
AnyBlok / Pyramid Documentation

Release 0.6.2

Jean-Sébastien Suzanne

June 27, 2016

1	Front Matter	3
1.1	Project Homepage	3
1.2	Project Status	3
1.3	Installation	3
1.4	Unit Test	3
1.5	Dependencies	4
1.6	Contributing (hackers needed!)	4
1.7	Author	4
1.8	Contributors	4
1.9	Bugs	4
2	MEMENTO	7
2.1	Add route, view, ... in pyramid config	7
2.2	Get AnyBlok registry in view	8
2.3	Define view which are used only if one blok is installed	8
2.4	WorkingSet	9
2.5	Define the name of the database	9
2.6	Authentication and authorization	10
2.7	JSON adapter	11
3	AnyBlok / Pyramid framework	13
3.1	anyblok_pyramid.pyramid_config	13
3.2	anyblok_pyramid.scripts module	14
4	Helper for unittest	17
4.1	PyramidTestCase	17
4.2	PyramidDBTestCase	17
4.3	PyramidBlokTestCase	17
5	ROADMAP	19
5.1	To implement	19
5.2	Functionnality which need a sprint	19
6	CHANGELOG	21
6.1	0.6.2 (2016-06-20)	21
6.2	0.6.1 (2016-04-18)	21
6.3	0.6.0 (2016-04-18)	21
6.4	0.5.3 (2016-03-17)	22
6.5	0.5.2 (2016-01-15)	22

6.6	0.5.1 (2016-01-08)	22
6.7	0.5.0 (2016-01-07)	22
6.8	0.4.1 (2015-10-9)	22
6.9	0.4.0 (2015-08-25)	22
6.10	0.3.2 (2015-06-22)	23
6.11	0.3.1 (2015-05-04)	23
6.12	0.3.0 (2015-05-04)	23
6.13	0.2.0 (2015-03-15)	23
6.14	0.1.0 (2015-02-07)	23
7	Mozilla Public License Version 2.0	25
7.1	1. Definitions	25
7.2	2. License Grants and Conditions	27
7.3	3. Responsibilities	28
7.4	4. Inability to Comply Due to Statute or Regulation	29
7.5	5. Termination	29
7.6	6. Disclaimer of Warranty	29
7.7	7. Limitation of Liability	30
7.8	8. Litigation	30
7.9	9. Miscellaneous	30
7.10	10. Versions of the License	30
7.11	Exhibit A - Source Code Form License Notice	31
7.12	Exhibit B - “Incompatible With Secondary Licenses” Notice	31
8	Indices and tables	33
	Python Module Index	35

Contents:

Contents

- *Front Matter*
 - *Project Homepage*
 - *Project Status*
 - *Installation*
 - *Unit Test*
 - *Dependencies*
 - *Contributing (hackers needed!)*
 - *Author*
 - *Contributors*
 - *Bugs*

Front Matter

Information about the AnyBlok / Pyramid project.

1.1 Project Homepage

AnyBlok is hosted on [github](https://github.com/AnyBlok/AnyBlok_Pyramid) - the main project page is at https://github.com/AnyBlok/AnyBlok_Pyramid. Source code is tracked here using [GIT](#).

Releases and project status are available on Pypi at http://pypi.python.org/pypi/anyblok_pyramid.

The most recent published version of this documentation should be at <http://doc.anyblok-pyramid.anyblok.org>.

1.2 Project Status

AnyBlok with Pyramid is currently in beta status and is expected to be fairly stable. Users should take care to report bugs and missing features on an as-needed basis. It should be expected that the development version may be required for proper implementation of recently repaired issues in between releases; the latest master is always available at http://bitbucket.org/jssuzanne/anyblok_pyramid/get/default.tar.gz.

1.3 Installation

Install released versions of AnyBlok from the Python package index with [pip](#) or a similar tool:

```
pip install anyblok_pyramid
```

Installation via source distribution is via the `setup.py` script:

```
python setup.py install
```

Installation will add the `anyblok` commands to the environment.

1.4 Unit Test

Run the test with `nose`:

```
pip install nose
nosetests anyblok_pyramid/tests
```

1.5 Dependencies

AnyBlok works with **Python 3.3** and later. The install process will ensure that [AnyBlok](#), [Pyramid](#) are installed, in addition to other dependencies. The latest version of them is strongly recommended.

1.6 Contributing (hackers needed!)

Anyblok / Pyramid is at a very early stage, feel free to fork, talk with core dev, and spread the word!

