

软件之启蒙 - 初识软件工程专业

软件工 程 的 发 展 及 趋 势

徐晓飞 教授、博导



视频位置

软件危机与软件工程的产生

◆ 软件危机

- 1960年代后期，随着软件规模及开发难度的增加，软件开发周期长、成本高、质量差、维护难，导致软件危机爆发
- **问题：**对软件开发工作量和成本估计不准；软件开发进度难以控制；软件产品质量与可靠性差强人意。

◆ 软件工程的产生

1968年10月，北大西洋公约组织NATO召开计算机科学会议，Fritz Bauer首次提出“软件工程”概念及克服“软件危机”的策略，强调按照工程化原则和方法组织软件开发工作。软件工程技术领域由此应运而生。

软件工程发展历史

1946-1956：程序设计时代；个体手工制作，采用机器语言/汇编语言编程，主要依靠个人编程技巧。

1956-1968：程序系统时代；作坊式小团队合作，采用高级语言编程，以个人编程技巧为主，开始有结构化方法。

1968以来，软件工程时代；工程化生产方式，采用软件工程方法，并逐步发展形成了结构化方法、面向数据结构方法、面向对象的方法、构件化方法、面向服务的方法等。

软件工程发展历史

◆ 软件工程方法发展历程

- 1960's-1970's: 结构化方法

- 1980's: 面向对象的方法

- 1990's: 构件化方法和Web Services

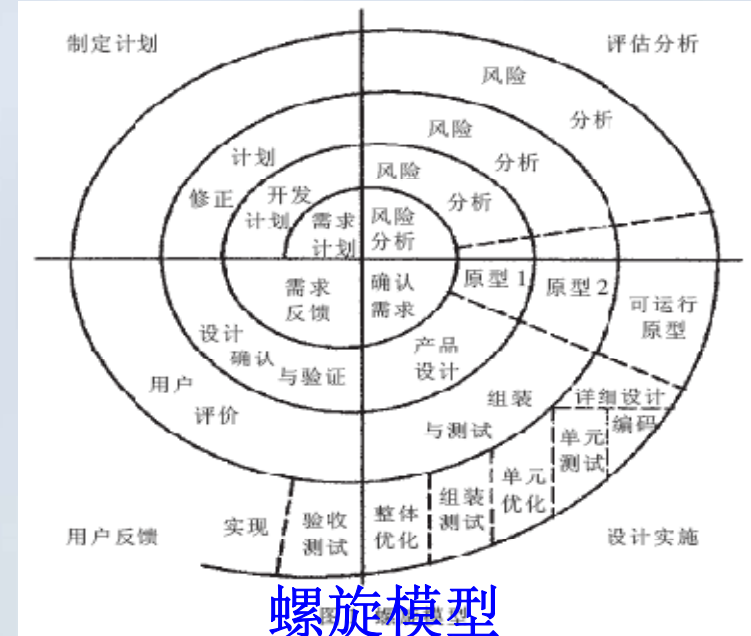
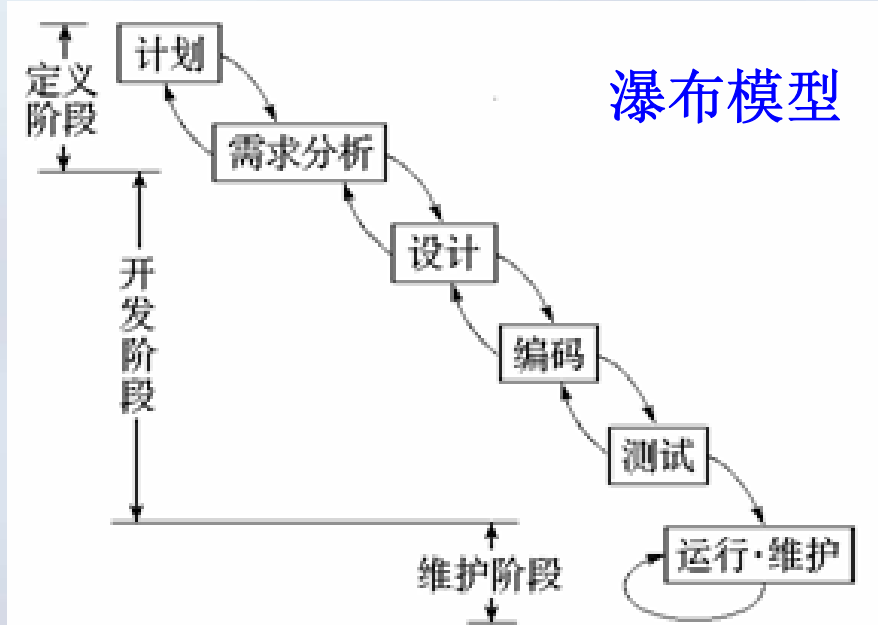
- 2000's: 面向服务的SOA方法

- 2010's: 基于互联网与云计算的软件开发方法

软件工程方法的演变：结构化方法

● 1960's-1970's: 结构化方法

- ◆ 方法：结构化程序设计方法、瀑布模型、螺旋模型等
- ◆ 编程语言：Fortran语言、Pascal语言、C语言
- ◆ 结构化方法好比建平房或用平房建设技术建造建筑物



软件工程方法的演变：结构化方法

● 1960's-1970's: 结构化方法

- ◆ 方法：结构化程序设计方法、瀑布模型、螺旋模型等
- ◆ 编程语言：Fortran语言、Pascal语言、C语言
- ◆ 结构化方法好比建平房或用建平房的技术建造复杂建筑

```
#include <stdio.h>
#define N 4
main() {
    //printf("Hello! " + CITY_NQ);
    int D[N][N];
    int S[N];
    int Sum;
    int i;
    int j;
    int K;
    int L;
    int Dtemp;
    int Found;

    S[0]=0;
    Sum=0;
    D[0][1]=2;
    D[0][2]=6;
    D[0][3]=5;
    D[1][0]=2;
    D[1][2]=4;
    D[1][3]=4;
    D[2][0]=6;
    D[2][1]=4;
    D[2][3]=2;
    D[3][0]=5;
    D[3][1]=4;
    D[3][2]=2;

    i=1;
    do {
        K=1;
        Dtemp=10000;
        do {
            L=0;
            Found=0;
            do {
                if(S[L]==K) {
                    Found=1;
                    break;
                }
                else L++;
            } while(L<1);
            if(Found==0 && D[K][L-1]<Dtemp) {
                j=K;
                Dtemp=D[K][L-1];
            }
            while(K<N) {
                S[i]=j;
                Sum=Sum+D[i][j];
                Sum=Sum+D[j][i];
                for(j=0;j<N;j++) {
                    printf("%d\t",S[i]);
                }
                printf("\n");
                i++;
            }
        } while(K<N);
    } while(i<N);
    printf("Total Length:%d",Sum);
}
```

主函数()

数据结构与数据

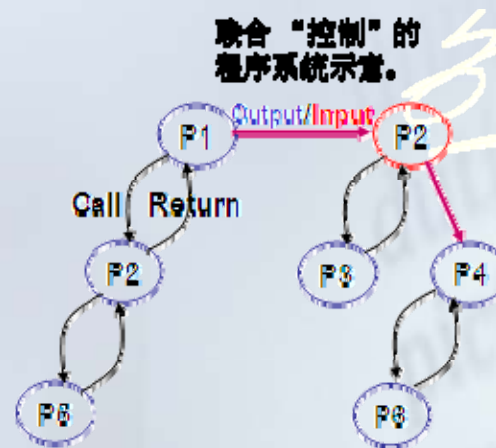
算法 (的实现)

函数()

函数()

系统 = 算法 + 数据结构 (1960's)

系统 = 子程序/函数 + 函数调用 (1980's)



```
2009-11-12 16:53 <DIR> ..
2009-11-10 00:41 <DIR> Administrator
2011-02-07 22:20 <DIR> All Users
2011-06-13 10:10 <DIR> andin
0 个文件
5 个目录 18,356,539,392 可用字节

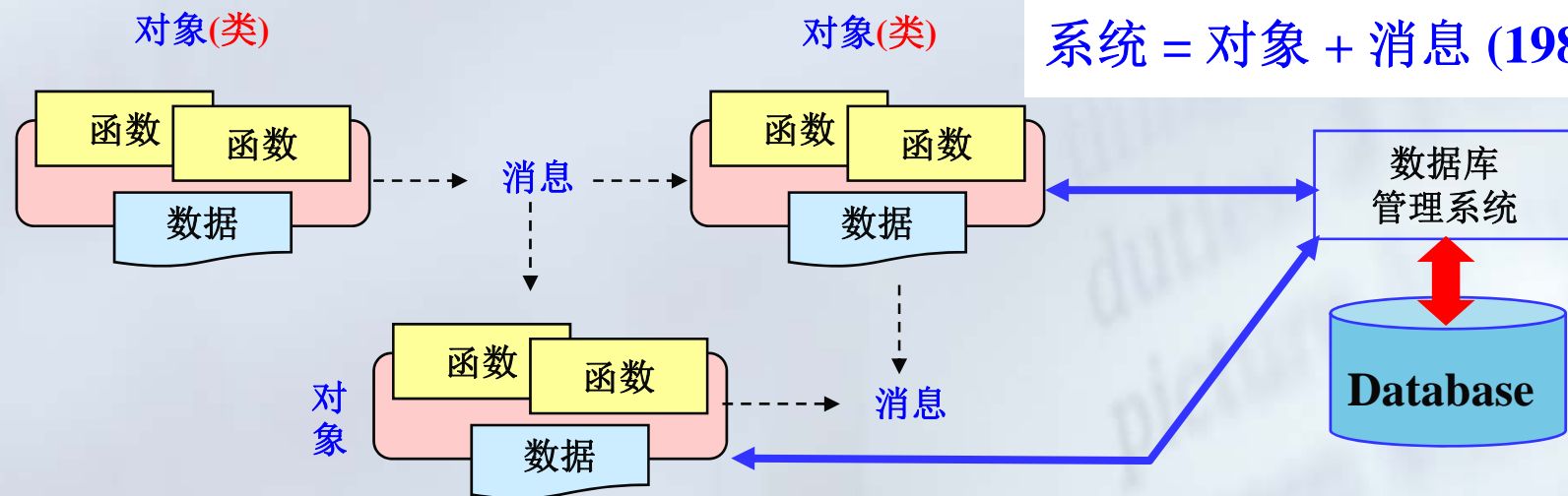
C:\Documents and Settings\andin\dir \
驱动器 C 中的卷是 Preload
卷的序列号是 ECAD-25EC

C:\ 的目录
2011-09-07 15:58 1,024 .red
2011-05-06 18:04 3688ec
2011-05-17 13:11 <DIR> S6Download
2010-03-22 23:04 44,364 abc.log
2009-11-10 00:24 <DIR> AuthLog
2006-05-15 21:21 0 AUTOEXEC.BAT
2011-03-20 07:05 532,712 bar.mf
2009-11-07 23:47 <DIR> CHPMNIS
2006-05-15 21:21 0 CONFIG.SYS
2009-11-12 16:53 <DIR> Documents and Settings
2011-08-21 07:12 <DIR> Downloads
2011-02-10 00:31 <DIR> drivers
2010-04-28 15:48 <DIR> eclipseCEP
2009-11-12 16:54 <DIR> 1386
2010-07-28 07:32 <DIR> icons
2009-11-10 00:00 <DIR> Intel
```

软件工程方法的演变：面向对象方法

● 1980's: 面向对象的方法

- ◆ 方法：面向对象方法、面向对象模型及建模工具等
- ◆ 编程语言：C++ (83)、Java (95)、Visual 系列语言 (90) 等
- ◆ 面向对象方法好比建高楼，可以更方便地构建复杂建筑



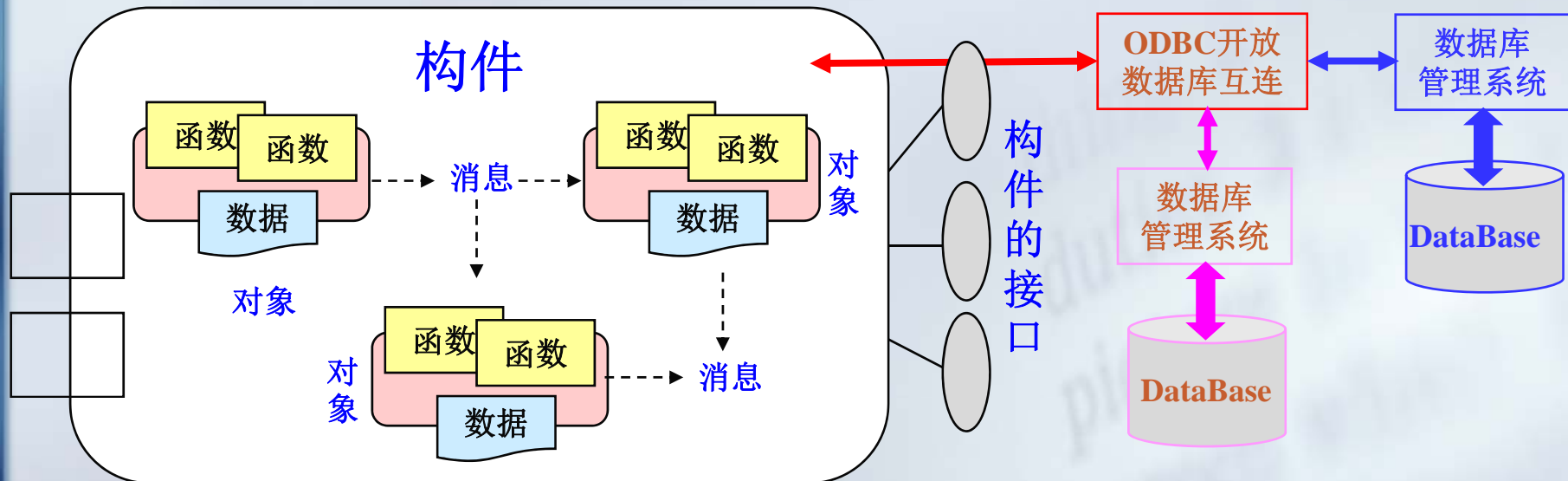
对象 = 函数 + 数据

系统 = 对象 + 消息 (1980's)

软件工程方法的演变：构件化方法

● 1990's: 构件化方法

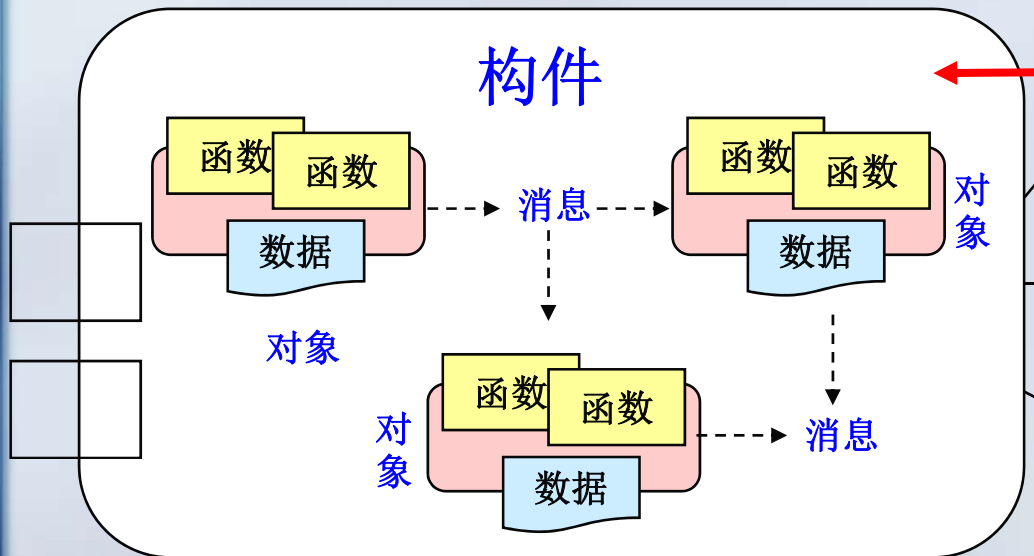
- ◆ 方法：软构件方法、Web Services、软件复用方法等
- ◆ 编程语言：Visual系列语言、Windows操作系统等
- ◆ 构件化方法好比堆积木、造预制件等，可以批量地、快速地构建更为复杂的建筑。



软件工程方法的演变：构件化方法

● 1990's: 构件化方法

- ◆ **方法**：软构件方法、Web Services、软件复用方法等
- ◆ **编程语言**：Visual 系列语言、Windows 操作系统等
- ◆ **构件化方法**好比堆积木
构件 = 对象 + 消息 or 构件 = 实体 + 接口
地构建更为复杂的建筑系统 = 构件 + 连接件 (1990's)

