

10 | 输入和输出设备：交互的演进

2019-05-17 许式伟

许式伟的架构课

[进入课程 >](#)



讲述：姚迪迈

时长 10:56 大小 10.02M



你好，我是七牛云许式伟。

前面我们已经介绍了操作系统的存储管理：内存和外存。今天，让我们来聊一聊操作系统是如何管理输入和输出设备的。

输入和输出设备有非常非常多。例如输入设备除了你常见的键盘、鼠标、触摸屏外，还有一些采集信息的传感器，如 GPS 位置、脉搏、心电图、温度、湿度等。我们会把关注点收敛在人机交互相关的设备管理上。

交互的演进

在计算机外部设备的演进上，人机交互设备的演进毫无疑问是最为剧烈的部分。

计算机刚出现的时候，人们使用“**打孔卡 + 打印机**”作为人机交互方式。这个时期还没有操作系统，表达意图非常痛苦，只能在打孔卡上打孔来发送指令。

但很快，人们找到了“**键盘 + 显示器**”，这对最为经典的人机交互设备。而随着计算机使用人群越来越多，以及使用场景的变化，人机交互的方式也相应地发生了一次又一次的改变。

时期	输入设备	输出设备
“史前”时期	打孔卡	打印机
字符界面时期	键盘	显示器
图形界面时期	键盘+鼠标	显示器+音箱
移动时期	触摸屏+麦克风	触摸屏+内置扬声器
IoT萌芽期(纯语音)	麦克风	内置扬声器
拟真交互(可能的未来)	摄像头+麦克风	VR

总结这些改变，我们会发现，人机交互在往越来越自然的方向发展。所谓自然，就是越来越接近于两个人直接的面对面沟通。

那么，这些人机交互的改变，对于操作系统来说又意味着什么呢？

输入设备

我们先看输入设备：键盘、鼠标、麦克风、摄像头。

键盘

键盘输入的管理机制和窗口系统的设计密切相关。为了让用户清楚键盘输入的目的地，窗口系统往往会有一个焦点窗口。

在窗口系统里面，窗口间还有父子关系，焦点窗口还会有父窗口，还有父窗口的父窗口，这些窗口属于活动窗口。

大部分情况下，键盘输入的事件会先发给焦点窗口，焦点窗口不处理则发给其父窗口，按此传递，直到有人处理了该按键事件，或者直到顶层窗口。

键盘从功能上来说，**有两个不同的能力：其一是输入文本，其二是触发命令**。从输入文本的角度来说，要有一个输入光标（在 Windows 里面叫 Caret）来指示输入的目的窗口。目的窗口也必然是焦点窗口，否则就会显得很 unnatural。

这个交互的呈现方式非常稳定，从 DOS 到 Windows/Mac，到 iOS/Android 都是如此。但是从触发命令的角度来说，命令的响应并不一定是在焦点窗口，甚至不一定在活跃窗口。

比如 Windows 下就有热键（HotKey）的概念，能够让非活跃窗口（Inactive Window）也获得响应键盘命令的机会。一个常见的例子是截屏软件，往往需要一个热键来触发截屏。

到了移动时代，键盘不再是交互主体，但是，键盘作为输入文本的能力很难被替代（虽然有语音输入法），于是它便自然而然地保留下来。

不过移动设备不太会有人会基于键盘来触发命令，只有常见的热键需求比如截屏、调大或调小音量、拍照等等，被设计为系统功能（对应的，这些功能的热键也被设计为系统按键）保留下来。

鼠标

鼠标输入的管理机制和键盘一样，和窗口系统的设计密切相关。但鼠标因为有位置，确定鼠标事件的目的地相比键盘事件要简单的多，大部分情况下，鼠标事件总是交给鼠标位置所属的窗口来处理，但也会有一些例外的场景，比如拖放。

为了支持拖放，Windows 操作系统引入了鼠标捕获（Mouse Capture）的概念，一旦鼠标被某个窗口捕获，哪怕鼠标已经移出该窗口，鼠标事件仍然会继续发往该窗口。

