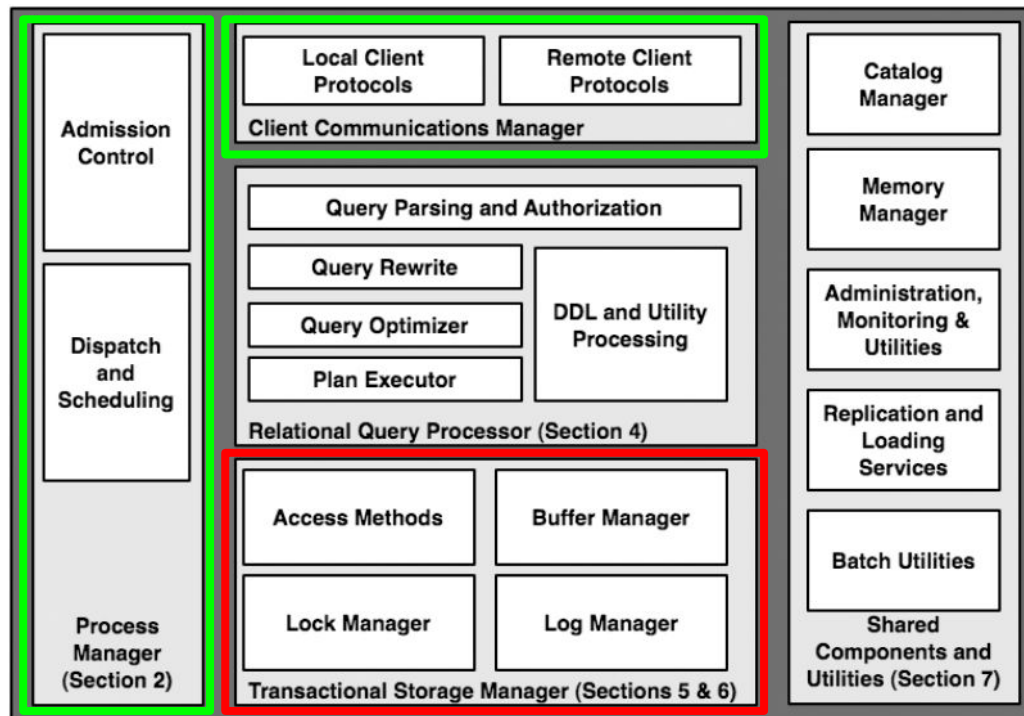


# Технологии и разработка СУБД

## Лекция 3. Хранение данных на диске

Анастасия Лубенникова  
Александр Алексеев

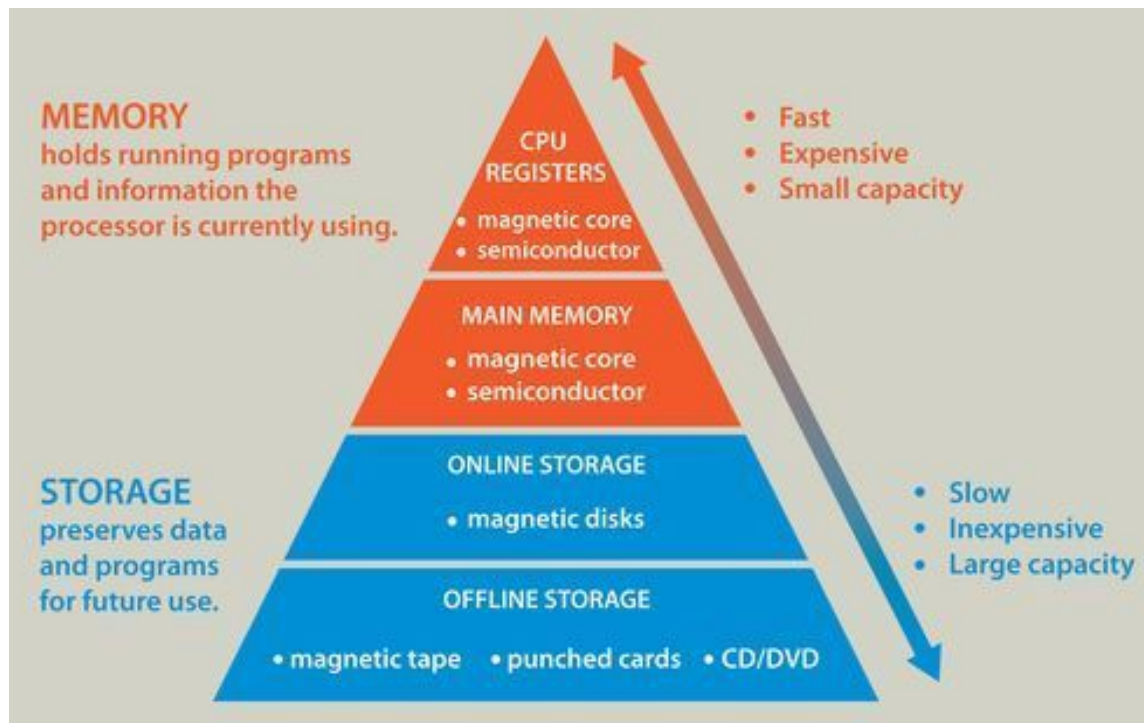
# После прошлого занятия



# Лекция 3

- Часть 1: Storage manager
- Часть 2: Buffer manager
- Часть 3: Lock manager

# Иерархия памяти



# Память и диски

- Иерархическая структура:
  - регистры - байты (  $< 0.5$  нс)
  - кэши - Kb, Mb (0.5 - 7 нс)
  - ОЗУ - Gb (100 нс)
  - жесткие диски - Tb ( $\sim 10\,000\,000$  нс)
- HDD vs SSD
  - SSD дороже, HDD дешевле
  - HDD: быстрый последовательный доступ, медленный произвольный
  - SSD: быстрый *произвольный* доступ

