GUIDE FOR THE ANALYSIS OF PUBLISHING PLATFORMS USING ezPAARSE

To analyze a publisher's platform and its links (URLs) using the free ezPAARSE software, you will need to use different tools, such as:

- Trello (for tracking platforms),
- The Analogist platform, hosted on the ezpaarse domain

The principle of platform analysis – and that of their URLs – is to recover as much documentary information as possible that can describe an access URL to a platform, its resources (journals, books) and its "types of documents" (abstract, summary, protocol, etc.) and formats (PDF, html, etc.).

The purpose of an analysis is to allow the writing of parsers that will recognize readers' access to these analyzed platforms, resources and documents.
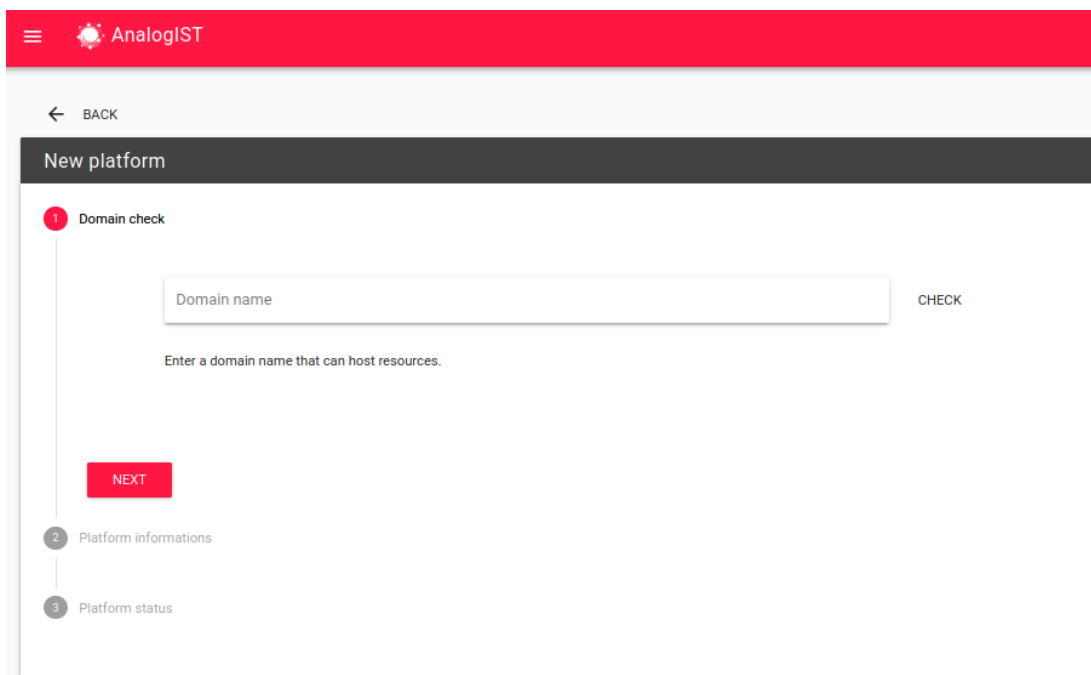
Analogist