1.7 Author

Jean-Sébastien Suzanne

1.8 Contributors

[Anybox](#) team:

- Georges Racinet
- Jean-Sébastien Suzanne
- Simon André
- Pierre Verkest

[Sensee](#) team:

- Franck Bret

1.9 Bugs

Bugs and feature enhancements to AnyBlok should be reported on the [Issue tracker](#).

Contents

- *MEMENTO*
 - *Add route, view, ... in pyramid config*
 - * *By includeme*
 - * *By blok*
 - *Get AnyBlok registry in view*
 - *Define view which are used only if one blok is installed*
 - *WorkingSet*
 - *Define the name of the database*
 - * *Define an AnyBlok init function*
 - * *Define the db name in the request path*
 - *Authentication and authorization*
 - *JSON adapter*

MEMENTO

Anyblok / Pyramid mainly depends on:

- Python 3.3+
- AnyBlok
- Pyramid

2.1 Add route, view, ... in pyramid config

2.1.1 By includeme

1. define the view in one file

in the file views.py:

```
from pyramid.view import view_config
from pyramid.response import Response

@view_config(route_name='hello')
def say_hello(request):
    return Response('Hello %(name)s !!!' % request.matchdict)
```

2. define the entrypoint function

in the file foo.py:

```
def update_pyramid_config(config):
    config.add_route('hello', '/hello/{name}')
    config.scan('.views')
```

2.1.2 By blok

1. define the view in one file of the blok

in the file views.py:

```
from pyramid.view import view_config
from pyramid.response import Response

@view_config(route_name='hello')
```

```
def say_hello(request):  
    return Response('Hello %(name)s !!!' % request.matchdict)
```

2. add the class method `pyramid_load_config`
in the file `foo.py`:

```
from anyblok.blok import Blok  
  
class MyBlok(Blok):  
    ...  
  
    @classmethod  
    def pyramid_load_config(cls, config):  
        config.add_route('hello', '/hello/{name}')        config.scan(cls.__module__ + '.views')
```

2.2 Get AnyBlok registry in view

By default the registry load is the registry of the Configuration `db_name` key.

Define a simple view:

```
from pyramid.view import view_config  
from pyramid.response import Response  
  
@view_config(route_name='foo')  
def bar(request):  
    registry = request.anyblok.registry  
    nb_installed_bloks = registry.System.Blok.query().filter_by(  
        state='installed').count()  
    return Response("Welcome in AnyBlok application, you have %d installed "  
                    "bloks" % nb_installed_bloks)
```

2.3 Define view which are used only if one blok is installed

See the link [view and route predicated](#)

the goal of the predicate is to get the access of the route or the view only if the predicate condition is validated.
AnyBlok / Pyramid add the predicate `installed_blok`:

```
from pyramid.view import view_config  
from pyramid.response import Response  
  
@view_config(route_name='foo')  
def bar1(request):  
    """ This is the default view """  
    return Response("Welcome in AnyBlok application, you have 0 installed "  
                    "bloks")  
  
@view_config(route_name='foo', installed_blok='anyblok-core')  
def bar2(request):
```

```

""" This view id call if the anyblok is installed """
registry = request.anyblok.registry
nb_installed_bloks = registry.System.Blok.query().filter_by(
    state='installed').count()
return Response("Welcome in AnyBlok application, you have %d installed "
                "bloks" % nb_installed_bloks)

```

Note: Installed predicated detect if the registry is load, without registry, the installed blok can no be verify.

Note: you can use the `current_blok` function to not write the blok name:

```

from anyblok_pyramid import current_blok

@view_config(route_name='foo', installed_blok=current_blok())
def bar2(request):
    """ This view id call if the anyblok is installed """
    registry = request.anyblok.registry
    nb_installed_bloks = registry.System.Blok.query().filter_by(
        state='installed').count()
    return Response("Welcome in AnyBlok application, you have %d installed "
                    "bloks" % nb_installed_bloks)

```

2.4 WorkingSet

Anyblok / Pyramid add two function to use callback:

- *set_callable*: save a callback, the name of the callable is the name of the callback
- *get_callable*: return a callback in function of this name

for example, see the callable *get_db_name*:

```
db_name = get_callable('get_db_name')(request)
```

2.5 Define the name of the database

The name of the database determine the registry use by the view.

By default the name of the database come from the Configuration `db_name` key. But it is possible to define a callback to define the good db name.

