CLSINFRA COMPUTATIONAL LITERARY STUDIES INFRASTRUCTURE

Dockerizing DraCor

A Container-based Approach to Reproducibility in Computational Literary Studies

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004984

CLSINFRA

This presentation:

bit.ly/dockerizing-dracor

CLSINFRA

Reproducibility as a Challenge for CLS

A reproducibility problem

"The humanities have a 'reproducibility' problem"

O’Sullivan 2019

A reproducibility problem

"The humanities have a 'reproducibility' problem"

O'Sullivan 2019

" …some serious and relevant challenges for the field of CLS: notably, the difficulty to reproduce work in this field, starting with issues of access to data and code, but also concerning questions of lacking reporting standards, limited scholarly recognition, and missing community commitment and capacity that would all be needed to foster a culture of [Repetitive Research] in [Computational Literary Studies] and beyond."

Schöch [Preprint] 2023: 8

CLS INFRA

- Community and research project “CLS INRA. Computational Literary Infrastructure”, funded by European Union’s Horizon 2020 research and innovation programme
- Work Package 7: “Building the Ecosystem of and for Programmable Corpora”,
  - Cf. the report Börner, Trilcke et al. (2023): “On Programmable Corpora”: “… corpora that expose an open, transparently documented and (at least partly) research-driven API to make texts machine-actionable.”

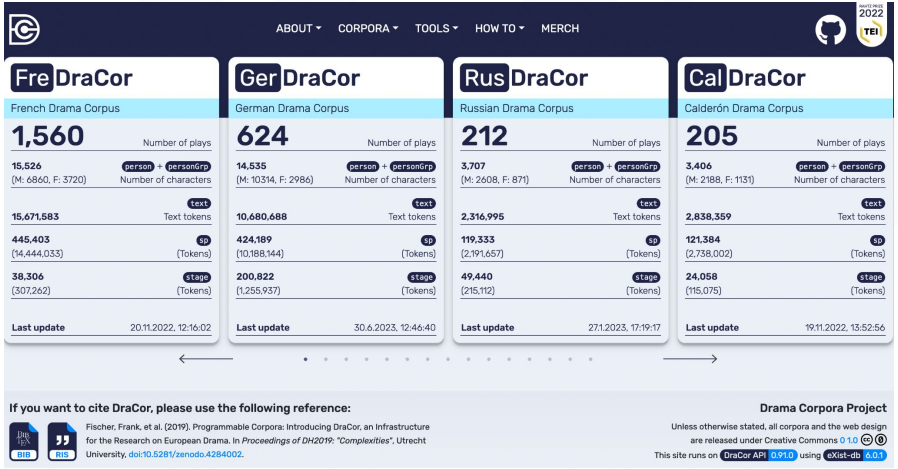
CLS INFRA

CLSINFRA

- Currently working on Task 7.3: “Technical stability of APIs (and Programmable Corpora) for versioning and reproducibility”
- Focus: replication (of research)
  - " … the term *replication* (of research) designates practices of [Repetitive Research] in which the research question, the dataset and the method of analysis of the repeating study are all identical to the original study." (Schöch [Preprint] 2023: 14)
  - "The requirements for enabling replication (of research) are quite high, as the complete dataset and all code need to be available in a form that allows running the code without modifications or re-implementations." (Schöch [Preprint] 2023: 15)

Our Approach: Dockerizing DraCor

- Inspired by the concept of *research artifacts* from computer science (cf. e.g. Arvan et al. 2022: 2351-2352)
- Based on DraCor (Drama Corpora Plattform)
- Self-set Challenge: Make the study Trilcke et al. [Preprint, 2022] “Detecting Small Worlds in a Corpus of Thousands of Theater Plays” replicable.



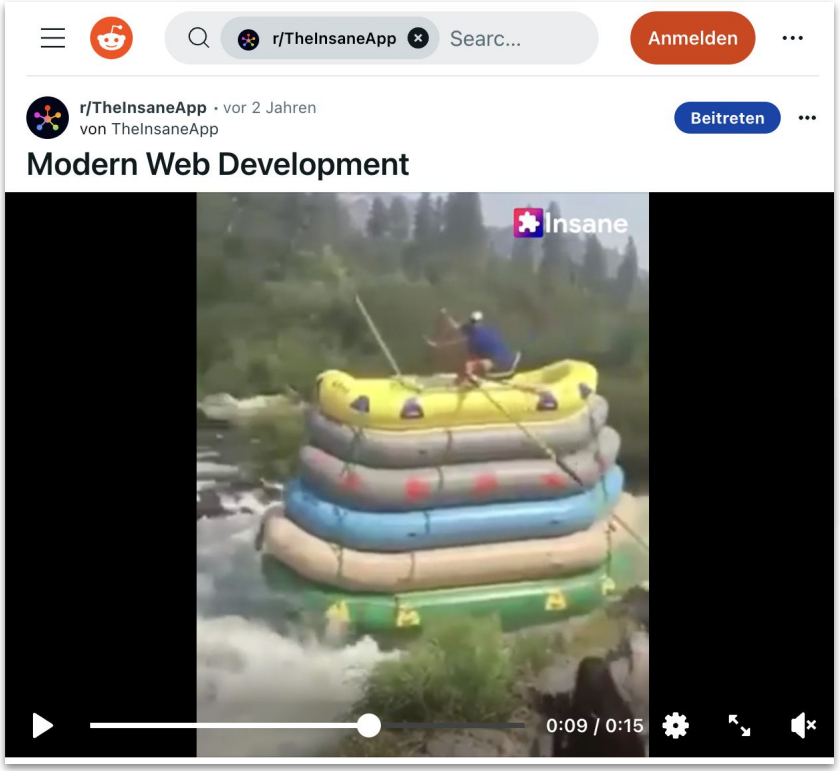
CLSINFRA

Using Docker to Enhance Reproducibility

Advantages of Docker in IT / “DevOps”

CLSINFRA

- streamline communication processes between Development (“Dev”) and Operations (“Ops”) Teams
- software dependencies are handled by developers – provide run-able apps



Key Components of Docker

CLSINFRA

- **Dockerfile**: Text file (YAML) defines steps how to create an image.
- Docker **Image**: contains instructions for creating a container; built on top of other images
- Docker **Container**: running instance of an image
- Docker **compose** file**:** defines system consisting of multiple services run as containers in a single YAML file

From *DevOps* to CLS Research

CLSINFRA

"You are free to use the code for your own purposes or to try to reproduce the results I put forth here. I make no claims to elegance in programming, but I am confident that **the scripts work, at least as of today**. Durability has taken on a new scale of meaning when seen against the long timescales of bibliographic preservation." (Piper. Enumerations, xii)

CLSINFRA

Use Case:

“Small World” study in containers

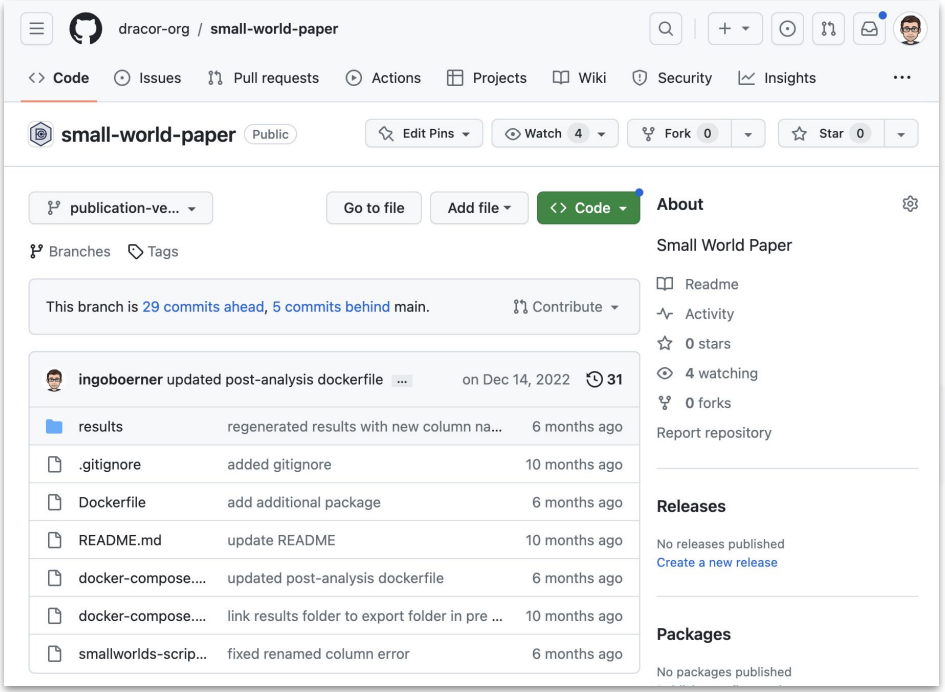
Study on “Detecting Small Worlds in a Corpus of Thousands of Theater Plays”

