Веб-сервер

Основы Веб-программирования

Кафедра Интеллектуальных Информационных Технологий, ИнФО, УрФУ

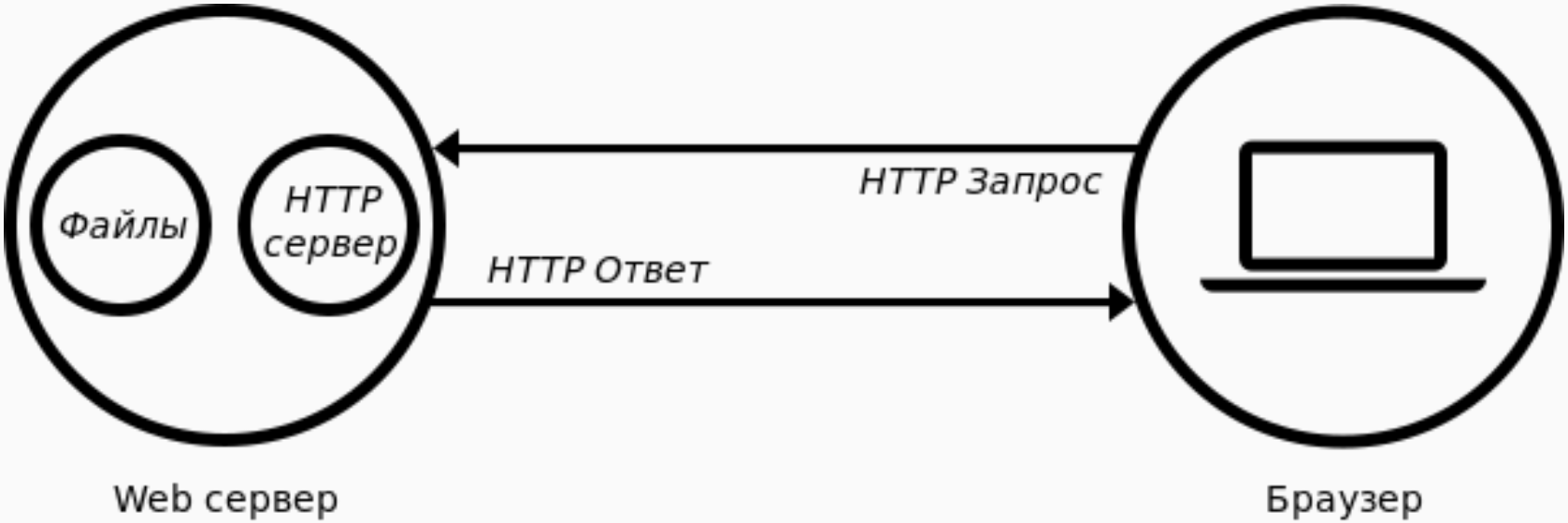
Ресурсы

https://ru.wikipedia.org/wiki/Веб-сервер

http://lectureswww.readthedocs.org/5.web.server/index.html

Называют

- Железо, на котором запущенны сервисы и есть подключение с сети.
- Программу, которая слушает 80 порт и отвечает на HTTP запросы



Типы HTTP серверов

- **Статический веб-сервер**, или стек, посылает размещенные на нем файлы в браузер “как есть”. (nginx, Apache)
- **Динамических веб-сервер** состоит из статического веб-сервера плюс дополнительного программного обеспечения, наиболее часто **сервером приложений** и базы данных. (Gunicorn, Waitress, Tornado)

