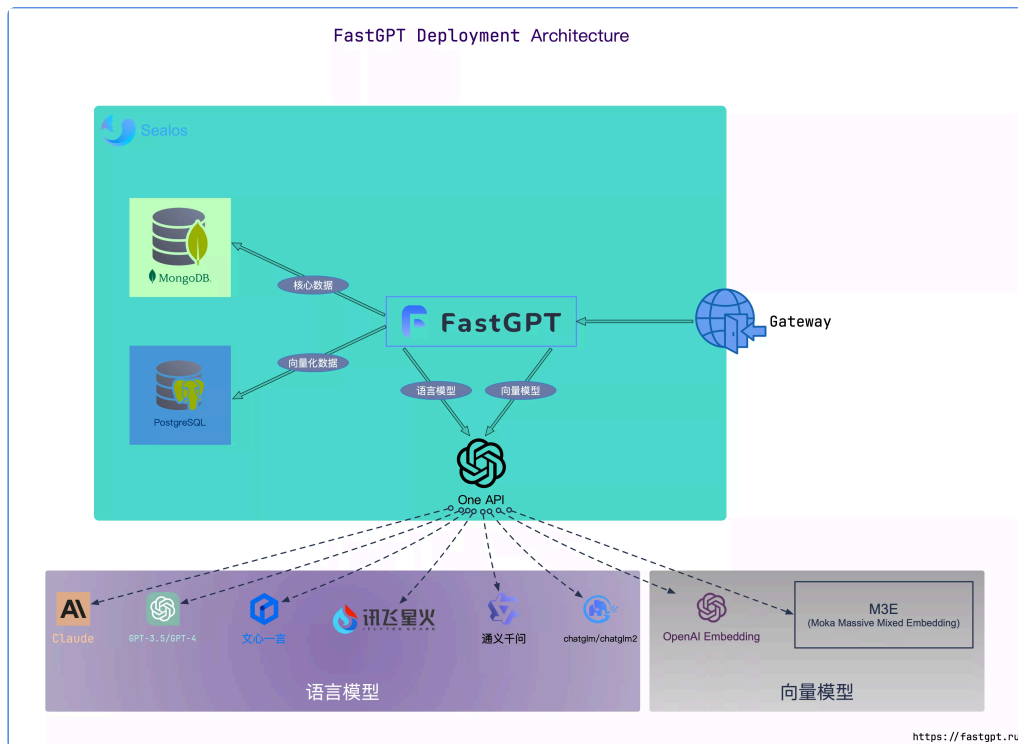


[🏠](#) / [开发与部署指南](#) / Docker Compose 快速部署

📖 Docker Compose 快速部署

使用 Docker Compose 快速部署 FastGPT

部署架构图



MongoDB: 用于存储除了向量外的各类数据

PostgreSQL/Milvus: 存储向量数据

OneAPI: 聚合各类 AI API, 支持多模型调用 (任何模型问题, 先自行通过 OneAPI 测试校验)

推荐配置

PgVector版本

体验测试首选

环境

最低配置 (单节点)

推荐配置

测试	2c2g	2c4g
100w 组向量	4c8g 50GB	4c16g 50GB
500w 组向量	8c32g 200GB	16c64g 200GB

Milvus版本

对于千万级以上向量性能更优秀。

[点击查看 Milvus 官方推荐配置](#)

环境	最低配置（单节点）	推荐配置
测试	2c8g	4c16g
100w 组向量	未测试	
500w 组向量		

zilliz cloud版本

亿级以上向量首选。

由于向量库使用了 Cloud，无需占用本地资源，无需太关注。

前置工作

1. 确保网络环境

如果使用 `OpenAI` 等国外模型接口，请确保可以正常访问，否则会报错：`Connection error` 等。
方案可以参考：[代理方案](#)

2. 准备 Docker 环境

`Linux` `MacOS` `Windows`

```
# 安装 Docker
curl -fsSL https://get.docker.com | bash -s docker --mirror Aliyun
systemctl enable --now docker
# 安装 docker-compose
curl -L https://github.com/docker/compose/releases/download/v2.20.3/docker-compose
chmod +x /usr/local/bin/docker-compose
# 验证安装
docker -v
docker-compose -v
# 如失效，自行百度~
```

开始部署

1. 下载 docker-compose.yml

非 Linux 环境或无法访问外网环境，可手动创建一个目录，并下载配置文件和对应版本的 `docker-compose.yml`，在这个文件夹中依据下载的配置文件运行 `docker`，若作为本地开发使用推荐 `docker-compose-pgvector` 版本，并且自行拉取并运行 `sandbox` 和 `fastgpt`，并在 `docker` 配置文件中注释掉 `sandbox` 和 `fastgpt` 的部分

