



ALESSANDRO COCCO

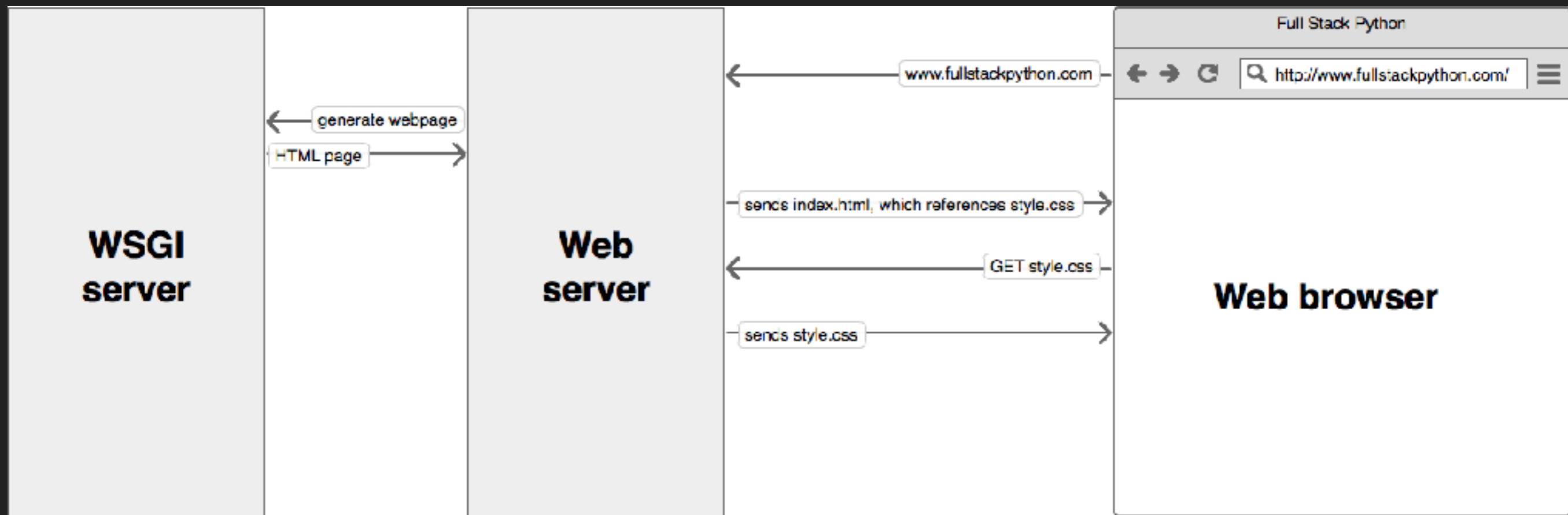
PYTHON E IL WEB

PERCHE' SCEGLIERE PYTHON

- ▶ Ha una bassa curva di apprendimento
- ▶ Ha moltissime librerie su cui fare affidamento
- ▶ E' attualmente uno dei linguaggi più usati
- ▶ E' molto versatile
- ▶ Fortemente orientato agli eventi asincroni
- ▶ Ha "pochi" framework di riferimento (PHP: laravel,codeigniter, symphony, cakephp,zend ...)

COME FACCIAMO COMUNICARE SERVER E APPLICAZIONE?

- ▶ Un server WEB tradizionale non sa come interagire o far girare un applicazione in python
- ▶ Attraverso un interfaccia di comunicazione
- ▶ Web Server Gateway Interface(WSGI)
- ▶ WSGI permette di mettere in comunicazione le richieste che arrivano al server e "girarle" all'applicazione(WSGI compliant), e ritornare la risposta al client.



COME DEV COSA DEVO SAPERE DI WSGI?

- ▶ Per cosa sta l'acronimo WSGI :)
- ▶ WSGI è un contenitore che gira in un processo separato e che gira in una porta differente dal vostro web server
- ▶ Che il vostro web server è configurato in modo tale da passare le richieste a WSGI che fa girare la vostra applicazione web, e che ritorna la risposta al richiedente.

