

Chapter 2

问题分析过程描述文档

2.1 明确问题

2.1.1 对问题达成共识

要素	内容
ID	P1
涉众	技术部经理、项目经理
问题	项目间调配开发人员的决策容易出现失误，很难保证将合适人选放在合适项目中去。
影响	部分需要增添人手的项目常常得不到合适的加入项目的人选。

要素	内容
ID	P2
涉众	项目经理、开发人员
问题	项目管理混乱、低效、无条理。(不明确!)
影响	导致开发者经常不清楚自己的当前任务是什么，进而引起项目延期或失败。

要素	内容
ID	P3
涉众	开发人员
问题	交流不便，缺乏与同公司的同行进行交流的途径。
影响	开发者在交流过程中浪费了时间，工作负担增加，而且缺乏向技术大牛交流学习的机会，对个人水平提高不利。

Table 2.1: “软件开发协作系统” 问题描述表

2.1.2 判断问题的明确性

1. 分析客户所提出的各个问题, 列出不明确的问题, 以便进一步与客户交流。
2. 不明确问题如下: P2: 项目管理混乱、低效、无条理。

3. 不明确性质：导致项目管理混乱、低效、无条理深层原因是什么？

2.1.3 分析不明确问题

对于列出的不明确问题,进行逐步分析,并通过查阅背景资料,画出鱼骨图,准备与客户交流。

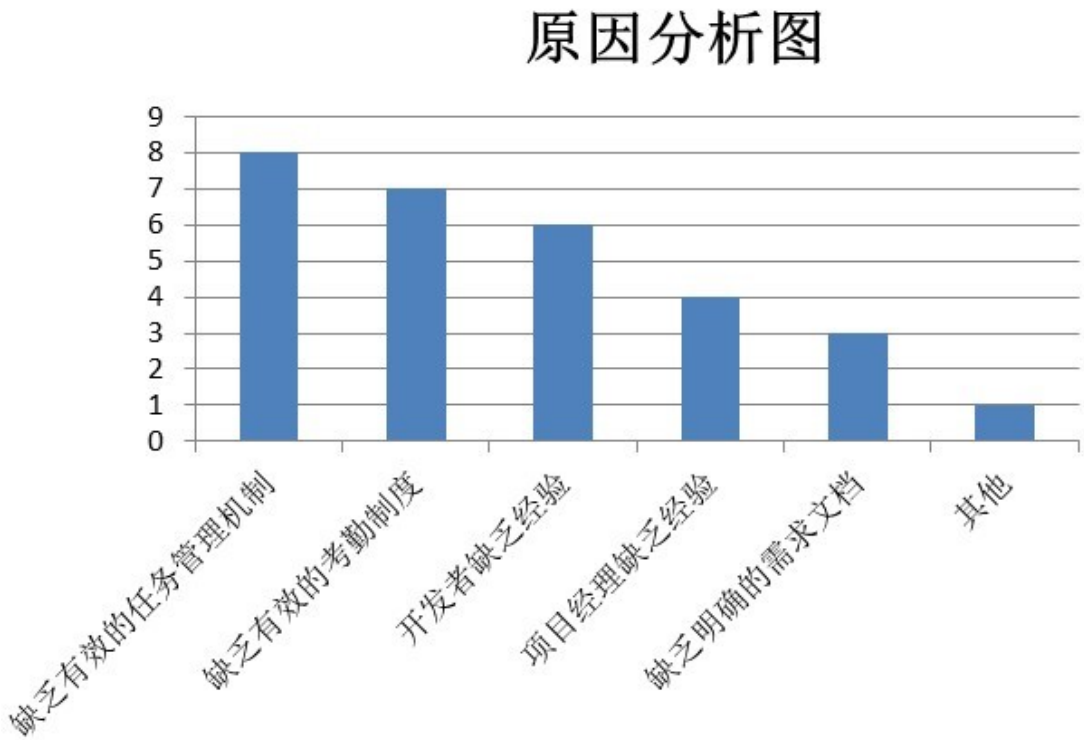


Figure 2.1: P2 原因分析图

通过在开发者与项目经理之间进行问卷调查 (原因有：缺乏一个有效的任务管理机制、缺乏有效的考勤制度、开发者缺乏经验、项目经理缺乏经验、缺乏明确的需求文档、其他)，绘制 P2 原因分析图，明确真正的问题所在。重新定义 P2 如下：

