



## Cygwin系列（十二）：了解X



silaoA

爱好编程的电气工程师, silaoa.github.io

已关注

29 人赞同了该文章

发布于 2020-04-25 19:33，编辑于 2022-07-07 10:17

本文共4500余字，预计阅读时间14分钟，本文知乎连接：[Cygwin系列（十二）：了解X](#)，本文同步发布于[silaoA的博客](#)和[微信公众号](#)平台。

关注学习了解更多的Cygwin、Linux、Python技术。

### 目录

- 0x00 前言
- 0x01 X Window System
- 规范
- XFree86和X.Org
- 桌面环境
- 应用场景
- 0x02 X的竞争对手
- Wayland
- XWayland
- Mir
- 其他

- 总结
- 参考
- 更多阅读



## 0x00 前言

图形环境不是计算机系统与生俱来的，现在习惯了Windows、Mac OS的用户，对于图形用户界面（GUI）得心应手，然而早期的计算机系统只有个黑框框，仅少数专业人员能够驾驭。一般认为，是施乐公司帕洛阿尔托研究中心 (Xerox PARC)在1973年首次研发出图形用户界面，具备了现代图形环境基本元素。这个激动人心的发明，还没有被施乐公司好好利用起来挣钱的时候，很快被苹果、微软等多家公司“借鉴”，伴随着PC发展，各家GUI系统群雄逐鹿，没有统一标准。

## 0x01 X Window System

- 规范

1984年MIT大学创造了X规范，就类似TCP/IP协议栈一样。本义是在**UNIX系统构建图形化视窗系统的设计方案**，准确名称叫“X Window System”，注意拼写**不是**“Windows”，与微软、与Windows没任何关系。X规范在1987年形成了第11版（Version 11），1994年发布第11版的第6次发行（Release 6），确切版本名即为“X11R6”，后来成为UNIX上图形视窗系统标准规范，也简称“X11”或者“X”，目前最新版本是2012年发布的“X11R7.7”。

X11 图源：维基百科

“X11”采用了C/S的架构，在其设计下，整个图形视窗系统主要分为3个部分：

- X Server（X服务器）。X Server一方面负责和设备驱动交互，监听显示器和键盘鼠标，另一方面响应X Client需求传递键盘、鼠标事件、（通过设备驱动）绘制图形文字等。**反直觉之一，X Server运行在本地。**
- X Client（X客户端）。X Client也叫X应用程序，负责实现程序逻辑，在收到设备事件后计算出绘图数据，由于本身没有绘制能力，只能向X Server发送绘制请求和绘图数据，告诉X Server在哪里绘制一个什么样的图形。**X Client可以和X Server在同一个主机上，也可以通过TCP/IP网络连接。**
- Window Manager（窗口管理器，简称WM），或者叫合成器（Compositor）。多个X Client向X Server发送绘制请求时，各X Client程序并不知道彼此的存在，绘制图形出现重叠、颜色干扰等问题是大概率事件，这就需要一个管理者统一协调，即Window Manager，它掌管各X Client的Window（窗口）视觉外观，如形状、排列、移动、重叠渲染等。**反直觉之二，Window Manager并非X Server的一部分，而是一个特殊的X Client程序。**

3个部分，**X Server是整个X Window System的中心**，协调X客户端和窗口管理器的通信。

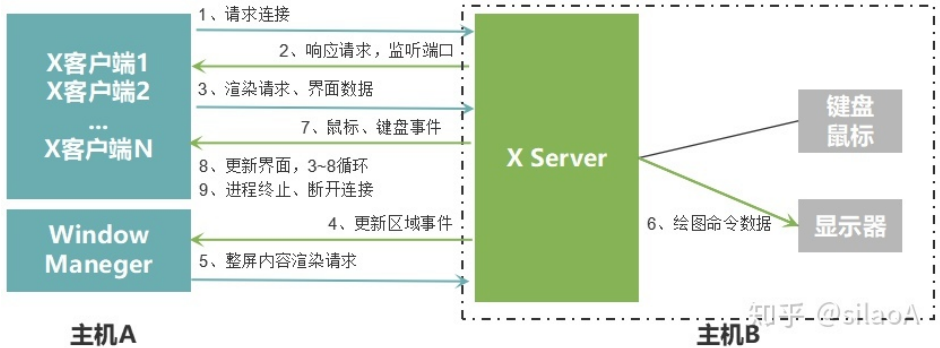


X11的C/S架构 图源：维基百科

“X11”充分遵循UNIX设计哲学，尽可能简单，除非必要绝不新增功能，其实也意味着粗糙、原始；提供机制而非具体策略，如菜单、按钮等元素细节并不做规定，交给窗口管理器、客户端程序等自主实现，于是不同的实现风格各异。任何遵循了X11规范的客户端程序和服务器程序配合起来，即可正常运行，X11得益其开放性，迅速发展，成为UNIX系统的事实标准。

