Реализация системы прав на Haskell

Александр Вершилов

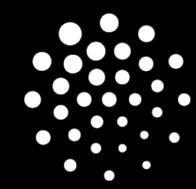
Про меня



- Разработчик с 2008 года
- Программирую на Haskell с 2011 года
- Senior software developer @ Tweag I/O с 2014 года

https://github.com/qnikst

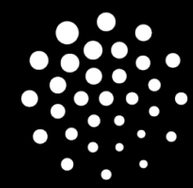
https://ru.linked.com/in/qnikst



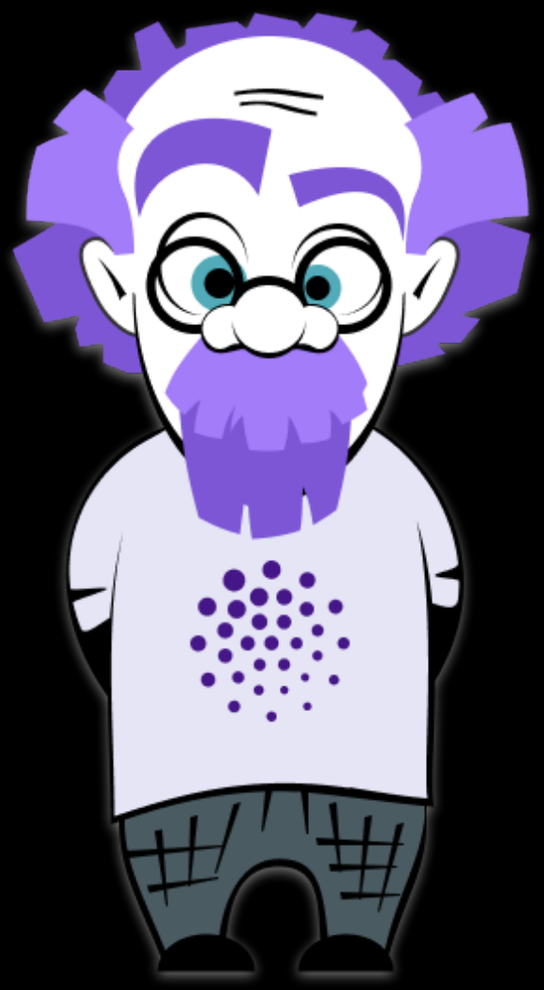
Про меня



- Разработчик с 2008 года
- Программирую на Haskell с 2011 года
- Senior software developer @ Tweag I/O с 2014 года
- Разработчик Сириус.Курсов с 2018 года



3 месяца



Сириус Курсы

https://edu.sirius.online/#/

Сириус Курсы Beta

Мои курсы

Доступен до 15 июня 2019 г. ✔ 1

Дополнительные главы геометрии. 7 класс

11 модулей, 34 видеолекции, 149 задач

ПРОДОЛЖИТЬ

Доступен до 15 июня 2019 г.

Дополнительные главы геометрии. 9 класс

23 модуля, 64 видеолекции, 392 задачи

ПРОДОЛЖИТЬ



Проф. Хеопс
Лучше учиться часто и понемногу, чем изредка, но большими кусками.

Доступные курсы

Доступен до 15 июня 2019 г.

