DevConnect 2023: EZproxy, ezPAARSE, and Community Collaboration

Presented by the EZproxy Product Team at OCLC
Sean Duffy
Product Manager

Karen Coombs
Senior Product Analyst

Justyn Shye
Product Analyst

Carly DiVito
Community Manager
Poll question
What do you hope to learn from this session?
Problem Statement

Campus libraries hold massive stores of data, but information about their collections isn’t always tied holistically to academic outcomes and educational activity.

**EZproxy logs hold valuable data**, but the data is confusing. How is it possible for a single person or department to get a handle on all of this?
EZproxy logs 101
About EZproxy logs:
The base that EZproxy Analytics is built upon

- **Audit logs**: User login and logout events
- **Starting Point URL (SPU) logs**: The initial click that starts user’s EZproxy journey
- **EZproxy logs**: The activity of a user while in proxy
What is ezPAARSE
ezPAARSE

• An open-source tool designed to process and analyze EZproxy log files
• Developed and maintained by the French consortium COUPERIN
• Transforms raw EZproxy Log data into readable/useable eResource Usage Data using parsers
Parsers – The core of ezPAARSE

- Sort of reverse of an EZproxy Stanza
- Community created and maintained
- Link to Github: https://github.com/ezpaarse-project/ezpaarse-platforms
- Each resource needs its own parser
Methods for utilizing ezPAARSE
EZproxy and ezPAARSE

- Download and Maintain ezPAARSE
- Build and run the jobs
- Responsible for log maintenance, ensuring matching log format statements, updating ezpaarse/parsers
- Responsible for repository for enriched data
OCLC’s EZproxy Analytics

- OCLC manages deployment from creation to updates
- Work with OCLC staff to recognize vision, work with implementers to get setup
- Work with OCLC to troubleshoot issues
- Ability to search and visualize other EZproxy logs such as Audit and SPU
How to get involved with ezPAARSE
Multiple ways to help with parser development

AnalogIST
http://analyses.ezpaarse.org/

Developing Parsers
- Create parsers from scratch
- Updating existing parsers
  - Fixing URLs in manifests
ezPAARSE now comes with a list of 557 parsers, which are the result of an analytical work provided by the community, and which provide users with a good coverage of their institution's subscriptions to electronic resources.

The community platform, called AnalogiST, is the place where different types of URLs are collected, semi-automatically analyzed and commented so that the corresponding parser can be accurate and complete. For publishers or providers who are not yet recognized by ezPAARSE, or that simply need an update, it is easy to contribute to a platform analysis: just open a Trello account and follow the guide!

Once the analysis is considered satisfactory, or complete, a programmer (usually from the ezPAARSE/izMeSURE team will take over and implement the corresponding parser. This parser will be added to the existing parsers in the dedicated github repository and can be retrieved by all instances of ezPAARSE, through a simple update mechanism embedded in the software administration area. This mode of operation ensures that each new parser, as well as each software and resource update, benefits all existing installations of ezPAARSE: work is required only in one place and only once.
### Platforms (784)

<table>
<thead>
<tr>
<th>Platform Name</th>
<th>Last Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>123rf [123rf]</td>
<td>18 days ago</td>
</tr>
<tr>
<td>20th Century German History Online [dgo]</td>
<td>about 1 year ago</td>
</tr>
<tr>
<td>4 Canoes [4cn]</td>
<td>5 days ago</td>
</tr>
<tr>
<td>4nmedia</td>
<td>6 months ago</td>
</tr>
<tr>
<td>A to Z World Food [AZfood]</td>
<td>18 days ago</td>
</tr>
<tr>
<td>Aarhus University Press / Aarhus Universitetsforlag [AUF]</td>
<td>18 days ago</td>
</tr>
<tr>
<td>Academic Writer [aw]</td>
<td>18 days ago</td>
</tr>
</tbody>
</table>
New Platform

1. Domain Check

   Domain Name

   Enter a domain name that can host resources.