到了移动时代，鼠标已经完全消失（虽然在智能手机之前还是出现过 WinCE 这样的支持鼠标的移动操作系统），取而代之的是触摸屏。窗口系统也和 PC 时期完全不同，在屏幕可见范围内只有单个应用程序占满整个屏幕，这让交互的目的地确认不再是个问题。

麦克风

麦克风是一个非常有潜力的下一代输入设备。今天 IoT 领域如汽车、智能音箱都是很好的发展语音交互能力的场景。包括今天大行其道的手机，语音交互也是一个很好的补充。

交互方式不管怎么变化，其核心需要实现的都是这样的两大能力：输入文本和触发命令，这一点是不变的。

语音交互今天仍然还很不成熟，究其原因，语音交互在 IoT 领域还停留在触发命令为主，且哪怕是触发命令这一件事情，也还有重重关卡需要去突破。

在手机软件中，语音输入文本在部分软件中已经有较多应用，但是主要优势还在日常用语和长文本，在个性化场景如“输入人名之类”，较难达到好的结果。

从更本源的角度看，语音交互今天仍然在相对封闭的应用技术场景里面发展为主，而作为操作系统的主体交互手段，其能力必须是开放的。因为操作系统是开放的，场景是开放的。

摄像头

摄像头作为交互设备，除了引入语音，也引入了手势、表情。从表达能力来说，这是最为丰富也是最为自然的一种表达方法。但是技术所限，这种交互方式还只在萌芽阶段。微软的 Kinect 是一个非常经典的案例，它能够让玩家通过语音和手势发指令来玩游戏。

输出设备

输出设备主要负责向用户反馈信息。比如：显示器（显卡）、音箱（声卡）、打印机。输出设备的演化并不大，最主要的输出设备还是以显示器为主。

显示器

显示器虽然经历了 CRT 到液晶屏多代更新，但也只是支持的色彩更多（从黑白到彩色到真彩色），分辨率越来越高。实际上，从操作系统的软件治理角度来看，显示器并没有发生过实质性的变化。

为了让不同软件可以在同一显示器屏幕上呈现，操作系统引入了窗口系统的概念。每个软件有一个或多个窗口（Window，有时候也叫视图，即 View）。在 PC 操作系统中，不同窗口还可以层叠（Cascade）或平铺（Tile）。

通过引入窗口，操作系统在逻辑上把显示器屏幕这个有限的设备资源，分配给了多个软件。和 PC 不同的是，移动设备由于屏幕过小，所以操作系统选择了让软件的顶层窗口全屏占据整个屏幕。这让显示器屏幕的管理变得更为简单。

除了窗口系统，显示设备管理的另一大挑战是绘制子系统。窗口里面的内容是什么，呈现成什么样子，完全是软件来决定的，这就意味着软件需要绘制能力。

绘制能力牵涉面非常之广，在操作系统里面，往往有一个独立的子系统（通常叫 GDI）与之对应。这里我们简单罗列一下 GDI 子系统会涉及哪些东西。

2D 图形相关。包含 Path(路径)、Brush(画刷)、Pen(画笔) 等概念。

3D 图形相关。包含 Model(模型)、Material(材质)、Lighting(光照) 等概念。

文本相关。包含 Font(字体) 等概念。而字体又分点阵字体和 TrueType 字体。TrueType 字体的优势是可以自由缩放。今天我们见到的大部分字体都是 TrueType 字体。

图像处理相关。包含 Bitmap(位图) 对象及常见图像格式的编解码器 (Encoder/Decoder)。

窗口系统结合输入设备对应的事件管理系统、绘制 (GDI) 系统，我们就可以随心所欲地实现各类用户体验非常友好的视窗软件了。

但是，为了进一步简化开发过程，操作系统往往还提供了一些通用的界面元素，通常我们称之为控件 (Control)。常见的控件有如下这些：

静态文本 (Label);

按钮 (Button);

单选框 (RadioButton);

复选框 (CheckBox);

输入框 (Input, 也叫 EditBox/EditText);

进度条 (ProgressBar);

