

数据库系统实现作业说明

一、引言

数据库管理系统软件是计算机中除操作系统之外的一个重要系统软件，数据库管理系统是信息系统的基础和核心，数据库管理系统已成为整个信息基础设施的核心支撑软件。随着 Internet 及其应用技术的发展，信息管理的内容、形式和规模不断扩展与变化，数据库技术面临着前所未有的挑战。数据库领域的研究是当前信息领域中最为热点的研究领域之一。

要深入掌握数据库系统的原理和技术，进而从事数据库管理软件和工具的开发，必须学习和研究数据库管理系统实现技术。要深入了解数据库系统的内部结构，以开发出高效的数据库应用系统，也需要学习和研究数据库管理系统实现技术。

为培养具有创新意识的一流人才，必须重视实践。目前学院已开设的“数据库原理及其应用”课程是一门概念性和实践性都很强的专业基础课程，通过该课程的学习，学生可以具备数据库管理系统基本概念、数据库应用设计的系统方法，同时学会商用数据库系统的操作使用，掌握应用 SQL 语言进行程序设计的开发方法。有一个现象：许多学生：“我很喜欢数据库，因为我觉得我学得比其它编程语言好（如 C++），也增强了学习的信心”。说明同学们对系统软件的认识很薄弱，殊不知数据库管理系统是两大系统软件之一，涉及计算机知识面大而广且深，除了编程语言，还包括“数据库管理系统”、“离散数学”、“算法设计与分析”、“数据结构”、“编译原理”、“软件工程”、“计算机网络”、“计算机组成”等软硬件知识的综合应用。

课程教学遵从实践、认识、再实践、再认识的认知理论，“不闻不若闻之，闻之不若见之，见之不若知之，知之不若行之。…，故闻之而不见，虽博必谬；见之而不知，虽识必妄；知之而不行，虽敦必困。”（《荀子·儒效》）。因此，十分重视实践环节。

本课程属于高年级专业选修课，必须加强培养独立动手实践能力，非验证性实践。不同于验证、操作类实验，系统实现类实验对学生能力要求较高、难度大，不是简单 2 节课可以完成的，需要学生有一个事先预习，事先准备和实现的过程。在实验课中，实际上是用于回答实验中遇到的一些关键问题、困惑现象的分析。尽管对每类实验给出实验目的、内容、步骤，但这显然不符合系统实现类实验。这类实验应当是启发式的、探索式的。其特点是：

- （1） 综合性：实践内容具有将前期所学课程知识综合应用的特点，不是某门课程单点知识的简单重复，而是涉及多门课程知识的深入、综合应用，既有广度又有深度。
- （2） 实践性：顾名思义，“数据库系统实现”不仅涉及理论更涉及技术，属于实践性很强的课程，注重学生的动手研发与创新能力的培养。
- （3） 探索性：由于数据管理的复杂性、数据类型的多样性，有许多理论与技术有待探索，因此本身具有探索性、科研性、先进性，如查询处理算法。
- （4） 专业性：这方面的训练不是其它专业可以具备的，能够突出专业特点——

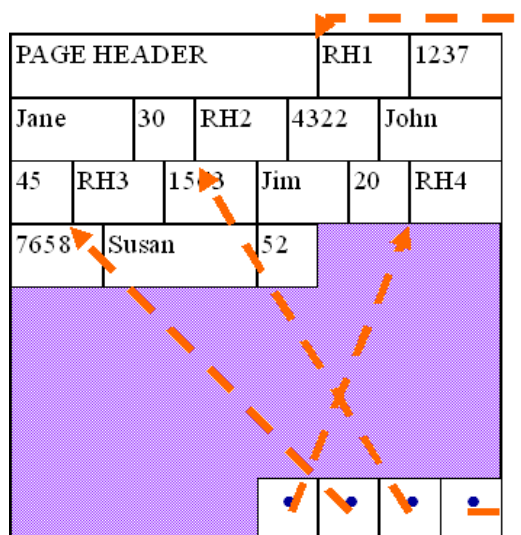
“专门性”，增强学生竞争力和自信心，也能够培养学生对专业的深层理解而不流于“专业趋同”的时弊。因此这个特性也是最重要的。

二、作业说明

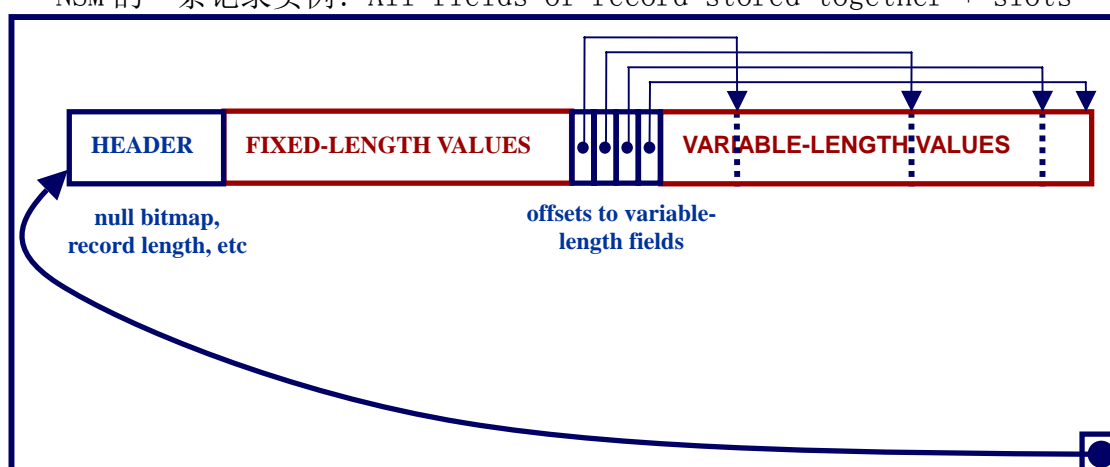
根据课程性质，课程布置大型实践作业。学生 3-5 人分组进行工作，以小组为单位，选择实现数据库系统中的某一个部件，分析各实现方案优劣。形成完整的软件文档说明。

