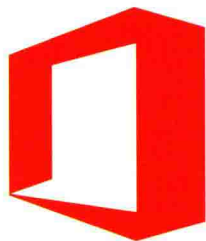




微软官方推荐教材

# Office 365

## 开发入门指南



陈希章◎著



北京大学出版社  
PEKING UNIVERSITY PRESS

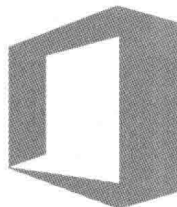


Microsoft

官方推荐教材

# Office 365

## 开发入门指南



陈希章◎著



北京大学出版社  
PEKING UNIVERSITY PRESS

## 内 容 提 要

微软的Office 365是业界知名的生产力平台,自2011年6月28日正式推出以来,已在全世界拥有数以亿计的活跃用户,也为开发者提供了广阔的发展机遇。本书是第一本围绕Office 365开展展开的中文书籍,出自微软高级产品经理之手,不仅全面介绍了Office 365开发的架构,还有丰富翔实的案例,同时也有国内版和国际版的比较,相信对于开发者来说是一本实用的指南。

本书的主要目的是帮助广大Office开发人员实现从传统的、分散的客户端开发体验向Office 365提供的一致的、跨平台与跨设备的体验过渡。此外,相信独立开发商(ISV)的开发团队、项目经理及产品经理也可以从本书中获得一定的启示。

### 图书在版编目(CIP)数据

Office 365开发入门指南 / 陈希章著. —北京: 北京大学出版社, 2018.9  
ISBN 978-7-301-29751-3

I. ①O… II. ①陈… III. ①办公自动化—应用软件—程序设计—指南 IV. ①TP317.1-62

中国版本图书馆CIP数据核字(2018)第171494号

书 名 Office 365 开发入门指南

Office 365 KAIFA RUMEN ZHINAN

著作责任者 陈希章 著

责任编辑 吴晓月

标准书号 ISBN 978-7-301-29751-3

出版发行 北京大学出版社

地 址 北京市海淀区成府路205号 100871

网 址 <http://www.pup.cn> 新浪微博: @北京大学出版社

电子信箱 [pup7@pup.cn](mailto:pup7@pup.cn)

电 话 邮购部 010-62752015 发行部 010-62750672 编辑部 010-62570390

印 刷 者 北京大学印刷厂

经 销 者 新华书店

787毫米×1092毫米 16开本 16印张 450千字

2018年9月第1版 2018年9月第1次印刷

印 数 1—4000册

定 价 69.00 元

---

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究

举报电话: 010-62752024 电子信箱: [fd@pup.pku.edu.cn](mailto:fd@pup.pku.edu.cn)

图书如有印装质量问题,请与出版部联系,电话: 010-62756370

## 从Office 365平台化发展看微软转型

微软（亚洲）互联网工程院副院长、Office 365 产品工程总监 叶盛华

作为第一本Office 365开发方面的中文书籍，本书系统地介绍了Office 365给广大开发人员带来的新的平台和机遇，并且给出了很多实践案例及代码。希章在加入我的团队之前，已经有超过15年的Office平台扩展和开发的实战经验，在开发者技术社区有长期的贡献，而且在一线为很多客户提供过架构设计、解决方案集成方面的服务。这些经历使他所写的文字和内容都比较“接地气”，这是我看了本书后的第一个感受。下面我想结合我的工作经历，谈谈对Office 365产品及其平台化发展的想法，也分享一些微软的转型思维和实践成果。

我在微软工作已超过22年，今年是我受命回国准备Office 365的落地并主持产品工程研发的第10个年头。我的大部分职业生涯都与Office这个有着光荣传统和历史的产品密切相关，我很荣幸成为这个有趣过程的一分子。

在有Office 365之前，我们有着广受欢迎的Office客户端套件，以及功能强大的Office服务器产品群（包括Exchange Server、SharePoint Server和Lync Server等）。5年前，我们迎来了微软的“云优先、移动优先”的大趋势，快速地将Office相关的产品实现SaaS化，这是一个极具挑战的工作，但我们做到了。我们将一个分散的、多服务器上部署的，以及以PC为主的产品有机地组合到一起，在一个强大的共享云平台上实现，通过世界各地高速及强大的数据中心研发出了世界领先的云生产力Office 365智能服务及平台。此外，我们将每一个客户端都进行了移动再造，让用户得到一个真正能在任何地方、任何时候，以及任何平台和设备上都能得心应手地处理文书、回复邮件、和同事沟通及协作的超级应用。就像微软的使命所说的，Office 365赋予每个人一个强大的生产力工具，让大家每一天都能在自己的工作中做得更好。

Office 365正在快速地被全世界的用户接受和使用，每月经由Office 365发出的邮件超过4T，每天的Skype通话超过30亿分钟，每月的会议超过10亿次，文件存储和共享超过470PB。在今天智能为先的时代，面对这些最有价值的用户数据，怎样让广大的开发人员和微软一起打造一个更智能、更广泛的服务呢？对此我们提出了Office 365的平台化发展战略，希望把Office 365建设成一种基础性的设施，不再是简单地做产品，而是希望将Office 365的能力公布出来，让全世界的开发人员都能轻松地调用、享用我们的研究成果。这不仅要求我们进一步审视自己的技术架构，同时还要有很多思维上的革新，如下几个方面。

