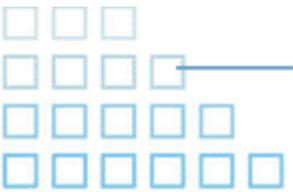




服务计算 Service Computing

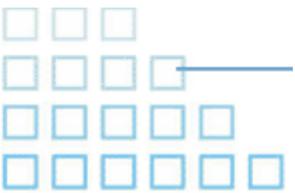
王旭

wangxu@act.buaa.edu.cn





一、服务计算概述





提纲

- 产生发展历程
- 释义和现状
- 核心理念
- 技术体系
- 发展趋势



提纲

- **产生发展历程**
 - 第一代服务计算技术
 - 第二代服务计算技术
- **释义和现状**
- **核心理念**
- **技术体系**
- **发展趋势**



Gartner首次提出SOA

- 早在1996年，Gartner就首次提出了SOA（Service-Oriented Architecture，即面向服务的架构）的概念，并预言SOA将成为下一代软件的革命性技术
- 但因为当时缺乏实现SOA的技术基础，SOA并没有立即引起企业用户和IT公司的重视
- 直到后来XML、SOAP、WSDL、UDDI等Web服务标准逐渐成熟，SOA才真正成长为可部署的技术、产品和下一代应用系统的方法论，开始被业界广泛接受，进入了部署期。

Gartner研究报告

"Service Oriented" Architectures, Part 1

12 April 1996

W. Roy Schulte Yefim V. Natis



比尔·盖茨: Microsoft .NET Today

June 14, 2001

- 大多数业务信息都存储在集中式的数据库中，在需要时，向个人用户一次提供一个页面。糟糕的是，这种页面**只是一种数据的“图画”，而不是数据本身**，迫使开发人员去分析抓取“屏幕”的信息。
- 这种孤立的应用和 Web 网站创建的是**功能和数据的孤岛**，解决这些问题是对下一代 Internet 的关键性挑战
- 解决方案的**核心是**可扩展的标注语言（eXtensible Markup Language, or **XML**。）在以XML为基础的技术（如SOAP、UDDI等）的帮助下创建一类新的软件。



比尔·盖茨：明日工作新世界

2005年《计算机世界日报》

- 由于XML和丰富Web services的出现，企业能够日益与合作伙伴**无缝分享信息和过程**，并建立一些虽然跨越多个机构但作为一个统一整体工作的供应链。
- 基于XML Web services标准的.NET 开发平台可以**实现个人之间，个人与企业之间，和企业之间的信息互连**，这样就实现人们可以随时随地存取和使用信息的梦想。



各软件公司纷纷提出对策

- Microsoft : 微软宣布推出Windows .NET Server RC1, 预见下一代平台技术浪潮
- Sun: 面向服务的体系结构因其固有的松散耦合与互操作性, 成为许多企业应用的自然选择。使用 J2EE 提供的 Web 服务功能可以很容易地构建能够访问现有业务流程的 SOA 系统
- IBM: 2001年开始, 将动态电子商务(Dynamic e-Business)的理念转向Web Services
- Oracle: 应用程序网格计算的特征和优点恰恰是面向服务体系结构(SOA) 的特征和优点

IT理念殊途同归

- Veritas强调，在“效用计算”模式下，可用性将使数据和应用“永远在线”，从而确保最终用户随时都可以获得服务；
- BEA 指出“流体计算”能帮助企业将IT响应时间从几个月缩短为几分钟，其最终目标是使实时的业务目标与企业的每一步变化紧密结合在一起；
- Microsoft比尔盖茨表示，在“无缝计算”的世界里，人们可以实现在任何时候、任何地点、任何设备得到任何想要的信息。
- IT厂商理念虽然层出不穷，按需应变、效用计算、流体计算、无缝计算、实时企业、动成长企业等等理念看似复杂、毫不相干，但是却都透着面向服务、整合资源、按需分配、敏捷响应共性。
- 面向服务架构（SOA）是实现上述理念的技术基础。

提出SOA的具体原因

- (1) Internet环境下的企业交互
- 现代企业已经不再是封闭的企业，市场分工的日益专业化使得企业之间可能存在大量频繁的交互行为，以发挥各自的竞争优势：
 - 供应链：供应商-制造商；
 - 客户关系管理：制造商-物流商-客户
- 这种业务上的交互体现为企业业务流程的交互/互操作，同时一定需要企业信息系统的的支持，因此体现为软件系统之间的集成与互操作
 - 互操作(Interoperability)：能够在异构的、分布式的系统之间交换和使用信息的能力；
 - 不仅是不同企业之间，甚至一个企业内部的各个部门之间都有可能存在大量的交互。

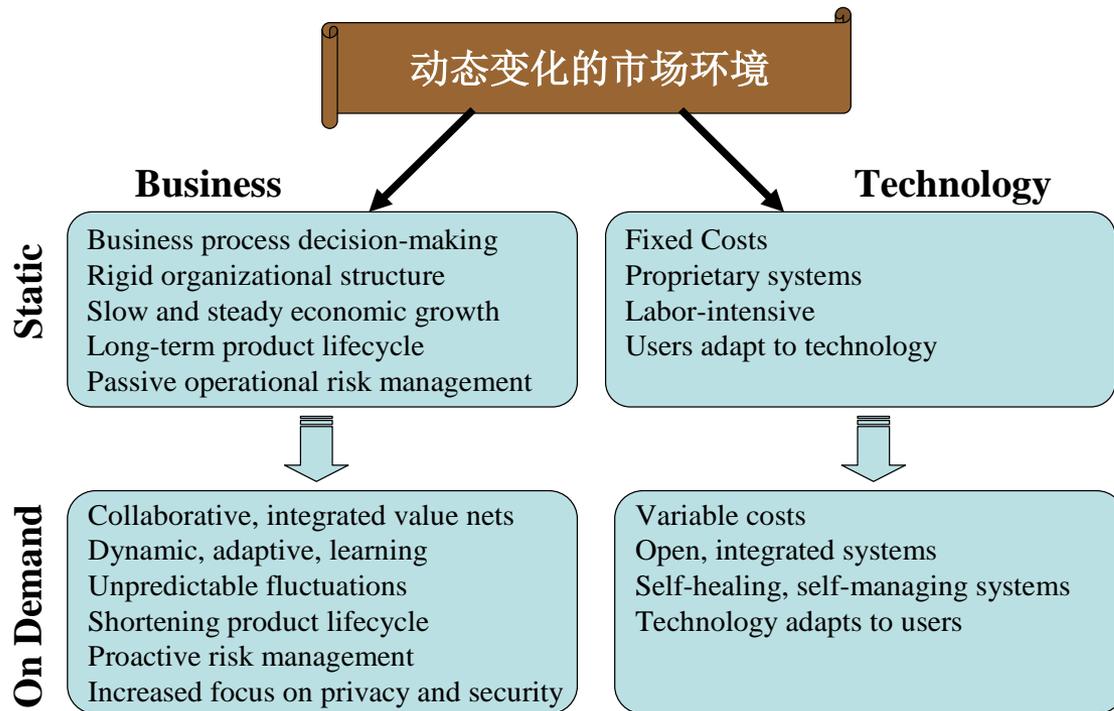
提出SOA的具体原因

- (2) 异构系统的集成与互操作
- 不同企业所应用的软件系统是不同的(异构的):
 - 技术平台不同: J2EE-based、.Net-based
 - 软件体系结构不同: message-based、file-based、process-based
 - 数据格式不同: 同样的“订单”对象, 不同的属性集合
 - ...
- 集成这些分布式的软件系统, 在它们之间传递数据和消息, 是一件非常困难的事情



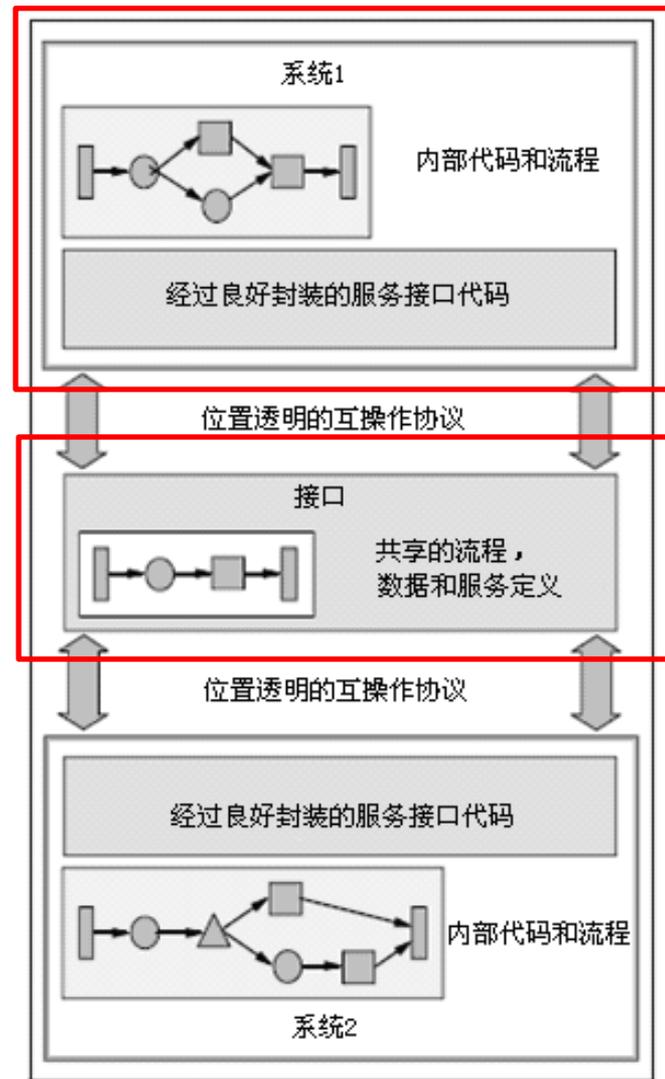
提出SOA的具体原因

- (3) 频繁变化的互操作与集成需求
- 企业的业务是频繁变化的
- 企业间的协同关系也不是固定的，随着业务流程的变化而随之变化
- 企业的IT应用系统要能够快速支持这种变化的需求



归纳：SOA所要解决的问题

- 分布式企业间业务的协同。
- 通过Internet连接在一起的异构企业应用软件系统的集成、交互与互操作。
- 当业务(Business)发生变化时，IT系统能够快速响应。

