

Introducing Middlefield – Custom generic developer tool

<https://github.com/elcaminoreal/>
<https://github.com/moshez/introducing-middlefield> –
reinventing plugins (gather), cli parsing (caparg) and
dependency injection (elcaminoreal)

Moshe Zadka – <https://cobordism.com>

Pyninsula January 2017

Build Pex with plugins to add commands

```
$ pip install middlefield
$ mf echo --what hello
Usage:
    self-build
$ mf self-build --output mf.pex \
  --package middlefield_echo \
  --shebang '/usr/bin/env python3'
$ ./mf.pex echo --what hello
hello
$ upload-to-internal-place mf.pex
```

Example plugin – middlefield echo (setup.py)

```
<snip>  
setuptools.setup(  
    entry_points={"gather":["gather=middlefield_echo"]}  
<snip>
```

Example plugin – middlefield echo (deps.py)

<snip>

```
@middlefield.COMMANDS.dependency()  
def echo_printer(_deps, _maybe_deps):  
    return print
```

Example plugin – middlefield echo (command.py)

<snip>

```
@middlefield.COMMANDS.command(
    parser=command( ' ',
        what=option( type=typing. List[ str ],
                      have_default=True)),
    dependencies=[ 'echo_printer ' ])
def echo( args , dependencies ):
    dependencies[ 'echo_printer ' ](* args[ 'what ' ])
```

What's next?

- ▶ Write a plugin (internal or public)
- ▶ Distribute internally
- ▶ Talk to me zadka.moshe@gmail.com <https://cobordism.com>