

# 离线部署与运维手册

欢迎使用本程序！本文档将为您提供详尽的部署、运行、维护及问题排查指南。

## 目录

- [系统架构概览](#)
- [部署环境要求](#)
- [首次部署步骤](#)
- [日常操作指南](#)
- [健康检查与问题排查](#)
- [高级运维](#)

## 系统架构概览

本应用基于 Docker 容器化技术，由以下几个核心部分组成：

- Docker 环境:** 提供隔离且一致的运行环境，是应用运行的基础。
- 后端服务 (windsim-backend):**
  - 镜像名: `windsimproj-backend:latest`
  - 容器名: `windsim-backend`
  - 职责: 负责核心业务逻辑、数据处理与计算。
- 前端服务 (windsim-frontend):**
  - 镜像名: `windsimproj-frontend:latest`
  - 容器名: `windsim-frontend`
  - 职责: 提供用户交互界面，运行在 `8080` 端口。
- 数据卷 (windsim\_data):**
  - 职责: 用于持久化存储用户上传的文件和计算结果，确保应用停止或重启后数据不丢失。
- 网络 (windsim-net):**
  - 职责: 为前端和后端服务提供一个隔离的内部通信网络。

## 部署环境要求

- 操作系统:** Windows 10 / 11 (64位) 或更高版本。
- 硬件:** 建议至少 8GB 内存，20GB 可用磁盘空间。
- 核心依赖:** **Docker Desktop**。安装程序已包含在 `Docker Desktop Installer.exe` 文件中。

## 首次部署步骤

请严格按照以下步骤操作，确保部署成功。

## 第一步：安装并启动 Docker

这是整个部署过程中最关键的一步。

1. **安装:** 如果您的电脑尚未安装 Docker，请双击运行本文件夹内的 `Docker Desktop Installer.exe`。按照屏幕提示完成安装。
2. **启动:** 安装完成后，务必从“开始”菜单或桌面快捷方式 **启动 Docker Desktop**。
3. **验证 (至关重要):** 启动后，请观察电脑右下角任务栏的 **鲸鱼图标**。请耐心等待，直到该图标 **完全静止** (没有启动中的动画)。这标志着 Docker 环境已准备就绪，可以进行下一步。

## 第二步：加载镜像并启动应用

1. **运行启动脚本:** 回到本文件夹，找到并双击 `start.bat` 文件。
2. **观察进程:**
  - 一个黑色命令行窗口将会打开。
  - 脚本首先会 **检查 Docker 环境** 是否正常。
  - 接着，脚本会自动 **加载后端和前端的镜像** (`windim-backend.tar` 和 `windim-frontend.tar`) 到 Docker 中。
    - **注意:** 首次执行此步骤时，加载过程可能需要 5-10 分钟，具体时间取决于您的电脑性能。这是正常现象，请耐心等待。如果镜像已存在，脚本会自动跳过。
  - 镜像加载完毕后，脚本会自动使用 `docker-compose.yml` 配置启动所有服务。
3. **确认成功:** 当您在窗口中看到以下信息时，代表应用已成功启动：

```
1  =====
2      Application started successfully!
3  =====
4
5      - You can now access the application at: http://localhost:8080
6      - To stop the application, run stop.bat
```

4. **保持运行:** 请不要关闭这个黑色的命令行窗口，您可以将其最小化。关闭此窗口将导致应用停止服务。

## 第三步：访问应用

1. 打开您的浏览器（推荐使用 Chrome, Edge 或 Firefox）。
2. 在地址栏输入并访问：`http://localhost:8080`
3. 现在您应该能看到 WindSim 的界面并开始使用了。

---

## 日常操作指南

### 启动应用

1. 确保 Docker Desktop 正在运行。
2. 双击运行 `start.bat`。

**注意:** 如果电脑重启，您需要先手动启动 Docker Desktop，再运行 `start.bat`。

## 停止应用

1. 双击运行 `stop.bat` 文件。
2. 该脚本会使用 `docker-compose down` 命令安全地停止所有应用容器。
3. 您的所有数据都会被保留在数据卷中，下次启动时可继续使用。

---

## 健康检查与问题排查

如果您遇到无法访问应用或其他异常，可以使用以下工具进行诊断。

### 1. 快速状态检查

- **操作:** 双击运行 `check_status.bat`。
- **功能:** 此脚本会执行以下操作并显示结果：
  1. 检查所有相关容器的当前状态 (运行中/已停止)。正常情况下，`windsim-backend` 和 `windsim-frontend` 都应处于 `up` 状态。
  2. 显示后端服务最近的 50 条日志。
  3. 显示前端服务最近的 50 条日志。
- **应用场景:**
  - 应用无法访问时，检查容器是否正在运行。
  - 功能异常时，查看日志中是否有 `Error` 或 `Exception` 等关键信息。

### 2. 常见问题 (FAQ)

- **问题:** `start.bat` 窗口一闪而过或提示 "Docker is not running"。
  - **原因:** Docker Desktop 未启动或未完全准备好。
  - **解决:** 手动启动 Docker Desktop，并等待其右下角鲸鱼图标静止后再试。
- **问题:** 浏览器访问 `http://localhost:8080` 无法打开页面。
  - **原因:** 服务可能仍在启动过程中，或启动失败。
  - **解决:**
    1. 请在 `start.bat` 提示成功后，再等待 1-2 分钟，因为前端服务可能需要额外时间初始化。
    2. 运行 `check_status.bat`，确认两个容器都处于 `up` 状态。如果不是，请将窗口中的所有内容截图并联系技术支持。

---

## 高级运维

### 1. 后端 Python 环境诊断

如果遇到与数据处理、计算相关的特定错误，可能需要深入诊断后端容器的内部环境。

- **操作:** 双击运行 `diagnose_python.bat`。
- **功能:** 此脚本会临时启动一个后端容器的实例，并执行一系列内部检查，包括：
  1. 操作系统版本。

2. Python 版本。
  3. 已安装的中文字体。
  4. `numpy` 和 `pandas` 库的版本。
  5. 尝试在 Python 中导入 `pandas` 和 `numpy` 并打印版本号（核心功能测试）。
- **应用场景:** 主要用于排查由环境依赖（如特定库版本、字体缺失等）引发的疑难杂症。如果脚本输出任何 `Error` 或 `not found` 信息，请截图并联系技术支持。

## 2. 手动管理

在某些情况下，您可能需要通过命令行手动管理应用。请打开 PowerShell 或 CMD，导航到本文件夹，然后执行：

- **启动服务 (后台模式):**

```
1 | docker-compose up -d
```

- **停止服务:**

```
1 | docker-compose down
```

- **查看实时日志:**

```
1 | docker-compose logs -f
```

- **强制重新构建镜像并启动:**

```
1 | docker-compose up -d --build
```

祝您使用愉快！