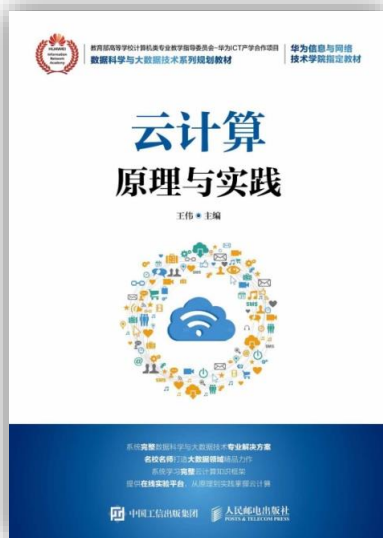


# 云计算原理与实践

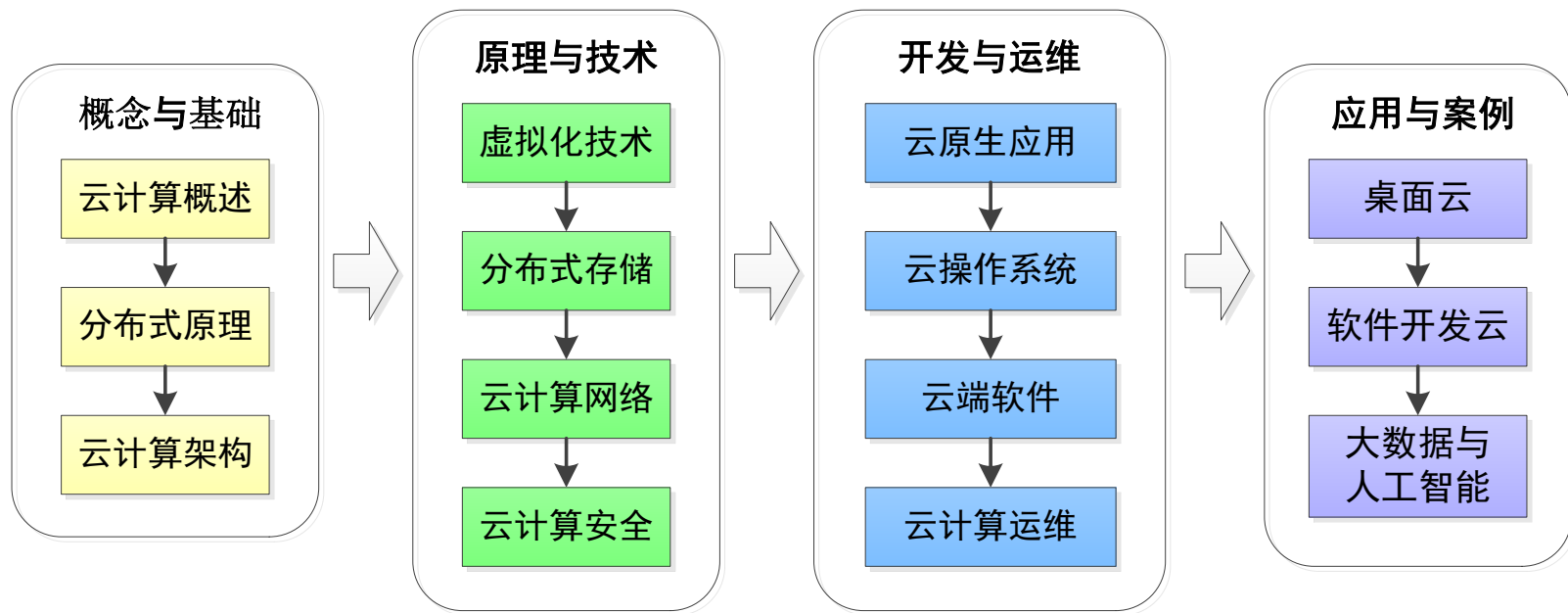
Principles and Practice of Cloud Computing

# 云计算原理与实践

Principles and Practice of Cloud Computing



# 《云计算原理与实践》课程总览



# Outline

- 1.1 初识云计算
- 1.2 云计算的公共特征与分类
- 1.3 云计算的三元认识论
- 1.4 云计算的开源方法论
- 1.5 实践：GitHub

# 1.1 初识云计算

1.1.1 云计算的定义

1.1.2 计算模式的演进过程

1.1.3 云计算简史

1.1.4 云计算的推动力

# 图1.1 超市模式、电厂模式和云计算模式



- 业界有一种很流行的说法，将云计算模式比喻为发电厂集中供电的模式。
- 通过云计算，用户可以不必要去购买新的服务器，更不用去部署软件，就可以得到应用环境或者应用本身。
- 对于用户来说，软硬件产品也就不再需要部署在用户身边，这些产品也不再是专属于用户自己的产品，而是变成了一种可利用的、虚拟的资源。



# 1.1.1 云计算的定义

## 1. 分析师和分析机构对云计算的理解

- **美林证券（Merrill Lynch）**：云计算是通过互联网从集中的服务器交付个人应用（E-mail、文档处理和演示文稿）和商业应用（销售管理、客户服务和财务管理）。
- **《信息周刊》（InformationWeek）**：云计算是一个环境，其中任何的IT资源都可以以服务的形式提供。
- **《华尔街日报》（The Wall street Journal）**：云计算使企业可以通过互联网从超大数据中心获得计算能力、存储空间、软件应用和数据。客户只需要在必要时为其使用的资源付费，从而避免建立自己的数据中心并采购服务器和存储设备。

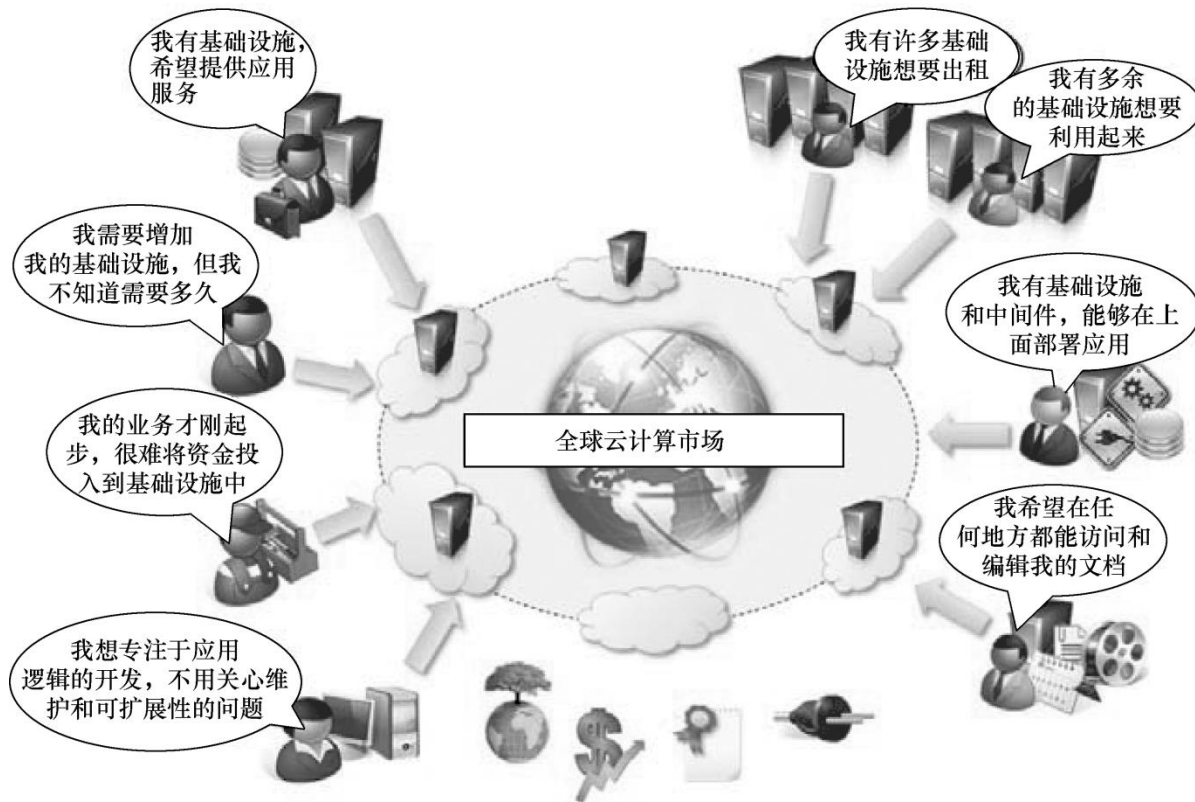
# 云计算的定义

## 2. 不同IT厂商对云计算的理解

- **IBM公司**：云计算是一种计算风格，其基础是用公共或私有网络实现服务、软件及处理能力的交付。云计算的重点是用户体验，核心是将计算服务的交付与底层技术相分离。
- **Google公司**：云计算把计算和数据分布在大量的分布式计算机上，这使得计算力和存储获得了很强的可扩展能力，并方便了用户通过多种接入方式（例如计算机、手机等）方便地接入网络获得应用和服务。其重要特征是开放式的，不会有一个企业能控制和垄断它。
- **微软公司**：未来的计算模式是云端计算，而不是单纯的云计算。这里的“端”是指客户端，也就是说云计算一定要有客户端来配合。时至今日，随着Azure云的崛起，微软已经全面拥抱云计算了。



