How to get the CLI

- User: Download OMERO.server and OMERO.py package
- Developer: Clone https://github.com/openmicroscopy/openmicroscopy
  → $ ./build.py build-py

Documentation:
https://docs.openmicroscopy.org/omero/5.4.0/users/cli/index.html

Plugins:

- Get the plugin python script and copy it into lib/python/omero/plugins or any .../omero/plugins directory on your PYTHONPATH
- Install with pip

Code organisation

Repository: https://github.com/openmicroscopy/openmicroscopy

Location: components/tools/OmeroPy

bin/omero[.bat] Launch script for

src/omero/cli.py Entry point, command line wrapper, runs

“Official” CLI core plugins:
src/omero/plugins/*.py

Tests for the “official” CLI core plugins:
test/integration/clitest/test_*.py Integration tests

Code organisation

[CLI] [plugin] [cmd] [args]

./omero admin restart

OmeroPy/bin/omero

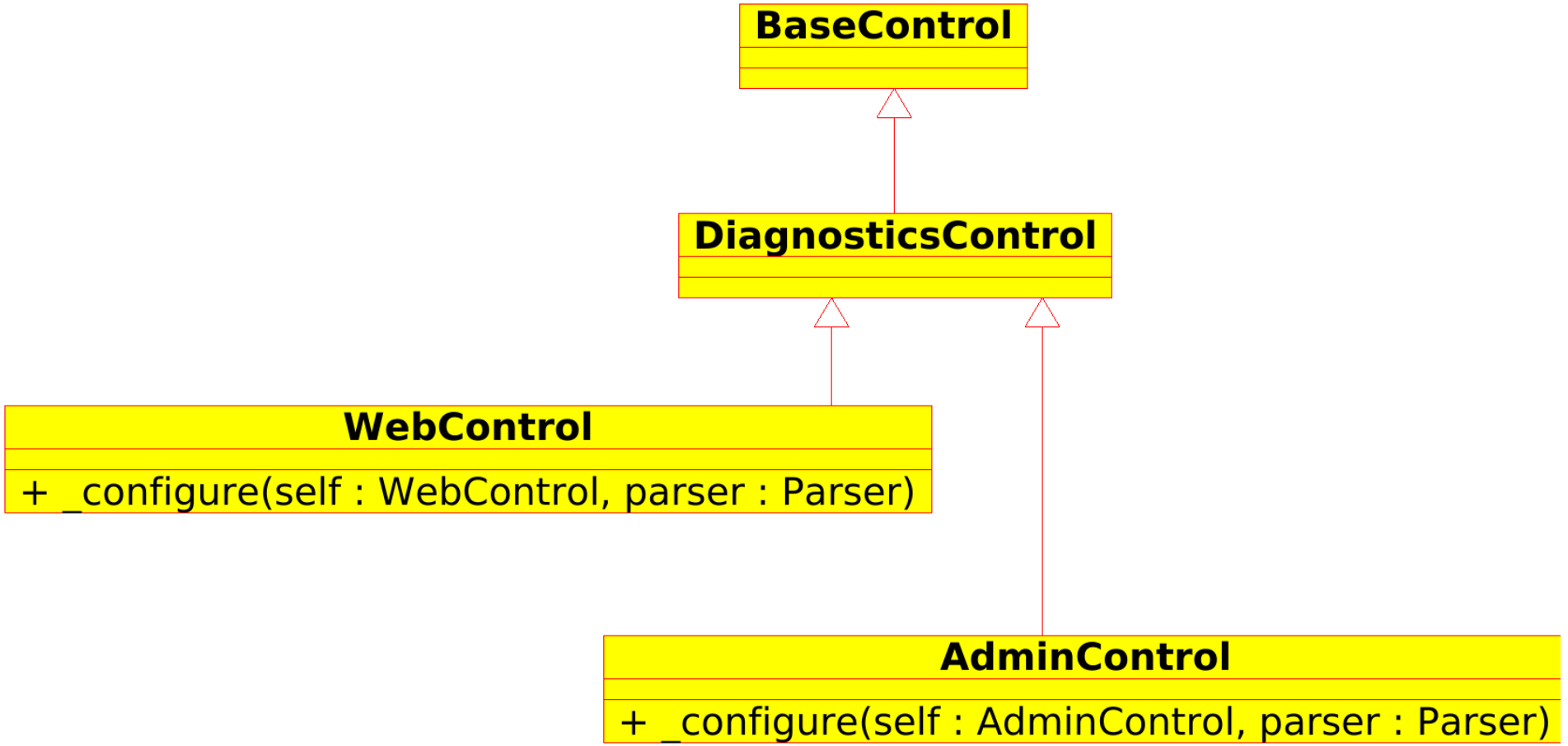
→ OmeroPy/src/cli.py

→ OmeroPy/src/omero/plugins/admin.py

→

```
@with_config
def restart(self, args, config):
    if not self.stop(args, config):
        self.ctx.die(54, "Failed to shutdown")
    self.wait_for_icedb(args, config)
    self.start(args, config)
```

