Az ARP platform adathozzáférési és -megosztási modelljei

Pallinger Péter, SZTAKI DSD

pallinger.peter@sztaki.hu

2023. 04. 13.

ELKH Eötvös Loránd Research Network

ELKH-ARP projekt

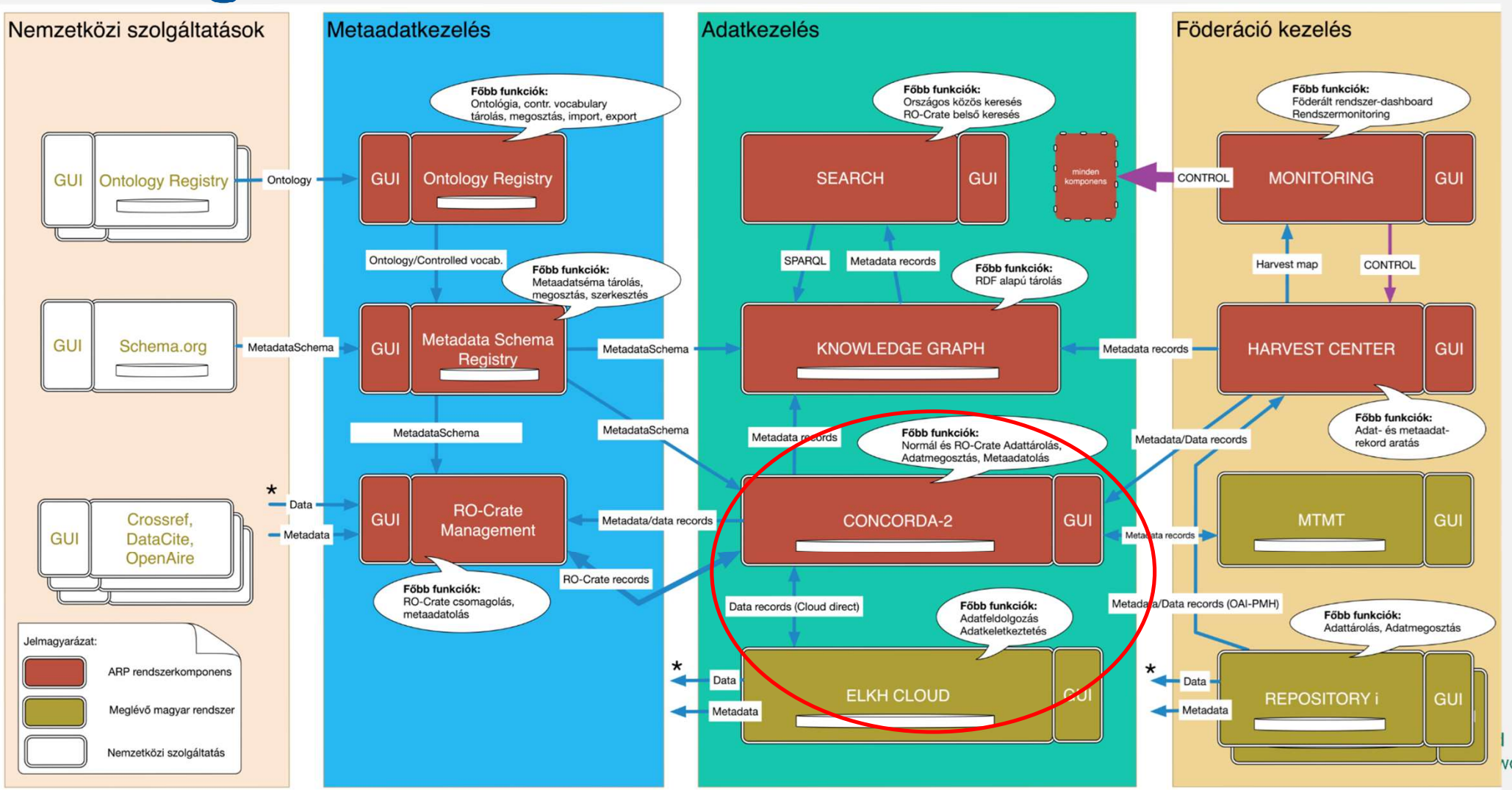
- Cél
  - A teljes ELKH kutatóhálózat számára folyamatos és hosszú távú kutatási repozitóriumi infrastruktúra-szolgáltatás megvalósítása
- Résztvevők
  - Számítástechnikai és Automatizálási Kutatóintézet (SZTAKI)
  - Társadalomtudományi Kutatóközpont (TK)
  - Wigner Fizikai Kutatóközpont (Wigner FK)
- Időtartam
  - 2021.11.01. - 2023.12.31.