CLSINFRA

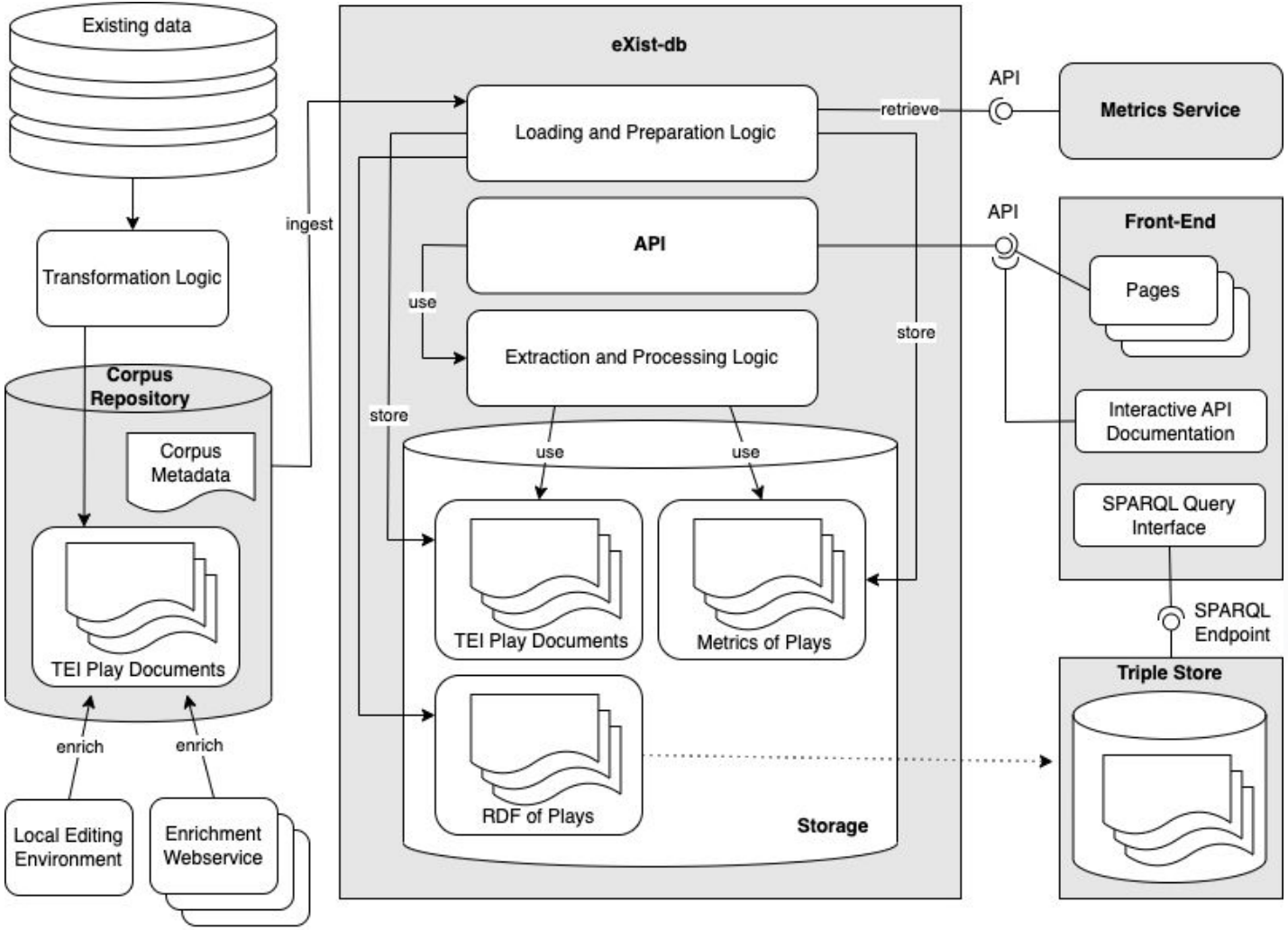
Study: [Trilcke et al. [Preprint, 2022] “Detecting Small Worlds in a Corpus of Thousands of Theater Plays”]

- based on a Very Big Drama Corpus (“**VeBiDraCor**”) of approx. 3.000 plays,
- analyzed with an R script fetching data from a locally running DraCor API

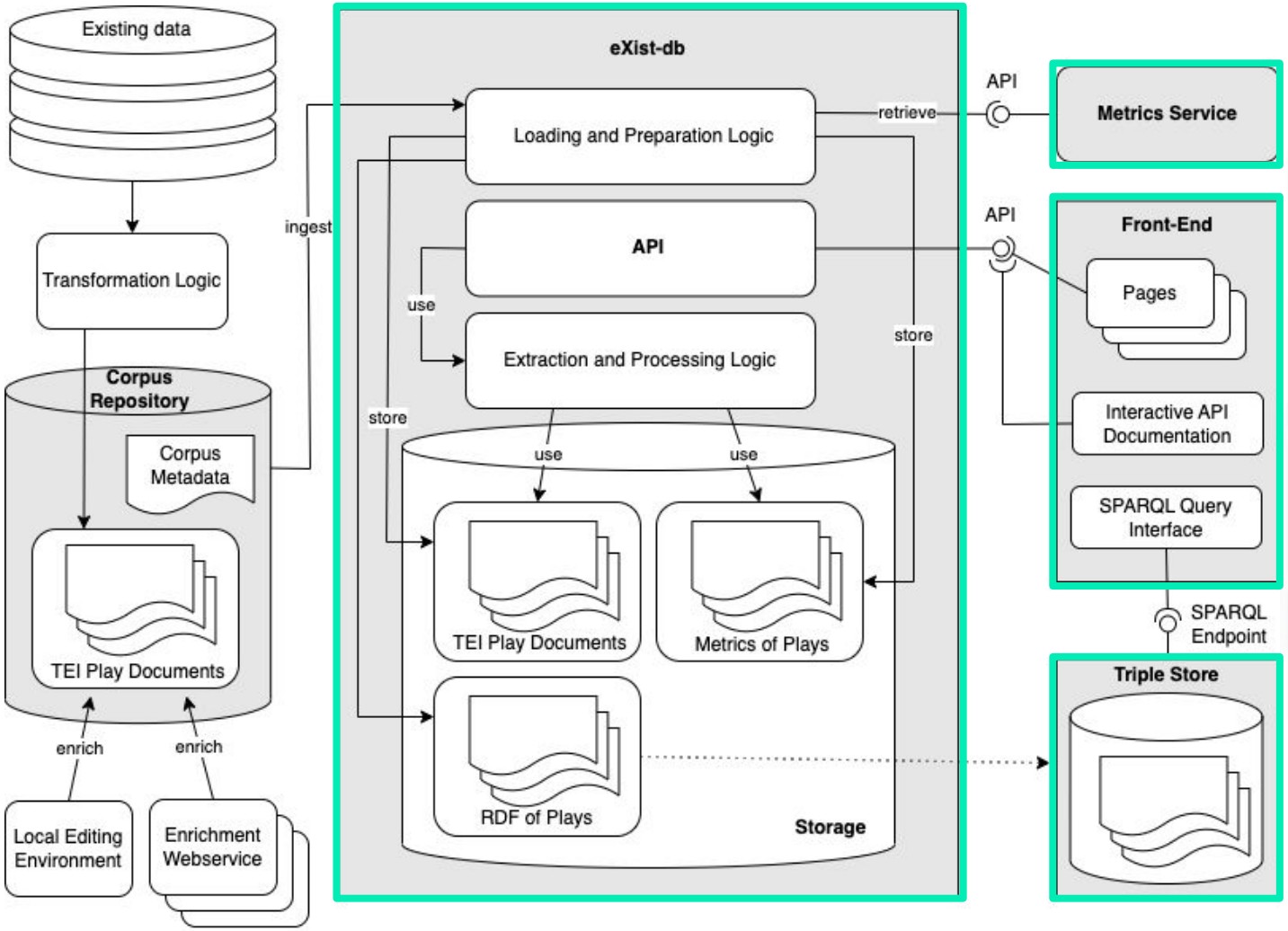
stabilize not only the ”data” – the ‘living’ DraCor corpora – but all necessary infrastructural components: DraCor API, Research Environment (RStudio)



