



信息与软件工程学院

School of Information and Software Engineering

# 高级软件工程

## 第二章 软件工程师的专业技能

姓名 | 许毅

[xuyi0421@uestc.edu.cn](mailto:xuyi0421@uestc.edu.cn)

2024/2/21



## 第二章目录

**2.1 软件需求工程**

**2.2 软件设计工程**

**2.3 软件项目管理**

# 内容

## ❑ 软件项目及其特点

- ✓ 软件项目及其管理的特殊性

## ❑ 软件项目管理概述

- ✓ 软件项目管理的对象和要素

## ❑ 3. 软件项目管理的方法

- ✓ 软件项目管理的常见方法

## ❑ 4. 软件项目管理的方法和标准

- ✓ 国内外标准、ISO和CMM系列



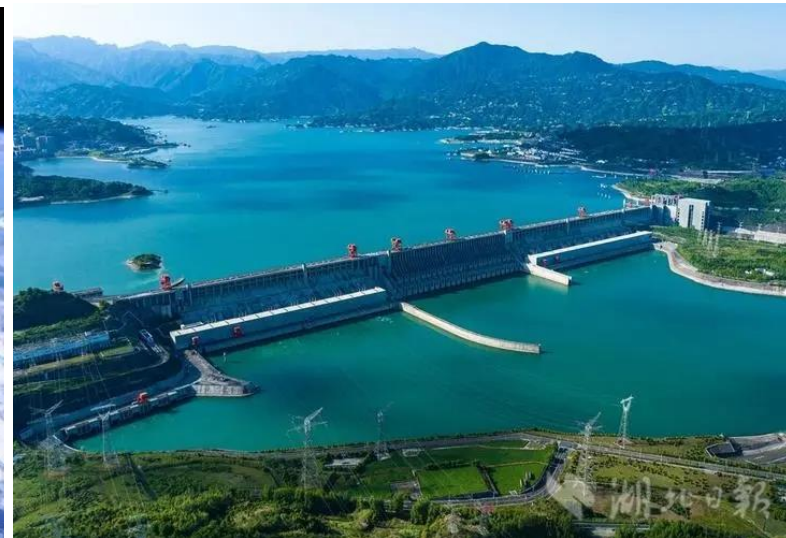
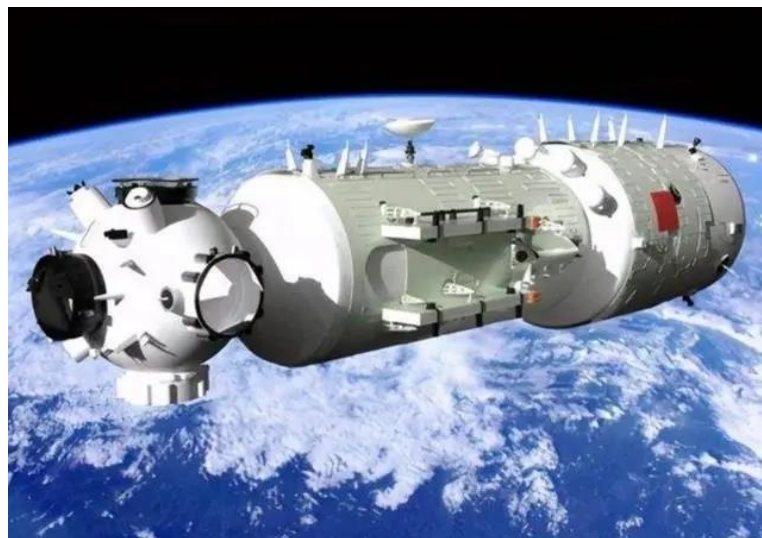
# 1.1 何为项目 (Project)?

## ❑ 项目概念

- ✓ 项目是指为**创建**一个**唯一**的**产品**或者提供唯一的**服务**而进行的**努力**
- ✓ 项目是基于既定**资源与约束**，为实现既定**目标**而实施的**活动**，它是一份临时工作，目的是创造独特产品、服务或者结果

## ❑ 典型项目示例

- ✓ 阿波罗登月项目
- ✓ Windows 7开发项目
- ✓ 三峡水利项目
- ✓ 载人飞船项目
- ✓ .....



**项目也称工程，存在约束，实施活动，提供产品和服务**

# 1.2 项目特点

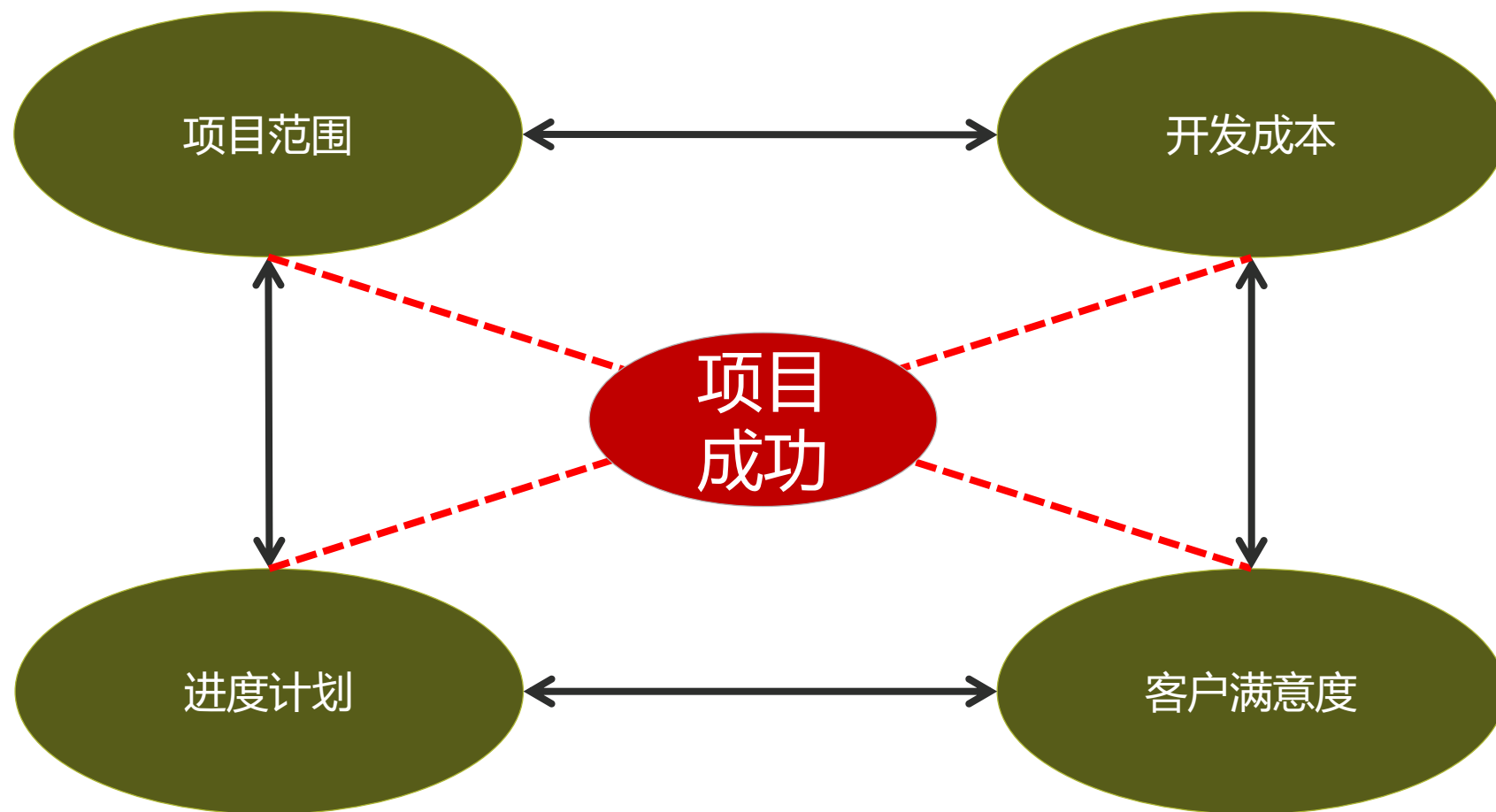
- ❑ **目标性:** 获得预期的结果
- ❑ **进度性:** 在限定期间完成
- ❑ **约束性:** 具有有限的资源（如人员、经费、工具等）
- ❑ **多方性:** 涉及多个不同人与组织
- ❑ **独立性:** 项目间无重复性
- ❑ **不确定性:** 项目的实施及其结果不确定性

中国探月工程项目如何体现项目的特点



软件开发项目如何体现项目的特点

# 影响项目成功的因素



项目的开展和实施受多要素影响，其结果具有不确定性



# 思考和讨论

- 列举你所知道的项目，说明项目的目标性、约束性、多方性、独立性、不确定性等特征



# 1.3 软件项目及其特点

## ❑ 何为软件项目

- ✓ 针对**软件**这一特定产品和服务的项目努力开展 “**软件开发活动**”

## ❑ 软件项目的特点

- ✓ **对象**：作为逻辑产品的软件
- ✓ **过程**：不以制造为主，没有重复生产过程
- ✓ **属性**：实施要素难以度量和估算，如成本、进度、质量
- ✓ **复杂性**：作为逻辑产品的复杂性非常高
- ✓ **易变性**：软件需求通常难以确定且经常变化



# 示例：软件项目

- ❑ 火炮的火控软件
- ❑ 一体化指挥信息系统
- ❑ 卫星图像处理软件
- ❑ 导弹飞控软件
- ❑ 机载软件系统
- ❑ 微信软件
- ❑ 12306软件
- ❑ .....



软件项目有其特点：对象、属性、过程、复杂性、易变性等

# 1.4 军用软件项目的特点

## ❑ 软件形态

- ✓ 往往与特定的物理或者硬件环境相互关联，**人机物融合系统**

## ❑ 软件需求

- ✓ 环境的恶劣性和多变性，以及军用软件对**质量**提出更高要求，需要更高的**可靠性、安全性、灵活性**等
- ✓ 由于需要对事件作出及时响应，军用软件项目对系统的**实时性**提出更高的要求