A CONCORDA adatrepozitórium története

- 2020-ban jött létre
  - SZTAKI és WIGNER intézetek összefogásában
  - Régi MTA/ELKH felhőben
  - mind a SZTAKI, mind a WIGNER telephelyen
  - Valmint a DSD felhőjében
  - Harvard dataverse 4.20 alapokon
  - condorda.sztaki.hu, concorda.hu címeken
- 2021 végén az ARP projekt karolta fel
  - Adattárolás fő elemévé választottuk
  - science-data.hu címre költözött
  - Háttértár kiegészítését fogjuk elvégezni
  - Dataverse verziófrissítések, jelenleg 5.10 az éles
  - Integráció az ARP által fejlesztett szolgáltatásokkal

ARP SZTAKI ELKH Eötvös Loránd Research Network

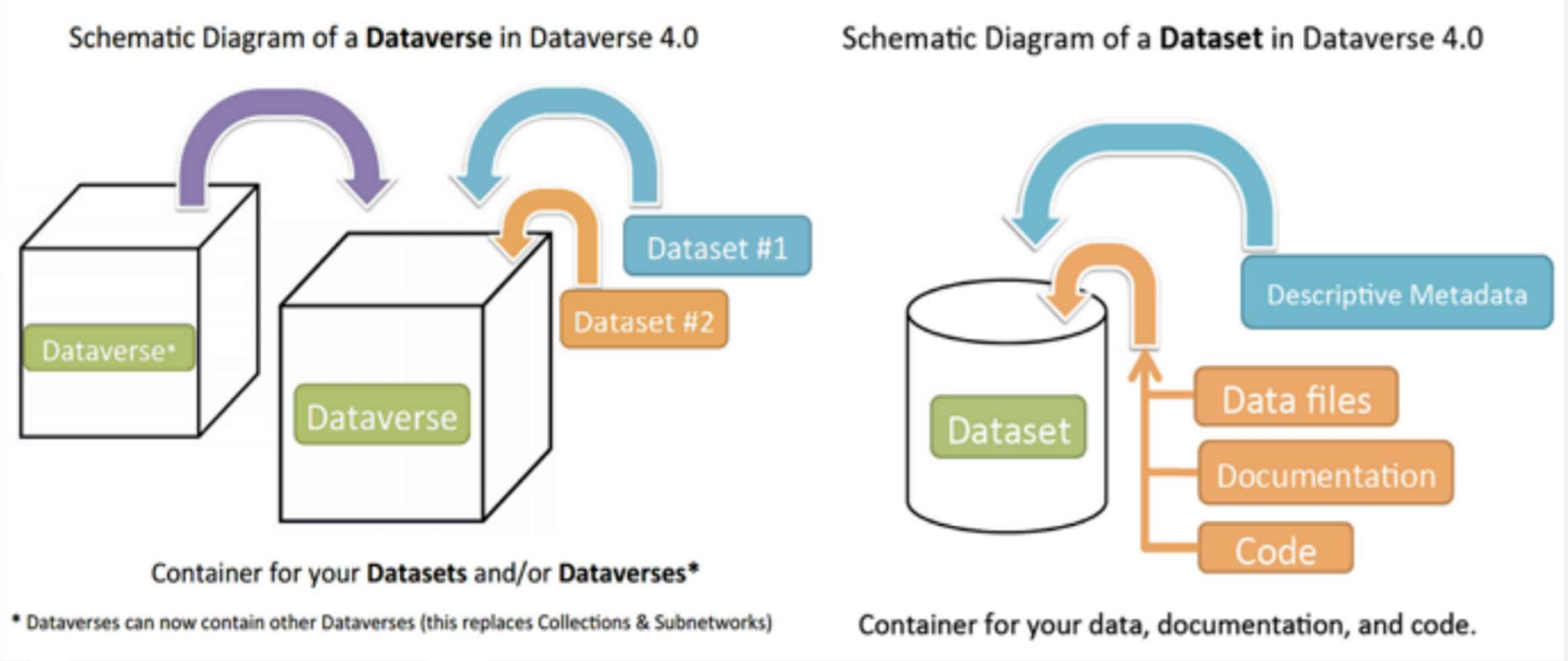
ARP logikai architektúra



Concorda - Dataverse

- A Dataverse szoftver:
  - Harvard által fejlesztett adatrepozitórium kifejezetten kutatási adatok megosztására
- Adatmodell
  - Tárolók (Dataverse)
    - Tárolókban további tárolók vagy adatcsomagok lehetnek
    - Elsősorban hozzáférés-szabályozás szempontjából van jelentőségük
  - Adatcsomagok (Dataset)
    - Részletes és gazdag metaadatok az adatcsomagokban
    - Adatcsomagok csak fájlokat tartalmazhatnak

