

带您走进全新Windows的世界

perfect, Windows 10

十全十美



帶 您 走 進 全 新 Windows 的 世 界

目录

第一章 外观新印象 /P1

第一节 窗口与任务栏 /P2

第二节 Modern UI 的调整 /P5

第三节 外观发展的趋势 /P7

第二章 新改进, 新技术 /P10

第一节 开始和操作中心 /P11

第二节 窗口管理的改进 /P14

第三节 细微之处的改进 /P16

第四节 DirectX 12 /P18

第五节 Dism 与 Compact /P20

第三章 无处不在的应用 /P23

第一节 应用体验的改进 /P24

第二节 全新的电脑设置 /P28

第三节 Cortana /P31

第四节 Microsoft Edge /P34

第四章 通用应用的深究 /P39

第一节 初识通用应用 /P40

第二节 构筑开发环境 /P42

第三节 开发通用应用 /P43

附录一 技术文章大全 /P56

附录二 Windows 10 开发历史 /P66

序言

“有人问过我，Windows 10 相对于以前的系统，改进了什么？我一时语塞，后来就有了这篇文章的创作。”

——主编

Windows 10 已经正式发布了，作为看着 Windows 10 一步一步开发完善的 Insider，我们心中有感而发，决定做一个别人都还没有开始做的 Windows 10 新功能介绍文章，这就是《十全十美，Windows 10》

《十全十美》这个计划的提出大概是三个月以前吧，在和兔子一次无聊的闲聊中他提到了这个想法（结果大家都懂），我想都没想就同意了，于是便开始不负责任的干活（由于身体和内心的原因偷懒 N 长时间），基本上我见证了《十全十美》从两三行乱糟糟的文字到现在作为一篇文章发布给大家，我用自己体验告诉大家，这本书我们的确是用心做的。

我们虽然没有那么浩浩荡荡的编写队伍，但是每个人都是独一无二、各有所能的，这本书离不开我们的通力合作，一字一句反复斟酌，每一个章节都是经过层层审核最终呈现给大家的，即使在发布后我们还是会不断进行修改，力争做到没有错误，客观、真实的给大家展示 Windows 10 的各个方面的优缺点，让大家能够深入的了解就是我们最初的宗旨。

我觉得，Windows 10 承载了太多太多，微软大一统的愿望，用户经过 Windows 8 的巨变之后要求的妥协太多太多难以言表，所以我们认为，Windows 10 是一个值得让我们为他而付出的操作系统，我们虽然在编写的过程中遇到过瓶颈，但是我们的那份信念让我们坚持了下来，没有放弃，没有半途而废。

现在，这本汇聚了我们整个编写组心血的《十全十美》即将全本发布，但这并不是重点，这只是一个开端，这就像 Windows 10 的 RTM，并不意味着结束，而会有一个一个的补丁，对其进行修改，我们也会不断完善，查漏补缺，让《十全十美》做到尽可能的完美，这就是我们的愿望。

最后祝你身体健康，生活顺心，阖家幸福。

由几个穷学生组成的《十全十美》编写组敬上

Project Group

主 编：ITechDeveloper

副 主 编：不离不弃的包子

lzj20021216

eugenewangtw

美术设计：OverWhite

贴吧运营：若夏黎光

编写人员：Mouri_Naruto

123dfdc5

最爱龙星

driver1998

感谢为梦想支持过我们的人：

wgl836912613

JiongDream

拿铁不加糖

diy371

blue_forever

Morn_zck

zhaoxile

宇文雨曦

13120339174

wondersnefu

厦门宇恒

山外的鸭子哥

本书完全出自个人兴趣目的，任何人不得以任何理由进行未经许可的添加或修改、补充行为以及禁止用于商业行为！

本书完全是非盈利的个人行为，字体应为个人免费授权：

方正兰亭黑、方正兰亭细黑、方正兰亭粗黑、方正兰亭特黑、方正大标宋、方正小标宋。

此外工作组不承担任何法律责任。



窗口与任务栏

Modern UI 的调整

外观发展的趋势

外观新印象

第一节 窗口与任务栏

外观的变化，可以从窗口和任务栏中窥知一二。

窗口

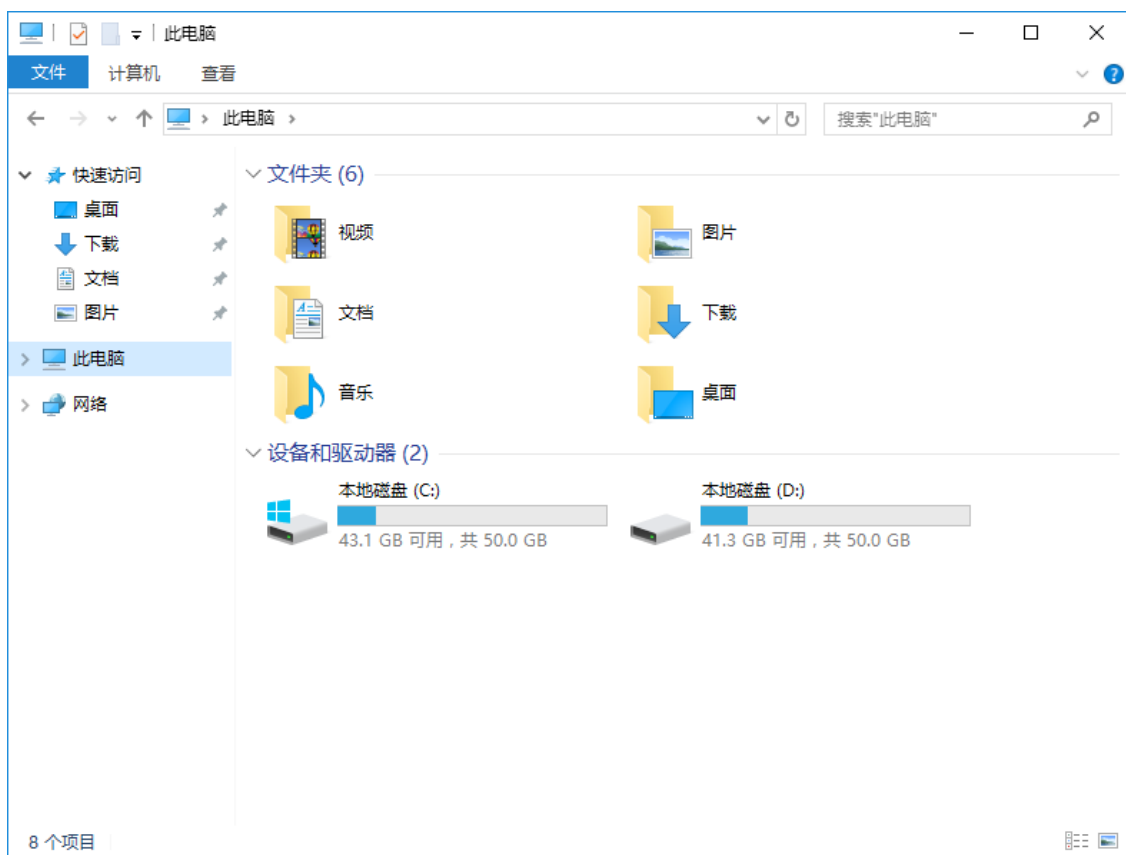


图 1-1 Windows 10 的窗口

Windows 10 的窗口借鉴了Office的设计风格,采用“无边框”设计(图1-1(其实四周仍然存在一个像素的边框));这种设计沉浸了工作区域,弱化了窗口的存在,同时激活的窗口四周带有阴影效果以方便识别(实质上这才是真正的边框区域)。

标题栏默认为白色,进一步体现了沉浸设计(可以通过手段修改颜色);最小化,最大化以及关闭按钮经过重绘,易于触摸,也更凸显线条感;标题栏文字改回居左,更符合Windows一向的传统。

而对于 Windows 7 用户而言, 最大的外观变化是 Ribbon UI 的引入。这种工具栏设计充分考虑了键鼠操作的合理性, 相对于 Windows 7 的“组织栏”, Windows 10 的 Ribbon 栏可以以更简单的步骤执行文件操作。

窗口特效也得到了更新, 例如最大化时向屏幕四周延伸。这些新特效更使 你觉得 Windows 10 更加耳目一新。

任务栏与托盘



图 1-2 Windows 10 的任务栏



图 1-3 任务栏图标状态, 从左往右依次为未打开状态, 复制文件状态, 鼠标移过状态, 激活状态和后台状态。

任务栏外观(图 1-2)也有所改变, 意图向 Vista 致敬, 默认主题色为黑色。但有所不同的是, Windows 10 的任务栏外观不如 Vista 模糊, 实质上只是半透明效果。

任务栏左侧增加了搜索框(可打开 Cortana)和任务视图按钮; 开始按钮 和任务栏图标与之前相比都缩小了; 任务栏的图标状态(图 1-3)变得更加直观; 任务栏图标右键菜单和任务预览也跟随平面化设计, 重新设计了外观。

通知区域的图标经过了重绘, 同时更改了全局外观, 更显得平面化(图 1-4)。



图 1-4 系统托盘

电量显示中电池百分比字号加大, 同时加入了节电模式和亮度按钮帮助你延长电池寿命。音量调节增加了当前音量显示, 音量进度条改成左右拖动, 同时也显示当前的声音设备及其名称。略微麻烦的是音量合成器必须通过音量图标右键打开。

网络选择经历了 Windows 8 的侧边栏样式后重回窗口样式, 不过相较于 Windows 7, 网络选择吸收了 Windows 8 的网络开关。你仍然可以自由打开 WLAN 开关, 乃至使用飞行模式。

时间和日历页面去掉了指针式时钟, 而以放大后的数字时钟来显示当前时间, 紧接着时间的是当天日期; 日历区也有所放大, 选中的日期会带蓝色底板, 而选中当天日期后还会增加一个黑色方框, 以示区别。

第二节 Modern UI 的调整

为了大一统战略，小方块也有大改变。

汉堡菜单

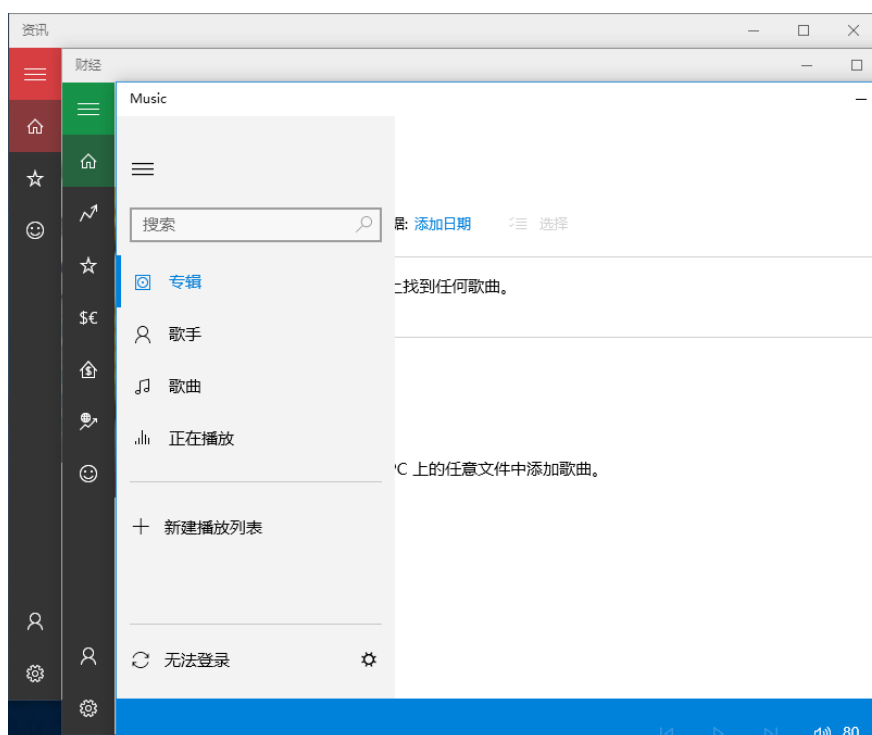


图 1-5 Modern 应用中的汉堡菜单

在 Windows 8 的应用设计规范中，“枢纽”菜单 (Pivot) 占据了应用的顶部空间。即使这种设计虽然简洁，但反响平平。微软为了大一统战略，修改了菜单框架。凡是符合 Windows 10 标准的应用，大多在左上角添加了汉堡菜单(图 1-5)。因为三条横线看起来很像汉堡的形状，故名“汉堡菜单”。

汉堡菜单并非微软的创新，在 Android、iOS 乃至网页的设计，它就已经广泛使用。考虑到用户学习成本，微软也采用了这种方案。

你可以点击汉堡按钮，扩展左侧菜单或查看文字说明；也可以直接点击汉堡菜单下的一排图标(如果有)。

不过这种设计还是引来了用户吐槽，主要体现在外观相似性和操作体验上。在这方面，微软仍要继续努力。

响应式菜单



图 1-7 一款应用在不同窗口大小下的布局

伸到较小的尺寸时还会显现不一样的变化。Windows 10 也修改了应用的页面滚动方向，竖向滚动更符合系统全局。

不过并非所有应用都支持响应式布局。以 Windows 8 标准开发的应用拉伸时就存在诸多问题。

既然要实行大一统战略，电脑端和移动端的应用显示就要保持相对一致。在 Windows 10 的应用设计规范中，来源于网页设计的响应式布局也得到了应用。响应式布局，顾名思义就是指能够根据对象大小而调整显示内容(图 1-7)。

你可以像桌面程序一样随意拉伸窗口，以调整显示内容，某些应用拉

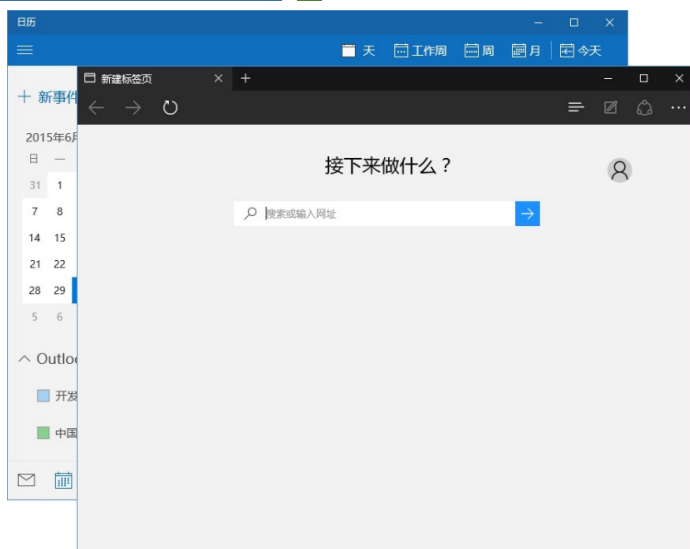
沉浸式状态栏与“暗黑”主题

Windows 10 也修改了 Modern App 的外观颜色框架(图 1-8)。早期，当应用窗口化时，标题栏都为系统的指定颜色，不仅过于乏味，还影响了部分应用的外观(例如早期的 Edge，标签栏在标题栏最下方)。

现在，开发者可以自定义标题栏颜色，以保持界面风格的统一，甚至还可以在标题栏上加控件。

为了与移动端界面风格统一，同时

兼顾用户长时间工作的需求，微软引入了暗黑主题。暗黑主题在 Windows 10 的其他地方也有所应用，例如任务栏、开始菜单和通知中心等。



第三节 外观发展的趋势

世界正不断的趋向扁平化设计。

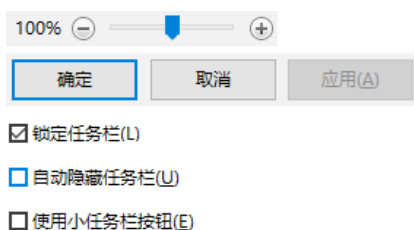


图 1-9 Windows 10 中部分按钮控件的变化



图 1-10 扁平化的右键菜单

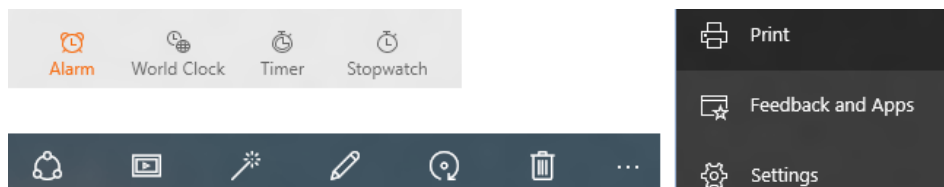


图 1-11 一些 Modern App 内的按钮控件

微软从 Windows 8 开始,就着手“压扁”桌面端的按钮控件。而在 Windows 10 内,重新设计的按钮控件更是体现了这一点。扁平化也成功渗透到右键菜单,新的右键菜单从视觉上来说风格更加统一(图 1-9 到图 1-10)。

Modern App 内的线条化图标也能证明这一点。这种图标设计,“细若蚊足,勾画了了”,而且兼顾直观(图 1-11),线条化图标甚至渗透到了系统其他地方,例如任务栏托盘区域的一些图标。



图 1-12 Windows 10 的新桌面图标

Windows 10 也重绘了大部分桌面图标 (图 1-12)。这些图标非常类似 NT6 的图标,但也出现了一些变化。微软在新图标的制作上付出了心血,使得其看起来大气,简约,也兼顾了平面化趋势。

尤其值得一提的是,新图标跟随了时代脚步。例如,手机图标被更新成了类似 Windows Phone 的形状。与此同时,新图标也不忘回顾历史,回收站的图标就从形状上看简直就是从 Windows95 上复制粘贴出来的

透明回归，无处不圆



图 1-13 Windows 10 中的“圆”

微软为了 ARM 设备性能和电池续航的妥协,从 Windows 8 的千万行代码中完全删除了有关透明效果的代码。于是,Windows 变成了各种色块的世界,虽然符合 Modern 的设计风格,但遭到用户的一致吐槽和反对。

在 Windows 10 中微软听取用户意见选择性恢复了透明效果 (例如开始菜单、通知中心)。这些效果可以在“设置”中关闭。另外,Windows 似乎是受到了 iOS 设计的启发,Modern 固有的方形被部分圆形设计所取代 (图 1-13)。

安装进度条和磁盘饼图经过了重新设计, 配色变得柔和, 同时也更加直观。其余的圆形设计变化, 更加凸显 Windows 10 是一个兼具现代和美观的操作系统。



Windows10 微百科

Windows Insider 计划

微软在每一个新系统正式发行之前, 都会向开发者提供预览版系统, 以便更快的找到 Bug 并解决。而 Windows Insider 的开放注册让更多的人加入到 Windows 10 的改进和完善的工作中来, 让 Windows 更加适应消费者的需求。早期的 Windows 版本虽然也有类似机制, 但意见常常不被微软采纳, 可以说这是直接导致 Windows 8 的失败的原因之一。

为了弥补这个遗憾, 微软在 Windows 10 中加入了 Windows Insider, 并完善了反馈机制。如果说你想试用预览版的话, 加入此计划是一个不错的选择。若你想加入此计划, 点击这个网站 <http://insider.windows.com/>。


截至发稿前夕, Windows Insider 计划在 Windows 10 的应用上已经告一段落。我们期待下一代的 Insider 计划是否有进步。

Windows 10 的默认壁纸

微软历来都很重视系统桌面壁纸, Windows XP 的蓝天白云更是给人带来了深刻印象。而“英雄”壁纸真是来之不易, 在于使用了照明技术拍摄, 不是用计算机合成的。

这张壁纸看起来富有科技感, 同时可能暗喻 Windows 10 将作为“英雄”拯救世界 PC 市场。Bradley 用一台 5000 万像素的相机拍了数千张照片, 比较了许多视觉效果, 比如烟雾, 激光和散景(由镜头产生的失焦效应)。最终的效果由 10 张照片合成, 它整合了不同的光影效果, 带来深度感。从制作过程中, 我们确实看到微软十分强调视觉效果, 也很想改变消费者看待微软的方式。

无论效果如何, 只要有心, 用户也欣慰了。



开始和操作中心

窗口管理的改进

细微之处的改进

DirectX 12

Dism 与 Compact

新改进，新技术

第一节

开始和操作中心

有时候，用户的力量是无穷的。

开始菜单



图 2-1 Windows 10 的开始菜单

尽管开始屏幕展现出了微软即将迈向移动终端的想法,但其使用逻辑可能仍然不令用户满意,于是微软重新设计了开始菜单。从外观上看,新的开始菜单有些类似 Windows 7 和 Windows 8 的集合。

开始菜单左侧为 Windows 7 样式,记录了最常用的或是新安装的应用;左下角是一些链接,默认指向文件管理器和电脑设置(可从电脑设置中调整,图 2-2)。

单击“电源”可以像以前一样进行关机、重启等操作,单击“所有应用”,也可以像以前一样检索 Modern 应用和开始目录下所有文件夹,

按首字母顺序排列。当你点击这些首字母的时候,就会出现一个检索页面用于快速检索(图 2-3)。

开始菜单右侧为 Windows 8 样式,可用来固定应用以及快捷方式(方法是在开始菜单左侧或桌面环境对准程序右键,点击“固定到开始菜单/屏幕”),甚至还能给磁贴组重命名(方法是鼠标悬停到标题区,点击右侧等号图标更名磁贴组)。而一个视觉上的改动是跟随 Modern UI 的变化改为纵向滑动。

开始菜单可以调整高度或右侧磁贴区的指定宽度(方法是鼠标移至开始边框,直至出现箭头后拖动鼠标调整)。



图 2-2 开始菜单的链接

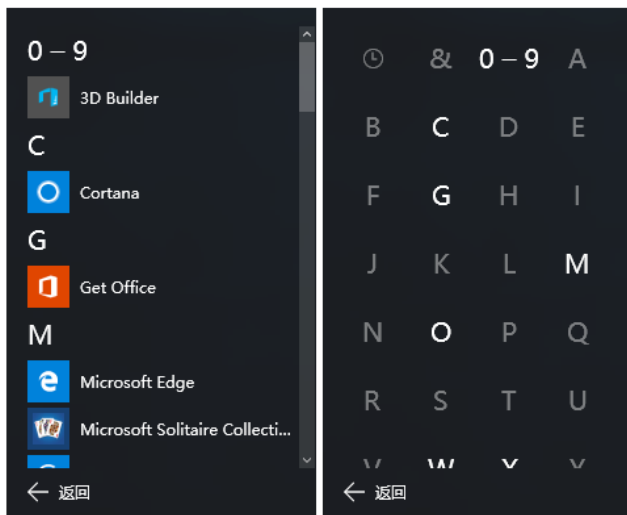
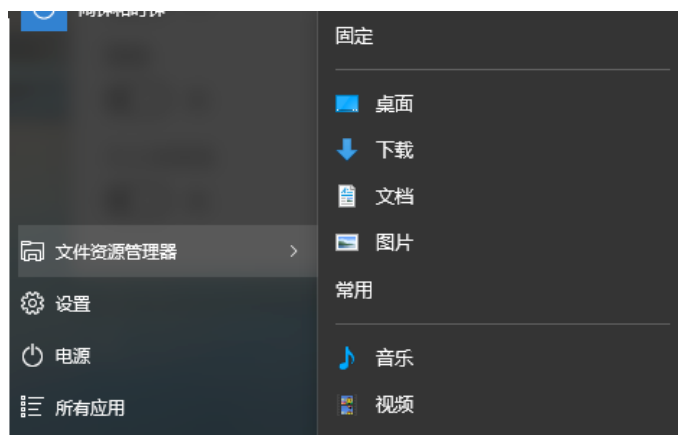


