

于春功师兄代码研讨会

主讲：王冰

outline

- 1, c++ 基础
- 2, google c++编程规范
- 3, 基因拼接基础
- 4, ARCS框架

C++基础

- 静态变量，静态函数，静态局部变量
- 函数重载
- STL（使用迭代器或BOOST_FOREACH遍历）
- 运算符重载(仿函数)，虚函数，友元
- 泛型

头文件 作用域

- 使用前置声明，减少.h文件的 # include数量，减少编译依赖
- 函数输入参数在前（const references），输出参数在后（non – const pointer）
- include顺序：c，c++，其他库，本项目库
- 函数，类内部使用 using ::foo::bar 或 namespace fez = ::foo::bar::baz
- 减少不必要的构造函数和析构函数调用(定义初始化，变量位置，++前置)

类

- 对单参数构造函数使用 C++ 关键字 `explicit`，避免不期望的隐式类型转换
- `#define DISALLOW_COPY_AND_ASSIGN(ClassName) \`
- `ClassName(const ClassName&); \`
- `void operator=(const ClassName&) // singleton模式`
- 用做base class的destruct声明为virtual
- 编写短小的函数（40行）！！
- 仅在输入参数类型不同、功能相同时使用重载函数
- 使用`static_cast<>`,`lexical_cast<>`，除测试不使用`dynamic_cast<>`

命名注释

- 类型名：大写开头不含下划线 (UrlTable)
- 成员函数：驼峰型 (setUrlTable)
- 变量：小写下划线分割(url_table)，成员变量下划线开始(_url_table)！！
- 宏和枚举：全大写加下划线(INSTALL_TABLE)！！
- TODO注释：// TODO(kl@gmail.com): Use a "*" here for concatenation operator.

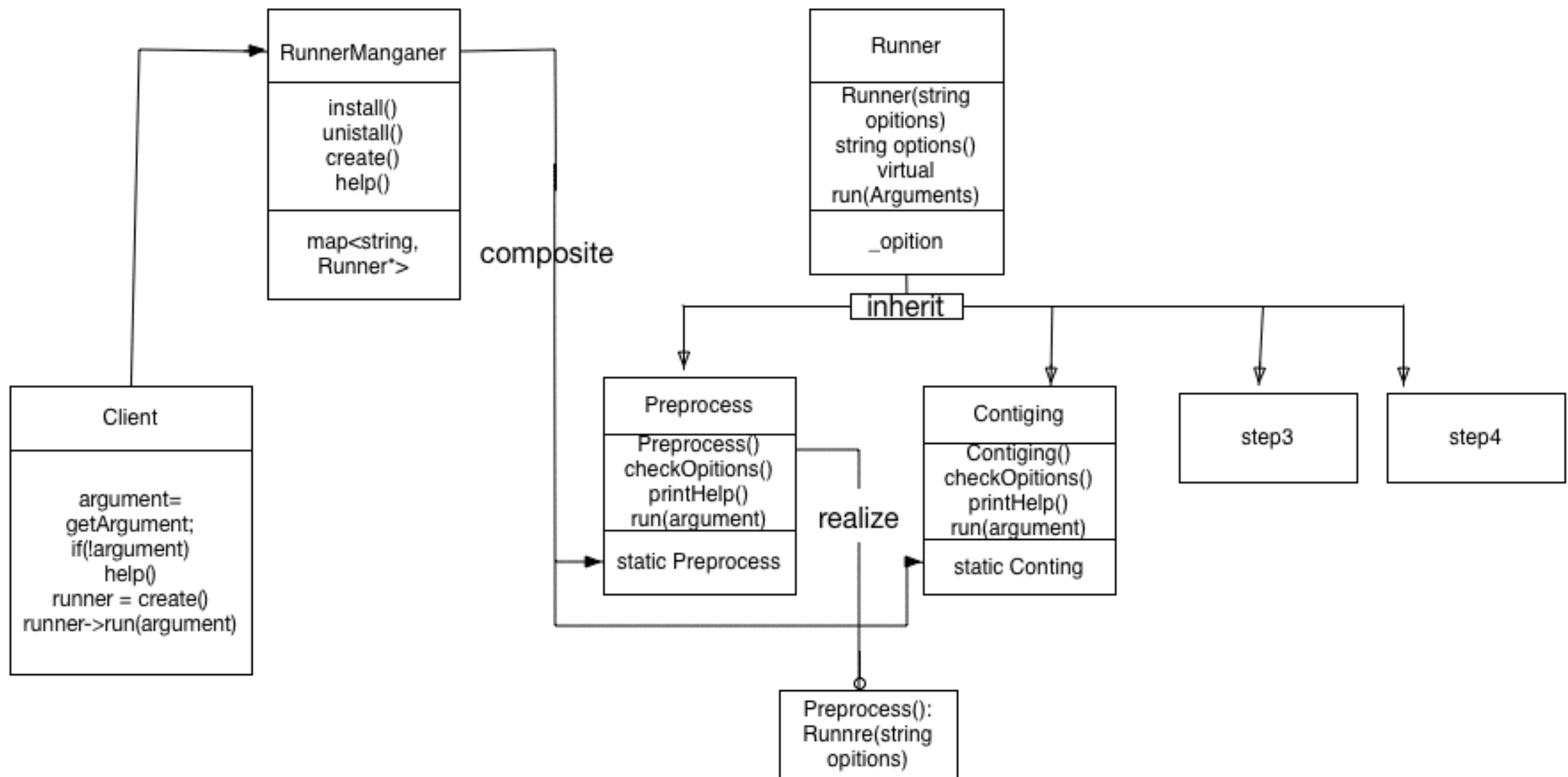
其他

- 使用boost被认可库
- 不要重复造轮子
- 说不出自己为什么比别人好，就用通用的解决方案

基因拼接基础

- preprocess filter and quality-trim reads
- assemble generate contigs from an assembly graph
- copy_num_estimate estimate copy number of contigs
- scaffold
 distance estimates generate ordered sets of contigs using
- solveLP solveLP
- remove_repeats remove_repeats
- gapfill fill intra-scaffold gaps

ARCS框架



- google c++ 编程规范: baidu
- ARCS: <http://github.com/bigict/ARCS>

- 疑问和建议
- 谢谢