

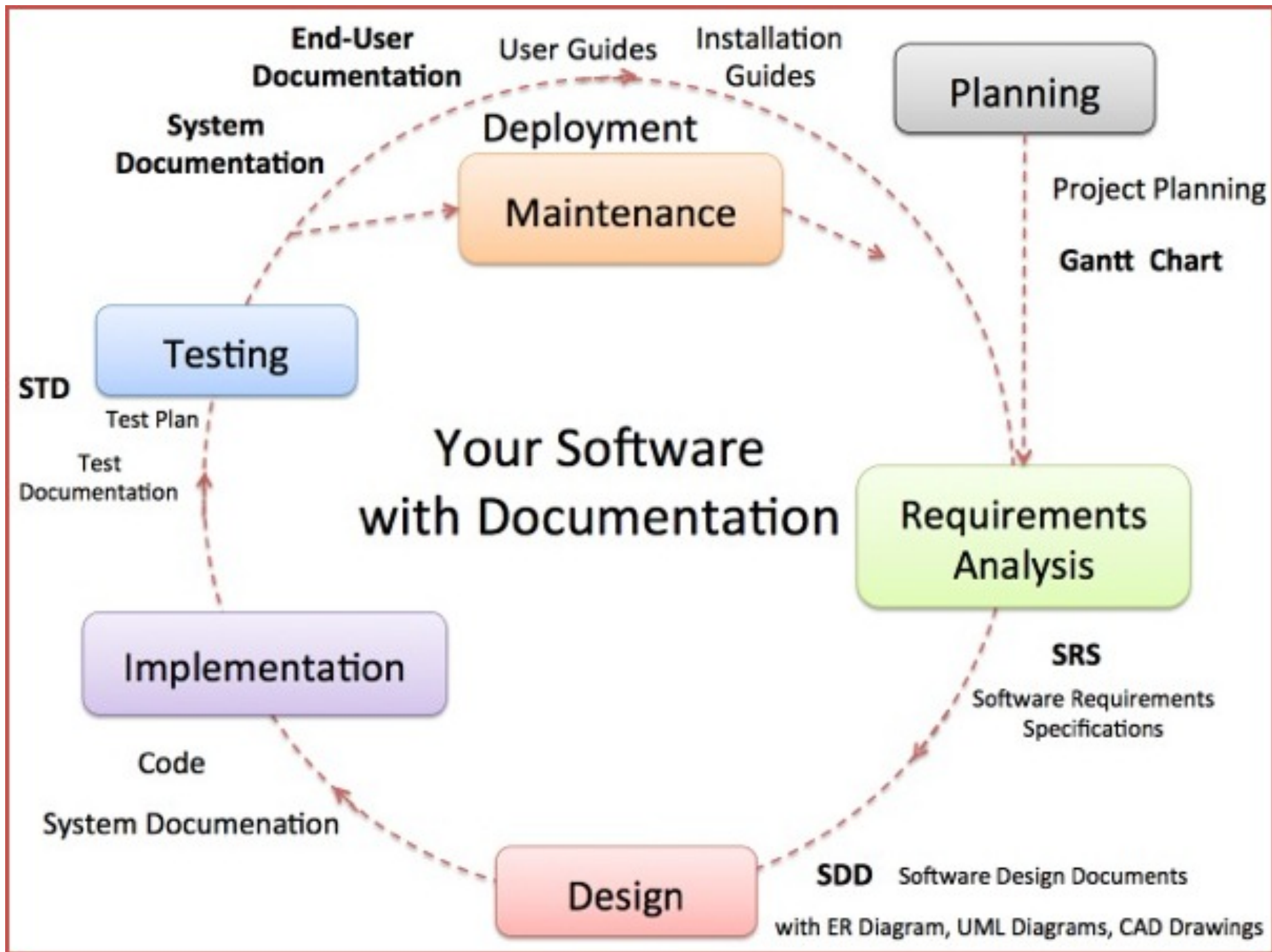
A Byte of Software Deployment

@dylanninin

@2016-03-07

What is Software Deployment ?

Software Development Life Cycle



Definition

It refers to:

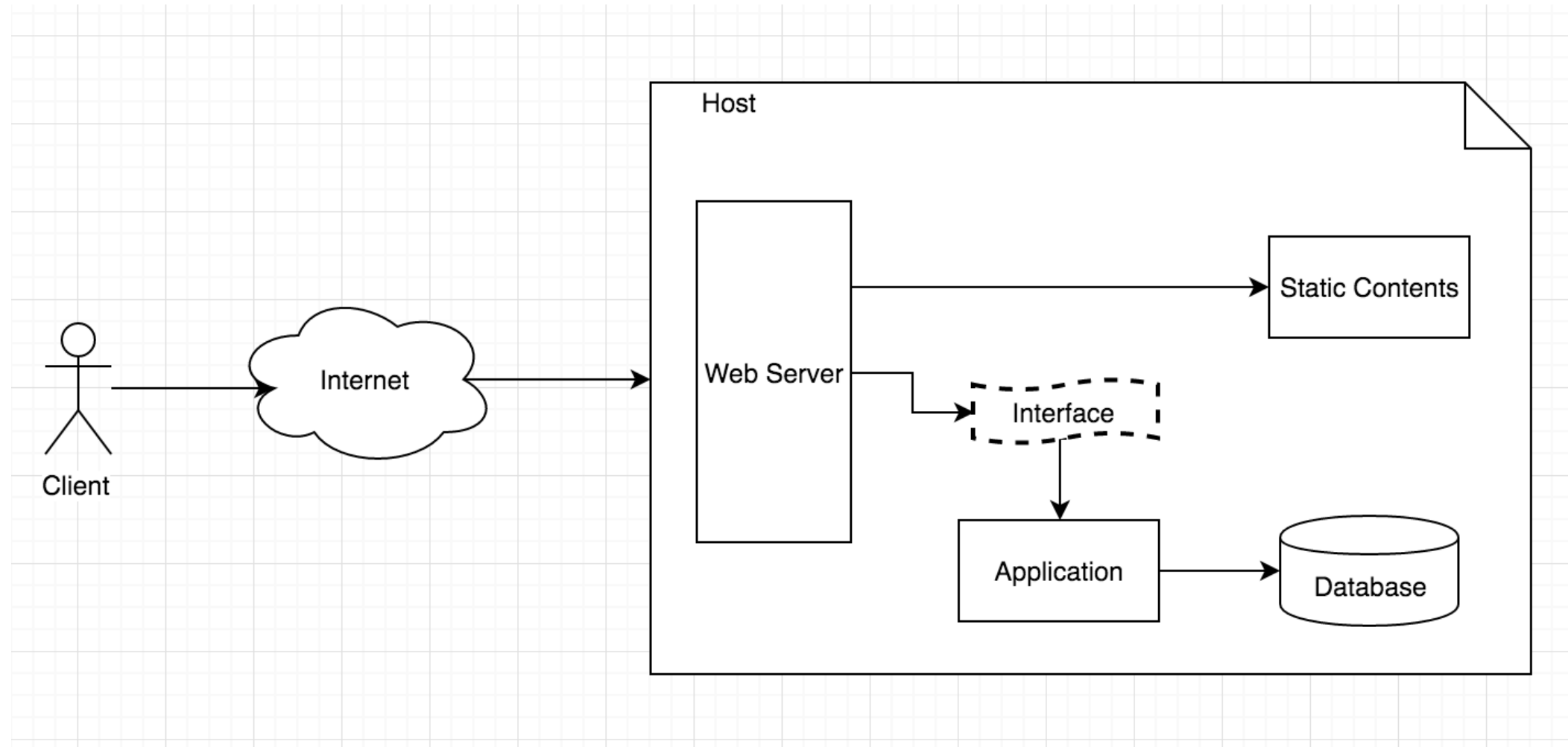
**All the Activities
that Make a Software System
Available for Use**

