



结构化开发方法的成与败

北京理工大学计算机学院
金旭亮

开始



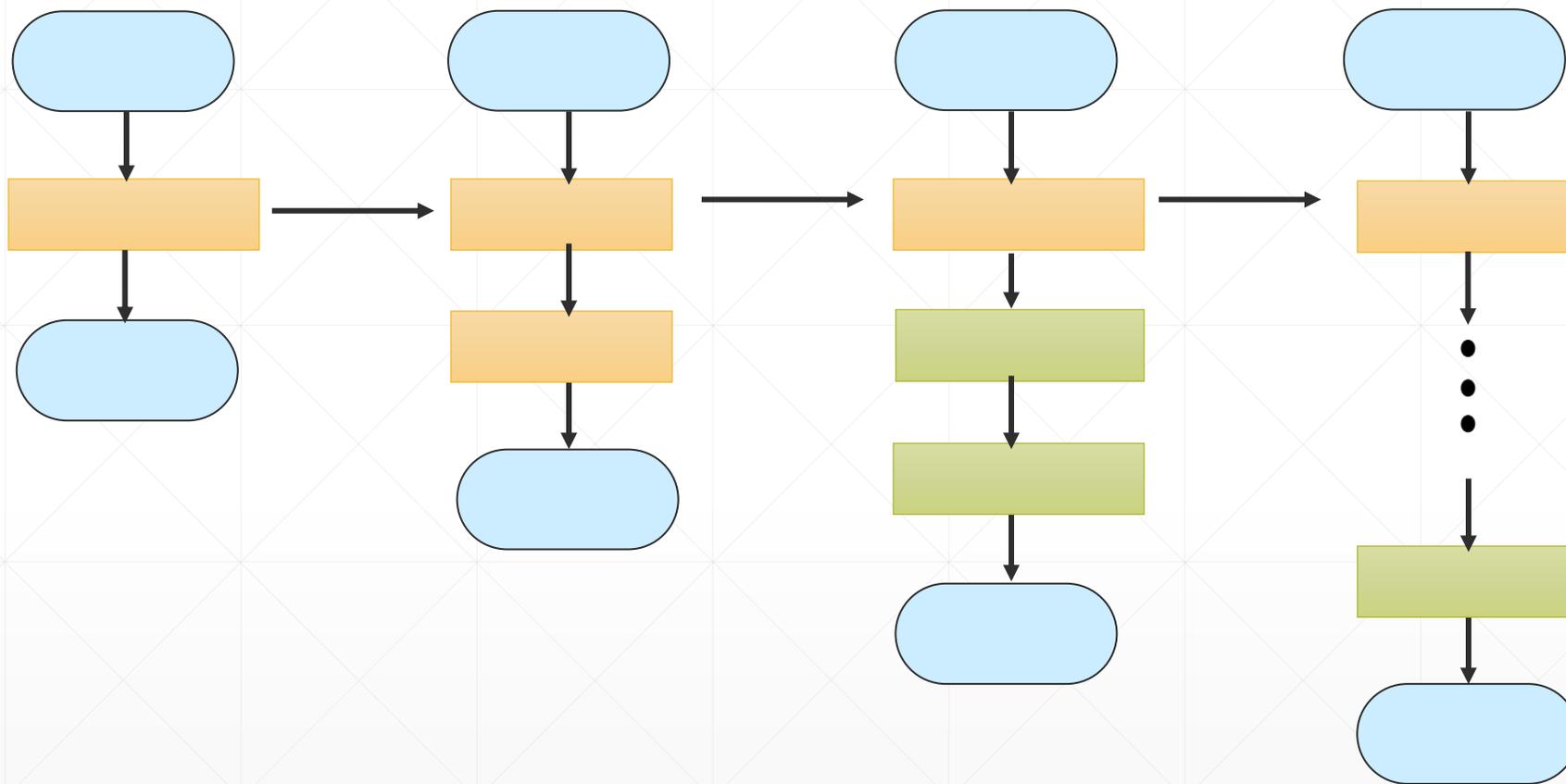
