

БЛОК 3. АРХИТЕКТУРА

БУФЕРНЫЙ КЕШ

A white curly brace symbol, commonly used to denote code or a code block.

{ }

A blue square button with the word "begin" in white lowercase letters.

begin



ЦЕЛЬ УРОКА

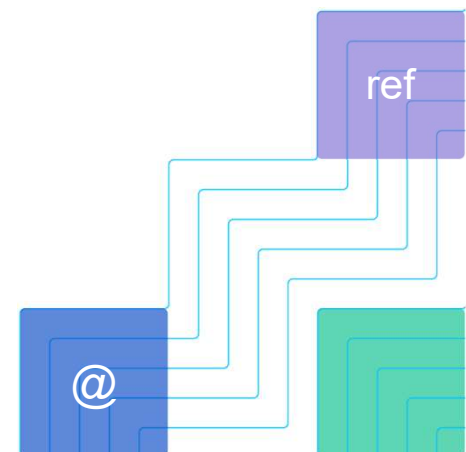


01

Узнать подробное устройство буферного кеша

02

Понять принцип его работы



СОДЕРЖАНИЕ УРОКА

1

Буферный кеш

2

Утилита pg_prewarm

3

Практика

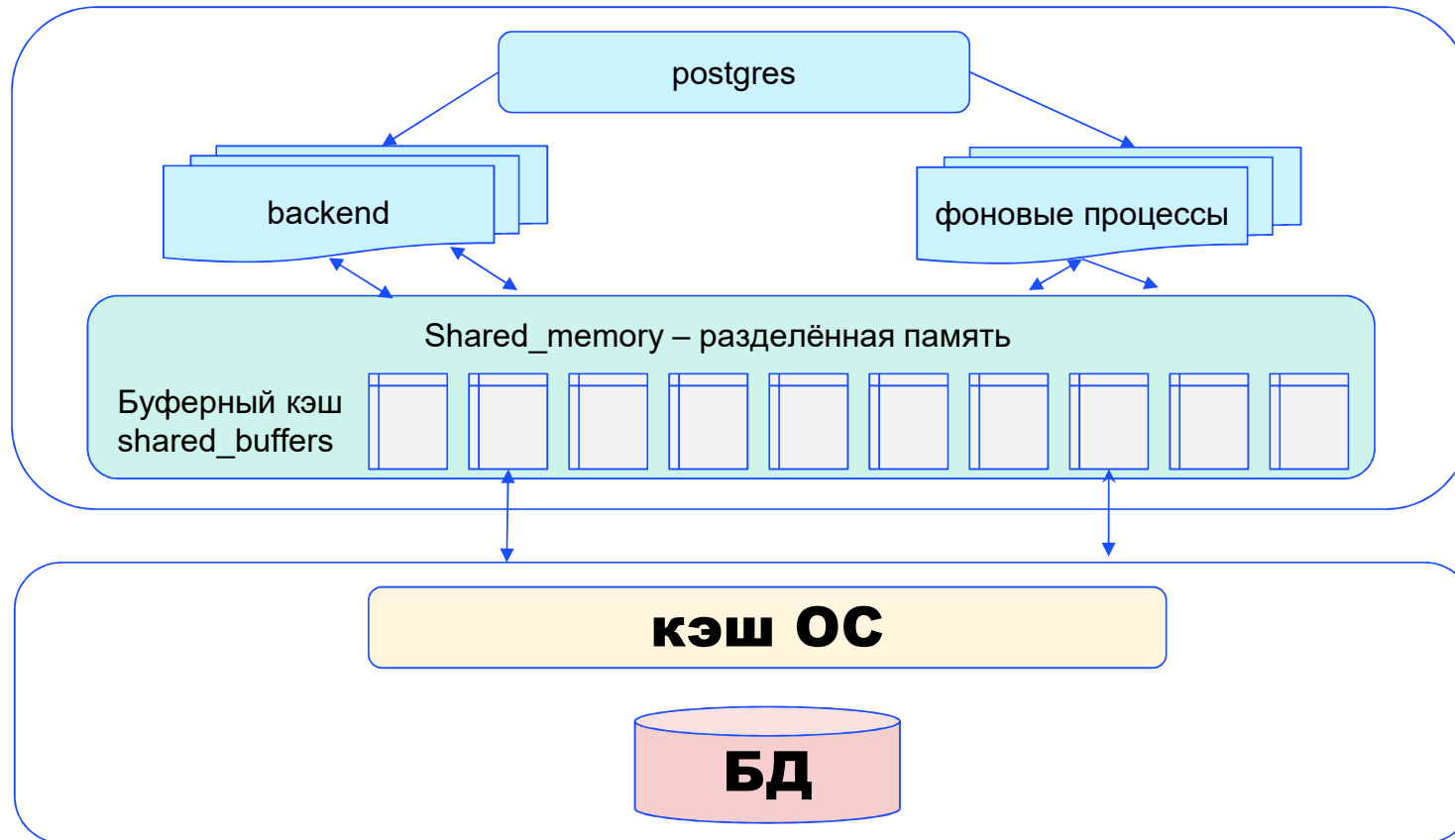


STEP
UP

ref

@

БУФЕРНЫЙ КЕШ



PostgreSQL

ОС

БУФЕРНЫЙ КЕШ. ЗАДАЧА

Ускоряем работу всей системы.

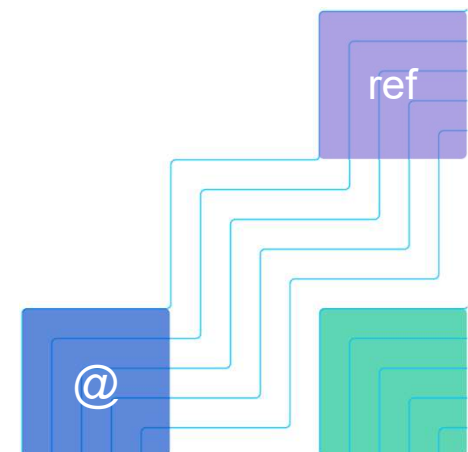
любой кеш нужен для того, чтобы сглаживать разницу в производительности двух типов памяти, одна из которых относительно быстрая, но ее на всех не хватает, а другая — относительно медленная, но имеющаяся в достатке. Вот и буферный кеш сглаживает разницу между временем доступа к оперативной памяти (наносекунды) и к дисковой (миллисекунды).

