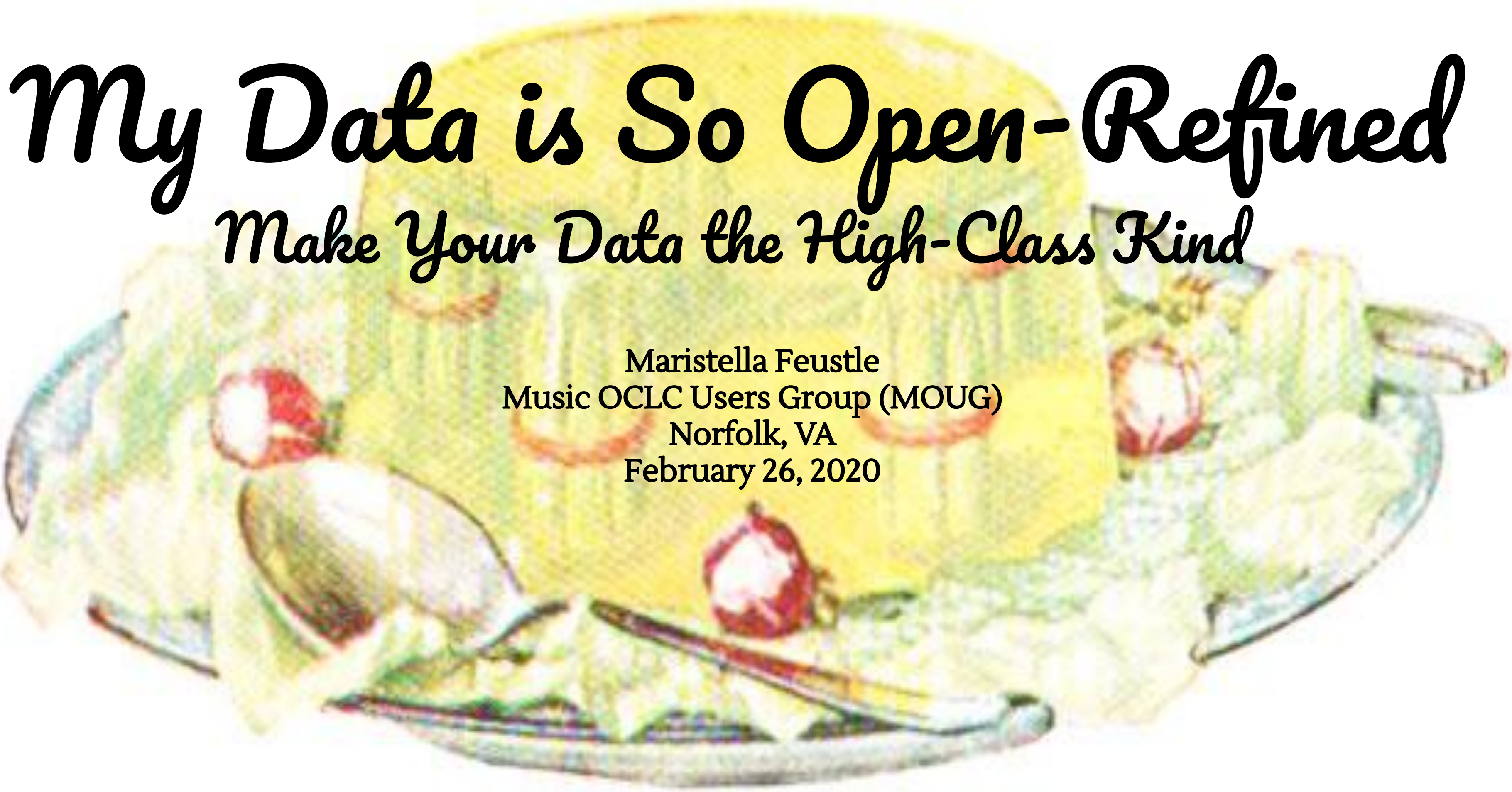
Today's data:

https://tinyurl.com/rphfun5



My Data is So Open-Refined

Make Your Data the High-Class Kind

Maristella Feustle
Music OCLC Users Group (MOUG)
Norfolk, VA
February 26, 2020

Why OpenRefine?

The ability to get useful information from machine-readable data requires that the data are being read as intended.

Use OpenRefine to:

- Clean up data - standardize, correct, rearrange
- Automate tedious editing
- Prep and export data for use in other programs, like MarcEdit
- Repurpose pre-existing data for a new task

Tasks I’ve taken on with OpenRefine

Converting Encoded Archival Description (EAD, an XML format) to tabular data

Turning raw lists into tables with columns

Converting inventories from collection donors into CSV files to import as finding aids.