假定多个X客户端程序及窗口管理器在主机A上，某个X Server（如下文 x.org Server ）运行在主机B上，程序运行过程可简化如下过程。

- 1. 某个X客户端进程启动，向主机B发送连接请求，目标地址可通过命令行或配置文件指定，如果给定的地址已有X Server正在监听端口，则进行下一步；
- 2. 主机B上的X Server返回一个连接正确响应，X Server也可以配置接受或拒绝某些地址的请求；
- 3. X客户端向X Server发送渲染请求及窗口界面数据；
- 4. X Server一方面将窗口界面数据交给显示驱动计算渲染缓冲，另一方面综合各个X客户端的渲染请求，计算更新区域，但它并不知道如何将多个窗口“合成到一起”，于是将更新区域事件发给窗口管理器；
- 5. 窗口管理器了解到需要在屏幕上重新合成一块区域，再向X Server发送整个屏幕的绘制请求和数据；
- 6. X Server将绘图数据交给显示驱动计算所有渲染缓冲，并最终绘制图形；
- 7. 运行过程中，X Server可能收到主机B上的鼠标、键盘事件，经计算后，X Server决定发给哪个X客户端（即获得焦点）；
- 8. X客户端收到鼠标、键盘事件后，回调事件处理，并计算界面该如何更新；
- 9. 循环第3~8，直至X客户端收到关闭事件，进程终止、连接断开。



X绘图过程示意

以上过程，主机A和B的CPU架构、操作系统可能都不相同，若A和B是同一个主机，就相当于在本地绘图、显示了。

- XFree86和X.Org等实现

**X11只是规范，并不是代码实现。** MIT另外只给了工具包和参考实现，参考实现的意思是不考虑性能只演示工作原理。



1992年，XFree86 项目顺利开展，作为X Window System的一个早期开源实现，兼容IBM PC机（XFree86 中86的含义），得到广泛应用，项目地址<http://www.xfree86.org>。至2004年，由于项目内部争论和许可证分歧，XFree86 从GPL协议改为XFree86 License发布，项目从开放走向封闭。

2004年，X.Org基金会成立，开始领导X项目，不仅推动X规范本身发展，还在XFree86 4.4 RC2版基础上开发了X规范的另一个实现——X.Org Server，原来XFree86上很多开发者也转向了X.Org Server，由于更加开放而被多数GNU/Linux发行版使用，项目官网<https://www.x.org>，源代码库地址 [cgit.freedesktop.org/xo/](https://cgit.freedesktop.org/xo/)，2019年发布的稳定版为1.20.5。

- 在X Client方面，X.Org Server 实现了2个开发库，Xlib 和 XCB，便于开发者编写X应用程序。
- 在X Server方面，X.Org Server 实现了“X Window System”核心规范及多个扩展组件。
- 在Window Manager方面，X.Org Server 继承XFree86 项目的 twm。
- 其他，实现了一个X Display Manager（X显示管理器）——xdm。通常透过字符终端登录Linux主机时，用户仅能获得字符环境，而X显示管理器实现XDMCP规范（X Display Manager Control Protocol），为用户提供图形登录、会话管理服务，允许用户登录时自主选择使用哪个Window Manager等。