Deployment Activities

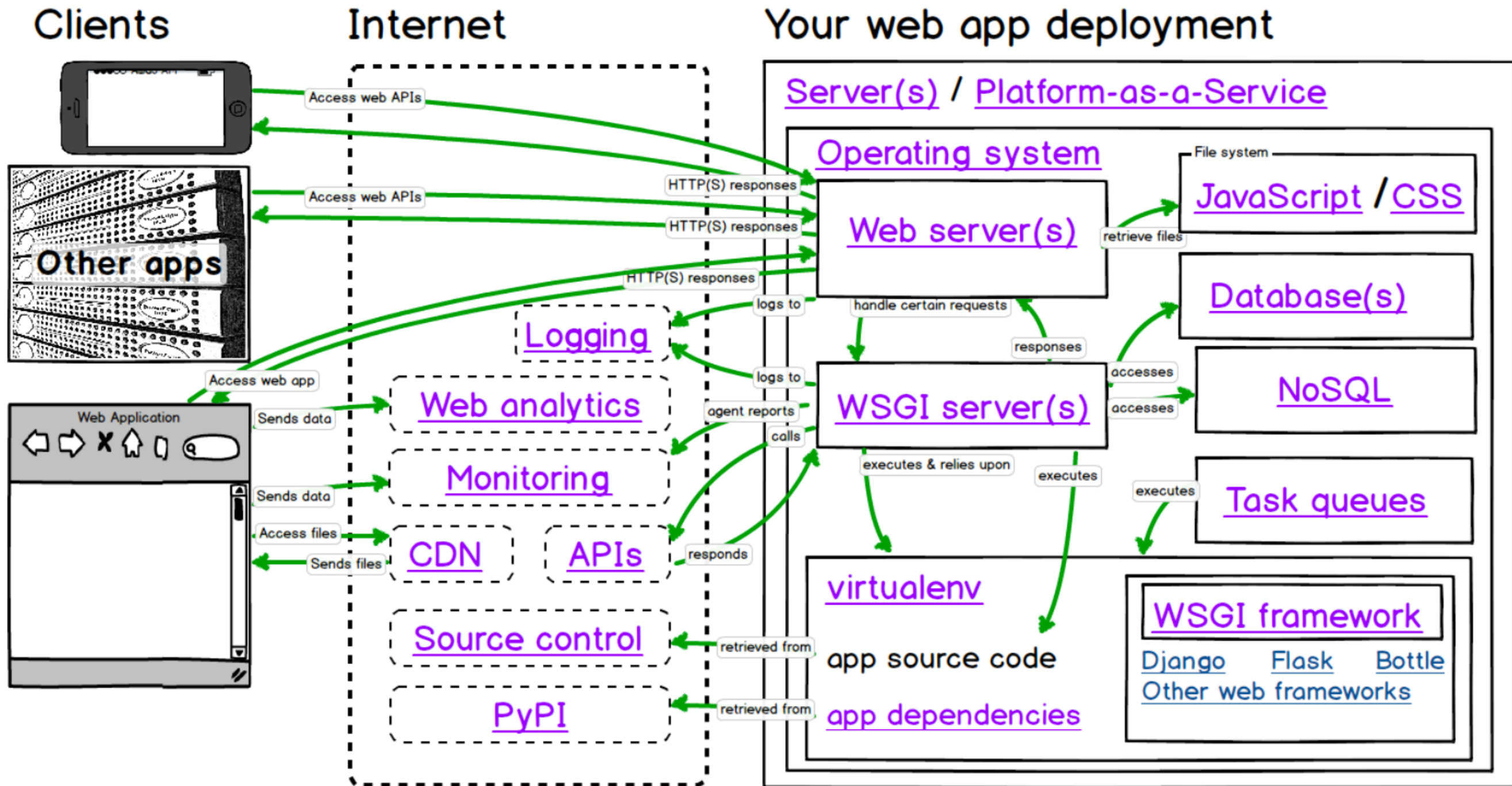
- **Requirements** — The Missing Piece of Software Development
- **Planning** — An Objective of Each and Every Activity
- **Designing** — Overall Pictures of Both High/Low-Level
- **Testing** — Do Not Ship Bugs At All
- **Deployment** — What Does this Deployment Mean

How to Deploy Software

A Minimum Python Web Application



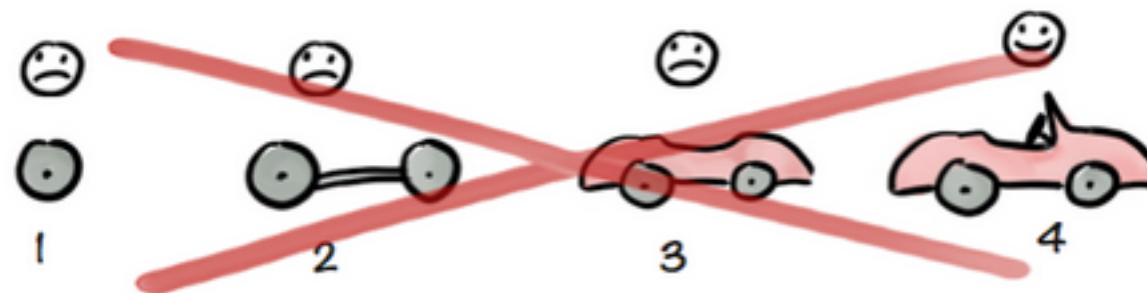
An Overview of Realworld Web Application



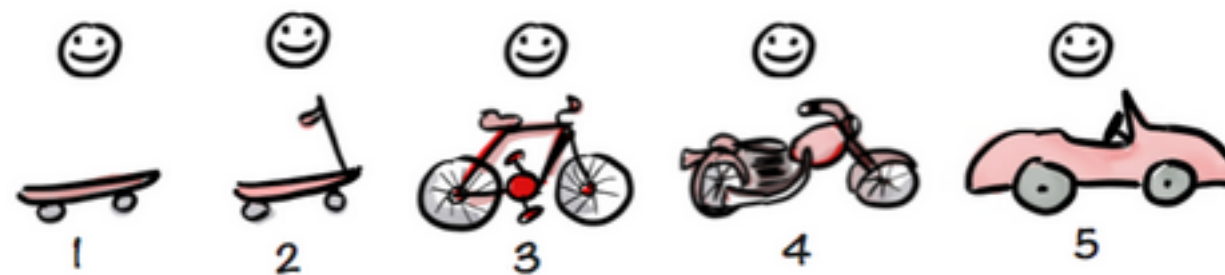
It's so complicated

MVP In Deployment

Not like this....



Like this!



Just Before You Start

- **Test your code**
 - We Do Not Ship Bugs At All
 - Confirm That It Is Deployable

Let's start

- **Get a Server**
 - Try AWS or Linode/DigitalOcean ...
 - Login via SSH/VNC ...
 - Initialize Security Rules ...
- **Apply a Public IP**
 - Apply an IP
 - Attatch the IP to Your Server

Preparing

- **Setup Dependencies**
 - Install Redis/Mongodb/ElasticSearch/...
 - Make them Daemon
- **Build Code/Config into Release**
 - Edit YAML/TOML/JSON Configuration
 - Maintain Environment Aware of Production/Staging/Test
- **Install Required Packages**
 - Install Project Based Libraries

Run Stand-alone

- **Run the App**
 - `export PYTHON_ENV=production`
 - `python run.py --port=8888`
 - `http :8888/the/actual/url`

Web Server

- **DNS**
 - Choose a Domain Name
 - Get the Public IP of Your Server
 - Add an A Record in DNS Console
- **Nginx**
 - Listen on 80/443
 - Proxy Pass Requests to Your App
 - `http https://your_domain_name/the/actual/url`

It is simple!