要素	内容
ID	P2
涉众	项目经理、开发人员
问题	缺乏一个有效的任务管理机制
影响	导致开发者经常不清楚自己的当前任务是什么，进而引起项目延期或失败。

2.2 发现业务需求

与客户交流,达成一致的业需求。

问题 ID	业务需求
P1	在系统投入使用六个月后，在项目经理中进行问卷调查，询问人才分配是否有所改善，回答“是”的比例不低于 60%。
P2	在系统投入使用六个月后，单个项目在规定期限内完成的比例超过未使用系统之前的 15%。
P3	在系统投入使用六个月后，在开发者中进行问卷调查，询问是否从公司内交流平台中获取学习经验或提升工作效率，回答“是”的比例不低于 60%。

Table 2.2: “软件开发协作系统” 业务需求表

2.3 定义解系统及系统特性

2.3.1 确定高层次的解决方案

对各个问题的目标进行分析, 并对每一个问题作出对应的几个可能的解决方案, 供客户选择。

问 题 ID	解 决 方 案 ID	方案描述	业务优势	代价
P1	S11	实现一个数据库,记录所有开发人员属性(包括技术水平、知识背景、交流能力、性格、健康程度、工作经历),这写信息从项目经理处获取,且每隔半年进行一次更新。技术部经理可以根据这些情报调配人员到不同的项目,合理安排人力资源。	技术部经理可以较精确地掌握开发人员属性,从而使决策正确率更高。	技术部经理需要花较多精力了解开发人员属性
P1	S12	实现一个数据库,记录所有开发人员属性(包括技术水平、知识背景、交流能力、性格、健康程度、工作经历),这写信息从项目经理处获取,且每隔半年进行一次更新。根据开发人员擅长的领域和工作经验,系统自动计算出合适的人选,给技术部经理一个推荐名单。然后技术部经理就可以直接选择这些人员,将其调配到某个项目中。	技术部经理不需要花较多精力了解开发人员属性	系统做出的决策可能会由于数据过时等原因与现实情况不符,因而不够灵活。
P2	S21	实现一个数学上基于 PetriNet 的图形化表示的任务管理功能模块,项目经理可以编辑、管理任务的 PetriNet 状的依赖关系。	可以清晰直观地呈现任务之间的依赖关系与整个项目的运行进度,不易出现遗漏。	未与日期相对应,不能在时间管理上为提供帮助,使用户难以把握各任务耗时。
P2	S22	实现一个日程管理模块,安排每天的任务	可作为每天工作的指导工具。使任务按期完成。	无法展示任务之间的逻辑关系,无法体现整个项目的运行进度,容易迷失在某些任务的细节之中而忽视了另一些必要的任务。
P3	S31	实现一个公司内部的实名制论坛。	可以方便疑难问题解答。	缺乏时效性,难以提升工作效率。
P3	S32	实现一个公司内部的实名制即时通讯工具。	具有时效性。	不能将某些话题与所有人讨论。

Table 2.3: “软件开发协作系统”高层次解决方案表

2.3.2 确定系统特性和解决方案

按照客户所选择的解决方案,进一步分析,确定每个解决方案的系统特性,并画出边界图。

针对 P1, 客户选择了 S11, 因为技术经理希望得到更多的掌控权限。

问题 ID	P1
解决方案	S11
系统特性	(1) 仅技术经理有权限登陆具有“指定开发者加入或退出某个项目团队”这一功能的客户端; (2) 仅项目经理在每年规定时间可以修改开发人员属性; (3) 所有用户都有权查看开发人员信息;

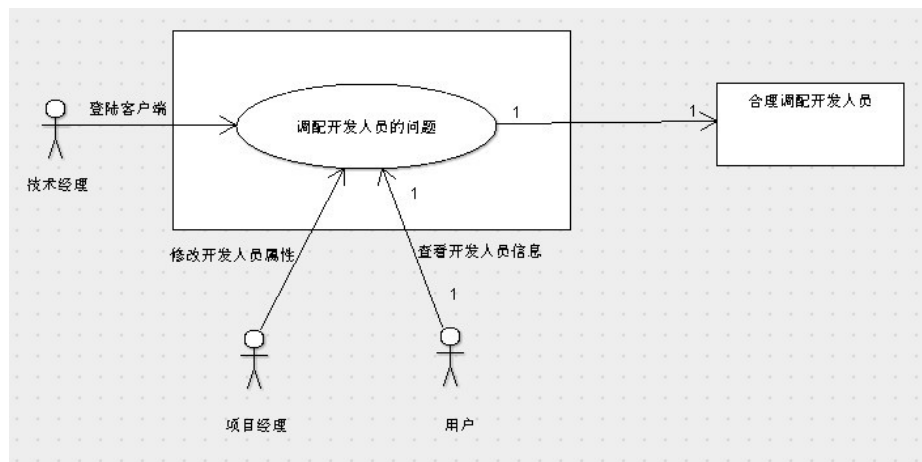


Figure 2.2: 解决方案 S11 边界图

针对 P2, 客户同时选择了 S21 和 S22 这两种解决方案, 因为他们认为这两种方案具有正交性。

问题 ID	P2
解决方案	S21
系统特性	(1) 仅项目经理可以使用“编辑任务”这一功能。 (2) 所有用户都可以查看任务。 (3) 系统提供一个基于 PetriNet 的图形化的任务表示方式。