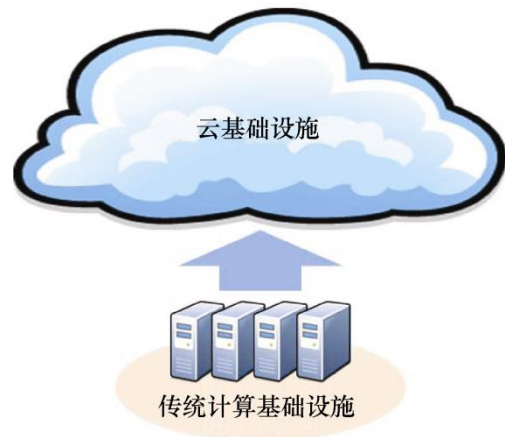
# 图1.2 云计算的全局视角



# 1.1.1 云计算的定义

## 3. 学术界对云计算的理解

- “**网格计算之父**” **Ian Foster**：云计算是一种大规模分布式计算的模式，其推动力来自规模化所带来的经济性。
- **加州大学伯克利分校**：云计算既是指透过互联网交付的应用，也是指在数据中心中提供这些服务的硬件和系统软件。
- **美国国家标准技术学院（NIST）**：云计算是一种模型，这个模型可以方便地通过网络访问一个可配置的计算资源（例如网络、服务器、存储设备、应用程序以及服务等）的公共集。这些资源可以被快速提供并发布，同时最小化管理成本以及服务供应商的干预。

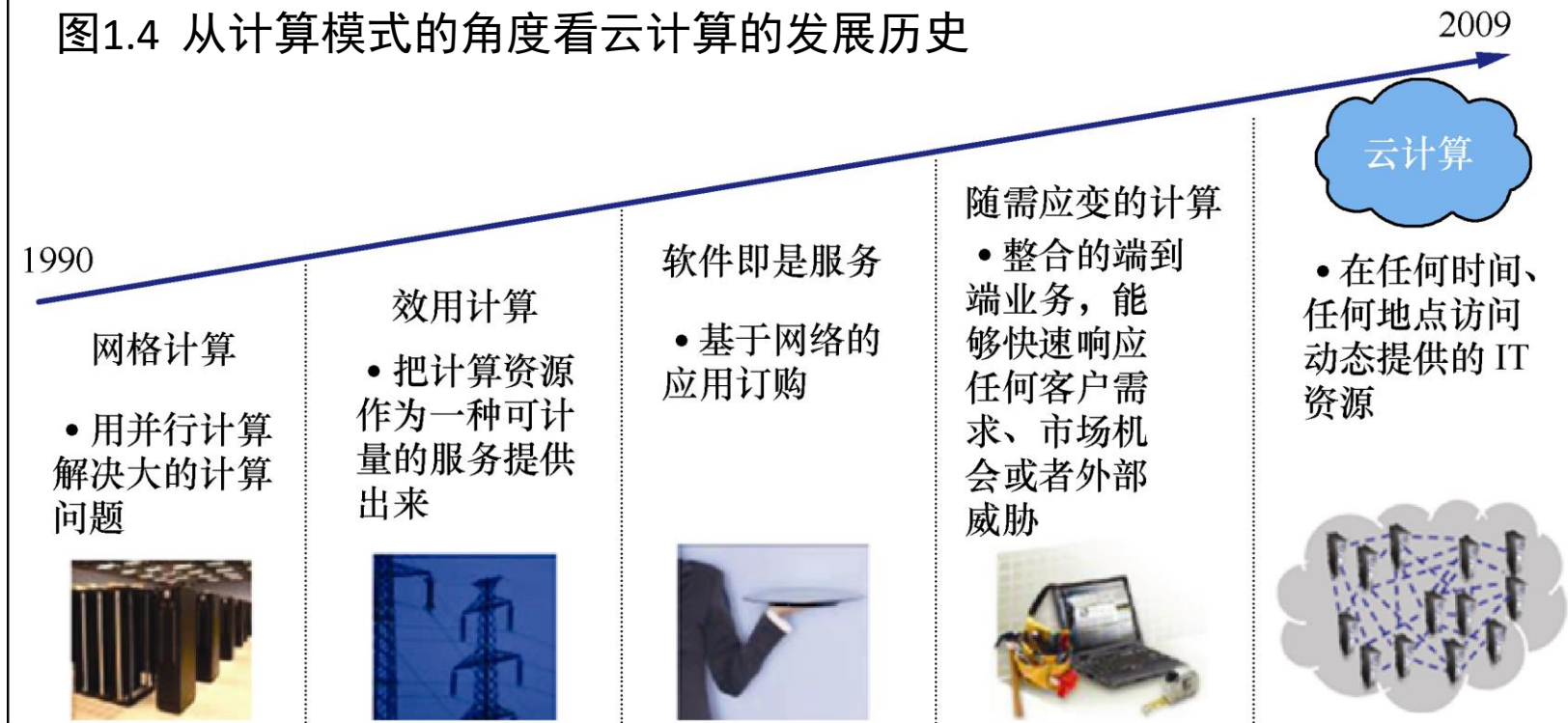


# 云计算总结

- 云计算是各种虚拟化、效用计算、服务计算、网格计算、自动计算等概念的混合演进并集大成之结果。
- 它从主机计算开始、历经小型机计算、客户机/服务器计算、分布式计算、网格计算、效用计算进化而来，它既是技术上的突破（技术上的集大成），也是商业模式上的飞跃。
- 对于用户来说，云计算屏蔽了IT的所有细节，用户无须对云端所提供服务的技术基础设施有任何了解或任何控制，甚至根本不用知道提供服务的系统配置和地理位置，只需要“打开开关”（接上网络），坐享其成即可。
- 云计算在技术和商业模式两个方面的巨大优势，确定了其将成为未来的IT产业主导技术与运营模式。

# 1.1.2 计算模式的演进过程

图1.4 从计算模式的角度看云计算的发展历史



# 1. 主机系统与集中计算

- 1964年，世界上第一台大型主机System/360诞生，引发了计算机和商业领域里的一场革命。
- 大型主机的一个特点就是资源集中，计算、存储集中，这是集中计算模式的典型代表。
- 其实主机系统就是最早的“云”，只不过这些云是面向专门业务、专用网络和特定领域的。
- 云计算与主机计算的一个重要的区别是其面向的用户群体不同。主机计算的用户通常是大型机构，并为关键应用所准备，如人口普查、消费统计、ERP、财务交易等；而云计算则面向普罗大众，可以运行各种各样的大、中、小型应用程序。

## 2. 效用计算

- **效用计算**的目标是把服务器及存储系统打包给用户使用，按照用户实际使用的资源量对用户进行计费，类似于水、电、气和电话等服务的提供方式，使用户能够像把灯泡插入灯头一样来使用计算机资源。
- 效用计算的实际运用以IBM公司为主要代表。IBM公司将自己的主机资源按照时间租给不同的用户，主机仍然存放在IBM的数据中心，用户在远程或者IBM数据中心现场使用IBM的资源。
- 与云计算比较起来，效用计算仅规定了IT资产的计费模式，对IT资产的其他方面，如技术、管理、配置、安全等诸多方面并不做限定。
- 而付费模式仅仅是云计算所考虑的一个因素，云计算要考虑的因素还包括许多。

### 3. 客户机/服务器模式

- **客户机/服务器**模型则泛指所有的能够区分某种服务提供者（服务器）和服务请求者（客户机）的分布式系统。
- 从服务的访问模式上看，云计算确实有客户机/服务器模式的影子，不过，在这种形似的背后，云计算提供的这个“远程服务器”具有无限的计算能力、无限的存储容量，且从来不会崩溃，几乎没有什么软件不能运行在其上。用户还可以发布自己的应用程序到这个“远程服务器”，而这个“远程服务器”可以为应用程序自动配置所需的资源，并随需变化。



## 4. 集群计算

- 服务器**集群计算**是用紧密耦合的一组计算机来达到单个目的，而云计算是根据用户需要提供不同支持来达到不同的目的。
- 服务器集群计算是有限度的分布式计算，其面临的挑战不如云计算所面临的分布式计算复杂。
- 集群计算并不考虑交互式的终端用户，而云计算恰恰需要考虑。
- 显然，云计算包含了服务器集群计算的元素。

# 5. 服务计算

- **服务计算**也称为面向服务的计算，其更为准确的名称是软件即服务（SaaS）。此种计算范式将所有应用程序都作为服务来予以提供，用户或其他的应用程序则使用这些服务，而不是买断或拥有软件。
- 在服务计算模式下，不同服务之间相对独立，松散耦合，随意组合。对服务计算来说，服务的发现是重点。
- 云计算大量采用了服务计算的技术和思维方式，重要区别在于：
  - 虽然服务计算一般在互联网上实现，但服务计算不一定必须在云中提供，单台服务器、小规模集群、有限范围的网络平台就可以提供服务计算；
  - 服务计算一般仅限于软件即服务，而云计算将服务的概念推广到了硬件和运行环境，囊括了基础设施即服务、平台即服务的概念。也就是说，云计算的服务理念比传统的服务计算概念更加广泛。

