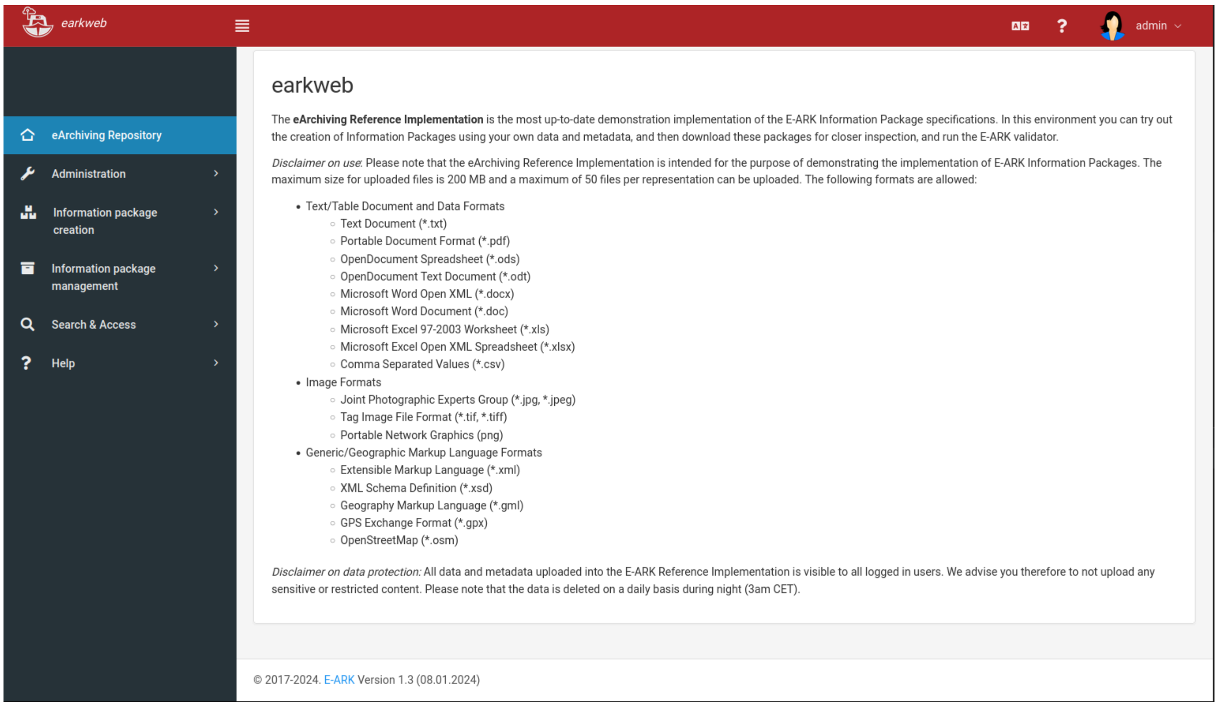
earkweb – Digital preservation repository

earkweb is an open-source archiving and digital preservation system which is based on the reference model for an Open Archival Information System (OAIS)[1] providing functions for Ingest, Archival Storage, Access, and Management of Information Packages. Information packages for Ingest, Archival Storage, and Access adhere to the eArchiving (E-ARK) information package specifications defined by the European Commission's eArchiving initiative.[2]

The life cycle of an information package starts with providing an E-ARK Submission Information Package (E-ARK SIP)[3] for ingest which can be either created by any external tool able to produce Information Packages conformant with the E-ARK SIP specification or using earkweb's integrated SIP creator. During the ingest the system executes a series of workflow steps – including the validation of the Submission Information Package against the requirements of the specification – which in case of success ends with the creation of the Archival Information Package (AIP). It also supports the creation of Dissemination Information Packages (DIPs) and the indexing of them to enable access to and full-text search in Information Packages.

Figure 1 shows the initial page after logging into the repository system with the navigation bar on the left showing the main functions for SIP creation (Information Package Creation), access to archival storage (Information Package Management) and Access (Search & Access).



earkweb

The **eArchiving Reference Implementation** is the most up-to-date demonstration implementation of the E-ARK Information Package specifications. In this environment you can try out the creation of Information Packages using your own data and metadata, and then download these packages for closer inspection, and run the E-ARK validator.

Disclaimer on use: Please note that the eArchiving Reference Implementation is intended for the purpose of demonstrating the implementation of E-ARK Information Packages. The maximum size for uploaded files is 200 MB and a maximum of 50 files per representation can be uploaded. The following formats are allowed:

- Text/Table Document and Data Formats
  - Text Document (*.txt)
  - Portable Document Format (*.pdf)
  - OpenDocument Spreadsheet (*.ods)
  - OpenDocument Text Document (*.odt)
  - Microsoft Word Open XML (*.docx)
  - Microsoft Word Document (*.doc)
  - Microsoft Excel 97-2003 Worksheet (*.xls)
  - Microsoft Excel Open XML Spreadsheet (*.xlsx)
  - Comma Separated Values (*.csv)
- Image Formats
  - Joint Photographic Experts Group (*.jpg, *.jpeg)
  - Tag Image File Format (*.tif, *.tiff)
  - Portable Network Graphics (png)
- Generic/Geographic Markup Language Formats
  - Extensible Markup Language (*.xml)
  - XML Schema Definition (*.xsd)
  - Geography Markup Language (*.gml)
  - GPS Exchange Format (*.gpx)
  - OpenStreetMap (*.osm)

Disclaimer on data protection: All data and metadata uploaded into the E-ARK Reference Implementation is visible to all logged in users. We advise you therefore to not upload any sensitive or restricted content. Please note that the data is deleted on a daily basis during night (3am CET).

