Introducing Middlefield

https://github.com/elcaminoreal/
https://github.com/moshez/introducing-middlefield

Moshe Zadka – https://cobordism.com

Pyninsula January 2017

Custom Generic Developer Tool

- Writing custom developer tools is fun!

Custom Generic Developer Tool

- ▶ Writing custom developer tools is fun!
- ▶ Too much repetition though

Why not reinvent everything?

- `gather` – plugin system
- `caparg` – argument parsing
- `elcaminoreal` – dependency injection
- `middlefield` – a multifunctional tool

Only built-in command

```
$ mf self-build --output mf.pex
```

Example plugin – middlefield echo

```
# setup.py
# ...
setuptools.setup(
# ...
    install_requires=['middlefield'],
    entry_points=dict(
        gather=["gather=middlefield_echo"],
    )
# ...
)
```

Example plugin – middlefield echo

```python
# middlefield_echo/_impl.py
# ... imports ...
@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'])
# ... continues ...
```

Example plugin – middlefield echo

```python
# ... continues ...
@middlefield.COMMANDS.dependency()
def echo_printer(_dependencies,
                 _maybe_dependencies):
    return print
```

Serving suggestion

```
$ mf self-build --package middlefield_echo \
                --output mf.pex
$ <upload mf.pex to your favorite internal place>
```

What's next?

- Write a plugin (internal or public)

What's next?

- Write a plugin (internal or public)
- Distribute internally

What's next?

- Write a plugin (internal or public)
- Distribute internally
- Talk to me

What's next?

- ▶ Write a plugin (internal or public)
- ▶ Distribute internally
- ▶ Talk to me zadka.moshe@gmail.com

What's next?

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