



软件工程概论

梅 宏

北京大学信息科学技术学院软件所
高可信软件技术教育部重点实验室



北京大学



主要内容

- ❖ 软件与软件产业
- ❖ 软件工程：概念及范围
- ❖ 软件工程发展回顾
- ❖ 软件工程现状及未来
- ❖ 关于本课程的学习





软件概念 (1)

❖ 软件—计算机系统中的程序及其文档

- 程序：计算任务的处理对象和处理规则的描述
- 文档：为了便于了解程序所需的阐明性资料
 - 张效祥主编. 计算机科学技术百科全书 (第2版)

❖ 其他定义

- Ian Sommerville : Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.
- R.S. Pressman : Software is a set of items or objects that form a “configuration” that includes : programs 、 documents 、 data ...

❖ 软件的作用

- 作为计算机用户和硬件间的接口界面
- 在计算机系统中起指挥管理作用
- 计算机体系结构设计的重要依据





软件概念（2）

- ❖ 软件一词用于描述计算机中的非硬件成分（特指计算机程序）出现于 50 年代，1960 年前后才广为流传
 - 高级语言的出现极大促进了计算机在各行业的应用，也使得术语“软件”得以诞生并广泛流行
 - 60 年代初，IBM 将 IBM1400 系列上的应用程序库改造为更为灵活易用的软件包形式。软件包概念的出现启动了软件从计算机系统中分离出来成为独立成分的过程
 - 同时，软件相对与硬件和体系结构，成为一个学科分支
- ❖ “软件”一词具有三层含义
 - 指一个科学研究领域或一门学科
 - 指计算机系统中相对于硬件的成分
 - 指某特定的应用软件



软件概念 (3)

❖ 软件的分类 (1)

➤ 基础设施类

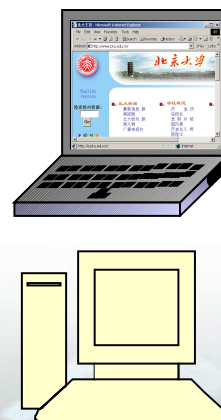
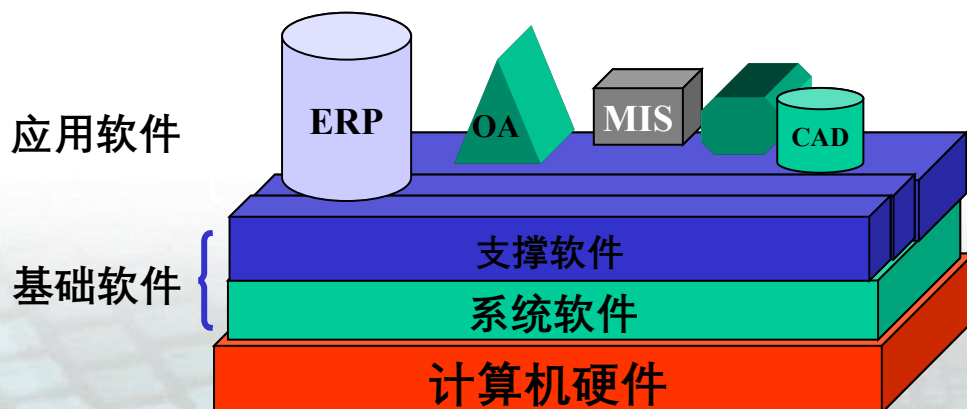
- 运行支撑类（操作系统、中间件、应用服务器等）－系统软件
- 开发支撑类（各类软件工具、可复用软件资源等）－支撑软件

➤ 应用类

- 各种领域、行业的应用软件系统（业务逻辑的体现）－应用软件

❖ 软件的分类 (2)

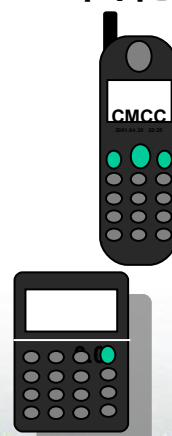
➤ 一般性软件和嵌入式软件



装载式



固化式



北京大学



软件概念 (4)

❖ 软件研究的 3 个层次

➤ 软件模型层

- 主要体现为计算模型（程序模型）的研究

➤ 运行开发层

- 针对具体计算模型，提供运行平台及开发方法、技术和工具

➤ 应用层

- 根据具体应用需求，基于上述模型、平台和方法开发应用系统

❖ 软件技术研究主要针对前两个层次，形成两条主线

➤ 软件系统技术

- 软件的本质、特征与模型，即软件的基本元素、体系结构、交互协议、行为模式、效率机制和质量保障机制等

➤ 软件开发技术

- 软件开发方法论及相应的工程原则、支撑工具和环境等





软件发展的阶段

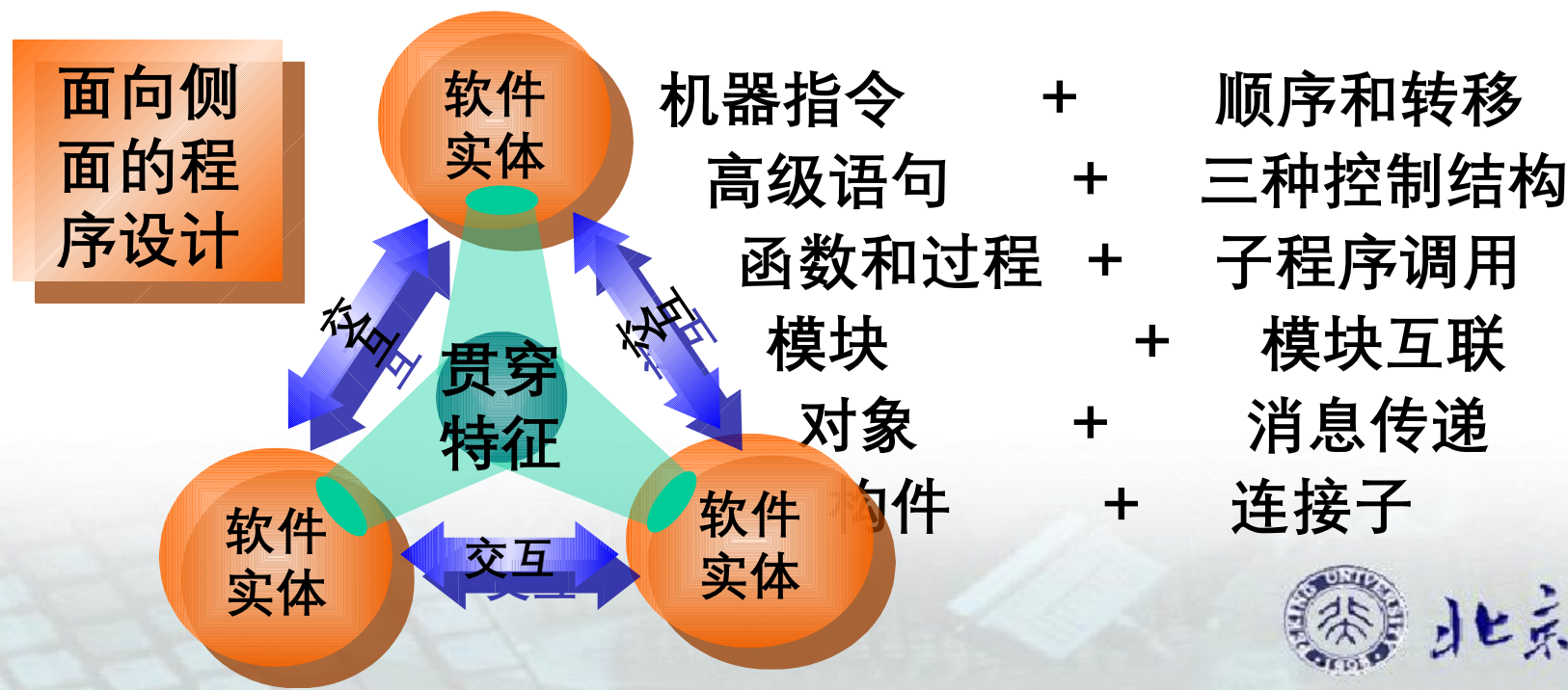
- ❖ 第一阶段： 40 年代中期到 50 年代中期，实用高级语言的出现前
- ❖ 第二阶段： 50 年代中期到 60 年代后期，软件工程出现前
- ❖ 第三阶段： 60 年代后期以来，软件工程出现以后
- ❖ □□□□□□□□□□□□□□□□□□



软件技术发展的驱动力

❖ 追求更具表达能力、更符合人类思维模式、**更具可构造性和演化性的计算模型** **(1)**

软件的基本模型 = 实体元素 + 连接和交互

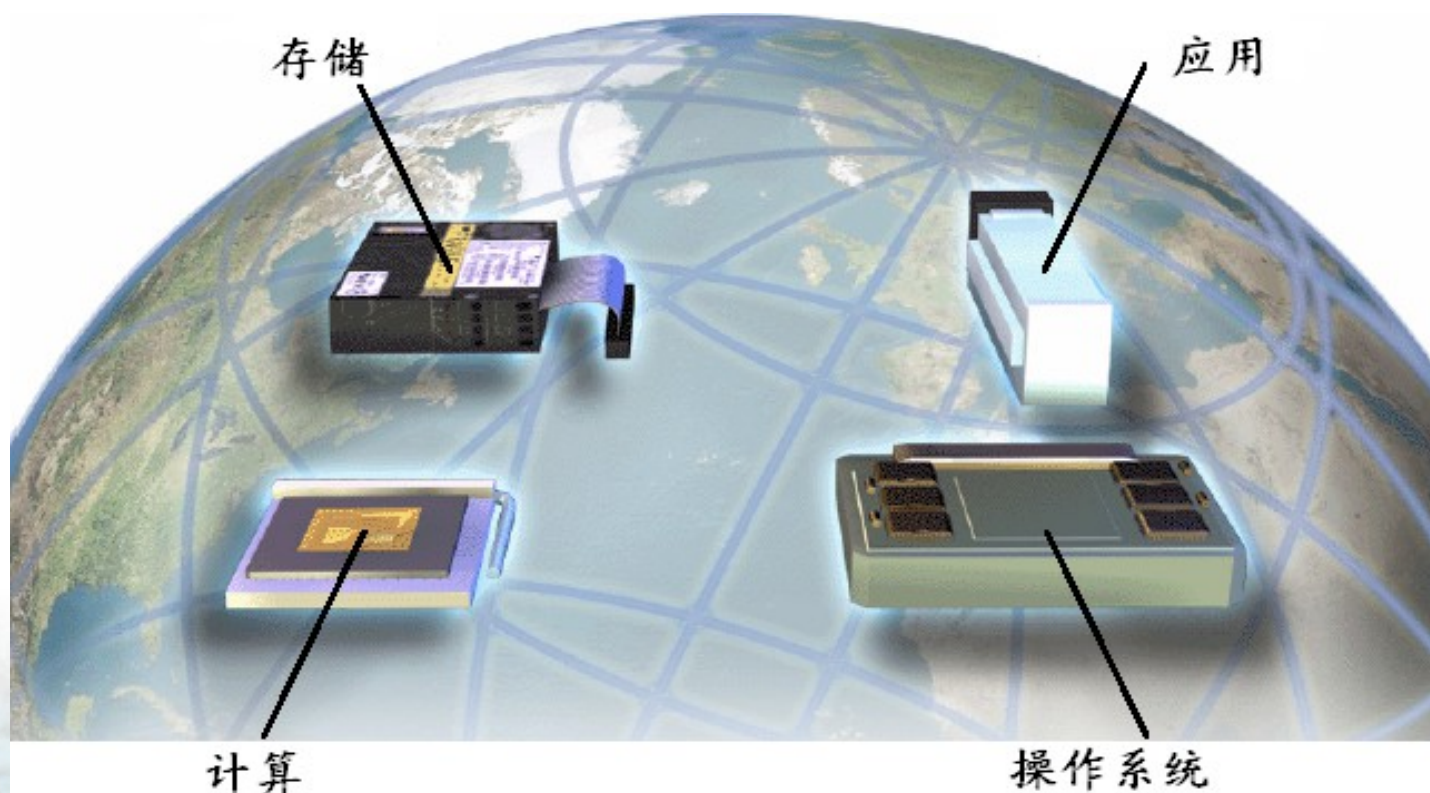


北京大学

软件技术发展的驱动力

(2)

