**Coverage使用总结**

1. **介绍**

由于python是解释性语言并非编译性语言，所以很多代码错误的要到运行时才能发现，我们需要用代码覆盖工具弥补缺少编译器损失的检查环节．Coverage是一个python的代码覆盖检查工具，除了跑代码，他还可以生成一些专业的统计报表．

1. **使用方法**
   1. 安装：可通过easy\_install来下载，easy\_install是setuptools的一个模块，所以在使用easy\_install之前要先安装setuptools。

或者通过官网下载

* 1. 开始代码覆盖
* 终端输入：coverage run test.py arg1 arg2
* 通过API覆盖：

from coverage import coverage

cov = coverage( ) #生成coverage对象

cov.start( )#开始分析

function\_to\_be\_tested()#待测函数

cov.stop()#分析结果

cov.save()#将覆盖结果保存到数据文件

* 1. 显示覆盖结果

coverage report

* 1. 将覆盖结果生成HTML文件

coverage html –d testhtml(指定一个文件名)在本地生成一个名为testhtml的文件夹里面有每次执行后生成的HTML页面

或：coverage html在本地生成一个默认文件名为htmlcov文件夹

* 1. 将结果生成XML文件

Coverage XML

* 1. Combine组合覆盖结果
* 这个方法很重要，首先要把需要组合的文件放在同一个文件夹下。
* 要求：对文件格式的要求，执行：

coverage run –p a.py

coverage run –p b.py

生成两个符合格式要求的文件，格式如下：

.coverage.hostname.processname

* 执行coverage combine

会生成一个.coverage文件就是合并后的文件

* 同上可生成HTML文件