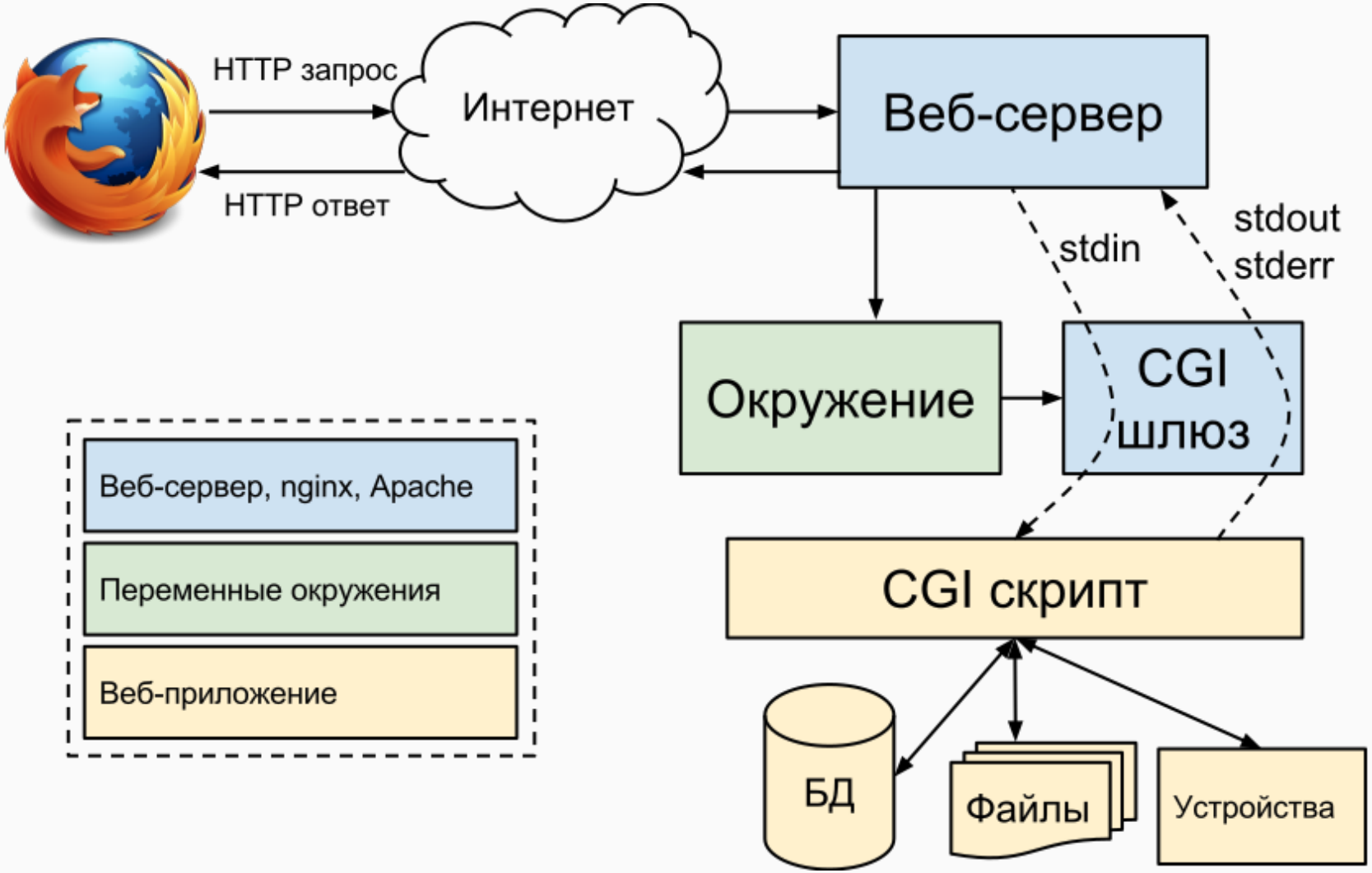
Типы сайтов

- Статические Веб-сайты проще всего установить. К ним относятся HTML страницы, картинки, шрифты, CSS стили и прочее…
- Динамический Веб-сайт означает, что сервер обрабатывает данные или даже генерирует их на лету из базы данных. Это обеспечивает больше гибкости, но технически сложнее в обслуживании, что делает его более сложным в разработке.

Связь Веб-приложения с HTTP-сервером

- CGI
- FastCGI
- Встроенный сервер
- WSGI

CGI


HTTP запрос
HTTP ответ
Интернет
Веб-сервер
stdin
stdout
stderr
Окружение
CGI
шлюз
Веб-сервер, nginx, Apache
Переменные окружения
Веб-приложение
CGI скрипт
БД
Файлы
Устройства

CGI

CGI — это не язык программирования!
Это простой протокол, позволяющий Веб-серверу передавать данные через **stdin** и читать их из **stdout**.

Поэтому, в качестве CGI-обработчика может использоваться программа на любом языке, способном работать со стандартными потоками ввода-вывода.

CGI. Самый простой сервер.

```
python3 -m http.server –cgi 8000
```

CGI. Hello World!