Comparing two sets of data for matches

Getting an overview of what a set of data (like a collection inventory) is “about,” based on frequently occurring elements

Thoughts on learning new software tools

1. User interfaces are like a visual language: Visual cues with meanings prompt interaction.
2. Learning a new program requires requires repetition and continued engagement.
3. Mastery comes from getting lost and unlost, building resourcefulness and resilience.
4. No one was born knowing how to do this.

Basic transformations

Direct editing

Clearing stray whitespace

Major features in OpenRefine: faceting, clustering

Moving columns

Note: While OpenRefine runs in your browser, the data lives on your computer → security pros and cons

Then what?

How you proceed depends on:

1. The final form you want your data to have
2. How much intervention your data requires to get there

Before and after

When you let your mom cut your hair and she tells you what a handsome young man you are

Today’s main project

From Internet Archive’s Great 78 Project: search results for Pathé records

Why?

1. Loosely formatted, crowdsourced data is great for demonstration.
2. To match it against UNT’s internal holdings for overlap.

Data: https://tinyurl.com/rphfun5

GREL

General Refine Expression Language

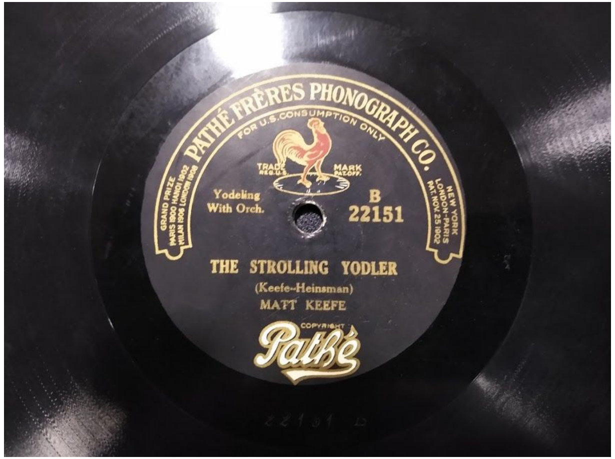
You don’t need to be fluent to make GREL do useful things.

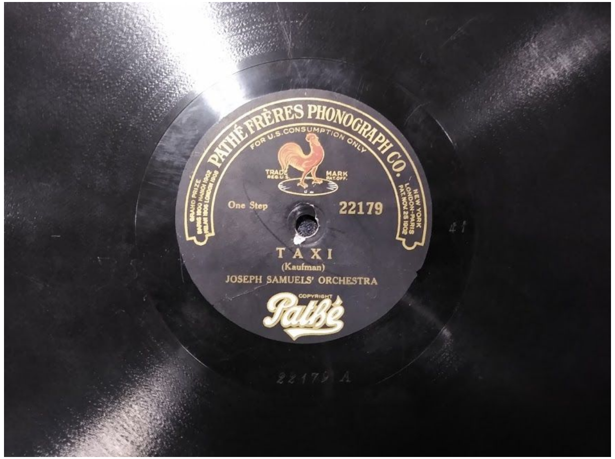
Just like learning a non-computer language, you start with some useful phrases.

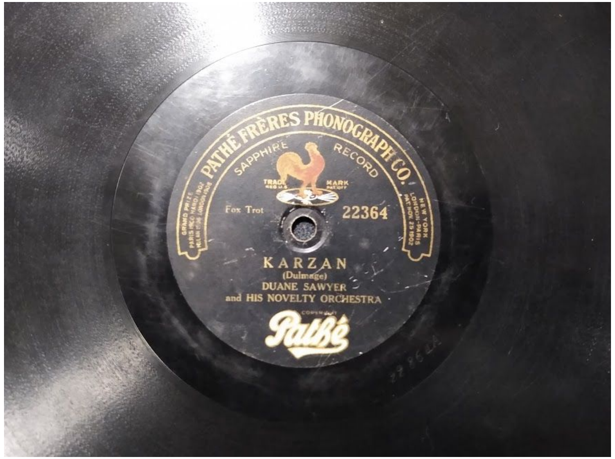
Just like there are language phrase books, there are GREL cheatsheets.

When in doubt, Google for the thing you’re trying to do, and find it or something close enough to adapt.