除上X Server，X11系统客户端、窗口管理器也有着很多种实现。

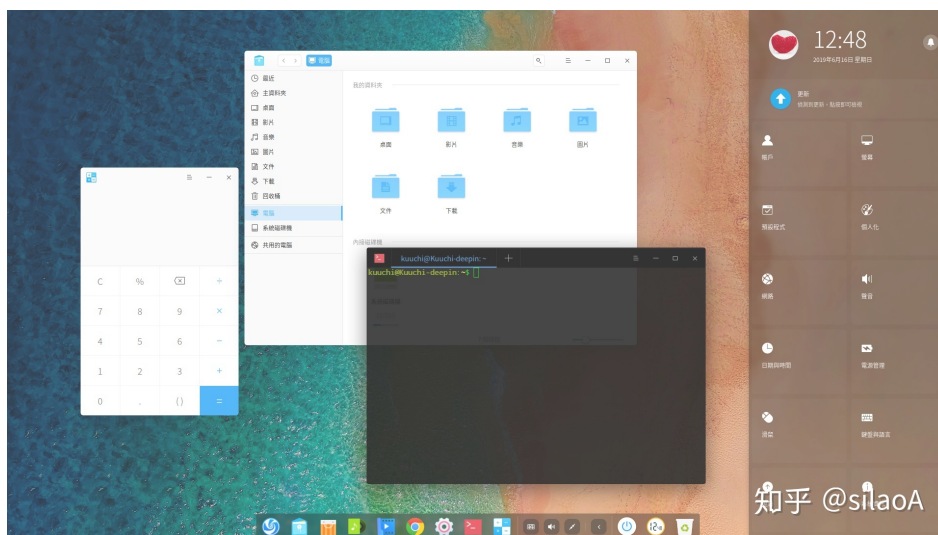
- 客户端。Xlib 及其替代者 XCB 开发库过于靠近底层，于是就出现了进一步封装、抽象级别更高的GUI库，如QT、GTK等，比方说前者提供绘制点线面的方法，后者提供绘制按钮、滚动条、下拉菜单等控件的方法，更高层次的抽象对于简化开发工作的重要性不言而喻。在众多X客户端程序中，有一个比较著名——xterm，它是基于X的终端模拟器，兼容VT102，也是X研发初期所计划的少数客户端程序之一。GNU/Linux发行版中设备/dev/tty1~tty6对应着6个全屏纯字符的终端模拟器，通过Ctrl-Alt-F1~F6切换，显示能力有限；如果运行到图形环境（通过Ctrl-Alt-F7切换），一般都集成了xterm，可配置的特性十分丰富，其他的诸如Gnome Terminal、Konsole、xfce-term等终端模拟器，或基于xterm，或参考xterm，增加了更多高级功能，如tab标签页、图片显示等。
- 窗口管理器。常见如Metacity、Mutter、KWin、vtwm、Xfwm、Compiz等，功能、风格各异，有的注重简洁高效，有的注重外观酷炫。
- 服务器。X.Org Server 占绝对统治地位，但也有其他衍生变种或移植，如XQuartz以及Cygwin/X、Xming、WeirdX，它们所支持的系统平台、底层库也都不一样。

需注意的是，**X Window System诞生比Linux早，也不是UNIX或Linux内核的一部分，而是用户态的软件组件，用户在系统上可以选择启用或禁用X。**

#### • 桌面环境

X Window System各个部分都有着多种实现，X规范本身特别偏底层，加之GNU/Linux自由开放的特点，有不少项目在X规范基础之上进一步封装，并把上述各组件、图标、主题、及其他特色组件打包在一起，形成开箱即用的桌面环境（Desktop Environment），常见如Gnome、KDE、Unity、XFCE、LXDE等，不一一列举，每种桌面环境都有不少用户群体，所支持的系统平台、底层库也各不一样。

另值得一提的是，中国武汉深之度科技公司基于Debian（早先基于Ubuntu）打造的GNU/Linux发行版Deepin，拥有自己的桌面环境DDE，即Deepin Desktop Environment，Deepin系统在DistroWatch上排名比较靠前，DDE也同时支持其他GNU/Linux发行版。



Deepin桌面环境 图源：Deepin官网

## • 应用场景

X11规范的模块化设计，最大的好处是X Client和X Server相互独立，并不需要了解彼此所处的硬件、软件环境。常见的X应用程序诸如 xterm、输入法、Web浏览器、Office套件等等，部署在一台Linux主机上，如果这个主机也同时部署了X Server，那么运行这些X应用程序毫无障碍，这是作为桌面计算机的常见应用场景。

