

模块设计 - 耦合与聚合

一、模块

- 模块

是可以组合、更换的单元，是组成系统、易于处理的基本单元。模块对外表现为实现某功能的输入与输出，内部表示为程序代码与内部数据的结合。
- 目的

系统模块化设计的目的，是降低系统的开发难度，加强系统的可维护性。
- 要求

系统模块化设计的要求是，低耦合，模块间的联系尽可能的少；高内聚，模块内的联系尽可能的多。其中耦合是主导因素，聚合则辅助耦合共同衡量模块独立性。

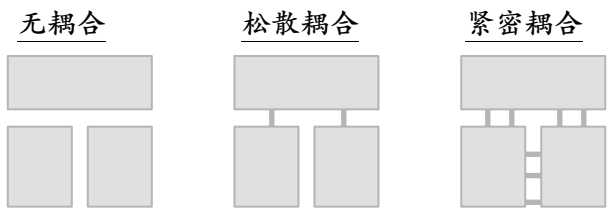
二、耦合

- 耦合度

衡量不同模块间相互依赖的紧密程度。耦合度越高，模块独立性越弱。[耦合是指模块间的联结关系]
- 衡量因素

1.一个模块对另一个模块的引用；2.引用接口的复杂程度；3.一个模块对另一个模块传递的数据量；4.一个模块对另一个模块施加的控制数量。
- 分类

系统模块间耦合程度，可粗略划分为无耦合、松散耦合与紧密耦合。按模块设计的要求，低耦合，系统的模块耦合度关系应尽量设计为松散耦合。



耦合类型 包括非直接耦合、数据耦合、标记耦合、控制耦合、公共耦合与内容耦合。其中非直接耦合与数据耦合是模块化设计中的广泛应用的耦合类型。

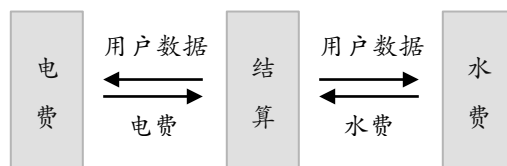
耦合度	耦合类型	模块独立性
低	★ 非直接耦合	强
↓	★ 数据耦合	↑
	标记耦合	
	控制耦合	
	公共耦合	
	内容耦合	
高		弱

无直接耦合 指两个模块无直接关系，仅有高层模块控制调用。模块独立性最强。

数据耦合 一模块调用另一模块时，被调用模块的输入和输出都是简单的数据，属于松散耦合。数据耦合是模块间必要的通信，是不可避免的。



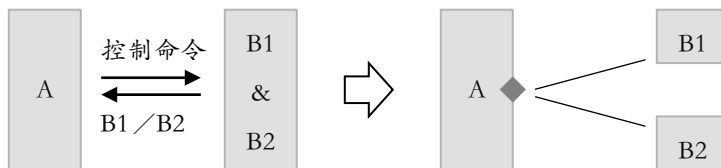
标记耦合 也称特征耦合，指两个模块通过传递数据结构（而不是简单数据，如记录、数组）进行通信，而被调用模块只需要其中的部分数据项。标记耦合传递的数据结构，会使得互不相关的模块建立依赖关系，往往会造成侦错上的困难，因此应尽量将标记耦合转化为数据耦合。



* 数据结构[用户数据]中包含了数据项用电量及用水量

控制耦合 指一个模块将控制信息传递给另一个模块，已控制被调用模块的内部处理逻辑。

从分解角度看，导致控制耦合的主要原因是分解不彻底，被调用的模块不是执行单一的功能。使得控制模块必须知道被控制模块的内部逻辑，增强了依赖。因此，应尽量去除模块间的控制耦合，首先将被控制模块内的功能判定上移到控制模块中进行，然后将被控制模块分解成若干单一功能模块。



公共耦合 指多个模块访问同一公共数据区，如全局数据结构、共享通信、内存公共覆盖区等。公共耦合将导致可理解性降低、可维护性变差、可靠性变差等问题。因此，全局数据类型、公共数据区需要慎重选择使用。

内容耦合 也成为病态耦合，指一个模块与另一个模块的内部属性产生关联（一个模块调用另一模块的内容，即内部数据、程序代码）。内容耦合是最高程度的耦合，务必规避。

原则 尽量使用数据耦合；少使用控制耦合；限制使用公共耦合的范围；坚决避免使用内容耦合。

接口复杂度 衡量因素见表。

接口方式	直接引用	内容耦合
	语句调用	其他耦合
接口数据	数据项作参数	数据耦合
	数据结构、变量名作参数	标记耦合
	开关量作控制变量	控制耦合
	全局数据结构公共数据区	公共耦合
无接口关系		非直接耦合

三、聚合

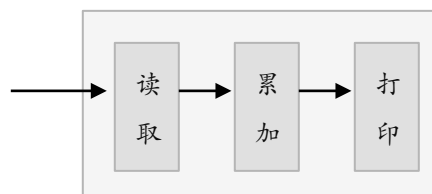
聚合度 衡量一个模块内部个组成部分之间的结合的紧密程度。

聚合类型 包括功能聚合、顺序聚合、通信聚合、过程聚合、时间聚合、逻辑聚合与偶然聚合。

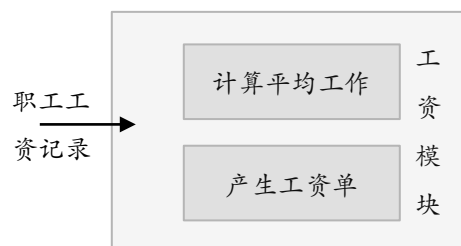
聚合度	聚合类型	模块独立性
低	偶然聚合	弱
↓	逻辑聚合	↓
	时间聚合	
	过程聚合	
	通信聚合	
	顺序聚合	
高	功能聚合	强

功能聚合 指一个模块内所有成分的处理动作全部为了完成某一个功能，且只执行一个功能，缺一不可。

顺序聚合 指一个模块中各成分的工作是有次序的，能完成多个操作且前一个操作处理的输出是下一个操作的输入，成分之间的关系较为紧密。



通信聚合 指一个模块内各部分使用相同的输入数据，或产生相同的输出结果。



过程聚合 指一个模块内个处理成分的动作各不相同，彼此没有什么关系，这些成分却收到一个控制流的支配，并由控制流决定它们的执行次序。与顺序聚合不同，过程聚合的成分并没有固定的执行次序。

时间聚合 指一个模块内的各成分必须在同一时间内执行，这些功能只因为时间因素关联在一起。如系统初始化模块。

逻辑聚合 指一个模块执行若干种类似的功能（仅在逻辑上有相似的功能，没有意义上的相同），每次调用该模块时，由控制参数决定执行哪一项模块功能。增加了模块间耦合。

偶然聚合 模块内各部分无联系。