1

RAM, Оперативная
память очень быстра,
но ее мало.
Для часто-читаемых
данных.

2

HDD огромный, но медленный





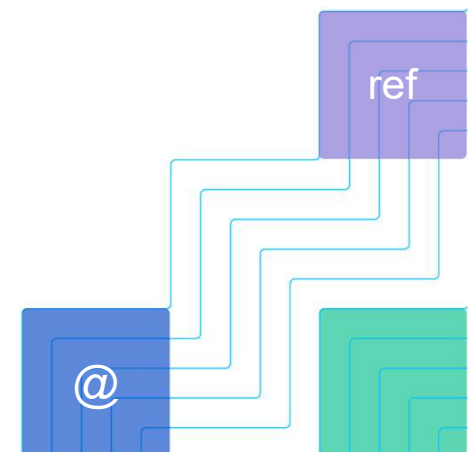
БУФЕРНЫЙ КЕШ. СОСТАВ



Каждый буфер состоит из одной Страницы данных + Заголовка. Header, по-умолчанию 8 кб, содержит:



- расположение страницы на диске (файл и номер страницы в нем),
- число обращений к буферу usage count (счетчик увеличивается каждый раз, когда процесс читает или изменяет буфер, максимально значение 5 (0-4)),
- признак того, что данные на странице изменились и рано или поздно должны быть записаны на диск (грязный буфер).





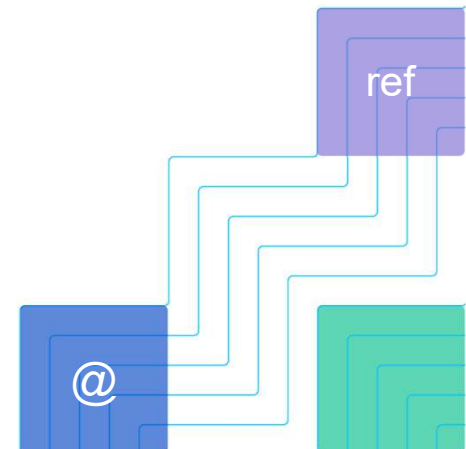
БУФЕРНЫЙ КЭШ. СОСТАВ



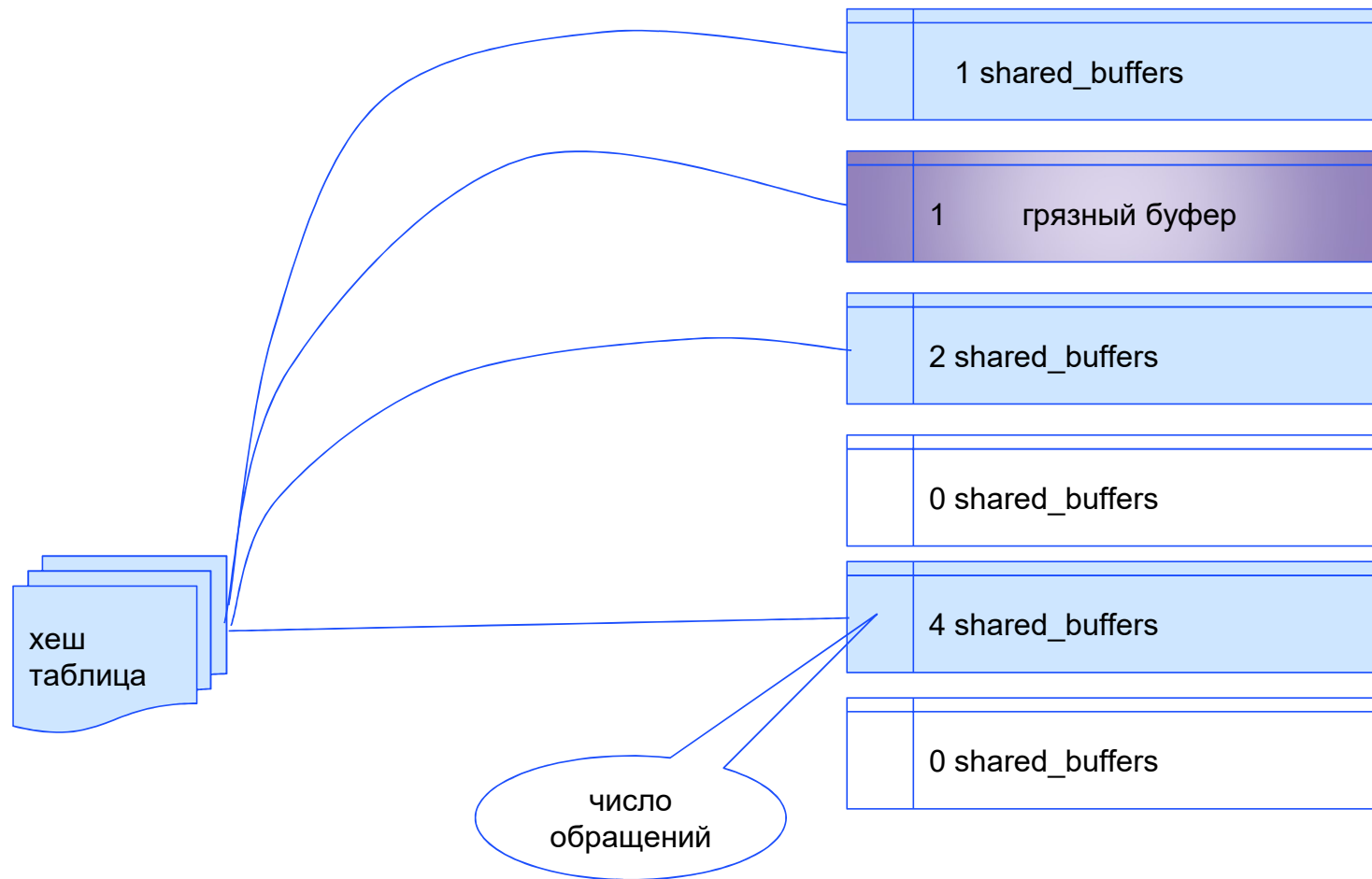
Изначально кэш содержит:



