

mosn mecha 使用指南

文档修订历史

版本号	作者	备注	修订日期
1.0	诣极 < https://github.com/zonghaishang >	初始版本	2022.5.25

Goland: 一站式mosn代码自动生成器，具备以下特性：

1. 一键创建插件project, 支持编译、调试、编译能力(请参阅自动生成README.md)
2. 支持全部插件生成，协议/拦截器/协议转换插件代码，全场景rpc协议代码生成
3. 支持右键project新增单个插件
4. 最小依赖mosn二进制文件 <<https://yuque.antfin-inc.com/docs/share/4793a223-23bb-495d-9616-f4b52b88f056?#>> , 即可上手mesh
5. 自动生成服务pub、sub代码
6. 自动生成服务调用脚本
7. 自动生成插件使用指南
8. 更多特性，待自行探索

功能增强版本说明：

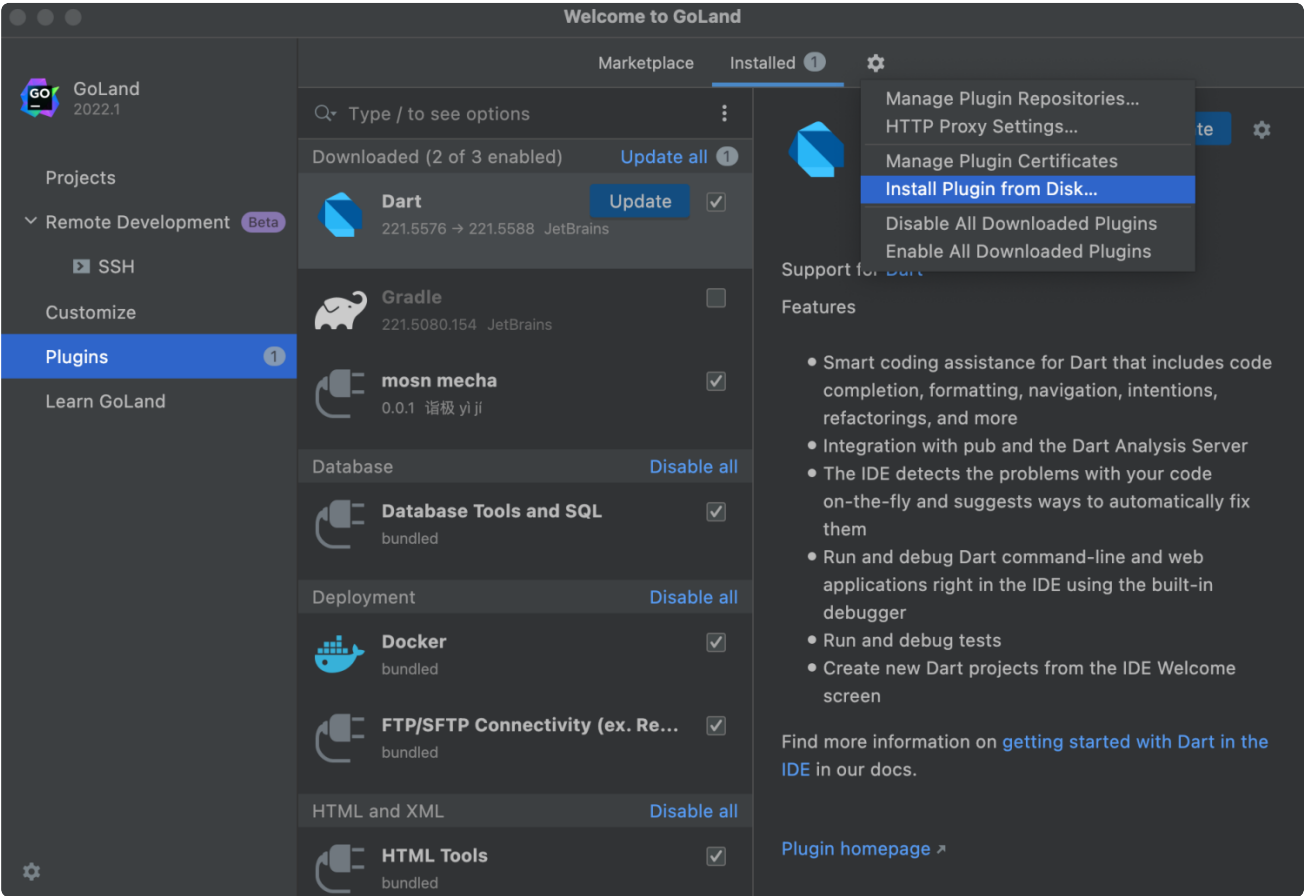
goland : 《[插件部署&升级场景 <https://yuque.antfin-inc.com/yppo6v/vadb69/fymt7kp7dq5drpvz?singleDoc#>](https://yuque.antfin-inc.com/yppo6v/vadb69/fymt7kp7dq5drpvz?singleDoc#>) 》

vs code: 《[vs code 命令行部署与升级 <https://yuque.antfin-inc.com/yppo6v/vadb69/arhqlq04dwa9ion?singleDoc#>](https://yuque.antfin-inc.com/yppo6v/vadb69/arhqlq04dwa9ion?singleDoc#>) 》