❖ 通过虚拟的运行平台追求更高效地发挥下层硬件资源所提供的计算能力



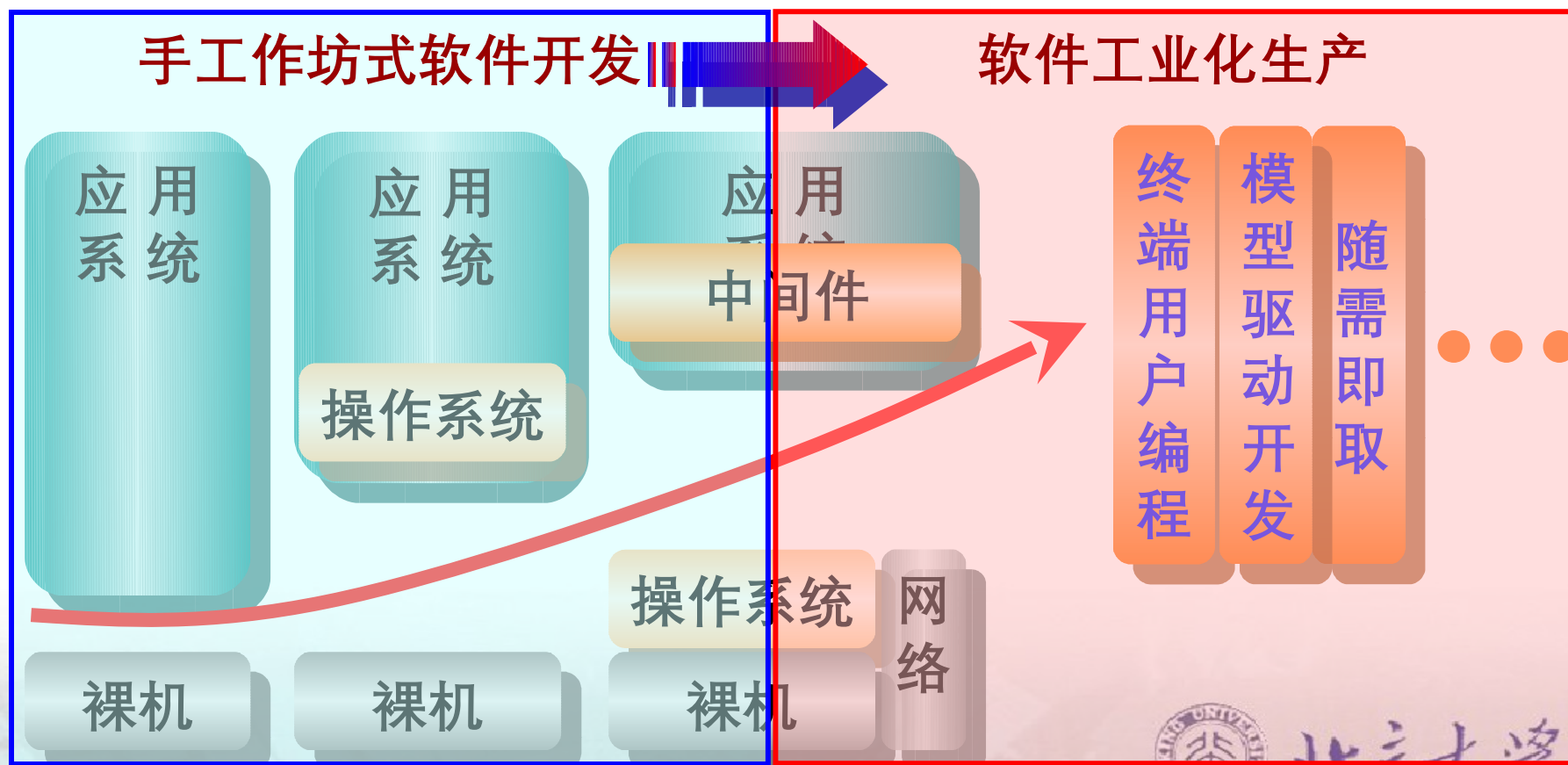
高效
可信
统一的
虚拟资源



北京大学

软件技术发展的驱动力

❖ 尽可能多地凝练共性，提高软件开发的效率和质量 (3)

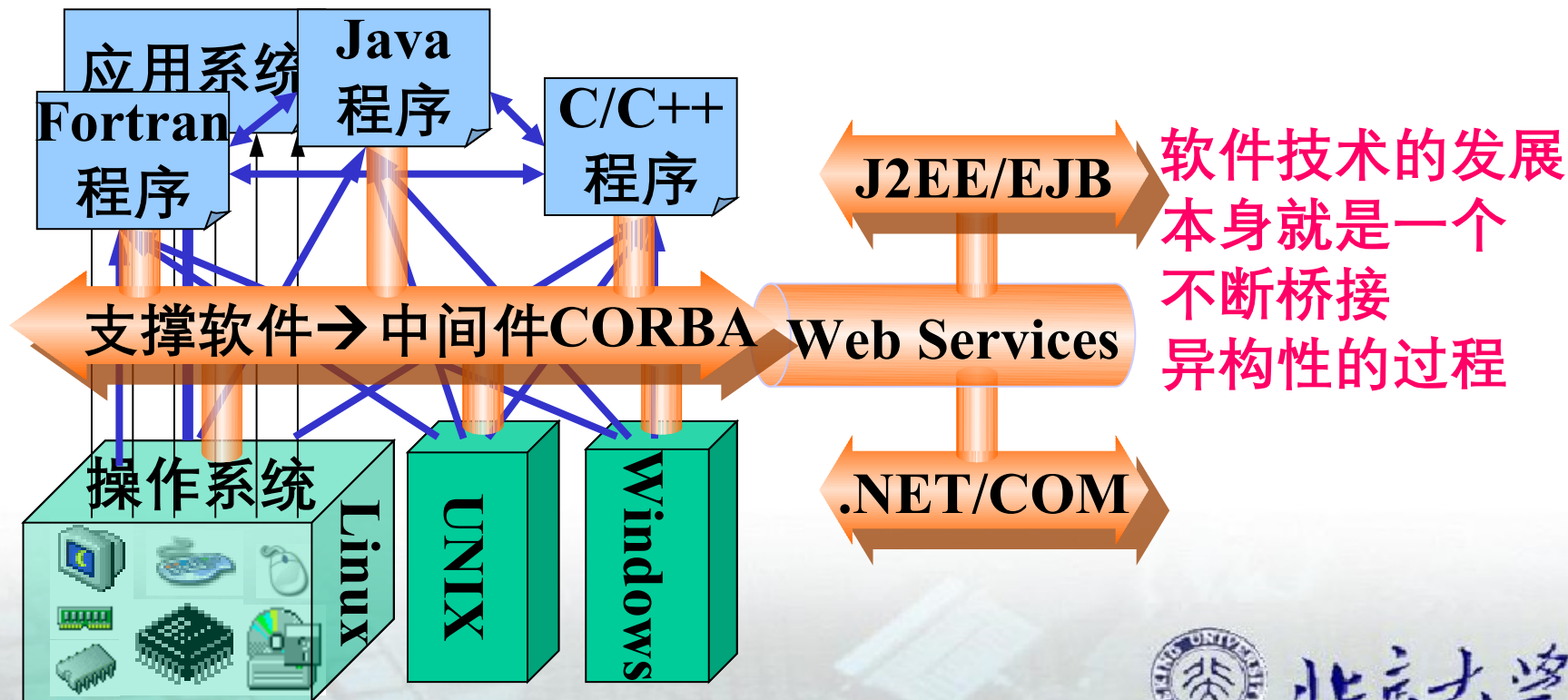


北京大学

软件技术发展的驱动力

(4)

- ❖ 尽可能平滑地桥接异构性，增加互操作性
- ❖ 异构是市场行为的自然产物，开放互操作同样是市场行为的产物



❖ 形成

- 60 年代，软件包概念的出现及 IBM360 系列机的研发，使得软件脱离硬件系统，开始自己的独立发展道路
- 80 年代，微软公司的成功促进了软件产业作为独立产业形态的发展
- 90 年代，美国软件产业每年以 12.5% 的速度增长，增长速度比整个经济增长高出 2.5 倍
- 当前，软件及软件服务业已成为信息产业中的支柱产业

❖ 和传统产业相比，软件产业还未成熟

- 智力密集、人力密集，“工业化”程度低
 - 合理的产业分工和协作、行业标准的形成、质量和过程的可控、可度量、……
 - 手工作坊式为主，自动或半自动工具应用弱





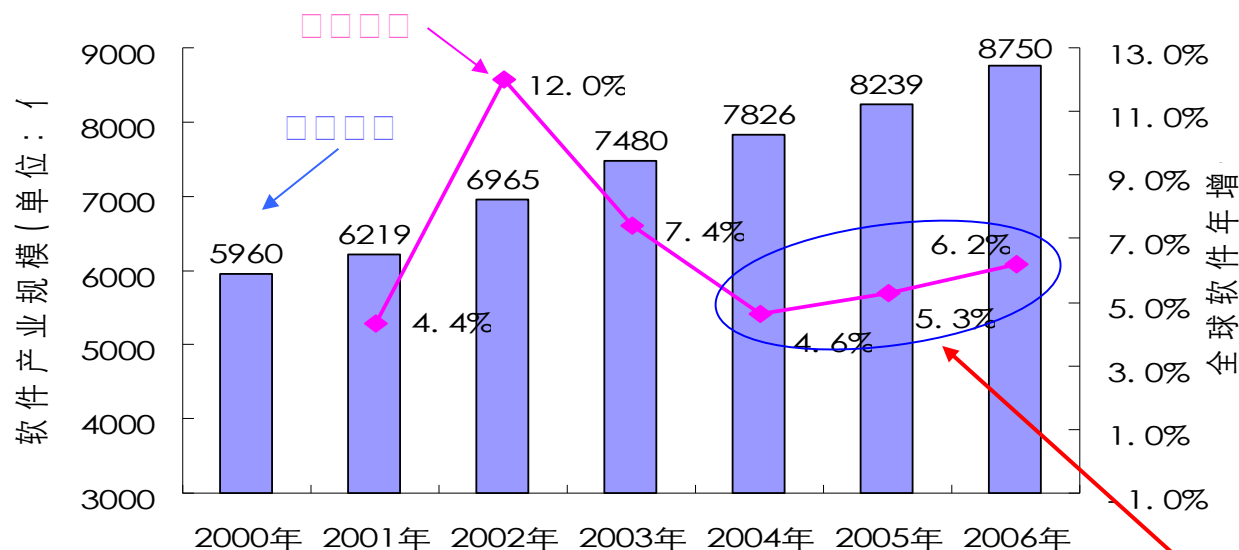
软件产业的地位

- ❖ 国际上将软件产业视为国家的战略性新兴产业，是国际竞争的焦点和制高点。当今发达国家竞相发展软件产业
- ❖ 我国高度重视软件产业
 - 2000 年 6 月，国务院发布《鼓励软件产业和集成电路产业发展的若干政策》（18 号文件）
 - 这是自 1994 年国务院发布《汽车工业产业政策》之后的又一重要举措，对中国软件产业发展具有举足轻重的意义
 - 2002 年：振兴软件产业行动纲要（2002 年至 2005 年）
 - 2005 年：国家中长期科学和技术发展规划纲要（2006-2020 年）
 - 软件技术和软件产业的核心与基础作用得到了进一步的明确，其在国家信息化发展需求中的战略地位日益凸显
- ❖ 我国高度重视软件人才培养
 - 教育部创办示范性软件学院，培养软件工程人才



全球软件产业发展 (1)

- ❖ 全球软件产业规模一直保持快速增长，自 2004 年起增长率趋于平稳



2001-2006 年全球软件产业规模及年增长率

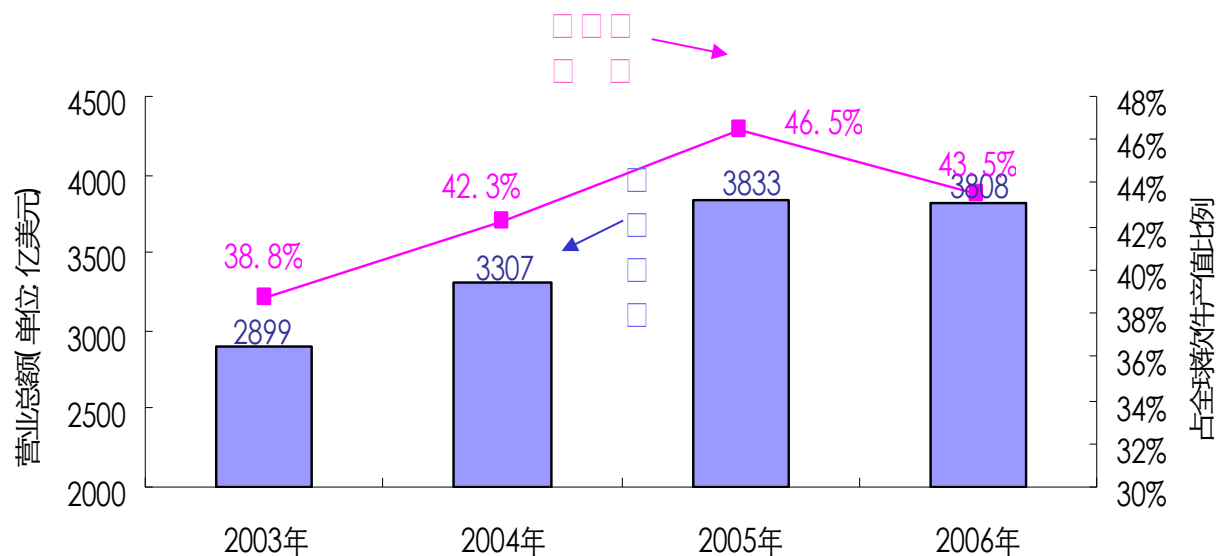
- ❖ 从 2000 年 5960 亿美元 增加到 2006 年 8750 亿美元
- ❖ 全球软件产业 年增长率 保持在 5%-10% 高于 全球经济年平均增长率的 4%