程序构造块

不断分解和细化

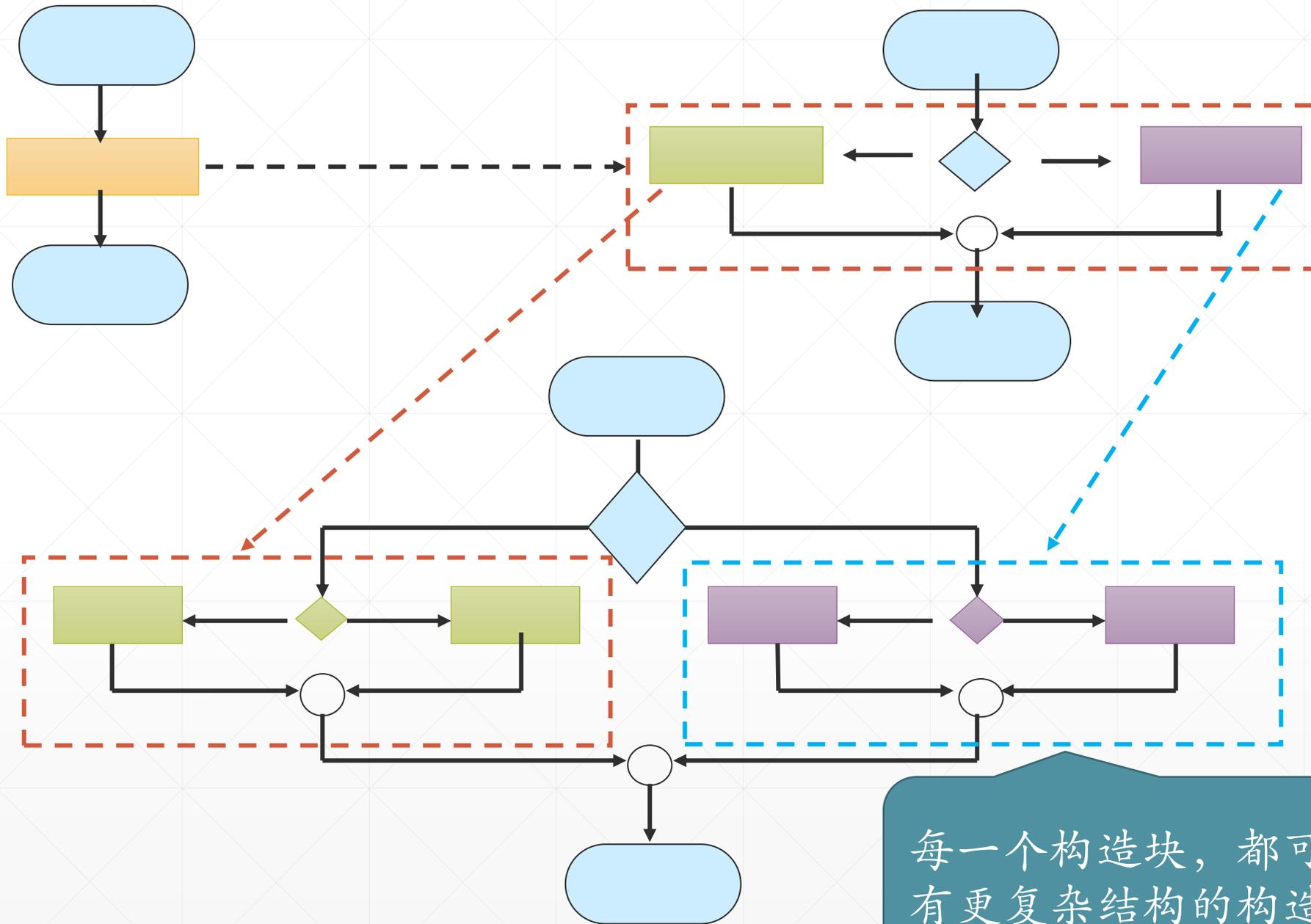


结束

从最简单的流程图出发.....