The Well-Known Three

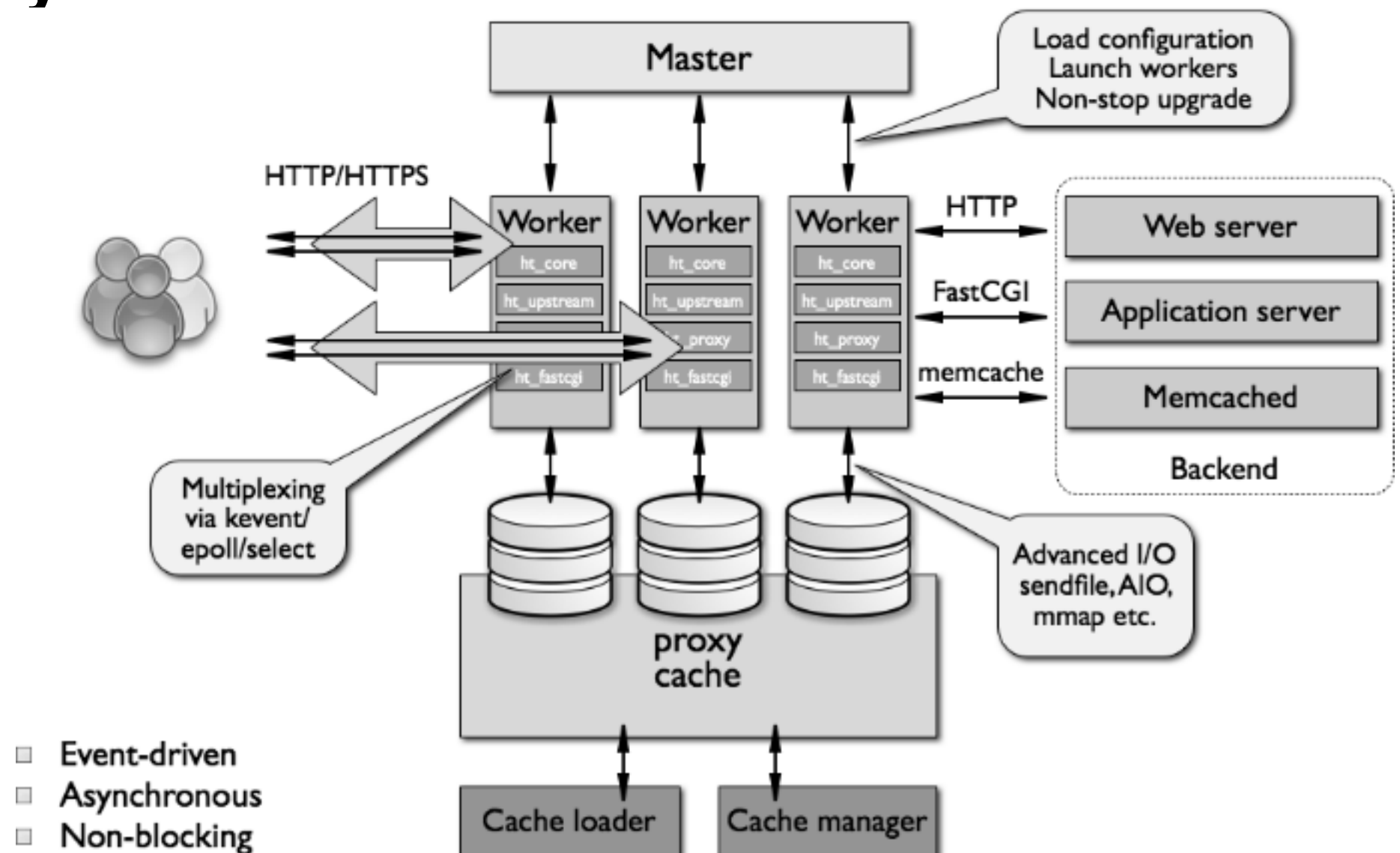
Supervisor: Progress Manager

- **Convenience**
- **Accuracy**
- **Centralised**



Nginx - Web Server

- **Simplicity**
- **High Concurrency**
- **Extensibility**



Sentry - Logging

- Log as Event Stream

The screenshot displays the Sentry web interface for an event stream. A circular callout highlights the event details:

- server_name = web-7
- level = error
- environment = beta
- sentry:release = 7af6cfafc5151

The main interface shows the event details, a code snippet, and a sidebar with charts and tags.

Tags:

- server_name = web-7
- level = error
- environment = beta
- sentry:release = 7af6cfafc5151

Exception: App Only Full Raw

Value: invalid literal for int() with base 10: '10.11'

Code Snippet:

```
sentry/db/models/Field (most recent call first)  
89.         return value  
90.         if value:  
91.             value = long(value)  
92.             assert value <= self.MAX_VALUE  
93.             return super(BoundedBigAutoField, self).get_prep_value(value)  
  
sentry/api/endpoints/project_rule_details.py in put at line 122 | application  
+ id=rule_id,  
  
sentry/api/base.py in dispatch at line 137 | application  
+ response = handler(request, *args, **kwargs)  
  
sentry/api/base.py in handle_exception at line 81 | application  
+ return super(Endpoint, self).handle_exception(exc)
```

Charts:

- Last 24 Hours
- Last 30 Days

First seen: 13 days ago
When: August 10, 2015 10:31 PM
Date: 4807130da8b9
Release: 7af6cfafc5151

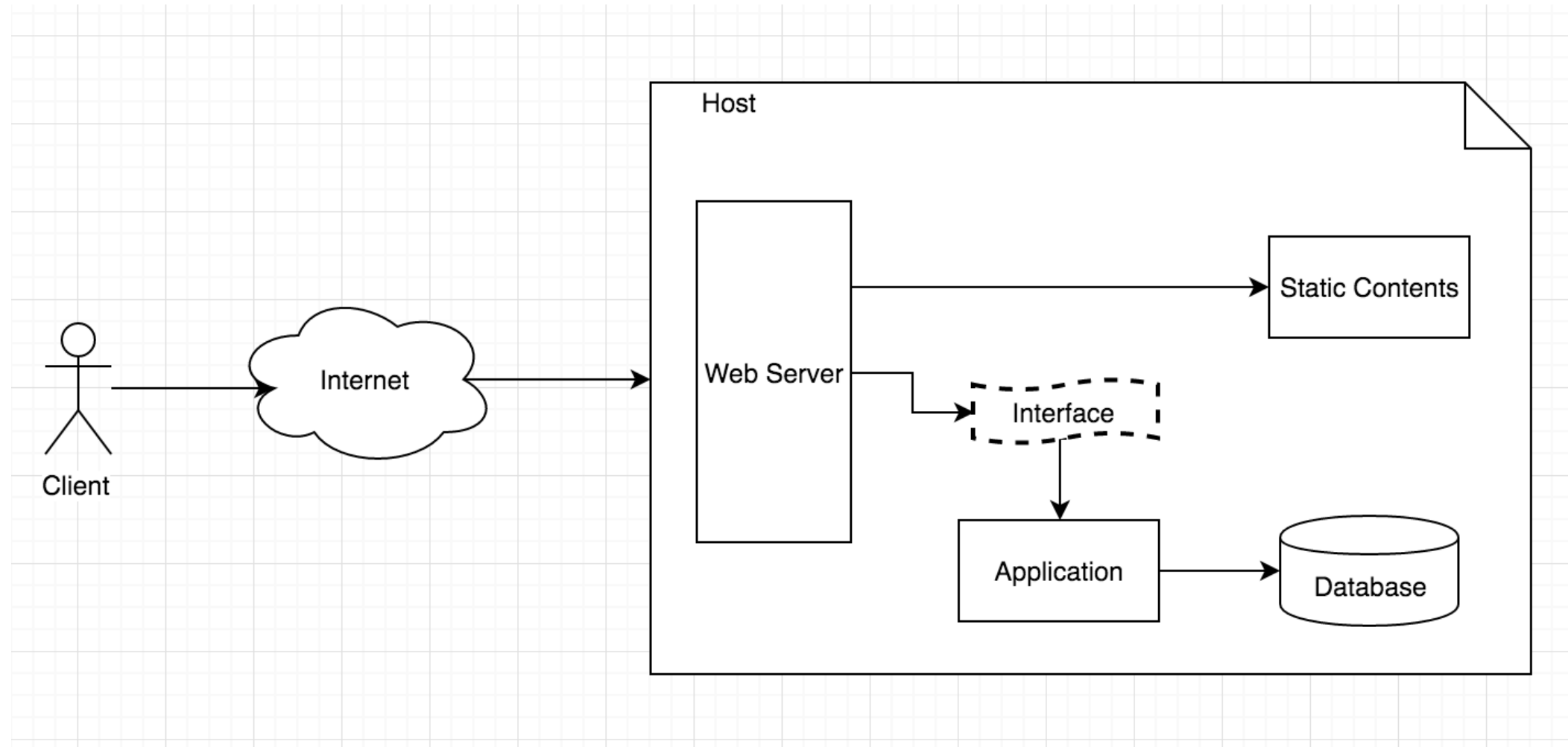
Last seen: a day ago
When: August 23, 2015 1:37 AM
Date: 7af6cfafc5151
Release: 7af6cfafc5151

Tags:

- Browser: 70% Chrome 44.0
- Device: 100% Other
- Environment: 100% beta
- Level: 100% error