清华大学

全球软件产业发展（2）

❖ 全球软件企业 500 强营业总额 2006 年开始回落，中小企业的贡献增强



2003-
2006
年全球
软件企
业 500
及占全
球软件
产值比
例

2003-2005 年间

全球软件企业 500 强营业总额 由 2899 亿美元上升至 3833 亿美元
占全球软件产业产值的比例 由 38.78% 上升至 46.52%

2005-2006 年间

500 强营业总额降到 3808 亿美元 占全球比例 降到 43.50%
同期内 全球软件产业以 6.2% 的速度 平稳增长



北京大学

全球软件产业发展（3）

❖ 开源软件持续增长，促进全球软件产业多元化发展

产 品	
整体构架	开源构架 LAMP 同 J2EE .Net 三足鼎立
操作系统	市场份额从 2004 年 20% 将增至 2008 年 27%
中间件	2006 年 Apache 在 欧洲 美国 市场份额分别为 84% 66%
	2005 年 JBoss 市场份额为 37% 而 IBM Websphere 为 37.2%
数据库	2006 年 MySQL 市场份额 25% 仅次于 SQLServer 成为 全球第二
桌面办公	OpenOffice 在北美 德国份额 8% 以上

区 域	
全球	全球应用开源软件企业占总数 50%
美国	使用开源软件企业比例为 87%
欧洲	使用开源软件的企业达 40% 其中 45 % 将开源软件部署为核心业务系统
	商业 金融 通讯传媒等企业 使用率超过 75 %
拉美	使用开源软件的企业超过 60%
	巴西政务领域采用开源软件率 80 %
中国	2004 年政府采购额的操作系统中 Linux 占 39 % 2006 年 Linux 市场收入达 1450 万美元 增长 23%

2006 年 微软 在开源软件方面 与 Novell 开展 大规模合作
2006 年 Oracle 宣布为 Red Hat Linux 提供 完整的技术支持
IBM SAP Google 等企业 纷纷加入 开源软件领域

全球软件产业：
开源软件与私有软件的 正被混合架构
将成为 世界软件产业发展

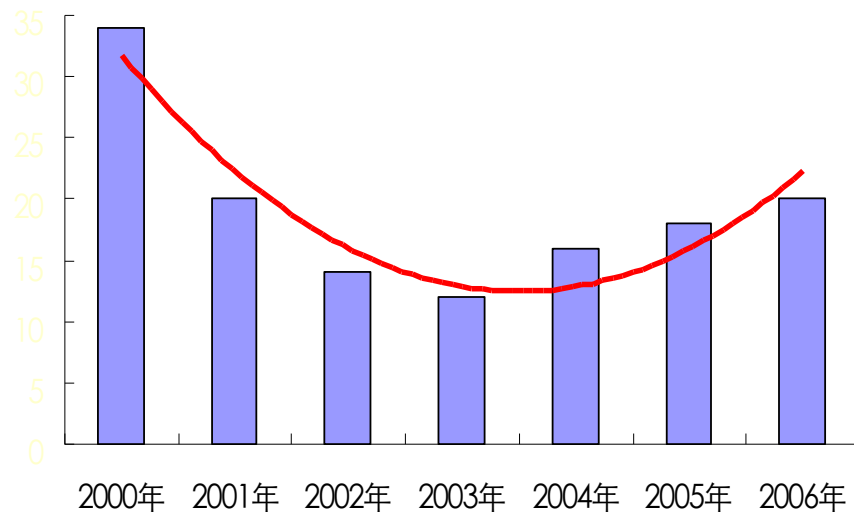


全球软件产业发展（4）

❖ 企业并购重新升温，市场竞争加剧

并购

是软件企业
整合产品线
争夺客户资源
拓展市场空间
重要方式



2004-2006 年间 并购活动 重新活跃

2005 年美国并购 10176 起

其中软件企业并购 1628 起 占 16% 位居各行业之首

2006 年 Google 以 16.5 亿美元收购 YouTube

HP 以 45 亿美元收购 Mercury Interactive

IBM 以 16 亿美元收购 FileNet

EMC 以 21 亿美元收购 RSA Security

2000-2006 □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ 2007 □

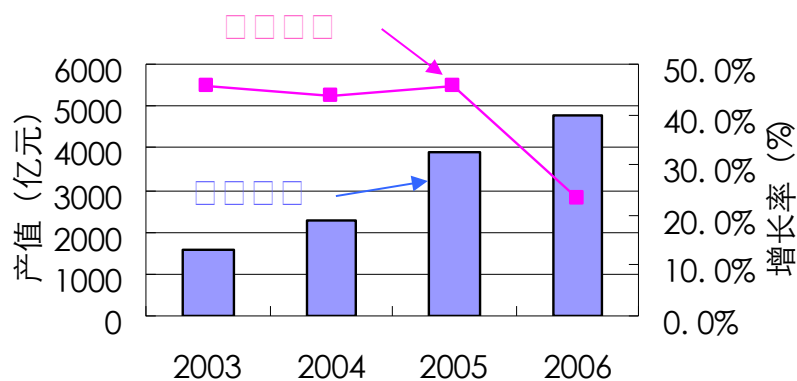


北京大学



中国软件产业现状（1）

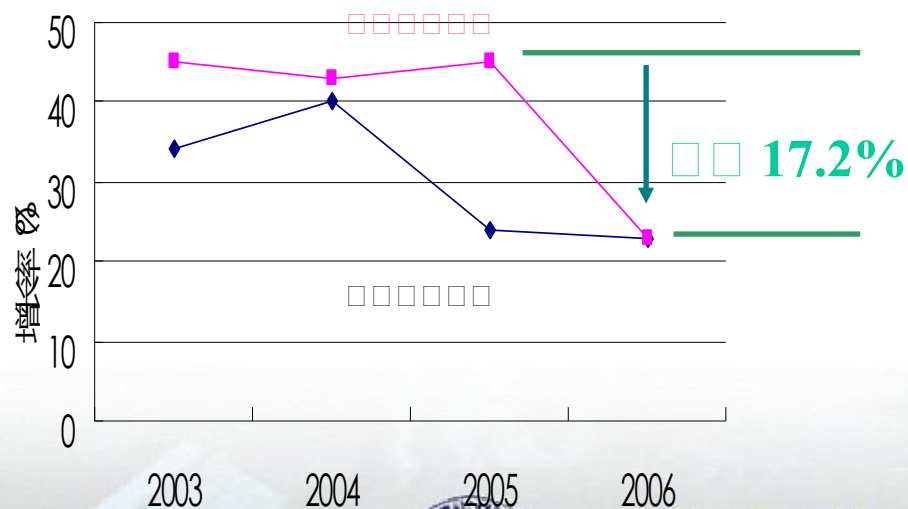
❖ 产业规模增长十分迅速，但增长率自 2006 年起明显回落



从 2003 年到 2005 年
年增长率超过 40%
2006 年增长率为 23.1%

2006 年发展速度比 2005 年回落

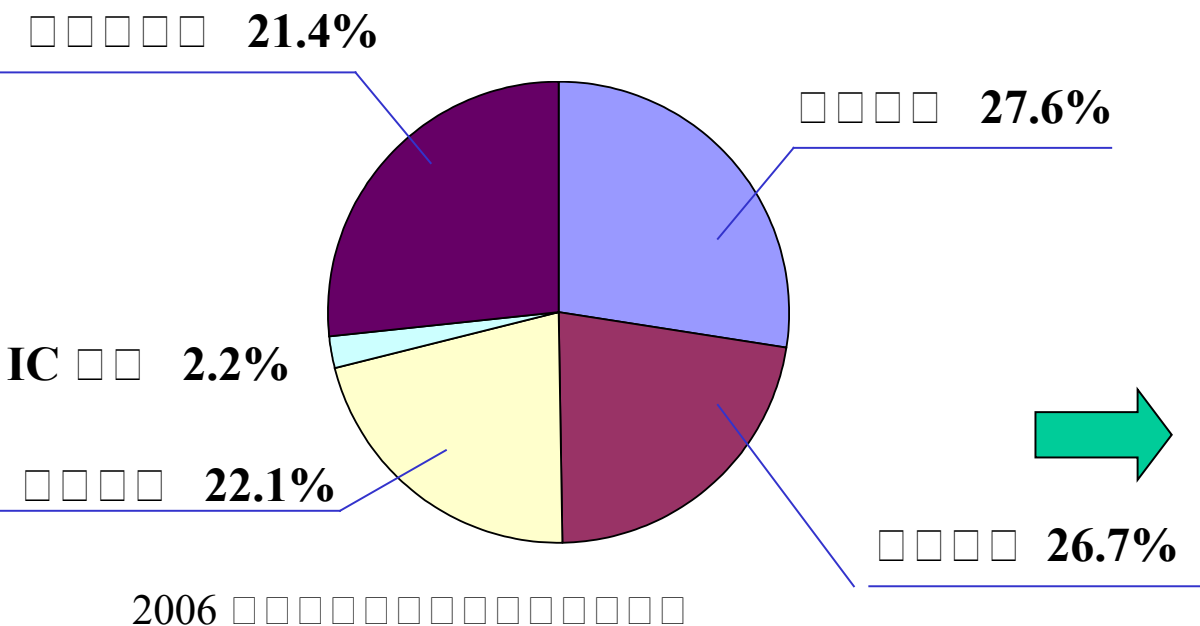
17.2 个百分点
从 2006 年下半年起
增速低于硬件产品





中国软件产业现状（2）

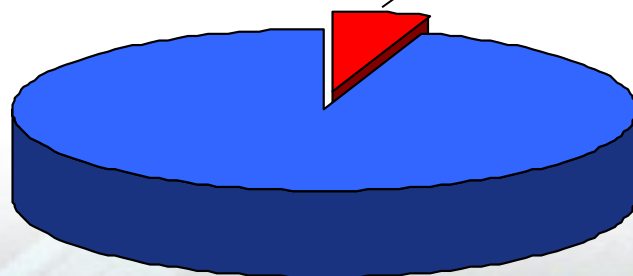
❖ 软件产业产值逐年增加，但产业增加值仍很小



❖ 2006 年
软件产值 4800 亿元
但 产业增加值
仅 1838 亿元

❖ 系统集成
占产业产值的 27.6%
所占份额 最多
但 增加值极少

2005 中国软件产业增加值构成



2005



清华大学

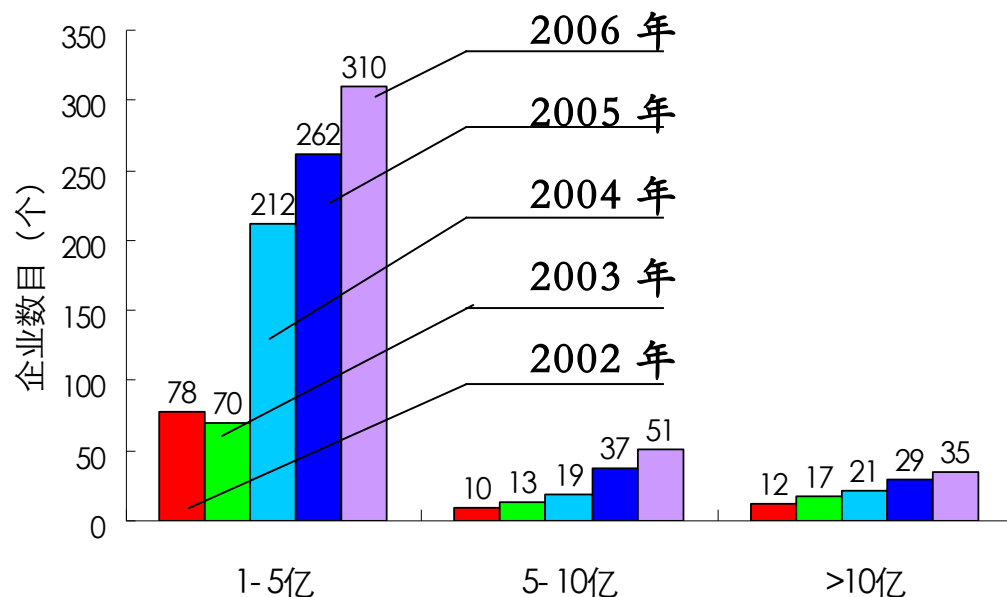


中国软件产业现状（3）

❖ 企业实力增强，但仍缺乏具有国际竞争力的龙头企业

❖ 中国软件企业规模不断扩大，到2006年

- 10 亿以上的企业有 35 家
- 5—10 亿的企业有 51 家
- 1—5 亿的企业有 310 家



2006 年中外软件代表性企业软件销售额对比 (单位：亿美元)

2002——2006 □□□□□□□□□□□□□□□□

中国代表性软件企业	中科红旗	达梦	用友
	0.05	0.75	1.74
国外代表性软件企业	Microsoft	Oracle	SAP
	365	117	94

❖ 与国外巨头相比
中国软件企业普遍较小
缺乏大型跨国软件企业集团



北京大学



中国软件产业发展的瓶颈

- ❖ 从软件产业分类上看，基础软件是瓶颈
- ❖ 从软件产业组织上看，缺乏大企业、软件产业生态环境不完善是瓶颈
- ❖ □□□□□□□□□□□□□□□□
- ❖ 从软件产业要素配置上看，高级人才缺乏、融资难是瓶颈

