

实战演练丛书（开发系列）

# Oracle 与 PowerBuilder 实战演练

樊东 赵雪梅 赵永生 孙星 编著

杨宝珍 审校

380R

人民邮电出版社  
[www.pptph.com.cn](http://www.pptph.com.cn)

附赠范例光盘



实战演练丛书（开发系列）

# Oracle 与 PowerBuilder

## 实战演练

樊东 赵雪梅 赵永生 孙星 编著

杨宝珍 审校

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>  
上由“馆藏检索”该书详细信息后下载，  
也可到视听部复制

人 民 邮 电 出 版 社

## 内 容 提 要

本书以商场管理信息系统的软件开发为主线,在基于 Oracle 数据库和 Power Builder 开发工具的基础上,系统地介绍了软件工程生存期的各个阶段,详细地描述了一个软件开发过程的实例。

全书共分为三篇 19 章。第一篇介绍了软件工程开发的基础知识和开发工具;第二篇详细描述了一个商场管理信息系统软件开发过程,其中包括了需求分析、概要设计、详细设计和具体实现等步骤;第三篇对软件运行维护方法进行了介绍。

本书适用于软件开发领域的大学生、研究生、各单位相关技术人员等软件开发工作者,是一本具有较强指导性和实用性的软件开发工具书。

实战演练丛书(开发系列)

### Oracle 与 PowerBuilder 实战演练

---

◆ 编 著 樊 东 赵雪梅 赵永生 孙 星

审 校 杨宝珍

责任编辑 李振广

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ pptph.com.cn

网址 <http://www.pptph.com.cn>

北京鸿佳印刷厂印刷

新华书店总店北京发行所经销

◆ 开本:787×1092 1/16

印张:26

字数:645 千字

2000 年 11 月第 1 版

印数:1-5 000 册

2000 年 11 月北京第 1 次印刷

ISBN 7-115-08834-9/TP·1859

---

定价:53.00 元

# 前 言

对于许多软件开发的人员来说，也许已经对软件开发所需要的开发知识和工具及其相关书籍进行过详细的阅读和研究，但真到了联系实际的时候又不知如何下手，特别是开发一个大型的软件工程，总感觉到理论和实际之间还缺少点什么，这就是经验。本书就是对这个“经验”进行详细描述“指导说明书”。

本书是根据作者多年的软件开发实践经验编写而成的，并将这些经验融合在商场管理信息系统的开发实例中。全书以开发实例为主线，在基于 Oracle 数据库的数据存储平台和 PowerBuilder 开发工具的基础上，详细描述了软件工程开发的具体过程。第 1 章概括介绍了软件工程的理论知识；第 2 章简述了 Oracle 数据库的基础知识；第 3 章概要介绍了 PowerBuilder 界面开发工具；第 4 章描述了实例的需求分析过程和方法；第 5 章叙述了实例总体方案设计方法；第 6 章对实例的详细设计进行了论述；第 7 章至第 17 章详细描述了实例的几个主要功能模块的具体设计和实现过程，其中还包括了 Oracle 数据库的具体应用方法和过程以及使用 PowerBuiler 开发界面等方法的过程；第 18 和 19 章对实例的系统验收及系统的运行与维护进行了介绍。

本书所附光盘的内容包括了开发实例的所有程序源码,实例是在 PowerBuilder 5.0 上实现的，稍加转换即可在 Powerbuilder 6.0 以上版本中运行，实例的运行、安装及使用详见盘中提供的“Readme.txt”文件。

樊东负责全书的设计、统稿和修改，并编写了第 1、2、4、5、6、18 和 19 章；赵雪梅负责编写了第 8、9、10 和 12 章，并负责光盘内容的制作和例子测试；赵永生负责编写了第 3、11、13 和 17 章，并负责配套光盘的制作；孙星负责编写了第 7、14、15 和 16 章。参加本书编写和指导工作的还有李罡、李玉东、符洁、陈卫东及柴海萍等同志。最后对所有关心、支持和帮助过本书编写工作的领导、朋友和家人表示诚挚的谢意！

