

Dubbo 源码分析 — 【3】 SPI 扩展机制

作者: [zsr251](#)

原文链接: <https://ld246.com/article/1543399950437>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

- 采用 Microkernel + Plugin 模式，Microkernel 只负责组装 Plugin，Dubbo 自身的功能也是通过扩展点实现的，也就是 Dubbo 的所有功能点都可被用户自定义扩展所替换。
- 采用 URL 作为配置信息的统一格式，所有扩展点都通过传递 URL 携带配置信息。

扩展点配置

来源

Dubbo 的扩展点加载从 JDK 标准的 SPI (Service Provider Interface) 扩展点发现机制加强而来。

JDK 标准的 SPI

详细请参考：[Java中SPI机制深入及源码解析](#)

- 主要类：`ServiceLoader` 默认从 `META-INF/services/` 路径下读取文件
- JDK 标准的 SPI 会一次性实例化扩展点所有实现，如果有扩展实现初始化很耗时，但如果没用上也载，会很浪费资源
- 如果扩展点加载失败，连扩展点的名称都拿不到了