不断迭代替换.....



每一个构造块，都可以使用拥有更复杂结构的构造块替换

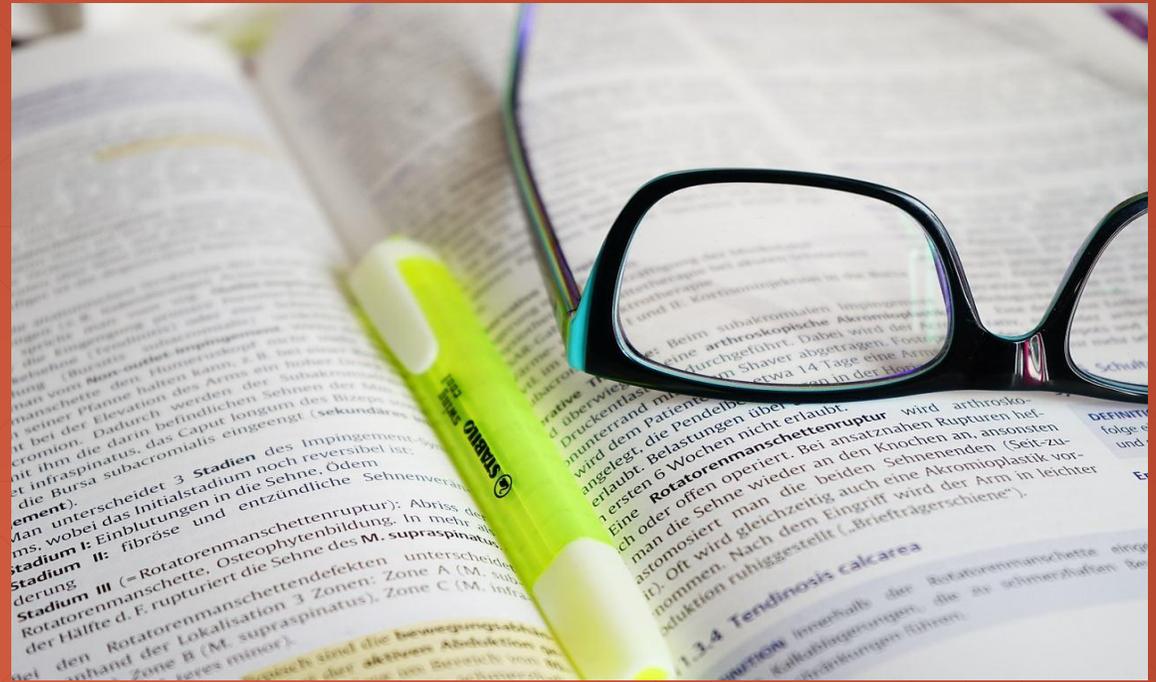
结构化开发的具体实施方式