1. 新建project

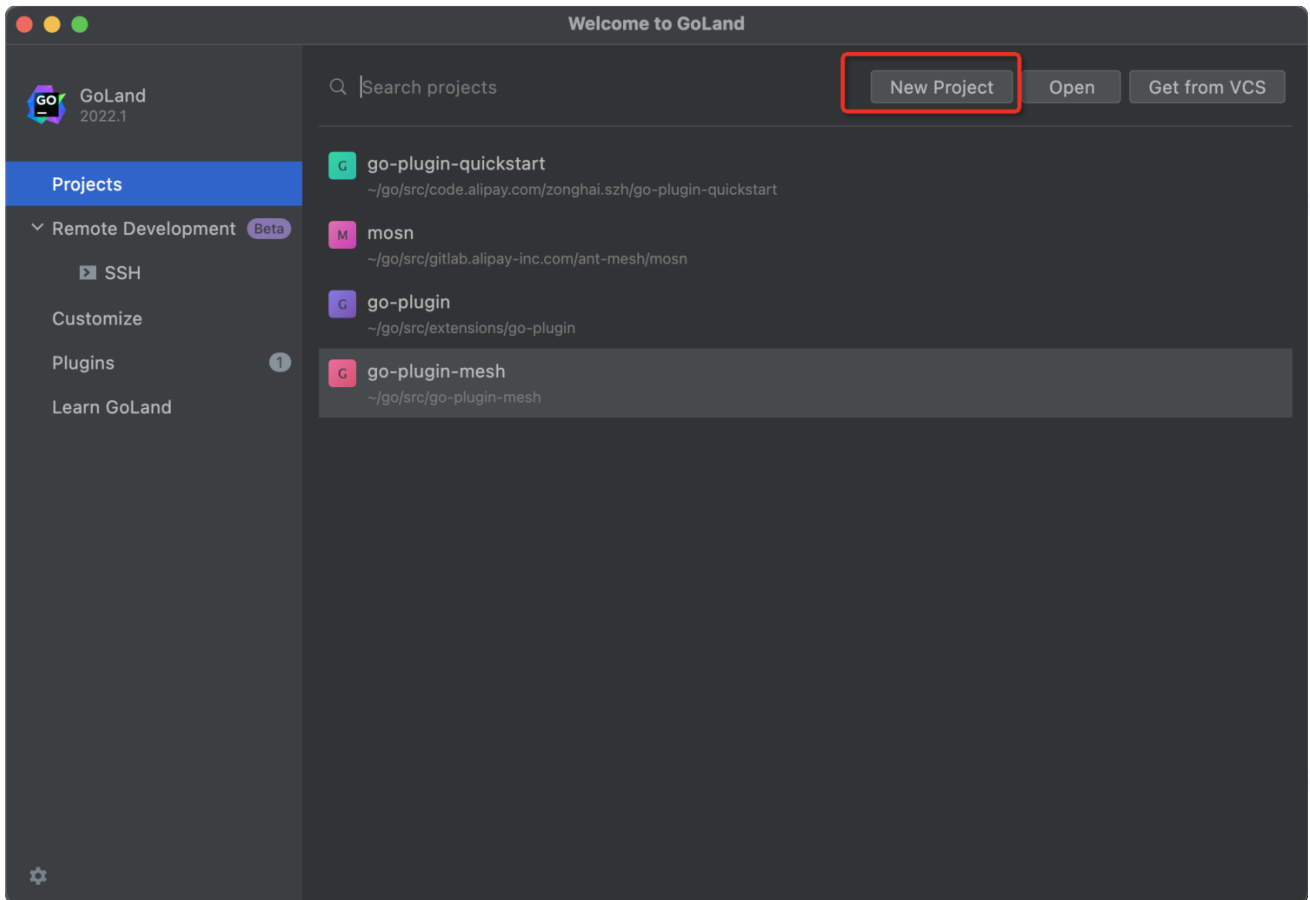
1.1 安装mosn mecha

通过GoLand插件面板，进入Installed的界面右侧工具图标，点击从本地安装(Install Plugin from Disk)，从本文第3章节获取最新mecha插件安装包即可，未来会发布到插件市场，允许在线安装。