© 2017-2024. E-ARK Version 1.3 (08.01.2024)

Figure 1: Screenshot of earkweb

[1] Consultative Committee for Space Data Systems (CCSDS), ISO standard 14721:2012, https://public.ccsds.org/pubs/650x0m2.pdf

[2] https://digital-strategy.ec.europa.eu/en/activities/earchiving

[3] https://dilcis.eu/specifications/sip

Technical features

earkweb provides a web frontend together with a scalable task execution system based on Celery[4] with the following beneficial properties:

- The task execution backend allows distributing tasks across multiple worker nodes, allowing for parallel processing and efficient resource utilization.
- As workload increases, additional workers can be added to handle higher volumes of tasks.
- It provides built-in mechanisms for handling task failures and retries, ensuring that tasks are completed successfully even in the presence of errors or failures.
- The asynchronous task execution allows freeing up application resources to handle other tasks while long-running or resource-intensive tasks are processed in the background.
- Tasks can be prioritised based on their importance or urgency, ensuring that critical tasks are processed promptly while less critical tasks can be queued for later execution.
- The system integrates tools for monitoring task execution, tracking task progress, and managing worker nodes, allowing for effective monitoring and optimization of task processing performance.

Task execution can be controlled and monitored using a REST API without using the web frontend.

The ingest process is implemented as a set of modular and extendible backend tasks. e*arkweb* also offers a pre-defined workflow for batch processing which executes the full chain of tasks for fully automated ingest of large volumes of data.

earkweb is Python/Django-based web application which uses a MySQL database for storing information about data sets and a Celery/RabbitMQ/Redis backend for asynchronous task processing. A docker-compose configuration file[5] allows easily setting up a local instance of the application to try out the data set creation, packaging, and storage functionality.

The e*arkweb* application is prepared for container-based deployment based on Docker[6] in order to support simple and modular installation of the software in cloud environments.

Docker is an open-source engine that automates the deployment of any application as a lightweight and portable container that will run on any platform where the Docker engine is supported.[7] In order to allow deploying services on a Docker platform, Docker containers for the individual services of *earkweb's* frontend and backend have been created.

Figure 2 gives an overview about the containers used for deployment. Each component with a “blue whale” symbol represents a component which is available as a Docker component.

[4] http://www.celeryproject.org/

[5] https://gitlab.com/datamarket/conduit/blob/master/docker-compose.yml

[6] https://www.docker.com

[7] https://docs.docker.com/engine/installation

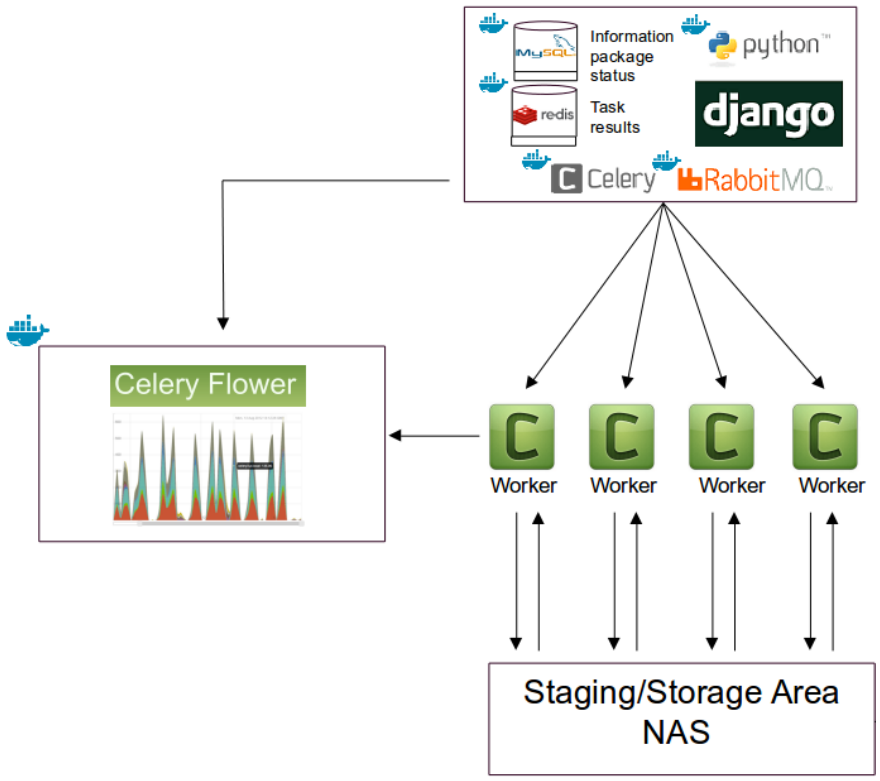


Figure 2 earkweb components overview

earkweb is based on the following container components:

- MySQL[8]
- SolR[9]
- RabbitMQ[10]
- Redis[11]
- Earkweb[12]
- Celery[13]
- Celery Flower[14]

The ingest process is composed of a set of individual tasks which are executed in a specific order to convert E-ARK submission information packages (SIPs) into E-ARK archival information package (AIPs). It is an extensible workflow which can be adapted to specific needs by inserting new tasks at any point of the workflow. *earkweb* uses a modular approach for defining atomic tasks which perform a specific transformation step of the ingest, such as the extraction of an SIP or the validation of the descriptive metadata it contains. However, a specific task does not necessarily execute one single action but can initiate a series of tasks or a complete workflow as well.

[8] http://www.mysql.com
[9] https://lucene.apache.org/solr
[10] http://www.rabbitmq.com
[11] http://redis.io
[12] http://github.com/eark-project/earkweb
[13] http://www.celeryproject.org
[14] https://github.com/mher/flower