图 2-3 所有应用列表

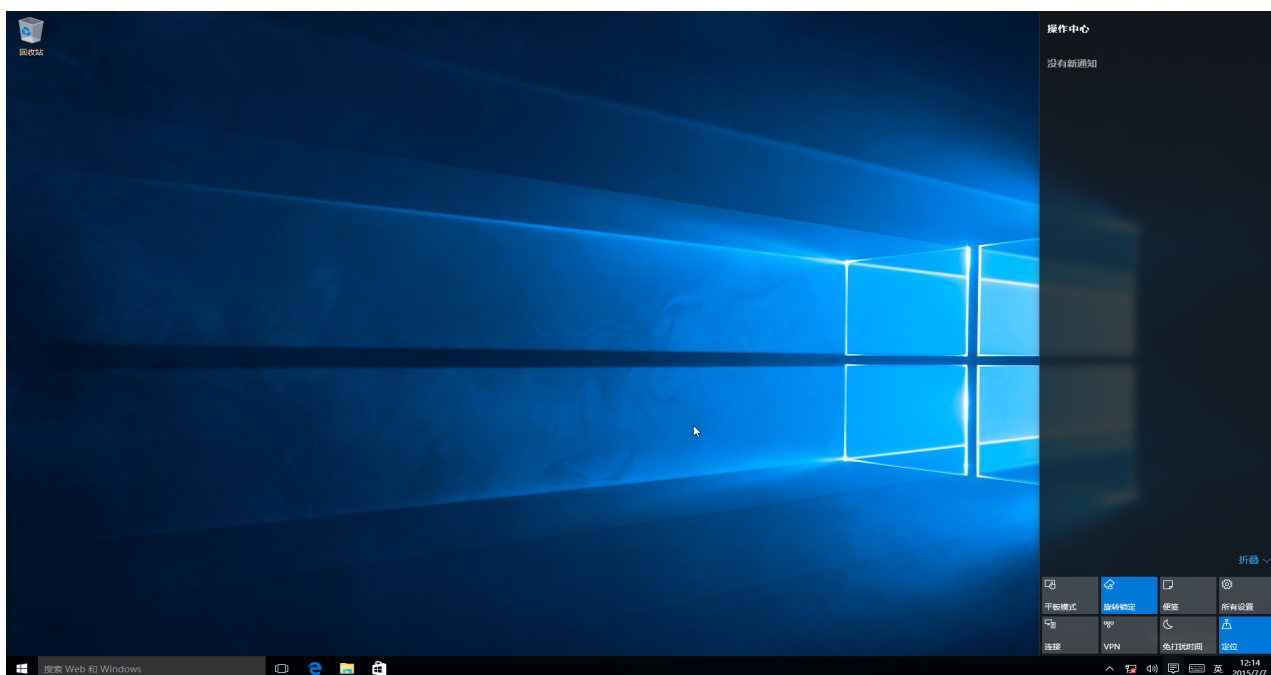
开始菜单也恢复了跳转列表, 可以点击**项目旁边的小箭头**展开(图 2-4)。




如果你在设置中选择打开全屏开始, 或使用平板模式, 则 Windows 10 会启用另一种样式的开始(图 2-5) 全屏开始模式外观更类似 Windows 8 中“开始屏幕”的模式, 不过也具备了 Windows Phone 的一些特征。单击左上角的汉堡菜单可以展开左侧列表; 当然, 如果你只是想关机或进入所有应用列表, 单击左下角的两个图标, 其他操作方法则与普通开始菜单



图 2-5 全屏模式开始



操作中心(图 2-6)位于屏幕右侧,是 Windows 7 同名功能。但是在 Windows 10 中它已不再只是显示系统维护信息,而是所有通知的集合站。操作中心可从任务栏上的  打开。

说到操作中心,就不得不说到系统通知的方式。一是传统的程序通知(图 2-7)。这类通知以前是作为通知气泡存在的,而现在都统一为长条状形式展现在屏幕右下角。二是新出现的交互式通知(图 2-8)。这类通知多用于 Modern 应用的提醒。例如,当其他人回信给你时,你就可以快速通过“交互式通知”的输入框来回复;或者,当你预定的时间到达时,你可以选择推迟或取消提醒。

操作中心被分为两个部分。上方是通知区域,用于收纳系统通知。如果文字一行内显示不完,你可以点击文字右侧的箭头展开。若要清除通知,你可以逐一点击通知右上方 × 号或按右上角的“全部清除”。下方是操作区域,你可以在这里进行一些快速设置,也可以单击“展开”或“折叠”以显示或隐藏设置按钮。当然,你可以在“设置”中改变最上面一行菜单的设定,然后在收拢状态时能够更方便的控制。

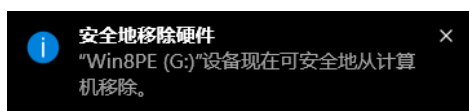


图 2-7 程序的通知

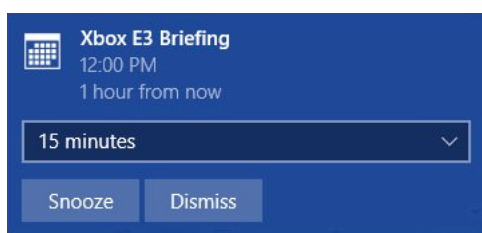


图 2-8 交互式通知

第二节 窗口管理的改进

世界正不断的趋向扁平化设计。

任务视图和虚拟桌面

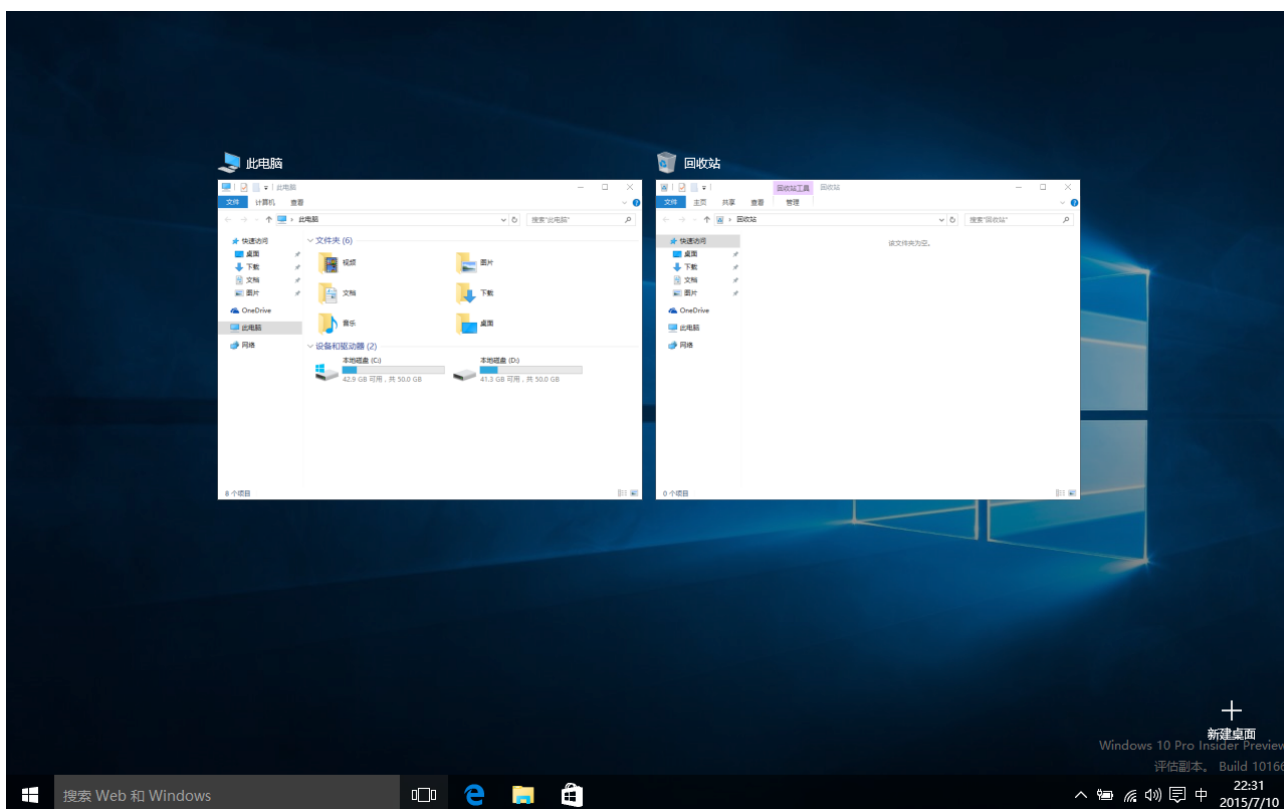


图 2-9 Windows 10 中的任务视图

任务视图和虚拟桌面 (图 2-9) 是 Windows 10 中的一项新功能。你可以按任务栏上的

 或键盘上的  +  来打开。

上方是当前运行的任务缩略图, 可以点击来切换, 或者鼠标移到缩略图, 当出现关闭按钮时可以关闭这个任务; 下方是虚拟桌面区, 可以切换虚拟桌面。如果只存在一个桌面, 则只在右下角显示添加按钮。

打开虚拟桌面, 你可以执行更多任务, 甚至可以进行桌面分工管理; 同时任务视图也是各自显示任务。当你在一个桌面打开另一桌面的任务时, 会出现一个过渡动画, 然后跳转到另一桌面打开任务。

你也可以按键盘上的 **Win + Ctrl + <** 或 **Win + Ctrl + >** 在虚拟桌面间切换; 键盘上的

Win + Ctrl + D 打开新桌面; 键盘上的 **Win + Ctrl + F4** 关闭一个虚拟桌面。

窗口贴靠

图 2-10 Windows 10 可以做四个窗口的分屏

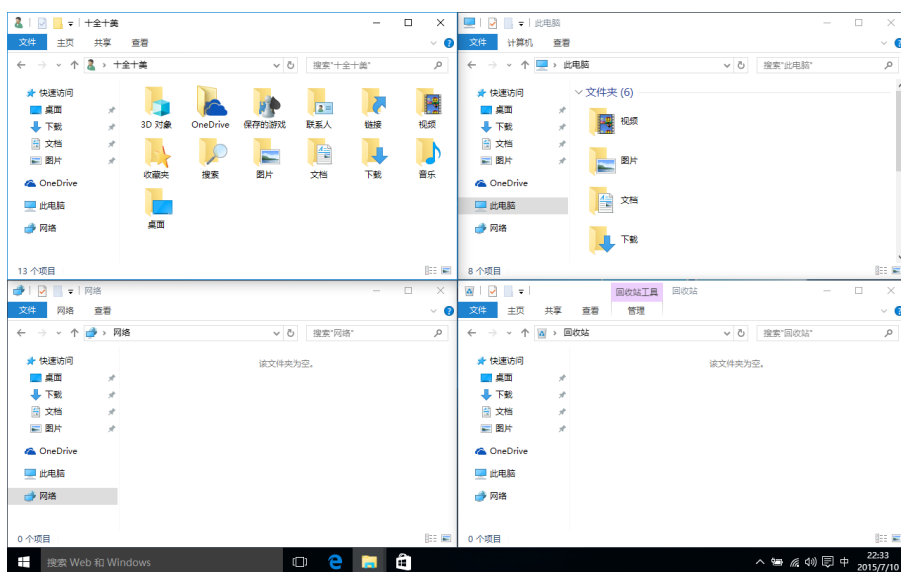


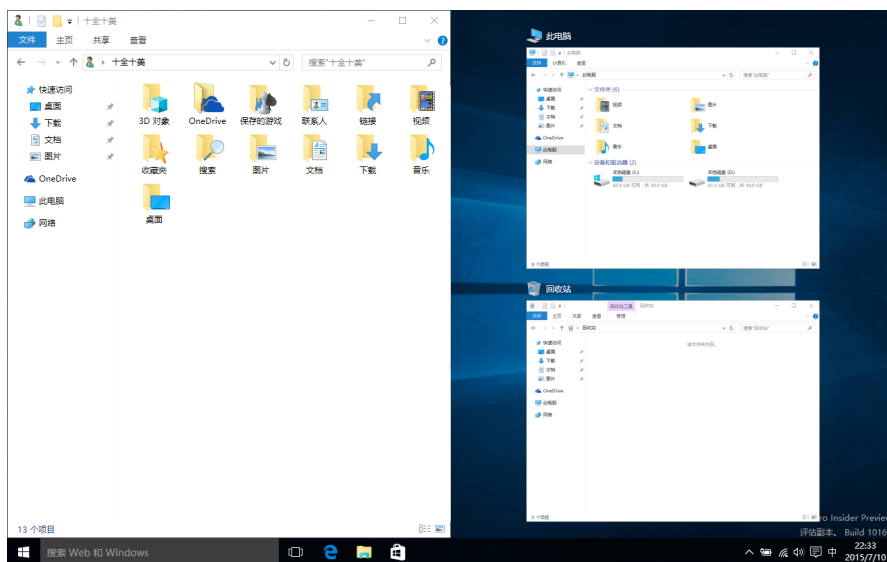
图 2-11 新加入的 Snap Assist 功能

早在 Windows 7 时, 微软就加入了 Aero Snap (窗口贴靠) 功能, 便于同屏操作几个窗口。而 Windows 10 进一步改进了这个功能。

你可以拖动窗口到屏幕的四个角落, 使窗口占据屏幕 1/4 的位置(图 2-10)。

同时, 微软引进了一个叫做“Snap Assist”的新功能, 当你拥有两个或以上窗口时, 且某一窗口已向左或右占据半边屏幕贴靠后, 系统会自动询问你是否需要选择另一窗口来填满屏幕空间, 当你执行操作后, 系统会自动贴靠你所选定的窗口(图 2-11)。

窗口贴靠功能在 Modern 应用中仍然适用, 操作方法和桌面窗口是一样的。



第三节

细微之处的改进

有时候，你察觉不到，但这也是 Windows 10 的改变。

图 2-12 快速访问页面

快速访问(图 2-12)用来显示已固定的文件夹,以及系统记录的最近使用文件,同时取代左侧窗格的“收藏夹”(实际上它们功能是相近的)。请不要同 Ribbon 栏左上方那一排按钮(快速访问工具栏)相混淆。快速访问也是一种特殊的跳转列表。

点击任务栏上的资源管理器图

标,默认就是启动快速访问(可以从文件夹属性中修改)。

快速访问被划分为两个部分,上方是“常用文件夹”列表,这里会显示用户最常访问的若干个文件夹。同时它还保留了之前“收藏夹”特性,允许用户手动添加文件夹。值得一提的是,它不仅支持本地设备上的文件夹,还能支持 OneDrive 上的文件夹。在“常用文件夹”下方是“最近使用的文件”列表,这里会记录你最近访问过的文件名和文件所在位置,按照最近访问时间的先后排序排列。

如果欲将某个文件夹固定到快速访问,你可以选中文件夹,然后点击主页——固定到快速访问即可(或右键后点击“固定到快速访问”),反向操作(包括最近使用的文件)则可以右键点击这些文件/文件夹,然后选择“从‘快速访问’中删除/取消固定”即可。

快速访问有泄露个人隐私的风险,因此要养成定期清理最近使用文件列表的习惯,这个操作也是在文件夹属性中执行,点击“隐私”中的“清除”即可。当然,如果你不喜欢系统在快速访问中显示历史,就可以把“在快速访问中显示最近使用的文件”和“在快速访问中显示常用文件夹”这两个复选框给取消。

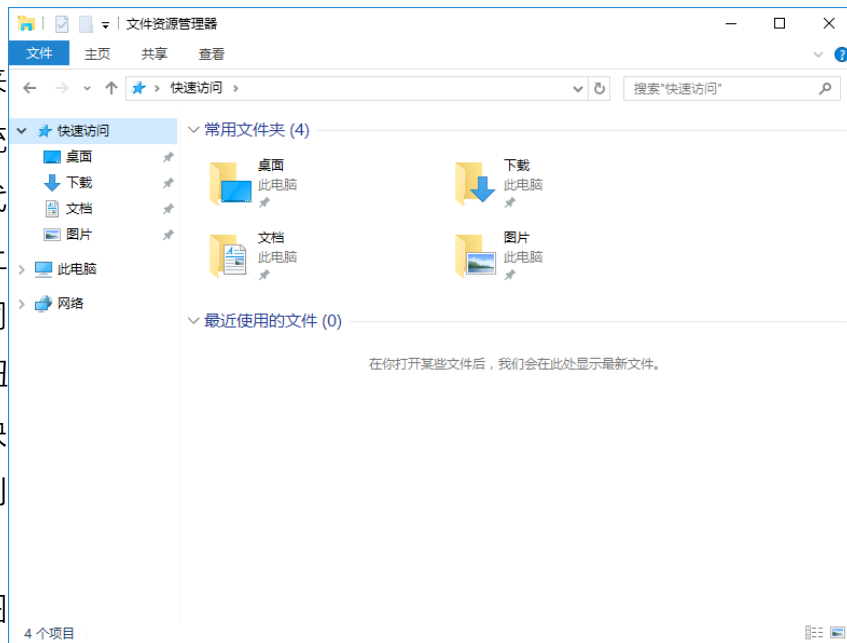


图 2-14 命令提示符内文字支持换行显示，窗口主体支持半透明。尽管如今 DOS 已经几近过时，但命令行的操作方式一直保留了下来。在 Windows 10 中，命令提示符(图 2-13)也做出了改进。以前，当你在命令提示符使用一些快捷键的时候，通常会输出 ^V 等错误的结果，而现在，你可以从容的用快捷键进行文字操作。这些快捷键是老生常谈的了：

Ctrl+C	复制
Ctrl+V	粘贴
Ctrl+A	全选
Ctrl+F	查找

图 2-14 命令提示符内文字支持换行显示，窗口主体支持半透明

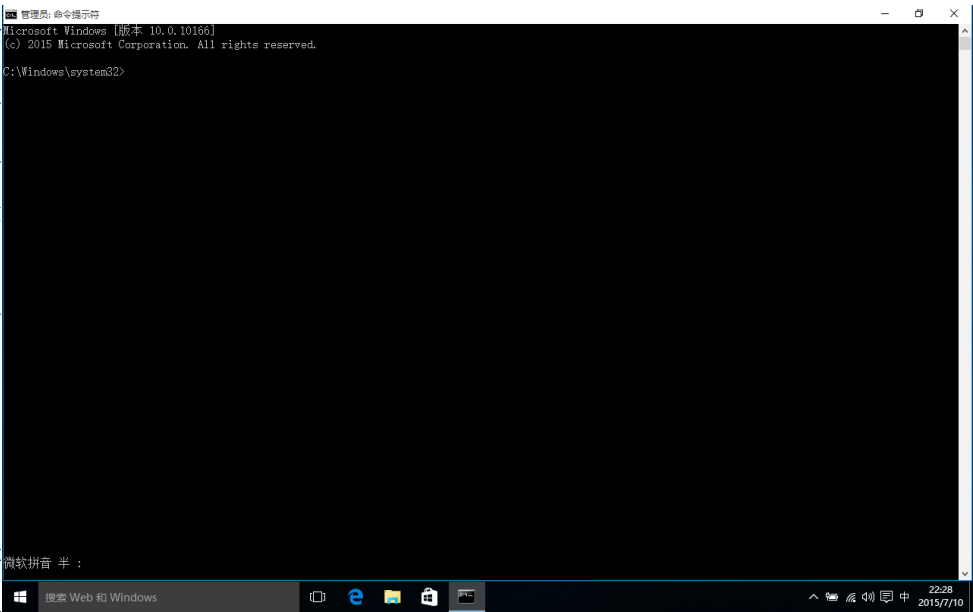
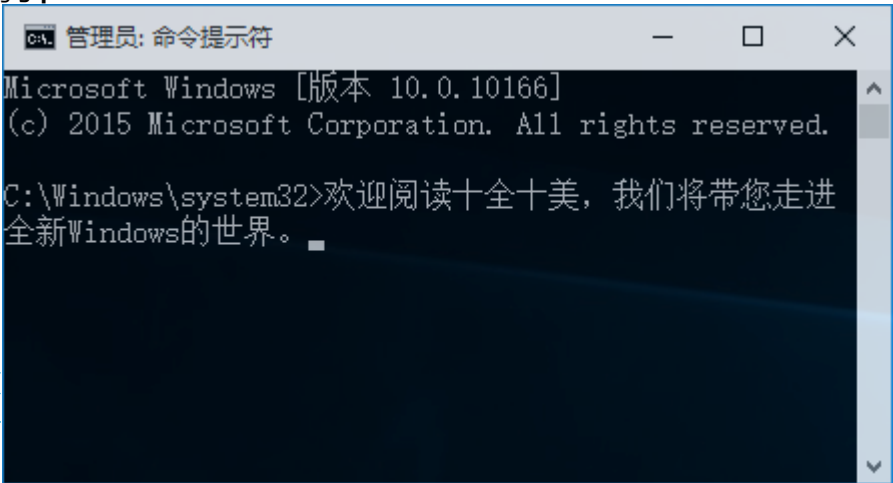


图 2-13 命令提示符最大化状态下



另外，命令提示符的外观也做出了重大的调整。之前系统的命令提示符，点击最大化按钮后，其窗口并不能占据整个屏幕，而在 Windows 10 中，这个“错误”被修正过来了。同时，命令提示符内文字支持换行显示，窗口主体支持半透明(图 2-14)，解决了命令太长而无法完整显示命令字符的问题，也提高了命令提示符外观的美观度。窗口透明度可以在命令提示符的属性——颜色中拖动滑块调整。

如果更喜欢经典样式的命令提示符，你就可以在属性中给“使用旧版控制台(需要重新启动)”打上勾。

* 此外，Windows 10 还增加了注册表编辑器同类项跳转、原生支持文档转化为 PDF、原生支持视频录制、Windows Media Center 彻底移除等改变。限于篇幅，不再叙述。

第四节

DirectX 12

对于游戏玩家来说, DirectX 12(图 2-15)可能是 Windows 10 最具闪光点的更新了。然而这次更新并没有 DirectX10 那么具有革命性,但它一经推出,就受到了游戏玩家和开发者的强烈关注。DirectX12 同 AMD Mantle API 相类似,但仅支持 Windows 10。



DirectX12 一个最重要的特性就是支持双显卡同时渲染,将可拆分的、相邻的工作负载分配到不同的 GPU,显卡间都能协作执行。这将大大提高显卡的性能和游戏的画质。

另外,DirectX12 大大提高了多线程效率,可以充分发挥多线程硬件的潜力。DirectX11 在这方面受 CPU 性能的严重制约,主要是因为不能有效利用多核心。微软宣称,微软对多核心 CPU 的利用几乎是完美线性增长的,也就是说四核心能接近单核心的四倍。DirectX12 还有渲染流水线、渲染特性、资源管理、命令列表与绑定、描述符跳跃等等方面的改进。

不过,至今完全支持 DirectX12 的游戏还比较少,开发商也要顾及 Windows 10 以下系统的兼容性。所以,DirectX12 普及,尚需时日。

需要说明的是,只要显卡厂家有针对 Windows 10 的更新驱动,就应当能支持 DirectX12,而不受上面列表之限。Feature Level 12_1 代表支持 DirectX12 全部特效,Feature Level 12_0 则是部分特效。同时,DirectX12 Feature Level 11_1、11_0 说明它们对应支持的特效更少,但也兼容 DirectX12,能够享受 DirectX12 底层加速。因此,等级越高,代表显卡支持程度越高,相应的价格也越昂贵。

支持的硬件列表

以下列举一些支持的硬件列表（图 2-16），你可以通过这个列表确定显卡是否在受支持范围内。

兼容支持^①

NVIDIA

Geforce 桌面显卡系列 (420 及以上)
Geforce 移动显卡系列 (410M 及以上)

AMD

Radeon R9/R7 系列显卡
Radeon R5 240 显卡
OEM 市场的 Radeon HD 8000 系列 (8570 及以上型号)
笔记本市场的 Radeon HD 8000M 系列
Radeon HD 7000 系列 (7730 及以上型号)
笔记本市场的 HD 7000M (7730M 及以上型号)

INTEL

Core i7/i5/i3 (4xxx 及以上)

基础支持^②

NVIDIA

GTX750/GTX750Ti
GeForce 830M, GeForce 840M, GeForce GTX 850M, GeForce GTX 860M
GeForce 930M, GeForce 940M, GeForce GTX 950M, GeForce GTX 960M

AMD

Radeon HD 7790、Radeon R7 260(X)
Radeon R9 285、Radeon R9 290(X), Radeon R9 295X2、Radeon R9 360(X)、Radeon R9 370(X)、Radeon R9 380(X)
AMD A4/A6/A8/A10 系列 (Kaveri 架构)
AMD A6/A8/A10 Pro-7000 系列 (Kaveri 架构)
AMD E1/A4/A10 Micro-6000 系列 (Mullins 架构)
AMD E1/E2/A4/A6/A8-6000 系列 (Beema 架构)

完整支持^③

NVIDIA

桌面平台:
GTX Titan X、GTX 980 Ti、GTX 980、GTX 970、GTX 960
移动:
GTX 980M、GTX 970M、GTX 965M

AMD

Radeon R9 390(X)、Radeon R9 395X2

①: 指 Feature Level 11_0 和 11_1

②: 指 Feature Level 12_0

③: 指 Feature Level 12_1

第五节

Dism 和 Compact

把磁盘空间节约到极致。

Dism 和 Compact

同 DirectX 一样, Dism 在 Windows 10 中也得到改变。Compact (直译为压缩) 是 Windows 10 开始出现的新功能, 和 WIMBoot 一样都是通过 WOF.sys 这个 Filter 驱动实现的。但是 Compact 和 WIMBoot 最大的不同是: Compact 是一对一的 (一个文件压缩成一个文件存储到磁盘, 类似 NTFS 压缩), 而 WIMBoot 是 n 对一的 (一堆文件压缩成一个 WIM 存储到磁盘)。

