Using AboutCode to Document Your Software Assets

AboutCode Defined

AboutCode is a tool for your software development team to document your code inside your codebase, typically in preparation for a product release, side-by-side with the actual code. AboutCode files have a simple, standard format that identifies components and their associated licenses. The current AboutCode tools (Python programs) are:

- **genabout.py**: Create AboutCode files from a Software Inventory file (.csv format) which is typically created from a software audit, and insert these AboutCode files into your codebase. You can regenerate the AboutCode files from a new Software Inventory file whenever you make changes.
- **genattrib.py**: Generate a Product Attribution notice document (HTML format) from your AboutCode files. You can also generate documents for other purposes (such as a License Reference) by varying your input control file and your .html template.
- **about.py**: Generate a Software Inventory list (.csv format) from your codebase based on your AboutCode files. Note that this Software Inventory will only include components that have AboutCode data. So if you do not create AboutCode files for your own original software components, these components will not show up in the generated inventory.

Additional AboutCode information is available at:

- http://www.aboutcode.org/ for an overview and a link to the ABOUT File specification.
- https://github.com/dejacode/about-code-tool for the AboutCode tools.

Key Terminology

Some key terminology that applies to AboutCode tool usage:

- **Software Inventory or Inventory** - means a list of all of the components in a Development codebase and the associated data about those components with a focus on software pedigree/provenance- related data for open source and third-party components.
- **Product BOM or BOM** - means a subset list of the components in a Development codebase (Software Inventory) that are Deployed on a particular Product Release (a Product Bill of Materials).

Using genabout.py to Generate AboutCode Files

Prepare Your Software Inventory for genabout.py Standard Column Names

You should start with a software inventory of your codebase in spreadsheet format. You need to prepare a version of it that will identify the column values that you want to appear in your .ABOUT files. Note the following standard column names (defined in the ABOUT File Specification), which genabout.py will use to look for the values that it will store in your generated .ABOUT files, as well as any additional text files that you identify, which it will copy and store next to the .ABOUT files.

Standard Column Name	Description	Notes
about_file	File or directory name. If this is a path name, use a "/" forward slash as path separators.	Mandatory. Tells the tool where to generate the AboutFiles. Note that genabout.py will use this to construct the "about_resource" field in the generated .ABOUT file, setting it to a "." if the about_file names a directory, otherwise using the file name.
name	Component name	Mandatory
version	Component version	Mandatory (optional in future releases)
spec_version	The version of the ABOUT file format specification used for this file.	Optional
description	Component description	Optional
description_file	Component description file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
download_url	Direct URL to download the original file or archive documented by this ABOUT file	Optional
home_url	URL to the homepage for this component	Optional

Using AboutCode to Document Your Software Assets

date	The date ('YYYY-MM-DD') when this ABOUT file was created or last validated.	Optional
readme	readme text	Optional
readme_file	readme text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
changelog	changelog text	Optional
changelog_file	changelog text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
news	news text	Optional
news_file	news text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
news_url	URL to a news feed for the component	Optional
notes	notes text	Optional
notes_file	notes text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
owner	name of the organization or person that owns or provides the component	Optional
contact	contact information for the component owner	Optional
author	author name(s)	Optional
author_file	author name(s) text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
copyright	copyright statement for the component	Optional
copyright_file	copyright statement text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.

notice_file	notice text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
license_text_file	license text file name	Optional. genabout will look for the file name (if a directory is specified in the --license_text_location option) to copy that file to the .ABOUT file target directory.
license_url	URL to the license text for the component	Optional
license_spdx	The SPDX license short form identifier for the license of this component. See http://spdx.org/licenses/ for details.	Optional
redistribute	Yes/No. Does the component license require source redistribution.	Optional
attribute	Yes/No. Does the component license require publishing an attribution or credit notice.	Optional
track_changes	Yes/No. Does the component license require tracking changes made to the component.	Optional
dje_component	A DejaCode Enterprise component URN or component name.	Optional
dje_license_key	A DejaCode Enterprise license URN, license name or license key for the component.	Optional. genabout will obtain license information from DejaCode Enterprise, including the license text, in order to create and write the appropriate .LICENSE file in the .ABOUT file target directory.
dje_owner	A DejaCode Enterprise owner URN for the component.	Optional
checksum_sha1	Checksum sha1 value for the file	Optional
checksum_sha1_file	File containing checksum data	Optional
checksum_md5	Checksum md5 value for the file	Optional
checksum_md5_file	File containing checksum data	Optional

Optionally Define Custom Fields for genabout.py with MAPPING.CONFIG

Optionally, you can control the generated label names and contents in your .ABOUT files using a MAPPING.CONFIG file. You can start with the default version provided at
https://github.com/dejacode/about-code-tool/blob/develop/about_code_tool/MAPPING.CONFIG and

you can customize a copy of that file to map the software provenance information that is important to you. When you are ready to run genabout.py, you will want to specify the --mapping option to tell it to look for the MAPPING.CONFIG file and use it.