GitHub Repository of the Study



Overview DraCor system components



CLSINFRA

dracor/dracor-api · ⬇ 10 · ☆ 0
By dracor · Updated 21 days ago
Image

dracor/dracor-frontend · ⬇ 11 · ☆
By dracor · Updated 21 days ago
Image

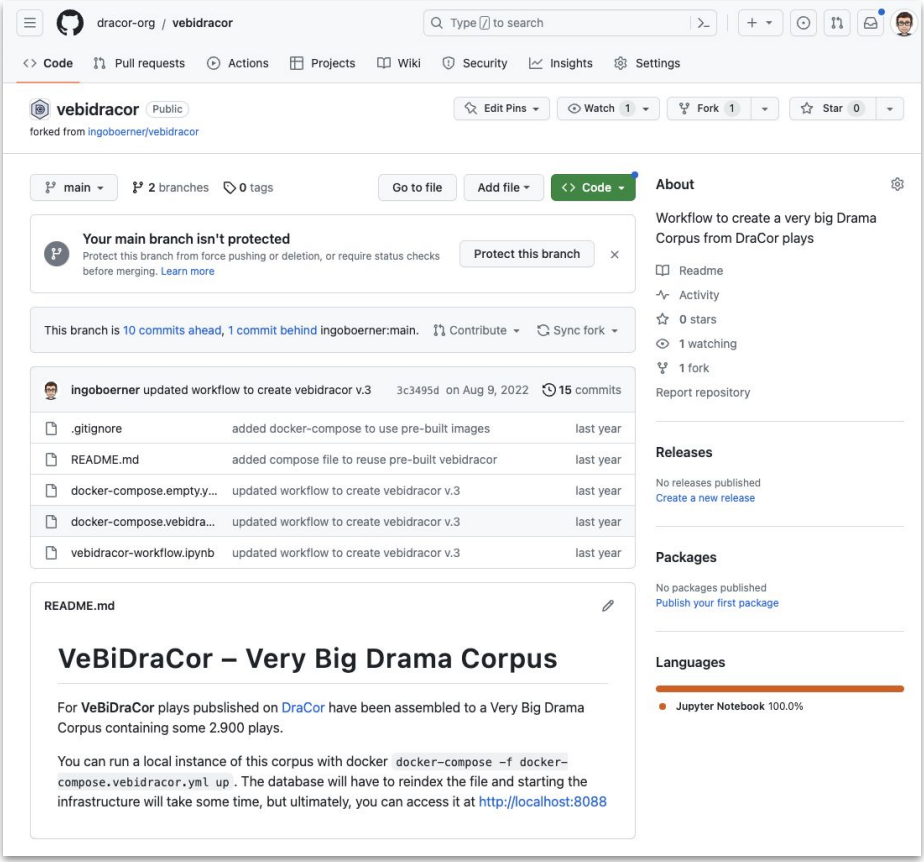
dracor/dracor-metrics · ⬇ 7 · ☆ 0
By dracor · Updated 21 days ago
Image

dracor/dracor-fuseki · ⬇ 7 · ☆ 0
By dracor · Updated 21 days ago
Image

Docker images of DraCor Core Services [DockerHub]

VeBiDraCor – A Very Big Drama Corpus

CLSINFRA




```
#list of corpora to include
corpora_to_include = [
    {
        "corpusname": "als",
        "repository": "https://github.com/dracor-org/alsdracor",
        "commit" : "c87ea41aac9412e4bd84a28e9c7632c53904f77c"
    },
    {
        "corpusname": "bash",
        "repository" : "https://github.com/dracor-org/bashdracor",
        "commit" : "c16b58ef3726a63c431bb9575b682c165c9c0cbd"
    },
    {
        "corpusname": "cal",
        "repository": "https://github.com/dracor-org/caldracor",
        "commit": "6cb804d415051d5f18bc4841fa1ce4343a7f0ab5"
    },
    {
        "corpusname": "fre",
        "repository": "https://github.com/dracor-org/fredracor",
        "commit": "65e93f6ff632b367cdc7e16e3e390956856c4b98",
        "exclude" : ["fre000038", "fre000057", "fre000065", "fre000099"]
    },
    {
        "corpusname": "ger",
        "repository": "https://github.com/dracor-org/gerdracor",
        "commit": "9135bd4598f54133f23df6edfc983b79f1616fb5",
        "exclude" : ["ger000480"]
    },
    {
        "corpusname": "greek",
        "repository": "https://github.com/dracor-org/greekdracor",
        "commit": "7397aafa1927c3e0a0720bf3c00bf367ab679f26"
    },
```

https://github.com/dracor-org/vebidracor

From Corpus-Container to Image: *docker commit*

CLSINFRA

In [30]:
```
!docker container ls -a | grep vebidracor
```

```
13ada6df6ea8   ingoboerner/dracor-frontend:v1.4.3_local                  "/docker-entrypoint.…"   22 hou
rs ago    Up 22 hours                 0.0.0.0:8088->80/tcp            vebidracor_frontend_1
fb0395a7e6d3   ingoboerner/dracor-api:v0.86.3_local                     "java org.exist.star…"   22 hou
rs ago    Exited (0) 2 minutes ago                                    vebidracor_api_1
5eed285d8f57   ingoboerner/dracor-metrics:v1.2.1                        "pipenv run hug -f m…"   22 hou
rs ago    Up 22 hours                 0.0.0.0:8030->8030/tcp          vebidracor_metrics_1
517897c784a1   stain/jena-fuseki                                        "/docker-entrypoint.…"   22 hou
rs ago    Up 22 hours                 0.0.0.0:3030->3030/tcp          vebidracor_fuseki_1
```

In [31]:
```
!docker commit -m "prepared VeBiDraCor based on pre-built images, loaded corpora, added metrics" $vebidracor_api_
```

```
sha256:24305e7b9509418e24b411cbef5e63702b25f95020b9ceb6d247d8cd1689dd45
```

In [32]:
```
#list
!docker images | grep vebidracor
```

```
ingoboerner/vebidracor-api                                3.0.0               24305e7b9509    9 seconds ago
2.26GB
```

Pushed the images to Dockerhub `docker push ingoboerner/vebidracor-api:3.0.0`. Had to create the repos first on Dockerhub, e.g. https://hub.docker.com/repository/docker/ingoboerner/vebidracor-api

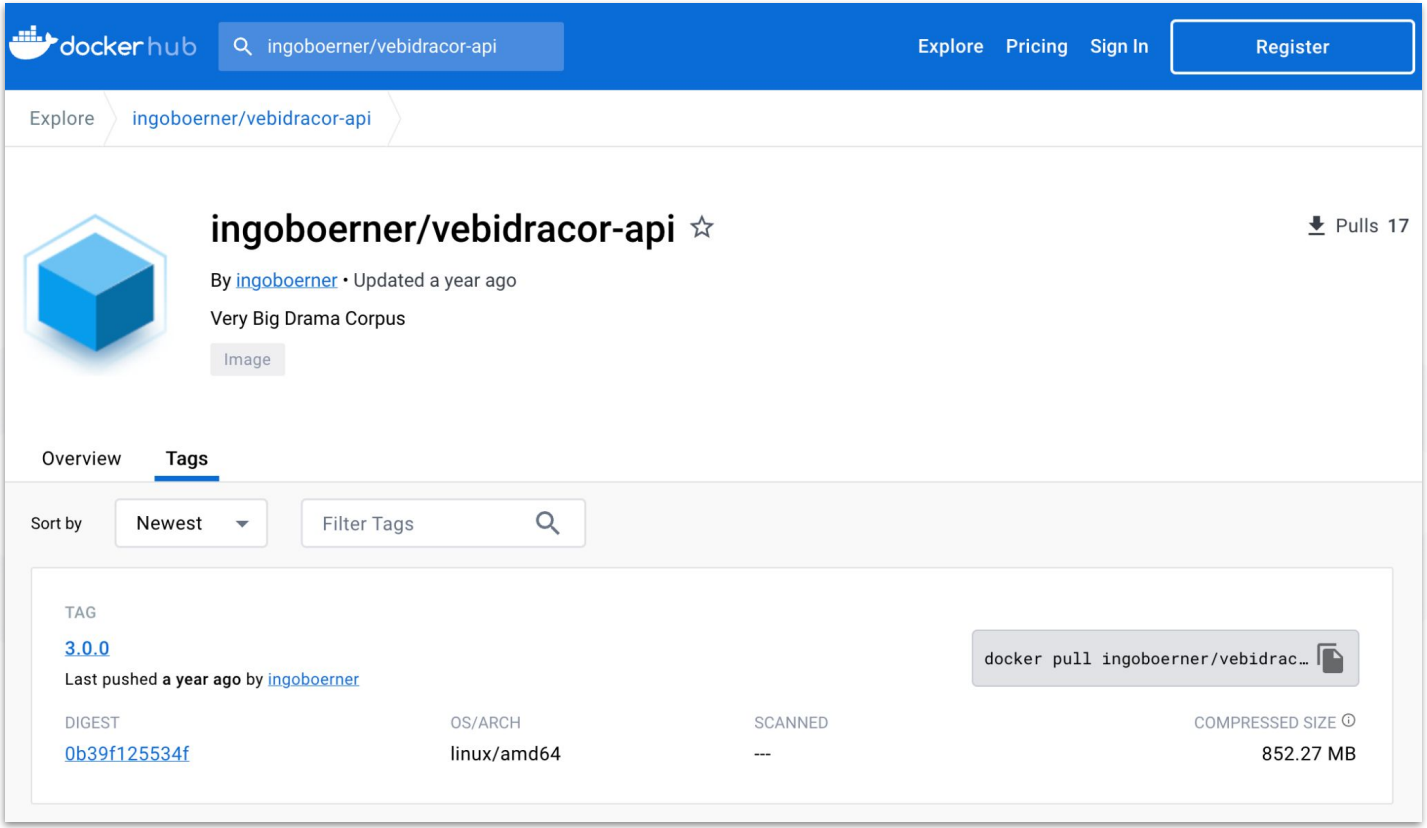
In [33]:
```
!docker push ingoboerner/vebidracor-api:3.0.0
```

```
The push refers to repository [docker.io/ingoboerner/vebidracor-api]

bb6a64ae: Preparing
7f4d745d: Preparing
abe1e1b1: Preparing
d1cd4f1d: Preparing
5d1e4382: Preparing
b2c72703: Preparing
```

