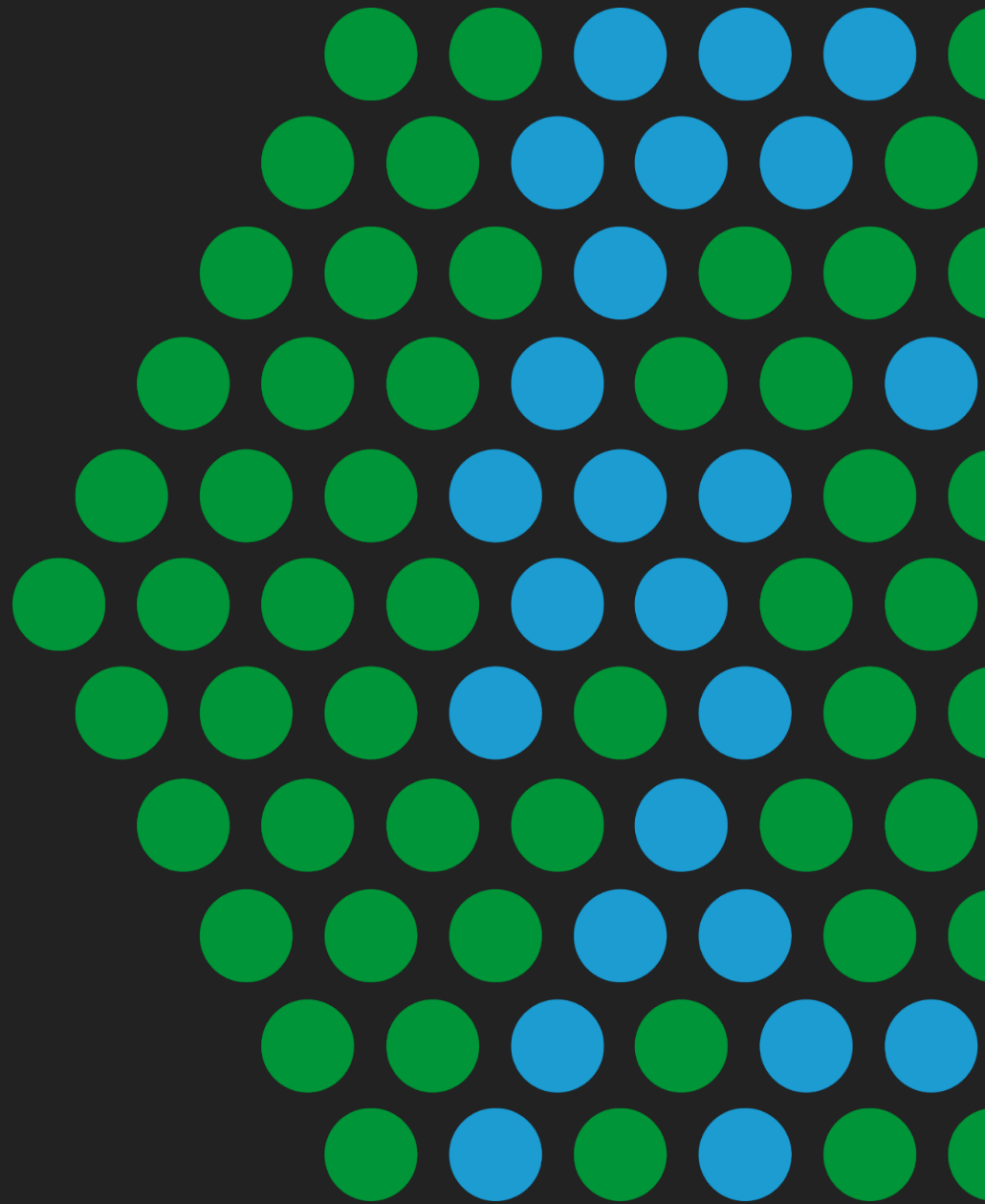


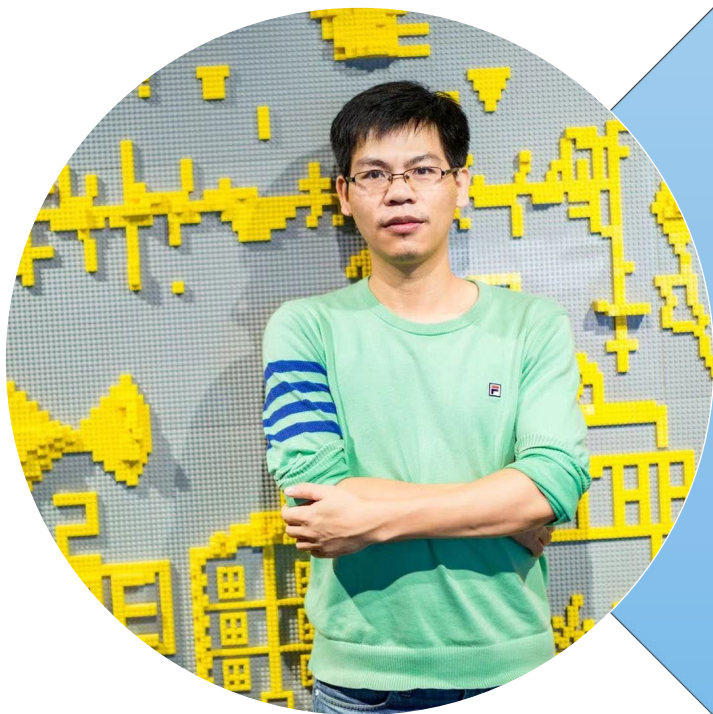


NGINX Unit

现代应用服务器



关于我



洪志道

- NGINX核心团队成员
- NGINX Unit核心研发

主题

SPOILER ALERT

1. 介绍

2. 架构

3. Q&A

介绍



NGINX Unit是NGINX作者Igor Sysoev亲自设计，由NGINX团队核心成员开发的应用服务器软件。

关键特性



配置：完全热加载, 用RESTful API管理JSON配置



应用：支持各版本的主流语言



性能：高性能, 低内存, 低延迟

配置

- JSON格式
- RESTful API管理
- 任意粒度修改
- 实时生效

```
{
  "listeners": {
    " *:8081": {
      "pass": "applications/myblog",
    }
  },
  "applications": {
    "myblog": {
      "type": "php",
      "processes": {
        "max": 10,
        "spare": 2
      },
      "root": "/srv/demo/php",
      "user": "www-php",
      "index": "index.php",
      "limits": {
        "timeout": 10
      },
      "environment": {
        "DB_HOST": "127.0.0.1"
      }
    }
  }
}
```