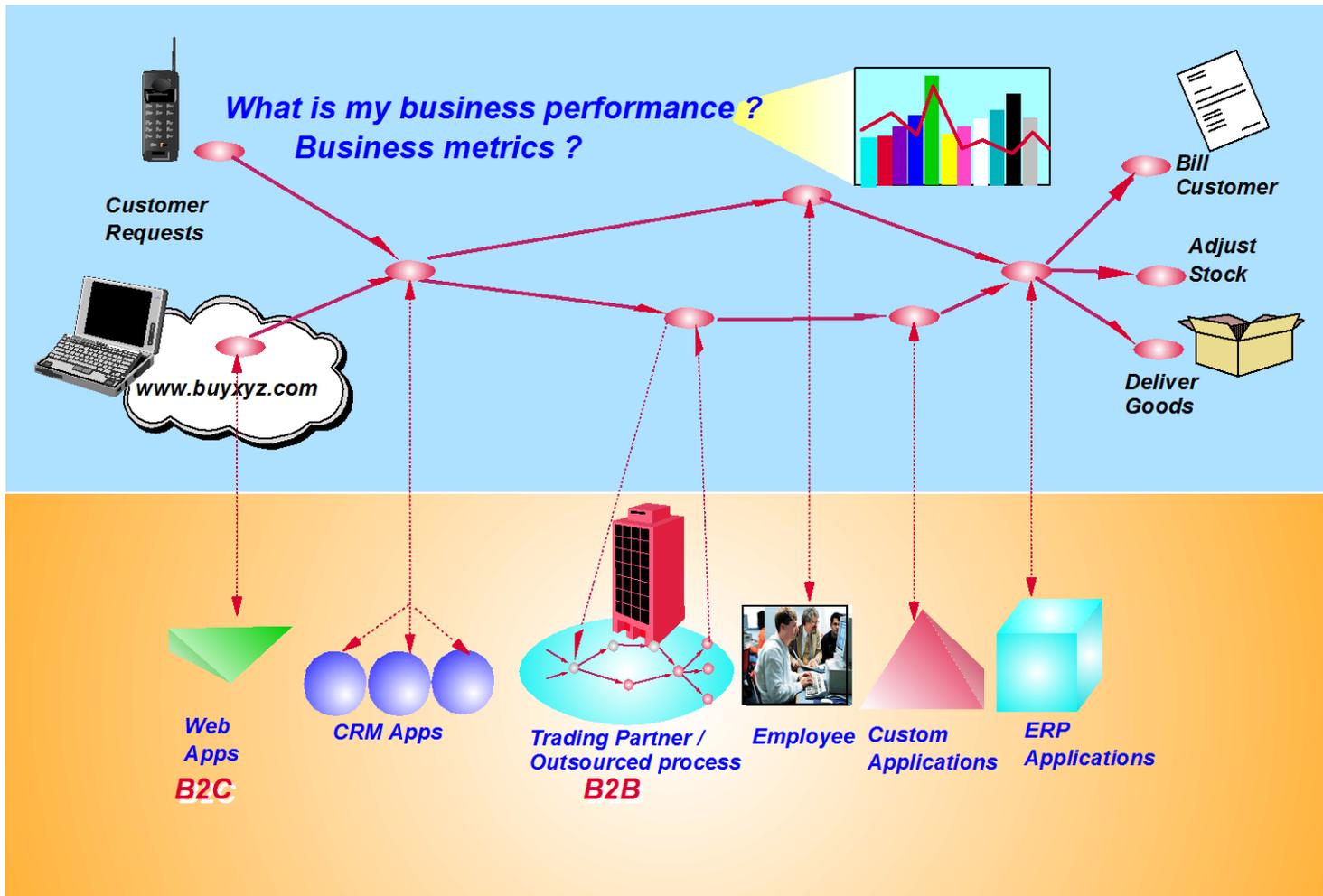




敏捷业务 - SOA 的首要目的; SOA - 敏捷业务的实现之道 (水平整合、灵活应变的业务流程; 可重用、可组装的 IT 资产)

做什么

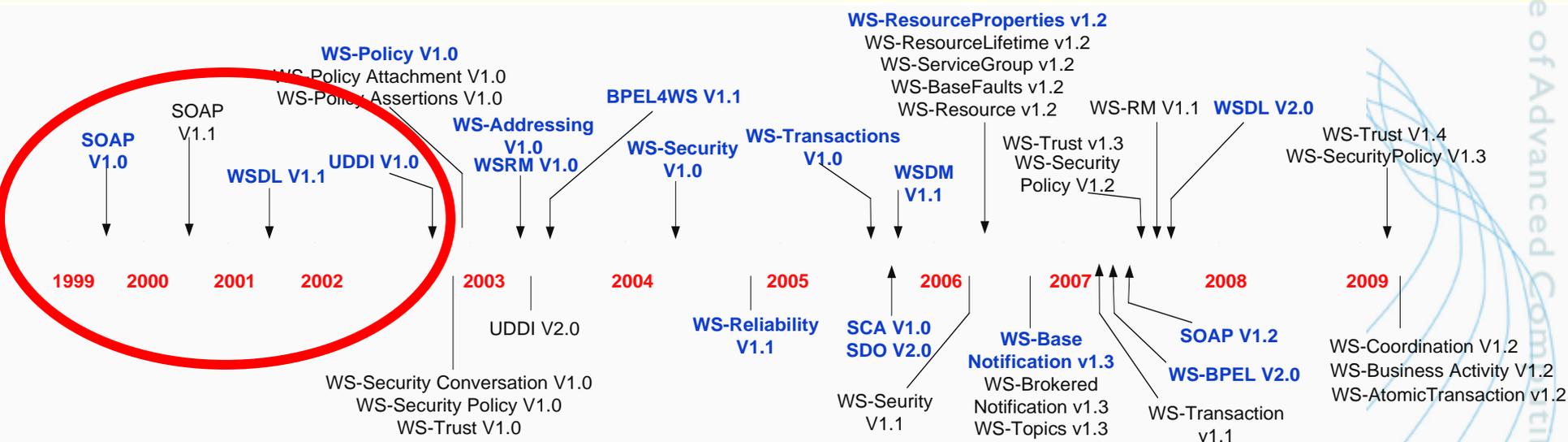
怎么做



敏捷业务

可重用IT资产

Web服务产生及发展





服务计算产生及发展

- 2002年6月，美国拉斯维加斯举行国际互联网计算会议(International Conference on Internet Computing)
 - 张良杰承办Web服务计算(Web Services Computing)专题讨论，首次将服务与计算结合，强调Web服务在分布式计算和动态业务集成中的重要作用
- 2003年6月，美国拉斯维加斯举行第一届Web服务国际会议(International Conference on Web Services)
- 2003年11月，IEEE成立服务计算技术社区(Technical Community of Services Computing)
- 2004年5月，改名为服务计算技术指导委员会(Technical Steering Committee for Services Computing)
 - 张良杰先生出任委员会的首任主席
 - 致力于推动服务计算学科发展和相关标准制定
 - 标志着服务计算正式成为一门独立的计算学科
- 2004年9月，服务计算技术指导委员会在上海召开第一届服务计算专题国际会议(IEEE International Conference on Services Computing)

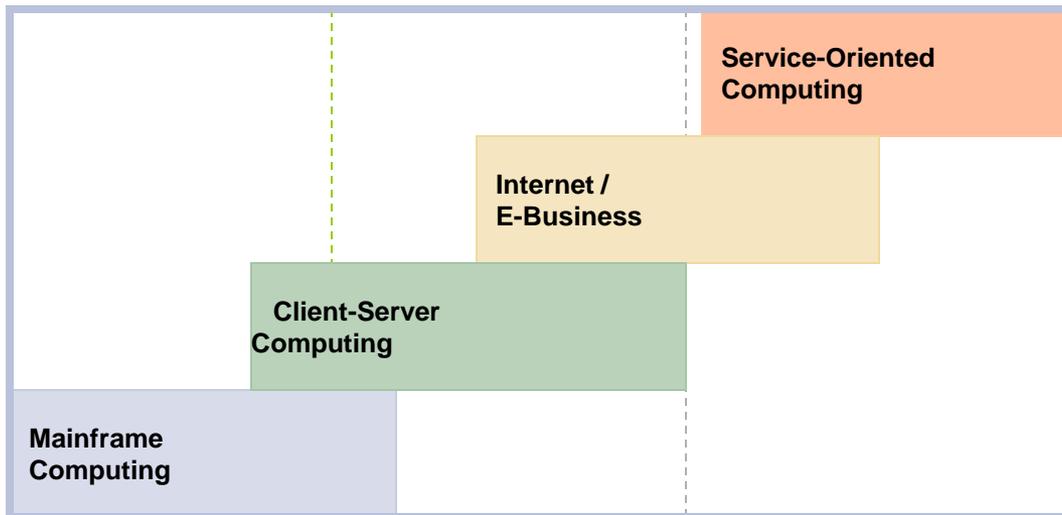


服务计算带来分布式计算演进

1980's

1990's

2000's



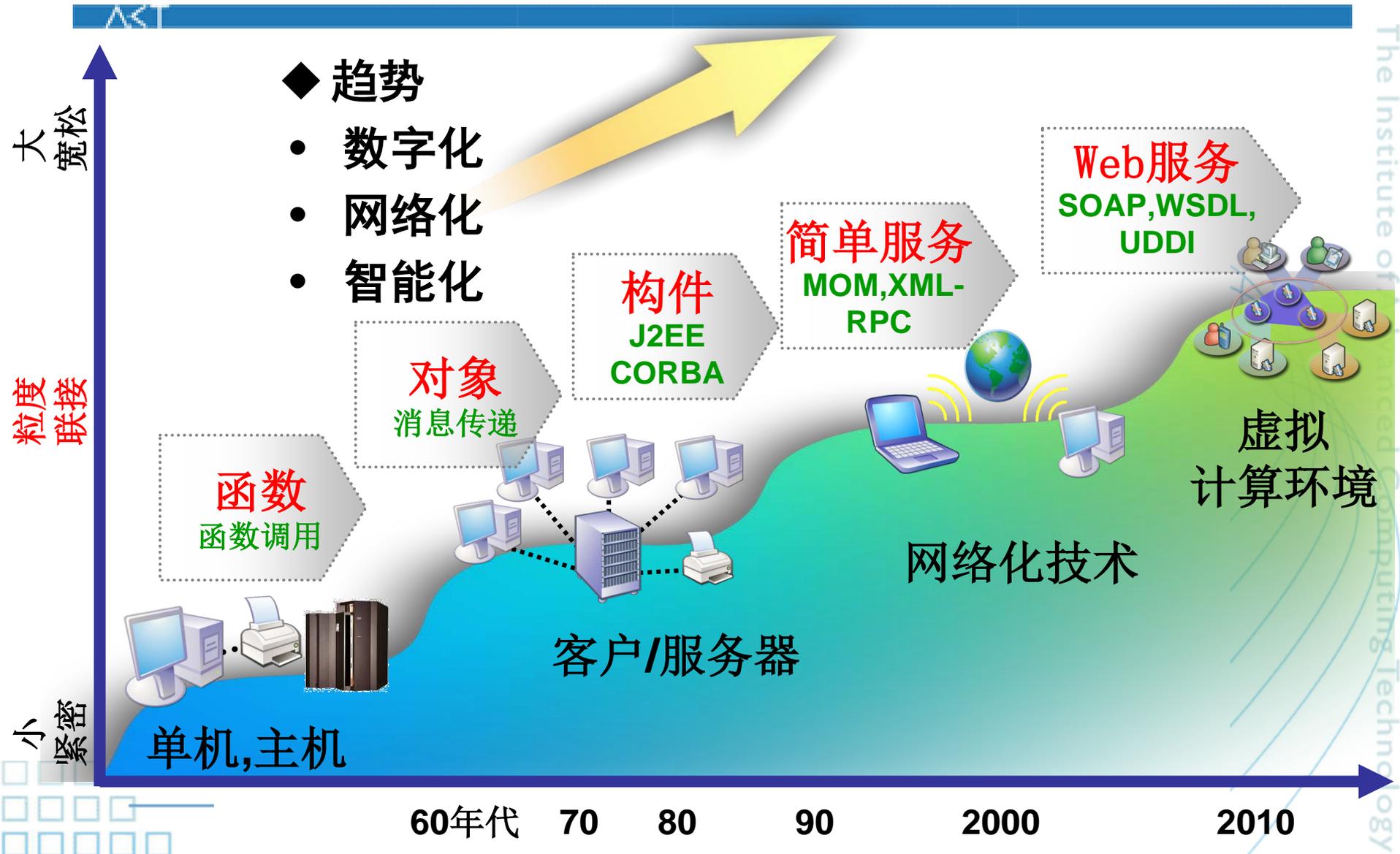
Return on Investment
 Interoperability / Flexibility
 Time to Market
 Infrastructure Leverage

Period Characteristics

- | | | |
|---|--|---|
| <ul style="list-style-type: none"> • Packaged & Custom Mainframe Applications • Top-Down, Procedural Development • Centralized Computing Models • Non-Distributed Solutions • Emerging PC-Based Technologies | <ul style="list-style-type: none"> • Client-Server Applications • Object-Oriented Development • Enterprise-Centric Computing • Internet Proliferation • E-Commerce Applications | <ul style="list-style-type: none"> • Enterprise Web Applications & Suites • Web Services & Software Components • Next-Generation Integrations • Real-Time Application Assembly • Rapid Deployment & Management |
|---|--|---|



