



vs.



微框架 vs. 微創業

I am Joe

我是 魯蛇 喬

Work at Startup

Use Bottle

Exp. 1.5y

Write Services

By **Bottle**



PiPoSea



No Users

微

失敗



Share Exp.

如何快速做出沒人要用的服務



Intro.

- Micro Web framework
- Single file (3000+)
- WSGI
- No dependencies other than STD
- Routing / Template / Utilities for http-related access
- Lack of ORM, session management support ...

Simple

is better than complex

Reinvent the Necessary wheel

Dynamic Routes

Static Files

Error pages

Redirect

Status code

Headers

Cookies

Templates

Plugins

Mount

request / response

And ...



Loves decorator

```
from bottle import route, run
```

```
@route('/hello')
```

```
def hello():
```

```
    return "Hello World!"
```

```
run(host='localhost', port=8080, debug=True)
```

get, post, put, delete, hook, error,
auth_basic ...



Bottle is a fast, simple and lightweight WSGI micro web-framework for Python.

Some Links

Issue Tracker
Development Blog
Bottle @ PyPI
Bottle @ GitHub
Bottle @ Google Groups
Bottle @ Twitter
Bottle @ Freenode (Chat)

Installation

Install Bottle with
`pip install bottle` or
download the source package
at PyPI.

Documentation

Download this documentation
as PDF or HTML (zip) for
offline use.

Sources

Warning: This is a preview for Bottle-0.12-dev, which is not released yet. Switch to the latest [stable release](#)?

Bottle: Python Web Framework

Bottle is a fast, simple and lightweight [WSGI](#) micro web-framework for [Python](#). It is distributed as a single file module and has no dependencies other than the [Python Standard Library](#).

- **Routing:** Requests to function-call mapping with support for clean and dynamic URLs.
- **Templates:** Fast and pythonic [built-in template engine](#) and support for [mako](#), [jinja2](#) and [cheetah](#) templates.
- **Utilities:** Convenient access to form data, file uploads, cookies, headers and other HTTP-related metadata.
- **Server:** Built-in HTTP development server and support for [paste](#), [fapws3](#), [bjoern](#), [Google App Engine](#), [cherrypy](#) or any other WSGI capable HTTP server.

Example: "Hello World" in a bottle

```
from bottle import route, run, template

@route('/hello/:name')
def index(name='World'):
    return template('<b>Hello {{name}}</b>!', name=name)

run(host='localhost', port=8080)
```

Visit official website !

Do what ?

像這樣的小玩具能幹嘛

Technology

Extensibility

ORM, Form Validation, Template engines,
session management ...

打造自己的工具箱

Performance

Under: Gevent, Tornado, GAE, Gunicorn ...

可處理高負載的服務

Writability

API , Restful, small app ...

而且學習成本超低 ...

S t a r t u p

Part of ...

Big service



Small service



PiPoSea



Rethink ...

Micro service

Do not customize it too much...

把最核心的服務早點寫出來，
給使用者玩看看才是王道...

Conclusion

Think "bottle" !!

微廣告 (工商服務)

- Taipei.py 5月份聚會

我人可能不會到...

微廣告 (工商服務)

乾脆來弄一個

Tainan.py

微廣告（工商服務）

或是弄一個

Tainan.p

微廣告（工商服務）

最後弄一個

Tainan.web

微廣告 (工商服務)

請與我聯絡

微感謝 (誠心誠意)

謝謝大家