（1）统一的标准化开放。Office 365是强大的、多元的，我们全面开放所有产品的开发接口，毫无保留。通过给开发者提供一个统一的Graph API标准，让开发人员易学易用。此外，我们提供的接口都是完全支持跨平台的。



(2) 以用户为中心，场景驱动。用户和场景是我们在设计某个功能和某个接口时的重要考虑因素。例如，我们在国内有一些合作伙伴，基于 Office Add-in 把他们的成熟业务（如流程审批、电子签章等）引入 Office 客户端。他们在设计解决方案时，我们会听取反馈，发现有当前功能或接口不符合用户需求和业务场景需要的情况，就把这个作为优先级很高的需求对待。过去我们的产品是每3年更新一次，现在 Office 365 每个月都有近百项新功能发布。

(3) 和开发者一起走进智能时代。我们一直在 Office 的重生再造上下功夫，人工智能的飞速发展是一个新机遇。我们一直致力于通过人工智能技术重新定义我们的服务。例如，你可以用前所未有的方式工作——让 Word 将文档读给你听，让 Excel 自动提出数据见解，让 PowerPoint 提供设计参考，等等；同时，Office 365 会前所未有地了解你，为你提供很多个性化的服务。通过 Office 365 Graph API，我们把大量的数据提供给开发者，希望他们也可以通过大数据及人工智能技术带给用户更智能、更高效的服务。

Office 365 开发平台是强大的，在上面提到的几个关键转型思维的指导下，我们的产品工程团队通过使用同样的平台 API 做了不少本地化的开发尝试。通过这些尝试，我们既亲身体验和测试了自己的 API 的实用性，也给中国用户提供了更“接地气”的服务。下面分别介绍我们在企业和消费者领域及公共事业领域取得的成果。

在2017年11月1日的微软技术暨生态大会上，我们发布了 Office 365 微助理。这是一款基于微信平台打造的 Office 365 个人助理办公套件，让简易熟悉的使用环境和 Office 365 的强大工作场景组合在一起，同时还能有效实现对数据的保护和再利用。通过微助理不仅可以完成邮件收发、日程管理、文档协作、团队沟通等移动办公场景，还可以实现很多有意思的人工智能应用场景，如重要邮件分类、图片识别和处理、翻译、知识库与智能问答机器人等。据不完全统计，在上线后的短短几个月里，有超过 600 家不同规模的企业客户部署了 Office 365 微助理。

在公共事业领域，微软与教育部签署了《中国教育信息化第三期合作战略合作备忘录》，为中国学校提供基于 Office 365 的微软教育云服务。通过多层次的合作，包括“全球创新学院”“长城计划”及“国培计划”等，不遗余力地参与和推动教育信息化的升级。其中，“家校通”这个项目是微软海外产品体系中尚未出现的产品形态，是完全在与本地合作伙伴的共同努力下打造的“中国智造”的成果。这个产品将 Office 365 的云盘、笔记本、团队网站及视频会议功能无缝整合成教育信息化平台，为老师和学生提供了强大的协作、沟通工具，低成本地快速构建可信赖的数字化课堂，在极端天气的情况下实现停课不停学，甚至为偏远山村的儿童实现了远程教学应用。

以上提到的两个应用场景是 Office 365 平台化能力的一个有力佐证。使用希章在书中提到的 Microsoft Graph 和人工智能的技术，也可以开发出类似的应用。Office 365 产品工程团队将一如既往地支持越来越多的中国合作伙伴和开发人员更好地利用 Office 365 的平台能力，一方面解决好、服务好中国的客户和用户，另一方面也将这些好的创意和成果推广到全世界。

# 从《淮南子》谈Office 365的赋能之道

微软大中华区Office 365产品市场总经理 郑弘亦

《淮南子》开篇有一段关于“道”的描述：“夫道者，覆天载地，廓四方，析八极，高不可际，深不可测。”在我看来，“道”应该是世界观和方法论的统一，贯穿于万事万物之中。

Office 365的世界观是赋能于员工，打造更智能的生产力，而对应的方法论就是把针对性的办公模块（如办公“三剑客”——Word、Excel、PowerPoint）、内容管理SharePoint、远程沟通Skype for Business、网盘OneDrive、企业邮件Outlook及团队管理Teams等进行有机整合。这种“有机性”已经接近一个自然的人体：各个模块就像人体中职能不同的器官，统一接口的Microsoft Graph就像神经中枢，Office Add-in等四大开发方向就像促进人体生长发育的组织腺体。Office 365是人类历史上很好地体现整体性的一款SaaS应用。

赋能于人类，就是Office 365的“道”。

《淮南子》有云：“置猿槛中，则与豚同，非不巧捷也，无所肆其能也。”意思是把猿类关在笼子里，它就会像猪一样，并不是它没有灵巧攀登、轻捷跳跃的本领，而是它没有发挥本领的机会。