H4 ▾ 1.2 新建协议插件

通过Goland创建新的project，会在工程面板左侧选择Mosn Plugin。

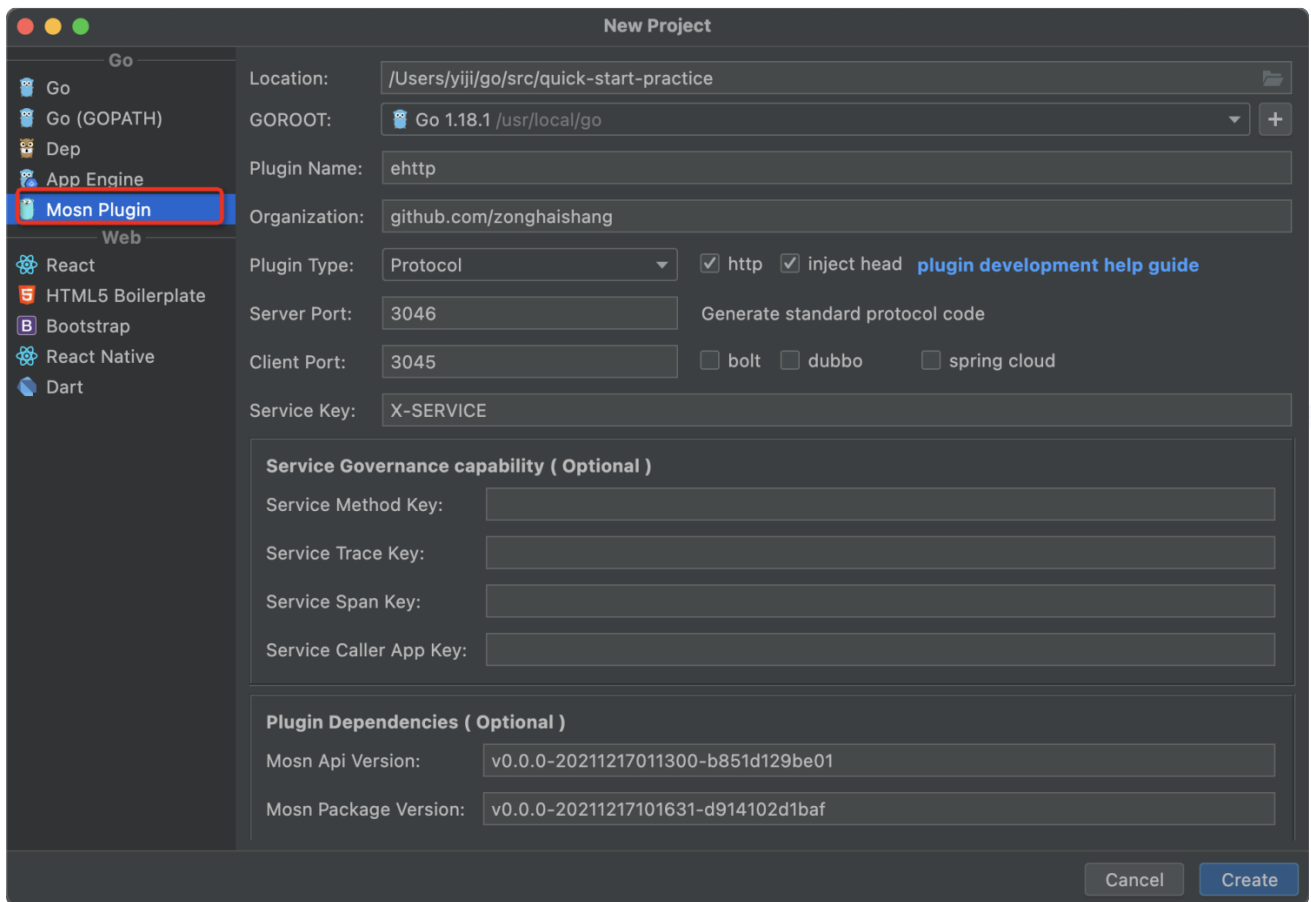


在创建协议插件时，必填项：

- 指定项目路径，项目路径也是项目名称，图示：quick-start-practice
- 选择GO版本，支持创建工程时自动下载go sdk
- 插件名称，用于编译插件，全小写
- 模块名称，和项目名称共同组成go module名称，一般填写域名/命名空间
- 指定监听的服务端和客户端端口
- 服务标识key，改key可以任意定义，mosn用来读取key用于服务发现和服务治理