由于作者的能力和水平有限，本书可能会有谬误之处，恳请读者批评指正。

编著者

# 第一篇

# 战前准备

第 1 章 软件工程理论的准备

第 2 章 Oracle 数据库基础的准备

第 3 章 PowerBuilder 界面开发工具基础的准备

## 本篇导读

本篇主要针对开发一个大中型信息系统所必须掌握的计算机科学技术知识进行一些概要性地说明，以便使初次进行大中型信息系统开发的读者能迅速入门，少走弯路。

随着计算机科学技术的发展和推广应用，当今社会已进入信息时代。在这个时代，大量的信息系统需要去开发和建设。如何能成功地建立起一个实用的信息系统，是用户和开发技术人员共同期望的目标。然而，现实的情况是建立起能称得上实用的信息管理系统并不那么容易。原因是多方面的，其中就包括数据库的结构和框架设计的合理性。而造成设计不合理的根源就是开发人员没能很好地理解和掌握软件工程理论思想和开发工具。为了给后面的实例提供充分的技术准备，下面将对软件工程理论、Oracle 数据库和 PowerBuilder 开发工具进行有针对性地介绍。

# 第1章

## 软件工程理论的准备

系统开发计划  
软件需求分析  
软件设计  
程序编写  
软件测试  
编制系统文档阶段  
系统运行与维护期的工作



软件工程技术发展到今天，已基本上形成了一套成熟的技术理论。本章撇开其复杂的系统论述（如有需要者，请参阅相关的专著），单刀直入地对软件工程生存期的六个基本阶段进行简要说明与解释。

本章主要内容：

- 系统开发计划
- 软件需求分析
- 软件设计
- 程序编写
- 软件测试
- 编制系统文档阶段
- 系统运行与维护期的工作

## 1.1 系统开发计划

系统开发计划在软件工程生存期中属于定义阶段。它确定开发软件系统的总目标，并列出其功能、性能、可靠性和接口等方面的要求；由该系统开发人员和用户密切合作，完成其软件部分的可行性研究，探讨解决问题的可能方案，还要对可利用的资源（如计算机硬件、软件、人员等）、成本、可取得的效益、开发的进度做出估计，然后制定出完成开发任务的实施计划，并将和可行性研究报告一起提交给主管部门审查。

### 1.1.1 可行性研究

可行性与风险分析是密切相关的。如果项目的风险很大，就会降低产生高质量软件的可行性。可行性研究主要集中在以下四个方面：

（1）经济可行性：进行开发成本的估算及可能取得效益的评估，确定待开发系统是否值得投资开发。

（2）技术可行性：对待开发的系统进行功能、性能和限制条件的分析，确定在现有的资源条件下，技术风险有多大，系统是否能实现。这里，资源包括已有的或可以搞到的硬件资源、软件资源、现有的技术人员的技术水平与已有的工作基础。

（3）法律可行性：确认待开发系统是否会涉及任何侵犯、妨碍法律等责任问题。

（4）抉择：对系统开发的不同方案进行比较评估。成本和时间的限制，会给方案的选择带来局限性，对于一些合理的方案都应加以考虑。

对于大多数系统考虑经济可行性时，一般先有一个投资规模底线，然后据此配置系统，以达到最好的性价比。经济可行性研究涉及范围较广，它包括成本/效益分析、长期的公司经营策略、对其他的单位或产品的影响、开发所需的成本和资源，以及潜在的市场前景。

技术可行性常常是最难决断的方面。因为系统的目标、功能、性能比较模糊。技术可行性的评估与分析 and 定义过程并行进行是十分必要的。一般情况技术可行性要考虑的包括：