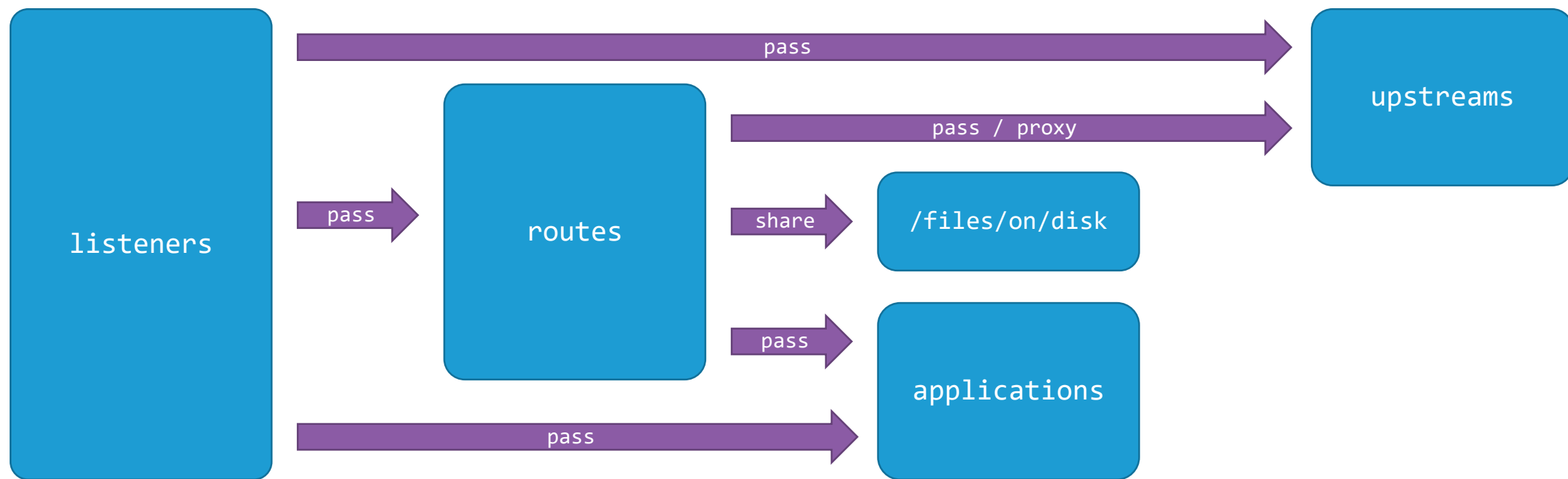
```
> cat config.json
{ "type": "php", "root": "/www/blogs/scripts" }
```

```
> curl -X PUT --data-binary @config.json --unix-socket /path/to/control.unit.sock \
http://localhost/config/applications/blogs
```

```
{ "success": "Reconfiguration done." }
```

```
{
  "listeners": {
    "*:80": {
      "pass": "routes"
    }
  },
  "routes": [
    {
      "match": {
        "uri": [ "*.php", "*.php/*", "/wp-admin/" ]
      },
      "action": {
        "pass": "applications/wordpress/direct"
      }
    },
    {
      "action": {
        "share": "/path/to/app/",
        "fallback": {
          "pass": "applications/wordpress/index"
        }
      }
    }
  ],
  "applications": {
    "wordpress": {
      "type": "php",
      "targets": {
        "direct": {
          "root": "/path/to/app/"
        },
        "index": {
          "root": "/path/to/app/",
          "script": "index.php"
        }
      }
    }
  }
}
```


