

Tools Catalog Version 2.0

This page is under construction.

Background

Tools Catalog is the main web application that the tool contributors of Brown Dog will interact with. Current version of Tools Catalog (0.4.0) needs to be simplified and made more user friendly, in order for that to happen. Tools Catalog 0.4.0 is written in Scala using the Play framework, which is less popular than languages like Python, JavaScript, etc. There are also many features in Tools Catalog 0.4.0, which are either not needed or not clearly understood by the users. This will be an opportunity to re-design and re-implement Tools Catalog with only the required set of functionalities, which are user friendly as well.

Name

We have this ongoing task  **BD-736** - More accurate name for Tools Catalog  and will *tentatively* use the name "Brown Dog Transformations" (previously suggested by Ben) or "BD Transformations" in short as we describe Tools Catalog "2.0" on this page. For the remaining part of this document we will use BD Transformations in place of Tools Catalog 2.0. This will help clarify thoughts related to Tool vs Software.

Definitions

This section will define the main concepts that will need to be clearly understood for easily interacting with BD Transformations. An example can be Software vs Transformation or any new terminology that will be used in BD Transformations.

Converter: Program that does data format conversion.

Extractor: Program that process and extract metadata from data.

Transformation: Any converter or extractor or extractor program.

Software: Software tool / library that is used by an extractor or converter to do data transformation.

converters_info.json: A JSON file that describes a converter.

extractors_info.json: A JSON file that describes an extractor.

Minimum Viable Product (MVP): Version of BD Transformations with only a few basic features. This will be developed through some kind of a hackathon.

User Roles

The user roles in BD Transformations will be:

1. **Administrator**
 - a. Update / Delete transformations created by contributors
 - b. Update / Delete software added by contributors?
 - c. All **Viewer** permissions
2. **Contributor**
 - a. Create new transformation and link it to one or more existing software.
 - b. Add a new software to BD Transformations
 - c. Update created transformations [adding the word "created" to make it clear that a contributor can only manage the transformations they contributed]
 - d. Delete created transformations
 - e. View created transformations
 - f. All **Viewer** permissions
3. **Viewer** (can be any user who does not login)
 - a. View all transformations
 - b. View all software

Features

- All users can view all transformations that are linked to a software.
- All users can view a single transformation and the software that it uses.
- Contributors will be able to add a transformation through the BD Transformations interface.
 - This will be made available through a form that the contributor can fill and submit.
 - Ability to include icon
 - Ability to include link to software web page (e.g. ImageJ)
 - Ability to add a URL to the transformation - source code / docker image name / website
- Contributors will also be able to import a new transformation into the catalog by uploading an `extractors_info.json` or `converters_info.json` file.
 - Any missing piece of information can be filled in by the contributor.

- Contributors will be able to link their transformation with one more existing software in BD Transformations. They will be able to add a new software if it doesn't already exist in BD Transformations.
- There won't be any formal approval process for the submitted transformations. All transformations that get submitted will be available to the public.
- Administrators can update or delete any transformation as they feel appropriate.

Wireframes



1. Home (Yan)
2. View Transformation (Sandeep)
3. Login (Bing)
4. POST Endpoint
5. Administrator
6. Contributor (Low priority for now)

Technologies

- Flask Templates
- Bootstrap
- JQuery

- Flask Services
- MongoDB
- Python LDAP Library
- Python MongoDB Library

Tasks

- Script for exporting data from Tools Catalog to BD Transformations
- Code to talk to NCSA LDAP for authentication
- Clusterman endpoint to read from BD Transformations database
- Flask Services
 - POST <extractors/converters>_info.json to MongoDB
 - GET <extractors/converters>_info.json for a specific transformation
 - GET all transformations
 - GET all transformations that uses a specific software

JSON Document Example

Mapping Table

Note: Changing to camel case notation for field names as MongoDB documents follow JSON.