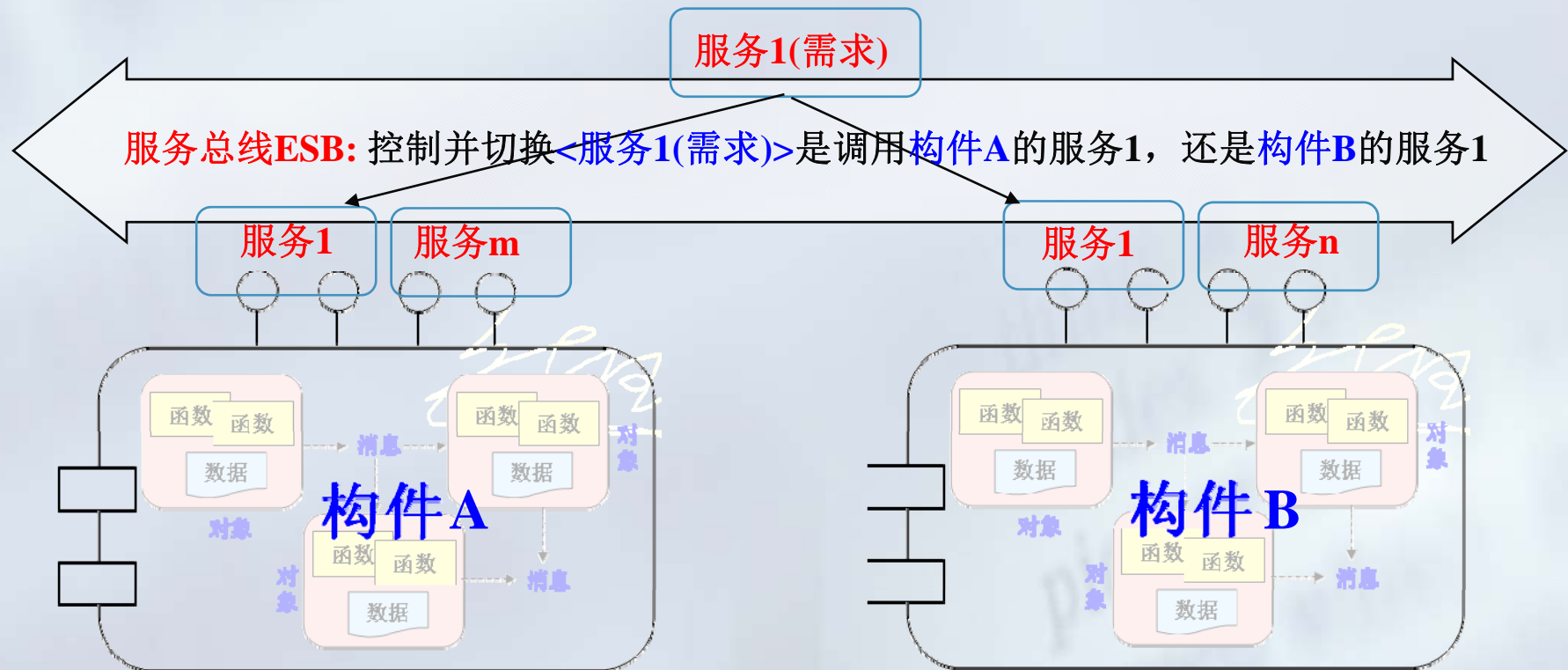


软件工程方法的演变：面向服务的方法

● 2000's: 面向服务的方法

◆ 面向服务的体系结构SOA方法 →

基于互联网与云计算的软件开发方法

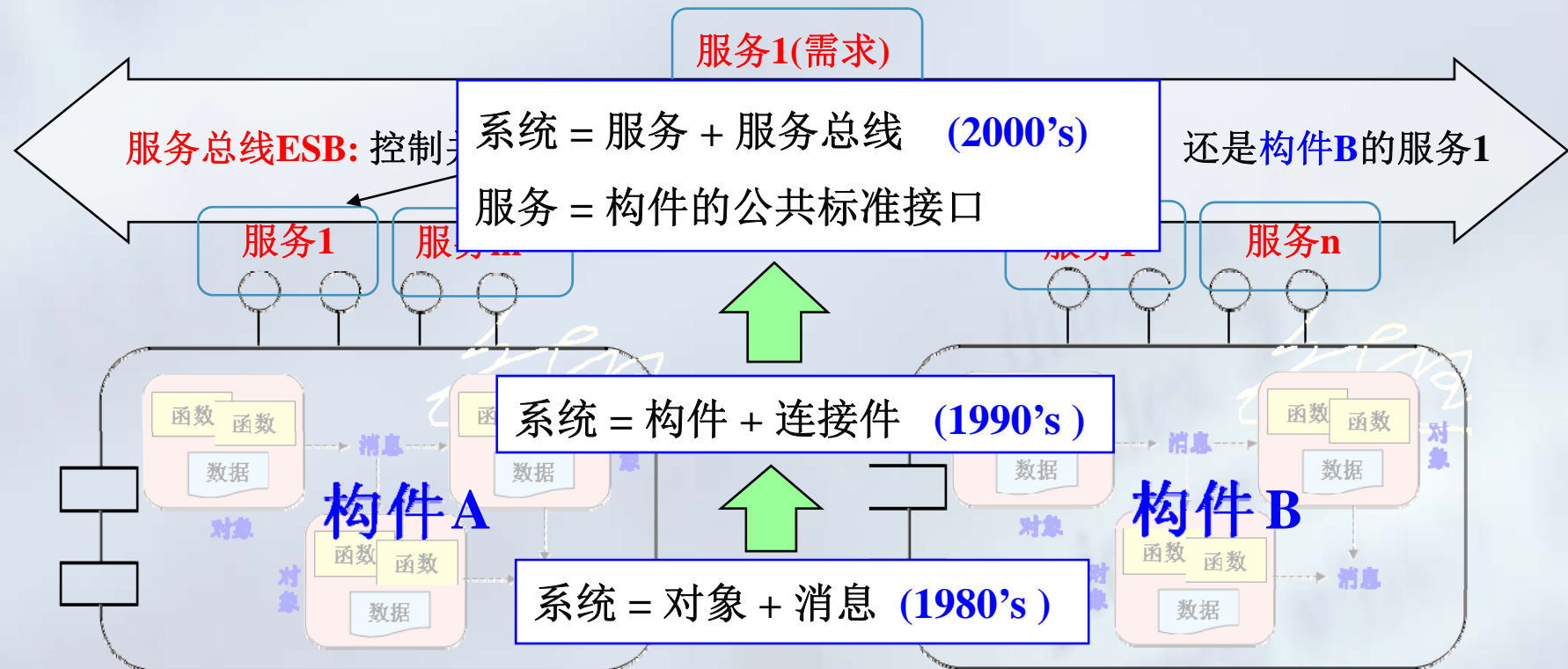


软件工程方法的演变：面向服务的方法

● 2000's: 面向服务的方法

◆ 面向服务的体系结构SOA方法 →

基于互联网与云计算的软件开发方法

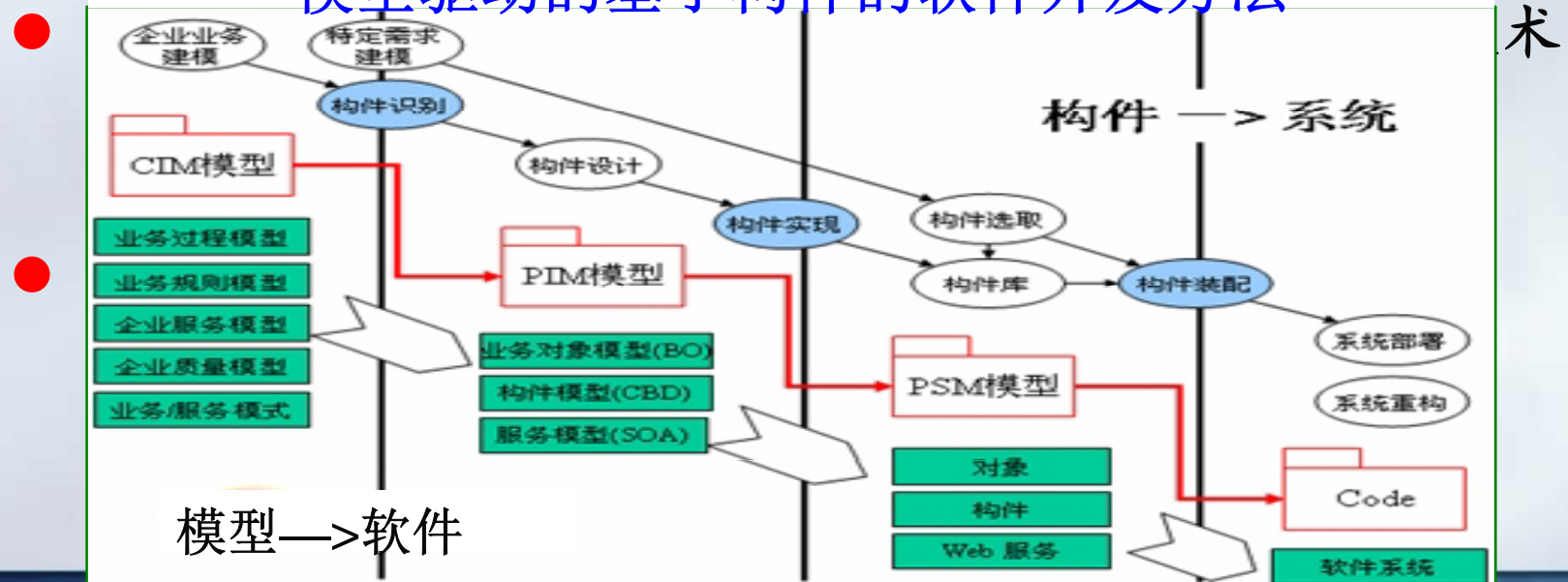


软件工程技术发展趋势

◆ 软件工程技术主要发展趋势

- 新型软件体系结构及开发方法 —— 基于云计算平台的软件体系结构、模型驱动的开发方法MDA、敏捷软件开发方法、软件集成开发环境及工具

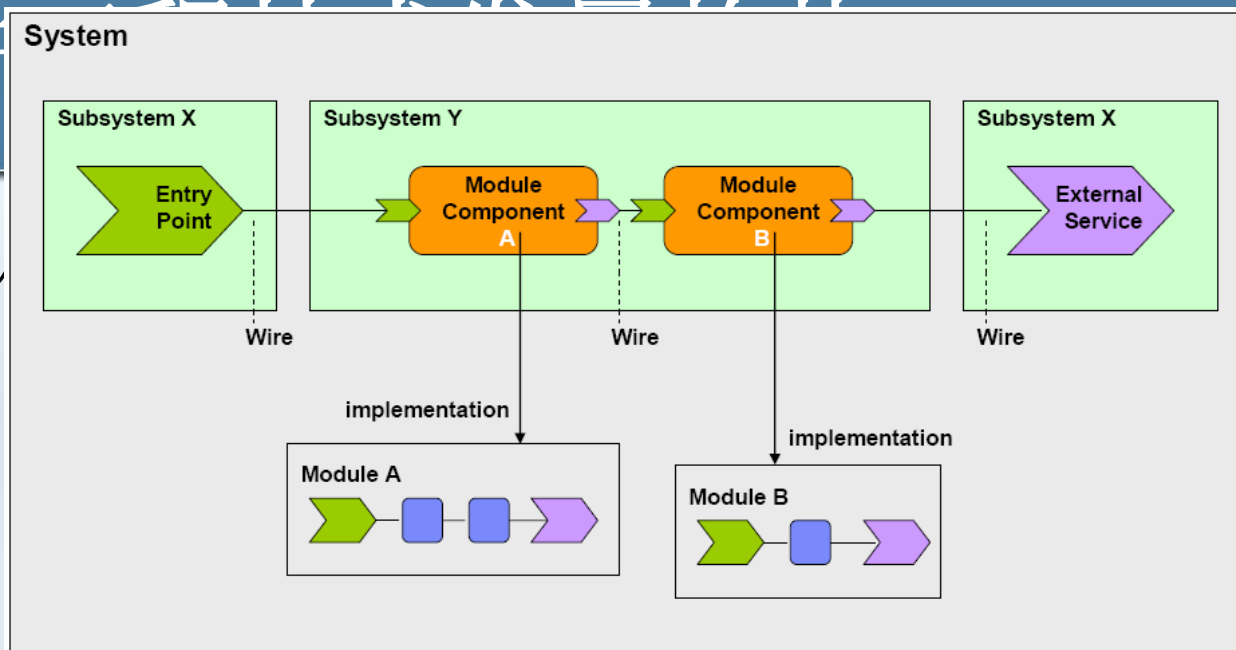
模型驱动的基于构件的软件开发方法



软件



软

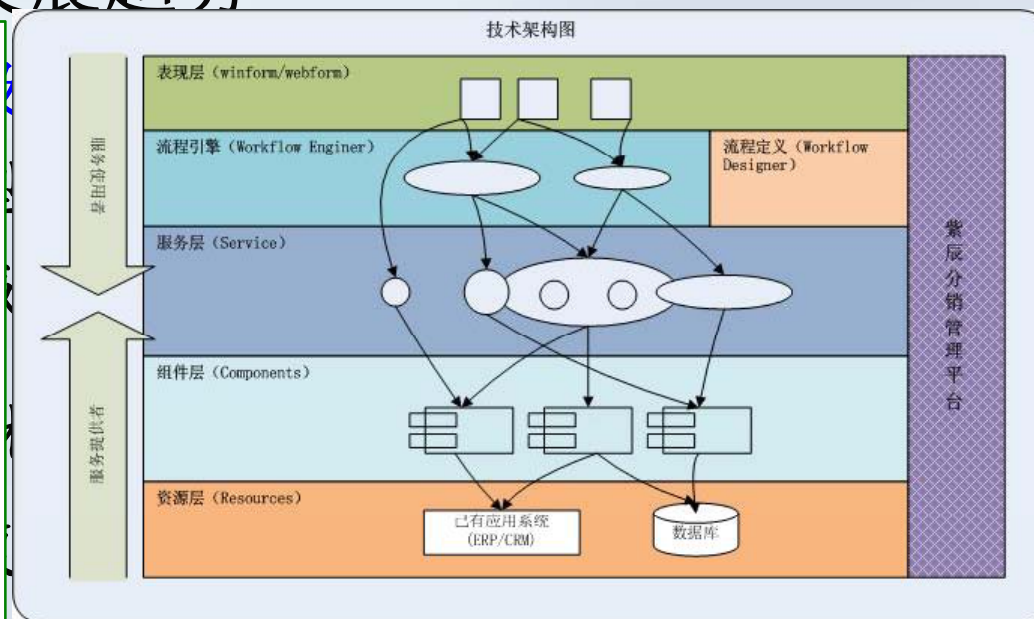


算平台的
敏捷软件

- 软件构件化 —— 软构件（Software Component）技术、基于构件的软件复用（Software Reuse）
- 软件服务化 —— 面向服务的体系结构SOA、Web Services、软件即服务SaaS、软件服务工程

软件工程技术发展趋势

◆ 软件工程技术主要发展趋势

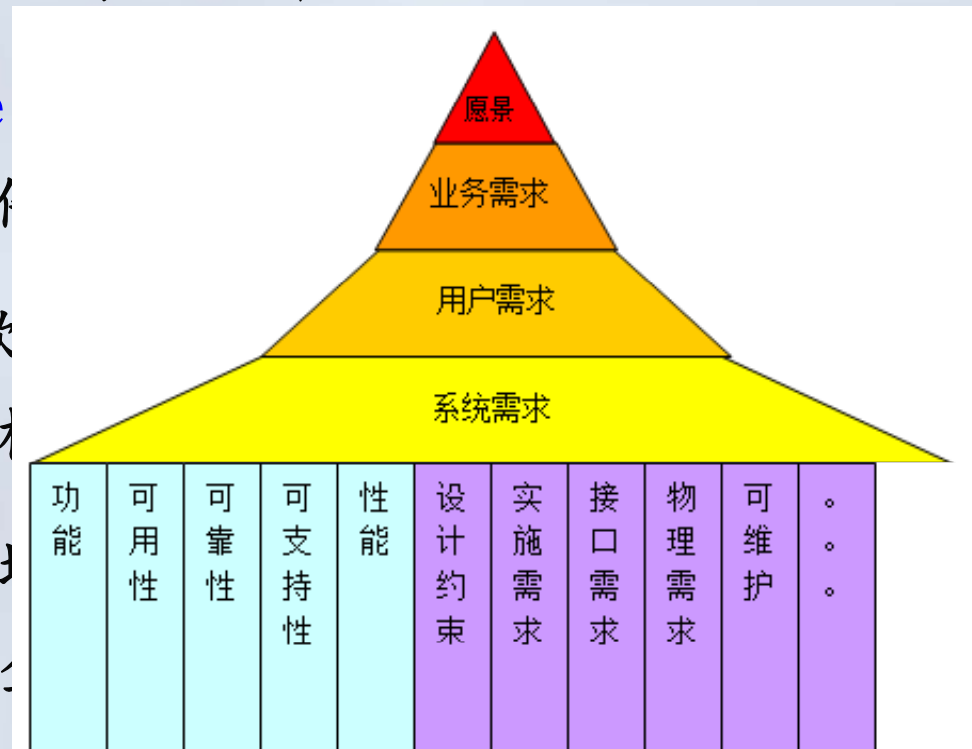


- 软件服务化 —— 面向服务的体系结构SOA、Web Services、软件即服务SaaS、软件服务工程

软件工程技术发展趋势

◆ 软件工程技术主要发展趋势

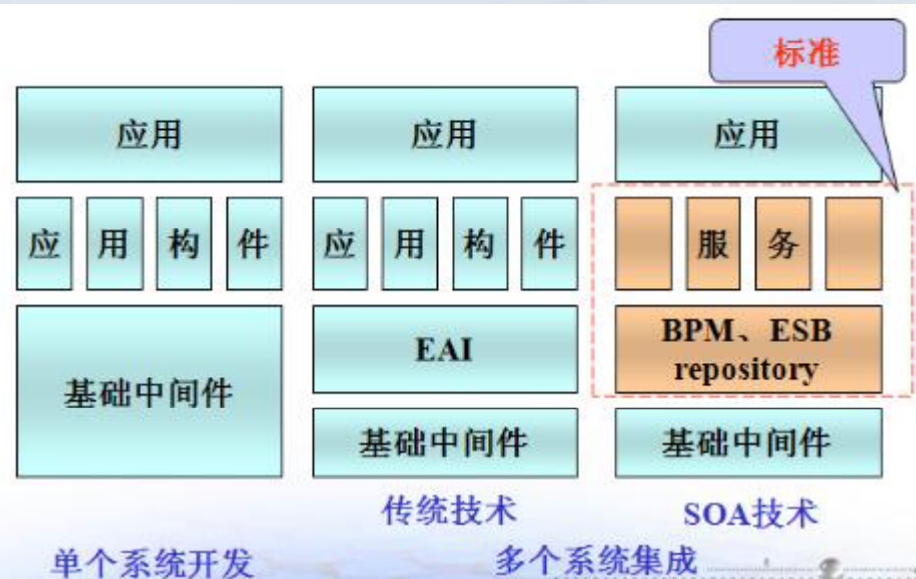
- 软件需求工程（**Requirement Engineering**）—— 基于知识的软件需求分析、需求分析自动化
- 中间件（**Middleware**）—— 总线中间件、消息总线ESB、网络构件
- 软件质量保障 —— 软件测试技术、软件过程改进技术
- 软件领域化 —— 领域特定语言、行业应用软件、企业级应用