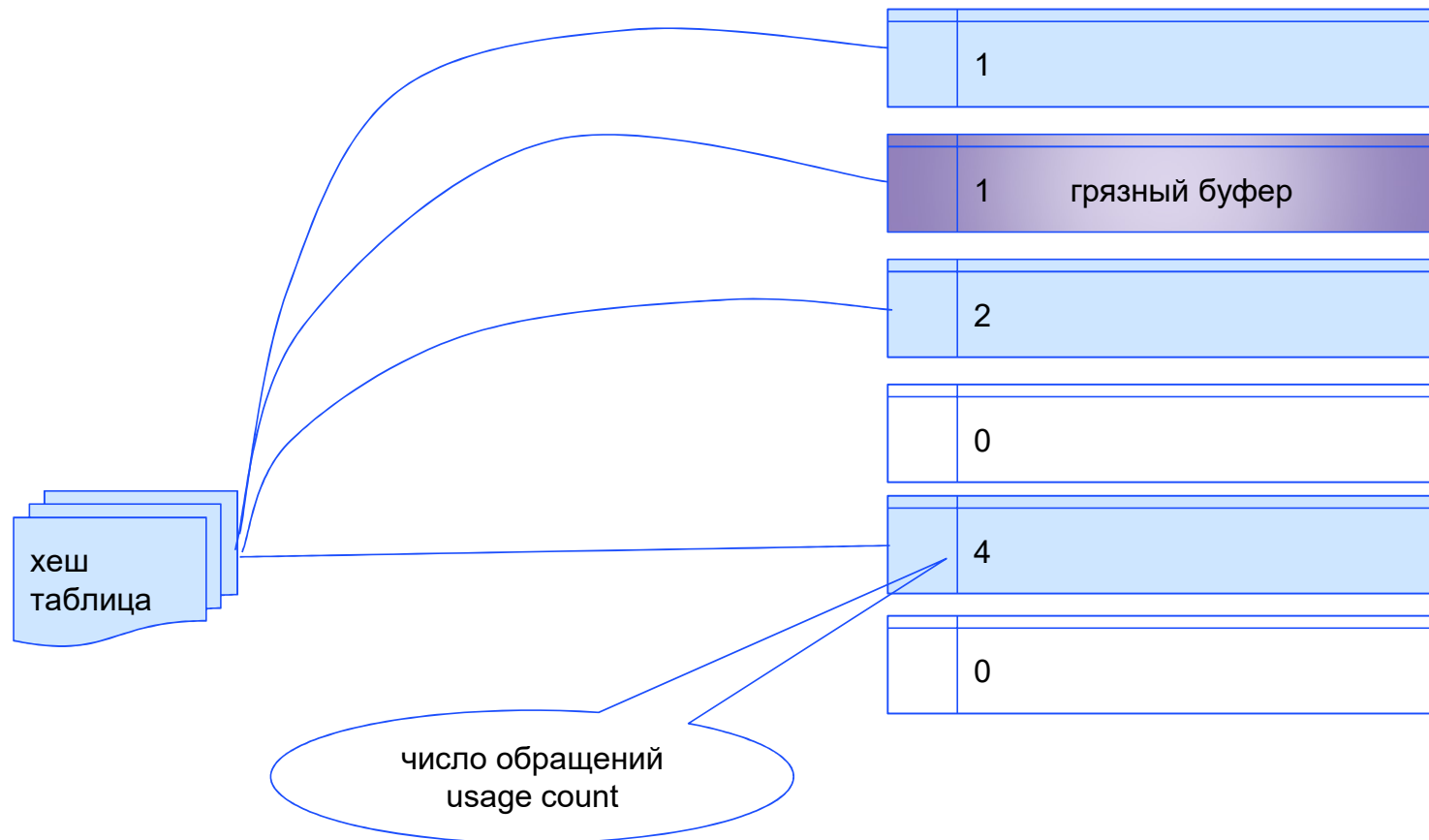
- пустые буферы, и все они связаны в список свободных буферов,
- указатель на «следующую жертву» при вытеснении старых буферов на HDD,
- также используется хеш-таблица, чтобы быстро находить нужную страницу в кэше.



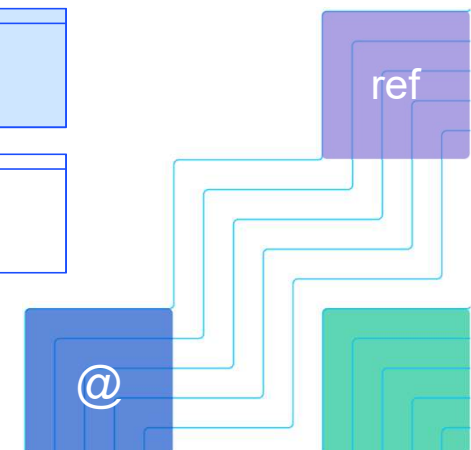
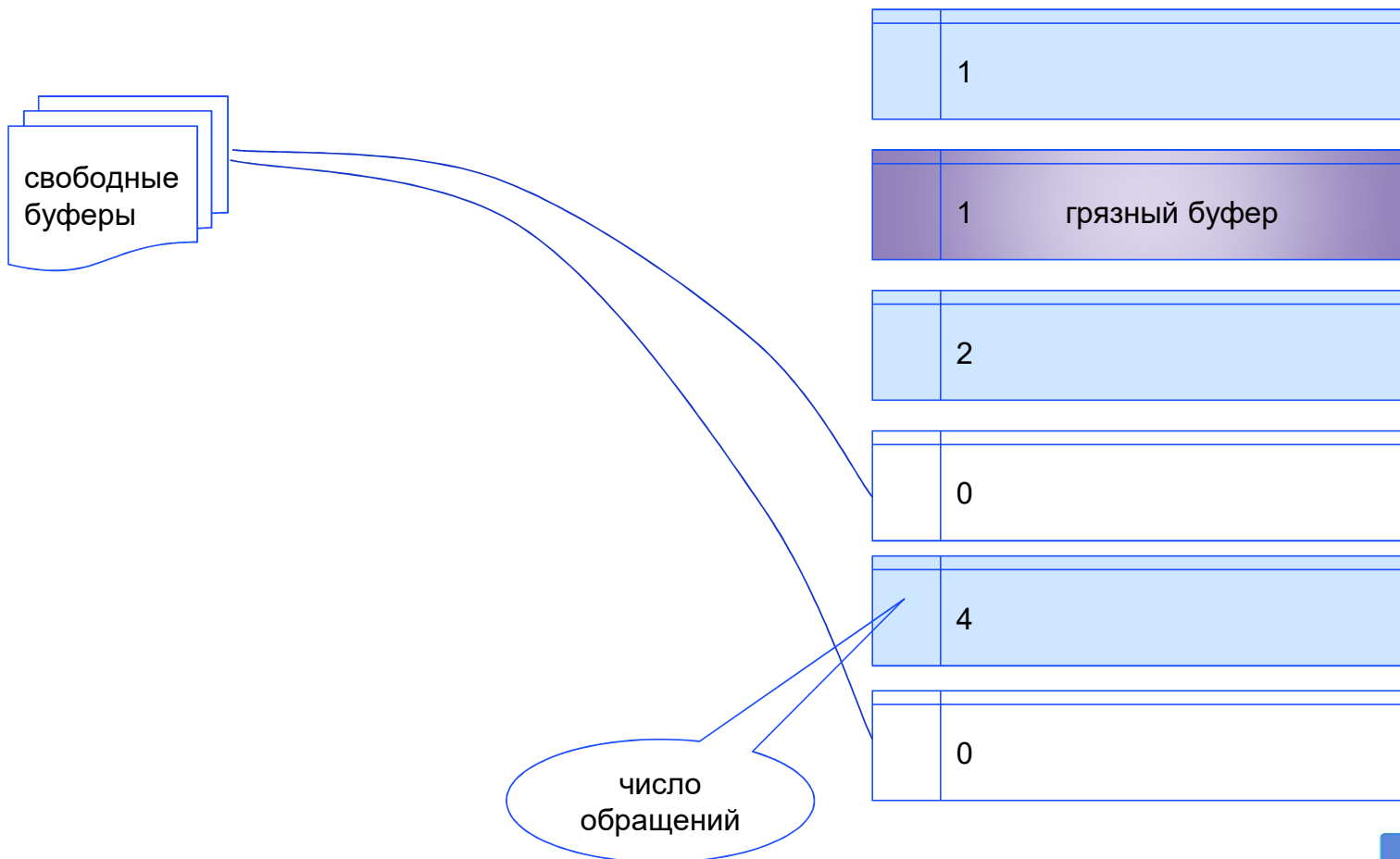
БУФЕРНЫЙ КЕШ. МЕХАНИЗМ РАБОТЫ