   NEXT

2. Platform Information

3. Platform Status
## New Platform

### Domain Check
- Domain Check

### Platform Information

<table>
<thead>
<tr>
<th>Name</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Homepage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

### Platform Status
EMIS [emis]

Updated 5 days ago by Sean Duffy

REPORT

Path Parameters

<table>
<thead>
<tr>
<th>Value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>php</td>
<td></td>
</tr>
<tr>
<td>search</td>
<td></td>
</tr>
<tr>
<td>docpdf</td>
<td></td>
</tr>
</tbody>
</table>

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>doc_id</td>
<td>766831747</td>
</tr>
</tbody>
</table>
ezPAARSE – Manifest Updates

- Manifest contains the host names to match in the URL
- When new hosts names are added by a provider they need to be added to the manifest
ezPAARSE – Skills to Writing Parsers

• Understanding of Git workflows
• Node.js (Javascript) knowledge
• Ability to write RegExs
• Command line (optional)
ezPAARSE – Updating a Parser

1. Fork the ezpaarse parser repo
   – https://github.com/ezpaarse-project/ezpaarse-platforms
2. Checkout the code
3. Make a branch for your changes
4. Pull in new test file from Analogist
5. Create a regex for each new or changed url
6. Test your stuff
7. Commit your changes
8. Create a pull request
<table>
<thead>
<tr>
<th>Author</th>
<th>Description</th>
<th>Date</th>
<th>Commits</th>
</tr>
</thead>
<tbody>
<tr>
<td>@thmlu</td>
<td>Added parser for Know BC (refactored version) ezpaarse</td>
<td>7 hours ago</td>
<td>3,048</td>
</tr>
<tr>
<td>Husky</td>
<td>Fix: grep return exit code</td>
<td>2 years ago</td>
<td></td>
</tr>
<tr>
<td>Jib</td>
<td>Test: add checks for some manifest fields</td>
<td>2 months ago</td>
<td></td>
</tr>
<tr>
<td>123rf</td>
<td>Merge pull request ezpaarse-project#654 from OCLC-Developer-Net</td>
<td>6 months ago</td>
<td></td>
</tr>
<tr>
<td>4cn</td>
<td>Refactor: lint</td>
<td>5 days ago</td>
<td></td>
</tr>
<tr>
<td>MIT</td>
<td>Adds parser for MIT Press [MIT]</td>
<td>5 years ago</td>
<td></td>
</tr>
<tr>
<td>aaajournals</td>
<td>Update docurl (ang link) for MIT Press</td>
<td>5 years ago</td>
<td></td>
</tr>
<tr>
<td>aaas</td>
<td>aaas: create parser</td>
<td>Last year</td>
<td></td>
</tr>
<tr>
<td>afl</td>
<td>Rename REF to RECORD_VIEW</td>
<td>8 months ago</td>
<td></td>
</tr>
<tr>
<td>ab</td>
<td>lint: remove useless catch</td>
<td>Last year</td>
<td></td>
</tr>
<tr>
<td>abcn</td>
<td>Added parser for Ancient Books</td>
<td>Last year</td>
<td></td>
</tr>
<tr>
<td>access-engineering</td>
<td>Created parser for access engineering</td>
<td>2 years ago</td>
<td></td>
</tr>
<tr>
<td>acl</td>
<td>Added parser for access learning</td>
<td>2 years ago</td>
<td></td>
</tr>
</tbody>
</table>

**About**

Platforms parsers, scrapers and PKBs for ezPAARSE

- **Readme**
- 0 stars
- 0 watching
- 21 forks

**Releases**

No releases published

Create a new release

**Packages**

No packages published

Publish your first package

**Languages**

- JavaScript 99.2%
- Other 0.8%
<table>
<thead>
<tr>
<th>Platform</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCESS</td>
<td>10 - Plateforme à jour</td>
<td>This platform is not yet certified</td>
</tr>
<tr>
<td>BOOK PAGE</td>
<td>9 months ago</td>
<td>This platform is not yet certified</td>
</tr>
</tbody>
</table>
'use strict';
const Parser = require('./lib/parser.js');

/**
 * Recognizes the accesses to the platform Thomson Reuters Proview
 * @param {Object} parsedUrl an object representing the URL to analyze
 *   main attributes: pathname, query, hostname
 * @param {Object} ec an object representing the EC whose URL is being analyzed
 * @return {Object} the result
 */
module.exports = new Parser(function analyseEC(parsedUrl, ec) {
  let result = {};
  let path = parsedUrl.pathname;
  // uncomment this line if you need parameters
  let param = parsedUrl.query || {};

  // use console.error for debugging
  // console.error(parsedUrl);

  if (RegExp(/catalog\..*/i).test(path)) {
    result.type = 'SEARCH';
    result.mime = 'HTML';
  } else if (RegExp(/\title\..*/i).test(path)) {
    // https://proview.thomsonreuters.com/title.html?redirect=true&titleKey=5692FFULLX2FSASAP2E2K2Fv1.&titleStage=F#ttitleAcct=10nd629760000001746c10a657d25c47ed#s1=p6&lid=3
    // https://proview.thomsonreuters.com/title.html?redirect=true&titleKey=5692FFULLX2FSASAP2E2K2Fv1.&titleStage=F#ttitleAcct=10nd629760000001746c10a657d25c47ed#s1=p6&lid=4
    result.type = 'BOOK_PAGE';
    result.mime = 'HTML';
    result.title_id = param.titleKey;
    result.unitid = param.titleKey;
  }

  return result;
});
'use strict';
const Parser = require('./lib/parser.js');