Код 1: Си

```c
#include <stdio.h>
int main(void) {
   printf("Content-Type: text/plain\n\n");
   printf("Hello, world!\n\n");
   return 0;
}
```

Код 2: Python

```python
#!/usr/bin/python
print("Content-Type: text/plain\n\nHello, world!")
```

CGI. Переменные окружения

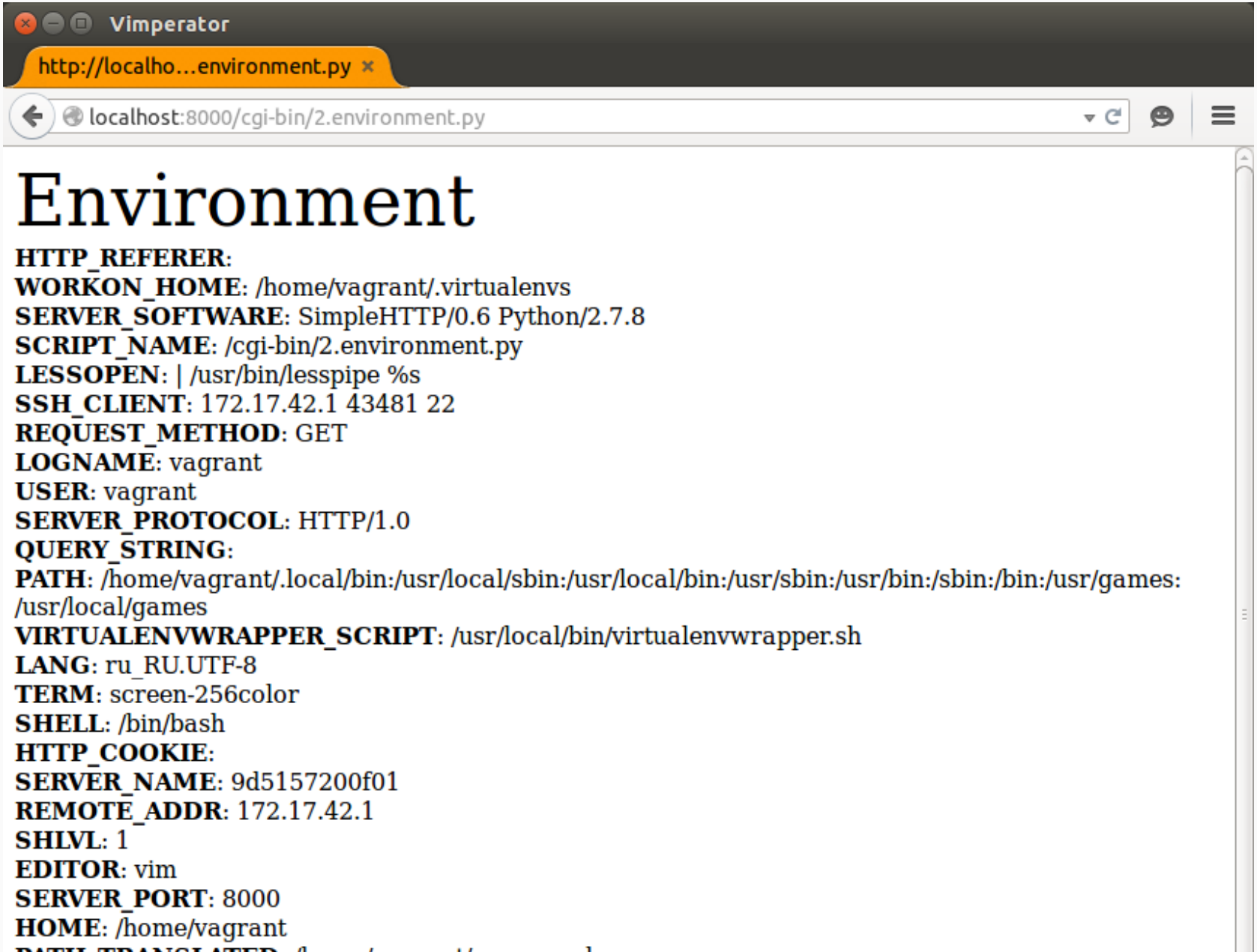
Код 3: Python

```python
#!/usr/bin/python
import os

print("Content-type: text/html\r\n\r\n")
print("<font size=+10>Environment</font><br>")

for param in os.environ.keys():
    print("<b>%20s</b>: %s<br>" %
            (param, os.environ[param]))
```