QUALI SONO I WSGI DI RIFERIMENTO?

- ▶ Ovviamente ce ne sono parecchi.
- ▶ **uWSGI**:
 - ▶ Supporta pienamente applicazioni WSGI
 - ▶ è scritto in C
 - ▶ ha un veloce ciclo di release ed è vastamente supportato

QUALI SONO I WSGI DI RIFERIMENTO?

▶ GUNICORN:

- ▶ Supporta pienamente applicazioni WSGI
- ▶ è basato su un modello pre-fork
- ▶ per questo offre diversi worker e tipi di configurazioni, gestione automatica dei workers ecc ...
- ▶ Architettura semplice e chiara e supporta grossomodo tutte le versioni di python

QUALI SONO I WSGI DI RIFERIMENTO?

- ▶ Ce ne sono molti altri fra cui scegliere e testare:
- ▶ Tornado, Twisted WEB, Waitress , mod_wsgi (apache mod) ecc

I FRAMEWORK DI RIFERIMENTO

- ▶ Ci sono svariati framework di riferimento, di cui però solo alcuni spiccano in maniera consistente, e si suddividono in:
 - ▶ **Full stack frameworks** (ha tutto il necessario per poter costruire la nostra applicazione)
 - ▶ **Non Full Stack framework** (offre solo una parte degli strumenti e lavora con "altri strumenti" scelti da chi sviluppa il codice, motore template ecc...)

ELENCHIAMONE ALCUNI

- ▶ Full stack frameworks
 - ▶ DJANGO
 - ▶ WEB2PY
- ▶ Non Full Stack framework
 - ▶ FLASK
 - ▶ HUG
 - ▶ PYRAMID



DJANGO

- ▶ E' probabilmente il web framework più conosciuto e diffuso al momento
- ▶ Fornisce di base moltissime utility "out of the box"
- ▶ Ha una comunità vastissima che mette a disposizione moduli/customizzazioni già pronte
- ▶ Ci sono conferenze annuali in EU e USA
- ▶ Probabilmente oggi la maggiorparte delle applicazioni web in python sono scritte usando Django



DJANGO COSA INCLUDE

- ▶ MVC
- ▶ Astrazione alle chiamate del DB
- ▶ Possibilità di installare plugin
- ▶ Gestione di moduli e permessi e utenze
- ▶ View generiche per l'uso di casi comuni evitando "riscritture"
- ▶ sistema di template basato su `{% tags %}`
- ▶ Localizzazione
- ▶ e molto altro



DJANGO INFO, DOC

- ▶ Installazione molto semplice tramite PIP o sistema a pkg
- ▶ sito <https://www.djangoproject.com/>
- ▶ tantissima documentazione a disposizione
- ▶ <https://docs.djangoproject.com/en/1.10/>

FLASK

- ▶ E' un microframework (ma questo non vuol dire che nn sia potente)
- ▶ E' molto duttile proprio perché lascia libera scelta nella gestione e uso di altri framework con cui si può abbinare
- ▶ Piuttosto che sviluppare tutto si concentra sulle cose fondamentali, routing, requests, response objects e template.
- ▶ A volte è meglio usare "meno" se nn serve.



Flask

web development,
one drop at a time

FLASK

- ▶ Essendo minimale può facilmente crescere con la vostra app.
- ▶ SQLAlchemy (è una abbinata classica)
- ▶ Celery ecc ...
- ▶ E' vastamente supportato, release frequenti
- ▶ Ottima doc
- ▶ <http://flask.pocoo.org/>



CHERRYPY

- ▶ Un ottima alternativa a Flask
- ▶ ha una struttura di plugin molto flessibile
- ▶ ha incluse diverse cose come authentication, sessions,
- ▶ Ordinato e robusto
- ▶ <http://cherrypy.org/>
- ▶ <http://docs.cherrypy.org/en/latest/>