服务计算带来中间件技术演进



The Institute of Advanced Computing Technology



提纲

- 产生发展历程
 - 第一代服务计算技术
 - 第二代服务计算技术
- 释义和现状
- 核心理念
- 技术体系
- 发展趋势

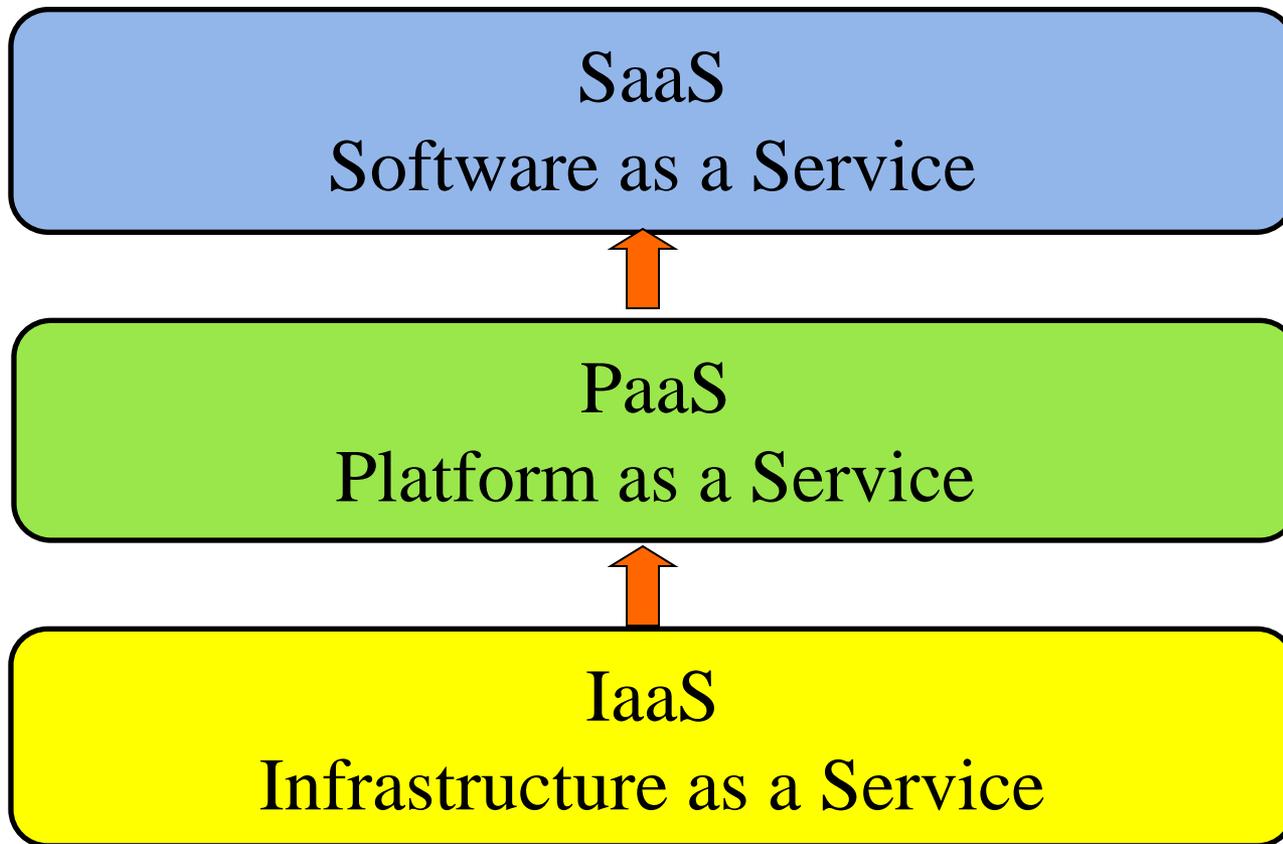


云计算技术的发展

• 云计算

- **2007.10**, **Google**与**IBM**开始在美国大学校园, 包括卡内基美隆大学、麻省理工学院、斯坦福大学、加州大学柏克莱分校及马里兰大学等, 推广云计算的计划。
- **Amazon**使用弹性计算云(**EC2**)和简单存储服务(**S3**)等为企业提供计算和存储服务。
- **Google**推出**App Engine**, **Doc**, **Web Mail**等云服务和**GFS**, **Map/Reduce**, **BigTable**等底层技术来推动云计算。
- **IBM** 在**2007年11月**推出包括一系列云计算产品在内的**蓝云(Blue Cloud)**。
- 微软在**2008年10月**也推出了**Azure**云服务平台, 包括了**Windows Azure**, **SQL**数据库服务, **.Net**服务等主要组件。
- 同时, **Yahoo**、**Salesforce**、**Facebook**、**SUN**、**HP**、**Oracle**、**EMC**、**Rackspace**等等公司也先后推出云计算相关产品

云服务的类型



laaS

■ laaS: 基础设施即服务

- 以网络服务的形式对外提供基础设施，包括计算、存储和数据等

■ 典型系统

■ Amazon



▶ Infrastructure web services

- EC2 (Elastic Compute Cloud) - now with Windows (99.95% availability!)
- S3 (Simple Storage Service)
- SimpleDB
- SQS (Simple Queue Service)

■ Rackspace

- ▶ Mosso
- ▶ JungleDisk
- ▶ SliceHost

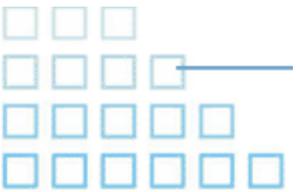
■ 阿里云





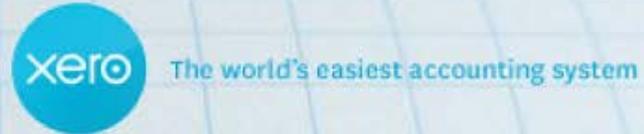
PaaS

- **Google Apps**
 - ▶ Python only
 - ▶ BigTable
- **Heroku**
 - ▶ Ruby on Rails hosted on EC2
- **Force.com**
 - ▶ Apex
- **Microsoft!**
 - ▶ Azure - .NET hosted in MS datacentres
- **华为PaaS平台**





SaaS

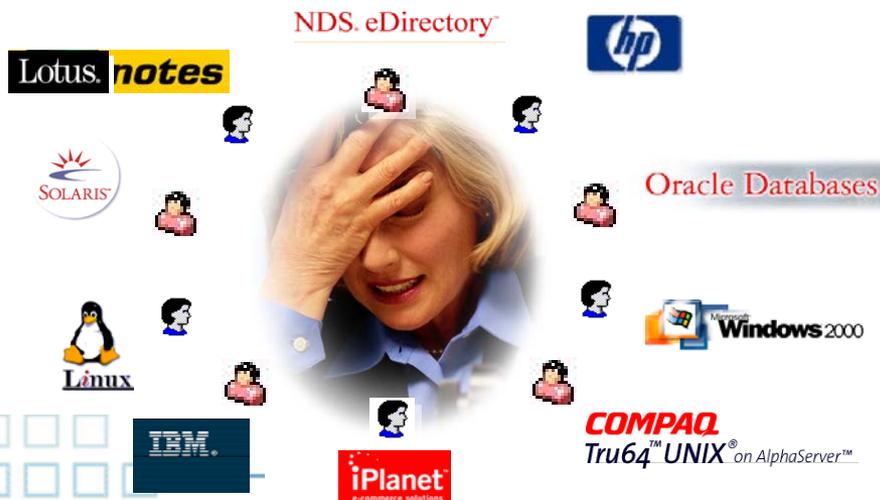


软件服务带来的便利

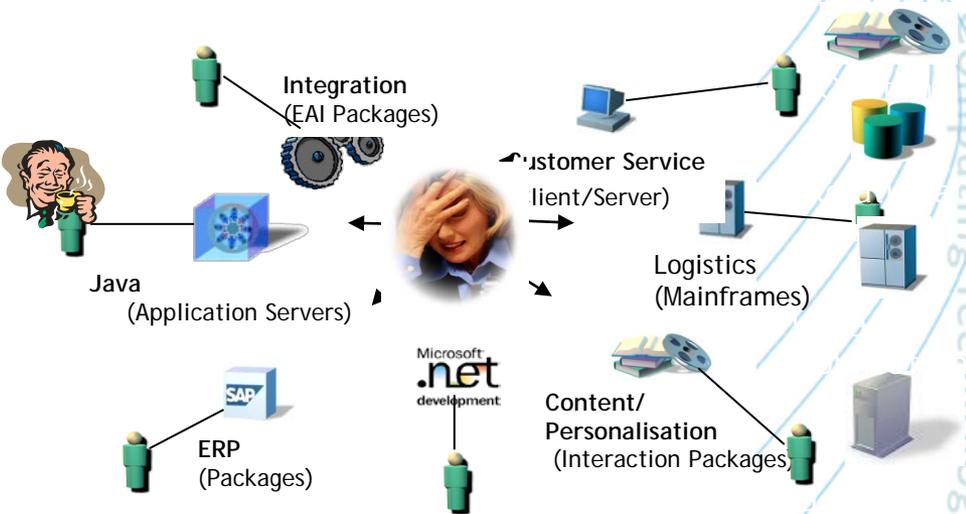


软件职责

面临众多的IT技术



面临众多的IT产品



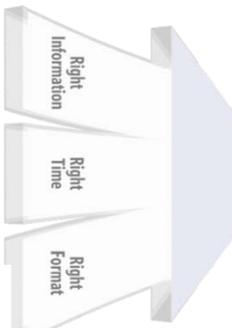
The Institute of Advanced Computing Technology

软件服务带来的便利



软件职责

软件提供商





第一代Web服务的缺陷：笨重

- 技术实现复杂：XML and SOAP
 - XML解析处理和转换
 - SOAP协议支持
- 性能低
 - 每一次调用都需要处理XML and SOAP
- 大范围应用受限
 - 由于前两者的原因，没有在Web上大范围应用



RESTful Service: 轻量级

- HTTP-based
 - 直接使用HTTP/HTTPS协议，利用已有的Web技术发展成果
- JSON或XML，大部分情况下是JSON
 - 使用工业界广泛接受的数据传输技术
 - 数据容量小
- 技术体系成熟
 - 直接扩展已有成熟的Web技术栈
 - 比如Java Spring等

RESTful Services

- REST(*Representational State Transfer*):表述性状态转移, 分布式超媒体软件的一种架构风格。源自2000年*Roy Thomas Fielding*的博士论文。
- 一套简单的设计原则、一种架构风格(或模式), 不是一种具体的标准或架构。
- 基于使用HTTP、URI等现有的广泛流行的协议和标准, 并由几个核心抽象概念支撑。
- 对Web的本质回归: 一种真实描述Web的方式, 不被特定时期的特定应用程序概念歪曲。



REST设计准则

- 网络上的所有事物都被抽象为资源
- 每个资源对应一个唯一的资源标识URI
- 通过HTTP协议方法作连接器对资源进行操作
- 对资源的任何操作不改变资源标识URI
- 所有的服务器操作都是无状态的