当创建http协议扩展时，允许自动生成解析http body的代码：

- 勾选http复选框，自动生成http代码和配置
- 勾选inject head, 会自动生成解析http body的拦截器filter代码，方便开发者进行给http 头部种服务标识key



除了http场景，针对tcp私有协议场景，做了灵活的支持：

- 允许生成请求id是字符串场景
- 支持协议长链接/短连接模型
- 支持固定长度+body的协议场景

The screenshot shows the 'New Project' dialog in Mosn Mecha. The sidebar on the left lists various frameworks: Go, Go (GOPATH), Dep, App Engine, Mosn Plugin (selected), Web, React, HTML5 Boilerplate, Bootstrap, React Native, and Dart. The main area contains the following fields and options:

- Location:** /usr/local/go
- GOROOT:** Go 1.18.1
- Plugin Name:** xrpc
- Organization:** github.com/zonghaishang
- Plugin Type:** Protocol (dropdown), with checkboxes for http, bolt, dubbo, and spring cloud.
- Server Port:** 3046
- Client Port:** 3045
- Pool Mode:** Multiplex (dropdown), with a checkbox for 'request id is string type'.
- Codec Type:** FixedLength (dropdown), with fields for 'Fixed Length' (8) and 'Append Prefix' (0).
- Service Key:** service
- Service Governance capability (Optional):**
 - Service Method Key:
 - Service Trace Key:
 - Service Span Key:
 - Service Caller App Key:
- Plugin Dependencies (Optional):**
 - Mosn Api Version: v0.0.0-20211217011300-b851d129be01
 - Mosn Package Version: v0.0.0-20211217101631-d914102d1baf

At the bottom right are 'Cancel' and 'Create' buttons.