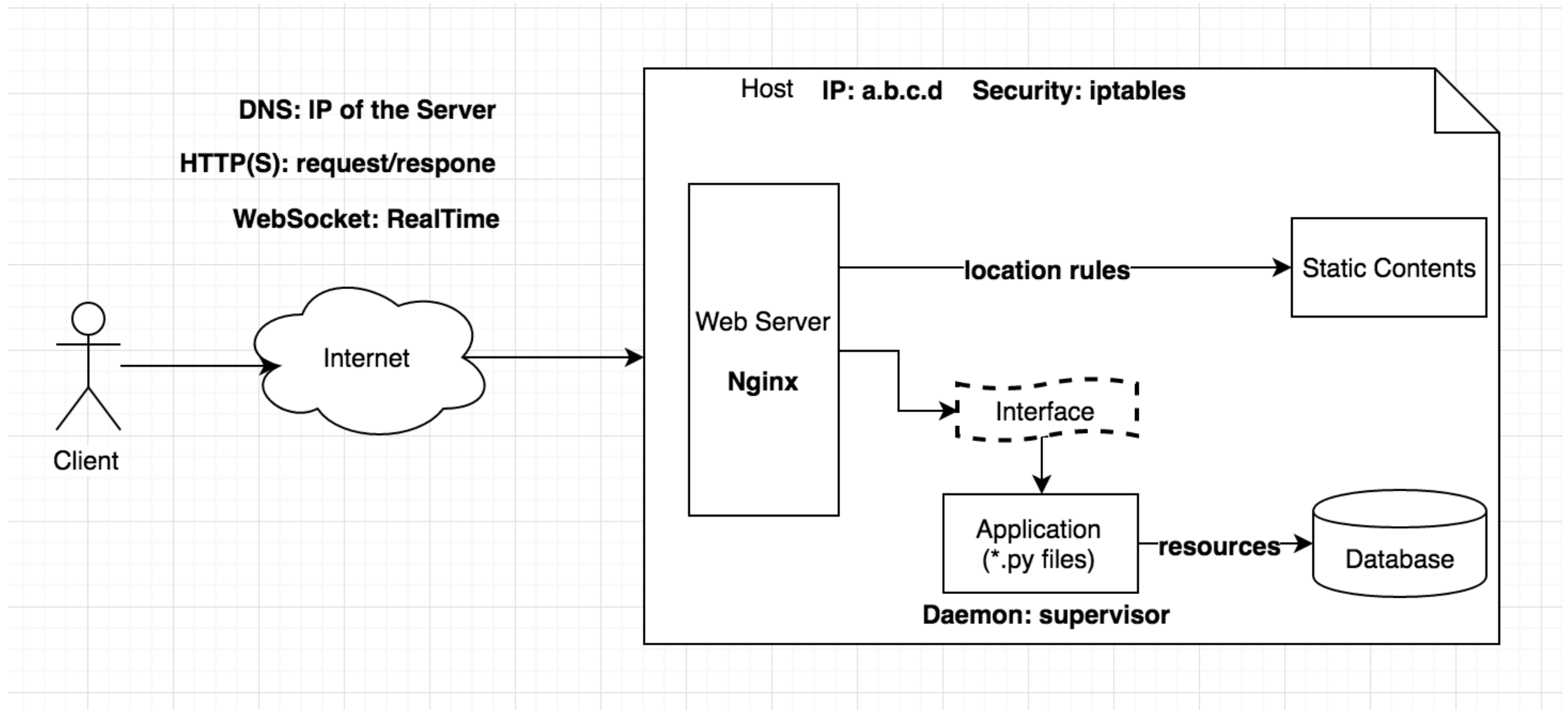
Trouble Shooting

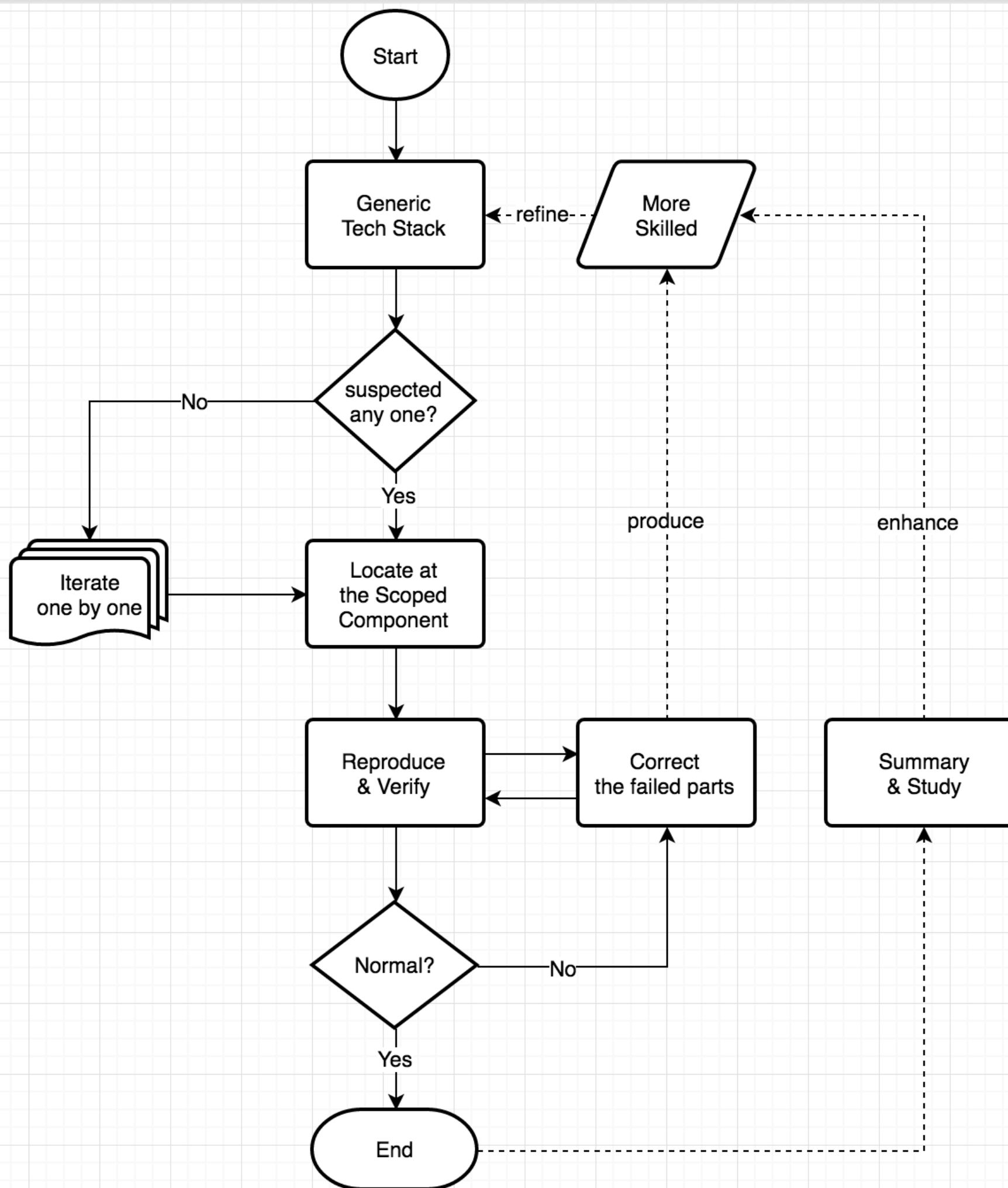
A Minimum Python Web Application



There may be problems anywhere

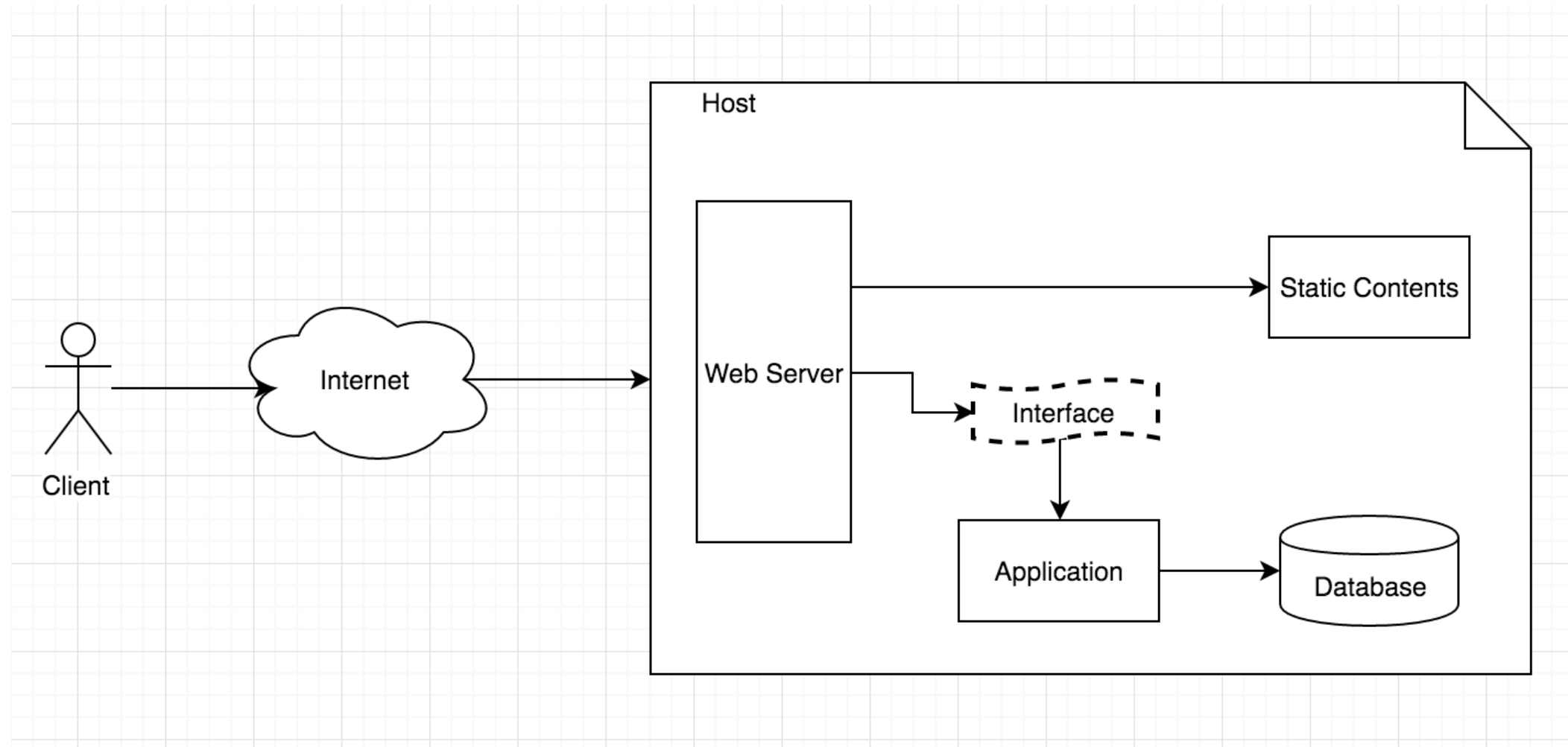
With Outlined Tech Stack





