Automating Date Parsing in ArchivesSpace

Alexander Duryee
ArchivesSpace Data Cleanup
2020-05-06
ArchivesSpace Implementation Goals

- Minimize repetitive data entry
- Minimize complexity of processing
- Maximize machine readability
Default Date Forms

- Encourages double data entry
- Can be complex to parse dates
- Is machine-readable, but only if Start/End are used
Existing Functionality

- timetwister
  - [https://github.com/alexduryee/timetwister](https://github.com/alexduryee/timetwister)
- Already in Ruby, already used to parse archival dates
Features of timetwister

- Parses most freetext dates into ISO-8601 EAD normal
  - May 6, 2020
  - April-May 2020
  - Early 2000s
  - Spring 2020
  - [May 6, 2020]
  - ca. May 6, 2020
timewalk

- ArchivesSpace implementation of the timetwister library
- [https://github.com/alexduryee/timewalk](https://github.com/alexduryee/timewalk)
- Transforms Expressions into Start/End/Certainty
- Removes extended Date fields for non-administrators
Benefits

● Data entry is much simpler
● Dates are consistently controlled across our instance
● SUI lives up to the ASpace slogan
Installation

- Download [https://github.com/alexduryee/timewalk](https://github.com/alexduryee/timewalk) to /plugins/timewalk
- Add `timewalk` to AppConfig[:plugins] list
- Restart ArchivesSpace
- Enjoy!
One Caveat

- timewalk will not automatically parse old dates!
- You’ll need to open and save old records for timewalk to parse them
  - Can be done manually or via the API
  - If you find any SUI tricks, please let me know!
Many Thanks!

- Staff across NYPL
- Contributors to timetwister/timewalk
- You!

alexanderduryee@nypl.org