The first operation when analyzing a new platform is to verify that it is not already declared in analogIST (search in the search bar using the name of the platform).
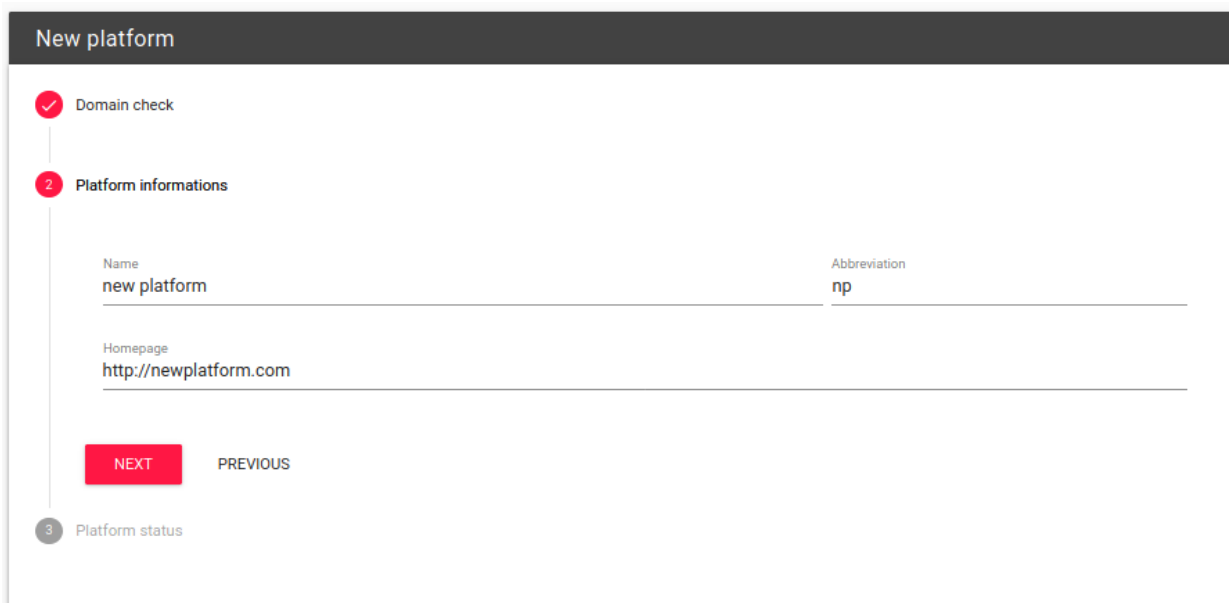
If it doesn't exist, you can create the platform by clicking on   .

Enter the domain (e.g.: *onlinelibrary.wiley.com*) to see whether it is already supported by a parser.

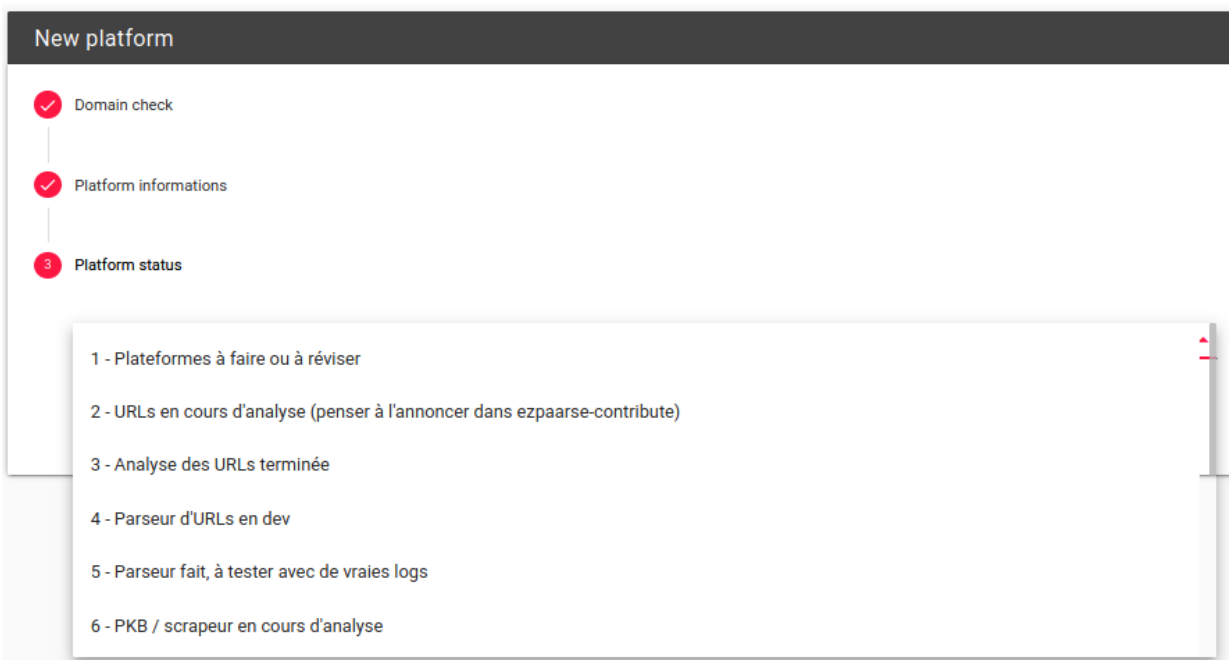If the domain does not exist, fill-in the information about the platform (name, abbreviation, homepage)