Compact 功能首先出现在 Windows 10 Build 9879 (当时这个功能叫做 System Compression, 即系统压缩) System Compression 的原理是把 Windows 目录和 Program Files 目录下的所有的 EXE 和 DLL 文件采用 WofAlgorithm 压缩 (Wof 算法), 也就是对每个文件进行 WIMBoot 压缩。

Compact 可以用 *XPRESS4K* (最快速) (默认)、*XPRESS8K*、*XPRESS16K*、*LZX* (压缩程度最高) 四种听起来很复杂的算法对文件进行压缩。我们列举的四种压缩率是从低到高排列的。

如果你不会用到 Windows10 之前的版本 (包括 PE), 那么 Compact 不会碰到任何兼容性问题; 压缩率高; 无需清理 WinSXS 冗余文件 (在 OSCompact 模式下的 Windows 10; 安装完系统补丁自动会对 WinSXS 进行清理); 今后系统可以使用 ISO 直接升级, 不会像 WIMBoot 一样报错而无法升级。

当你使用的是不带 WOFADK 驱动的旧版本操作系统或者 PE 的时候; 那你可就碰上大麻烦了。被 Compact 过的文件统统无法访问; 如果你用的是 Windows10 之前的系统进行 OSCompact 或者对 Windows 10 的系统分区进行非默认算法 (即 *XPRESS4K*) 的 OSCompact; 是需要对一些文件进行排除的; 否则你休想进入系统; 也许你需要定期执行 Compact。因为由于你应用的更新, 很多被 Compact 的文件不处于 Compact 状态。

Compact 基本上不会影响性能, 这和 WIMboot 不同, WIMboot 对性能有一定影响。但 Compact 容易产生难整理型文件碎片, 在机械硬盘上应注意。

操作：如何使用 Compact^①

同时按下按键盘上的 Win 和 X 键, 出现菜单; 点击命令提示符 (管理员); 接下来弹出用户账户控制对话框, 点是就出现了命令提示符窗口。

若安装时全盘压缩, 可以执行这样的指令:

```
DISM /Apply-Image /ImageFile:install.wim /Index:1 /ApplyDir:D:\ /compact
```

另外介绍几个不是很常用的指令:

如果你要查询你的系统有没有被 OSCompact, 可以在管理员权限的命令提示符下输入以下命令并回车 (在线系统适用)

```
compact /compactos:query
```

如果你的系统被 OSCompact 的话, 会显示以下信息“系统处于压缩状态。它将保持此状态, 除非管理员更改它。”

如果你想要进行 OSCompact 的话, 可以在管理员权限的命令提示符下输入以下命令并回车 (在线系统适用)

```
compact /compactos:always
```

如果你要撤销 OSCompact, 可以在管理员权限的命令提示符下输入以下命令并回车 (在线系统适用)

```
compact /compactos:never
```

如果你想单独对一个文件进行 Compact 压缩, 那怎么办呢? 举个例子, 如果我要对 E:\ToolKit\procexp.exe 进行 lzx 算法的 Compact 压缩, 我可以输入以下命令并回车 (在线和离线都可以)。

```
compact /c /exe:lzx E:\ToolKit\procexp.exe
```

既然介绍了压缩, 那么介绍如何解压也是不可少的。

```
compact /u /exe E:\ToolKit\procexp.exe
```

如果你的系统不符合要求, 或者不习惯命令提示符操作, 推荐使用 Dism++。使用方法参考 Dism++ 的相关说明, 这里不再叙述。

^①: 使用命令提示符, 这个用法只适用于 Windows 10 Build 10074 或更高版本, 或者旧版本系统有集成 Compact 的 Wof.sys。



Windows Hello

Windows Hello 也是 Windows 10 出现的新技术。从名字上就可以看出这个技术与系统登录相关。实际上, Windows Hello 是一种生物特征授权方式。用户只需要露一下脸, 动动手指, 就能立刻被运行 Windows10 的新设备所识别。Windows Hello 不仅比输入密码更加方便, 也更加安全。同时, 它能够让你无需存储在本地或网络服务器的密码即可进行认证应用、企业内容甚至包括某些在线体验。

从实现原理上说, 诸如摄像头和指纹识别器之类的识别设备早就已经普及了。不过, Windows Hello 需要特定的硬件才能使用。诸如支持 Windows Hello 的摄像头就必须要有英特尔的 Realsense 认证。

为什么没有 Windows 9 ?

Windows 10 发布后, 我们都不禁要问: 为什么没有 Windows 9 ? 有以下几种说法供娱乐

1. 微软官方认为, Windows 10 开辟了大一统的新纪元。
2. 乔北峰衣服上的二进制码告诉我们: seven eight nine (eight 谐音 ate), 中文翻译为七八(把)九十(食)。
3. 有人认为 Windows 8.1 就是 Windows 9, 微软因而特意跳过。
4. 有 IT 专家指出, Windows 10 如果是 Windows 9, 会与早期的 Windows 9x 系列混淆, 出现兼容问题。
5. 根据历史典故, 温酒 (Windows 9) 斩华雄去了。

Windows 10 发行版本间的区别 (一)

Windows 10 根据用途不同, 分为七个发行版本 (SKU), 我们这里介绍其中的三个版本。

1. 家庭版: 适合一般消费者使用, 拥有最基础的 Windows 功能。这个版本可以由 Windows 7 家庭普通 / 高级版或 Windows 8/8.1 Core/China 升级而来。Windows 10 中家庭版的特性是 Windows Update (下称 WU) 推送补丁完全受微软控制, 无法取消, 也无法推迟或决定补丁。

2. 专业版: 适合电脑爱好者使用, 除了家庭版的功能, 还有组策略, Bitlocker 等特性。这个版本可以由 Windows 7 专业 / 旗舰版或 Windows 8/8.1 Pro 升级而来。Windows 10 中专业版的特性是可以推迟 WU 推送补丁的时间 (最多八个月)。

3. 企业版: 适合企业用户使用, 除了专业版的功能, 还有企业级管理、Applocker 等特性。这个版本需要重新购买授权, 不能直接从旧版本的企业版推送升级。Windows 10 中企业版的特性是 Windows Update 不仅可以决定 WU 推送补丁的时间, 还能决定所需要的补丁, 甚至可以只选择安全性更新。



应用体验的改进

全新的电脑设置

Cortana

Microsoft Edge

无处不在的应用

第一节

应用体验的改进

更加易用的应用世界已呈现在我们眼前。

Modern 应用的桌面端改进

微软在 Windows 8 的时候引入了应用这个概念, 也引入了移动端的操作理念。但是这种理念与桌面端相违背, 割裂了操作体验; 全屏化的运行模式有时也给操作带来不便。在 Windows 10 中, 微软尊重了桌面端的操作习惯, 加入了 Modern 应用的窗口化功能。



图 3-2 返回按钮



图 3-1 一款应用运行在窗口模式下

在窗口模式下运行的 Modern 应用(图 3-1), 右上方同样具有最小化, 最大化(还原窗口)和关闭按钮, 操作体验尽可能趋近于桌面端程序。根据前两章可知, 窗口化下的应用也可以自适应内容显示以及做窗口贴靠。

如果应用具有系统级后退功能, 则会在左上角显示一个箭头, 你可以按这个按钮返回到上一页(图 3-2)。

但是现阶段, 应用商店里能下载到的应用基本上按照 Windows 8 的 SDK 开发, Windows 10 需要向下兼容这些应用。于是我们能在这些应用中找到额外的这两个按钮。

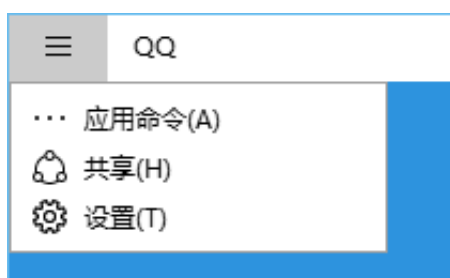


图 3-3 兼容式汉堡菜单

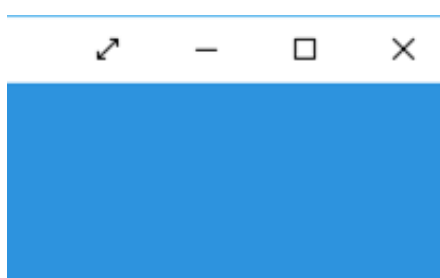


图 3-4 全屏按钮

左上角的按钮是汉堡菜单(图 3-3)。请不要同第一章中的“汉堡菜单”相混淆。实际上, 这是为了向下兼容 Windows 8 的 Charm 栏而设计的。按下汉堡菜单后, 你可以像之前呼出 Charm 栏一样来对这个应用进行共享和设置。应用命令相当于之前在应用内按鼠标右键。右上角多出来的按钮是全屏按钮(图 3-4), 这个按钮允许 Windows 8 应用在全屏模式下运行。对于 Windows 10 应用, 由开发者自行决定是否显示全屏按钮。如果没有显示全屏按钮而想使应用全屏, 请使用平板模式(可参看下面的内容)。

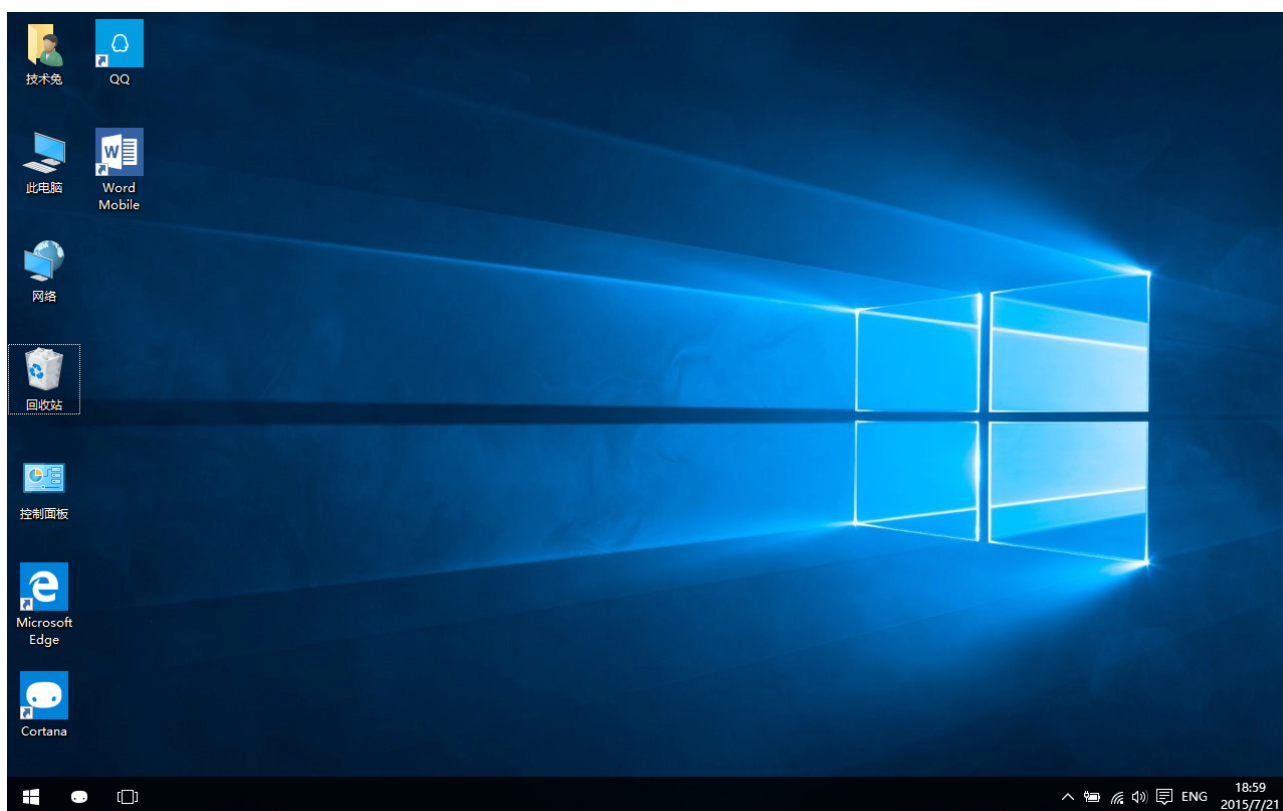


图 3-5 Modern 应用固定到桌面

除此之外, 还有这些地方体现出应用的桌面端改进:

应用选取文件的对话框改变成同桌面程序一样。

某些应用支持多开窗口, 例如计算器和 Edge。

应用的快捷方式可以直接固定到桌面(方法是打开开始菜单的所有应用列表, 直接拖动欲固定应用到桌面, 图 3-5)。

平板模式

图 3-6 平板模式下的任务栏，左起第二个图标是返回按钮

微软在 Windows 8 的基础上对平板操作做出了改良，在 Windows 10 中引入了平板模式（也叫做 Continuum 模式，即互通），提供了二合一的触屏体验。混合形态设备可以通过插拔键鼠来很容易地切换到平板模式，若要手动打开，你可以点击通知中心按钮或屏幕从右边缘轻扫（触屏设备），然后打开“平板模式”开关。

平板模式中默认启用了全屏开始，且传统桌面和显示桌面按钮被屏蔽，这个时候，你只能固定应用以及快捷方式到全屏开始。

平板模式中任务栏得到了简化（图 3-6），默认情况下只显示开始，返回，搜索和任务视图这四个按钮，以及右侧的系统通知图标和时间。任务栏中首次加入后退按钮，支持系统级后退的应用（后退按钮显示在标题栏的），可以通过该后退按钮来进行后退操作，不支持系统级后退的应用，仍然使用应用中的后退按钮，点击任务栏上的后退按钮将回到全屏开始。切换窗口要通过任务视图进行，此时虚拟桌面功能将被禁用。这些按钮都放大了间距，以适合手指触摸。当然，你也可以单击鼠标右键，然后自定义需要显示的项目。

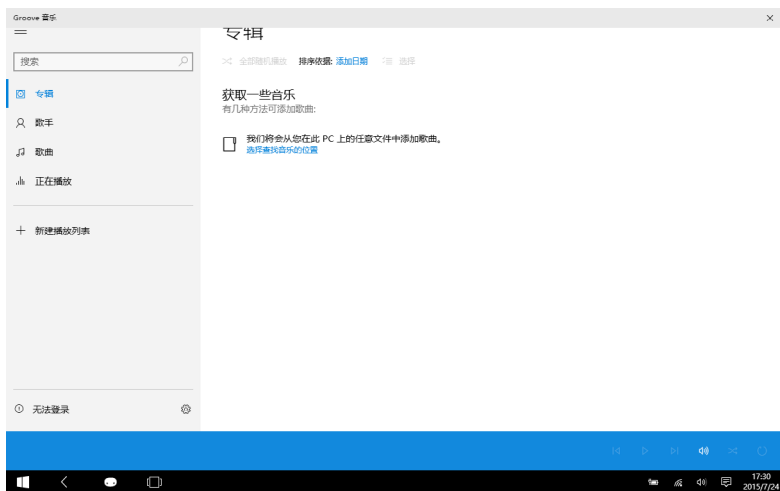


图 3-7 平板模式下的 Modern 应用

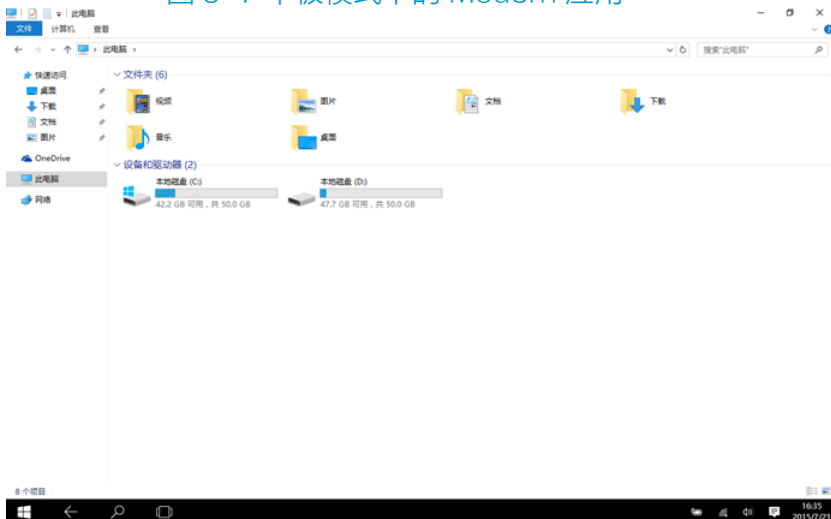


图 3-8 平板模式下的桌面窗口

平板模式中的窗口, 桌面程序与 Metro 应用同等看待, 都将以全屏模式运行, 类似于 Windows 8.1 (图 3-7 和图 3-8)。Modern 应用若要呼出标题栏, 在 Modern 应用中鼠标移到上方或者手指向下滑动。这时 Modern 应用只能执行关闭操作。桌面程序窗口锁定为最大化 (任务栏保留在下方)。这时窗口只能最小化或关闭。锁定大小或者是置顶的窗口不受影响, 依然可以自由移动 (如 QQ)。部分桌面程序会出现强行拉伸的错误, 也就是本来不能调整大小的窗口被强行拉伸到全屏。

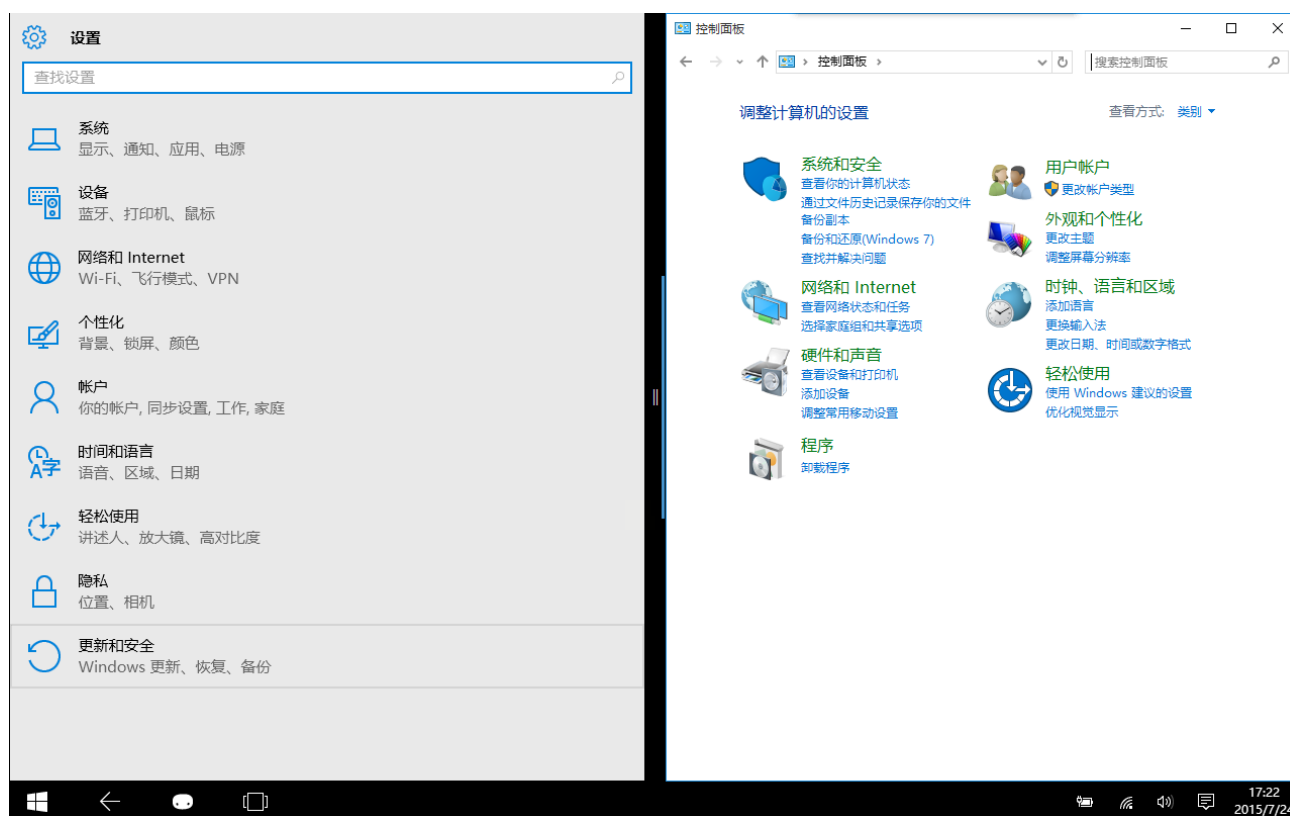


图 3-9 平板模式下的分屏

Windows 10 的平板模式下也支持手势拖拽 (包括桌面程序), 用以分屏或关闭进程 (方法是从上边缘一直拖到底) (图 3-9)。

相对于 Windows 8 的改变在于:

应用分屏时也提供 “Snap Assist” 分屏建议, 你可以决定用于分屏的其他应用。

分屏的快捷键 (WIN+ 句号切换焦点, WIN+ 左 / 右进行分屏) 被去除。

应用分屏的尺寸限制变得相对宽松了, 不再限定指定屏占比 (若要改变应用尺寸, 可拖动应用间的长条)。

应用分割条间, 当前应用 (焦点) 用分隔条上的标记标注, 便于区分。

第二节

全新的电脑设置

新的系统中枢。

说到设置，我们第一个想到的就是控制面板。这个从 Windows 3.0 时代就存在的系统工具，曾经是管理系统所有设置的中枢。从 Windows 8 开始，微软另外开发了 Modern 电脑设置图 3-10)。不过很长一段时期以来，电脑设置并没有引起重视。在 Windows 10 中，

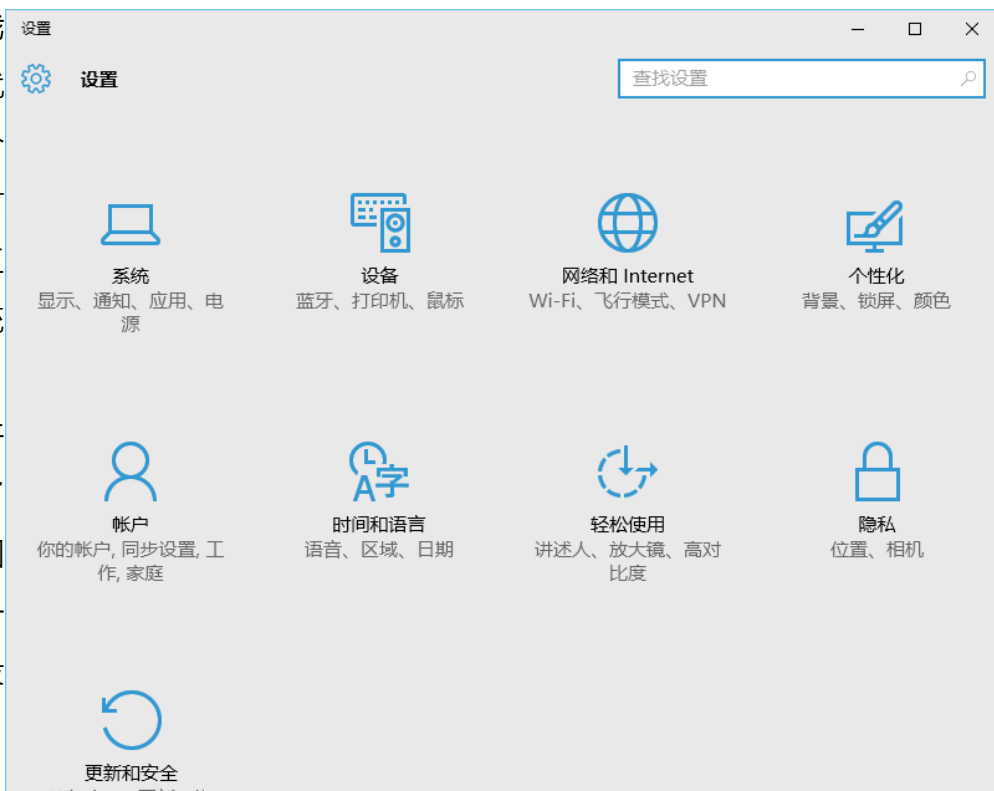


图 3-10 电脑设置窗口

电脑设置得到了进一

步完善，同时也开始删改控制面板的设置项目。例如，“Windows Update”现在在控制面板就已经看不到了，“个性化”也只剩下主题设置。控制面板目前仍然保留，但合二为一为大势所趋。打开电脑设置可以从开始菜单中的“设置”连接打开。

下面我们通过问答简要介绍各设置项的作用。时刻注意，你可以通过搜索框来搜索具体的设置项。部分设置项仅限笔记本或平板可用。

系统和设备（相关设置：系统、设备）

Q: 我该如何修改屏幕分辨率或其他屏幕设置？

A: 点击“显示”或桌面右键点击“显示设置”，可以修改包括缩放、屏幕方向、亮度等的设置。

Q: 感觉通知太多太乱了，我该如何管理通知？

A: 点击“通知和操作”，设定全局通知设置或开关通知项目；也可以设定操作中心折叠状态时显示的快捷按钮。

Q: 最近想删除一些应用, 我该如何做?