核心功能



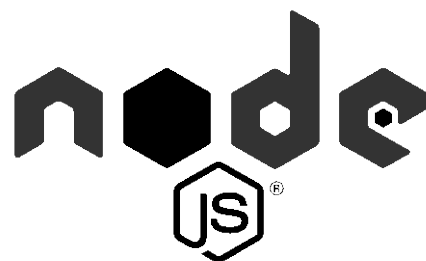
Layer 4
TCP/TLS/ports

Layer 7
Headers, URIs

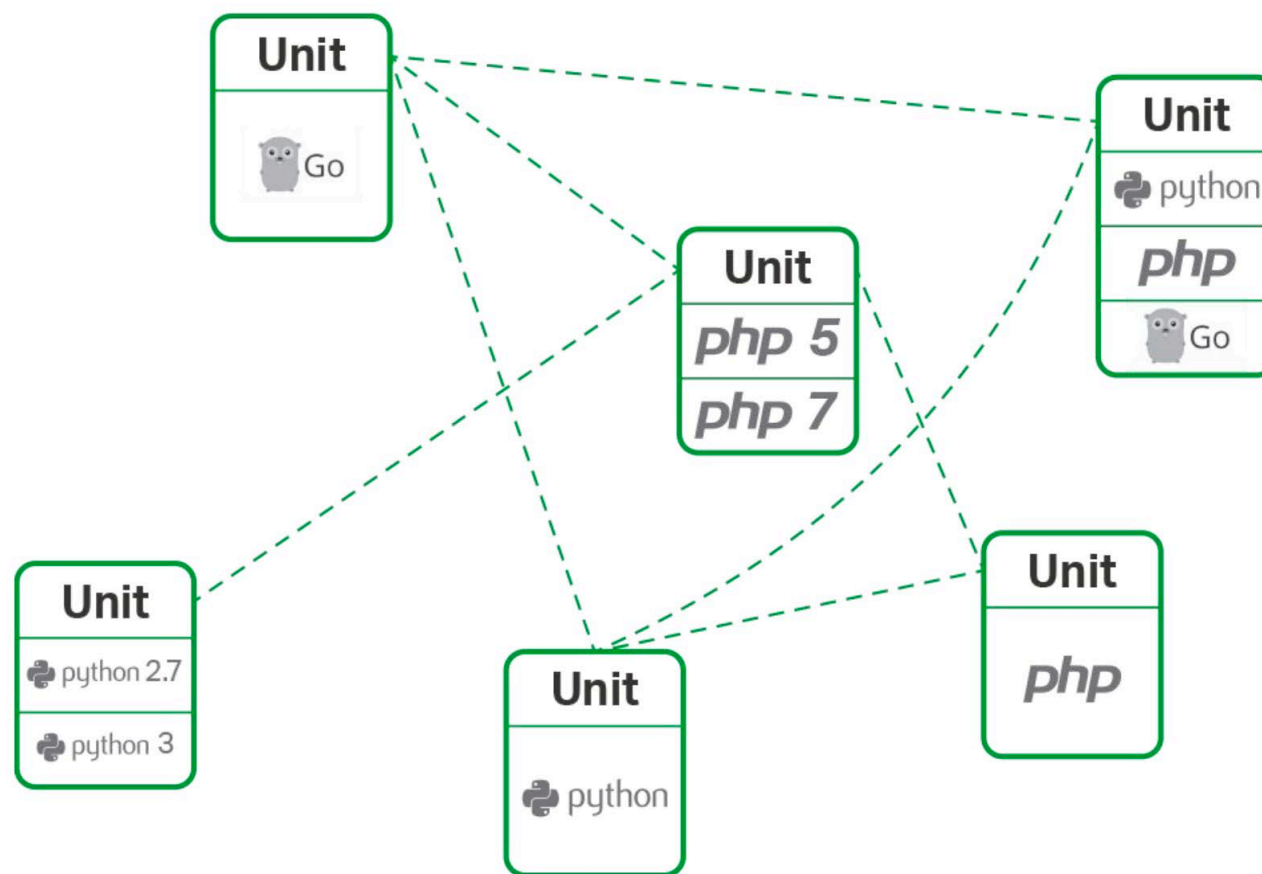
User Space
Code/Files

Layer 7
IP/ports

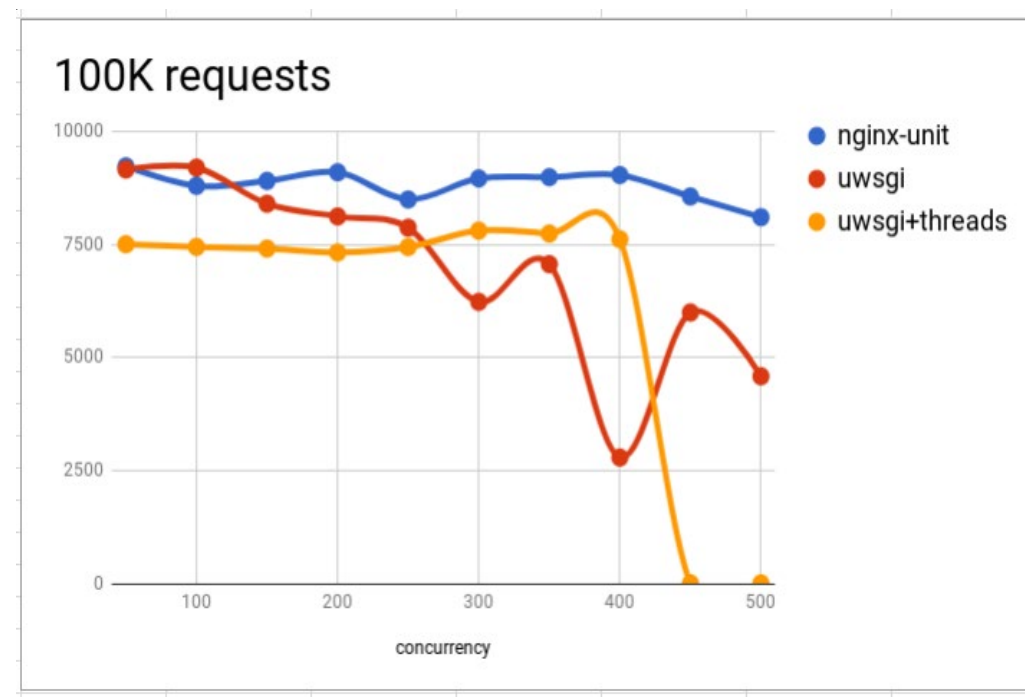
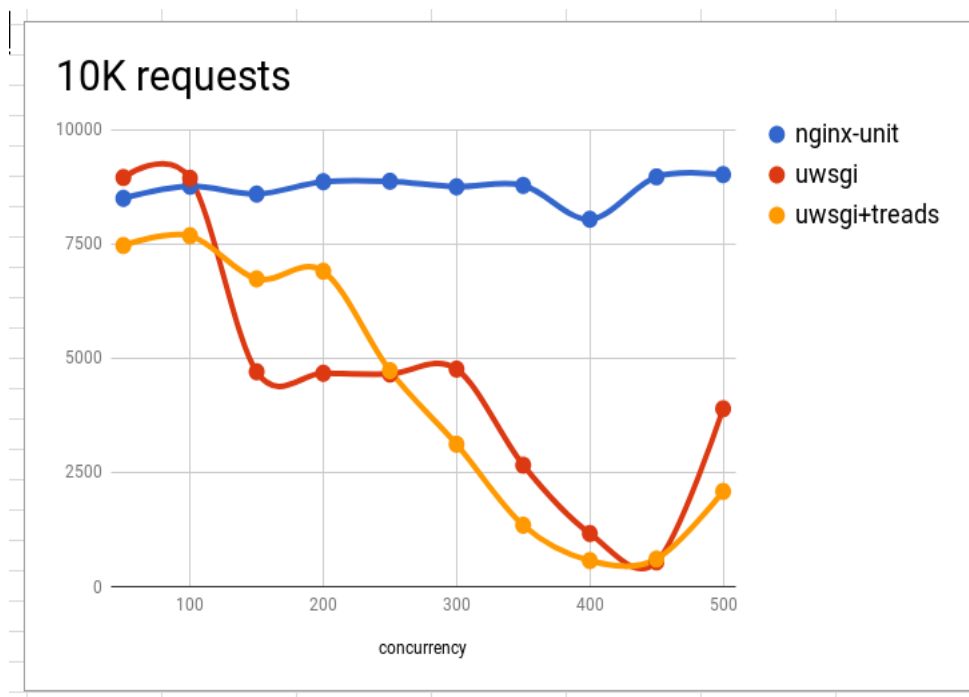
应用语言