❖ 注：关于软件产业的描述摘自中国科学院学部咨询评议项目报告





软件危机

❖ 应用的繁荣带来了软件危机：不断增加的为大系统制造可靠软件的困难度

- 软件系统的规模和复杂性不断增长，对软件的需求增长超过供应（开发、演化和维护）能力。软件开发人员极度短缺、开发效率和软件质量不能满足用户的需求，最终延缓经济和社会的发展
- 1968 年 NATO 会议（ Garmisch , Germany ）第一次提出了“软件危机”的说法
- 例如， IBM 公司 1963 年至 1966 年开发的 IBM 360/OS
 - 耗资几千万美元，投入 5000 人年的工作量，近一百万行源程序，拖延了几年才交付使用，交付使用后仍不断发现新的错误
 - 每次发行新版本都修改了上一版本的一千个左右的程序错误
 - 项目负责人 F.D.Brooks 曾说：“就像一只逃亡的野兽落入泥潭做垂死挣扎，越是挣扎，陷得越深，最后无法逃脱灭顶之灾。”

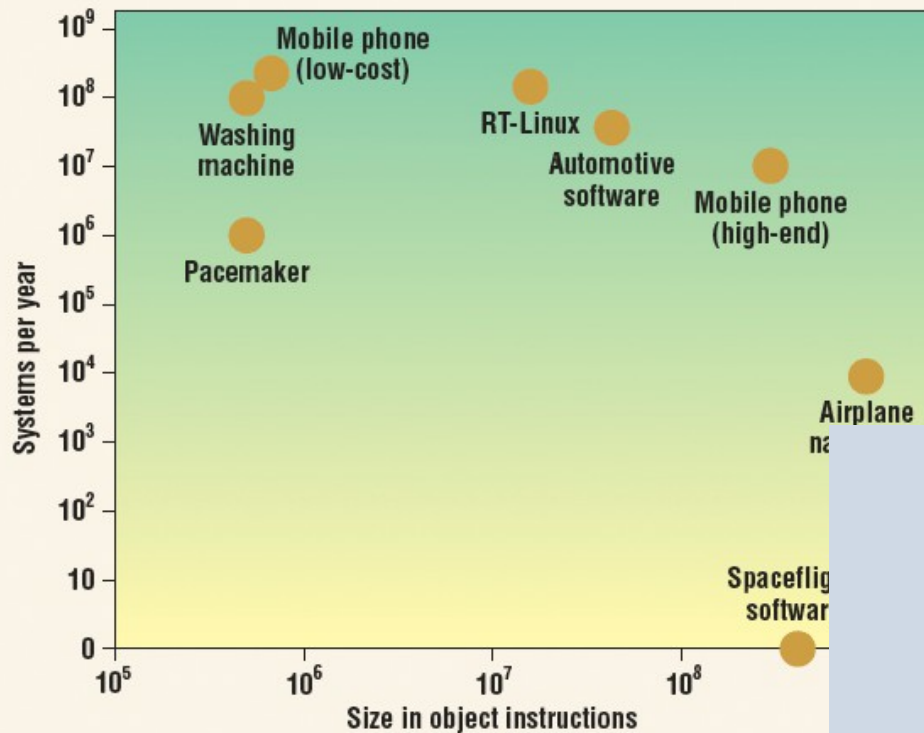


软件复杂性的增长

越来越高

： 3000 开发人员， 59 个开

（ 1970s ） - 1 亿行 （ 2010



- 空间站：软件代码 10 亿行
- 应用软件系统也越来越复杂，代码已是常见现象！

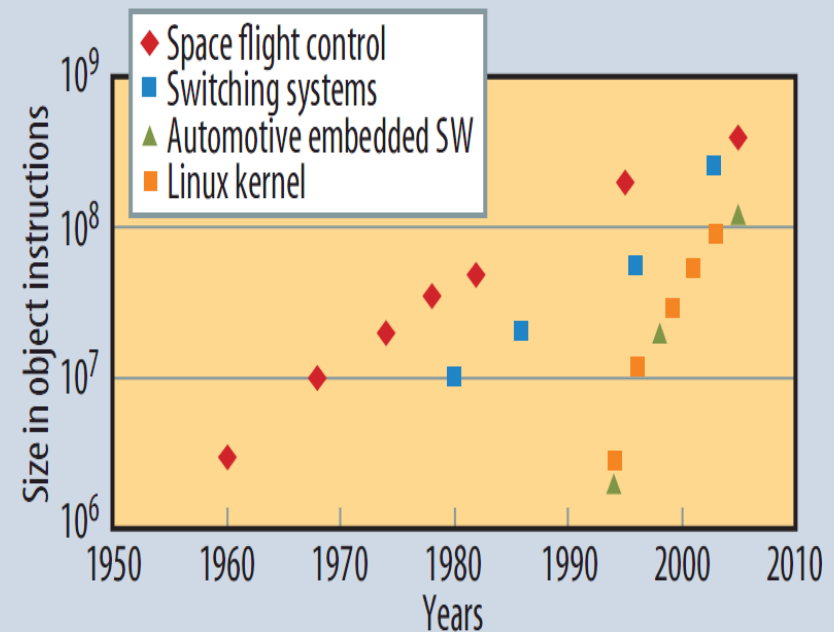


Figure 2. Complexity growth of embedded systems.



软件现状－危机仍然存在

- ❖ 对软件的需求远超出现有的生产能力
 - 信息技术的爆炸性增长促进了对新软件的空前需求
 - 软件系统的规模和复杂性不断增长
 - 用来开发这些软件的资源没有与上述需求保持同步
- ❖ 人们依赖于脆弱的软件
 - 不可靠、缺乏安全性、性能下降、出错、难以升级
 - 73 %的软件项目被延迟、超资、取消或失败
- ❖ 构造可靠、安全软件的技术能力不足
 - 过去 40 年，硬件性能至少跨越了 8 个重要的阶段
 - 开发软件的能力未能与硬件提供的机会保持同步





软件工程的诞生

- ❖ 1968 年 10 月北大西洋公约组织（NATO）的科学委员会在德国加尔密斯（Garmisch）开会讨论软件可靠性及软件危机的问题，产生了关于“软件工程”的加尔密斯报告，简称 G- 报告。在这次会议上，首次提出了“软件工程”概念。
- ❖ 1969 年 10 月 NATO 的委员会在意大利罗马开会讨论软件开发技术问题，产生了关于“软件工程技术”的罗马报告，简称 R- 报告。
- ❖ 软件工程试图以工程的原理、原则和方法进行软件开发，以解决软件危机
 - 将软件作为产品，借鉴传统工程途径，实现软件的工程化开发和工业化生产





软件工程概念

❖ 软件工程

➤ 应用计算机科学、数学及管理科学等原理，开发软件的工程

。

■ 借鉴传统工程的原则、方法，以提高质量、降低成本为目的。其中，计算机科学和数学用于构造模型和算法，工程科学用于制定规范、设计范型、评估成本和确定权衡，管理科学用于计划、资源、质量、成本等管理