2.5.1 Define an AnyBlok init function

In the setup of the package add new entry point:

```

setup(
    ...
    entry_points={
        ...
        'anyblok.init': ['get_db_name=package.path:add_get_db_name'],
    }
)

```

```
    ...  
    },  
    ...  
)
```

In the file path of the package add the method `add_get_db_name`:

```
def add_get_db_name():  
    from anyblok_pyramid import set_callable  
  
    @set_callable  
    def get_db_name(request):  
        return ``My db Name``
```

2.5.2 Define the db name in the request path

This is an example to define the good db name in function of the path of the method.

This example work if the path id define like this:

```
config.add_route('one_route', '{dbname}/foo/bar')
```

The definition of `get_db_name` is:

```
def add_get_db_name():  
    from anyblok_pyramid import set_callable  
  
    @set_callable  
    def get_db_name(request):  
        return request.matchdict.get(  
            dbname',  
            Configuration.get('db_name'))
```

2.6 Authentication and authorization

Authentication can be add directly in configuration with `includeme`.

Links to the official documentation :

- <http://docs.pylonsproject.org/projects/pyramid/en/latest/tutorials/wiki2/design.html>
- <http://docs.pylonsproject.org/projects/pyramid/en/latest/tutorials/wiki2/authorization.html>
- <http://docs.pylonsproject.org/projects/pyramid/en/latest/tutorials/wiki2/authentication.html>
- http://docs.pylonsproject.org/projects/pyramid/en/latest/quick_tutorial/authorization.html
- http://docs.pylonsproject.org/projects/pyramid/en/latest/quick_tutorial/authentication.html

Link to an official tutorial If you want to replace default pyramid component by your own:

- <http://docs.pylonsproject.org/projects/pyramid/en/latest/narr/security.html#creating-your-own-authentication-policy>
- <http://docs.pylonsproject.org/projects/pyramid/en/latest/narr/security.html#creating-your-own-authorization-policy>

Add a root factory:

```
class RootFactory(object):

    def __init__(self, request):
        self.request = request

    __acl__ = [
        (Allow, Everyone, 'all'),
    ]
```

Add the authentication callback:

```
def group_finder(email, request):
    return ("all",)
```

Add the includeme callable:

```
def pyramid_security_config(config):
    # Authentication policy
    secret = Configuration.get("authn_key", "secret")
    authn_policy = AuthTktAuthenticationPolicy(secret=secret,
                                              callback=group_finder)
    config.set_authentication_policy(authn_policy)
    # Authorization policy
    authz_policy = ACLAuthorizationPolicy()
    config.set_authorization_policy(authz_policy)
    # Root factory: only added if set in config file (no default one)
    config.set_root_factory(RootFactory)
```

Add the includeme in the entry point:

```
setup(
    ...,
    entry_points={
        'anyblok_pyramid.includeme': [
            'pyramid_security_config=path:pyramid_security_config',
            ...
        ],
    },
    ...,
)
```

Note: You can get the session, with the callback `get_registry`:

```
from anyblok_pyramid import get_callable
# only if get_registry is implemented for you use case
registry = get_callable('get_registry')(request)
```

Note: You can merge the authorization of Pyramid and the authorization of AnyBlok

2.7 JSON adapter

In the case where you need to return json value you can format the data with:

- Define an adapter for the python type:

```
def datetime_adapter(obj, request):  
    return obj.isoformat()
```

- Add the adapter at the pyramid configuration:

```
def declare_json_data_adapter(config):  
    from pyramid.renderers import JSON  
    json_renderer = JSON()  
    json_renderer.add_adapter(datetime, datetime_adapter)  
    config.add_renderer('json', json_renderer)
```

- Add the includeme:

```
setup(  
    ...,  
    entry_points={  
        'anyblok_pyramid.includeme': [  
            'json_adapter=path:declare_json_data_adapter',  
            ...  
        ]  
    },  
    ...,  
)
```

Contents

- *AnyBlok / Pyramid framework*
 - *anyblok_pyramid.pyramid_config*
 - * *pyramid_config.settings*
 - * *pyramid_config.includeme*
 - *anyblok_pyramid.scripts module*

AnyBlok / Pyramid framework

3.1 anyblok_pyramid.pyramid_config

class anyblok_pyramid.pyramid_config.**Configurator** (*args, **kwargs)

Bases: pyramid.config.Configurator

Overwrite the Pyramid Configurator

default_setting()

Call all the entry point anyblok_pyramid.settings to update the argument setting

the callable need to have one parametter, it is a dict:

```
def settings_callable(setting):
    ...
```