调用已有的函数模块

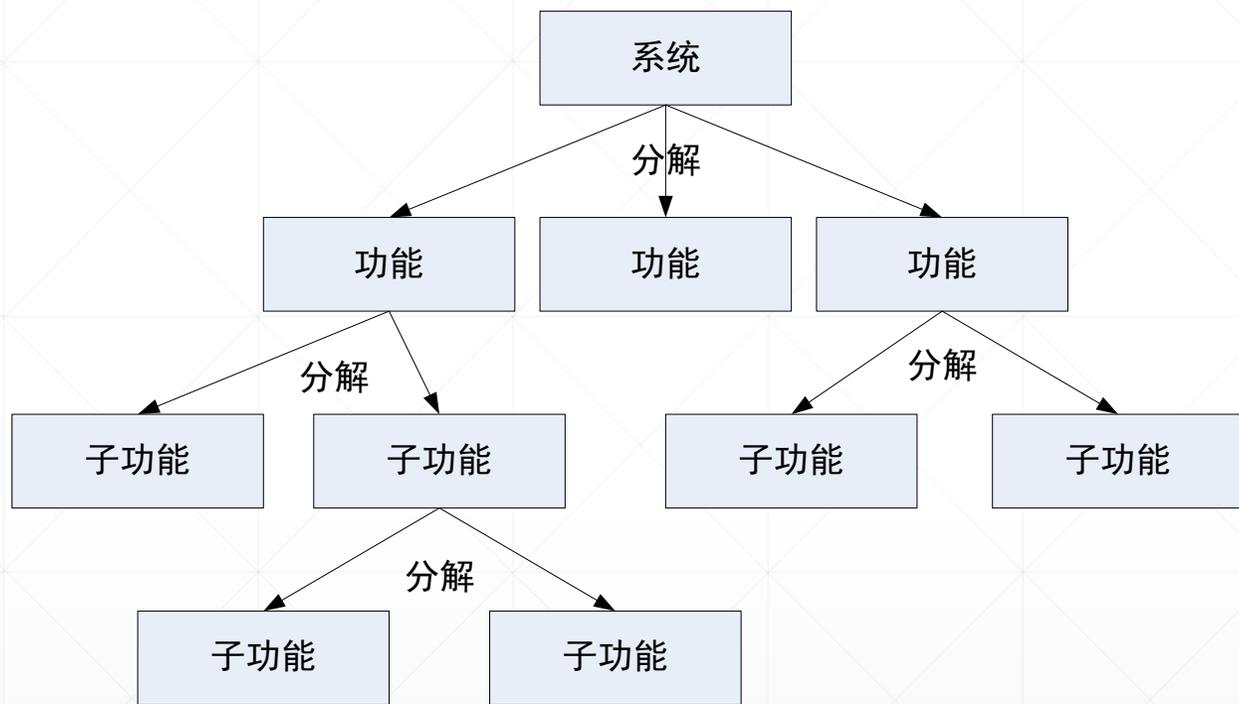
开发自己的函数模块

将所有模块装配起来



开发过程中如何进行结构化分析?