/***/
* Recognizes the accesses to the platform Thomson Reuters Preview
* @param {Object} parsedUrl an object representing the URL to analyze
* main attributes: pathname, query, hostname
* @param {Object} ec an object representing the EC whose URL is being analyzed
* @return {Object} the result
*/
module.exports = new Parser(function analyzeEC(parsedUrl, ec) {
  let result = {};
  let path = parsedUrl.pathname;
  // uncomment this line if you need parameters
  let param = parsedUrl.query || {};

  // use console.error for debugging
  // console.error(parsedUrl);

  if (path.endsWith('/i.test(path)')) {
    result.type = 'SEARCH';
    result.mime = 'HTML';
  } else if (path.endsWith('/i.test(path)')) {
    // https://preview.thomsonreuters.com/title.html?redirect=true&titleKey=5N2FFULN2FAISAP2ED2Fv1.0&titleStage=F&titleAccct=10cc62976800001746c1o662e25c47eod$sl=pl&eid=3
    result.type = 'BOOK_PAGE';
    result.mime = 'HTML';
    result.title_id = param.titleKey;
    result.unittest = param.titleKey;
  } else if (path.endsWith('/i/test(path)')) {
    // https://next-preview.thomsonreuters.com/title
    result.type = 'QUERY';
    result.mime = 'HTML';
  }
});
COOMBSK-3MBAHO:devnet-ezpaarse-platforms coombsk$ make test trp
cd .lib && EZPAARSE_PLATFORM_TO_TEST="trp" npm test

> ezpaarse-platforms@1.0.0 test
> mocha test/**-test.js

Thomson Reuters Proview
✓ Manifest is valid
ct=i0ad62976000001746c1a652e25c47ead#sl=p&eid=39fb769eefc451298626b5ec528616c5&eat=&pg=cover&psl=p
✓ Test https://proview.thomsonreuters.com/catalog.html#/catalog/anacat/results?keyword=Label&thesaurus=
ct=i0ad62976000001746c1a652e25c47ead#sl=p&eid=62148b13d108bac08d4131c19365222d&eat=&pg=89&psl=&nvgS=false
✓ Test https://next-proview.thomsonreuters.com/title
cnt=i0ad62976000001746c1a652e25c47ead#sl=p&eid=62148b13d108bac08d4131c19365222d&eat=&pg=89&psl=&nvgS=false
✓ Test https://proview.thomsonreuters.com/catalog.html#/catalog/anacat/results?keyword=Label&thesaurus=
cnt=i0ad62976000001746c1a652e25c47ead#sl=p&eid=39fb769eefc451298626b5ec528616c5&eat=&pg=cover&psl=p

8 passing (5ms)

make: `trp' is up to date.
COOMBSK-3MBAHO:devnet-ezpaarse-platforms coombsk$
updated parser for Thomson Reuters Proview #728

librarywebchic wants to merge 1 commit into ezpaarse-project:master from OCLC-Developer-Network:trp

librarywebchic commented now

No description provided.

librarywebchic requested a deployment to development now — with GitHub Actions in progress

Add more commits by pushing to the trp branch on OCLC-Developer-Network/ezpaarse-platforms.

This branch is being deployed
1 in progress deployment

Some checks haven't completed yet
1 queued and 1 in progress checks
- Run tests on pull request / run-tests (pull_request) Queued — Waiting to run this check... Details
- Run lister on pull request / run-lister (pull_request) In progress — This check has started... Details
ezPAARSE – Creating a Parser