直观的区别：Web服务

- SOAP- Web服务例子

POST search/beta2 HTTP/1.1

Host:api.google.com

Content-Type:application/soap+xml

SOAPAction: urn:GoogleSearchAction

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<soap:Envelope
```

```
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```
  <soap:Body>
```

```
    <gs:doGoogleSearch xmlns:gs="urn:GoogleSearch">
```

```
      <q>REST</q>
```

```
      ...
```

```
    </gs:doGoogleSearch>
```

```
  </soap:Body>
```

```
</soap:Envelope>
```



直观的区别: RESTful 服务

- RESTful Service例子

GET

services/rest?api_key=xxx&method=flickr.photos.search&tags=penguin HTTP/1.1

Host:www.flickr.com

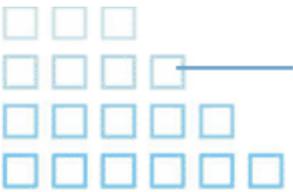
...

GET

member/corporation/crpHome!ListByUserId.jspa
HTTP/1.1

Host:member.alisoft.com

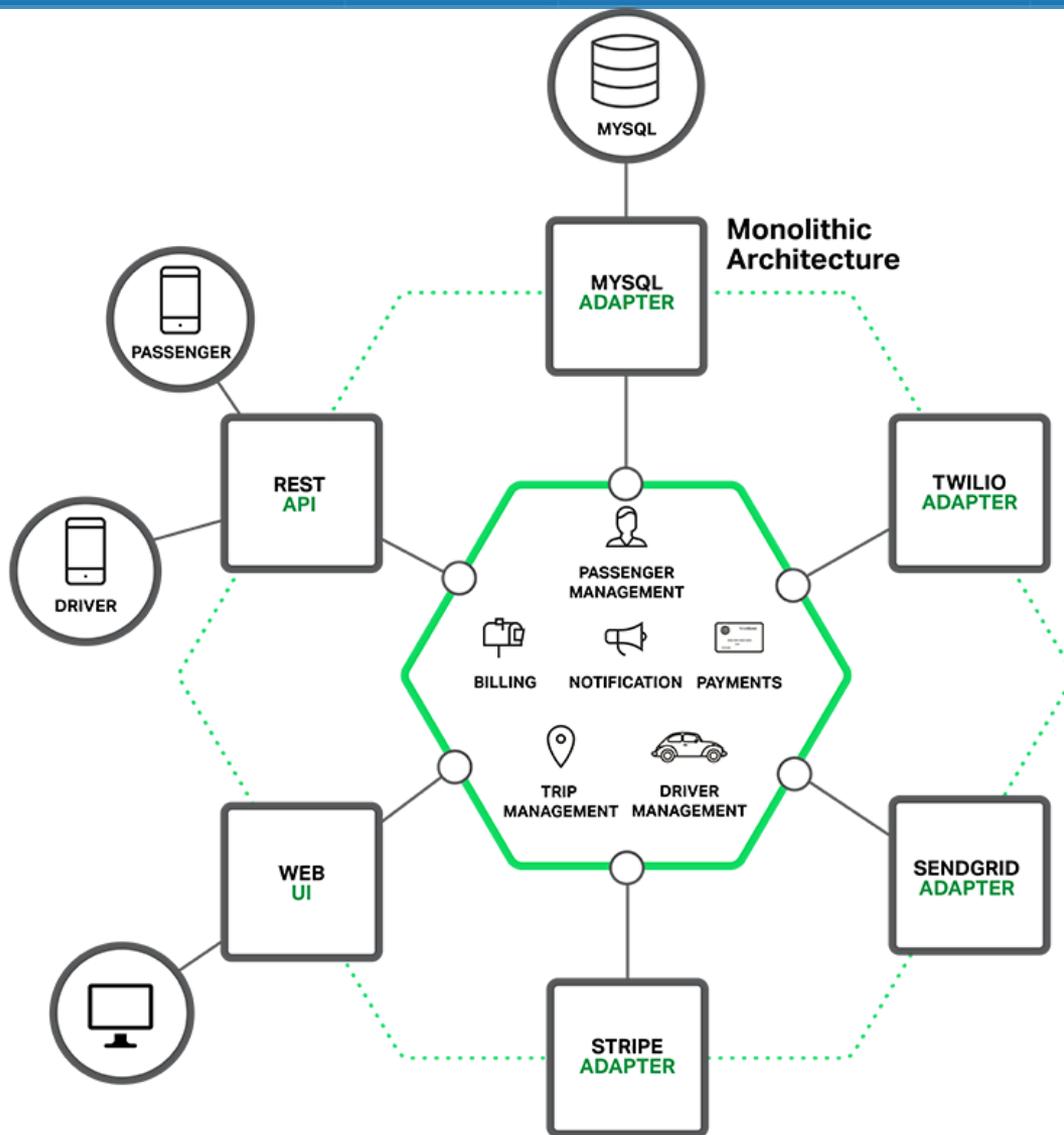
...



微服务

- **业务互操作→微服务：更细粒度**
 - **Web**服务强调不同企业或者不同业务之间的互操作，而微服务强调单一业务系统内部也采用服务架构
- **驱动力**
 - 在云服务背景下，单一业务系统有较大的负载和可用性需求
 - 单一业务系统本身也有模块重用的需求
 - 单一业务系统中不同模块的执行效率、资源消耗不一样，有些可能是性能瓶颈

微服务例子



微服务特点

- 一个微服务一般完成某个独立的功能，比如订单管理、客户管理等，服务间可以并行开发
- 每个微服务都是一个微型应用，有着自己的业务逻辑和各种接口
 - 有的微服务通过暴露 **API** 被别的微服务或者应用客户端所用
 - 有的微服务则通过网页 **UI** 实现
 - 运行时，每个实例通常是一个云虚拟机或者 **Docker** 容器
- 每个微服务之间通过API进行集成，单个微服务可以进行单独进行横向扩展



小结

• 第一代和第二代服务计算技术的比较

– 相同点

- 基于网络服务实现功能分割，每个组件可以通过网络单独调用并且单独运行互不干扰
- 复杂业务服务通过对组件服务的集成来实现

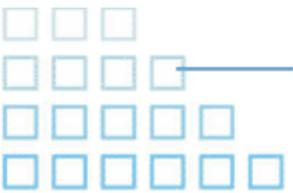
– 不同点

- Web服务基于XML和SOAP协议实现，效率低，有较好的自描述性和可理解性
- RESTful服务和微服务一般基于HTTP和原始数据进行远程调用和参数传递，轻量级
- Web服务强调不同业务系统间的业务集成和互操作，这些服务的管理域不同；而微服务更强调单一业务系统内部的组件服务分割以及集成，其组件服务一般是由同一个组织管理



提纲

- 产生发展历程
- **释义和现状**
- 核心理念
- 技术体系
- 发展趋势



什么是“服务计算”

- 荷兰科学家Mike P. Papazoulou
 - 从软件系统设计与开发的角度出发，认为“服务计算是一种以服务为基本元素进行应用系统开发的方式”
 - M. Papazoglou and D. Georgakopoulos. Introduction to a special issue on service oriented computing, Communication of ACM, 2003: p.25- 28





什么是“服务计算”

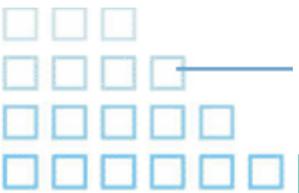
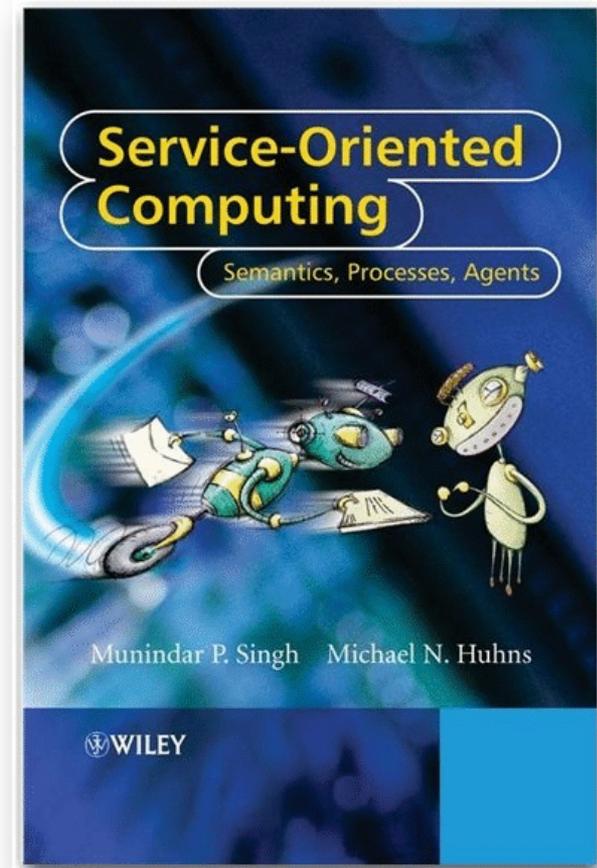
- 华人科学家 张良杰

- 从学科的角度出发，认为“服务计算是一门跨计算机与信息技术、商业管理与咨询服务的基础学科，其目标在于利用服务科学和服务技术消除商业服务和信息技术服务之间的鸿沟”