The So-Called “Interface”

A Minimum Python Web Application



What does “interface” mean?

Definition

It refers to:

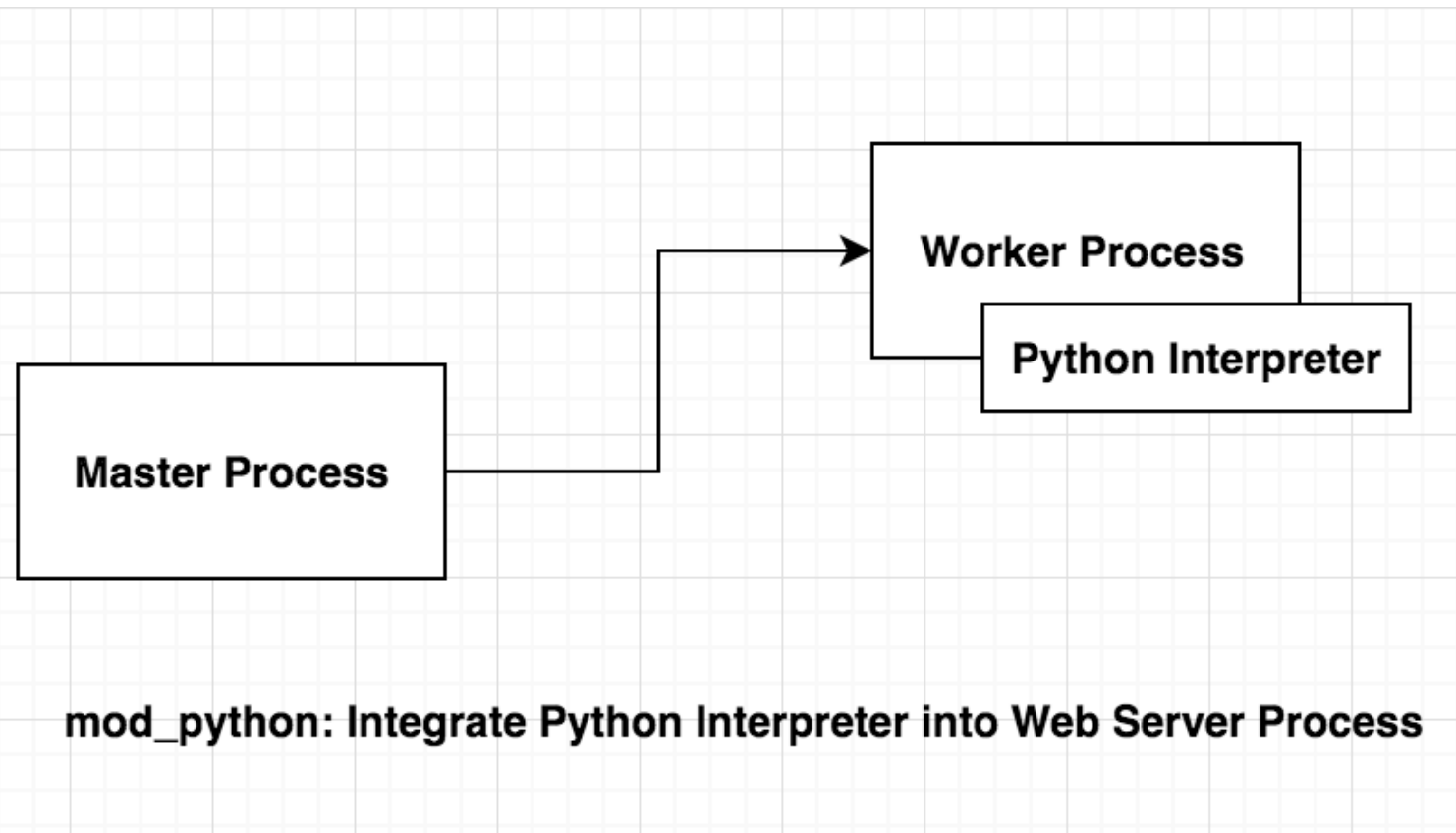
**a Protocol Describes
How the Web Server Talks with an Application**