Ground truth




PATHÉ FRÈRES PHONOGRAPH CO.
FOR U.S. CONSUMPTION ONLY
One Step
22179
TAXI
(Kaufman)
JOSEPH SAMUELS' ORCHESTRA
Pathé
22179 A



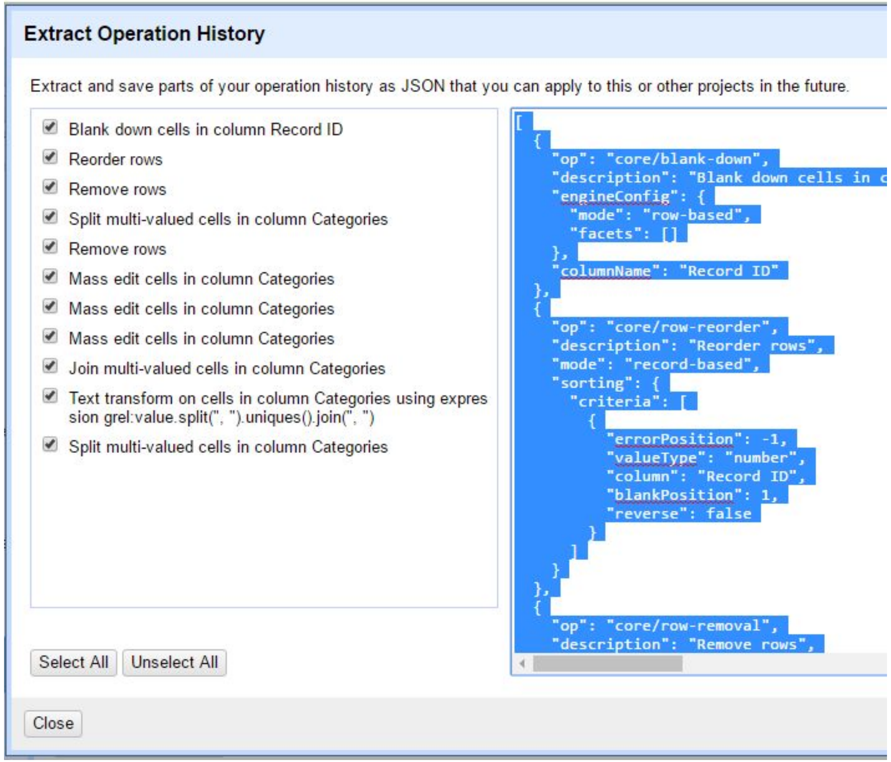
And another thing! XML imports

Useful for things like Bibframe and MARC XML.

LoC Comparison Tool data:
http://id.loc.gov/tools/bibframe/compare-id/full-rdf?find=5226

http://www.thinkgeek.com/product/6806/

Lather, rinse, repeat



Extract Operation History

Extract and save parts of your operation history as JSON that you can apply to this or other projects in the future.

- ☑ Blank down cells in column Record ID
- ☑ Reorder rows
- ☑ Remove rows
- ☑ Split multi-valued cells in column Categories
- ☑ Remove rows
- ☑ Mass edit cells in column Categories
- ☑ Mass edit cells in column Categories
- ☑ Mass edit cells in column Categories
- ☑ Join multi-valued cells in column Categories
- ☑ Text transform on cells in column Categories using expres sion grel:value.split(", ").uniques().join(", ")
- ☑ Split multi-valued cells in column Categories

```
[
  {
    "op": "core/blank-down",
    "description": "Blank down cells in c
    "engineConfig": {
      "mode": "row-based",
      "facets": []
    },
    "columnName": "Record ID"
  },
  {
    "op": "core/row-reorder",
    "description": "Reorder rows",
    "mode": "record-based",
    "sorting": {
      "criteria": [
        {
          "errorPosition": -1,
          "valueType": "number",
          "column": "Record ID",
          "blankPosition": 1,
          "reverse": false
        }
      ]
    }
  },
  {
    "op": "core/row-removal",
    "description": "Remove rows",
```

Extract command history to reuse on other datasets.

References and further reading

Comparing Two Sets of Data in OpenRefine: https://openlibraryenvironment.atlassian.net/wiki/spaces/GOKB/pages/655657/Comparing+Two+Sets+of+Data+in+OpenRefine

Terry Reese's MarcEdit videos: https://www.youtube.com/channel/UC7OLudoObYgiN_EmyDtZ_DQ

General Refine Expression Language (GREL): https://github.com/OpenRefine/OpenRefine/wiki/General-Refine-Expression-Language

GREL cheat sheet: https://code4libtoronto.github.io/2018-10-12-access/GoogleRefineCheatSheets.pdf

Thank you! (And here's a cat)

Maristella.Feustle@unt.edu

Twitter: @MFeustle

If you can read this, you don't need glasses.