Дополнительные главы геометрии. 8 класс

Сириус.Курсы — Открытая

https://edu.sirius.online/#/course/5

Дополнительные главы геометрии. 9 класс

0/19 0/4

Список модулей

Прямая Симсона

Прямая Симсона, лекция

Лемма о трезубце

Ортоцентр и вписанные углы

Гомотетия

Композиция гомотетий

Поворотная гомотетия

Счёт вписанных углов 2

Антипараллельность

Теорема Птолемея

Касательные к окружности

Описанные четырёхугольники

Вневписанные окружности

Счёт вписанных углов

Скалярное произведение

Конические сечения

Вписанные углы

Движения плоскости

Степень точки

Векторы

Геометрия масс

Радикальные оси

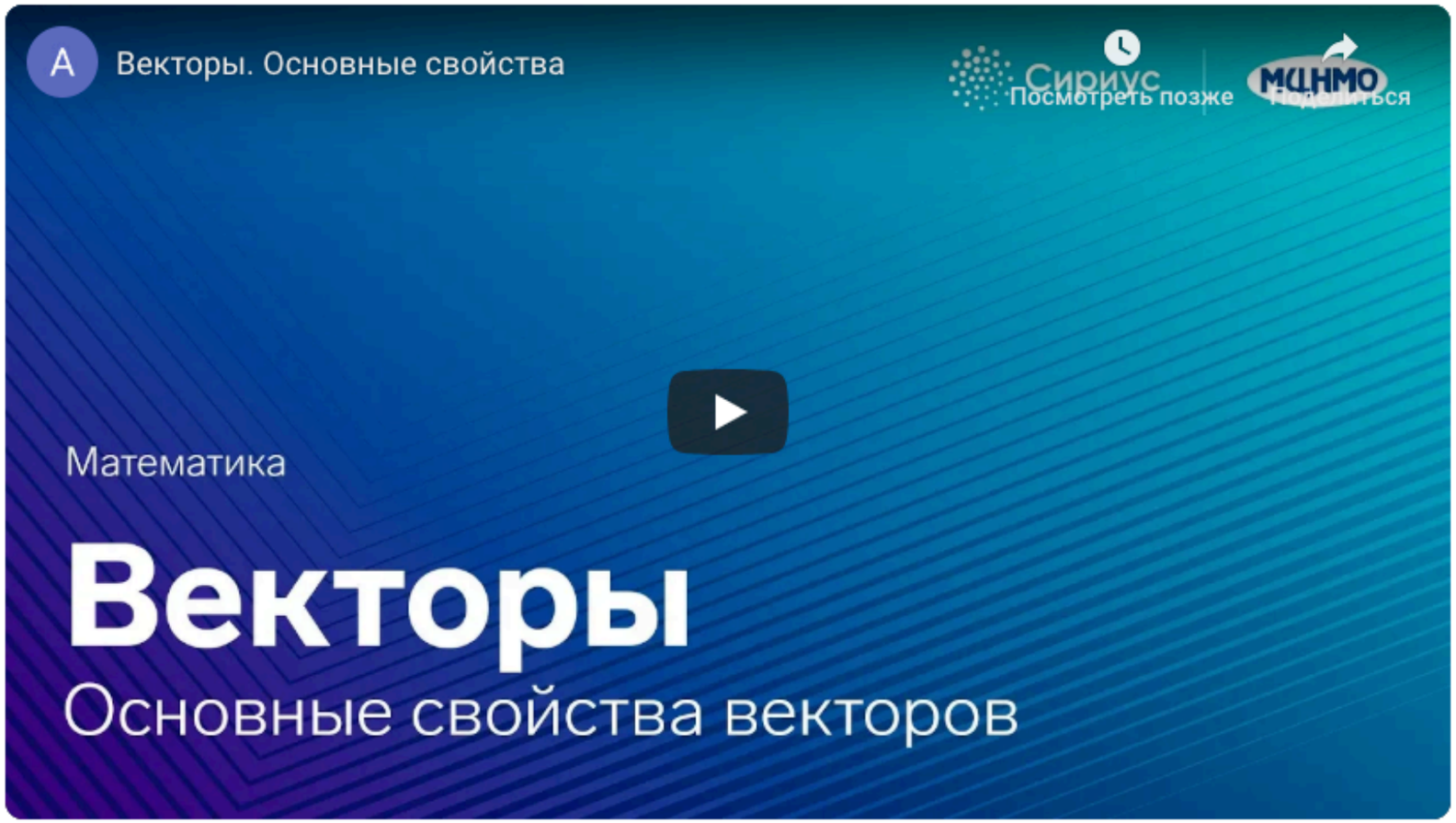
Дополнительные построения. Движения

ДОПОЛНИТЕЛЬНЫЕ ГЛАВЫ ГЕОМЕТРИИ. 9 КЛАСС

ПРОДОЛЖИТЬ

Дополнительные главы ге

https://edu.sirius.online/#/course/5/94



Краткий конспект

Вектор — это класс направленных отрезков, совпадающих по длине и направлению.

Выделяется также *нулевой вектор* $\vec{0}$, который имеет нулевую длину и не имеет направления.

Векторы можно складывать друг с другом, вычитать друг из друга, а также умножать на действительные числа. Эти операции подчиняются известным арифметическим законам.