[config.json](#)

[docker-compose.yml](#)（注意，不同向量库版本的文件不一样）

 所有 `docker-compose.yml` 配置文件中 `MongoDB` 为 5.x，需要用到AVX指令集，部分 CPU 不支持，需手动更改其镜像版本为 4.4.24**（需要自己在docker hub下载，阿里云镜像没做备份）

Linux 快速脚本

```
mkdir fastgpt
cd fastgpt
curl -O https://raw.githubusercontent.com/labring/FastGPT/main/projects/app/data/c

# pgvector 版本(测试推荐，简单快捷)
curl -o docker-compose.yml https://raw.githubusercontent.com/labring/FastGPT/main/
# milvus 版本
# curl -o docker-compose.yml https://raw.githubusercontent.com/labring/FastGPT/mai
# zilliz 版本
# curl -o docker-compose.yml https://raw.githubusercontent.com/labring/FastGPT/mai
```

2. 修改 docker-compose.yml 环境变量

[PgVector版本](#)

[Milvus版本](#)

[Zilliz版本](#)

 无需操作

3. 启动容器

在 `docker-compose.yml` 同级目录下执行。请确保 `docker-compose` 版本最好在2.17以上，否则可能无法执行自动化命令。

```
# 启动容器
docker-compose up -d
# 等待10s, OneAPI第一次总是要重启几次才能连上MySQL
sleep 10
# 重启一次oneapi(由于OneAPI的默认Key有点问题, 不重启的话会提示找不到渠道, 临时手动重启一次解决)
docker restart oneapi
```

4. 打开 OneAPI 添加模型

可以通过 `ip:3001` 访问OneAPI, 默认账号为 `root` 密码为 `123456`。

在OneApi中添加合适的AI模型渠道。 [点击查看相关教程](#)

5. 访问 FastGPT

目前可以通过 `ip:3000` 直接访问(注意防火墙)。登录用户名为 `root`, 密码为 `docker-compose.yml` 环境变量里设置的 `DEFAULT_ROOT_PSW`。

如果需要域名访问, 请自行安装并配置 Nginx。

首次运行, 会自动初始化 `root` 用户, 密码为 `1234` (与环境变量中的 `DEFAULT_ROOT_PSW` 一致), 日志里会提示一次 `MongoServerError: Unable to read from a snapshot due to pending collection catalog changes;` 可忽略。

FAQ

Mongo 副本集自动初始化失败

最新的 `docker-compose` 示例优化 Mongo 副本集初始化, 实现了全自动。目前在 `ubuntu20,22`, `centos7`, `wsl2`, `mac`, `window` 均通过测试。仍无法正常启动, 大部分是因为 `cpu` 不支持 `AVX` 指令集, 可以切换 `Mongo4.x` 版本。

如果是由于, 无法自动初始化副本集合, 可以手动初始化副本集:

- 1 终端中执行下面命令, 创建mongo密钥:

```
openssl rand -base64 756 > ./mongodb.key
chmod 600 ./mongodb.key
# 修改密钥权限, 部分系统是admin, 部分是root
chown 999:root ./mongodb.key
```

- 1 修改 `docker-compose.yml`, 挂载密钥

mongo:

```
# image: mongo:5.0.18
# image: registry.cn-hangzhou.aliyuncs.com/fastgpt/mongo:5.0.18 # 阿里云
container_name: mongo
ports:
  - 27017:27017
networks:
  - fastgpt
command: mongod --keyFile /data/mongodb.key --replSet rs0
environment:
  # 默认的用户名和密码，只有首次允许有效
  - MONGO_INITDB_ROOT_USERNAME=myusername
  - MONGO_INITDB_ROOT_PASSWORD=mypassword
volumes:
  - ./mongo/data:/data/db
  - ./mongodb.key:/data/mongodb.key
```

1 重启服务



```
docker-compose down
docker-compose up -d
```

1 进入容器执行副本集合初始化



```
# 查看 mongo 容器是否正常运行
docker ps
# 进入容器
docker exec -it mongo bash

# 连接数据库（这里要填Mongo的用户名和密码）
mongo -u myusername -p mypassword --authenticationDatabase admin

# 初始化副本集。如果需要外网访问，mongo:27017 。如果需要外网访问，需要增加Mongo连接参数：direct
rs.initiate({
  _id: "rs0",
  members: [
    { _id: 0, host: "mongo:27017" }
  ]
})
# 检查状态。如果提示 rs0 状态，则代表运行成功
rs.status()
```

