

数据库系统课程设计

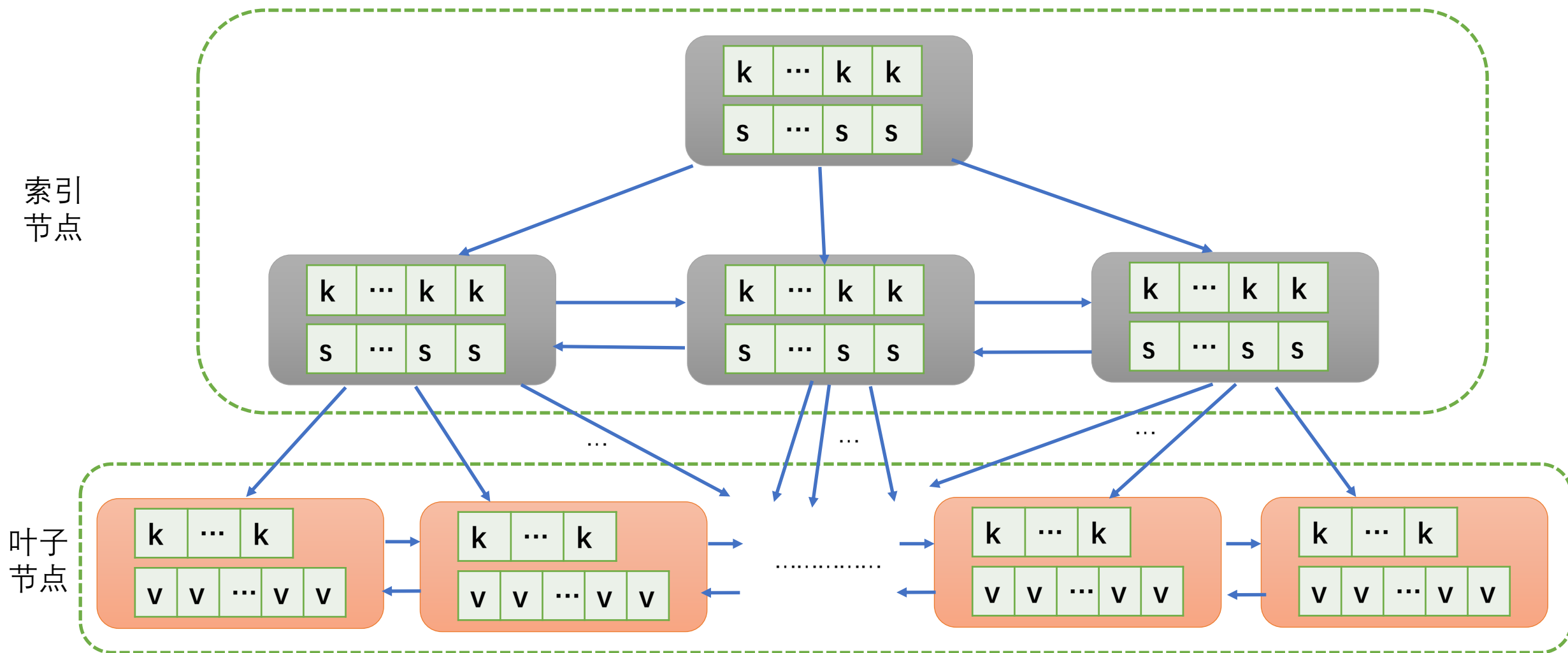
课程设计

- B+树bulkloading多核并行设计
 - 了解B+树的结构和bulkloading操作
 - 提供一份串行的B+实现bulkloading过程的代码，基于B+树实现并行化bulkloading操作。
 - 在单机版电脑用多核并行程序实现。

Bulkloading说明

- Bulkloading
 - 对有序的数据进行批量构建B+树的过程。
- 结合代码具体说明
 - 来源于近似最近邻检索算法QALSH的索引部分代码。
 - 数据形式：有序、键值对。
 - 批量连续插入的过程：自下往上
 - 先从底层叶子节点构建，从左往右按顺序构建一个双向链表。
 - 从下往上，一层层构建索引节点，每一层也是从左往右构建索引节点。

Bulkloading说明



Bulkloading说明

- 叶子节点
 - 从左往右构建一层双向链表。
 - 每个节点有两个数组，一个存储key，一个存储value，存储的key数量和value数量不相等，实际上是一个key对应16个value。（不影响并行程序设计，理解就好）
 - 相邻节点之间有互相指向的指针。
- 索引节点
 - 每一层从左往右根据下一层子结点构建一层双向链表。
 - 每个节点有两个数组，一个存储key，一个存储指向子节点的指针，数量相等。
 - 相邻节点之间有互相指向的指针。
- 与磁盘交互
 - 构建的B+树的每一个节点会依次存储到一个文件的不同区域中，每一个节点有一个block号，表示节点存储到文件的区域编号，上述所说的指针实际上是一个block号，表示节点存储在文件的位置。
 - 注意：串行bulkloading过程中，从底层往上，每一层从左往右的节点的block号是按顺序分配的，并行执行，可能会导致block号不连续，要考虑这种问题怎么解决，否则并行执行会出现错误。

编译运行

- Code
 - 环境: ubuntu
 - b_tree.h b_tree.cc: B+树的构建
 - b_node.h b_node.cc: B+树的节点
 - block_file.h block_file.cc: 磁盘交互的实现
 - main.cc: 调用的主程序
 - pri_queue、random、util、def: 辅助函数
 - Makefile: 编译运行, 先将各个代码文件生成可执行的.o文件, 然后再将所有的.o文件合并成一个可执行的文件run
 - 编译命令: make
 - 清除生成的.o文件: make clean
 - 运行命令: ./run

实验说明

- 实验说明

- 并行编程

- Pthread、MPI都可。推荐《并程序序设计导论》

- 数据集

- 提供一个有序键值对数据生成的代码文件make_data.cpp，自己生成不同数据量大小的数据，测试其并行设计在不同数据量下的加速效果。

- 查询检验

- 自己写一个遍历查询的函数，遍历输出每一层节点的数据，与串程序序相比，检验程序序设计是否正确。
 - 并行设计过程，多线程或者多进程同时执行构建不同的节点，有可能会出现中间的叶子节点数据不满的情况，可能导致叶子节点的数目有变化，可以结合实际情况分析是否正确，或者说这样的情况会有哪些好处或者坏处。

课程设计要求

- 课程设计要求
 - 理解B+树bulkloading操作过程的代码。
 - 掌握多核并行程序的设计。
 - 并行化实现
 - 实现底层叶子节点的并行构建。
 - 实现索引节点（非叶子节点）的并行构建。
 - 实验结果
 - 比较不同数据量的并行加速效果，将加速比绘制成折线图。

课程设计作业提交

- 作业代码。
- Readme文件，说明具体怎么运行代码，说明参数设置。
- 课程设计报告。
 - B+树bulkloading过程的理解（用自己语言写）
 - 算法并行的设计思路
 - 算法流程图
 - 关键代码描述
 - 实验结果
 - 实验分析
 - 性能调优、创新优化
- 实验心得。（小组每个同学写一份，合在一个文档中）
- 学者网提交作业。
- ddl: 2022年1月6日