1. Fork the ezpaarse parser repo
   – https://github.com/ezpaarse-project/ezpaarse-platforms
2. Checkout the code
3. Make a branch for your changes
4. Create the base parser
5. Create a regex for each url you care about
6. Test your stuff
7. Commit your changes
8. Create a pull request
<table>
<thead>
<tr>
<th>Platform Name</th>
<th>Type</th>
<th>Created</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT - 3</td>
<td>REPORT / HTML</td>
<td>about 7 hours ago</td>
</tr>
<tr>
<td>REPORT - 2</td>
<td>REPORT / PDF</td>
<td>about 7 hours ago</td>
</tr>
<tr>
<td>DATASET</td>
<td>DATASET / XLS</td>
<td>about 7 hours ago</td>
</tr>
<tr>
<td>RECORD</td>
<td>RECORD / HTML</td>
<td>about 7 hours ago</td>
</tr>
<tr>
<td>REPORT</td>
<td>REPORT / PDF</td>
<td>about 7 hours ago</td>
</tr>
<tr>
<td>SEARCH</td>
<td>SEARCH / HTML</td>
<td>about 7 hours ago</td>
</tr>
</tbody>
</table>
COOMBSK-3MBAH0:devnet-ezpaarse-platforms coombsk$ git branch emis
Switched to branch 'emis'
COOMBSK-3MBAH0:devnet-ezpaarse-platforms coombsk$ git checkout emis
COOMBSK-3MBAH5:devnet-ezpaarse-platforms coombsk$ make init
cd .lib && npm run create-parser

> ezpaarse-platforms@1.0.0 create-parser
> node ./platform-init.js

? What's the URL of the analysis? http://analyses.ezpaarse.org/platforms/63ab44187efb2804c14a6164
? Confirm analysis ID 63ab44187efb2804c14a6164
 ✓ Fetch data from Trello
 ✓ Fetch analyses from ANG
? What's the full name of the platform? EMIS
? What's the abbreviated name of the platform? emis
? How would you describe this parser? Recognizes the accesses to the platform EMIS
? Who's the contact for this parser? Sean Duffy
? Does it have a knowledge base? No

Now we are going to list the fully qualified domain names that should be handled by the parser.
(eg. www.example.org, test.example.org, etc...)
(in addition to those found in the knowledge base)

The following domain names were extracted from the remote analysis.

? Check the domains that should be handled www.emis.com
If some domains are missing, enter them now.
(empty answer to stop)
? Enter a domain name

Ready... go!

✓ Create root directory
✓ Create test directory
✓ Create pkg directory
✓ Create scrapers directory
✓ Create manifest.json
✓ Create parser.js
✓ Create test file

Parser generated at /Users/coomsk/git/devnet-زمةאר-Platforms/emis
COOMBSK-3MBAHO:devnet-ზ MMArse-Platforms coomsk$
```javascript
#!/usr/bin/env node

`use strict`;

const Parser = require('./lib/parser.js');

/**
 * Recognizes the accesses to the platform EMIS
 * @param {Object} parsedUrl an object representing the URL to analyze
 * @param {Object} ec an object representing the EC whose URL is being analyzed
 * @return {Object} the result
 */

module.exports = new Parser(function analyzeEC(parsedUrl, ec) {
  let result = {};
  let path = parsedUrl.pathname;
  // uncomment this line if you need parameters
  // let param = parsedUrl.query || {};

  // use console.error for debugging
  // console.error(parsedUrl);

  let match;

  if ((match = /^\//platform\//path\//to\//\[document-\([0-9]++\)\-test\].pdf$/i).exec(path)) !== null) {
    // http://parser.skeleton.js/platform/path/to/document-123456-test.pdf?sequence=1
    result.rtype = 'ARTICLE';
    result.mime = 'PDF';
    result.title_id = match[1];

    // unitid is a crucial information needed to filter double-clicks phenomenon, like described by COUNTER
    // it described the most fine-grained of what's being accessed by the user
    // it can be a DOI, an internal identifier or a part of the accessed URL
    // more at http://ezxparse.readthedocs.io/en/master/essential/ec-attributes.html#unitid
    result.unitid = match[2];
  }
});
```
} else if (/^\//search//searchv2$/i.test(path)) {
    // https://www.emis.com/php/search/searchv2
    result.rtype = 'SEARCH';
    result.mime = 'HTML';
}
} else if ( /^\/php\/companies\//index$/i. test(path)) {
    result.rtype = 'RECORD';
    result.mime = 'HTML';
    result.unitid = param.cmpy;
}

