bormeparser Documentation

Versión 0.1.0

Pablo Castellano

01 de octubre de 2019

Índice general

Esta documentación describe el funcionamiento de **bormeparser**.

Esta documentación está disponible en http://bormeparser.readthedocs.org/. Si la estás leyendo desde otro sitio, es posible que no sea la última versión.

CAPÍTULO 1

Introducción

bormeparser es una librería de Python para parsear los archivos del BORME (Boletín Oficial del Registro Mercantil en España).

1.1 Qué es BORME

El **Boletín Oficial del Registro Mercantil** es un documento publicado diariamente por el Registro Mercantil Central (RMC) en España que contiene un listado de las últimas sociedades creadas y disueltas así como otros datos que las empresas están obligadas a comunicar.

La librería aprovecha que desde la aprobación de esta ley, desde el año 2009 el BORME se publica también en formato electrónico con la misma validez que su versión en papel.

Los BORMEs se publican en https://boe.es/diario_borme/.

Desgraciadamente debido al acuerdo actual con el Registro Mercantil, no pueden publicar todos los datos en un formato útil y reutilizable como XML o JSON y los datos más interesantes están solo disponibles en los archivos PDF.

Puedes leer más sobre ello en:

- Wikipedia: https://es.wikipedia.org/wiki/Boletín_Oficial_del_Registro_Mercantil

CAPÍTULO 2

Instalación

Puedes obtener bormeparser sencillamente desde PyPI:

```
pip install bormeparser
```

O si lo prefieres, puedes clonar el repositorio e instalarlo desde ahí, compilando previamente sus dependencias:

```
sudo apt-get install python3-dev libxslt1-dev
git clone https://github.com/PabloCastellano/bormeparser
cd bormeparser
python setup.py install
```

CAPÍTULO 3

Tutorial

3.1 Uso básico

bormeparser proporciona distintas funciones para tratar los archivos del BORME.

Empezamos con las funciones para generar las urls de descarga:

```
import bormeparser
date = (2015, 6, 2)
xml_url = bormeparser.get_url_xml(date)
pdf_url = bormeparser.get_url_pdf(date, bormeparser.SECCION.A, bormeparser.PROVINCIA.
↪MALAGA)
```

```
>>> print(xml_url)
https://www.boe.es/diario_borme/xml.php?id=BORME-S-20150602
>>> print(pdf_url)
https://boe.es/borme/dias/2015/06/02/pdfs/BORME-A-2015-102-29.pdf
```

Pero podemos usar otras funciones para descargar el BORME directamente de ese día:

```
import bormeparser

date = (2015, 6, 2)
path = '/tmp/BORME-A-2015-102-29.pdf'
downloaded = bormeparser.download_pdf(date, path, bormeparser.SECCION.A, bormeparser.
↪PROVINCIA.MALAGA)
```

```
>>> print(downloaded)
True
```

Para conocer la url de un PDF, bormeparser internamente descarga el archivo XML del día y ahí encuentra la ruta. Podemos obtener dicho archivo XML así:

```
>>> bormeparser.download_xml(date, '/tmp/20150602.xml')
True
```

Parsear un archivo PDF de BORME:

```
borme = bormeparser.parse('/tmp/BORME-A-2015-102-29.pdf', bormeparser.SECCION.A)
```

```
>>> print(borme)
<Borme(2015-06-02) seccion:SECCIÓN PRIMERA provincia:MÁLAGA>
```

3.2 Uso avanzado

Descargar y parsear un PDF de BORME:

```
borme = bormeparser.download_pdf(date, path, bormeparser.SECCION.A, bormeparser.
↪PROVINCIA.MALAGA, parse=True)
```

Si no ha habido ningún error (problema de conexión, el BORME de esa fecha no existe, . . .) la variable borme es una instancia de Borme:

```
>>> print(borme)
<Borme(2015-06-02) seccion:SECCIÓN PRIMERA provincia:MÁLAGA>
```

3.3 Borme y BormeActo

De la instancia BORME puedes obtener información básica como la fecha, la sección, la provincia. . .

```
>>> borme.cve
'BORME-A-2015-102-29'
>>> borme.num
102
>>> borme.info
{}
>>> borme.date
datetime.date(2015, 6, 2)
>>> borme.provincia
'MÁLAGA'
>>> borme.seccion
'SECCIÓN PRIMERA'
```