Векторы. Задание 1 из 20

https://edu.sirius.online/#/course/5/94

На координатной плоскости заданы т … −5), $C(3, 4)$. Найдите координаты (x, y) вектора $\overrightarrow{AB} + 2\overrightarrow{BC} + \frac{1}{2}\overrightarrow{C}$

x y

3

верно!

ДАЛЕЕ

Вопросы - ответы про задачу

https://edu.sirius.online/#/course/5

Дополнительные главы геометрии. 9 класс

0/19 0/4

Список модулей

Счёт вписанных углов

Скалярное произведение

Вписанные углы

Степень точки

Векторы

Движения плоскости

Геометрия масс

дикальные оси

Дополнительные построения. Движения

ДОПОЛНИТЕЛЬНЫЕ ГЛАВЫ ГЕОМЕТРИИ. 9 КЛАСС

ПРОДОЛЖИТЬ

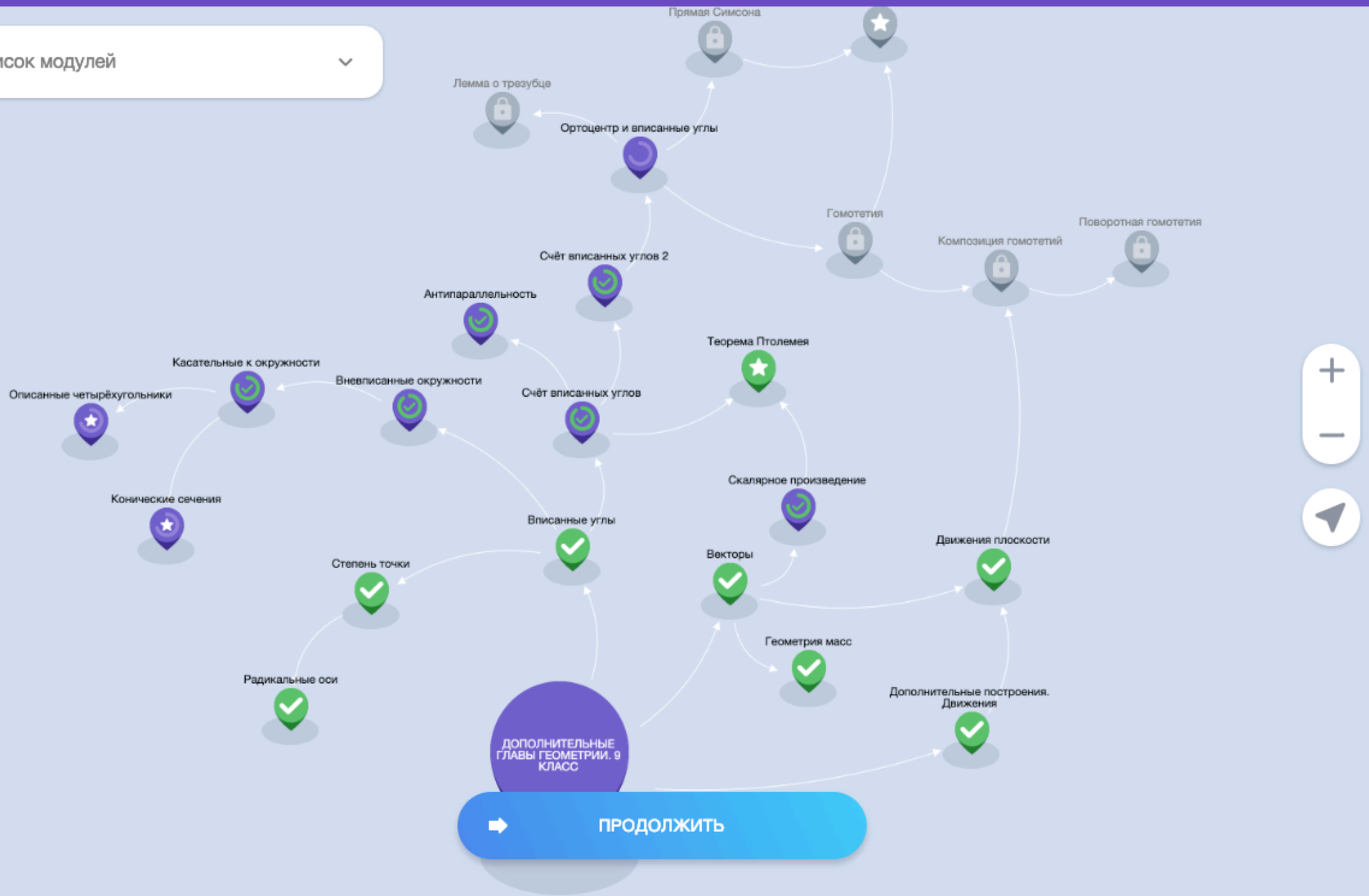
Дополнительные главы ге

https://edu.sirius.online/#/course/5

Дополнительные главы геометрии. 9 класс

17 13/19 1/4

Список модулей



ПРОДОЛЖИТЬ

Дополнительные главы ге

https://edu.sirius.online

https://lab.sirius.online/#/course

Cheops

Ольга Большакова

Олимпиады

Курсы

Пространства

Курсы

Геометрия. Исправленное	45	По 31 декабря
