An HTTP request's journey through a platform-as-a-service

Giles Thomas, EuroPython 2014, Berlin
Introducing Web Platform Docs

Tim Berners-Lee
Web Developer
Welcome

Hello World

How did you get here?
1. You are successfully running web2py
2. You visited the url /welcome/default/index
3. Which called the function index() located in the file web2py/applications/welcome/controllers/default.py
4. The output of the file is a dictionary that was rendered by the view web2py/applications/welcome/views/default/index.html
5. You can modify this application and adapt it to your needs

Copyright © 2014
Powered by web2py
HSK东西 Scripts

Tools

- Analyse Your 汉字 Vocabulary/Text
- Fast Simple Chinese-English Dictionary
- Advanced Hanzi Search
- Radicals List
- Homophones (same pinyin)

Static Pages

- Mandarin Companion Vocabulary Analysis
- 'New' HSK Lists
  - HSK Words 2012/2013
  - HSK Characters 2012/2013
  - HSK Words 2010 (outdated)
  - HSK Characters 2010 (outdated)
- HSK Comparisons
  - Where the HSK 2010 words are in 2013
  - Where the HSK 2010 characters are in 2013

Page generated in 0.000003 seconds
Obey the Testing Goat!

TDD for the Web, with Python, Selenium, Django, JavaScript and pals...

The book

This is my book about Test-Driven-Development for web programming, published by the excellent O'Reilly Media.

There are 2 ways you can read the book:

- **Buy it now via O'Reilly**
  - Obviously this is my favourite option! O'Reilly are great, they have DRM-free versions you can download in all sorts of formats (MOBI/epub/pdf), and of course it's the only way to get a hold of an actual physical copy of the book. You can also read it at Safari if you have an account there.

- **Read it online for free**
  - Via O'Reilly Chimera. I see this as a "try-before-you-buy" scheme, and I hope that if you enjoy it you'll buy a copy -- if not for yourself, then perhaps for a friend!

And do get in touch with comments, suggestions, corrections etc! obeythetestinggoat@gmail.com

Blog

Kent Beck on the limits of TDD

Read the book

I'm writing a book all about TDD and Web programming. Read the draft and let me know what you think!

Reviews & Testimonials

"Hands down the best teaching book I've ever read" — "Even the first 4 chapters were worth the money" — "Oh my gosh! This book is outstanding" — "The testing goat is my new friend" — Read more..

Resources

A selection of links and videos about TDD, not necessarily all mine. eg this tutorial at PyCon 2013, how to motivate coworkers to write unit tests, thoughts on Django's test tools, London-style TDD and more.

Old TDD/Django Tutorial

This is my old TDD tutorial.
PICK YOUR DRUG
WE'LL MAKE YOUR PLAYLIST

PLAYLIST TYPE:
USER PLAYLISTS OFFICIAL PLAYLISTS
>>> the stack
>>> http request, part 1

Load balancer (nginx/openresty)

Backend server 1 (nginx/uwsgi)
  - Web app 1
  - Web app 2

Backend server 2 (nginx/uwsgi)
  - Web app 3
  - Web app 4

Backend server 3 (nginx/uwsgi)
  - Web app 5
  - Web app 6
>>> http request, part 3

Load balancer (nginx/openresty)

Backend server 1 (nginx/uwsgi)
  - Web app 1
  - Web app 2

Backend server 2 (nginx/uwsgi)
  - Web app 3
  - Web app 4

Backend server 3 (nginx/uwsgi)
  - Web app 5
  - Web app 6
http {
    # ... 
    init_by_lua_file /etc/nginx/init_backends.lua; 
    # ...

    server {
        listen 80;
        listen 443 ssl;
        # ...
        location / {
            set $route_host $http_host;
            set $backend_ip '';
            access_by_lua_file /xxx/get_backend_ip.lua;
            proxy_pass $backend_ip;

            #...
        }
    }
}
canonicalHost = string.lower(ngx.var.route_host)

local hash = 5381
local limit = (2 ^ 32)
for i = 1, #canonicalHost do
  local c = canonicalHost:sub(i, i)
  hash = ((hash * 33) % limit) + string.byte(c)
end

serverNo = (hash % #web_server_ips) + 1

ngx.var.web_server_ip = ngx.var.scheme.."://"..web_server_ips[serverNo]
>>> http request, part 2

Load balancer
(nginx/openresty)

Backend server 1
(nginx/uwsgi)

- Web app 1
- Web app 2

Backend server 2
(nginx/uwsgi)

- Web app 3
- Web app 4

Backend server 3
(nginx/uwsgi)

- Web app 5
- Web app 6
>>> the backends, nginx

```nginx
server {
    listen  80;
    listen  443 ssl;
    server_name ~(?<domain>.+)$;

    location / {
        include uwsgi_params;
        uwsgi_pass unix:/var/sockets/$domain/socket;
    }
}
```
the backends, uWSGI

[uwsgi]
# The socket that nginx is talking to
socket = /var/sockets/www.obeythetestinggoat.com/socket

# Run the web app
file = /path/to/obeythetestinggoat.com/source/application.py

# shutdown app (but not master) after 26hrs of no hits
idle=93600
server {
    listen 80;
    listen 443 ssl;
    server_name ~(?<domain>.+)$;

    location / {
        error_page 502 = @fallback;
        include uwsgi_params;
        uwsgi_pass unix:/var/sockets/$domain/socket;
    }

    location @fallback {
        if (-f /etc/uwsgi/vassals/$domain.ini) {
            rewrite ^ /xxx/502-backend.html;
        }
        proxy_pass https://localhost/initialize_webapp/$scheme/$domain/$uri?$query_string;
    }
}
>>> scaling

Load balancer (nginx)

Backend server 1
- Web app 1
- Web app 3

Backend server 2
- Web app 5
- Web app 6

Backend server 3
- Web app 7
- Web app 2

Backend server 4
- Web app 5
- Web app 4
>>> resilience

Load balancer (nginx)

Backend server 1 (nginx/uwsgi)
  - Web app 1
  - Web app 2

Backend server 2 (nginx/uwsgi)
  - Web app 1
  - Web app 3
  - Web app 4

Backend server 3 (nginx/uwsgi)
  - Web app 2
  - Web app 5
  - Web app 6
Questions?

Giles Thomas @gpjt
Glenn Jones @millenniumhand
Harry Percival @hjwp