在我看来，这个猿类也应该包括“程序猿”——21世纪从事IT开发的一类人群。而这个笼子很大程度上并不是客观限制，而是“程序猿”对Office 365的认知深度。有的开发人员只是蜻蜓点水，满足于常规功能的维护与管理，不但令企业数字化转型的投入大打折扣，而且自己的能力提升也有限；有的人则可以从业务需求反馈中通过Office 365挖掘出更多的生产力，塑造各类更高效的办公场景，自身功力也得以精进。

希章集20年功力创作的这本《Office 365开发入门指南》，对于“程序猿”打破脑中的笼子、深度拓展Office 365的世界观，在学习和实践中提高开发能力，一定大有裨益！



## 人工智能时代生产力的新篇章

微软（中国）有限公司现代办公“专业技术解决方案专家”团队总监 刘浩

自2011年6月28日微软发布Office 365至今，我们团队Office领域的技术专家亲眼目睹“她”一天天成长，直至成为当今世界上最大的SaaS云服务平台。截至2017年6月，Office 365的收益已经全面超越了传统的本地部署版的Office套件。

在“她”成长期间，我们不仅感受到了Office 365的日新月异，同时也切身体会到微软文化转型及组织转型带来的巨大能量，可以说文化的转型和Office 365是相辅相成的。目前，全球500强中近90%的企业都在使用Office 365，个人订阅用户已经达到1亿，这些变化使Office 365成为一个巨大的数据金矿，等待着全球的独立软件开发人员来挖掘。这是一个巨大的“风口”。

当我们意识到这座“金矿”的存在后，接下来的难题就是如何去“采矿”了。我们需要对“采矿”的方式方法及对“矿场”的结构进行必要的解析。

陈希章创作的这本指南由浅入深，不仅告诉大家如何去开发Office 365中的组件，还对每个组件的业务价值做了相应的说明。书中的例子都经过了陈希章的反复推敲与实践验证，指南中也针对中国由世纪互联(21V)运营的Office 365数据中心做了较多的说明和优化解析。同时，由于Office Graph涵盖和整合了之前所有的服务，所以在此书的编写过程中，有些预览版的API经常会有一些调整。在此期间，陈希章经历了很多的磕磕绊绊。此书的创作是一项了不起的工作，可以让开发人员少走很多弯路。

特别值得注意的是，指南的最后简单地讲述了Office 365的发展趋势之一——智能化。这也是微软未来的三大主题(混合现实、人工智能和量子计算)之一。现在都在谈万物互联(IoT)，殊不知万物互联发展到最后，如何进行“人机互动”就成了新的难题。例如：IoT得到的数据如何分析、计算，并展现给相应角色的人员？当数据发生问题时，该角色的运营人员如何快速地与另外一组人员进行沟通互动？因此，对万物互联(IoT)的理解还应该加上“人”这一核心要素，我给它的定义为IoPeople。我们是否可以用这样的模型来定义一个企业的智能程度： $A.I./ (IoT + IoP)$ 。A.I.是指一个企业全部的A.I.能力，相信随着量子计算的实现，这一能力将有质的飞跃。IoT+IoP是一个企业所有“物”的数量集，这样就可以得到每一个“物”所具有的智能指数值。相信在未来，我们会看到最新版的混合现实(MR)带来更多令人振奋的人机互动方式。

微软的Office 365是业界知名的生产力平台，自2011年6月28日正式推出以来，已在全世界拥有数以亿计的活跃用户，帮助用户实现现代化办公，也为广大的开发者提供了广阔的发展机遇。用户可以利用这个平台所提供的能力，快速构建“云优先、移动优先”的应用，以全新的方式分发给全世界的用户。

Office 365是在2014年4月正式进入中国市场的，而且中国拥有一个由世纪互联独立运营的特殊版本。对于广大的中国用户和开发人员来说，这是一个很好的机遇。我作为一个从传统的Office定制开发一路走来的“老兵”，同时又在微软Office 365产品组担任产品经理的角色，有机会较系统地了解产品和平台发展的内在设计和规律，希望将这些经验分享给大家，让更多的人受益。

## 为什么要写这样一本书

我是中国区最早的Office平台开发方向的MVP（微软最有价值专家），曾经获得Excel MVP和SharePoint MVP，所以对于Office的开发和定制比较熟悉，对它提供的强大能力也深有体会。在过去的十多年里，我写过很多这方面的技术文章，并分享在技术社区。

2017年2月上旬，我从西雅图总部学习交流回来，有感于Office 365在全球范围内的蓬勃发展，以及它给开发人员带来的全新动力和机遇。我接触了一些独立开发商和开发人员，他们的兴奋之情溢于言表；在与总部产品团队交流时，我也了解到Office 365在加速向平台化完善，不止步于提供强大的协作和沟通功能，更重要的是还要让合作伙伴在该平台基础上定制、集成自己的应用和解决方案，实现更大的价值。