A Dataverse adatmodellje

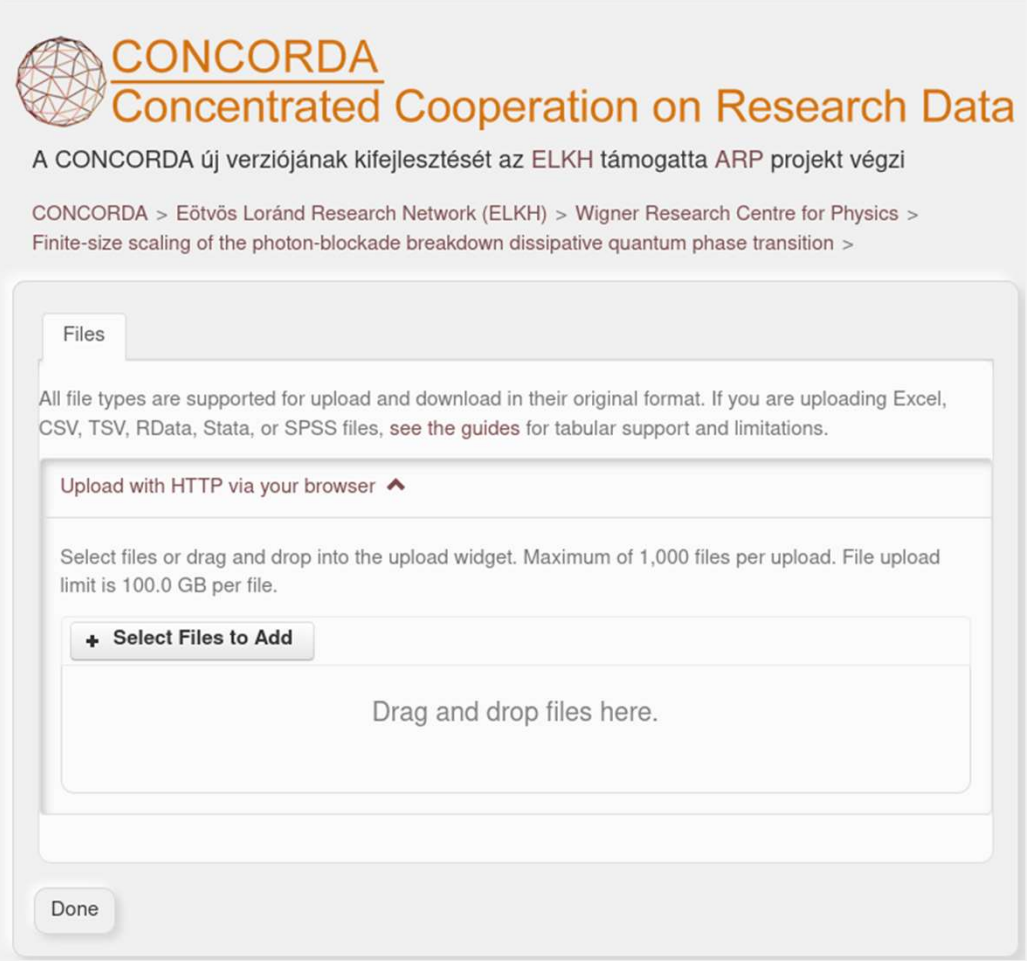


Fájlok és adatcsomagok feltöltése helyi tárolókból

- Mit értünk helyi tároló alatt
  - Intézményekben (kutatóintézet, egyetem) elsődleges adattárolásra használt szerver
  - Elsősorban Synology NAS-t célozzuk meg
    - Ilyet vettünk, több kutatóintézetben ilyet használnak
  - De bármilyen linux alapú szerveren működnek a lenti módszerek
- Megoldások helyi tárolókból feltöltésre
  - Manuális feltöltés weben
  - Közvetlen API elérés: curl, java/python/ruby/R/stb. könyvtárak
  - Dataverse Mass Uploader
  - DVUploader
  - RDM Integration Dashboard
- Helyi tároló helyett vagy mellett
  - Köztes tároló az ELKH felhőben