## 6. 个人计算机与桌面计算

- 个人计算机的出现极大地推动了软件产业的发展，各种面向终端消费者的应用程序涌现出来。个人计算机可以完成绝大部分的个人计算需求，这种模式也叫**桌面计算**。
- 在互联网出现之前，软件和操作系统的销售模式都是授权（License）模式，也就是通过软盘或者光盘，将软件代码复制到计算机之上，而每一次复制，都需要向软件开发商付费。
- 这种模式一些问题，例如费用太高、软件升级烦琐等。升级的目的是解决之前的一些问题，或是使用新的功能，但是升级的过程有时会很烦琐。对于一个大型企业来讲，它的IT部门可能需要管理上百种软件、上千个版本、上万台计算机，每个版本的软件都需要维护，包括问题追踪、补丁管理、版本升级和数据备份等，这绝非一项简单的工作。

# 7. 分布式计算

- **分布式计算**依赖于分布式系统。分布式系统由通过网络连接的多台计算机组成。每台计算机都拥有独立的处理器及内存。这些计算机互相协作，共同完成一个目标或者计算任务。
- 分布式计算是一个很大的范畴，它包含了很多人们熟悉的计算模式和技术，例如网格计算、P2P计算、客户机/服务器计算（C/S）和浏览器/服务器计算（B/S），当然也包括云计算。
- 在当今的网络时代，非分布式计算的应用已经很少了，只有部分单机运行的程序属于这一范畴，例如文字处理、单机游戏等。

## 8. 网格计算

- **网格计算**出现于20世纪90年代。它是伴随着互联网而迅速发展起来的、专门针对复杂科学计算的新型计算模式。
- 这种计算模式利用互联网把分散在不同地理位置的计算机组织成一台“虚拟的超级计算机”，其中每一台参与计算的计算机就是一个“节点”，而整个计算是由成千上万个“节点”组成的“一堆网格”，所以这种计算方式叫网格计算。
- 为了进行一项计算，网格计算首先把要计算的数据分割成若干“小片”，然后将这些小片分发给分布的每台计算机。每台计算机执行它所分配到的任务片段，待任务计算结束后将计算结果返回给计算任务的总控节点。

## 9. SaaS

- SaaS全称为Software as a service，中文译为“软件即服务”。其实它所表达的也是一种计算模式，就是把软件作为服务。
- 它是一种通过Internet来提供软件的模式，厂商将应用软件统一部署在自己的服务器上，客户可以根据自己的实际需求，通过互联网向厂商订购所需的软件应用服务，按订购的服务多少和时间长短向厂商支付费用，并通过互联网获得厂商提供的服务。
- 用户不用再购买软件，而改为向提供商租用基于Web的软件，来管理企业经营活动，且无须对软件进行维护，服务提供商会管理维护软件。
- 软件厂商在向客户提供互联网应用的同时，也提供软件的离线操作和本地数据存储，让用户随时随地都可以使用其订购的软件和服务。

## 10. 云计算的出现

- 总体来看，云计算至少有以下四个优势：
  - 按需供应的无限计算资源；
  - 无须事先花钱就能使用的IT架构；
  - 基于短期的按需付费的资源使用；
  - 单机难以提供的事务处理环境。



# 1.1.3 云计算简史

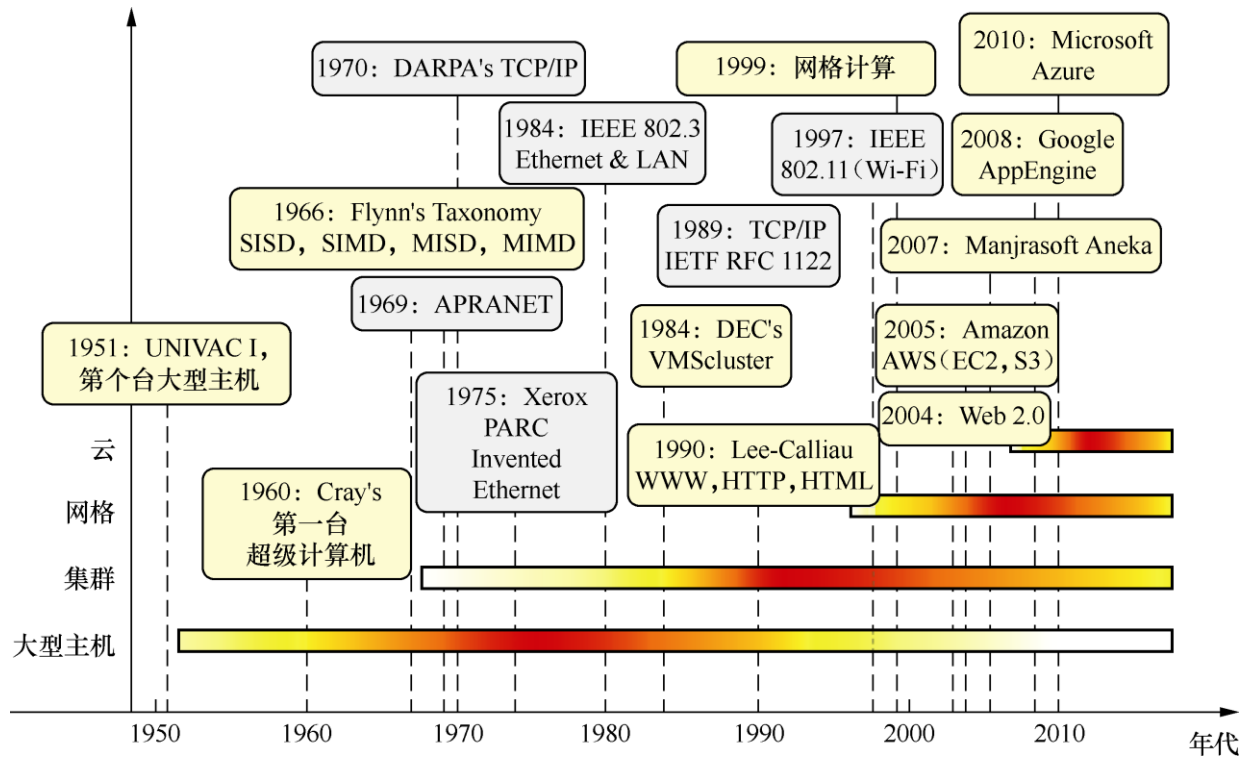


图1.5 云计算的发展简史

## 1.1.4 云计算的推动力

云计算并不是凭空出现的，它的出现是由多种因素促成，具有一定的必然性。

1. 网络带宽的提升
2. 技术成熟度
3. 移动互联网的发展
4. 数据中心的演变
5. 经济因素
6. 大数据

# 1.2 云计算的公共特征与分类

1.2.1 云计算的公共特征

1.2.2 云计算的分类

1.2.3 与云计算有关的技术

## 1.2.1 云计算的公共特征

通过对云计算方案的特征进行归纳和分析，可发现这些方案所提供的云服务有着显著的公共特征，这些特征也使云计算明显区别于传统的服务。

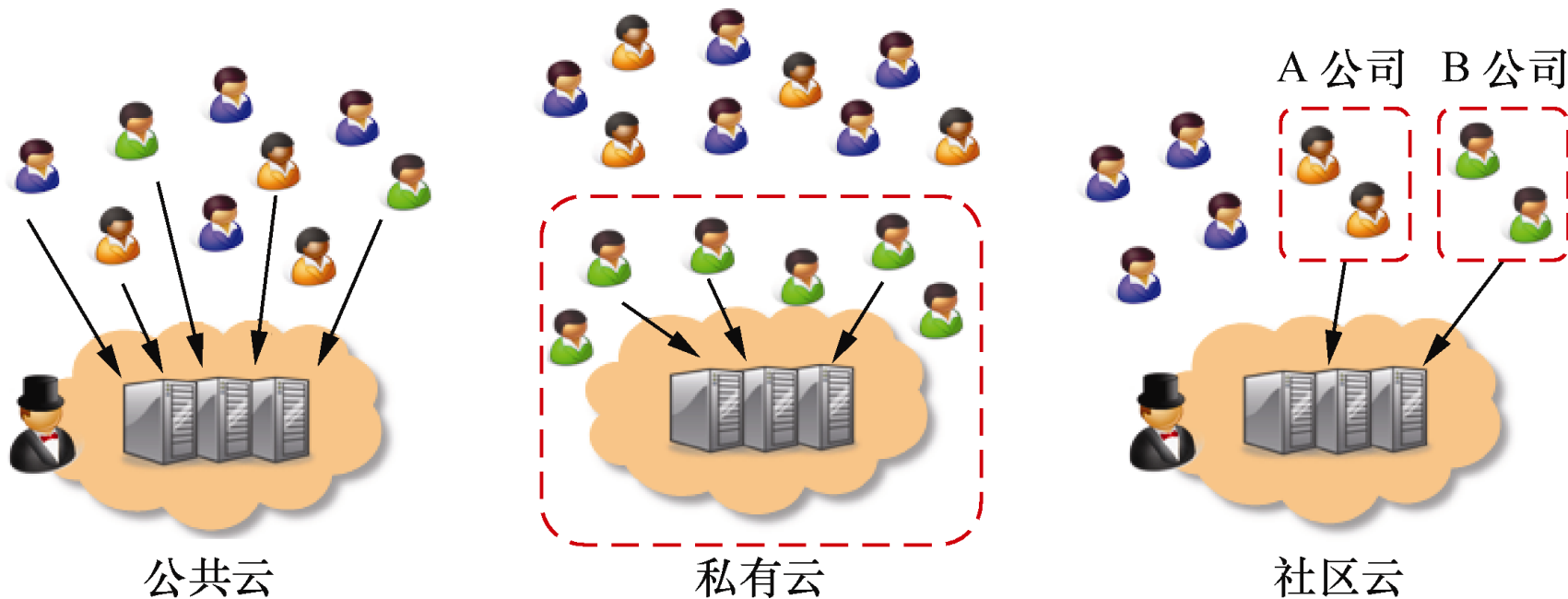
1. 弹性伸缩
2. 快速部署
3. 资源抽象
4. 按用量收费
5. 宽带访问