软件工程技术发展趋势

◆ 软件工程技术主要发展趋势

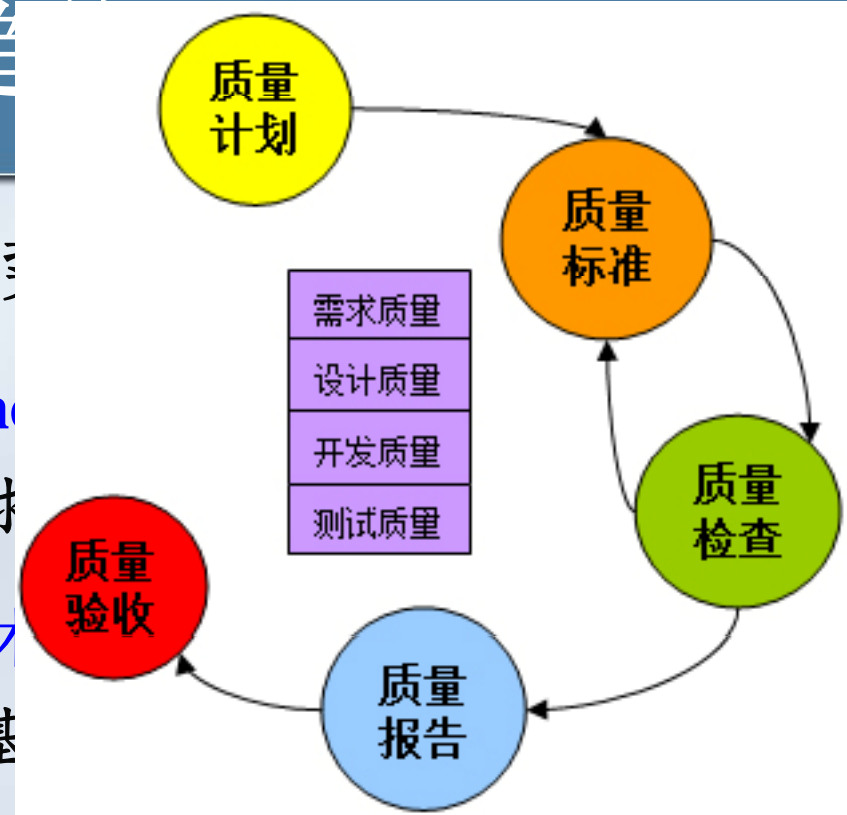
- 软件需求工程（**Requirement Engineering**）—— 基于知识的软件需求分析、需求分析自动化
- 中间件（**Middleware**）技术—— 中间件平台、企业服务总线ESB、网络构件、基于中间件的软件集成技术
- 软件质量保障 —— 软件质量保障技术、软件过程改进技术
- 软件领域化 —— 领域应用软件开发、行业应用软件、企业应用软件开发



软件工程技术发展趋势

◆ 软件工程技术主要发展趋势

- 软件需求工程（**Requirements Engineering**）—— 基于形式化知识的软件需求分析、需求管理
- 中间件（**Middleware**）技术—— 面向消息的中间件、企业服务总线ESB、网络构件、基础中间件
- 软件质量保障 —— 软件质量评测与度量、软件可靠性技术、软件过程改进模型
- 软件领域化 —— 领域软件工程（**Domain Engineering**）、行业应用软件、企业应用软件



软件工程技术发展趋势

◆ 软件工程技术主要发展趋势



—— 基于

企业服
成技术

件可靠性

● 软件领域化 —— 领域软件工程（Domain Engineering）、行业应用软件、企业应用软件

软件产业的发展历程

阶段1: 独立的程序(Independent Programming Service)

- 1950s-1960s: 机器为中心, 专业服务公司, 如IBM

阶段2: 软件产品(Software Product)

- 1960s-1970s: 软件业独立发展, 软件产品公司, 如ADR

阶段3: 企业解决方案(Enterprise Solution)

- 1970s-1980s: 应用为中心, 企业解决方案公司, SAP, Oracle

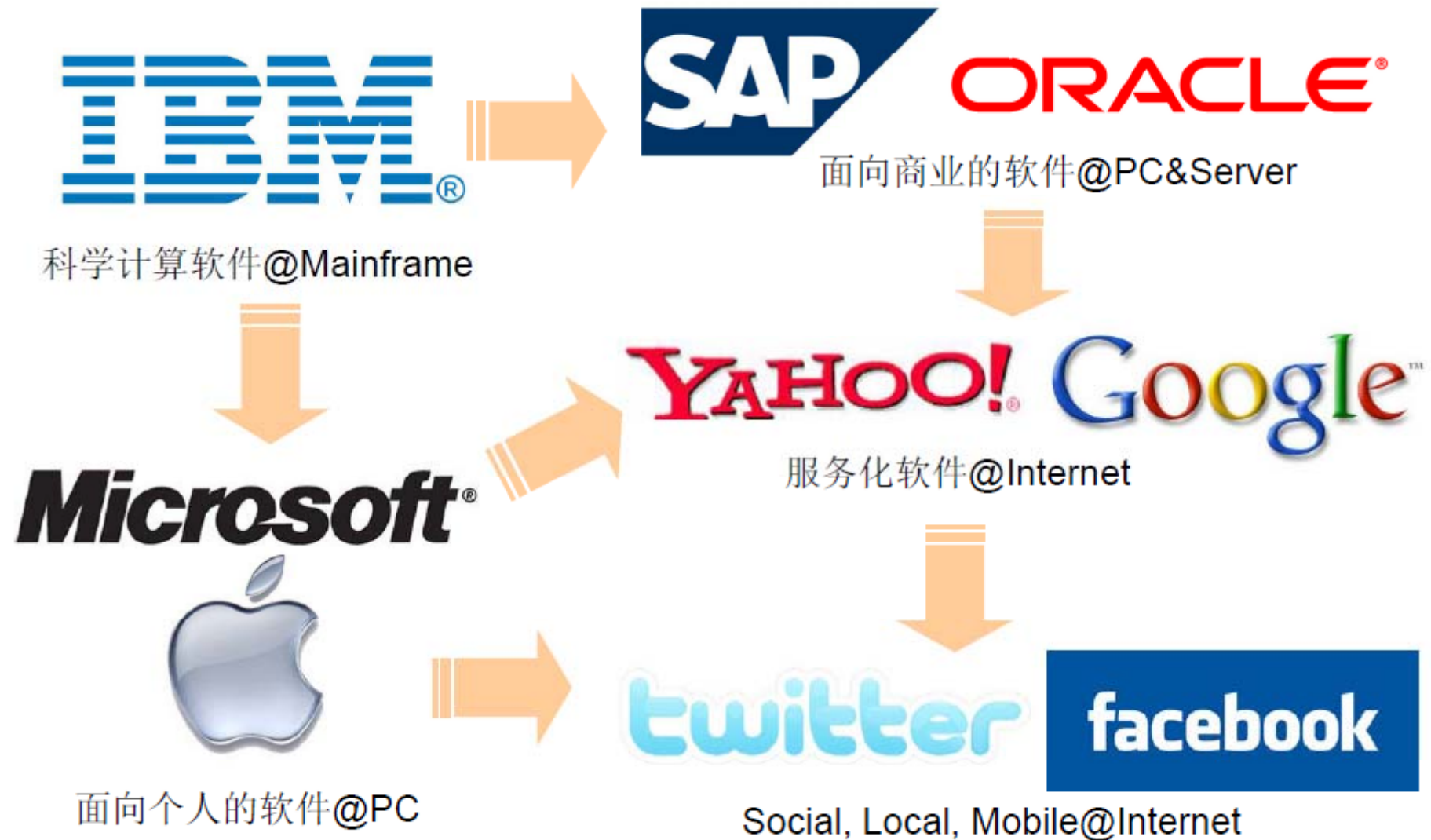
阶段4: 支持大众应用的软件包(Packaged Software for Mass)

1980s-1994: 个人为中心, 大众软件公司, Microsoft, Apple

阶段5: 网络软件与服务(Internet - Based Software and Service)

- 1990s, 网络为中心, 互联网公司, Netscape, Yahoo, Google
- 2000s, 社会为中心, 服务公司, Facebook, Twitter, YouTube

软件产业的发展历程



软件之启蒙 - 初识软件工程专业

软件工程专业学科及范畴

徐晓飞 教授、博导



视频位置

软件工程学科的兴起

- **萌芽：**1970s末，美国制定研究生教育计划时采纳了IEEE/CS提出的制定软件工程教程的建议。
- **酝酿：**1980s末，美国卡内基-梅隆大学软件工程研究所（CMU SEI）开始研究软件工程教育，组织软件工程教育研讨会，出版教程；建立软件工程硕士教育计划。
- **准备：**1993年，IEEE CS和ACM成立了软件工程协调委员会（SWECC），提出了“软件工程职业道德规范”、“本科软件工程教育计划评价标准”和“软件工程知识体系”（SWEBOK），全面描述软件工程实践所需的知识，为软件工程教育发展打下坚实基础。

软件工程学科的兴起

- **软件工程学科的产生：**2004年8月，世界**500**余位大学、研究所和企业界教授专家推出了软件工程知识体系、软件工程教育知识体系（**SEEK**）两个文件的最终版本，标志着软件工程学科在世界范围正式确立，并成为计算学科领域的独立学科。

中国软件工程专业学科：

- **2001年**, 教育部建立**35**所国家示范性软件学院，并批准新增“软件工程”本科专业。现已有**400**多高校设有软件工程专业。
- **2011年**, 国务院学位委员会批准新增“软件工程”一级学科博士点与硕士点。现已有**100**多高校设有软件工程博士/硕士点。
- **2013年**, 教育部成立独立的“软件工程专业教学指导委员会”。

软件工程科学技术体系

◆ 软件工程科学技术体系

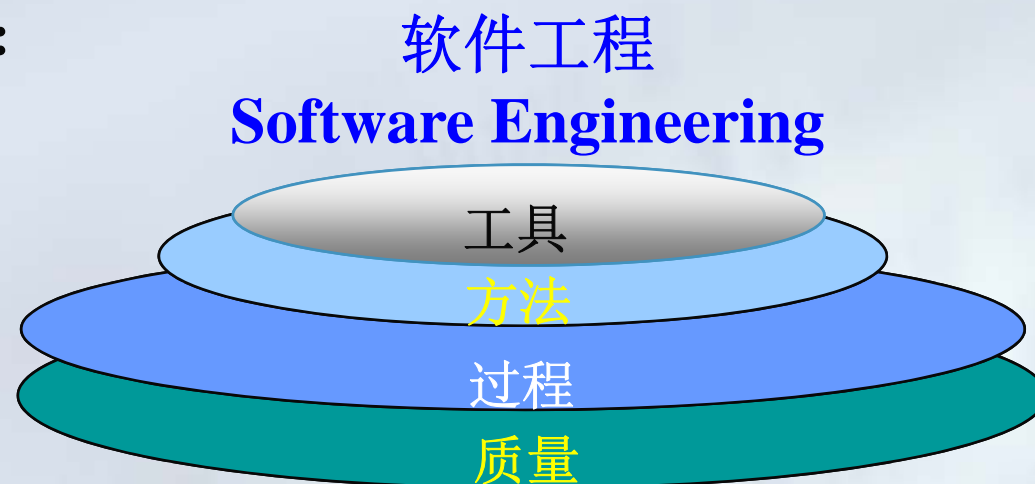
➤ 软件工程的研究对象：

软件 + 工程

（理论、技术、管理、方法；目标、原则、活动）

➤ 软件工程的关注点：

- 软件开发过程
- 软件开发方法
- 软件开发工具
- 软件开发质量



软件工程科学技术体系

◆ 软件工程科学技术范畴

- 软件过程 (**Software Process**)
- 软件开发方法 (**Software Development Method**)
- 软件需求工程 (**Software Requirement Engineering**)
- 软件体系结构 (**Software Architecture**)
- 软件开发工具与环境 (**Software Development Environment**)
- 软件复用与软构件 (**Software Reuse & Component**)
- 软件中间件 (**Software Middleware**)
- 软件测试 (**Software Testing**)
- 软件维护 (**Software Maintenance**)

软件工程科学技术体系

◆ 软件工程科学技术范畴

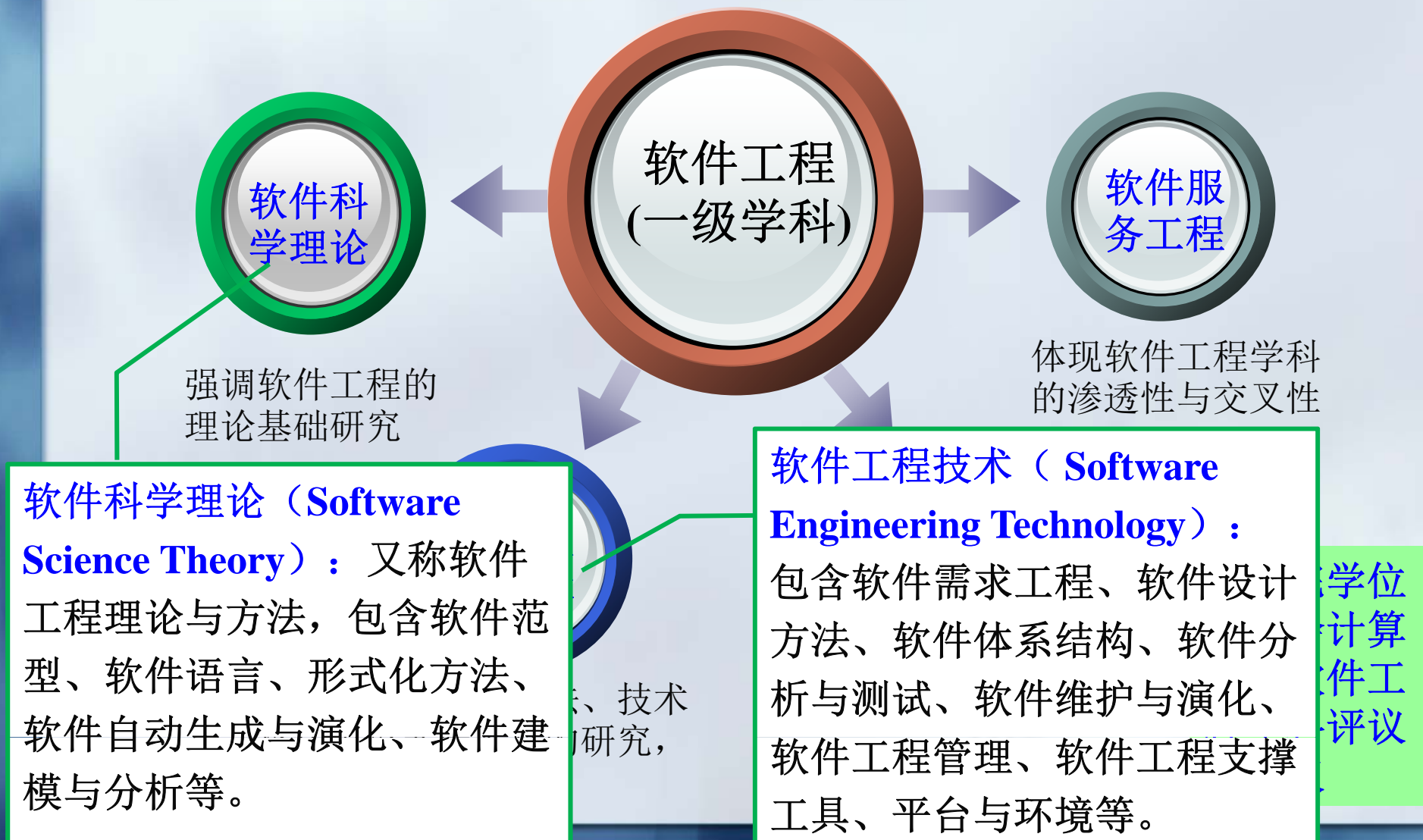
- 软件过程度量 (Software Process)
- 软件工程管理 (Software Engineering Management)
- 软件质量(Software Quality)
- 软件可信性 (Software Dependability)
- 软件服务工程 (Software Service Engineering)
- 领域工程 (Domain Engineering)
- 软件逆向工程 (Software Reverse Engineering)
-