CGI. Переменные окружения



CGI. Преимущества

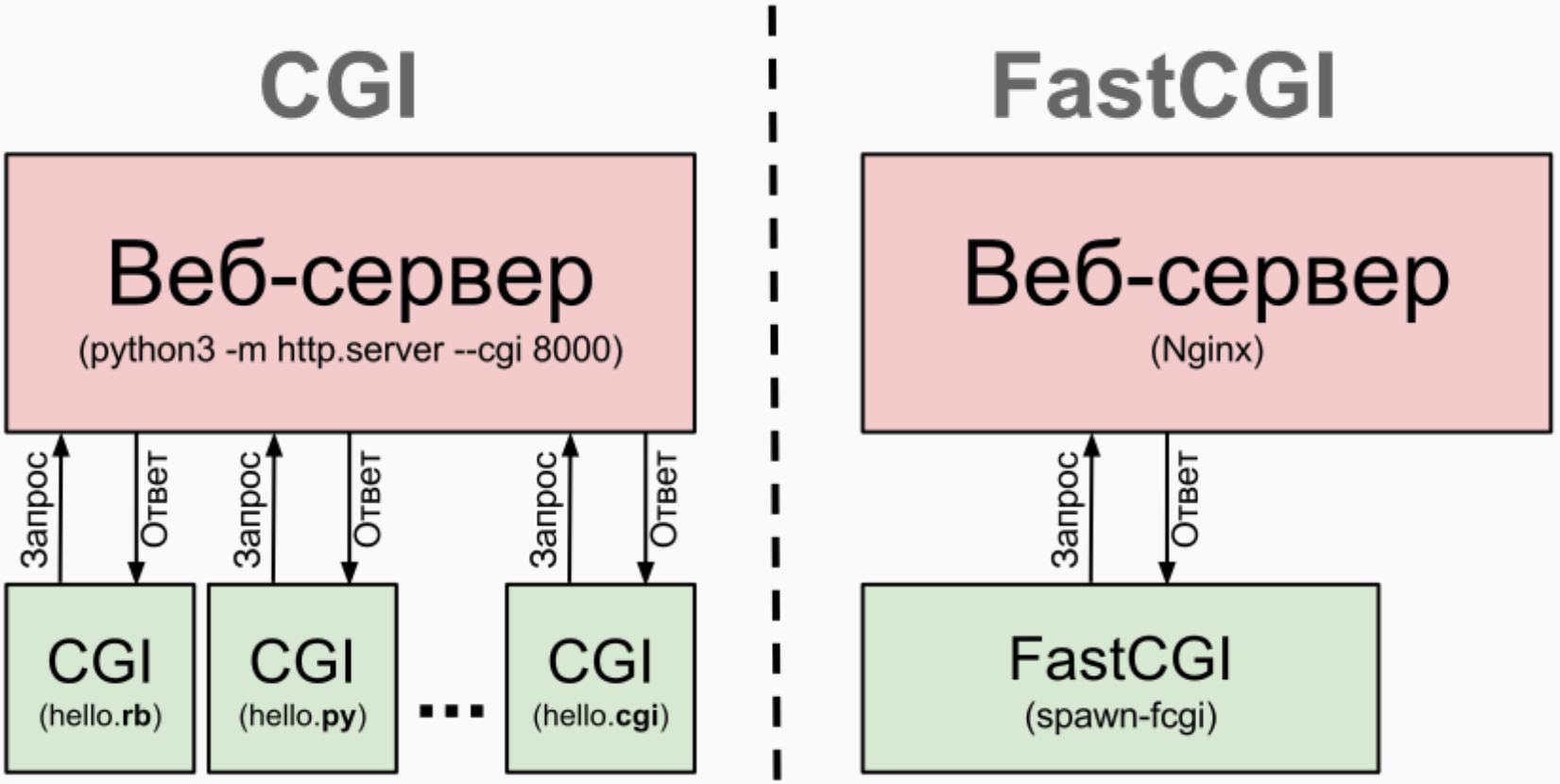
- Процесс CGI скрипта не зависит от Веб-сервера и в случае падения ни как не отразится на работе последнего.
- Может быть написан на любом языке программирования.
- Поддерживается большинством Веб-серверов.

CGI. Недостатки

Потребление ресурсов

Дело в том, что каждое обращение к CGI-приложению вызывает порождение нового процесса, со всеми вытекающими отсюда накладными расходами.

FastCGI


CGI
Веб-сервер
(python3 -m http.server --cgi 8000)
Запрос
Ответ
Запрос
Ответ
Запрос
Ответ
CGI
(hello.rb)
CGI
(hello.py)
CGI
(hello.cgi)
FastCGI
Веб-сервер
(Nginx)
Запрос
Ответ
FastCGI
(spawn-fcgi)

FastCGI

- В отличие от CGI, FastCGI использует постоянно запущенные процессы для обработки множества запросов.
- FastCGI-процессы используют для связи с сервером **Unix Domain Sockets** или **TCP/IP**.