## ❑ 软件复杂性

- ✓ 随着信息化的开展，军用软件项目的**规模越来越大**，甚至可能是系统之系统，或者超大规模系统

# 内容

## 1. 软件项目及其特点

- ✓ 软件项目管理的特殊性

## 2. 软件项目管理概述

- ✓ 软件项目管理的对象和要素

## 3. 软件项目管理的方法

- ✓ 软件项目管理的常见方法和已有标准

## 4. 软件项目的标准

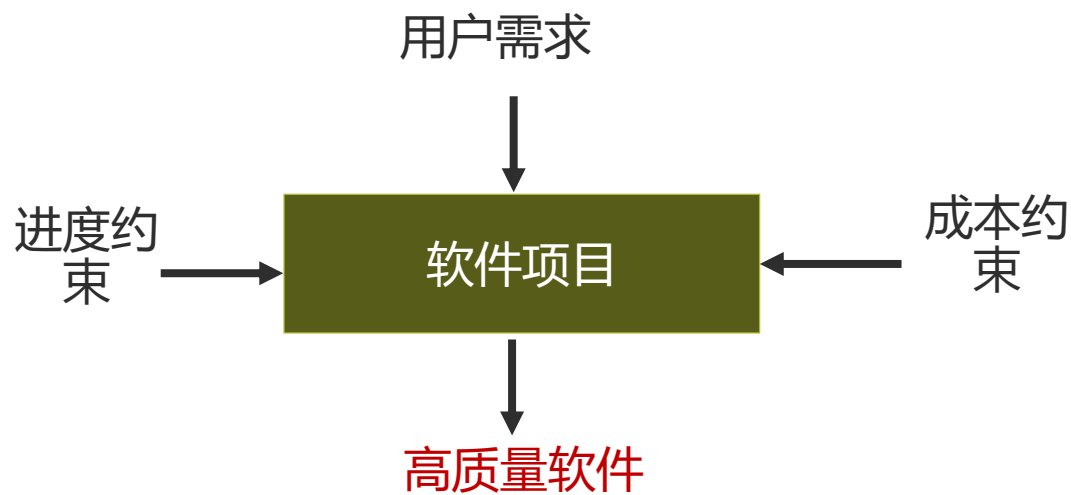
- ✓ 国内外标准、ISO和CMM系列



# 2.1 软件项目的任务

□ 按照预定的**进度、成本和质量**，开发出**满足用户要求**的软件产品

- ✓ 用户需求
- ✓ 确保质量
- ✓ 成本限制
- ✓ 进度限制



- ✓ 目标性
- ✓ 进度性
- ✓ 约束性
- ✓ 多方性
- ✓ 独立性
- ✓ 不确定性

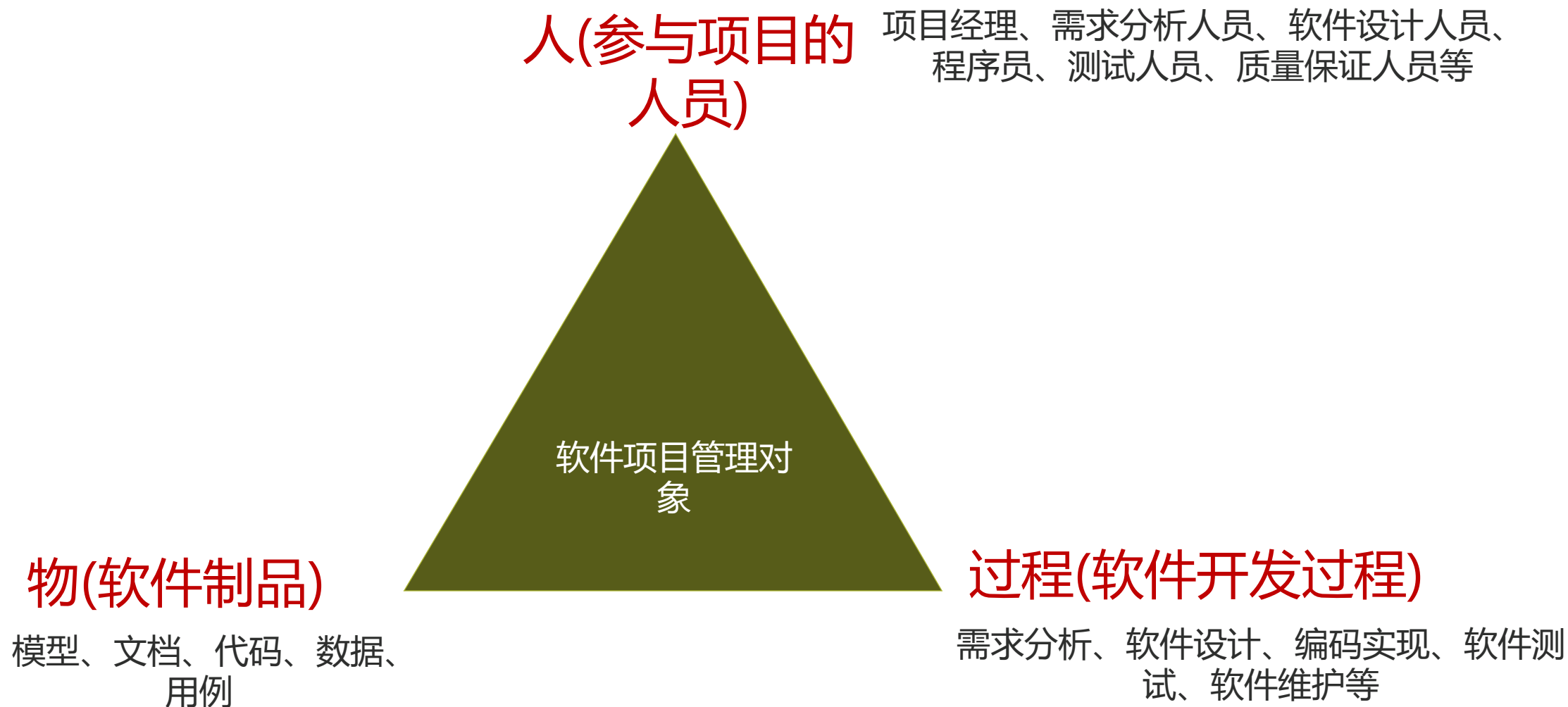
# 软件项目实施方法 – 工程化

- ❑ **按照过程** – 如瀑布模型
- ❑ **遵循原则** – 如设计原则
- ❑ **注重质量** – 如正确性
- ❑ **借助工具** – 如质量分析
- ❑ **适时评审** – 如需求评审
- ❑ **开展测试** – 如单元测试
- ❑ **撰写文档** – 如需求文档
- ❑ **遵照规范** – 如军方规范
- ❑ .....

用软件工程方法来实施软件项目


- 军用软件工程过程
- 军用软件产品规范
- 军用软件质量保证方法
- .....

## 2.2 软件项目涉及的对象



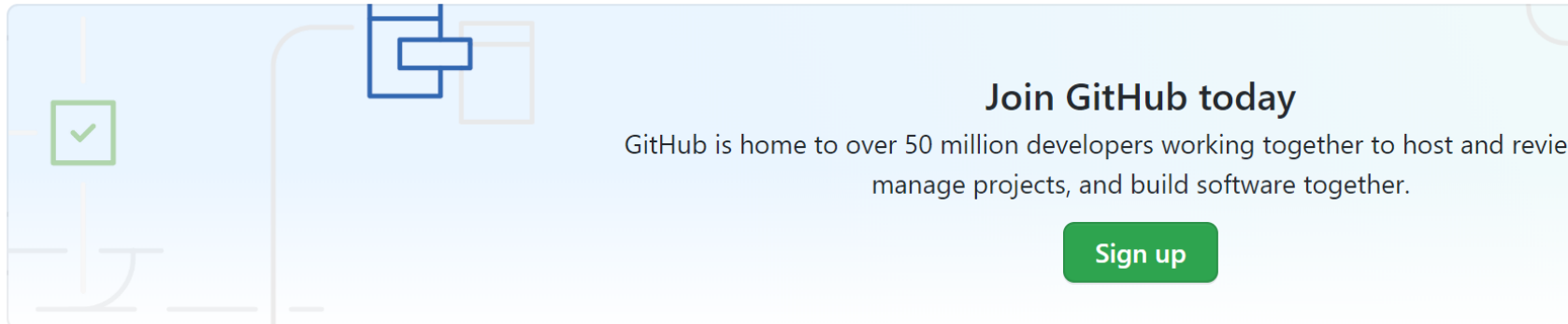
软件项目要管理好“人、物和过程”三类对象

# 示例：Hadoop开源软件项目

 [apache / hadoop](#)

[Watch](#) 1k [Star](#) 11.1k [Fork](#) 6.9k

[Code](#) [Pull requests](#) 345 [Actions](#) [Projects](#) [Security](#) [Insights](#)







**Join GitHub today**

GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

[Sign up](#)

[trunk](#) [282 branches](#) [341 tags](#) [Go to file](#) [Code](#)

 **bilaharith** HADOOP-17191. ABFS: Run the tests with various combinations of con... 4c033ba 9 hours ago 24,455 commits

 .github	HADOOP-15184. Add GitHub pull request template. (#1419)	15 months ago
 dev-support	HADOOP-17352. Update PATCH_NAMING_RULE in the personality file. (...)	last month
 hadoop-assemblies	HDFS-15346. FedBalance tool implementation. Contributed by Jinglun.	6 months ago

## About

Apache Hadoop

[hadoop.apache.org/](#)

[hadoop](#)

[Readme](#)

- ✓ 成百上千人参与
- ✓ 采用Git来管理代码和文档制品
- ✓ 采用分支和群体化的过程和方法

管理好复杂和不断变化的软件制品并确保其质量非常重要！

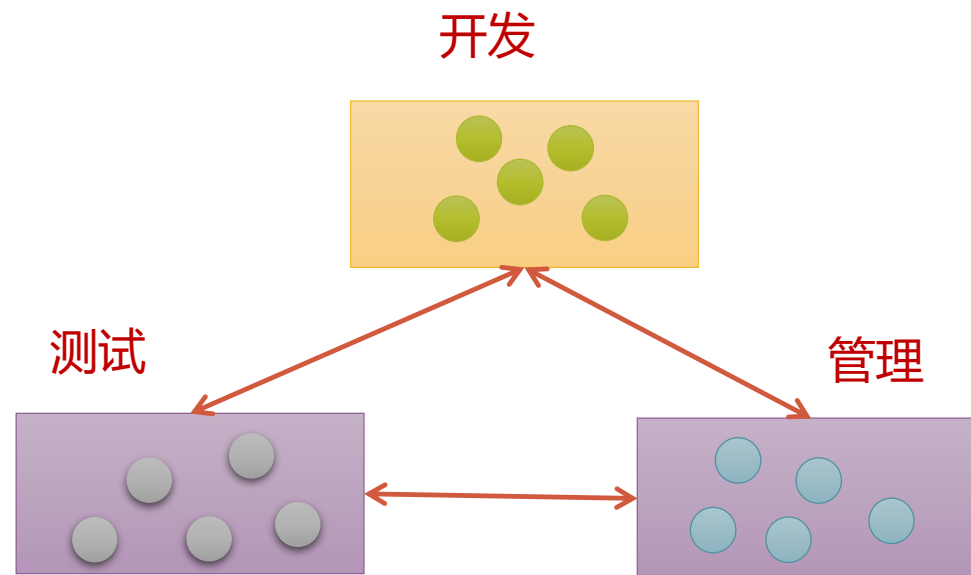


# 示例：软件项目的人员及团队

## ❑ Windows 7项目

- ✓ 25个功能**团队**
- ✓ 每个团队由40名开发人员，40名测试人员，20名项目经理组成
- ✓ 整个项目总体约500个项目经理，1000名开发人员，1000名测试人员

**加强人员之间的交流和合作  
非常重要！**



## 2.3 何为软件项目管理?

- 对软件项目所涉及的过程、人员、产品、成本和进度等要素进行度量、分析、规划、组织和控制的过程，以确保软件项目按照预定的成本、进度、质量要求顺利完成



# 软件项目管理的对象

## ❑ 过程管理

✓ 过程：怎么做(How)

## ❑ 人员管理

✓ 人员：谁来做(Who)

## ❑ 产品管理

✓ 产品：结果(What)

软件项目管理要管好三类对象

- 过程
- 人员
- 产品

## 2.4 软件项目的管理要素

### ❑ 管理软件过程

- ✓ 明确软件开发活动及过程：**过程定义**
- ✓ 估算软件项目工作量成本：**软件度量**
- ✓ **制定计划、跟踪过程、风险控制等**

### ❑ 管理软件产品

- ✓ 明确有哪些产品，呈什么形式(规范文档)
- ✓ **质量保证、配置管理、需求管理，风险控制**

### ❑ 管理项目人员

- ✓ 组建开发团队、调动积极性和激情
- ✓ **团队建设与沟通、机制设计、风险控制**



# 软件项目管理要素

管理对象	人员	过程	产品
管理内容	<ul style="list-style-type: none"><li>➤ 团队建设</li><li>➤ 团队纪律</li><li>➤ 激励机制</li></ul>	<ul style="list-style-type: none"><li>➤ 过程定义</li><li>➤ 软件度量</li><li>➤ 项目计划</li><li>➤ 项目跟踪</li><li>➤ 风险管理</li></ul>	<ul style="list-style-type: none"><li>➤ 软件质量管理</li><li>➤ 软件配置管理</li><li>➤ 软件需求管理</li></ul>

# 管理在软件项目中的重要性

- ❑ 70%的软件项目由于**管理不善**导致难以控制进度、成本和质量，三分之一左右的软件项目在时间和成本上超出额定限度125%以上
- ❑ **管理**是影响软件项目成功实施的**全局性因素**，而**技术**仅仅是**局部因素**
- ❑ 如果软件开发组织不能对软件项目进行有效管理，就难以充分发挥软件开发方法和工具的**潜力**，也无法高效率地开发出高质量的软件产品
- ❑ 历史上由于管理不善而导致软件项目失败的例子**比比皆是**，如美国国税局税收现代化系统、美国银行的MasterNet系统等

# 2.5 软件度量、测量和估算

## ❑ 任务

- ✓ 对软件项目的**过程、产品、资源的属性**的**定量描述**，如软件项目的规模、成本、工作量、质量等
- ✓ 目的是为了对软件项目进行**更好管理**，如制定计划、质量保证等

## ❑ 问题

- ✓ 需要对哪些方面进行度量？
- ✓ 如何进行度量？
- ✓ 如何将度量的结果指导软件项目的管理？
- ✓ 有哪些工具和模型可辅助进行软件度量.....

