# [自动化工具设计]软件兼容性工具——包安装依赖检查工具

### 背景

参与 操作系统标准工具SIG 项目，与4院以及其他公司进行合作，进行标准化工具的开发，助力国产操作系统兼容性。

### 需求

我作为一个应用厂商，希望能够保证我的应用的兼容性，从安装依赖的角度考虑，要求如下：

（1）包只依赖标准规定的1类库或2类库，且版本符合标准

兼容性级别

基于Linux内核的服务器操作系统和微型计算机操作系统发行版的系统运行库的兼容性，分为以下级别：

L1 长期稳定库：标准下一个版本会保持平滑兼容的库，操作系统应实现对该类库的跨主版本兼容；

L2 短期兼容库：标准下一个版本发布时会尽量保持兼容性，操作系统在主版本内应保持兼容；兼容性变化时会标准通常会给出新版库或替代的同类型库（比如QT5会被QT6替代）；

L3 试用库：建议操作系统提供的库，或者属于接收持续更新的组件和开发人员工具，此兼容级别将尝试尽可能地保持但不保证兼容性；使用此类库的应用宜密切关注标准及发行版支持情况，必要时可以考虑使用自包含等方式保证软件的兼容性。

### 逻辑

1. 输入
   1. 标准文件
   2. 应用包
   3. 桌面还是服务器
   4. 应用包管理体系
2. 测试方法
   1. 获取应用的安装依赖列表
      1. deb包使用 dpkg -f $deb Pre-Depends 和 dpkg -f $deb Depends 获取依赖包列表
   2. 遍历依赖库集合
      1. 若包不在标准中，存在兼容问题

若包在标准中：

若是放弃类型的，存在兼容问题；

若否，检查兼容等级：

若是L3级别，存在兼容问题；

若否，进行版本检查：

若没有版本要求，则通过；

否则执行如下检查：

若版本号是=或<=，存在兼容问题（rpm包默认为>=）；

若否，检查版本号；

若=标准，则通过；

若否，存在兼容问题；

结果表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **是否在标准中** | **是否废弃deprecated** | **库等级** | **版本要求** | **版本比较** | **结果** | **提示** |
| 否 | / | / | / | / | WARNING | 该包为不推荐使用的包，请您使用标准中的包 |
| 是 | 是 | / | / | / | WARNING | 本库即将在下一个版本中废弃，不建议使用 |
| 否 | L0L3 | / | / | WARNING | 该包系统不保证兼容，不推荐使用 |
| L1L2 | / | / | PASS |  |
| <=/= | / | WARNING | 该依赖要求版本过严，建议修改 |
| >= | > | WARNING | 依赖版本高于标准 |
| = | PASS |  |
| < | WARNING | 依赖版本低于标准 |

输出

**控制台/日志输出格式**

[2022-09-06 16:56:06,308] [appchecker\_pkg.py:234]:[INFO] ########## 软件包依赖检测开始 ##########  
[2022-09-06 16:56:06,308] [appchecker\_pkg.py:236]:[INFO] ##### 正在检查包 deb1 #####  
[2022-09-06 16:56:06,335] [appchecker\_pkg.py:240]:[INFO] 获取到依赖包列表 [{'name': 'dep1', 'limit': '', 'version': ''}, {'name': 'dep2', 'limit': '>=', 'version': '1.0.16'}]  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:116]:[INFO] ### 正在检查dep1 ###  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:120]:[INFO] 该包未在标准中  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:121]:[INFO] 结果为warning  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:168]:[INFO] ### dep1 检查完毕 ###  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:116]:[INFO] ### 正在检查 dep2 ###  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:120]:[INFO] 该包未在标准中  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:121]:[INFO] 结果为warning  
[2022-09-06 16:56:06,336] [appchecker\_pkg.py:168]:[INFO] ### dep2 检查完毕 ###  
[2022-09-06 16:56:06,347] [appchecker\_pkg.py:258]:[INFO] ##### 包 deb1 检测完毕 #####  
[2022-09-06 16:56:06,347] [appchecker\_pkg.py:259]:[INFO] ########## 软件包依赖检测完成 ##########

**测试结果**：PkgChecker.json

点击此处展开...

{

"result": "warning",  
"data": [

{

"name": "deb1",  
"result": "warning",  
"detail": [

{

"item": "dep1",  
"level": "none",  
"result": "warning",  
"info": "该包为不推荐使用的包，请您使用标准中的包"

},  
{

"item": "dep2",  
"level": "none",  
"result": "warning",  
"info": "该包为不推荐使用的包，请您使用标准中的包"

}

]

}

]

}

### 设计

PkgChecker类：

* init方法
  + 实例化标准
  + 实例化result
* get\_standard 方法：
  + 获取包信息
* check方法：
  + 遍历标准进行测试
* pkg\_check方法：
  + 检查依赖包的兼容性
* export方法
  + 调用result的export，输出结果csv
* stat方法
  + 调用result的stat，计算统计结果

{'total': 140, 'pass': 103, 'warning': 36, 'fail': 1}

main方法：

主函数，串流程，返回报告路径