“功能分解”法



细化分解到每个子功能都可以使用一个函数（或若干个函数配合）实现。

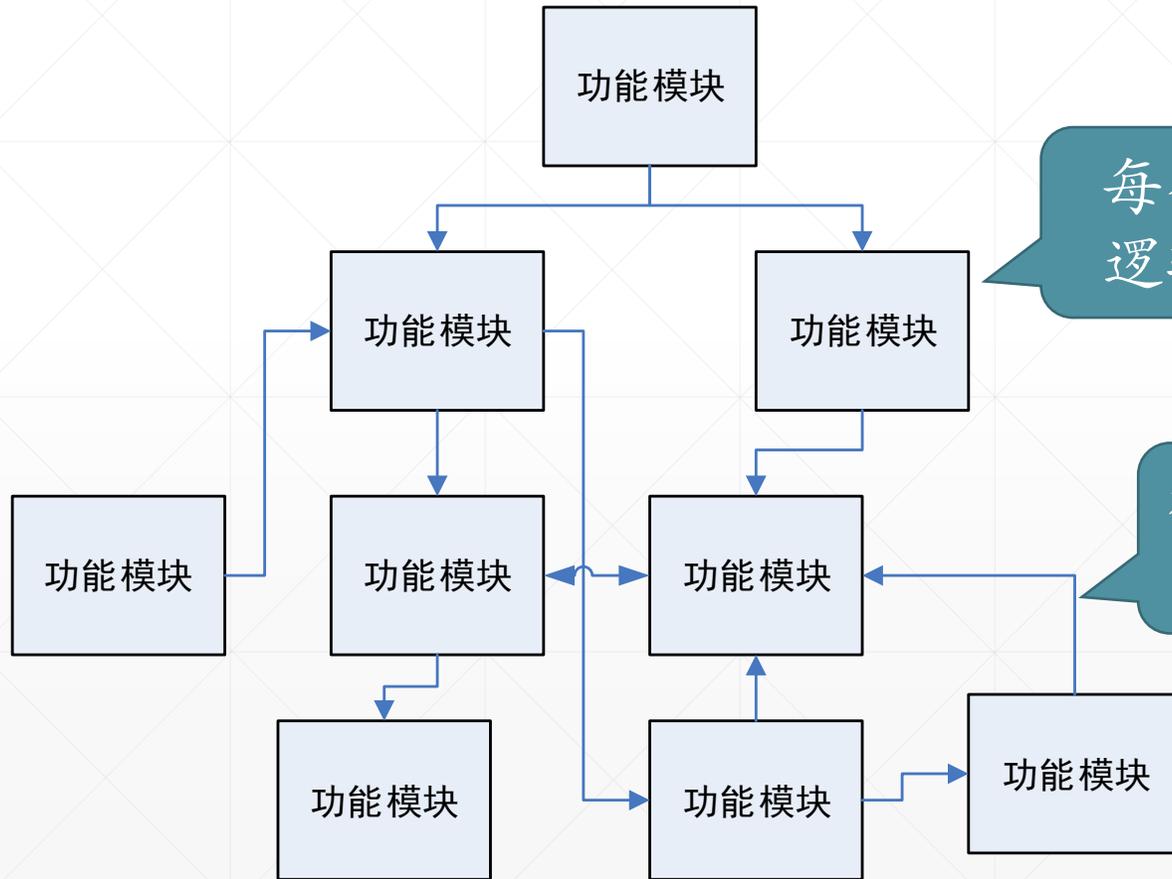
逐步细化分解的结果：



得到一棵庞大无比的大树!