软件项目管理需  
定量的数据支持



# 软件开发项目需要定量数据

## ❑ 签订合同之时

- ✓ 客户和开发商都关心**项目规模和合同额**

## ❑ 项目策划之时

- ✓ 项目组关心**开发时间、投入成本和人力**

## ❑ 开发过程之中

- ✓ 项目组关注**开发进度**，用户和开发人员关注**产品质量**

## ❑ 开发完成之后

- ✓ 项目**实际投入和开销**，项目的**利润**



# 软件度量

## ❑ 软件度量(Metrics)是指对软件产品、软件开发过程或者资源的简单属性的定量描述

- ✓ **产品**：软件开发过程中所生成的各种文档和程序
- ✓ **过程**：与软件开发有关的各种活动，如软件设计等
- ✓ **资源**：软件开发过程中所需支持，如人员、费用等

## ❑ 注意点

- ✓ **简单属性**：无需参照其它属性便可直接获得定量描述
- ✓ **定量描述**而不是定性描述

## ❑ 简单属性示例

- ✓ 例子：\*\*软件系统的代码行数目为1132 *KLOC*

# 软件测量

## ❑ 软件测量(Measure)

- ✓ 对软件产品、软件开发过程和资源**复杂属性**的**定量描述**，它是简单属性度量值函数，用于**事后或实时状态**，如软件可靠性

## ❑ 注意点

- ✓ **复杂属性**-不可直接获得、需参照其它属性的度量值
- ✓ **实时或者事后状态**，用于对历史进行评估
- ✓ **定量描述**而不是定性描述

## ❑ 复杂属性示例

- ✓ 软件质量、软件可靠性等等，如软件可靠性为0.95

# 估算

## ❑ 估算(Estimation)

- ✓ 对软件产品、软件开发过程和资源**复杂属性**的**定量描述**，它是简单属性度量值的函数，软件估算**用于事前**

## ❑ 注意点

- ✓ **复杂属性**，不可直接获得、需参照其它属性的度量值
- ✓ **事前状态**，可采用经验公式
- ✓ **定量描述**，而不是定性描述

## ❑ 示例

- ✓ 如在招投标时（开发之前）估算某个软件的开发成本为125万元

# 软件度量的对象

	产品	过程	资源	关注对象	难易程度
<b>内部属性</b>	(1)代码长度 (2)程序功能 (3)重用性 (4)模块的耦合和内部聚度 .....	(1)工作量 (2)计划和进度 ....	(1)人 (2)软硬件环境 (3)方法 (4)经验	软件开发人员和项目管理人员	相对比较容易
<b>外部属性</b>	(1)可靠性 (2)可理解性 (3)质量 (4)可维护性 (5)可移植性	(1)成本 (2)可控制性	(1)成本 (2)时间	用户和软件项目管理人员	相对比较困难，由内部属性决定

# 示例：面向规模的软件度量

## ❑ 用软件代码行数来表示软件项目规模

✓ **生产率**  $PM = L / E$

L表示代码总量(单位：KLOC)，E表示软件工作量(单位：人月)

✓ 每千行代码的**平均成本**  $CKL = S / L$

S为软件项目总开销

✓ **文档与代码比**  $DI = Pd / L$

Pd表示文档页数

✓ **代码出错率**  $EQRI = Ne / L$

Ne表示代码出错的数目

软件代码行越多，意味着软件的规模就越大

# 示例：面向规模的软件度量

## ❑ 基于经验模型的估算

- ✓ **构造性成本模型**CoCoMo (Constructive Cost Model)
- ✓ 基本CoCoMo模型： $E = a * (kLOC)^b$ 。其中E是软件系统的工作量(单位：人月)，a和b是经验常数
- ✓  $D = c * E^d$ 。其中D是开发时间（单位:月），c和d是经验常数

软件类型	a	b	c	d	适用范围
组织型	2.4	1.05	2.5	0.38	各类应用程序
半独立型	3.0	1.12	2.5	0.35	各类实用程序、编译程序等
嵌入型	3.6	1.20	2.5	0.32	各类实时软件、OS、控制程序等



# 内容

## 1. 软件项目及其特点

- ✓ 软件项目及其管理的特殊性

## 2. 软件项目管理概述

- ✓ 软件项目管理的对象和要素

## 3. 软件项目管理的方法

- ✓ 软件过程管理：软件项目计划、软件项目跟踪

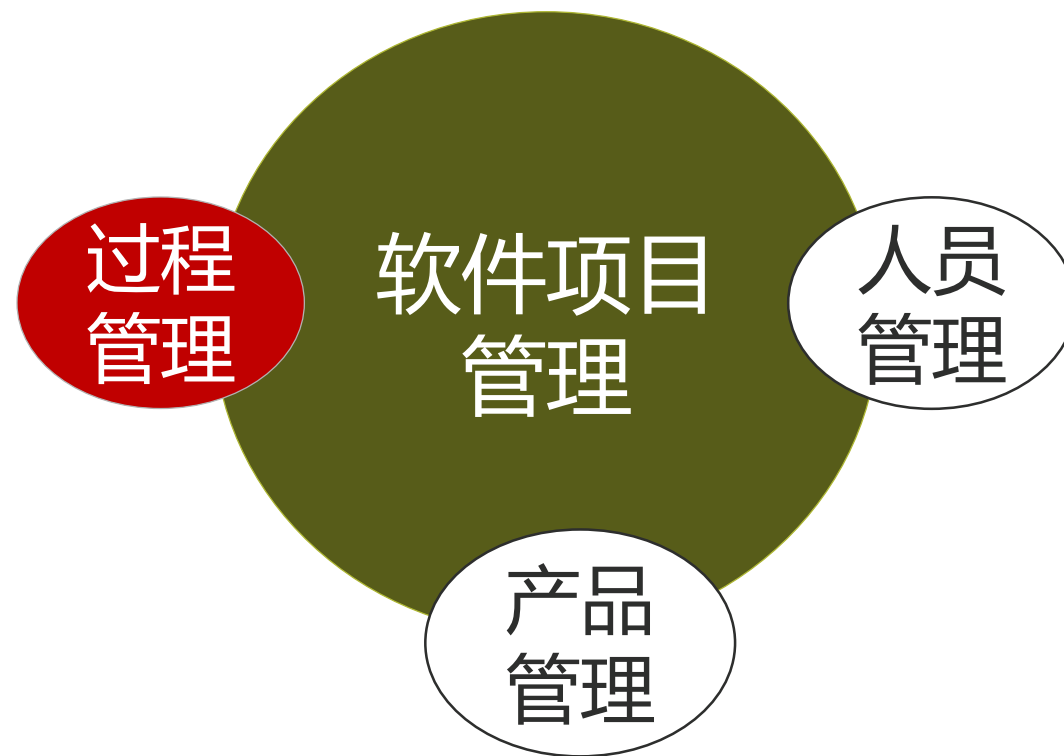
## 4. 软件项目的标准

- ✓ 国内外标准、ISO和CMM系列



# 3.1 过程管理

- ❑ 过程定义
- ❑ 软件度量
- ❑ 项目计划
- ❑ 项目跟踪
- ❑ 风险管理



# 1. 过程定义

## ❑ 任务

- ✓ 定义、文档化软件开发过程、明确软件开发活动，得到一个**良定义、全面、灵活、简洁和可供剪裁的软件开发过程**

## ❑ 问题

- ✓ 如何根据要求定义软件开发过程？
- ✓ 如何对它文档化软件过程？
- ✓ 如何确保软件开发过程的有效性(包含必须的活动)、简洁性(舍弃不必要的过程和活动)和灵活性(根据具体项目情况进行剪裁)？
- ✓ 如何根据项目的特点剪裁过程？ .....

**定义和明确过程是软件开发的前提**

# 示例：根据软件项目的特点来定义过程

## ❑ 明确软件需求、严格质量管控

- ✓ 采用以文档为中心的重型过程
- ✓ 如瀑布模型、迭代和增量模型、螺旋模型等
- ✓ 大部分军用软件项目采用该方法

## ❑ 变化和未确定的软件需求、需要快速应对变化

- ✓ 采用敏捷方法
- ✓ 如互联网软件、企业信息系统等

# 2. 软件项目计划

## ❑ 任务

- ✓ **制定和文档化软件项目计划**，确保软件开发计划是可行、科学、符合实际的

## ❑ 问题

- ✓ 要对软件开发过程中的哪些方面制定计划？
- ✓ 制定软件项目的计划的基础和依据是什么？
- ✓ 要考虑哪些方面的问题？
- ✓ 如何确保计划是科学的和可行的？(软件度量)
- ✓ 如何描述计划？利用哪些工具可辅助计划的制定？ .....

**软件项目计划有助于项目有序地开展和实施**

# 软件开发需要制定和实施计划

## ❑ 合同和策划阶段

- ✓ 制定软件开发计划
- ✓ 软件项目计划分发到各个开发成员
- ✓ 每个成员准确了解各自任务和工作以及实施进度要求

## ❑ 项目实施过程

- ✓ 根据软件开发计划有序实施项目
- ✓ 发现项目实施中存在的问题：如进度滞后
- ✓ 出现偏差后调整软件项目计划

## ❑ 项目完成之后

- ✓ 总结项目计划实施情况
- ✓ 分析原因，指导后续其他项目的计划制定

计划有助于项目的有序实施

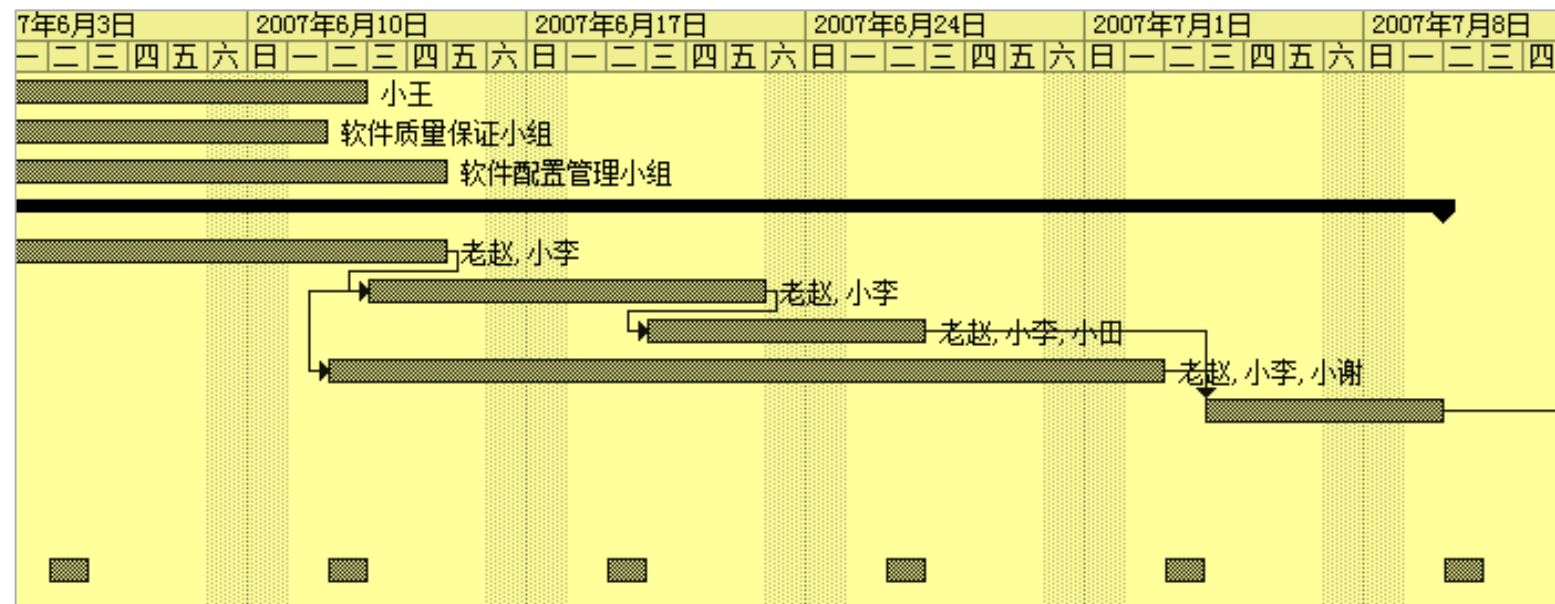
# 什么是软件项目计划?

## ❑ 软件项目计划

- ✓ 对软件项目实施所涉及的活动、人员的安排、任务的划分、开发进度、资源的分配和使用等方面作出的**预先规划**

## ❑ 计划内容

- ✓ 软件开发**进度计划**
- ✓ 人员任务**分工计划**
- ✓ 资源分配**使用计划**
- ✓ .....