软件工程学科的设置

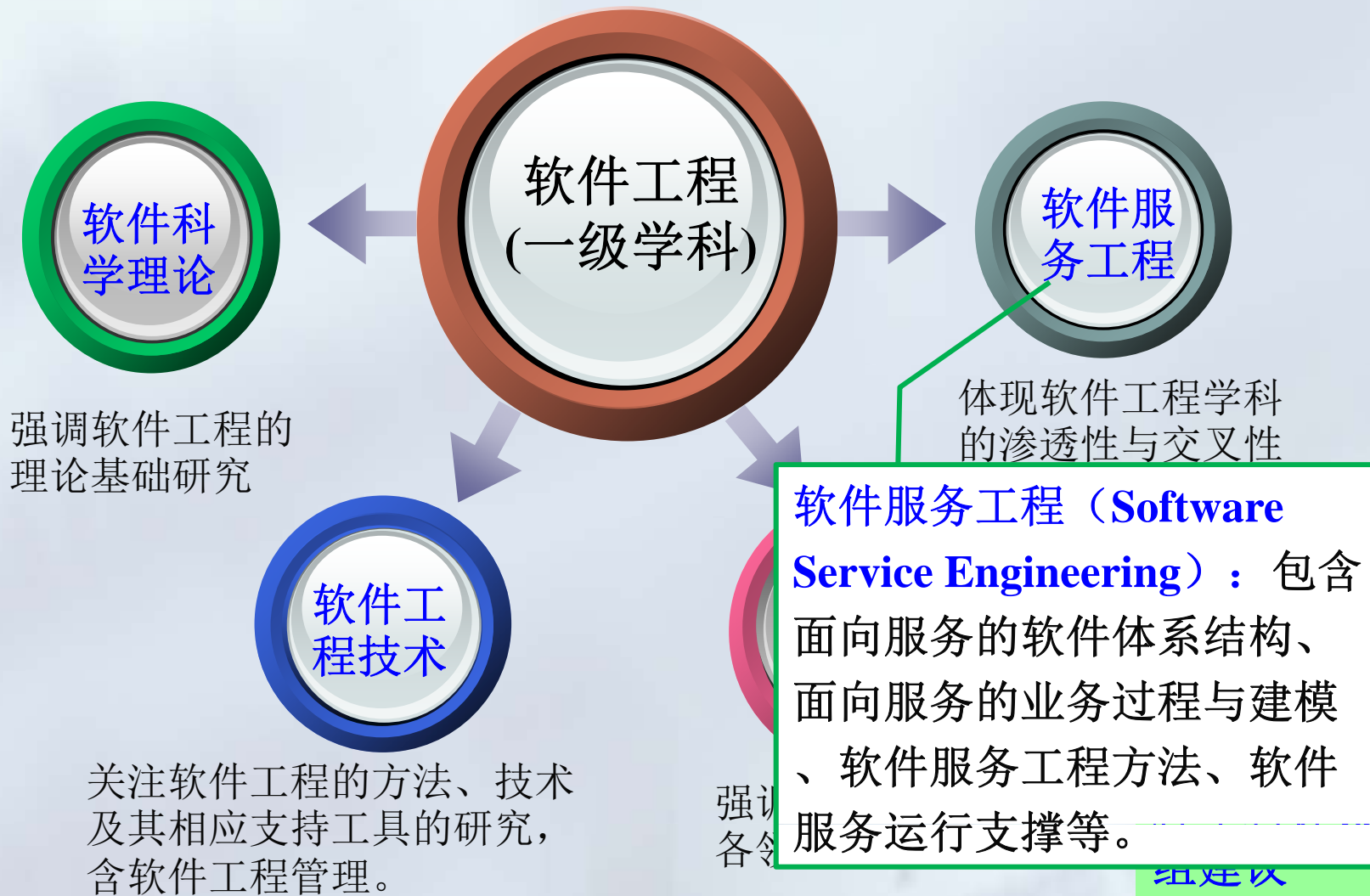


国务院学位
委员会计算
机与软件工
程学科评议
组建议

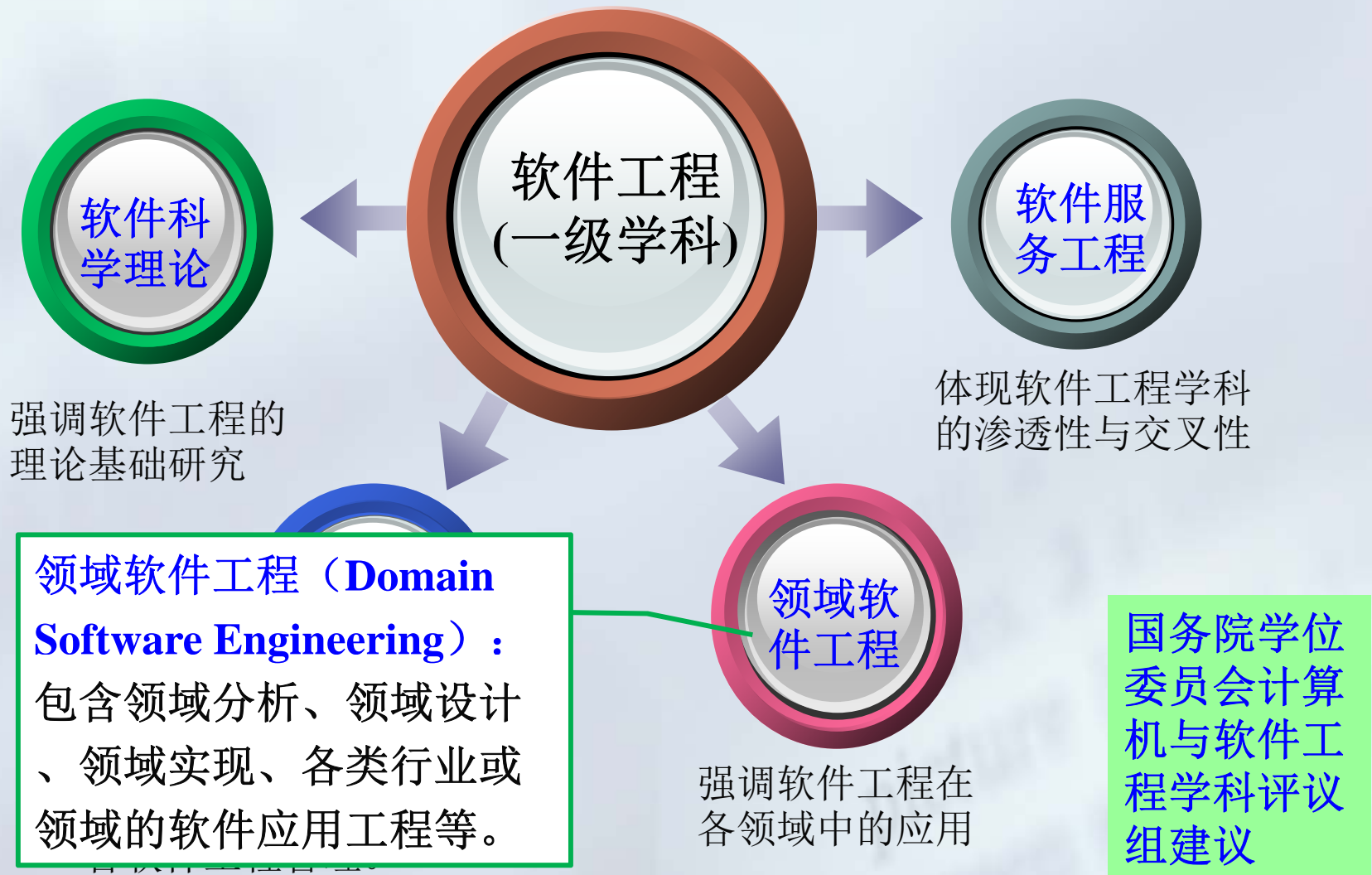
软件工程学科的设置



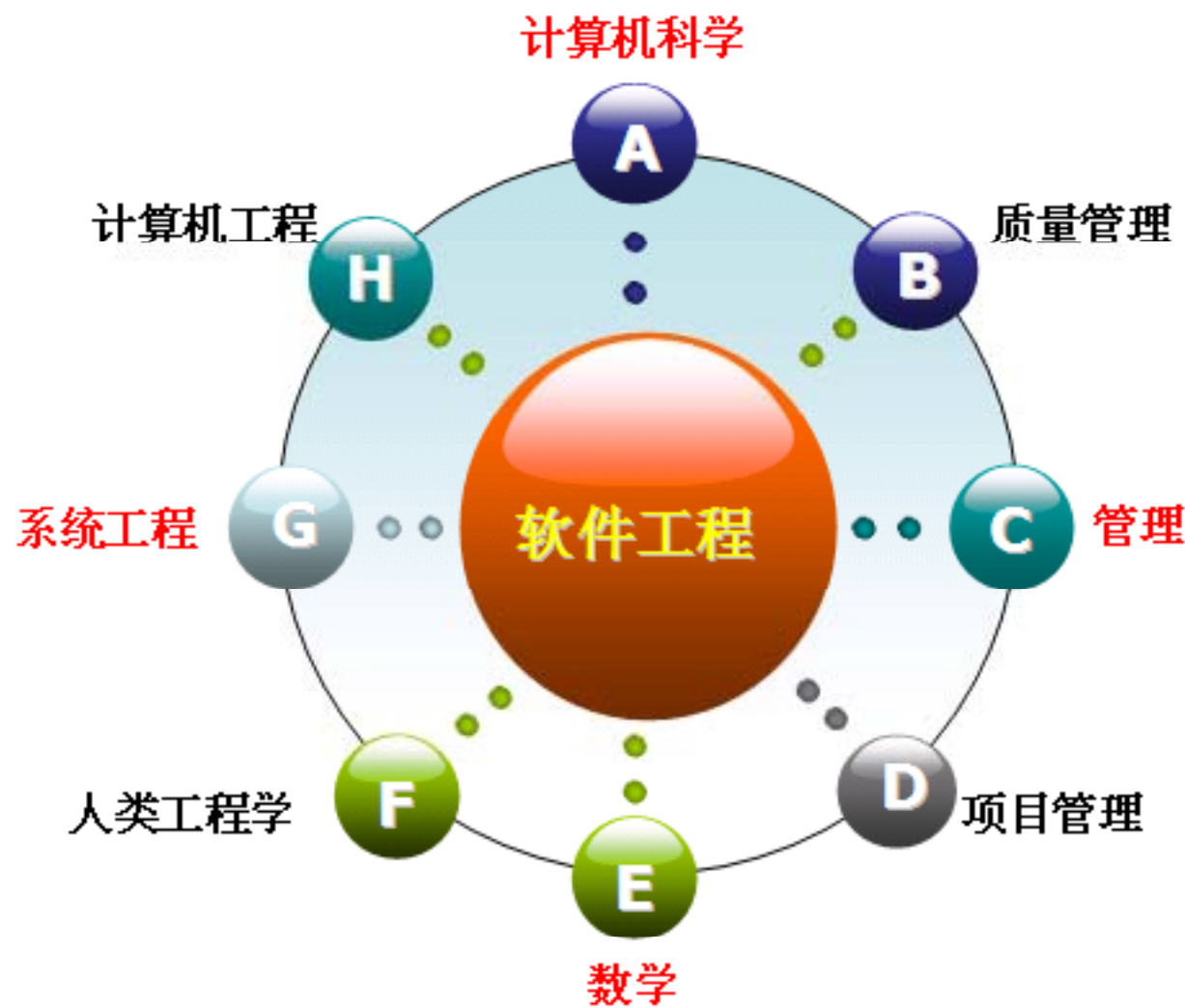
软件工程学科的设置



软件工程学科的设置



软件工程学科的相关学科



软件之启蒙 - 初识软件工程专业

软件工程的知识体系

徐晓飞 教授、博导



视频位置

软件工程专业知识体系

 **SWEBOK®**
V3.0
Guide to the Software
Engineering Body of Knowledge

Editors
Pierre Bourque
Richard H. Clark

IEEE
computer society

◆ IEEE CS 软件工程专业知识体系SWEBOK

- 美国电子电气工程师学会IEEE CS与美国计算机联合会ACM成立了软件工程协调委员会，于1994年开始研究软件工程知识体系（SWEBOK）；历经草人阶段、石人阶段和铁人阶段；IEEE CS于2001、2004年先后发布SWEBOK 1.0，2.0版。
- 2014年3月，IEEE CS正式发布SWEBOK 3.0版，成为软件工程知识体系的样板。
- **SWEBOK**是“普遍接受的知识，并应用于大多数软件项目；有广泛一致的意见都确认其价值和效力”。
- **SWEBOK**意味着“有能力的软件工程人员，为了胜任潜在的应用应该具有这些知识”。

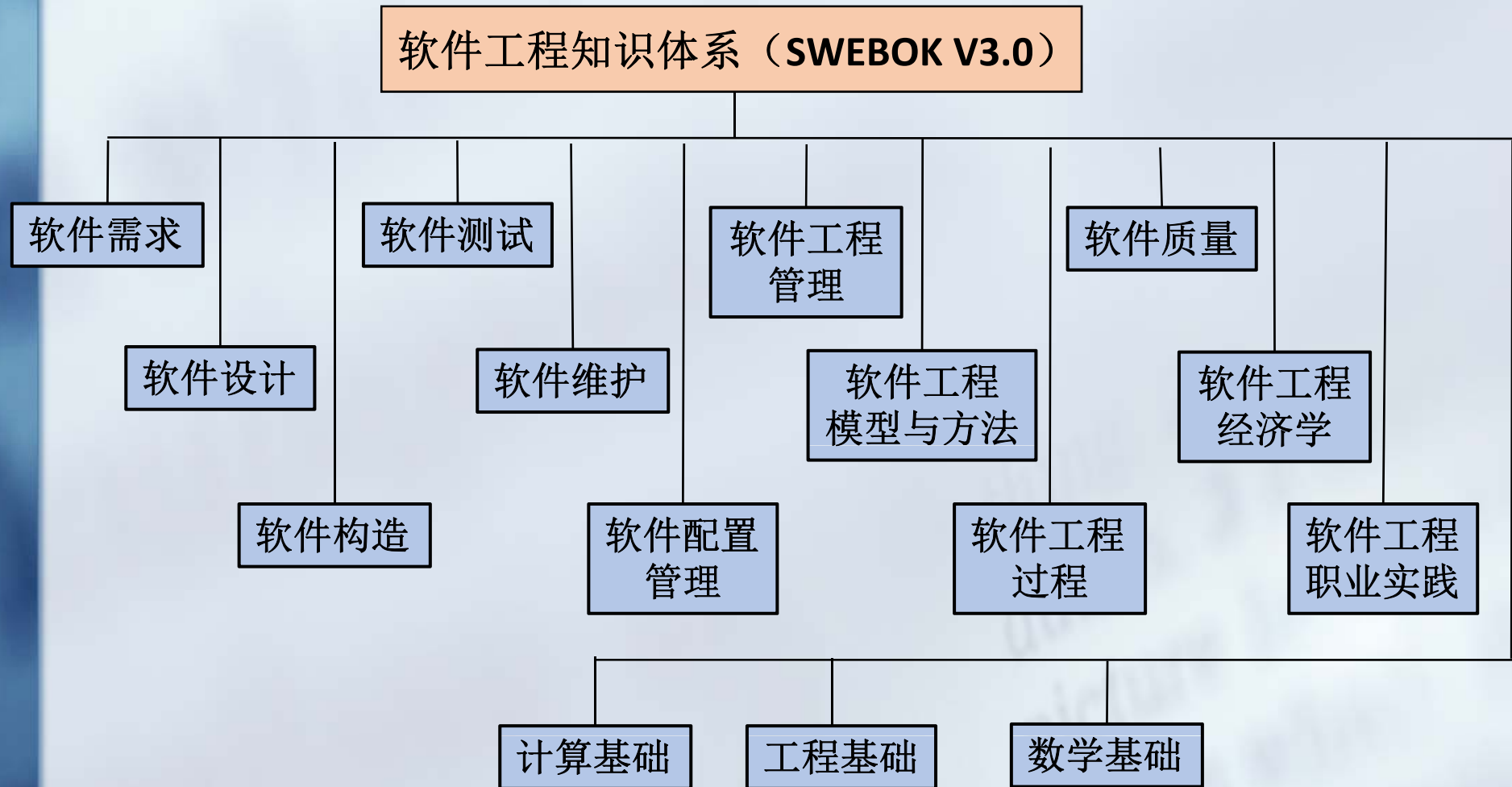
软件工程专业知识体系

IEEE CS: SWEBOK V3.0

SWEBOK[®]
V3.0
Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
computer society



