GROBID manual

| 1. | Overview. | . 1 |
|----|--|-----|
| 2. | Build and set up environment for local deployment | . 1 |
| | Build and set up environment for remote deployment | |
| | Use of grobid-service. | |

1. Overview

The project grobid-service is a RESTful service implementation for accessing the grobid system. grobid-service is an open source project under the Apache License 2.0. It comes as a war file for deploying on a web container e.g. tomcat. The project also contains the libraries of grobid-core, doing the extraction work.

2. Build and set up environment for local deployment

To build grobid for local deployment, you just have to go to the root of the project and run the following command:

mvn clean install

Then deploy the generated war to the server. The artifact is in:

grobid-service/target/grobid-service-<version>.war

3. Build and set up environment for remote deployment

To build grobid for remote deployment, you have to go to the root of the project and run the following command:

mvn clean install -PgenericBuild

It will generate 2 artifacts, 1 in grobid-home, 1 in grobid-service:

grobid-home/target/grobid-home-<version>.zip
grobid-service/target/grobid-service-<version>.war

Copy these 2 artifacts to your remote server.

grobid-home-<version>.zip contains the needed native libraries, the models, lexicons and a config directory that contains 2 properties files grobid.properties and grobid_service.properties.

You have to unzip grobid-home wherever you want on your server.

unzip grobid-home-<version>.zip

In grobid-service-<version>.war, the file web.xml has 3 parameters to set before starting the server:

org.grobid.property: path to grobid.property

org.grobid.property.service: path to grobid service.properties

org.grobid.home: path to grobid home

These properties are filled by the following variables: _GROBID_PROPERTY, _GROBID_SERVICE_PROPERTY, and _GROBID_HOME so that it is possible to fill these values with a script given the environment. It is also possible to set manually these variables.

4. Use of grobid-service

Welcome page is available at http://server instance name/<root context name (i.e. for local tomcat http://localhost:8080/sname of the war deploy in webapp). From there you can access to about grobid (Fig4.1), process some conversion from the interface "Test Rest Interface" (Fig4.2) and access the administration parameters contained in grobid.properties and grobid_service.properties (Fig4.3):

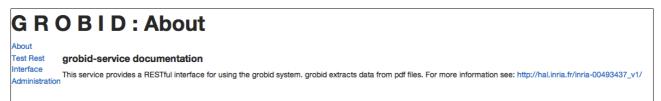


Fig 4.1: About



Fig 4.2: Test Rest Interface

| GROBID: Administration | | | | | | | |
|---------------------------------|------------------------------|--|--|--|--|--|--|
| About Test Rest Interface | Enter administrator password | Log in | | | | | |
| Administration | - | allel.execution updated with success | | | | | |
| | Property | value | | | | | |
| | org.grobid.service.admin.pw | 9d4e1e23bd5b727046a9e3b4b7db57bd8d6ee684 | | | | | |
| | ution true | | | | | | |
| | grobid.resources.inHome | true | | | | | |

Fig 4.3: Administration

The array below shows the provided resources corresponding to the HTTP verbs, to use the grobid-service. All url described bellow are relative path, the root url is <a href="http://<server instance">http://<server instance name>/<root context>.

| | URL | Parameter name | Requesting type | MIME Type | | |
|-----------------|--|----------------|-----------------|---------------------------------------|----------------------|--|
| Type of request | | | | Request input type | Response output type | Description |
| | /admin | _ sha1 | POST | application/x-www-form- urlencoded | text/html | Request to get parameters of grobid properties and grobid_service.properties formatted in html table. |
| | /admin?sha1= <pwd></pwd> | | GET | String | | |
| | /sha1 | - sha1 | POST | application/x-www-form- urlencoded | text/html | Request to get an input string hashed using sha1. |
| | /sha1?sha1= <input string=""/> | | GET | String | | |
| | /allProperties | sha1 | POST | application/x-www-form- urlencoded | text/xml | Request to get all properties key/value/type as xml. Sent xml follow the following schema: <pre></pre> |
| Administration | /allProperties?sha1= <password></password> | | GET | String | | |
| | /changePropertyValue | xml | POST | application/x-www-form- urlencoded | text/xml | Change the property value from the property key passed in the xml input. Xml input has to follow the following schema: <changeproperty></changeproperty> |
| | /changePropertyValue?xml= <some xml></some | | GET | String | | |

| General | /grobid | N/A | GET | N/A | text/html | Gives a very brief description about grobid. |
|---------------------------|--------------------------|--------------|-----------|---------------------------------------|------------------|---|
| Pdf to tei.xml conversion | /processHeaderDocument | Any name | POST, PUT | multipart/form-data | application/xml | Extract the header of the input document and convert it to tei.xml format. |
| | /processFulltextDocument | Any name | POST, PUT | multipart/form-data | application/xml | Convert all the input document to tei.xml format. |
| | /processDate | date | POST, PUT | application/x-www-form- urlencoded | application/text | Parse a raw date and return the corresponding normalized date. |
| Extract/normalize | /processHeaderNames | names | POST, PUT | application/x-www-form-urlencoded | application/text | Parse a raw sequence of names from a header section and return the corresponding normalized authors. |
| data | /processCitationNames | names | POST, PUT | application/x-www-form- urlencoded | application/text | Parse a raw sequence of names from a header section and return the corresponding normalized authors. |
| | /processAffiliations | affiliations | POST, PUT | application/x-www-form- urlencoded | application/text | Parse a raw sequence of affiliations and return the corresponding normalized affiliations with address. |
| | /processCitations | citations | POST, PUT | application/x-www-form- urlencoded | application/text | Parse a raw sequence and return the corresponding normalized citations. |

| The page "Test Rest Interface" allows you to test these requests quickly and easily. For technical look in the code, GrobidRestService class is the entry point for each rest service of Grobid. |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |