大学计算机-计算思维导论

南京理工大学计算机科学与工程学院

Fall, 2015

课程安排

课程对象：理工类大学新生

时间：　5周

学分：　1学分

讲师：　张静 jzhang@njust.edu.cn 18724002066

课程主页：http://jz81.github.io/course/ict/ict.html

考试时间在学期末（请关注课程主页）

第1讲 计算机、计算与计算思维

当今信息社会，每个人都离不开计算机，计算机的出现已改变了人们很多的工作和生活习惯！

◆学习大学计算机，应该学习计算思维，学习计算机科学家进行问题求解的思维方式！

计算机是什么？

形形色色的计算机

传统“计算机器”

多样化的“计算机器”，各种设备的“大脑”系统

形形色色的计算机


中国联通 3G
23:05
49 %
编辑
出发
12.9 公里 22 分钟（高交通流量）
荔枝角
Lai Chi Kok
深水埗区
Sham Shui Po
昂船洲
Stonecutters Island
维多利亚港
Victoria Harbour
Google
搜索
路线
40 chips
elsewhere
ATM, cell phone, PDA,
80 chips家用电器
TV, DVD player, phone, games, washer, dryer, dishwasher, etc.
40 chips
工作
printers, scanners, PC, phone systems, etc.
每个人每天使用250个芯片
70 chips
汽车
door opener, ABS, airbag, GPS, radio, engine control, etc.

形形色色的计算机

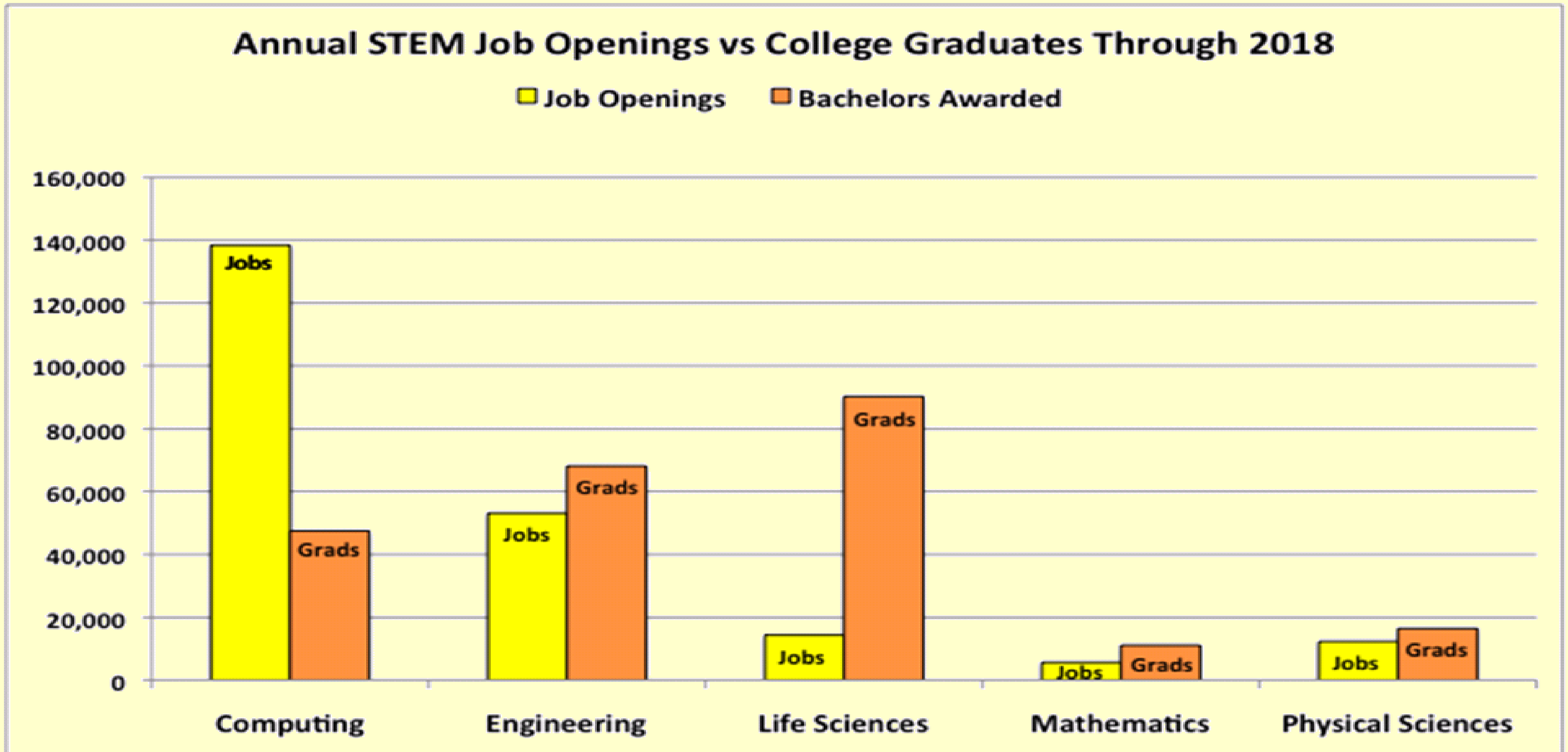
马克.安德森 ---前Netscape公司创始人，现风险投资人

软件正在占领全世界

为什么要学习和怎样学习
大学计算机课程

计算学科的供需关系

Understanding Supply Versus Demand

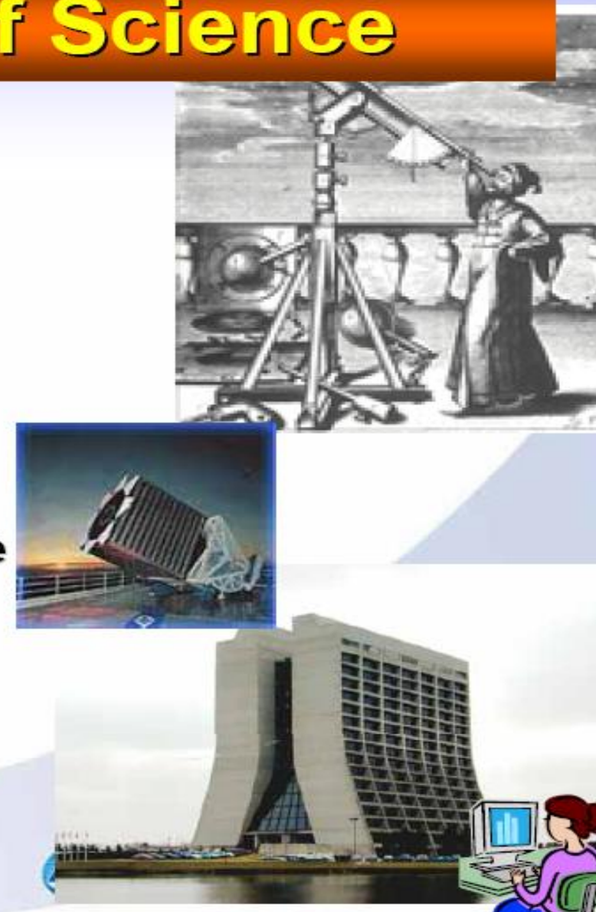


Data Sources: *US-BLS Employment Projections, 2008-2018* (http://www.bls.gov/emp/ep_table_102.pdf), *National Science Foundation Division of Science Resource Statistics* (http://www.nsf.gov/statistitcs/nsf08321/tables/tab5.xls), and *National Center for Education Statistics* (http://nces.ed.gov/programs/digest/d08/tables/dt08_286.asp).

计算科学 与 信息探索科学

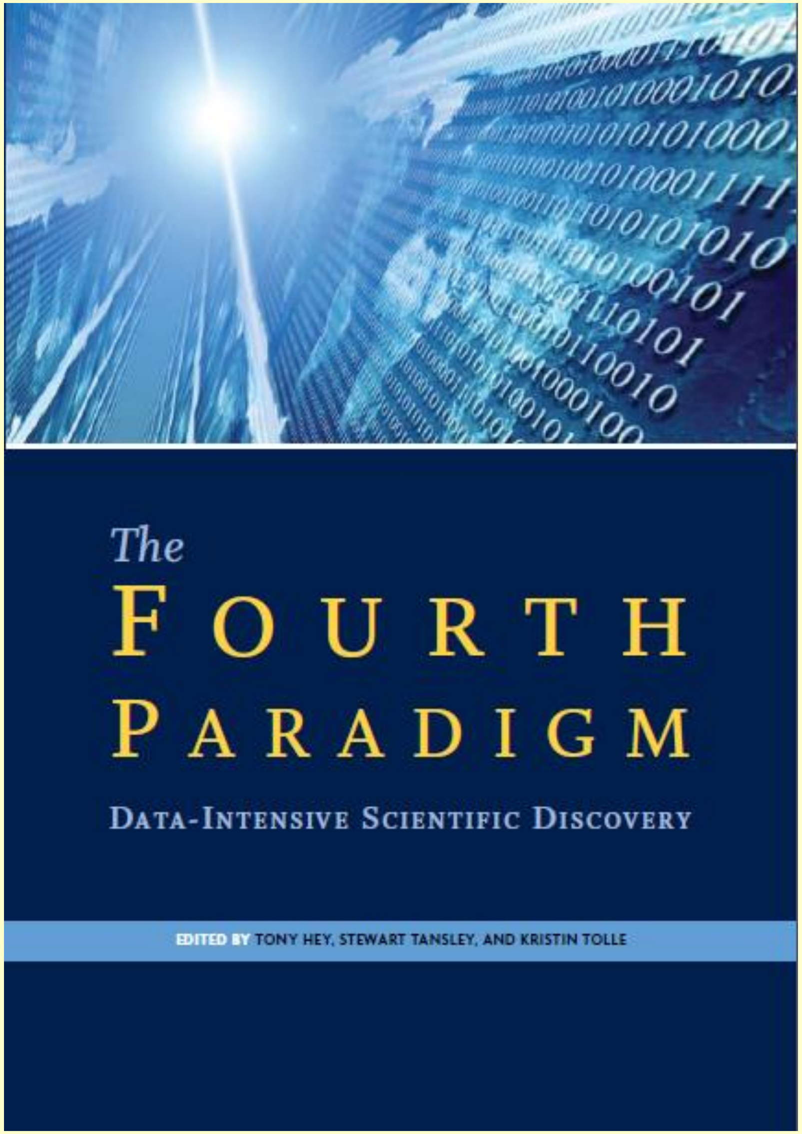
The Evolution of Science

- **Observational Science**
  - Scientist gathers data by direct observation
  - Scientist analyzes Information
- **Analytical Science**
  - Scientist builds analytical model
  - Makes predictions.
- **Computational Science**
  - Simulate analytical model
  - Validate model and makes predictions
- **Science - Informatics**
  Information Exploration Science
  Information captured by instruments
  Or Information generated by simulator
  - Processed by software
  - Placed in a database / files
  - Scientist analyzes database / files



大数据时代的第四范式

- 地球和环境科学
- 健康和社会福利
- 科学基础设施
- 学术社区



各学科人才的计算思维/计算能力需求

计算思维的学习和训练

计算思维/计算能力

计算机及其通用计算手段的应用

知识/技能

Gap

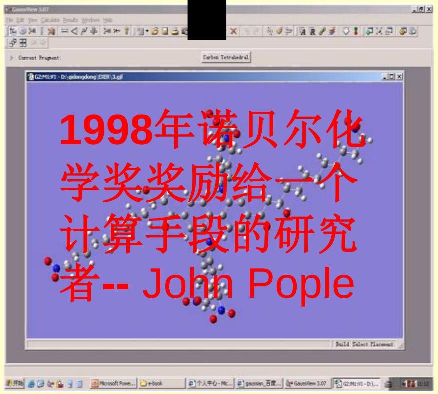
支持各学科研究创新的新型计算手段

应用计算手段进行各学科研究和创新

当前的非计算机专业计算机关注点

非计算机专业学生的未来计算能力

化学学科工作者利用计算手段进行学科的科学研究



1998年诺贝尔化学奖奖励给一个计算手段的研究者-- John Pople

计算思维

- ●“计算机”的思维: 计算机是如何工作的? 计算机的功能是如何越来越强大的?
- ●利用计算机的思维: 现实世界的各种事物如何利用计算机来进行控制和处理?
- ◆计算思维(Computational Thinking)是运用计算机科学的基础概念去**求解问题**、**设计系统**和**理解人类行为**，其本质是**抽象**和**自动化**----from 周以真。

训练与实践

不断训练，不断理解，才能找出本质，才能创新

“看山还是山，看水还是水”

“众里寻她千百度，蓦然回首，那人却在灯火阑珊处”

联想与贯通

浮想联翩，由此及彼，才能发现

贯通，看得远，才能认识准确

“看山不是山，看水不是水”

“衣带渐宽终不悔，为伊销得人憔悴”

概念与知识

“看山是山，看水是水”

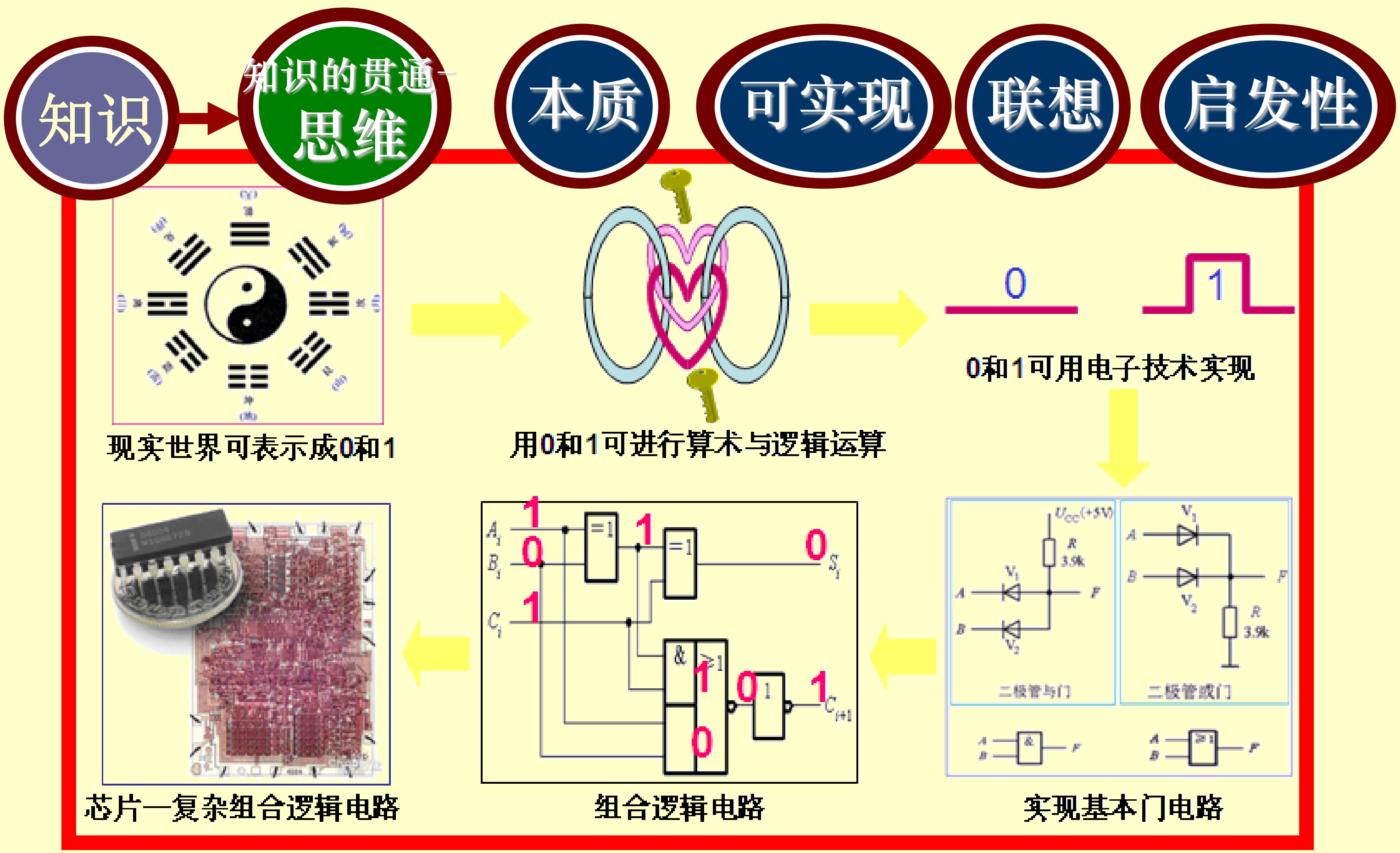
“昨夜西风凋碧树，独上高楼，望尽天涯路”

From 王国维“人间词话”关于“境界”的阐述

思维是创新的源头，技术与知识是创新的支撑

知识 vs. 贯通知识的思维—计算思维

---本门课程学习需要注意的



知识 vs. 思维 vs. 能力

视野：
宽度---知识面宽，前沿性
深度---贯通性，深入性

打通知识脉络，
融贯各门课程，
内功强化基础，
外功灵活应变。

能力
(Ability & Capability)

知识与视野拓展
Knowledge Expansion

云计算与云服务

企业资源规划
与供应链管理

....

技能：练习，练习，… …

知识/技能
(Technique/Skill)

训练与掌握
Training

计算机语言
与程序设计

数学建模
或
非数学建模

数据库

....

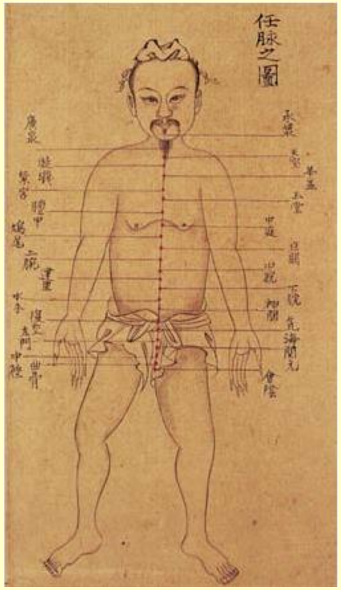
思维
(Thought/Thinking)

启发与理解
Understanding

大学计算机
计算机专业导论
（计算思维导论）

....

思维：好奇、思考、联想、贯通



- 能力----内功(贯通的脉络)
- 实践----锻炼,使脉络贯通
- 思维----脉络(穴位链)
- 知识----穴位

计算与自动计算

计算学科的计算 vs. 数学学科的计算

简单计算I： **数据**计算，**计算规则**，应用计算规则进行计算并获得计算**结果**

$$3+2=5;\quad 5-4=1;\quad 3\times 2=6;$$

复杂计算II： f(x)，函数，计算规则及其简化计算方法，便于人应用规则进行计算，获得计算结果

$$\int x^{-1}dx=\int\frac{1}{x}dx=\ln|x|+c$$

复杂计算III： 如丢番图方程，判定，计算规则，人可能无法完成但却可由机器自动完成，借助于机器获得计算结果

$$a_1x_1^{b_1}+a_2x_2^{b_2}+\ldots+a_nx_n^{b_n}=c$$

◆ “人”计算 vs. “机器”计算

例如：求$ax^2+bx+c=0$的根

人-求解	机器-求解
$x=\dfrac{-b\pm\sqrt{b^2-4ac}}{2a}$ 利用上述公式计算得到x值	(1)从-n到n，产生x的每一个整数值； (2)将其依次代入到方程中计算； (3)如果其值使方程式成立，则即为其解；否则不是

人进行计算：

- 规则可能很复杂, 但计算量却可能很小
- 人需要知道具体的计算规则
- 特定规则，只能求：

$$a_1x^2+a_2x=c$$

机器-自动计算：

- 规则可能很简单, 但计算量却很大
- 机器也可以采用人所使用的计算规则
- 一般性的规则，可以求任意：

$$a_1x_1^{b_1}+a_2x_2^{b_2}+\ldots+a_nx_n^{b_n}=c$$

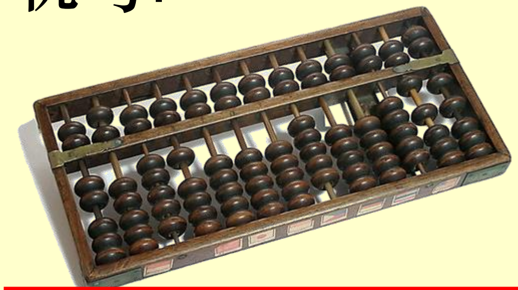
自动计算要解决的几个问题: 表示-存储-执行

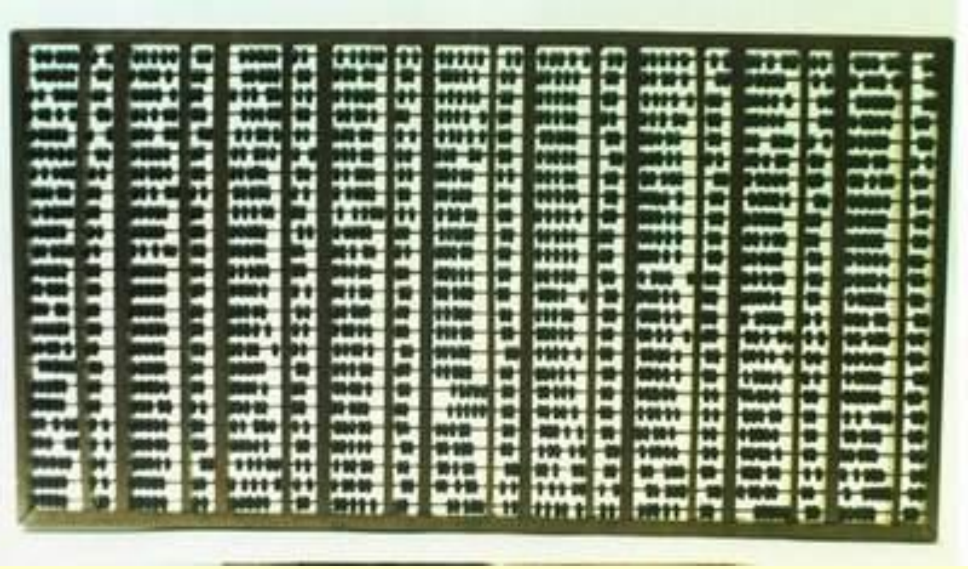
- "数据" 的表示
- "计算规则" 的表示
- 数据与计算规则的 "自动存储"
- 计算规则的 "自动执行"

$$a_1x_1^{b_1}+a_2x_2^{b_2}+\ldots+a_nx_n^{b_n}=c$$

算盘能被认为是计算机吗?

珠算珍品





九层算盘

一、加法口诀

	直加	满五加	进十加
一：	一上一	一下五去四	一去九进一
二：	二上二	二下五去三	二去八进一
三：	三上三	三下五去二	三去七进一
四：	四上四	四下五去一	四去六进一
五：	五上五	五去五进一	
六：	六上六	六去四进一	六上一去五进一
七：	七上七	七去三进一	七上二去五进一
八：	八上八	八去二进一	八上三去五进一
九：	九上九	九去一进一	九上四去五进一

二、减法口诀

	直减	破五减	退位减
一	一下一	一上四去五	一退一还九
二	二下二	二上三去五	二退一还八
三	三下三	三上二去五	三退一还七
四	四下四	四上一去五	四退一还六
五	五下五	五退一还五	
六	六下六	六退一还四	六退一还五去一
七	七下七	七退一还三	七退一还五去二
八	八下八	八退一还二	八退一还五去三
九	九下九	九退一还一	九退一还五去四

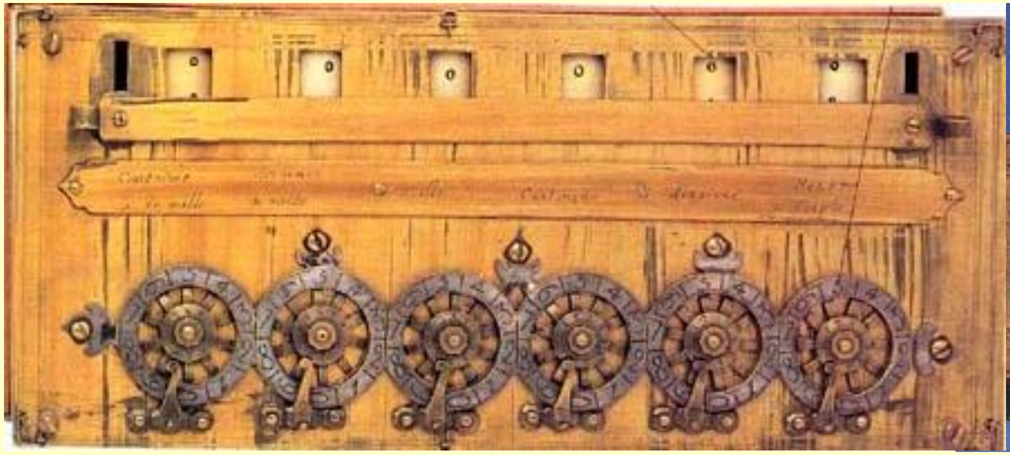
帕斯卡与帕斯卡机----(准)第一台机械计算机

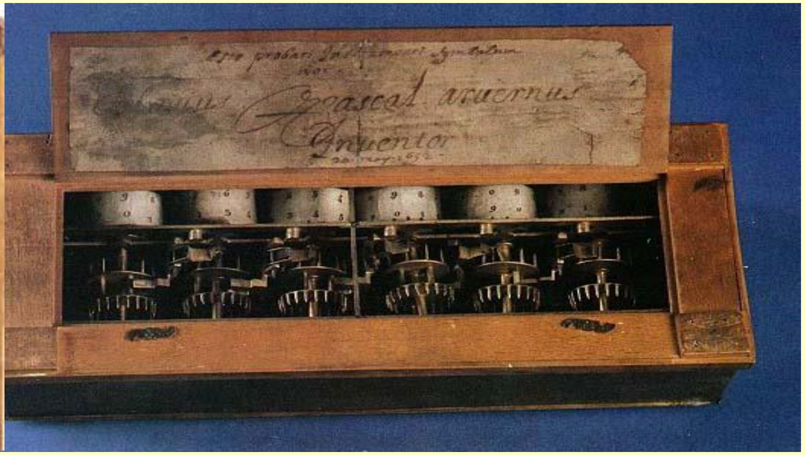
◆ **Blaise Pascal** (1623～1662)

◆1642年研制成功一种齿轮式计算机器

帕斯卡机的意义：它告诉人们**"用<u>纯机械装置</u>可代替人的思维和记忆"**。开辟了<u>自动计算</u>的道路。

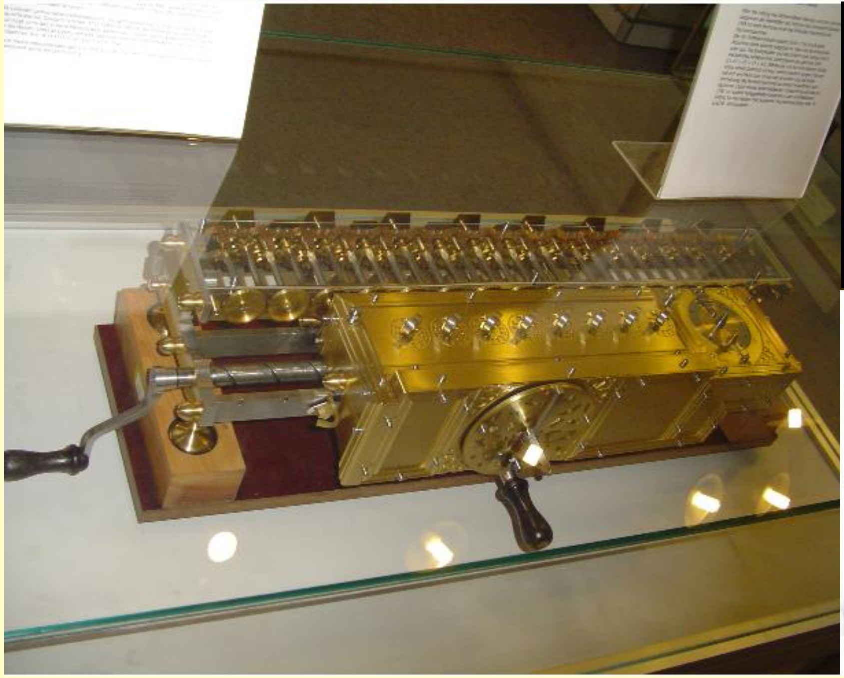
帕斯卡，B.





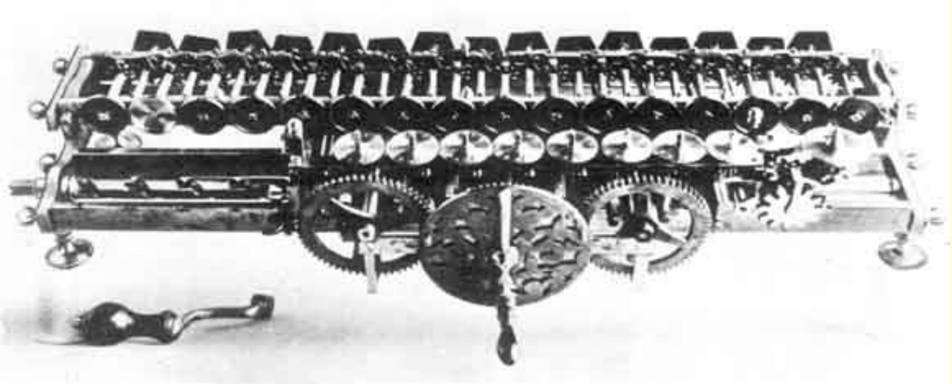
莱布尼茨

- Gottfried Wilhelm Leibniz (1646～1716)，德国数学家。
- 莱布尼茨机的意义：**连续重复自动执行**。
- 提出了<u>二进制数</u>及其<u>计算规则</u>；
- <u>数理逻辑</u>的创始人



是基于十进制设计机器，还是基于二进制设计机器？

如果基于二进制设计机器，那其处理规则又是怎样的呢？



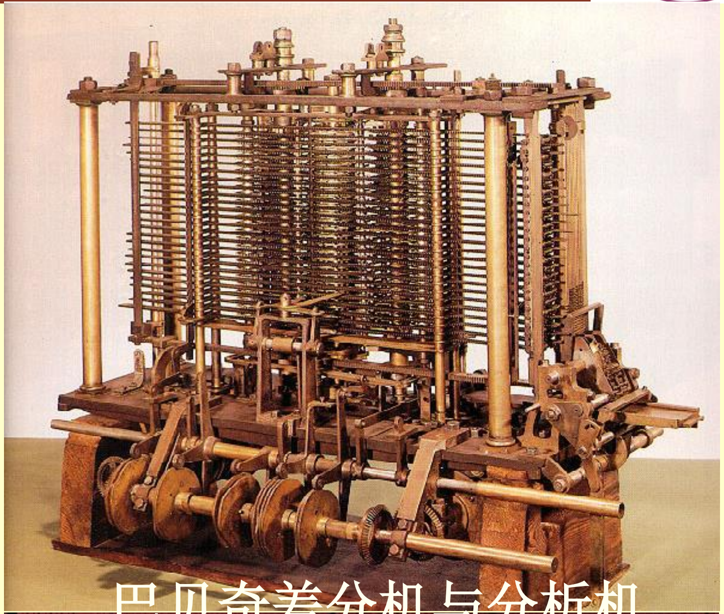
其他重要工作

◆1834年：巴贝奇(**Charles Babbage**)，<u>**分析机**</u>的概念----<u>**可执行程序的机器**</u>。

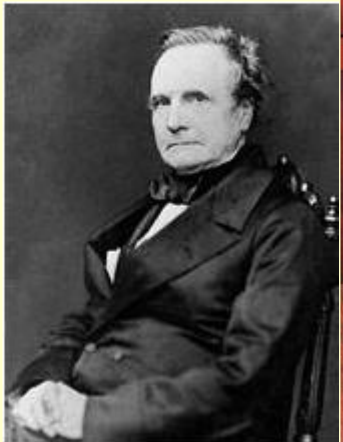
◆1805年： 杰卡德(J.Jacquard)，<u>**打孔卡**</u>，实践了<u>**输入手段**</u>问题。

◆1854年：布尔创立<u>**布尔代数**</u>，为<u>**数字计算机的电路设计**</u>提供了理论基础。

◆… …(请同学课后补充)



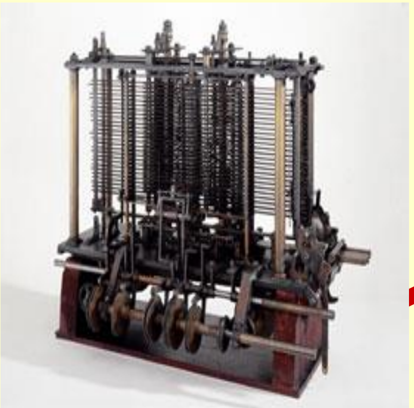
巴贝奇差分机与分析机



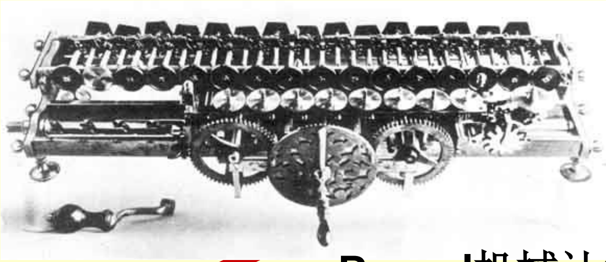
机械计算的简要发展历程是怎样的?

◆从表示-自动存储-自动执行的角度

现代计算机：一般程序



Babbage机械计算机: (特定)程序

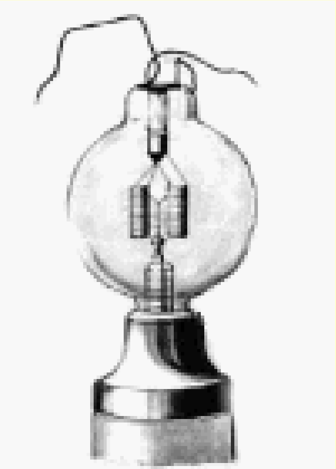


Pascal机械计算机: 自动计算

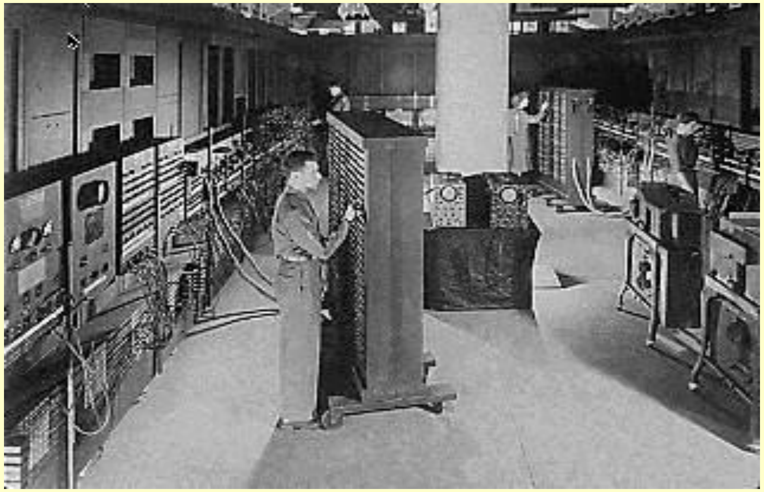
计算辅助工具

电子自动计算-元器件

电子管时代的计算机器



人类第一只电子管
(真空二极管),1895



电子管计算机ENIAC,1946年,17468只电子管

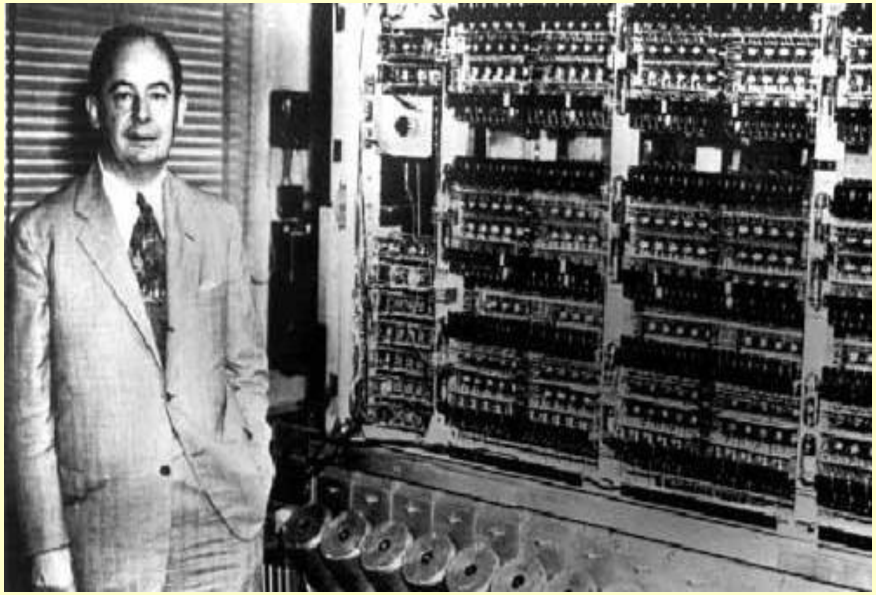
存储0和1的元器件

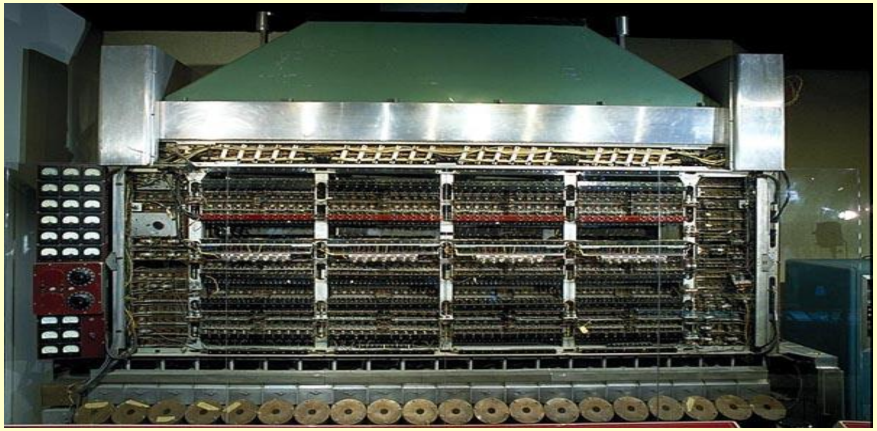
电子管时代的计算机器

◆ 冯.诺伊曼(Von Neumann)电子计算机EDVAC问世

◆将运算和存储分离，运算速度却比拥有18000个电子管的“ENIAC”提高了10倍

◆结构上的创新：“冯.诺伊曼计算机”。





?

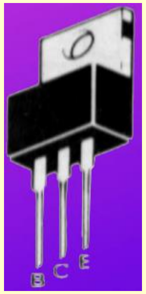
运算规则和数据是否可事先存储于存储器中，以便机器连续的执行呢？

运算和存储怎样分离呢？

晶体管时代的计算机器

"The first transistor ever assembled, invented in Bell Labs in 1947." Photo and text from Porticus.org, www.porticus.org/bell/belllabs_transistor.html. (Follow that link to see more historical documents and images about Bell Labs and the transistor.)

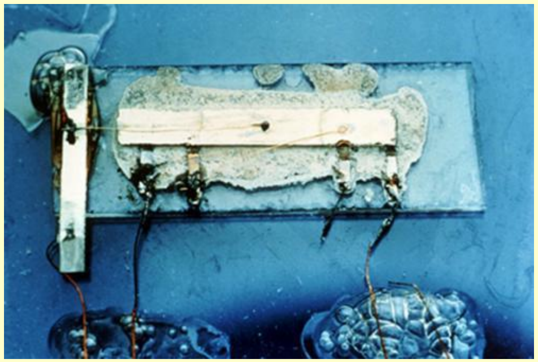
人类第一只**晶体管**
(真空二极管),1947



第一台晶体管计算机TRADIC，1953

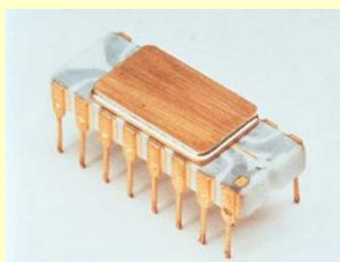
? 怎样使体积更小? 可靠性更高? 可控性更灵活呢?

集成电路时代的计算机器

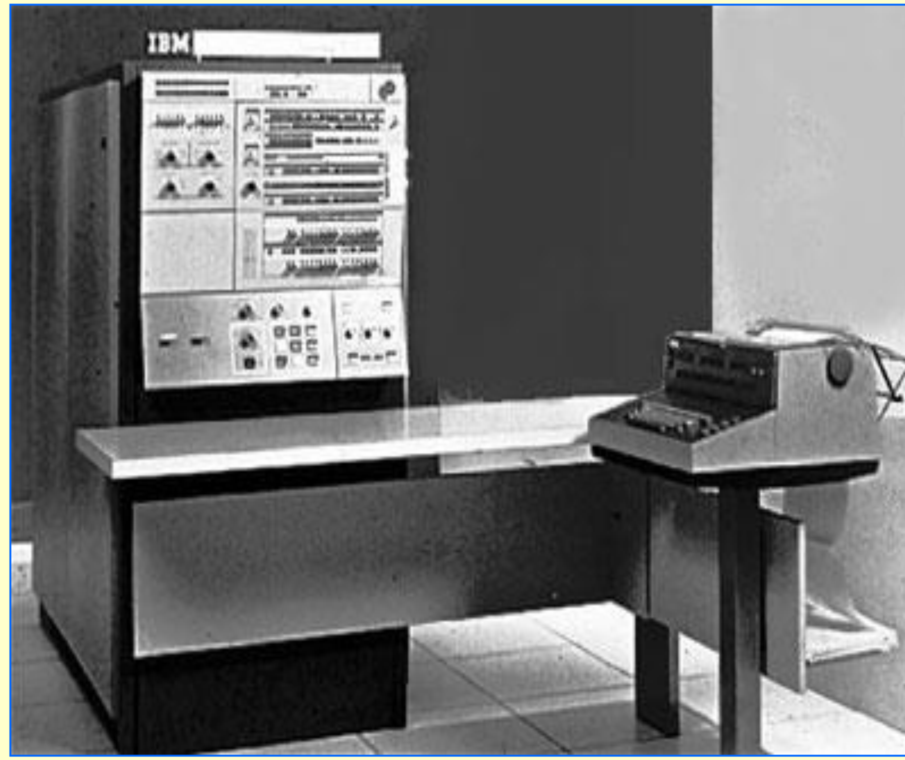


集成电路的发明,1959

J.Kilby，集成电路发明者



封装后的集成电路芯片

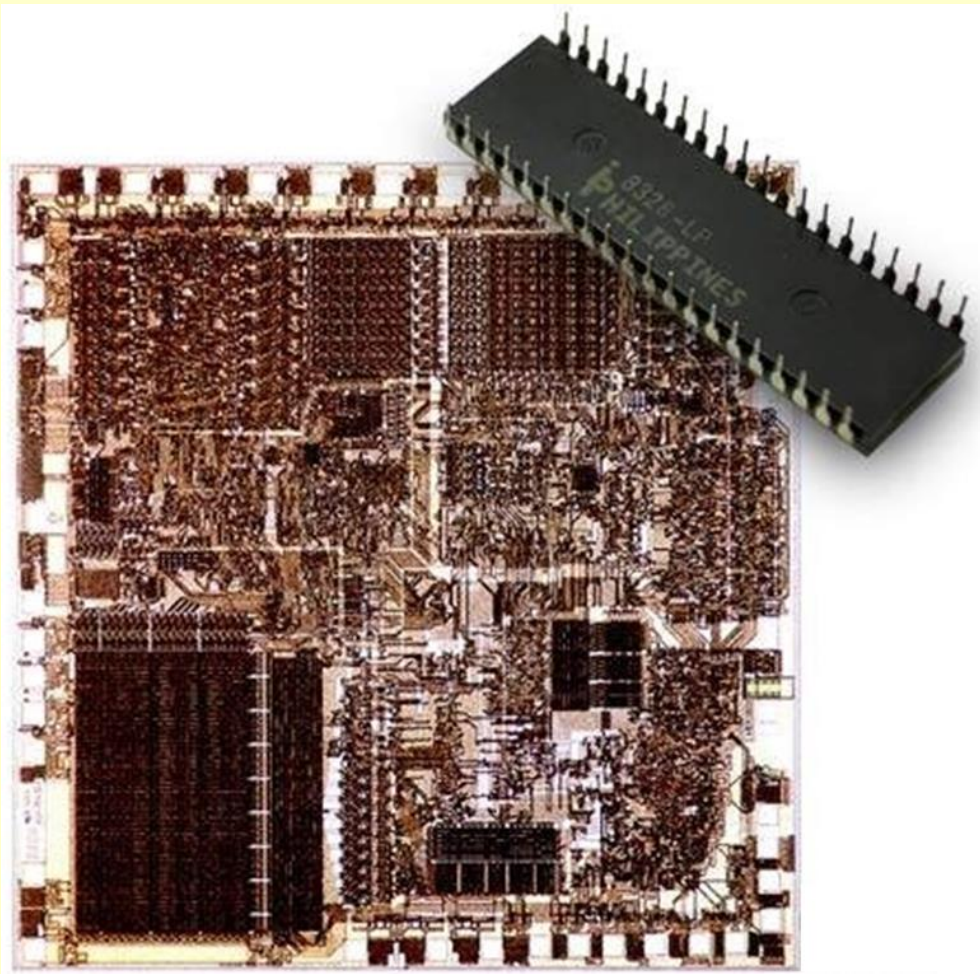


第三代计算机IBM360,1964

能否将复杂的电路封装后作为新电路设计的元件呢?