VeBiDraCor Docker Image

CLSINFRA



Dockerizing Research Environment (RStudio Image)

CLSINFRA

small-world-paper / Dockerfile

ingoboerner add additional package — 5622479 · 7 months ago — History

Code | Blame | 14 lines (14 loc) · 545 Bytes | Raw

```
1   #need to specify version of dockerized R
2   FROM rocker/verse:4
3   # set the API base
4   ENV DRACOR_APIBASE=http://localhost:8088/api/
5   # set environment variable corpusname
6   ENV CORPUSNAME=vebi
7   #install the packages; should also later use designated versions!
8   RUN R -e "install.packages(c('purrr', 'igraph', 'stringr', 'jsonlite', 'dplyr'))"
9   #add the script to use in R Studio server instance
10  ADD smallworlds-script.R  /home/rstudio/smallworlds-script.R
11  #create an export folder for the results
12  RUN mkdir /home/rstudio/export
13  #expose the port
14  EXPOSE 8787
```

The Rocker Project
Docker Containers for the R Environment

vebidracor — Explore — Repositories — Organizations — Help — Upgrade — dracor

Explore / ingoboerner/smallworld-rstudio

ingoboerner/smallworld-rstudio ☆ — Pulls 9

By ingoboerner • Updated 7 months ago

Image

Docker Compose Files

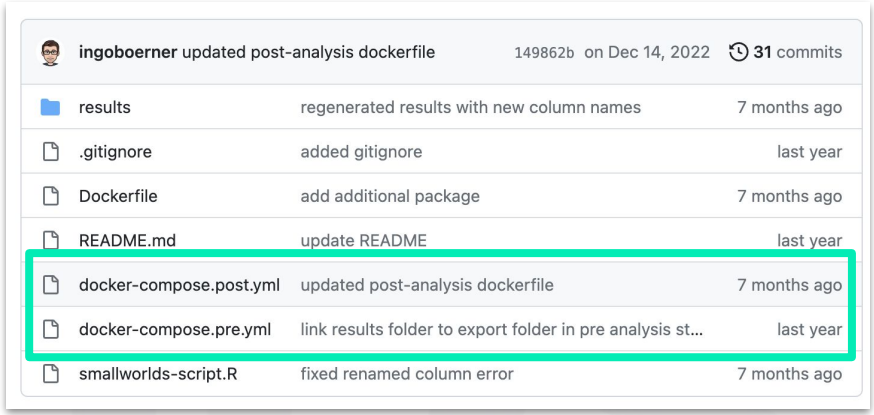
CLSINFRA

small-world-paper / docker-compose.pre.yml

ingoboerner link results folder to export folder in pre analysis state container bef7ead · last year History

Code Blame 27 lines (27 loc) · 572 Bytes Raw

```yaml
version: '2'
services:
  api:
    image: ingoboerner/vebidracor-api:3.0.0
    ports:
      - "8080:8080"
  frontend:
    image: ingoboerner/dracor-frontend:v1.4.3_local
    ports:
      - "8088:80"
    depends_on:
      - api
  rstudio:
    image: "dracor/smallworld-rstudio"
    build:
      context: .
      dockerfile: Dockerfile
    environment:
      - PASSWORD=smallworld
      - DRACOR_APIBASE=http://api:8080/exist/restxq/
      - DRACOR_CORPUSNAME=vebi
    volumes:
      - ./results:/home/rstudio/export
    ports:
      - "8787:8787"
    depends_on:
      - api
```



Docker compose files allow to recreate

- pre-analysis state
- post-analysis state

Single command to start infrastructure:

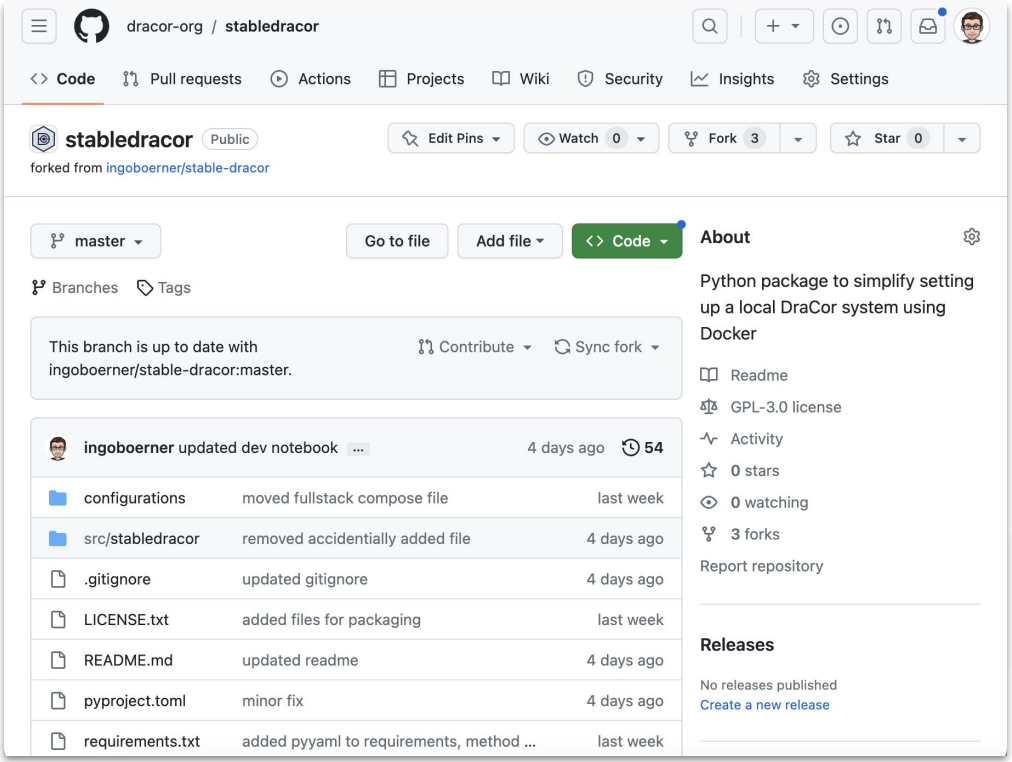
```
docker compose -f
docker-compose.post.yml up
```

CLSINFRA

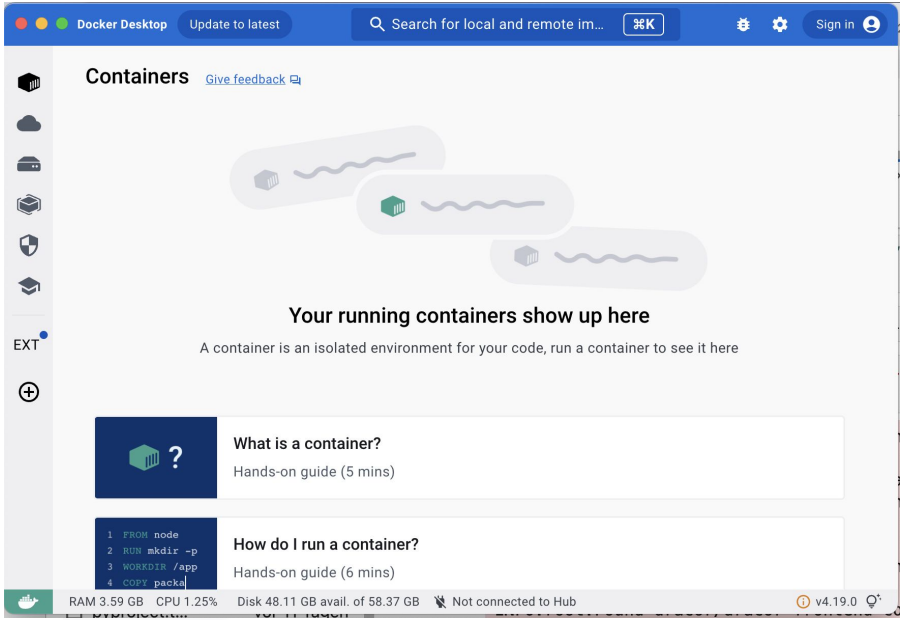
Outlook: StableDraCor Client

StableDraCor Client

CLSINFRA



https://github.com/dracor-org/stabledracor



CLSINFRA

Official **Docker Desktop** app to manage images and containers [Download]

