

# 软件需求工程

## 课后讨论 2

徐鸣飞

# 目录

<b>第一章 课后讨论</b>	<b>2</b>	<b>1</b>
1.1 技术发展跟不上软件需求的原因与应对措施		1
1.1.1 原因		1
1.1.2 应对措施		1
1.2 不同软件开发过程的特点		2
1.3 限制软件发展的原因？		3
1.4 解决需求冲突的方法？		4
1.5 协调不一致业务需求的方法		5

# 第一章 课后讨论 2

## 1.1 技术发展跟不上软件需求的原因与应对措施

### 1.1.1 原因

以下为常见的原因：

- **复杂性增加：**软件系统的复杂性在不断增加，新的需求可能涉及到复杂的技术挑战。这使得开发人员需要花费更多的时间来理解和解决这些问题，导致技术发展的速度跟不上需求的增长。
- **技术债务：**在软件开发中，有时为了迅速推出产品，开发团队可能会采用一些权宜之计，导致技术债务的积累。这种债务可能会阻碍新技术的引入和系统的更新，从而使技术发展跟不上需求。
- **资源限制：**开发团队可能面临有限的资源，包括时间、人力和资金。这可能阻碍他们采用最新的技术和工具，从而使技术发展的速度受到限制。
- **技术碎片化：**技术领域的碎片化和快速演进，使得开发人员需要不断学习新技术，同时软件系统可能会面临不同技术的混合使用，增加了整合和升级的难度。
- **遗留系统：**很多组织依然在使用过时的、难以维护的遗留系统。这些系统可能无法轻松地与现代技术集成，从而限制了新技术的应用。

### 1.1.2 应对措施

应对这些问题的措施包括：

- 敏捷开发：采用敏捷开发方法，使开发团队能够更灵活地响应变化，并在较短的周期内交付可用的软件。
- 持续集成和持续交付（CI/CD）：CI/CD 流程可以加速软件交付，减少开发周期，使得软件更容易适应变化。
- 云计算：利用云计算平台可以帮助组织更灵活地扩展基础设施和服务，减轻硬件和基础设施管理的负担。
- 技术培训和成长：给开发团队提供不断学习的机会，确保团队对新技术的了解，并能够灵活地应对技术的变化。
- 创新文化：鼓励组织内部建立一种创新的文化，促使团队不断寻求新的解决方案和技术。
- 等等。

## 1.2 不同软件开发过程的特点

### 瀑布模型（Waterfall Model）：

- 特点：瀑布模型是一种线性顺序的开发过程，各个阶段是依次进行的，每个阶段的输出作为下一个阶段的输入。
- 优点：易于理解和使用，适用于小规模项目，有清晰的文档输出。
- 缺点：缺乏灵活性，不适应变化，风险难以管理，客户只能在最后验收阶段看到成果。

### 迭代模型（Iterative Model）：

- 特点：开发过程被分成小的迭代周期，每个迭代都包含开发、测试和部署等阶段。
- 优点：更灵活，允许反复调整和改进，能够及早发现和纠正问题。
- 缺点：需要更多的管理和沟通，可能导致一些阶段的重复。

**螺旋模型（Spiral Model）：**

- 特点：结合了瀑布模型和迭代模型的优点，以风险管理为核心，通过迭代的方式逐步增强系统。
- 优点：注重风险管理，适用于大型和复杂的项目，有助于及早发现问题。
- 缺点：复杂度较高，需要更多的资源和时间。

**敏捷开发（Agile Development）：**

- 特点：以灵活性和快速响应变化为核心，强调团队合作、持续交付和客户反馈。
- 优点：高度灵活，能够适应变化，客户参与程度高，快速交付价值。
- 缺点：对团队和客户的要求较高，不适用于所有类型的项目。

**DevOps：**

- 特点：结合开发（Development）和运维（Operations），通过自动化和协作实现快速、可靠的软件交付。
- 优点：缩短开发周期，提高交付质量，促进团队协作。
- 缺点：需要文化和组织层面的变革，可能面临技术和文化挑战。