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

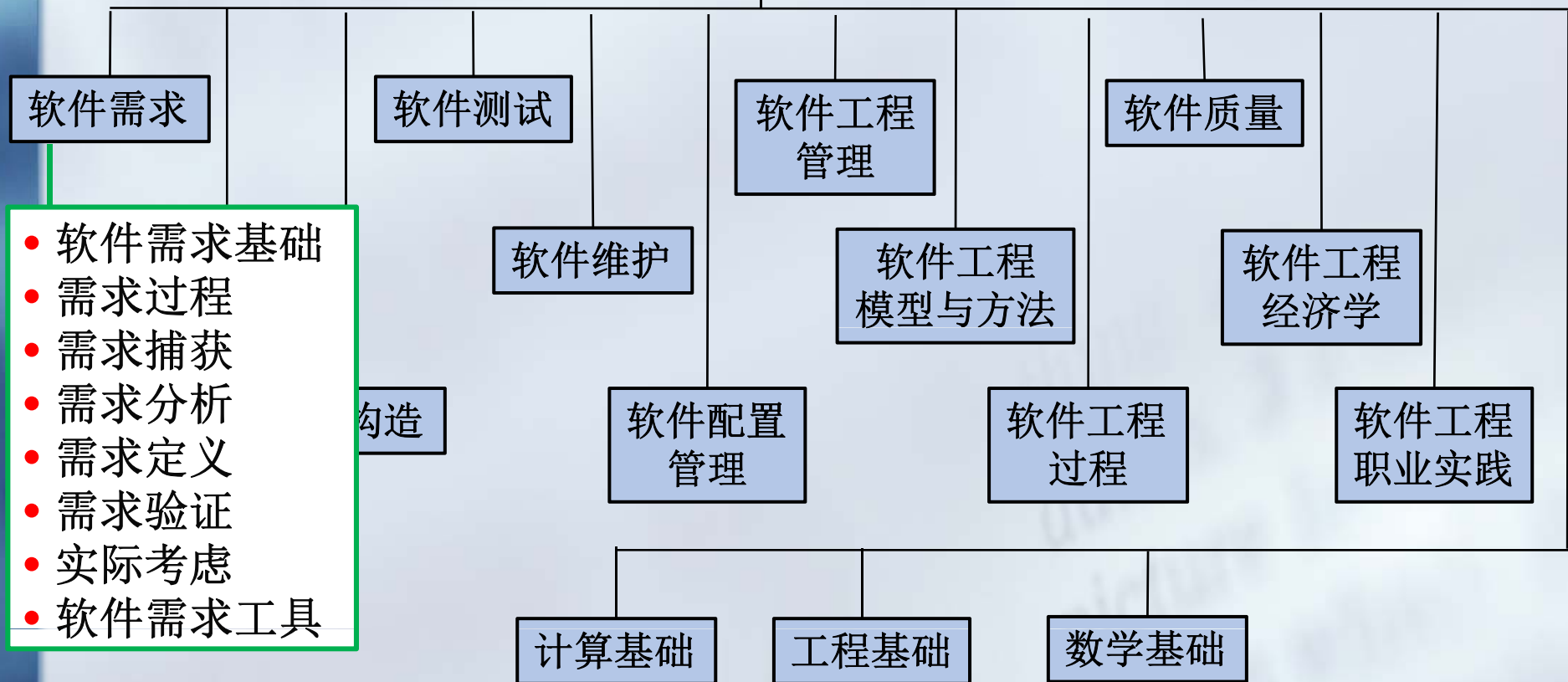
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
© 2011 IEEE Computer Society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

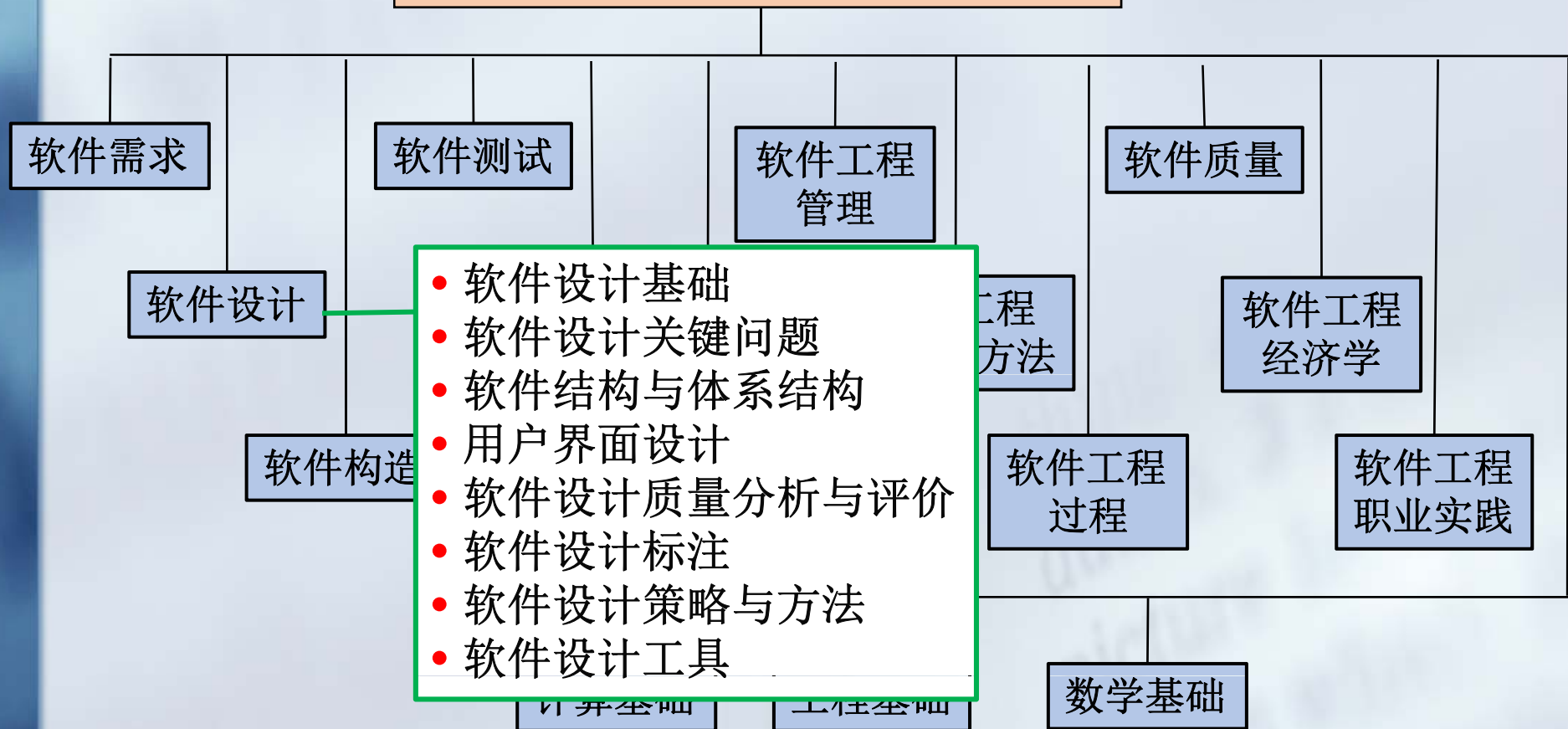
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
computer society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

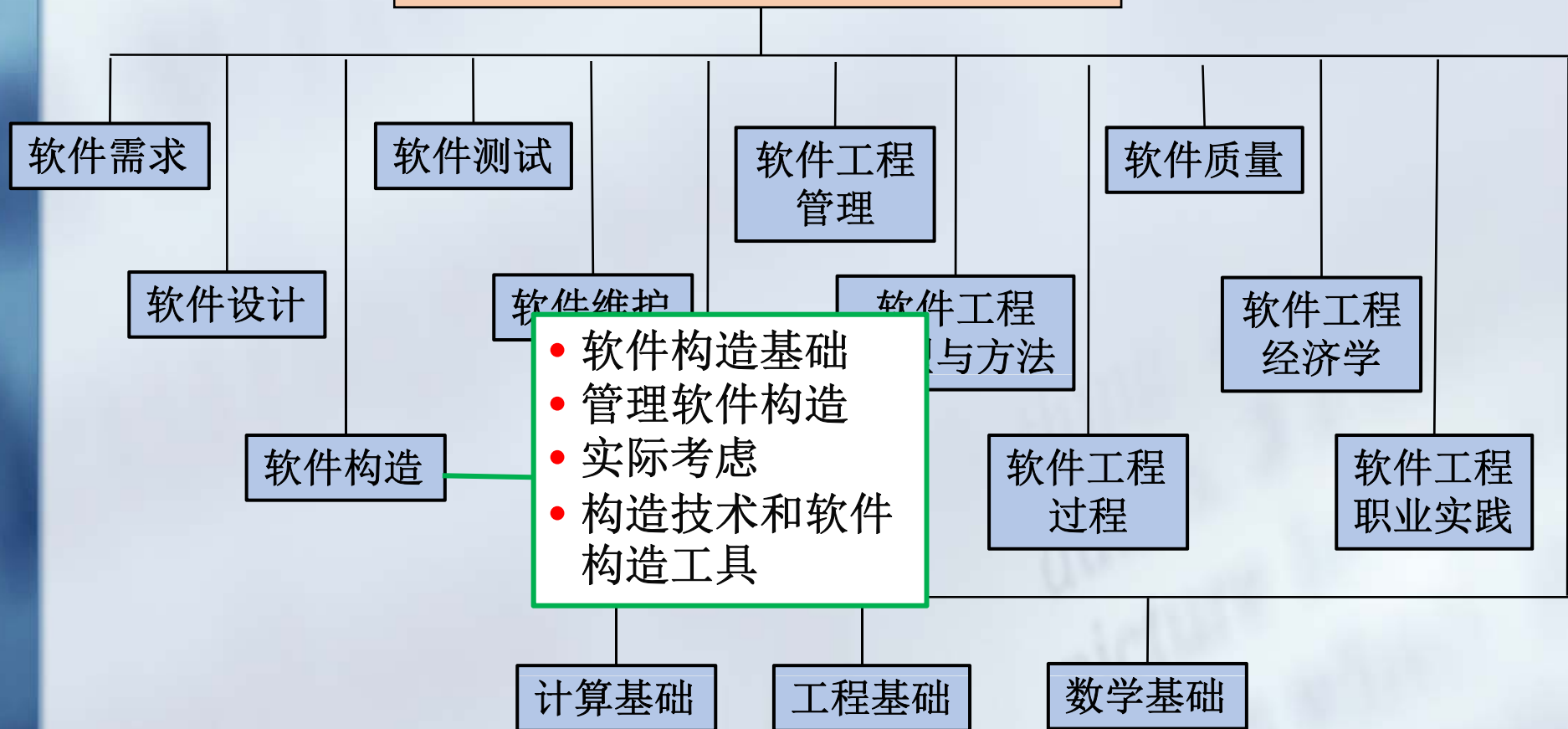
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
computer society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

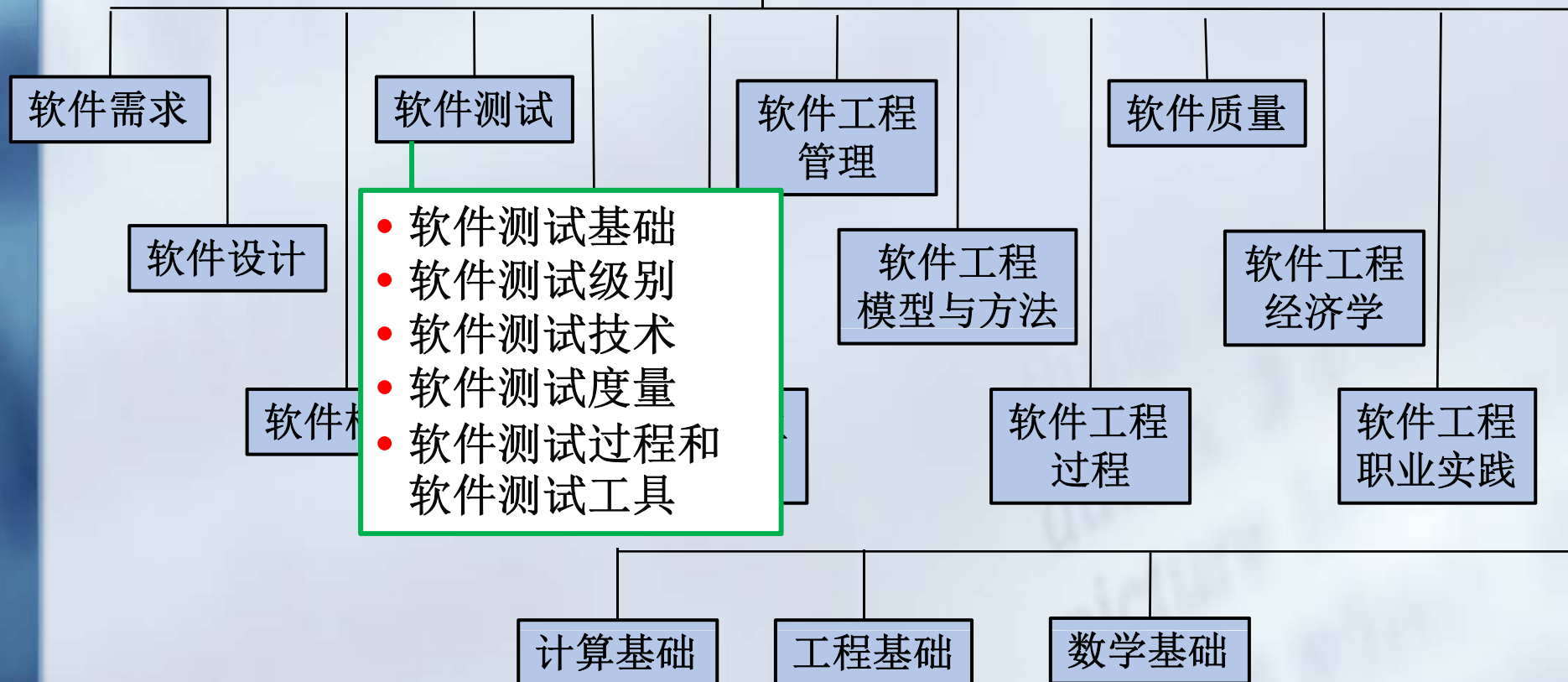
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilkerson

IEEE
computer society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

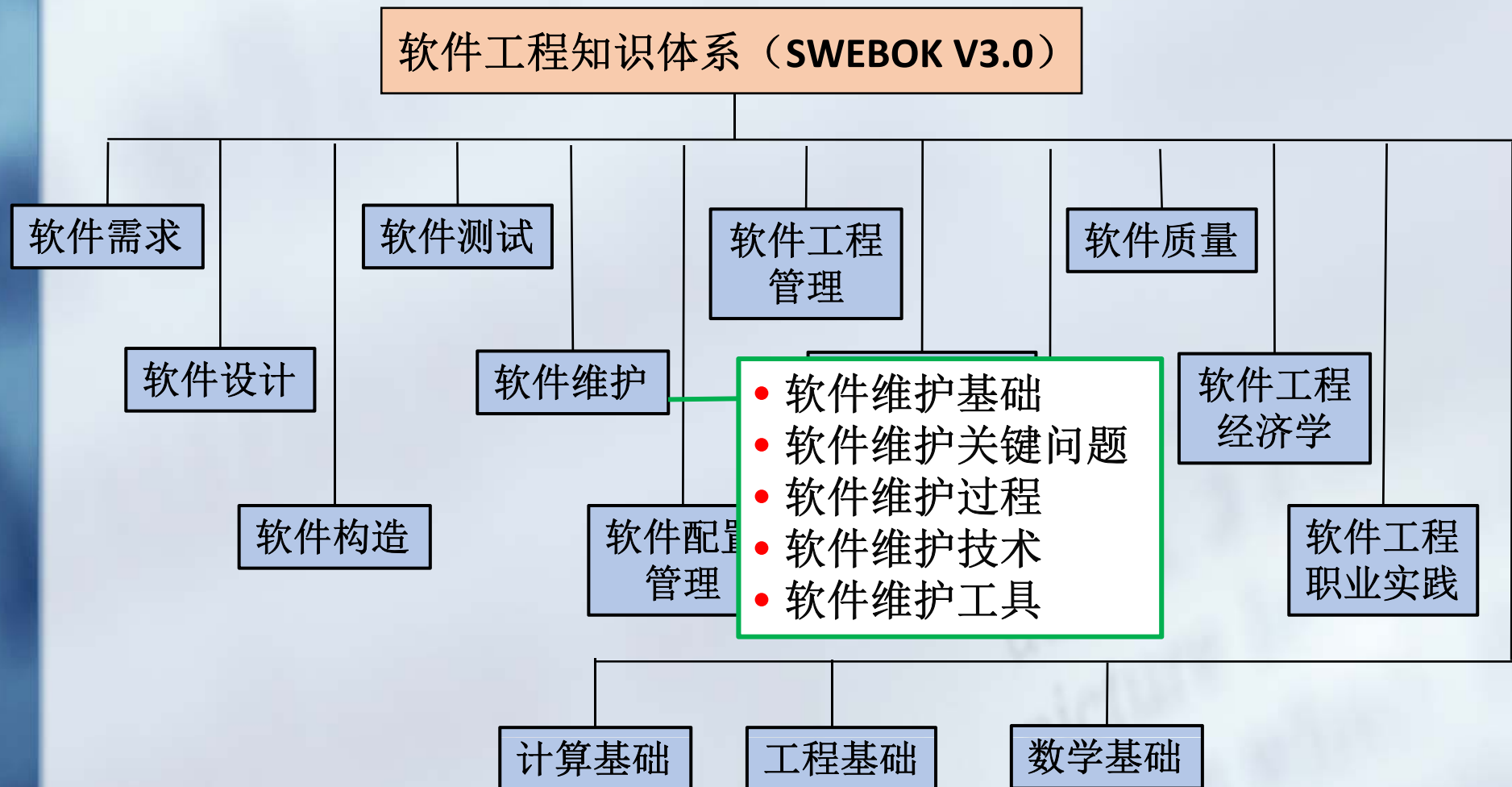
IEEE CS: SWEBOK V3.0

SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
© 2011 IEEE Computer Society