Initialize a StableDraCor instance (Demo in Jupyter notebook)

```
[7]: local_api = StableDraCor(name="my-stable-dracor",
                              description="My custom DraCor")
```

```
INFO:root:Initialized new StableDraCor instance: 'my-stable-dracor' (ID: a93e0efb-0088-45fc-a641-bc16d6c50918).
WARNING:root:Local DraCor API is not available at http://localhost:8088/api/.
WARNING:root:Personal GitHub Access Token is not supplied. Requests to the GitHub API might be affected by rate limiting.
INFO:root:Docker is available.
WARNING:root:Could not detect a running Docker container derived from a dracor/dracor-api image.
WARNING:root:Could not detect a running Docker container derived from a dracor/dracor-frontend image.
WARNING:root:Could not detect a running Docker container derived from a dracor/dracor-metrics image.
WARNING:root:Could not detect a running Docker container derived from a dracor/dracor-fuseki image.
WARNING:root:Could not detect a running Docker container derived from a dracor/stable-dracor image.
```

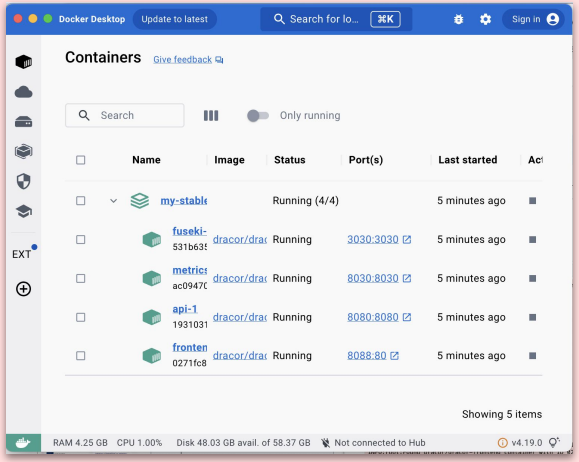
Running containers from within the client

CLSINFRA

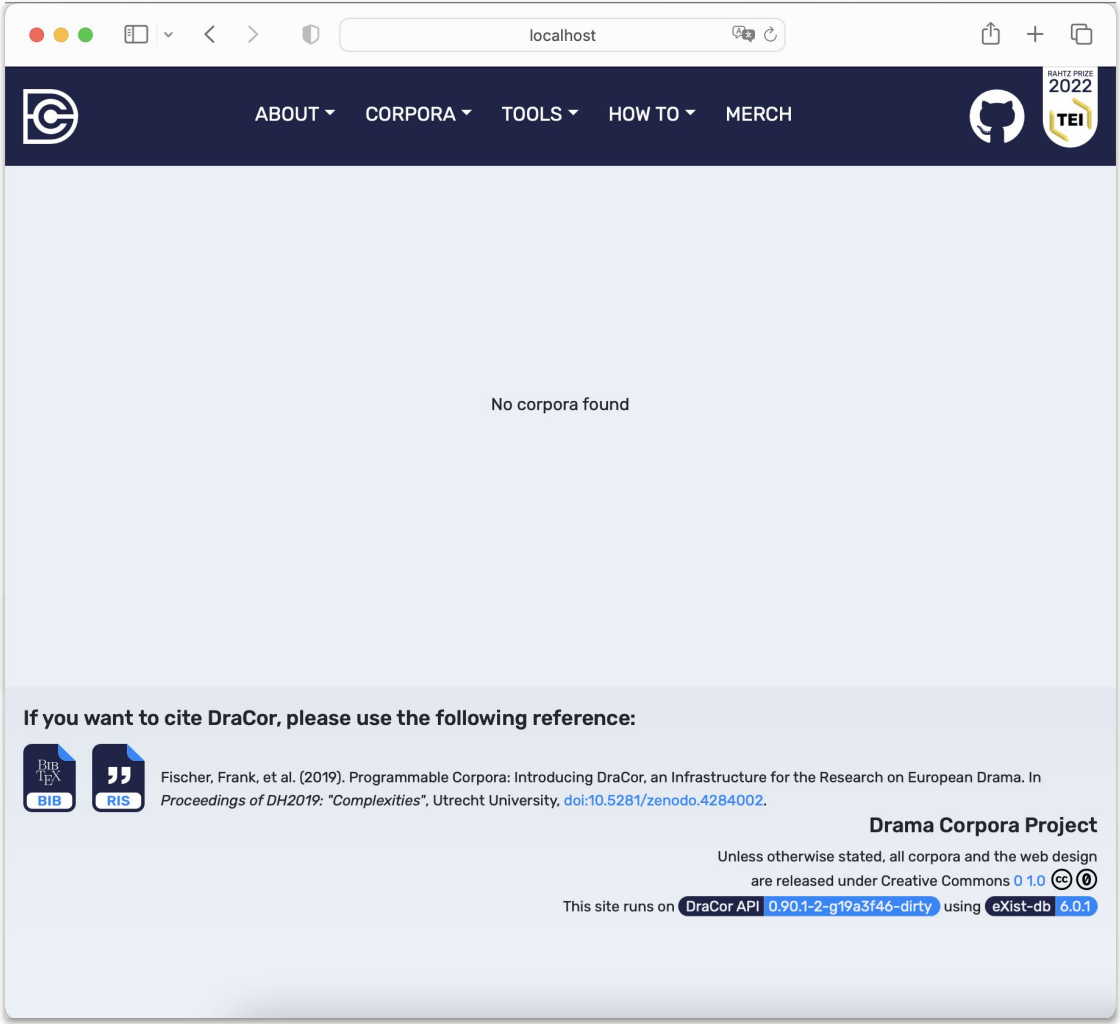
```
[8]: local_api.run()

INFO:root:Fetched default compose file (configuration) from https://raw.githubusercontent.com/dracor-org/stabledracor/master/configurations/compose.fullstack.empty.yml.
 Network my-stable-dracor_default  Creating
 Network my-stable-dracor_default  Created
 Container my-stable-dracor-fuseki-1  Creating
 Container my-stable-dracor-metrics-1  Creating
 Container my-stable-dracor-metrics-1  Created
 Container my-stable-dracor-fuseki-1  Created
 Container my-stable-dracor-api-1  Creating
 Container my-stable-dracor-api-1  Created
 Container my-stable-dracor-frontend-1  Creating
 Container my-stable-dracor-frontend-1  Created
 Container my-stable-dracor-metrics-1  Starting
 Container my-stable-dracor-fuseki-1  Starting
 Container my-stable-dracor-fuseki-1  Started
 Container my-stable-dracor-metrics-1  Started
 Container my-stable-dracor-api-1  Starting
 Container my-stable-dracor-api-1  Started
 Container my-stable-dracor-frontend-1  Starting
 Container my-stable-dracor-frontend-1  Started
INFO:root:Started with downloaded docker compose file.
INFO:root:Found dracor/dracor-api container with ID 1931031ecfda. Image is: dracor/dracor-api:v0.90.1-local
INFO:root:Found dracor/dracor-frontend container with ID 0271fc83ddc3. Image is: dracor/dracor-frontend:v1.6.0-dirty
INFO:root:Found dracor/dracor-metrics container with ID ac09470773c9. Image is: dracor/dracor-metrics:v1.2.0
INFO:root:Found dracor/dracor-fuseki container with ID 531b6351a8b9. Image is: dracor/dracor-fuseki:v1.0.0
INFO:root:Trying to connect to the local DraCor API. This can take some time ...
INFO:root:DraCor API can be reached at 'http://localhost:8088/api/'.

[8]: True
```



use one single command [`run()`] to fetch images
based on a default Docker compose file and start containers