Демо-курс	16	По 15 июня
Дополнительные главы геометрии. 9 класс	23	По 15 июня
Дополнительные главы геометрии. 8 класс	19	По 15 июня
Дополнительные главы геометрии. 7 класс	11	По 15 июня

Сириус.Курсы — Открыта

https://lab.sirius.online/#/course/5

Cheops

Ольга Большакова

Олимпиады

Курсы

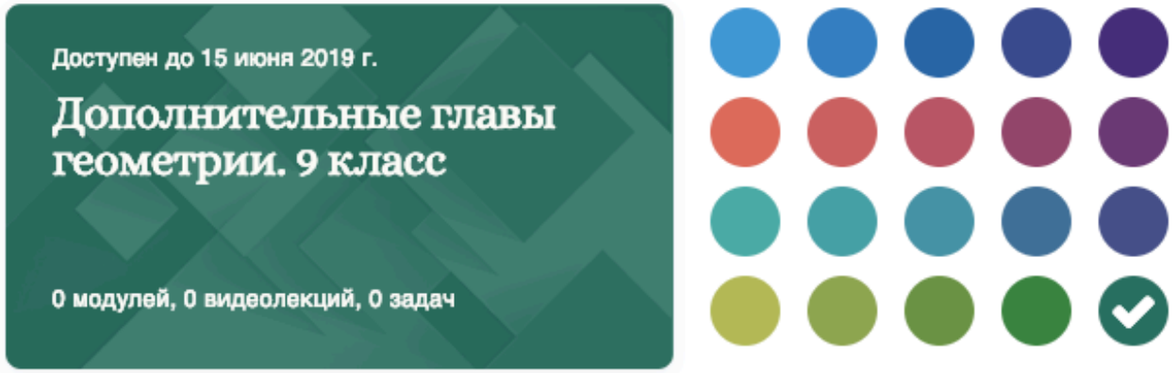
Пространства

Дополнительные главы геометрии. 9 класс

Отмена

Сохранить

Карта | Настройки | Новости | Вопросы-ответы | Выгрузки



Название

Дополнительные главы геометрии. 9 класс

О чем курс (выводится под названием курса на лендинге, описывает содержание курса)

Старайтесь уложиться в 200 знаков. Пишите по делу, наример: “Углубленное изучение школьной программы по геометрии 8 класса. Вы изучите темы: ...”

Правила курса (вкладка на странице курса, доступна ученикам)

Обучение в курсе

Присоединиться к курсу можно до 1 мая 2019 года, учебные материалы доступны для изучения до 15 июня 2019.

О курсе (вкладка на странице курса, доступна ученикам)