## 1.2.2 云计算的分类

### 1. 根据云的部署模式和云的使用范围进行分类

- (1) 公共云
- (2) 私有云（或称专属云）
- (3) 社区云
- (4) 混合云
- (5) 行业云
- (6) 其他云类型

# 图1.6 公共云、私有云与社区云



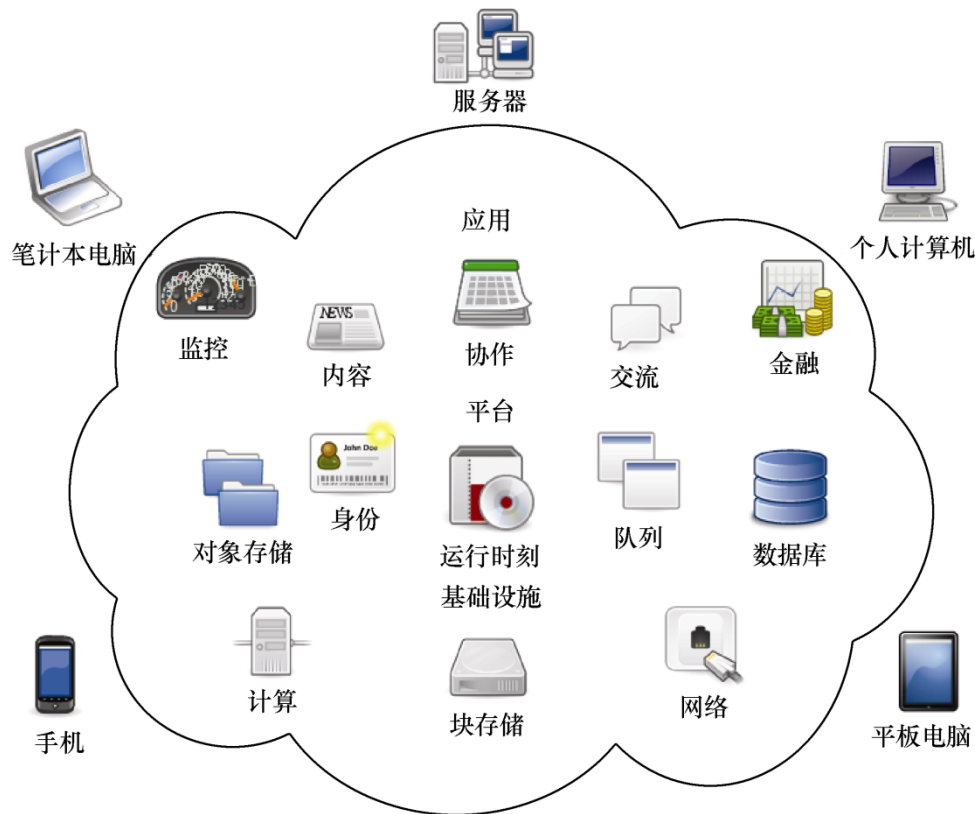
# 云计算的分类

## 2. 针对云计算的服务层次和服务类型进行分类

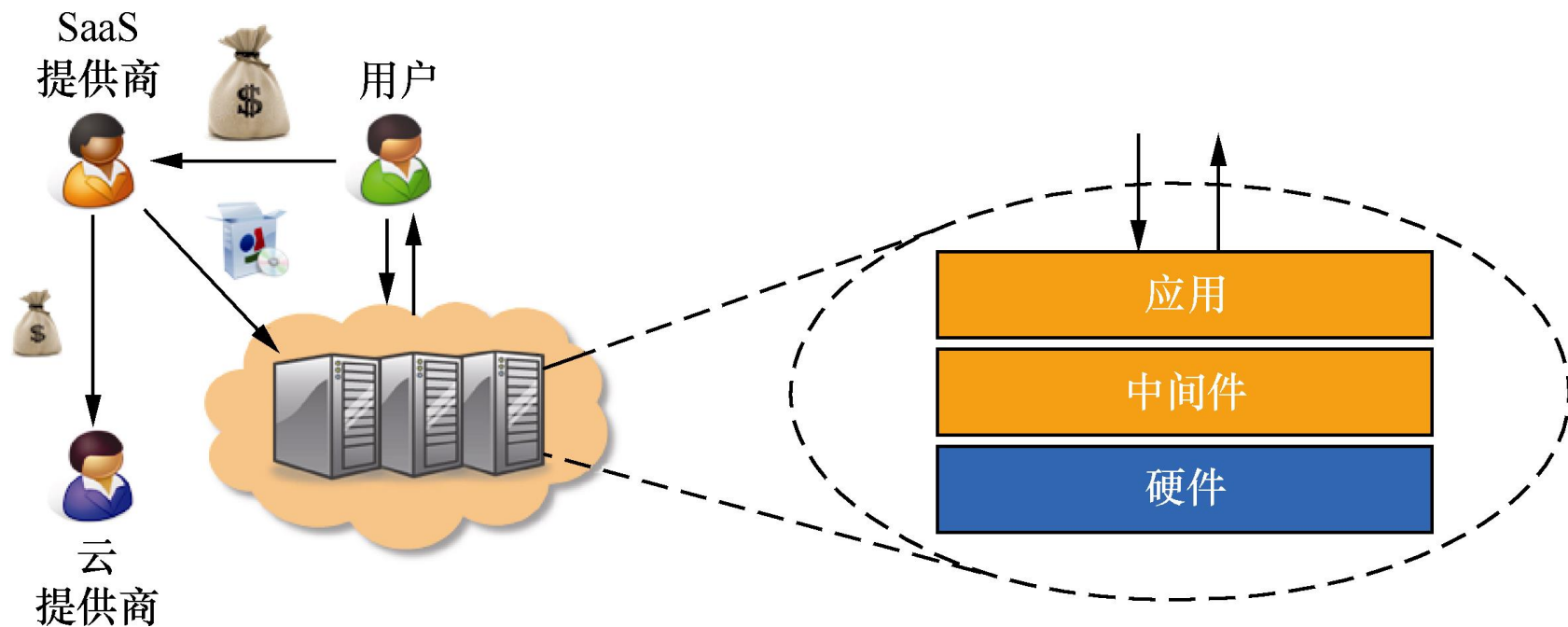
- (1) 基础设施即服务 (Infrastructure as a Service, IaaS)
- (2) 平台即服务 (Platform as a Service, PaaS)
- (3) 软件即服务 (Software as a Service, SaaS)