A: 点击“应用和功能”, 可以查看当前安装的所有程序和应用并选择卸载; 也可以点击“管理可选功能”用以管理有关本地化的条目。

Q: 我要更好的养护我的电池, 同时也不想自动待机, 有什么电源设置能帮上忙?

A: 点击“节电模式”, 当电池接近耗尽时, 打开以提高电池续航, 在节电模式设置中还能设置启用的电源百分比, 是否允许推送通知, 以及调低亮度; 或者点击“电源和睡眠”, 调整屏幕关闭和执行睡眠的时间。

Q: 不用打开文件资源管理器, 我怎么知道磁盘的大致占用?

A: 点击“存储”, 可以查看所有硬盘的内容占比; 也可以更改指定目录的保存位置。

Q: 我使用第三方浏览器, 设置为默认时经常会弹出恼人的提示, 怎么解决?

A: 点击“默认应用”, 可以按给定的条目设置默认应用, 也可以按文件类型或协议设置默认应用。不过截止目前, Firefox 最新版已经支持直接跳转到“默认应用”。

Q: 我该如何查看电脑属性?

A: 除了传统的对“此电脑”右键点击属性, 在设置中点击“关于”也可以查看。

Q: 我新买了一台打印机和摄像头, 我该如何添加到系统?

A: 正常情况下系统会自动安装驱动并添加。如果看不到, 点击“打印机和扫描仪”或“已连接设备”, 然后点击添加。这里也可以设置是否在按流量计费的连接中下载更新驱动。

Q: 我最近添加了一些蓝牙设备, 我如何找到它们?

A: 点击“蓝牙”, 在这里可以管理搜索到的蓝牙设备。

Q: 每次插入 U 盘都会自动播放, 我担心安全问题, 怎么禁用?

A: 点击“自动播放”。在这里可以设置是否对所有媒体设备使用自动播放, 以及修改针对特定设备的自动播放设置。

网络和个性化 (相关设置: 网络和 Internet、个性化)

Q: 我该如何连接到网络?

A: 点击任务栏上的网络图标, 或者点击“WLAN”、“拨号”、“以太网”以连接或配置网络。

Q: 桌面右键点击“个性化”, 怎么没有出现熟悉的窗口?

A: 实际上在设置的“个性化”中都保留了相应的功能。点击“背景”可以修改壁纸以及契合度。点击“颜色”可以修改窗口边框颜色和任务栏指示器颜色, 或开关“从我的背景自动选取一种主题色”、开始菜单和任务栏的透明选项和主题色。点击“锁屏壁纸”可以修改锁屏壁纸, 以及在锁屏上通知的应用。点击“主题”可以跳转到主题、声音主题、桌面图标和鼠标指针的相关设置。

Q: 开始菜单左侧列表和跳转列表看起来好杂乱, 我该如何关闭它们?

A: 点击“开始菜单”可以修改这些设置。另外针对习惯开始屏幕的用户, 也可以选择全屏开始。用户也可以像 Windows 7 一样, 自定义在开始菜单显示的链接。

账户和安全 (相关设置: 账户、隐私、更新和安全)

Q: 听说 Windows 10 引入了一个新的账户机制, 我该如何启用它?

A: 打开“你的账户”, 点击“使用 Microsoft 账户登录”, 按要求输入账号密码即可。这里也可以设置用户头像。

Q: 前文提到过 Windows Hello, 我也想体验, 怎么打开?

A: 打开“登录选项”, 在 Windows Hello 下面的按钮中选择一个设置即可。不过, 这需要硬件支持。“登录选项”还能选择离开电脑后是否需要重新登录, 或者调整有关登录密码的设置。

Q: 我担心系统后台自动传输个人信息, 或者应用滥用某些访问权限, 我该如何做?

A: 点击“隐私”设置大项, 这里可以修改有关隐私的常规选项, 给予应用某些访问权限, 以及设置可以在后台静默运行的应用。

Q: 我在控制面板没有发现 Windows Update 了, 怎么打补丁?

A: 点击“Windows 更新”。这里可以更新补丁以及推送新版本系统(需加入 Windows Insider 计划)。“高级选项”中可以设置获取更新的方式, 选择打开 Microsoft Update, 或者查看更新记录。

Q: 用不惯 Windows 10 了, 想回滚以前的系统, 怎么做?

A: 点击“恢复”, 可以执行回滚操作(条件: 升级 Windows 10 后的一个月内, 且保留了 Windows.old 文件夹), 在这里也可以重置电脑或进入高级启动选项。

Q: 我该如何查看激活情况? 当系统提示我不是正版, 我该如何做?

A: 点击“激活”可以查看激活情况。若系统没有激活, 会提示你重新输入密钥, 或者要求进入应用商店购买。



第三节 Cortana

人工智能的新时代来临了。

Cortana 概述

Cortana, 也叫科塔娜或小娜, 下称小娜(图 3-11)——是 Windows 10 的招牌功能, 于 Windows Phone 8.1 GDR1 首次发布。Build 9901 开始首次在 Windows 10 出现。小娜是微软发布的全球第一款个人智能私人语音助理她能够“了解”用户的喜好和习惯, 用户可以对她说话, 或者让她做一些事情, 小娜可以说是微软在机器学习和人工智能领域方面的尝试。

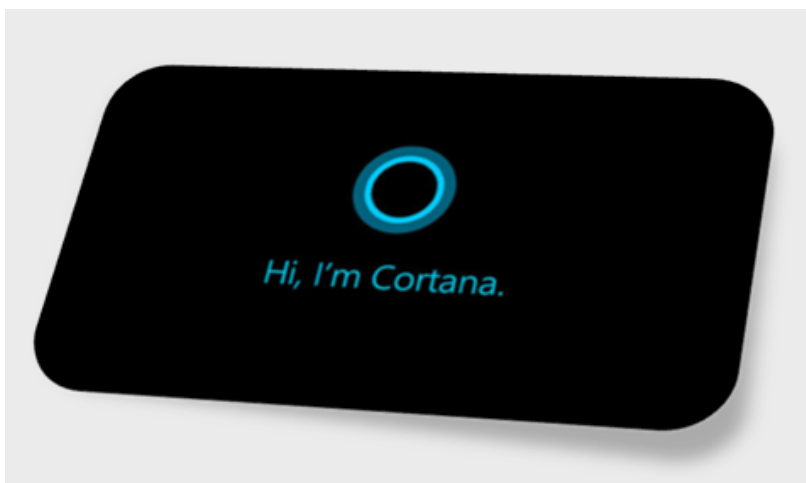


图 3-11 Cortana

小娜的语音功能非常强大, 当你说出“你好小娜”时, 她就会作出回应, 开始聆听你说话, 然后, 你便可以问她一些问题, 问她天气, 让她提醒你在某时间做某事, 让她讲个笑话, 讲个鬼故事或背首诗, 还可以让她唱首歌, 安排行程, 打开应用等等。

小娜支持多种语言, 包括简体中文, 英语, 法语, 德语, 日语, 俄语, 西班牙语、意大利语等。

中国版的小娜有两种形象供用户选择——小娜圆圈与面团(图 3-12), 当选择不同形象时, 所发出的特效音也会有所改变。

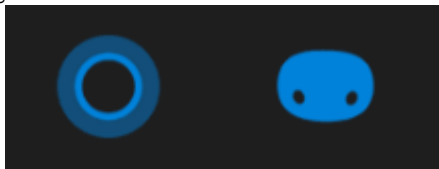


图 3-12 小娜的两种形象

小娜、小冰与搜索的使用

要使用小娜非常简单, 任务栏默认会放置搜索框或搜索按钮, 点击即可开始使用(图 3-13)。(若没有显示, 可以在任务栏右键菜单自行勾选显示, 或使用之前版本的搜索快捷键使用)。

要使用小娜非常简单, 任务栏默认会放置搜索框或搜索按钮, 点击即可开始使用(图 3-13)。(若没有显示, 可以在任务栏右键菜单自行勾选显示, 或使用之前版本的搜索快捷键使用)。

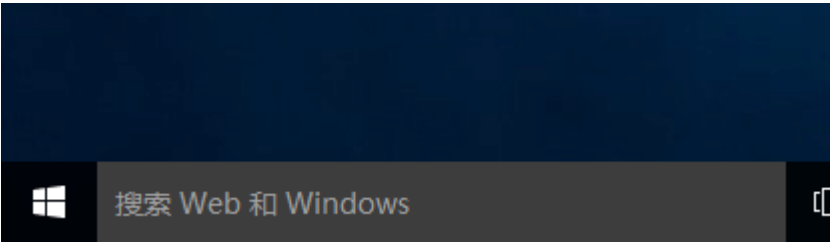


图 3-13 搜索框



图 3-14 小娜初始化过程

用户第一次启用小娜, 系统会引导用户进行初始化, 其中包括提示“我需要你提供一些信息”和设置小娜对你的称呼(图 3-14), 如果没有打开定位, 系统会要求打开。如果没有使用微软账户, 系统会要求你切换到微软账户才可使用小娜。



图 3-15 小娜信息页

配置完毕后, 小娜会展现一些信息, 例如当地天气、当日新闻以及股票涨跌情况, 上下拖动可以查看更多信息(图 3-15)。与小娜对话可以单击搜索框右侧的麦克风按钮, 或按键盘上的 **Windows + C**, 或者直接说”你好小娜”。汉堡图标下第二个图标是笔记本, 这里可以查看小娜记录的你的所有信息, 以及设置有关小娜的选项; 第三个图标是提醒, 这里可以查看你的待办事项。

这里介绍一些小娜的调戏办法:

功能	你可以跟她说
聊天	给我讲个笑话、成语接龙、给我唱首歌、模仿赵本山、模仿周星驰
电话	给 XXX 打电话
发短信	给 XXX 发短信、向 XXX 发送消息
日历	将我下午 3 点的事件更改到 4 点、今天下午 6 点有什么事?、我下一步该做什么?
提醒	提醒我明天去……、当 XXX 打电话时, 提醒我……
约会	新建一个 X 点的约会、新建约会“去和 XXX 做……”
闹钟	早上 6 点叫醒我、关闭下午 3 点的闹钟
备注	写便笺、注意: 将我的车留在第 4 层上
播放音乐	播放 [艺术家]、开始播放 [歌曲]、播放一些流行音乐、播放 [专辑]
交通	向我提供到 XXX 的路线、我在哪里?、为我显示 XXX 大街的地图、我离 XXX 有多远?、去 XXX 的路上的交通情况如何
要去的地点	我附近有星巴克吗?、告诉我附近评价高的餐馆
询问天气	这个周末会下雨吗?、外面冷吗?、今天天气如何?、下周的天气预报是怎么说的?、XXX 现在很热吗?、XXX 的天气
事实	世界上最高的女人是谁?、葡萄牙总统是谁?、卡塔尔的首都是什么?
查询航班	中国东方航空公司航班 737 的状态如何?、XXX 航班能在 XXX 准时到达吗?
应用	打开“相机”、推荐应用
新闻	头条资讯、国内资讯、科技资讯、商业资讯、体育资讯、娱乐资讯
电影	热映电影排行榜、XX 电影排行榜
电视剧	盗墓笔记、古剑奇谭
彩票	双色球、大乐透
名人	王祖蓝、范冰冰微博

小娜还有一个隐藏功能, 可以呼出小冰并与其聊天, 在搜索框中键入或直接说“召唤小冰”即可(图 3-16)。小冰是微软亚洲研究工程院研发的一款聊天机器人, 单独面向中国用户, 于 2014 年推出, 现已进化到小冰二代, 功能更为全面。目前小冰只接受文字交谈。如果不选择启用小娜, 你也可以使用搜索框, Windows 10 仍然支持搜索桌面程序和 Modern 应用并运行, 也可以搜索网络内容。

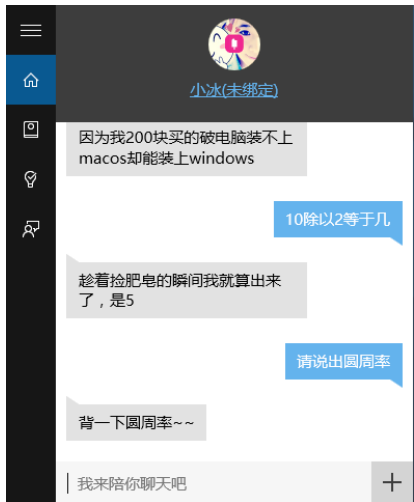


图 3-16 小冰聊天页面

第四节

Microsoft Edge

抛弃负担，轻装上阵。

Edge 概述

Microsoft Edge (下称 Edge, 图 3-17) 是微软重新开发的一款新浏览器(实际上, Edge 基于 IE11 的内核改造而来)。在 Edge 出现之前, Windows 预装默认的浏览器一直是 Internet Explorer (N 版除外), 但其功能甚少, 性能体验欠佳, 一直被多数用户戏称为“用来下载其他浏览器的浏览器”。因此, 微软决定另起炉灶, 以 Modern 应用的形式, 创造出了 Edge。当然,

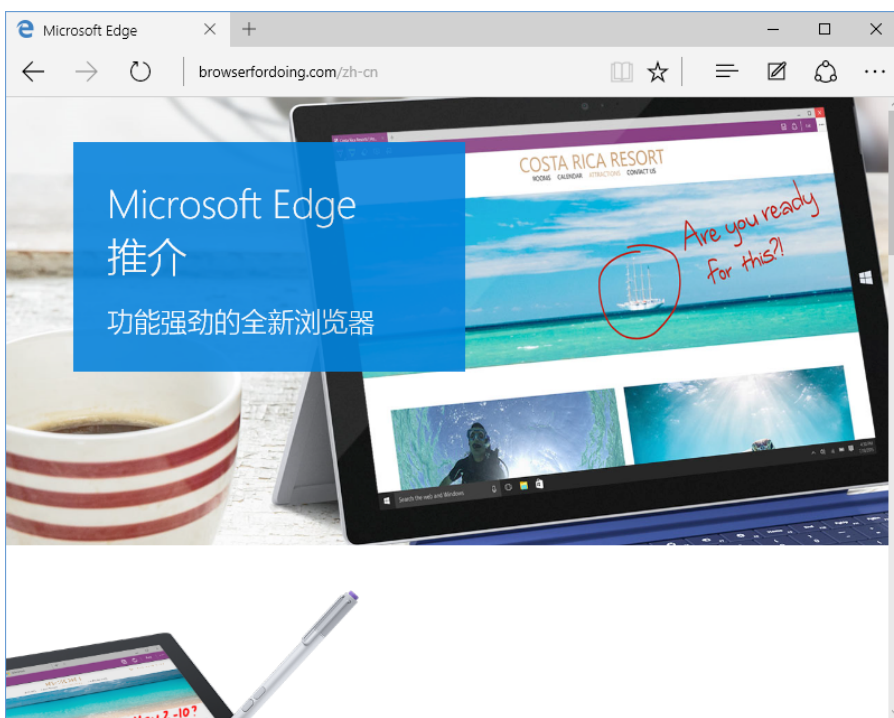



图 3-17 Edge 浏览器界面

Internet Explorer 也会保留, 供使用了较老技术和代码的网页使用。Edge 目前尚未有官方中文译名。

Edge 在早期开发阶段的开发代号为 Project Spartan (音译为斯巴达, 同 Cortana 正好都是微软游戏《光晕》中的角色), 后期微软宣布了其正式命名(意为“边缘”)。


Edge 经过精心设计后, 使用界面非常简洁而大气, 给人一种清爽的感觉, 因此被很多用户寄予厚望。这款浏览器也是目前唯一一款内建即使在网页上增加注解功能和与语音助手小娜深度整合的浏览器。

Edge 使用方法与特色功能

Edge 使用相当容易, 只需要从任务栏或开始菜单里找到  图标即可开始使用。Edge

从图标到外观都尽可能与其他浏览器近似, 用户应该不会感到很强烈的陌生感, 甚至在某些方面, Edge 可能要来的更加容易。

然而并非任何时候 Edge 都能完美显示网页。对于网银等使用了ActiveX 插件技术的网页, Edge 会提示你使用 Internet Explorer 访问。

Edge 具有一个“阅读列表”功能, 与收藏夹不同, 你可以使用阅读列表来保存你希望稍后阅读的文章或其他内容。按地址栏上的“收藏”按钮, 然后点击“阅读列表”, 更改条目名称为你喜欢的名字, 保存即可。阅读列表清单可以点击工具栏上的  (中心) 打开(图 3-18)。这里还可以查看你的收藏夹、历史记录和下载。

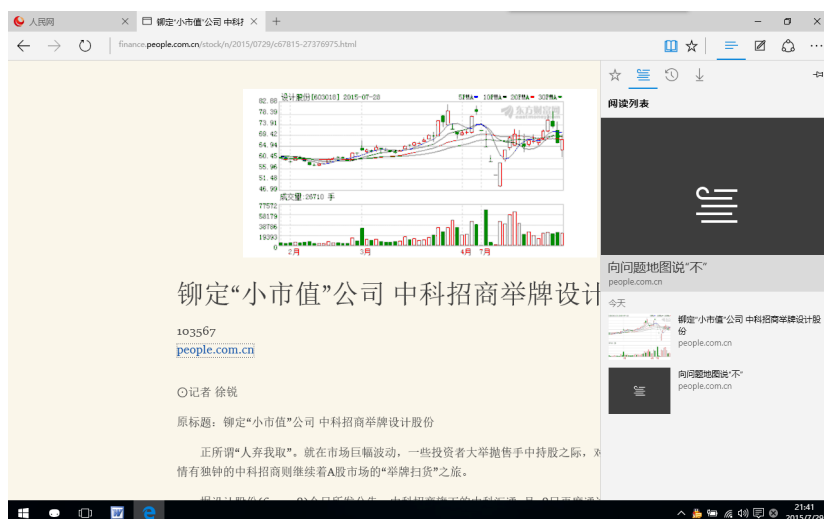


图 3-18 阅读列表清单


Edge 也具有一个“阅读视图”功能, 它可以帮助用户自动删除掉与正文无关的杂乱图文。当阅读视图可用时, 按地址栏上的  , 网页将切换为阅读模式(图 3-19)。



图 3-19 阅读模式下的网页


Edge 还有一个独门绝技: “Web 笔记功能”。它也是目前唯一一款能够让你直接在网页上记笔记、书写、涂鸦和突出显示的浏览器。按工具栏上的  以使用它(图 3-20)。工具栏左侧分别为普通笔(即手写“远景论坛”)、荧光笔(即手写“PCBeta”)、橡皮擦(用来擦除笔迹)、标记(用于键入文字, 即“欢迎阅读十全十美”三行文字)和剪辑(用于选取部分网页内容到剪贴板); 右侧为保存按钮(可选择保存为 OneNote 笔记或阅读列表)和共享按钮(可共享到 OneNote)。



图 3-20 Web 笔记示例

前文已经提到了 Edge 的前身——斯巴达与小娜的关系。实际上它们确实是相互协作的好伙伴,当启用小娜后,在默认情况下小娜会将网页搜索结果反馈给 Edge;不仅如此,Edge 还深度集成了小娜的一些功能。例如,用户可以在搜索框中直接搜索天气信息(图 3-21),也可以在页面选中文字,然后按右键“询问 Cortana”,小娜会自动筛选出最有价值的信息显示在右侧(图 3-22)。

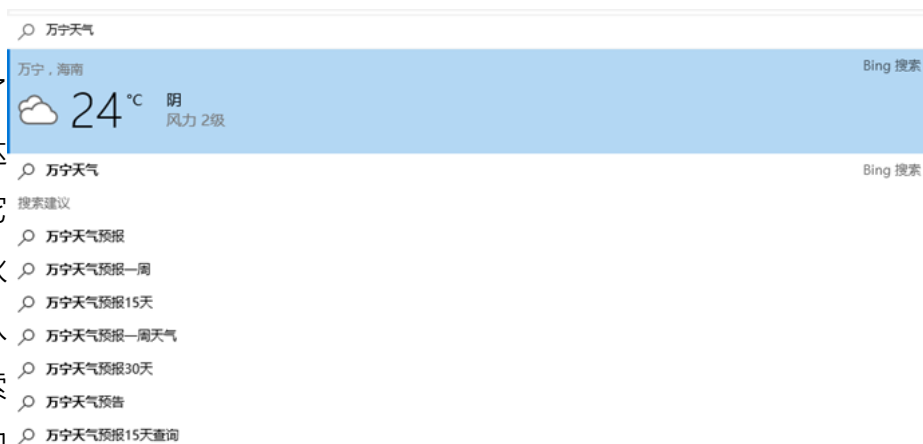


图 3-21 搜索天气信息

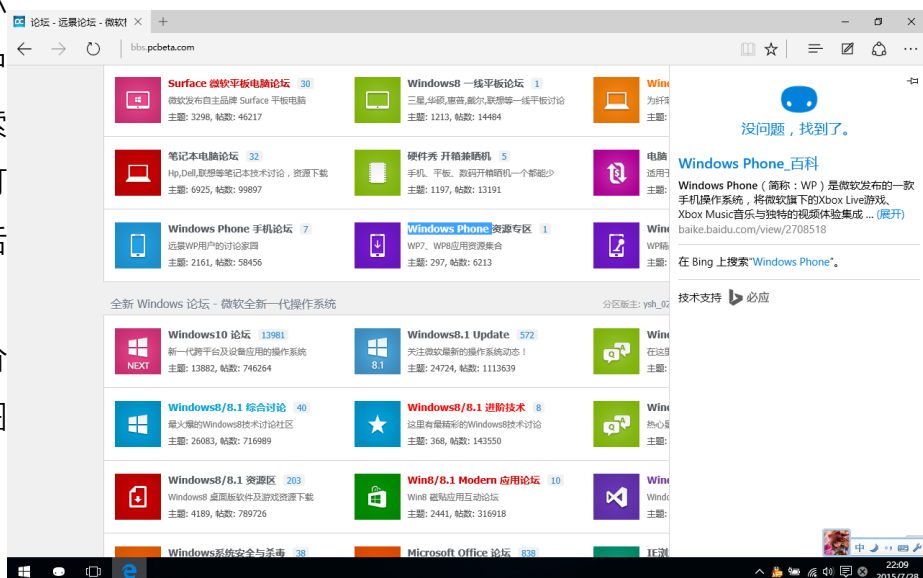


图 3-22 询问小娜

现阶段, Edge 缺少如下特性, 有待微软今后更新中添加:

对于浏览器插件的支持。

每个网页的实时缩略图。

不支持收藏当前所有网页。

不能通过文件夹打开一组标签。

后退和前进按钮右键没有显示最近网页列表。



Windows10 微百科

Windows 应用商店

应用商店的概念最早出现在 iOS 和 Android 上, 用于用户下载或购买应用。后来, 微软在 Windows 中也引入了应用商店, 里面并非只有 Modern 应用, 也同时提供传统的桌面程序, 不过应用商店会要求用户跳转到软件制造商页面下载。

在 Windows 8 开发者预览版之前, Windows 8 Milestone 3 Pre-release 就已经出现了应用商店。Build 843x 时, 现在的商店图标已经定型。Windows 8.1 时, 商店外观有所改造。

Windows 10 时代, 应用商店又被重新构建, 即集合了应用, 游戏, 音乐和电影的一站式应用商店, 实现了一次购买, 全平台通用的体验。今后当 Edge 支持插件, 插件也可以从应用商店获取。不过在中国, 暂时不支持获取音乐和电影内容。

同其他应用商店一样, Windows 应用商店主页也有滚动 Banner 用于显示最近受到关注的应用, 也支持查看排行榜和应用专题。右上角具有一个搜索框可以用来搜索应用, 单击头像可以管理账户以及设置商店。

Windows 10 Mobile

应该有人会问,《十全十美, Windows 10》通篇都在写电脑知识,怎么突然话锋一转到了手机?但是, Windows 10 是一个总称,电脑,手机等设备的操作系统都囊括其中, Windows 10 Mobile (下称 Win10M) 就是 Windows 10 的手机版。