БУФЕРНЫЙ КЕШ. МЕХАНИЗМ РАБОТЫ



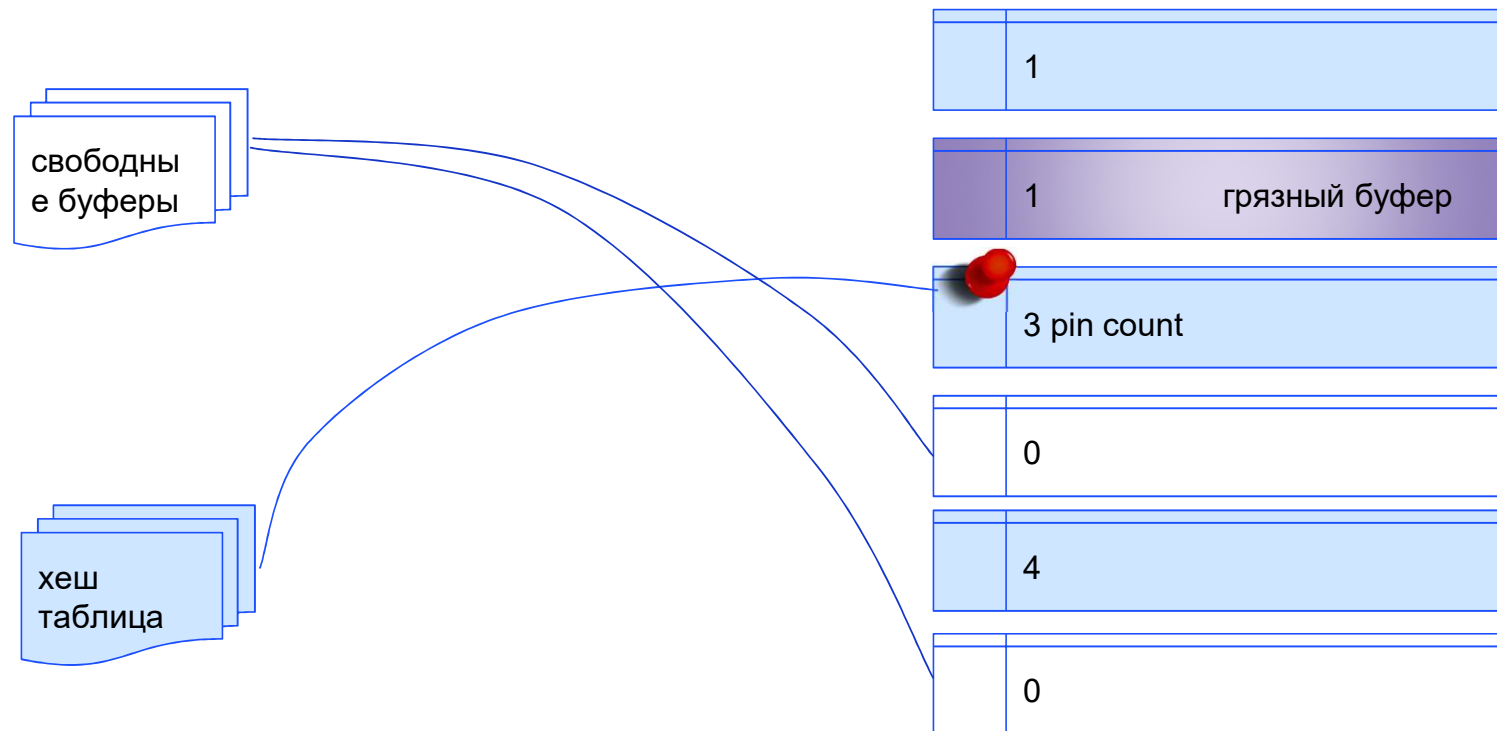
БУФЕРНЫЙ КЕШ. МЕХАНИЗМ РАБОТЫ



БУФЕРНЫЙ КЭШ. МЕХАНИЗМ РАБОТЫ

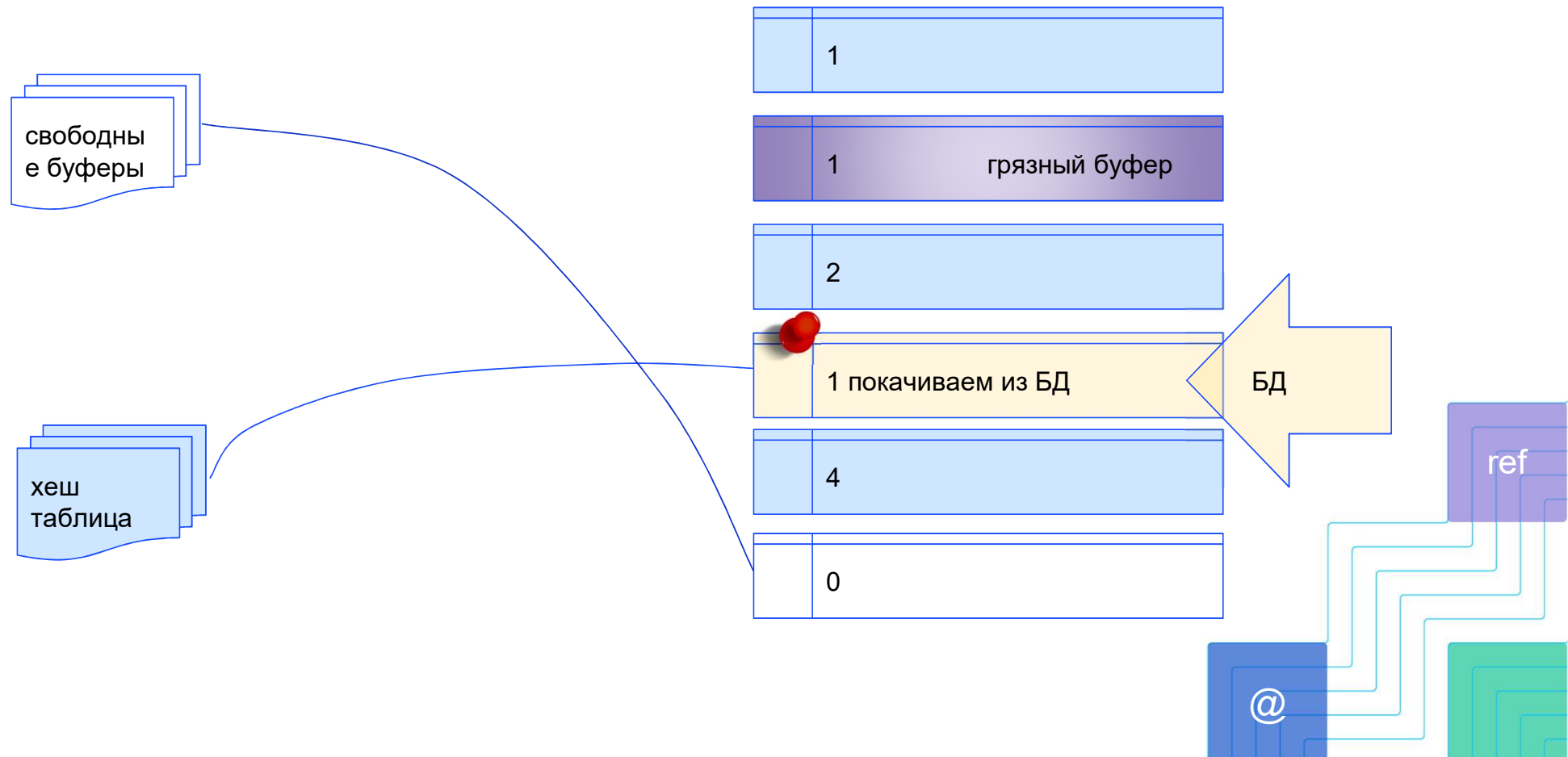
Сначала ищем в буферном кэше по хешу. Если нужная страница найдена в кэше, процесс должен:

- «закрепить» буфер (pin count, увеличить счетчик) и
- увеличить число обращений (usage count, счетчик)



БУФЕРНЫЙ КЕШ. МЕХАНИЗМ РАБОТЫ

Если нет страницы в кэше, то грузим с диска в первый пустой блок и добавляем строку в хеш таблицу. Блок буфера такой же как и данные с HDD.