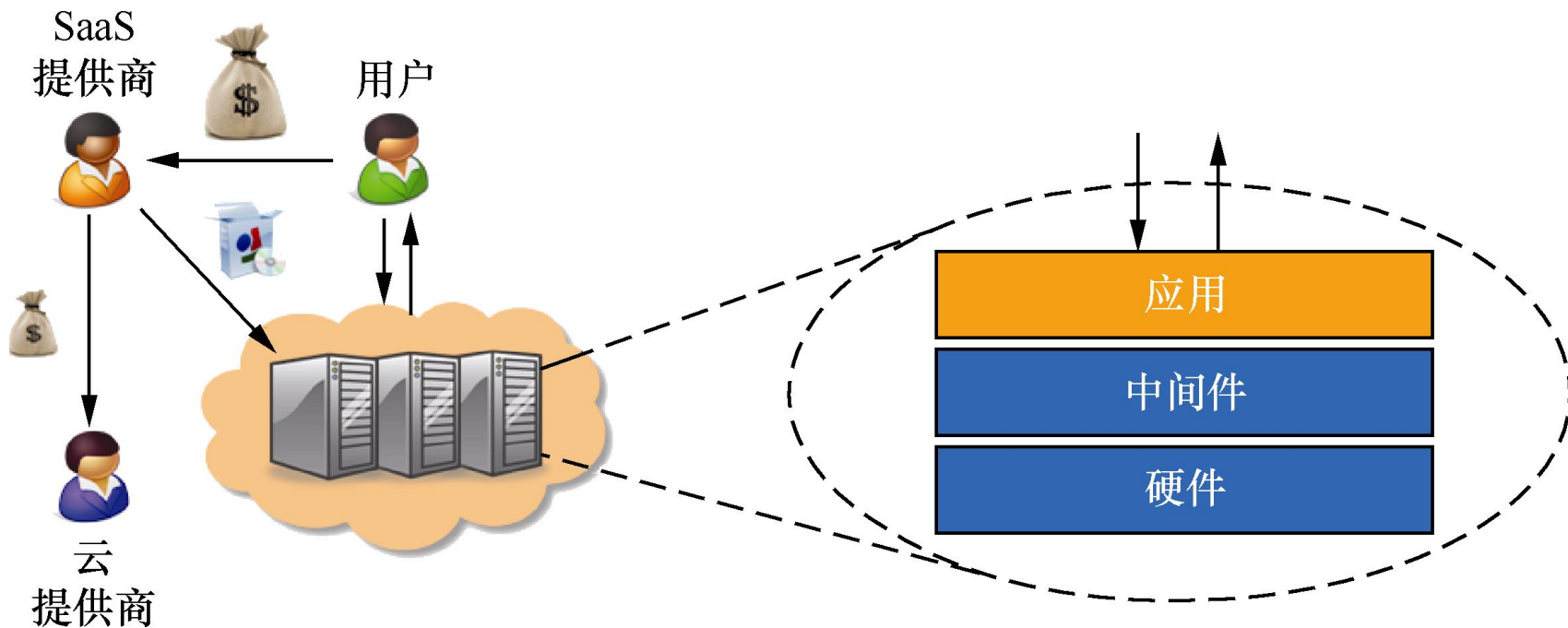
# 图1.7 云计算的组成元素



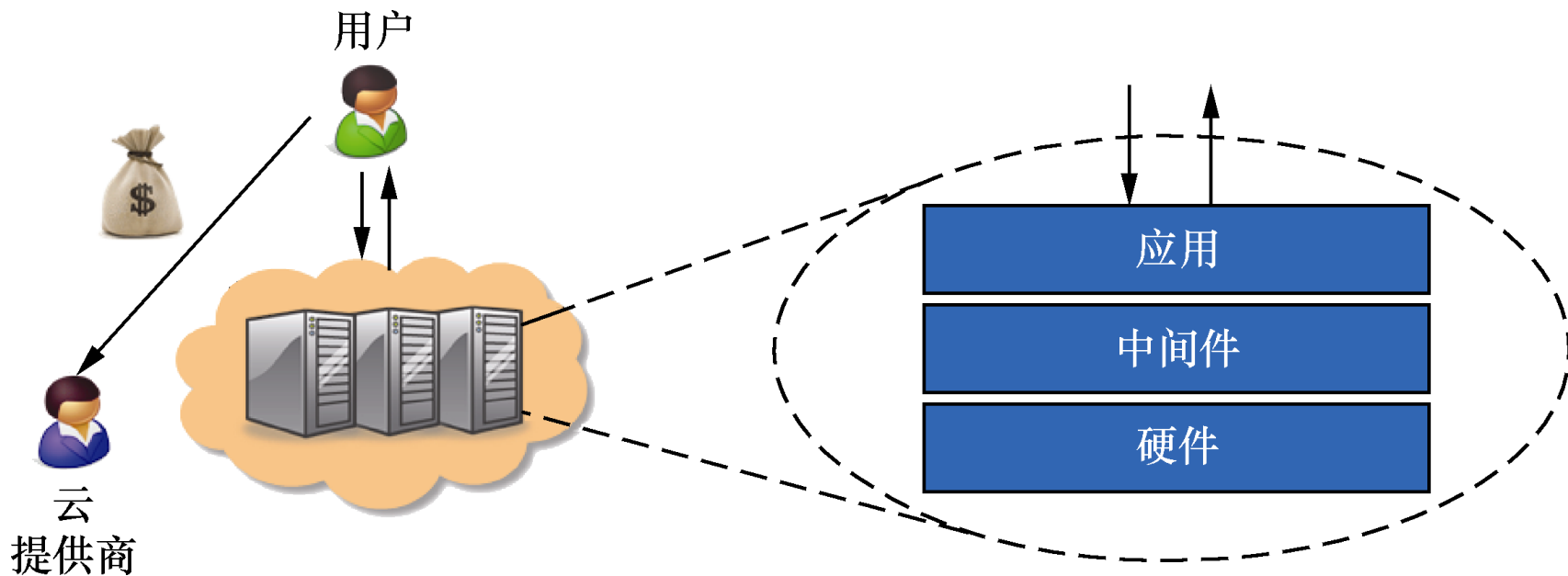
# 图1.8 基础设施即服务的层次



# 图1.9 平台即服务的层次



# 图1.10 软件即服务的层次



## 1.2.3 与云计算有关的技术

- 从技术的角度看，云计算体现出分布式系统、虚拟化技术、负载均衡等各种技术有着千丝万缕的联系。
- 在具体的技术实现上，云计算平台创新性地融合了多种技术的思想，通过不同的组合，解决在具体应用时遇到的不同问题。
- 就技术而言，云计算在本质上源自超大规模分布式计算，是一种演进的分布式计算技术。
- 云计算还延伸了SOA的理念，并融合了虚拟化、负载均衡等多种技术方法，形成了一套新的技术理念和实现机制。
- 云计算表现出的核心意义不仅在于技术的发展，还在于通过组织各种技术，使人们建立IT系统的思路 and 结构发生根本性的变化。

# 1. 并行计算

- **并行计算**（Parallel Computing）是指同时执行多个指令的计算模式，其原理为一个“大”问题可以被分解为多个同时处理的“小”问题。
- 并行计算的主要动力在于加快计算速度，因此确定问题分解的并行算法，对于并行计算而言至关重要，所以在结构上并行计算是紧耦合（Tight Coupling）的概念。
- 在软件工程中，“耦合”指的是互相交互的系统彼此间的依赖。紧耦合表明模块或者系统之间关系紧密，存在明显的依赖关系。

## 2. SOA

- SOA是**面向服务的体系结构**（Service Oriented Architecture）的简称，依照结构化信息标准促进组织（OASIS）所给出的定义，“SOA是一种组织和利用可能处于不同所有权范围控制下的分散功能的范式。”
- 通常所说的SOA是一套设计和开发软件的原则和方法，它将应用程序的不同功能单元（即“服务”）通过这些服务之间定义良好的接口和协议联系起来，以使实现服务的平台或系统中，所构建的各类服务可以通过一种统一和通用的方式进行交互。
- 简而言之，SOA是一种理念，即给定一种标准接口和一个约束接口的服务协议，则任何业务应用，只要能满足服务协议，即可通过给定的标准接口进行通信和交互，实现对接。



# 3. 虚拟化

- **虚拟化（Virtualization）**是为某些事物创造的虚拟（而非真实）版本，例如硬件平台、计算机系统、存储设备和网络资源等。
- 其目的是为了摆脱现实情况下物理资源所具有的各种限制，即“虚拟化是资源的逻辑表示，它不受物理限制的约束。”
- 云计算所涉及的虚拟化，是经过发展之后的更高层次的虚拟化，是指所有的资源——计算、存储、应用和网络设备等连接在一起，由云计算平台进行管理调度。
- 借助于虚拟化技术，云计算平台可以对底层千差万别的资源进行统一管理，也可以随时方便地进行资源调度的管理，实现资源的按需分配，从而使大量物理分布的计算资源可以在逻辑层面上以一个整体的形式呈现，并支撑各类应用需求。