CLSINFRA

DraCor frontend container and API are exposed at http://localhost:8088

no corpora are pre-loaded

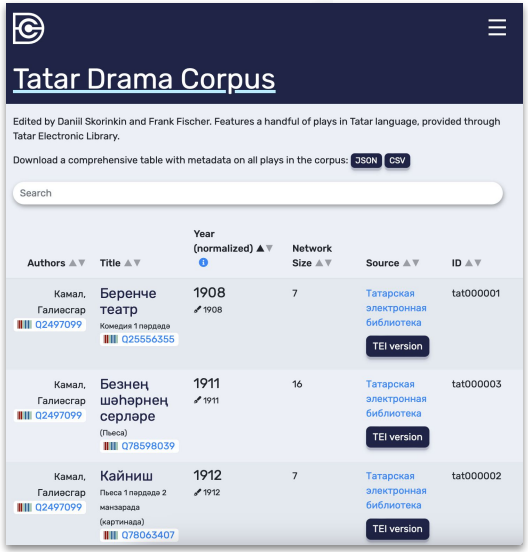
Copy Corpus from dracor.org

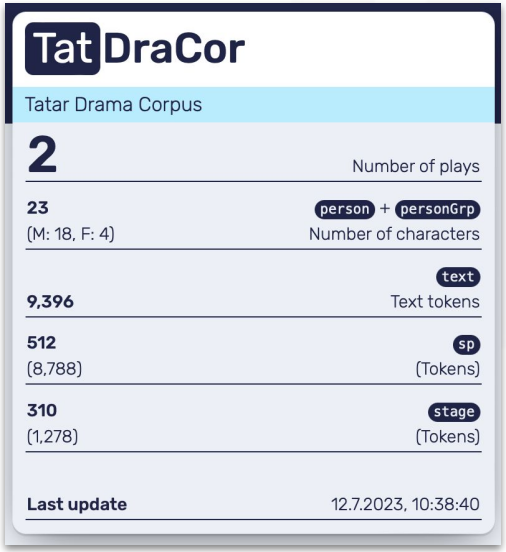
CLSINFRA

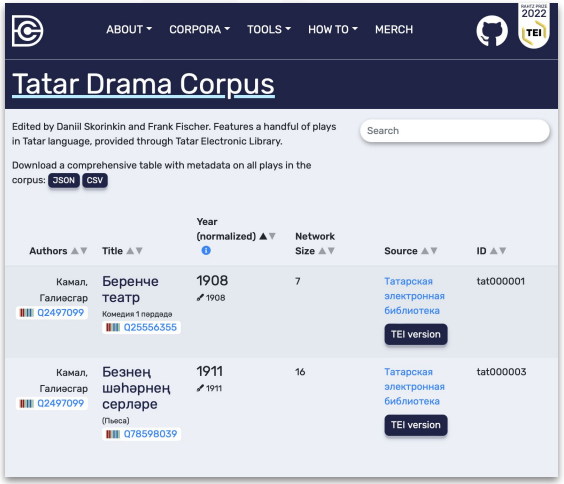
```
[13]: local_api.copy_corpus(source_corpusname="tat", exclude=["qamal-kaynish"])

INFO:root:Successfully created corpus tat. All metadata is available. Plays have not been added yet.
INFO:root:Added contents of corpus tat from https://dracor.org/api/. 2 plays were added.
INFO:root:Copying tat (as tat) was successful. Plays (that were not excluded) were also copied entirely.

[13]: True
```





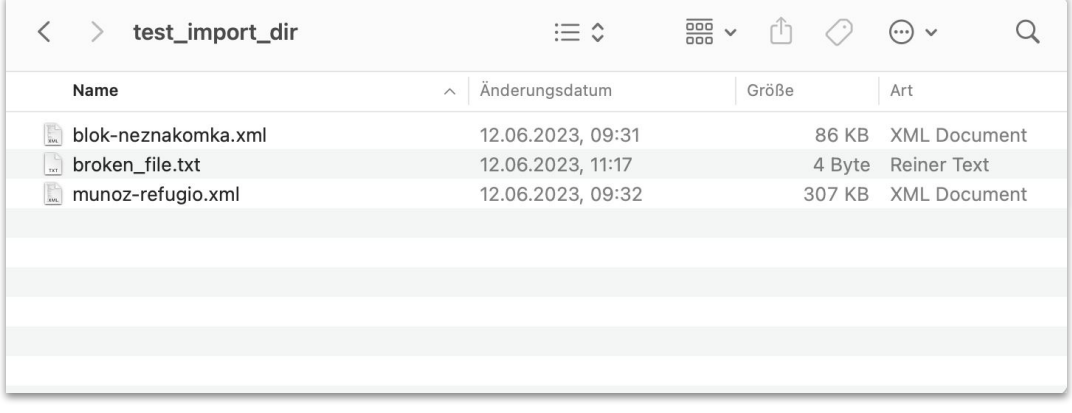


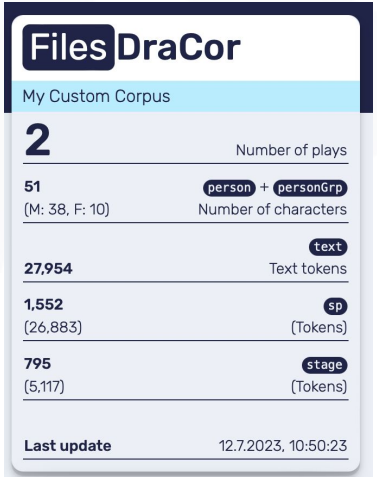
Create a corpus from a local folder

CLSINFRA

```
[15]: imported_corpus_metadata = {
          "name" : "files",
          "title": "My Custom Corpus",
          "description": "This is a custom corpus that was generated with the 'StableDraCor' client. It contains files from a local directory."
      }
      local_api.add_plays_from_directory(corpusname="files",directory="test_import_dir", corpus_metadata=imported_corpus_metadata)

      INFO:root:Successfully created corpus files. All metadata is available and plays are available.
      INFO:root:Added TEI data from file 'blok-neznakomka.xml' to corpus 'files'.
      INFO:root:Added TEI data from file 'munoz-refugio.xml' to corpus 'files'.
      INFO:root:Imported 2 files from test_import_dir as corpus 'files'.
[15]: True
```





Custom corpus
(from several sources – like *VeBiDraCor*)

CLSINFRA

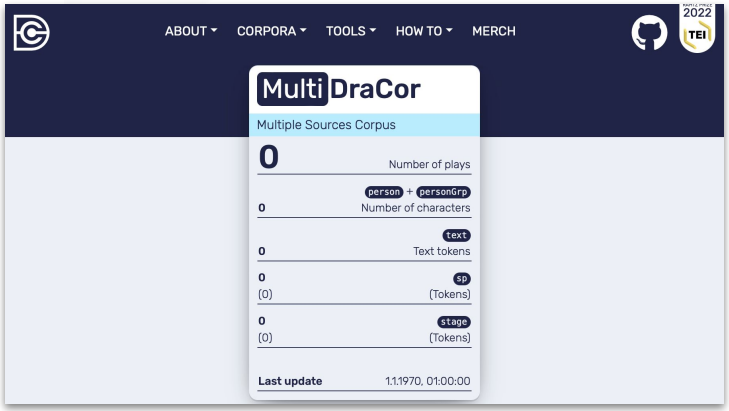
Corpus from several sources

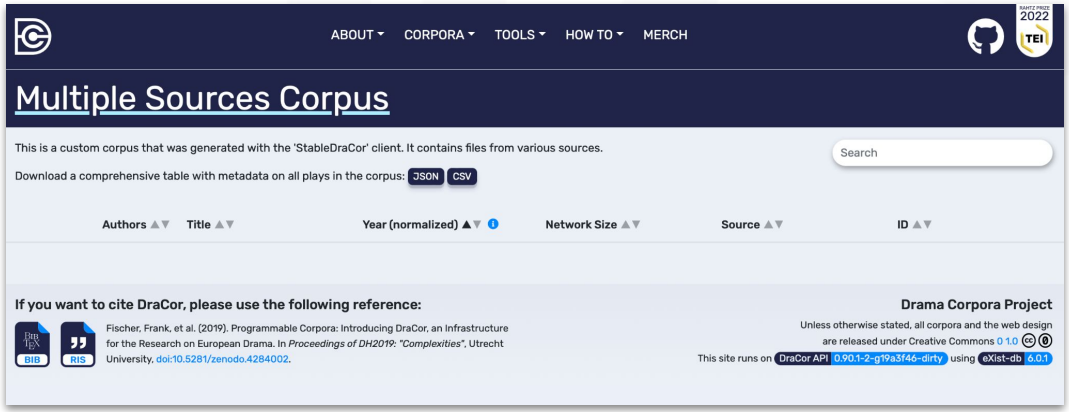
[9]:
```
multi_source_corpus_metadata = {
    "name" : "multi",
    "title": "Multiple Sources Corpus",
    "description": "This is a custom corpus that was generated with the 'StableDraCor' client. It contains files from various sources."
}
```