软件项目计划对项目实施进度、人员组织、资源使用等进行规划



# 制定软件项目计划的基础和依据

## ❑ 软件开发过程

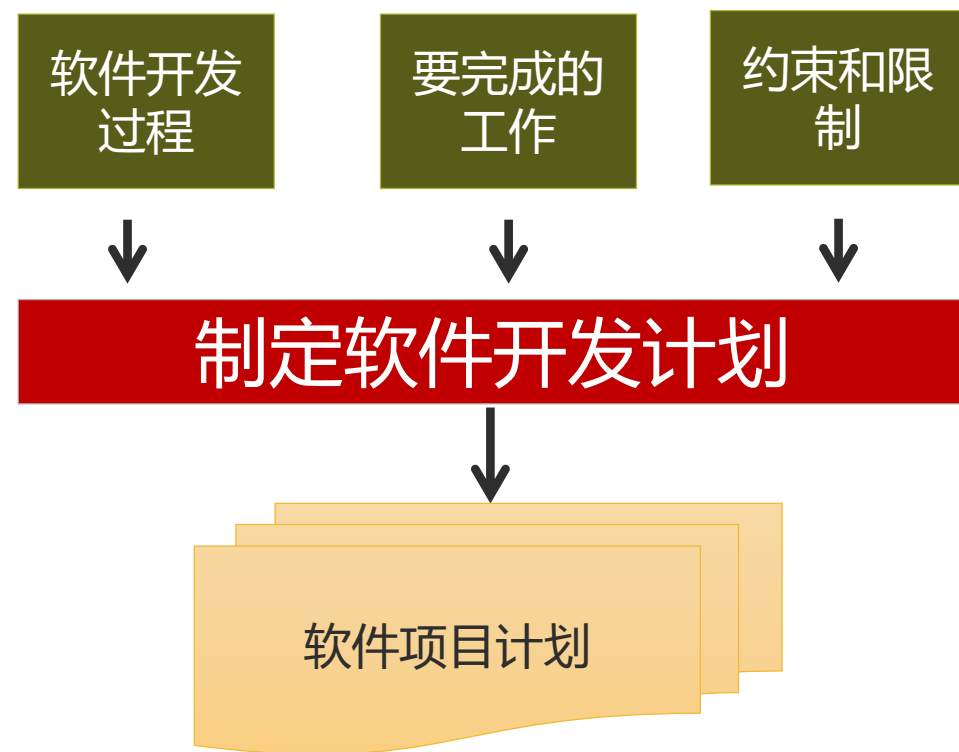
- ✓ 选择什么样的软件开发过程
- ✓ 支持敏捷方法过程：迭代模型

## ❑ 要完成的工作

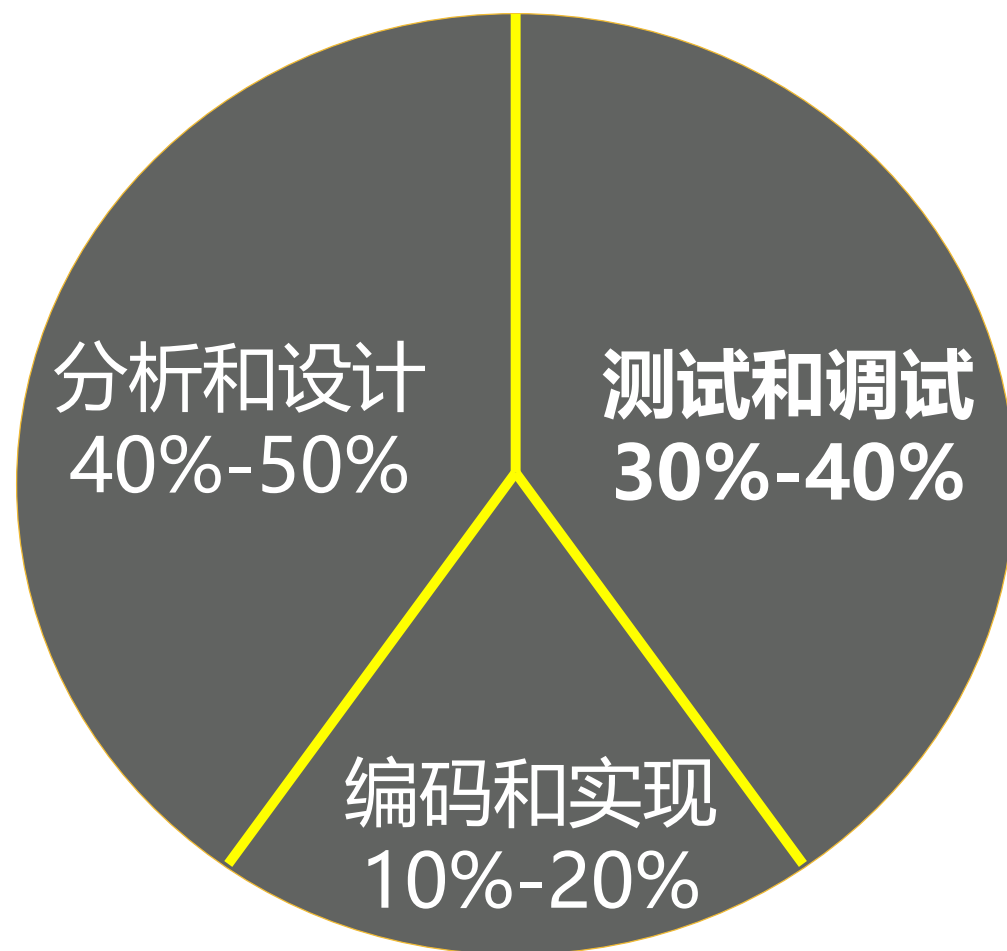
- ✓ 工作说明和软件需求
- ✓ 历史数据和估算模型
- ✓ 估算工作量和成本

## ❑ 项目约束限制

- ✓ 项目投入资源及其质量
- ✓ 资源（经费、时间、人员、...）



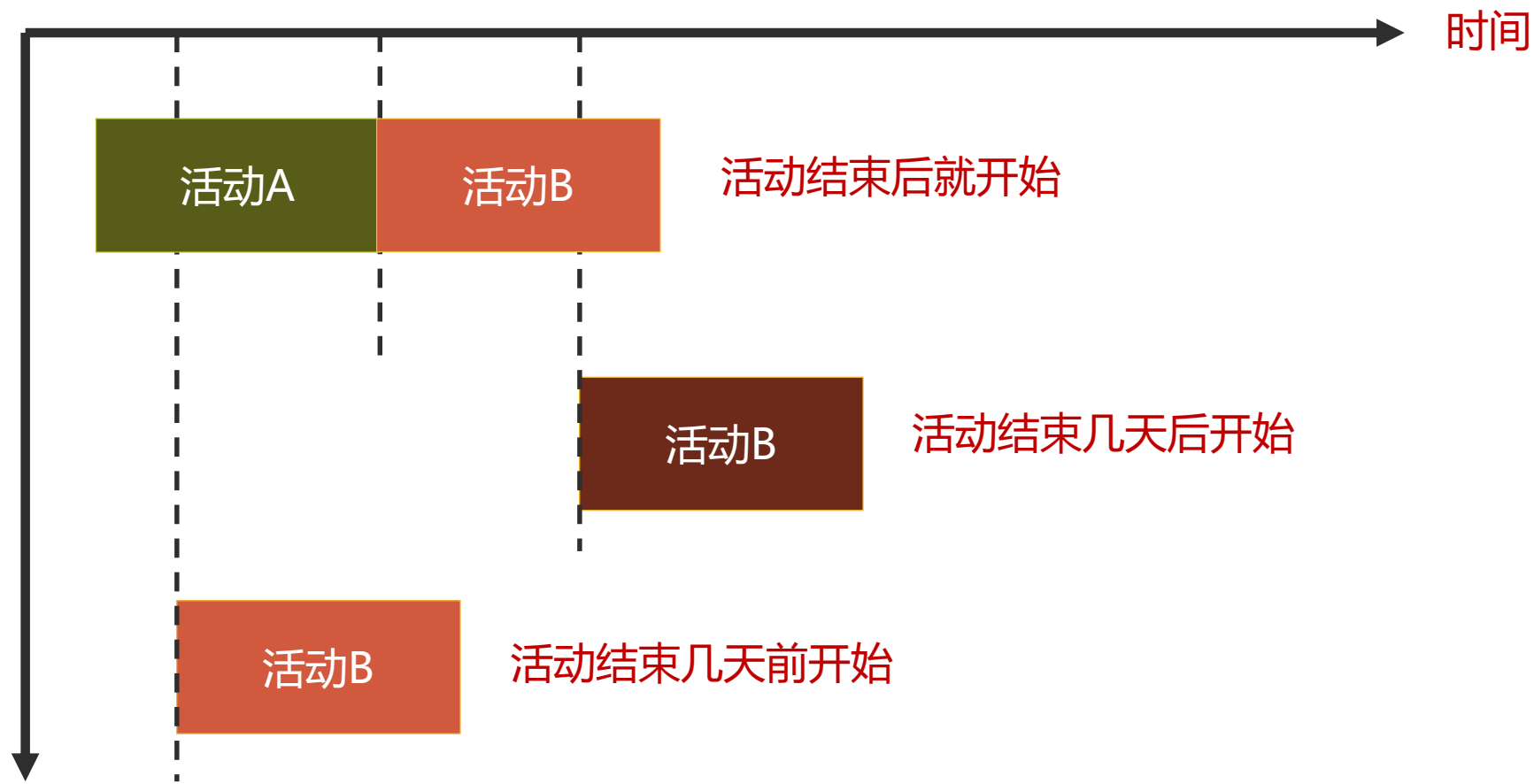
# 软件开发工作量的大致分布



软件开发的大部分工作量用于分析和设计、测试和调试

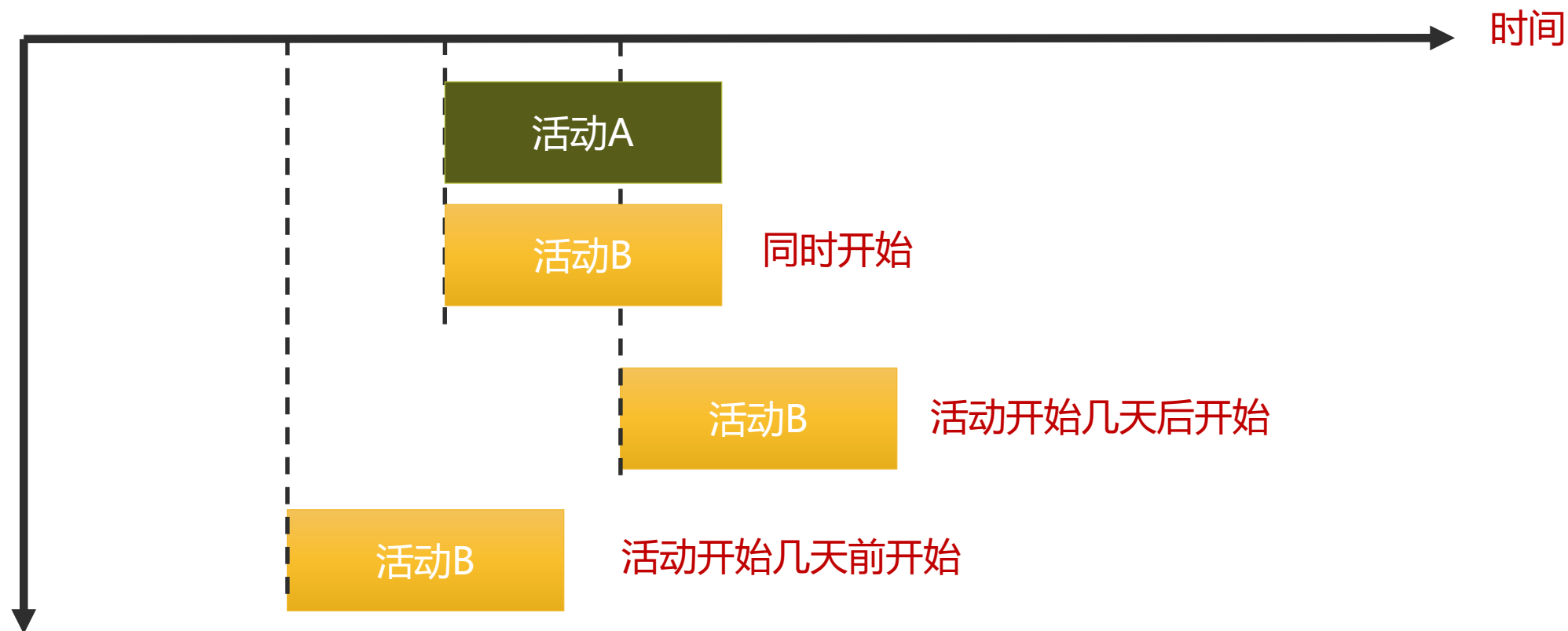
# 软件开发活动之间的关系 (1/3)