Manuális feltöltés webes felületen

- Egyszerű
- Szinkron
  - A feltöltő asztali számítógépe / laptopja bekapcsolva kell maradjon
  - 1 TB feltöltése minimum 3 óra
    - de inkább 30 óra
- Megszakadt feltöltés esetén manuálisan kell ellenőrizni, hogy mi lett már feltöltve
- Ha az adat helyi tárolóban van tárolva, akkor további hibalehetőségek lépnek fel



ARP SZTAKI ELKH Eötvös Loránd Research Network

Dataverse Mass Uploader

- Python script
  - Létező adatcsomagokba tölt fel fájlokat
  - pyDataverse könyvtárat használja, de lehet 2 GiB fölötti fájlokat is feltölteni, ha van curl telepítve
  - `python3 uploadFilesToExistingDataset.py -u https://science-data.hu -d HANDLE_OR_DOI -k $APIKEY FILE1 FILE2 ......`
- Szkriptelhető, csomagolható
  - Pl. több adatcsomagba feltöltéshez
- Elérhető
  - https://github.com/dsd-sztaki-hu/DataverseMassUploader
- Leendő fejlesztések
  - Fájlok felülírása
  - Adatcsomag- és fájl-metaadat feltöltés
  - RO-Crate támogatás

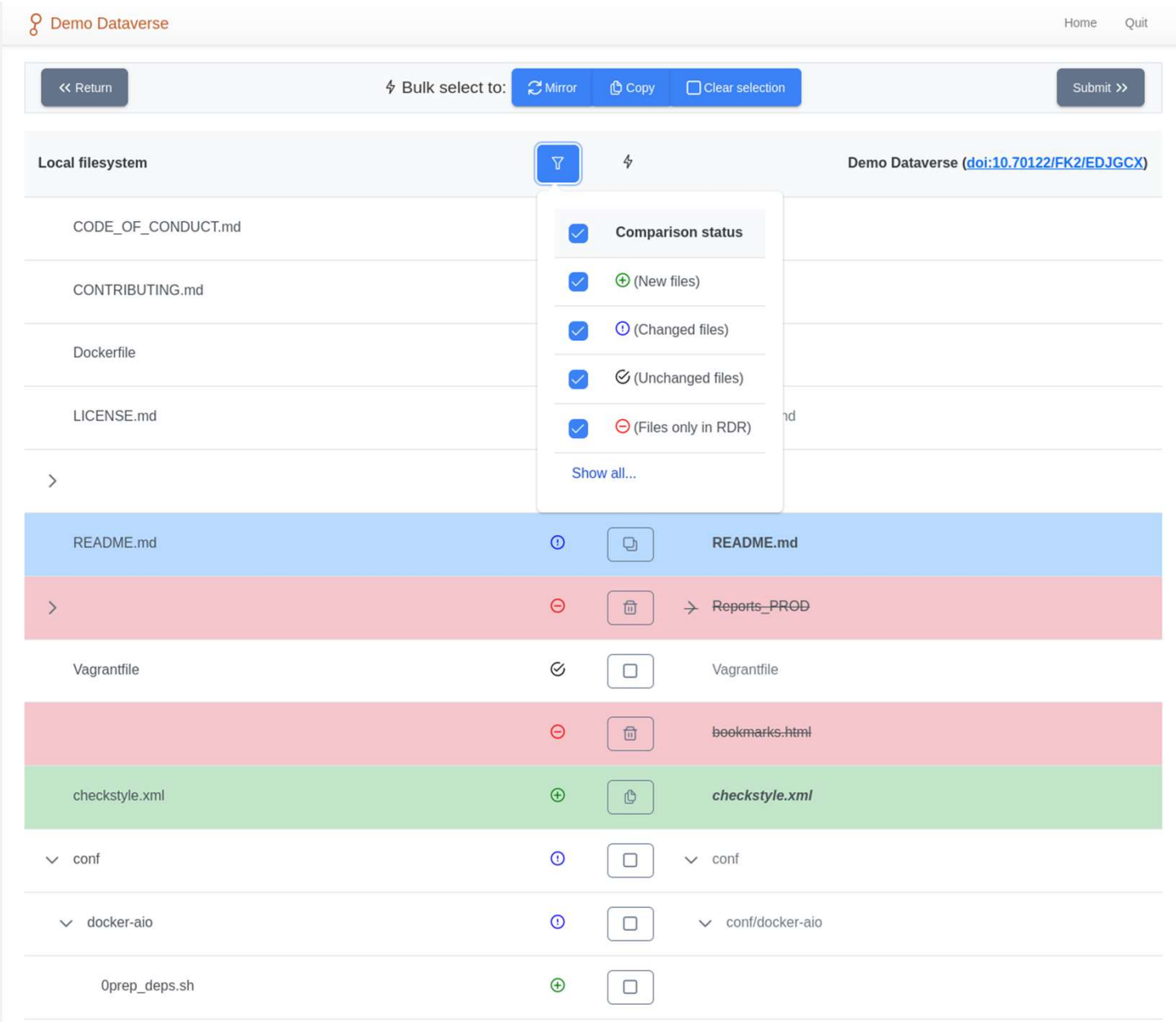
ARP SZTAKI ELKH Eötvös Loránd Research Network