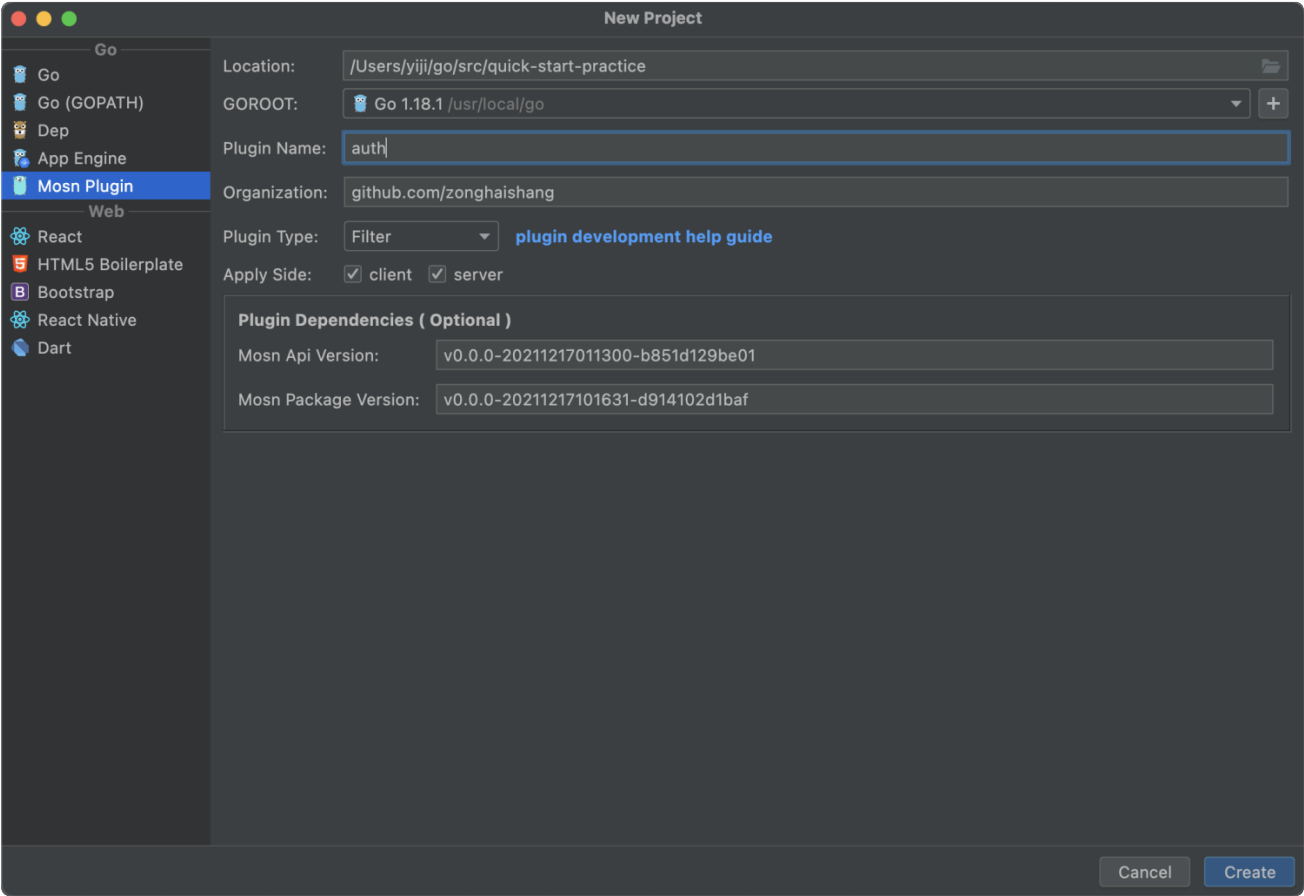
协议插件补充说明：

- 允许自动生成内置协议代码，方便poc等场景做协议转换
- 如果支持更多服务治理能力，可以在解码阶段生成服务治理所需要的key
 - 对应上图 Service Governance capability 的key
- 允许覆盖插件依赖的api和package版本

1.3 新建拦截器插件

在创建拦截器插件时，必填项：

