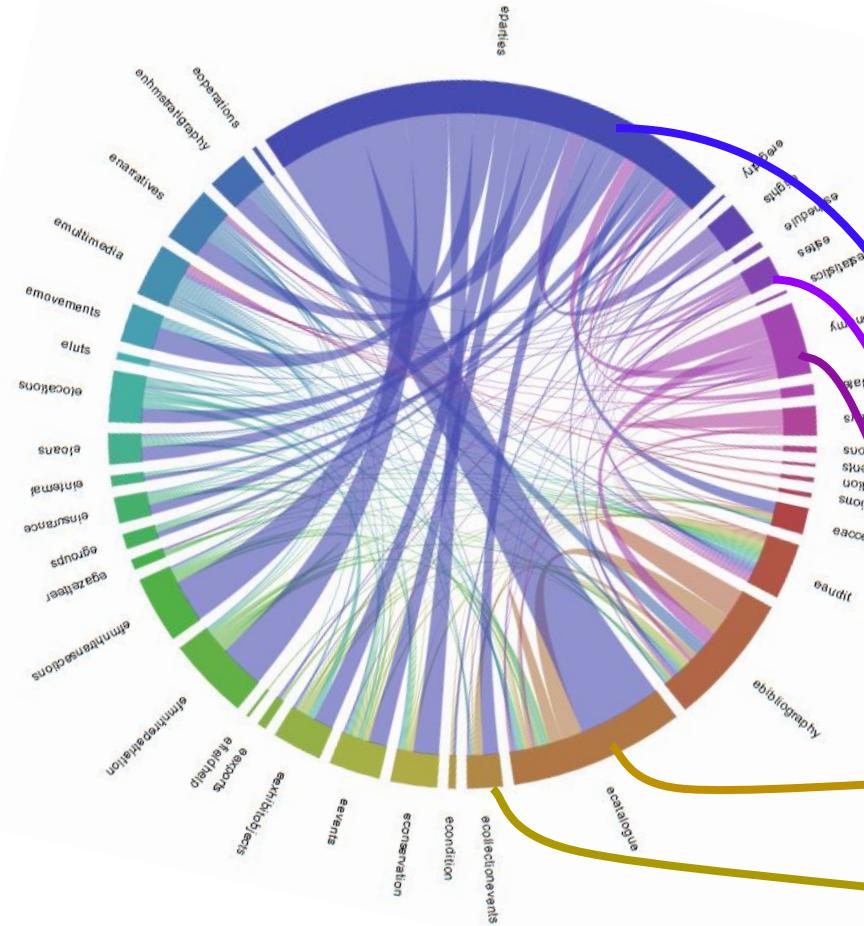
But maybe not for data-structure.



...

How to
build + maintain
ONE SYSTEM
for
DIVERSE DATA?

Database structure



<https://github.com/fieldmuseum/Collections-Scripts/>

- + [ShinySchemaVis](#)
- + [EMu database schema](#)

```
recordedBy: "J. W. Woodhead",  
country: "Australia",  
decimalLatitude: "-42",  
decimalLongitude: "147",  
scientificName: "Vombatus ursinus hirsutus",  
basisOfRecord: "PreservedSpecimen",  
occurrenceID: "8864ef77-15ff-4ea8-bd97-f8bc39",  
catalogNumber: "49085",  
collectionID: "Mammals",  
year: "1939"
```

networkD3::chordNetwork()

EMu Schema
text file

```

eaccessionLots :>
    Colnames => 'eaccessionLots'
    Coltypes => 'Bikem'
    Colcheck => 'Bikem'
    Colupdate => 'Bikem'
    Coldele => 'Bikem'

    #and many others

```

```

Audit :>
    Colnames => 'Audit'
    Coltypes => 'Bikem'
    Colcheck => 'Bikem'
    Colupdate => 'Bikem'
    Coldele => 'Bikem'

    #and many others

```

prep-script
tidy, zoo

```

# Setup Link-table of Source/Target nodes
schemaB <- schema[!(schema$type=="table", c("source", "target"))]
schemaB <- schemaB[order(schemaB$source, schemaB$target),]

schemaBlevels <- unique(append(schemaB$source, schemaB$target))
schemaBlevels <- as.factor(schemaBlevels[order(schemaBlevels)])

schemaB$sourceNum <- factor(schemaB$source,
                             levels = schemaBlevels)
schemaB$targetNum <- factor(schemaB$target,
                             levels = schemaBlevels)

schemaB$sourceNum <- as.numeric(schemaB$sourceNum) - 1
schemaB$targetNum <- as.numeric(schemaB$targetNum) - 1

# Show # of attachments b/t tables as line width
schemaBlinks <- dplyr::count(schemaB, sourceNum, targetNum)
schemaBNodes <- data.frame("id" = as.numeric(schemaBlevels) - 1,
                           "name" = as.character(schemaBlevels),
                           stringsAsFactors = FALSE)

schemaB2 <- dplyr::count(schemaB, source, target)
g <- graph.data.frame(schemaB2, directed = FALSE)
schemaMatrix <- get.adjacency(g,
                               attr = "n",
                               sparse = FALSE)