■ 是一门交叉学科

❖ 其他定义

➤ **NATO 会议**：软件工程是用来建立和使用合理的工程原则，以经济地获取可靠的、且在真实机器上可高效工作的软件

➤ **IEEE [IEE93]**：

■ (1) 将系统化的、规范的、可量化的方法应用到软件的开发、运行及维护中，即将工程化方法应用于软件；

■ (2) 在 (1) 中所述方法的研究

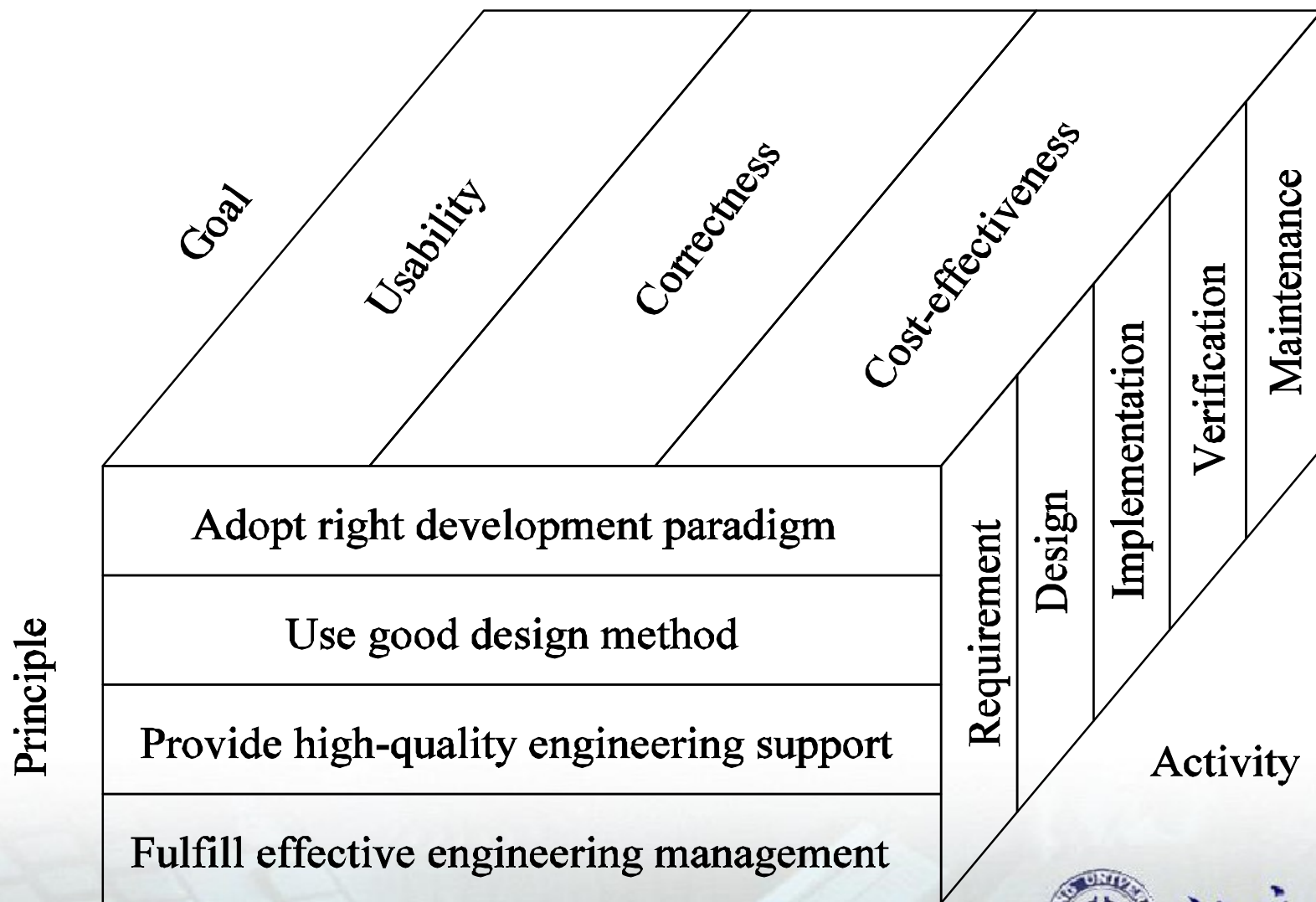


软件工程范围





软件工程框架

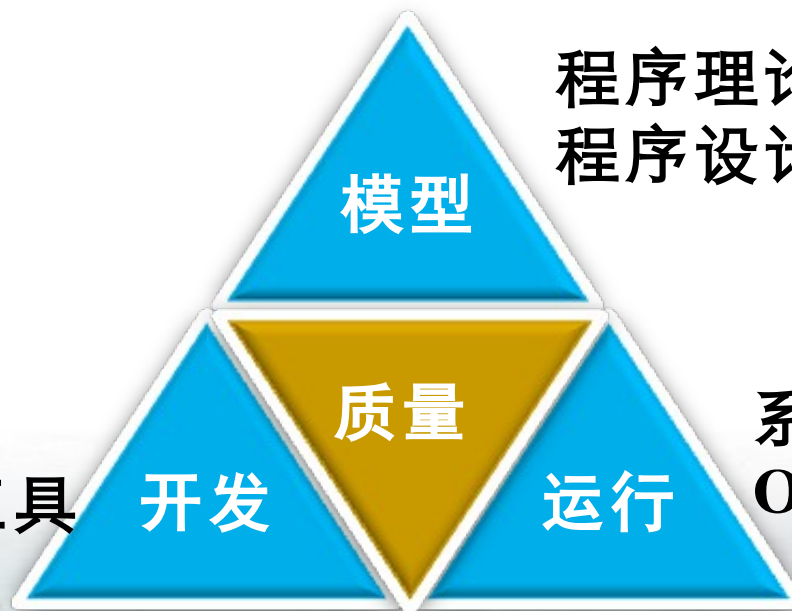




软件技术与软件工程

❖ 系统软件技术的发展是软件工程发展的主要推动力

- 程序语言、操作系统、数据库、中间件…
- 程序设计语言是软件开发的基础
- 操作系统等系统软件是软件及其理论与技术的综合载体
- …



软件工程：
方法 + 技术 + 工具

程序理论、范型和模型
程序设计语言

系统软件：
OS + DBMS + 中间件



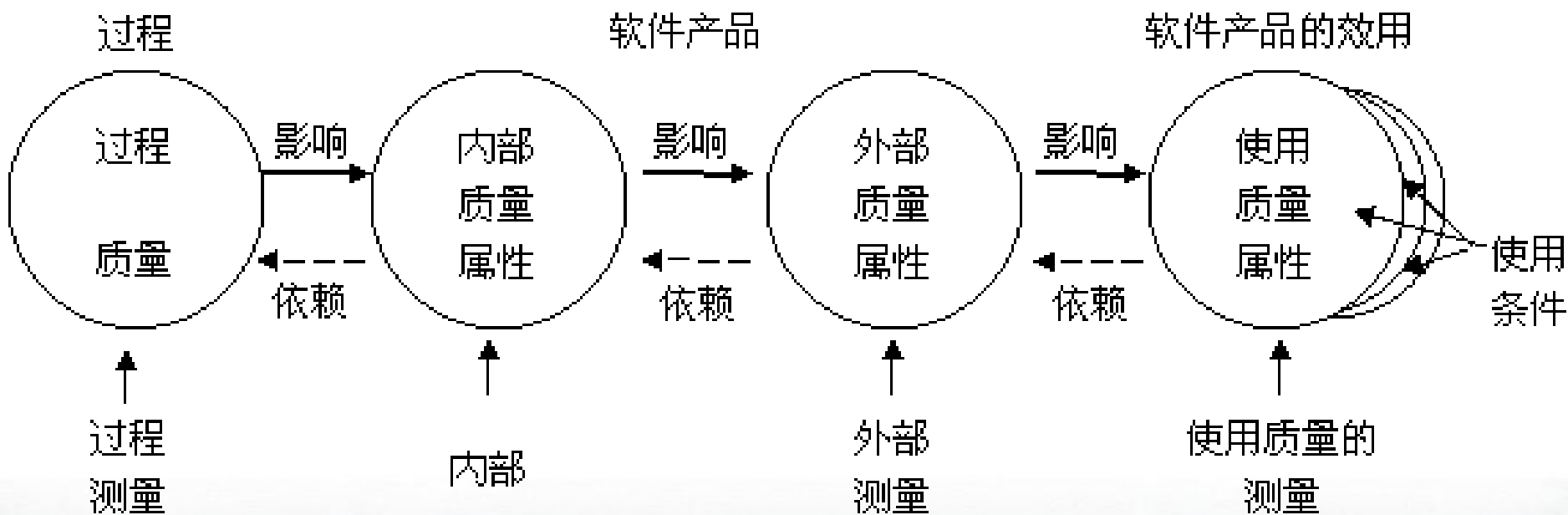
北京大学



ISO/GB 软件产品质量模型

(1)

❖ 软件质量：软件产品特性的总和，表示软件产品满足明确或隐含要求的能力



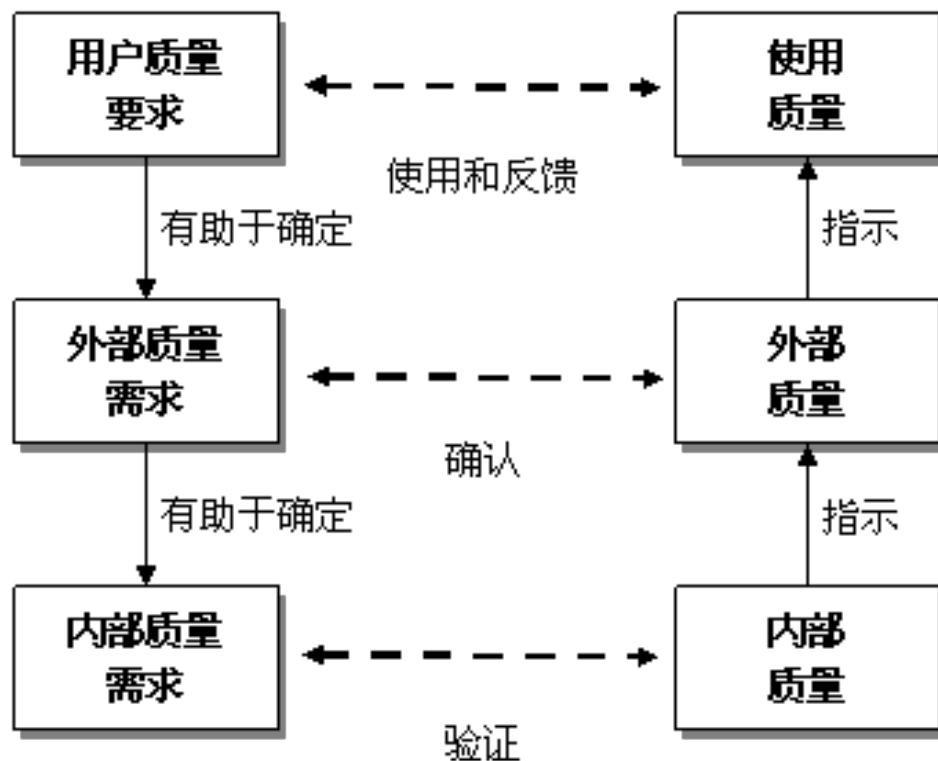
ISO/GB 软件产品质量模型 (2)

❖ 内部质量需求：

- 基于内部观点的软件产品特性的总体

❖ 外部质量需求：

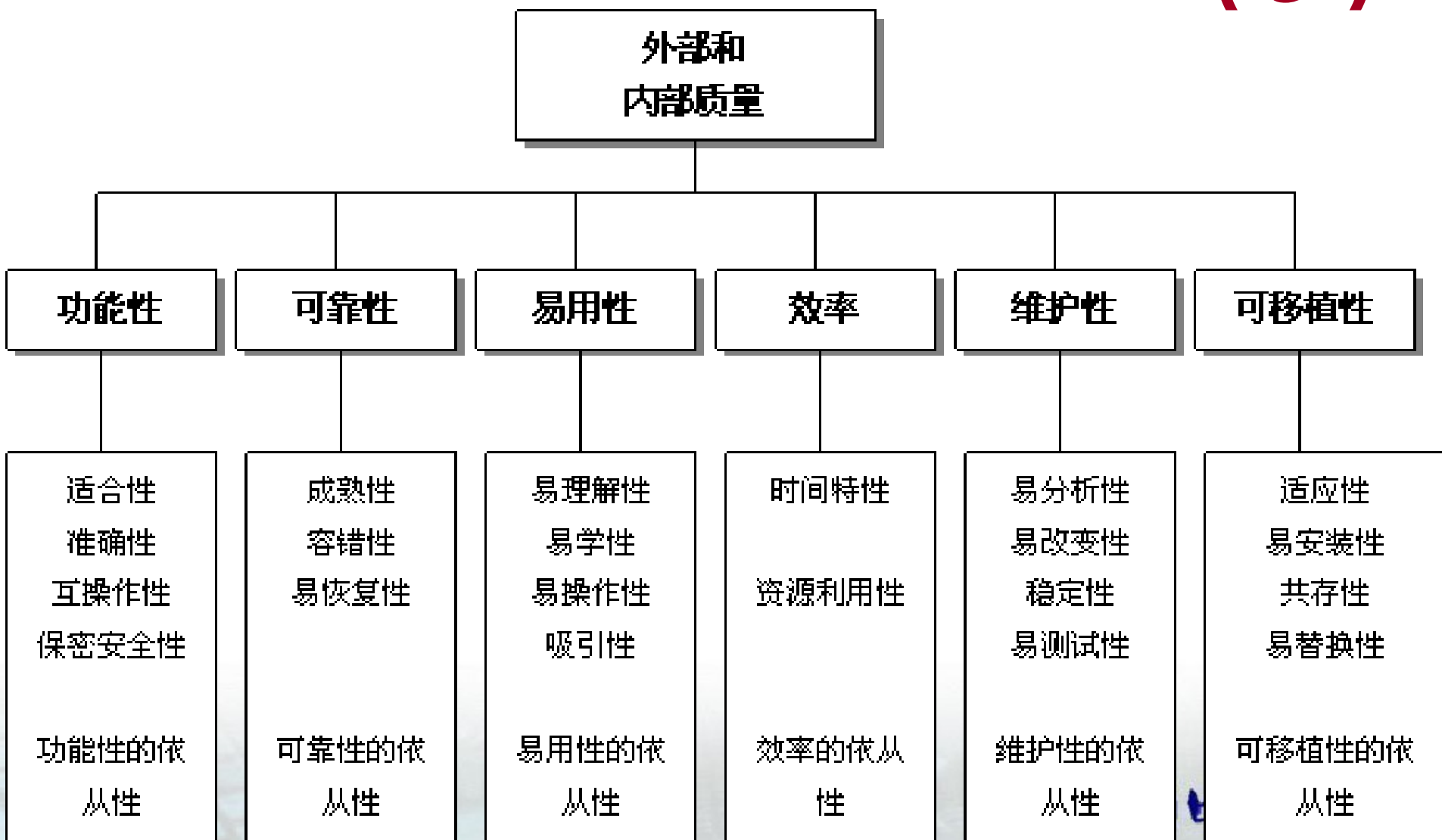
- 基于外部观点的软件产品特性的总体





ISO/GB 软件产品质量模型

(3)

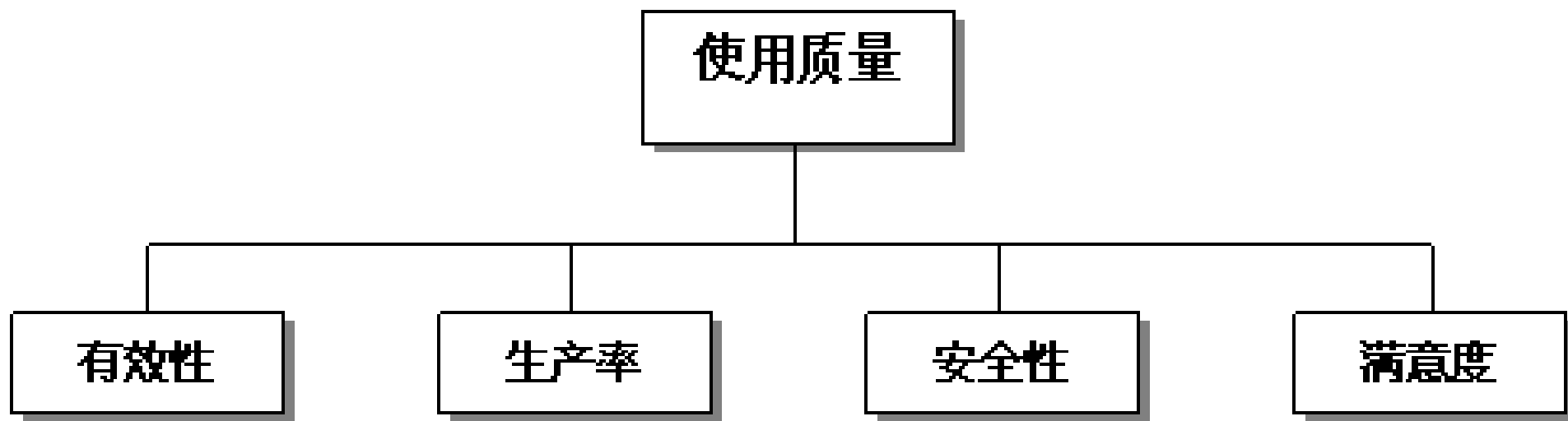




ISO/GB 软件产品质量模型

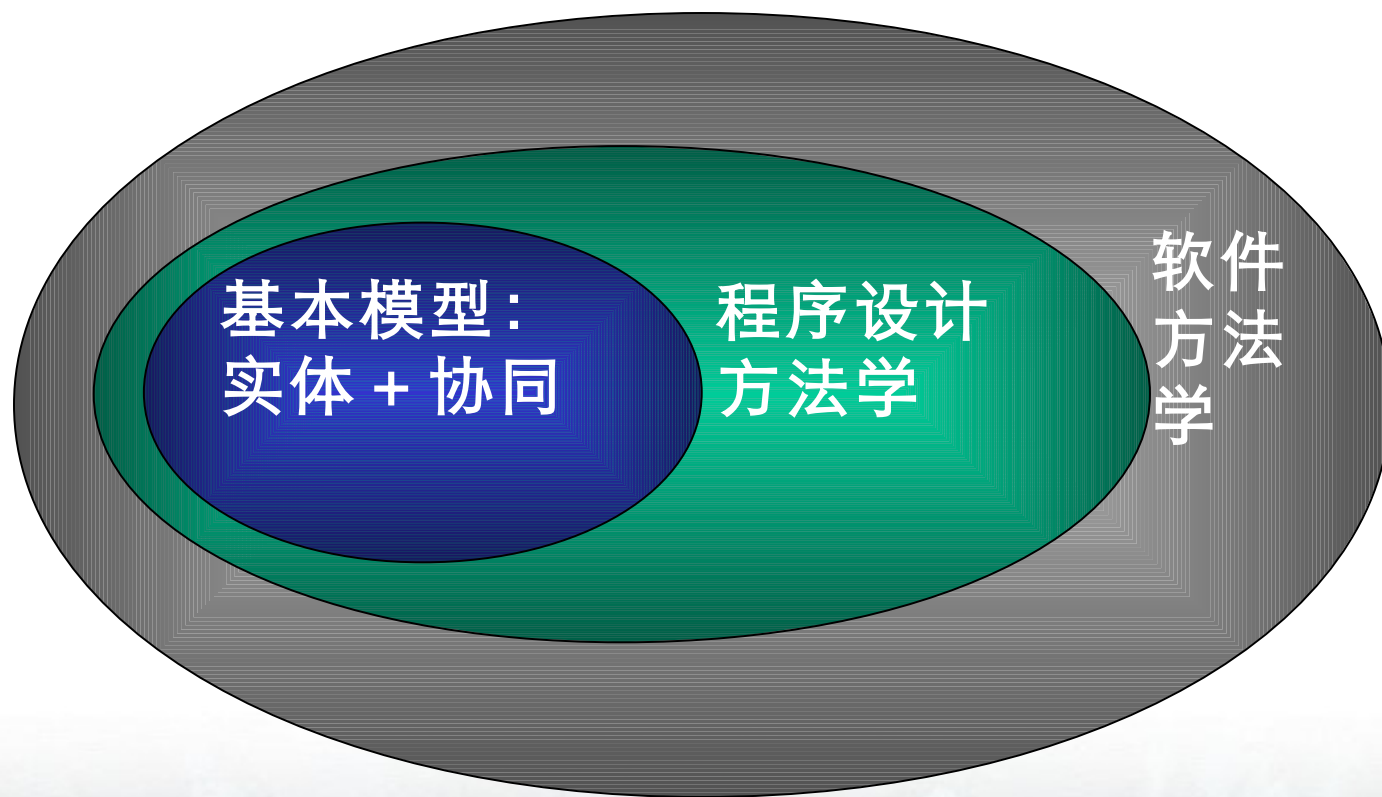
(4)