Code organisation



How to write a plugin

Create a directory for plugin development and add it to PYTHONPATH:

```
$ mkdir -p ~/omero-cli-training/src/omero/plugins
(or $ git clone https://github.com/dominikl/omero-cli-training )
$ export PYTHONPATH=~/omero-cli-training/src:$PYTHONPATH
```

(“Official” core plugin: `components/tools/OmeroPy/src/omero/plugins`)

- Plugin class must inherit from BaseControl (or its children)
- Must implement method _configure
- Must call parser.set_defaults (typically within the _configure method)
- Must call the CLI.register method to register itself

How to write a plugin

Create a `xyz.py` in `~/omero-cli-training/src/omero/plugins`

```
import sys
from omero.cli import BaseControl, CLI

HELP = """What this plugin does"""

class XYZControl(BaseControl):
    """
    Some documentation
    """

    def _configure(self, parser):
        parser.add_argument(
            "some_argument",
            help="What this argument does")
        parser.set_defaults(func=self.__call__)

    def __call__(self, args):
        # Dispatch to certain methods depending on args
        self.some_method(args.some_argument)

    def some_method(self, some_argument):
        print("Do something with "+some_argument)

try:
    register("xyz", XYZControl, HELP)
except NameError:
    if __name__ == "__main__":
        cli = CLI()
        cli.register("xyz", XYZControl, HELP)
        cli.invoke(sys.argv[1:])
```

(https://github.com/dominikl/omero-cli-training/blob/master/skeleton.py)

→ PyCharm : https://www.jetbrains.com/pycharm/

- Create a new project for ~/omero-cli-training
- Add components/tools/OmeroPy directory to the project

Example: list.py - Simple 'list' command to list Projects, Datasets, Screens or Plates available to a logged in user.

How to write an integration test for the plugin

Create a directory for the integration test:

```
$ mkdir -p ~/omero-cli-training/test/integration/clitest
```

(“Official” core plugin test: `components/tools/OmeroPy/test/integration/clitest`)

- The integration test must inherit from CLITest
- Registers the plugin Control in setup_method
- Runs several usually parametrized tests

How to write an integration test for the plugin

Create `test_xyz.py to ~/omero-cli-training/test/integration/clitest/`

```
from xyz import XYZControl
from test.integration.clitest.cli import CLITest
import pytest

# some parameter values to test
test_arguments = ["argument0_value0", "argument0_value1", "argument0_value2"]


class XYZList(CLITest):

    def setup_method(self, method):
        super(XYZList, self).setup_method(method)
        self.cli.register("xyz", XYZControl, "TEST")
        self.args += ["xyz"]

    @pytest.mark.parametrize("test_arguments", arg)
    def test_xyz(self, capsys, arg):

        # assemble the arguments and invoke CLI
        self.args += ['%s' % object_type]
        self.cli.invoke(self.args, strict=True)

        # capture and check the output
        out, err = capsys.readouterr()
        assert out is not None
```

(https://github.com/dominikl/omero-cli-training/blob/master/test_skeleton.py)

How to run the integration test

You need a local server running, then you can run the test with the build.py script.

First add the `~/omero-cli-training/src/omero/plugins` directory itself to the PYTHONPATH:

```
$ export PYTHONPATH=~/omero-cli-training/src/omero/plugins:$PYTHONPATH
```

Then run the test with:

```
$ ./build.py -f components/tools/OmeroPy/build.xml test \
-DTEST=~/omero-cli-training/test/integration/clitest/[Your_Test].py
```

Integration tests for “official” core plugins located in components/tools/OmeroPy/test/integration/clitest are automatically run by the CI integration test job

“Publish”

“Official” core plugin: Open a PR against https://github.com/openmicroscopy/openmicroscopy

“Official” additional plugin: Create a repository under ome called omero-cli-xyz ; including a .travis.yml file to enable travis build checks

Make plugin pip installable: Create a setup.py, package plugin and upload to PyPI , see https://packaging.python.org/tutorials/distributing-packages/

As reference for a pip installable additional plugin see https://github.com/ome/omero-cli-render in particular https://github.com/ome/omero-cli-render/pull/1