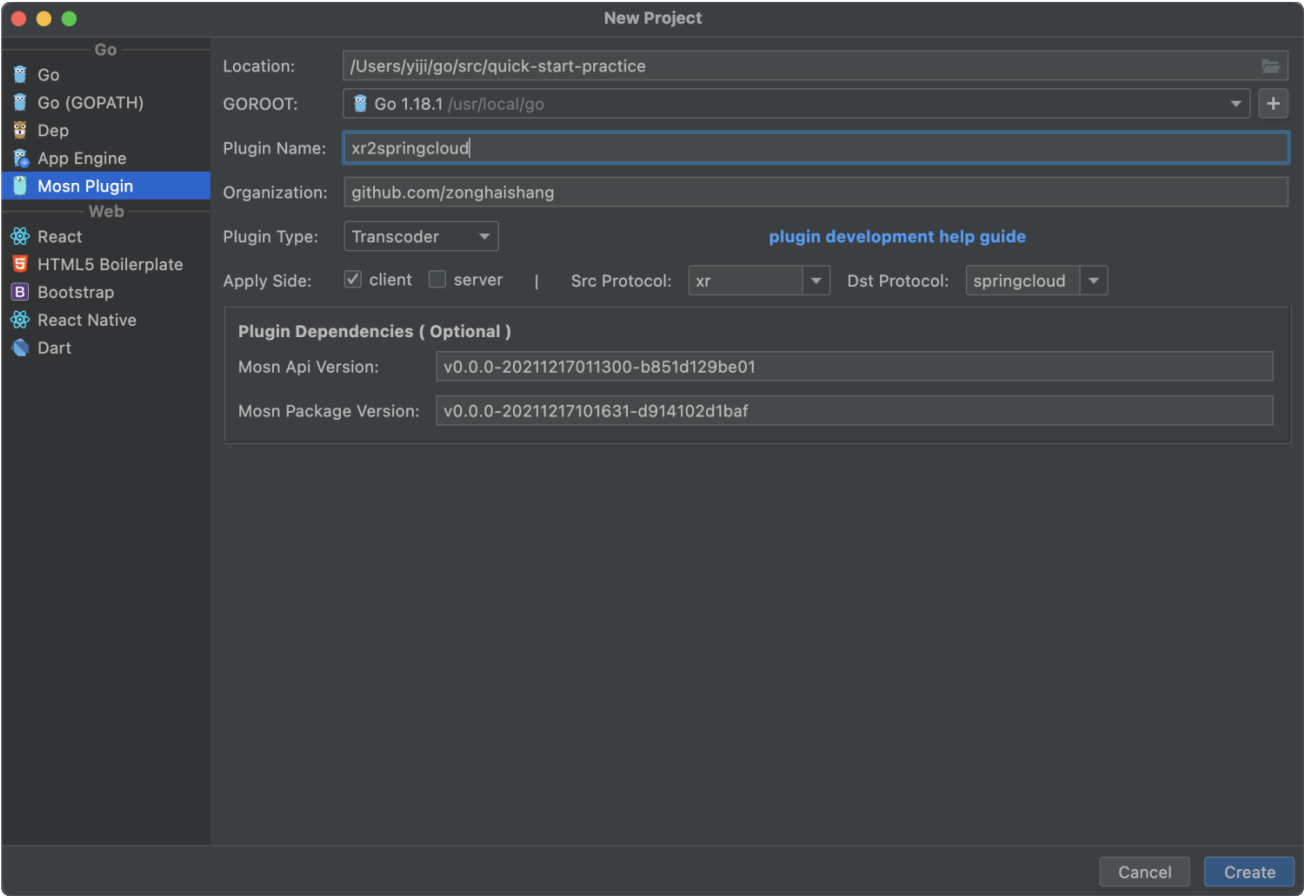
- 指定项目路径，项目路径也是项目名称，图示：quick-start-practice
- 选择GO版本，支持创建工程时自动下载go sdk
- 插件名称，用于编译插件，全小写
- 模块名称，和项目名称共同组成go module名称，一般填写域名/命名空间
- 执行拦截器生效范围，客户端/服务端，允许同时生效