Choose the status of the platform (for a creation, use status 1 "platforms to create or revise") in the pull-down menu:

End the creation by clicking on "create".
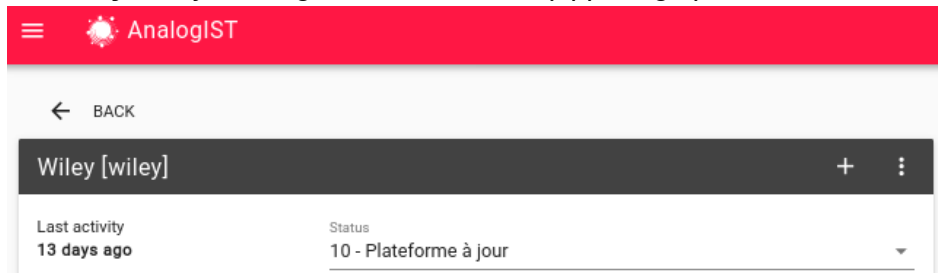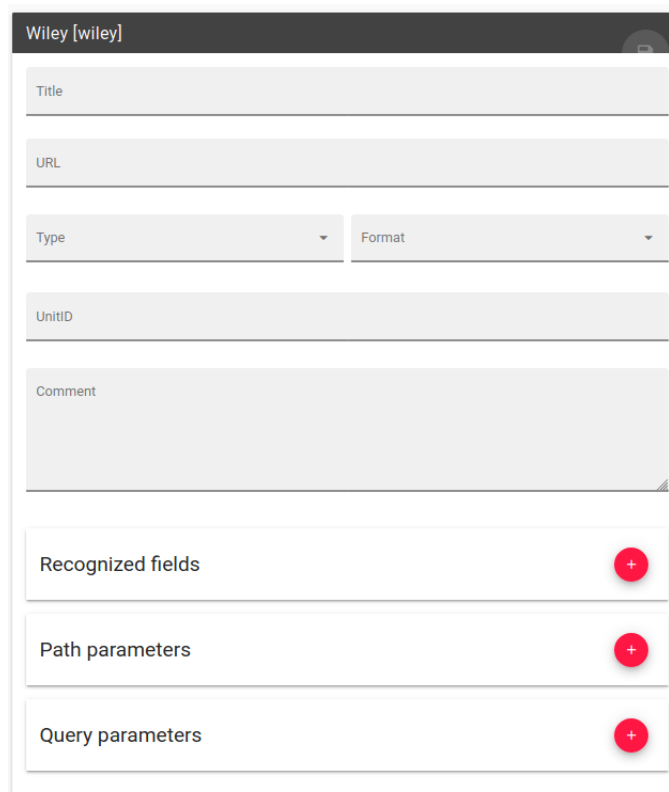

You can also return to the Trello tracking board. Your identifier will be filled-in automatically on the card concerning your platform.
https://trello.com/b/wEaLnz8d/ezpaarse-analogist-suivi-des-plateformes

When the **platform** document has been created,
begin the **URL analysis** by clicking on the "+" button (upper right)

A new screen will open, and you can begin the analysis:

Fill-in the form, by indicating:
- the title of your analysis (e.g.: viewing an article in PDF format)
- the URL consulted
- the type of document (ARTICLE, TOC, ABS, BOOK, etc.)
- the format (HTML, PDF, etc.)
- the UnitID (the identifier of the article)
- a comment (if desired)

Click on:

to add a recognized field to the URL (doi, pii, print_identifier, online_identifier, etc.)

The card is automatically saved.

❖ *Guidelines/Recommendations:*
Describe accurately all the identifiable elements in a URL.
It is **essential** to fill in the resource identifier (*title_id*) and the article or chapter identifier (*UnitID*).
The *title_id* identifier of the resource (journal, ebook) is important because later it will serve to allow the association with a knowledge base (Kbart lists) and the creation of a PKB to precisely identify each resource (publication_title, ISSN, publisher, etc.) in the resulting Access Events (AEs).

Do not hesitate to make comments that may be useful for the later writing of parsers.

Analyze recent issues of journals as well as archives, supplements etc. The writing of URLs may differ within a single journal. This observation is even more pronounced for a platform giving access to several journals. Thus, for a single platform you can have a multitude of URL descriptions, for the same type of articles in pdf format for example.