Y lo más importante: los anuncios mercantiles.

```
>>> for anuncio in borme.get_anuncios()[:10]:
...         print(anuncio)
...
<BormeAnuncio(223966) POLYESTER MALAGA SA (1)>
<BormeAnuncio(223967) RED MOUNTAIN PARK SL (3)>
<BormeAnuncio(223968) ISOFT SANIDAD SA (1)>
<BormeAnuncio(223969) RUILERENA SL (4)>
<BormeAnuncio(223970) REMOTONIO SL (4)>
<BormeAnuncio(223971) GARPAPACIA SL (4)>
```

(continué en la próxima página)

(proviene de la página anterior)

```
<BormeAnuncio(223972) GARIETOCIA SL (4)>
<BormeAnuncio(223973) PROAS INGENIERIA SL (2)>
<BormeAnuncio(223974) LORECUALAR SL (4)>
<BormeAnuncio(223975) CUALERENA SL (4)>
```

El segundo número entre paréntesis indica el número de actos mercantiles que contiene dicho anuncio.

Para analizar un anuncio mercantil en concreto, podemos obtenerlo de la instancia Borme a través de su id:

```
>>> anuncio = borme.get_anuncio(223969)
>>> anuncio.datos_registrales
'T 4889, L 3797, F 13, S 8, H MA109474, I/A 2 (21.05.15).'
>>> import pprint
>>> anuncio.get_actos()
<generator object get_actos at 0x7fed96cceb40>
>>> actos = list(anuncio.get_actos())
>>> pprint.pprint(actos)
[('Ceses/Dimisiones',
  {'Adm. Solid.': {'PASCUAL GARCIA LORENA', 'RUIZ GARRIDO JUAN ANTONIO'}}),
 ('Nombramientos', {'Liquidador': {'PASCUAL GARCIA LORENA'}}),
 ('Disolución', 'Voluntaria.'),
 ('Extinción', True)]
```

CAPÍTULO 4

Backends

Bormeparser soporta diferentes backends a la hora de parsear los archivos PDF.

4.1 Usar un backend específico

```
import bormeparser.backends.pypdf2

parser = bormeparser.backends.pypdf2.parser.PyPDF2Parser('examples/BORME-A-2015-27-10.
↪pdf')
borme = parser.parse()
```

4.2 Implementar un nuevo backend

Para implementar un nuevo backend, es necesario crear un nuevo paquete en el directorio bormeparser/backends/ con la siguiente estructura:

```
nuevoparser/
├── __init__.py
├── parser.py
```

__init__.py deberá estar vacío.

parser.py deberá contener una clase que herede de BormeAParserBackend e implemente el método *_parse()*:

```
from bormeparser.backends.base import BormeAParserBackend

class NuevoParser(BormeAParserBackend):
    def _parse(self):
        # Do your parsing here
        return DATA
```

_parse() debe retornar un diccionario de la siguiente forma:

```
{214028: {'Actos': {'Ceses/Dimisiones': [('Adm. Unico', {'JUAN GARCIA GARCIA'})],
                    'Datos registrales': 'T 5188, L 4095, F 146, S 8, H MA120039, I/A␣
↪4 (25.05.15).',
                    'Nombramientos': [('Adm. Unico', {'PEDRO GOMEZ GOMEZ'})]},
          'Empresa': 'EMPRESA RANDOM SL.'},
 214017: {'Actos': {'Datos registrales': 'T 2226, L 1139, F 102, S 8, H MA 33737, I/A␣
↪6 (25.05.15).',
                    'Modificaciones estatutarias': '8. Administración y␣
↪Representacion.-.'},
          'Empresa': 'EMPRESA ALEATORIA SL.'},
 'borme_cve': 'BORME-A-2015-102-29',
 'borme_fecha': 'Martes 2 de junio de 2015',
 'borme_num': 102,
 'borme_provincia': 'MÁLAGA',
 'borme_seccion': 'SECCIÓN PRIMERA'}
```

Es decir, debe contener los atributos *borme_fecha*, *borme_num*, *borme_provincia*, *borme_seccion* y todos los actos estructurados de la misma manera. Para más información consulta el código fuente de los parsers ya disponibles.

Por último añada el nuevo parser a *backends/__init__.py*:

```
from .parser1.parser import Parser1
from .pypdf2.parser import PyPDF2Parser
from .nuevoparser.parser import NuevoParser

__all__ = ['parser1', 'pypdf2', 'nuevoparser']
```

CAPÍTULO 5

Contribuir

Puedes mandar tus Pull Requests directamente a través de GitHub, donde también hay una lista de issues puedes ayudar a arreglar.

https://github.com/PabloCastellano/bormeparser/issues

CAPÍTULO 6

Changelog for bormeparser

6.1 0.4.1 (unreleased)

- Nothing changed yet.