❖ 使用质量：基于用户观点的软件产品用于指定的使用环境和条件时的质量





软件方法学的层次



软件工程发展的主要历程

软件复用和软件构件技术被视为解决软件危机的一条现实可行途径
基于构件的软件开发方法成为主流技术之一

1990 年代
软件复用和构件技术受到关注

CASE 工具 and 环境的研制成为热点
面向对象技术开始出现并逐步流行

1980 年代
软件开发方法学成为研究热点

出现了结构化分析和设计方法

1970 年代
程序设计方法学成为研究热点

软件工程被正式提出
开始注重程序结构的研究
程序设计语言和编译系统得到应用

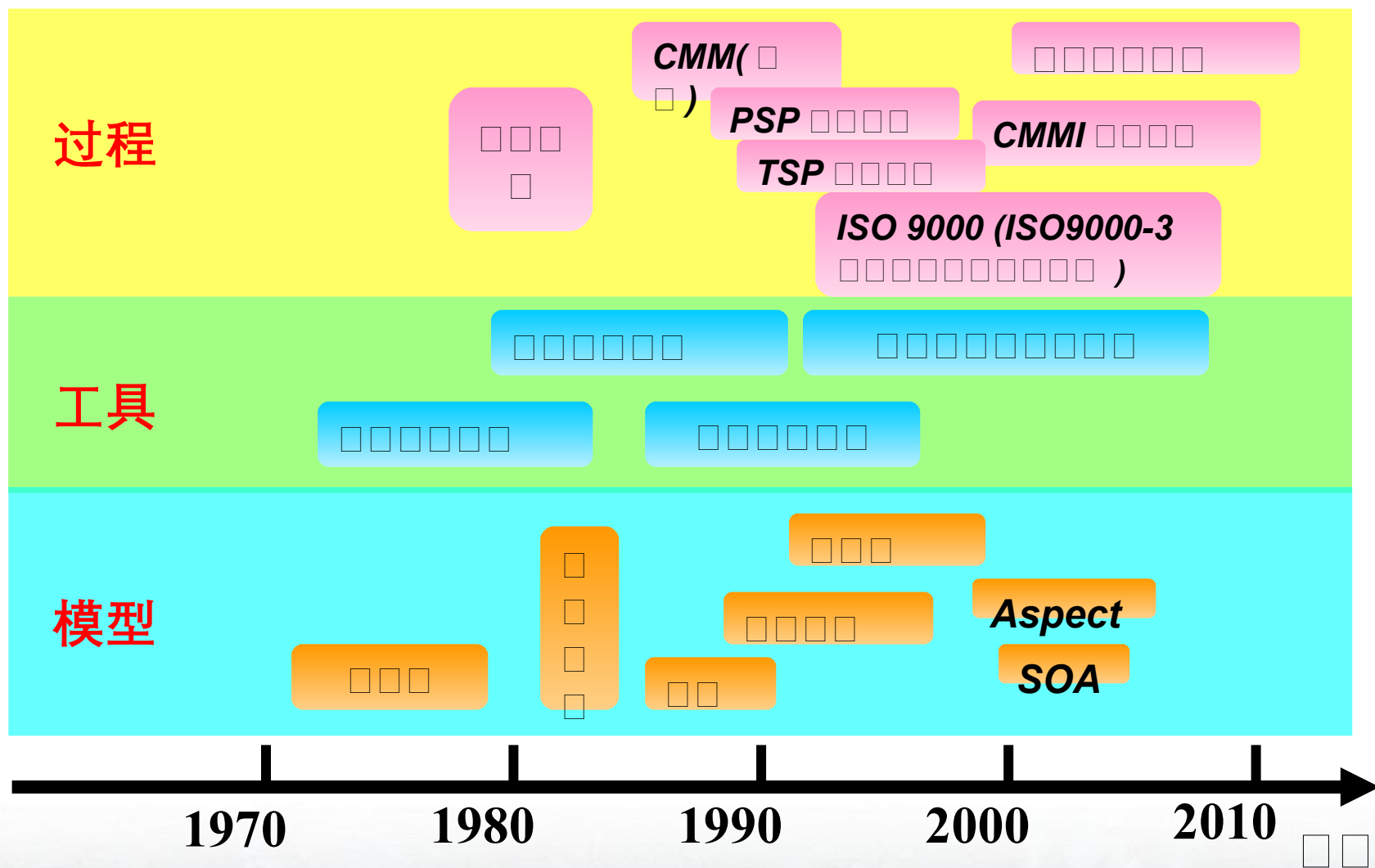
1960 年代
出现了软件危机



北京大学



软件工程里程碑



软件工程呈**多线、并行、交叉**发展
每个线路都有各自的里程碑



北京大学



软件方法学发展回顾

❖ 与软件方法学发展密切相关的三个要素

- 计算机平台、人的思维模式和问题基本特征

❖ 复杂性控制—软件方法学发展的驱动力

- 高级语言：控制计算机硬件平台的复杂性
- 结构程序设计：控制程序开发过程和执行过程的复杂性
- 面向对象方法：控制系统需求易变所导致的复杂性

❖ 一些重要的方法学概念

- 分而治之、逐步求精
- Separation of Concerns
- 局部化、信息隐蔽、封装、抽象
- 良构性、结构化、演化性



对软件工程认识的发展

❖ 如何达成软件工程的目标——提高开发质量和生产率？

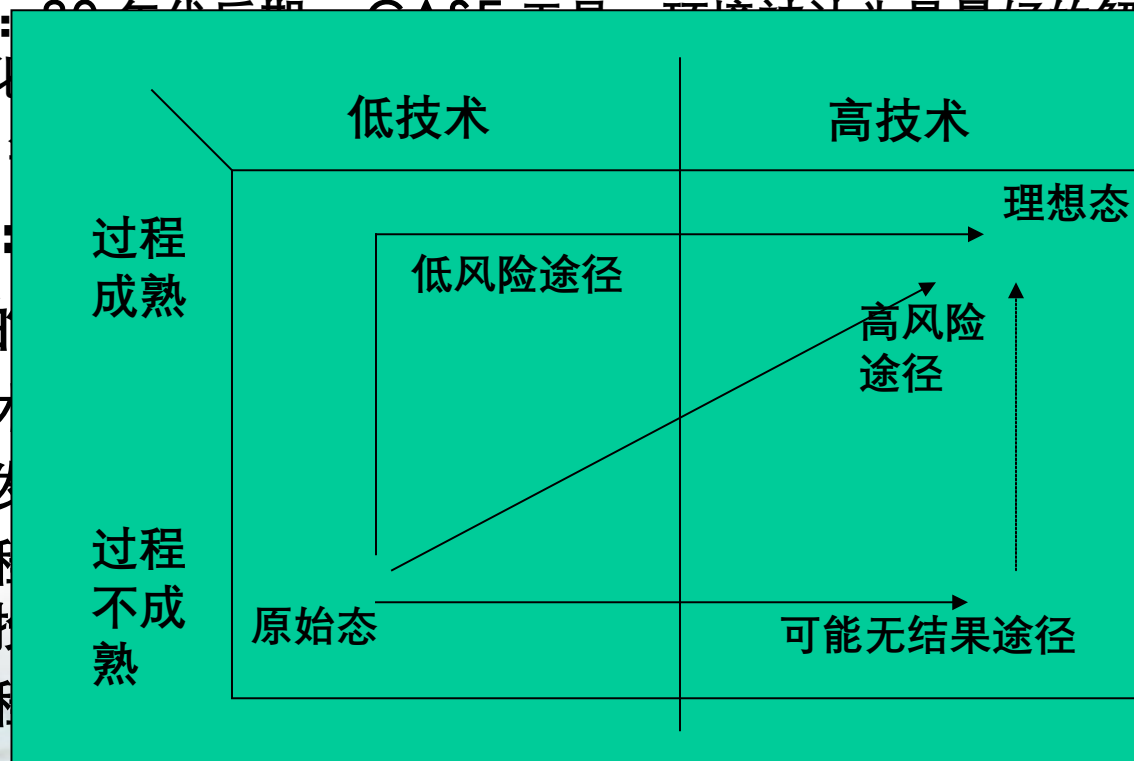
➤ 较长时间的认识：技术的发展是达到上述目标的主要（唯一）途径

- 如：80年代后期，CASE工具，环境设计工具最佳解决方案——通过自动化
- 90

➤ 原因：

➤ 认识的

- 技术
- 开发
- 过程
- 的推
- 过程



得到收益

承诺和长期

京大学



软件工程技术研究热点（1）

❖ 基于构件的软件开发（CBSD）

- CBSD 提供了一种自底向上的、基于预先定制包装好的类属元素（构件）来构造应用系统的途径
- 典型的产品复用模式
- 构件复用的三个基本过程
 - 可复用构件的生产
 - 可复用构件库的管理
 - 基于可复用构件的应用开发（组装）
- CBSD 和中间件发展密不可分
 - CCM/CORBA、JavaBeans/J2EE、DCOM/.Net
- 面临的问题
 - 需要系统化的 CBSD 方法学，需要足够的可复用构件
 - Web Service 和 SOA 的出现，需要 CBSD 的相应调整和发展





软件工程研究热点（2）

❖ 面向服务的体系结构（SOA）

- 某种意义上，可视为 CBSD 面向“软件作为服务”思想的进一步发展
- SOA 支持在更高抽象级别上的复用，将软件开发的关注点从低层的实现转到高层的业务流程组合
- SOA 过程涉及 3 种角色：application builders, service developers, and service brokers；支持松耦合的服务间组装
- SOA 的灵活性和易扩展性适合于产业级系统和大型分布式应用
- SOA 远未成熟
 - 可用服务的数量规模
 - 服务的动态和按需组合
 - 服务的可信性
 - 复合服务的确认和评估……





软件工程研究热点 (3)

❖ 软件过程

- 主要涉及用于评估、支持和改善软件开发活动的方法和技术
- 主要研究包括：
 - 过程模型和过程建模
 - 过程评估和过程改善
- 过程和技术是相辅相成的两方面
- 关于过程的代表性工作
 - ISO 9001 (欧洲 TickIT)
 - CMM/CMMI (CMU SEI)
 - Bootstrap (欧洲 ESPRIT 计划对 CMM 模型的扩展)
 - 面向团队和个人的 TSP 和 PSP





软件工程研究热点 (4)

❖ 需求工程

- 软件开发的早期活动，需求分析活动的系统化发展
- 主要考虑将要开发的软件系统的现实世界目标、功能和约束
- 需求工程是以人为中心的过程，需要多学科技能的综合，包括：计算机科学、逻辑学、认知心理学、人类学、社会学、语言学等
- 基本活动：需求诱导、确认和验证、规约和文档化、可变性和演化管理
- 常见 RE 技术：
 - ERD、DFD、state transition diagram、state machine, state chart and Petri-Net 等
 - OO 建模技术：覆盖了需求的结构、行为和过程方面
 - 特征建模和产品线技术：领域需求建模
- 挑战性问题
 - 更好地对问题域建模
 - 非功能需求建模
 - 需求模型的复用……



