

Pythonanygui项目预览 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/181/2021_2022_Pythonanyg_c103_181698.htm 在 Python 世界中有一个非常有趣的 [anygui] 项目，它已经进入了早期的开发阶段。[anygui] 项目打算作为许多主要图形工具箱的下层 API。一旦完全开发成功，Python 程序员就可以调用一个公共 [anygui] 函数 — 例如，为创建一个窗口 — 可由“最适当好用”的工具箱来完成这项工作。在 Windows 上，可以使用到 Win32 API（或者 wxWindows）；在 MacOS 上，可能本机调用；在 BeOS 上，使用 Bethon；在 Linux 上，使用 TKinter 或者 GTK；在 Telnet 屏幕上使用 ncurses — 所有这些都取决于给定的机器上安装的和可用的软件。本文讨论了 [anygui] 当前的开发状态，以及该项目要达到的目标。编写一次，到处显示！许多年前当 Java 刚出现时，它的一个重要承诺就是实现代码“编写一次，随处运行”的想法。起先，主要考虑将 Java 用户界面作为 applet，嵌入到 Web 浏览器中。一段时间后，独立的 AWT 应用程序成为更流行的概念。接下来，AWT 通常被 Swing 所取代。Swing 又变成了 Bean（构建在 Swing 上，但有另外的要求）。这样依次下来，Swing 类从 Java 版本中添加、删减，不断来回变化着。有关 Java 的一个很流行的笑话是，“编写一次，到处调试”。至少可以确定的是，您不可能编写了一个 Java 应用程序后，非常自信地认为它可以运行在您应用程序的每个用户机器上 — 除非您愿意要求每个用户做相当多的工作来获取 Java 版本和配置，使它们完全符合您特定的应用程序。应用程序是否运行取决于 Java 版本，以及甚至特定的供应商和安装

Java 虚拟机 (JVM) 的平台。就大多数方面而言, 如 Python、Perl 和 Tcl 这样的脚本语言, 要比 Java 具有更好的可移植性。例如, 对于大多数 Python 脚本, 程序员感到十分自信的是, 发送到多个用户的脚本在每台目标机器上都将正确和完全一致地运行 (可能有最低版本的要求 — 这要比 Java 上的要求简单得多、可靠得多)。当然, Java 除了不完美的移植性外, 它也有许多优势: 静态输入 (许多人想要它)、庞大的类库、卓越的文档、细心的设计选择。但是有关那些语言的注意事项并不是我在这里感兴趣的话题。Python 脚本可移植性中有一个地方比 Java 差很多, 那就是在用户界面中。对于一个命令行工具, 这一点不成问题。但当您希望复杂的用户交互时 — 特别专门针对图形界面时 — Python 实际上什么也不能提供。对于所有的小故障和小错误, Java 通常确实为每个带 JVM 的平台提供了 Swing 和 AWT 这些基本的 GUI。相反, Python 完全没有任何 “标准” 的 GUI 库。许多人都表示他们期望有标准的 Python GUI。Tkinter 恰在这时到来 — 它有 Windows 和 UNIX/X Window 系统稳定的版本, 并且有一个过得去的 MacOS 版本。但您还需要在系统上安装 TCL 和 TK, 以及被忽略的 “非主流” 平台, 如 BeOS 和 OS/2。各类倡导者建议, 采用一些其它库 / 绑定将会是一个更好的选择 (有许多库可以进行选择; 请参阅 参考资料)。而每个库 / 绑定支持一个期望平台的子集; 并且最重要的是, 没有一个库 / 绑定被一致地接受 (因此, 没有一个成为标准而随 Python 分发版一起发行)。我们无法编写带用户界面的应用程序, 并且无法确保实际用户可以与它进行相互操作。重新考虑这个问题 Java 思想创建了每个 JVM 必须实现的能力的一个标准集

