

Shell 其实就是一种脚本语言，也是一个可以用来连接内核和用户的软件，我们编写完源码后不用编译，直接运行源码即可。

bash, a general purpose scripting language.

command [arguments]

interactive and batch / 交互式和批处理

Gnuplot (在Linux下基于Qt的一套绘图工具)

https://blog.csdn.net/weixin\_42014622/article/details/82962634

LaTeX（LTEX，音译“拉泰赫”）是一种基于ΤΕΧ的排版系统

https://blog.csdn.net/pipisorry/article/details/54571521

关于shell 和 bash 之间的关系

<https://blog.csdn.net/weixin_42432281/article/details/88392219>

|  |  |  |
| --- | --- | --- |
| Platform | Linux | Windows |
| 字符界面模式 | shell | Command Prompt |
| 命令行 | bash | command |
| 批处理模式 | batch mode | 批处理 bat |

commands: aliases, functions, built-ins, keywords, and executables.

Text, letter

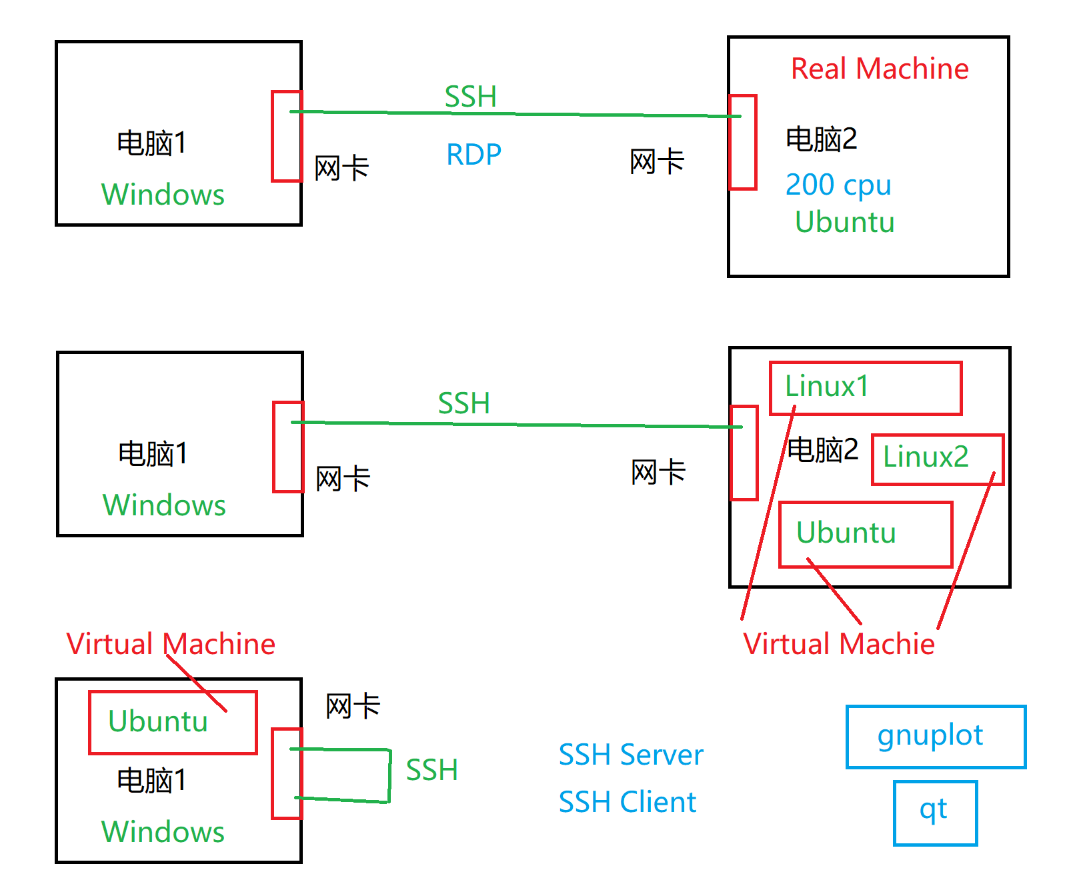
Description automatically generated

**Here document** 的使用总结

https://blog.csdn.net/liumiaocn/article/details/86715953

Sudo apt install / upgrade/update \*\*\*\*

Sudo apt-get install \*\*\*\*\*



关于makefile和make:

<https://blog.csdn.net/ZBraveHeart/article/details/123187908>

Makefile文件中的命令有一定规范，一旦该文件编写好以后，在Linux命令行中执行一条make命令即可自动编译整个工程。

Text

Description automatically generated

它表示，为了达成目标，必须1，先有依赖，2，顺序执行下面的命令

Makefile并不会关心命令是如何执行的，仅仅只是会去执行所有定义的命令，和我们平时直接输入命令行是一样的效果。

1、目标即要生成的文件。如果目标文件的更新时间晚于依赖文件更新时间，则说明依赖文件没有改动，目标文件不需要重新编译。否则会进行重新编译并更新目标文件。

2、默认情况下Makefile的第一个目标为终极目标。

3、依赖：即目标文件由哪些文件生成。

4、命令：即通过执行命令由依赖文件生成目标文件。注意每条命令之前必须有一个tab保持缩进，这是语法要求（会有一些编辑工具默认tab为4个空格，会造成Makefile语法错误）。

5、all：Makefile文件默认只生成第一个目标文件即完成编译，但是我们可以通过all 指定所需要生成的目标文件。