另一种应用场景，Linux主机通常作为服务器使用，未配备比较好的图形显卡等硬件，也没有必要部署X Server，从终端登录到Linux主机也仅有字符界面，那X应用程序该如何运行呢？可以在图形显卡等硬件设备比较好的Windows/Mac主机上部署X Server，保持与Linux主机网络连通，在Linux主机上写好配置文件，配置好X Server地址和X11请求转发（X11 Forwarding），从任意一个终端登录到Linux主机并启动X客户端应用程序，就可以使程序的图形界面显示在Windows/Mac主机上，并与用户交互。事实上，在1990年代初期，就有这样的硬件设备，称为“X终端机（X terminal）”，专门部署X Server，将远端UNIX主机上的图形界面程序显示出来，这也正是MIT研发X的初衷之一。



X 终端机 图源：维基百科

从场景描述看出，与常规Client/Server所不同的是，通常X Server部署在本地用户主机上，监听、调控本地用户主机的硬件设备，而X Client可能部署在远端服务器/嵌入式设备上，也可能在本地。

## 0x02 X的竞争对手





一方面，X工作在用户空间，X客户端程序不能直接访问显示驱动、X Server与X客户端及窗口管理器之间繁琐的通信流程，拉低了绘图效率；另一方面，通过扩展、补丁手段允许在本地直接访问显示驱动，又造成稳定性问题。总之，X的设计随着新技术发展而显得臃肿、过度复杂，背负了沉重的历史包袱。

## • Wayland

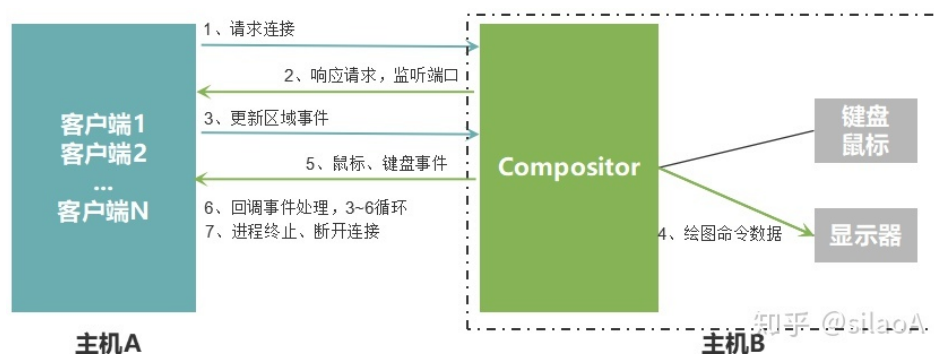
2008年，红帽公司（RedHat）的开发者Kristian Høgsberg利用业余时间搞起了Wayland项目，2010年Wayland加入了Freedesktop.org项目，项目官网为<https://wayland.freedesktop.org>，代码开源地址[cgit.freedesktop.org/wayland](https://cgit.freedesktop.org/wayland)。

Wayland是一个新的图形窗口系统方案，一套旨在取代X的新规范。与X最大的不同是，Wayland将X中的Server和窗口管理器整合到一起作为服务端，称为合成器（Compositor），架构上只分了客户端和合成器两大部件，有没有觉得终于看着舒服了一些。

- 客户端（Wayland Client），直接计算各自界面的渲染缓冲数据，客户端程序需要和渲染库（如OpenGL）链接。
- 合成器（Wayland Compositor），汇总所有客户端的渲染数据，实现各界面窗口“合成”，最后交给显示驱动绘图。Wayland项目提供了两套底层库libwayland-server和libwayland-client，简化图形程序开发，还给了一个Compositor参考实现——Weston。

假定多个Wayland客户端程序在主机A上，某个Wayland Compositor（如Weston）运行在主机B上。

1. 某个客户端进程启动，向主机B发送连接请求，目标地址可通过命令行或配置文件指定，如果给定的地址已有Compositor正在监听端口，则进行下一步；
2. 主机B上的Compositor返回一个连接正确响应，Compositor也可以配置接受或拒绝某些地址的请求；
3. 客户端自行生成UI界面和渲染缓冲，不需要向Compositor发送绘制请求，但需要发送更新区域事件，告知渲染缓冲中更新了哪些内容；
4. Compositor综合各客户端的区域更新事件，重新合成整个屏幕，并交给显示驱动绘制图形；
5. 运行过程中，Compositor可能收到主机B上的鼠标、键盘事件，经计算后，Compositor决定发给哪个客户端（即获得焦点）；
6. 客户端收到鼠标、键盘事件后，回调事件处理；
7. 循环第3~6，直至客户端收到关闭事件，进程终止、连接断开。