Win10M, 前身即 Windows Phone, 是微软开发的一款手机操作系统。请不要同微软之前发布的 Windows Mobile 相混淆。一直以来, Windows Phone 的市占额都相对较低, 随着通用程序的逐渐增多, 将有望改善这一现状。

Win10M 与 WP8.1 相比, 多出了几项新特性:

像 PC 一样工作: 连接键盘、鼠标和监视器以提高效率。

全新浏览器: Microsoft Edge 非常适合用来阅读文章和在线完成工作。

操作中心: 交互式通知和设置集中于一处; 更多可自定义的快速操作。

Windows Hello: 通过视觉安全登录; 无需密码。

适用于手机的 Continuum: 以无线方式漂亮地投影到第二块屏幕, 而您所使用的仍然只是您的手机。

过改进的语音转文字技术: 无需动手, 即可方便地将语音内容转换为任何数据字段。

全新照片应用: 使用自动增强功能美化照片; 可自动创建相册。

全新的设置: 同 Windows 10 上的设置一样, 新的 Win10M 设置更加简单明了, 容易查找。

Windows 10 发行版本间的区别 (二)

上一章我们已经提到了 Windows 10 其中三个发行版本, 这章结合 Win10M 继续聊聊发行版本。

1. 企业版 LTSC: LTSC 即“长期服务支持分支”, 与企业版内容基本一致, 但 LTSC 删除了几乎所有 Modern 应用, 而且只会推送安全更新。这也就意味着, 使用 LTSC, 将不能接收到后续的功能性更新, 例如 Redstone 等重大的更新版本。除 LTSC 以外的发行版本均为当前分支 (Current Branch)。

2. 教育版: 这个版本功能同企业版比较接近, 主要面向学校教职员工和学生, 它将通过学术批量授权服务提供给用户。

3. Windows 10 IOT: IOT 即“物联网”版。这个版本用于嵌入某些特定硬件, 例如树莓派。Windows 10 IOT 将用于广泛的智能物联网设备。它为设备体系提供了具有企业级安全性的融合平台, 并通过 Azure 物联网服务提供机器到机器和机器到云的本地连接。根据设备的不同, 系统外观和硬件要求也有所不同。

4. “360 版”和“腾讯版”。它们其实指推送方式, 因为通过 360 或腾讯的升级工具推送来的还是家庭 / 专业版。同官方推送相比, 360 和腾讯推送的 Windows 10 修改了 Edge 设置, 并且捆绑了一些软件。不过相比较起来, 这种升级方式对于普通用户来说更容易接受。



初识通用应用

构筑开发环境

开发通用应用

通用应用的探究^①

①: 此章节可能涉及的内容并不常用且难以理解。如果您没有太大的需要, 此章节您只需要浏览一遍即可。

第一节

初识通用应用

Windows 的未来，尽在通用应用。

通用应用概述

图 4-1 通用应用的战略目标



通用应用最先于 Windows 8.1 提出，其意义为同一个应用，在不同种类的 Windows 设备（手机、平板电脑、个人电脑）上皆可顺利运行（图 4-1）。通用应用的最强之处是开发人员可以很轻易的编写跨平台应用，只需编写一次代码，稍作修改即可发布至全平台。在商店中，有些应用便标示着“获取一次，在兼容的 Windows Phone 上免费获取”，此举除了可以让用户在不同的设备上使用同一款应用，增加熟悉感，也因为无需重复购买，因此可以提升用户的购买意欲。

在 Windows 10 中，通用应用有了进一步完善。应用界面可以自适应窗口尺寸（可参看第一章第二节“响应式布局”）；应用可以实现多开，也支持“Continuum”新特性。所有这些，都凸显了通用应用的通用性。

通用应用的出现，有助于拯救 Windows Phone。一直以来，Windows Phone 的市占率较低，应用数量和质量也不尽人意。为了改变这一局面，Windows 10 允许运行从安卓和 iOS 转制来的通用应用。这将大大增加应用数量，以及减少开发者对通用应用的忧虑。开发者甚至可以将传统的桌面程序转制为通用应用，饱受诟病的注册表残留、滥用系统权限、推送广告等问题都将不复存在。

总之，我们看好通用应用的发展，但在专业领域尚不能完全用通用应用代替。

早在 Windows 8 时, 系统就开始了应用化的步伐, 随附了一些基础性的 Modern 应用, 同时添加了新的电脑设置。但兼顾到一般用户的使用习惯, 外观上显得激进的 Windows 8 并没有彻底砍除传统程序组件, 这样就存在了组件重复的情况。在 Windows 10 中, 微软正着手解决这个问题。

Windows 10 虽然表面上处处表现妥协 (比如开始菜单, 公用对话框重新使用 Win32 样式等); 但实际上一点也不妥协。Windows 10 的激进到底体现在哪里呢? 其实就是 Windows 系统组件的应用化。

截至 Windows 10 RTM, 已经完全被应用替代的系统组件主要有: 那就是开始菜单、OOBE 登陆界面、Windows Update、Windows Defender 设置页面、计算器、录音机、音量托盘、网络托盘、时钟托盘等。控制面板的设置选项也正逐步搬到设置应用。不仅是系统组件朝着完全应用化, 其实 Windows 10 添加的大部分新功能是用应用实现的, 比如 Microsoft Edge、小娜、操作中心等。增强的大部分组件也都是应用, 比如应用商店、照片、相机、音乐、视频等。

对于一些不喜欢应用的人而言, Windows 10 只是一个悲情的开始, 因为微软不会放弃系统组件的应用化。接下来, 记事本和文件管理器是最可能被应用替代的两个组件。

系统组件的应用化, 有以下优点:

组件化, 不用通过 WinSXS 可以省去不少麻烦; 对高 DPI 和触控的完美支持; Windows 10 的系统组件应用化会在一定程度上提升开发者对 UWA 的兴趣。

有优点则必然有缺点:

性能损失, 在低端平板上滑动开始屏幕, CPU 占用奇高; 奇葩 Bug 增多, 例如开始菜单的崩溃; Broker 满天飞, 即应用权限低, 当系统组件需要应用没有的权限时, 则通过 Socket 与 Broker 通信。这也是性能下降的罪魁祸首之一 (图 4-2)。

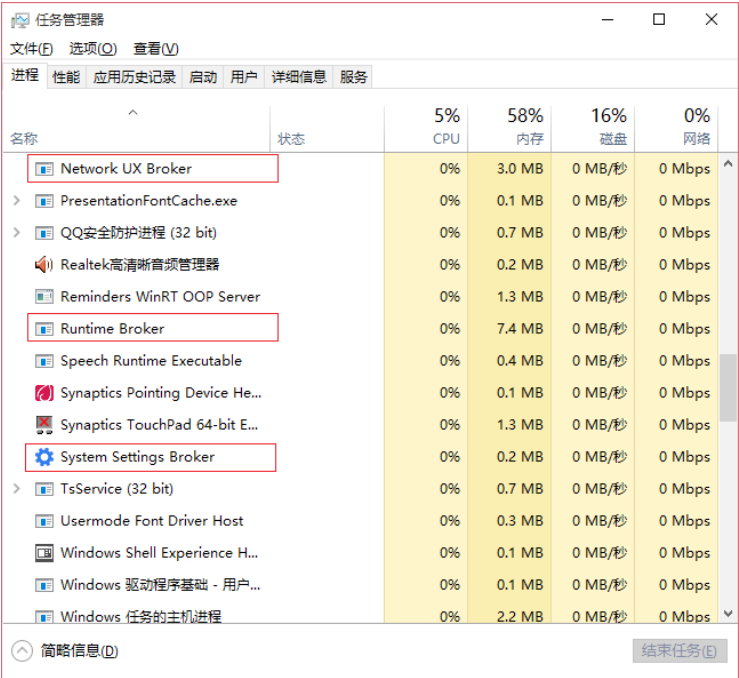


图 4-2 任务管理器中的 Broker

虽然应用并不完美; 但是我们相信随着微软的逐渐改进最终会好很多。

第二节

构筑开发环境

千里之行，始于足下，开发通用应用也是如此。

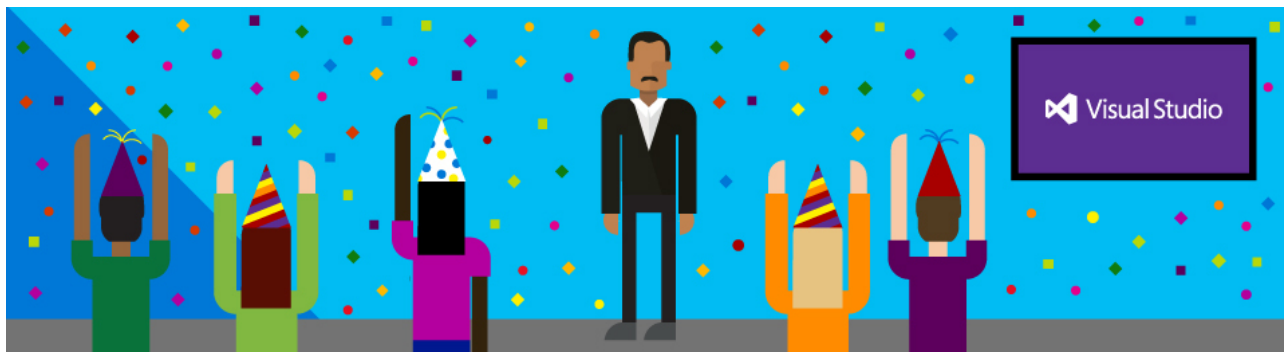


图 4-3 Visual Studio

Microsoft Visual Studio(简称 VS, 图 4-3) 是美国微软公司的开发工具包系列产品。VS 是一个基本完整的开发工具集, 它包括了整个软件生命周期所需要的大部分工具, 如 UML 工具、代码管控工具、集成开发环境 (IDE) 等等。所写的目标代码适用于微软支持的所有平台, 包括 Microsoft Windows、Windows Mobile、Windows CE、.NET Framework、.NET Compact Framework 和 Microsoft Silverlight 及 Windows Phone。Visual Studio 是目前最流行的 Windows 平台应用程序的集成开发环境。最新版本为 Visual Studio 2015 版本, 基于 .NET Framework 4.5.2。

简体中文版下载地址^①

专业版(包括 X86 和 X64):

文件名 cn_visual_studio_professional_2015_x86_x64_dvd_6846645.iso

SHA1 629E7154E2695F08A3C692C0B3F6CE19DF6D3A72

文件大小 3.84GB

发布时间 2015-07-20

下载链接: ed2k://file|cn_visual_studio_professional_2015_x86_x64_dvd_6846645.iso|4125556736|6EECD10D20D74704C919CC60F088823E|/

企业版(包括 X86 和 X64):

文件名 cn_visual_studio_enterprise_2015_x86_x64_dvd_6846222.iso

SHA1 4FFA1EE3E2D3337D3EDAE550A3583ABE9C426BEF

文件大小 3.89GB

发布时间 2015-07-20

下载链接 ed2k://file|cn_visual_studio_enterprise_2015_x86_x64_dvd_6846222.iso|4172560384|E558149A422E9FBF7D1D37FB0A2F1F53|/

①: 更多版本请访问 MSDN。

第三节

开发通用应用

创建你的第一个 Windows 10 通用应用。

“Hello World”是程序员初学的一句编程套用语。在这节中,我们将以 C++ 和 XAML 为例,开发一个含有“Hello World”的通用应用。

与常规情况的差异

如果你习惯使用 C++ 编写 Windows 桌面程序,你可能会发现 Windows 应用商店应用和 Windows Phone 应用编程的某些方面与此类似,但在其他一些方面则需要了解更多知识。相同之处有:

你可以使用 STL、CRT (有某些例外) 以及任何其他 C++ 库,只要代码不尝试调用不可从 Windows 运行时环境访问的 Windows 函数。

如果你习惯可视设计器,你仍然可以使用内置于 Microsoft Visual Studio 的设计器,或者你可以使用功能更加完整的 Blend for Visual Studio。如果你习惯手动编写 UI 代码,则可以手动编写 XAML 的代码。

你仍然可以创建使用 Windows 操作系统类型和你自己的自定义类型的应用。

你仍然可以使用 Visual Studio 调试器、探查器和其他开发工具。

你仍然可以通过使用 Visual C++ 编译器创建用来编译原生机器代码的应用。使用 C++ 编写的 Windows 应用商店应用不能在托管的运行时环境中执行。

新增内容

Windows 应用商店应用和 Universal Windows App 的设计准则与桌面应用的设计准则有很大差别。设计的重点不再是窗口边框、标签和对话框等。内容才是最重要的。出色的 Universal Windows App 从最开始的规划阶段就严格遵循这些准则。

你将使用 XAML 定义整个 UI。在 Universal Windows App 中,UI 与核心程序逻辑之间的分离远比在 MFC 或 Win32 应用中更为清晰。你在代码文件中处理行为的同时,其他用户可以在 XAML 文件中处理 UI 的外观。

尽管在 Windows 设备上 Win32 仍然可用于某些功能,但你将主要面向一个易于导航且面向对象的全新 API (即 Windows 运行时) 进行编程。

使用 C++/CX 使用和创建 Windows 运行时对象。C++/CX 支持 C++ 异常处理、代理、事件和动态创建的对象自动引用计数。使用 C++/CX 时,基础 COM 和 Windows 体系结

构的详细信息从应用代码中隐藏。有关详细信息, 请参阅 C++/CX 语言参考。

你的应用将编译为一个程序包, 其中还包含有关你的应用所包含的类型、它使用的资源以及它需要的功能(文件访问、Internet 访问和相机访问等)的元数据。

在 Windows 应用商店和 Windows Phone 应用商店中, 你的应用通过一个验证流程确定为安全之后, 即可面向无数潜在客户发布。

开始开发应用

我们的第一个应用是“Hello World”, 它演示了交互性、布局和样式的一些基本功能。我们将通过 Universal Windows App 项目模板创建应用。如果你之前开发过 Windows 8.1 和 Windows Phone 8.1 应用, 你可能还记得在 Visual Studio 中, 你必须具有三个项目: 一个用于 Windows App, 一个用于手机应用, 另一个用于共享代码。Windows 10 Universal Windows Platform (UAP) 使只有一个项目成为可能, 该项目可在所有设备上运行, 包括运行 Windows 10 的台式机和笔记本电脑, 平板电脑、手机等设备, 等等。

我们将从基础开始:

如何使用 Visual Studio 2015 创建 Universal Windows App。

如何了解创建的项目和文件。

如何了解 Visual C++ 组件扩展中的扩展以及何时使用它们。

首先, 在 Visual Studio 中创建一个解决方案(图 4-4):

1. 在 Visual Studio 中, 依次选择“文件”>“新建”>“项目”。
2. 在“新建项目”对话框的左侧窗格中, 展开“已安装”>“Visual C++”>“Windows”>“Windows 通用”。
3. 在中心窗格中, 选择“空白应用 (Windows 通用)”。
4. 输入项目名称。我们将其命名为 HelloWorld。
5. 选择“确定”按钮。已创建项目文件。

在继续之前, 让我们看一下解决方案的内容(图 4-5)。

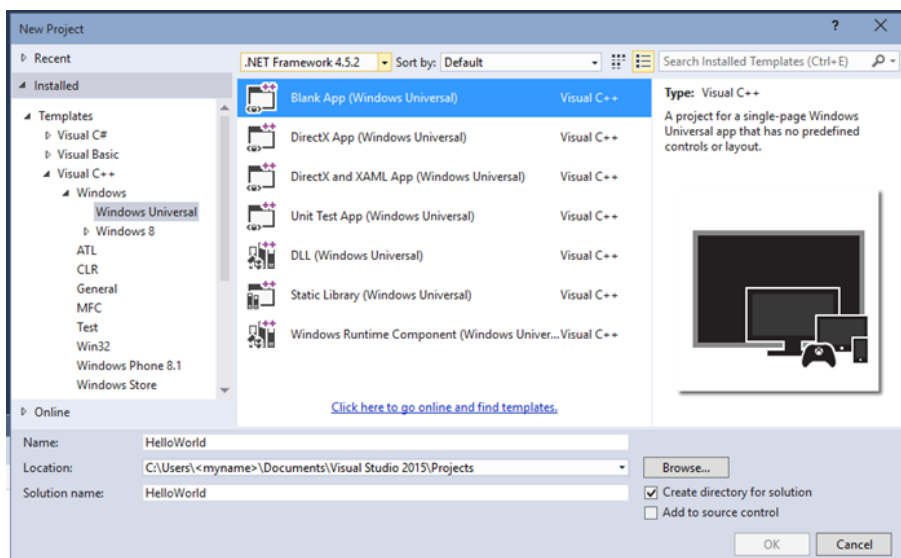


图 4-4 创建空白通用应用

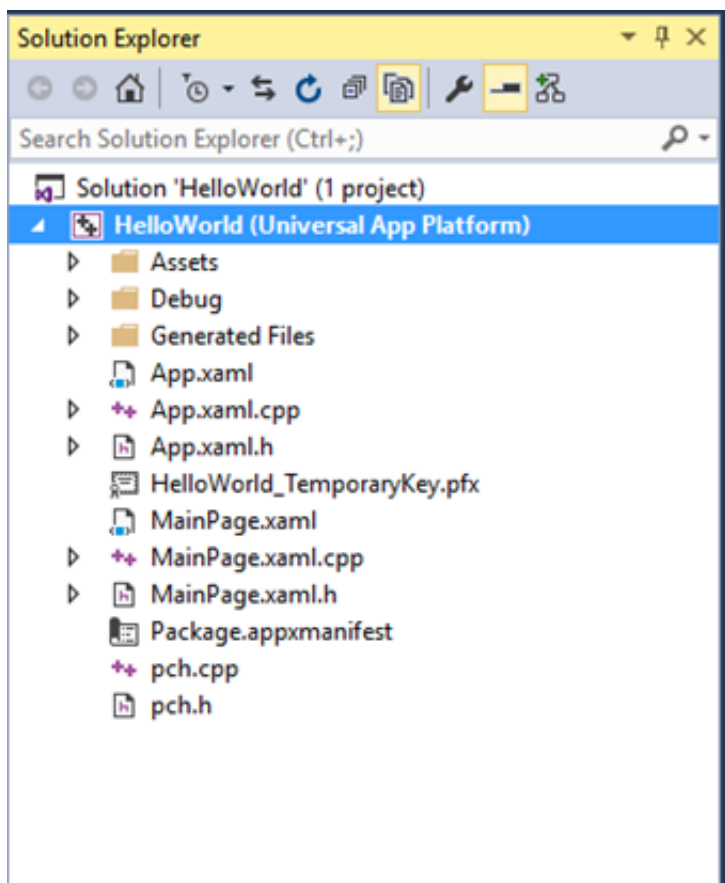


图 4-5 解决方案的内容

关于项目文件

项目文件中的每个 .xaml 文件都在同一个文件夹有对应的 .xaml.h 文件和 .xaml.cpp 文件, 在“生成的文件”文件夹中有 .g 文件和 .g.hpp 文件。修改 XAML 文件以创建 UI 元素, 并将其连接到数据源 (DataBinding)。修改 .h 和 .cpp 文件以为事件处理程序添加自定义逻辑。自动生成的文件表示 XAML 标记到 C++ 的转换。不要修改这些文件, 但是请研究它们以更好地了解代码隐藏的工作原理。基本上, 生成的文件包含 XAML 根元素的部分类定义; 此类是你在 *.xaml.h 和 .cpp 文件中修改的相同类。生成的文件将 XAML UI 子元素声明为类成员, 以便你可以在你编写的代

码中引用它们。在构建时, 生成的代码和你的代码合并为完整的类定义, 然后进行编译。

让我们先看一下项目文件。

App.xaml、App.xaml.h、App.xaml.cpp: 代表应用程序对象, 这是应用的入口点。App.xaml 不包含特定于页面的 UI 标记, 但是你可以添加希望可以从任何页面访问的 UI 样式和其他元素。代码隐藏文件包含 OnLaunched 和 OnSuspending 事件的处理程序。通常, 你在此处添加自定义代码以应用启动时初始化应用并在它挂起或终止时执行清理。

MainPage.xaml、MainPage.xaml.h、MainPage.xaml.cpp: 包含应用中的默认“开始”页面的 XAML 标记和代码隐藏。它没有导航支持或内置控件。

pch.h, pch.cpp: 预编译标头文件和在你的项目中包含它的文件。在 pch.h 中, 你可以在解决方案中包含任何不经常更改且包含在其他文件中的标头。

package.appxmanifest: 描述应用所需的设备功能、应用版本信息以及其他元数据的 XML 文件。要在“清单设计器”中打开此文件, 只需右键单击它。

HelloWorld.Windows_TemporaryKey.pfx: 在 Visual Studio 中, 支持在此计算机上进行应用部署的密钥。

代码一览

如果检查已共享项目的 App.xaml.h 和 App.xaml.cpp 中以及每个特定于平台的项目的 MainPage.xaml.h 和 .cpp 中的代码, 你将注意到大部分 C++ 代码看起来都很熟悉。但是, 如果你不熟悉 Windows RT 应用, 或者你使用过 C++/CLI, 某些语法元素可能不那么熟悉。以下是你将在 C++/CX 中看到的最常见的非标准语法元素:

引用类

几乎所有 Windows 运行时类, 包括 Windows API--XAML 控件中的所有类型、应用中的页面、应用类本身, 所有设备和网络对象、所有容器类型, 都声明为 ref class。(一些 Windows 类型是 value class 或 value struct)。引用类可从任何语言使用。在 C++ 中, 这些类型的生存期由自动引用计数管理(非垃圾集合), 以便你从来不会明确地删除这些对象。你也可以创建自己的引用类。

```
namespace HelloWorld
{
    public ref class MainPage sealed
    {
    public:
        MainPage();
        ...
    };
}
```

所有 Windows 运行时类型必须在命名空间中进行声明, 而且与 ISO C++ 中不同, 这些类型本身具有辅助功能修改器。public 修改器使类对命名空间之外的 Windows 运行时组件可见。sealed 关键字表示该类无法用作基类。几乎已密封所有引用类; 不广泛支持类继承, 因为 Javascript 不理解它。

ref new 和 ^ (hats)

你通过使用 ^ (乘幂号) 运算符声明引用类的变量, 并且使用引用新关键字实例化该对象。然后你使用 -> 运算符访问对象的实例方法, 就像 C++ 指针一样, 使用 :: 运算符访问静态方法, 就像 ISO C++ 中一样。

在以下代码中, 我们使用完全限定的名称来实例化对象, 并使用 -> 运算符调用实例方法。

```
Windows::UI::Xaml::Media::Imaging::BitmapImage^ bitmapImage =
    ref new Windows::UI::Xaml::Media::Imaging::BitmapImage();

bitmapImage->SetSource(fileStream);
```

通常, 在 .cpp 文件中, 我们将添加 using namespace Windows::UI::Xaml::Media::Imaging 指令和自动关键字, 因此相同的代码看似如下:

```
auto bitmapImage = ref new BitmapImage();
bitmapImage->SetSource(fileStream);
```

属性

引用类可以具有属性, 和在托管的语言中一样, 这些属性是对使用代码显示为字段的特殊成员函数。


```

public ref class SaveStateEventArgs sealed
{
    public:

        // Declare the property
        property Windows::Foundation::Collections::IMap<Platform::String^, Platform::-
Object^>^ PageState
        {
            Windows::Foundation::Collections::IMap<Platform::String^, Plat-
form::Object^>^ get();
        }
    ...
};

...
// consume the property like a public field
void PhotoPage::SaveState(Object^ sender, Common::SaveStateEventArgs^ e)
{
    if (mruToken != nullptr && !mruToken->IsEmpty())
    {
        e->PageState->Insert( "mruToken" , mruToken);
    }
}

```

代理

和在托管语言中一样，代理是包装带有特定签名的函数的引用类型。它们经常与事件和事件处理程序一同使用

```

// Delegate declaration (within namespace scope)
public delegate void LoadStateEventHandler(Platform::Object^ sender, LoadStateEventArgs^ e);

// Event declaration (class scope)
public ref class NavigationHelper sealed
{
    public:
        event LoadStateEventHandler^ LoadState;
};

// Create the event handler in consuming class
MainPage::MainPage()
{
    auto navigationHelper = ref new Common::NavigationHelper(this);
    navigationHelper->LoadState += ref new Common::LoadStateEventHandler(this, &MainPage::LoadState);
}

```

向应用添加内容

让我们来向应用添加一些内容。

步骤 1: 修改你的起始页

在“解决方案资源管理器”中，打开 MainPage.xaml。

通过向根 Grid 添加以下 XAML 来为 UI 创建控件，紧挨在其结束标记之前。它包含 StackPanel，后者具有一个询问用户名的 TextBlock、一个接受用户名的 TextBox 元素、一个 Button 和其他 TextBlock 元素。

```

<StackPanel Margin=" 120,30,0,0" >
  <TextBlock HorizontalAlignment=" Left" Text=" Hello World" FontSize=" 36" />
  <TextBlock Text=" What' s your name?" />
  <StackPanel Orientation=" Horizontal" Margin=" 0,20,0,20" >
    <TextBox x:Name=" nameInput" Width=" 300" HorizontalAlignment=" Left" />
    <Button Content=" Say "Hello"" />
  </StackPanel>
  <TextBlock x:Name=" greetingOutput" />
</StackPanel>

```

我们将在导航、布局和视图文章中谈论有关 XAML 布局的详细信息。

此时, 你已创建了一个非常基本的 Windows 应用。若要查看 Windows 应用商店应用的外观, 请按 F5, 在调试模式下构建、部署并运行该应用。

首先会出现默认的初始屏幕。它有一个图像 (Assets\SplashScreen.scale-100.png) 以及在应用的清单文件中指定的背景色。若要了解如何自定义初始屏幕, 请参阅添加初始屏幕。

当初始屏幕消失时, 将显示你的应用。它显示黑屏和标题 "My Application"。

没有关闭应用的按钮或命令。尽管你可以使用关闭手势或 Alt+F4 来关闭它, 但是通常不关闭 Windows 应用商店应用。(第 2 部分: 管理应用生命周期和状态中讨论了这一点。) 按 Windows 键可转到“开始”菜单的应用列表, 而且请注意, 部署应用时会将其磁贴添加到“开始”菜单。若要再次运行应用, 只需点击或单击该应用的磁贴, 或在 Visual Studio 中按 F5 在调试模式下运行该应用。

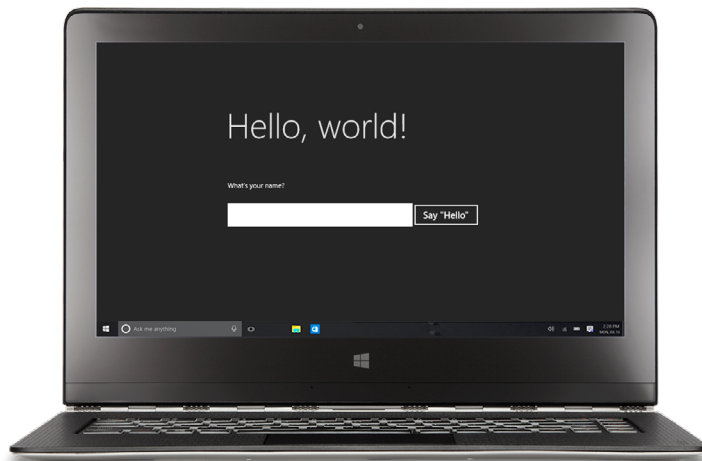


图 4-6 创建的通用应用页面

它还做不了多少事, 但是祝贺你, 你已经构建了你的第一个 Windows 应用商店应用!