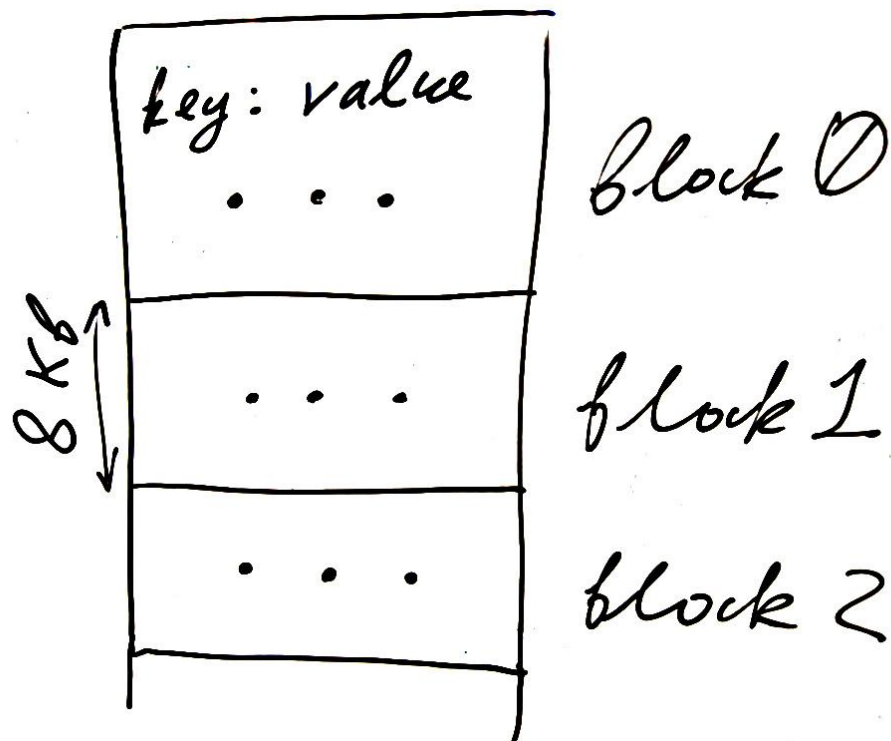
# Немного терминологии

- Block = Page
  - единица ввода/вывода (I/O)
  - типичный размер от 4K (в книгах) до нескольких Mb
- Relation (Table)
- Attribute (Column, Field)
- Tuple (Row, Record)

# Storage manager

- Нижние уровни реализуют логику работы с диском
  - чаще всего всё-таки не напрямую, а поверх файловой системы
  - одной таблице может соответствовать несколько файлов
  - readblock / writeblock / createfile / dropfile
  - опционально: сжатие, шифрование, чексуммы, и тд
- Методы доступа (Access method)
  - структуры данных и алгоритмы работы с данными на диске
  - используют примитивы чтения и записи блоков и оперируют уже с логической структурой данных

Иллюстрация ;)





# Buffer Manager

- Buffer
  - область памяти, доступная для хранения копии дисковой страницы
- Для чего?
  - для минимизации количества операций ввода/вывода
  - кэш файловой системы не знает специфику нашей СУБД
  - hit ratio - процент удачных попаданий
- Как работает?
  - страницы с диска зачитываются в разделяемую память, когда они понадобятся
  - страницы вытесняются на диск по мере необходимости
- Варианты:
  - только кэш файловой системы
  - только кэш СУБД - Direct I/O
  - mmap / MapViewOfFile (например, так сделано в MongoDB)

# Buffer Manager: dirty pages

- Если мы что-то изменили на странице, нужно пометить буфер как dirty.
- Для чего?
  - чтобы при вытеснении мы знали, надо ли сбросить изменения на диск или просто переиспользовать буфер

# Стратегии замещения (вытеснения)

- Какой буфер вытеснить в первую очередь?
- Метрика успешности - “hit ratio”
- Варианты:
  - Random
  - FIFO
  - LRU
  - Clock algorithms

# Счётчик ссылок

- Reference count (pin counts)
  - как много транзакций сейчас используют эту страницу
  - если счетчик больше нуля - вытеснять страницу из буфера нельзя
- `pin_buffer` / `unpin_buffer`

# Lock manager

- для обеспечения конкурентной работы с данными
- lock\_buffer / unlock\_buffer
- shared lock, exclusive lock
- дополнительно: deadlock detection, отображение информации о локах пользователю и тд

# На следующей лекции

- Методы доступа
- “Как разместить данные на страницах?”

Тест! ;)



# Семинар



# Домашнее задание №2. Проверка

- Нужно было выбрать домашнее задание.

# Дополнительные материалы

- Числа, которые должен знать каждый программист  
<https://habrahabr.ru/post/108537/>
- Random notes on improving the Redis LRU algorithm  
<http://antirez.com/news/109>
- Википедия: SSD, HDD, LRU, etc

## Вопросы и ответы.

- [a.lubennikova@postgrespro.ru](mailto:a.lubennikova@postgrespro.ru)
- [a.alekseev@postgrespro.ru](mailto:a.alekseev@postgrespro.ru)
- Telegram: <https://t.me/dbmsdev>