（1）开发的风险：在给出的限制范围内，能否设计出系统，并实现必需的功能和性



能？

(2) 资源的有效性：可用于开发系统的人员是否存在问题？可用于建立系统的其他资源（如硬件、软件等）是否具备？

(3) 技术：相关技术的发展是否能支持这个系统？

开发人员在评估技术可行性时，一旦估计错误，将会出现灾难性的后果。

可行性研究可以归档为一个单独的报告，提供给上级部门，也可以包括在“系统规格说明”的附录中。虽然可行性报告的形式可以有多种，但最重要的内容应当有：

(1) 项目背景：问题描述；实现环境；限制条件。

(2) 管理概要与劝告：重要的研究结果；说明；劝告；影响。

(3) 候选方案：候选系统的配置；选择最终方案的准则。

(4) 系统描述：简略的范围描述；分配元素的可行性。

(5) 经济可行性：经费概算；预期的经济效益。

(6) 技术可行性：技术实力；已有工作基础；设备条件。

(7) 法律可行性：系统开发可能导致的侵权、违法和责任。

(8) 用户使用可行性：用户单位的行政管理、工作制度、使用人员的素质。

(9) 其他与项目有关的问题：其他方案介绍；未来可能的变化。

可行性研究报告首先由项目负责人审查内容是否可靠，再上报给上级主管部门审阅。从可行性研究应当得出“可行或不可行”的决断。当然，在以后的开发阶段，还要作出其他的“可行或不可行”决定。

### 1.1.2 系统需求鉴别

在系统开发计划中，不仅仅只考虑软件系统，而是一个整体的计算机系统工程分析。分析的目标就是：

(1) 识别用户要求；

(2) 评价系统的可行性；

(3) 进行经济分析和技术分析；

(4) 将功能分配给硬件、软件、人、数据库和其他系统元素；

(5) 建立成本和进度限制；

(6) 生成系统规格说明，形成所有后续工程的基础。

这需要硬件和软件专家共同合作来实现以上目标。

计算机系统工程分析过程的第一步就是识别用户要求。要求的识别是系统开发的起点。为此，分析员必须回答以下问题，才能完成这一步骤：

(1) 所期望的功能和性能是什么？

(2) 可靠性和质量问题有哪些？

(3) 总的系统目标是什么？

(4) 成本与进度如何？

(5) 研制的需求是什么？

(6) 市场与竞争情况怎样？

(7) 有效的技术有哪些？

### (8) 将来可能有哪些扩充?

刚开始时, 系统分析员帮助用户定义系统的目标: 将产生什么信息? 要求提供什么信息? 要求的功能和性能如何? 系统分析员要查明用户要求(对系统成功地实现是关键的项目)与用户希望(是有用的但不是必须的或无法实现的)之间的区别。

一旦识别了总目标, 系统分析员进一步评估某些辅助信息, 如建立系统的技术是否存在? 需要哪些开发和制造资源? 成本与进度的限制如何? 如果新系统是一个要卖给多个用户的产品, 则还要考虑该产品的潜在市场如何? 与其他产品相比, 该产品的竞争力如何? 在公司的所有产品中该产品的地位如何?

把在要求识别步骤所收集的信息定义到一个“系统概念文档”中, 原始的概念文档有时由用户与系统分析员之间要不断进行对话, 对文档进行修改。

### 1.1.3 成本/效益分析

成本/效益分析的目的, 是从经济角度评价开发一个新的软件项目是否可行。成本效益分析首先是估算待开发系统的开发成本, 然后与可能取得的效益(有形和无形)进行比较和权衡。有形的效益可以用货币的时间价值、投资回收期、纯收入等指标进行度量。无形的效益主要从性质上、心理上进行衡量, 很难直接进行量的比较。例如, 通过重复优化得到更好的设计质量; 通过可编程控制使用户更满意; 通过对销售数据重定格式和预定义产生更好的商业决策等, 就很难进行直接的量的比较。无形的效益在某些情况下会转化成有形的效益。例如, 一个高质量的设计先进的软件产品可以使用户更满意, 从而影响到其他潜在的用户也会喜欢它, 一旦需要时就会选择购买它, 这样使得无形的效益转化成有形的效益。

由于成本效益分析和技术开发关系不大, 具体详细的度量和分析方法这里略述, 需要对此深入了解的读者请阅读有关专著。

### 1.1.4 技术分析

在技术分析时, 系统分析员评价系统概念的技术价值, 同时收集附加的有关性能、可靠性、可维护性及生产率方面的信息。在这个分析步骤中还包括一些研究和设计工作。

技术分析提交系统技术可行性评估。指明为完成系统的功能和性能需要什么技术? 需要哪些新设备、方法、算法或者过程? 有什么开发风险? 这些技术问题对成本的影响如何?