CGI

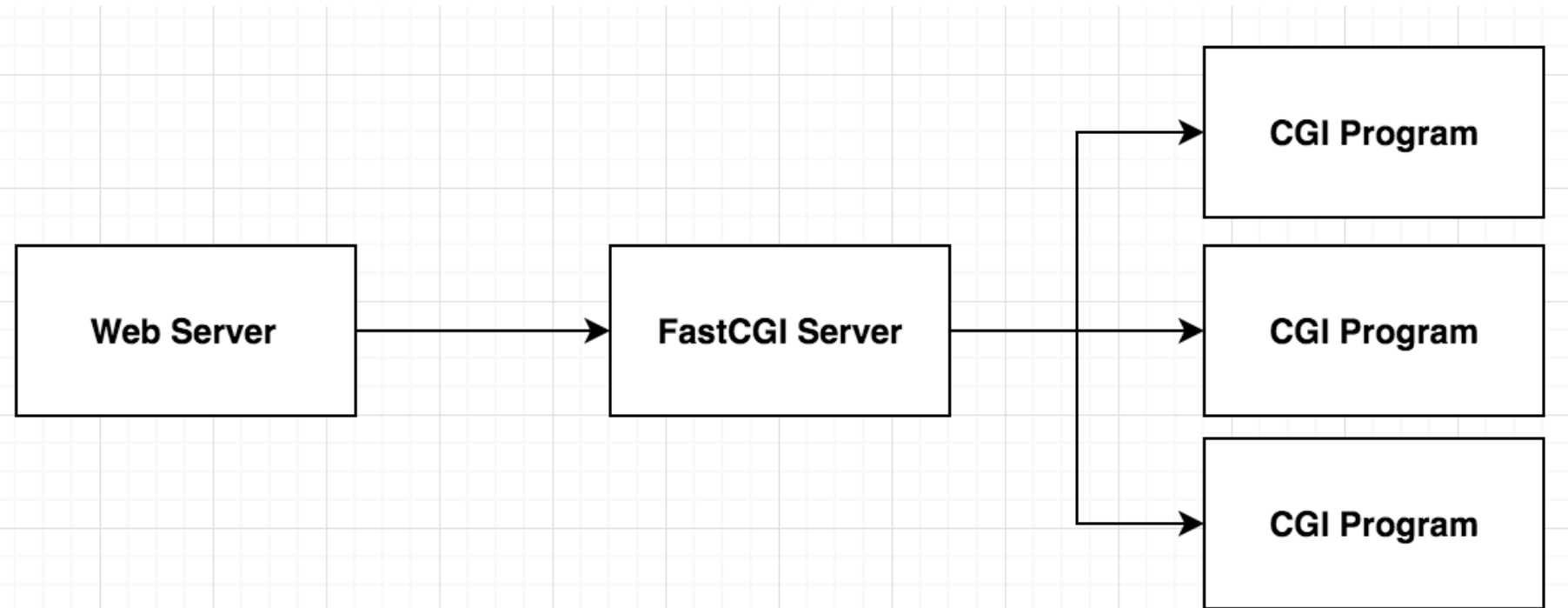


Web Server execute CGI program directly

mod_python

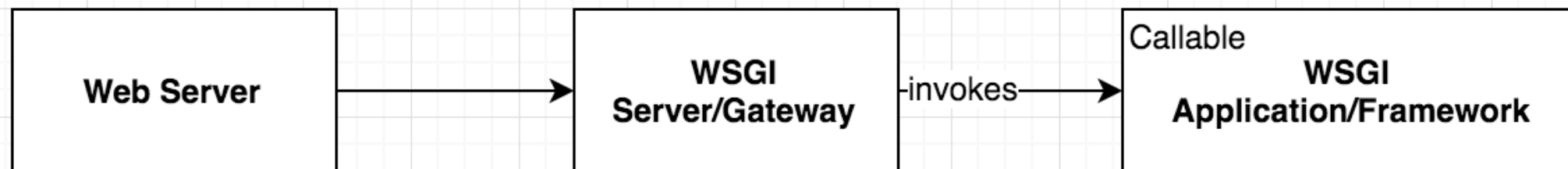


FastCGI/SCGI



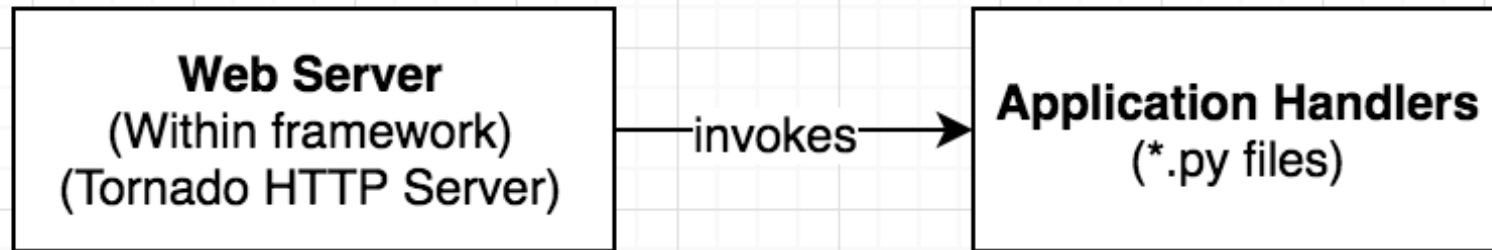
Web Server execute CGI program through FastCGI Server

WSGI



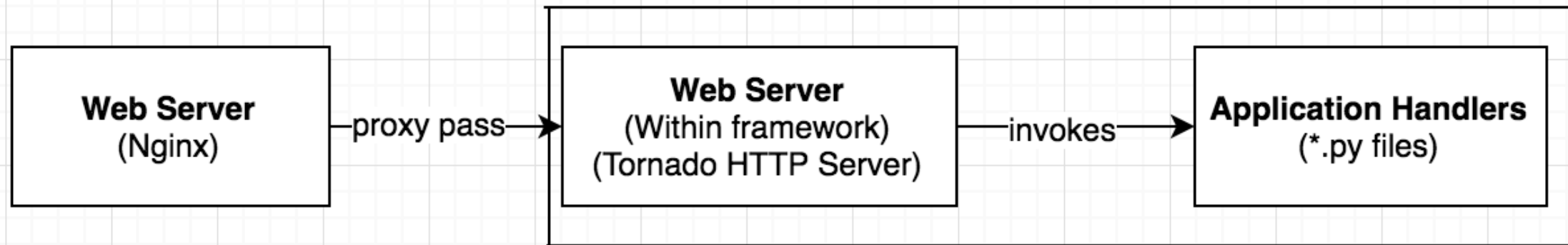
a WSGI server simply invokes a callable object on the WSGI application

Integrated Web Server



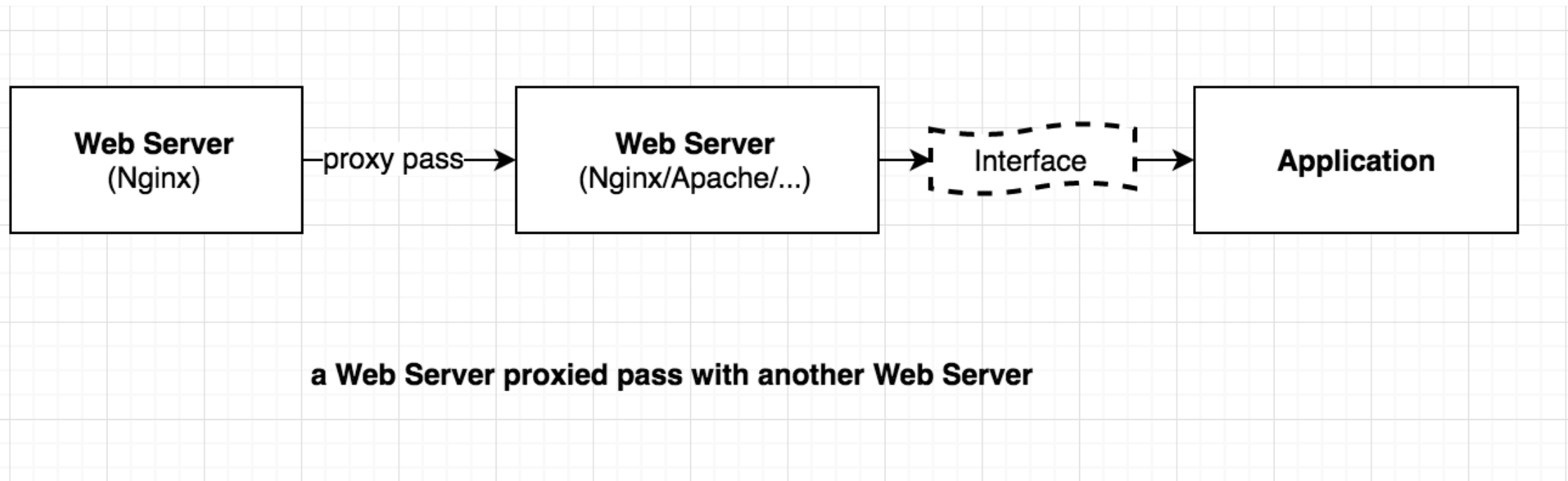
some Frameworks integrate HTTP Server, and invoke handlers all by themselves