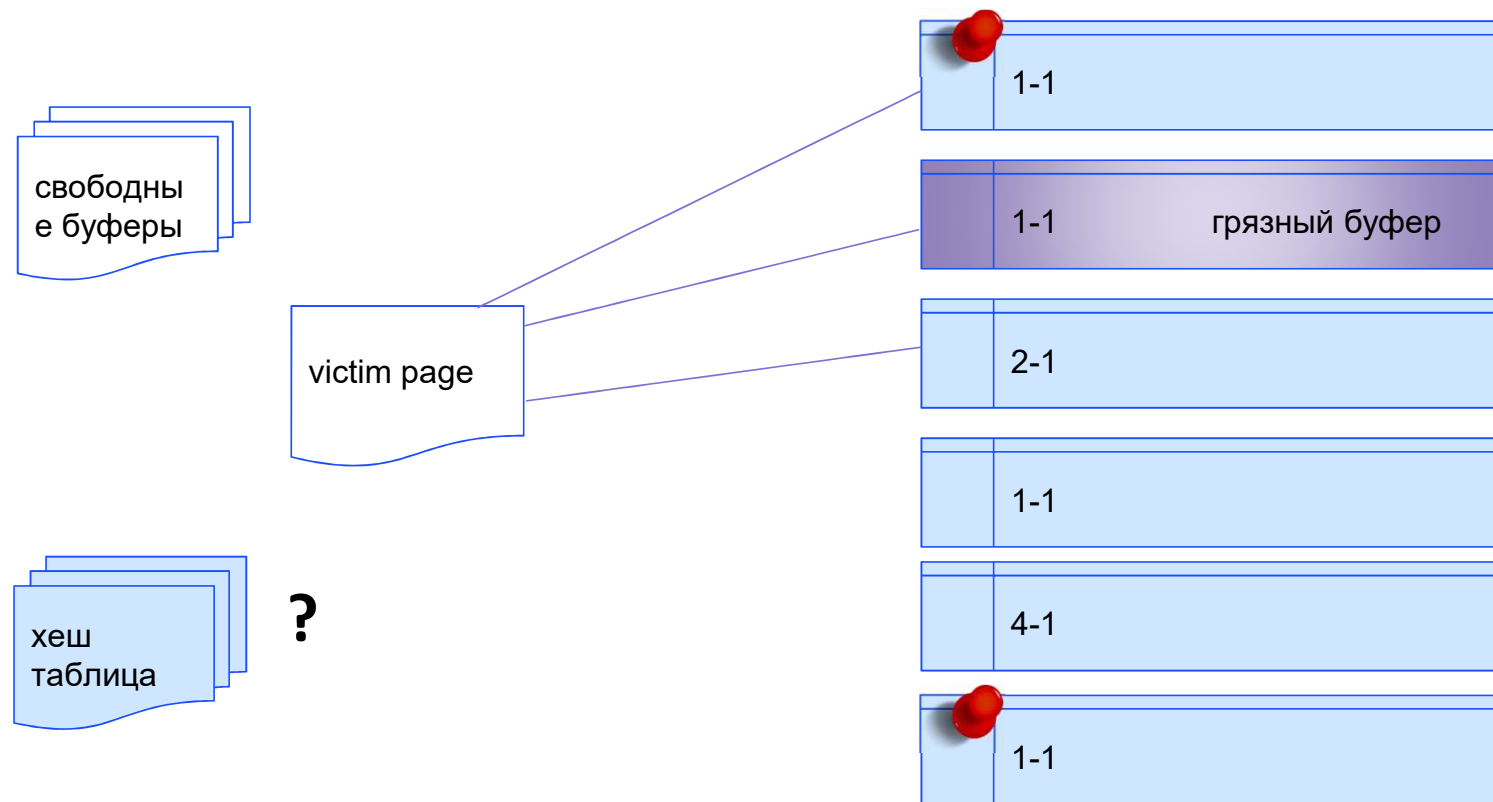


БУФЕРНЫЙ КЕШ. ЧТЕНИЕ С ВЫТЕСНЕНИЕМ



В какой-то момент все свободные блоки заканчиваются.. Что делать дальше?

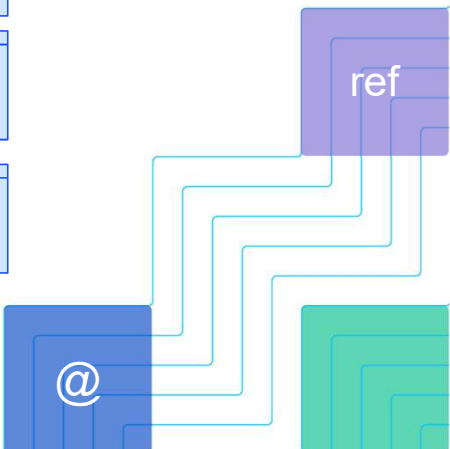
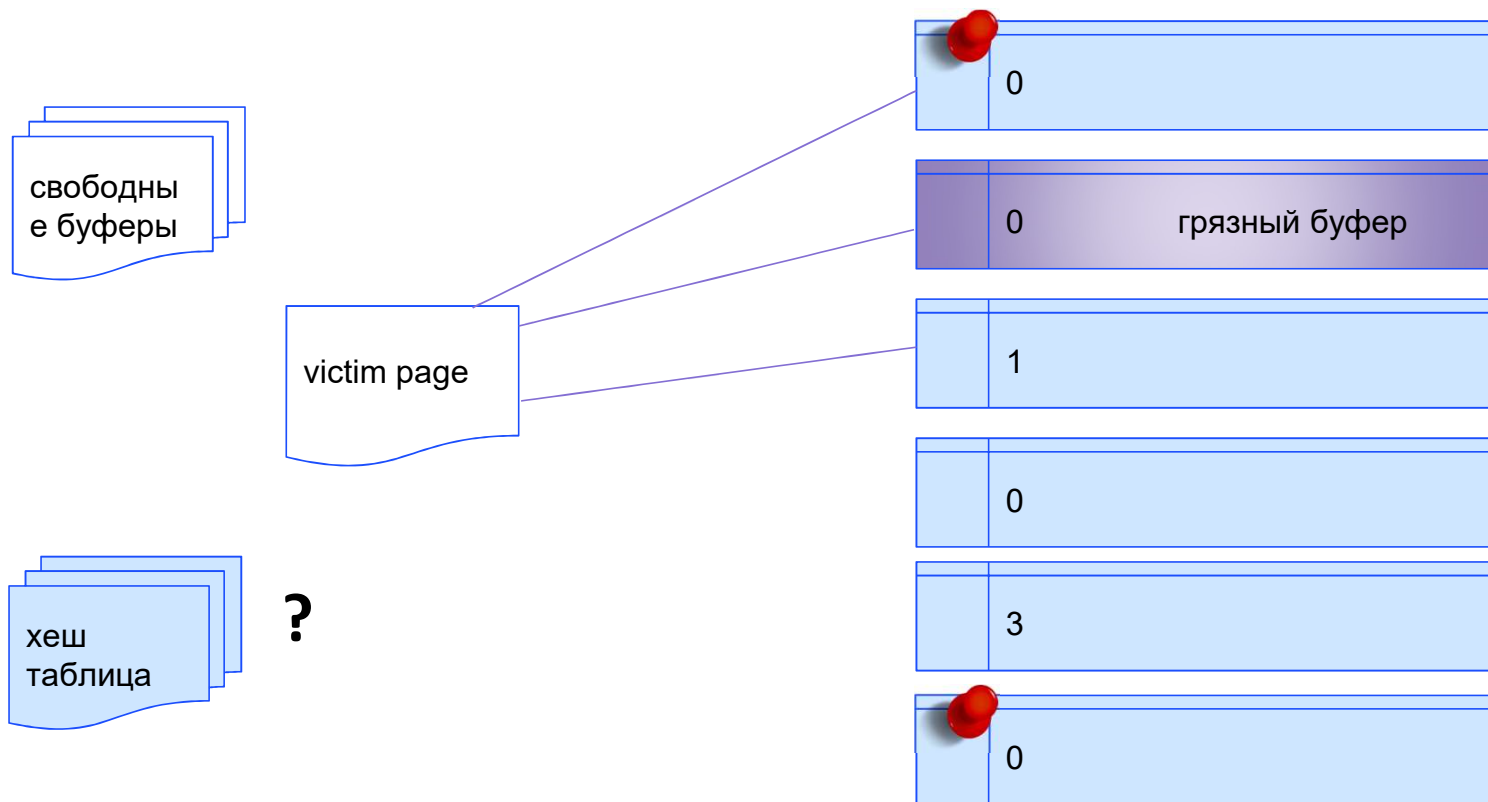
- Уменьшить usage count -1;