复杂的电路 ➔ 集成 ➔ 封装 ➔ 应用?

超大规模集成电路(VLSI)时代的计算机器



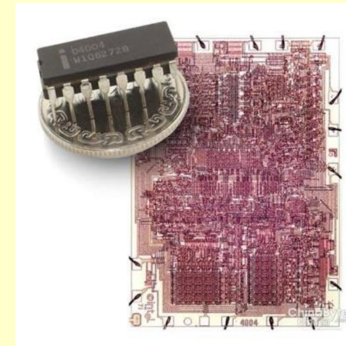
VLSI芯片及其封装的内部电路

摩尔定律----每18个月芯片能力增长一倍

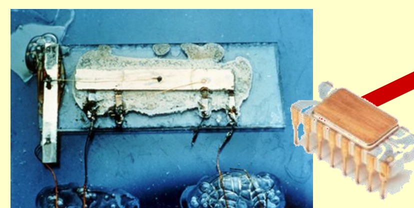
第四代计算机—个人计算机,1981

自动计算中的元器件的发展

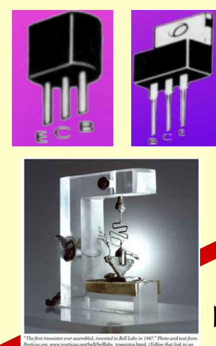
◆从表示-自动存储-自动执行的角度



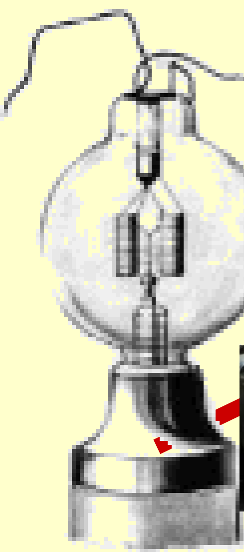
超大规模集成电路(VLSI)



集成电路：可自动实现一定变换的元件



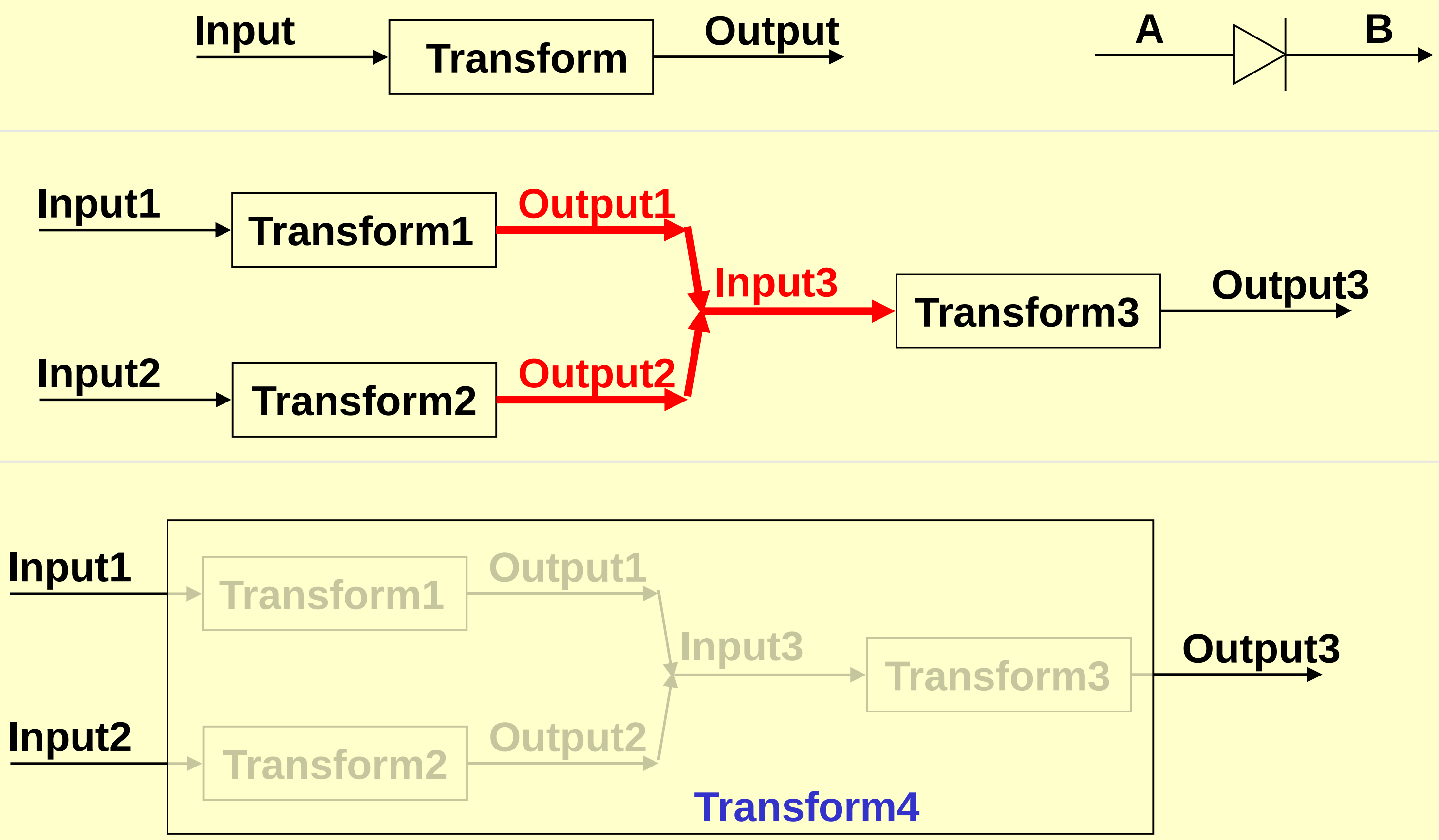
晶体管




电子管：可自动控制0和1变化的元件

- ●体积越来越小；
- ●可靠性越来越高；
- ●电路规模越来越大；
- ●速度越来越快；
- ●功能越来越强大；

自动计算中的元器件的发展启示



电子自动计算-计算机系统

计算机系统要解决的几个问题

- "存储与计算"----微处理器
- "输入"----如何将外部信息输入到计算机中?
- "输出"----如何将计算机中信息输出到外界(显示或打印)?
- "永久存储"----如何将计算机中的信息永久保存?

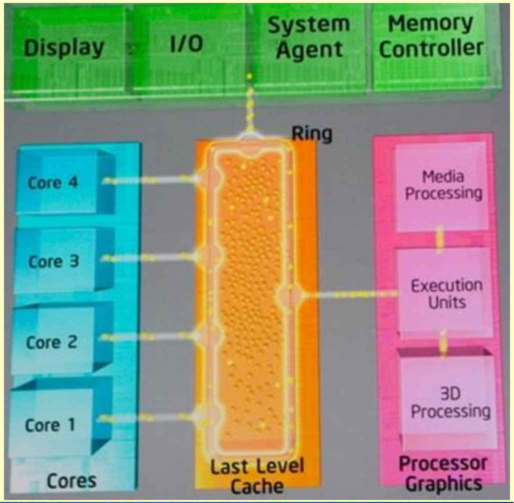
微处理器的发展

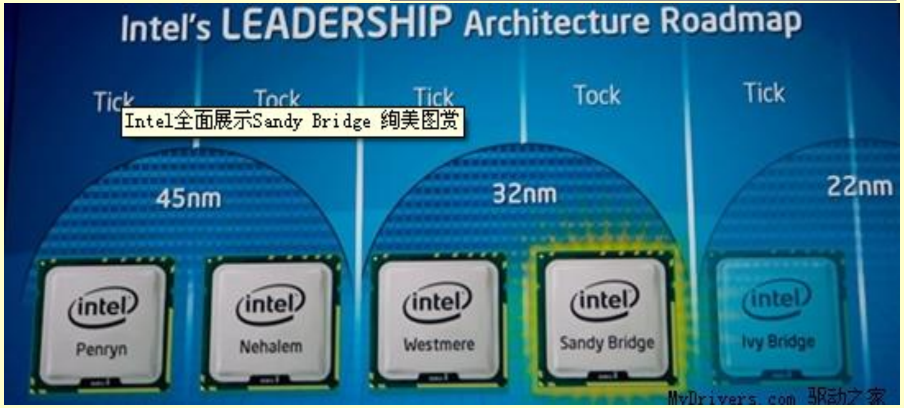
◆**字长**：8位➔16位➔32位➔64位

◆**主频**：几MHz➔几百MHz➔几GHz

◆**晶体管数量**：几万➔几百万➔几亿颗

◆**功能/规模**：微处理器➔**微处理器+协处理器(浮点运算)➔微处理器+图形处理单元GPU➔微处理器+3D处理器+多媒体处理器➔多核微处理器**





8086/8088	Intel 80286	Intel 80386	Intel 80486	Pentium	Pentium II	Pentium III	双核 CORE	八核-十核 CORE i 系列
1974	1982	1985	1989	1993	1997	1999	2005	2010

存储设备

发展水平

纳米存储/量子存储

固态硬盘

USB Removable disk

FlashRAM

光盘存储(CD-ROM, CD R/W, DVD)

磁盘存储(硬盘与软盘)

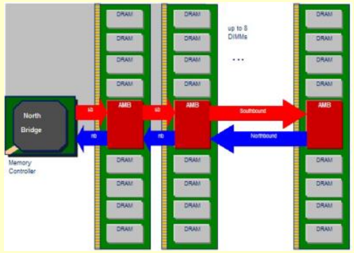
半导体存储(ROM/RAM)

磁带/磁芯/磁鼓存储

汞延迟线

存储设备

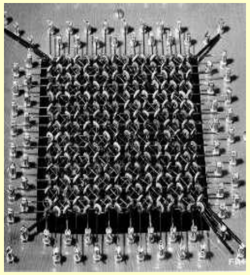
- 体积越来越小
- 容量越来越大
- 访问速度越来越快
- 可靠性越来越高
- 功耗越来越低
- 持久性越来越好



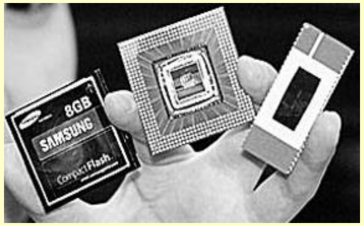
DRAM半导体存储器芯片

固态硬盘

U-Disk



磁芯存储器



FlashRAM存储器

类别

输入设备

发展水平

感知输入
(接触式感知输入
非接触式感知输入)

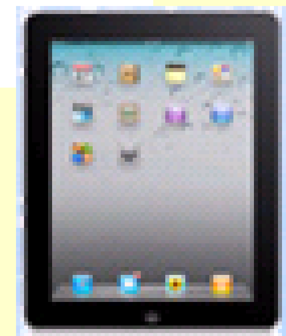
鼠标输入
(点输入)

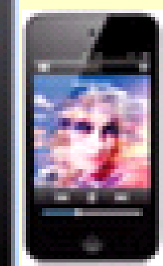
键盘输入
(符号输入)

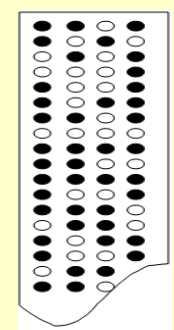
穿孔纸带
(0/1输入)

输入设备

类别







输出设备---显示及显示控制

发展水平

3D显示器：3维图形

数字显示器：高清图形
(液晶、等离子技术)

CRT: 数字光栅扫描显示器
(基于内存的显示: 输出图形)

CRT: 字符发生器
(向量式模拟显示器: 输出字符)

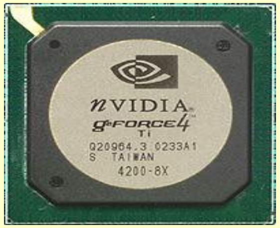
CRT：阴极射线管
(模拟显示器：黑白与彩色)

输出设备
(显示器)

