第一章  
怎样理解整体大于部分之和？  
系统要素协同配合时，将发挥出好的作用和效益，这就是整体大于部分之和。  
当各个要素以有序、合理、优化的结构形成系统时，系统整体功能就会大于各  
部分功能之和。这是因为，系统整体是相互作用的各部分所构成的整体，整体  
的存在不仅依赖于部分的存在，而且依赖于各部分之间的相互性即内在联系，  
正是在部分之间的这种相互性和内在联系中产生了高效能的、有机的统一整体。  
最优化是系统方法所要达到的目标，它可以根据需要和可能为系统定量地确定  
出最优目标并运用最新技术手段和处理方法把整体系统逐阶分级，分成不同等  
级、层次结构，在动态中协调整体与部分的关系，使部分的功能和目标服从系  
统总体的最佳目标，以达到总体最佳。  
第二章  
信息系统的定义：信息系统就是输入数据，通过加工处理，产生信息的系统。  
在任何一个地方，只要有管理，就离不开信息系统。以计算机为基础的信息系  
统可以定义为：结合管理理论和方法，应用信息技术解决管理问题，为管理决  
策提供支持的系统。  
信息系统的基本功能：具有数据的输入、传输、存储、处理、输出等基本功能。  
信息系统的物理结构：集中式（单机系统） / 分布式（文件服务器 F/S、客户  
机-服务器 C/S、浏览器-服务器 B/S）  
集中式系统：是将软件、数据和主要外部设备集中在一套计算机系统中，采用  
集中计算模式。单机系统是典型的集中式系统。  
分布式系统：通过计算机网络把不同地点的计算机硬件、软件、数据等资源联  
系在一起，共同服务于一个目标。实现不同地点的资源共享，是这种系统的主  
要特征。  
缩写：  
EDPS (electronic data processing system)数据处理系统  
TPS（transaction processing system）事物处理系统  
MIS（Management information system）管理信息系统  
DSS(decision support system)决策支持系统 +group GDSS 群体决策支持系统  
EIS（executive information system）主管信息系统  
Data warehouse 数据仓库  
Data mining 数据挖掘  
Business intelligence BI 商务智能  
ERP (enterprise resource planning)企业资源计划  
EC (electronic commerce)电子商务  
CRM(customer relationship management)客户关系管理  
CMS（content management system）内容管理系统  
SCM （supply chain management） 供应链管理  
描述未来信息系统的发展趋势？  
阿里云 分布式 人工智能平台 内容分发  
ABCD AI Blockchain Cloud Data  
第三章  
为什么说信息系统建设是复杂的社会过程？  
复杂性：技术手段复杂、内容复杂，目标多样，投资大，周期长，效益难以计  
算；环境复杂多变；思维方式不同。  
社会过程:将信息系统建设与一般技术工程比较，信息系统建设的困难不仅来自  
技术方面，还来自企业内部环境。比如：体制、政策、法规、概念、技术等多  
种因素。  
建设系统不单单是一个人的事，是关于整个团队的，要做许多的分析调研，工  
具开发，要通过很多建模工具来设计分析。  
信息系统的生命周期（信息系统建设的基本步骤）  
系统规划（可行性分析报告、系统设计任务书）、分析（系统说明书）、设计  
(系统设计说明书)、实施（实施进度和状态报告、系统测试分析报告）、运行  
与维护（记录系统运行的情况，定期整理成新需求建议书。为下一周期的系统  
规划做准备）  
常用信息系统模型  
业务流程图——不同职能部门业务活动分工和活动的过程  
数据流图——数据的产生、处理、存储和去向的信息处理模型 系统分析阶段  
程序流程图——程序完成顺序、分支、循环  
（ER 图）实体关系图——有价值的实体及其关系 系统设计阶段  
组织结构图——部门及其从属关系  
模块结构图——功能模块，调用关系  
判定表、判定树——决策条件及其行动关系  
UML——软件系统结构及行为  
甘特图——项目任务以及其完成日期的项目计划  
对系统进行概要描述：业务流程、数据流图  
数据部分详细描述：数据字典  
基于生命周期的开发方法：瀑布 原型 迭代 螺旋 敏捷  
基于开发技术的开发方法：结构化（自顶向下分析与设计系统） 面向对象(把  
数据和对数据的操作紧密结合在一起：现实世界由对象组成；对象由属性和方  
法组成) 面向服务（将程序进行封装）  
它们之间的联系 区别  
结构化  
优点：从系统整体出发，强调整体优化的条件下“自上而下”地分析与设计，  
保证了系统地整体性和目标的一致性；遵循用户至上原则，严格区分系统开发  
的阶段性；每一阶段的工作成果是下一阶段地依据，整体性较好，便于系统开  
发地管理与控制，是其他系统开发（面向对象）的基础。  
缺点：开发周期长  
适用范围：规模较大、结构化程度较高  
面向对象  
优点：实现软件复用，简化程序设计，有利于维护，缩短开发周期  
缺点：对开发人员要求高  
使用范围：适用面很广  
面向服务  
优点：松耦合、自描述、可重用，可以将已经有的服务进行重新重组后可以快  
速建立新的业务流程。是以结构化和面向对象为基础。  
额  
什么是 CASE？  
在系统开发生命周期各个阶段的帮助开发者提高工作质量和效率的一类软件。  
工具：分析与设计（Visio）、编程工具(Eclipse, Visual Studio 2010)、测  
试工具、运行维护工具、项目管理工具。  
第四章  
系统规划步骤：企业调查和分析、信息系统调查和分析、制定方向、评审及建  
议。  
第五章  
系统分析基本任务  
系统分析员与用户在一起，充分了解用户的要求，并把双方的理解用系统说明  
书表达出来。系统说明书审核通过之后，将成为系统设计的依据，也是将来验  
收系统的依据。  
第六章  
数据流程图 由 外部实体、数据处理、数据流、数据存储等基本成分组成。  
数据流程图的注意事项 P121  
数据字典种类：数据流、数据存储、处理过程、外部实体。  
业务规则的描述工具有结构化语言、判定树、判定表。P132  
第七章  
什么是用例？  
在面向对象方法中，用例描述系统需求。  
用例创始人雅各布森 认为用例是对于一组动作序列的描述，系统执行这些动作  
会对特定的参与者产生可观测的、有价值的结果。  
代表人物对上述用例概念进行扩展，强调用例是各种系统受益人之间的一种行  
为契约。  
用例功能：  
对系统需求进行规范化的描述，用例模型是面向对象分析的关键输入。  
用例图及其事件流描述集中体现了系统责任。  
通过用例交互图可以严格的对照检查系统的每一项功能需求是否得到满足，需  
求都落实到哪些对象，以及这些对象是如何来解决的。  
用例包括：用例名称、参与者、前/后置条件、主事件/备选事件流、非功能性  
需求、业务规则。  
第八章  
面向对象方法三个阶段 ：面向对象编程/设计/分析  
概念  
对象、类、封装、消息、继承和多态性、关系。