Sl. No.	Tools Catalog Field	Transformations Catalog Field	Type	Remarks
1.	status	N/A	Integer	
2.	creationDate	dateCreated	Date	
3.	updateDate	dateUpdated	Date	
4.	downloads	N/A	Integer	
5.	license	N/A	String	
6.	whatsnew	description	String	
7.	compatibility	N/A	String	
8.	url	url	String	URL
9.	sample_input	N/A	String	URL
10.	sample_output	N/A	String	URL
11.	interfaceLevel	N/A	Integer	
12.	dockerimageName	dockerImageName	String	
13.	vmImageName	N/A	String	
14.	dockerfile	N/A	String	Email ID
15.	author	author	String	
16.	dependencies	dependencies	Array	Array of Strings
17.	transformation_type	transformationType	String	Originally this field type was an integer with values 1 for converter and 2 for extractor
18.	transformation_id	transformationId	String	Extractor name / converter name
19.	title	title	String	

Transformations Catalog Example

```
{
  "_id" : ObjectId("53864aee7400009900ec8d22"),
  "dateCreated" : ISODate("2014-05-28T20:45:34.412Z"),
  "dateUpdated" : ISODate("2016-11-21T04:09:47.836Z"),
  "description" : "Identify tumors in microscopy images.",
  "url" : "https://opensource.ncsa.illinois.edu/bitbucket/projects/CATS/repos/extractors-cv/browse/cellprofiler/cellprofiler-tumor?raw",
  "dockerImageName" : "",
  "author" : "ruiliu@illinois.edu",
  "dependencies" : [
    "Cell Profiler"
  ],
  "externalServices" : [
  ],
  "transformationType" : "extractor",
  "transformationId" : "",
  "title" : "Tumors",
  "status" : "submitted"
}
```

Extractor Info Example

```
{
  "@context": "http://clowder.ncsa.illinois.edu/contexts/extractors.jsonld",
  "name": "ncsa.csvheaders",
  "version": "1.0",
  "description": "Extracts the headers of CSV file. Only looks at the first row.",
  "author": "Yan Zhao <yanzhao3@illinois.edu>",
  "contributors": ["Luigi Marini <lmarini@illinois.edu>"],
  "contexts": [
    {
      "csv_headers": "http://clowder.ncsa.illinois.edu/metadata/ncsa.csvheaders#csv_headers"
    }
  ],
  "repository": [
    {
      "repType": "git",
      "repUrl": "https://opensource.ncsa.illinois.edu/bitbucket/scm/iarp/extractors.git"
    },
    {
      "repType": "docker",
      "repUrl": "clowder/extractors-csvheaders"
    }
  ],
  "process": {
    "file": [
      "text/csv"
    ]
  },
  "external_services": [],
  "dependencies": [],
  "bibtex": []
}
```

Converter Info Example

```

{
  "name": "ImageMagick",
  "version": "1.0",
  "description": "Converts images from one format to another",
  "author": "Kenton McHenry <mchenry@illinois.edu>",
  "contributors": ["Bing Zhang <bing@illinois.edu>"],
  "repository": [
    {
      "repType": "git",
      "repUrl": "opensource.ncsa.illinois.edu/bitbucket/scm/pol/converters-imagemagick.git"
    },
    {
      "repType": "docker",
      "repUrl": "ncsapolyglot/converters-imagemagick"
    }
  ],
  "input_formats": [
    "png", "bmp"
  ],
  "output_formats": [
    "jpeg", "jpg", "pdf"
  ],
  "external_services": [],
  "dependencies": ["ImageMagick"],
  "bibtex": []
}

```

Current Tools Catalog Example

```

{
  "_id" : ObjectId("53864aee7400009900ec8d22"),
  "status" : 4,
  "creationDate" : ISODate("2014-05-28T20:45:34.412Z"),
  "updateDate" : ISODate("2016-11-21T04:09:47.836Z"),
  "downloads" : 2,
  "license" : "UIUC/NCSA Open Source License.\r\nhttp://otm.illinois.edu/uiuc_openSource",
  "whatsnew" : "Identify tumors in microscopy images.",
  "compatibility" : "",
  "url" : "https://opensource.ncsa.illinois.edu/bitbucket/projects/CATS/repos/extractors-cv/browse/cellprofiler/cellprofiler-tumor?raw",
  "sample_input" : "",
  "sample_output" : "",
  "interfaceLevel" : 2,
  "dockerimageName" : "",
  "vmImageName" : "",
  "dockerfile" : "",
  "author" : "ruiliu@illinois.edu",
  "dependencies" : [
    "Cell Profiler"
  ],
  "transformation_type" : "extractor",
  "transformation_id" : "",
  "title" : "Tumors"
}

```