1.4 新建协议转换插件

在创建协议转换插件时，必填项：

- 指定项目路径，项目路径也是项目名称，图示：quick-start-practice
- 选择GO版本，支持创建工程时自动下载go sdk
- 插件名称，用于编译插件，全小写
- 模块名称，和项目名称共同组成go module名称，一般填写域名/命名空间
- 执行拦截器生效范围，客户端/服务端，只允许一侧生效
- 指定协议转换名称，从哪个协议名转换到目标协议名

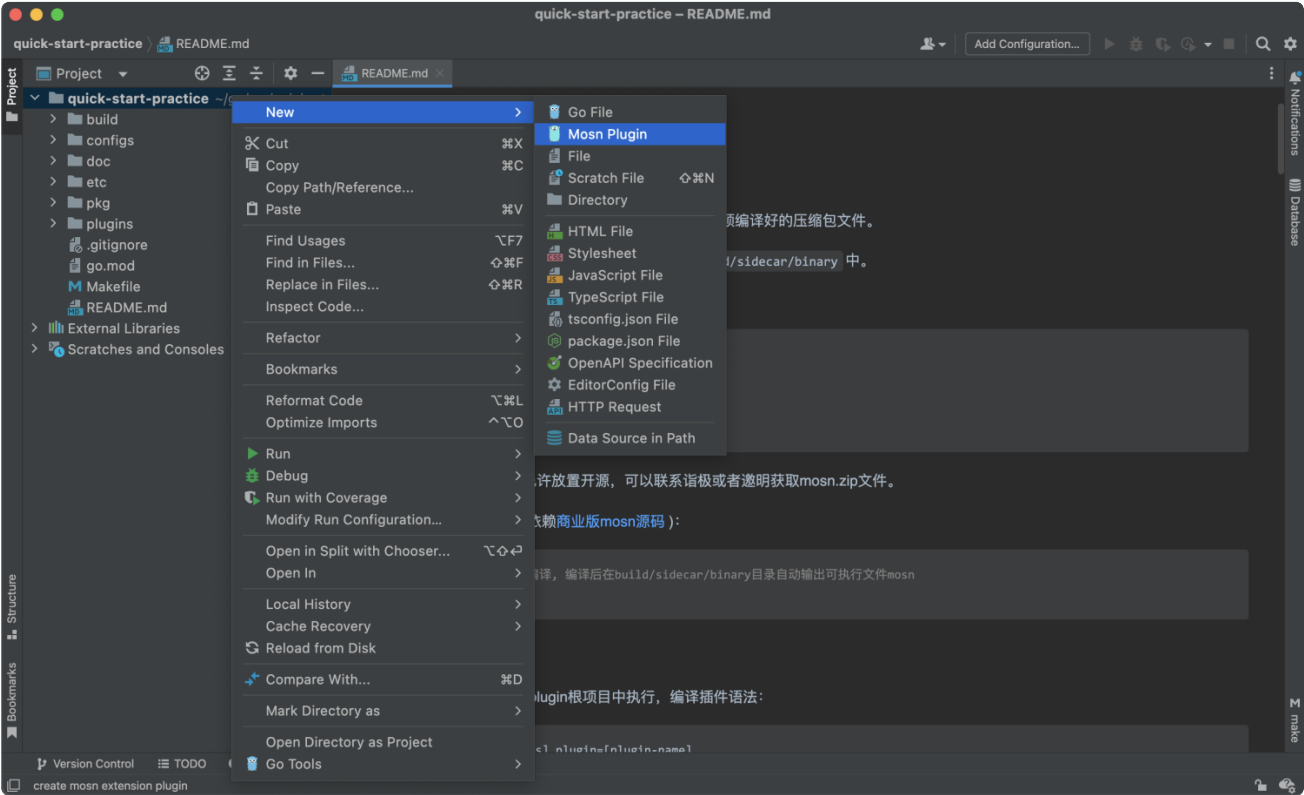


2. 新增插件

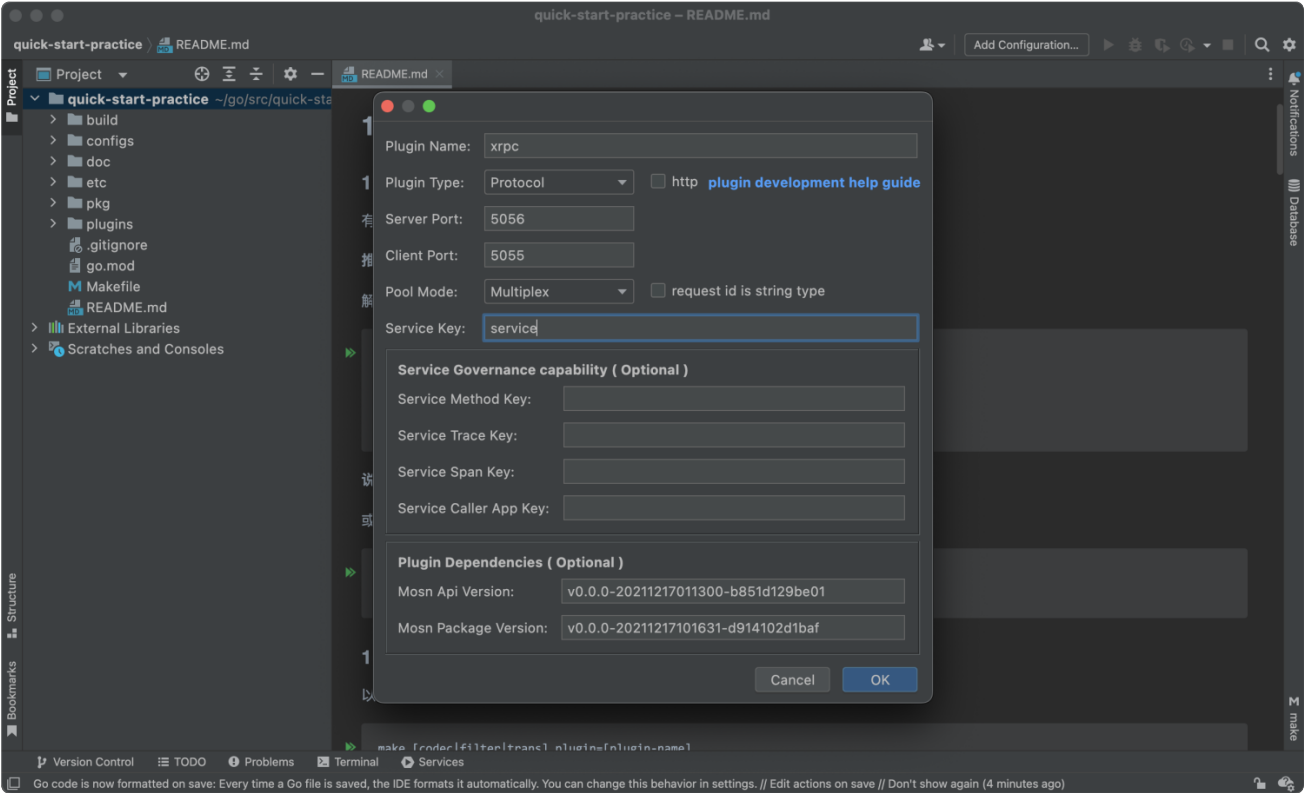
在已经创建好的项目上，新增插件尤其有用。可以在项目project面板右键New-> Mosn Plugin生成插件代码。

目前mecha支持：

- 新增协议插件、拦截器插件、协议转换插件
- 生成协议自动注册和订阅的脚本、服务调用脚本，项目根目录configs中



相比较首次创建工程，新增插件无需选择project位置以及go sdk的版本，mecha会自动获取识别，通过插件类型(Plugin Type) 自由切换待新增的插件类型。



3. 版本发布

3.1 goland 版本发布

*.zip 发布的插件安装包**仅用于GoLand安装**:

