Tahiti 构件使用文档

复用构件选择

构件	选择
配置管理 (CM)	Team1
性能监控 (PM)	Team1
许可证 (License)	Team1

构件使用：配置管理 (CM)

我们在 ConfigManager 构件上选择的是 Team1 的构件。

- 优点：代码经过少量重构（大多数为变量类型的修改和接口的修改），构件就可以被运用在我们的项目中。
- 优点：以 Maven 方式集成，无需手工管理依赖。
- 优点：文档和示例全面，没有使用上的问题。
- 缺点：构件只支持 Java Properties 方式的配置管理，且不具有可扩展性（即不能被我们扩展以便接受其他格式配置），因而我们只能将自己的 YAML 配置重写成 Java Properties 格式后才能投入使用。

构件使用：性能监控 (PM)

我们在 PerformanceManager 构件上选择的是 Team1 的构件。

- 优点：构件提供两种集成方式：Maven 或者手动集成 jar 包，方便不同项目根据自身的构建方式选择不同的集成方式。我们项目中采用的是 Maven 方式。
- 优点：使用文档很全面，我们可以方便地根据文档中的示例代码使用 PerformanceManager 替换掉我们的原构件，解耦程度很高，通过简单的初始化，就可以马上投入使用。
- 缺点：PM 模块中日志文件写入模块（LogUtils）提供的是静态全局函数接口，这意味着使用该构件后，项目中只能有一种日志报告输出
- 缺点：模块之间没有做到高内聚、低耦合。

构件选择：许可证 (License)

我们在 License 构件上选择的是 Team1 的构件。

- 优点：文档和使用示例清晰易懂。
- 优点：支持 Maven。
- 优点：对项目进行少量重构即可投入使用。
- 优点：License实例的功能可以自行关闭或开启，开启时可以选择重置计数或者继续计数。
- 缺点：两种类型的限流器合在一个模块中实现，违背了 KISS 原则，也使得它不具有可扩展性。