## 1.3 云计算的三元认识论

1.3.1 云计算作为一种商业模式

1.3.2 云计算作为一种计算范式

1.3.3 云计算作为一种实现方式

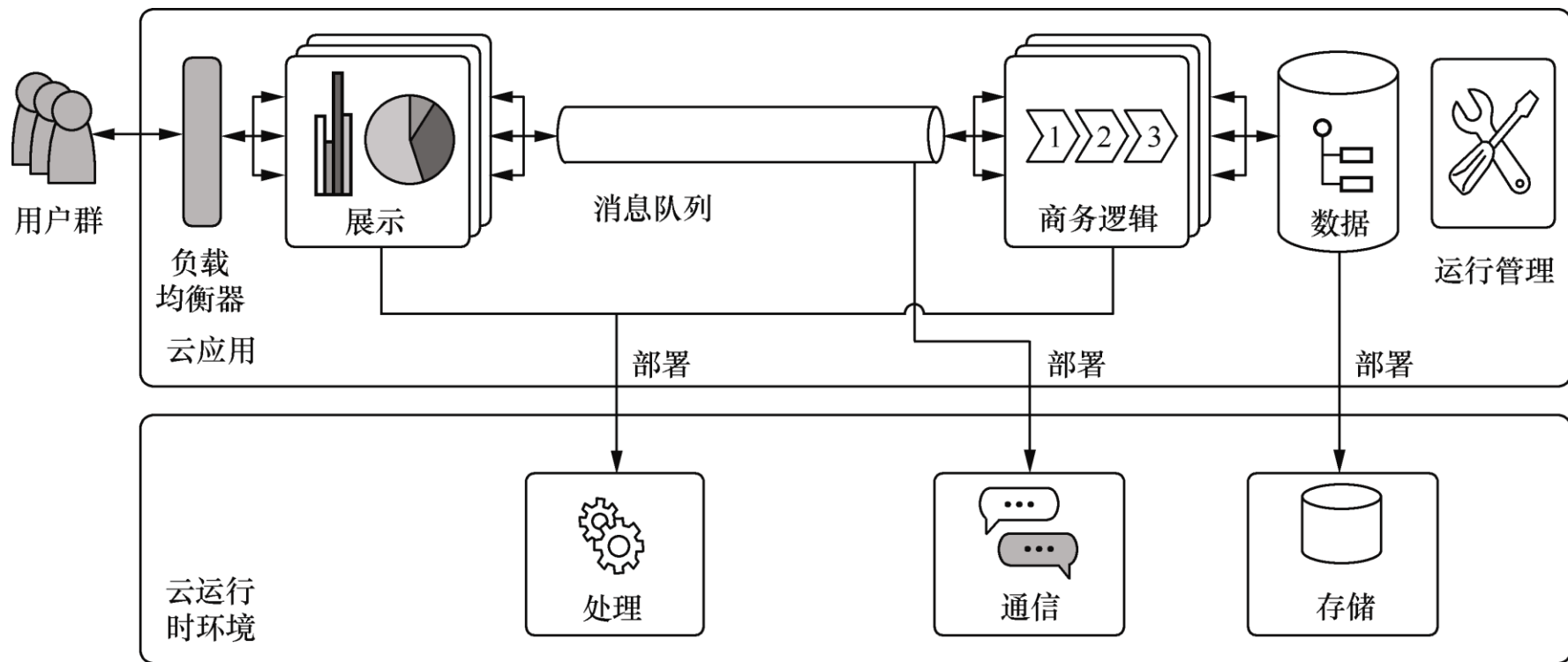
# 1.3.1 云计算作为一种商业模式

- 云计算服务代表一种新的商业模式，SaaS（软件即服务）、PaaS（平台即服务）和IaaS（基础设施即服务）是这种商业模式的表现形式。
- 伴随着云计算服务理念的发展，云计算也形成了一整套的软件架构与技术实现机制，而常常听到的云计算平台就是这套机制的具体体现。
- 对于任何一种商业模式而言，除了理论上可行之外，还要保证实践上可用。因此，伴随着云计算服务理念的发展，云计算也形成了一整套技术实现机制，而云计算平台则是这套机制的具体体现。

## 1.3.2 云计算作为一种计算范式

- 云计算作为一种计算范式，其计算边界既由上层的经济因素所决定，也由下层的技术因素所决定。经济因素自上而下决定这种计算范式的商业形态，实现技术自下而上决定这种计算范式的技术形态。作为云计算服务的计算范式又可以从两个角度来进一步理解：
  - 横向云体逻辑结构
  - 纵向云栈逻辑结构

# 图1.11 云计算的横向云体逻辑结构



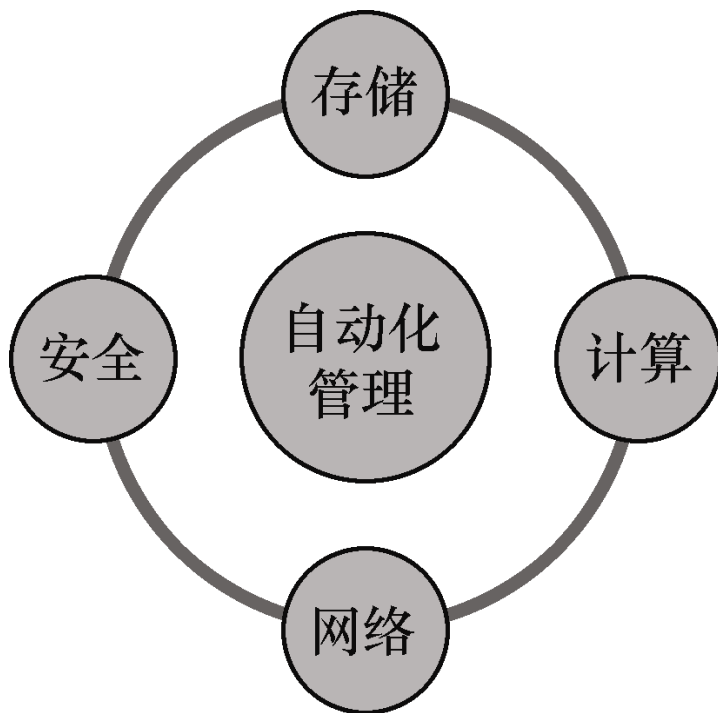
# 图1.12 云计算的纵向云栈逻辑结构



# 1.3.3 云计算作为一种实现方式

- 云计算最终的实现方式是需要新一代的软硬件技术推动，即目前流行的数据中心，并且朝着软件定义的数据中心（Software Defined Data Center, SDDC）所演进。
- 数据中心是云计算实现的最终归属，包括全方位的计算、存储和通信需求。
- 软件定义数据中心是一个比较新的概念，它将虚拟化概念（如抽象，集中和自动化）扩展到所有数据中心资源和服务，以实现IT即服务（ITaaS）。在软件定义的数据中心中，基础架构的所有元素（网络，存储，CPU和安全）都是被虚拟化并作为服务交付的。

# 图1.13 软件定义数据中心功能划分





## 1.4 云计算的开源方法论

1.4.1 开源定义和相关概念

1.4.2 开源的价值和意义

1.4.3 开源发展历程

1.4.4 开源是方法论

1.4.5 开源给云计算人才培养带来的挑战

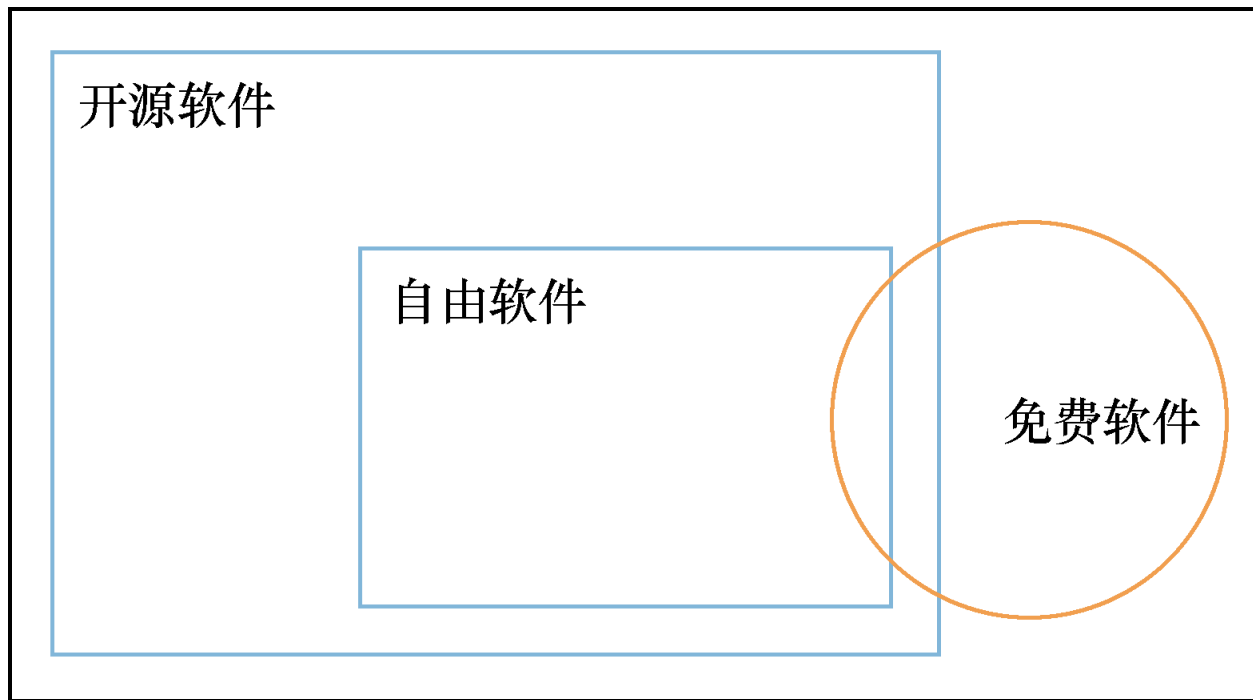
## 1.4.1 开源定义和相关概念

- **开源**，即开放一类技术或一种产品的源代码、源数据、源资产等，可以是各行业的技术或产品，其范畴涵盖文化、产业、法律、技术等多个社会维度。如果开放的是软件代码，一般被称作开源软件。
- **开源软件**是一种版权持有人为任何人和任何目的提供学习、修改和分发权利，并公布源代码的计算机软件。
- 开源软件促进会（Open Source Initiative, OSI）对开源软件有明确的定义，业界公认只有符合该定义的软件才能被称为开发源代码软件，简称开源软件。

# OSI对开源软件特征的定义

- 开源软件的许可证不应限制任何个人或团体将包含该开源软件的广义作品进行销售或者赠予；
- 开源软件的程序必须包含源代码，必须允许发布源代码及以后的程序；
- 开源软件的许可证必须允许修改和派生作品，并且允许使用原有软件的许可条款发布它们。