要停止调试并关闭应用, 请返回到 Visual Studio 并按 Shift+F5。

有关详细信息, 请参阅从 Visual Studio 运行 Windows 应用商店应用。

在应用中, 你可以在 TextBox 中键入, 但是单击 Button 不执行任何操作。在以后的步骤中, 为按钮的 Click 事件创建事件处理程序, 该事件显示了个性化问候。向 MainPage.xaml.h 和 MainPage.xaml.cpp 文件添加事件处理程序代码。

在移动设备仿真器上启动该应用

你的应用可在任何 Windows 10 设备上运行, 让我们看一下它在 Windows Phone 上的情况如何。

除了在桌面设备上执行调试的选项, Visual Studio 还提供用于在连接到计算机的物理移动设备上或移动设备仿真器上部署和调试应用的选项。你可以为带有不同内存和显示配置的设备在仿真器中进行选择。

设备

仿真器 10.0.0.0 WVGA 4 英寸 512MB

采用其他配置的各种仿真器

最好在带有小型屏幕和有限内存的设备上测试应用, 因此请使用“仿真器 10.0.0.0 WVGA 4 英寸 512MB”选项。

提示 有关使用手机仿真器的详细信息, 请参阅在仿真器中运行 Windows Phone 应用。若要在物理设备上调试你的应用, 你必须具有已注册用于开发的设备。有关详细信息, 请参阅注册你的 Windows Phone。

在移动设备仿真器上开始调试

1. 在“标准”工具栏上的目标设备菜单 ( Local Machine) 中, 选取“仿真器 10.0.0.0 WVGA 4 英寸 512MB”。
2. 单击工具栏中的“开始调试”按钮 (); 或在“调试”菜单中, 单击“开始调试”; 或按下 F5。

在移动设备仿真器中, 应用的外观如下所示。

Visual Studio 将启动选定的仿真器, 然后部署和启动你的应用。你会首先注意到, 在本地计算机上能正常显示的 120 像素左边距, 在屏幕较小的移动设备上会将内容推到屏幕之外。在本教程的后面部分中, 你将了解如何使 UI 适应不同的屏幕大小, 以使应用始终保持外观良好。

你可能还会注意到, 你可以在 TextBox 中键入内容, 但此时单击 Button 不会起任何作用。在接下来的步骤中, 你将为按钮的 Click 事件创建事件处理程序, 以显示个性化的问候。将事件处理程序代码添加到 MainPage.xaml.cs 文件。

步骤 2: 创建事件处理程序

1. 在 MainPage.xaml 的 XAML 或设计视图中, 在之前添加的 StackPanel 中选择 "Say Hello"Button。


2. 按 Alt+Enter 打开“属性窗口”, 然后选择事件按钮 ()。(图 4-8)



图 4-7 移动应用仿真器中的通用应用界面

3. 查找 Click 事件。在其文本框中, 键入处理 Click 事件的函数名称。对于本示例, 请键入 "Button_Click"。

4. 按 Enter。事件处理程序方法在 MainPage.xaml.cpp 中创建并已打开, 因此你可以添加在事件出现时执行的代码。

同时, 在 MainPage.xaml 中, 更新 Button 的 XAML 以声明 Click 事件处理程序, 如下所示:

```
<Button Content=" Say &quot;Hello&quot;" Click=" Button_Click" />
```

5. 在 MainPage.xaml.cpp 中, 将以下代码添加到你刚刚创建的 Button_Click 事件处理程序。此代码将从 nameInputTextBox 控件检索用户名并使用该用户名创建问候。greetingOutput TextBlock 将显示相关结果。

6. 按 Enter。事件处理程序方法在 MainPage.xaml.cpp 中创建并已打开, 因此你可以添加在事件出现时执行的代码。

同时, 在 MainPage.xaml 中, 更新 Button 的 XAML 以声明 Click 事件处理程序, 如下所示:

```
<Button Content=" Say &quot;Hello&quot;" Click=" Button_Click" />
```

7. 在 MainPage.xaml.cpp 中, 将以下代码添加到你刚刚创建的 Button_Click 事件处理程序。此代码将从 nameInputTextBox 控件检索用户名并使用该用户名创建问候。greetingOutput TextBlock 将显示相关结果。

```
void HelloWorld::MainPage::Button_Clicked(Platform::Object^ sender, Windows::UI::Xaml::RoutedEventArgs^ e)
{
    greetingOutput->Text = "Hello, " + nameInput->Text + "!";
}
```

8. 将项目设置为启动, 然后按 F5 生成并运行应用。当你在文本框中键入姓名并单击按钮后, 应用会显示个性化问候 (图 4-9)。

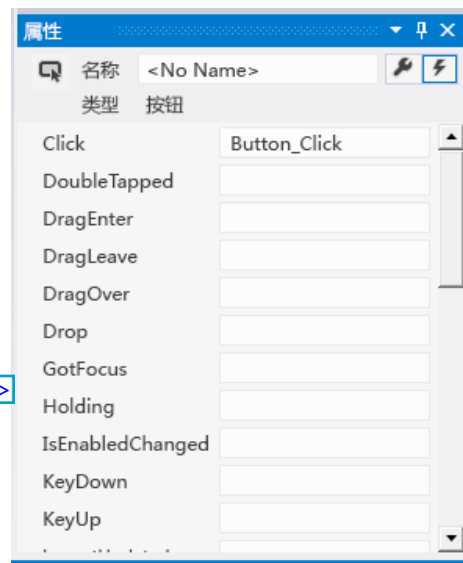


图 4-8 应用控件属性窗口

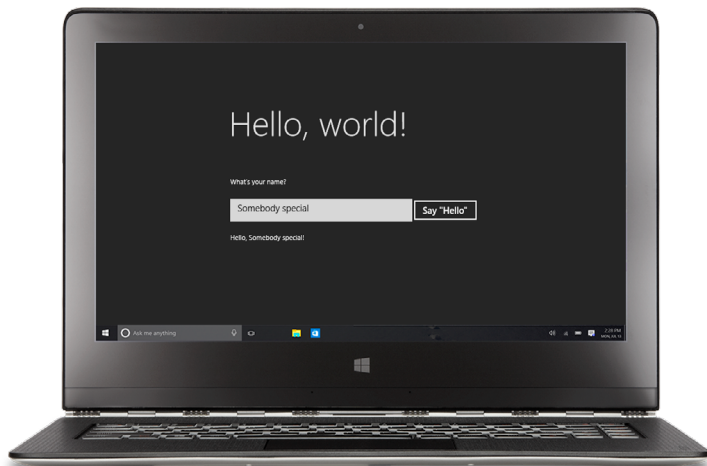


图 4-9 通用应用中的“问候”

步骤 3: 设置起始页的样式

选择主题

轻松自定义应用的外观。默认情况下,应用使用深色样式的资源。系统资源还包含浅色主题。我们来尝试一下并看看它的外观。

切换到浅色主题的步骤

1. 在“已共享”项目中, 打开 App.xaml。
2. 在开始标记 Application 中, 添加 RequestedTheme 属性, 并将其值设置为 Light:

RequestedTheme="Light"

以下是添加了浅色主题的完整 Application 标记:

```
<Application
  x:Class=" HelloWorld.App"
  xmlns=" http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x=" http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:local=" using:HelloWorld"
  RequestedTheme=" Light" >
```

1. 使其中一个平台项目成为启动项目然后按 F5 以构建并运行它。请注意它使用浅色主题。现在将其他平台项目设置为启动项目, 按 F5, 并且请注意它也使用浅色主题。这是因为 App.xaml 已由两个项目共享。



图 4-10 通用应用的浅色主题


你应使用哪个主题? 你需要的任何一个以下我们的观点对于主要显示图像或视频的应用, 我们建议深色主题; 对于包含大量文本的应用, 我们建议浅色主题(图 4-10)。如果你使用的是自定义配色方案, 则请使用最适合应用外观和感觉的主题。

注意 在应用启动时应用主题, 并且无法在应用运行时更改主题。

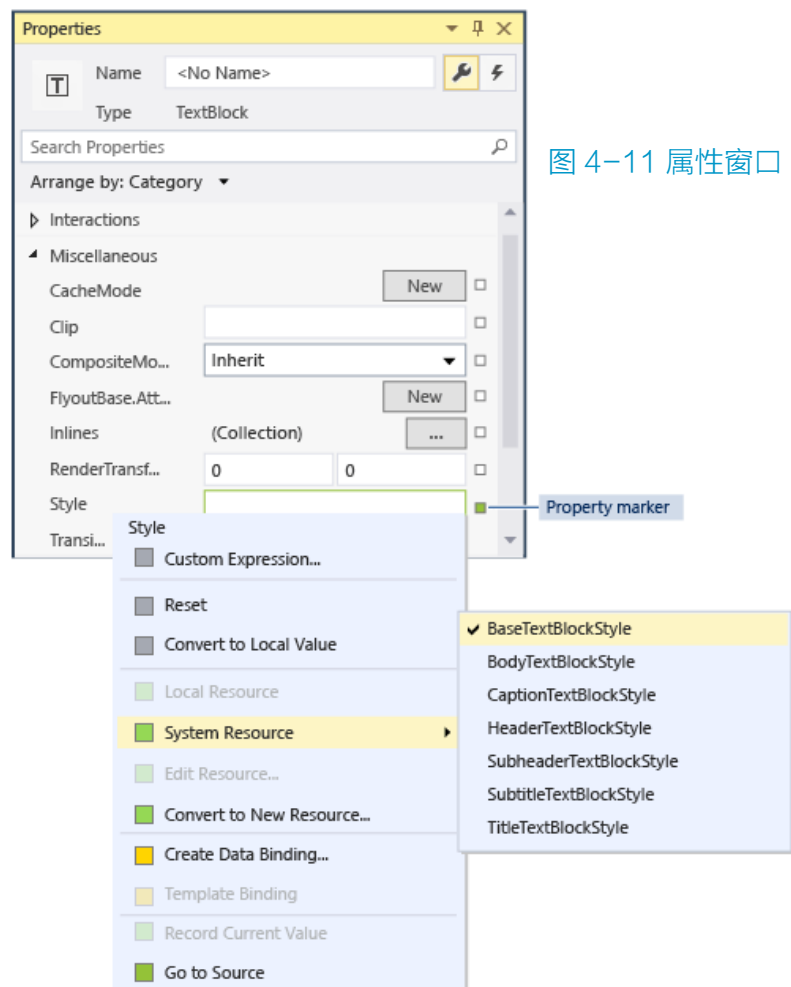
使用系统样式

此时, 在 Windows 应用中, 所有文本都非常小, 因此很难阅读。让我们通过应用系统样式来解决该问题。

更改元素样式的步骤

1. 在 Windows 项目中, 打开 MainPage.xaml。
2. 在 XAML 或设计视图中, 选择以前添加的 "你叫什么名字?" TextBlock。
3. 在“属性”窗口(图 4-11)中, 选择右上角的“属性”按钮 ()。
4. 展开“文本”组, 并将字体大小设置为 18 px。
5. 展开“其他”组并找到“样式”属性。
6. 单击属性标记 (“样式”属性右侧的绿色框), 然后在菜单上, 依次选择“系统资源” > “BaseTextBlockStyle”。

“BasicTextStyle”为在 <root>\Program Files\Windows Kits\10\Include\winrt\xaml\design\generic.xaml 中的 ResourceDictionary 中定义的资源。



在 XAML 设计图面中, 文本外观会发生更改。在 XAML 编辑器中, TextBlock 的 XAML 会进行更新:

```
<TextBlock Text=" What' s your name?" Style="{StaticResource BasicTextStyle}" />
```

7. 重复该过程可设置字体大小并将“BaseTextBlockStyle”分配到 greetingOutputTextBlock 元素。

提示 尽管此 TextBlock 中没有文本, 但当你将指针移动到 XAML 设计图面上时, 蓝色轮廓会显示它的位置, 以便你可以选择它。

现在的 XAML 如下所示:

```

<StackPanel Margin=" 120,30,0,0" >
  <TextBlock Style="{ThemeResource BaseTextBlockStyle}" FontSize=" 16" Text=" What' s your name?" />
  <StackPanel Orientation=" Horizontal" Margin=" 0,20,0,20" >
    <TextBox x:Name=" nameInput" Width=" 300" HorizontalAlignment=" Left" />
    <Button Content=" Say &quot;Hello&quot;" Click=" Button_Click" />
  </StackPanel>
  <TextBlock Style="{ThemeResource BaseTextBlockStyle}" FontSize=" 16" x:Name=" greetingOutput" />
</StackPanel>

```

8. 按 F5 构建并运行应用。现在它的外观如下所示 (图 4-12) :



图 4-12 浅色主题中通用应用的问候

步骤 4: 使 UI 适应不同的窗口大小

现在我们将使 UI 适应不同的屏幕大小, 以使其在移动设备上外观良好。若要执行此操作, 添加 VisualStateManager 并设置应用于不同视觉状态的属性。

1. 在 XAML 编辑器中, 在根 Grid 元素的开始标记后添加此 XAML 块。

```

<VisualStateManager.VisualStateGroups>
  <VisualStateGroup>
    <VisualState x:Name=" wideState" >
      <VisualState.StateTriggers>
        <AdaptiveTrigger MinWindowWidth=" 641" />
      </VisualState.StateTriggers>
    </VisualState>
    <VisualState x:Name=" narrowState" >
      <VisualState.StateTriggers>
        <AdaptiveTrigger MinWindowWidth=" 0" />
      </VisualState.StateTriggers>
      <VisualState.Setters>
        <Setter Target=" contentPanel.Margin" Value=" 20,30,0,0" />
        <Setter Target=" inputPanel.Orientation" Value=" Vertical" />
        <Setter Target=" inputButton.Margin" Value=" 0,4,0,0" />
      </VisualState.Setters>
    </VisualState>
  </VisualStateGroup>
</VisualStateManager.VisualStateGroups>

```

2. 在本地计算机上调试应用。请注意, UI 外观与以前相同, 除非窗口变得窄于 641 与设备无关的像素 (DIP)。

3. 在移动设备仿真器上调试应用。请注意 UI 使用你在 narrowState 中定义的属性并正确显示在小屏幕上。

如果你在以前版本的 XAML 中使用过 VisualStateManager, 你可能会注意到 XAML 在此处使用简化的语法。

名为 wideState 的 VisualState 具有一个 AdaptiveTrigger, 并且其 MinWindowWidth 属性设置为 641。这意味着仅在窗口宽度不小于 641 DIP 的最小值时应用该状态。你没有为此状态定义任何 Setter 对象, 因此它会将你在 XAML 中定义的布局属性用于页面内容。

名为 narrowState 的第二个 VisualState 具有一个 AdaptiveTrigger, 其 MinWindowWidth 属性设置为 0。当窗口宽度大于 0 但小于 641 DIP 时, 应用此状态。(在 641 DIP 时, 应用 wideState。)在此状态下, 定义一些 Setter 对象以更改 UI 中控件的布局属性:

将 contentPanel 元素的左边距从 120 降低为 20。

将 inputPanel 元素的 Orientation 从 Horizontal 更改为 Vertical。

将 4 DIP 的上边距添加到 inputButton 元素。



图 4-13 移动应用仿真器中的浅色主题通用应用界面

这样, 你的第一个通用应用就写成了。利用其他编程语言开发通用应用以及应用设计可以参考 dev.windows.com。



Windows 10 移植应用的原理

在本章第一节中我们已经谈到了,Windows 10 允许运行从安卓和iOS 转制来的通用应用,也可以对 Win32 程序进行移植。这里,我们就来探讨移植原理。

Windows 10 运行安卓应用,依靠的是 Project Astoria,也叫 Android on Windows (下称 AOW),也叫安卓子系统。它是微软为安卓应用搭建的 Windows 桥梁,通过这种方式可以让你的安卓应用只经过少量修改 / 不用修改就可以在 Win10M 上运行。

同黑莓运行安卓应用的情形相似, AOW 里也包含了运行安卓应用所依赖的资源,这意味着性能会比预期的好,并提供如同 Win10M 版应用的应用体验。有趣的是,安卓环境中还包括 WebKit 渲染引擎,意味着在 Win10M 上可以选择第二个渲染引擎。最近,经过 WPer 研究,安卓应用更是可以直接安装在 Win10M 中(但并非所有应用都兼容)。

Project Astoria 其实是微软为了解决 Win10M 应用匮乏推出的一个举措。因此,利用 AOW 运行的应用只能兼容 Win10M,这类应用不属于通用应用。

而 Windows 10 运行 iOS 应用,依靠的是 Project Islandwood。通过这种方式可以移植你的 iOS 应用,并自动将 iOS 应用内支付、本地通知等替换为 Windows 平台的功能。通过将 iOS API 转发到对应的 Windows API,尽量减少移植过程中的代码的修改,同时扩展 iOS 应用程序,以便让其充分利用 Win10 通用应用平台的特性。目前这个项目支持 Objective-C,未来将支持 Swift。

利用 Project Islandwood 移植来的应用,可以在所有平台上运行,是真正的通用应用,也是面对多种设备兼容最理想的移植方式。

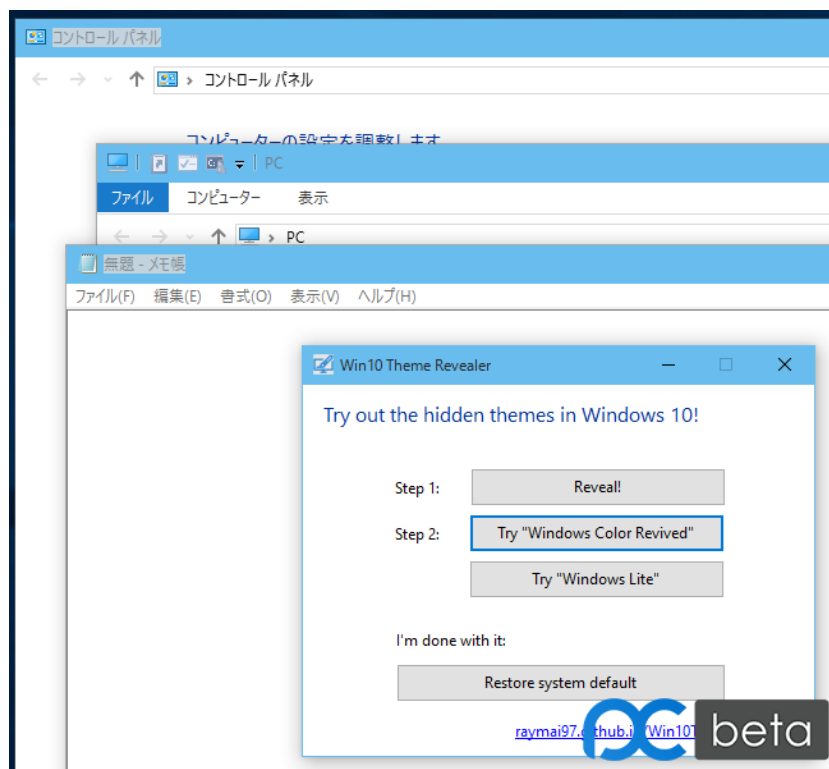
另外,微软还推出了两个项目,分别叫做 Centennial 和 Westminster。前者可以移植 Win32 程序,通过一些调整使得程序可以在 UWP 平台上运行;后者则可以将网站生成通用应用。

这四个计划,统称为 UWP Bridge (意即“沟通通用应用的桥梁”)。



附录一

技术文章大全



(图 /raymai97)

方法一：手动修改版

第一步：复制主题文件夹并重命名

定位到 C:\Windows\Resources\Themes，选取 aero 文件夹，然后马上复制并粘贴在原位，如果 MSS 文件存取被拒，直接略过；

现在把“aero – 复制”文件夹重命名为“Windows”；

进入 C:\Windows\Resources\Themes\Windows (谨记)，把 aero (.msstyles) 重命名为 windows (.msstyles)；

接着进入 en-US 和 / 或 zh-CN 文件夹，把“aero.msstyles.mui”重命名为“windows.msstyles.mui”。

第二步：修改主题文件

把 C:\Windows\Resources\Themes\ 文件夹下的 aero (.theme) 文件复制到桌面，重命名为 windows (.theme)。用记事本打开。

在记事本里面，找到 [VisualStyles] 这一区，然后删掉下面这一行：

```
Path=%ResourceDir%\Themes\Aero\Aero.msstyles
```

在那一行插入新的代码：

```
Path=%ResourceDir%\Themes\windows\windows.msstyles
```

步骤即为完成。

双击桌面的 Windows.theme 主题文件, 系统自动应用新的主题文件, 然后你的窗口标题栏就会呈现指定颜色了, 现在你可以在设置 - 个性化里面选择你想要的颜色。

方法二: 软件修改版

进入 <http://bbs.pcbeta.com/viewthread-1622189-1-1.html>, 下载软件修改, 按步骤操作即可。

(方 法 一 感 谢 eugenewangtw, 链 接: <http://bbs.pcbeta.com/viewthread-1611128-1-1.html>;

方 法 二 感 谢 raymai97 和 一 帆 风 顺 2001, 原 作 链 接: <http://bbs.pcbeta.com/viewthread-1619217-1-1.html>, 汉 化 版 链 接: <http://bbs.pcbeta.com/viewthread-1622189-1-1.html>)

如何替换登录背景

Windows 10 中为了与新的英雄桌面壁纸配套登录背景也做出了调整不再是单调的纯色。但是, 自带的登录背景比较压抑, 系统也并没有提供修改入口。替换登录背景?