从2017年2月开始动笔后，我最先在博客上发布了Office 365开发系列文章，期间陆续收到了不少反馈，也帮助我将写作的想法逐步完善起来。截至2017年12月，我共写作39篇，总计约10万字。这些文章经过多个微信公众号转载，被来自24个国家的上万名读者阅读分享，也更坚定了我创作本书的决心。

写作Office 365开发入门系列的文章前后花费了十个多月的时间，这也正好是Office 365平台化作用日趋明显的一段时间，尤其是Microsoft Graph的功能不断升级和更新，全球范围内有越来越多的成功案例。所以要写这样一本技术性的书籍并没有想象得那么容易，因为它本身是动态变化的——事实上，在我决定出书后，有部分内容在回头审核时就发现已经过时，以至于需要重写——但这也正是我想表达的一种谢意，这是包括微软的产品组、市场和销售，以及服务各条战线共同努力的结果，也是越来越多的合作伙伴、开发人员共同参与的结果：Office 365的开发与很多其他平台的开发一样，蕴含价值、富有活力。虽然这是第一本以中文写作的Office 365开发书籍，但我相信这只是一个开始。

## 本书内容和读者对象

本书共分为6章，第1章回顾Office平台开发的技术和场景，并引出Office 365开发的4个核心方向；第2章着重介绍基于Microsoft Graph的开发流程和案例；第3章详细介绍全新的Office Web Add-





in的架构和开发生命周期；第4章围绕SharePoint Online的开发技术进行探讨；第5章展示Office 365开发的新领域和快速开发面向主题的商业应用程序实践；第6章分析Office 365现有的人工智能技术，详细讲解基于Office 365开发智能服务机器人的过程。

写作本书的主要目的在于帮助广大的Office开发人员（包括Office客户端开发人员和SharePoint服务器端开发人员）实现从传统的、分散的客户端开发体验向Office 365提供的一致、跨平台与跨设备的体验过渡。如果读者已经有Office开发的经验（包括VBA和VSTO），通过本书将了解到新的平台（Office 365）及其带来的新机遇。Web Add-in通过主流的Web技术实现，一方面可以让应用更易于分发和更新，另外一方面也可以让开发人员的开发技能得到进一步扩展，建议先着重阅读第1章和第3章。如果读者已经有SharePoint开发的经验，通过本书将了解到SharePoint Online与本地版本的SharePoint Server在开发模式上的差异。通过阅读第1章和第4章，读者将深入了解SharePoint Add-in和SharePoint Framework在设计上的考虑和具体应用场景。

同时，独立开发商（ISV）的开发团队、项目经理及产品经理也可以从本书中获得一定的启示，因为Office 365提供了一套强大的接口（Microsoft Graph），可以通过这种新的技术将Office 365的能力集成到自己的解决方案中，为客户提供更多独特的价值。这些能力既包括Office 365标准的功能，如邮件、个人网盘、文档协作、联系人管理、会议室和日程管理等，也包括基于Office 365的大量数据来实现人工智能的能力。本书的第1章、第2章及第5章、第6章特别适合此类读者阅读。

## 鸣谢

我在初步决定要将这些在网上发表过的文章集结成书后，郑重地邀请了公司同事、行业内好友，以及技术社区的朋友进行审稿。在此过程中，我收到了很多中肯的意见，这些意见涉及内容、行文风格、可读性等方面，还有部分好友专门为我撰写了书评和推荐语。由于篇幅有限，这些文字未能全部出现在本书中，但这些反馈已经为我日后创作其他文章或书籍埋下了种子。在此一并表示感谢。

最后，我要感谢所有正在阅读本书的读者，衷心希望这本书可以为读者打开通往Office 365平台的大门。在创作过程中，我竭尽所能地为读者呈现最新、最实用的功能和内容，但仍难免有疏漏和不妥之处，敬请广大读者指正，也欢迎大家多提宝贵意见（请通过 [office365devguide@xizhang.com](mailto:office365devguide@xizhang.com) 邮箱与我联系）。