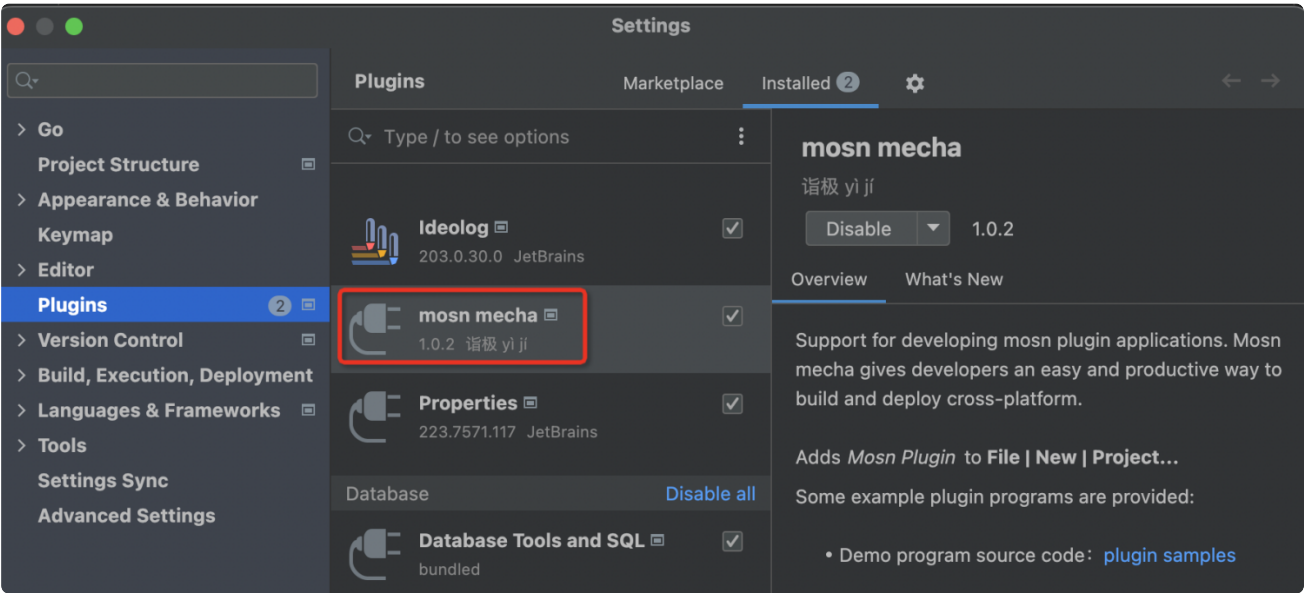
- **注意: 安装版本更新, 已有项目右键-> Mosn Go Plugin -> Refresh 刷新脚手架**

版本号	日期	插件安装包	描述
1.0.3	2023.3.2	 mosn-intellij-1.0.3.zip (9.1 MB)	修复linux启动失败
1.0.2	2023.2.8	 mosn-intellij-1.0.2.zip (9.1 MB)	修复vs code升级project

如何查看已安装的goland mosn intellij 版本?

GoLand 菜单 -> Settings... -> Plugins -> Installed 中找到 mosn mecha

用高版本直接安装重启, 不用管以前的版本也可以



3.2 vs code 版本发布

*.jar 发布的插件安装包**仅用于vs code命令行**:

- 注意：安装版本更新，在终端执行 `mecha fresh -d <project directory>` 刷新脚手架

版本号	日期	插件安装包	描述
1.0.3	2023.3.2	 mosn-intellij-1.0.3-all.jar (9.8 MB)	修复linux启动失败
1.0.2	2023.2.8	 mosn-intellij-1.0.2-all.jar (9.8 MB)	修复vs code升级project

4. 设计实现

产品设计：研发利器– mosn mecha <<https://yuque.antfin-inc.com/docs/share/d777bc77-9460-424f-b7b7-5dc6a9dd137f?#>>>

阿里云课程(推荐)：开放能力培训 <https://help.aliyun.com/document_detail/437892.html?spm=a2c4g.11186623.0.0.2d6a5fcfBweexn>

mecha视频：mosn mecha视频演示 <<https://h5.dingtalk.com/group-live-share/index.htm?encCid=6e1bc66364a266abe212dcb144404c72&liveUuid=af69b2b8-723f-4e52-8c17-f1ff83835af6>>

内部系列课程：Mesh开放技术赋能 <<https://yuque.antfin-inc.com/docs/share/8b0ffd3d-124d-4b0b-b76d-d5f7206c32a7?#>>>

公开系列课程：Mesh开放技术赋能培训课程（优化版） <<https://yuque.antfin-inc.com/docs/share/2bbd7dcc-3727-41f0-ab90-d5d0f6a8d2af?#>>>

mosn mecha项目：mecha源码仓库 <<https://code.alipay.com/ant-extensions/mosn-intellij.git>>

mosn二进制获取：mosn可执行文件 <<https://yuque.antfin-inc.com/docs/share/4793a223-23bb-495d-9616-f4b52b88f056?#>>>