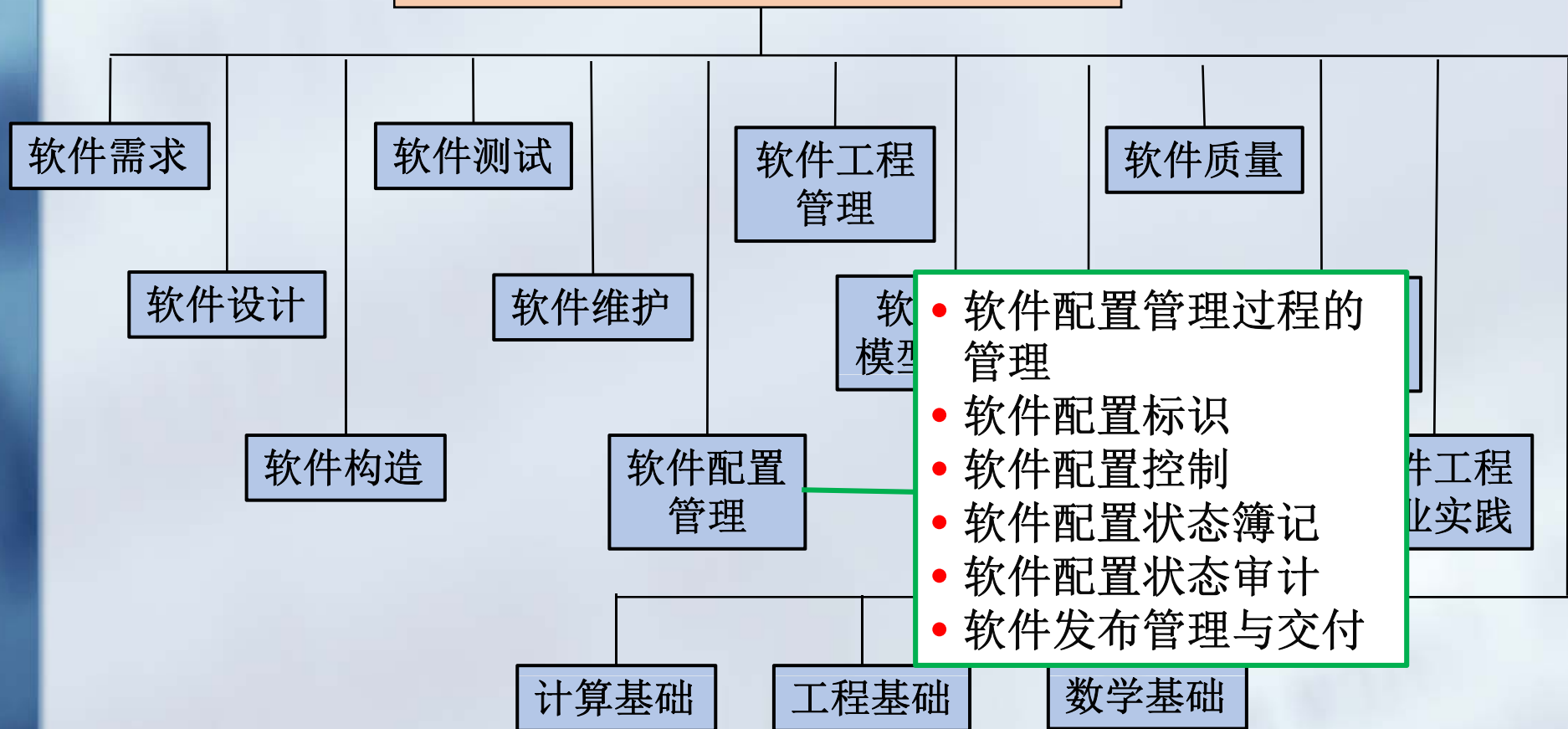


软件工程专业知识体系

IEEE CS: SWEBOK V3.0



软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

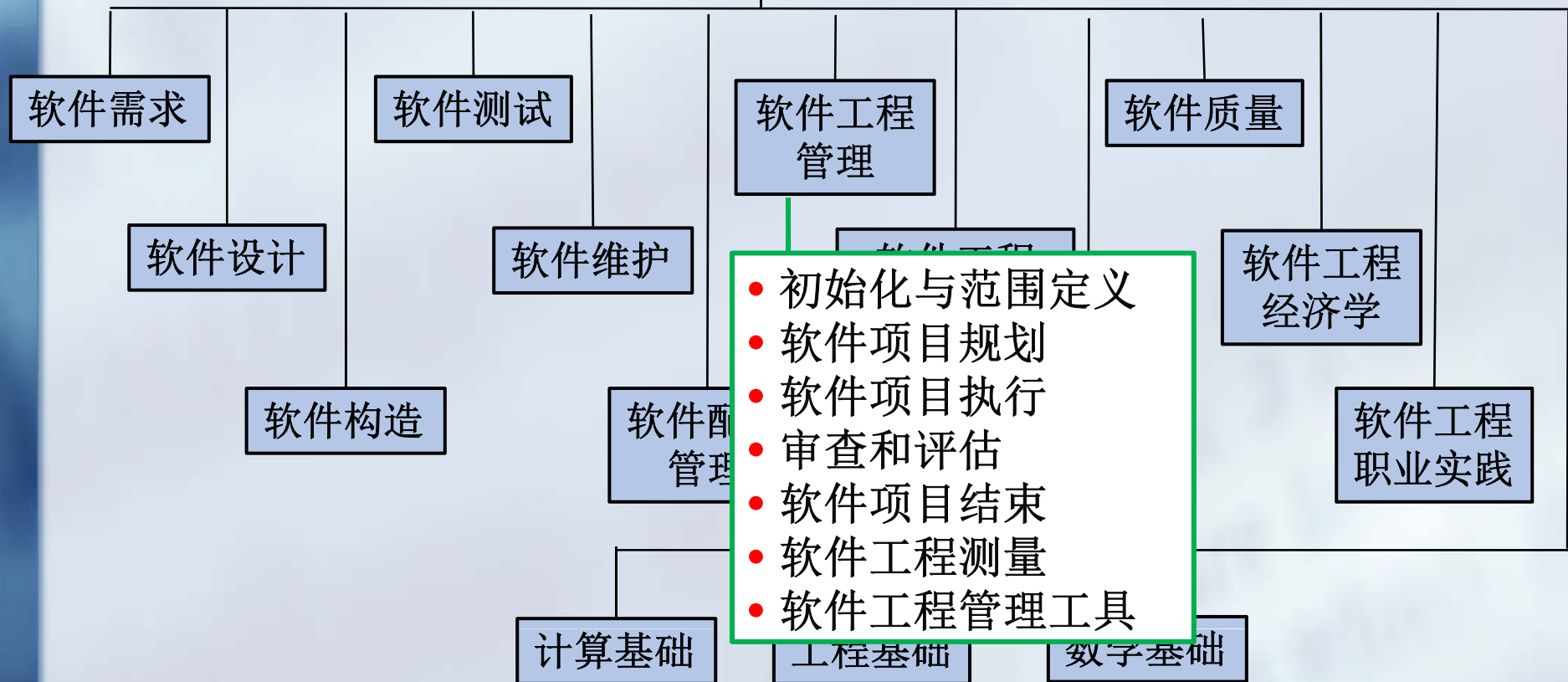
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. (Dick) Flory

IEEE
© 2011 IEEE Computer Society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

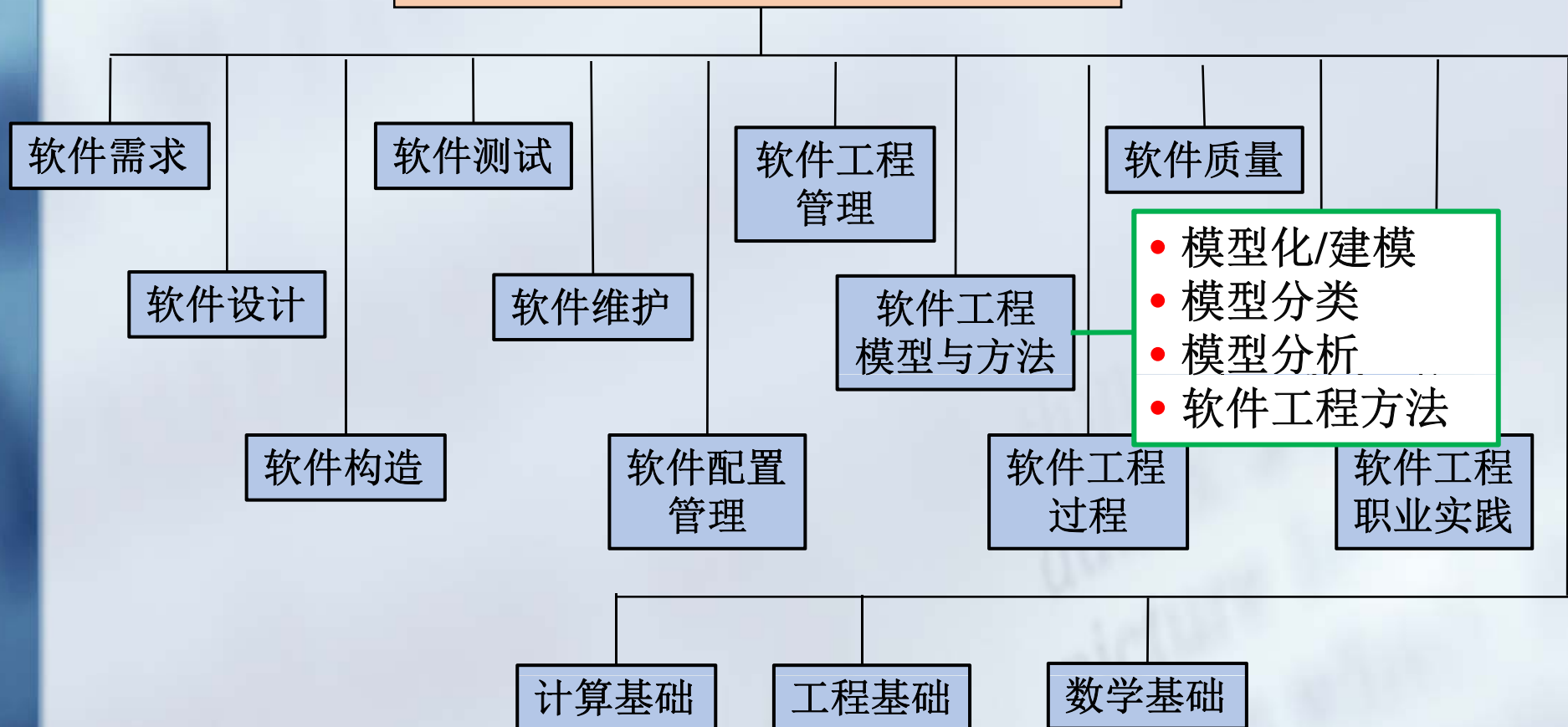
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
© 2011 IEEE Computer Society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

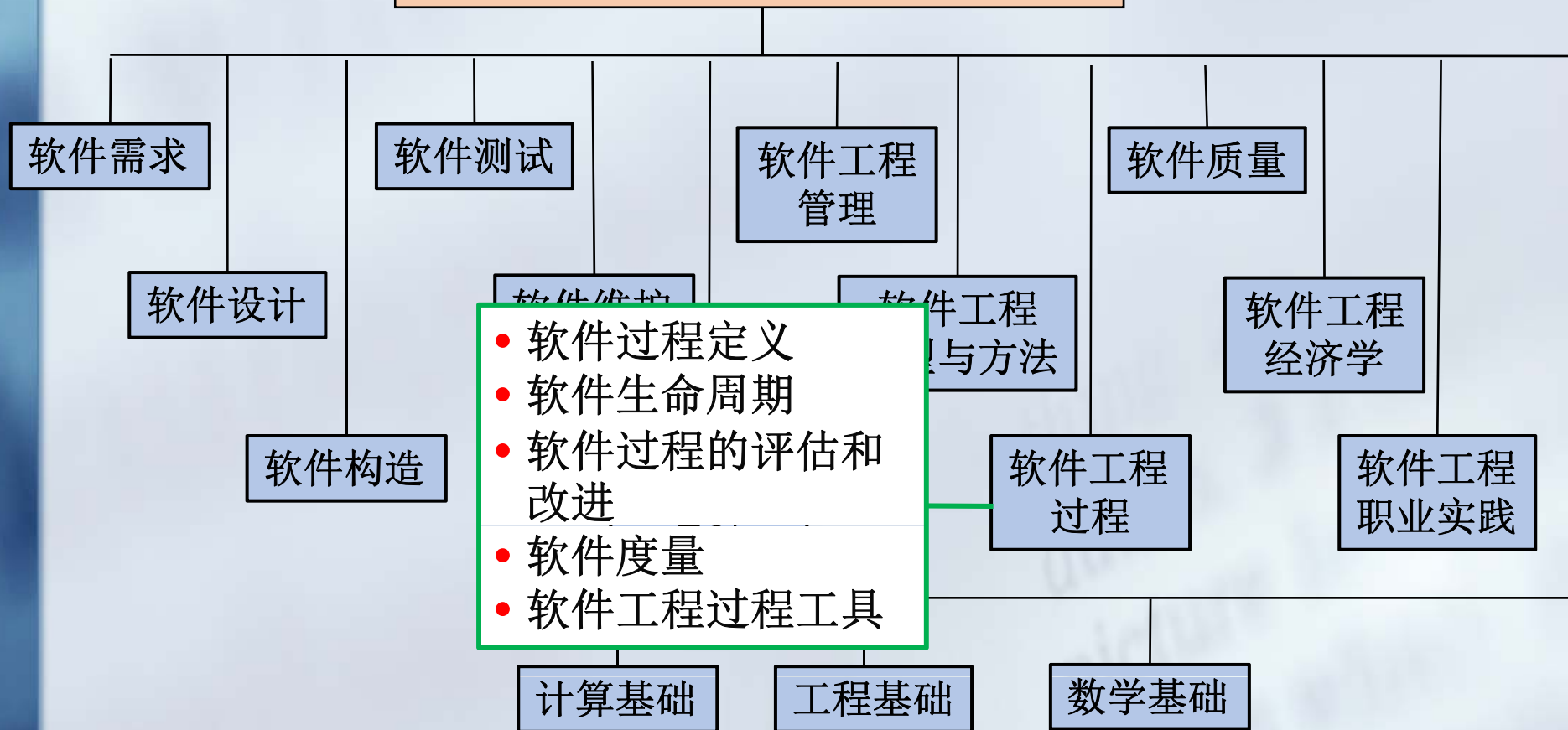
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
computer society

软件工程知识体系 (SWEBOK V3.0)