# 目 录

## 第1章 Office 365开发概述及

### 生态环境介绍 ..... 2

#### 1.1 回顾Office开发的基本情况 ..... 2

##### 1.1.1 Office XP ..... 2

##### 1.1.2 Office 2003 ..... 2

##### 1.1.3 Office 2007 ..... 3

##### 1.1.4 Office 2013 ..... 4

##### 1.1.5 Office 365横空出世 ..... 4

#### 1.2 Office 365开发概述 ..... 8

##### 1.2.1 Office 365是什么 ..... 8

##### 1.2.2 Office 365的开发场景 ..... 10

#### 1.3 Office 365“生态环境”介绍 ..... 14

#### 1.4 搭建Office 365开发环境 ..... 15

##### 1.4.1 申请Office 365一年免费的 开发者账号 ..... 15

##### 1.4.2 客户端开发环境介绍 ..... 17

## 第2章 Microsoft Graph开发 ..... 20

#### 2.1 Microsoft Graph 概述 ..... 20

#### 2.2 通过工具快速体验Microsoft Graph ..... 22

##### 2.2.1 通过Graph 浏览器体验 Microsoft Graph ..... 22

##### 2.2.2 通过tokenPlease( )函数获取 access token ..... 26

#### 2.3 应用程序注册 ..... 28

##### 2.3.1 注册Azure AD应用程序 ..... 28

##### 2.3.2 注册Azure AD 2.0应用程序 ..... 32

##### 2.3.3 中国版Office 365应用程序注册 ..... 35

##### 2.3.4 通过PowerShell创建应用程序 并定义服务和权限声明 ..... 37

#### 2.4 Microsoft Graph应用程序开发实战 ..... 40

##### 2.4.1 Microsoft Graph桌面应用程序 ..... 40

##### 2.4.2 在PowerShell脚本中集成 Microsoft Graph ..... 49

##### 2.4.3 Microsoft Graph Web应用 程序极致开发体验 ..... 53

##### 2.4.4 在无人值守程序(服务)中调用 Microsoft Graph ..... 60

#### 2.4.5 跨平台应用集成(在ASP.NET

Core MVC 应用程序中  
集成 Microsoft Graph) ..... 69

#### 2.4.6 扩展Microsoft Graph 数据结构(开放扩展) ..... 84

## 第3章 Office Add-in开发 ..... 90

#### 3.1 Office Add-in 架构和入门 ..... 90

##### 3.1.1 Office Add-in 开发概述 ..... 90

##### 3.1.2 Web Add-in技术架构 ..... 91

##### 3.1.3 Office Add-in 能做什么 ..... 92

##### 3.1.4 Office Add-in 概览 ..... 92

#### 3.2 Office Add-in开发实践 ..... 94

##### 3.2.1 在Visual Studio 中开发 Office Add-in ..... 94

##### 3.2.2 在Visual Studio Code中开发 Office Add-in ..... 100

##### 3.2.3 实战Excel Add-in的三种玩法 ..... 105

##### 3.2.4 详解Office Add-in 清单文件 ..... 109

##### 3.2.5 在企业和应用市场发布 Office Add-in ..... 114

#### 3.3 Office Add-in的技术原理和 常见问题剖析 ..... 117

## 第4章 SharePoint Online开发 ..... 124

#### 4.1 SharePoint 大局观 ..... 124

##### 4.1.1 SharePoint 向云迁移的 趋势和规律 ..... 124

##### 4.1.2 SharePoint Server 和 SharePoint Online的分工 ..... 126

##### 4.1.3 SharePoint和OneDrive for Business的分工 ..... 127

##### 4.1.4 SharePoint 在用户体验 方面的改进 ..... 129

##### 4.1.5 开发模式的变化 ..... 132

#### 4.2 SharePoint Online Add-in 开发简介 ..... 133

##### 4.2.1 SharePoint Add-in开发概述 ..... 133

##### 4.2.2 安装开发环境 ..... 134

##### 4.2.3 创建SharePoint Developer Site(开发者站点) ..... 134

4.2.4 创建SharePoint App Catalog Site (应用程序目录站点) .....	135	5.2.4 在PowerApps中操作引发的流程....	191
4.2.5 创建、测试和部署SharePoint- hosted Add-in.....	137	5.2.5 通过Power BI 警报引发的流程.....	192
4.2.6 创建、测试和部署Provider- hosted Add-in.....	143	5.3 Common Data Service (CDS) 初探 .....	194
4.3 SharePoint Framework 开发简介 .....	148	5.3.1 创建和管理数据库.....	195
4.3.1 SharePoint Framework的 主要特性 .....	149	5.3.2 在Excel中编辑实体数据 .....	196
4.3.2 SharePoint Framework能做什么.....	149	5.3.3 在Outlook中集成 Common Data Service .....	197
4.3.3 准备SharePoint Framework的 开发环境 .....	149	5.3.4 在PowerApps中使用 Common Data Service .....	199
4.3.4 开发和调试一个简单的 客户端WebPart.....	150	5.3.5 在Microsoft Flow 中使用 Common Data Service .....	201
4.3.5 部署SPFx WebPart.....	153	5.4 为PowerApps、Flow及Power BI 开发自定义连接器.....	202
<b>第5章 基于Office 365的按需应变 业务应用平台</b> .....	159	5.4.1 编写一个Web API服务 (适用于PowerApps和Flow) .....	204
5.1 使用PowerApps快速构建基于 主题的轻业务应用.....	161	5.4.2 在Flow中创建自定义连接器.....	206
5.1.1 先决条件 .....	161	5.4.3 在Flow中使用自定义连接器.....	209
5.1.2 基于一个保存在OneDrive for Business个人 网盘中的Excel文件创建业务应用 .....	162	5.4.4 在PowerApps中使用 自定义连接器 .....	212
5.1.3 基于SharePoint Online的 列表创建轻业务应用 .....	168	5.4.5 Power BI自定义连接器的 开发思路 .....	213
5.1.4 基于Dynamics 365创建 自定义应用.....	172	<b>第6章 人工智能背景下的Office 365 现状和发展趋势</b> .....	215
5.1.5 将PowerApps应用集成到 Microsoft Teams中.....	173	6.1 微软人工智能, 增强人类智慧.....	215
5.1.6 进阶话题 .....	175	6.2 Office 365已经具有的AI能力 .....	217
5.2 Microsoft Flow 概述 .....	183	6.3 基于Office 365的人工智能 发展方向与机遇.....	230
5.2.1 通过Microsoft Flow实现将特定邮件的 附件自动保存到SharePoint Online 文档库中 .....	184	6.4 Office 365机器人 ( Bot ) 开发入门指南 .....	232
5.2.2 实现周期性执行的流程.....	188	6.4.1 3种类型的Bot.....	232
5.2.3 实现用户手工启动的流程 .....	189	6.4.2 3种常见的Azure机器人 服务方案 .....	233
		6.4.3 Function Bot 开发和调试.....	234