什么是“服务计算”

- 北卡罗来纳州立大学Munindar P. Singh
- 南加州大学Michael N. Huhns
 - 从**服务技术的应用**角度出发，认为“**服务计算是集服务概念、服务体系架构、服务技术和**服务基础设施**于一体，指导如何使用服务的**技术集合**”**



什么是“服务计算”

- 从**分布式计算的角度**出发，认为“服务计算是从面向对象和面向构件的计算演化而来的一种**分布式计算模式**，它使得分布在企业内部或跨越企业边界的不同商业**应用系统**能实现**快捷、灵活的无缝集成与相互协作**”
 - Orłowska, M.E., Weerawarana, S., Papazoglou, M.P., and Yang, J. First International Conference Service-Oriented Computing (ICSOC 2003), Trento, Italy, December 15-18, 2003.

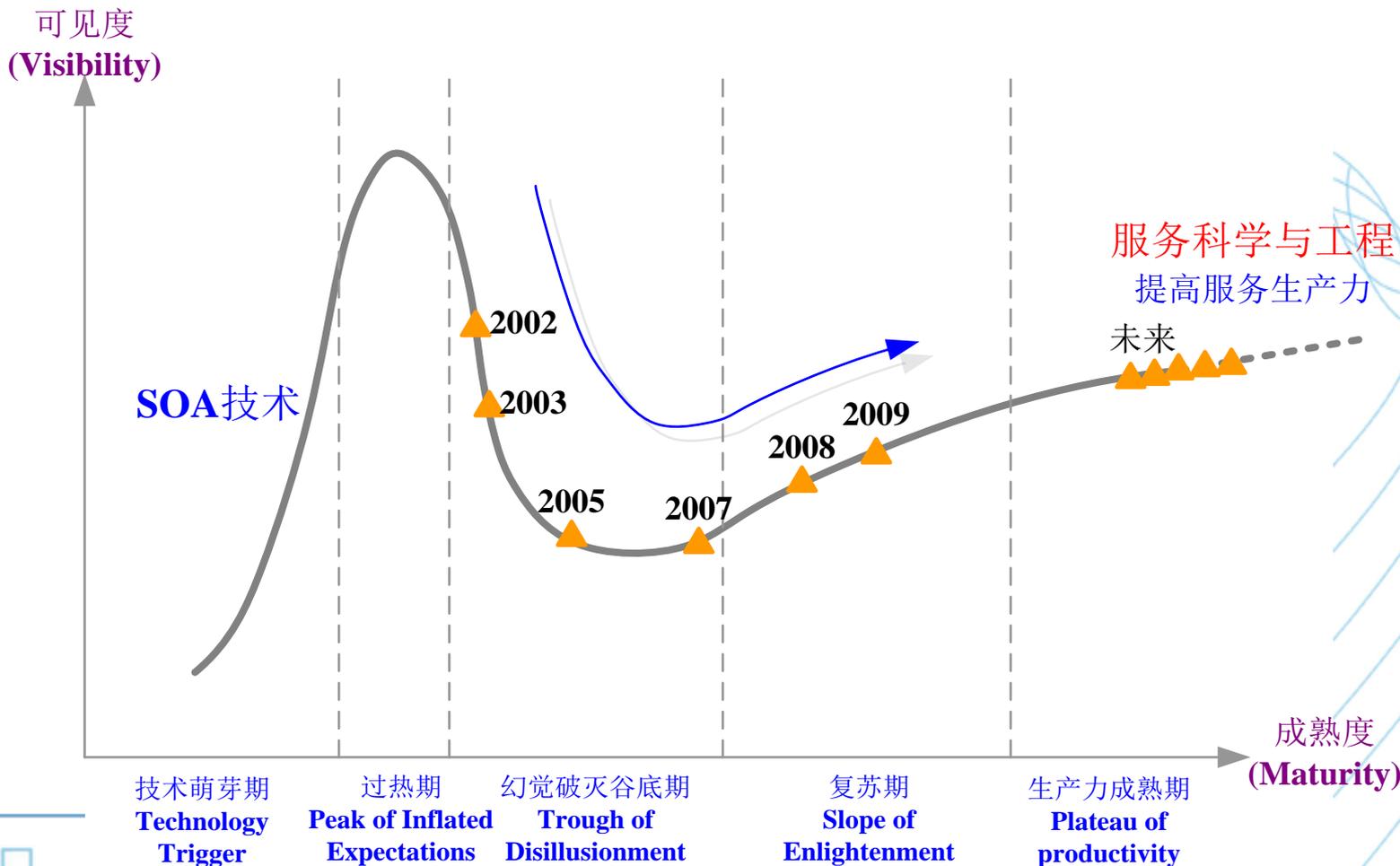
服务计算

- 它在形成自己独特的科学与技术体系的基础上有机整合了一系列最新技术成果
 - **SOA(Service Oriented Architecture, 面向服务的体系架构)及Web服务**
 - **REST服务和微服务架构**
 - **网格/效用计算(Grid & Utility Computing)**
 - **业务流程整合及管理(Business Process Integration & Management)**
 - 第一部分解决的是技术平台和架构的问题；第二部分解决是服务交付的问题；第三部分则是业务本身的整合和管理。



SOA/Web服务技术成熟度趋势

Gartner新兴技术光环曲线 (Gartner Hype Cycle for Emerging Technologies)





SOA软件产业：Web服务

最活跃的SOA标准组织：

- W3C - World Wide Web Consortium
- OASIS
- WS-I – Web services Interoperability Organization
- JCP – Java Community Process
- OSOA – Open SOA Collaboration

最活跃的SOA相关公司：

- Oracle BEA
- IBM Microsoft
- SUN SAP

开源组织：

- ObjectWeb
- Apache(Axis)
- JBoss



SOA软件产业: 微服务

- 服务开发层



- 数据层



- 容器层



学术界现状

• 刊物

- IEEE Transaction on Services Computing(TSC)
- International Journal of Web Services Research(JWSR)
- International Journal of Web and Grid Services(IJWGS)

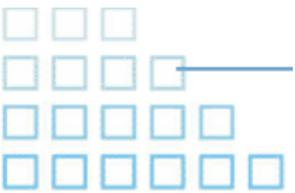
• 会议

- WWW
- IEEE International Conference on Services Computing(SCC)
- IEEE International Conference on Web Services(ICWS)
- IEEE World Congress on Services
- International Conference on Service Oriented Computing(ICSOC)



提纲

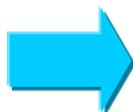
- 产生发展历程
- 释义和现状
- **核心理念**
- 技术体系
- 发展趋势



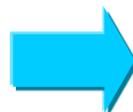
服务的故事之咖啡的故事



农业时代，咖啡是自给自足的



工业时代，咖啡馆，每杯2元

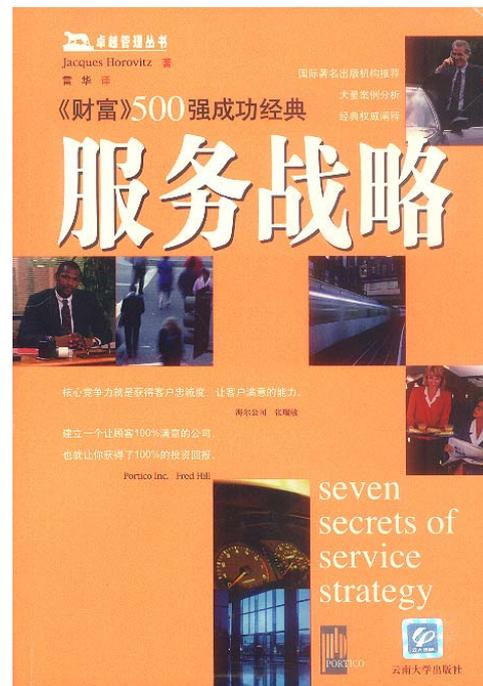
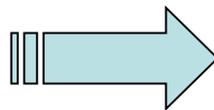


星巴克，体验经济的咖啡，每杯25元

- 星巴克不是只卖咖啡的，它更是提供工作、生活之外社交聚会服务的地方：25元/杯的咖啡=咖啡5元+体验轻松氛围20元