阅读文献[2]，了解数据库系统中磁盘页面布局的多种方法，了解其优缺点。并进行实现。几个磁盘页面布局的说明如下：

NSM (N-ary Storage Model): Traditionally, database systems use the N-ary storage model (NSM), a page-based storage layout in which tuples (or rows) are stored contiguously in pages. NSM may waste disk and memory bandwidth if only a small fraction of each row is needed.



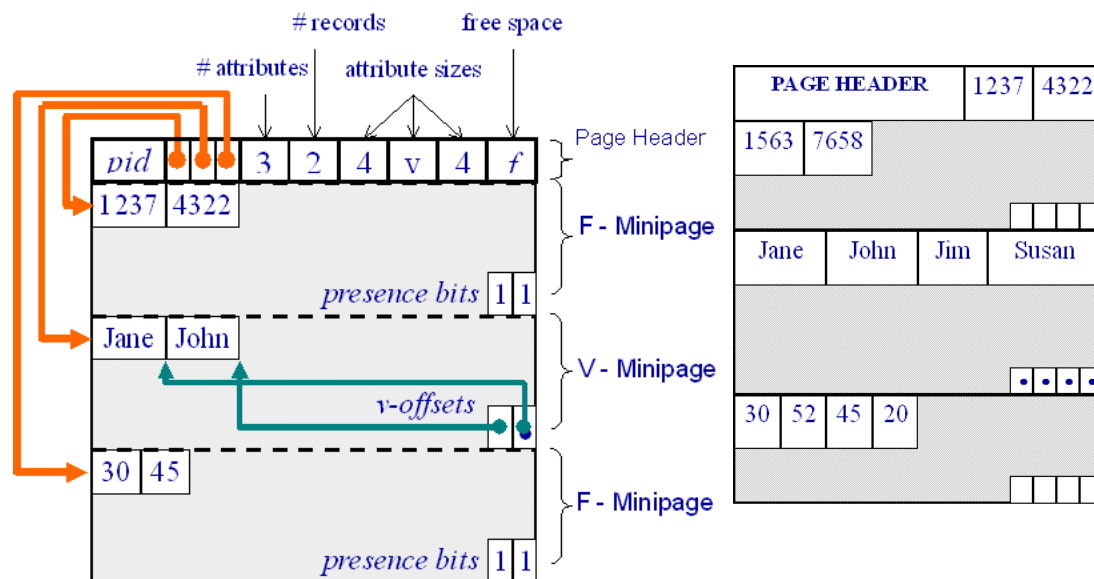
NSM 的一条记录实例: All fields of record stored together + slots



PAX (Partition Attributes Across) is a hybrid approach, essentially a DSM-like organization within an NSM page. While the disk access pattern of PAX is indistinguishable from that of NSM, it improves on the memory bandwidth requirements. On SSDs, PAX, in combination with the much faster seek time, allows reading only those columns needed by the query, essentially enjoying the read efficiency of DSM while retaining existing NSM functionality.

PAX 的详细设计:

PAX: Group fields + amortizes record headers



有兴趣的同学可以尝试 DSM:

DSM (Decomposition Storage Model) [1] proposed in the 80s, decomposes relations vertically, allocating one sub-relation per attribute. DSM had its own disadvantages, including storage overhead for storing tuple IDs and expensive tuple reconstruction costs. With changing market needs and more favorable technology trends, newer DSM-like (column-store) commercial products and academic prototypes have recently appeared (such as SybaseIQ, Vertica, C-store, and MonetDB/X100). These systems eliminate storage overhead through virtual IDs and offer fast scans of few attributes at the cost of additional disk seeks to fetch non-contiguous attributes.

[1] G. P. Copeland and S. N. Khoshafian. A decomposition storage model. In Proc. of SIGMOD Conf., p.268-279, 1985.

[2] A. Ailamaki, D. J. DeWitt, M. D. Hill, and M. Skounakis. Weaving relations for cache performance. In Proc. of VLDB Conf., p.169-180, 2001

[3] M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S.

O_ORDERSTATUS	CHAR(1) NOT NULL,
O_TOTALPRICE	DECIMAL(15,2) NOT NULL,
O_ORDERDATE	DATE NOT NULL,
O_ORDERPRIORITY	CHAR(15) NOT NULL,
O_CLERK	CHAR(15) NOT NULL,
O_SHIPPRIORITY	INTEGER NOT NULL,
O_COMMENT	VARCHAR(79) NOT NULL);