开始使用 wp-env

目录

- 快速开始
- 设置 Docker Desktop
- 安装并运行 wp-env
  - 在何处运行 wp-env
  - 卸载或重置 wp-env
- 故障排除
  - 常见错误
  - Ubuntu Docker 设置
- Additional resources

@wordpress/env 包（ `wp-env` ）让你能够为构建和测试插件及主题，设置一个本地 WordPress 环境（站点），无需任何额外配置。

在遵循本指南之前，如果你尚未安装 Node.js 开发工具，请先进行安装。

wp-env 基础示意图

快速开始

1. 根据操作系统指引下载、安装并启动 Docker Desktop。
2. 在终端中运行 `npm -g install @wordpress/env` 以全局安装 `wp-env`。
3. 在终端中进入现有的插件目录、主题目录或新的工作目录。
4. 在终端中运行 `wp-env start` 以启动本地 WordPress 环境。
5. 脚本运行后，访问 `http://localhost:8888/wp-admin`，使用用户名 `admin` 和密码 `password` 登录 WordPress 管理后台。

设置 Docker Desktop

wp-env 工具使用 Docker来创建运行本地 WordPress 站点的虚拟机。 Docker Desktop

应用程序对小型企业、个人使用、教育和非商业开源项目免费。更多信息请参阅其 FAQ。

使用以下链接为您的操作系统下载并安装 Docker Desktop。

适用于 Mac 的 Docker Desktop

适用于 Windows 的 Docker Desktop

适用于 Linux 的 Docker Desktop

如果您使用的是早于 20.04.1 版本的 Ubuntu，请参阅下方额外的 故障排除说明 。

成功安装后，启动 Docker Desktop 应用程序并按照提示进行设置。通常应使用推荐（默认）设置，创建
Docker 帐户是可选的。

安装并运行 wp-env
wp-env 工具用于通过 Docker 创建本地 WordPress 环境。因此，在设置好 Docker Desktop
后，请打开终端并通过运行以下命令安装 wp-env：

npm -g install @wordpress/env

这将全局安装 wp-env，允许从任何目录运行该工具。要确认其已安装且可用，请运行 wp-env --version ，版本号应会显示。

接下来，在终端中导航到现有的插件目录、主题目录或新的工作目录，并运行：

wp-env start

脚本完成后，您可以通过以下地址访问本地环境： http://localhost:8888。使用用户名 admin 和密码 password 登录 WordPress 仪表板。

某些项目（如 Gutenberg）包含其特定的 wp-env 配置，其文档可能会提示您运行 npm run wp-env
start 命令。

有关控制 Docker 环境的更多信息，请参阅 @wordpress/env 包 的自述文件。

在何处运行 wp-env

wp-env 工具几乎可以在任何地方运行。在开发单个插件时使用此脚本，从包含插件的目录运行 wp-env start 可以自动挂载并激活该插件。对于主题开发，从开发主题的目录运行同样有效。

如果在非插件或主题的目录中运行 `wp-env start`，将创建一个通用的 WordPress 环境。脚本会显示以下警告，但如果这是你的意图，可以忽略它。

```
!! 警告：未找到 .wp-env.json 配置文件，且无法确定 'DIR' 是否为 WordPress 安装、插件或主题。
```

你也可以使用 `.wp-env.json` 配置文件来创建适用于多个插件和/或主题的环境。更多配置详情请参阅 @wordpress/env 包的说明文档。

卸载或重置 `wp-env`

如果你需要重新开始或想要移除已安装的内容，这里有一些说明。

- 如果只想重置并清理 WordPress 数据库，请运行 `wp-env clean all`
- 要完全移除特定项目的本地环境，请运行 `wp-env destroy`
- 要全局卸载 `wp-env` 工具，请运行 `npm -g uninstall @wordpress/env`

故障排除

常见错误

使用 `wp-env` 时，常见错误：运行 `docker-compose` 命令时出错

- 检查 Docker Desktop 是否已启动并正在运行。
- 检查 Docker Desktop 仪表板中的日志，重启或移除现有虚拟机。
- 然后尝试重新运行 `wp-env start`。

如果看到错误：`主机已被另一个容器占用`

- 您尝试启动的容器已在运行，或另一个容器正在运行。您可以通过在启动它的目录中运行 `wp-env stop` 来停止现有容器。
- 如果不记得启动 `wp-env` 的目录，可以通过运行 `docker stop $(docker ps -q)` 停止所有容器。这将停止所有 Docker 容器，请谨慎使用。
- 然后尝试重新运行 `wp-env start`。

Ubuntu Docker 设置

如果您使用的 Ubuntu 版本早于 20.04.1，在使用 `wp-env` 设置本地 WordPress 环境时可能会遇到错误。

要解决此问题，请首先按照 Docker 的安装指南进行操作。同时还需要 `docker-compose`，您可能需要单独安装它。请参阅 Docker compose 文档。

安装好 Docker 和 `wp-env` 后，假设 `wp-env` 已全局配置，尝试在某个目录中运行 `wp-env start`。如果遇到此错误：

```
ERROR: Couldn't connect to Docker daemon at http+docker://localhost - is it running?

If it's at a non-standard location, specify the URL with the DOCKER_HOST environment variable.
```

首先，请确保 Docker 正在运行。您可以通过运行 `ps -ef | grep docker` 来检查，它应该返回类似以下的内容：

```
/usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock
```

如果 Docker 没有运行，请尝试通过运行 `sudo systemctl start docker.service` 来启动服务。

如果 Docker 正在运行，那么它没有监听 WordPress 环境尝试通信的方式。尝试添加以下服务覆盖文件以包含监听 `tcp`。请参阅 此 Docker 文档了解如何配置 Docker 守护进程的远程访问。

```
# /etc/systemd/system/docker.service.d/override.conf
[Service]
ExecStart=
ExecStart=/usr/bin/dockerd -H fd:// -H tcp://0.0.0.0:2376
```

从命令行重启服务：

```
sudo systemctl daemon-reload
sudo systemctl restart docker.service
```

重启服务后，设置环境变量 `DOCKER_HOST` 并尝试使用以下命令启动 `wp-env`：

```
export DOCKER_HOST=tcp://127.0.0.1:2376
wp-env start
```

现在您的环境应该已在 `http://localhost:8888` 设置完成。

Additional resources

- @wordpress/env (Official documentation)

- Docker Desktop (Official documentation)
- Quick and easy local WordPress development with wp-env (WordPress Developer Blog)
- wp-env: Simple Local Environments for WordPress (Make WordPress Core Blog)
- `wp-env` Basics diagram (Excalidraw)

□□□ #2
□ □□□ □□□ 2026-03-26 17:08:43 UTC
□ □□□ □□□ 2026-03-26 17:08:43 UTC