Abstracts, contents lists etc. are often in a format other than PDF or html. Most often, they must be listed as being in a miscellaneous format.

It is difficult to be exhaustive straight from the first passage in the analysis of a platform; it will be necessary to return and update platform analyses after the writing and/or parser testing phases, or even after the final phase (list of 10 / updated platform).

> - In the analysis of a platform and its URLs it is possible to find and describe a multitude of documents (to be taken in the generic sense of the term): review, contents list (TOC), abstract, PDF article, HTML article, BOOK, book chapter, protocol, etc. The exhaustive list is available on:
> https://ezpaarse-project.github.io/ezpaarse/essential/ec-attributes.html#resources-types-rtype
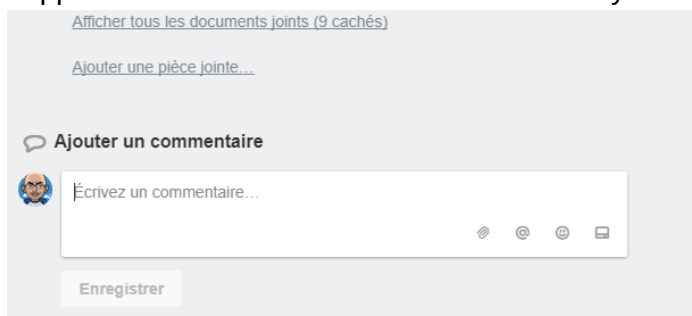
Trello

Our public Trello board can be found at: https://trello.com/b/wEaLnz8d/ezpaarse-analogist-suivi-des-plateformes

Trello is a collaborative tool for the follow-up in 10 steps of the platform analysis, from the platform creation/revision step to the final step where the platform is up-to-date at the ezPAARSE level, passing by the steps of URL analysis, parser creation, parser tests etc.

When a task is finished, don't forget to drag this label/card to the next list.

Instructions/recommendations:

It is possible to add comments to "cards" (even those you did not create yourself). As such, don't forget to click on "How" to save your comment (this may seem a trivial point, but it has happened several times that I didn't validate my comment).

You can also ask questions directly via the Trello board, to get help from a team member.

Follow-up of operations (parsers, PKB, scraper)

The information collected during the previous step of URL analysis is the starting point for the work on ezPAARSE. It is decisive for the continuation of operations which are more computer-linked than documentary in nature and require proven computer skills.

- The analysis information allows progression to the writing step of the **parser** (writing regular expressions from the information given in the documentary analysis).

See "Writing the parser (encoder)" column under http://doc.ezpaarse.org

In a nutshell, the resulting parser will look like a series of such *if* blocks:

```
if ((match = /S([0-9]{4})([0-9]{3}[0-9Xx])([0-9A-Za-z]*)/.exec(param._pii)) !
== null) {
// example : http://pdn.sciencedirect.com.gate1.inist.fr/science?
_ob=MiamiImageURL&_cid=282179&_user=4046392 //
&_pii=S221267161200100X&_check=y&_origin=browseVolIssue&_zone=rslt_list_item&_
coverDate=2012-12-31 // &wchp=dGLbVlB-
zSkWz&md5=79a307d3c9bdbea6d6a6092d73c25545&pid=1-s2.0-S221267161200100X-
main.pdf
result.unitid = 'S' + match[1] + match[2] + match[3];
result.title_id = match[1] + match[2];
result.print_identifier = match[1] + '-' + match[2];
result.rtype = 'ARTICLE';
result.mime = 'PDF';
```

This parser will then have to be tested with real logs.

- After this parser step (and parser tests), it is possible to create a Publisher Knowledge Base (PKB) file, often based on a provider's knowledge base (containing publisher's data such as publication title, title ID etc.).

See how to create a PKB under http://doc.ezpaarse.org

- as well as the creation of a scraper linked to that of the PKB, because a PKB can also be automatically generated by a scraper when no KBART file is available.

All parsers, PKBs, scrapers that have already been created are visible on our Github repository:
https://github.com/ezpaarse-project/ezpaarse-platforms

In conclusion, the analysis of a platform and its URLs is a decisive and crucial stage in the ezPAARSE workflow!