用于技术分析的方法, 有数学模型和优化技术、概率和统计、排队论、控制论等。但有时分析和评估是不可行的。模型化方法, 包括数学模型和物理模型, 是待开发系统进行技术分析的一种有效的方法。其具体的模型化过程这里略述。

根据对实际领域的观察或对系统目标的逼近, 建立模型。系统分析员评价模型的特性, 并将它与实际的或期望的系统特性做比较, 进而深入分析所建立系统的技术可行性。

技术分析的结果决定待开发系统软件项目可行, 还是不可行。如果技术风险严峻或者模型表明不能实现所期望的功能和性能, 或者某些部分不能一致配合, 那么就要决定取舍。

### 1.1.5 系统的软硬件分配与权衡

一旦解决了技术的可行性问题, 就开始考虑多种解决方案, 直观上说, 哪些系统功能

由硬件实现，哪些由软件实现。根据实际情况配置多个候选方案，然后进行权衡与折衷，从而选定某一方案作为实现方案。

### 1.1.6 系统结构的模型化

系统功能的实现分别由软、硬件进行，由此就能够建立起系统的模型，为稍后的需求分析和设计奠定基础。众所周知，任何一个基于计算机的系统都能够模型化为使用输入——处理——输出结构的一个信息变换，即可抽象为数据和方法。大的系统可划分成几个小系统，小系统还可以分解成子系统，如此分层下去，直到能清晰地划分出软、硬件信息传递的接口，并能用系统结构的规格说明定义它们。

### 1.1.7 系统结构的规格说明定义

为了下一步工程的工作需要，系统分析员要对各个子系统及它们之间的信息加以定义。系统结构的规格说明就是担当此任的。

系统结构规格说明是作为硬件工程、软件工程、数据库工程、人类工程的基础而使用的一个文档。它描述了系统的功能和性能，以及管理该系统开发的一些限制条件。给软件工程师指明了软件在整个系统，以及各种子系统环境中的作用。系统结构规格说明还描述了系统的输入/输出（数据与控制）信息。

系统规格说明的主要内容如下：

- (1) 引言
  - 文档的范围和目的。
  - 概述：目标；限制条件。
- (2) 功能和数据描述
  - 系统结构：结构环境图及其描述。
- (3) 子系统描述
  - 对于子系统的结构图描述：流程图；模块描述；性能问题；设计限制条件；系统部件的分配。
- (4) 系统模型化和模拟结果
  - 用于模拟的系统模型。
  - 模拟结果。
  - 特殊的性能问题。
- (5) 项目问题
  - 项目开发成本。
  - 项目进度安排。
- (6) 附录

在这里提醒读者的是：系统规格说明的实际格式和内容可以根据软件或系统工程标准，或者根据用户本身的要求来决定。

### 1.1.8 系统定义的评审

系统定义的评审主要是评价在系统规格说明中所做出规定的正确性。评审由开发人员

和用户合作进行，以保证：

- (1) 正确地定义了项目的范围；
- (2) 适当地定义了功能、性能和接口；
- (3) 环境的分析和开发风险证明了系统是可行的；
- (4) 开发人员与用户对系统的目的达成了共识。

系统定义评审分两步走。先从管理的角度进行审查，再对系统元素和功能进行技术评估。

管理方面考虑的关键问题有：

- (1) 是否已经建立了稳固的商业需求？系统可行性是否合理？
- (2) 特定的环境（或市场）是否需要所描述的系统？
- (3) 考虑了哪些候选方案？
- (4) 每一个系统元素的开发风险有哪些？
- (5) 资源对于系统的开发是有效的吗？
- (6) 成本与进度界限合理吗？