You can customize your copy of MAPPING.CONFIG to recognize your own software inventory column names in order to map them to ABOUT File contents. This is especially useful if you prefer not to change some of the actual column names in your software inventory before running genabout.py. Note that the name on the right side (for example "Directory/Filename") is the name of the field in your software inventory spreadsheet, and the name on the left, followed by a colon, is the field label to go into the .ABOUT file. Here is an example:

```
# Essential Fields
about_file: Directory/Filename

# Mandatory Fields
name: Component
version: Confirmed Version

# Optional Fields
copyright: Confirmed Copyright

# Custom Fields
audit_ref_nbr: audit_ref_nbr
confirmed_license: Confirmed License
```

Run genabout.py to Generate AboutCode Files

When your software inventory is ready, you can save it as a .csv file, and use it as input to run genabout.py to generate your AboutCode files. The official genabout.py parameters are defined here:

● https://github.com/dejacode/about-code-tool/blob/develop/USAGE.rst

Here is an example of a genabout.py command:

```
python genabout.py --extract_license --api_url='{{your license library api}}'
--api_key='{{your license library api key}}'  --mapping
--license_text_location=/Users/harrypotter/myAboutFiles/
/Users/harrypotter/myAboutFiles/myProject-bom.csv /Users/harrypotter/myAboutFiles/
```

Note that this example genabout.py command does the following:
● Activates the --extract_license option to get license text.
● Activates the --mapping option to use a custom MAPPING.CONFIG file.

Using AboutCode to Document Your Software Assets

- Activates the --license_text_location option to get any text files that you have specified in your software inventory to be copied next to the associated .ABOUT files when those are created.
- Specifies the path of the software inventory to control the processing.
- Specifies a target output directory.

Note that the value for `{{your license library api}}` should end with /api/v2/licenses/ .

Review your generated AboutCode files to determine if they meet your requirements. Here is an example of an AboutCode file (`config.sub.ABOUT`) file that documents the component `config.sub` from the e2fsprogs project:

```
about_resource: config.sub
name: config.sub
version:

attribute: yes
category: Copyleft
checksum_sha1: c6aba2045763f997fc81c1d881304755b09d2e02
component_type: posix shell script text executable
confirmed_license: GPL 2.0 with Autoconf exception
copyright: Copyright (c) 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000,
2001, 2002, 2003, 2004, 2005 Free Software Foundation, Inc.
dje_license_key: gpl-2.0-autoconf
dje_license_name: GNU General Public License 2.0 with Autoconf exception
dje_license_url:
https://enterprise.dejacode.com/urn/?urn=urn:dje:license:gpl-2.0-autoconf
license_text_file: gpl-2.0-autoconf.LICENSE
owner: Theodore Ts'o
redistribute: yes
track_changes: yes
```

You can make the appropriate changes to your input software inventory and/or your MAPPING.CONFIG file and then run genabout.py as often as necessary to replace the generated AboutCode files with the improved output. (Note that you will want to delete or move your previously generated output before running genabout.py again.)

Using genattrib.py to Generate a Product Attribution Notice Package

Prepare a Filtered Product BOM to Use as Input to genattrib.py

The Software Inventory that you prepared for genabout.py most likely includes components that do not need to appear in a product attribution notice package; for example:

- Components in your codebase that are not Deployed on the final product (e.g. build tools, testing tools, internal documentation).
- Components in your codebase under licenses that do not require attribution (e.g. proprietary packages, commercial products).

You should prepare a filtered version of your software inventory (the one that you used for genabout.py) by removing the rows that identify components which should not be included in a product attribution notice package, and saving that filtered version as your Product BOM. You should also order the rows in this Product BOM in the sequence that you would like them to appear in the product attribution notice package.

Prepare an Attribution Template to Use as Input to genattrib.py

You can run genattrib.py using the default.html template provided with the AboutCode tools:

https://github.com/dejacode/about-code-tool/blob/develop/about_code_tool/templates/default.html

If you choose to do that, you will most likely want to edit the generated .html file to provide header information about your own organization and product.

Running genattrib.py with the default.html file is probably your best choice when you are still testing your AboutCode process. Once you have a good understanding of the generated output, you can customize the template to provide the standard text that you want to see whenever you generate product attribution for your organization. You can also create alternative versions of the template to use genattrib.py to generate other kinds of documents, such as a License Reference.

Use jinja2 Features to Customize Your Attribution Template

The genattrib.py tool makes use of the open source python library **jinja2** (http://jinja.pocoo.org/docs/dev/templates/) in order to extend .html capabilities and transform AboutCode input data into the final format of the generated attribution file. The **default.html** file contains text that complies with jinja2 syntax specifications in order to support grouping, ordering, formatting and presentation of your AboutCode data. If your attribution requirements are complex,

you may wish to study the jinja2 documentation to modify the default.html logic; alternatively, here are a few relatively simple concepts that relate to the attribution document domain.
The simplest modifications to the default.html file involve the labels and standard text. For example, here is the default template text for the Table of Contents:

```
<div class="oss-table-of-contents">
    {% for about_object in about_objects %}
        <p><a href="#component_{{ loop.index0 }}">{{ about_object.name }}
            {% if about_object.version %} {{ about_object.version }}
            {% endif %}</a></p>
    {% endfor %}
</div>
```

If you would prefer something other than a simple space between the component name and the component version, you can modify it to something like this:

```
<div class="oss-table-of-contents">
    {% for about_object in about_objects %}
        <p><a href="#component_{{ loop.index0 }}">{{ about_object.name }}
            {% if about_object.version %}  - Version  {{ about_object.version }}
            {% endif %}</a></p>
    {% endfor %}
</div>
```

The "`if about_object.version`" is checking for a component version, and if one exists it generates output text that is either a space followed by the actual version value, or, as in this customized template, it generates output text as " `- Version` ", followed by the actual version value. You will, of course, want to test your output to get exactly the results that you need.

