Flask

March 29, 2013

Flask

web developme
one drop at a ti

Section 1

Cos'è Flask

Microframework web

- Poche componenti di base.
- Facile da integrare con altri strumenti (es.WTForms).
- Basato su Werkzeug, Jinja 2 e buoni intenti.

Basato su Werkzeug

- Toolkit per Web Server Gateway Interface
- Fornisce le “solite cose” :)

Basato su Jinja 2

- Linguaggio di templating
- Unica scelta imposta: flask ne presuppone le funzionalità.
- ma se ne può sovrapporre un altro.

Basato su buoni intenti

- Approccio pythonico.
- Design chiaro.
- Ben documentato.
- pocoo.org, detto tutto :)

Quando usarlo

- Sempre :)
- Siti web (es. il sito di flask).
- Piattaforme:
- Frontend nel browser a programmi.

Caso d'uso: frontend

- Facile integrazione.
- Snello: poche dipendenze (jinja2 e werkzeug).
- Autocontenuto: con un solo utente non serve un webserver.
- Facile e veloce fare cose facili.

Section 2

Come funziona

Hello world

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

Section 3

Demo: app di esempio

Struttura

- test: package con i test della nostra app
- demo: package con la nostra app
- demo−ctrl: script per la gestione
- sample_data: alcuni dati di esempio

test

- Si possono usare i consueti strumenti per i test.
- Nel nostro caso: unittest .
- A disposizione il werkzeug.test . Client :

```
self.client = demo.app.test_client()
```

demo/__init__.py

```
from views import app
```

demo/config.py

- Flask può prendere la configurazione da un modulo python:

```
demo.app.config.from_object('demo.config')
```

- Ogni variabile in MAIUSCOLO viene importata:
  ::
  DATA_DIR = 'sample_data'
- ... e può essere letta nel "dizionarioide" app.config:
  ::
  app.config['DATA_DIR']

demo/views.py

- app = flask.Flask(__name__)
- Views definite facendo uso di un decoratore:

```
@app.route('/')
def show_root():
        return flask.render_template('root.html')
```

demo/templates

- Sistema di template: Jinja2.
- Autoescaping attivo di default.
- Ereditarietà dei template.
- Quant'altro :)

demo/static

- File statici.
- Ad esempio, usato in un template:

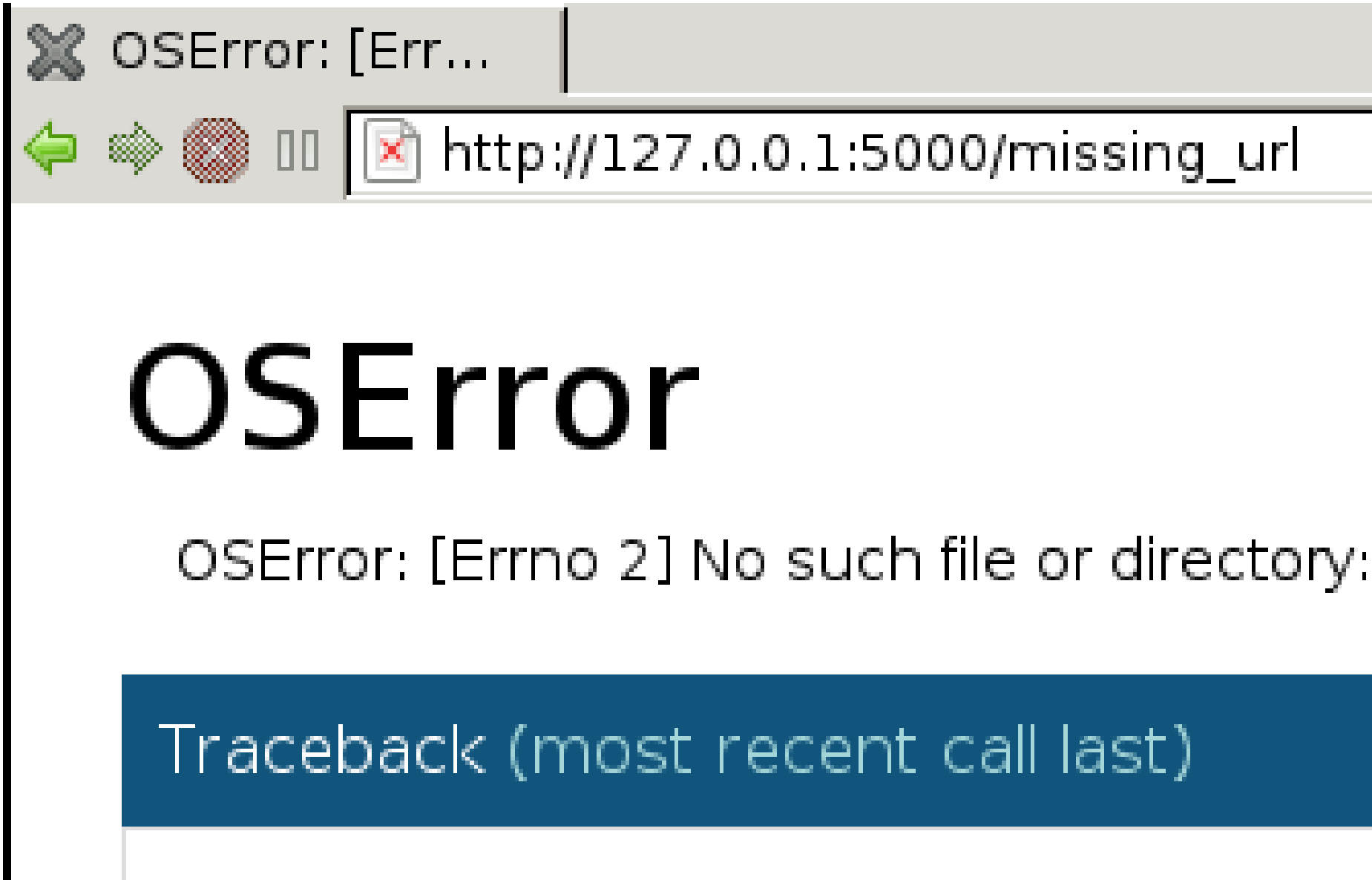
```
{{ url_for('static', filename='default.css') }}
```