系统技术评审时考虑的详细程度随功能分配工作时考虑的详细程度而改变。评审应当包括以下问题：

- (1) 系统的功能复杂性是否与开发风险、成本、进度的评估相一致？
- (2) 功能分配是否定义得足够详细？
- (3) 系统元素之间的接口，系统元素与环境的接口是否定义得足够详细？
- (4) 在规格说明中是否考虑了性能、可靠性和可维护性问题？
- (5) 系统规格说明是否为后续的硬件和软件工程步骤提供了足够的基础？

一旦完成了系统评审，工程开发就开始并行地开展。系统的硬件、人员、数据库元素将成为它们各自的工程过程的一部分，而软件工程的开发才真正展开。

## 1.2 软件需求分析

软件需求分析工作是软件生存期中重要的一步，也是关键性的一步。只有通过软件需求分析，才能把系统的功能和性能的总体概念描述为具体的软件需求规格说明，从而为后面的软件开发打下基础。软件需求分析是一个渐进的过程，通过开发人员与用户的不断相互交流，逐步加深对系统功能和性能的认识，并且细化、分解软件需求。该过程将软件计划阶段所确定的软件需求逐步细化到可详细定义的程度，并分析找到可行的解决方法。

制定软件的需求规格说明不只是软件开发人员的事，用户也起着至关重要的作用。用户必须对软件功能和性能提出初步要求，详细说明原有系统是怎样做的，新系统应该做什么，并澄清一些模糊概念。而软件开发人员则要认真了解用户的需求，细致地分析，把用户的要求转换成一个完全的、明确的软件逻辑模型，并写出与用户要求相符合的软件需求规格说明。

### 1.2.1 软件需求分析的任务

需求分析规定了系统设计统一的、最基本的要求，需求分析的结果是系统设计、实现、测试验收和维护的依据。

需求分析是在可行性研究的基础上，确定新项目必须完成哪些工作，也就是对项目最终目标提出完整、准确、清晰、具体的要求。

需求分析具体任务是：

- ① 确定用户对新系统的综合要求；
- ② 分析新系统的数据要求；
- ③ 导出新系统详细逻辑模型（数据流、数据字典等）；
- ④ 确定系统开发计划。

#### 1. 确定用户对新系统的综合要求

##### （1）系统的功能要求

开发人员在为用户确定系统的功能时，应尽量做到细化、具体、全面，使用户感到这些功能确实能解决他们的问题。确定主要功能和次要功能，并用文字、图形、逻辑或数学方法描述其特性。例如，系统具有资料检索功能，则应分解为数据录入、增加数据、修改数据、删除数据、数据输出以及统计计算等功能。又如，系统具有目标实时运动态势处理功能，则应分解为实时信息搜集、运动态势综合计算、地图处理与显示、运动目标标绘及图上计算等功能。

系统的功能要求应按以下几方面确定：

- ① 输入：必须确定与功能有关的所有输入信息，包括其来源、意义、格式、接收方法、数量、输入范围及换算方法，必须说明时间要求、优先顺序（常规作业，紧急情况），操作控制要求和所用的输入媒体。
- ② 处理：必须确定输入数据到中间数据直到获得预期输出结果的全部过程，操作的准确顺序，非正常情况的响应。对每种功能的算法及其实现作文字描述，必要时给出图形、逻辑描述或相应的数学描述。
- ③ 输出：必须确定与功能有关的所有输出信息，包括信息的传送方法、意义、格式、数量、输出范围及换算方法。必须说明时间要求、优先顺序和输出形式（显示、打印等）。
- ④ 特殊要求：必须确定系统是否有特殊要求或应急措施。

