Relationships between Files

Description:

Some extractors while working on an original input file generate derivative files. These derivative files are usually submitted to the extraction bus so that further metadata can be extracted. While the new metadata extracted from the derivative files are indeed related to them, they are also related to the original file based on which the derivative file was created. Thus, documenting the relationship between files might be beneficial when dealing with chains of extraction procedures so to facilitate the association of any metadata generated further in line with the original file. We assume this would be especially important when searching/querying based on metadata, since while it could be useful to get to a derivative file, the main target to such procedure should be the original file.

Current examples:

1. The basic OCR extractor extracts text from image files. Currently, if some text is detected, it is associated with the original file as metadata. Nevertheless, more information could be extracted leveraging some natural language processing extractors that work on text files. So, in addition to be used as metadata, the text content could be used to create a derivative text file that in its turn would be submitted to the extraction bus. Then, we would be able for example to detect the language of the text, extract a simple summary (if applicable), etc based on the results of the text extractors. Any new information extracted would be associated with the derivative text file. However, if we maintain a record of the relationship between the original and the derivative file, we will be able to indirectly associate the new information with the original file. (In some ways this is similar to section specific metadata: although it is associated with the section, it is also associated with the original file based on the link between the original file and the section)

2. CellProfiler extractors generate csv and image files. Currently the csv files' contents are extracted and added as metadata to the original file. In addition, in Medici, all files are uploaded to the same dataset. Two issues arise from this approach:
    a. Uploading the derivative files to the same dataset as the original one does not imply that they are indeed derivative files and does not differentiate between the original files (maybe more than one) and the derivative ones.
    b. The csv contents that are extracted are currently being associated to the original file as metadata. Probably a better way of approaching it would be to have a csv extractor that would associate csv files' contents with the derivative csv files later on when these are submitted to the extraction bus. This not only makes the process more flexible and generic, but also guarantees that there won't be information lost /complex results due to naming clashes. The csv metadata would now be associated with the derivative files that contain the information and would be accessible from the original file due to the link between the original and its derivative files.

Possible approach:

We have discussed that a "link" between the original file and its derivatives should be maintained. This would work much as a tree when derivative files could also themselves be "parents" to other new derivative files. Navigating the tree (from a node downwards to the leaves) would let us gather all the metadata associated with any node in the tree. From each file we would also need to be able to reach its parent so to facilitate user interaction later on (navigating the tree upwards from a node to its parent/s node/s).

To have access to this "relationship status" between files, any file should hold a list of children/derivative files ids and its parent id. We have to take into consideration that some derivative files might have been generated not from one single file but from a combination of the information processed by an extractor from multiple files. In this case, the file will have more than one parent.