- 所谓体验，就是企业以服务为舞台、以商品为道具，环绕着消费者，创造出值得消费者回忆的活动。核心就是活动
- 体验经济的表象似乎是出售产品，但其本质则是出售与消费者一齐度过的时光

什么是服务?

- 服务->服务业、现代服务业
- 服务->SSME、SS、SSE、ES
- 管理学家、经济学家、工程学家、信息家、法学家、心理学家、人文学家、政治学家的不同理解





服务的定义

- 服务是为满足顾客的需要,在与顾客的接触中,服务提供者的活动过程和活动的结果。

ISO9000

- 服务是行动、过程和表现

Zeithamland Bitner

- 包括所有产出为非有形产品或构建品的全部,通常在生产时被消费、并以便捷、愉悦、省时、舒适或健康的形式提供附加价值

Scientific American

- “服务”被定义为在一个应用软件内部的一种方法、过程、或通讯。这些“服务”或“方法”是旨在满足某些商业需求的应用程序的操作。

—— IT的说法

Elizabeth Chang

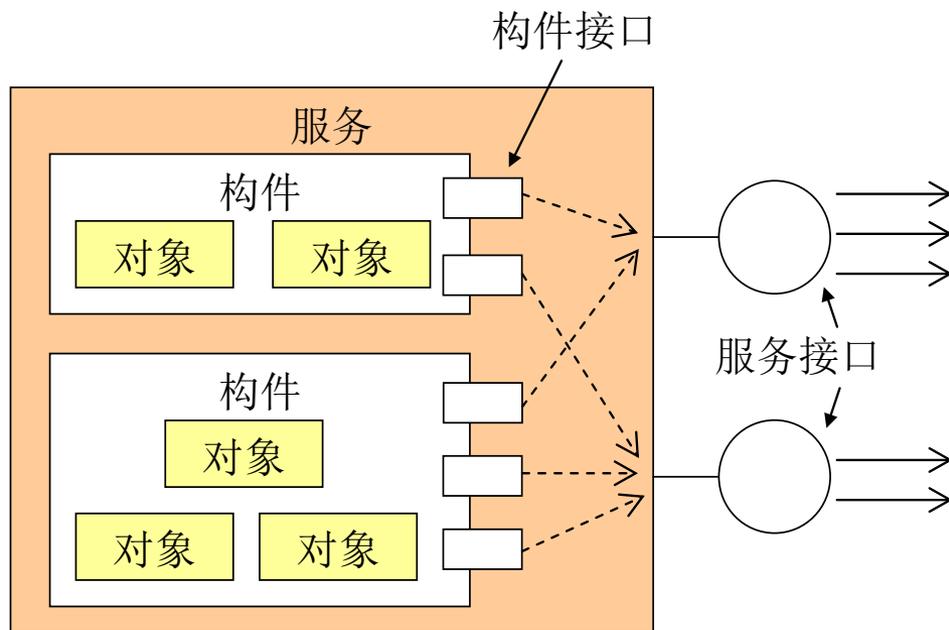


服务计算中的服务概念

- 函数、对象、组件的进一步抽象、提升，是整个SOA实现的核心
- SOA的基本元素是服务，SOA 指定一组实体(服务提供者、服务消费者、服务注册表、服务条款、服务代理和服务契约)，这些实体详细说明了如何提供和消费服务
- 自治的、平台独立的计算实体，可被描述、发布、发现、动态组装
- 这些服务是可**互操作的、独立的、模块化的、位置明确的、松耦合的**，并且可以通过网络查找其地址。

服务的含义

- 从外特性上看，一个服务被定义为显式的、独立于服务具体实现技术细节的接口。
- 从内特性上看，服务封装了可复用的业务功能，这些功能通常是大粒度业务，如业务过程、业务活动等。服务的实现可采用任何技术平台，如J2EE、.Net等。



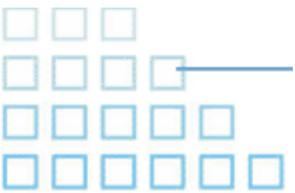
原子服务和组合服务

- 服务的定义是抽象、可迭代的
- 不同的复用和集成粒度对应于不同复杂度的业务功能
- 原子服务和组合服务
 - 都是属于某种服务
 - 当某个复杂服务是由其余多个较小粒度的服务集成而来，一般称该复杂服务为组合服务，而小粒度的服务为原子服务
 - 原子服务是相对概念，并不能说明该服务不能由更小粒度的服务构成



提纲

- 产生发展历程
- 释义和现状
- 核心理念
- **技术体系**
- 发展趋势



服务计算的经典模型

• 服务提供者:

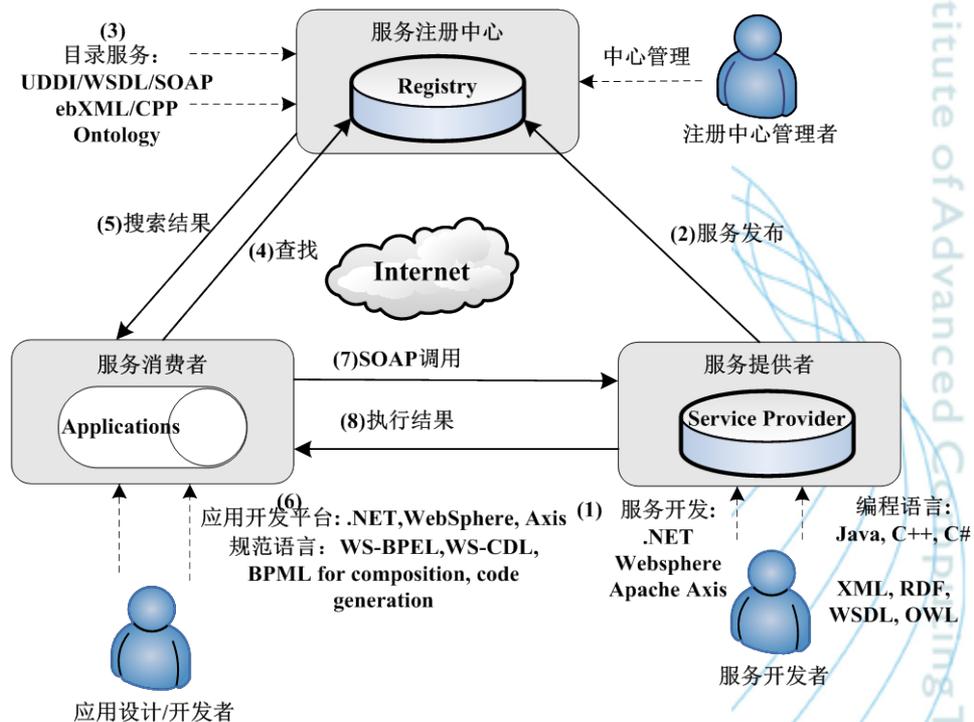
- 一个可通过网络寻址的实体, 它接受和执行来自使用者的请求。
- 它将自己的服务和接口契约发布到服务注册中心, 以便服务使用者可以发现和访问该服务。

• 服务消费者:

- 一个应用程序、一个软件模块或需要一个服务的另一个服务。
- 它发起对注册中心中的服务的查询, 通过传输绑定服务, 并且执行服务功能。
- 服务使用者根据接口契约来执行服务。

• 服务注册中心:

- 服务发现的支持者;
- 它包含一个可用服务的存储库, 并允许感兴趣的服务使用者查找服务提供者接口。





服务计算技术体系

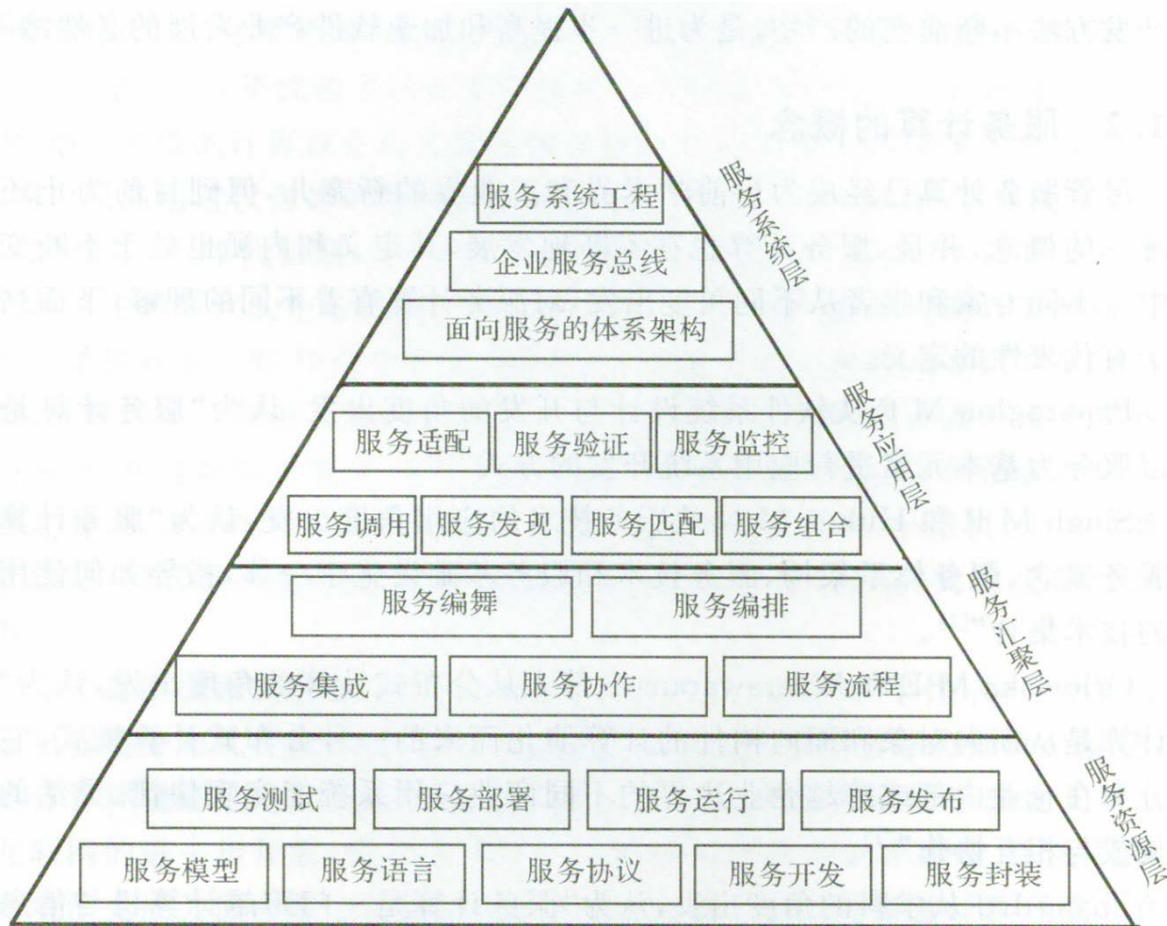
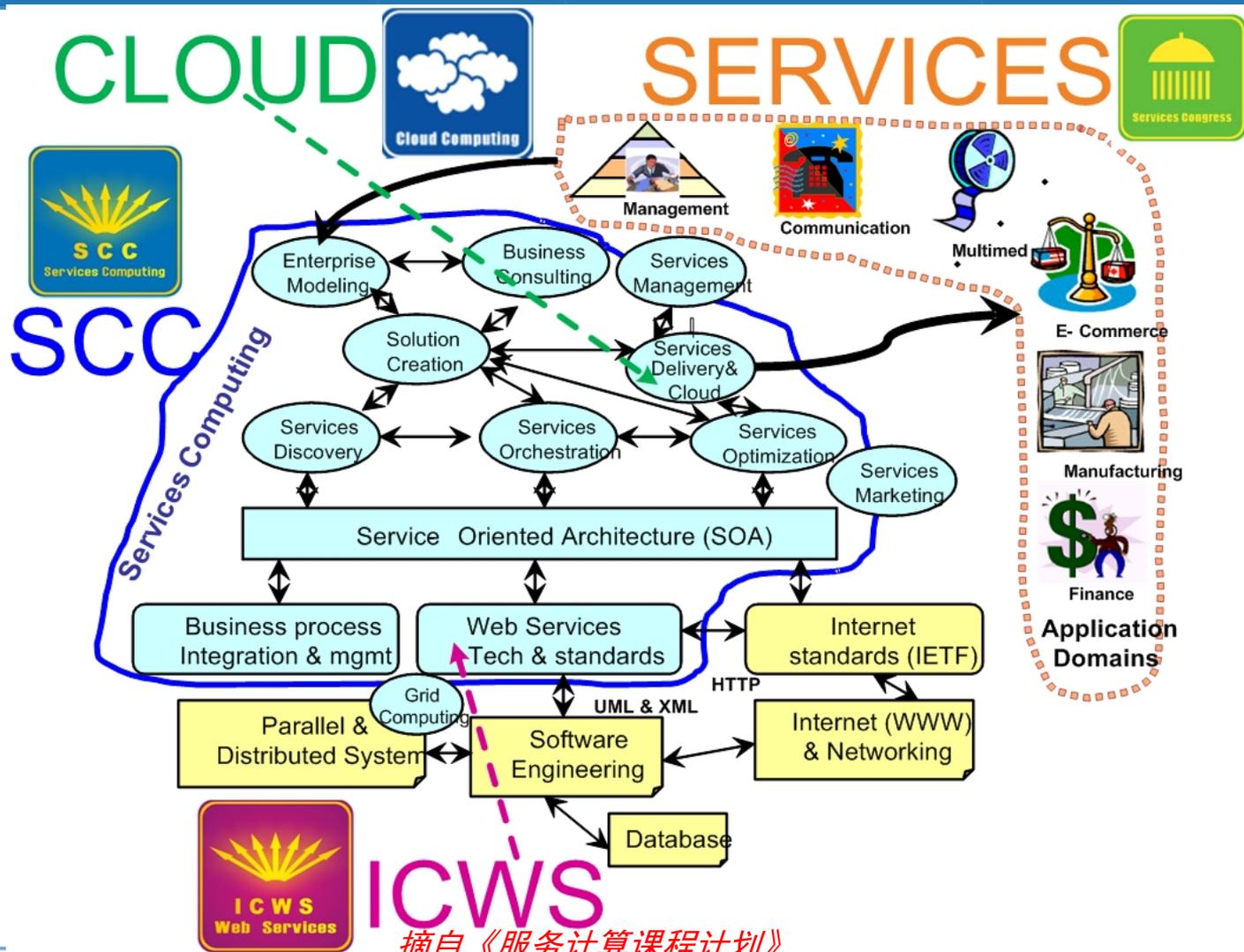


图 1-1 服务计算技术体系

服务计算全景图



摘自《服务计算课程计划》

IEEE计算机学会服务计算技术委员会

<http://servicescurriculum.org/index.php5?title=ServicesCurriculum>



服务计算研究内容

- **IEEE服务计算技术指导委员会给出的研究内容:**
- Mathematical foundation of Services Computing
- Services science
- Service-oriented architecture (SOA)
- Cloud Computing
- Service creation, development, and management
- Linkage between IT services and business services

- Web services security and privacy
- Web services agreement and contract
