

The 4 Components

General

Culturize exists out of 4 different components. These components being:

1. The CSV-file in which links between the PIDs and the URLs.
2. The tool made during Osoc18 that converts the CSV-file to htaccess files and interacts with GitHub.
3. The GitHub repository that stores the htaccess files
4. The server (that might use a CI tool to listen to the repository and pull in changes) which executes the redirections.

The beauty of the solution we developed is that control of each of these components is independent. This way it is possible to outsource any component to different organizations.

1. CSV-file

In the CSV-file users maintain the link between PID's and URL's, this creates the possibility of linking data to a persistent URI that will be activated by the tool. The columns of the CSV file should look like this:

PID	Document type (optional)	URL	Enabled (optional)
-----	--------------------------	-----	--------------------

More columns can be added but will be ignored by the culturize application. The CSV-file will mostly be edited by personnel of the museum itself, they can use a trusted excel-environment to modify the CSV-file and this way keep the links to which the persistent URI's redirect up to date.

2. Culturize tool

The culturize tool is a desktop application that will be used to convert the previously mentioned CSV files to htaccess files. (This type of file is used to give instruction to an apache server to execute redirections.) The tool also handles communication with the GitHub API, making the UI/UX nice and intuitive. The tool will probably be used by museum personnel but could also be used by only one maintainer an organization, it's up to the users to choose their workflow. A more detailed and technical explanation of the tool can be found in the flow.pdf file.

3. GitHub repository

The GitHub repository is used to store the .htaccess files used for the redirections. The repository should be maintained by a person with knowledge of git and the git workflow. His/her task would mainly be revising and accepting pull requests of new/updated htaccess files into the repository. This component will probably be outsourced by museums who don't have people with these skills in their ranks. It is easy

to centralize this part for multiple museums. It is however also possible for a museum to maintain their own repository if they wish to.

4. Redirection server

The server will handle redirection of the persistent URI's to their respective URL's. Culturize uses .htaccess files. This means an apache server is required. The server watches a GitHub repository using a continuous integration service (for example CircleCI <https://circleci.com/docs/2.0/>). This way the server detects incoming changes to repository, pulls and immediately deploys the changes. Provision and maintenance of a server will mostly be outsourced by museums who don't have the required equipment/people. Again, it is also possible for a museum to host the server themselves.