Atitit usrQC27模块化的规范模块化法 v4 t77

目录

[1. 模块化层级（软件项目 2](#_Toc12264)

[1.1. 子项目》lib程序集》命名空间package机制》类》类文件》方法函数级别》语句 2](#_Toc2863)

[1.2. 源码分支模块化 2](#_Toc27559)

[1.3. 项目大局的模块化 分支》子项目》pkg 2](#_Toc25097)

[2. 常见的模块化机制与实例 2](#_Toc11118)

[2.1. 文件夹树 2](#_Toc11726)

[2.2. 常见的package模块化机制 命名空间namespace机制 2](#_Toc7796)

[2.3. 前缀式模块化 2](#_Toc27362)

[2.4. 程序集 模块化 2](#_Toc22891)

[3. 模块化原则与最佳实践 3](#_Toc17798)

[3.1. 井字形模块化 3](#_Toc129)

[3.2. 核心模块gui，数据库，guice，spring 3](#_Toc7772)

[3.3. 优先模块化垂直切分。。 3](#_Toc12882)

[3.4. 自包含原则 3](#_Toc24590)

[3.5. 参考jdk netclr sdk是个模块化好参照 3](#_Toc17270)

[4. 常见模块化问题 3](#_Toc5910)

[4.1. 什么时候需要子项目化 3](#_Toc6045)

[4.2. Package化 3](#_Toc1283)

[4.3. 数据库编程模块化 3](#_Toc3148)

[4.4. 避免只有大规模水平模块化cotrole service doc模块化 4](#_Toc29360)

[5. java 模块化 4](#_Toc19127)

[5.1. Jdk9模块化 4](#_Toc24891)

[5.2. classloader 4](#_Toc16725)

[5.3. OSGI （面向Java的动态模型系统）  4](#_Toc30664)

[5.4. SPI 简介 4](#_Toc19226)

[6. ref 5](#_Toc15446)

# 模块化层级（软件项目

## 子项目》lib程序集》命名空间package机制》类》类文件》方法函数级别》语句

## 源码分支模块化

## 项目大局的模块化 分支》子项目》pkg

# 常见的模块化机制与实例

## 文件夹树

代码树 ast等

## 常见的package模块化机制 命名空间namespace机制

## 前缀式模块化

常见于一些不支持ns的语言或项目

## 程序集 模块化

* 更加容易轻量级部署
* 改进组件间的依赖管理，引入比Jar粒度更大的Module

# 模块化原则与最佳实践

## 井字形模块化

## 核心模块gui，数据库，guice，spring

## 优先模块化垂直切分。。

## 自包含原则

常用的模块划分，参考，java nodejs net sdk 系列。。

## 参考jdk netclr sdk是个模块化好参照

# 常见模块化问题

## 什么时候需要子项目化

减少主题atiplat\_ee体积..凡是外围的依赖jar很大的一律外移动，比如office系列。Spider系列。Poster系列。

依赖于http系列的外衣

## Package化

或者代码很多的，超过20个的，一律外衣，比如wechat等。。

## 数据库编程模块化

使用前缀机制。。不支持命名空间

## 避免只有大规模水平模块化cotrole service doc模块化

先使用垂直模块化划分

# java 模块化

## Jdk9模块化

替代osgi

## classloader

## ****OSGI （面向Java的动态模型系统）****

OSGi(Open Service Gateway Initiative)技术是Java动态化模块化系统的一系列规范。OSGi一方面指维护OSGi规范的OSGI官方联盟，另一方面指的是该组织维护的基于Java语言的服务（业务）规范。简单来说，OSGi可以认为是Java平台的模块层。

OSGi服务平台向Java提供服务，这些服务使Java成为软件集成和软件开发的首选环境。Java提供在多个平台支持产品的可移植性。OSGi技术提供允许应用程序使用精炼、可重用和可协作的组件构建的标准化原语，这些组件能够组装进一个应用和部署中。

## ****SPI 简介****

SPI 全称为 (Service Provider Interface) ,是JDK内置的一种服务提供发现机制。 目前有不少框架用它来做服务的扩展发现， 简单来说，它就是一种动态替换发现的机制， 举个例子来说， 有个接口，想运行时动态的给它添加实现，你只需要添加一个实现，  
而后，把新加的实现，描述给JDK知道就行啦（通过改一个文本文件即可） 公司内部，目前Dubbo框架就基于SPI机制提供扩展功能

# ref

JAVA9模块化详解（一）——模块化的定义 - 牛初九 - 博客园.html