Сириус.Курсы — Открыта

https://lab.sirius.online/#/course/5/qa

Cheops

Ольга Большакова

Олимпиады

Курсы

Пространства

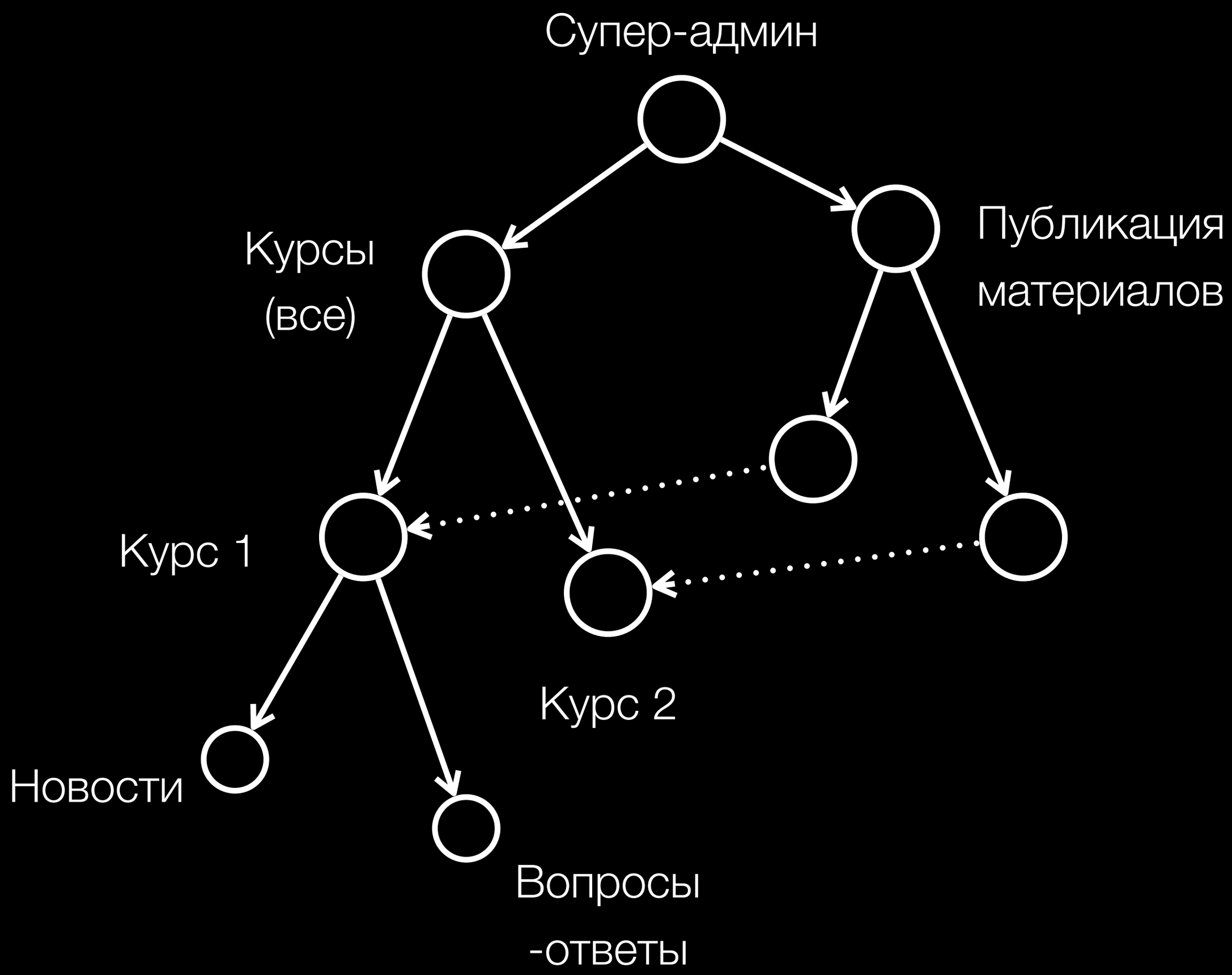
Дополнительные главы геометрии. 9 класс

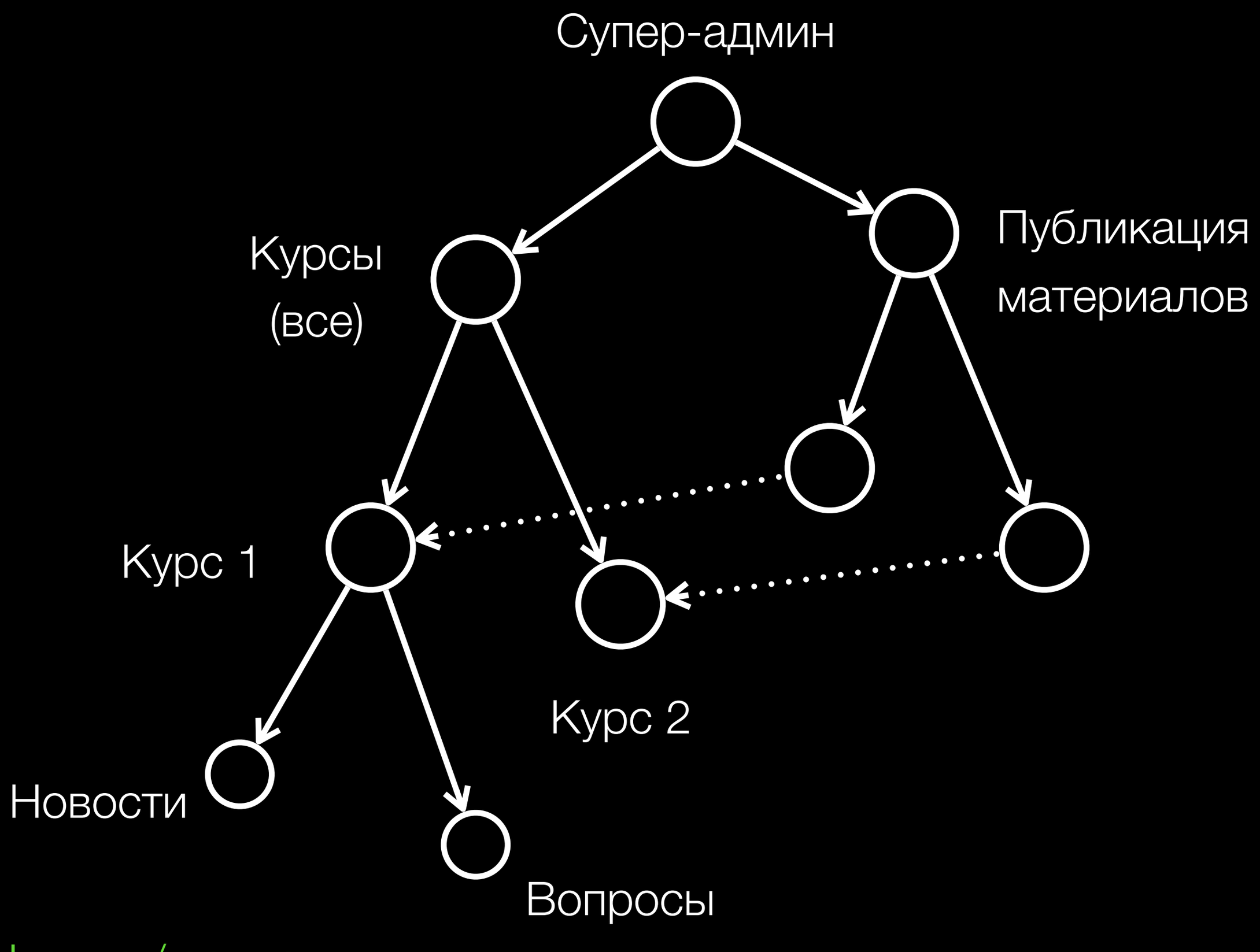
Карта | Настройки | Новости | Вопросы-ответы | Выгрузки

Векторы. Михаил Васильев: "В условии задачи опечатка. Середина отрезка должна …	16 марта
Степень точки. Артем Зубрицкий: "Скажите, в какой форме записывается ответ к э…	18 марта
Векторы. Глеб Сергеев: "не решается"	21 марта
Дополнительные построения. Движения. Егор Бородулин: "вершины отрезков могут …	24 марта
Вписанные углы. Дарья Брусникина : "А доказывать для выпуклого четырехугольник…	25 марта
Векторы. Артём Сидоров: "У меня при любом из решений получаются координаты (1…	27 марта
Радикальные оси. Роман Ященко: "Что-то вы в конце напутали."	27 марта
Радикальные оси. Роман Ященко: "Примерно на 1:40 говорится, что степень точки - …	28 марта
Вписанные углы. Мария Кёся: "Здравствуйте, у меня такой вопрос, почему каждый р…	29 марта
Вневписанные окружности. Роман Ященко: "Если такое можно уточнить, то сколько …	29 марта
Вневписанные окружности. Роман Ященко: "Уже не нужно. Извините!"	29 марта
Касательные к окружности. Роман Ященко: "Извините, но условие точно корректное?"	30 марта
Вписанные углы. Егор Иванов: "Что делать если я ничего не понял. Как в задаче 2 то…	1 апреля