- Web services discovery and negotiation
- Web services management
- Web services collaboration
- Quality of service for Web services
- Web services modeling and performance management
- Solution frameworks for building Web services applications
- Composite Web service creation and enabling infrastructures



服务计算研究内容

- Business and Scientific applications using Web services
- Business process integration and management using Web Services (e.g., BPEL4WS)
- Multimedia applications using Web services
- Communication applications using Web services
- Interactive TV applications using Web services
- Distance education applications using Web services
- Standards and specifications of Services Computing
- Business Grid covering business process integration and management in Grid environment
- Wireless Grid and communication services delivery platform
- Grid applications and system solutions in science and engineering
- Utility Models and Solution Architectures
- Resource acquisition models in Utility Computing
- Mathematical foundation of Business Process Modeling, Integration and Management
- Business Services modeling
- Ontology and Business Rules
- Business performance analysis and management
- Autonomic Programming Models, Tools, and Environments

服务计算知识体系

- 进一步阅读，请参考：
 - **Introduction to the Body of Knowledge Areas of Services Computing**
 - Liang-Jie (LJ) Zhang, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 1, NO. 2, APRIL-JUNE 2008
 - ***Microservice Architectures for Scalability, Agility and Reliability in E-Commerce***
 - Hasselbring, Wilhelm and Steinacker, Guido (2017) In: IEEE International Conference on Software Architecture 2017, April 03-07, 2017, Gothenburg, Sweden

Research road map

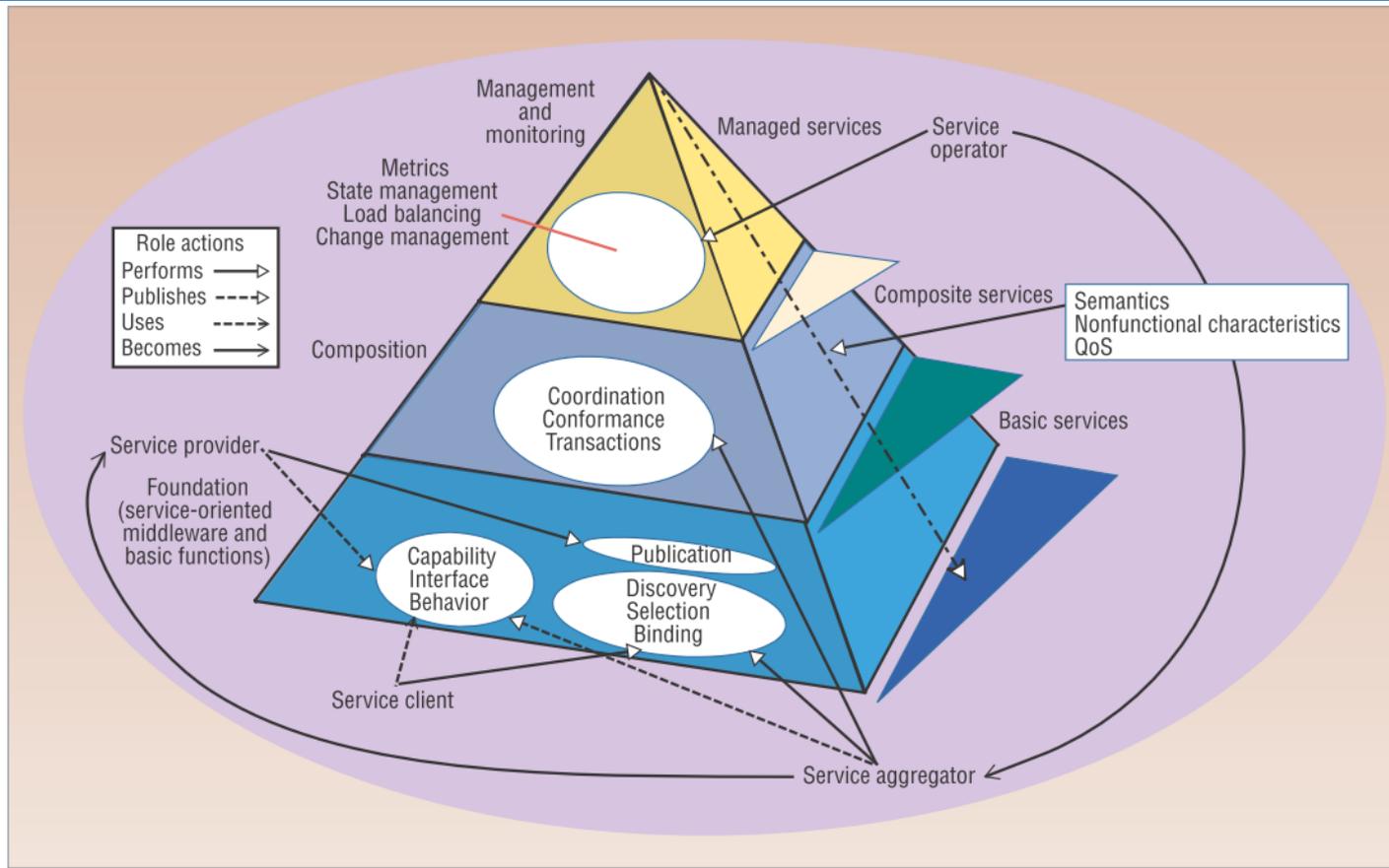
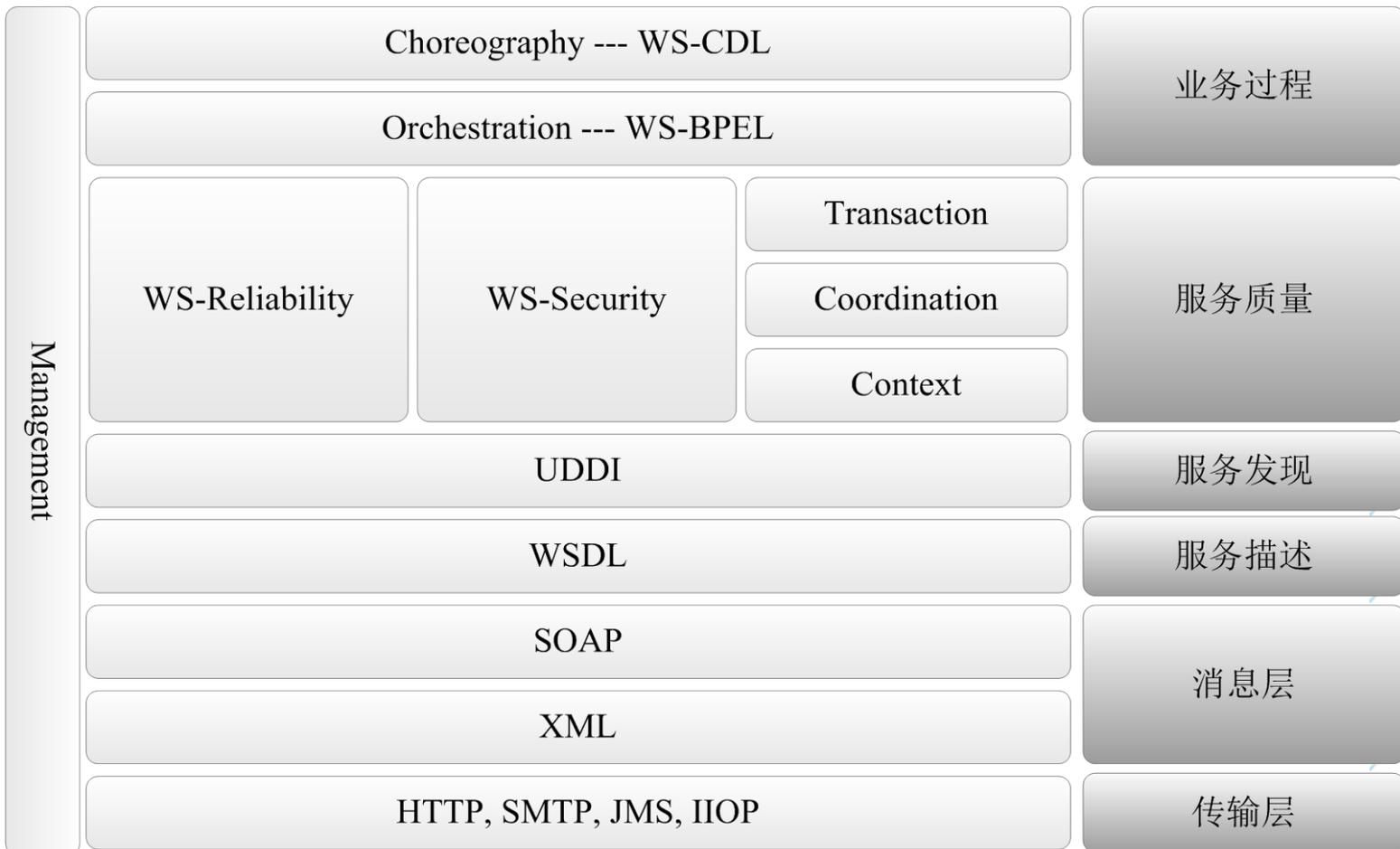


Figure 1. SOC research road map. The architectural layers provide a logical separation of functionality, while the perpendicular axis indicates service characteristics that cut across all three planes.