[10]:
```
# need to explicitly register this corpus
local_api.add_corpus(corpus_metadata=multi_source_corpus_metadata, register=True)
```

```
INFO:root:Successfully created corpus multi. All metadata is available and plays are available.
```

[10]:
```
True
```



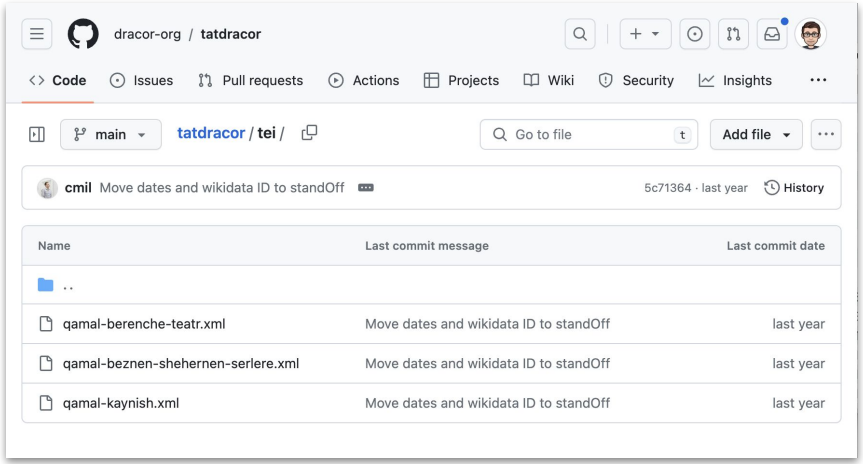


Adding plays from a Github Repository

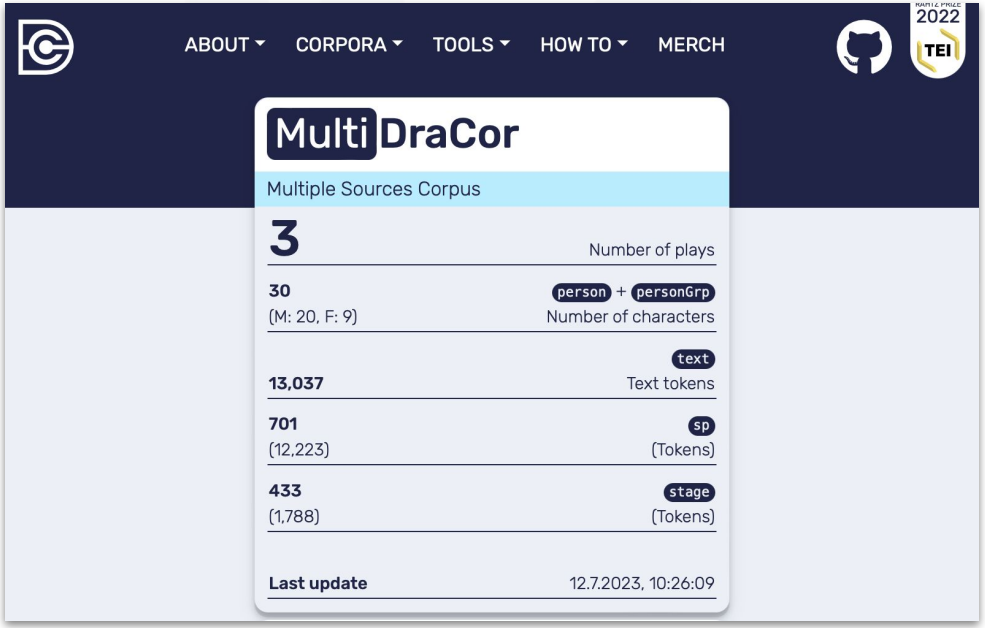
```
[11]: local_api.add_files_from_repo(corpusname="multi", repository_name="tatdracor")

INFO:root:Successfully added all 3 files of repository 'dracor-org/tatdracor' to corpus 'multi'.

[11]: True
```



Add the plays from [Tatar Drama Corpus](#)
to the “multi” corpus in the local instance



```
[14]: local_api.add_files_from_repo(corpusname="multi",
                                    repository_name="bashdracor",
                                    exclude=["khudayberdin-aq-bilettar", "karim-tashlama-utty"])
```

```
INFO:root:Successfully added all 1 files of repository 'dracor-org/bashdracor' to corpus 'multi'.
```

```
[14]: True
```

MultiDraCor

Multiple Sources Corpus

4 Number of plays

47 (M: 31, F: 14) `person` + `personGrp` Number of characters

25,851 `text` Text tokens

1,557 (24,323) `sp` (Tokens)

722 (2,960) `stage` (Tokens)

Last update 12.7.2023, 10:47:01

Multiple Sources Corpus

This is a custom corpus that was generated with the 'StableDraCor' client. It contains files from various sources.

Download a comprehensive table with metadata on all plays in the corpus: JSON CSV

Search

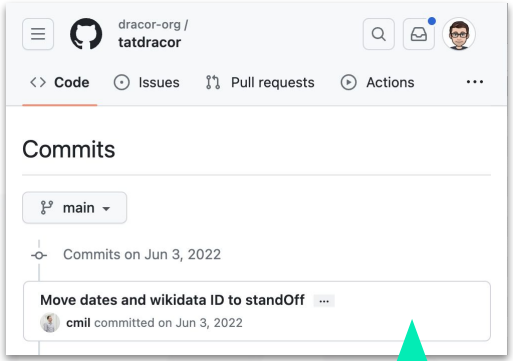
Authors	Title	Year (normalized)	Network Size	Source	ID
Камал, Галиәсгар Q2497099	Беренче театр Комедия 1 пәрдәдә Q25556355	1908 1908	7	Татарская электронная библиотека TEI version	tat000001
Камал, Галиәсгар Q2497099	Безнең шәһәрнең серләре (Пьеса) Q78598039	1911 1911	16	Татарская электронная библиотека TEI version	tat000003
Камал, Галиәсгар Q2497099	Кайниш Пьеса 1 пәрдәдә 2 манзарада (картинада) Q78063407	1912 1912	7	Татарская электронная библиотека TEI version	tat000002
Кәрим, Мостай Q2719262	Ай тотолған төндә Өс шаршаулы трагедия Q18401539	1964 1963 1964 1964	17	kitaptar.bashkort.org TEI version	bash00001

Manifest documenting system and corpora loaded

CLSINFRA

```
[12]: local_api.get_manifest()

[12]: {'version': 'v1',
 'system': {'id': 'a93e0efb-0088-45fc-a641-bc16d6c50918',
  'name': 'my-stable-dracor',
  'description': 'My custom DraCor',
  'timestamp': '2023-07-12T10:31:23.334919'},
 'services': {'api': {'container': '1931031ecfda',
   'image': 'dracor/dracor-api:v0.90.1-local',
   'version': '0.90.1-2-g19a3f46-dirty',
   'existdb': '6.0.1'},
  'frontend': {'co
   'image': 'draco
  'metrics': {'con
   'image': 'draco
  'triplestore': {
   'image': 'draco
 'corpora': {'mult
   'timestamp': '2
   'sources': {'ta
     'commit': '5c
     'url': 'https
     'timestamp':
     'num_of_plays': 3}},
   'num_of_plays': 3}}}
```