类别

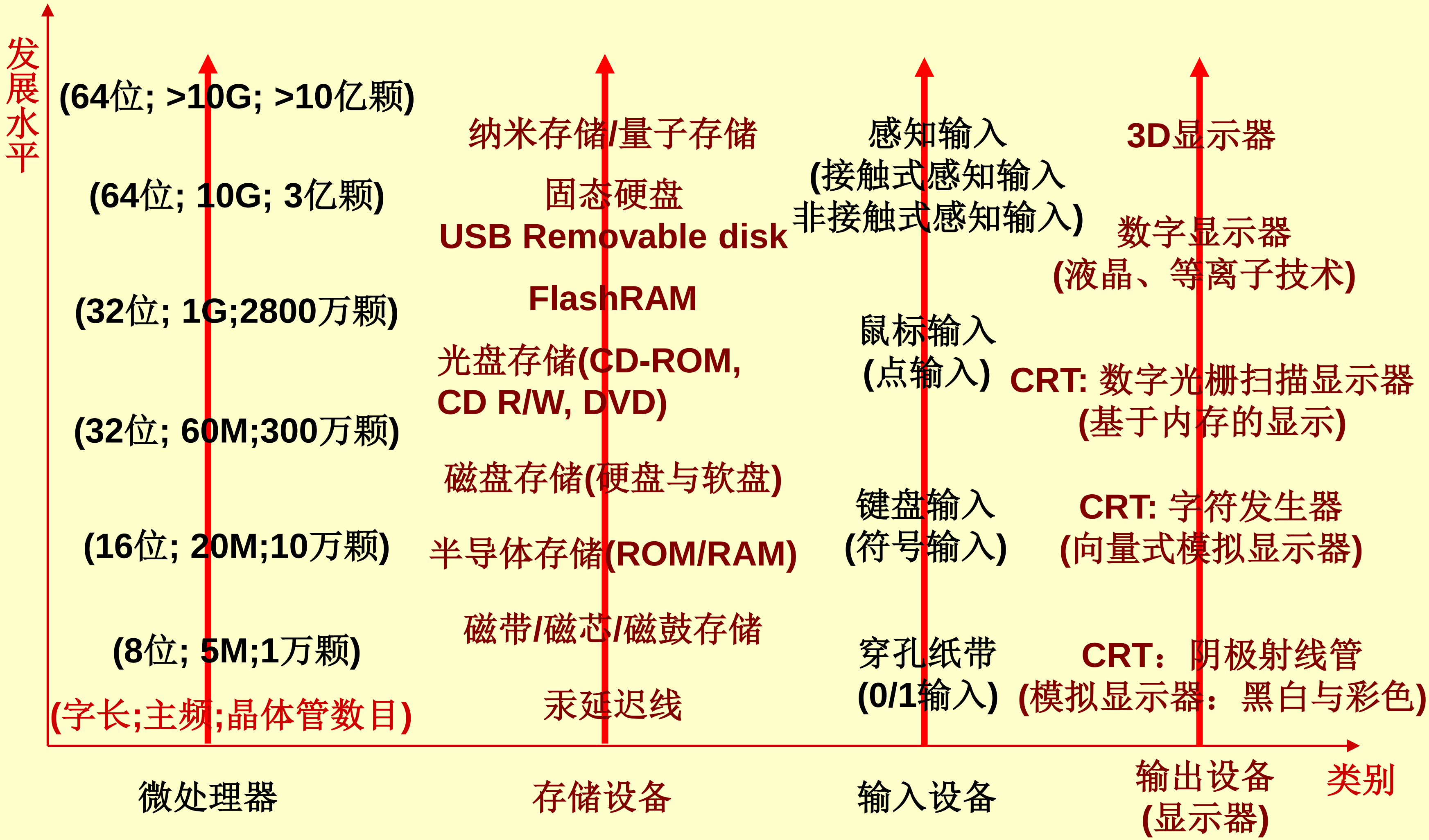
多显示卡并联

- 分辨率越来越高
- 颜色越来越逼真
- 显示速度越来越快(屏幕刷新速度和图形处理速度)
- 越来越薄，越清晰
- 可视角度越来越接近平角



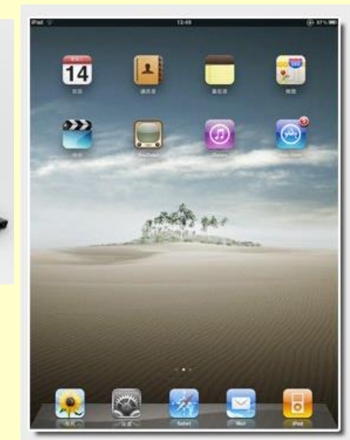
GPU芯片

计算机系统的发展

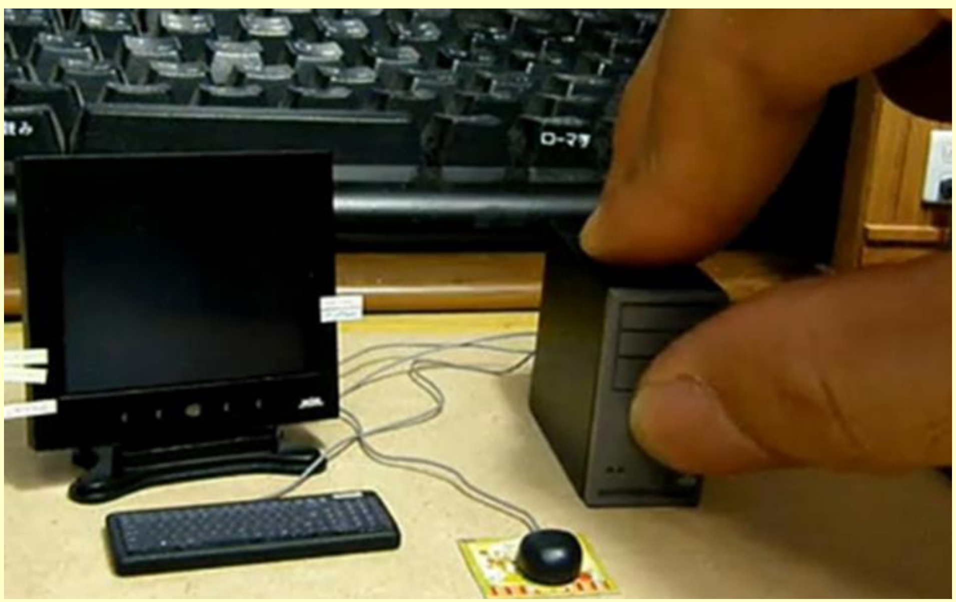

发展水平
(64位; >10G; >10亿颗)
(64位; 10G; 3亿颗)
(32位; 1G;2800万颗)
(32位; 60M;300万颗)
(16位; 20M;10万颗)
(8位; 5M;1万颗)
(字长;主频;晶体管数目)
纳米存储/量子存储
固态硬盘
USB Removable disk
FlashRAM
光盘存储(CD-ROM, CD R/W, DVD)
磁盘存储(硬盘与软盘)
半导体存储(ROM/RAM)
磁带/磁芯/磁鼓存储
汞延迟线
感知输入
(接触式感知输入
非接触式感知输入)
鼠标输入
(点输入)
键盘输入
(符号输入)
穿孔纸带
(0/1输入)
3D显示器
数字显示器
(液晶、等离子技术)
CRT: 数字光栅扫描显示器
(基于内存的显示)
CRT: 字符发生器
(向量式模拟显示器)
CRT：阴极射线管
(模拟显示器：黑白与彩色)
微处理器
存储设备
输入设备
输出设备
(显示器)
类别

电子自动计算-发展趋势

计算机的发展方向--微型化：可嵌入、可携带



平板电脑-Apple IPAD

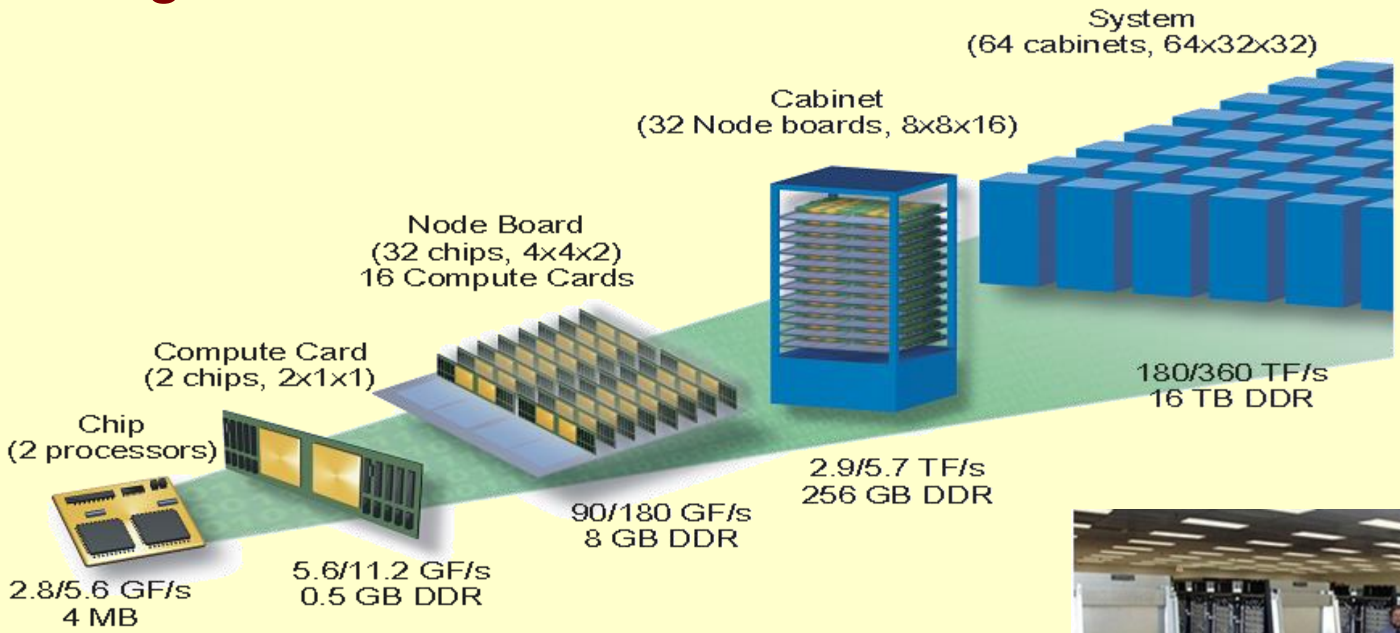


世界上最小台式电脑---- 如同拇指大小

智能手机

计算机的发展方向--大型化：可进行大规模、复杂计算

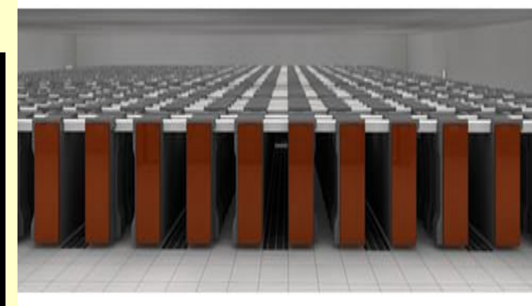
IBM-BlueGene(蓝色基因): Milestone of an Intelligent Machine



A massively parallel supercomputer using tens of thousands of embedded PowerPC processors supporting a large memory space With standard compilers and message passing environment

计算机的发展方向--大型化：可进行大规模、复杂计算

超级计算机500强



2010.11, 超级计算机500强第一名：天河一号A -- 中国

14336颗Intel Xeon X5670 2.93GHz六核心处理器

2048颗我国自主研发的飞腾FT-1000八核心处理器

7168块NVIDIA Tesla M2050高性能计算卡

总计: **186368**个核心, **224TB**内存。

实测运算速度可以达到**每秒2570万亿次**(这意味着，它计算一天，相当于一台家用电脑计算800年)

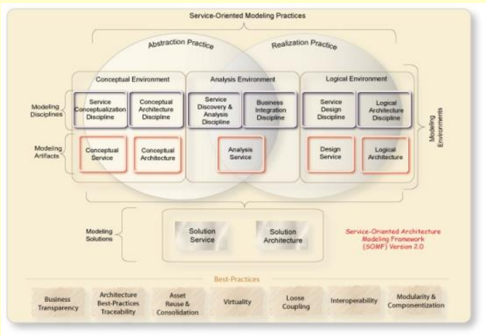
计算机的发展方向--智能化

理解自然语言，具有自适应性，自主完成复杂功能

计算机的发展方向--网络化

“未来互联网”-Future Internet

Internet of Services



For people

Internet of 3D Worlds

Internet of Things

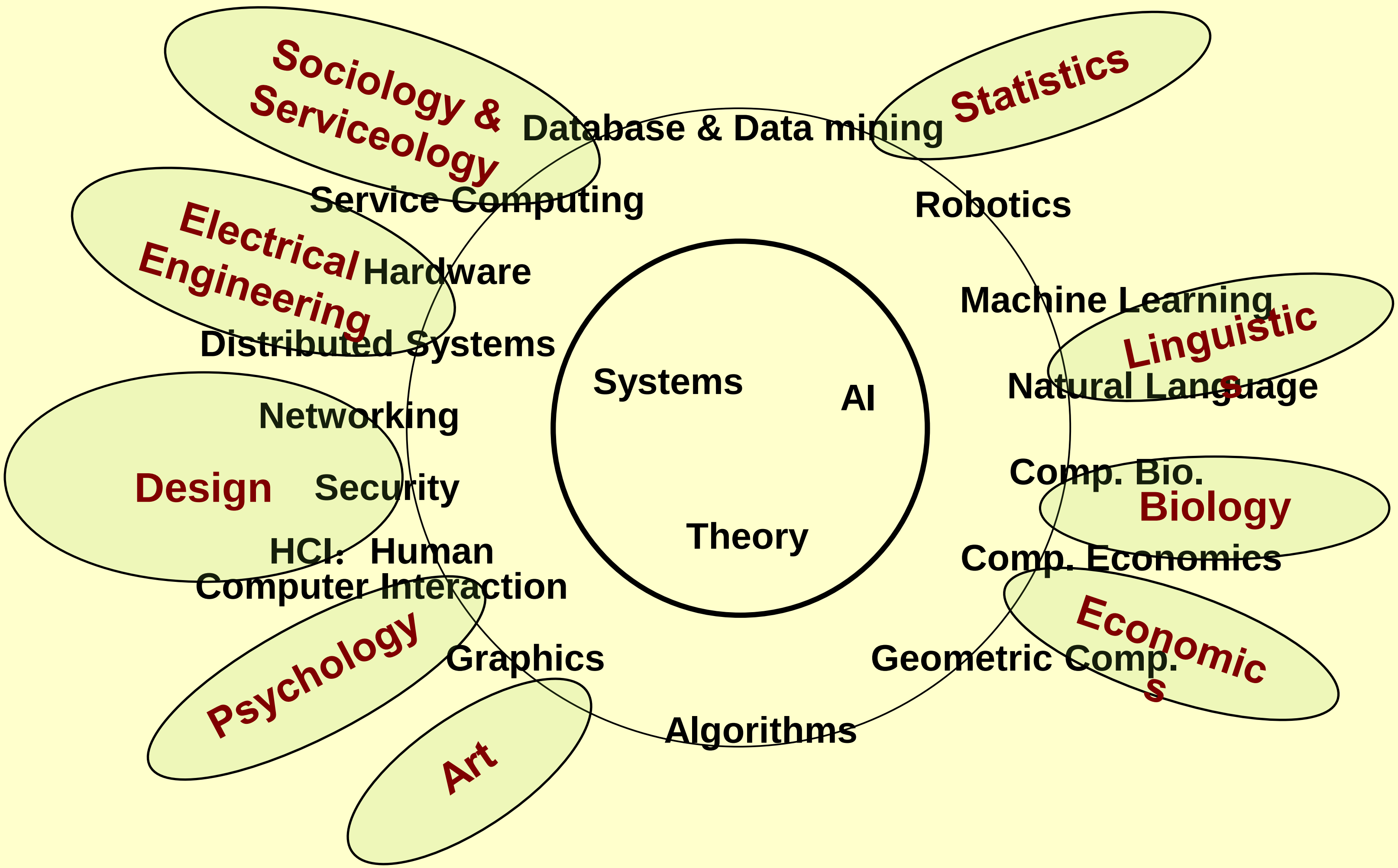
and enterprises

Internet of Networks

机-机相联，物-物相联，物-人相联，人-人相联

什么是计算思维?