### 1.3 限制软件发展的原因？

软件行业的发展受到多重因素的交织影响。首先，法规和合规性的要求对软件公司提出了严格要求，尤其是在数据隐私、网络安全和知识产权方面。这可能导致公司需要投入更多资源来确保其业务符合法规标准，从而限制了发展的速度。其次，知识产权和专利问题也是一个限制因素，一些公司通过专利保护其技术，形成了市场上的竞争壁垒，阻碍了其他公司的进入和创新。市场上的垄断现象和竞争不公平也可能限制了软件行业的整体发展。技术障碍是另一个方面，尽管软件行业取得了巨大的进步，但在引入新技术方面可能会受到一些技术上的制约，这可能导致行业发展的放缓。人才短缺、安全和隐私问题、经济不稳定、社会和道德考虑、以及环境可持续性问题都是软件行业发展中

的潜在障碍。因此，软件公司需要在克服这些多方面挑战的同时，谨慎应对各种因素，以实现可持续和健康的发展。

## 1.4 解决需求冲突的方法？

以下是一些解决需求冲突的方法：

1. **明确优先级：**对不同需求进行优先级排序，确保关键和紧急的需求先得到满足。这有助于确保在资源有限的情况下，优先满足最重要的需求。
2. **有效沟通：**促进各利益相关方之间的开放和透明的沟通，以理解他们的需求、担忧和目标。通过有效沟通，可以更好地协调并找到满足多方需求的平衡点。
3. **变更管理：**使用严格的变更管理流程，确保对需求的任何更改都经过审批和文档记录。这有助于防止不经过适当评估就进行的需求更改，从而减少潜在的冲突。
4. **协商和妥协：**寻找各方都能接受的解决方案，进行合理的协商和妥协。这可能涉及到权衡不同需求之间的关系，以寻找最佳平衡点。
5. **需求优化：**审查和优化需求，以确保其对项目目标的贡献最大化。有时候，通过更清晰地定义需求、删除冗余或过于具体的需求，可以减轻需求冲突。
6. **利益相关方参与：**将关键利益相关方纳入决策过程中，以确保他们的声音被充分考虑。这有助于建立共识和减少后期的需求变更。
7. **敏捷方法：**对于敏捷项目，采用迭代和增量的方式，可以更容易地适应变化并快速响应新的需求。敏捷方法的灵活性有助于缓解一些需求冲突。
8. **技术评估：**进行技术评估，了解不同需求对系统架构和性能的影响。这可以帮助决策者更好地理解技术上的限制和可能的妥协。
9. **项目管理工具：**使用项目管理工具来跟踪和管理需求，确保每个需求的状态和进展都能得到适当的监控和控制。
10. **权威决策：**在一些情况下，可能需要有权威的决策者做出最终的决定，以解决争议和需求冲突。

## 1.5 协调不一致业务需求的方法

以下是一些协调不一致业务需求的方法：

1. **需求工作坊：**组织需求工作坊，邀请各利益相关方共同参与，以明确并讨论业务需求。通过面对面的交流，可以更好地理解各方的期望和优先级。
2. **制定优先级：**在业务需求中确定优先级，确保关键和战略性的需求首先得到满足。这有助于减少不一致性，集中资源解决最重要的问题。
3. **建立联络人：**为每个关键利益相关方指定一个联络人，负责收集和传达他们的需求。这有助于确保信息的准确传递，减少信息失真。
4. **持续沟通：**建立定期的沟通渠道，包括会议、报告和邮件等方式，确保各方了解项目的进展和变化。及时沟通有助于发现并解决潜在的不一致性。
5. **需求跟踪工具：**使用专业的需求跟踪工具，追踪每个需求的状态和进展。这有助于确保每个需求都得到适当的处理和满足。
6. **制定共同的目标：**与各方合作，制定共同的项目目标和业务目标。通过建立共识，可以更容易地解决不同需求之间的矛盾。
7. **敏捷方法：**采用敏捷方法，通过迭代和反馈的方式不断调整业务需求。敏捷方法的灵活性有助于更好地适应变化和 inconsistency。
8. **冲突解决：**建立一个有效的冲突解决机制，当不一致性出现时，能够迅速而有效地解决。这可以包括调解、仲裁或其他冲突解决方法。