We add the entry point by the setup file:

```
setup(
    ...,
    entry_points={
        'anyblok_pyramid.settings': [
            settings_callable=path:settings_callable,
            ...
        ],
    },
    ...,
)
```

include_from_entry_point()

Call all the entry point anyblok_pyramid.includeme to update the pyramid configuration

the callable need to have one parametter(the instance of Configurator class, self):

```
def config_callable(config):
    config.include(...)
```

We add the entry point by the setup file:

```
setup(
    ...,
    entry_points={
        'anyblok_pyramid.includeme': [
            config_callable=path:config_callable,
            ...
        ],
    },
)
```

```
        ],
    },
    ...,
)
```

load_config_bloks()

loop on each blok, keep the order of the blok to load the pyramid config. The blok must declare the meth
pyramid_load_config:

```
def pyramid_load_config(config):
    config.add_route('hello', '/hello/{name}/')
    ...
```

class anyblok_pyramid.pyramid_config.**AnyBlokRequest** (*request*)

Bases: object

Add anyblok properties in the request

```
request.anyblok
```

registry

Add the property registry

```
registry = request.anyblok.registry
```

Note: The db_name must be defined

class anyblok_pyramid.pyramid_config.**InstalledBlokPredicate** (*blok_name, config*)

Bases: object

Predicate installed_blok

3.1.1 pyramid_config.settings

anyblok_pyramid.pyramid_config.**pyramid_settings** (*settings*)

Add in settings the default value for pyramid configuration

Parameters **settings** – dict of the existing settings

3.1.2 pyramid_config.includeme

anyblok_pyramid.pyramid_config.**static_paths** (*config*)

Pyramid includeme, add the static path of the blok

Parameters **config** – Pyramid configurator instance

3.2 anyblok_pyramid.scripts module

anyblok_pyramid.scripts.**anyblok_wsgi** (*application, configuration_groups, **kwargs*)

Parameters

- **application** – name of the application
- **configuration_groups** – list configuration groupe to load

- ****kwargs** – ArgumentParser named arguments

Contents

- *Helper for unittest*
 - *PyramidTestCase*
 - *PyramidDBTestCase*
 - *PyramidBlokTestCase*

Helper for unittest

For unittest, classes are available to offer some fonctionnalités

4.1 PyramidTestCase

```
from anyblok_pyramid.tests.testcase import PyramidTestCase
```

```
class anyblok_pyramid.tests.testcase.PyramidTestCase
    Bases: object

    init_web_server()

    setUp()
```

4.2 PyramidDBTestCase

Warning: this testcase destroys the test database for each unittest

```
class anyblok_pyramid.tests.testcase.PyramidDBTestCase (methodName='runTest')
    Bases:
        anyblok_pyramid.tests.testcase.PyramidTestCase,
        anyblok.tests.testcase.DBTestCase

    init_registry (function, **kwargs)
```

4.3 PyramidBlokTestCase

```
class anyblok_pyramid.tests.testcase.PyramidBlokTestCase (methodName='runTest')
    Bases:
        anyblok_pyramid.tests.testcase.PyramidTestCase,
        anyblok.tests.testcase.BlokTestCase

    webserver
```

Contents

- *ROADMAP*
 - *To implement*
 - *Functionnnality which need a sprint*

ROADMAP

5.1 To implement

- WebSocket

5.2 Fonctionnalité which need a sprint

- Internalization

Contents

- *CHANGELOG*
 - 0.6.2 (2016-06-20)
 - 0.6.1 (2016-04-18)
 - 0.6.0 (2016-04-18)
 - 0.5.3 (2016-03-17)
 - 0.5.2 (2016-01-15)
 - 0.5.1 (2016-01-08)
 - 0.5.0 (2016-01-07)
 - 0.4.1 (2015-10-9)
 - 0.4.0 (2015-08-25)
 - 0.3.2 (2015-06-22)
 - 0.3.1 (2015-05-04)
 - 0.3.0 (2015-05-04)
 - 0.2.0 (2015-03-15)
 - 0.1.0 (2015-02-07)

CHANGELOG

6.1 0.6.2 (2016-06-20)

- [FIX] utf-8 encoding in setup, need for readthedocs
- [REF] move from bitbucket (mercurial) to github (git)

6.2 0.6.1 (2016-04-18)

- [FIX] for Python < 3.5

6.3 0.6.0 (2016-04-18)