БУФЕРНЫЙ КЕШ. ЧТЕНИЕ С ВЫТЕСНЕНИЕМ

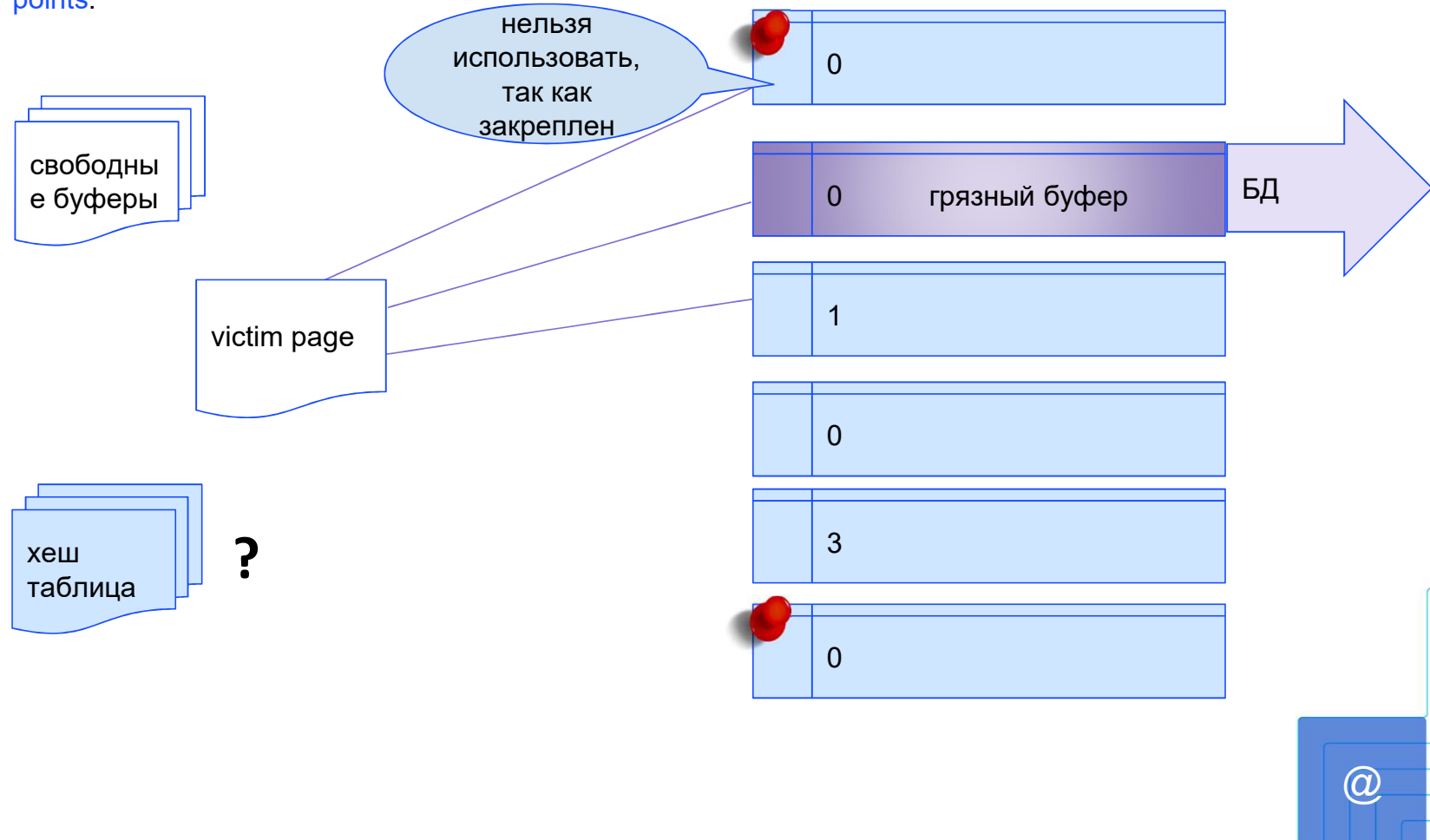


После уменьшения счётчика usage count появятся буферы с «0»