软件工程技术研究热点（5）

❖ 模型驱动的体系结构（MDA）

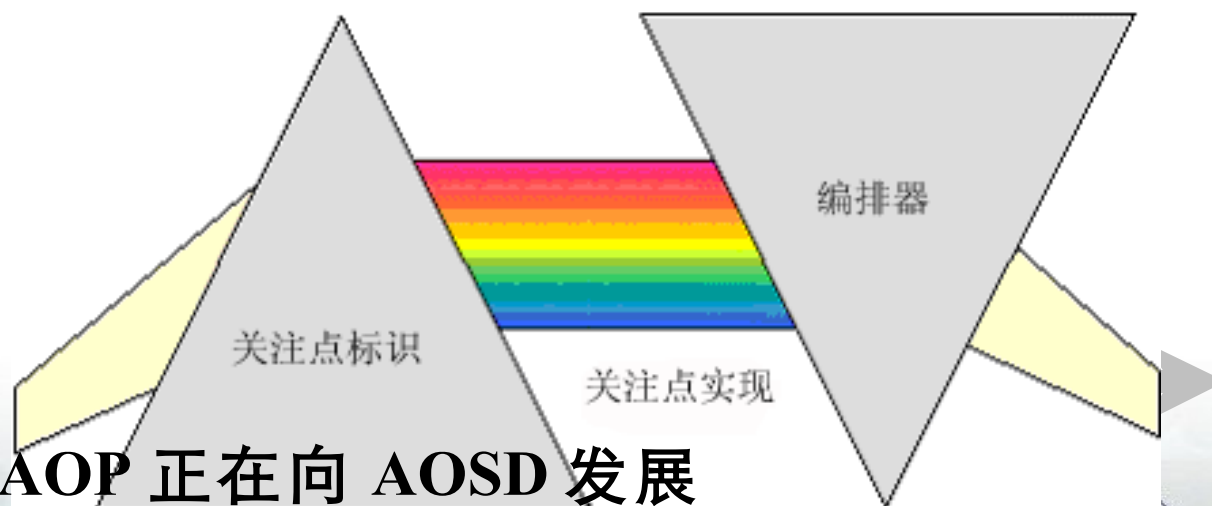
- 一种从系统设计层次出发的、解决中间件间互操作性的新的技术途径
- 关注集成和互操作问题，覆盖了从建模、设计，到构件的构造、组装、集成、部署、管理和演化的软件生命周期的全过程。其一个关键目标就是促进良定义的建模标准和中间件标准的使用
- MDA 中，模型成为一阶实体；主要生命期阶段：
 - 业务需求建模：CIM
 - 不考虑任何与实现平台相关的技术细节，根据需求规约建立和业务逻辑相关的模型：PIM
 - 通过适当的映射机制和工具支持，将平台独立的模型映射到平台特定的模型：PSM；并对 PSM 进行精化和完善
 - 通过工具支持从 PSM 自动产生实现代码
 - 在特定环境部署系统
- 挑战问题
 - 如何描述模型？模型规约的形式化程度
 - 如何自动转换？自动转换工具的研发



软件工程技术研究热点（6）

❖ 面向方面的软件开发（AOSD）

- 将复杂系统中的多种关注点分离，然后分而治之，经验证明是解决问题的一种有效方法
- AOP 将传统方法中分散处理的贯穿特性实现为系统的一阶元素— Aspect，并将它们从类结构中独立出来，成为单独的模块，从而使设计和代码更加模块化和更具结构性



- AOP 正在向 AOSD 发展
- 挑战问题：面向大型应用的实用性



软件工程研究热点（7）

❖ 面向 Agent 的软件工程

- 一种新的软件范型，试图建立基于 Agent 的软件开发的方法学和工具
- 软件有一组自治 Agent 构成，通过灵活的高层交互来求解问题。直觉上，这符合复杂系统的自然描述方式
- 自治 Agent 的引入使得面向 Agent 的软件成为 Internet 环境下软件形态的一种可行技术途径
- 挑战问题：
 - Agent 的实现形式
 - 面向 Agent 的编程语言
 - 和已有编程模型的结合
 -





软件工程技术研究热点（8）

❖ 开源软件开发

➤ 很多因素，技术的和非技术的，经济的和政治的，使得开源运动日益繁荣

➤ 开源项目的数量继增，如，在 <http://sourceforge.net> 上就有上百万个；很多开源应用已达到甚至超越其传统形态竞争者，成功例子如：

■ GNU/Linux 和 FreeBSD 操作系统

■ PostgreSQL 和 MySQL 数据库

■ GNU 编译器 GCC

■ 集成开发环境 Eclipse 等

➤ 传统的软件开发过程和工具将难以有效或高效支持

软件工程技术研究热点 (9)

❖ 可信软件技术

- 软件可信性是**软件质量**的一种特殊的表现形式，它所关注的是**使用层面**的综合化的质量**属性**及其**保障**形式，涉及多个质量属性的集合，涉及这些属性的综合和平衡
- 正确性、可靠性、安全性、保安性、可用性、可维护性、可预测性、……

传统封闭环境

人们只需关心质量属性的局部，如正确性、可靠性等

以系统质量为核心
兼顾使用质量

主观角度→可信度

Internet 开放环境

人们必须关心更多的质量属性及其综合，如生存性、安全性等

以系统质量为基础
以使用质量为核心

客观角度→服务质量

凸显了人们对
质量属性的新关注



其他热点研究领域

- ❖ 程序分析
- ❖ 软件测试
- ❖ 形式化方法
- ❖ 敏捷软件开发过程
- ❖





Internet 带来的挑战

❖ Internet 平台的特征

开放性

动态性

多变性

- □ □ □ □ □ “ □ ” □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □



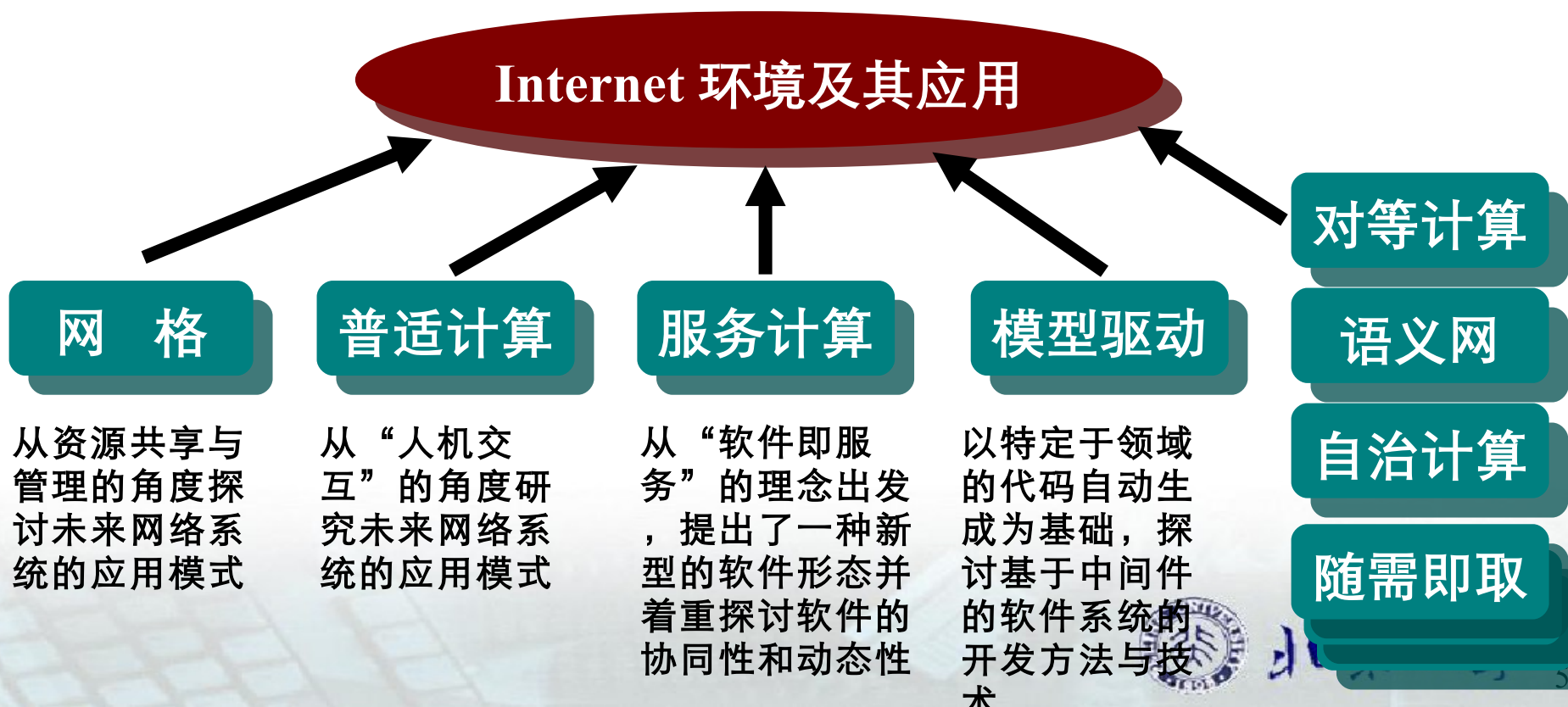
面向 Internet 的新兴研究

❖ 前沿研究十分活跃

- 往往从某种侧面或层次对信息技术提出一种新型模式或理念

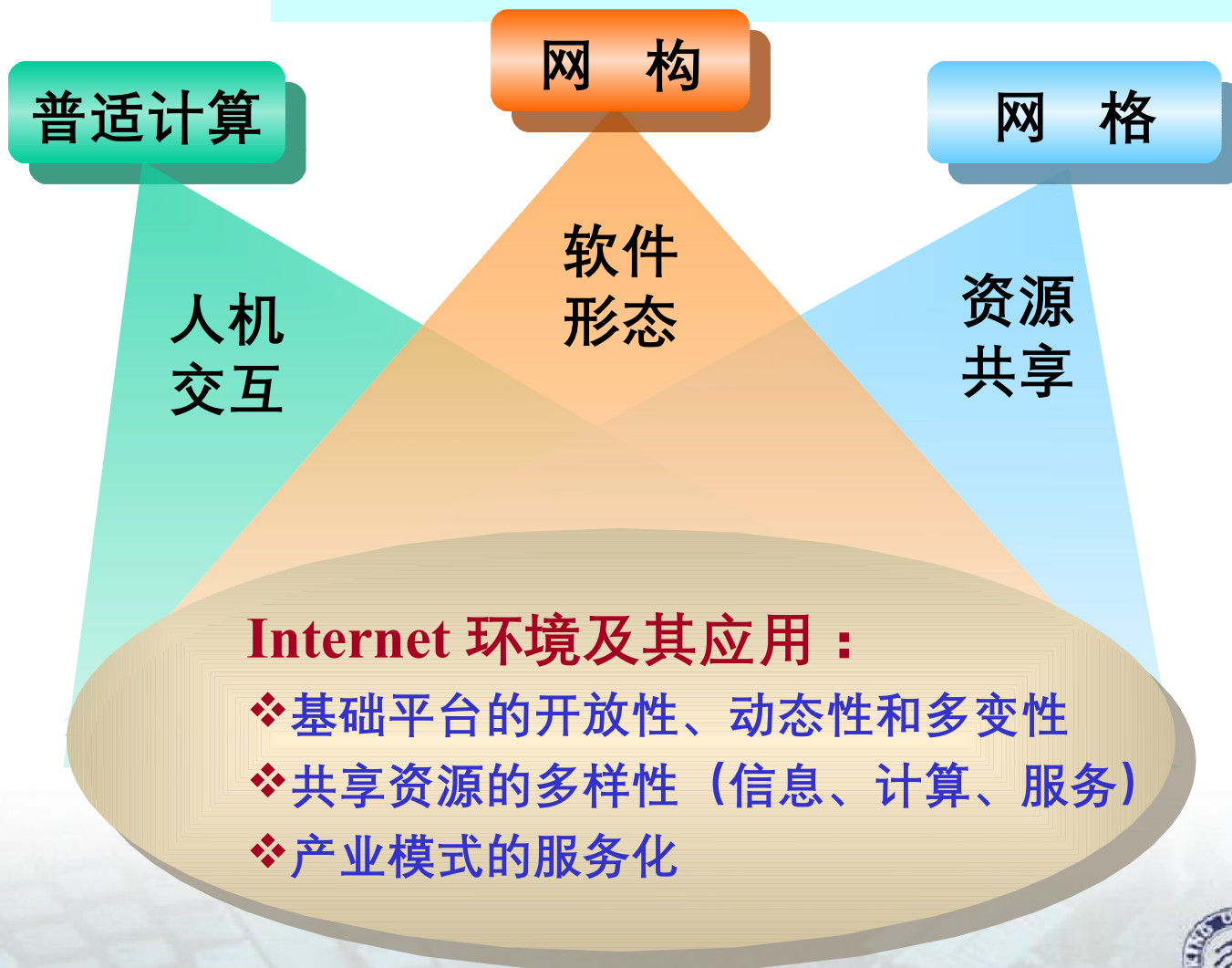
❖ 软件技术驱动力依然存在

- 随着计算平台从单机、机群、局域网到 **Internet** 的转移，软件技术的发展正进入一个变革性阶段





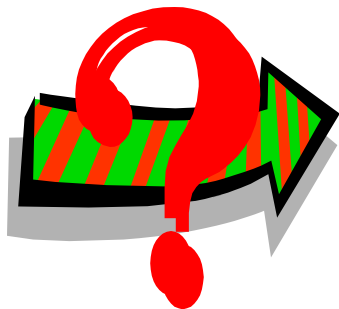
Internet 平台和环境的出现，
对软件形态 技术发展 理论研究
提供了 新的契机 提出了新的问题



未来软件面临的挑战

传统软件

- ❖ 传统软件结构
- ❖ 系统目标的确定性
- ❖ 实体单元的被动性
- ❖ 协同方式的单一性
- ❖ 系统演变的静态性
- ❖ 基于实体的结构分解
- ❖ 经验驱动的软件手工开发模式