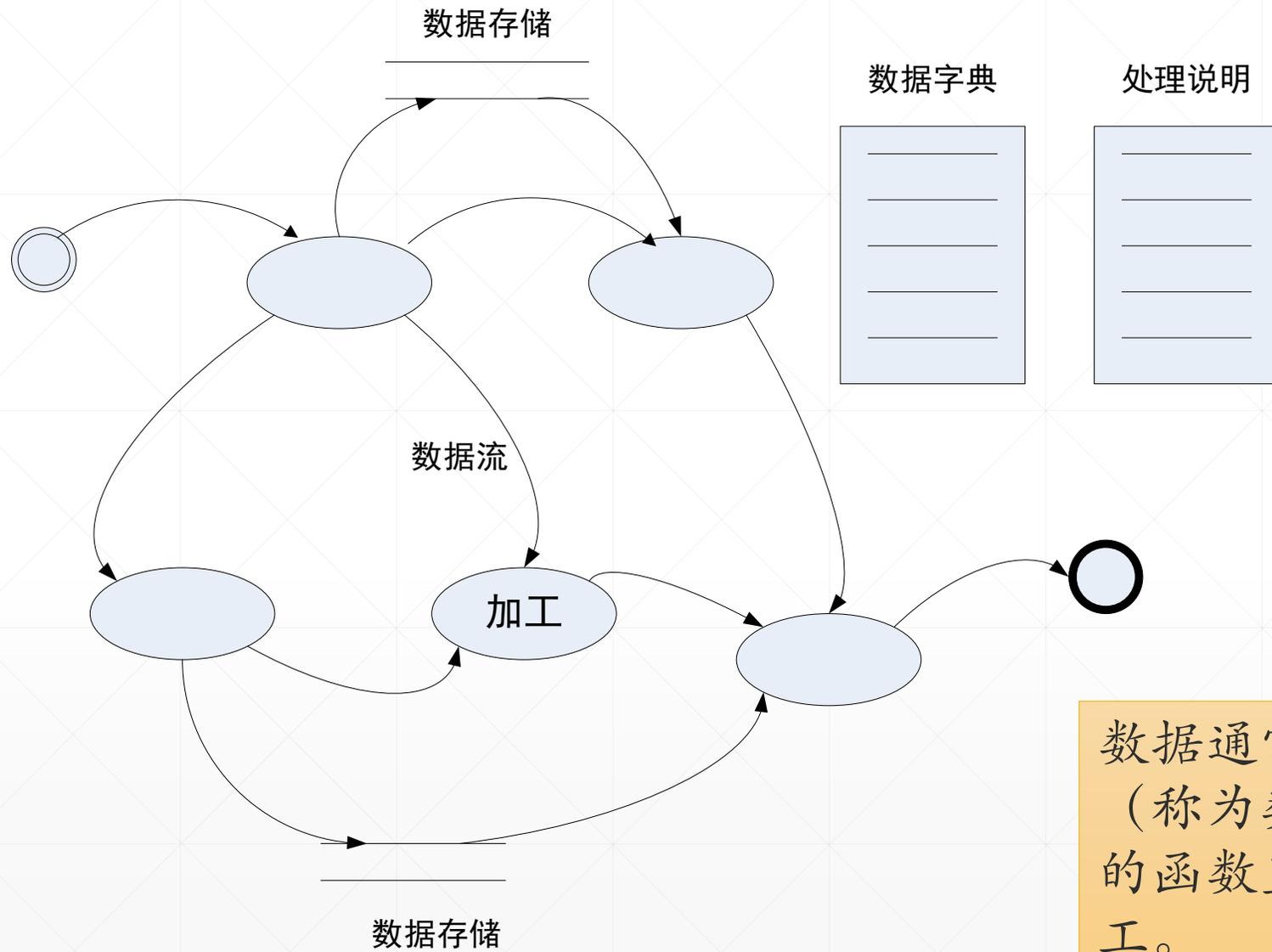
使用功能分解法开发软件

将软件系统划分为多个功能模块和接口函数



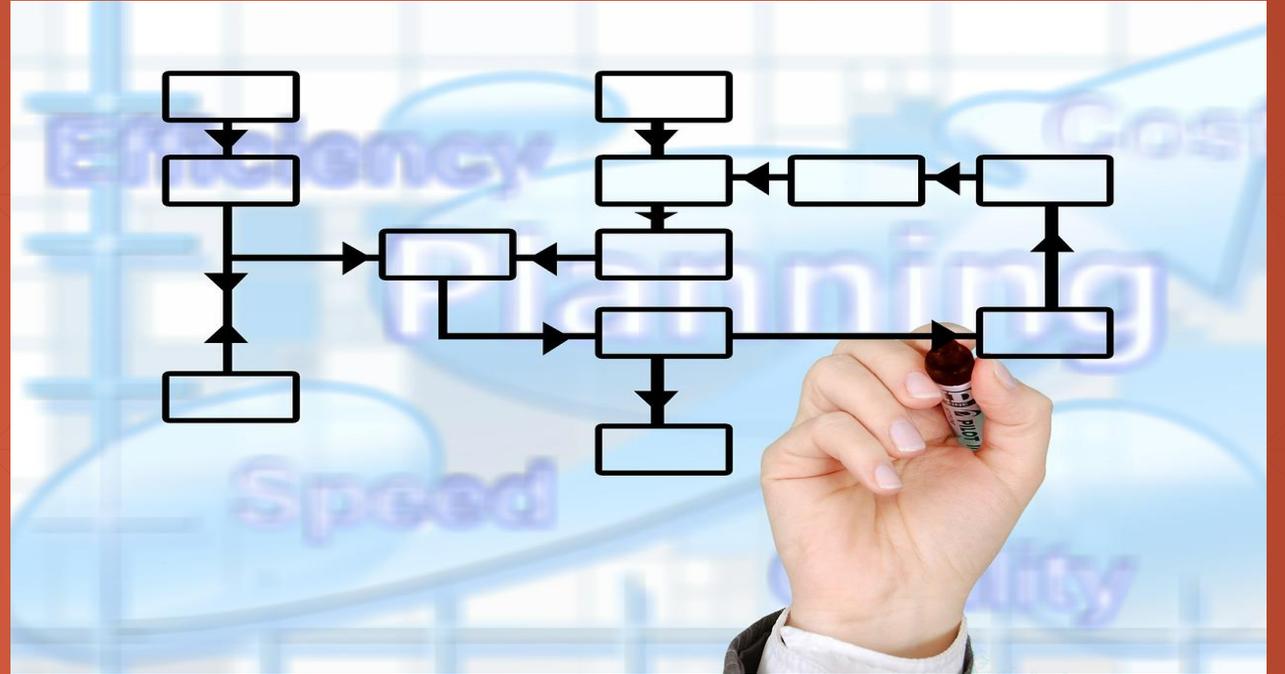