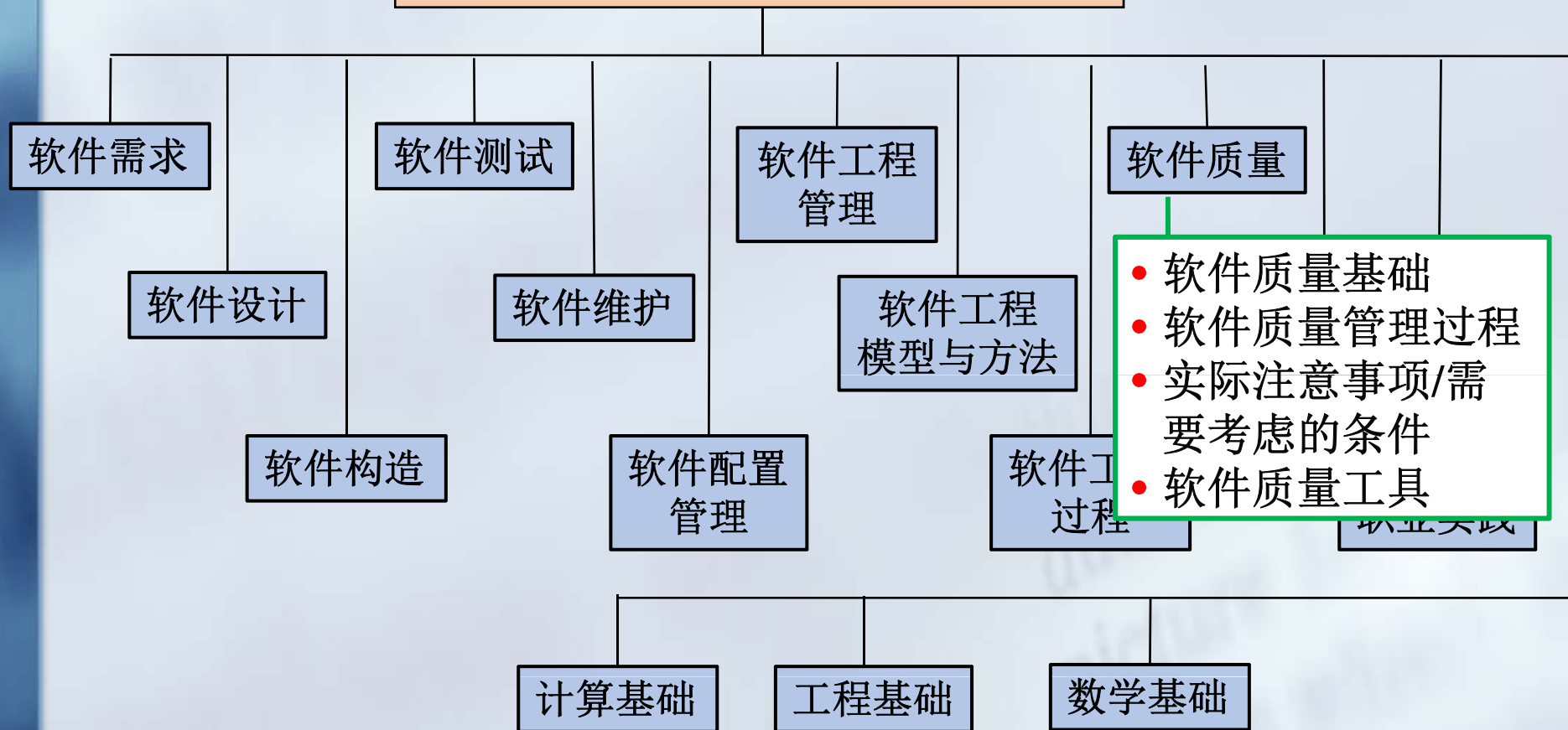


软件工程专业知识体系

IEEE CS: SWEBOK V3.0

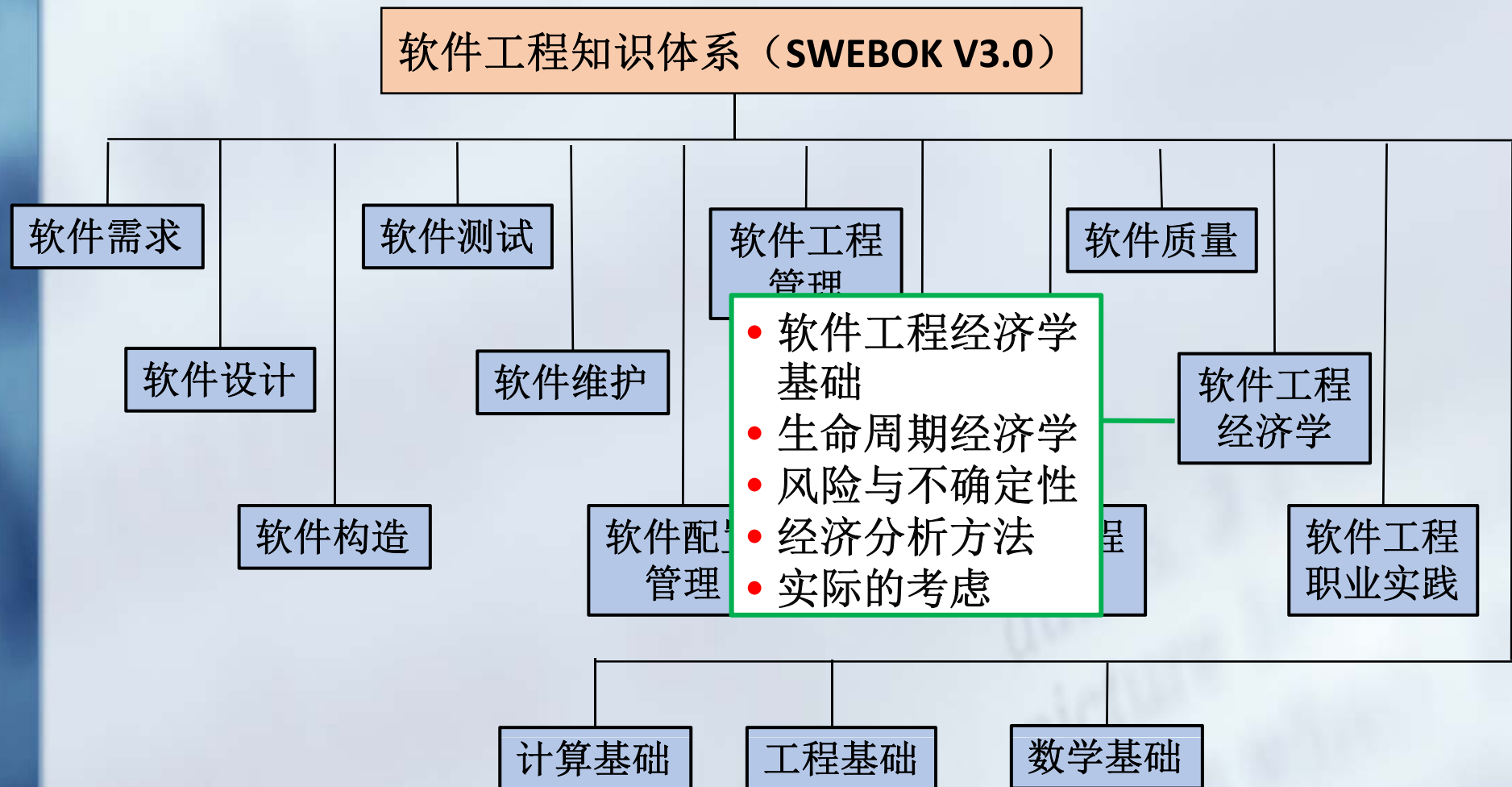


软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0



软件工程专业知识体系

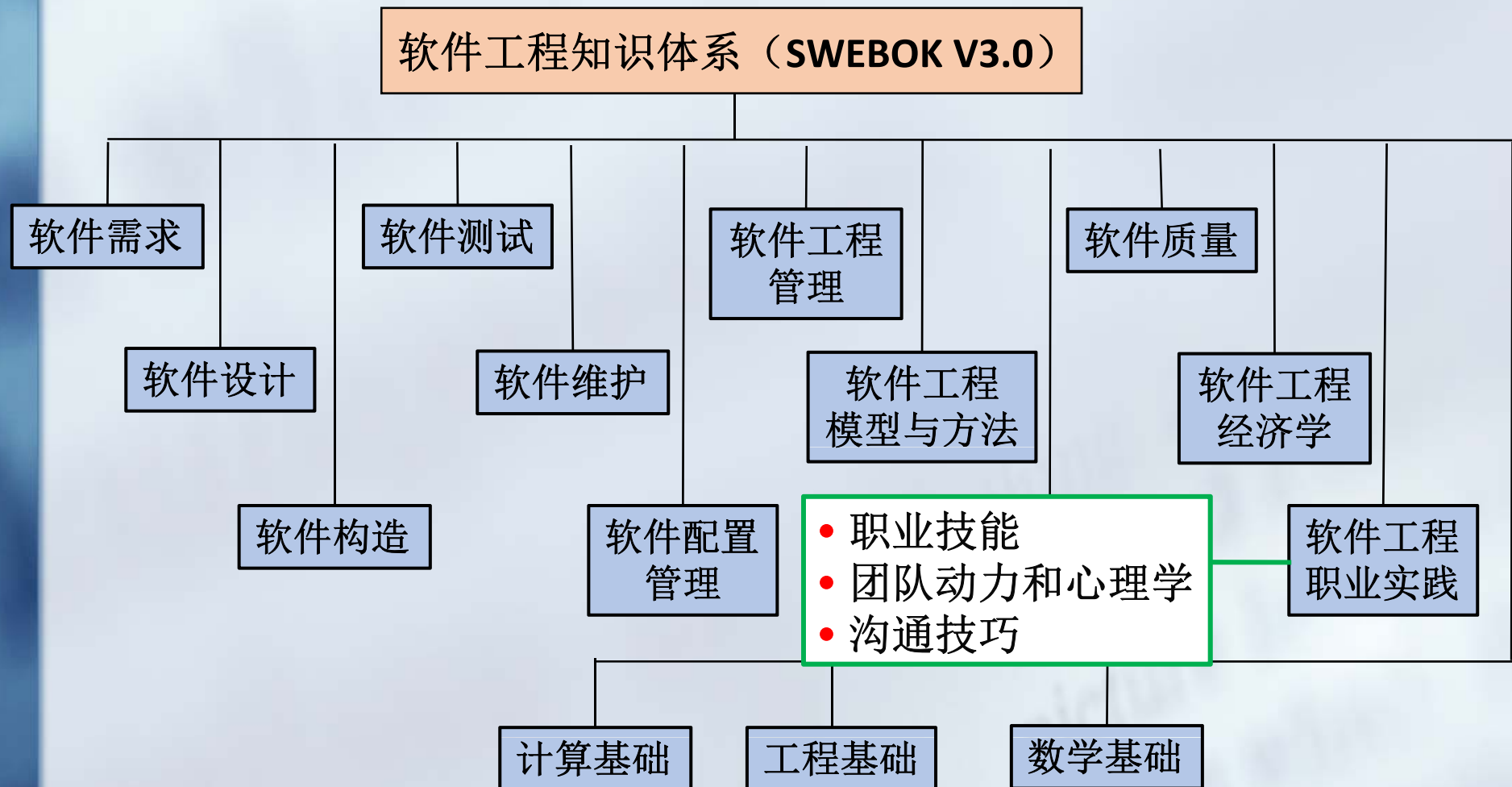
IEEE CS: SWEBOK V3.0

SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
© 2011 IEEE Computer Society



软件工程知识体系

IEEE CS: SWEBOK

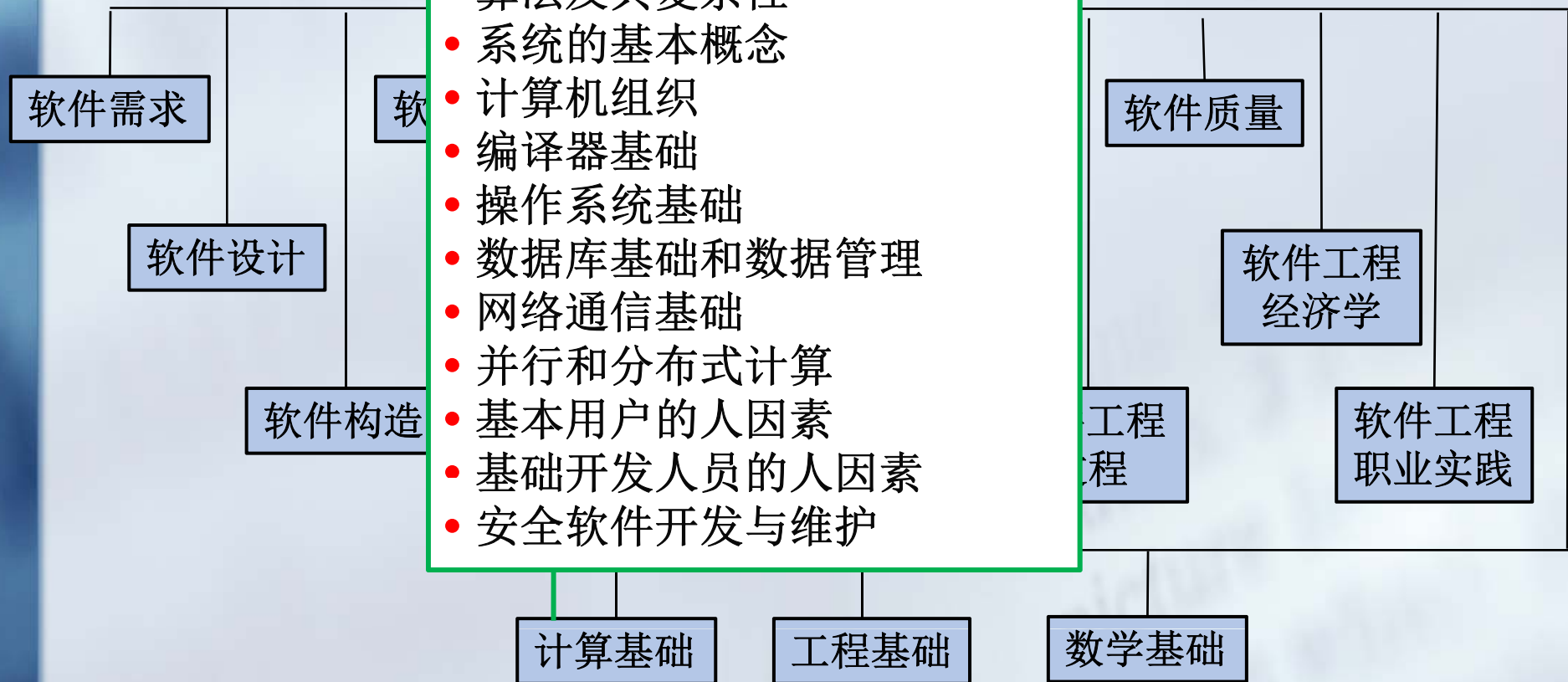
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilkerson

IEEE
computer society

- 问题求解技巧
- 抽象化
- 程序设计基础
- 编程语言基础
- 调试工具和技巧
- 数据结构和数据表示
- 算法及其复杂性
- 系统的基本概念
- 计算机组织
- 编译器基础
- 操作系统基础
- 数据库基础和数据管理
- 网络通信基础
- 并行和分布式计算
- 基本用户的人因素
- 基础开发人员的人因素
- 安全软件开发与维护



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

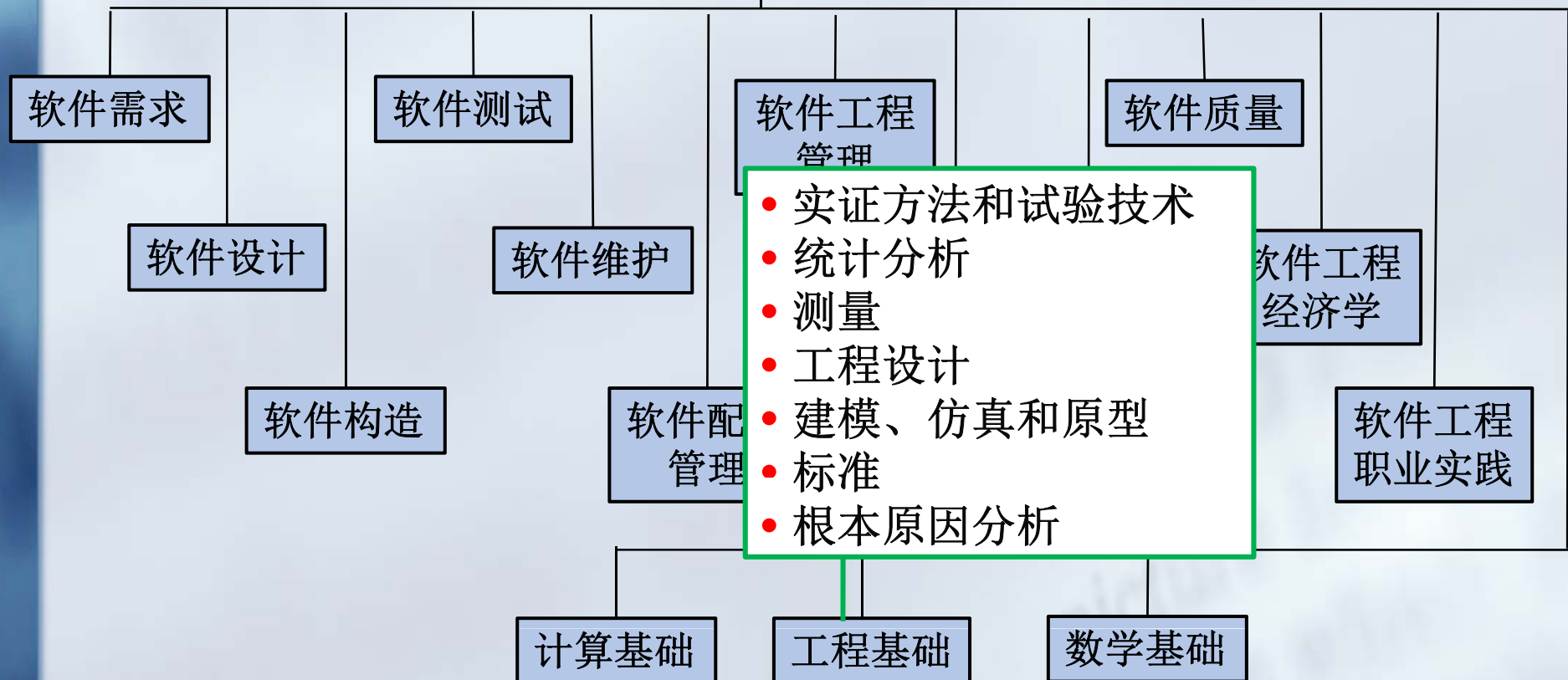
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. Gilk, Editor

IEEE
© 2011 IEEE Computer Society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业知识体系

IEEE CS: SWEBOK V3.0

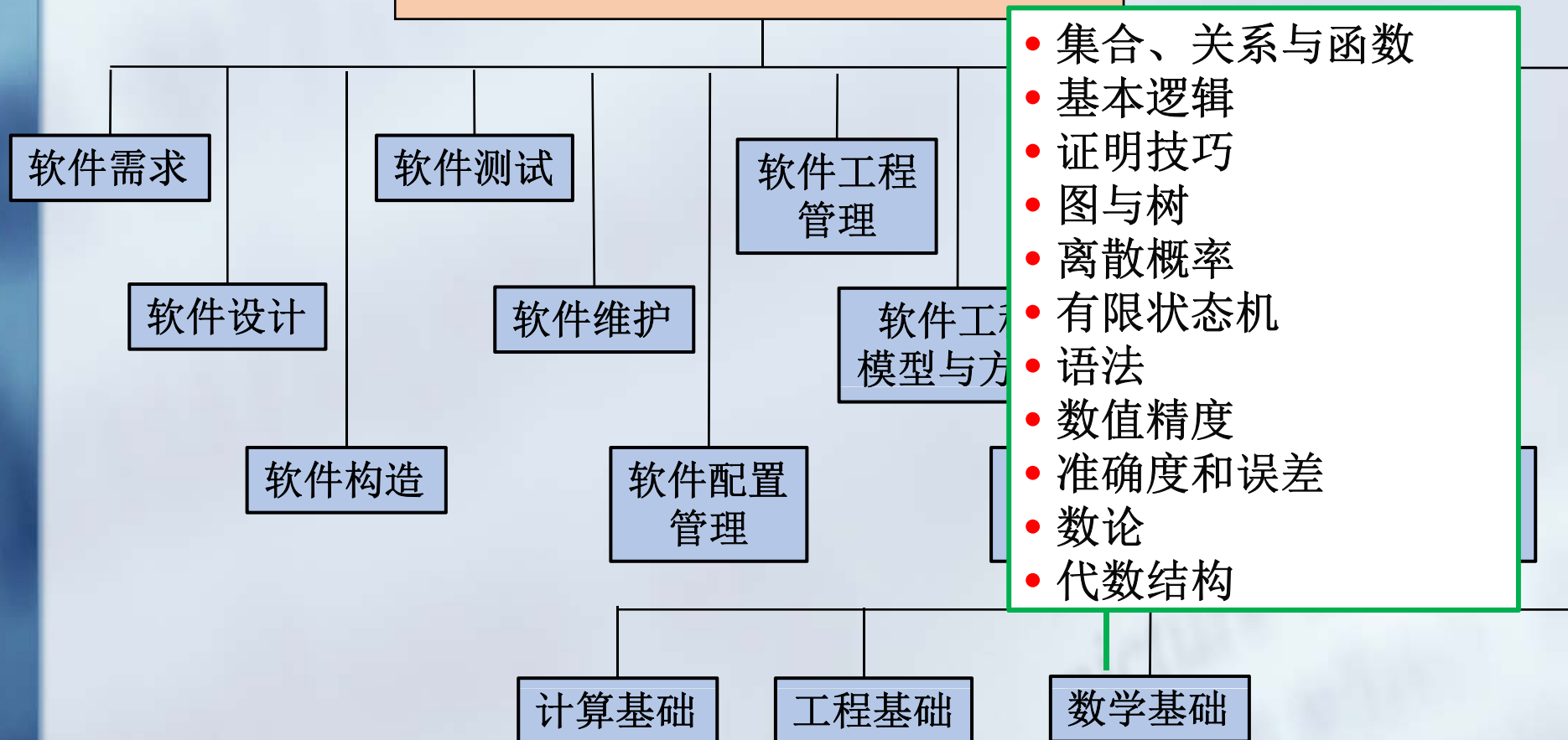
SWEBOK[®]
V3.0

Guide to the Software
Engineering Body of Knowledge

Editors
Barry Boehm
Richard F. (Dick) Flory

IEEE
© 2011 IEEE Computer Society

软件工程知识体系 (SWEBOK V3.0)