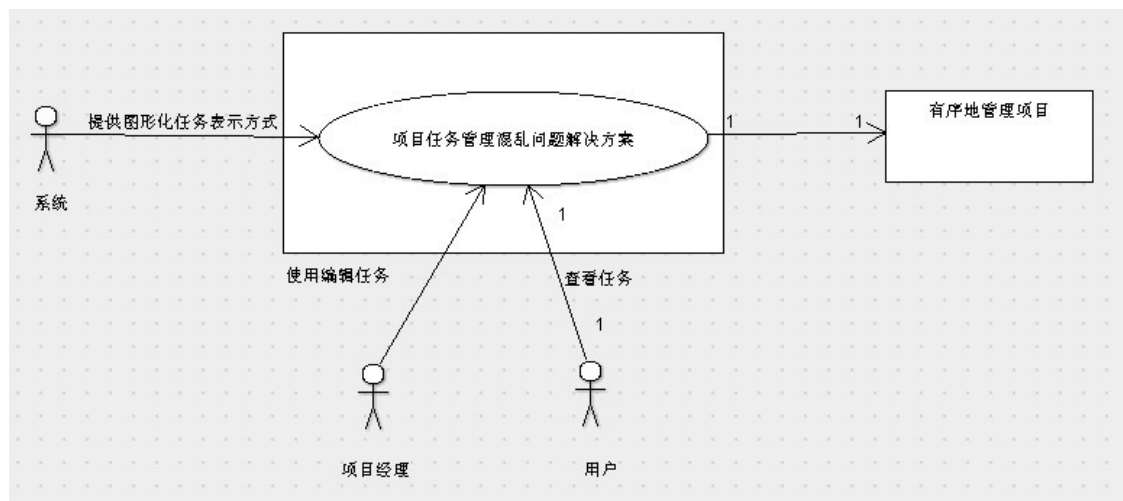


Figure 2.3: 解决方案 S21 的边界图

问题 ID	P2
解决方案	S22
系统特性	(1) 仅项目经理可以使用“编辑任务”这一功能。 (2) 所有用户都可以查看任务。 (3) 开发者可以通过留言对当天任务进行反馈，提出建议或者抗议。 (4) 系统提供一个日历，以日期为划分任务的依据。

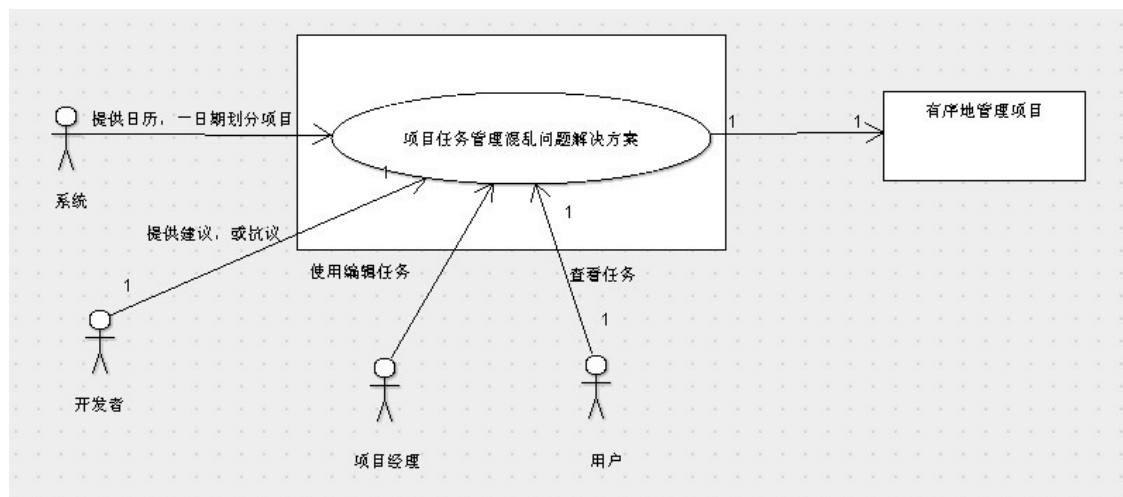


Figure 2.4: 解决方案 S22 的边界图

针对 P3，客户选择了 S32 这种解决方案，因为他们认为这种方案在公司内部可行性较高，上班期间随时可以进行即时通讯，解决沟通效率的迫切性大于提升员工水平的迫切性。