❑ 结束到开始：一个活动结束，随后另一项活动开始



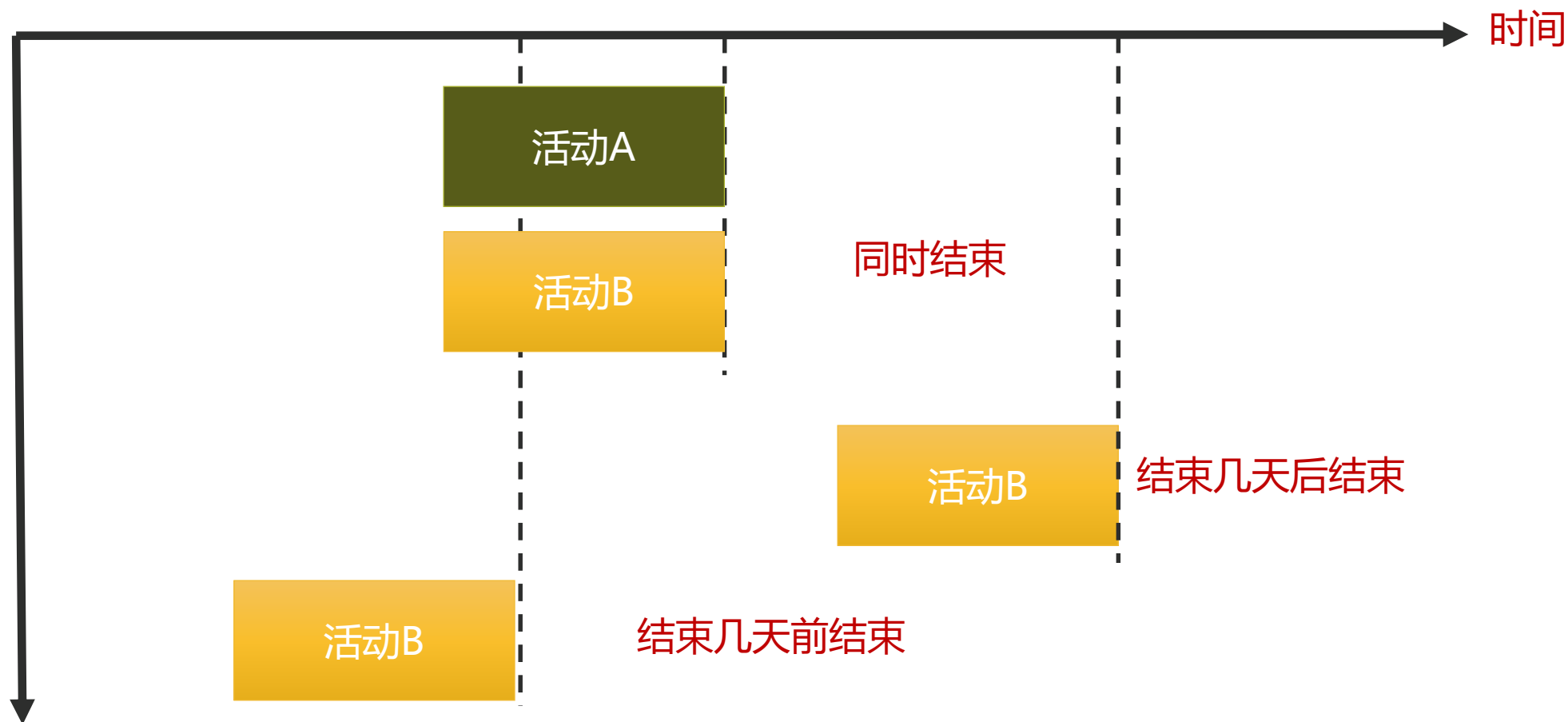
# 软件开发活动之间的关系 (2/3)

## 开始到开始：一项活动开始，预示着另一项活动开始

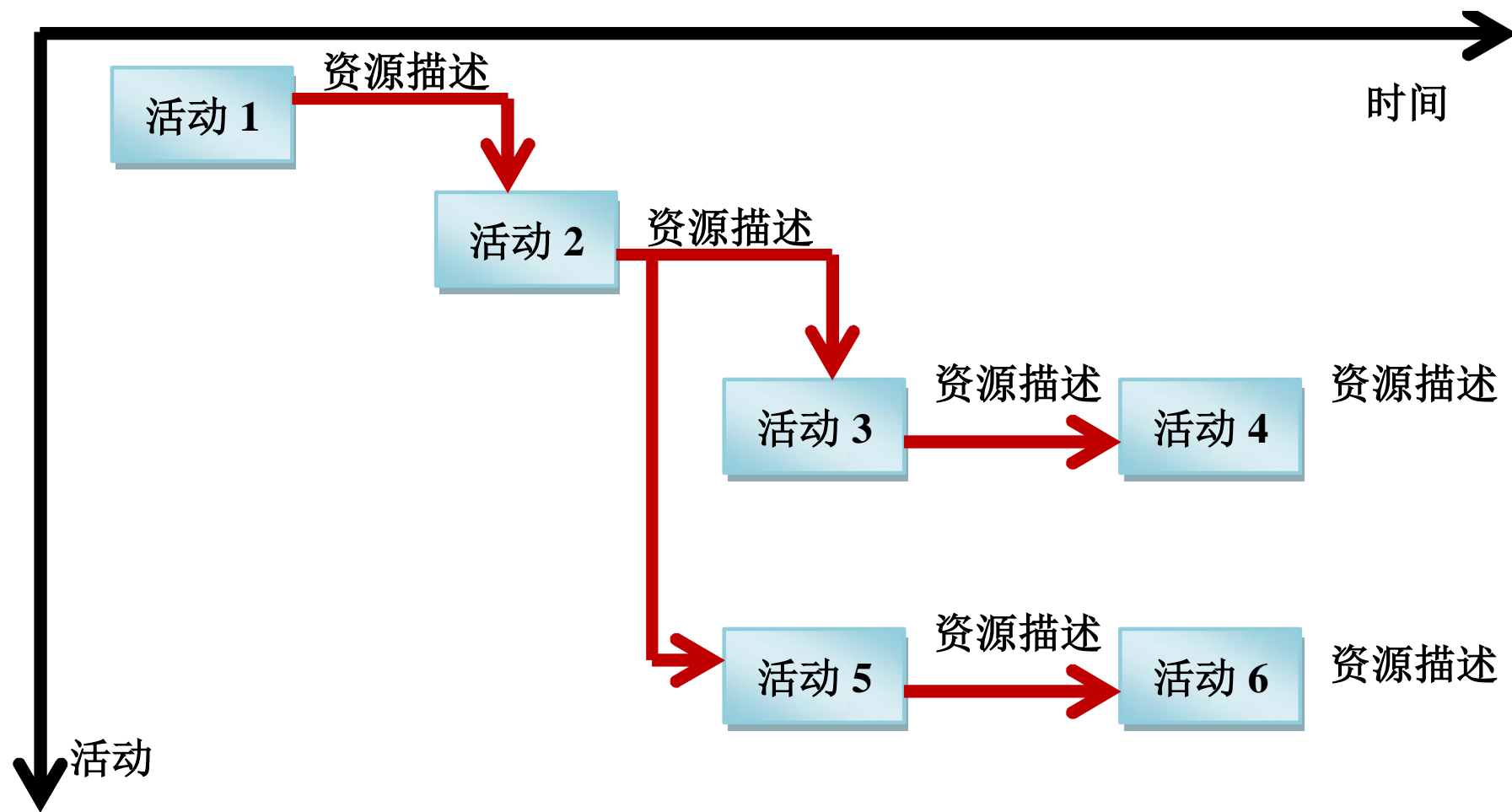


# 软件开发活动之间的关系 (3/3)

❑ 结束到结束：一项活动结束，预示着另一项活动结束



# 基于甘特图表示的软件项目计划

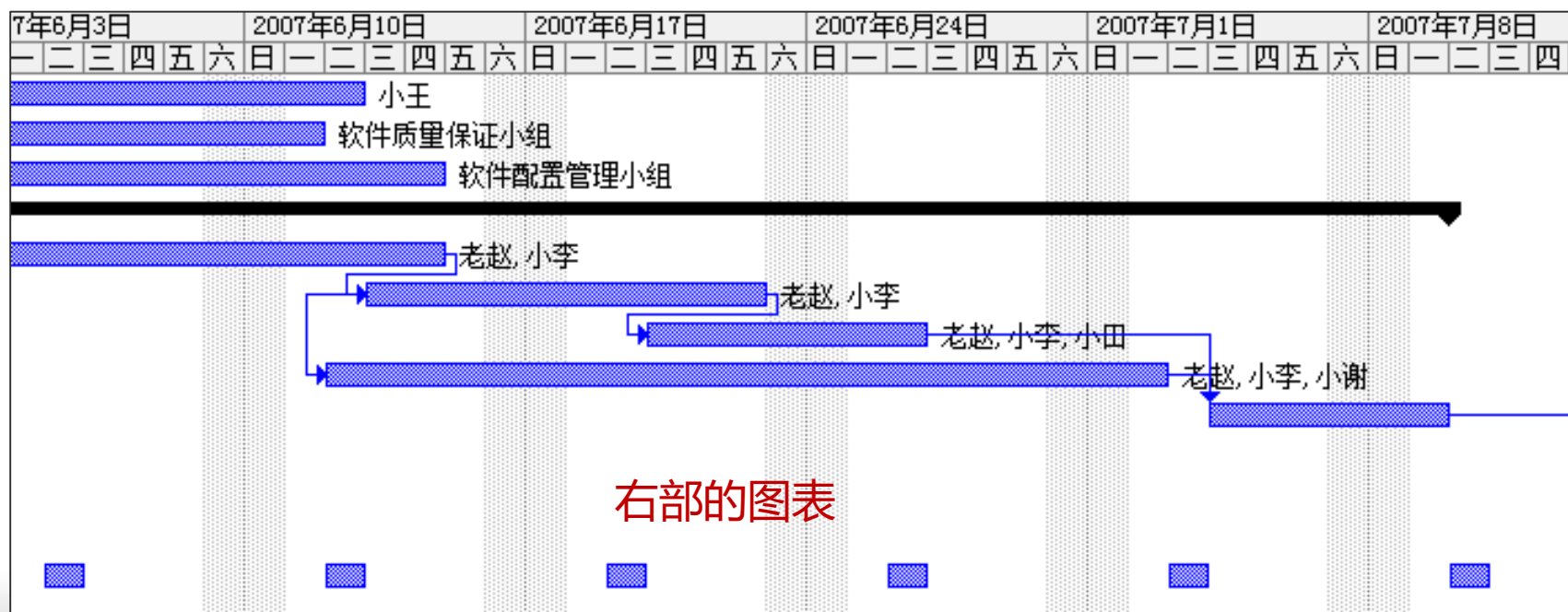


- 开发活动如何随时间而推进
- 活动之间的依赖关系

# 描述项目进度计划-甘特图

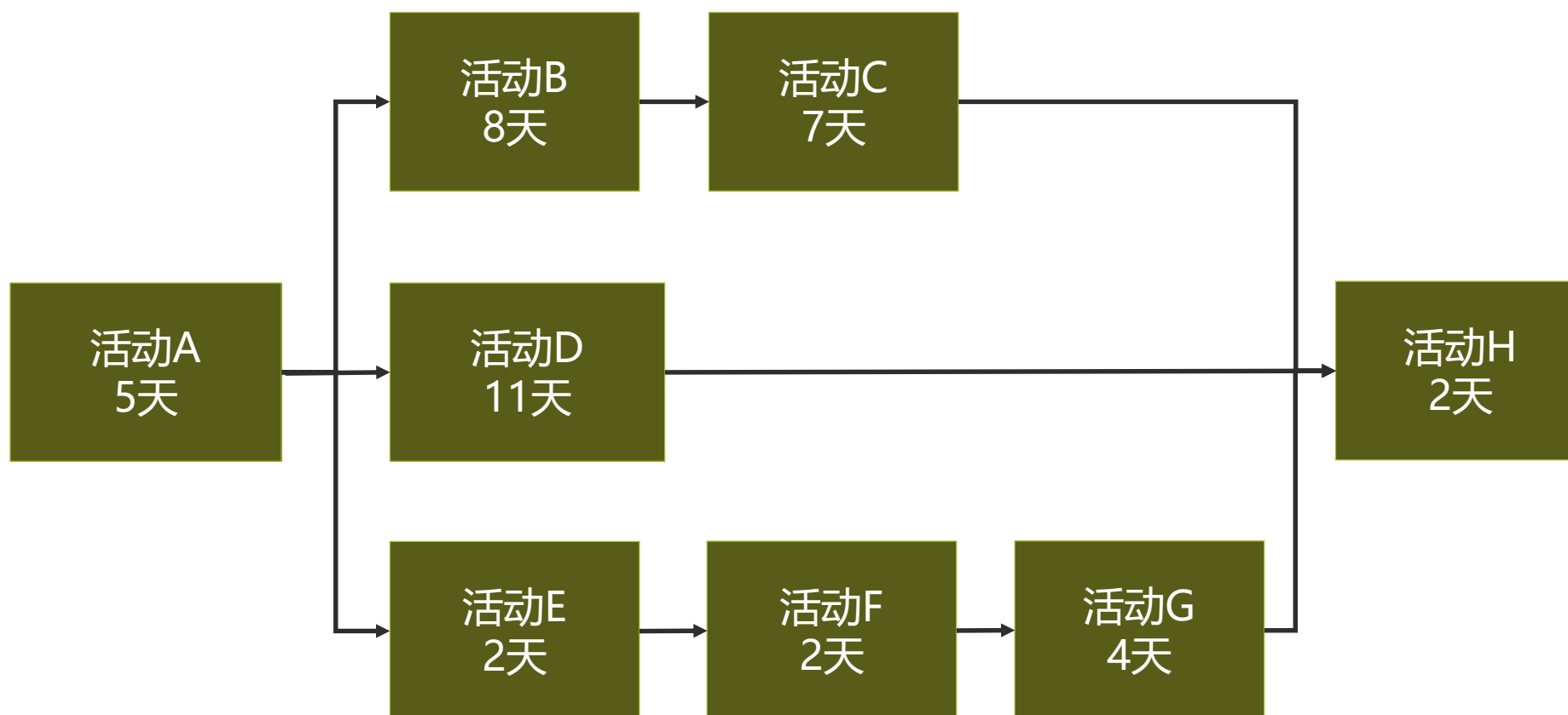
## 甘特图

- ✓ 左部工作表，以**文字方式**显示任务信息，如任务名称，开始和结束日期等
- ✓ 右边的图表，以**条形图**方式显示任务信息



# 关键路径分析

- **关键路径**是指软件项目进度计划中从起始活动开始到结束活动为止，具有**最长长度**的路径。  
**长度**是指软件开发时间





# 活动责任矩阵 (1/2)

- ❑ 用于定义与软件开发活动执行、评审和批准相关的人员和角色
- ❑ 软件开发活动 - 角色责任矩阵表

软件开发活动 \\角色	执行	负责	评审	批准
需求分析	需求分析小组	需求分析小组组长	用户方代表 需求分析小组 软件设计小组 质量保证小组 软件测试小组	软件项目负责人 用户方负责人