# 第1章

微软的Office 365是业界知名的生产力平台，自2011年6月28日正式推出以来，已在全世界拥有数以亿计的活跃用户。它不仅帮助用户实现了现代化办公，更为广大的开发者提供了广阔的发展机遇，开发者可以利用这个平台所提供的能力，快速构建“云优先、移动优先”应用，以全新的方式分发给全世界的用户。本章将介绍以下内容。

1. 回顾Office开发的基本情况。
2. Office 365开发概述。
3. Office 365“生态环境”介绍。
4. 搭建Office 365开发环境。

# 第 1 章 Office 365 开发概述及生态环境介绍

## 1.1 回顾 Office 开发的基本情况

我几乎使用过 Office 97 之后所有的 Office 版本，印象最深刻的是以下几个版本。

- (1) Office XP。
- (2) Office 2003。
- (3) Office 2007。
- (4) Office 2013。
- (5) Office 365。

### 1.1.1 Office XP

Office XP 没有用年份来编号(理论上应该是 Office 2002)，可能是为了配合 Windows XP 的整体市场宣传定位，它的特殊之处在于有一个所谓的开发版(Office 2000 也有开发版，但 Office XP 的这个版本更加完善)。值得一提的是，虽然同样带有 XP 的光环，但 Office XP (如图 1-1 所示)远没有 Windows XP 那么风光(后者“服役”超过 13 年，直到现在还有用户对其念念不忘)，它很快就被 Office 2003 取代了。



图 1-1

### 1.1.2 Office 2003

Office 2003 是一个非常重要的版本，它代表着 Office 产品技术的巅峰时代——这个版本的 Office 功能非常强大，可以说是无所不能。如果说 Office XP 是我用得比较全的一个版本——除了 Outlook 外，其他组件基本都用过，还用 FrontPage 做出了人生中第一个网站——那么 Office 2003 就是我真正意义上开始较为深入地使用的版本，尤其是 Excel 和 Access 这两个组件，结合当时的工作需要，我使用 VBA 开发





了从简单到复杂的各种小应用。

我在学习Excel的VBA时是非常认真的，一个佐证就是我那时愿意花50美元托人从国外辗转买来一本足有1000多页的书(如图1-2所示)。这本书及其作者John Walkenbach对我的影响之大，很难用一两句话讲清楚。在那个相对单纯的年代，我一头扎进Excel VBA的世界里，收获的可不仅是写代码带来的乐趣，还有在微软技术社区(那时称“新闻组”)认识的很多朋友。

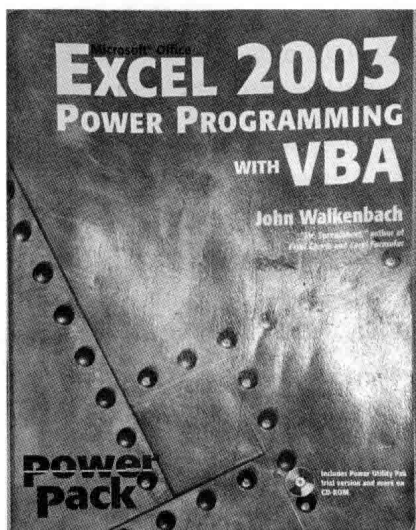


图 1-2

### 1.1.3 Office 2007

前面提到Office 2003是一个巅峰之作，那么Office 2007无疑是一个转型精品。表面上看，Office 2007带来了全新的UI风格——Ribbon，这是一次大胆的尝试，因为Office 2003的菜单已经非常多了，以至于很多新手经常找不到功能所对应的位置。这种界面的创新带有一定的冒险性(颠覆成熟的产品确实需要勇气)，但事实证明非常成功。

图1-3所示为Word 2007的界面，除了界面上可见的变化外，Office 2007的另外一个重要创新是重新定义了Office文档的格式。除了继续支持Office 2003及早期版本的二进制文件格式外，还有一种全新的、基于XML的文件格式(通常在默认的文件扩展名后面添加一个x以示区分，如Word 2003的格式是.doc，而Word 2007虽然支持.doc，但更推荐用户使用.docx文件格式)。这个格式后来被正式命名为OpenXML技术，微软在经过实践后将其贡献给ECMA，并被ISO和IEC等组织认定为开发文档格式的国际标准。