学科的发展，知识的膨胀


Sociology &
Serviceology
Database & Data mining
Statistics
Service Computing
Robotics
Electrical
Engineering
Hardware
Machine Learning
Linguistic
s
Distributed Systems
Systems
AI
Natural Language
Networking
Design
Security
Comp. Bio.
Biology
Theory
HCI: Human
Computer Interaction
Comp. Economics
Economic
s
Psychology
Graphics
Geometric Comp.
Art
Algorithms

计算思维，计算的伟大原理

PHILOSOPHICAL TRANSACTIONS — OF — THE ROYAL SOCIETY A

Phil. Trans. R. Soc. A (2008) **366**, 3717–3725
doi:10.1098/rsta.2008.0118
Published online 31 July 2008

Computational thinking and thinking about computing

By Jeannette M. Wing*

Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Computational thinking will influence everyone in every field of endeavour. This vision poses a new educational challenge for our society, especially for our children. In thinking about computing, we need to be attuned to the three drivers of our field: science, technology and society. Accelerating technological advances and monumental societal demands force us to revisit the most basic scientific questions of computing.

Keywords: computational thinking; abstraction; automation; computing; computable; intelligence

1. Computational thinking

Computational thinking is taking an approach to solving problems, designing systems and understanding human behaviour that draws on concepts fundamental to computing[1] (Wing 2006).

Computational thinking is a kind of analytical thinking. It shares with mathematical thinking in the general ways in which we might approach solving a problem. It shares with engineering thinking in the general ways in which we might approach designing and evaluating a large, complex system that operates within the constraints of the real world. It shares with scientific thinking in the general ways in which we might approach understanding computability, intelligence, the mind and human behaviour.

(*a*) *Computing: abstraction and automation*

The essence of computational thinking is *abstraction*. In computing, we abstract notions beyond the physical dimensions of time and space. Our abstractions are extremely general because they are symbolic, where numeric abstractions are just a special case.

In two ways, our abstractions tend to be richer and more complex than those in the mathematical and physical sciences. First, our abstractions do not necessarily enjoy the clean, elegant or easily definable algebraic properties of mathematical

*wing@cs.cmu.edu

[1] By 'computing' I mean very broadly the field encompassing computer science, computer engineering, communications, information science and information technology.

One contribution of 19 to a Discussion Meeting Issue 'From computers to ubiquitous computing, by 2020'.

The Great Principles of Computing

Peter J. Denning

Computing is integral to science—not just as a tool for analyzing data, but as an agent of thought and discovery.

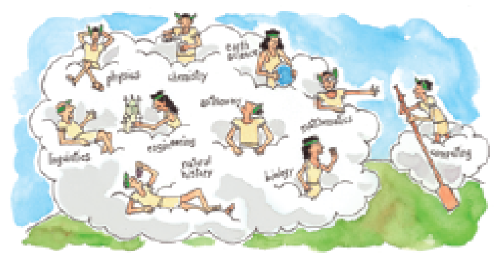
It has not always been this way. Computing is a relatively young discipline. It started as an academic field of study in the 1930s with a cluster of remarkable papers by Kurt Gödel, Alonzo Church, Emil Post and Alan Turing. The papers laid the mathematical foundations that would answer the question "what is computation?" and discussed schemes for its implementation. These men saw the importance of automatic computation and sought its precise mathematical foundation. The various schemes they each proposed for implementing computation were quickly found to be equivalent, as a computation in any one could be realized in any other. It is all the more remarkable that their models all led to the same conclusion that certain functions of practical interest—such as whether a computational algorithm (a method of evaluating a function) will ever come to completion instead of being stuck in an infinite loop—cannot be answered computationally.

At the time that these papers were written, the terms "computation" and "computers" were already in common use, but with different connotations from today. Computation was taken to mean the mechanical steps followed to evaluate mathematical functions; computers were people who did computations. In recognition of the social changes they were ushering in, the designers of the first digital computer projects all named their systems with acronyms ending in "-AC", meaning automatic computer—resulting in names such as ENIAC, UNIVAC and EDSAC.

At the start of World War II, the militaries of the United States and the United Kingdom became interested in applying computation to the calculation of ballistic and navigation tables and to the cracking of ciphers. They commissioned projects to design and build electronic digital computers. Only one of the projects was completed before the war was over. That was the top-secret project at Bletchley Park in England, which cracked the German Enigma cipher using methods designed by Alan Turing.

Many people involved in those projects went on to start computer companies in the early 1950s. Universities began offering programs of study in the new field in the late 1950s. The field and the industry have grown steadily into a modern behemoth whose Internet data centers are said to consume almost three percent of the world's electricity.

Computing may be the fourth great domain of science along with the physical, life and social sciences



During its youth, computing was an enigma to the established fields of science and engineering. At first, computing looked like only the applied technology of math, electrical engineering or science, depending on the observer. However, over the years, computing provided a seemingly unending stream of new insights, and it defied many early predictions by resisting absorption back into the fields of its roots. By 1980 computing had mastered algorithms, data structures, numerical methods, programming languages, operating systems, networks, databases, graphics, artificial intelligence and software engineering. Its great technological achievements—the chip, the personal computer and the Internet—brought it into many lives. These advances stimulated more new subfields, including network science, Web science, mobile computing, enterprise computing, cooperative work, cyberspace protection, user-interface design and information visualization. The resulting commercial applications have spawned

Peter J. Denning is Director of the Cebrowski Institute for Innovation and Information Superiority at the Naval Postgraduate School in Monterey, California, and is a past president of ACM. Email: pjd@nps.edu

◆《Computational Thinking》

from CMU，周以真（Jeannette M. Wing），Communications of ACM, Vol.49, No.3, March 2006, Pages 33-35

◆**Computational thinking is a way of solving problems, designing systems, and understanding human behavior that draws on concepts fundamental to computer science.**

◆**Computational thinking will be a fundamental skill used by everyone in the world by the middle of the 21st Century.**

– Just like reading, writing, and arithmetic.

– Imagine every person knowing how to think like a computer scientist!

– Computational thinking is not just for other scientists, it's for everyone.

–Thinking like a computer scientist means more than being able to program a computer

◆计算思维的本质就是**抽象(Abstraction)**与**自动化(Automation)**，即在**不同层面**进行抽象，以及将这些抽象“机器化”。

国内学者/专家的观点

■计算思维是人类应具备的第三种思维

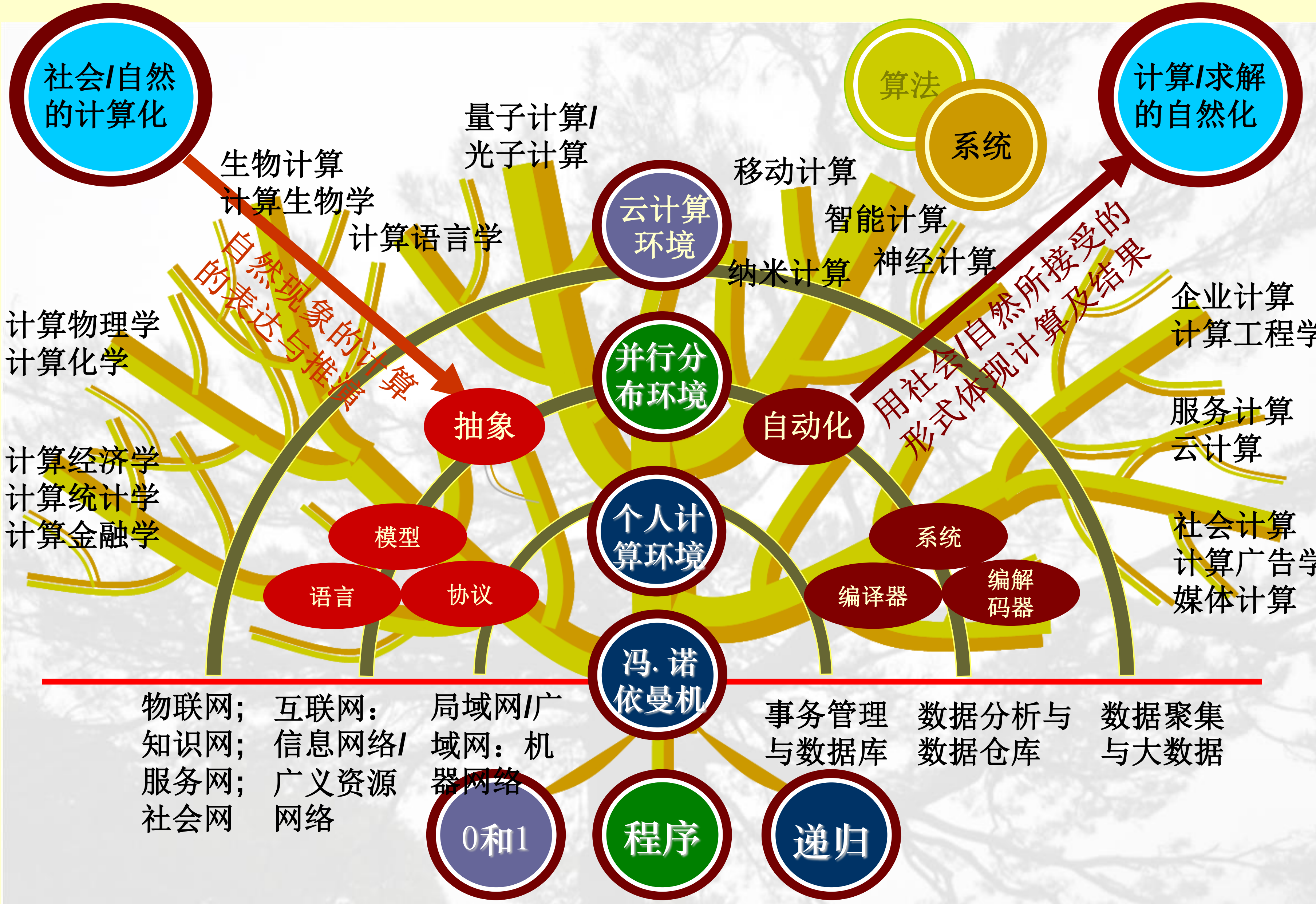
➢实验思维: 实验 ➔ 观察 ➔ 发现、推断与总结. ---观察与归纳

➢理论思维: 假设/预设 ➔ 定义/性质/定理 ➔ 证明. ---推理和演绎

➢计算思维: 设计, 构造 与 计算. ---设计与构造

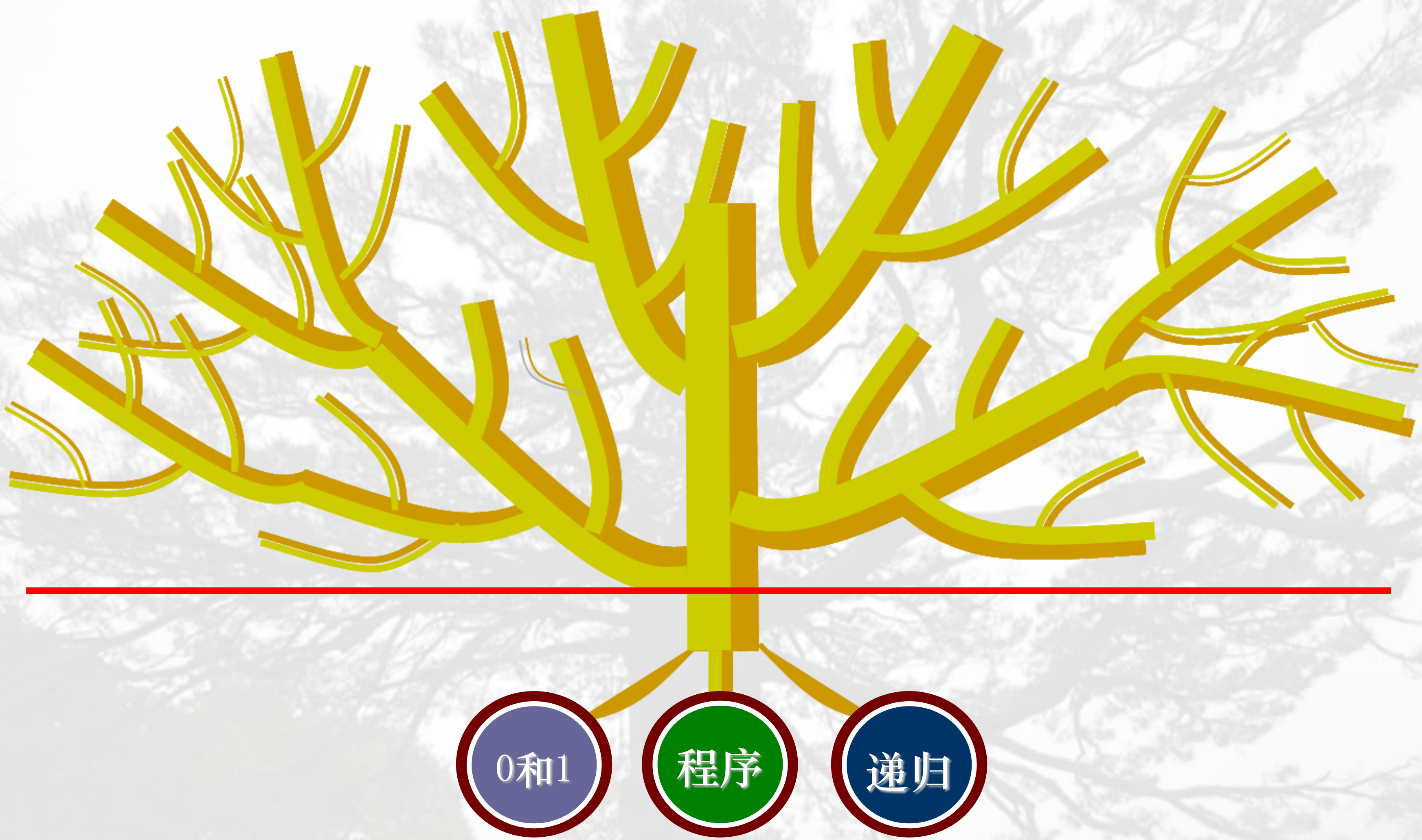
计算思维关注的是人类思维中有关可行性、可构造性和可评价性的部分

当前环境下，理论与实验手段在面临大规模数据的情况下，不可避免地要用计算手段来辅助进行。


社会/自然的计算化
计算/求解的自然化
算法
系统
量子计算/光子计算
生物计算
计算生物学
计算语言学
移动计算
智能计算
神经计算
纳米计算
云计算环境
并行分布环境
个人计算环境
冯. 诺依曼机
自然现象的计算的表达与推演
用社会/自然所接受的形式体现计算及结果
计算物理学
计算化学
计算经济学
计算统计学
计算金融学
企业计算
计算工程学
服务计算
云计算
社会计算
计算广告学
媒体计算
抽象
模型
语言
协议
自动化
系统
编译器
编解码器
物联网；知识网；服务网；社会网
互联网：信息网络/广义资源网络
局域网/广域网：机器网络
事务管理与数据库
数据分析与数据仓库
数据聚集与大数据
0和1
程序
递归

大学计算思维教育空间
—计算之树?