问题 ID	P3
解决方案	S32
系统特性	(1) 所有用户都可以使用该功能。 (2) 所有用户都是实名制的，其中开发者链接到其个人信息。

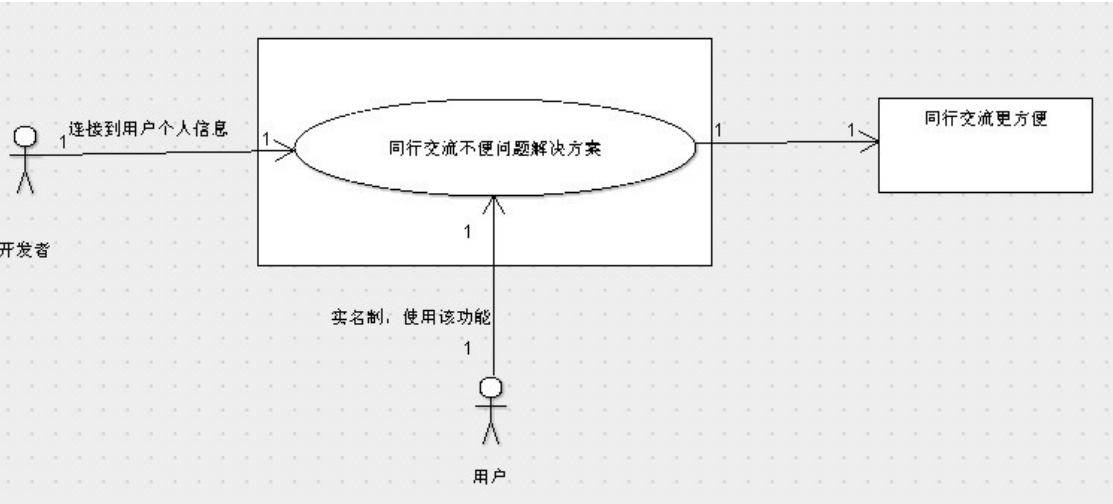


Figure 2.5: 解决方案 S32 的边界图

2.3.3 确定解决方案的约束

解决方案 S11 的约束

约束源	约束	理由
操作性	输入员工信息必须每隔半年就更新。	开发人员的水平会随时间而变化。
操作性	需要技术部经理花费较多精力整理、了解开发人员信息。	技术经理决策依据是系统中录入的开发人员信息。
设备和预算	系统必须在已有的服务器和主机上开发	成本控制和已有系统的维护。

解决方案 S21 的约束

约束源	约束	理由
操作性	项目经理需要了解 PetriNet 这一数学模型。	了解 PetriNet 后才能正确使用基于这一模型的图形化任务管理软件。
技术要求	要求客户端使用 C++ 开发高性能的本地图形界面应用程序。	绘图对计算性能要求较高。
设备和预算	系统必须在已有的服务器和主机上开发	成本控制和已有系统的维护。

解决方案 S22 的约束

约束源	约束	理由
操作性	要求项目经理每天或者至少常常使用该功能	如果项目经理长时间不更新, 基于日程的任务管理就无法发挥其时间管理上的优势。
设备和预算	系统必须在已有的服务器和主机上开发	成本控制和已有系统的维护。
人员资源	无外部资源	软件开发企业不会再给项目经理增派管理人员。

解决方案 S32 的约束

约束源	约束	理由
设备和预算	系统必须在已有的服务器和主机上开发	成本控制和已有系统的维护。
人员资源	无外部资源	即时通讯软件不需要管理员维护一个数据库, 信息都是点对点传递的。