多版本



性能



<https://itnext.io/performance-comparison-between-nginx-unit-and-uwsgi-python3-4511fc172a4c>

低延迟

纯应用



NGINX代理



Unit



读取请求



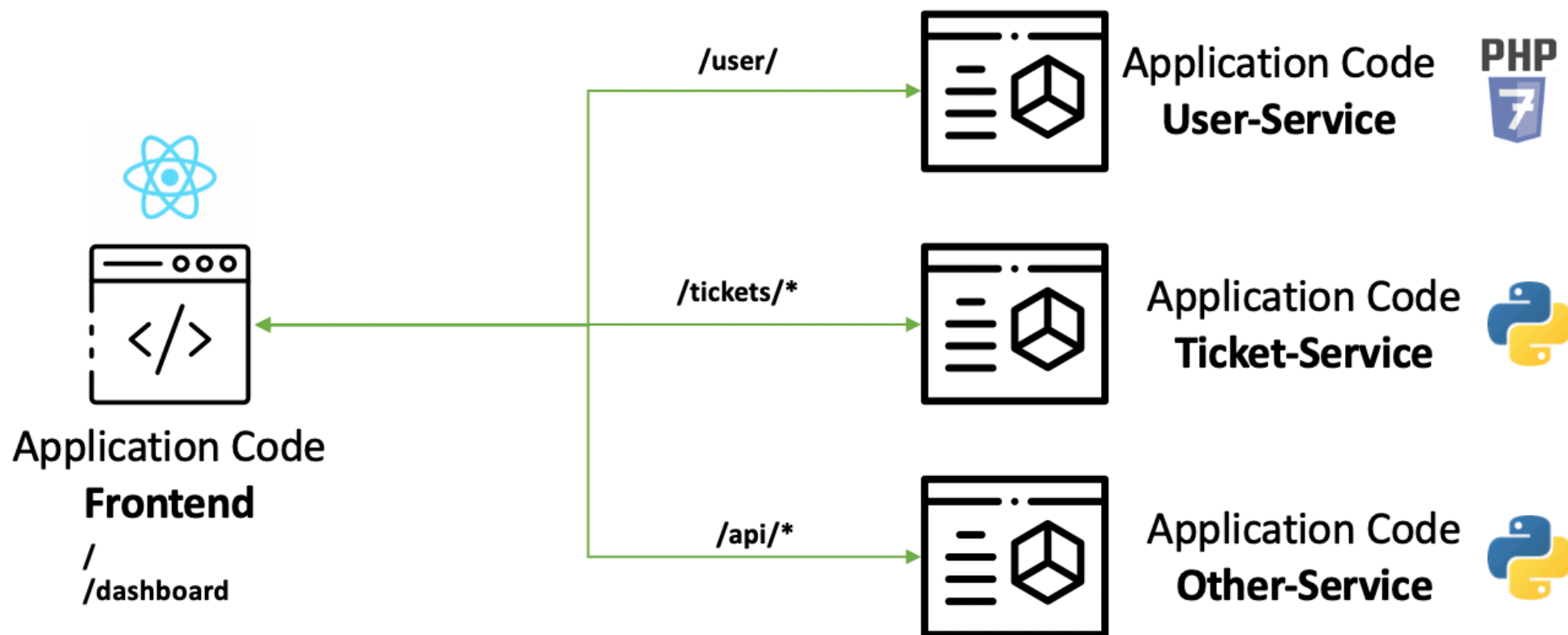
处理请求



响应请求

典型场景

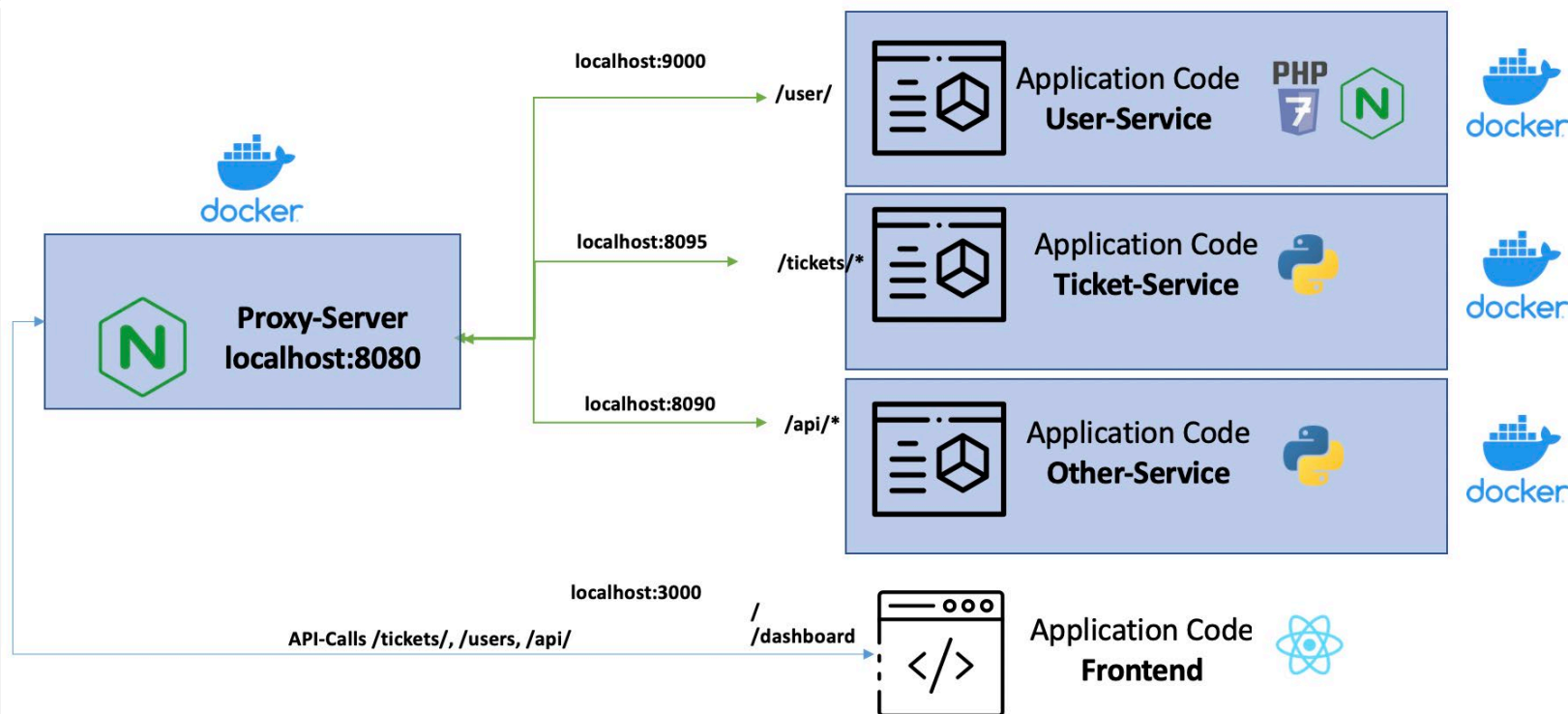
MULTI-TIER APPLICATION (FRONTEND – MULTI BACKENDS)



非Unit方式

local-proxy.conf

```
1 location /user {
2     proxy_pass http://127.1:9000;