计算之树的第一个维度—计算技术的奠基性思维



计算之树的第一个维度—计算技术的奠基性思维

■“0 和 1”思维--符号化➔计算化➔自动化

➢0和1是实现任何计算的基础；社会/自然与计算融合的基本手段； 0和1是连接硬件与软件的纽带；0/1是最基本的抽象与自动化机制。

■“程序”思维--千变万化复杂功能的构造、表达与执行

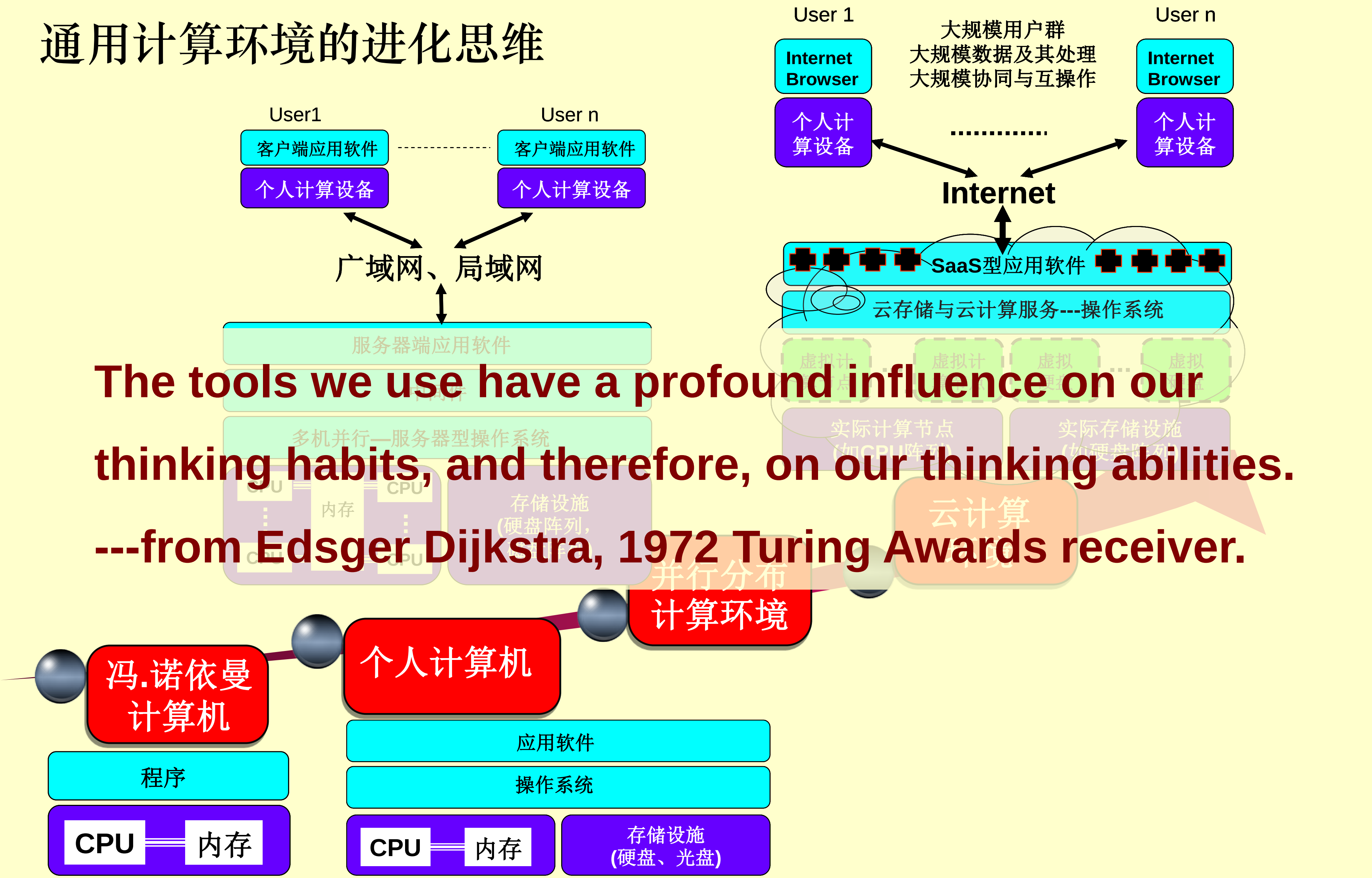
➢程序是基本动作(指令)的各种组合，是控制计算系统的基本手段

■“递归”思维--无限事物及重复过程的表达与执行方法

➢递归是最典型的构造程序的手段；递归函数是可计算函数的精确的数学描述；递归函数是研究计算学科理论问题的基础

计算之树的第二个维度—通用计算环境的进化思维

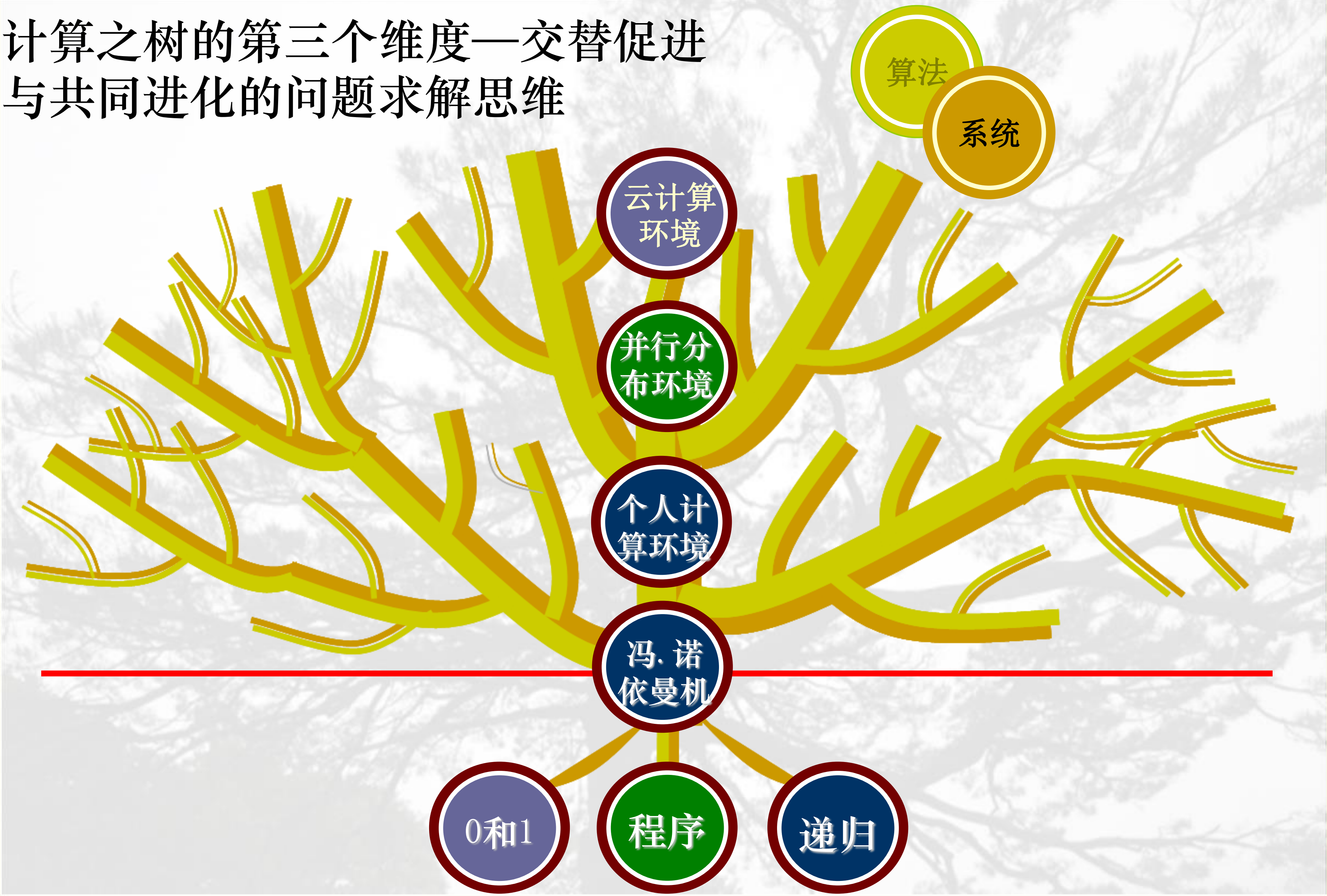
通用计算环境的进化思维



The tools we use have a profound influence on our thinking habits, and therefore, on our thinking abilities.

---from Edsger Dijkstra, 1972 Turing Awards receiver.