DVUploader

- Java command line alkalmazás
  - Létező adatcsomagokba tölt fel fájlokat
  - java8+ kompatibilis
  - részletesen konfigurálható, rengeteg opcióval
    - rate limiting
    - konfigurálható várakozási idők
    - reguláris kifejezések
    - stb.
  - `java -jar DVUploader-*.jar -server=https://science-data.hu -did=<Dataset DOI or handle> -key=<API Key> <file or directory list>`
- Szkriptelhető, csomagolható
  - Pl. több adatcsomagba feltöltéshez
- Nincs metaadat-feltöltés
  - és nem is terveznek
- Elérhető
  - https://github.com/GlobalDataverseCommunityConsortium/dataverse-uploader

RDM Integration Dashboard

- Hasonló céllal készült mint a korábbi kettő
  - https://github.com/libis/rdm-integration
- Szép grafikus interfésszel
  - Elsősorban szinkronizálásra kitalálva
- Sok storage backend támogatott
  - Lokális
  - Github, Gitlab, IRODS
- Go nyelven
  - csak 64 bites bináris van
  - libc2.32-re fordítva
  - redis szerver is kell neki
- alfa verzió
- Nincs metaadat-feltöltés
  - és nem is terveznek



Köztes tároló az ELKH felhőben

- “Concorda Data Staging (SSH, btrfs)” image az ELKH felhőben (SZTAKI ág)
  - Helyi tároló helyett/mellett használható
  - Bárki indíthat ilyet, viszonylag nagy diszkkel (több száz TB)
  - Ide ideiglenesen gyűjthetőek adatok (6 hónap)
  - Futtatható a DataverseMassUploader vagy DVUploader
- Ki lehet próbálni!
  - https://docs.google.com/document/d/1-zJdAHBUkq2Ob630ftIFRRoM6XNW-Tjz
- Fejlesztési lehetőségek
  - Előre installált DataverseMassUploader és DVUploader, dependenciákkal

Adathozzáférés szabályozása a dataverse-ben

- Az adatcsomagok létrehozáskor DRAFT állapotba kerülnek
  - Az ilyen adatcsomagokat csak (befoglaló tárolóban) különleges hozzáféréssel rendelkezők láthatják
  - Egyedi hozzáférés adható ilyen adatcsomagokhoz a “Private URL” funkcióval (pl. cikk bírálók számára)
- Közzététel után a metaadatok hozzáférhetőek mindenki számára
  - Az adatok hozzáférhetősége akár fájlonként állítható
  - Korlátozott fájlok esetén beépített workflow használható hozzáférés engedélyezésére

Fájlok és adatcsomagok letöltése

- Webes felületen
  - Egyesével
  - ZIP formátumban
    - 100MB maximum (szerveroldalon konfigurálható, de pár GB fölött gond lehet vele)
    - metaadatokat nem tartalmaz
- RO-crate formátumban
  - vagy egy kiterjesztett ZIP vagy könyvtárstruktúra
    - könyvtárstruktúra letöltéséhez kliensoldali scriptelés kell
  - metaadatokat tartalmaz
  - fejlesztés alatt
- API-n keresztül
  - curl/wget
  - python/ruby/R/java könyvtárak
- DataverseMassUploader
  - `downloadDatasetFiles.py`

Köszönöm a figyelmet!

SZTAKI: www.sztaki.hu

DSD: dsd.sztaki.hu

Adatrepozitóium projekt: science-research-data.hu

CONCORDA: science-data.hu

ELKH CLOUD: science-cloud.hu

SZTAKI ELKH Eötvös Loránd Research Network