

Linux手机平台让应用程序单实例运行 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/145/2021\\_2022\\_Linux\\_E6\\_89\\_8B\\_E6\\_9C\\_c103\\_145306.htm](https://www.100test.com/kao_ti2020/145/2021_2022_Linux_E6_89_8B_E6_9C_c103_145306.htm) 和PC上的应用程序不同，手机上的应用程序一般都只允许单实例运行。当应用程序已经在运行，再次运行该应用程序时，通常只是把该应用程序的窗口提到前面来，把新的命令行参数传递给第一个运行实例，而第二个实例退出。这在传统的单进程多线程的手机平台中，实现是简单而直接的，而在linux这样的多进程平台里，则会稍微困难一点。记得在Windows下，保证应用程序单实例运行的最简单办法就是创建一个命名的mutex，第二个实例运行时创建同名的mutex会失败，便认为当前运行实例不是第一个了。这种方法虽然可以解决单实例运行的问题，但是无法把第二个实例的命令行参数传递给第一个实例，也无法把第一个运行实例的窗口提到前面来。在PC上，把第一个运行实例的窗口提到前面来，完全可以让用户自己手工进行，而在手机上应该由程序来实现。在我们的linux手机平台中，起动应用程序的地方主要有：开始菜单、新事件查看和文件管理器。开始菜单中起动应用程序，参数是固定的，处理最简单。新事件查看即查看新短信、新提醒和新通话记录等等，它起动应用程序时，会把新事件的记录ID作为命令行参数，传递给对应的应用程序。文件管理器即可以起动应用程序，也可以用应用程序打开其关联的文件，后者把数据文件的文件名作为命令行参数，传递给对应的应用程序。这实际上由三个子问题组成：应用程序单实例运行、自动把窗口提到前面和把参数传递给第一个运行实例。对此我们考虑了下列方法：

方法一：遍历/proc目录下的进程信息，如果应用程序对应的进程存在，则让认为应用程序已经在运行。然后强制关闭该应用程序，用新的命令行参数重新运行它。这样做的优点是实现简单，到达了我们的预期的目标。缺点是强制kill第一个运行实例太野蛮，可能会造成数据丢失，另外创建新进程的开销会让用户感觉反应迟钝。

方法二：通过窗口管理器实现，根据应用程序的窗口判断应用程序是否在运行。如果发现对应的应用程序已经在运行，把对应的窗口提到前面来。这要修改应用程序才行，只有应用程序自己才知道其拥有的窗口的关系，若不加考虑把应用程序的主窗口提到前面来，这也并不见得合适。这种方法只能解决前两个小问题，命令行参数还是无法传递给第一个运行实例。

方法三：应用程序提供一个DBUS服务对象，第一个运行实例作为服务器运行，第二个运行实例作为客户端运行，第二个运行实例把命令行参数传递给第一个运行实例,然后就退出。第一个运行实例接受到新的命令行参数，作相应的处理，比如，把相应的窗口提到前面来，根据命令行参数进行处理。这是第二种方法的改进，可以实现全部的功能，缺点是要修改应用程序，不过DBUS服务对象可以通过一个公共库来实现，应用程序的改动并不大。

我们最终决定采用方法三，并实现了一个公共函数库，它提供下列功能：

- 判断当前运行实例是否是第一个实例。
- 第一个运行实例作为服务器运行，注册自己的命令行参数处理函数。
- 第二个运行实例作为客户端运行，更新自己的命令行参数到服务器（即第一个运行实例）上。

在起动机时，应用程序的常规处理方法：如果是第一个运行实例，则作为服务器运行，并注册命令行参数处理的回调函数，后续操作与正常

相同。再次运行该应用程序时，注册的回调函数就会被DBUS调用，并传入新的命令行参数。在该回调函数中，应用程序通常要把相应的窗口提到前面来，并根据新的命令行参数做些处理。比如，查看新短信时，短信应用程序要打开查看窗口，或者把查看窗口提到前面来，并显示新的短信内容。如果是第二个运行实例，它简单的把命令行参数传递给第一个运行实例，然后自己退出。其它的事情都让第一个运行实例去完成。比如打开新的mp3文件时，媒体播放器只是把该文件的文件名传递给媒体播放器的第一个运行实例，然后自己退出。 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)