3     ....
4 }
5
6 location /tickets {
7     proxy_pass http://127.1:8095;
8     ....
9 }
10
11 location /api{
12     proxy_pass http://127.1:8090;
13     ....
14 }
15
16
```



docker cp . . .



docker exec . . .



docker run -v . . .

chmod 777 . . .



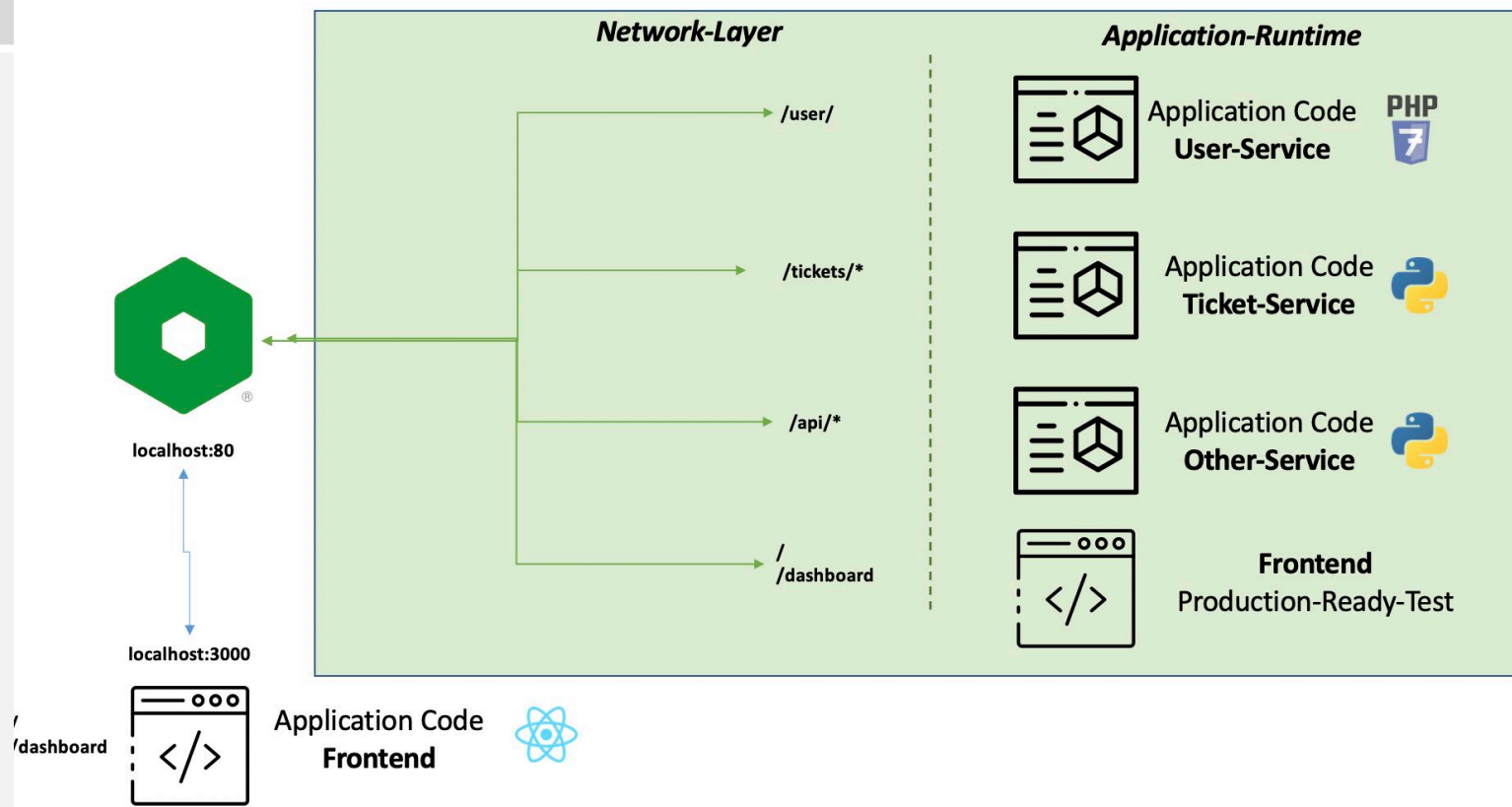
```
docker run ... postgres  
docker run ... dovecot
```