合。Java GUI 是根据规定而存在。Python 式的方法可能来自一个不同的角度。不再命令每台机器遵从某个确定的 API，取而代之的是，只是确定给定机器可以做什么，然后从那里开始工作。API 仅作为底层平台所完成工作的包装器。一旦以 Python 方式考虑问题时，anygui 做了您确实希望做的事情。从 anydbm 模块中取出它的名字和相关说明，在运行时找到“最适当可用”的数据库后端，anygui 在自身应用程序正在运行的系统上会找到最适当可用的 GUI 后端。anygui 强调提供一个可以和每个后端一起工作的实用的界面元素集合；特定的后端可能自己能够提供更高级的界面，但 anygui 提供的是它们的公共的界面。写本文的时候，anygui 还是一个 alpha 级的项目。对一个目标后端子集，anygui 已经做得非常好。但由于目标是建立一个（接近于）通用的包装器，只有一个子集在工作显然是不够的。最终，如果 anygui 达到了它的目标，则在每个 Python 分发版中，anygui 都将作为标准的 Python 包包含在其中，那才是有意义的（就如同不管什么依赖于系统的后端，Python 分发版中都包括了 anydbm 或者 xml.sax）。毕竟，这一点才能确保每个用户都拥有了它。顺便提一下，anygui 是纯 Python；它本身不需要 C/C 或者其他较低级语言的任何东西（当然，如果有用，anygui 应该找一些支持的 GUI 库）。台和图片 为了写本文，我快速研究了大多数工作后端。还有几个后端没有实现，或者只实现了部分功能。已经实现的有 Tkinter、Java Swing、win32all、PyGTK 和 wxPython。BeOS 本机（带 Bethon）只实现了部分功能，但可能每天都构建一次进行新的改进。PyQT 和 MacOS 本机已经编制了规划，并且已经进行了开发，还没有创建这些包

装器的实现；当然，随着时间的推移，这些都可能发生变化。关于直接的 xlib 后端，也一直有一些讨论，但目前没有人自愿负责这项工作。上面所有的图形工具箱以一种非常相似的方式工作，或将以一种非常相似的方式工作。我承认我在大多数后端工具箱方面的知识很有限——但从我的理解，anygui API 在很大程度上与 Tkinter 相似。本质上，这个策略是创建带回调的一串窗口小部件，然后进入一个主事件循环。将来还可能会有一些其它后端，它们会打破这个“标准”的 GUI 工具箱模型。在某些方面，它们看起来似乎最有趣，或者至少很新颖。一个已经规划好的后端被期望由我来负责——但是在开发最初的版本时，我有点懈怠。希望在您阅读本文时，这会有所改进。我自己开发的小型后端是 ncurses。如果它最终实现，则这开创了甚至在文本模式的终端（譬如，SSH/telnet 会话）或者只是在纯 UNIX 机器（不带 X Window 系统）上运行 anygui 应用程序的可能性。按照 curses 后端的风格，anygui 的项目负责人 Magnus Lie Hetland 已经建议了纯面向行界面（似乎有点倒退），它可以使用 readlines 支持。在这个方案中，菜单将简化成提示符，接着是选项选择，再接着是反馈或者结果等等。假想中的 anygui.backends.textgui 只需要 STDIN 和 STDOUT 来工作，这对于程序来说，是令人感兴趣的，因为它所需要的最少。否则，该程序可能（一点未变动）运行在复杂的图形化、事件驱动和 WIMP 界面（窗口、图标、鼠标指针）。当然，到目前为止，它还只是一个想法。还有一个古怪的想法也同样有趣。每人都有一个 Web 浏览器（几乎），即使那个浏览器碰巧是 lynx 或者是 links。Python 标准的 webbrowser 模块允许以

一种与 anygui 以及和 anygui 具有相似功能的软件的方式，灵活地启动一个“最适当可用”的 Web 浏览器。如果那个浏览器与某类 LOCALHOST 服务器通信，那么所有您想要的基本界面设备都完全在 Web 浏览器内（按钮、输入域、文本区域、图形等等）。这个后端也处于规划阶段。一幅图片抵得上用千言万语来表达的含义（至少有时候）。由于我的编辑们，出于善意，不希望将本专题出版为一万字的巨著，所以就让我们看几幅屏幕快照。为了说明需要，使用了一个小玩具应用程序，有一些按钮，并且启用了几个不活动性的按钮（它的源代码显示如下）。其中还包含了几个文本标签。其它窗口小部件示例包含在 anygui 分发版的 test 目录中。首先值得一看的，我们可能认为是作为“默认缺省值”的后端 Tkinter。这个版本看上去和使用起来就与它应该做到的完全一致。但是，`win.destroy()` 调用就有点好笑——它没有立即破坏窗口（并且关闭应用程序），而是一旦当窗口受到足够的注意（比如移动窗口），它就将窗口变为一个会消失的鬼怪。如同我说的，我们还停留在 alpha 阶段。这个示例在 Win98 下运行：Tkinter 下的按钮应用程序（在 Win98 上）在 Windows 下运行时，还有一个利用 `win32all` 模块使用 Windows 本机调用的选项。来自 ActiveState 的 ActivePython 分发版在缺省情况下有这个选项；否则，还需要另外获得此模块（也是来自 ActiveState）。总之，这种绑定是我看到的运行最好的一种——但那也只是反映在我做测试的版本上。标签的放置与 Tkinter 上的有一点不同，这表示，一个人对不同后端不可能得到完全相同的视觉审美观。Win32 下的按钮应用程序（在 Win98 上）

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