每个功能模块都包容着逻辑上相关的多个函数

模块之间通过调用特定的接口函数实现协作



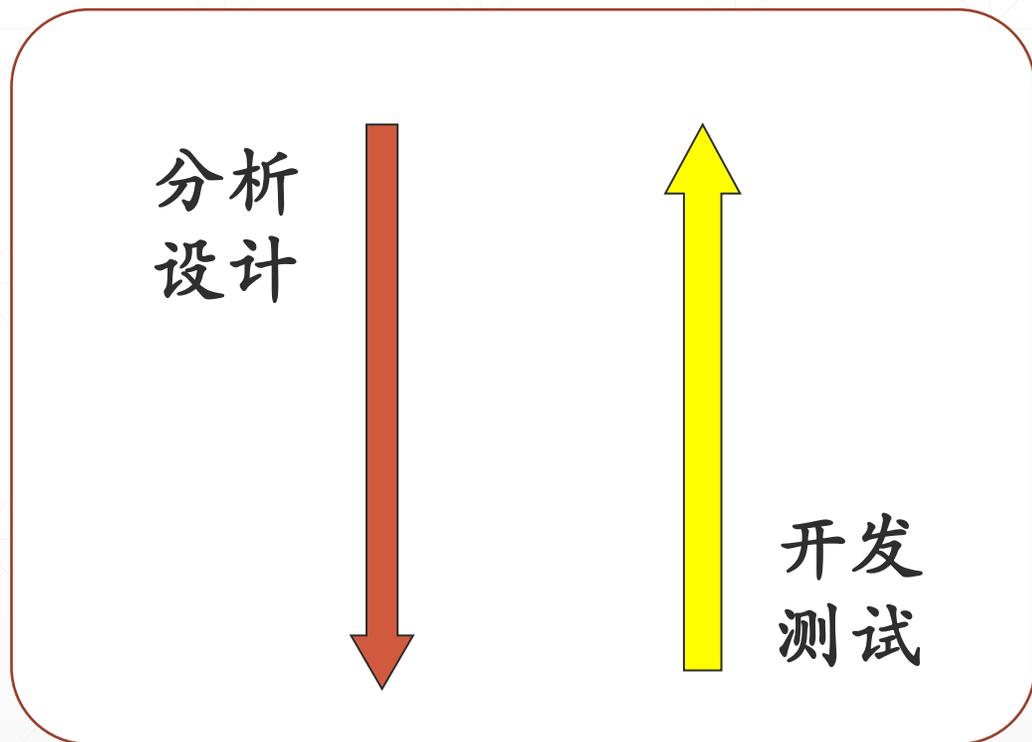
数据通常保存在功能模块外部（称为数据存储），功能模块中的函数直接访问数据存储进行加工。