Unit方式

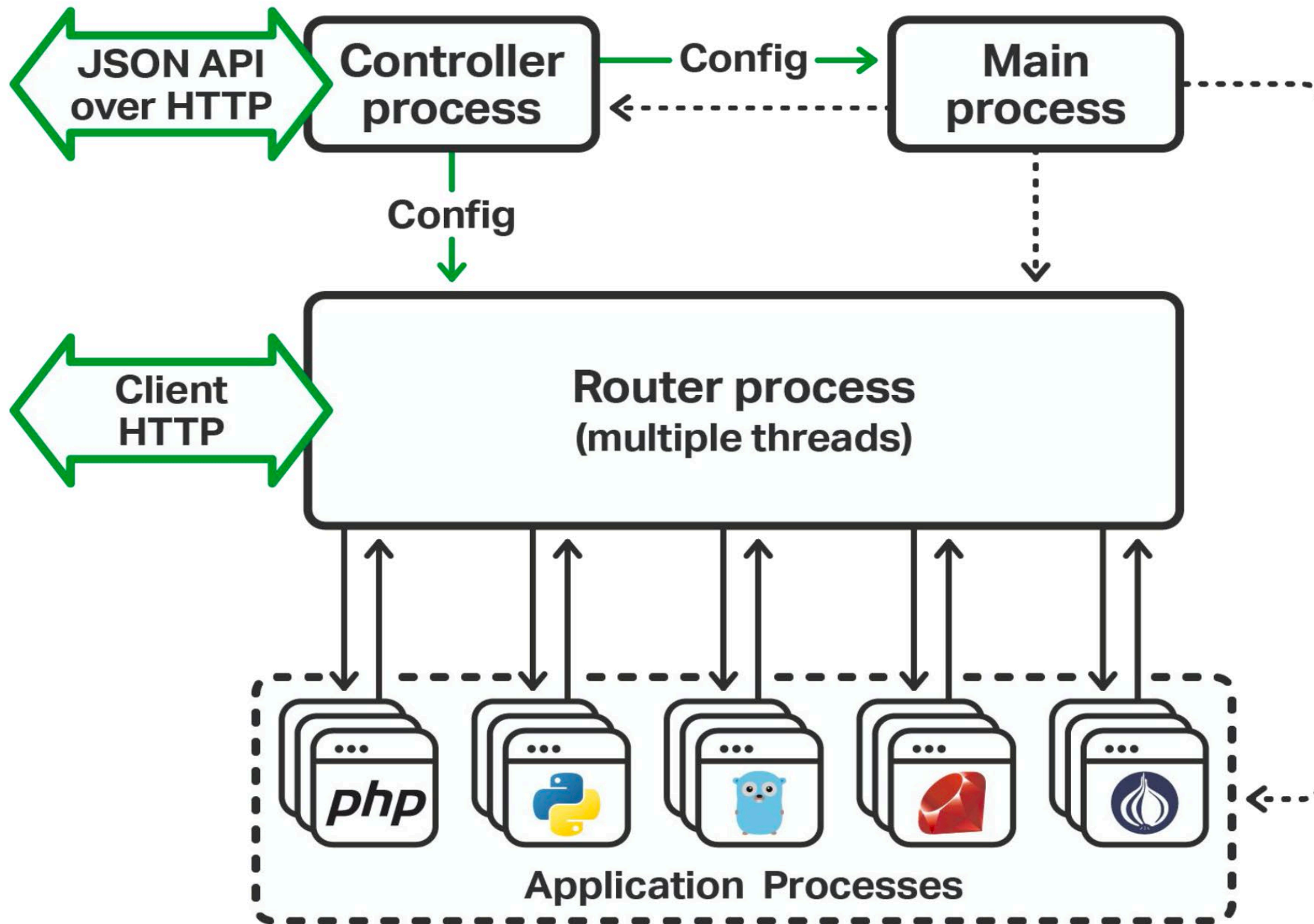
NETWORK- AND APPLICATION-STACK FULLY SEAMLESS INTEGRATED

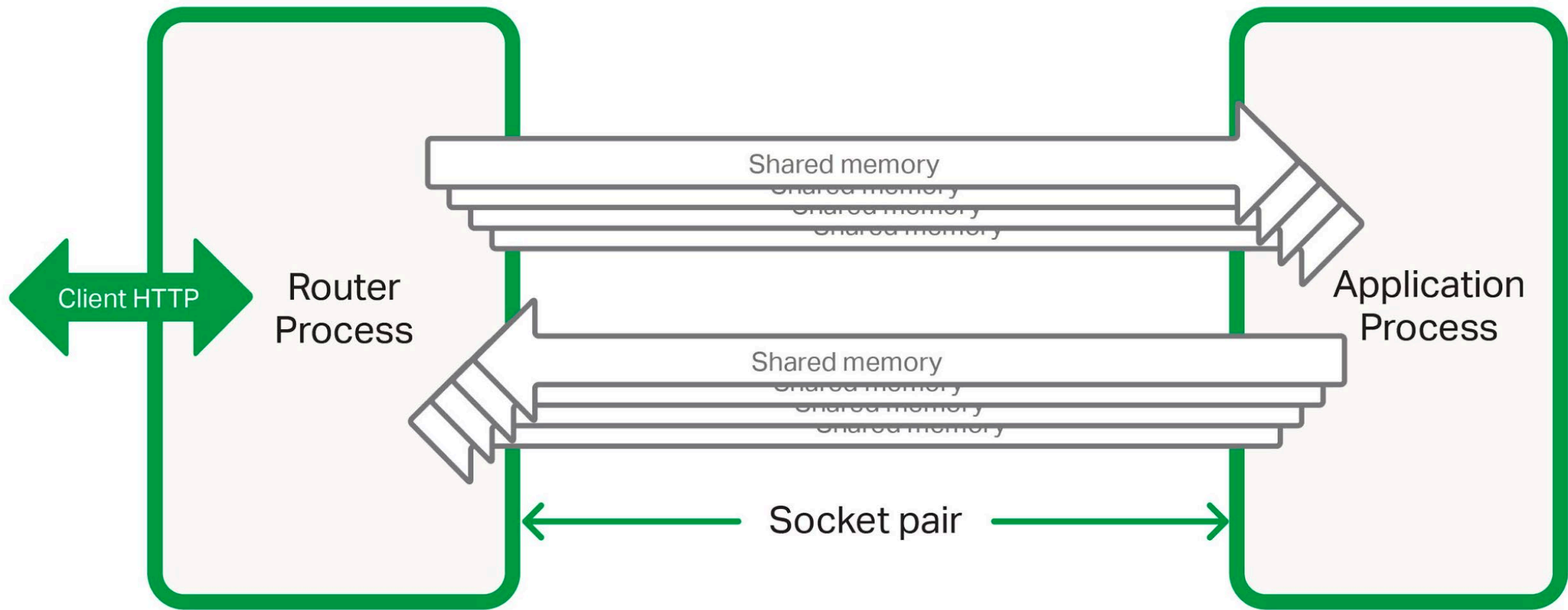
unit.conf.json

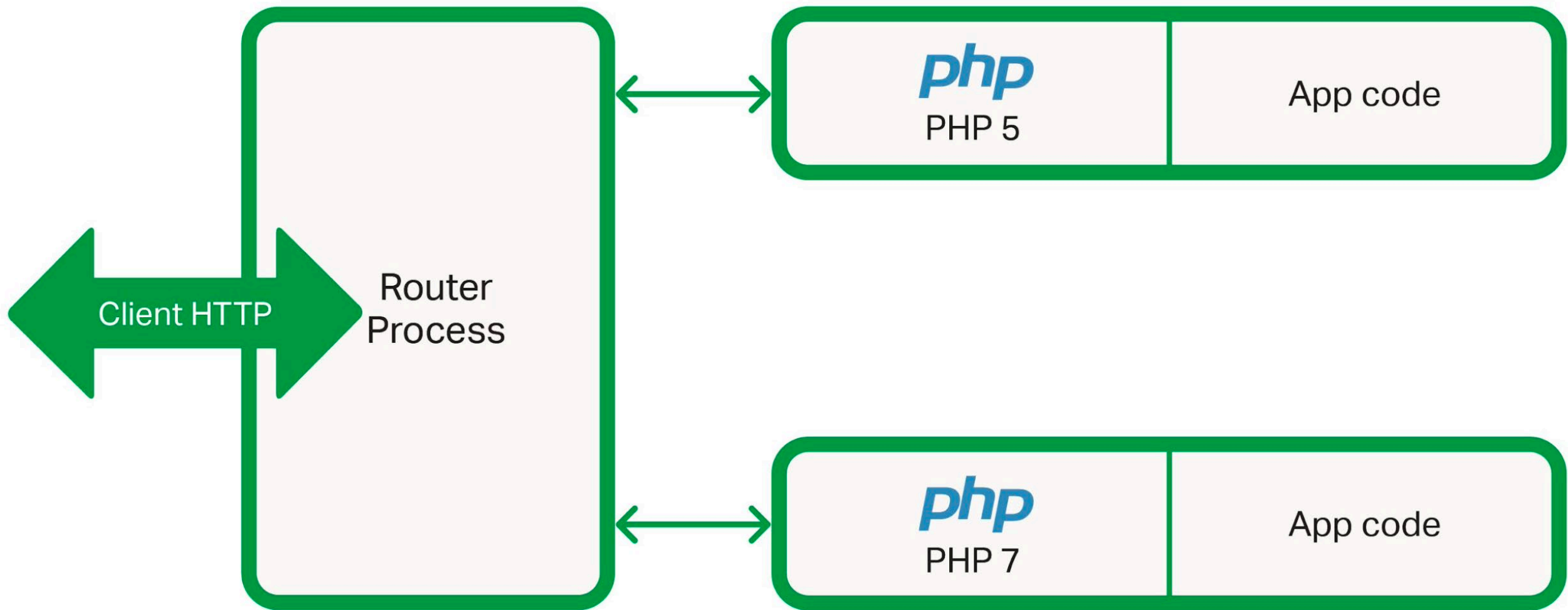
```
1  [
2    {
3      "match": {
4        "uri": ["/ticket", "/ticket/*"]
5      },
6      "action": {
7        "pass": "applications/support-api"
8      }
9    },
10   {
11     "action": {
12       "share":
13         "~/unit-demo-frontend/build",
14       "fallback":
15         {"pass": "applications/support-crm"}
16     }
17   }
18 ]
```