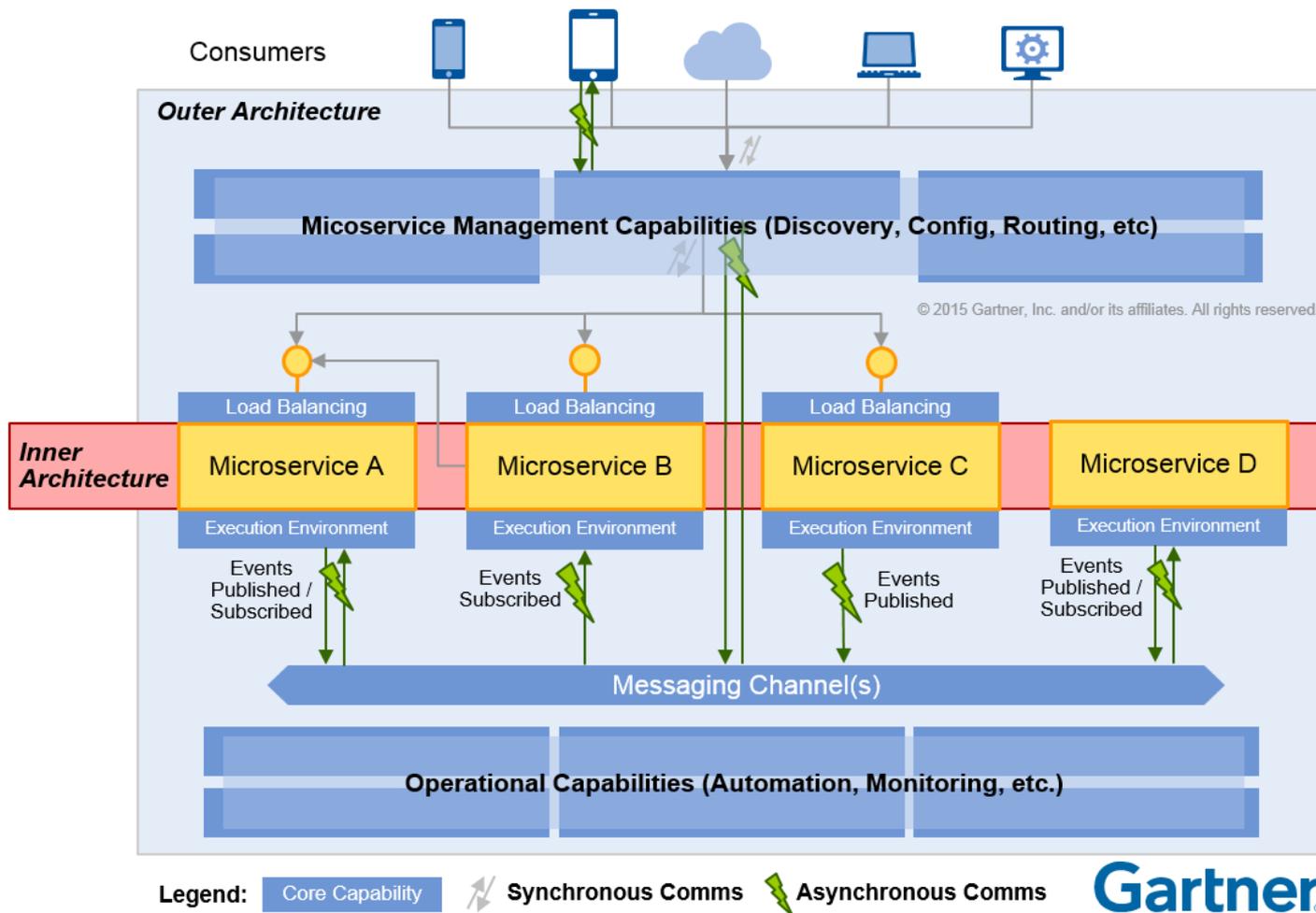
Michael P. Papazoglou, "Service-Oriented Computing: State of the Art and Research Challenges", IEEE Computer, 2007 November



服务计算的一种实现：Web服务



另一种实现：微服务





功能性方面—服务

- **传输协议**：用于将来自服务使用者的服务请求传送给服务提供者，并且将来自服务提供者的响应传送给服务使用者。
- **通信协议**：一种经过协商的机制，通过这种机制，服务提供者和服务使用者可以就将要请求的内容和将要返回的内容进行沟通。
- **服务描述**：一种经过协商的模式，用于描述服务是什么、应该如何调用服务以及成功地调用服务需要什么数据。
- **服务**：描述供使用的操作。



功能性方面—流程

业务流程

- 一个服务的集合，可以按照特定的顺序并使用一组特定的规则进行调用，以满足业务要求；
- 业务流程本身也看作是服务，这样就产生了业务流程可以由不同粒度的服务组成的观念。



功能性方面—服务注册中心

服务注册中心

- 一个服务和数据描述的存储库；
- 服务提供者可以通过服务注册中心发布它们的服务，而服务使用者可以通过服务注册中心发现或查找可用的服务；
- 服务注册中心可以给需要集中式存储库的服务提供其他的功能。

服务质量方面

可靠高效

- 低延迟高吞吐量、高可扩容
- 容错、高可用
- 缺陷少

安全性

- 应用于调用服务的**服务使用者的身份验证、授权和访问控制**

一致性事务

- 应用于**一组服务，以提供一致的结果。如果要使用一组服务来完成一项业务功能，则所有的服务都必须都完成，或者没有一个完成**

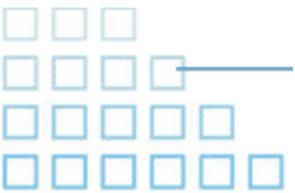
服务管理

- 应用于**管理提供的服务或使用的服务**



提纲

- 产生发展历程
- 释义和现状
- 核心理念
- 技术体系
- **发展趋势**





服务计算



服务科学

服务计算理论

- 建模
- 理论
- 实验
- 行为
- 规则



服务工程

服务系统的工程开发

- 规范、技术
- 方法、工具
- 自动化
- 度量性



服务管理

服务质量和性能

- 优化
- 效率
- 质量
- 最优化方法
- 管理经验

以服务为中心

科学

工程

管理



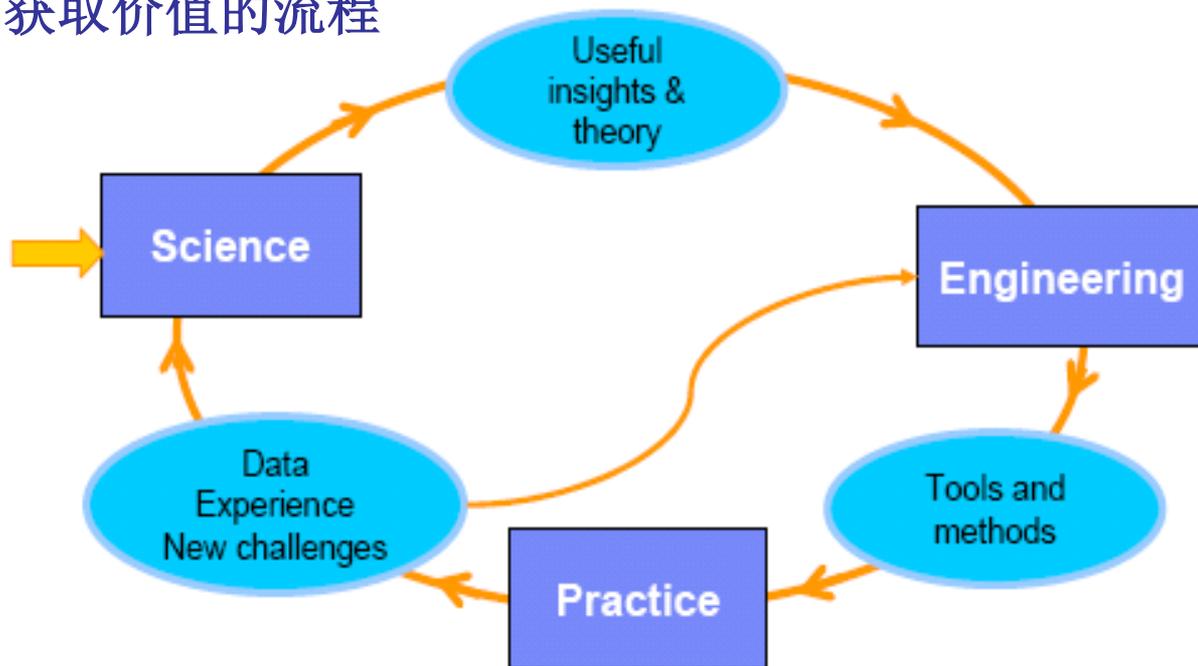
服务科学?

- 50多年前，随着信息技术时代的到来，**计算机科学从电子工程和数学分离出来**，成为一个独立的学科
- 随着服务经济的到来，**服务科学**会不会也像计算机学科一样，在不久的将来**成为一门独立的学科**？
- 服务科学可以将计算机科学、运筹学、产业工程、数学、管理学、决策学、社会科学和法律学等既定领域内的工作相融合，创建新的技能和市场，提供高价值的业务



科学、工程与管理

- 科学
 - 用来产生知识
- 工程
 - 应用知识去产生价值
- 管理
 - 改进产生和获取价值的流程

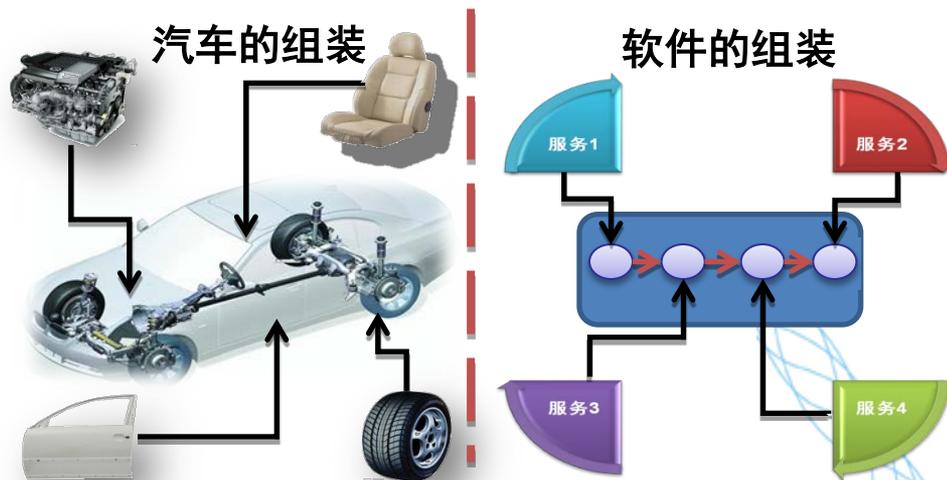


服务工程(Services Engineering)

- 面向服务软件的工程化开发
 - 高效地开发高质量的软件制品
 - 软件开发的工业化：软件生产线
- 面临的挑战
 - 服务工程的数学理论
 - 服务工程的设计方法
 - 服务工程的模型驱动架构
 - 服务工程的信任、安全、隐私
 - 服务工程的Agent技术

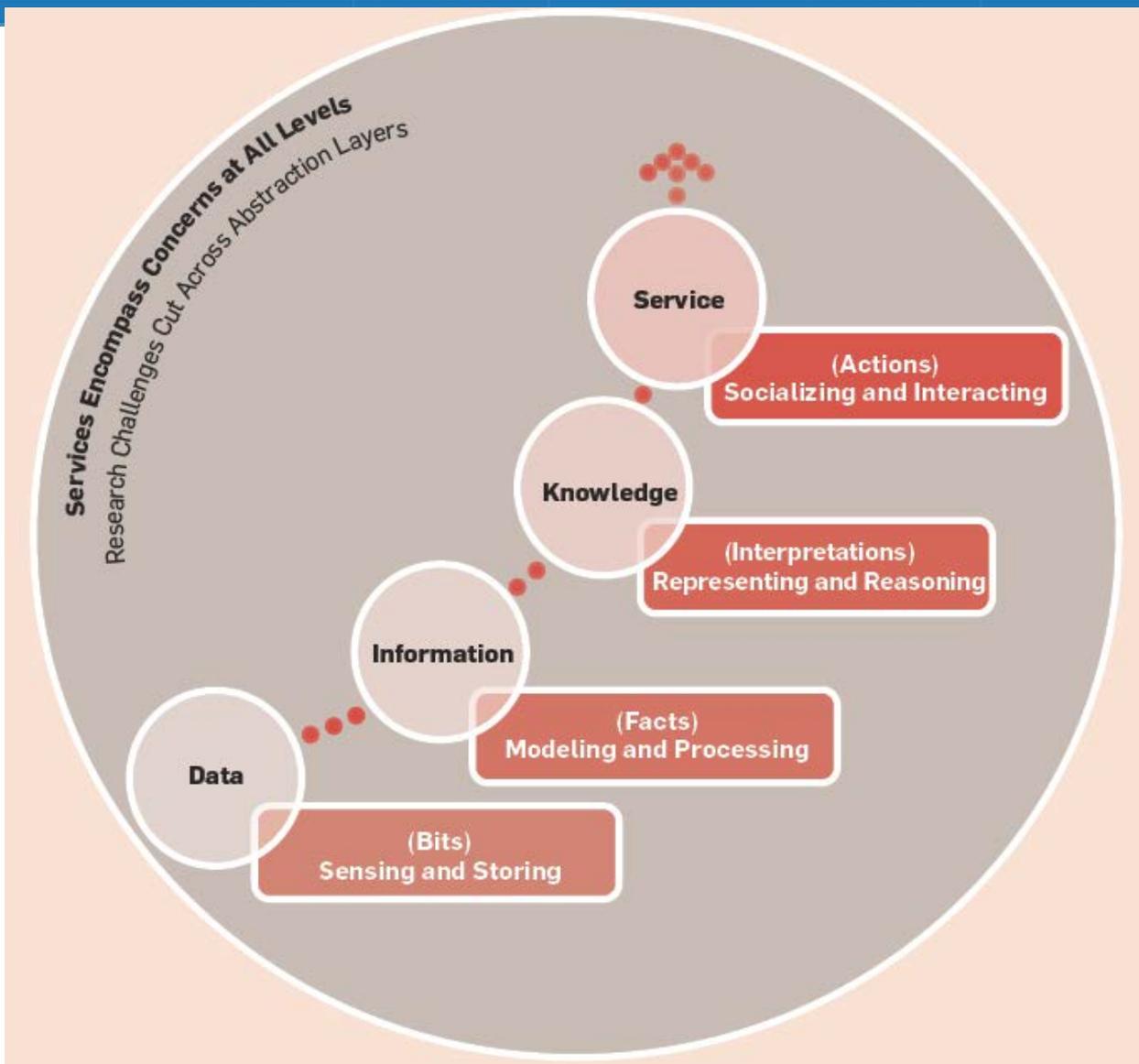
 - 服务计算应用的需求工程
 - 普适应用的服务开发

 - 服务的分析与设计模式
 - 服务的可信性、可生存性、重构、测试、验证与校验
 - 服务的本体化与语义匹配
 -



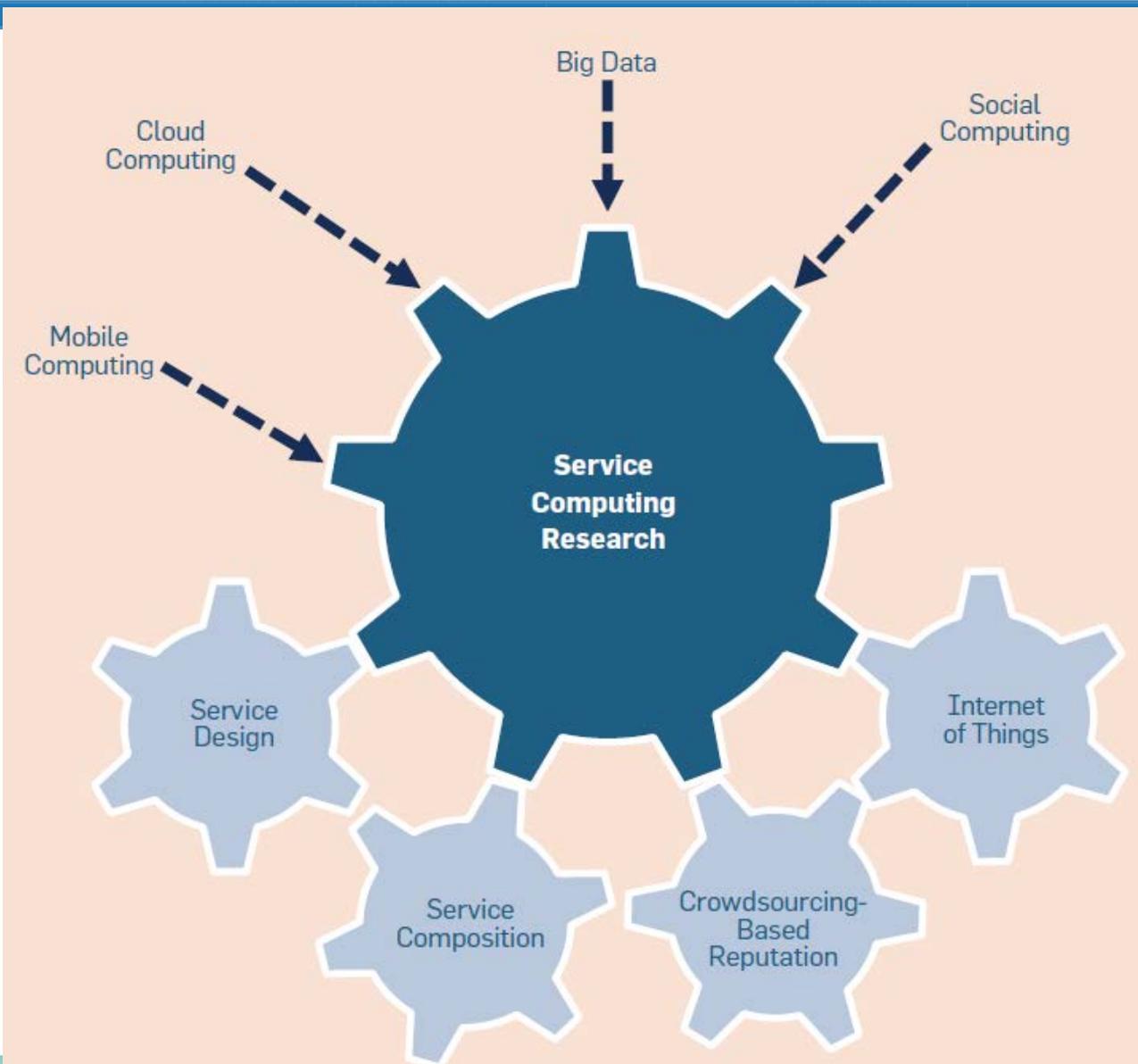


数据科学与服务





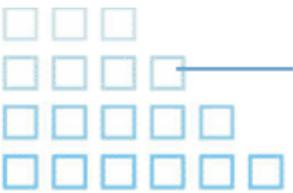
服务计算正在成为交叉前沿学科





总结

- SOA、Web服务、服务计算产生
- 服务计算发展历程与现状
- 服务计算概念
- 服务计算技术体系
- 服务计算发展趋势





结 束!