# 活动责任矩阵 (2/2)

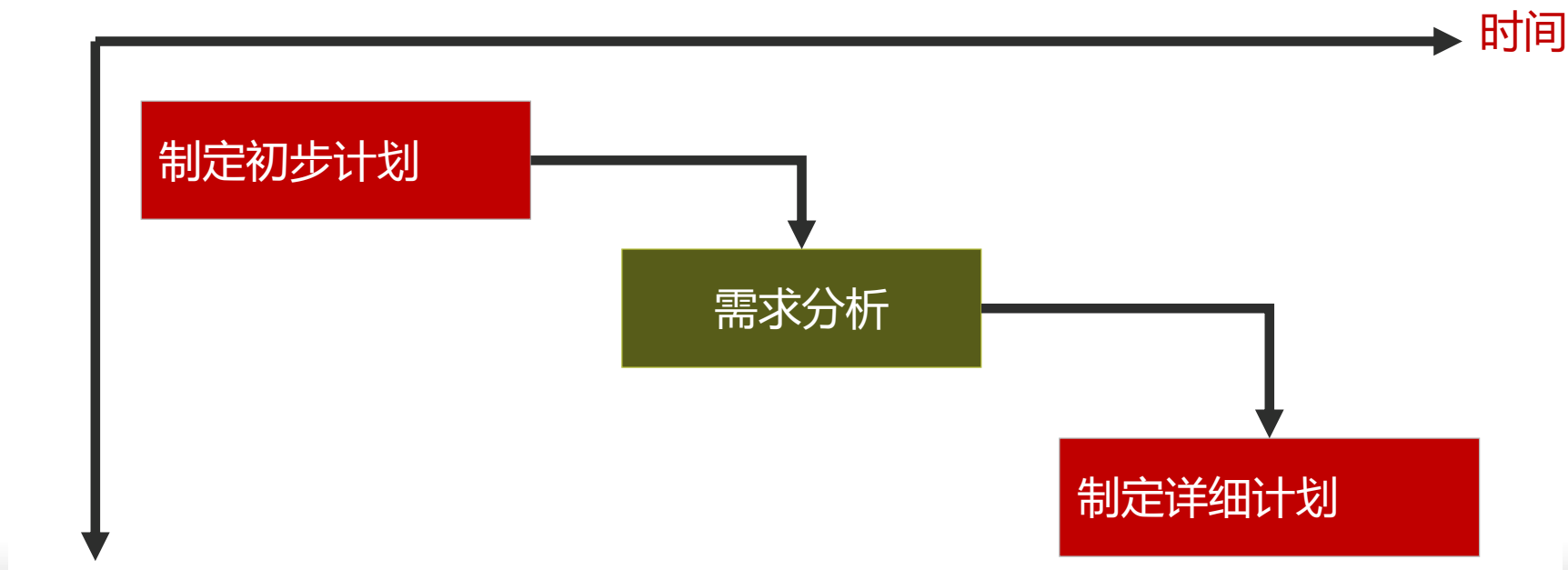
❑ 角色 - 人员责任矩阵表

角色	人员
需求分析小组	小张、小李、小王
需求分析负责人	小张
软件项目负责人	小宋
用户方代表	小张
用户方负责人	小董

# 制定软件项目计划的时机

## ❑ 软件项目计划一般是在软件项目**实施之初**制定

- ✓ **项目开始之初**，制定一个初步的软件项目计划，用于指导后续短期的软件开发工作，如需求分析工作
- ✓ **软件需求分析完成之时**，制定详细的软件项目计划，用于指导后续长期的软件开发工作



# 参与、承诺和分发软件项目计划

## ❑ 共同参与

- ✓ 一起讨论、明确任务、估算周期、确认进度

## ❑ 通过会议形式参与制定

- ✓ 便于加强交流

## ❑ 评审和承诺

- ✓ 共同评审，承诺计划内容

## ❑ 分发和宣传

- ✓ 分发给所有的项目组成员，让他们了解软件项目计划

# 3. 项目跟踪

## ❑ 任务

- ✓ **跟踪和及时调整**软件项目开发计划，**提供项目情况的可视性**，保证计划的适时调整

## ❑ 问题

- ✓ 要对哪些方面进行跟踪？
- ✓ 如何对软件项目进行跟踪？
- ✓ 当无法按预定计划实施时如何调整计划？
- ✓ 当跟踪发现问题时如何处理
- ✓ 如何提供工具辅助对软件项目进行跟踪.....

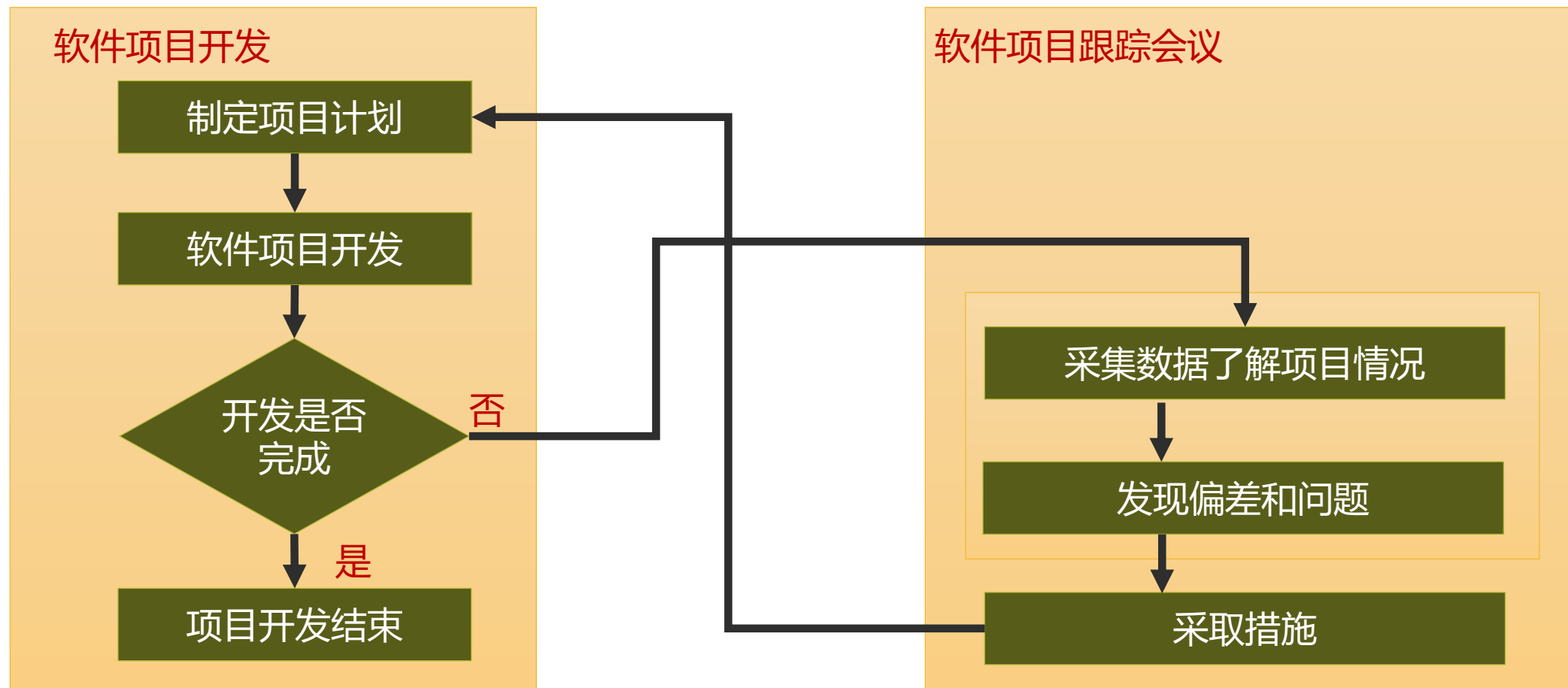
**项目跟踪有助于发现项目实施中存在的偏差：实际与计划不符**

# 为什么要进行软件项目跟踪

- ❑ 随时掌握软件项目的**实际开发情况**，提供软件项目的实施提供**可视性**
  - ✓ 哪些地方出现了什么样的问题
- ❑ **跟踪的对象**
  - ✓ **项目问题和风险**，软件项目在实施过程中会出现各种各样的问题和风险
  - ✓ **软件项目进展**，软件项目的实际进展与软件项目计划二者之间会产生偏差

# 项目跟踪的步骤和方法

## ❑ 周期性的项目跟踪会议



# 内容

## 1. 软件项目及其特点

- ✓ 软件项目及其管理的特殊性

## 2. 软件项目管理概述

- ✓ 软件项目管理的对象和要素

## 3. 软件项目管理的方法

- ✓ 软件风险管理

## 4. 软件项目管理的方法和标准

- ✓ 国内外标准、ISO和CMM系列





# 4. 风险管理

## ❑ 任务

- ✓ 对软件开发过程中各种**风险进行分析、预测、评估、监控**的过程

## ❑ 问题

- ✓ 什么是软件开发风险？
- ✓ 软件开发可能会有哪些风险？
- ✓ 如何客观地预测风险？
- ✓ 如何评估风险带来的影响？
- ✓ 如何避免和消除风险？
- ✓ 如何提供工具支持风险分析？ .....

风险管理有助于发现软件开发过程中存在的风险：实施中的问题

# 何为软件风险

## ❑ 软件风险

- ✓ 使软件项目的实施受到**影响和损失**、甚至导致**失败**的、可能会发生的**事件**
- ✓ 例如，人员的临时流失，计划过于乐观，设计的低劣

## ❑ 软件风险特点

- ✓ **事先难以确定**
- ✓ **带来损失**，影响项目实施，甚至会导致项目失败

# 示例：需求风险

- ❑ 软件需求已经成为项目基准，但仍在**变化**
- ❑ 软件需求定义**欠佳**：不清晰、不准确、不一致
- ❑ 软件增加了**额外**的需求
- ❑ 软件需求**没有得到**用户和客户的**确认**
- ❑ .....

# 示例：产品风险

- ❑ 错误率高的模块，需要更多时间对它进行测试、设计和实现
- ❑ 矫正质量低下的不可接受的产品需要更多的时间对它进行测试、设计和实现
- ❑ 由于功能错误，导致需要重新进行设计和实现
- ❑ 开发额外不需要的功能延长了进度
- ❑ 要满足产品规模和速度要求，需要更多的时间
- ❑ 严格要求与现有系统兼容，需要更多的时间
- ❑ 要求软件重用，需要更多的时间
- ❑ .....