在开发层面，Office 2007也有新的变化。虽然它依然支持VBA，但却规定所有包含代码的文件与不包含代码的文件从文件格式上要有明确区分。例如，Excel 2007的标准文件格式为.xlsx，而包含VBA代码的文件则必须重命名为.xlsm(这里的m是指macro)。此外，它开始支持使用Visual Studio 2005及.NET Framework，并对其进行开发定制，这就引出了一个全新的开发工具——VSTO(Visual Studio Tools for Office)，此传统一直沿用至今。

针对.NET开发人员，微软还专门提供了OpenXML SDK，支持从自定义程序中通过OpenXML的标准操作Office文档(不要求本地安装Office)。

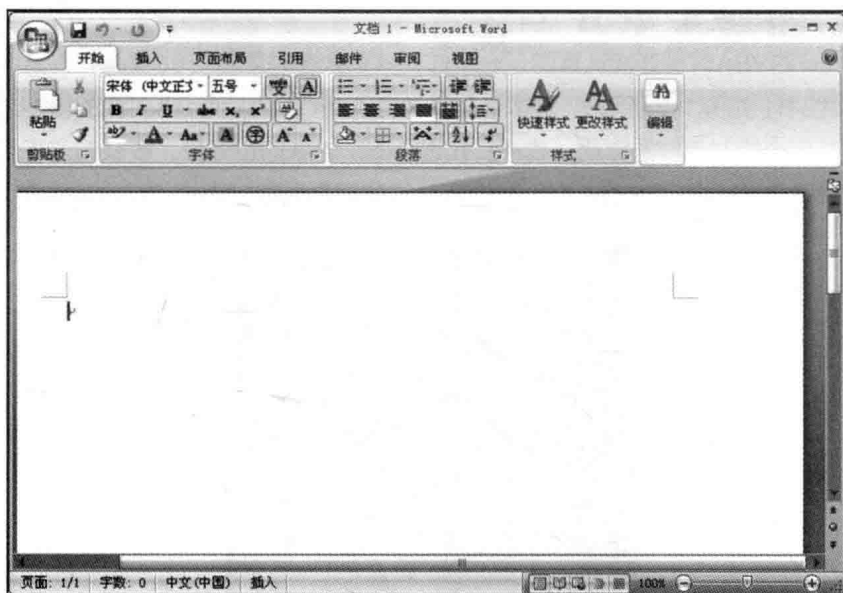


图 1-3

### 1.1.4 Office 2013

Office 2013有很多重要的创新，如增强了与云端服务整合的能力、跨平台和设备的能力，以及协同编辑的能力等，还有一个对开发人员来说至关重要的App开发模式，这个模式涵盖了客户端和服务端及云端完整的产品线。这从根本上解决了开发人员部署应用程序的困扰，同时，它将通过Office Store建立一个全新的生态环境。相关的架构如图1-4所示。

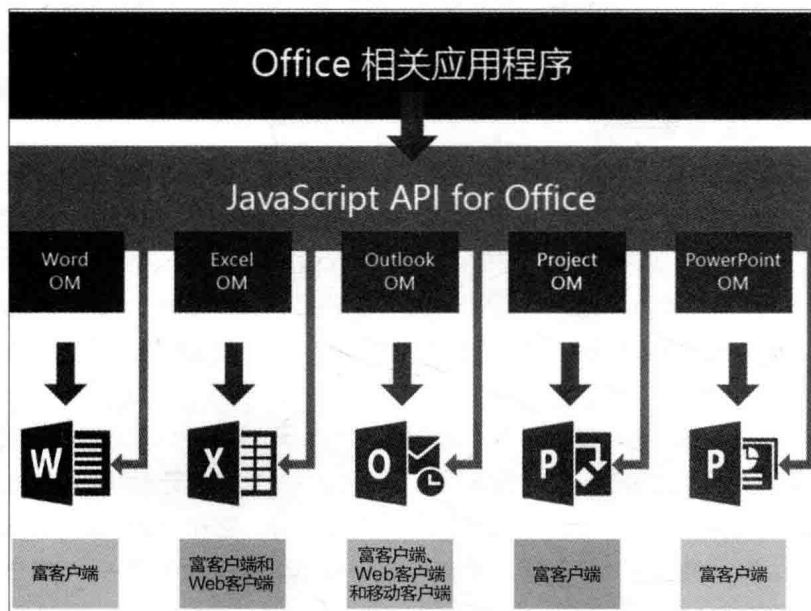


图 1-4

### 1.1.5 Office 365 横空出世

虽然Office 2013之后还有Office 2016，未来还将有Office 20xx这样按照年份编号的版本（称为本地版本），并不是说这些版本不重要但Office 365代表着微软对于广大Office用户的最终承诺，它已经有



了并且还将不断有各种创新，用技术的变革来推动生产力的进步。在展开 Office 365 之前，先对此前的两种开发技术/模式 (VBA 和 VSTO) 进行一个归纳，以向经典致敬。