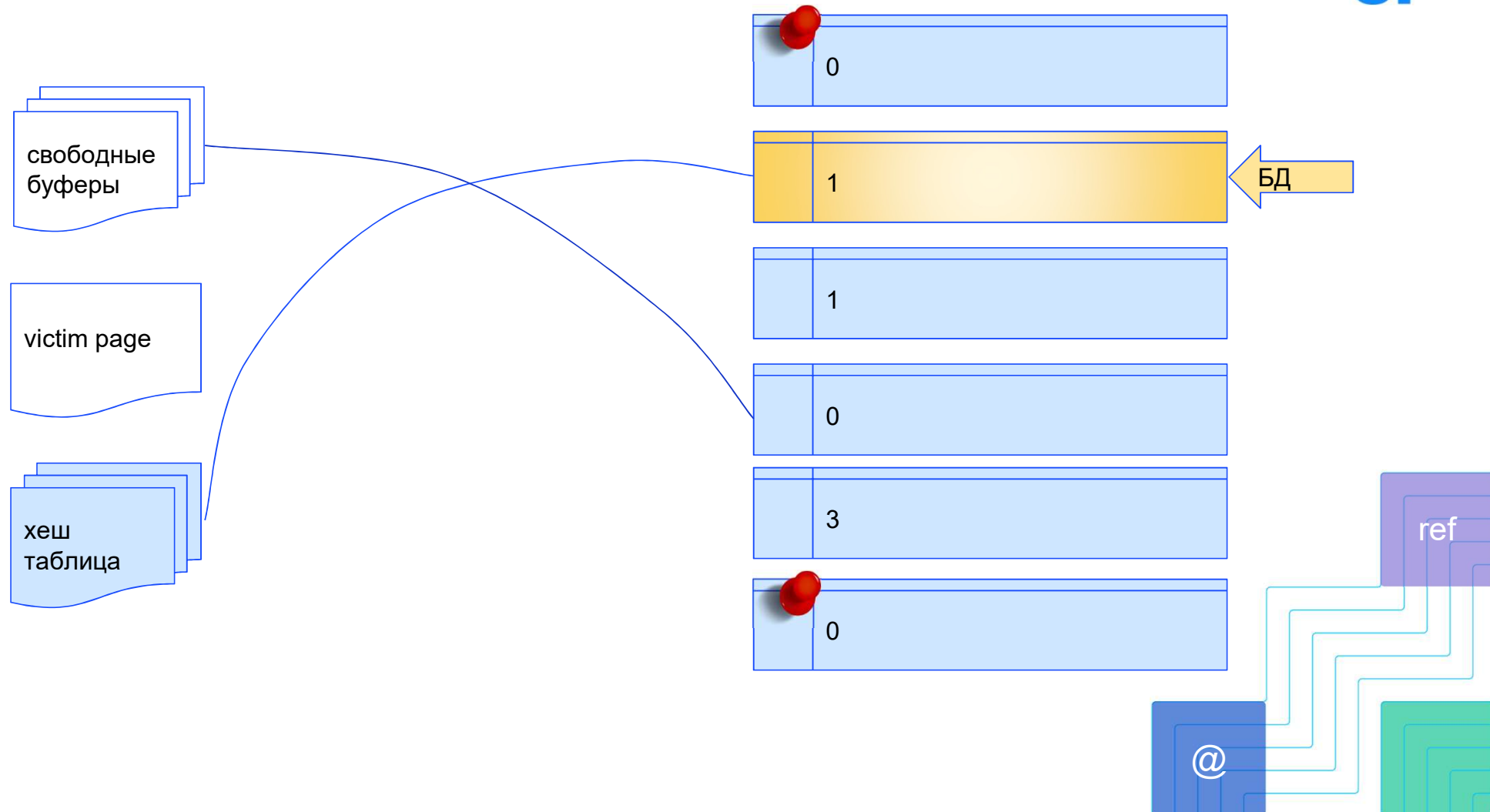


БУФЕРНЫЙ КЕШ. ЧТЕНИЕ С ВЫТЕСНЕНИЕМ

Грязную страницу нужно **заменить**, но для этого ее нужно сбросить на диск, что требует много времени. Частично сглаживается процессом **bgwriter** и механизмом контрольных точек **check points**.



БУФЕРНЫЙ КЕШ. ЧТЕНИЕ С ВЫТЕСНЕНИЕМ





БУФЕРНЫЙ КЭШ. НАСТРОЙКА



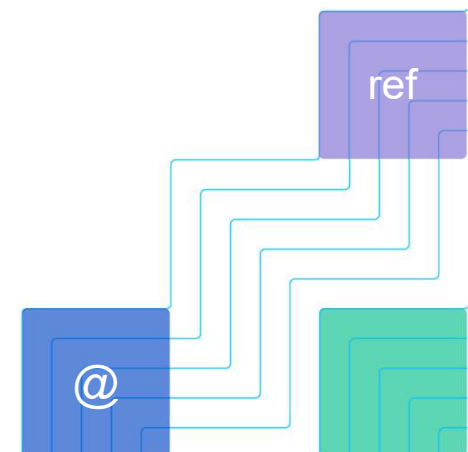
По умолчанию `shared_buffers = 128MB`

Буферный кэш должен содержать «активные» данные:

- при меньшем размере постоянно вытесняются полезные страницы
- при большем размере бессмысленно растут накладные расходы

начальная рекомендация — `25%` ОЗУ

Нужно учитывать двойное кэширование - если страницы нет в кэше СУБД, она может оказаться в кэше ОС, но алгоритм вытеснения ОС не учитывает специфики базы данных.






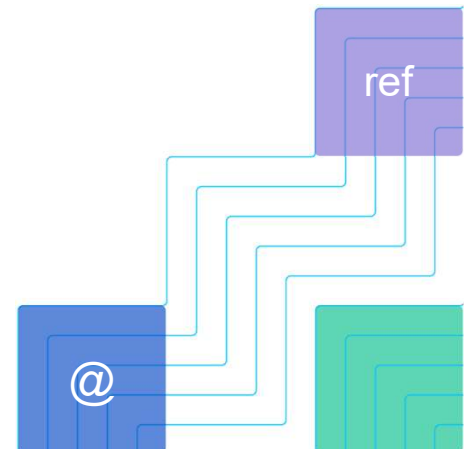
БУФЕРНЫЙ КЕШ. РАЗОГРЕВ КЕША



pg_prewarm

 используется после рестарта кластера