图1.14 开源软件、自有软件和免费软件的关系



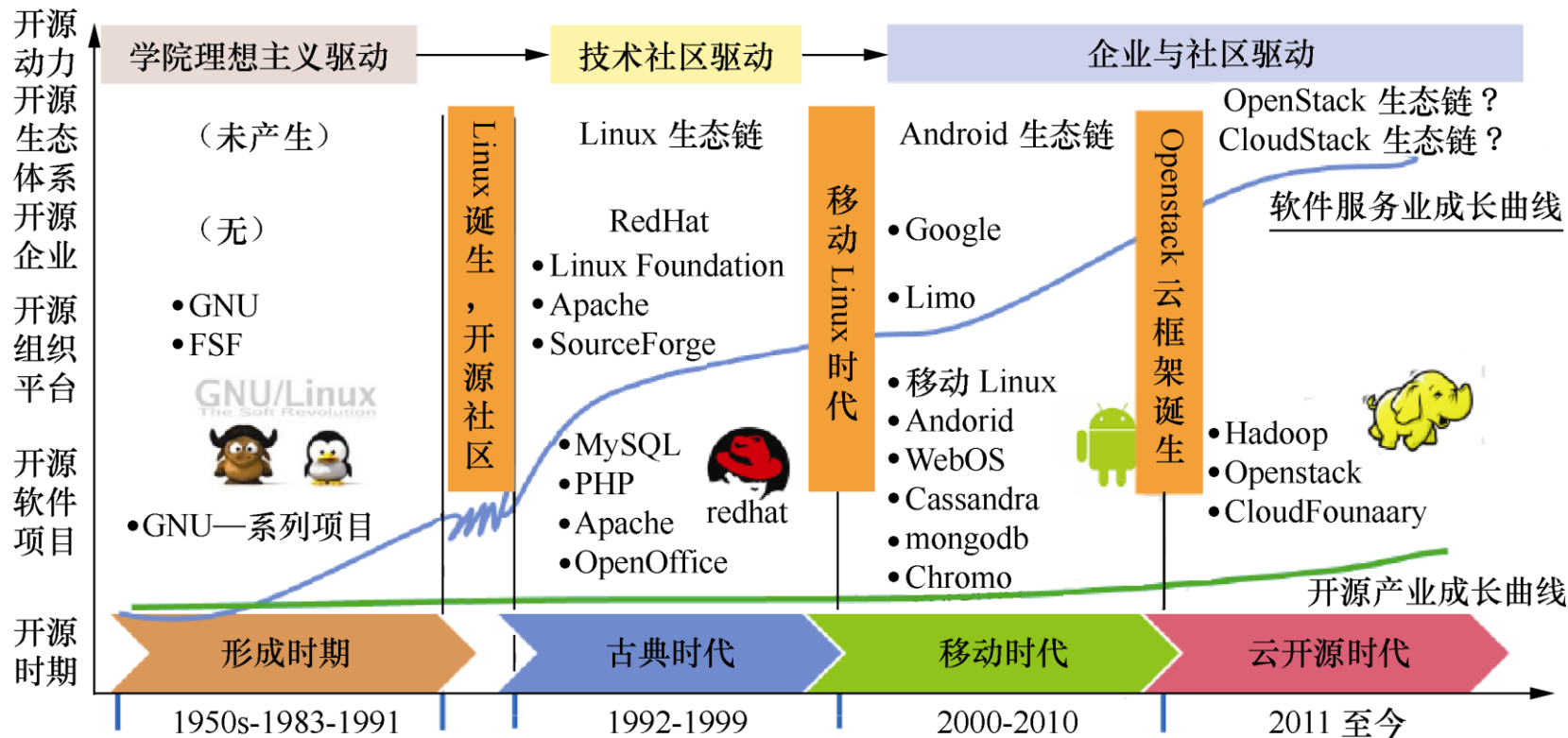
## 1.4.2 开源的价值和意义

1. 开源生态促进国家信息技术创新，带动经济发展
2. 软件厂商依托开源技术提升研发能力
3. 用户使用开源技术改变信息化路线
4. 企业自主开源，引领技术发展路径

## 1.4.3 开源发展历程

1. 开源项目形成
2. 开源社区形成，企业逐渐参与到开源社区做贡献
3. 企业和社区驱动开源
4. 开源代码托管

# 图1.15 开源发展历程



## 1.4.4 开源是方法论

### 1. 开源价值观的内涵

- (1) 奉献精神
- (2) 感恩意识
- (3) 开放精神
- (4) 勇敢精神
- (5) 追求持续进步的精神
- (6) 按照劳动获得公平价值回报的精神



# 开源是方法论

## 2. 开源方法论的内涵

- (1) 通过开放共享促进进步与创新的方法
- (2) 通过聚集、累积众多参与者的劳动与智慧来解决复杂性、系统性问题的方法
- (3) 通过社区平台完成开源项目的方法
- (4) 通过知名企业、个人的有效组织和引导来发展、完成项目的方法

## 1.4.5 开源给云计算人才培养带来的挑战

1. 开源模式对开发人员的能力有特别的需求
2. 对开源软件开发的学习与实践是增强开源能力的一种重要途径
  - (1) 开源文化教育。
  - (2) 开源意识教育。
  - (3) 开源技能教育。

## 1.5 实践：GitHub

1.5.1 GitHub简介

1.5.2 使用GitHub

1.5.3 GitHub界面总览

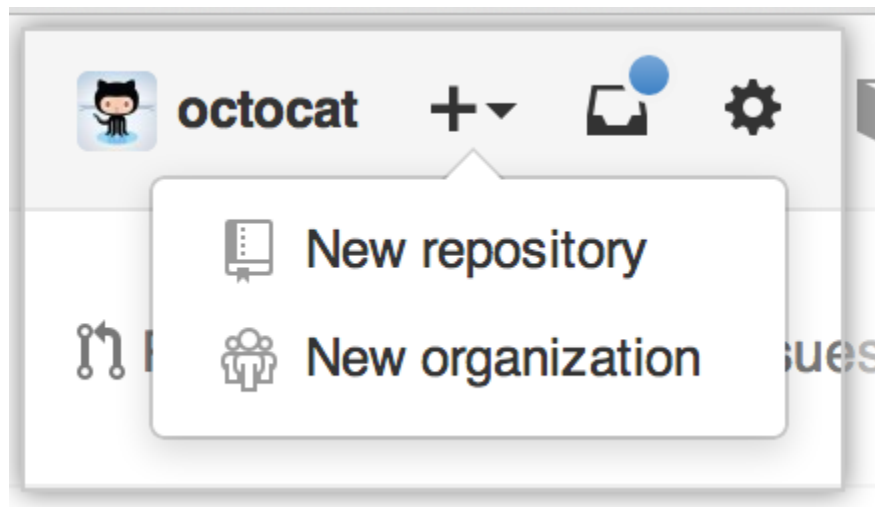
## 1.5.1 GitHub简介

- **Git**已经成为程序员必备技能之一，而GitHub作为流行的Git仓库托管平台，不仅提供Git仓库托管，还是一个非常优秀的技术人员社交平台，技术人员可以通过开源的项目进行协作、交流，是现在优秀的工程师必须娴熟运用的方法。
- Git是一个优秀的分布版本控制系统。在一个分布版本控制系统中，每个用户都有一份完整的源代码，可以对这个本地的数据进行操作。
- **GitHub**是一个面向开源及私有软件项目的托管平台，因为只支持Git作为唯一的版本库格式进行托管，故名GitHub。

## 1.5.2 使用GitHub



1. GitHub注册
2. 安装Git
3. 通过Git验证GitHub
4. 在GitHub上创建一个新仓库
5. 提交第一个更改
6. Fork一个示例仓库
7. 同步Fork仓库

# 图1.16 新建仓库



# 图1.17 创建名字

**Owner**

PUBLIC   octocat ▾ /


**Repository name**

hello-world ✓


Great repository names are short and memorable. Need inspiration? How about **spawncamping-tyrion**.

# 图1.18 为仓库添加描述

PUBLIC



Owner

 octocat ▾

Repository name

hello-world ✓

Great repository names are short and memorable. Need inspiration? How about **spawncamping-tyrion**.


Description (optional)

My first repository on GitHub.




# 图1.19 选择仓库类型

PUBLIC



Owner

 octocat ▾


Repository name


hello-world ✓

Great repository names are short and memorable. Need inspiration? How about **spawncamping-tyrion**.

Description (optional)

My first repository on GitHub.

☒  **Public**  
Anyone can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

## 图1.20 选择初始化文件



**Public**

Anyone can see this repository. You choose who can commit.



**Private**

You choose who can see and commit to this repository.



**Initialize this repository with a README**

This will allow you to `git clone` the repository immediately. Skip this step if you have already run `git init` locally.



Add .gitignore: **None** ▾

Add a license: **None** ▾




Create repository

# 图1.21 完成创建一个仓库

- ☒  **Public**  
Anyone can see this repository. You choose who can commit.
- ☐  **Private**  
You choose who can see and commit to this repository.