##### （2）系统的性能要求

除了功能满足用户要求外，系统的性能要求也很重要。如果新系统不能给用户带来工作上的方便和提高效率，用户是不能接受的。系统的性能要求一般包括：响应时间、处理容量、存储容量、人机交互的友好性、系统的安全性、可靠性以及容错能力等。例如，对资料检索的性能要求是：常用资料检索响应时间，非常用资料检索响应时间，资料信息存储容量，后备存储容量（可存储几年、几个月的数据），可以打印报表的类型、数量等。

系统的性能要求应按以下几方面确定：

- ① 容量要求：确定系统的容量要求，如处理的记录数和处理数据的最大容量等。
- ② 精度要求：确定系统的精度要求。如数据或数值计算的精度要求、数据传输的精度要求等。

③ 时间特性要求：确定系统的时间特性要求。如处理时间、响应时间及其峰值负载期间允许的偏离范围，系统各项功能的顺序关系，由于输入类型的不同和操作方式的变化而引起的优先顺序的变化等。

④ 适应性要求：必须指明反映系统环境变化和系统适应能力的各种参数。说明当需求发生某些变化时系统的适应能力，指出为适应这些变化而需要设计的软件成分和过程。

⑤ 人机交互友好性：要有适用于用户特点的智能化人机交互界面，便于用户操作使用。

⑥ 系统可靠性：为提高系统可靠性必须采用模块化、结构化设计，对需要容错处理的模块提供备份，以便减少系统故障。

⑦ 系统可维修性：系统应具有较好的可维修性，以便在系统发生故障时，迅速恢复系统运行。

## 2. 分析新系统的数据要求

新系统的数据包括向系统输入的数据和用户所需要输出的数据。

新系统的所有数据应根据系统功能确定，当系统细化到分系统、子系统及模块时，应将外部输入系统的数据及系统向外输出的数据全部确定下来。

分析新系统的数据要求的目的是：准确地描述系统所要处理的数据元素组成，这有利于全系统的数据一致性，以便进行系统的详细设计。

## 3. 导出新系统的数据字典

对于一个新系统来说一下子列出所有数据字典是不可能的，通常先将数据字典分成若干类，建立数据关系结构，然后再逐一细化各类中数据字典。

## 4. 构造新系统详细逻辑框图

开发人员应根据系统功能要求及系统数据要求，逐一分解出所有数据在系统中处理过程。

## 5. 绘制系统流程图

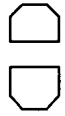
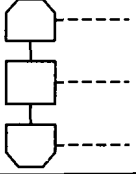
绘制系统流程图：系统流程图表达的是在系统各部件（硬件与软件）中信息流动的情况，这个流程图只描述信息如何流动，不对信息进行加工处理。系统流程图是高度概括的模型，不像程序流程图那样对信息加工处理，控制过程描述得很细致。系统流程图与程序流程图所有符号相同。

常用的系统流程图符号如表 1-1 所示（参照国家标准 GB1526-89）。

表 1-1 常用的系统流程图符号

| 符号                                                                                  | 名称     | 说 明                         |
|-------------------------------------------------------------------------------------|--------|-----------------------------|
|  | 加工处理   | 对数据进行加工处理的部件,程序或人工处理等都是加工处理 |
|  | 数据     | 表示数据                        |
|  | 存储数据   | 表示以一种适合于处理的形式表达的存储数据        |
|  | 内存储器   | 表示数据，媒体为内存储器                |
|  | 直接内存储器 | 可以直接存储的数据，媒体为磁盘、软磁盘等        |