Proxy Pass



an integrated Web Server Application proxied pass with a Web Server

Proxy Pass

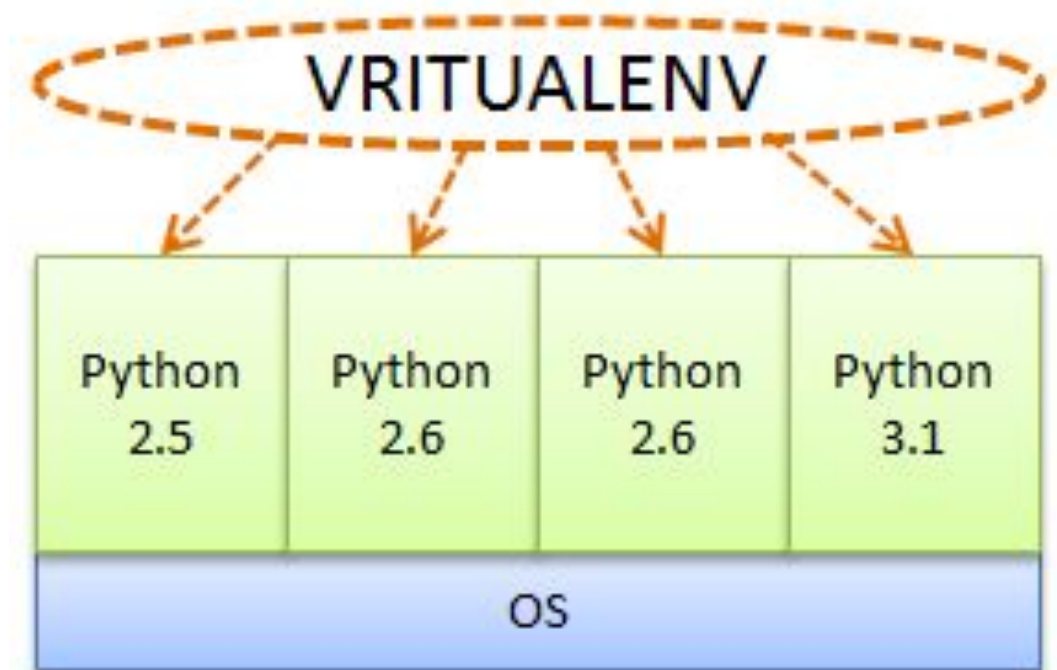


There Are Some Other Topics

Standardisation

Explicitly declare and isolate dependencies

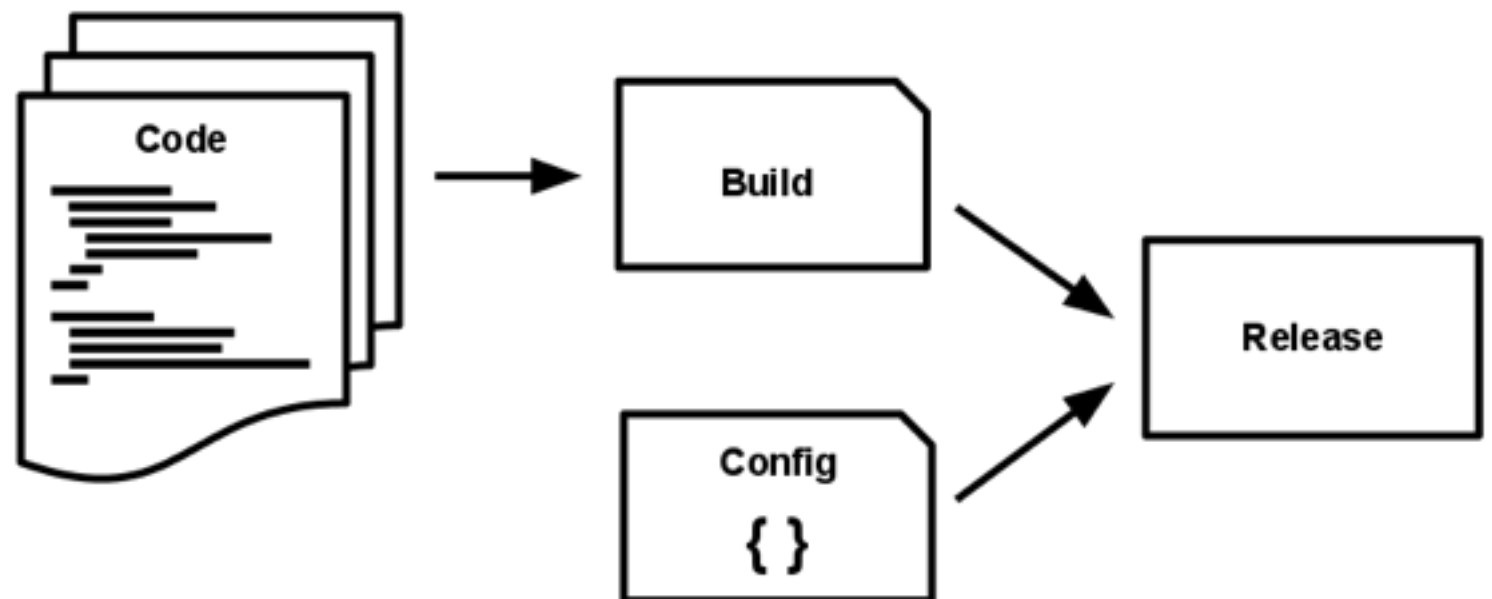
- Virtual Environment for Isolation
- requirements.txt for Dependencies



Standardisation

Store config in the environment

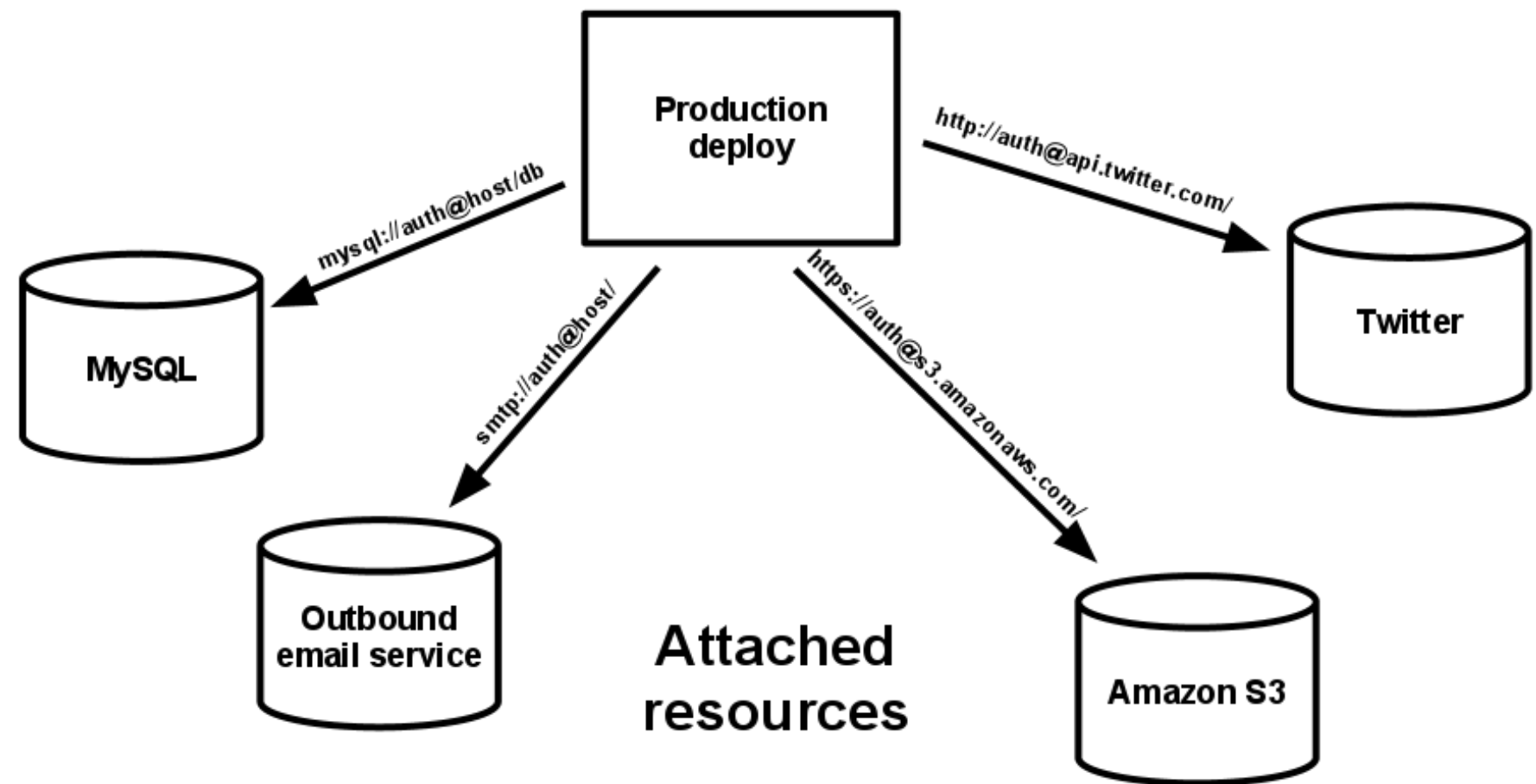
- YAML/TOML/JSON
- Dev/Test/Staging/Production