} else if ( /^\/php\/search\//searchv2$/i. test(path)) {
    // https://www.emis.com/php/search/searchv2
    result.rtype = 'SEARCH';
    result.mime = 'HTML';
}
else if (/^\//php\//search\//pdf2html$\//i.test(path)) {
    result.rtype = 'REPORT';
    result.mime = 'HTML';
    result.unitid = param.doc_id;
} else if (/^\//php\//companies\//index$\//i.test(path)) {
    result.rtype = 'RECORD';
    result.mime = 'HTML';
    result.unitid = param.cmpy;
} else if (/^\//php\//search\//searchv2$\//i.test(path)) {
    // https://www.emis.com/php/search/searchv2
    result.rtype = 'SEARCH';
    result.mime = 'HTML';
}
if ("^/php\//search\//docpdf\$/i.test(path)) {
  result.rtype = 'REPORT';
  result.mime  = 'PDF';
  result.unitid = param.doc_id;
} else if ("^/php\//search\//pdf2html\$/i.test(path)) {
  result.rtype = 'REPORT';
  result.mime  = 'HTML';
  result.unitid = param.doc_id;
} else if ("^/php\//companies\//index\$/i.test(path)) {
  result.rtype = 'RECORD';
  result.mime  = 'HTML';
  result.unitid = param.cmpy;
} else if ("^/php\//search\//searchv2\$/i.test(path)) {
  // https://www.emis.com/php/search/searchv2
  result.rtype = 'SEARCH';
  result.mime  = 'HTML';
} else if ("^/php\//search\//searchv2\$/i.test(path)) {
  // https://www.emis.com/php/search/searchv2
  result.rtype = 'SEARCH';
  result.mime  = 'HTML';
}
make test emis

cd .lib & & EZPAARSE_PLATFORM_TO_TEST="emis" npm test

> ezpaarse-platforms@1.0.0 test
> mocha test/*-test.js

EMIS
✓ Manifest is valid
✓ Test https://www.emis.com/php/search/searchv2
✓ Test https://www.emis.com/php/search/docpdf?id=766831747
✓ Test https://www.emis.com/php/companies/index?pc=HK&cmpy=9737982
✓ Test https://www.emis.com/php/companies/index/keystatsbox?pc=HK&cmpy=9737982&hideValues=&currency=HKD&display_units=3&excel=1&tbl=keystats-page-table-exchange
✓ Test https://www.emis.com/php/search/pdf2html?pc=BR&doc_id=719516695&type=1

7 passing (4ms)

make: `emis' is up to date.
COOMBSK-3MBAH0:devnet-ezpaarse-platforms coombsk
$ git add emis/
COOMBSK-3MBAH0:devnet-ezpaarse-platforms coombsk
$ git commit -m "added parser for EMIS"
$ cd .lib && EZPAARSE_PLATFORM_TO_TEST="emis" npm test

> ezpaarse-platforms@1.0.0 test
> mocha test/*-test.js

EMIS
✓ Manifest is valid
✓ Test https://www.emis.com/php/search/searchv2
✓ Test https://www.emis.com/php/search/docpdf?doc_id=766831747
✓ Test https://www.emis.com/php/companies/index?pc=HK&cmpy=9737982
✓ Test https://www.emis.com/php/companies/index/keystatsbox?pc=HK&cmpy=9737982&hideValues=&currency=HKD&display_units=3&excel=1&tbl=keystats-page-table-exchange
✓ Test https://www.emis.com/php/search/pdf2html?pc=BR&doc_id=719516695&type=1

7 passing (5ms)

make: `emis' is up to date.
[emis a470d95] added parser for EMIS
3 files changed, 72 insertions(+)
create mode 100644 emis/manifest.json
create mode 100755 emis/parser.js
create mode 100644 emis/test/emis.2023-04-12.csv
Community Collaboration
Community Insights:

Houses content created and submitted by your fellow EZproxy users.

[oc.lc/ezproxy-insights]
Community Roundtable: Various Ways of Collecting EZproxy Usage Statistics

oc.lc/ezproxy-usage
Upcoming Webinar:

Improving learning outcomes with EZproxy Analytics

Date: 9 May 2023
Time: 11:00 am-12:00 pm EDT

oc.lc/learning-outcomes
Community Center: upcoming functionality and resources

1. Subscribe to enhancement suggestions
2. Search across communities
3. Improved subscription management with Digest Notification options

To see planned updates from the community, visit oc.lc/community-enhancements
Accessing the Community Center

• Request credentials:
  – oc.lc/cc-request

• Access the Community Center:
  – oc.lc/community
  – If you have access for another community already. You can use your existing credentials.
Next Up:

25 April:
Automated Checking of Items in a Talis Reading List

2 May:
WMS Connect x DevConnect - Let Me See Your License: Adding Title Level Notes to Limited-Access Ebooks

16 May:
Using jq with OCLC APIs

23 May:
Seamless-smart-fulfillment-between-Tipasa-Alma-Primo

See full schedule here: oc.lc/devconnect-23
Access OCLC’s list of nearly 600 parsers, today.

oc.lc/ezpaarse-platforms