续表

| 符号                                                                                 | 名称            | 说 明                                                                                                                                  |
|------------------------------------------------------------------------------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------|
|   | 文件            | 表示人可阅读的数据，媒体为打印输出、光学符号识别文件、输入表格等                                                                                                     |
|   | 人工输入          | 表示数据，媒体可为任意类型，如键盘、开关、按钮、光笔输入、条形码输入器，处理过程中信息以人工方式输入                                                                                   |
|   | 显示            | 表示数据，媒体可为任意类型，如视频屏幕                                                                                                                  |
|   | 既定处理（或称预定义过程） | 表示一个已命名的处理，它由在别处已详细说明的一个或多个操作或程序步骤所组成，例如子程序、模块                                                                                       |
|   | 准备            | 表示对影响随后活动的一条或一组命令的修改，例如程序初始化                                                                                                         |
|   | 判断            | 表示判断或开关类型功能。该符号只有一个入口，但可以有若干个可选的出口，在对符号中定义的条件进行求值后，有一个且只有一个出口被激活                                                                     |
|   | 控制转移          | 表示从一个处理到另一个处理的直接转移                                                                                                                   |
|   | 流线            | 表示数据流或控制流，箭头表示流向                                                                                                                     |
|   | 通信连接          | 表示通过远程通信线路连接进行的数据传输                                                                                                                  |
|   | 虚线            | 表示两个或更多符号间的选择关系，也可用于标出注释区域                                                                                                           |
|  | 循环界限          | 此符号分为两个部分，分别表示循环的开始和结束。例如<br><br>循环名<br>处理<br>终止条件 |

对于一个大系统很难将总流程图绘制出来，通常可以按照分系统、子系统、功能模块逐一绘制。

6. 确定和评价系统方案

开发人员在绘制系统流程图后，可从技术操作和经济几方面考虑，提出几种可供选择的方案。系统方案可根据系统平台类型、规模大小、投资强度等方面要求制定，一般可提出2~3种方案供用户选择。在可行性分析阶段给出的系统方案只是一个初步方案，在需求分析阶段制定的系统方案为更具体的可操作方案。

1.2.2 需求分析的过程

需求分析大体分成以下四个阶段：对问题的识别、分析与综合、制定规格说明和评审。

1. 问题识别

首先系统开发人员要研究可行性分析报告，主要是从系统的角度来理解软件并评审用于产生计划估算的软件范围是否恰当。确定对目标系统的综合需求。并提出这些需求实现条件，以及需求应达到的目标。这些需求包括：

- 功能需求：列举出所开发软件功能上应做什么。这是最主要的需求。

- 性能需求：给出所开发软件的技术性能指标。如存储容量限制、运行时间限制、安全保密性等。
- 环境需求：对软件系统运行时所处环境的要求。在硬件方面，应采用何种机型、外部设备和通信接口等等。在软件方面，采用哪个公司的系统软件（操作系统、网络软件、数据库管理系统等）。在使用方面还需要使用部门在制度上、操作人员的技术水平上应具备什么样的条件等等。
- 可靠性需求：在需求分析时应对所开发软件在投入运行后不发生故障的概率，按实际的运行环境提出要求。对于那些重要的软件应当提出高可靠性要求，以便在开发的过程中采取必要的措施，使软件产品能够高度可靠地稳定运行，避免因运行事故而造成损失。
- 安全保密要求：有些软件对其安全、保密的要求显然是不同的。应当把这方面的需求恰当地做出规定，以便对软件给予特殊的设计，使其在运行中其安全保密方面的性能得到必要的保证。
- 用户界面需求：软件与用户界面的易操作性是用户能够方便地使用该软件的关键之一。从市场角度来看，具有友好用户界面的软件有很强的竞争力。因此，在需求分析时，为用户界面进行细致地设计是非常必要的。
- 资源需求：指软件运行时所需的数据、软件、内存空间等各项资源需求。
- 成本消耗与开发进度需求：在软件项目立项后，要根据合同规定，对软件开发的进度和各步骤的费用提出要求，以便开发管理。

## 2. 分析与综合

需求分析的第二步是问题分析和方案的综合。系统分析员需从数据流和数据结构出发，逐步细化所有的软件功能，找出系统各单元之间的联系、接口特性和设计上的限制，分析它们是否满足功能要求，是否合理。依据功能需求，性能需求，运行环境需求等，剔除其不合理的部分，增加所需要部分。最终综合成为整个系统的需求解决方案。