```

matrix
igraph::
graph.data.frame()

```

schemaB2 <- dplyr::count(schemaB,
                          source,
                          target)

g <- graph.data.frame(schemaB2,
                      directed = FALSE)

schemaMatrix <- get.adjacency(g,
                               attr = "n",
                               sparse = FALSE)

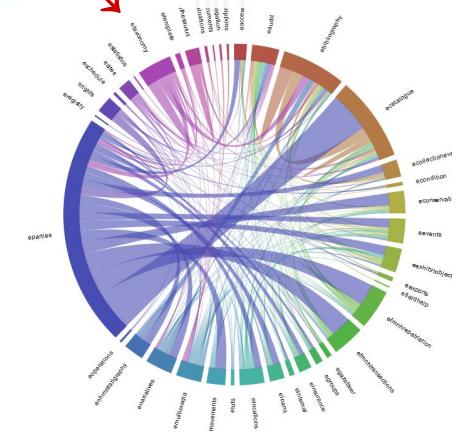
```

networkD3::chordNetwork()

```

chordNetwork(Data = schemaMatrix,
             height = 850, width = 800,
             initialOpacity = 0.6,
             colourScale = col2hex(palette(rainbow(NROW(colnames(sch
# colourScale = input$file$colorSchema, # schemaRainbow,
fontSize = 10,
padding = 0.035,
labels = colnames(schemaMatrix),
labelDistance = 130,
pdf(file = NULL))

```



	eaccessionlots	eaudit	ebibliography
eaccessionlots	0	1	
eaudit	1	1	
ebibliography	0	1	
ecatalogue	1	1	
ecollectionevents	1	1	
econdition	0	1	
econservation	0	1	
eevents	0	1	
eexhibitobjects	1	1	
eexports	0	1	
efieldhelp	0	1	

Registry / Admin Patterns

- `sunburstR::sunburst()`

EMu Registry
CSV



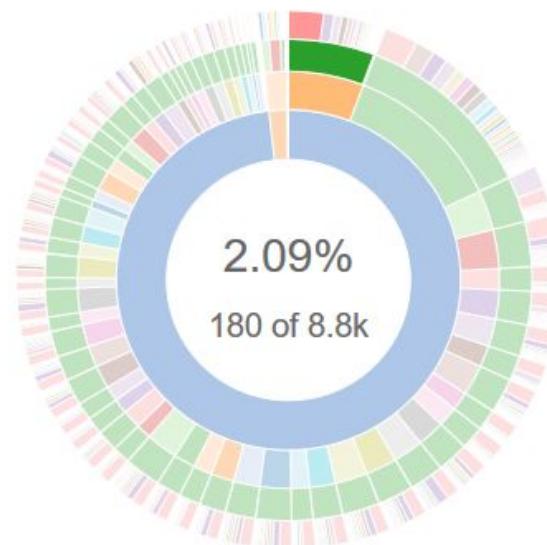
prep-script
dplyr



`sunburst(regD3, count = TRUE)`

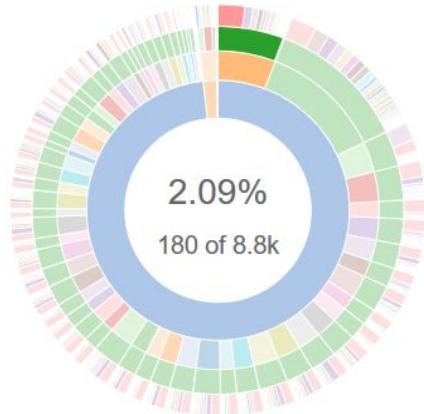
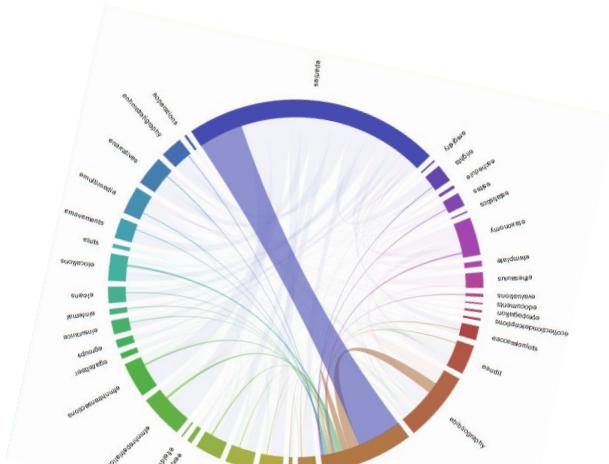


Group Admin Table ecatalogue 2.09%



Key1	Key2	Key3	Key4	Key5
Group	Default	Table	Default	Operations
Group	Default	Table	ecatalog	Tabs
Group	Admin	Table	eevents	Tabs
Group	IZMgr	Table	esites	Report
User	emu	Table	esites	Tabs

Light at the end of a tunnel...



...Or an oncoming train?

Cited Things:

- **networkD3**
- **sunburstR**
- github.com/fieldmuseum/Collections-Scripts
 - Shiny**SchemaVis**
 - Shiny**RegVis**
- **Darwin Core**
TDWG - <https://dwc.tdwg.org/>
- Science Map / **W. B. Paley**
- Children's book / **Homestar Runner**
- Screamy lamb / twitter.com/screaminglxmb
- Charlton Heston / Soylent Green

Acknowledgements:

FMNH
Technology Department

Sharon Grant
Janeen Jones
Pete Herbst
Rob Zschernitz

Action Department

Tomomi Suwa