首先, 请从开始菜单中找到 Windows PowerShell ISE 并运行, 复制 59 页全部文字并保存 ps1 文件至桌面: (img101.JPG 代表桌面上已有的一张壁纸)

右键单击保存的文件文件, 选择“使用 PowerShell 运行”, 此时会弹出 PS 运行窗口, 稍等片刻窗口消失后就会在桌面生成我们想要的 Windows.UI.Logon_new.pri 文件, 把它重命名为 Windows.UI.Logon.pri。

然后对 C:\Windows\SystemResources\Windows.UI.Logon 获取管理员所有权。进入 Windows.UI.Logon 目录, 先在原文件名后面添加 .Backup 后缀以作备份, 然后把自己生成的 Windows.UI.Logon.pri 文件复制到其中。这期间如果提示需要权限, 统统点击“继续”。这时候锁定电脑即可看到替换结果

```

$priPath = "$env:windir\SystemResources\Windows.UI.Logon\Windows.UI.Logon.pri"
$outputPath = ".\Windows.UI.Logon_new.pri"
$replacementPath = ".\img101.JPG"

$inputStream = [System.IO.File]::OpenRead($priPath)
$outputStream = [System.IO.File]::Create($outputPath)
$replacementStream = [System.IO.File]::OpenRead($replacementPath)

$inputReader = New-Object System.IO.BinaryReader -ArgumentList $inputStream
$outputWriter = New-Object System.IO.BinaryWriter -ArgumentList $outputStream

$inputStream.CopyTo($outputStream)

$replacementLengthAligned = ([Math]::Ceiling($replacementStream.Length / 8) * 8)

# header
$inputStream.Seek(0x14, "Begin" ) | Out-Null
$headerLength = $inputReader.ReadUInt32()
$inputStream.Seek(0xB8, "Begin" ) | Out-Null
$dataitemOffset = $inputReader.ReadUInt32()
$origDataitemLength = $inputReader.ReadUInt32()
$dataitemLength = $origDataitemLength + $replacementLengthAligned
$outputStream.Seek(0xBC, "Begin" ) | Out-Null
$outputWriter.Write([int]$dataitemLength)

# dataitem
$outputStream.Seek($headerLength + $dataitemOffset + 0x18, "Begin" ) | Out-Null
$outputWriter.Write([int]$dataitemLength)
$inputStream.Seek($headerLength + $dataitemOffset + 0x24, "Begin" ) | Out-Null
$stringCount = $inputReader.ReadUInt16()
$blobCount = $inputReader.ReadUInt16()
$origDataLength = $inputReader.ReadUInt32()
$outputStream.Seek(0xC, "Current" ) | Out-Null
$outputWriter.Write([int]($origDataLength + $replacementLengthAligned))
$outputStream.Seek($stringCount * 4, "Current" ) | Out-Null
for ($i = 0; $i -lt 10; $i++)
{
    $outputWriter.Write($origDataLength)
    $outputWriter.Write([int]$replacementStream.Length)
}
$outputStream.Seek(($blobCount - 10) * 8, "Current" ) | Out-Null

# data
$outputStream.Seek($origDataLength, "Current" ) | Out-Null
if ($outputStream.Length - $outputStream.Position -ne 0x18)
{
    Write-Error "Not compatible with this PRI file."
}
$replacementStream.CopyTo($outputStream)

# footer
$outputStream.Seek($replacementLengthAligned - $replacementStream.Length, "Current" ) | Out-Null
$outputWriter.Write(0xDEF5FADE)
$outputWriter.Write([int]$dataitemLength)
$outputWriter.Write(0xDEFFFADE)
$outputWriter.Write(0x00000000)
$outputWriter.Write([char[]] " mrm_pri2" )

$outputStream.Seek(0xC, "Begin" ) | Out-Null
$outputWriter.Write([int]$outputStream.Length)
$outputStream.Seek(-0xC, "End" ) | Out-Null
$outputWriter.Write([int]$outputStream.Length)

$inputReader.Close()
$outputWriter.Close()
$replacementStream.Close()

```

如何优化 Chrome 内核浏览器字体显示效果^①

Windows 10 中的雅黑加入了 Y 轴抗锯齿, 在高分屏下效果比较明显。但是 DirectWrite 的渲染机制只在 Edge 中效果良好, 在 Chrome 内核浏览器下显示效果惨不忍睹。我该如何让字体更加易读?

方法一: 直接关闭 DirectWrite



方法二: 直接替换雅黑字体

版本 6.20 之前的雅黑无渲染问题, 直接在 RE 下替换即可。注意, 需要提前准备好字体文件。例如, 你已准备好了 msyh.ttc 到 D 盘, 则在命令提示符中输入 `copy D:\msyh.ttc C:\Windows\Fonts`, 按 Y 确认。

雅黑粗体替换同理。替换完后重启进系统即可。

方法三: 利用扩展替换显示字体

雅黑生来就存在一些显示缺陷, 经过上面任一方法操作仍不能达到满意的, 可以使用扩展替换显示字体。注意, 这个方法同方法一正好相反, 需要打开 DirectWrite。

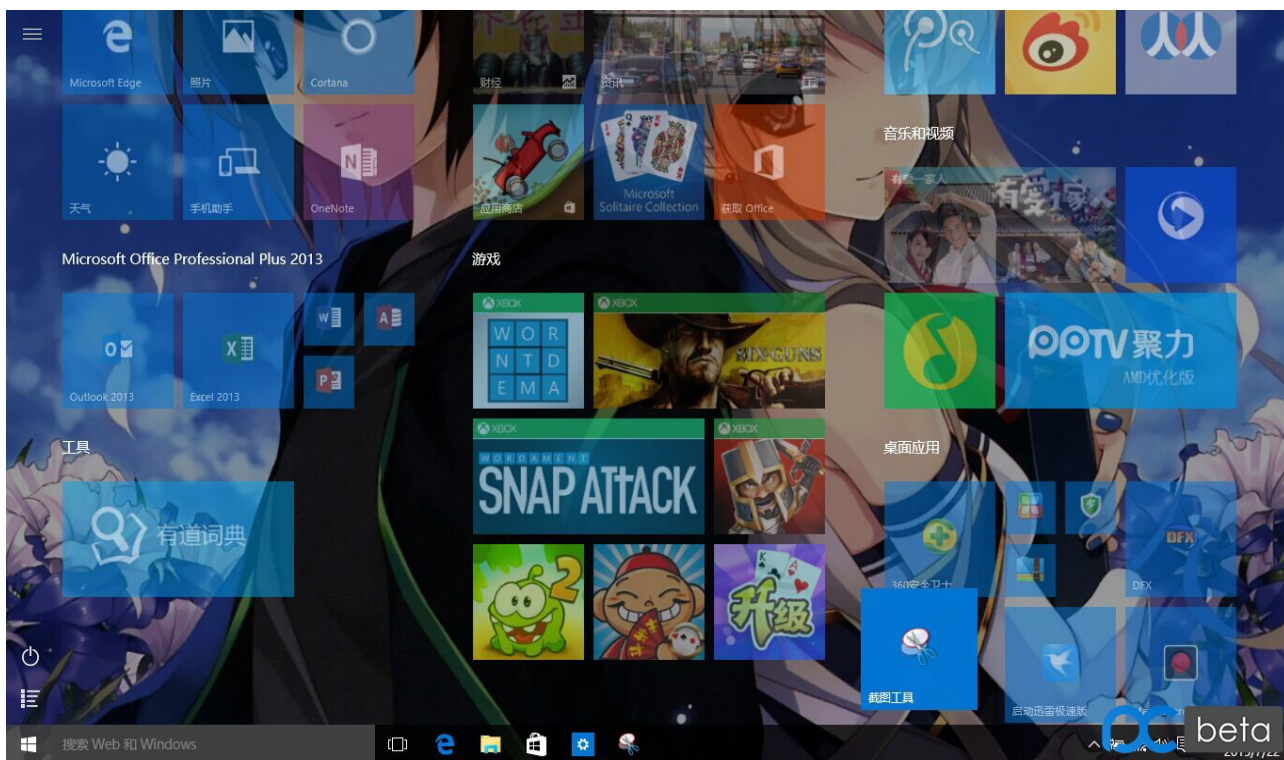
扩展地址: <http://pan.baidu.com/s/1sjBlIm9> 密码: gq60

①: 方法一: ITechDeveloper, <http://bbs.pcbeta.com/viewthread-1618905-1-1.html>; 方法三: blue_forever, <http://bbs.pcbeta.com/viewthread-1618950-1-1.html>

应用无法固定到开始的解决办法

Windows 10 听取民意, 恢复了开始菜单, 但使用过程中会发现, 将应用固定到开始菜单的功能貌似失效了, 无论你尝试点它几百下, 都没用! 将应用固定到任务栏的功能都好好的! 下面就是这个问题的解决方案。

进入设置——个性化——开始, 开启“使用全屏幕开始菜单”。然后, 来到开始菜单, 找到你想固定到开始菜单的应用。

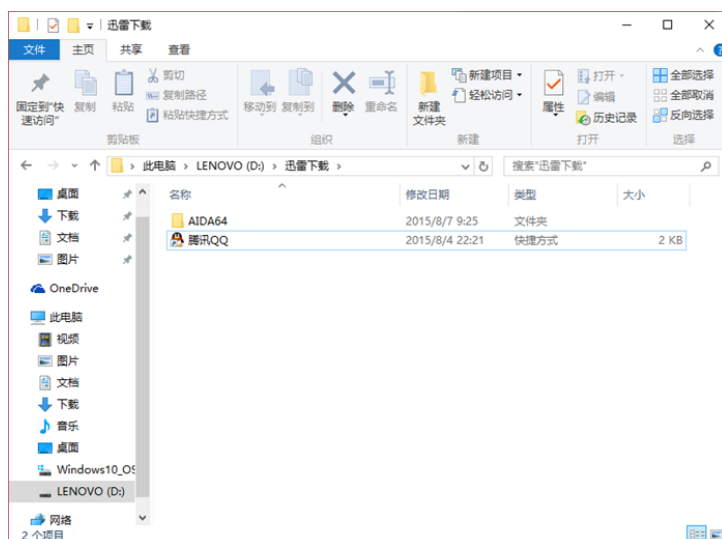



例如: 我想固定截图工具那么就把截图工具, 直接拖拽到开始菜单右侧的磁贴区域, 松手即可固定。其他应用以此类推。

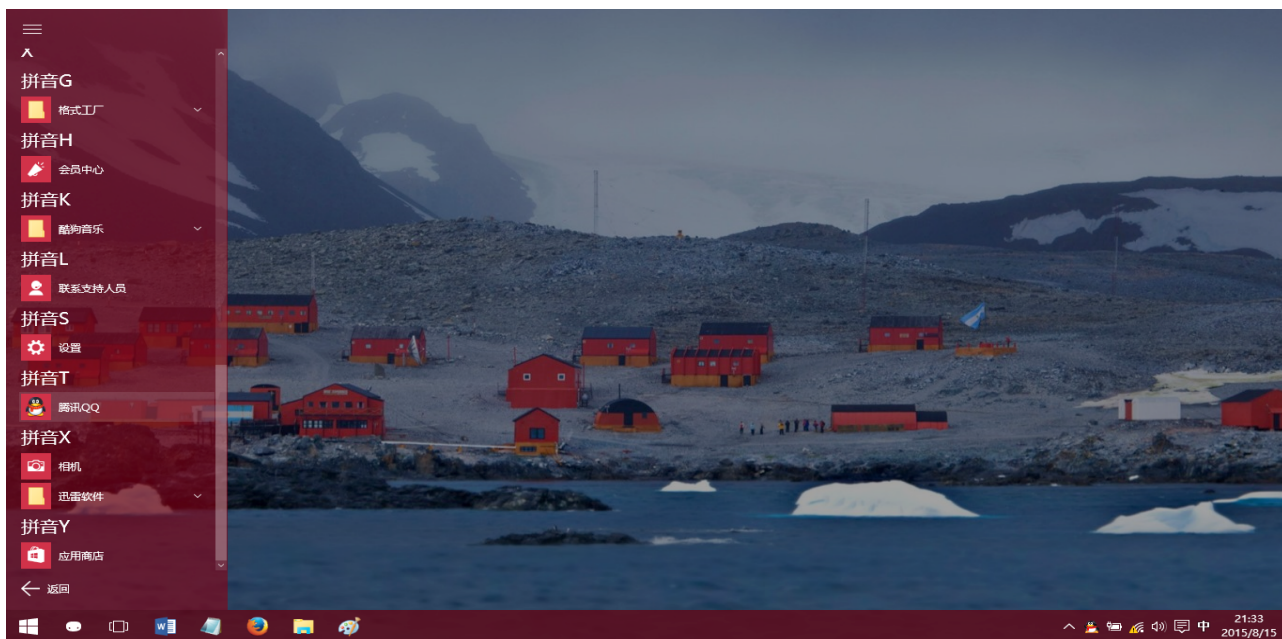
如果欲固定项目不在左侧列表, 可以执行这个操作:

首先找到欲固定到开始磁贴的程序, 并创建快捷方式。这里以“腾讯 QQ”为例:

然后复制这个链接到地址栏, 打开开始菜单文件夹: C:\ProgramData\Microsoft\Windows\Start Menu\Programs。复制欲固定开始菜单磁贴的项目。如果提示“目标文件夹访问被拒绝”, 点击继续即可。



按全屏开始下  的，进入所有应用列表：

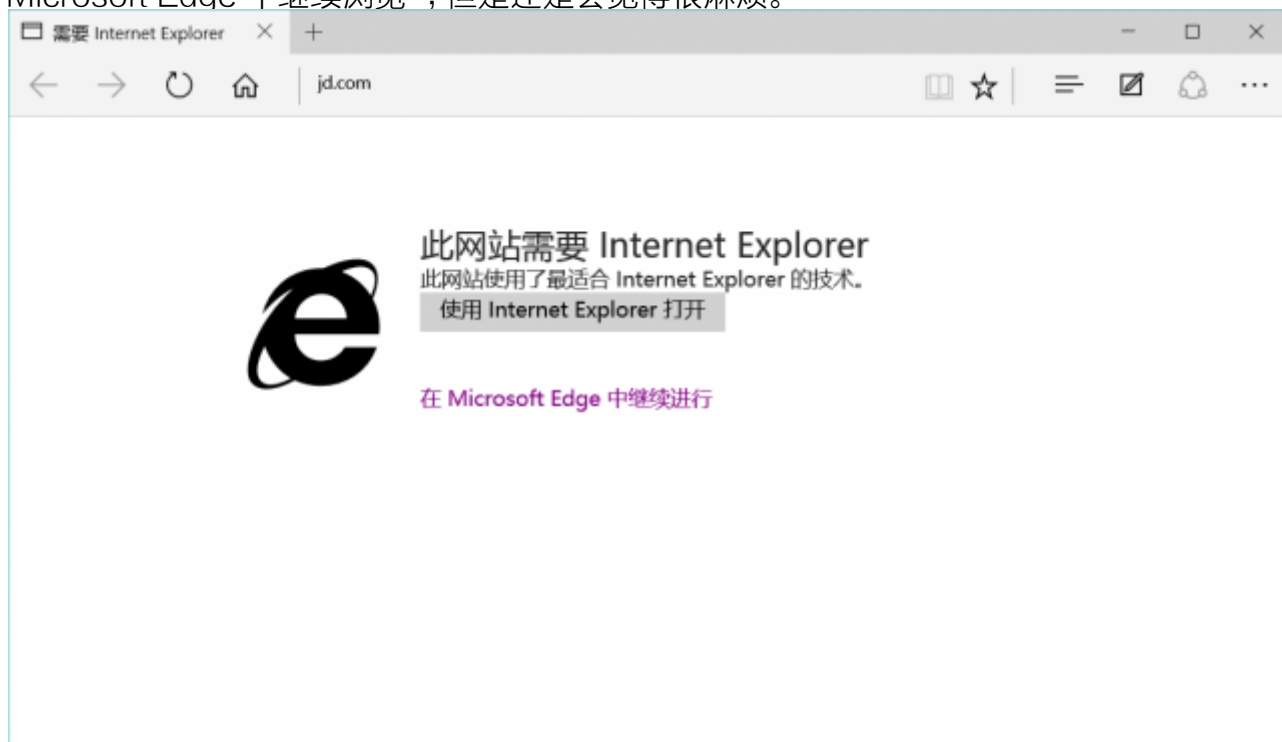


找到腾讯 QQ 的快捷方式，如前文所述，固定到开始即可。

(感谢 lzj20021216 提供图文，链接：<http://bbs.pcbeta.com/viewthread-1617130-1-1.html>)

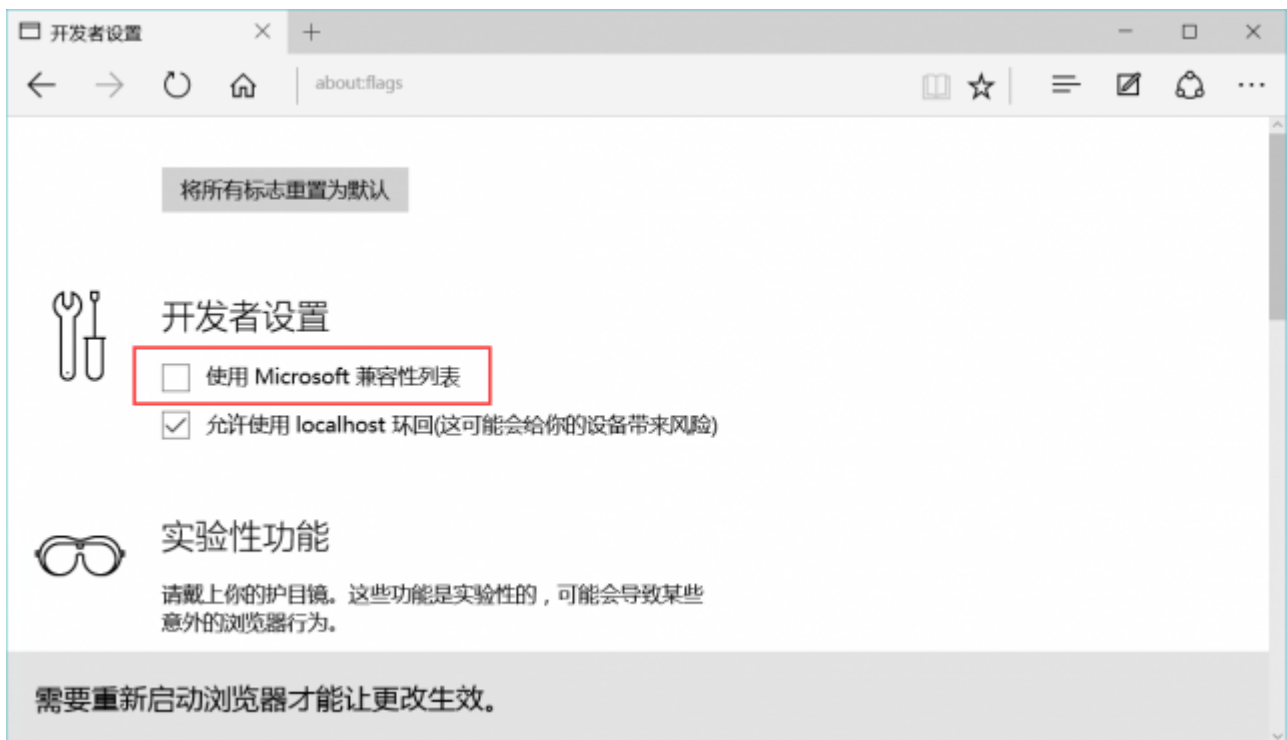
取消 Microsoft Edge 打开某些网站显示“此网站需要 Internet Explorer”的提示

众所周知，Microsoft Edge 在浏览国内绝大部分购物网站的时候，都会被微软标记为不兼容，于是就会变成“此网站需要 Internet Explorer”的提示，虽然可以手动点击“在 Microsoft Edge 中继续浏览”，但是还是会觉得很麻烦。



解决方法如下：

1. 在 Microsoft Edge 的地址栏输入 about:flags 回车，开启开发者设置：



2. 将“使用 Microsoft 兼容性列表”前面的选勾取消。



3. 重启浏览器，使更改生效。

这样做了之后再也不会弹出“此网站需要 Internet Explorer”的提示了。

来源 I451841014: <http://bbs.pcbeta.com/viewthread-1617263-1-1.html>

如何卸载 Windows 10 内置应用

Windows 10 中预装了很多应用,基本可以满足使用需求;但用户想卸载这些应用的时候,要么卸载不干净,要么右键根本就没有提供卸载选项,我该如何卸载应用?

你可以在开始菜单中找到 Powershell,右键点击,选择“以管理员身份运行”,然后进行卸载。这是卸载一些应用的指令:

3D 建模应用:

```
Get-AppxPackage *3d* | Remove-AppxPackage
```

闹钟和时钟:

```
Get-AppxPackage *alarm* | Remove-AppxPackage
```

财经、资讯和天气:

```
Get-AppxPackage *bing* | Remove-AppxPackage
```

计算器:

```
Get-AppxPackage *calc* | Remove-AppxPackage
```

相机:

```
Get-AppxPackage *camera* | Remove-AppxPackage
```

邮件和日历:

```
Get-AppxPackage *communi* | Remove-AppxPackage
```

获取 Office、获取 Skype 和入门:

```
Get-AppxPackage *get* | Remove-AppxPackage
```

OneNote:

```
Get-AppxPackage *onenote* | Remove-AppxPackage
```

人脉:

```
Get-AppxPackage *people* | Remove-AppxPackage
```

手机助手:

```
Get-AppxPackage *phone* | Remove-AppxPackage
```

照片:

```
Get-AppxPackage *photo* | Remove-AppxPackage
```

纸牌游戏:

```
Get-AppxPackage *solit* | Remove-AppxPackage
```

录音机:

```
Get-AppxPackage *soundrec* | Remove-AppxPackage
```

Xbox:

```
Get-AppxPackage *xbox* | Remove-AppxPackage
```

Groove 音乐、电影与电视:

Get-AppxPackage *zune* | Remove-AppxPackage

注意:

1. 小娜和 Edge 等系统级应用无法用此方法卸载。
2. 部分程序卸载时会报错误, 无视即可。
3. 以上列表仅为举例, 你可以提前查看应用包名自行卸载。

(感谢 will08 和 ztsoft 提供应用卸载指令和应用列表)

如何在 Windows 10 中恢复图片查看器

如何在 Windows 10 中恢复图片查看器?

Windows 10 开始, 默认照片应用打开图片。但是无论是开启速度还是操作体验都不可与 Windows 图片查看器同日而语。那么该如何恢复呢? 将下列字符保存为 reg, 导入即可。

```
Windows Registry Editor Version 5.00

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.jpg]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.jpeg]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.gif]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.png]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.bmp]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.tiff]
@="PhotoViewer.FileAssoc.Tiff"

; Change Extension's File Type
[HKEY_CURRENT_USER\Software\Classes\.ico]
@="PhotoViewer.FileAssoc.Tiff"
```

(感谢 wgl836912613, 链接: <http://bbs.pcbeta.com/viewthread-1608743-1-1.html>)

A full-page background image showing a person in a red shirt and black pants climbing a massive, vertically layered rock face. The rock has distinct horizontal strata and some vertical fissures. The climber is positioned on the right side of the frame, appearing small against the scale of the rock.