Warning: This version break the compatibility with previous version. The goal is to use all the fonctionnality of pyramid, and give the tools to make the bind with AnyBlok easily

- [REM] remove old Controller declarations:

- Declarations.Pyramid
- Declarations.PyramidHTTP
- Declarations.PyramidJSONRPC
- Declarations.PyramidXMLRPC

- [ADD] add anyblok request property

```
registry = request.anyblok.registry
```

- [ADD] installed_blok predicate for route and view

```
@view_config(route_name='hello', installed_blok='my-blok')
def say_hello(request):
    ...
```

- [ADD] need_anyblok_registry predicate for route and view

```
@view_config(route_name='hello', need_anyblok_registry=True)
def say_hello(request):
    ...
```

6.4 0.5.3 (2016-03-17)

- [REF] Preload database, add log and check if the database exist before load it
- [FIX] catch simple exception to reput in real rpc exception

6.5 0.5.2 (2016-01-15)

- [FIX] use anyblok parser for config with gunicorn
- [REF] entry point init is now in anyblok

6.6 0.5.1 (2016-01-08)

- [REF] Adapt with the new version of AnyBlok
- [IMP] Add new entry point to load function before load AnyBlok bloks

6.7 0.5.0 (2016-01-07)

- [ADD] pyramid_pm and zope.sqlalchemy to isolate each controller call

6.8 0.4.1 (2015-10-9)

- [ADD] console script, implementation with gunicorn only
- [ADD] wsgi script to give un app for wsgi server

6.9 0.4.0 (2015-08-25)

Warning: this version can not be capable with the previous version

Note: Works only with AnyBlok 0.5.1 and after

- [REF] Add entry point to add new pyramid includeme and settings
- [DEL] properties decorator, it is useless because pyramid have a better behaviour
- [REF] add workingset to define overwritable callback used for application, no for the blok, add first callback, get_registry
- [REF] unit test cause of new version of AnyBlok 0.5.0
- [FIX] unit test case, update controller to unload the declaration when BlokManager are unloaded

6.10 0.3.2 (2015-06-22)

- [REF] cause of upgrade version of AnyBlok 0.4.0

6.11 0.3.1 (2015-05-04)

- [FIX] default value for beaker, None is better than ‘

6.12 0.3.0 (2015-05-04)

- [IMP] console script argsparse for pyramid and beaker
- [ADD] MANIFEST.in
- [FIX] script cause of remove logging configuration from AnyBlok

6.13 0.2.0 (2015-03-15)

- [ADD] configurator callable
- [REF] Adapt the import of python module of the blok, cause of the change in AnyBlok version 0.2.2

6.14 0.1.0 (2015-02-07)

Main version of AnyBlok / Pyramid. You can with this version

- Declare Views / Routes for application
- **Declare controller (Views / Routes) which depend of the installation of bloks**
 - XHR
 - JsonRPC
 - XmlRPC
- Possibility to check some property as authentication
- Possibility to define properties check

Contents

- *Mozilla Public License Version 2.0*
 - *1. Definitions*
 - * *1.1. “Contributor”*
 - * *1.2. “Contributor Version”*
 - * *1.3. “Contribution”*
 - * *1.4. “Covered Software”*
 - * *1.5. “Incompatible With Secondary Licenses”*
 - * *1.6. “Executable Form”*
 - * *1.7. “Larger Work”*
 - * *1.8. “License”*
 - * *1.9. “Licensable”*
 - * *1.10. “Modifications”*
 - * *1.11. “Patent Claims” of a Contributor*
 - * *1.12. “Secondary License”*
 - * *1.13. “Source Code Form”*
 - * *1.14. “You” (or “Your”)*
 - *2. License Grants and Conditions*
 - * *2.1. Grants*
 - * *2.2. Effective Date*
 - * *2.3. Limitations on Grant Scope*
 - * *2.4. Subsequent Licenses*
 - * *2.5. Representation*
 - * *2.6. Fair Use*
 - * *2.7. Conditions*
 - *3. Responsibilities*
 - * *3.1. Distribution of Source Form*
 - * *3.2. Distribution of Executable Form*
 - * *3.3. Distribution of a Larger Work*
 - * *3.4. Notices*
 - * *3.5. Application of Additional Terms*
 - *4. Inability to Comply Due to Statute or Regulation*
 - *5. Termination*
 - * *5.1.*
 - * *5.2.*
 - * *5.3.*
 - *6. Disclaimer of Warranty*
 - *7. Limitation of Liability*
 - *8. Litigation*
 - *9. Miscellaneous*
 - *10. Versions of the License*
 - * *10.1. New Versions*
 - * *10.2. Effect of New Versions*
 - * *10.3. Modified Versions*
 - * *10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses*
 - *Exhibit A - Source Code Form License Notice*
 - *Exhibit B - “Incompatible With Secondary Licenses” Notice*

Mozilla Public License Version 2.0

7.1 1. Definitions

7.1.1 1.1. “Contributor”

Means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

7.1.2 1.2. “Contributor Version”

Means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor’s Contribution.