Note that you can actually use genattrib.py to generate an AboutCode-sourced document of any kind for varying business purposes, and you may want to change the grouping/ordering of the data for different reporting purposes. (Here we get into somewhat more complex usage of jinja2 features, and you may wish to consult the jinja2 documentation to reach a more comprehensive understanding of the syntax and features.) The default ordering is by component, but In the following example, which is intended to support a "license reference" rather than an attribution document, the customized template modifies the data grouping to use a custom field called "confirmed license":

```
<div class="oss-table-of-contents">
        {% for group in about_objects | groupby('confirmed_license') %}
        <p>
                <a href="#group_{{ loop.index0 }}">{{ group.grouper }}
                </a>
        </p>
        {% endfor %}
</div>
```

After the table of contents, this example customized template continues with the license details using the jinja2 for-loop capabilities. Notice that the variable "group.grouper" is actually the license name here, and that "License URL" can be any URL that you have chosen to store in your .ABOUT files:

```
{% for group in about_objects | groupby('confirmed_license') %}
<div id="group_{{ loop.index0 }}">
<h3>{{ group.grouper }}</h3>
<p>This product contains the following open source software packages licensed under the terms of
the license: {{group.grouper}}</p>

<div class="oss-component" id="component_{{ loop.index0 }}">
      {%for about_object in group.list %}
      {% if loop.first %}
      {% if about_object.license_url %}
            <p>License URL: <a href="{{about_object.license_url}}
                ">{{about_object.license_url }}</a> </p>
            {% endif %}
      {% endif %}
      <li>{{ about_object.name }}{% if about_object.version %}  - Version
            {{ about_object.version }}{% endif %}</li>
            {% if about_object.copyright %}<pre>{{about_object.copyright}}</pre>{% endif %}
            {% if about_object.notice %}<pre>{{ about_object.notice }}</pre>
            {% elif about_object.notice_file %} <pre class="component-notice">
                {{ about_object.notice_text }}</pre>
            {% endif %}
            {% if loop.last %}
            <pre>
                {{about_object.license_text}}
            </pre>
            {% endif %}
      </div>
      {% endfor %}
      </div>
      {% endfor %}
      <hr>
</div>
<hr>
```

In summary, you can start with simple, cosmetic customizations to the default.html template, and gradually introduce a more complex structure to the genattrib.py output to meet varying business requirements.

Run genattrib.py to Generate a Product Attribution Notice Package

When your Product BOM (your filtered software inventory) is ready, you can save it as a .csv file, and use it as input to run genattrib.py to generate your product attribution notice package. Note that genattrib.py will use the "about_file" column in your software inventory to get all the fields that it needs from your previously generated AboutCode files. The official genattrib.py parameters are defined here:

- https://github.com/dejacode/about-code-tool/blob/develop/USAGE.rst

Here is an example of a genattrib.py command:

```
python genattrib.py
--template_location=/Users/harrypotter/myAboutFiles/my_attribution_template_v1.html
--mapping /Users/harrypotter/myAboutFiles/
/Users/harrypotter/myAboutFiles/myProject-attribution-document.html
/Users/dclark1330/cipher/myProject-attribution-input.csv
```

Note that this example genattrib.py command does the following:
- Activates the --template_location option to specify a custom output template.
- Activates the --mapping option to use a custom MAPPING.CONFIG file.
- Specifies the path of the AboutCode files needed to generate the output document.
- Specifies the full path (including file name) of the output document to be generated.
- Specifies the path of the filtered software inventory to control the processing.

A successful execution of genattrib.py will create a .html file that is ready to use to meet your attribution requirements.

Using about.py to Generate a Software Inventory

Generate a Software Inventory of Your Codebase from AboutCode Files

One of the major features of the ABOUT File specification is that the .ABOUT files are very simple text files that can be created, viewed and edited using any standard text editor. Your software development and maintenance processes may require or encourage your software developers to maintain .ABOUT files and/or associated text files manually. For example, when a developer addresses a software licensing issue with a component, it is appropriate to adjust the associated AboutCode files manually.

If your organization adopts the practice of manually creating and maintaining AboutCode files, you can easily re-create your software inventory from your codebase using about.py. The official about.py parameters are defined here:

- https://github.com/dejacode/about-code-tool/blob/develop/USAGE.rst

A successful execution of about.py will create a complete software inventory in .csv format.