# 示例：人员风险

- ❑ 招聘人员所需的时间**比预期要长**
- ❑ 作为人员参与工作的先决条件(如培训、其他项目的完成等)**不能按时完成**
- ❑ 开发人员与管理层**关系不佳**导致决策迟缓、影响全局
- ❑ 项目组成员**没有全身心地投入**到项目中，因而无法达到所需的产品功能和性能需求
- ❑ **缺乏激励措施**、士气低下，降低生产能力
- ❑ **缺乏必要的规范**，增加工作失误，重复工作，降低工作质量
- ❑ **缺乏工作基础**(语言、经验、工具等)
- ❑ 项目结束前，项目组成员**离开项目组**

# 风险管理模式

## ❑ 危机管理

- ✓ 类似于救火模式，听任软件风险的发生，及至造成麻烦后才着手进行处理

## ❑ 失败处理

- ✓ 听任软件风险的发生和演化，只是在风险发生之后才采取应对措施

## ❑ 风险缓解

- ✓ 识别软件风险，事先制定好风险发生后的补救措施，但不做任何防范措施

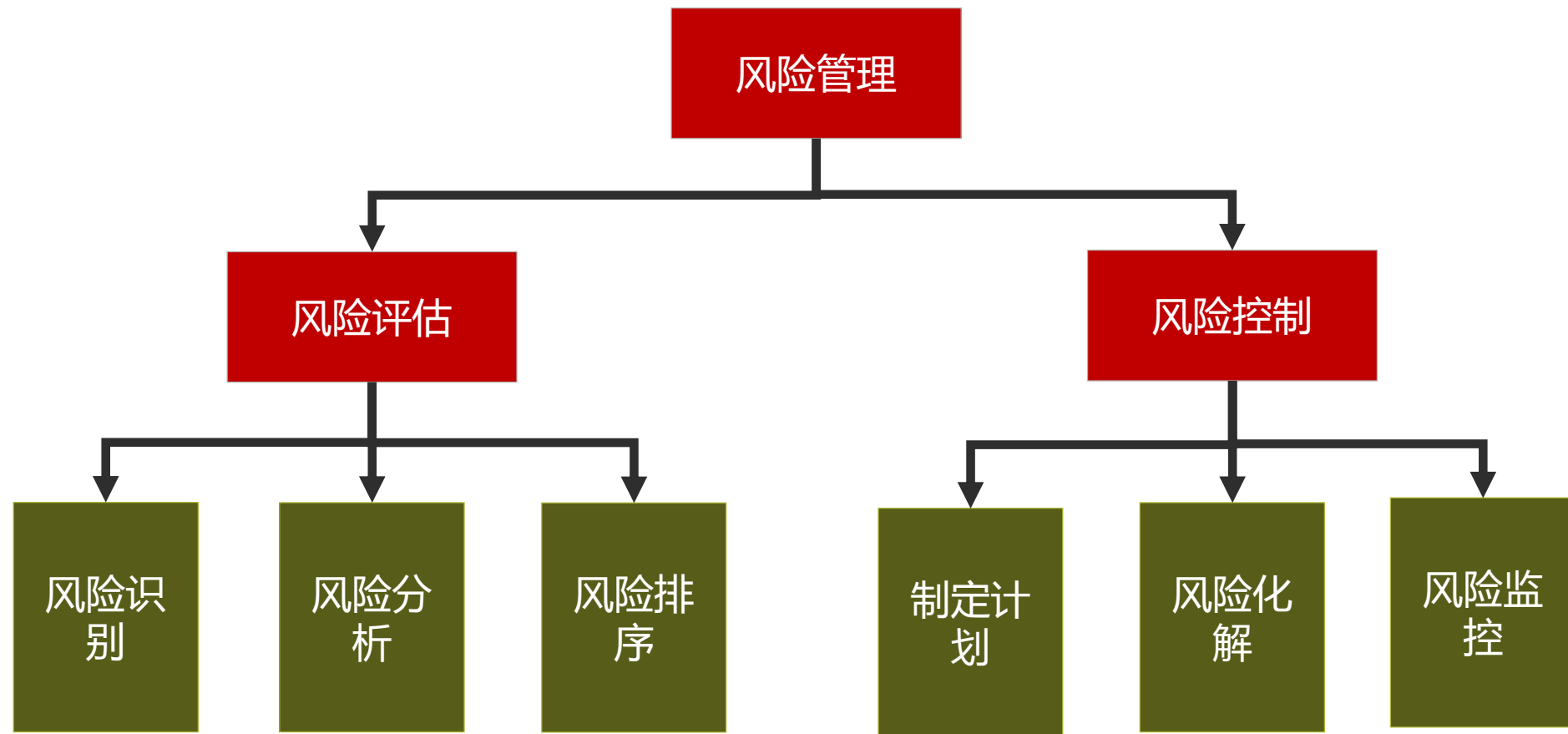
## ❑ 风险预防

- ✓ 预先识别和分析哪些不好事件可能会发生，制定好了万一发生的应对措施，同时采取措施防止它发生

## ❑ 消灭根源

- ✓ 不仅要识别出软件开发过程中各种潜在的软件风险，而且还要分析导致这些软件风险发生的主要因素，并采取积极的措施消除软件风险产生的根源

# 风险管理的组成



# 风险管理 (1/4)

## ❑ 风险识别

- ✓ 识别软件项目可能存在的各种**潜在软件风险**

## ❑ 风险分析

- ✓ 评估各项软件风险发生概率、可能造成损失及软件风险危险度

编号	风险名称	风险概率	损失 (人周)	危险度(人周)
1	软件项目规模的估算结果过于乐观	0.7	8	5.6
2	软件产品的交付日期提前	0.2	4	0.8
3	用户增加了额外的需求	0.8	5	4.0
4	需求分析工程师不能按时到位	0.9	2	1.8
5	需求分析所需的软件工具尚未到位	0.5	3	1.5



# 风险管理 (2/4)

## ❑ 风险优先级

✓ 根据软件风险的危险数，可以对软件风险的**优先级进行排序**

编号	风险名称	风险概率	损失 (人周)	危险度(人周)
1	软件项目规模的估算结果过于乐观	0.7	8	5.6
6	由于业务繁忙，用户没有足够多的时间配合需求分析小组开展需求调查工作	0.7	6	4.2
3	用户增加了额外的需求	0.8	5	4.0
4	需求分析人员不能按时到位	0.9	2	1.8
5	需求分析分析所需的软件工作尚未到位	0.5	3	1.5
2	软件产品的交付日期提前	0.2	4	0.8

# 风险管理 (3/4)

## ❑ 制定风险管理计划

软件风险管理计划	
风险编号	2
风险名称	小刘离开项目组
风险发生的对象	小刘
风险发生的原因	未知
风险可能发生的时机	二周后
消除风险的措施	由软件项目负责人小王和小刘交互， 询问离开软件项目组的真正原因，并 及时向高层反映情况
风险发生后的应对措施	让小陈接替小刘的工作

## □ 风险化解方式

- ✓ 避免风险、转移风险、消除发生软件风险的根源

## □ 风险监控

- ✓ 对软件风险的化解程度及其变化（如发生概率、可能导致的损失和危险度）进行**检查和监控**，并记录收集到的有关软件风险信息，以促进对软件风险的**持续管理**

# 5. 软件质量保证

## □ 任务

- ✓ 为软件产品的**质量提供某种可视性**，知道**哪些地方有质量问题**，便于改进方法和措施，提高软件产品的质量

## □ 问题

- ✓ 高质量的软件体现在哪些方面？
- ✓ 如何发现和保证软件产品的质量？
- ✓ 如何制定软件产品质量保证计划？
- ✓ 如何有效落实和实施计划？
- ✓ 如何提供工具支持软件质量保证？ .....

软件质量保证是软件项目管理的重要目标

# 软件质量保证

## ❑ 什么是软件质量保证

- ✓ 为管理层提供为获知产品**质量信息**所需的**数据**，从而获得产品质量是否符合预定目标的**认识和信息**

## ❑ 软件质量保证的目的

- ✓ 为软件产品的质量提供**可视性**
- ✓ 知道哪些地方有质量问题
- ✓ 便于改进方法和措施

## ❑ 示例

- ✓ 对软件进行测试和静态分析，发现有哪些错误
- ✓ 掌握这些错误的性质(严重、一般)

# 软件质量保证的内容

## ❑ 掌握软件产品质量

- ✓ 软件测试

## ❑ 提交软件质量报告

- ✓ 软件测试报告，说明质量问题

## ❑ 汇报项目组和管理层

- ✓ 例行的质量回报，便于改进管理和技术手段

# 从哪些方面关注软件质量

## ❑ 软件产品

- ✓ 软件需求规格说明书
- ✓ 软件设计规格说明书
- ✓ 源程序代码

## ❑ 开发活动

- ✓ 需求分析
- ✓ 软件设计
- ✓ 编码实现

## ❑ 标准和规程

- ✓ 代码编写标准
- ✓ 文档规范

# 如何保证软件质量

## ❑ 软件产品

- ✓ 文档类：审核，产生审核报告
- ✓ 代码类：测试，产生测试报告

## ❑ 开发活动

- ✓ 审查：产生审查报告

## ❑ 制定标准和规程

- ✓ 组织内部或者在项目开始时要制定软件开发的标准和规程



# 谁来执行和实施软件质量保证

- ❑ 软件项目**质量保证小组**(SQA小组)
- ❑ 独立于项目开发小组
- ❑ 具有比较大的权限

# 软件质量保证活动和手段

- ❑ 正确理解用户要求
- ❑ 制定相关标准规程
- ❑ 审核软件工作产品
- ❑ 审查软件开发活动
- ❑ 测试程序代码
- ❑ 记录各种偏差
- ❑ 记录不符合项，报告高级管理者

# 6. 产品管理 – 软件配置管理

## ❑ 任务

- ✓ 对软件产品进行**标识、存储、更动和发放**，记录和报告其**状态**，验证软件产品的**正确性和一致性**，并对上述工作的审计的过程

## ❑ 问题

- ✓ 如何标识和描述软件产品
- ✓ 如何对其软件产品的版本进行控制？
- ✓ 如何控制软件产品的变更？
- ✓ 如何制定软件配置计划？
- ✓ 如何利用工具支持软件配置活动.....

软件配置管理是有效管理软件产品及其之间关系、应对变化传播和影响

# 软件配置项的概念

## ❑ 软件生命周期内产生、需进行配置管理的工作产品

- ✓ 文档：SRS/SDS/...
- ✓ 代码：源代码、中间代码、可执行代码、...
- ✓ 数据：配置数据、数据库、数据文件、...
- ✓ 标准和规约：编码规范、...

# 软件配置项的描述

## ❑ 配置项的命名和编号

- ✓ 如SRS

## ❑ 配置项的属性

- ✓ 版本(1.0)
- ✓ 类型(文档)

## ❑ 配置项之间的关系

- ✓ 与软件概要设计文档、数据设计文档关联

# 基线的概念

## ❑ 已经通过正式复审和批准的**软件产品、标准或规约**,

- ✓ 它们可以作为进一步开发的基础
- ✓ 只能通过正式的变化控制过程才允许对它们进行变更

## ❑ 基线示例

- ✓ 经过评审后的, 发现的问题已经得到纠正, 用户和项目组双方认可并且正式批准的软件需求文档

# 什么是软件配置管理？

- ❑ 在软件生命周期中对SCI进行的以下工作
  - ✓ 系统地控制SCI的**标识、存储、更动和发放**
  - ✓ **记录、报告其状态**
  - ✓ **验证SCI的正确性和一致性**
  - ✓ 对上述工作的**审计**

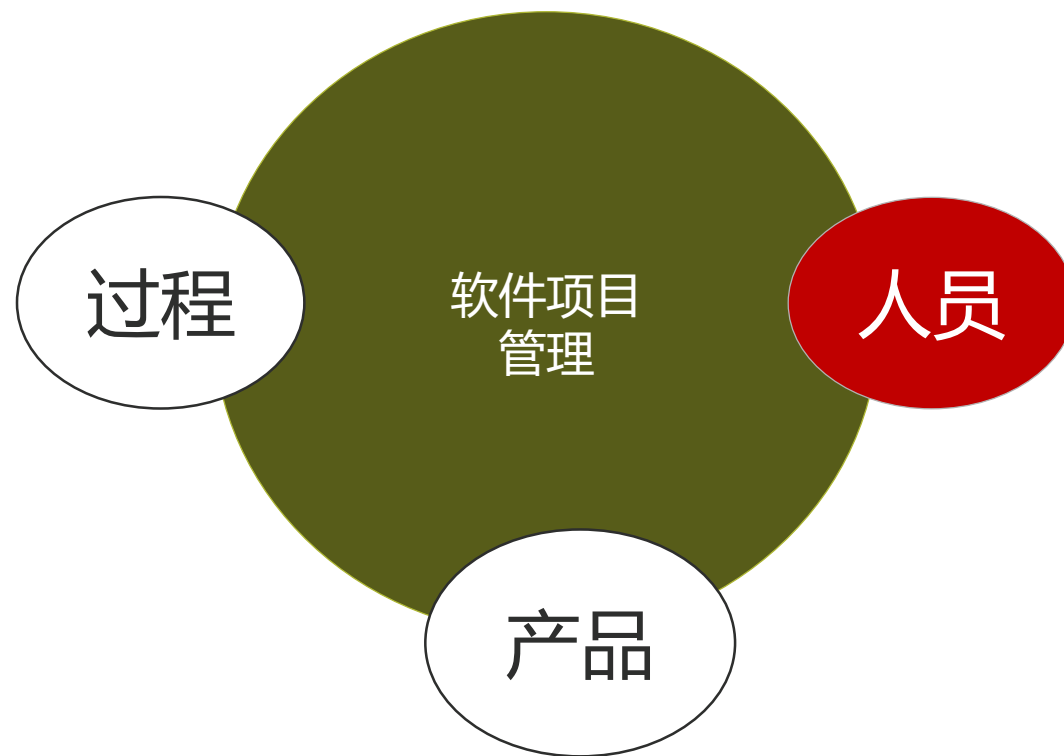
# 为什么需要SCM?