6.2 0.4.0 (2019-09-18)

- Bump requirements
- Require python 3.5

6.3 0.3.5 (2019-02-18)

- Mejora en la detección del encoding de BORME-XML
- No fallar cuando la cabecera Content-Length no esté presente
- Pequeños cambios en los niveles de logging
- download: ajustados los valores por defecto de sleep y threads
- Borrados los checks de python2

6.4 0.3.4 (2018-09-24)

- Nuevo método BormeXML.get_cve_url()

6.5 0.3.3 (2018-09-23)

- is_company(): llama a clean_empresa() y comprueba si contiene la palabra «SOCIEDAD»

6.6 0.3.2 (2018-09-23)

- Actualización de las dependencias
- Tests arreglados
- Nuevo acto mercantil: Adaptación Ley 44/2015
- download_url() ahora reintenta la descarga si hubo un error
- BormeXML.save_to_file() crea el directorio si no existe
- clean_empresa(): quita «EN LIQUIDACION» y «SUCURSAL EN ESPAÑA» del nombre

6.7 0.3.1 (2018-05-29)

- Añadidos más tipos de sociedad
- Borme.from_json: permitir un objeto file como argumento filename

6.8 0.3.0 (2018-03-12)

- Eliminado soporte de Python 2
- Cambios en el formato BORME-JSON
- Nombres de actos repetidos en el mismo anuncio (issue #3)
- Usar requests en lugar de urllib
- Archivo de configuración ~/.bormecfg
- Mejoras en el parser
- Añadidos 4 nuevos actos y 41 cargos directivos
- Borme.to_json ahora permite especificar un path (archivo o directorio) en lugar de solo archivo
- Borme._set_url evita conexión a Internet si existe BORME-XML
- Sociedades y registros tienen su propio módulo
- Funciones de limpieza de datos en bormeparser.clean
- Incluye nombre del R.M. en BORME-JSON
- Cambios menores en los scripts
- Borme.XML devuelve str en lugar de list si solo hay un elemento
- BormeXML.get_provincias

6.9 0.2.4 (2016-09-21)

- BormeXML: get_url_pdfs, get_cves y get_sizes ahora permiten especificar sección y provincia
- Nueva constante ALL_PROVINCIAS en bormeparser.provincia
- Detección de nuevos tipos de sociedades
- Scripts: limpieza, uso de argparse en los scripts, unificación de parámetros
- Mejoras menores en la documentación
- Nuevos campos «version» y «raw_version» en el formato JSON de BORME

6.10 0.2.3 (2016-04-26)

- Mejora en el parser de BORME C

6.11 0.2.2 (2016-04-26)

- Mejoras en el parser de BORME C

6.12 0.2.1 (2016-04-25)

- Corregidos fallos de compatibilidad con Python 2

6.13 0.2 (2016-04-25)

- Eliminado primer argumento «date» de BormeXML.get_url_pdfs()
- Nuevo método: BormeXML.get_urls_cve()
- Arregladas algunas incompatibilidades con Python 2
- Nuevas funciones: get_borme_website(), get_url_seccion_c()
- Se incorpora parser para BORME C
- Añadido el acto «(Primera inscripcion O.M. 10/6/1.997)»
- Renombradas constantes y funciones
- BormeXML: BORME C
- script download_borme_pdfs_C.py
- Mejora parsing de cargos repetidos en el mismo acto (issue #4)

6.14 0.1.5 (2015-09-25)

- Añadidos nuevos cargos
- Mejoras en setuptools

6.15 0.1.4 (2015-09-24)

- Grandes mejoras en el parser en general
- Añadidos cargos y actos nuevos
- Mejoras en las expresiones regulares
- Los objetos Provincia ahora son comparables
- download_pdfs() ahora admite los parámetros seccion y provincia
- Nuevos scripts: borme_to_json, download_borme_pdfs_A, borme_sort, xml_poller
- Uso de OrderedDict en lugar de dict
- Uso de OrderedDict en lugar de dict
- Usar la librería logging
- Más tests
- Actualización de los requisitos

6.16 0.1.3 (2015-08-08)

- Fixed missing packages that weren't distributed

6.17 0.1.2 (2015-08-07)

- Fixed UnicodeWarning that caused tests to fail in Python 2

6.18 0.1.1 (2015-08-07)

- setup.py install now installs requirements

6.19 0.1 (2015-08-07)

- First release
- Download and parse BORME PDF files
- Main parser is PyPDF2
- Python 2 and 3 support
- Tests suite

CAPÍTULO 7

Índices y tablas

- genindex
- modindex
- search