

实验二 工资支付系统的需求分析建模

姓名： 高树林

班级： 2020185

学号： 202018526

实验学时：2（必修）

一、目的与任务

目的：确定项目要做什么及其可实施性，在此基础上完成系统的逻辑功能模型的建立。

任务：可采用不同的需求分析技术，完成对项目的需求分析过程，给出系统的逻辑功能模型，数据字典等

二、内容

1、实验内容：

熟悉系统项目的业务流程，根据现有分析资料绘制系统数据流图，IPO图，编写数据字典，数据加工处理的描述。

2、实验结果：

名字：工资总额 别字：总工资 描述：扣除个税，公积金和保险之前一个教职工的月工资 格式：数，最大值=9999.99 位置：工资明细表
--

图1 工资支付系统的数据字典（工资总额）

名字：个人所得税 别名：个税，所得税 描述：政府本月征收的个人收入所得税 格式：数，最大值=9999.99 位置：工资明细表
--

图2 工资支付系统的数据字典（个人所得税）

IPO 表	
系统：工资支付	作者：高树林
模块：工资总额算法	日期：2023.05.16
编号：	
被调用：	调用：
输入：基本工资 课时费、岗位津 贴、生活补贴、书 报费、交通费洗理 费	输出：工资总额
处理：工资总额基本工资+课时费+岗位津贴+生活补贴+书 报费+交通费+洗理费	
局部元素：	注释： 教室岗位津贴为 0