软件工程专业的知识体系

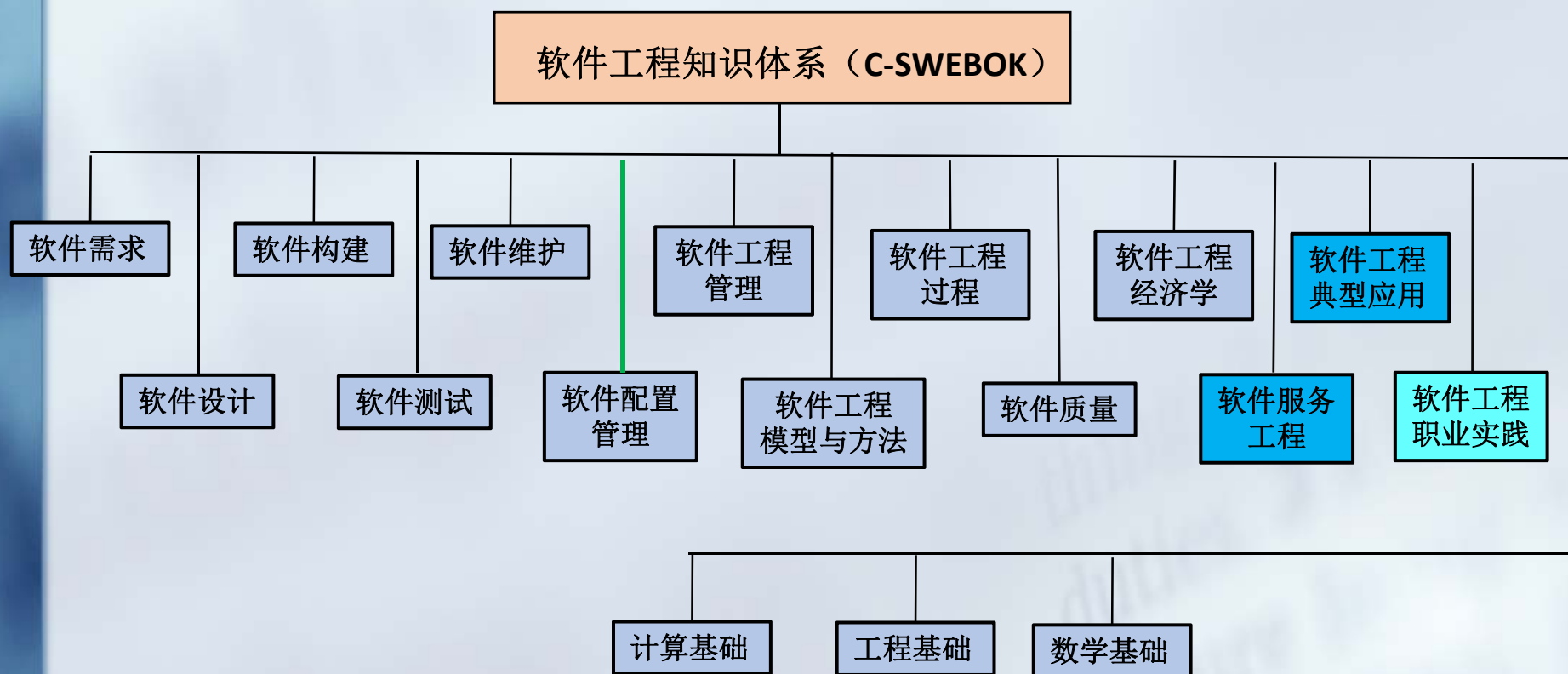
中国软件工程教指委：C-SWEBOK知识体系

- 中国软件工程知识体系**C-SWEBOK**

- 面向中国软件工程教育需要，基于**SWEBOK 3.0**及相关知识体系，结合中国软件工程教育的实践经验，提出面向中国的软件工程知识体系**C-SWEBOK**，支持中国高校的软件工程专业教育；
- 在中国建立软件工程学科知识结构的一致化概念，为中国高校软件工程专业教育与课程体系提供依据和指导。
- 增加软件服务工程、软件工程典型应用等**2**个知识领域**KA**，改造与扩充软件工程职业实践知识领域
- **C-SWEBOK**包含**17**个知识领域**KA**。

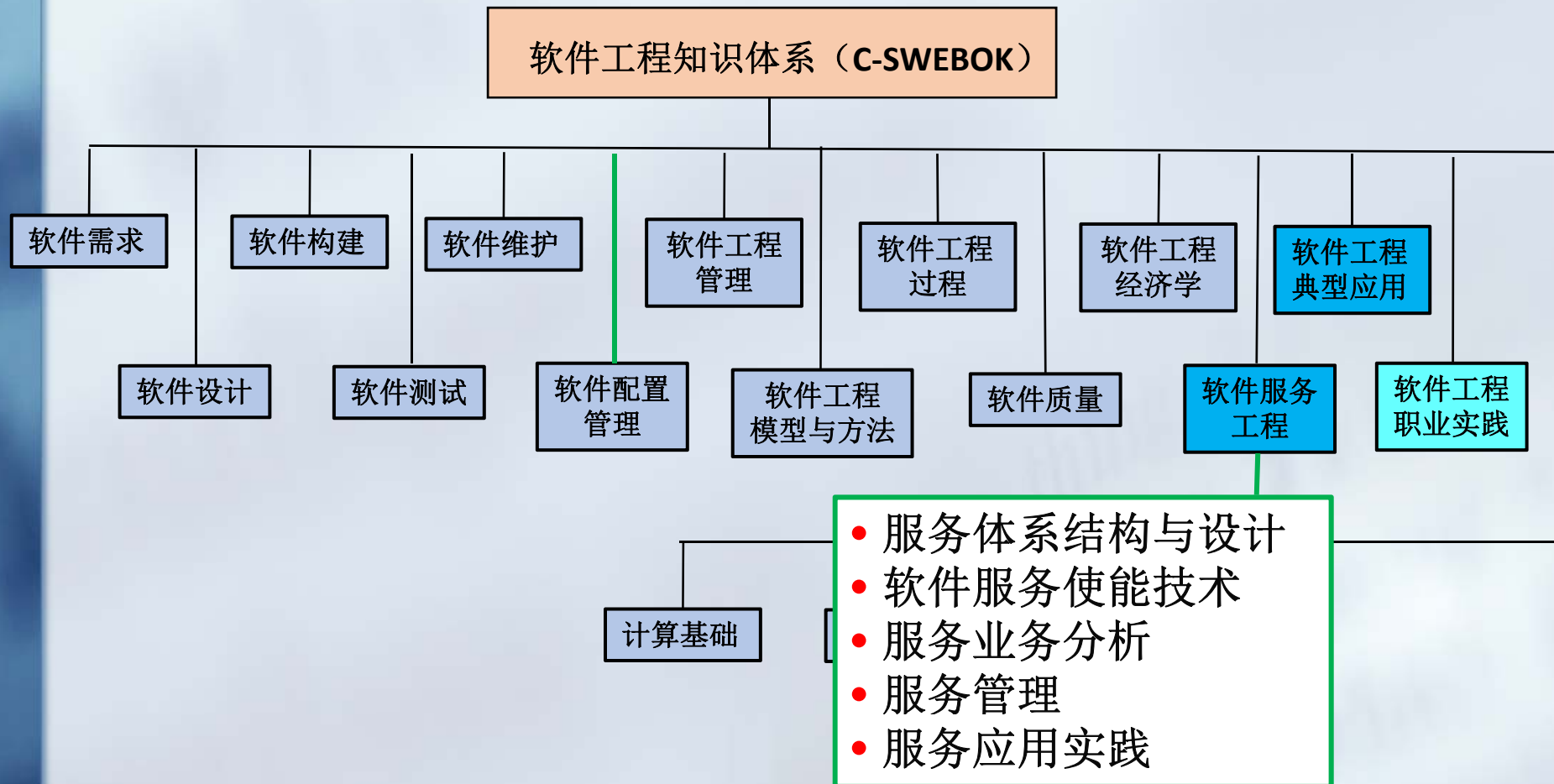
软件工程专业知识体系

中国软件工程教指委：C-SWEBOK知识体系



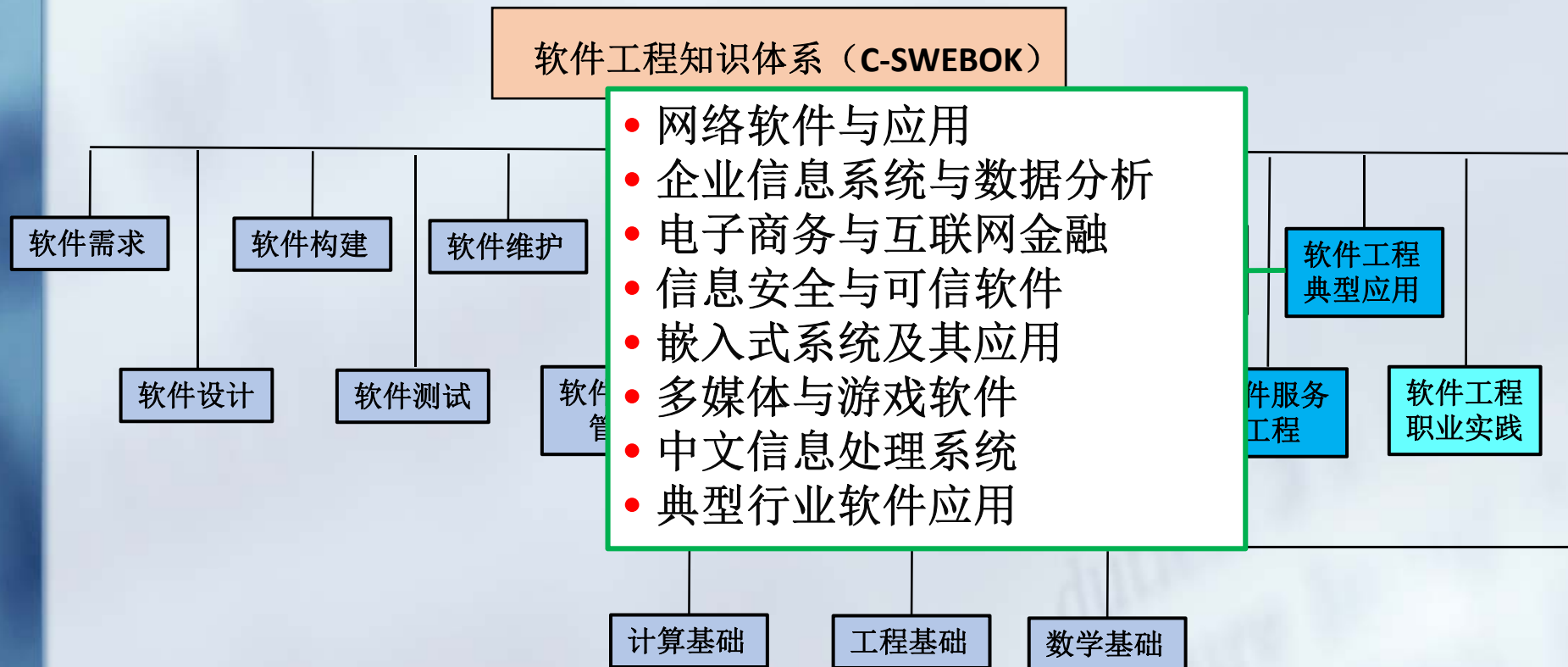
软件工程专业知识体系

中国软件工程教指委：C-SWEBOK知识体系



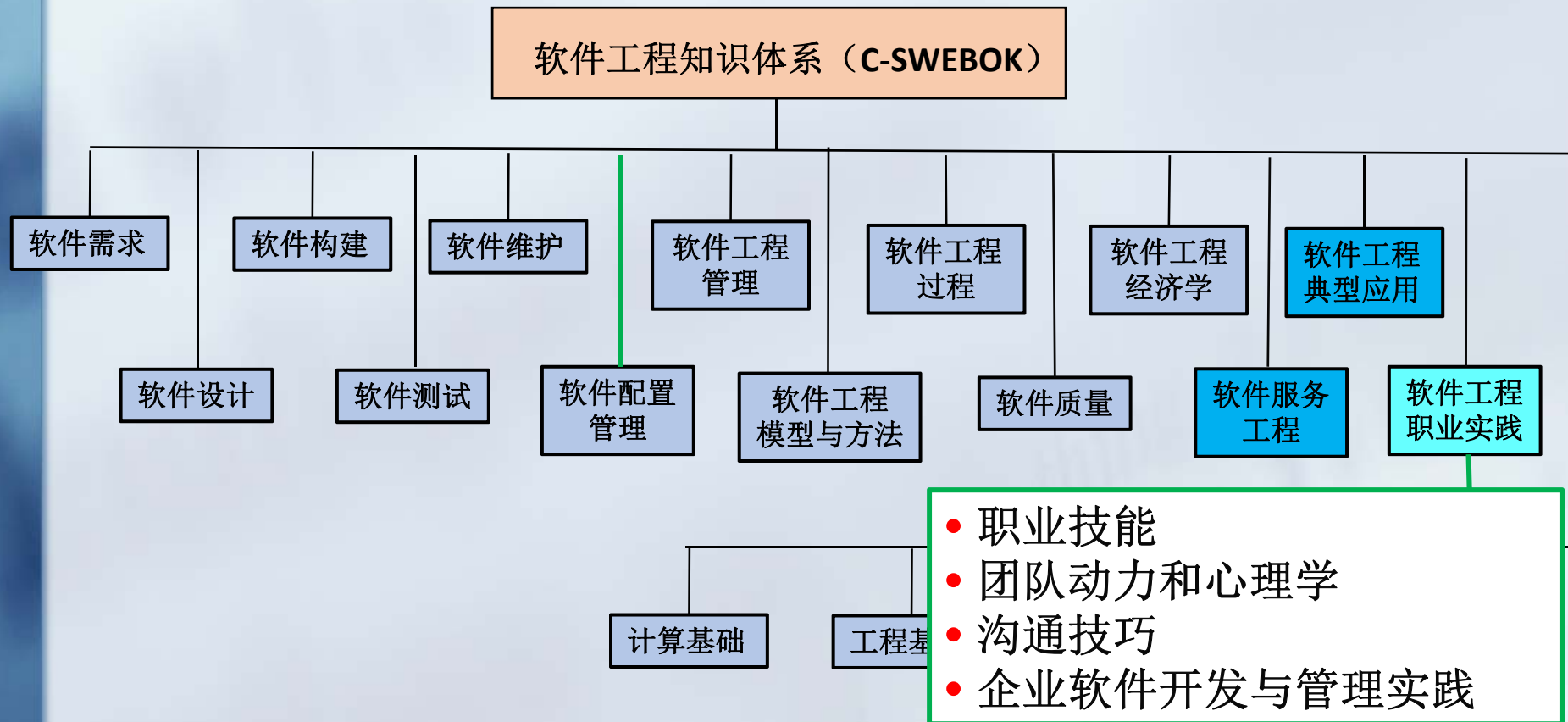
软件工程专业知识体系

中国软件工程教指委：C-SWEBOK知识体系



软件工程专业的知识体系

中国软件工程教指委：C-SWEBOK知识体系



小 结

第一讲 软件之启蒙 使您 ——

- 认识计算机、软件工程及基本概念
- 了解计算机发展历程及现状
- 了解软件工程发展历程与趋势
- 初识软件工程科学技术体系与学科范畴
- 了解软件工程的知识体系

视频位置

关于软件工程的更多知识与故事 ——

请您继续学习后续10讲

软件工程专业导论

Introduction to Software
Engineering Program

第一讲 软件之启蒙 —
初识软件工程专业

视频位置

谢谢!

徐晓飞

哈尔滨工业大学