## 2 VBA

在多个 Office 客户端应用程序中，一直保留着对 VBA (Microsoft Visual Basic for Applications) 这个编程方式的支持。Visual Basic 是微软公司于 1991 年推出的开发语言，直到现在还保持着强大的活力，在编程语言排行榜中名列前茅。除了它本身的易用性之外，还有一个非常关键的原因就是它在 Office 产品家族中的嵌入式编程支持，甚至一些非微软产品 (如 AutoCAD) 也支持 VBA。

虽然 VBA 可以做很多事情，但它最擅长的是对应用程序内部操作的自动化。例如，可以根据 Excel 表格中的数据，每一行生成一个表单，然后将其发送到打印机中打印出来。

现在能找到的任何一个 Office 版本，在打开某个应用 (如 Excel) 后，按下 “Alt+F11” 组合键都可以进入 VBA 的编辑器界面，如图 1-5 所示。

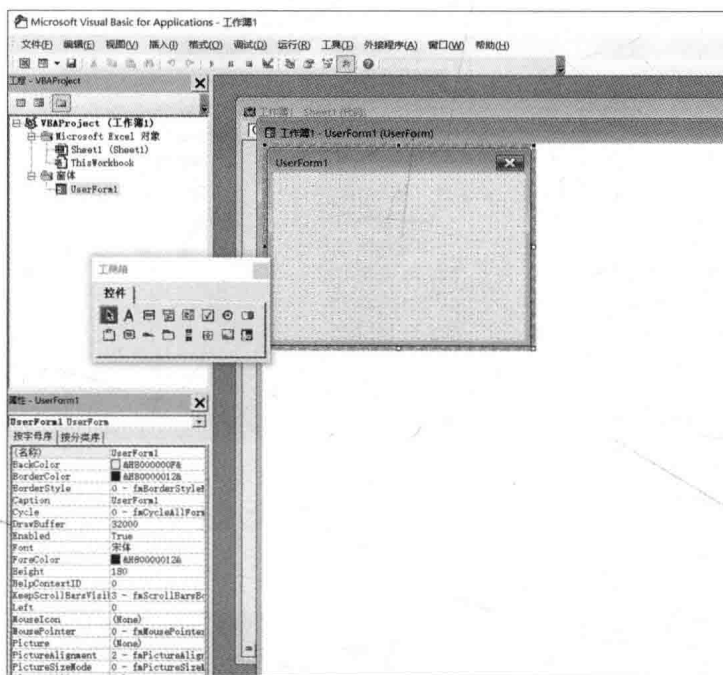


图 1-5

绝大部分应用程序的 VBA 编辑器都支持 3 类模块：一是该应用本身的对象模块，通常与该应用程序的行为—主要体现为事件—密切相关；二是 Forms (Visual Basic 中的 Visual，即可视化的编程)；三是类模块，由于之前提到 VBA 主要是对 Office 的自动化，因此相当一部分 VBA 程序代码都集中在应用本身的对象模块中，而某些标准化较高的通用组件 (如 John 的不朽杰作—Power Pack)，则有大量代码在类模块或 Forms 中。

经历很多次的错误提示消息 (如图 1-6 所示) 操作后，我在技术上才有了一定的提升。学习 VBA 的首要工作就是要比较清楚地了解应用程序的对象模型，其实这个并不难，微软提供了相当丰富且详细的帮助文档 (如 Excel 的不完全对象模型，如图 1-7 所示)，但是熟能生巧，只有经过大量的实践才能做到得心应手。



图 1-6





图 1-7

一个好消息是，Office 应用程序中提供了录制宏的功能。也就是说，可以先按照使用者的想法进行操作，然后录制工具会把相应的代码记录下来。通常这些代码直接就可以运行，但是理想情况下，略加修改后的代码才真正有实用价值。毫不避讳地说，这是我早年学习VBA的一个重要法宝。编程工具能做到这个层面，不仅是业界良心，而且在技术上也是相当先进的。在Excel中录制宏的操作如图1-8所示。

宏 (Macro) 是VBA中的一个重要概念，通常可以简单理解为一组代码。

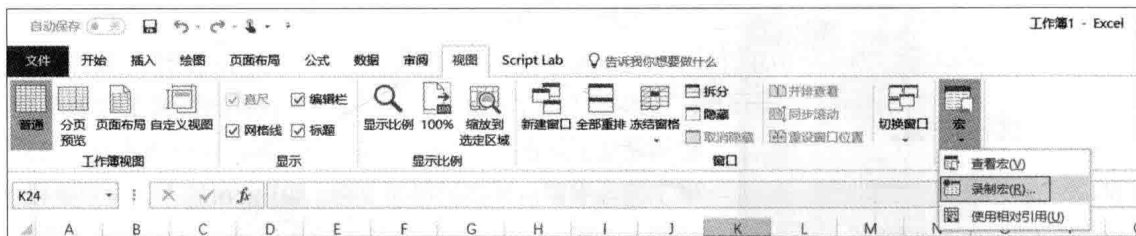


图 1-8