图 3 描述工资总额初步算法的 IPO 图

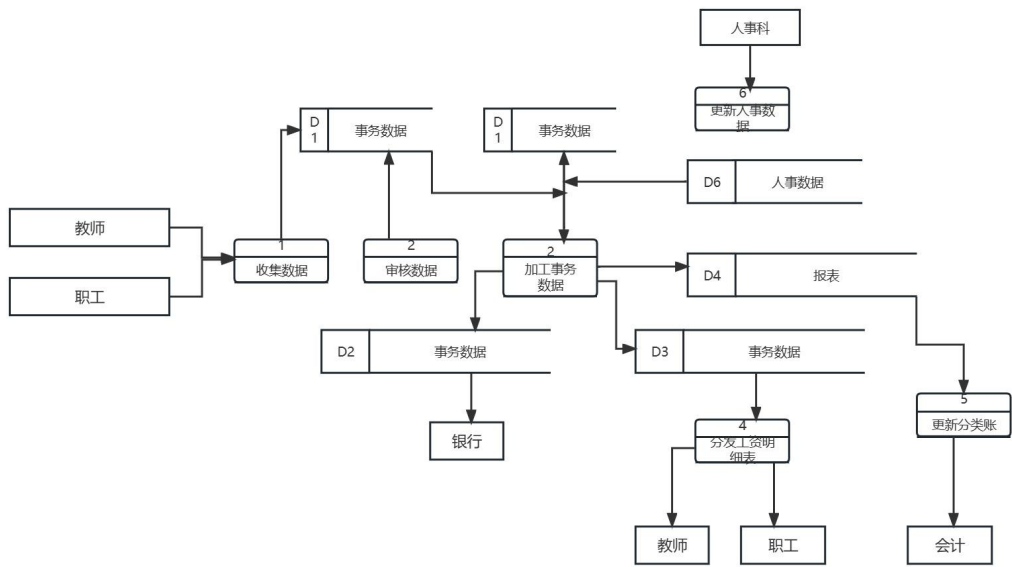


图 4 工资支付系统数据流图（更新）

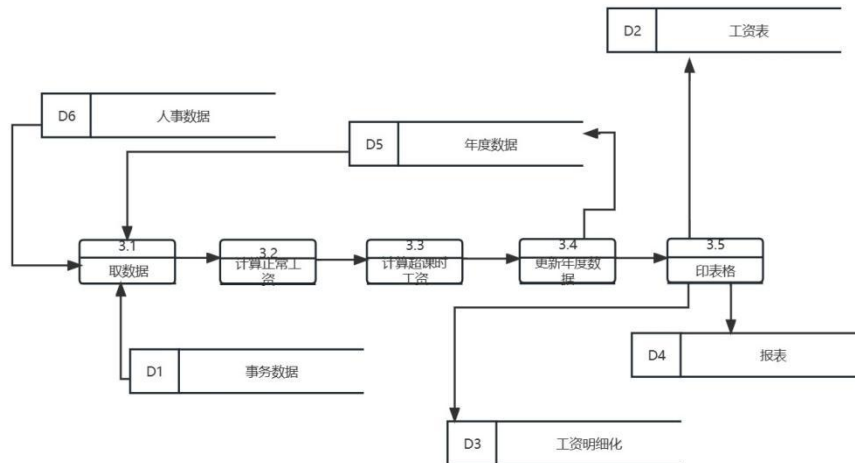


图 5 对“加工事务数据”的细化数据流图

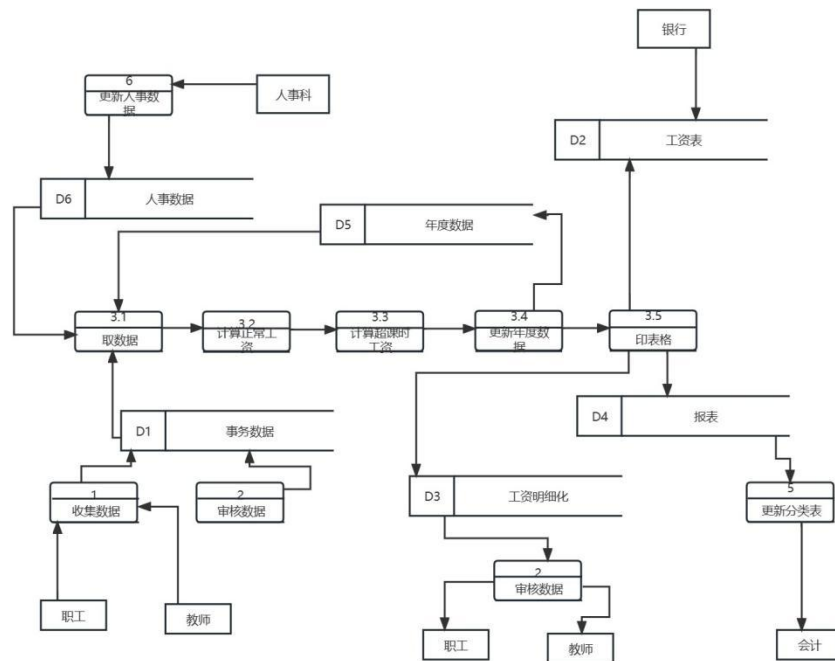


图 6 工资支付系统完整数据流图

三、问答题

1、分析系统流程图，数据流图之间的差别和各自的特点。

系统流程图和数据流图是两种常用的图形工具，用于分析和描述系统或过程的不同方面。它们之间存在一些差别和特点，如下所述：

系统流程图：

定义：系统流程图是一种图形化表示系统组件和其交互关系的工具。它用于显示系统中

各个模块、子系统或部件之间的流程、控制和数据传递。

用途：系统流程图通常用于展示系统的整体结构和各个部分之间的相互作用。它显示了系统的主要功能、处理步骤、条件分支、循环等。

图形元素：系统流程图使用了不同的符号来表示不同的系统组件和交互元素，如矩形框表示模块或子系统，箭头表示数据流和控制流，菱形表示条件或判断，圆角矩形表示数据存储等。

优点：系统流程图具有易于理解和解释的优势，可以清晰地展示系统的流程和结构，便于开发人员和用户之间的沟通和理解。

局限性：系统流程图的绘制和修改可能需要一定的时间和精力，对于复杂的系统，图形可能变得复杂难懂。

数据流图：

定义：数据流图是一种图形化的工具，用于显示系统或过程中数据的流动路径和处理过程。它着重于数据的流动和处理，并将系统看作是一系列相互关联的数据流和处理功能。

用途：数据流图通常用于分析系统或过程的数据流动和转换。它显示了数据流的来源、流向、处理和存储，以及系统中的不同角色和功能。

图形元素：数据流图使用了一些基本符号，如箭头表示数据流，圆圈表示处理功能，矩形表示数据存储，菱形表示数据转换等。DFD 一般分为不同层次，从上层到下层逐步展开细节。

优点：数据流图能够清晰地表示数据的流向和处理过程，使分析人员能够更好地理解系统的数据要求、转换和存储，有助于识别问题和改进系统。

局限性：数据流图不太适合展示系统的顺序和控制流程，较少关注时间和顺序。对于复杂系统，DFD 可能需要多层次的分解和详细说明。

列成表格如下表 1 所示：

表 1 系统流程图和数据流程图的分析

	系统流程图	数据流图
定义	描述系统组件和交互关系的图形化工具	显示数据流动和处理的图形化工具
用途	展示系统的整体结构和部分之间的相互作用	分析系统的数据流动和转换
图形元素	模块/子系统、箭头、条件判断、数据存储等	数据流、处理功能、数据存储等
优点	易于理解和解释	清晰表示数据流向和处理过程
局限性	图形可能复杂难懂	不太适合展示顺序和控制流程

2、数据流图应该从哪些方面修改完善？

要完善和修改数据流图，可以从以下几个方面入手：

系统边界和外部实体：检查数据流图的系统边界是否正确定义，并确保涵盖了系统与外部实体之间的数据流动。如果有遗漏或不准确的部分，需要进行修正。

数据流：审查数据流是否完整、准确地描述了系统中的数据流动路径。确认数据流是否正确地传递数据元素，并且数据流的命名清晰、一致。如果发现缺失的数据流或者命名混乱的情况，需要进行修正。

数据存储：检查数据流图中的数据存储是否恰当地表示了数据的存储和持久化。确认数据存储的命名和描述是否清楚，数据存储之间的关系是否准确。如果需要添加、修改或删除数据存储，应进行相应的修改。

进程和功能：审查进程和功能模块是否正确地表示了系统中的处理和计算过程。确认进程的命名是否准确，每个进程是否包含了合适的功能和计算步骤。如果发现功能模块缺失、命名不当或者包含冗余功能，需要进行相应的修改和优化。

控制流和数据流方向：分析数据流图中的控制流和数据流的方向，确保正确反映了系统中的控制逻辑和数据流动方向。检查是否存在混淆或错误的数据流方向，以及缺失的控制流。必要时，进行调整和修正。

数据处理逻辑：分析每个进程中的数据处理逻辑，确认数据的转换、计算和验证过程是否正确表示。检查是否需要添加或修改数据处理步骤，以确保数据在系统中得到正确处理。

细化和扩展：根据实际需求，对数据流图进行细化和扩展，以更详细地描述系统中的数据流动和处理过程。可以添加更多的数据流、数据存储、进程和功能模块，以便更全面地表示系统的功能和业务流程。

通过对数据流图进行上述方面的完善和修改，可以提高数据流图的准确性、清晰性和可读性，从而更好地理解和分析系统的数据流动和处理过程。