demo−ctlr

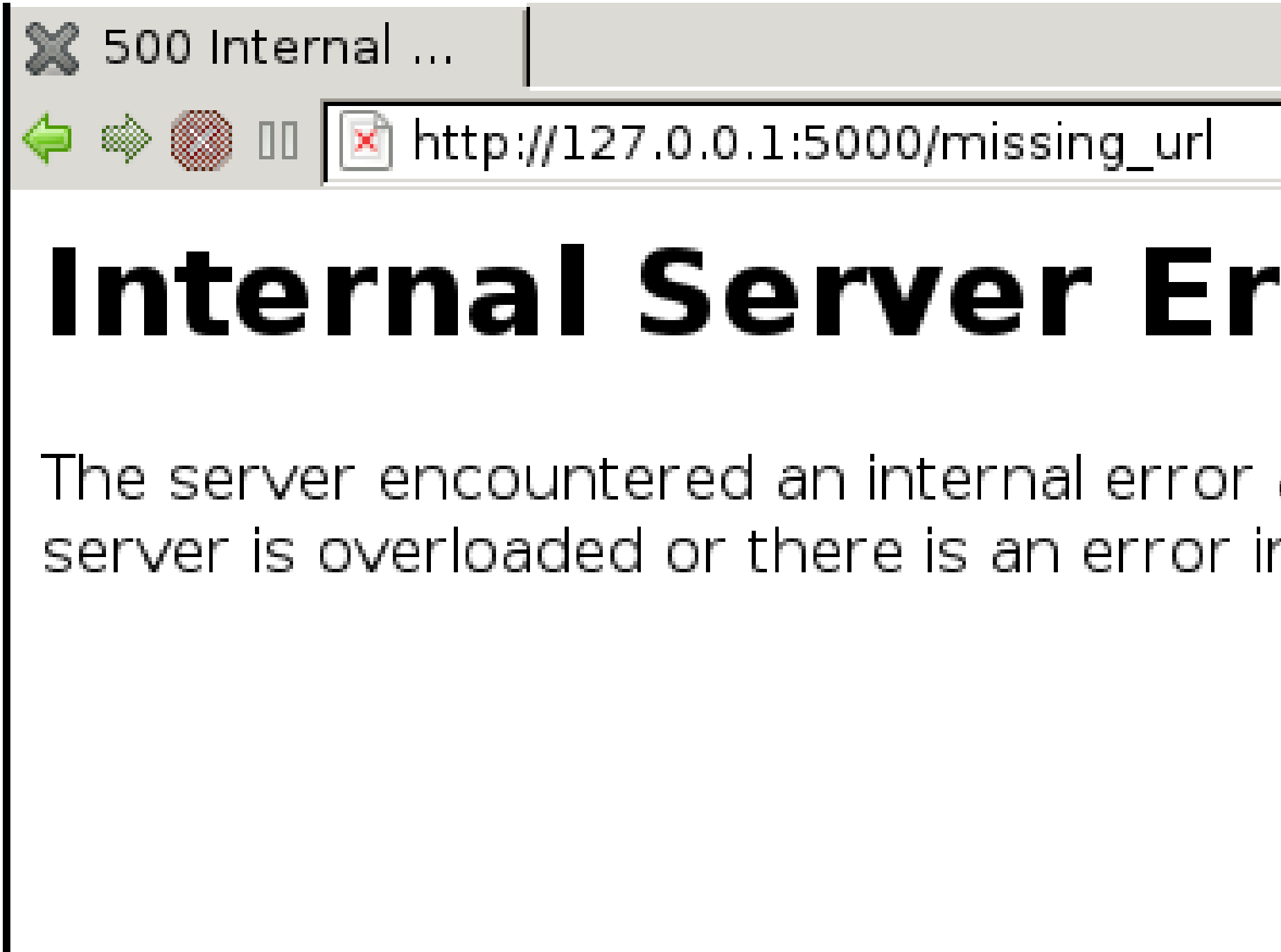
- Uso: demo−ctrl [−h] [−d] {start}
- Carica la configurazione.
- Lancia il server locale.

Modalità debug

"Brought to you by *DON'T PANIC*, your friendly Werkzeug powered traceback interpreter."



Modalità definitiva



Section 4

Per concludere

Altri esempi

- Microblog:
  https://github.com/mitsuhiko/flask/tree/master/examples/flaskr/
- Clone di Twitter:
  https://github.com/mitsuhiko/flask/tree/master/examples/minitwit
- Sito web di flask (pagine statiche + archivi di mailing list)
  https://github.com/mitsuhiko/flask/tree/website

Approfondimenti

- http://flask.pocoo.org/
- http://flask.pocoo.org/docs/
- http://jinja.pocoo.org/docs/
- http://werkzeug.pocoo.org/docs/
- http://flask.pocoo.org/community/poweredby/

Domande?