等等。

不同操作系统提供的基础控件大同小异。不过一些处理细节上的差异往往会成为跨平台开发的坑，如果你希望一份代码多平台使用，在这方面就需要谨慎处理。

音箱

相比显示器的管理，音箱的设备管理要简单得多。我们很容易做到多个软件同时操作设备，而有合理的结果。

例如，调整音量我们遵循覆盖原则即可，谁后设置音量就听谁的。而声音的播放则可进行混音处理，多个软件播放的声音同时播放出来，让人听起来像是同时有多个人在说话。

当然，特定情况下要允许某个软件禁止其他软件播放出来的声音，比如接听电话的软件，需要在电话接通的时候屏蔽掉所有其他软件播放的声音。

打印机

打印机的管理方式又很不一样，软件使用打印机的过程基本上是互斥的。一个软件在打印文档的时候，其他的软件只能等待它打印完毕后，才能进行打印。

打印机的使用是以文档为互斥的单位。为了避免软件之间出现长时间的相互等待，操作系统往往在打印机的管理程序中引入很大的打印缓冲。

软件操作打印机的时候，并不是等待打印机真把内容打印出来，而是把文档打印到打印缓冲中就完成打印。这样，在大部分情况下多个软件不需要因为使用打印机而出现相互等待。

结语

后面我们在谈“桌面开发”一章中，还会涉及人机交互的更多细节，这一章侧重点在于领域无关的通用操作系统相关的问题域，相关的内容这里仅做概要性的阐述。

但是，仅通过简单对比所有输入和输出设备的管理方式，我们就可以看出，不同输入和输出设备的管理方法差异非常大，没有太大的共性可言。

尽管对 CPU 而言，所有外部设备有着相同的抽象，但这些设备的业务逻辑却如此不同，并不能统一抽象它们。正是因为有了操作系统这样的基础软件，这些设备业务逻辑的复杂性才从我们的软件开发过程中解放出来。

人机交互演化的核心变化是输入设备的变化。我们看到，输入手段的变化是非常剧烈的，且每一次演变都是颠覆性的变化。

事实上输入意图的理解越来越难了，**因为交互在朝着自然（Nature）和智能（Intelligence）的方向发展**。我们不可能让每一个软件都自己去做输入意图的理解（今天的现状是每个软件自己做），**在未来，必然将由操作系统来实现智能交互的基础架构**。

今天的内容就到这里。你对今天的内容有什么思考与解读，欢迎给我留言，我们一起讨论。如果你觉得有所收获，也欢迎把文章分享给你的朋友。感谢你的收听，我们下期再见。



许式伟的架构课

从源头出发，带你重新理解架构设计

许式伟
七牛云 CEO



新版升级：点击「 请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 09 | 外存管理与文件系统

下一篇 11 | 多任务：进程、线程与协程

精选留言 (19)

写留言



jueyoq
2019-05-17

11

交互即人机交流。
对于交流，人们更趋向于使用更自然更本能的方式。
人做为中心，我们就更加注重输入设备的优化。

从使用上来说，本质上输入设备要解决的问题就是文本的输入与命令的解析 命令的解释 是离散化的，而未来 会逐渐过度到连续化的意图的识别。...

展开 ▾

作者回复: ㇏



涵

2019-05-17

👍 7

老师好，现在输入设备越来越多样化，意图理解越来越复杂，甚至需要专用的芯片，而如果人机交互意图理解放在操作系统层，会否每一种输入设备都需要一套操作系统?如果是这样，那么会否人机交互意图理解的架构下沉到硬件也就是芯片层，例如人脸识别?谢谢。

作者回复: 人机交互肯定需要统一的交互范式设计。从这个角度来说，把意图理解放到芯片层并不划算，因为迭代反而慢了。



我在你的视...

2019-05-18

👍 4

越来越nature，意味着越来越artifiaci。估计设备最好的交互就是类人，而不是代替人。

作者回复: 嗯



苟范儿

2019-05-17

👍 3

老师今天讲的输入、输出设备交互的变化也是针对终端设备的。而对于开发者而言，一般理解的架构更多的是服务端架构、后台架构，掌握这些人机交互对我们这些后台开发人员所理解的架构设计有什么影响呢?