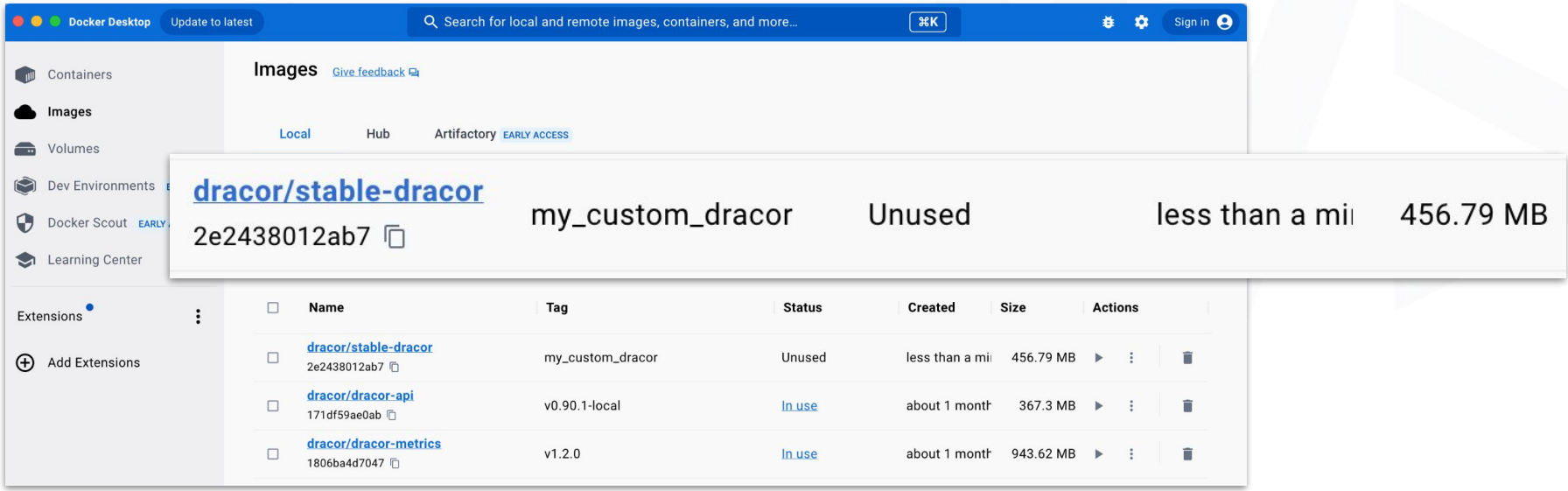

```
 'corpora': {'multi': {'corpusname': 'multi',
   'timestamp': '2023-07-12T10:21:56.346942',
   'sources': {'tatdracor': {'type': 'repository',
     'commit': '5c71364f39f6533baa3a2e04217fd39e0898c851',
     'url': 'https://github.com/dracor-org/tatdracor',
     'timestamp': '2023-07-12T10:26:04.848341',
     'num_of_plays': 3}},
```

Create image

CLSINFRA

```
[16]: local_api.create_docker_image_of_service(service="api", image_tag="my_custom_dracor")
```

```
WARNING:root:The dracor-api container is running. There might be issues with the image, if it is create f
eating the image.
INFO:root:Committed container 1931031ecfda as dracor/stable-dracor:my_custom_dracor. Image identifier sha
7893a3ae15f0d
.
```



```
[17]: local_api.list_docker_images()
```

```
[17]: [{'Containers': 'N/A',
  'CreatedAt': '2023-07-12 10:55:54 +0200 CEST',
  'CreatedSince': '4 minutes ago',
  'Digest': '<none>',
  'ID': '2e2438012ab7',
  'Repository': 'dracor/stable-dracor',
  'SharedSize': 'N/A',
  'Size': '457MB',
```

CLSINFRA

Extract Docker Object Labels from Docker Image

Manifest data converted to labels in *org.dracor.stable-dracor.** namespace

```
[18]: local_api.get_labels_from_docker_image("2e2438012ab7")
```

```
[18]: {'com.docker.compose.config-hash': 'f3e7dbfe4bebdf9f284e9ee661814e9b4623d214ea7d49854b4a651aa8e98c7d',
 'com.docker.compose.container-number': '1',
 'com.docker.compose.depends_on': 'fuseki:service_started:false,metrics:service_started:false',
 'com.docker.compose.image': 'sha256:171df59ae0ab650356c45feeabe2a65b63b77c3e9bd7bf362926bfdd78e931f8',
 'com.docker.compose.oneoff': 'False',
 'com.docker.compose.project': 'my-stable-dracor',
 'com.docker.compose.project.config_files': '-',
 'com.docker.compose.project.working_dir': '/Users/ingoboerner/Projekte/dracor/stable-dracor',
 'com.docker.compose.service': 'api',
 'com.docker.compose.version': '2.17.3',
 'org.dracor.stable-dracor.corpora': 'multi,tat,files',
 'org.dracor.stable-dracor.corpora.files.corpusname': 'files',
 'org.dracor.stable-dracor.corpora.files.num-of-plays': '2',
 'org.dracor.stable-dracor.corpora.files.sources': '9edc7a49',
 'org.dracor.stable-dracor.corpora.files.sources.9edc7a49.timestamp': '2023-07-12T10:50:20.352739',
 'org.dracor.stable-dracor.corpora.files.sources.9edc7a49.type': 'files',
 'org.dracor.stable-dracor.corpora.files.timestamp': '2023-07-12T10:50:20.352730',
```

Manifest <> Image Labels

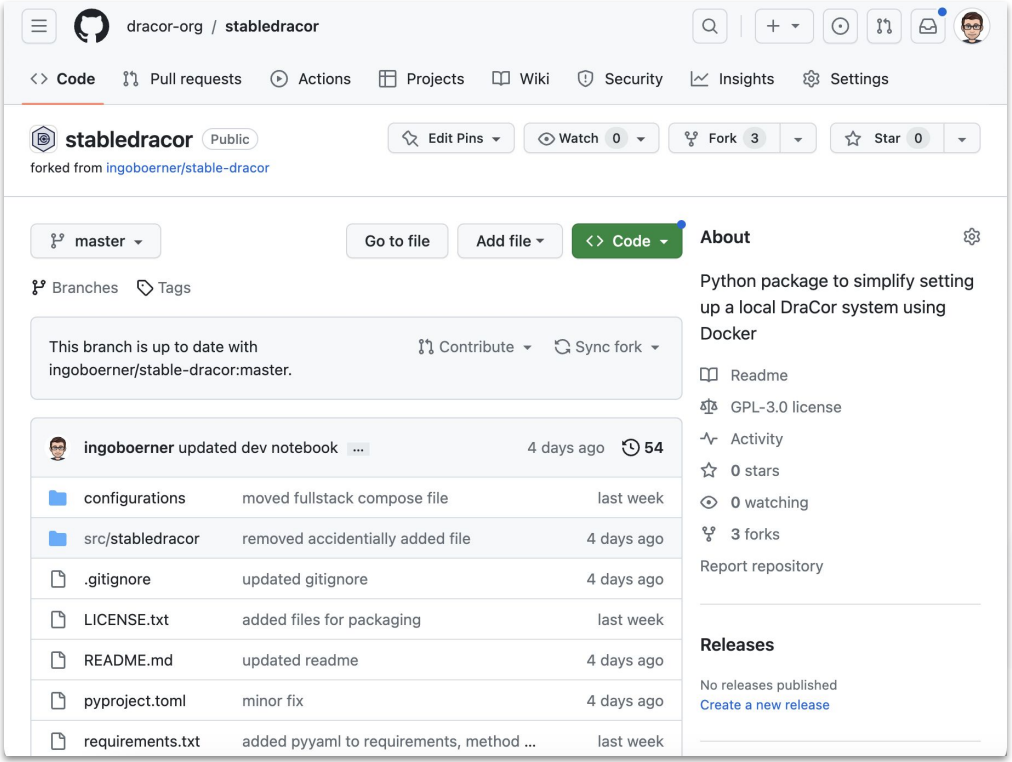
CLSINFRA

```
'tat': {'corpusname': 'tat',
 'timestamp': '2023-07-12T10:38:37.104030',
 'sources': {'tat': {'type': 'api',
   'corpusname': 'tat',
   'url': 'https://dracor.org/api/corpora/tat',
   'timestamp': '2023-07-12T10:38:37.104047',
   'exclude': {'type': 'slug', 'ids': ['qamal-kaynish']},
   'num_of_plays': 2}},
 'num_of_plays': 2},
```

```
'org.dracor.stable-dracor.corpora.tat.corpusname': 'tat',
'org.dracor.stable-dracor.corpora.tat.num-of-plays': '2',
'org.dracor.stable-dracor.corpora.tat.sources': 'tat',
'org.dracor.stable-dracor.corpora.tat.sources.tat.corpusname': 'tat',
'org.dracor.stable-dracor.corpora.tat.sources.tat.exclude.ids': 'qamal-kaynish',
'org.dracor.stable-dracor.corpora.tat.sources.tat.exclude.type': 'slug',
'org.dracor.stable-dracor.corpora.tat.sources.tat.num-of-plays': '2',
'org.dracor.stable-dracor.corpora.tat.sources.tat.timestamp': '2023-07-12T10:38:37.104047',
'org.dracor.stable-dracor.corpora.tat.sources.tat.type': 'api',
'org.dracor.stable-dracor.corpora.tat.sources.tat.url': 'https://dracor.org/api/corpora/tat',
'org.dracor.stable-dracor.corpora.tat.timestamp': '2023-07-12T10:38:37.104030',
```

StableDraCor Client

CLSINFRA



https://github.com/dracor-org/stabledracor

CLSINFRA

THANK YOU

ingo.boerner@uni-potsdam.de
trilcke@uni-potsdam.de