 заполняет кэш указанными таблицами





ИТОГИ ЗАНЯТИЯ



01



Узнали подробное
устройство
буферного кеша

02

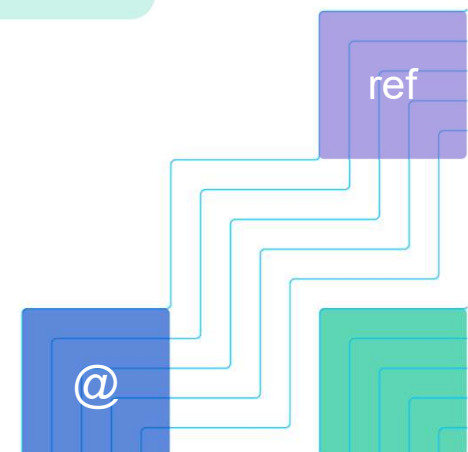


Поняли принцип его
работы

03



Узнали зачем нужна
утилита pg_prewarm



ПРОВЕРОЧНОЕ ЗАДАНИЕ

Цель задания: Загрузить в буфер данные для горячего доступа после рестарта сервера

begin

{ }



ЗАДАНИЕ ДЛЯ САМОПРОВЕРКИ



01

использовать
существующую или
создать новую
таблицу с данными

02

перезагрузить
кластер
Постгреса

03

создать
расширение
pg_prewarm

04

использовать утилиту
pg_prewarm на
таблице из 1 пункта

05

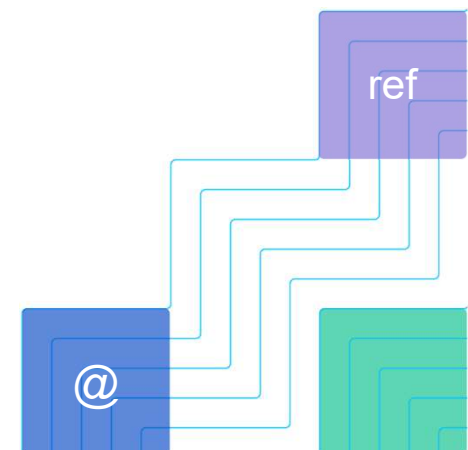
создать
расширение для
просмотра
содержимого
буферного кеша

06

создать
представление для
просмотра
буферного кеша
как было показано
на занятии

07

убедиться, что
данные загружены
в буферный кеш



СПАСИБО

На следующем занятии мы рассмотрим тему:

- Журналы

{ }

End