附录二

Windows 10 开发年表

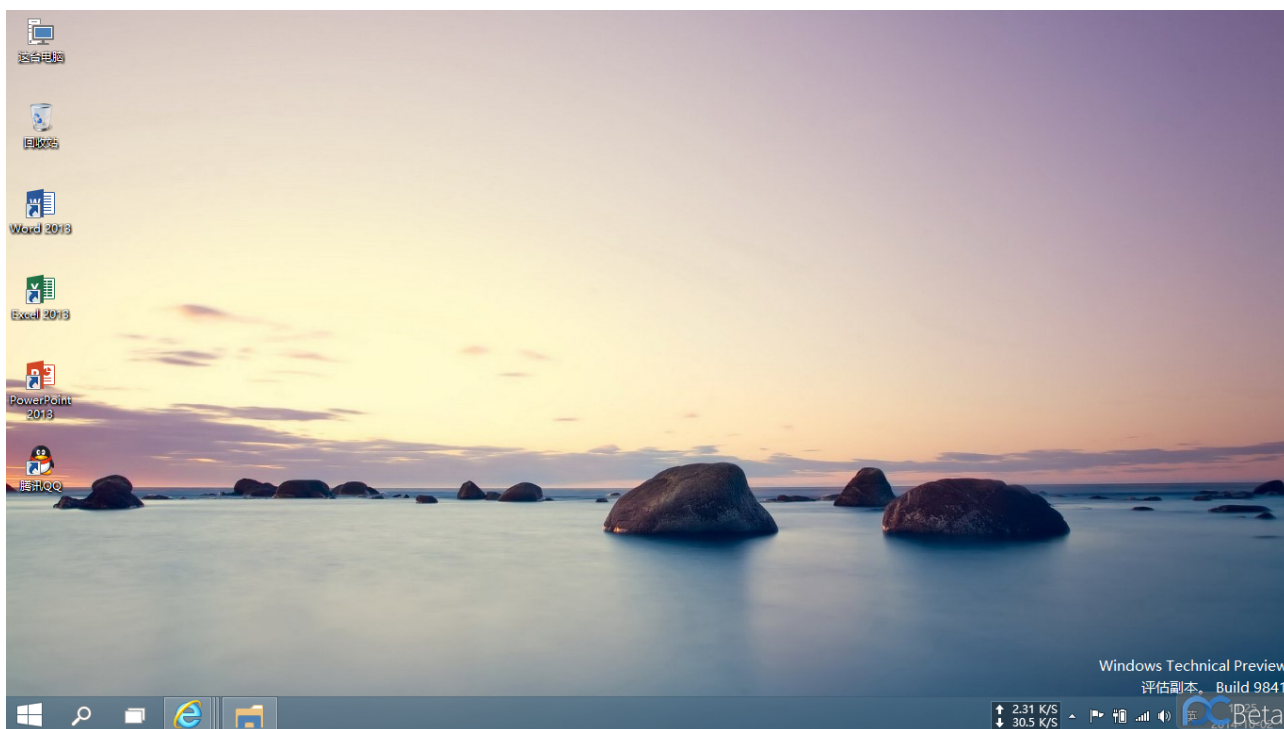
从 2014 年中旬开始传出 Windows 8.1 Update2 (一说为 Windows 9) 的消息, 到 2014 年 10 月 2 日发布第一个公开测试版本, 再到 2015 年 7 月 29 日正式发布 RTM, 开发历时将近一年时间。

(备注: 发布代表官方推送过, 并有 ISO 或 ESD 镜像。未注明来源的图均为主编所截)

2014 年中旬, Windows 8.1 Update2 的消息一度在网络传疯。很多人都认为 Update2 会带回开始菜单, 但实际上这只是一个常规补丁的集成包而已。也有人认为真正的开始菜单回归在下一代 Windows 中才能实现。

Technical Preview 阶段

2014 年 9 月, 网上开始出现 Build 9834 的截图。在这个版本中, 已经出现了早期的开始菜单、操作中心 (Build 9841 一度删除)、快速访问以及 Modern App 窗口化等 Windows 10 新功能。资源管理器的“无边框”设计也在这时候使用。



(图 / 晨星 05361)

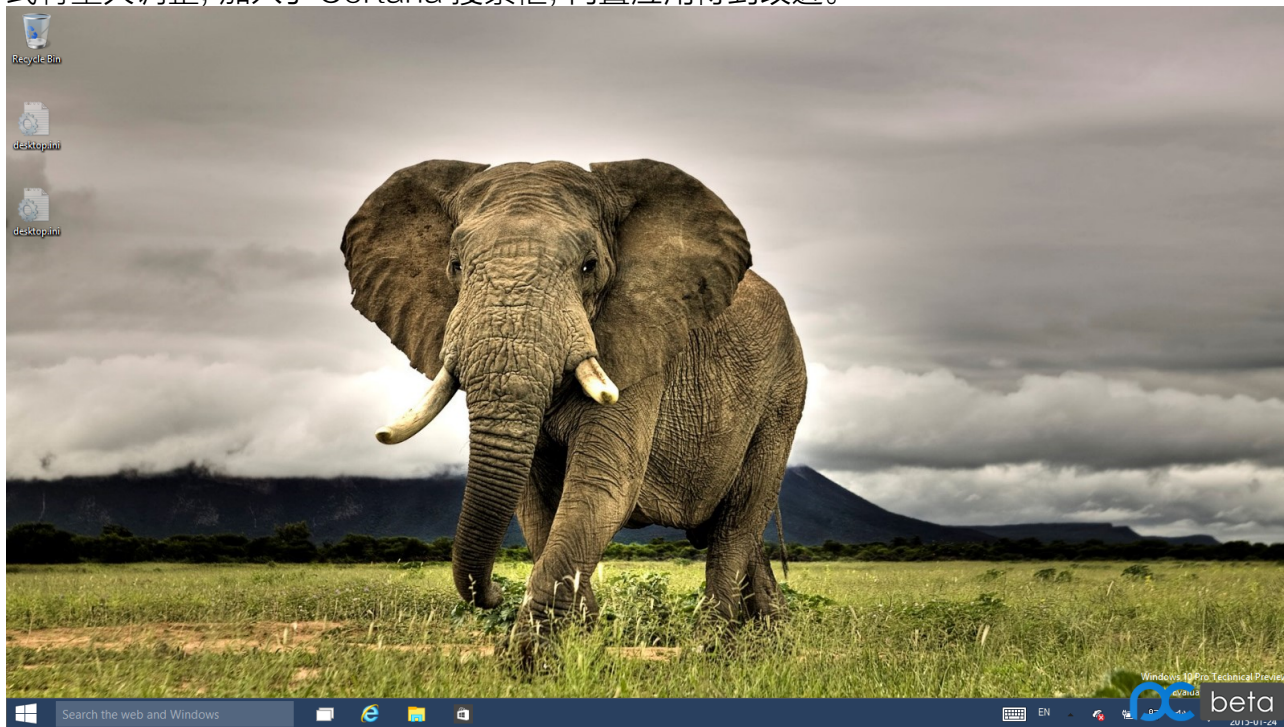
2014 年 10 月 2 日, Build 9841 发布。同时微软宣布下一代 Windows 版本为 Windows 10 (具体可参看微百科“为什么没有 Windows 9”)。Windows Insider 也开放用户注册。

2014 年 10 月 22 日, Build 9860 发布。这个版本重新加入操作中心, 新增了流量感知和节电模式, 并新增了一些窗口动画特效。

2014 年 11 月 14 日, Build 9879 发布。这个版本改变了一些系统图标, 支持固定文件到快速访问, 并且可以隐藏任务栏上的搜索和任务视图按钮。Build 9926 式的开始菜单也开始出现。但这个版本一度导致硬盘丢失。

2014 年 12 月 2 日, Build 9888 泄露。这个版本首次升级内核为 10.0, 右键菜单样式也获得更新。

2014 年 12 月 14 日, Build 9901 泄露。这个版本更换了默认壁纸为雪山风景, 任务栏样式有重大调整, 加入了 Cortana 搜索框, 内置应用得到改进。



(图 /eugenewangtw)

2015 年 1 月 24 日, Build 9926 发布。这个版本首次启用了现在 Windows 10 中的开始菜单样式, 最小化, 最大化和关闭按钮样式得到更新, 也开始出现 Windows 10 的第二套系统图标。

2015 年 3 月 14 日, Build 10036 泄露。这个版本删除了 9841 式菜单和开始屏幕, 新开始菜单支持透明, 网络选择不需要跳转到设置, 也完善了系统图标。

2015 年 3 月 19 日, Build 10041 发布。这个版本支持固定设置到开始菜单, Cortana 加入对中文的支持, 虚拟桌面下可以相互托拽应用程序。

2015 年 3 月 23 日, Build 10014 泄露。这是目前已经公开泄露的含有斯巴达(Edge 前身, 下称 Edge)的第一个版本。

2015 年 3 月 31 日, Build 10049 发布。这个版本是官方第一个出现 Edge 的版本, 同时也修复了开始菜单全屏后会导致桌面内容完全显示的 Bug。

2015 年 4 月 5 日, Build 10051 泄露。这个版本主要是修改了 Modern 应用的文件选取框为 Win32 式。

2015 年 4 月 11 日, Build 10056 泄露。这个版本率先修改了回收站图标, “个性化”选项整合进设置中, 针对平板模式, 可不显示任务栏图标, 开始菜单中的电源移到左下方, 正式启用了 Modern 风格的时钟和日历界面, 暗黑风格设计也在这时候开始启用, Edge 支持 PDF 文档另存为, 加入了邮件和日历的通用应用。

2015 年 4 月 23 日, Build 10061 发布。

2015 年 4 月 30 日, Build 10102 泄露。同 Build 10074 功能基本相同, 但编译时间较早。这个版本主要是 OOBE 的变化以及设置外观的微调, 翻转磁贴特效也于这个版本加入。

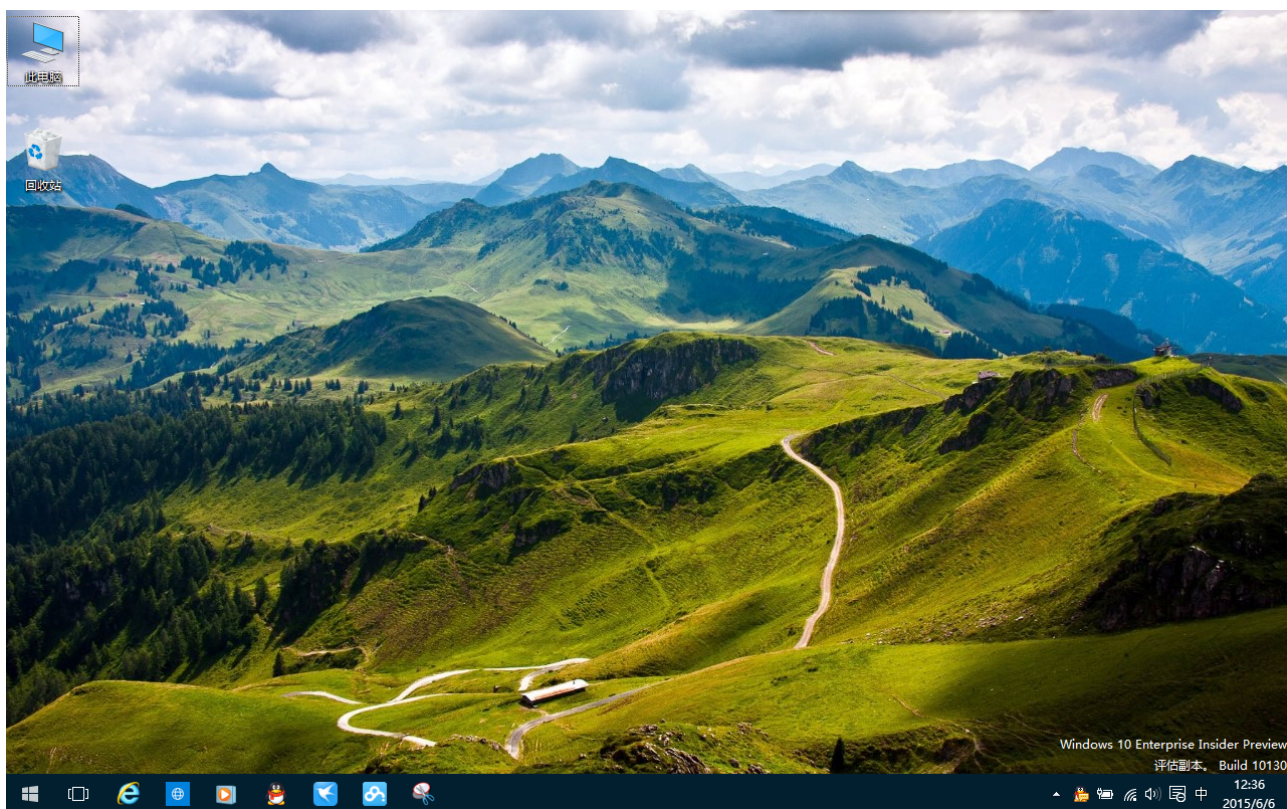


Insider Preview 阶段

2015 年 4 月 30 日, Build 10074 发布。这个版本首次改为“Insider Preview”（内测会员预览版）。这个版本中, 开始菜单的关机按钮已经移动到左下角, 加入了部分 Aero 特效, 也加入了横向音量控制面板。

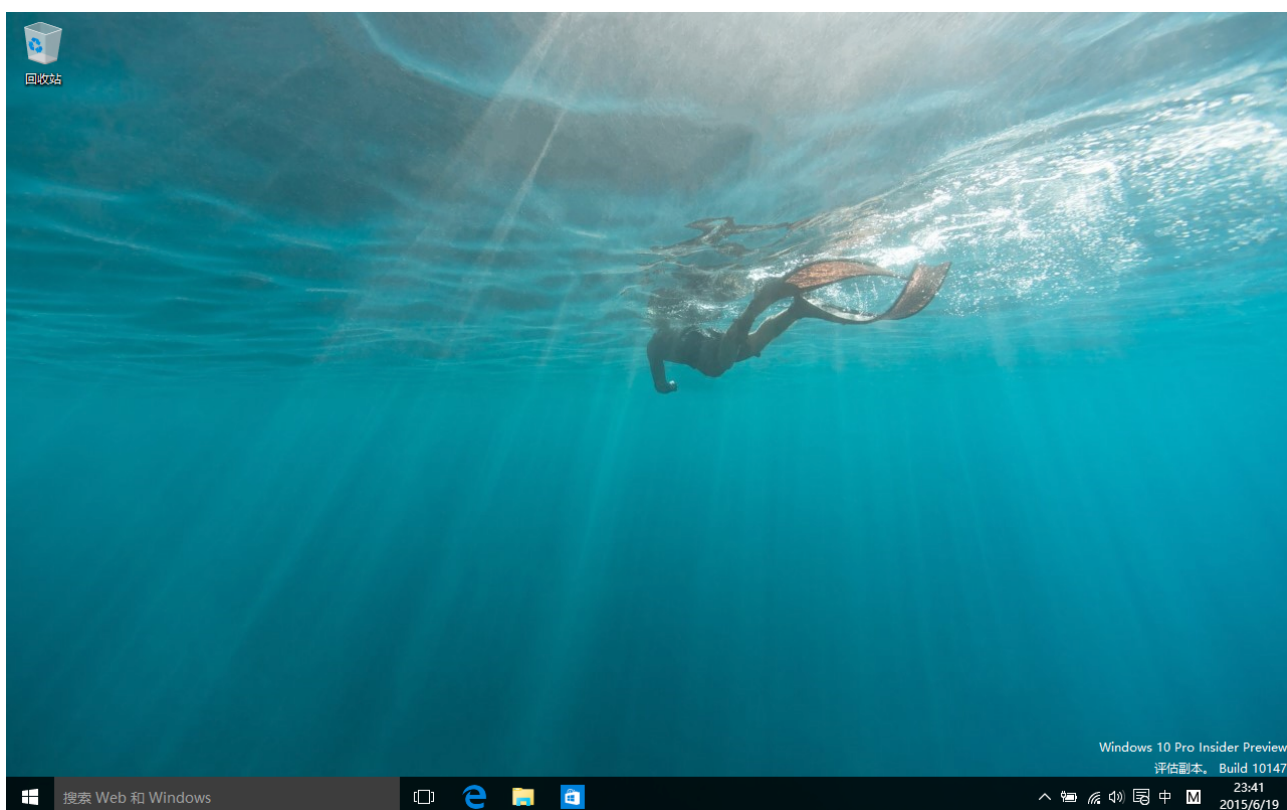
2015 年 5 月 21 日, Build 10122 发布。这个版本的窗口标题栏变成白色, 开始中的全屏按钮被取消Edge拥有了全新的新标签页加入了隐私浏览模式将网站固定到开始菜单的功能、历史记录和全新的阅读模式动画按钮, 标签页中新增了声音图标。

2015 年 5 月 25 日, Build 10125 泄露。这个版本出现了定型的系统图标, 加入了 Windows Hello 功能。



2015 年 5 月 30 日, Build 10130 发布。这个版本可以自定义开始左下角的条目, 改进了任务栏跳转列表的 UI, 使其风格与任务栏保持一致, 开始菜单的快速访问得到优化, 去除了通知中心的边框。这个版本也可以使用原 Charms 快捷键来启动小娜。

2015 年 6 月 5 日, Build 10134 泄露。这个版本加入了截图工具的延时功能。



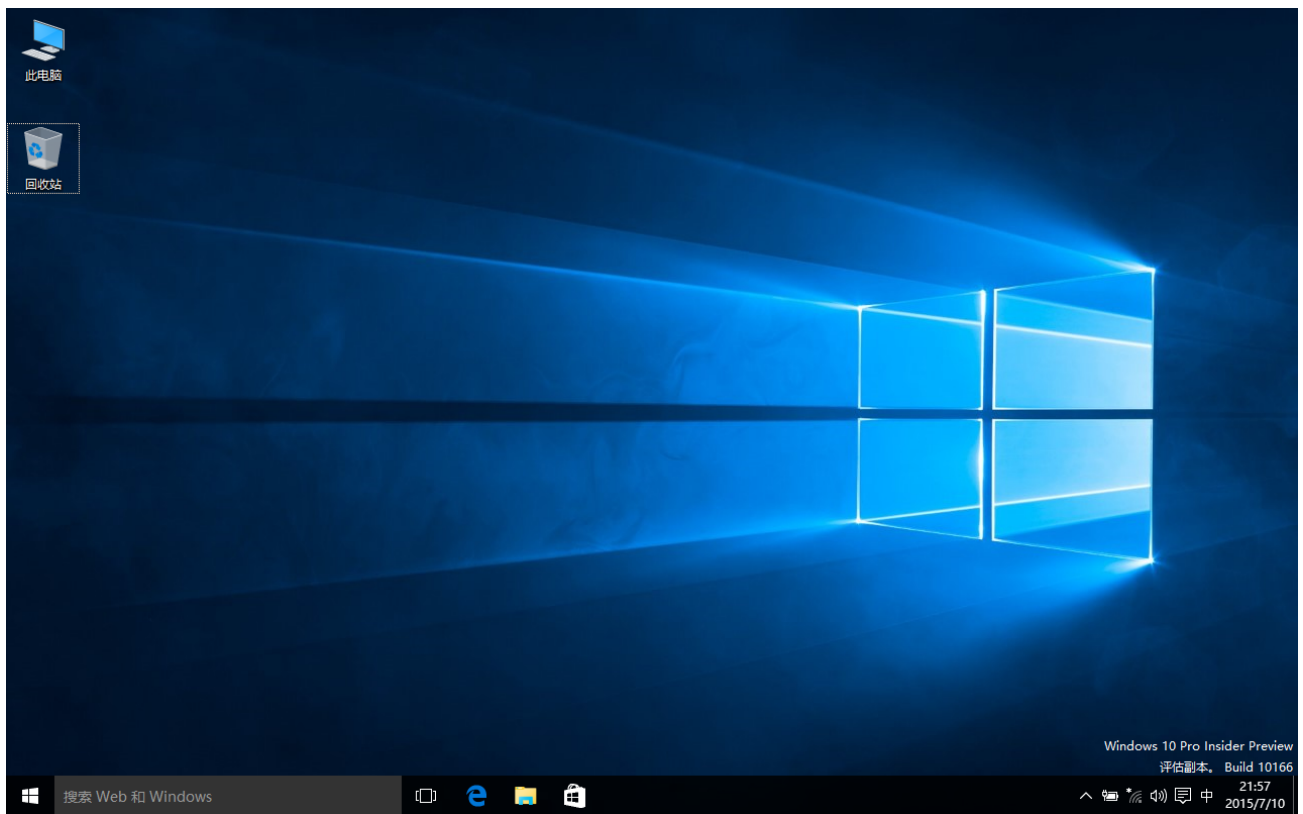
2015 年 6 月 19 日, Build 10147 泄露。这个版本更换了默认壁纸, 全屏开始加入了汉堡菜单, 时钟和网络等替换为暗黑透明外观, Edge 的名字和图标正式启用。



2015 年 6 月 30 日, Build 10158 发布。这个版本更换了默认壁纸, 全屏开始加入了电源和所有应用按钮, 控制面板图标进行了更改, 小娜的外观进行了调整, 会员中心不再随系统自带。

2015 年 7 月 1 日, Build 10159 发布。这个版本启用了“英雄”壁纸和与之配套的登录界面, 修复了近 300 项 Bug。

2015 年 7 月 3 日, Build 10162 发布。这个版本注重于 Bug 修复。



2015 年 7 月 10 日, Build 10166 发布。Groove 品牌正式启用, Edge 浏览器微调, 加入 Microsoft WiFi 应用。这个版本仍注重 Bug 修复。

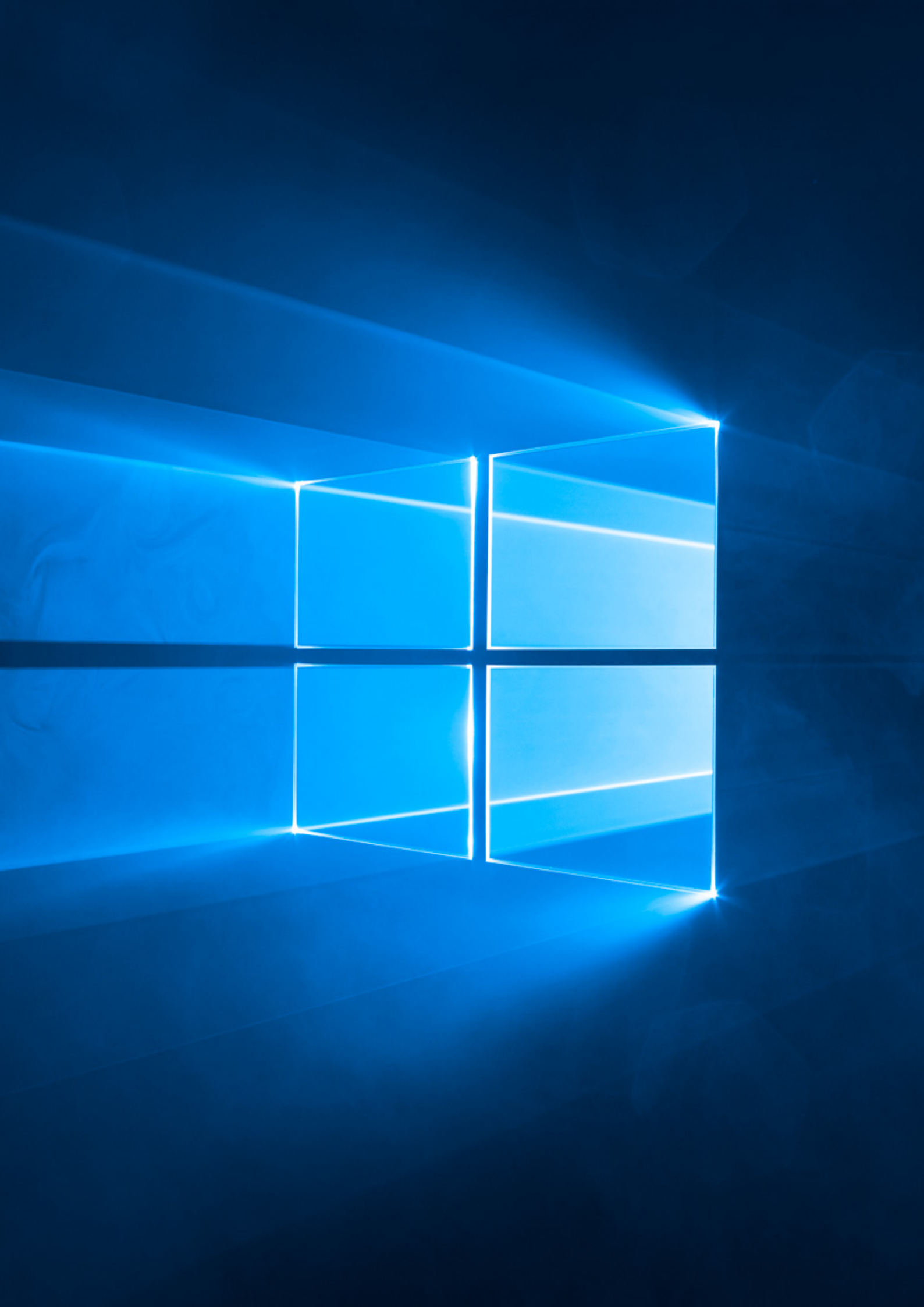
RC 和 RTM 阶段

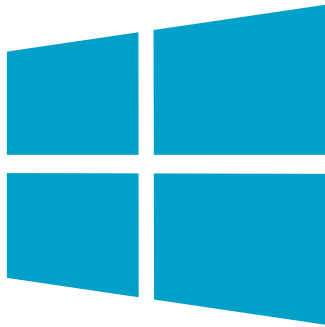
2015 年 7 月 10 日, Build 10176 泄露。这个版本中关于和系统属性徽标改为“Windows 10”, 系统无过期时间。



2015 年 7 月 16 日,
Build 10240 发布。
这是官方推送的首个 RTM 候选版本。

2015 年 7 月 29 日,
微软宣布 Windows 10 正式发布
并面向全球 Windows 7 和
Windows 8.1 用户推送升级,
版本号确定为 Build 10240。





论坛鸣谢