在这个步骤中，分析和综合过程是不断反复地进行。在对现行需求问题和期望的信息（如输入和输出需求）进行分析的基础上，系统分析员综合出一个或几个解决方案，然后检查是否符合软件计划中规定的需求范围，再进行修改等等。总之，对问题进行分析和综合的过程将一直持续到系统分析员与用户双方都感到有把握正确地制定该软件的规格说明为止。

系统分析员及开发人员常用的分析方法有面向数据流的结构化分析方法（简称 SA）、面向数据结构的 Jackson 方法（简称 JSD）、面向对象的分析方法（简称 OOA）等，以及用于建立动态模型的状态迁移图或 Petri 网等。这些方法都采用图文结合的方式，可以直观地描述软件的逻辑模型。

## 3. 编制需求分析的文档

需求的文档又称为软件需求规格说明书。为了确切表达用户对软件的输入输出要求，还需要制定数据要求说明书及编写初步的用户手册，着重反映被开发软件的用户界面和用户使用的具体要求。

此外，根据在需求分析阶段对系统的进一步分析，从系统的精细模型出发，可以更准确地估计所开发项目的成本与进度，从而进一步修改、完善与确定软件开发实施计划。



#### 4. 需求分析评审

作为需求分析的最后一步，应该对系统功能的正确性、完整性，以及其他需求给予评价。评审的主要内容应是：

- 系统的定义与用户要求的一致性；
- 系统需求分析阶段提供的文档资料要齐全；
- 文档中的所有描述应完整、清晰、准确地反映用户要求；
- 所有其他系统单元的重要接口应有描述；
- 所开发项目的数据流与数据结构是否足够确定；
- 所有图表清楚，在不补充说明时便于理解；
- 主要功能已包括在规定的软件范围之内，是否都已充分说明；
- 设计的约束条件或限制条件要符合实际；
- 开发的技术风险要有描述；
- 最好能提供多个软件需求方案；
- 能考虑到将来可能会提出的软件需求；
- 应详细制定检验标准；
- 用户是否审查了初步的用户手册。

为确保软件需求定义的质量，评审最好指定专门的技术人员负责，并按规程严格进行。除系统分析员之外，用户和开发人员都应当参加评审工作。一旦评审的结果包括了一些修改意见，待修改完成后再经评审，通过后，方可进入设计阶段。

### 1.2.3 数据及数据库需求分析

在数据词典中，强调的是对数据存储结构的逻辑设计，并用数据结构表达数据项之间的逻辑关系。但是，这种结果不能满足需求分析阶段的要求。任何一个软件系统都可能有成千上万个数据项，仅仅描述这些数据项是不够的，更重要的是如何把它们以最优的方式组织起来，以满足系统对数据的要求。

#### 1. 有关数据库的基本概念

在软件中需要处理的数据是客观世界中存在的事物及其联系的反映。客观世界中的事物可分成对象和性质两大类。对象可以是人、是物，还可以是实际的东西或概念的东西，例如，大学、城市等。对象还可以指事物与事物间的联系。性质则是指事物的性质或特征。通常，把客观世界中的事物叫做实体，实体是由若干属性的属性值组成。属性是实体某一方面的特征，相应于事物的性质。例如，一个学生实体是一个 5 元组：（200088，李芳，女，19，软件）。5 元组中每一元素是学生的某一属性的属性值。他们对应的属性集合是：（学号，姓名，性别，年龄，专业）。这些属性的集合就表示了“学生”这个实体的类型，叫做实体型。同一类型的实体的集合叫做实体集。

在数据处理中用数据模型描述客观世界中的事物及其联系，便把描述每一实体的数据称为记录，把描述属性的数据叫做数据项或字段。因此，与实体集相对应的称为文件。例如，学生文件就由多个记录组成，这些记录放在一起构成一个二维表。表中每一横排叫做一个记录或元组，每一纵列叫做一个属性。

一个实体由若干属性的属性值组成，所以记录也是由数据项组成。一般数据项沿用属