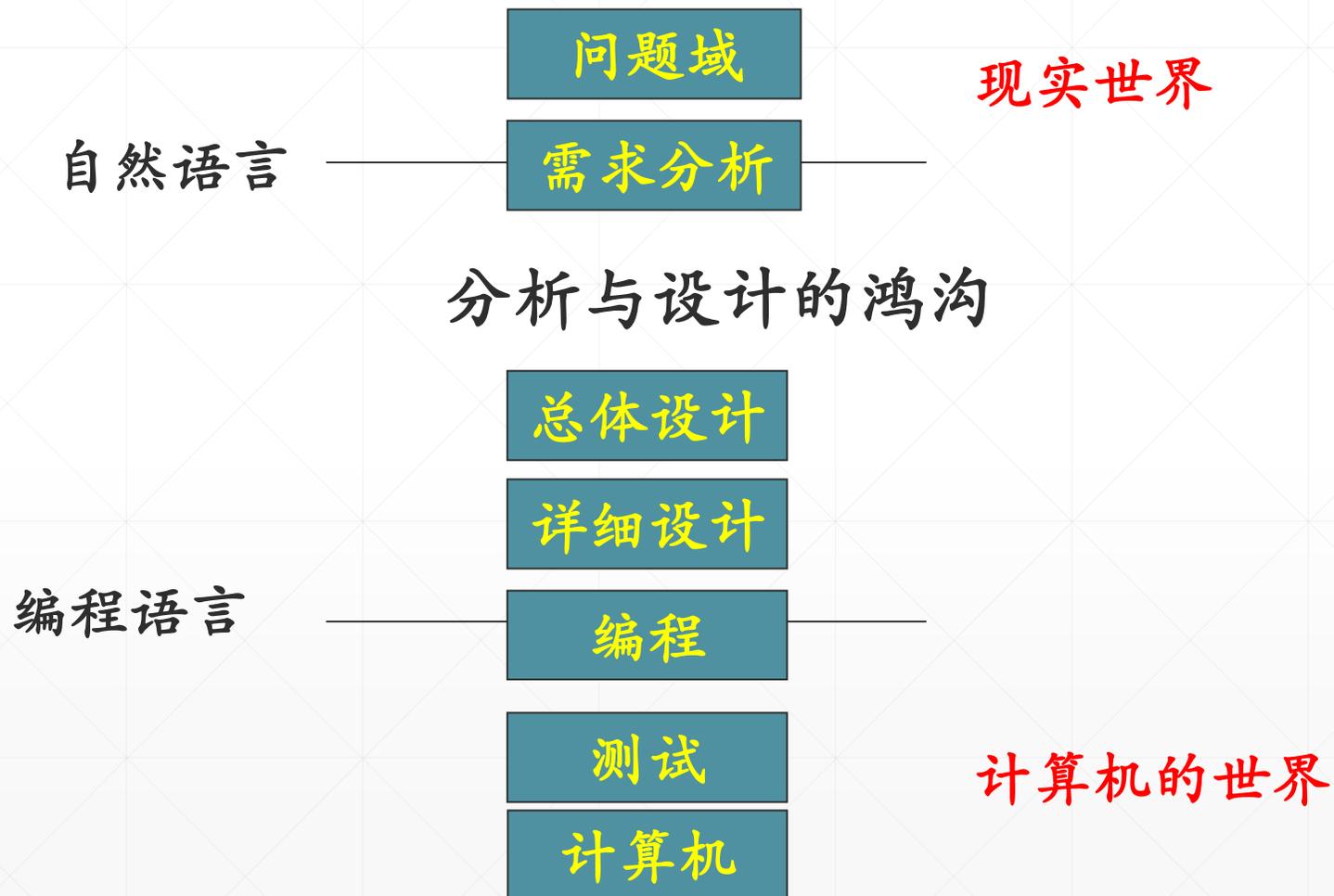
结构化软件的开发过程

结构化软件开发过程的主要特性 (1)



双向车道

传统结构化软件开发的缺陷



结构化方法开发复杂软件系统的“痛苦”

复用困难

牵一发而动全身

大量出现复杂的多参数的函数

实例：Win32 API中那可怕的上千个函数.....

```
int SetWindowRgn(  
    HWND hWnd,    // handle to window  
    HRGN hRgn,    // handle to region  
    BOOL bRedraw // window redraw option  
);
```

面向对象方法跨越“楚河汉界”