- ❑ 软件产品的**易改性与可控性**
- ❑ 修改很可能**引入新的错误**, 使结构变坏
- ❑ **牵一发而动全身**(影响域)
- ❑ 团队开发时, 多人**并发存取**需加控制(存取控制)
- ❑ 多应用开发时, 同一软件的不同版本可能对应于不同应用, 对此需加控制(版本控制)
- ❑ 应对软件更动状态予以**追踪**, 并及时向有关人员通报状态情况
- ❑ 如果软件产品不能自始至终地保持清晰、互相一致, 造成混乱、丢失, 那么该软件系统会因**无法使用而不得不报废**



# 7. 人员管理

- ❑ 软件项目团队
- ❑ 纪律激励机制



# 软件项目团队

## ❑ 任务

- ✓ 确定开发**团体的结构、明确人员的角色和任务、加强人员间的交流与合作**，结构合理、任务明确、团结协作、交流顺畅

## ❑ 问题

- ✓ 如何根据开发组织、软件项目和开发人员的特点来组建项目团队？
- ✓ 如何采取有效的措施来加强和促进人员之间的交流、沟通和合作？
- ✓ 如何提高团队的合作精神？
- ✓ .....

# 纪律和激励机制

## ❑ 任务

- ✓ 制定和实施纪律约束团体，通过激励机制**激发人员的激情，严格的纪律，饱满的激情**

## ❑ 问题

- ✓ 如何制定有效的纪律确保项目得以顺利的实施
- ✓ 如何制定措施激励员工的积极性和热情
- ✓ .....

# 软件项目的团队组织方法

## ❑ 团队的特点

- ✓ 由一帮**志同道合**的人员组成
- ✓ 一致、共同的**集体目标**
- ✓ **分工明确，相互合作**，共同完成任务

## ❑ 团队示例

- ✓ 跑步接力赛
- ✓ 软件项目开发
- ✓ .....

# 软件项目团队的运行模式(1/2)

## ❑ 一窝蜂模式

- ✓ 无组织，一窝蜂，无序和随意
- ✓ 典型例子是小孩子游戏

## ❑ 主治医生模式

- ✓ 主治医生主刀，其他人员协助
- ✓ 容易产生一人干活，其余打酱油

## ❑ 社区模式

- ✓ 志愿者没有报酬，众人拾柴火焰高
- ✓ 只烤火不拾柴，柴火质量低



# 软件团队的运行模式(2/2)

## ❑ 功能团队模式

- ✓ 平等协作，共同完成功能
- ✓ 功能完成后又重组织，完成其他功能
- ✓ 人员间没有管理关系，小组内部交流频繁

## ❑ 官僚模式

- ✓ 大领导--》小领导--》员工
- ✓ 存在明显的领导和管理关系
- ✓ 跨组织合作变得困难



# 团队中的合作

- ❑ **交流与沟通：面对面、基于互联网和平台**
- ❑ **共同解决问题**
- ❑ **协商和消解冲突**
- ❑ **多为他人提供帮助**
- ❑ **联合开展工作(如联调)**
- ❑ **.....**

# 内容

## 1. 软件项目及其特点

- ✓ 软件项目及其管理的特殊性

## 2. 软件项目管理概述

- ✓ 软件项目管理的对象和要素

## 3. 软件项目管理的方法

- ✓ 项目人员管理

## 3. 软件项目的标准

- ✓ 国内外标准、ISO和CMM系列





# 4.1 与软件项目管理相关的ISO标准 (1/2)

## ❑ ISO9001 -质量管理体系认证标准

- ✓ ISO颁布的、在全世界范围内通用的关于**质量管理和质量保证方面**的系列标准
- ✓ 主要应作好两方面的工作：一是建立健全质量保证体系，二是作好与体系认证直接有关的各项工作

## ❑ 相关组织（如企业）可根据ISO9001标准的具体要求，**制定和实施**针对特定产品和服务（如软件）的**质量体系**

## ❑ 第三方组织可依据ISO9001标准，对相关组织（如企业）的**质量体系进行认证**，以评判该组织的质量体系是否满足ISO9001标准

# 与软件项目管理相关的ISO标准 (2/2)

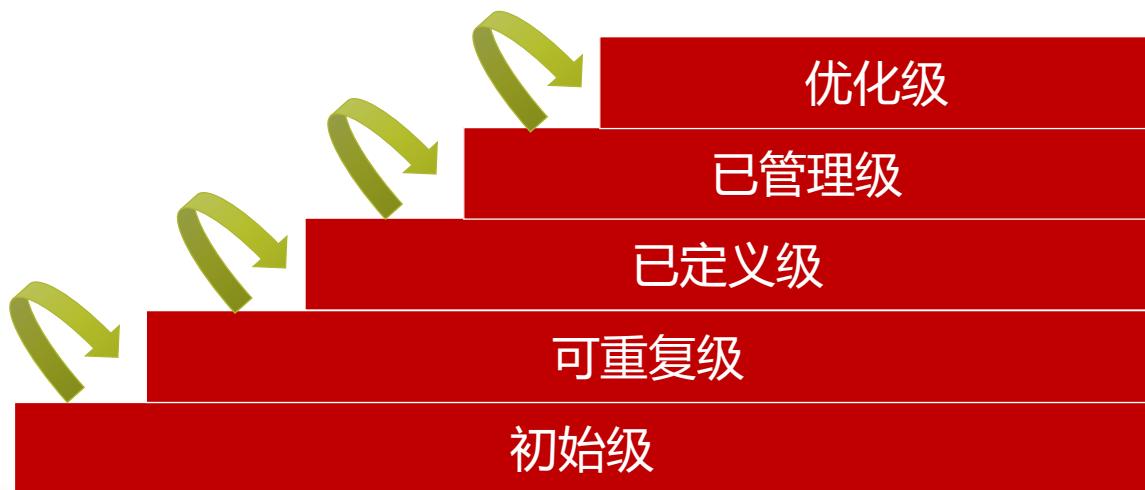
## ■ GJB9001系列

- ✓ 我国军方根据军用产品的特点和要求，制定了GJB9001国家军用系列标准
- ✓ 对承担军用产品、研制、生产、试验和维修任务的组织是必须执行的质量管理标准
- ✓ 1996年制订的**GJB9001**,2001年制订的**GJB9001A**，2009年制订的**GJB9001B**，2017年制订的**GJB9001C**
- ✓ 提出了武器装备“六性”的概念，即**可靠性、维修性、保障性、安全性、测试性、环境适应性**
- ✓ 强化了风险管理和软件工程管理要求

## 4.2 与软件项目管理相关CMM系列标准

### ❑ CMM

- ✓ 软件能力成熟度模型是一种对软件组织在**定义、实施、度量、控制和改善其软件过程**的实践中各个发展阶段的描述形成的标准
- ✓ CMM分为五个等级：**一级为初始级，二级为可重复级，三级为已定义级，四级为已管理级，五级为优化级**



- **关键过程域** (Key Process Area) 以明确每个成熟度层级的软件过程能力需达成的目标 (18个)
- 每个关键过程域都有其**任务和目标**

# CMM不同等级的关键过程域

等级\关键过程域	管理方面	组织方面	工程方面
L1初始级			
L2可重复级	<ul style="list-style-type: none"><li>— 需求管理</li><li>— 软件项目计划</li><li>— 软件项目跟踪与监控</li><li>— 软件转包合同管理</li><li>— 软件质量保证</li><li>— 软件配置管理</li></ul>		
L3已定义级	<ul style="list-style-type: none"><li>— 集成软件管理</li><li>— 组间协调</li></ul>	<ul style="list-style-type: none"><li>— 组织过程焦点</li><li>— 组织过程定义</li><li>— 培训程序</li></ul>	<ul style="list-style-type: none"><li>— 软件产品过程</li><li>— 同行评审</li></ul>
L4已管理级	<ul style="list-style-type: none"><li>— 定量过程管理</li></ul>		<ul style="list-style-type: none"><li>— 软件质量管理</li></ul>
L5优化级		<ul style="list-style-type: none"><li>— 技术更新管理</li><li>— 过程变更管理</li></ul>	<ul style="list-style-type: none"><li>— 缺陷防范</li></ul>

## 4.2 与软件项目管理相关CMM系列标准

### □ CMMI

- ✓ 能力成熟度模型集成模型，是在**CMM基础上**发展而来的
- ✓ 集成了**软件工程、系统工程、集成化制品和和过程开发**等三个过程改进模型
- ✓ CMMI共有**5个级别**：一级为执行级，二级为管理级，三级为明确级，四级为量化级，五级为优化级
- ✓ CMMI提供了**24个关键过程域**，并对相关关键过程域的名称和内涵做了适当的调整

# CMMI不同等级的关键过程域

等级\关键过程域	过程管理	项目管理	工程	支持
L1初始级				
L2可重复级		<ul style="list-style-type: none"><li>项目计划</li><li>项目监控</li><li>供应商合同管理</li></ul>	<ul style="list-style-type: none"><li>需求管理</li></ul>	<ul style="list-style-type: none"><li>配置管理</li><li>度量和分析</li><li>过程和制品质量保证</li></ul>
L3已定义级	<ul style="list-style-type: none"><li>组织过程焦点</li><li>组织过程定义</li><li>组织培训</li></ul>	<ul style="list-style-type: none"><li>集成项目管理</li><li>风险管理</li><li>组建团队</li></ul>	<ul style="list-style-type: none"><li>需求开发</li><li>技术方案</li><li>制品集成</li><li>验证</li><li>确认</li></ul>	<ul style="list-style-type: none"><li>决策分析和措施</li><li>组织集成环境</li></ul>
L4已管理级	<ul style="list-style-type: none"><li>组织过程性能</li></ul>	<ul style="list-style-type: none"><li>项目定量管理</li></ul>		
L5优化级	<ul style="list-style-type: none"><li>组织改革与实施</li></ul>			<ul style="list-style-type: none"><li>因果分析和和措施</li></ul>

# GJB5000系列标准

## ❑ GJB5000系列标准是由我国军方主导制订的军用软件能力成熟度模型

- ✓ 有多个版本，包括2003年颁布的**GJB5000-2003**、2008年颁布的**GJB5000A-2008**、2021年颁布的**GJB5000B-2021**
- ✓ **2024年3月**后全部贯彻实施GJB 5000B标准，并按此进行军用软件研制能力评价

## ❑ GJB5000系列标准规定了**军用软件能力成熟度的模型和军用软件论证、研制、试验和维护活动中的相关实践**

- ✓ 适用于军用软件论证、研制、试验和维护能力的**评价和过程改进**
- ✓ 相关组织需通过GJB5000的认证以展示其军用软件能力成熟度的水平，并依此作为资质参与军用软件研制的招投标

# GJB5000系列标准

## ❑ GJB5000系列标准是由我国军方主导制订的军用软件能力成熟度模型

- ✓ 有多个版本，包括2003年颁布的**GJB5000-2003**、2008年颁布的**GJB5000A-2008**、2021年颁布的**GJB5000B-2021**
- ✓ **2024年3月**后全部贯彻实施GJB 5000B标准，并按此进行军用软件研制能力评价

## ❑ GJB5000系列标准规定了**军用软件能力成熟度的模型和军用软件论证、研制、试验和维护活动中的相关实践**

- ✓ 适用于军用软件论证、研制、试验和维护能力的**评价和过程改进**
- ✓ 相关组织需通过GJB5000的认证以展示其军用软件能力成熟度的水平，并依此作为资质参与军用软件研制的招投标



# 小结

## ❑ 软件项目的**工程特征**

- ✓ 进度、成本、质量等约束，涉及人员、过程、产品等管理对象

## ❑ 软件项目的**管理内容**

- ✓ 包含**人员管理、过程管理、产品管理**等多个方面

## ❑ 软件项目的**常见管理方法**

- ✓ 结合项目特点选择合适的过程模型、组织项目团队
- ✓ 管理好软件制品，确保质量，发现风险

## ❑ 软件项目管理的**支撑标准**

- ✓ **国际、国内、军方**



信息与软件工程学院  
School of Information and Software Engineering

高级软件工程

THANK YOU