Standardisation

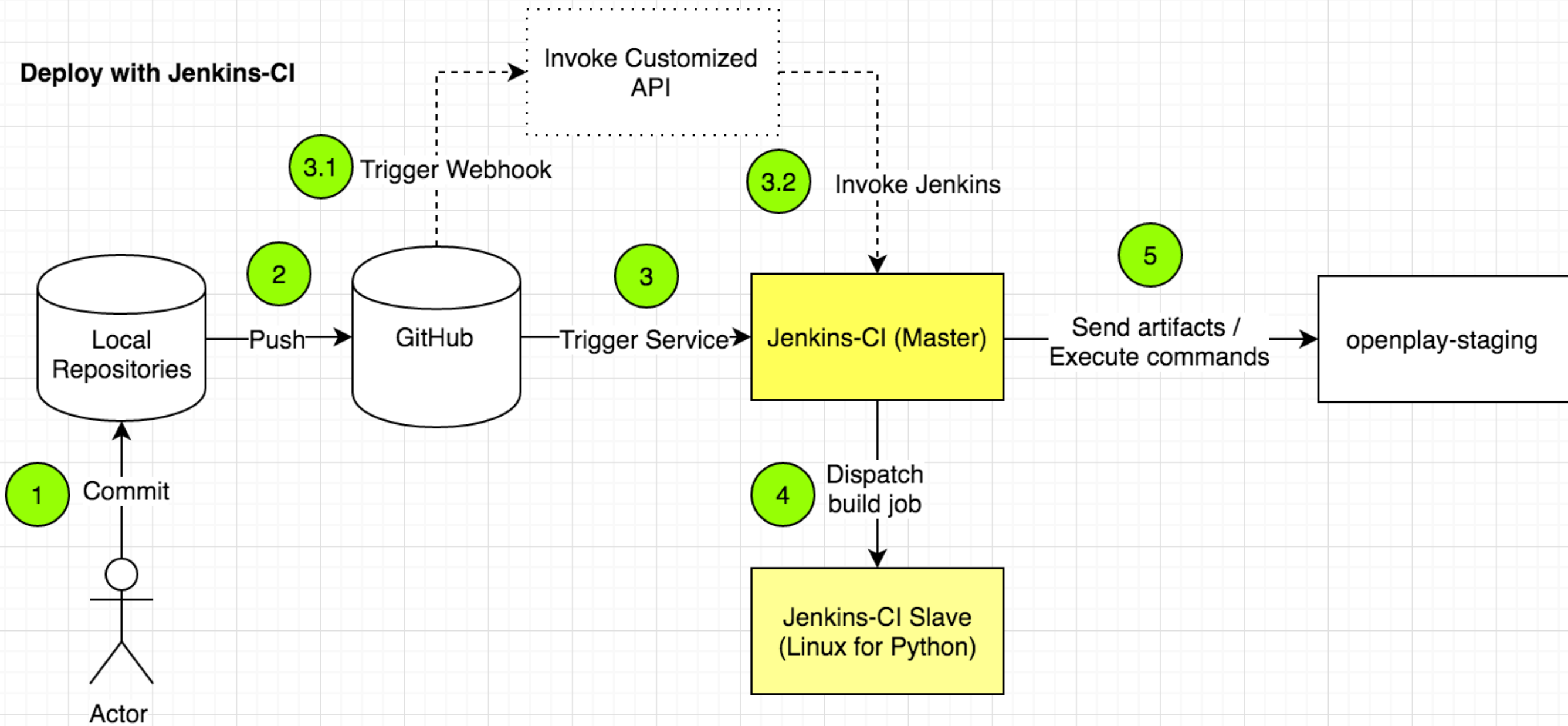
Treat backing services as attached resources

- Configurable
- Detached/Attached



Automation

Deploy with Jenkins-CI



Monitoring

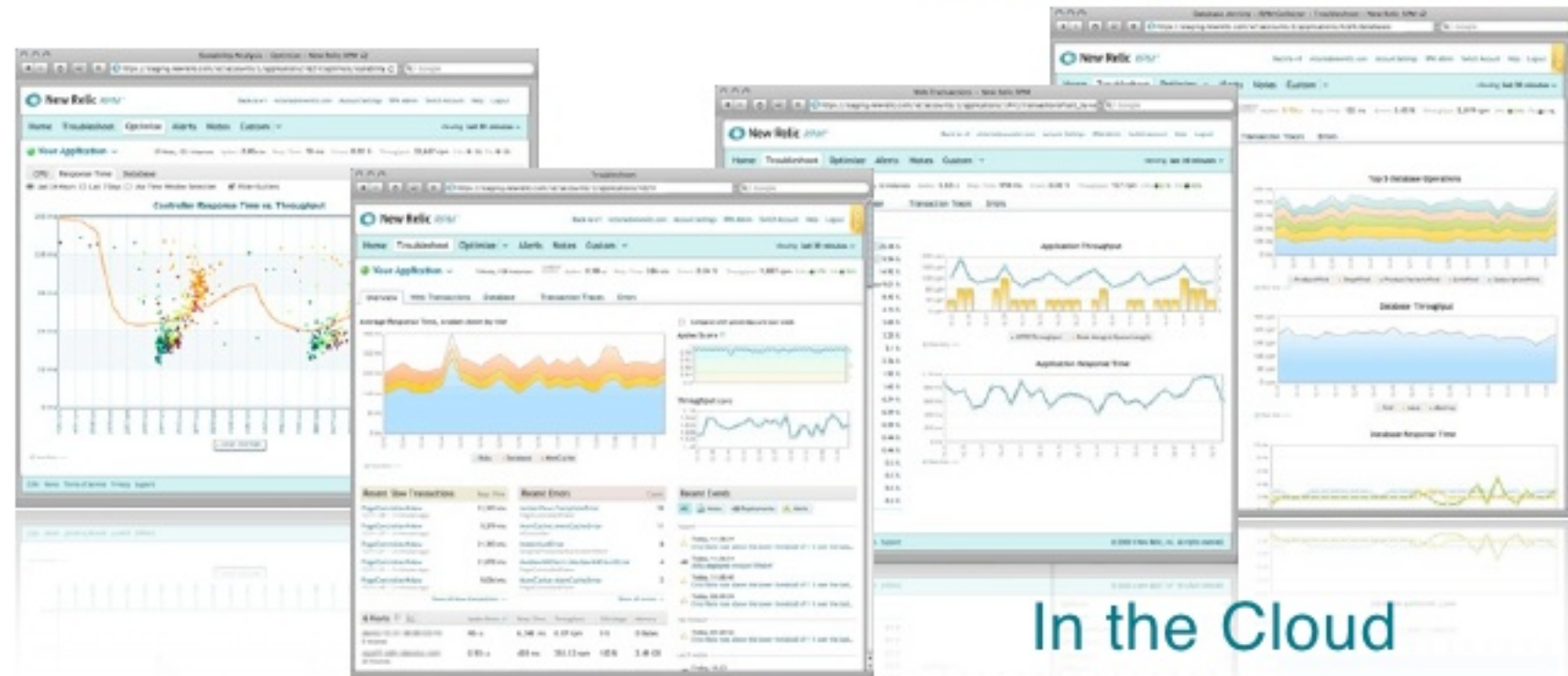
Aware of Your Application

- Console
- Utilities
- Services

RPM

✓ Monitors
✓ Troubleshoots
✓ Tunes

Web Applications



In the Cloud

Availability

Max Availability

- **Max Mean Time between Failures**
- **Min Mean Time to Recover**
- **$\geq 99.9\%$**

Level of Availability	Percent of Uptime	Downtime per Year	Downtime per Day
1 Nine	90%	36.5 days	2.4 hrs.
2 Nines	99%	3.65 days	14 min.
3 Nines	99.9%	8.76 hrs.	86 sec.
4 Nines	99.99%	52.6 min.	8.6 sec.
5 Nines	99.999%	5.25 min.	.86 sec.
6 Nines	99.9999%	31.5 sec.	8.6 msec

Q & A

Reference

- Comparison of web server software
- <http://www.fullstackpython.com/deployment.html>
- <http://www.fullstackpython.com/wsgi-servers.html>
- <https://docs.python.org/3.5/howto/webrowsers.html>
- <https://github.com/fouber/blog/issues/6>
- <https://ruby-china.org/topics/12686>
- <http://supervisord.org/>
- <http://nginx.org/>
- <https://jenkins.io/>
- <https://newrelic.com/>
- <http://12factor.net/>
- <https://github.com/alex/what-happens-when>