Wayland绘图过程示意

Wayland适用于桌面设备，也适用于移动设备，原来基于X的GUI库、桌面环境很快适配了对Wayland的支持，如Qt、Gnome、KDE、Compiz等，Ubuntu更是早在2010年就将Unity切换到适配Wayland。Wayland依赖了较多Linux内核技术，不易移植到其他系统平台。

## • XWayland

Wayland开发者在X.Org Server基础上打上系列补丁，称为XWayland，目的是在Wayland规范之上搞成新的X Server，在Wayland环境中运行X客户端程序时，Wayland合成器（如Weston）调用XWayland进行服务，以便在过渡到Wayland过程中保持对X的兼容性。

## • Mir



2013年，Ubuntu背后的开发商Canonical公司搞出来Mir项目，计划用Mir取代X并兼容X客户端程序，并且同时支持桌面设备和移动设备，统一用户体验。这种搞分裂的行为招致了批评，万万没想到的是，2017年Canonical公司又放弃了Mir项目并加大对Wayland的支持。

- 其他
- 苹果公司在MacOS和iOS中，使用Quartz作为自己的图形窗口系统。
- Android系统上，使用SurfaceFlinger作为自己的图形窗口系统。
- 历史上还有一些激进的方案，逐渐被淘汰。

## 0x03 总结

尽管有这样那样的问题，X仍然是GNU/Linux、UNIX上的主力，Wayland作为强有力的竞争对手，在远程桌面方面亦存在问题，重度依赖Linux内核技术不易移植到其他系统平台。在微软的WSL崛起之时，X的优势倒可以尽情发挥出来。

敬请期待下一篇，在Windows上折腾X。

## 参考

- [腾讯科技：电脑操作系统图形用户界面38年进化史](#)
- [en.wikipedia.org/wiki/X...](http://en.wikipedia.org/wiki/X...)
- <http://www.xfree86.org>
- <https://www.x.org>
- [en.wikipedia.org/wiki/X...](http://en.wikipedia.org/wiki/X...)
- [invisible-island.net/xt...](http://invisible-island.net/xt...)
- <https://wayland.freedesktop.org>

## 更多阅读

- [上一篇 Cygwin系列（十一）：折腾终端2](#)
- [伪码人专栏目录导航 持续更新...](#)
- [微软WSL——Linux桌面版未来之光](#)
- [Linux Cygwin知识库（一）：一文搞清控制台、终端、shell概念](#)
- [GNU Wget 爬虫？试一试](#)
- [Python操作Excel文件（0）：盘点](#)
- [silaoA的博客.https://silaoa.github.io](https://silaoa.github.io)

---

如本文对你有帮助，或内容引起极度舒适，欢迎分享转发与留言交流

► 本文为原创文章，如需转载请私信知乎账号silaoA或联系公众号伪码人（We\_Coder）。

都看这里了，不妨点个赞再走呗

发布于 2020-04-25 19:33，编辑于 2022-07-07 10:17

「真诚赞赏，手留余香」

赞赏

还没有人赞赏，快来当第一个赞赏的人吧！



写评论... | 你赞过作者的回答

8 条评论

默认 时间



PEMudy

...

感谢，写的很好，真的解惑了

05-10 · 作者回复了

回复 赞



silaoA 作者

...

点赞关注转发3连走起🤖

05-10

回复 赞



ccdad

...

看了很多X方面的文章,你写的最好,给你点赞

2021-07-30 · 作者回复了

回复 赞



silaoA 作者 > ccdad

...

上wine

2021-08-05

回复 赞



ccdad > silaoA

...

能写出来就是好的,我最近也在折腾X window, 我们公司想把window 产品移植到linux 上,真是一个头两个大

2021-08-05 · 作者回复了

回复 赞

展开其他 1 条回复 >



追随者

...

下一篇啥时候出啊? --" 在Windows上折腾X "

2020-07-24 · 作者回复了

回复 赞



silaoA 作者

...

小小透露一下，大概会写在Windows和Cygwin里安装X Server，对比效果。本人10月有考试，应该在10月之后了。

2020-07-24

回复 1

文章被以下专栏收录



伪码人 We\_Coder

专注Cygwin、Linux、电气等技术交流分享