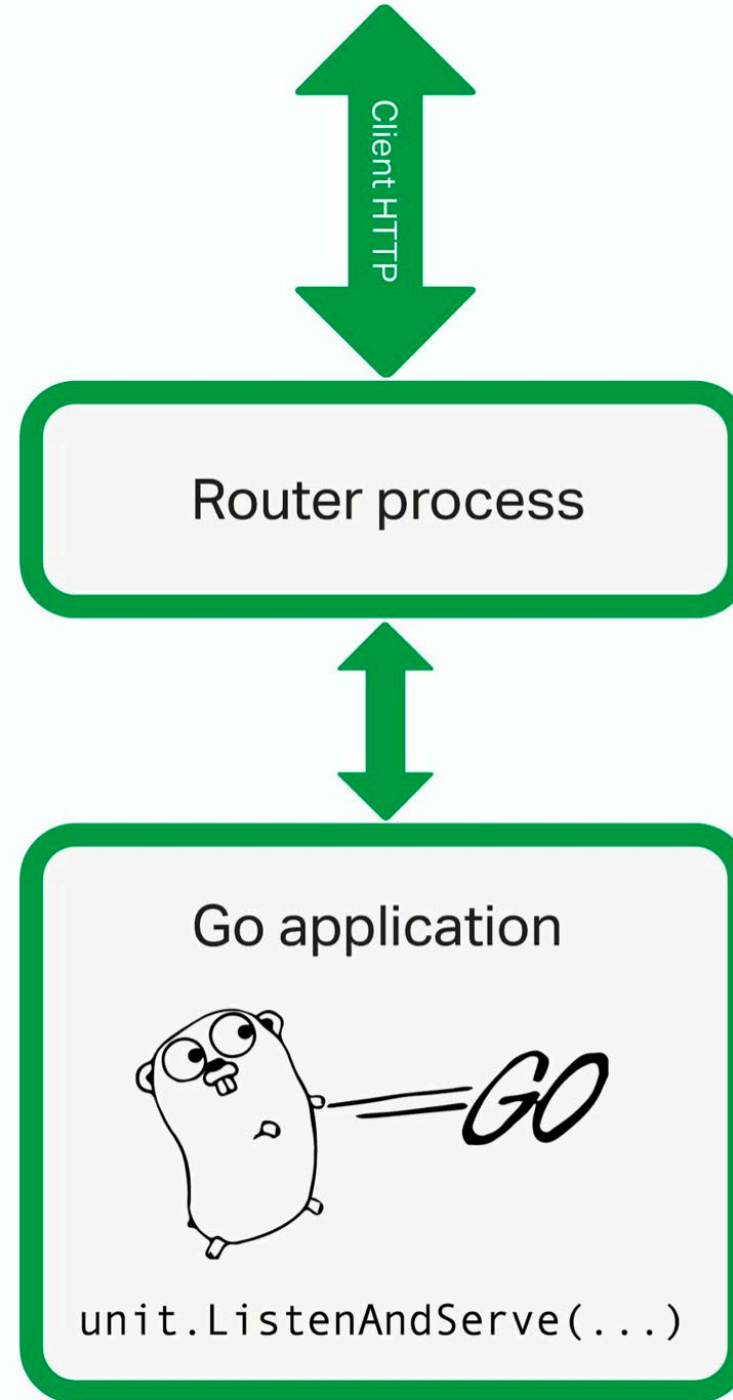
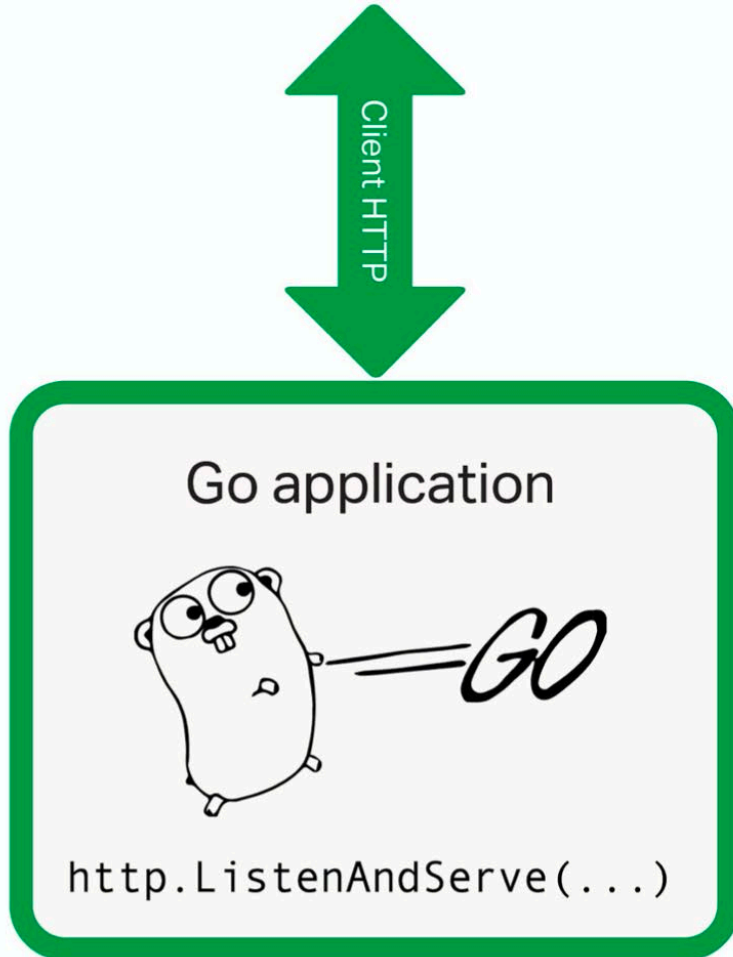


架构









Q&A

资源

unit.nginx.org

- 文档
- 教程

github.com/nginx/unit

- 代码
- 问题
- 计划



NGINX Unit®

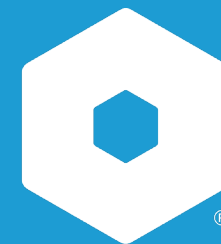
关注我们

nginx.org.cn

- Q&A
- 教程
- 好文

NGINX开源社区

- 微信公众号
- B站
- 知乎



NGINX Unit[®]