展开 ▾

作者回复: > 对于开发者而言，一般理解的架构更多的是服务端架构、后台架构

这个是错误的理解。桌面端软件也是需要架构的。如果你定位自己是后台开发人员，那么理解交

互确实意义不大，但是如果希望自己是一个优秀的架构师，还是需要关注交互对我们架构的影响的。



Tesla

2019-05-17

👍 2

老师好。人机交互智能化，更趋近于人与人的交流。但是就算人类面对面沟通，也有很多的理解歧义 还会受限于表达能力。这样是不是意味着，除非有其他学科领悟的理论算法支持，如心理学这类，否则智能交互很难在交互效率和交互准确性上做到平衡取舍？

作者回复: 这有一定的可能性。人机交互范式目前是确定性交互，但是越来越模糊是趋势，最终如何我觉得还不太好判断。



Being

2019-05-28

👍 1

老师，您说的由操作系统来实现智能交互的基础架构，可不可以理解为，对于智能处理，在操作系统抽象出一套标准作为一大模块，为以智能为核心的交互提供统一处理，让交互则更聚焦于场景。

展开 ▾

作者回复: 是的



williamcai

2019-05-27

👍 1

许老师，操作系统层面支持智能交互，当前现状是咋样的

展开 ▾

作者回复: 现在还只是萌芽阶段



Adoy

2019-05-18

👍 1

让我想到了罗永浩的TNT（狗头护体）

展开 ▾



Aaron Che...

2019-05-18

👍 1

补打卡10 交互是非常重要的的一环

展开 ▾



有铭

2019-05-17

👍 1

我认为人机交互其实就是所谓的大前端

展开 ▾



wucx

2019-05-31

👍

通用能力在未来会下沉来提高上层的效率及建立标准。

多样的输入处理能力、标准范式会在未来操作系统上，这个和微服务中的service mesh是一样的感觉，让软件/应用更关注提供的服务，而非基础的能力需要，这个找底层拿来用就好了。



Mr_杨

2019-05-31

👍

操作系统识别语音，动作还是有一定难度，毕竟不像文字等有标准的字符集，应该还是有专业的软件做这些解析。



Luke

2019-05-28

👍

我感觉出老师的一个意图或者观点：无论未来的交互形式如何变迁，从系统架构的角度考虑，本质上还是在做输入输出的角色，输入输出都应该放在系统层面统一处理，提供服务给外围软件或者上层。

老师列举了输入输出设备的变迁，是否也是想说明过去的发展一直是遵循这个规律的， ...

展开 ▾



醉里挑灯...

2019-05-21



目前输入设备的操作，鼠标键盘，麦克风，摄像头，而输出其实更常见的就是显示器窗口，架构迭代的话，可能就是ai人工智能，立体投影交互，就跟科幻大片中的那种操作吧，对于将输入放在操作系统层来统一，只需要开启系统通过某一种形式输入指令操作系统进一步做我们想要做的事情，指令处理完成，立体展现，是不是输入输出都可以统一处理了，这样就不需要不同的规范了！

展开 ∨



Geek_9102

2019-05-21



在未来，必然将由操作系统来实现智能交互的基础架构。未来是这样，是不是也意味着，如果只能交互的应用，会逐渐被系统公司所吞并，商业模式也会有问题

展开 ∨



Geek_88604...

2019-05-19



在洋葱架构和整洁架构中平等地看待输入和输出，不知道许老师是什么看法？

展开 ∨



Fz

2019-05-18



操作系统作为软件运行的必要基础。小软件要智能。这个管家更需要智能，才能更好和应用软件配合，服务人类



八哥

2019-05-18



语音输入要能解决两个问题就好很多，1.识别正确率，2.能在边缘端自己处理大多数。即离线处理能力。VR输出最开始大规模应用估计会在游戏，教学和医疗诊断。

展开 ∨



觉

2019-05-17



感恩大佬分享 随喜大佬

展开 ✓