

# 软件分析与测试实验报告

DZ1833003 曹雨露

## 背景介绍

回归测试：回归测试是指修改了旧代码后，重新进行测试以确认修改没有引入新的错误或导致其他代码产生错误。

回归测试的基本过程：

- （1）重点测试软件中被修改的部分；
- （2）从原基线测试用例库中，排除所有不再适用的测试用例，确定那些对新的软件版本依然有效的测试用例，其结果是建立一个新的基线测试用例库。
- （3）依据一定的策略从测试用例库中选择测试用例测试被修改的软件。
- （4）如果必要，生成新的测试用例集，用于测试无法充分测试到的软件部分。
- （5）用新软件测试用例集执行修改后的软件。

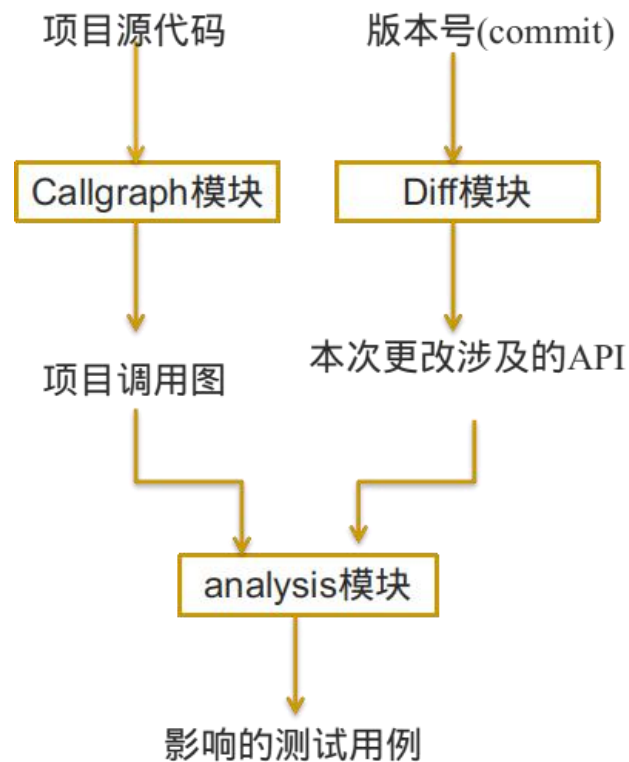
关于第（3）条，对于常常更改的代码，如果每一次代码更改之后，都执行全部的测试用例，十分耗时切不必要，因此需要做依存分析，对受影响的模块进行测试，回归测试重点测试软件中被修改的部分，这也是本工具需要实现的部分。

## 工具名称

回归测试工具

## 功能描述和设计框架

实现一个工具，通过依存分析，确定某次变更影响的测试用例。下图为设计框架。



## 详细设计

该工具的实现包含三个模块，Callgraph 模块、Diff 模块、Analysis 模块，首先将需要分析的项目代码从 github 上克隆到 projects 文件夹下，通过 Callgraph 模块生成项目的静态调用图，该调用图包含项目内部代码的调用关系，调用图保存在 dps 文件夹下，当需要对一次特定的修改进行影响分析时，需要提供本次更改的 commit sha 值和项目名称，然后工具会自动在 dps 文件夹寻找对应的调用图，如果没有该调用图，会提醒用户先生成调用图，如果找到了调用图，将通过 Diff 模块得到本次更改涉及到的 API，再通过 Analysis 模块得到更改影响到的测试用例。以下是三个模块的具体实现方式。在以下描述中，API 指函数或者方法。

Diff 模块：

用于解析一次 commit 中更改了文件中的哪些模块。首先通过 diff 语句，确定更改的行数，切换到项目的当时版本，然后解析文件中所有 API 定义的起始行，然后确定更改的对应是哪些 API。

Callgraph 模块：

该模块用于构建调用图，主要参考了 github 上开源项目 pyan，该模块采用解析抽象语法树的方法，构建了项目内部的调用图。

Analysis 模块：

对更改影响了哪些测试用例进行了分析，通过深度搜索得到调用该 API 影响的所有模块，当有测试用例调用了这些模块，则意味着该测试用例可能受到影响，则挑选该测试用例，作为重要模块进行测试。

## 使用方法

本工具使用 Python 编程，在调用时使用命令行参数，包含的参数有：

it is a tool of impact analysis for regression testing

optional arguments:

-h, --help	show this help message and exit
-l	list the callgraph of exist projects
-g G	generate the callgraph of the provided project
-p P	provide the name of the project and start analysis
-cm CM	please input the commit

用户在使用过程中，首先需要将项目源代码克隆到 projects 文件夹下，然后调用-g 参数生成项目调用图，然后调用-p 参数指定项目名称，-cm 参数指定版本号，如 commit sha，进行分析。

## 挑战性和解决办法

### 1. 静态调用图的生成

Python 是一个动态语言，没有类型检查等，在静态分析时，动态绑定和函数参数等调用关系无法正确解析，本工具是在 github 开源项目 pyan 上进行了改进，修改了其在运行可能出现的问题代码，最终生成了调用图。

### 2. 如何挑选测试用例

为了保证工具的实用性，对于每次更改挑选的测试用例不能太多，但重要的是覆盖到可能影响的所有测试用例，所以本工具将可能影响到测试用例全部挑选出来，将来可以改进的是，进一步对测试用例进行筛选。