# Generic compression, decompression, archive library for 7zip: **py7zr**

Hiroshi Miura
https://github.com/miurahr

# Popular compression  formats for data science

| Namep | Born | Algorithm and Strategy | Python |
|---|---|---|---|
| lzo | 2005 | Improved LZ77 | python-lzo (**) |
| quicklz | 2006 | Improved LZ77, speed | python-quicklz (**) |
| brotli | 2009 | Improved LZ77, Hafmann encode, 2nd context model | python-brotli(**) |
| lz4 | 2011 | Improved LZ77, speed | python-lz4 (**) |
| snappy | 2011 | Improved LZ77, speed | python-snappy(**) |
| Zstarndard | 2015 | Improved LZ77, speed、entropy encode | python-zstandard(**) |
| | | | ** Binding to C library |

# Popular data compression and archiving formats

| Name | Born | Compression Algorithm | Tools | Python |
|---|---|---|---|---|
| TAR | 1979 | None | GNU tar | tarfile |
| ZIP | 1989 | Deflate,(bzip2,LZMA,PPMd *) | PKZIP, WinZip | zipfile |
| GZIP | 1992 | Deflate | GNU gzip, zlib | gzip |
| xz | 1996 | LZMA, LZMA2 | XZ Utils, 7-zip | lzma |
| Bzip2 | 1996 | RLE,BWT,MTF,huffman code, delta | Bzip2, 7-zip, | bz2 |
| 7zip | 1999 | LZMA, LZMA2, Bzip2, PPMd, Deflate | 7-Zip p7zip | py7zr |

# Pure Python 7zip  library - py7zr

- Utilize lzma support on Python core (> python 3.3)
  - Python 3.7 Supports LZMA, LZMA2, BCJ, Delta
  - No support for BCJ2, PPMd compression algorithms.
- 7-zip compression and decompression with Pure python
  - Supports UNIX extensions for file permission as compatible with p7zip.
- Quality
  - CI/CD, coverage with azure-pipelines and travis-CI
  - Static type checks with mypy
  - Documentations

# Usage

```
$ pip install py7zr
$ py7zr l sample.7z
```

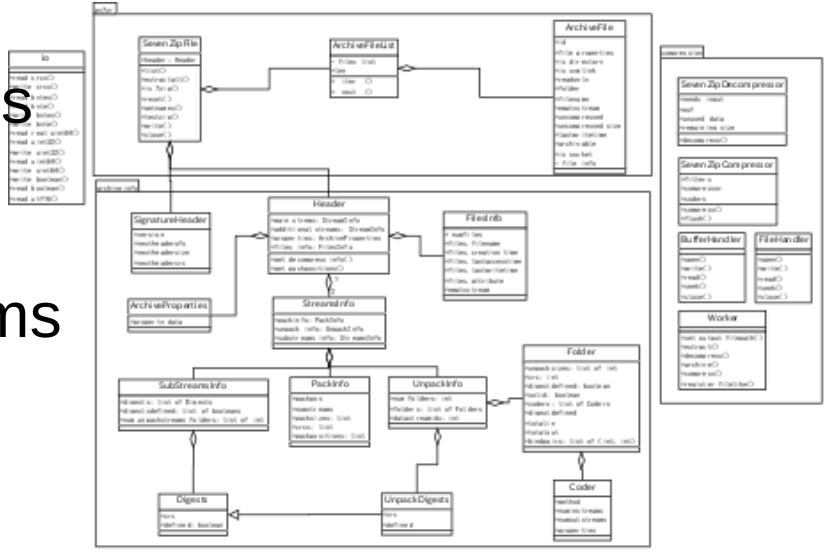
```
import py7zr

sf = py7zr.SevenZipFile("sample.7z", “r”)
sf.list()
sf.extractall(path=”tmp”)
sf.close()
```

# Inside py7zr:  Class design

a) **archiveinfo** package: hold classes to represent 7zip header structures.

b) **py7zr** package: provide compression/decompression APIs

**c)compressor** package: Implement compression algorithms

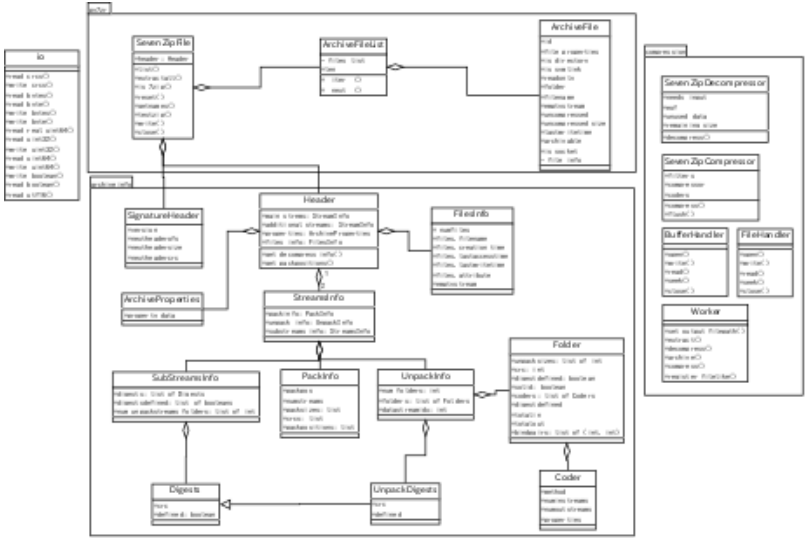


# Inside py7zr: design patterns

- Utilize **Observer** pattern

  - Minimum memory foot print for compression/decompression
  - Small number of file descriptor utilization.
  - Support selectable decompression from archive.

  - Not implement yet: progress display



# Inside py7zr: safety with type check

```
class ArchiveFile:
    def __init__(self, id: int, file_info: Dict[str, Any]) -> None:
        self.id = id
        self._file_info = file_info

    def file_properties(self) -> Dict[str, Any]:
        properties = self._file_info
        if properties is not None:
            properties['readonly'] = self.readonly
            properties['posix_mode'] = self.posix_mode
            properties['archivable'] = self.archivable
            properties['is_directory'] = self.is_directory
        return properties

    def _get_property(self, key: str) -> Any:
        try:
            return self._file_info[key]
```

# Inside py7zr

- Multi-threading compression/decompression for large scale archive file
  - lzma core library in Python core is not thread safe
  - Generate LZMADecompressor object for each threads.
- Unit tests and file extraction tests

# Copyright and license

- py7zr is distributed under GNU general public license 2.1 and later
- Copyrights
  - 2019 Hiroshi Miura
  - pylzma copyright(c) 2004-2015 by Joachim Bauch
  - 7-Zip copyright (c) 1999-2010 Igor Pavlov

# Active community

- Community development on github project

  https://github.com/miurahr/py7zr

- As usual, forks and pull requests are welcome.
- Decompression is now beta quality, compression is now alpha.
- Implementation of compression is under active development.