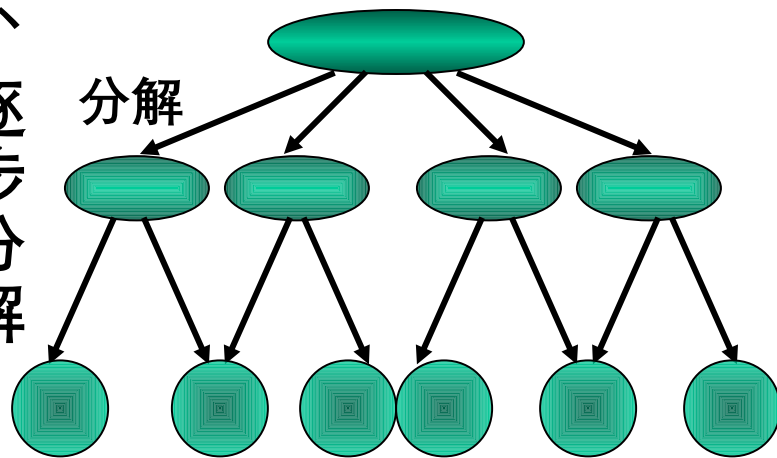
未来软件

- ❖ 新型软件结构
- ❖ 多重不确定性
- ❖ 主动自主性
- ❖ 灵活多面性
- ❖ 系统演化的动态性
- ❖ 基于协同的实体聚合
- ❖ 知识驱动的软件自动生成模式

挑战举例：开发方式的变化

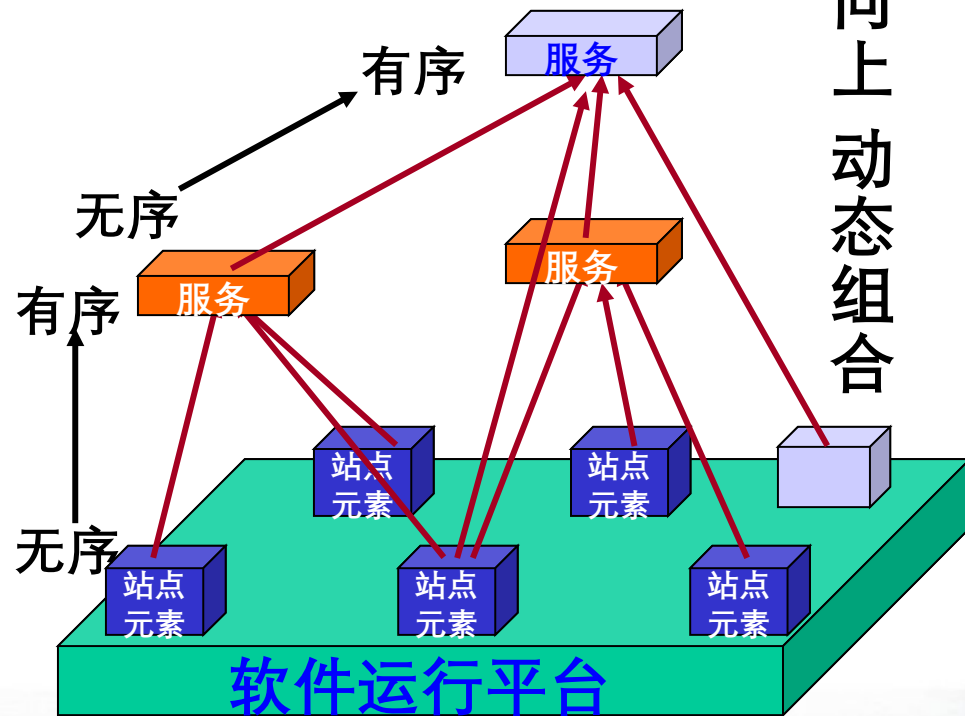
自顶向下
逐步分解

传统软件形态



确定目标、有序控制

新的软件形态



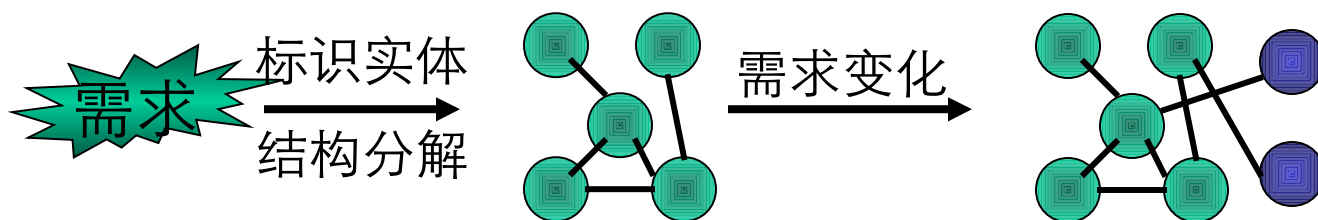
动态目标、渐趋稳态



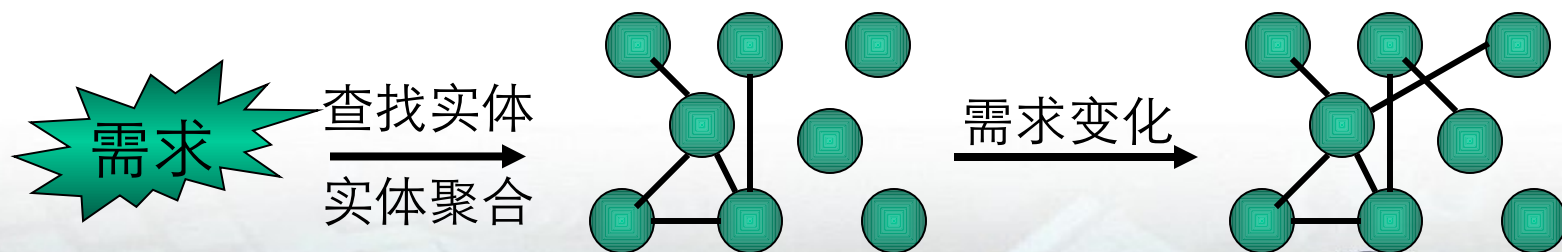
北京大学

挑战举例：开发关注点的变化

传统软件形态基于实体驱动



新的软件形态基于协同驱动



北京大学



挑战举例：对高可信的关注

❖ 可信性

- 安全性 (safety)
- 保安性 (security)
- 可靠性 (reliability)
- 可用性 (availability)
- 可预测性 (predictability)
-

❖ 软件的可信性

- 从用户的角度
 - 软件是否能按用户的期望提供安全、可靠的服务
- 从提供商的角度
 - 将软件 / 服务提供给可信的用户使用
- 从开发者的角度
 - 保证所开发软件产品的可信

❖ 如何实现软件的高可信

- 开发上 → 可信软件工程
 - 如可信需求工程，可信软件开发方法、过程和工具
 - 形式化方法
- 运行上 → 可信运行平台
 - 可信的操作系统、中间件、数据库
 - 可信的管理与监控设施
- 评估上 → 科学的、可操作的评估体系

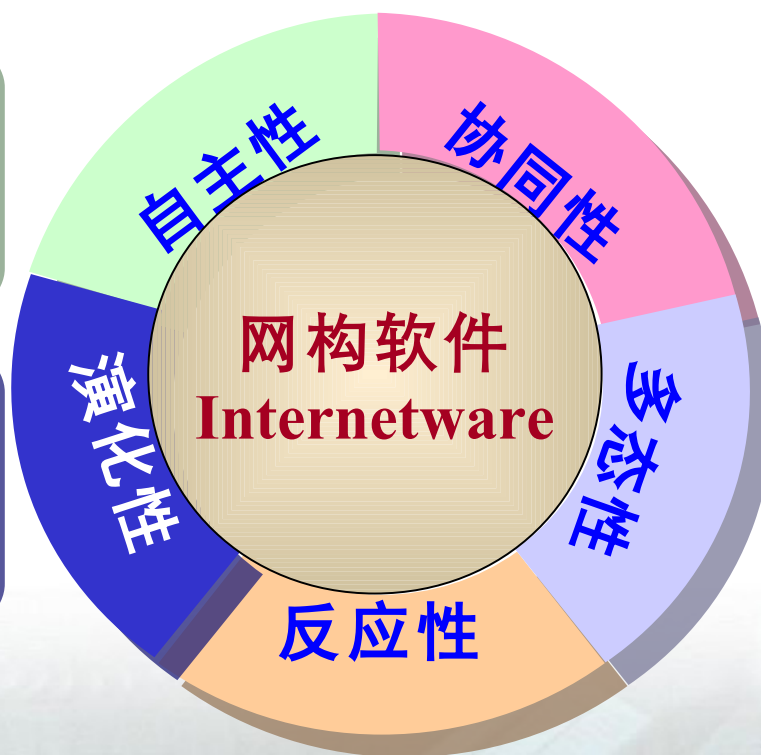


网构软件

网构软件是 Internet 开放、动态和多变环境下
软件系统基本形态的一种抽象
既是传统软件结构的自然延伸
又具有区别于传统软件形态的独有的基本特征

自主性：
软件实体具有相对
独立性、主动性和
自适应性

演化性：
元素数目可变性、
结构关系和形态的
动态可调性



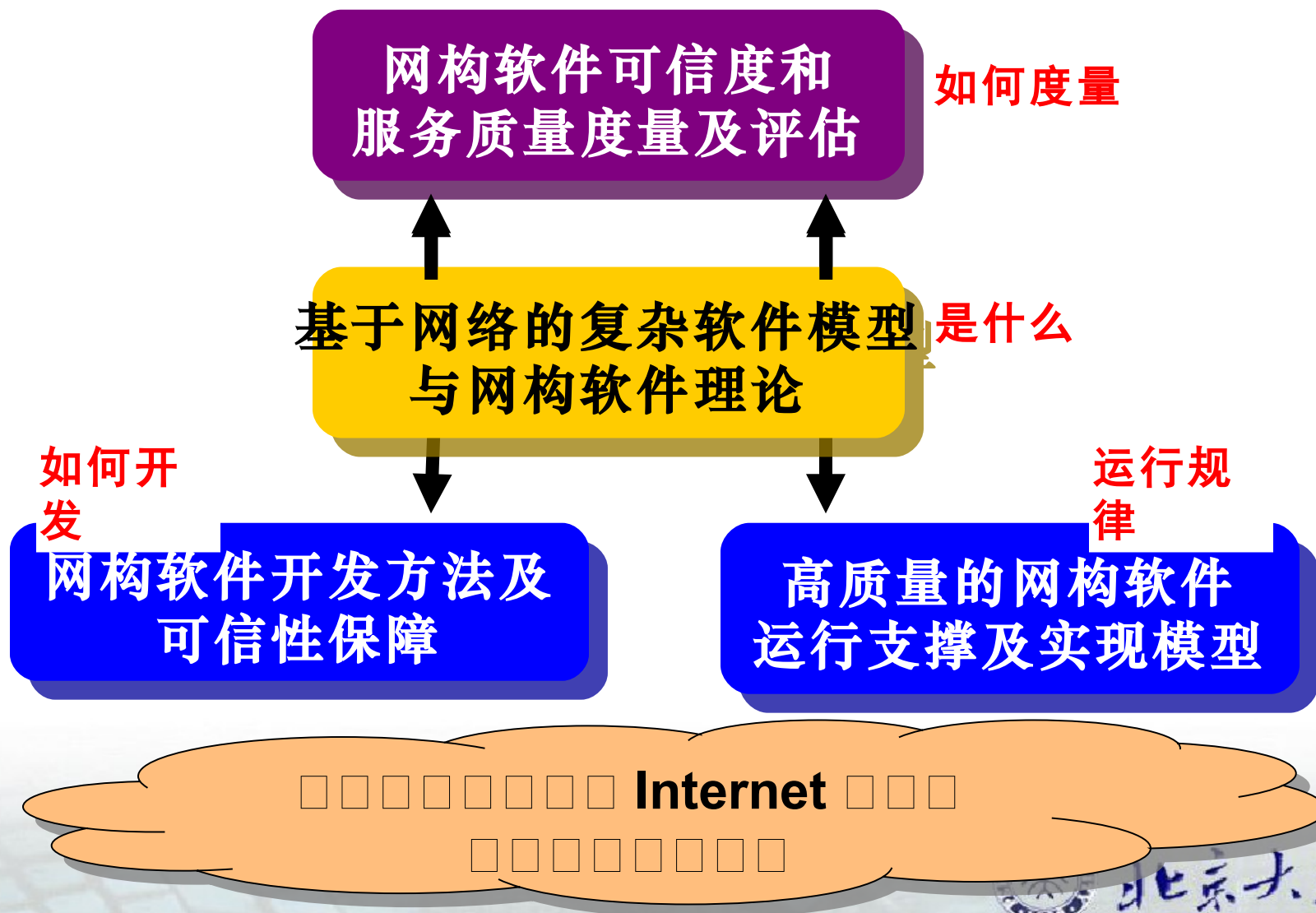
协同性：
软件实体之间多种
方式的互连、互通
、协作和联盟

多态性：
软件实体具有目标
制导和多目标的特
征协同能力

反应性：软件实体
具有感知外部运行
和使用环境的能力



网构软件技术体系





关于本课程的学习

❖ 软件工程课程教学的问题

- 教学形式抽象，似乎总在哲学、方法学层次说问题，无实践经验难以理解
- 软件工程覆盖面较宽，而教学时间有限，难以深入

❖ 建议

- 排遣“枯燥”情绪，认识上高度重视，为未来打基础
- 当成知识来学，在记忆中努力理解基本概念，理顺逻辑
- 类比传统行业来学，软件工程的很多理念都能在我们现实生活中找到源头
- 尽可能在自己的课程或科研实践中尝试应用



谢谢