计算之树的第三个维度—交替促进与共同进化的问题求解思维



计算之树的第三个维度—交替促进与共同进化的问题求解思维

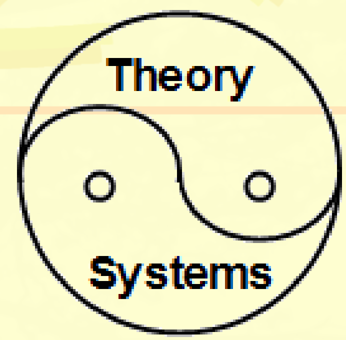
■“算法”：问题求解的一种手段—构造与设计算法

■算法是计算的灵魂；算法强调数学建模；算法考虑的是可计算性与计算复杂性；算法研究通常被认为是计算学科的理论研究。

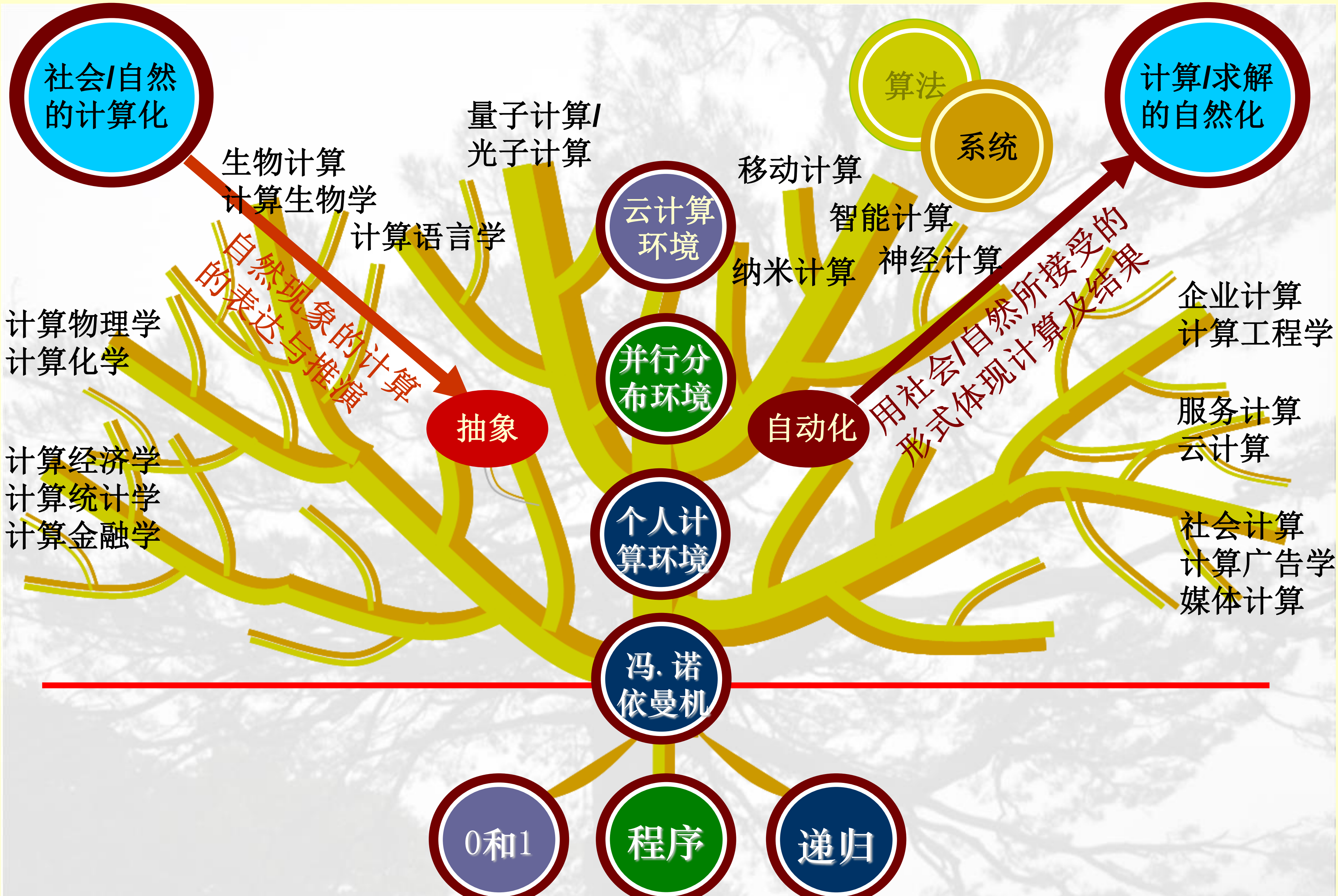
■“系统”：问题求解的另一种手段—构造与设计系统

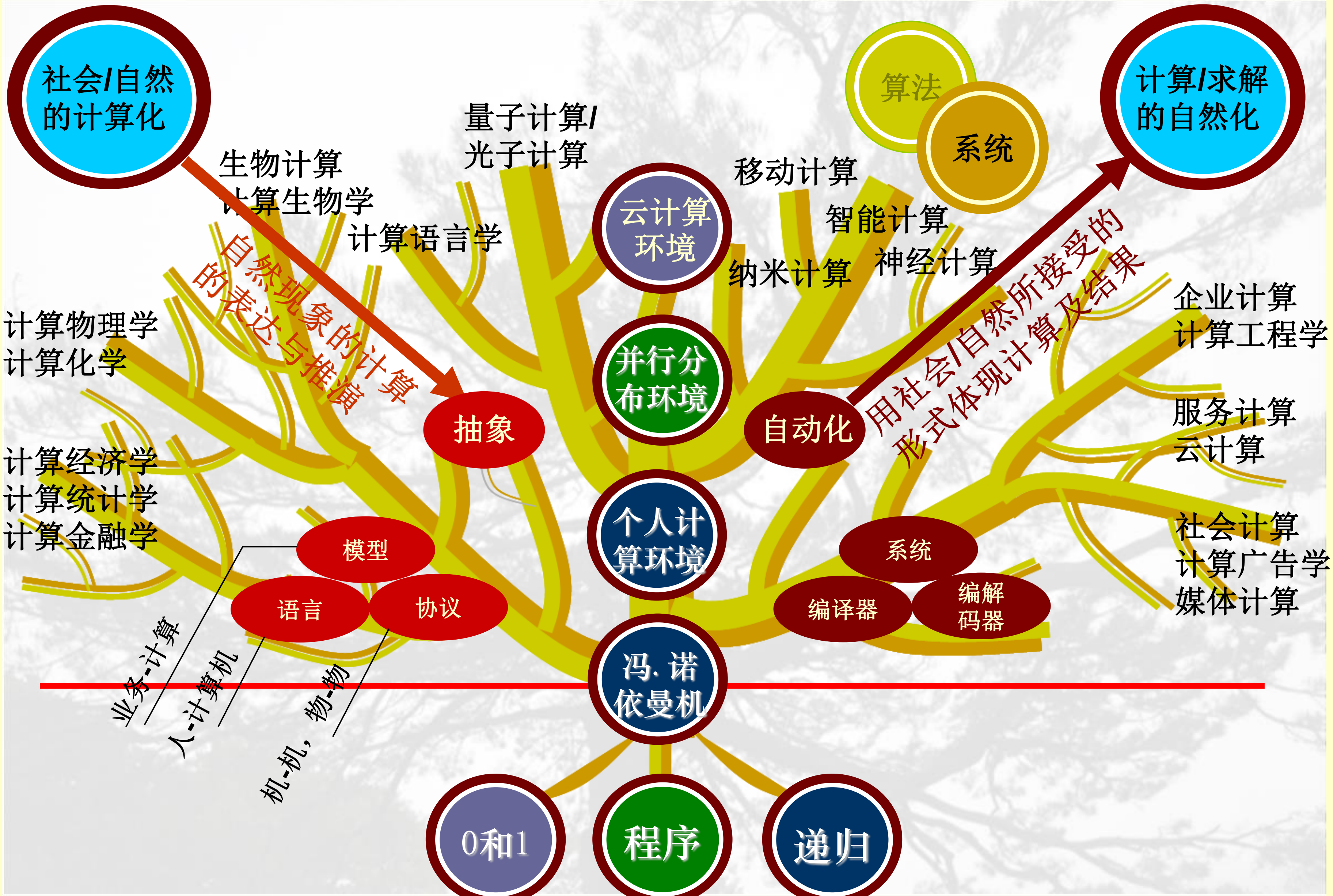
➢系统是改造自然的手段；系统还强调非数学建模；系统考虑的是如何化复杂为简单(使其能够被做出来)；系统还强调结构性、可靠性、安全性等。

系统是龙，算法是睛，画龙要点睛。

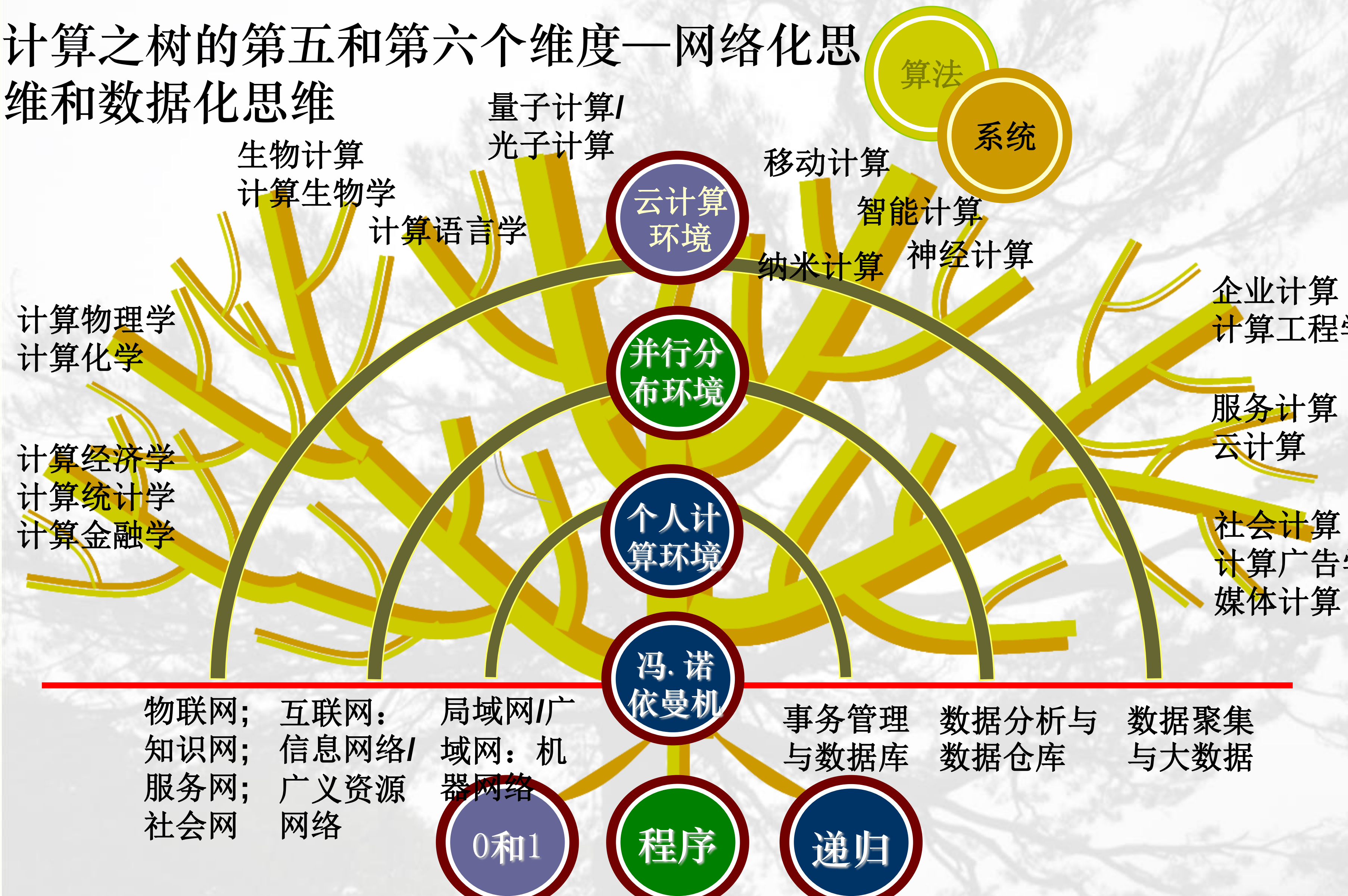


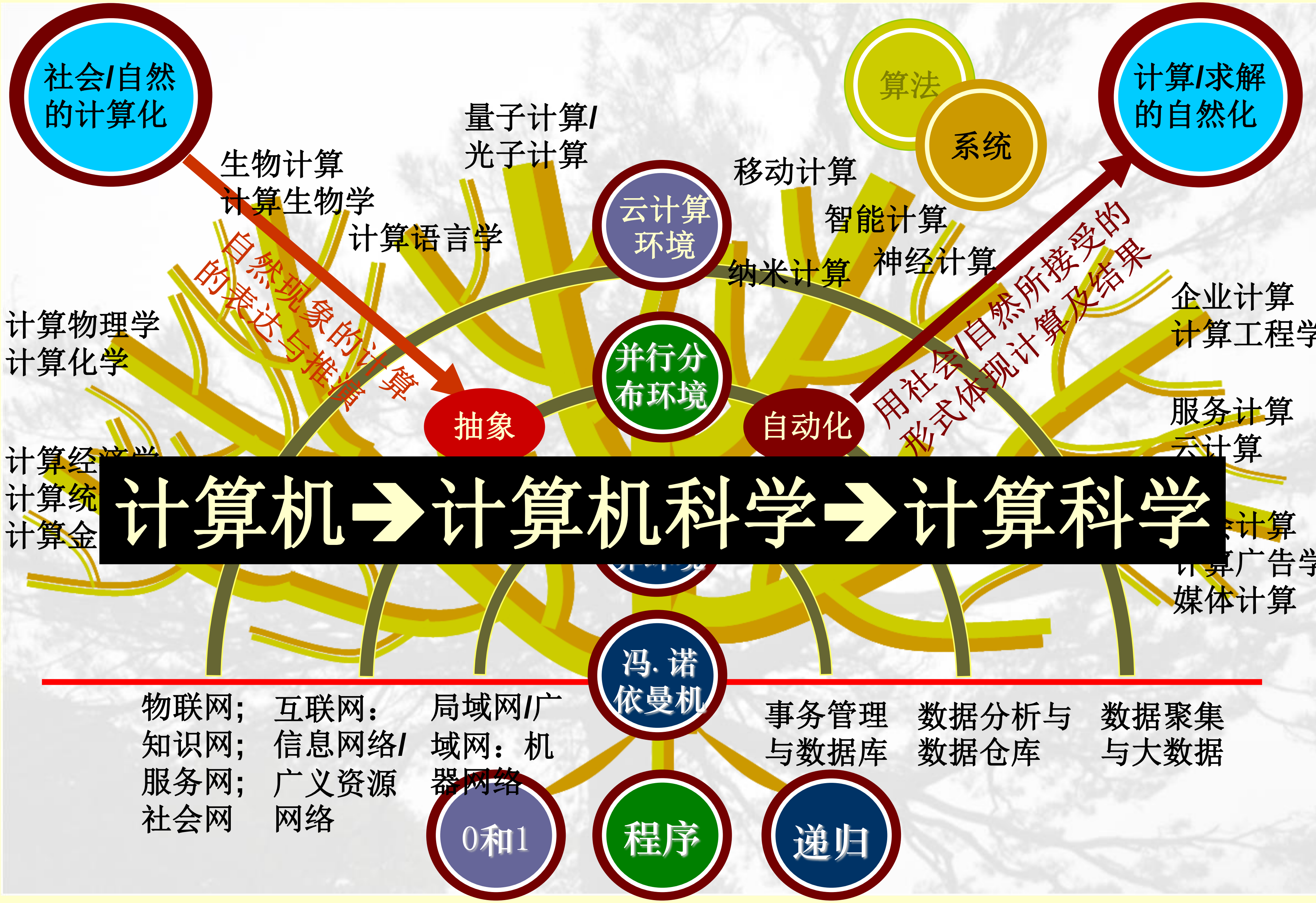
计算之树的第四个维度—计算与社会/自然环境的融合思维


社会/自然
的计算化
计算/求解
的自然化
算法
系统
量子计算/
光子计算
生物计算
计算生物学
计算语言学
移动计算
智能计算
神经计算
纳米计算
云计算
环境
并行分
布环境
个人计
算环境
冯. 诺
依曼机
计算物理学
计算化学
计算经济学
计算统计学
计算金融学
自然现象的计算
的表达与推演
抽象
自动化
用社会/自然所接受的
形式体现计算及结果
企业计算
计算工程学
服务计算
云计算
社会计算
计算广告学
媒体计算
0和1
程序
递归


社会/自然的计算化
计算/求解的自然化
算法
系统
量子计算/光子计算
生物计算
计算生物学
计算语言学
移动计算
智能计算
纳米计算
神经计算
云计算环境
并行分布环境
个人计算环境
冯. 诺依曼机
计算物理学
计算化学
计算经济学
计算统计学
计算金融学
企业计算
计算工程学
服务计算
云计算
社会计算
计算广告学
媒体计算
自然现象的计算的表达与推演
用社会/自然所接受的形式体现计算及结果
抽象
自动化
模型
语言
协议
系统
编译器
编解码器
业务-计算
人-计算机
机-机，物-物
0和1
程序
递归

计算之树的第五和第六个维度—网络化思维和数据化思维




社会/自然的计算化
量子计算/光子计算
生物计算
计算生物学
计算语言学
云计算环境
移动计算
智能计算
纳米计算
神经计算
算法
系统
计算/求解的自然化
自然现象的计算的表达与推演
计算物理学
计算化学
并行分布环境
抽象
自动化
用社会/自然所接受的形式体现计算及结果
企业计算
计算工程学
服务计算
云计算
计算机→计算机科学→计算科学
媒体计算
冯.诺依曼机
物联网；知识网；服务网；社会网
互联网：信息网络/广义资源网络
局域网/广域网：机器网络
事务管理与数据库
数据分析与数据仓库
数据聚集与大数据
0和1
程序
递归

◆计算思维的学习方法

（1）“知识/术语”随着“思维”的学习而展开，“思维”随着“知识”的贯通而形成，“能力”随着“思维”的理解而提高。

（2）从问题分析着手，强化如何进行抽象，如何将现实问题抽象为一个数学问题或者一个形式化问题，提高问题表述及问题求解的严谨性。

（3）通过图示化方法来展现复杂的思维可以一目了然；通过规模较小的问题求解示例来理解复杂问题的求解方法；通过从社会/自然等人们身边的问题理解到计算科学家是如何进行问题求解。

（4）追求“问题”及问题的讨论，通过逐步地提出问题，使自己从一个较浅的理解层次逐步过渡到较深入的理解层次，通过不同视角和递阶的讨论，使自己理解和确定前行的方向。

（5）宽度与深度相结合，从宽度学习开始，深度学习结束，既能够使自己理解相关的思维与知识，还能够有助于建立起较为科学的研究习惯与研究方法。

（6）思维蕴含在案例中，案例蕴含着思维。

阅读书籍、阅读文献、网上搜索、梳理思路(记笔记)

第1讲 计算机、计算与计算思维

Questions & Discussion?

2015