FastCGI. Приложение

Сборка: gcc -o hello.fcgi hello.cpp -lfcgi

Запуск: spawn-fcgi -p 5000 -n hello.fcgi

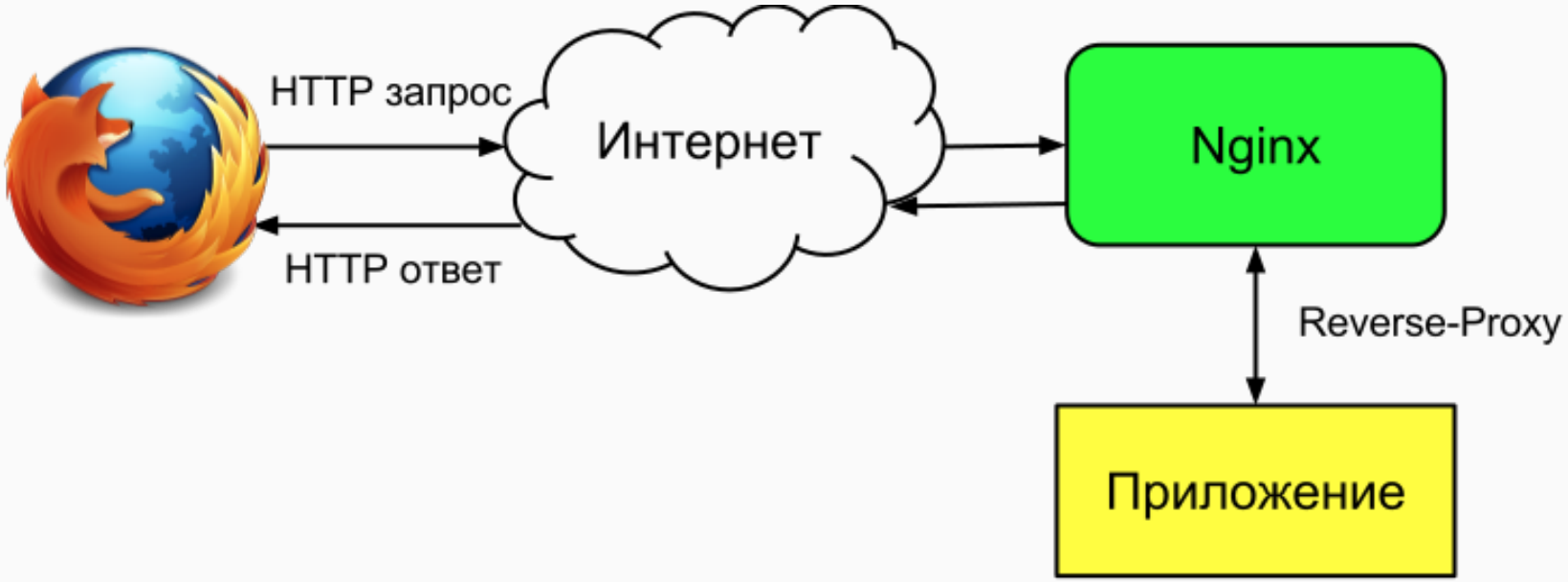
Код 4: Си

```c
#include "fcgi_stdio.h"
#include <stdlib.h>

int main(void)
{
 while(FCGI_Accept() >= 0)
 {
     printf("Content-type: text/html\r\n"
            "Status: 200 OK\r\n\r\nHello World!");
 }
 return 0;
}
```

Встроенный сервер

В некоторых языках, например Go, уже существует встроенный Веб-сервер.



Встроенный сервер. FastCGI

В этом случае не нужно запускать отдельно fcgi сервер, например spawn-fcgi.

Код 5: Go

```go
package main
import (
    "fmt"
    "net"
    "net/http"
    "net/http/fcgi"
)

func handler(res http.ResponseWriter,
             req *http.Request) {
    fmt.Fprint(res, "Hello World!")
}
```

Встроенный сервер. FastCGI

go run hello.go

Код 6: Go

```go
func main() {
    // For local machine
    // l, _ := net.Listen("unix",
    //                    "/var/run/go-fcgi.sock")

    // TCP 5000 listen
    l, err := net.Listen("tcp", "0.0.0.0:5000")
    if err != nil {
        return
    }
    http.HandleFunc("/", handler)
    fcgi.Serve(l, nil)
}
```

Встроенный сервер. HTTP

Может запускаться самостоятельно без дополнительных программ и настроек.

Код 7: Go

```go
package main

import (
    "fmt"
    "net/http"
)

func handler(w http.ResponseWriter,
             r *http.Request) {
    fmt.Fprintln(w, "Hello World!")
}
```

Встроенный сервер. HTTP

go run hello.go

Код 8: Go

```
func main() {
    http.HandleFunc("/", handler)
    http.ListenAndServe(":8000", nil)
}
```

WSGI. PEP-333. PEP-3333

WSGI предоставляет простой и универсальный интерфейс для соединения веб-серверов, веб-приложений и прослоек между ними.

- Application
- Server
- Middleware

Все Python фреймворки поддерживают WSGI.
Даже Django!