7.1.3 1.3. “Contribution”

Means Covered Software of a particular Contributor.

7.1.4 1.4. “Covered Software”

Means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

7.1.5 1.5. “Incompatible With Secondary Licenses”

Means:

- **That the initial Contributor has attached the notice described in Exhibit B** to the Covered Software; or
- **That the Covered Software was made available under the terms of version 1.1** or earlier of the License, but not also under the terms of a Secondary License.

7.1.6 1.6. “Executable Form”

Means any form of the work other than Source Code Form.

7.1.7 1.7. “Larger Work”

Means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

7.1.8 1.8. “License”

Means this document.

7.1.9 1.9. “Licensable”

Means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

7.1.10 1.10. “Modifications”

Means any of the following:

- Any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or
- Any new file in Source Code Form that contains any Covered Software.

7.1.11 1.11. “Patent Claims” of a Contributor

Means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

7.1.12 1.12. “Secondary License”

Means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

7.1.13 1.13. “Source Code Form”

Means the form of the work preferred for making modifications.

7.1.14 1.14. “You” (or “Your”)

Means an individual or a legal entity exercising rights under this License. For legal entities, “You” includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

7.2 2. License Grants and Conditions

7.2.1 2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- **Under intellectual property rights (other than patent or trademark)** Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- **Under Patent Claims of such Contributor to make, use, sell, offer for sale,** have made, import, and otherwise transfer either its Contributions or its Contributor Version.

7.2.2 2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

7.2.3 2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- For any code that a Contributor has removed from Covered Software; or
- **For infringements caused by: (i) Your and any other third party's** modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- **Under Patent Claims infringed by Covered Software in the absence of its** Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

7.2.4 2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

7.2.5 2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

7.2.6 2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

7.2.7 2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

7.3 3. Responsibilities

7.3.1 3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

7.3.2 3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- **Such Covered Software must also be made available in Source Code Form, as** described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- **You may distribute such Executable Form under the terms of this License, or** sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

7.3.3 3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

7.3.4 3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

7.3.5 3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

7.4 4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

7.5 5. Termination

7.5.1 5.1.

The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

7.5.2 5.2.

If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

7.5.3 5.3.

In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

7.6 6. Disclaimer of Warranty

Warning: Covered Software is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the Covered Software is free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the Covered Software is with You. Should any Covered Software prove defective in any respect, You (not any Contributor) assume the cost of any necessary servicing, repair, or correction. This disclaimer of warranty constitutes an essential part of this License. No use of any Covered Software is authorized under this License except under this disclaimer.

7.7 7. Limitation of Liability

Warning: Under no circumstances and under no legal theory, whether tort (including negligence), contract, or otherwise, shall any Contributor, or anyone who distributes Covered Software as permitted above, be liable to You for any direct, indirect, special, incidental, or consequential damages of any character including, without limitation, damages for lost profits, loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses, even if such party shall have been informed of the possibility of such damages. This limitation of liability shall not apply to liability for death or personal injury resulting from such party's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

7.8 8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

7.9 9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

7.10 10. Versions of the License

7.10.1 10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

7.10.2 10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

7.10.3 10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

7.10.4 10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

7.11 Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

Note: You may add additional accurate notices of copyright ownership.

7.12 Exhibit B - “Incompatible With Secondary Licenses” Notice

This Source Code Form is “Incompatible With Secondary Licenses”, as defined by the Mozilla Public License, v. 2.0.

Indices and tables

- `genindex`
- `modindex`
- `search`

a

`anyblok_pyramid.pyramid_config`, [13](#)

`anyblok_pyramid.scripts`, [14](#)

`anyblok_pyramid.tests.testcase`, [17](#)

A

`anyblok_pyramid.pyramid_config` (module), [13](#)

`anyblok_pyramid.scripts` (module), [14](#)

`anyblok_pyramid.tests.testcase` (module), [17](#)