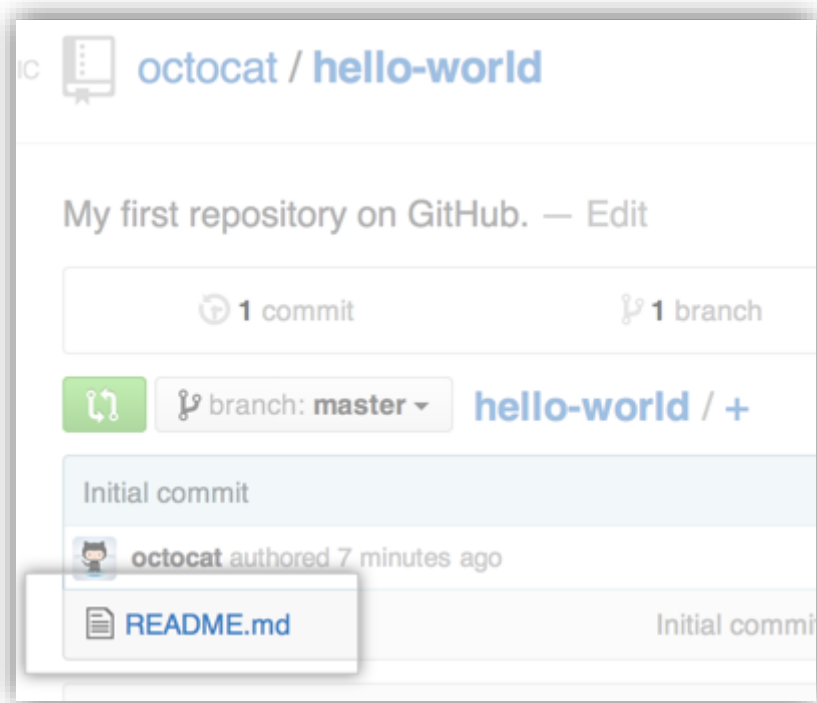
- ☒ **Initialize this repository with a README**  
This will allow you to `git clone` the repository immediately. Skip this step if you have already run `git init` locally.

Add .gitignore: **None** ▾

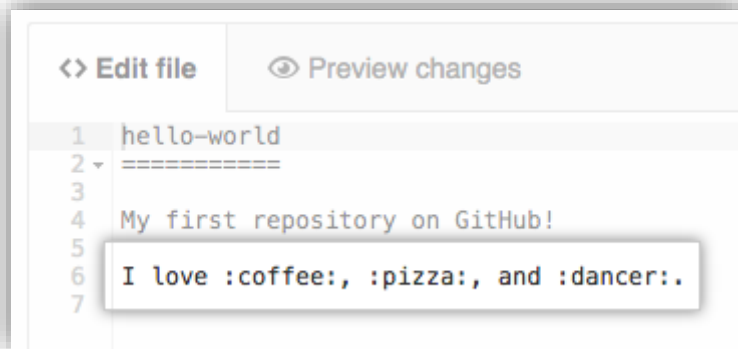
Add a license: **None** ▾ 

Create repository

# 图1.22 单击README.md文件



# 图1.23 输入用户的信息

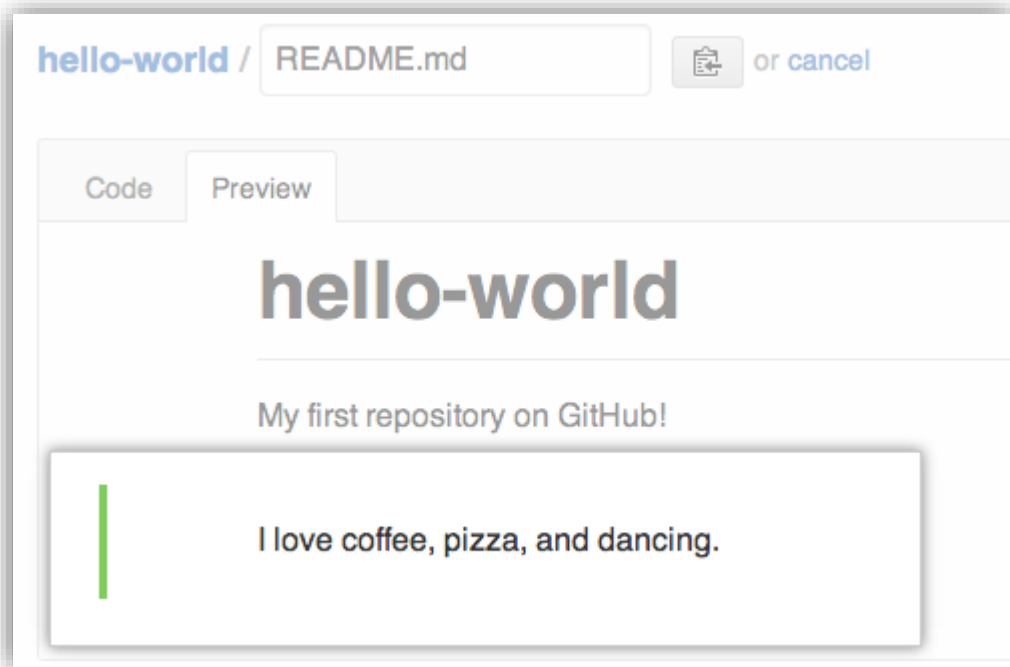


```
<> Edit file    Preview changes
1 hello-world
2 =====
3
4 My first repository on GitHub!
5
6 I love :coffee:, :pizza:, and :dancer:
7
```

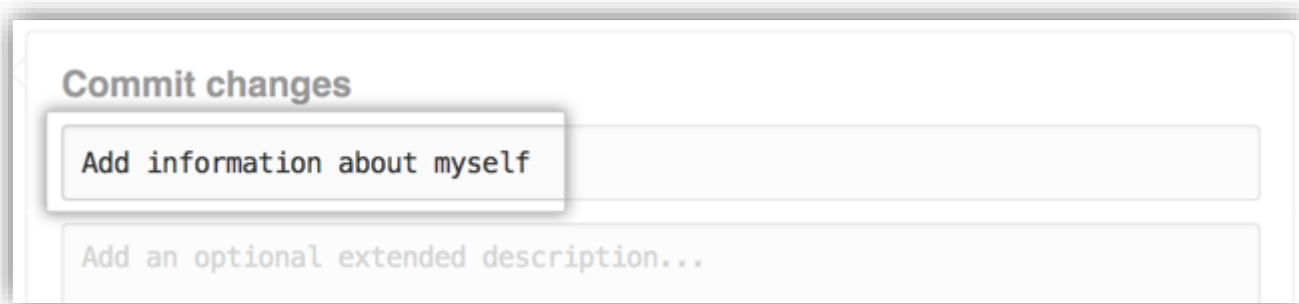
# 图1.24 预览



# 图1.25 新的内容



## 图1.26 注释修改



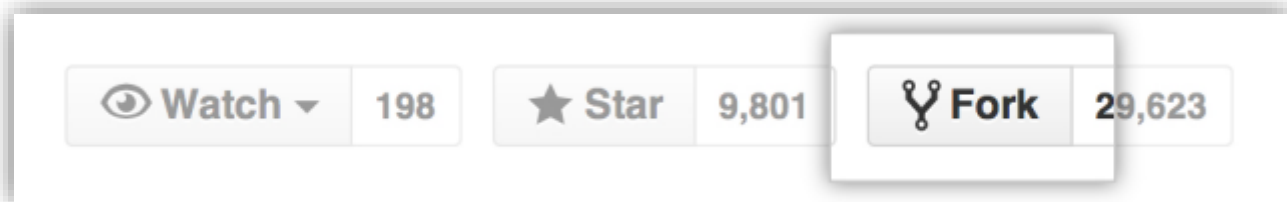
**Commit changes**

Add information about myself

Add an optional extended description...



# 图1.27 Fork一个仓库



# 1.5.3 GitHub界面总览



图1.28 Github界面总体功能

# 小结



summary

- 初识云计算
- 云计算的公共特征与分类
- 云计算的三元认识论
- 云计算的开源方法论
- 实践：GitHub

# 课内复习

1. 云计算的定义是什么？
2. 云计算的公共特征有哪几个？
3. 云计算按照部署方式和服务类型分别分成哪几类？
4. 如何从三元认识论的角度理解云计算？
5. 云计算作为一种计算范式可以分成哪两种结构？
6. 开源软件、自由软件和免费软件的区别与联系是什么？

# 课外思考

1. 计算系统是如何演变成今天的云计算的？
2. 如何理解“开源是种方法论”？
3. 开源技术是如何促进云计算发展的？

# 动手实践1

- Git是目前世界上最流行的开源分布式版本控制系统之一，用于敏捷高效地处理项目。Git是 Linus Torvalds 为了帮助管理Linux内核开发而开发的一个开放源码的版本控制软件。Git与常用的版本控制工具CVS、Subversion等不同，它采用了分布式版本库的方式，不必需要服务器端软件支持。
  - 任务：在Linux上安装某个版本的Git软件，试着创建和管理一个版本库，并熟悉Git的各种操作。

# 动手实践2

- Github是由克里斯•万斯特拉斯（Chris Wanstrath）、海伊特（PJ Hyett）与汤姆•普雷斯顿•沃纳（Tom Preston-Werner）三位开发者在2008年4月创办，主要提供基于Git的版本托管服务。GitHub目前已经成为最好用的免费开源项目托管站点。
  - 任务：在Github的官方网站上注册一个账号，然后通过实际的项目熟悉Github的各种操作。

# Thanks!