2.4 确定系统边界

将各个问题的分析结果进行综合处理, 并把所有的解决方案综合, 得到整个解系统的功能和边界。

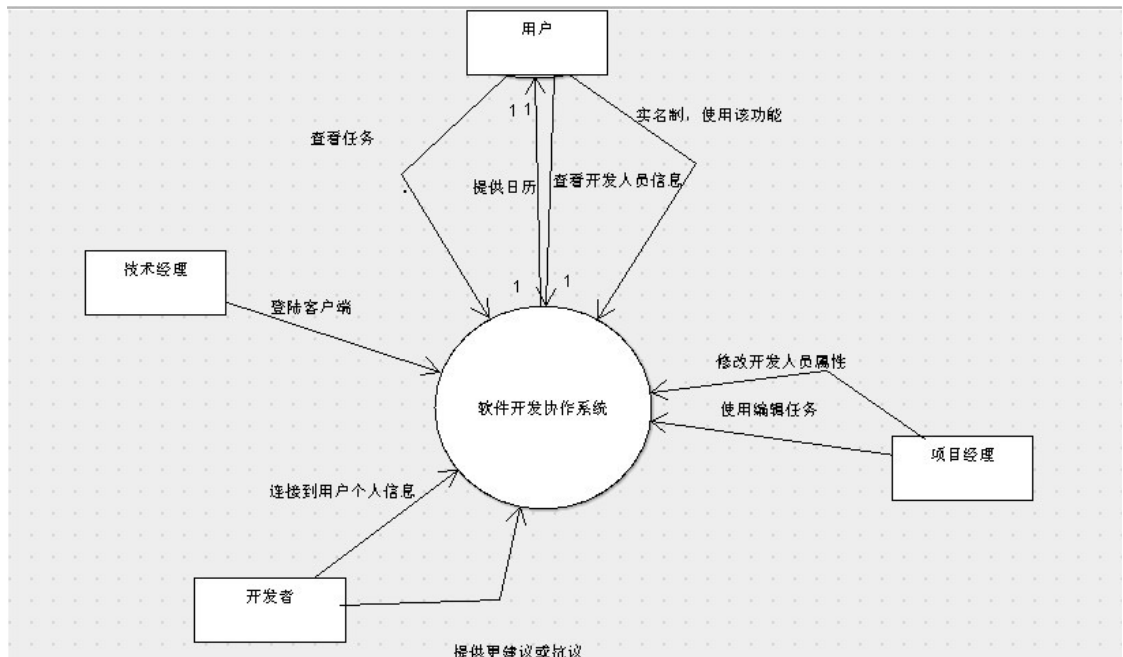


Figure 2.6: 系统边界图

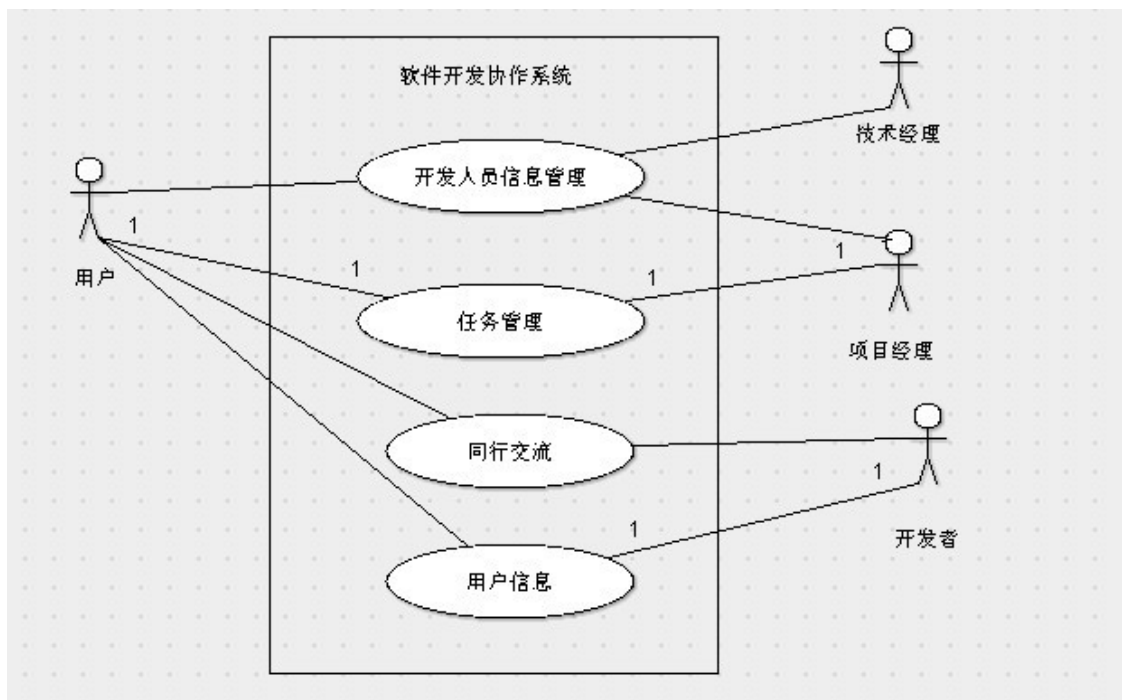


Figure 2.7: 系统用例图