如何修改API地址和密钥

默认是写了OneAPI的连接地址和密钥，可以通过修改 `docker-compose.yml` 中，fastgpt容器的环境变量实现。

`OPENAI_BASE_URL` (API 接口的地址，需要加/v1) `CHAT_API_KEY` (API 接口的凭证)。

修改完后重启：



```
docker-compose down
docker-compose up -d
```

如何更新版本？

- 1 查看[更新文档](#)，确认要升级的版本，避免跨版本升级。
- 2 修改镜像 tag 到指定版本
- 3 执行下面命令会自动拉取镜像：



```
docker-compose pull
docker-compose up -d
```

- 4 执行初始化脚本（如果有）

如何自定义配置文件？

修改 `config.json` 文件，并执行 `docker-compose down` 再执行 `docker-compose up -d` 重启容器。具体配置，参考[配置详解](#)。

如何检查自定义配置文件是否挂载

- 1 `docker logs fastgpt` 可以查看日志，在启动容器后，第一次请求网页，会进行配置文件读取，可以看看有没有读取成功以及有无错误日志。
- 2 `docker exec -it fastgpt sh` 进入 FastGPT 容器，可以通过 `ls data` 查看目录下是否成功挂载 `config.json` 文件。可通过 `cat data/config.json` 查看配置文件。

可能不生效的原因

- 1 挂载目录不正确
- 2 配置文件不正确，日志中会提示 `invalid json`，配置文件需要是标准的 JSON 文件。
- 3 修改后，没有 `docker-compose down` 再 `docker-compose up -d`，restart是不会重新挂载文件的。

如何检查环境变量是否正常加载

- 1 `docker exec -it fastgpt sh` 进入 FastGPT 容器。
- 2 直接输入 `env` 命令查看所有环境变量。

为什么无法连接本地模型镜像

`docker-compose.yml` 中使用了桥接的模式建立了 `fastgpt` 网络，如想通过 `0.0.0.0` 或镜像名访问其它镜像，需将其它镜像也加入到网络中。

端口冲突怎么解决？

`docker-compose` 端口定义为：`映射端口:运行端口`。

桥接模式下，容器运行端口不会有冲突，但是会有映射端口冲突，只需将映射端口修改成不同端口即可。

如果 `容器1` 需要连接 `容器2`，使用 `容器2:运行端口` 来进行连接即可。

(自行补习 `docker` 基本知识)

relation “modeldata” does not exist

PG 数据库没有连接上/初始化失败，可以查看日志。FastGPT 会在每次连接上 PG 时进行表初始化，如果报错会有对应日志。

- 1 检查数据库容器是否正常启动
- 2 非 `docker` 部署的，需要手动安装 `pg vector` 插件
- 3 查看 `fastgpt` 日志，有没有相关报错

Illegal instruction

可能原因：

- 1 `arm` 架构。需要使用 `Mongo` 官方镜像：`mongo:5.0.18`
- 2 `cpu` 不支持 `AVX`，无法用 `mongo5`，需要换成 `mongo4.x`。把 `mongo` 的 `image` 换成：`mongo:4.4.29`

Operation `auth_codes.findOne()` buffering timed out after 10000ms

`mongo` 连接失败，查看 `mongo` 的运行状态对应日志。

可能原因：

- 1 `mongo` 服务有没有起来（有些 `cpu` 不支持 `AVX`，无法用 `mongo5`，需要换成 `mongo4.x`，可以 `docker hub` 找个最新的 `4.x`，修改镜像版本，重新运行）
- 2 连接数据库的环境变量填写错误（账号密码，注意 `host` 和 `port`，非容器网络连接，需要用公网 `ip` 并加上 `directConnection=true`）
- 3 副本集启动失败。导致容器一直重启。
- 4 `Illegal instruction.... Waiting for MongoDB to start`： `cpu` 不支持 `AVX`，无法用 `mongo5`，需要换成 `mongo4.x`

首次部署，root用户提示未注册

日志会有错误提示。大概率是没有启动 Mongo 副本集模式。

无法导出知识库、无法使用语音输入/播报

没配置 SSL 证书，无权使用部分功能。

登录提示 Network Error

由于服务初始化错误，系统重启导致。

90%是由于配置文件写不对，导致 JSON 解析报错

剩下的基本是因为向量数据库连不上

如何修改密码

修改 `docker-compose.yml` 文件中 `DEFAULT_ROOT_PSW` 并重启即可，密码会自动更新。

 [编辑此页面](#)