Сириус.Курсы — Открыта

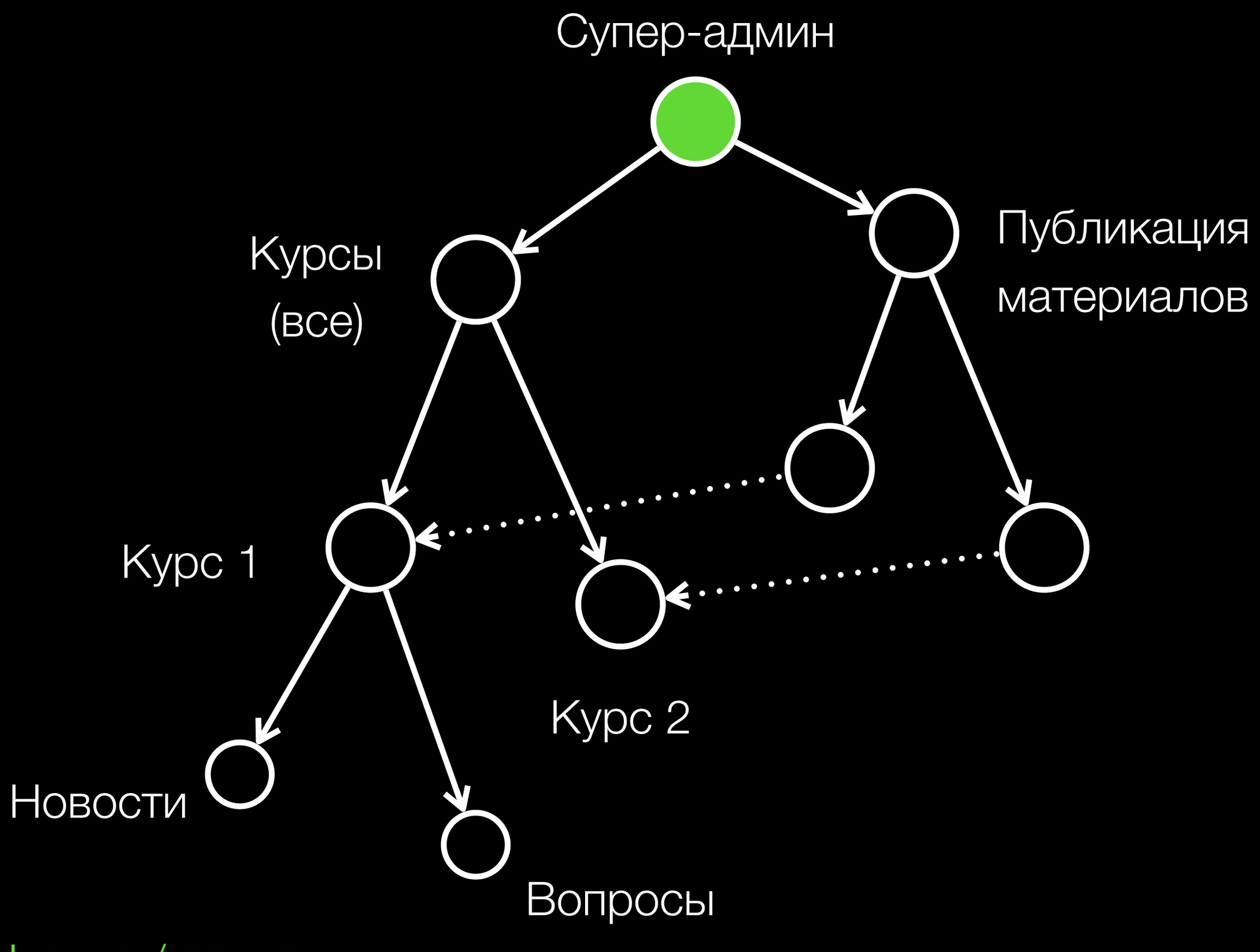
Система прав


Супер-админ
Курсы
(все)
Публикация
материалов
Курс 1
Курс 2
Новости
Вопросы
-ответы



Чтение / запись

Только чтение



Чтение / запись

Только чтение

Супер-админ

Курсы
(все)

Публикация
материалов

Курс 1

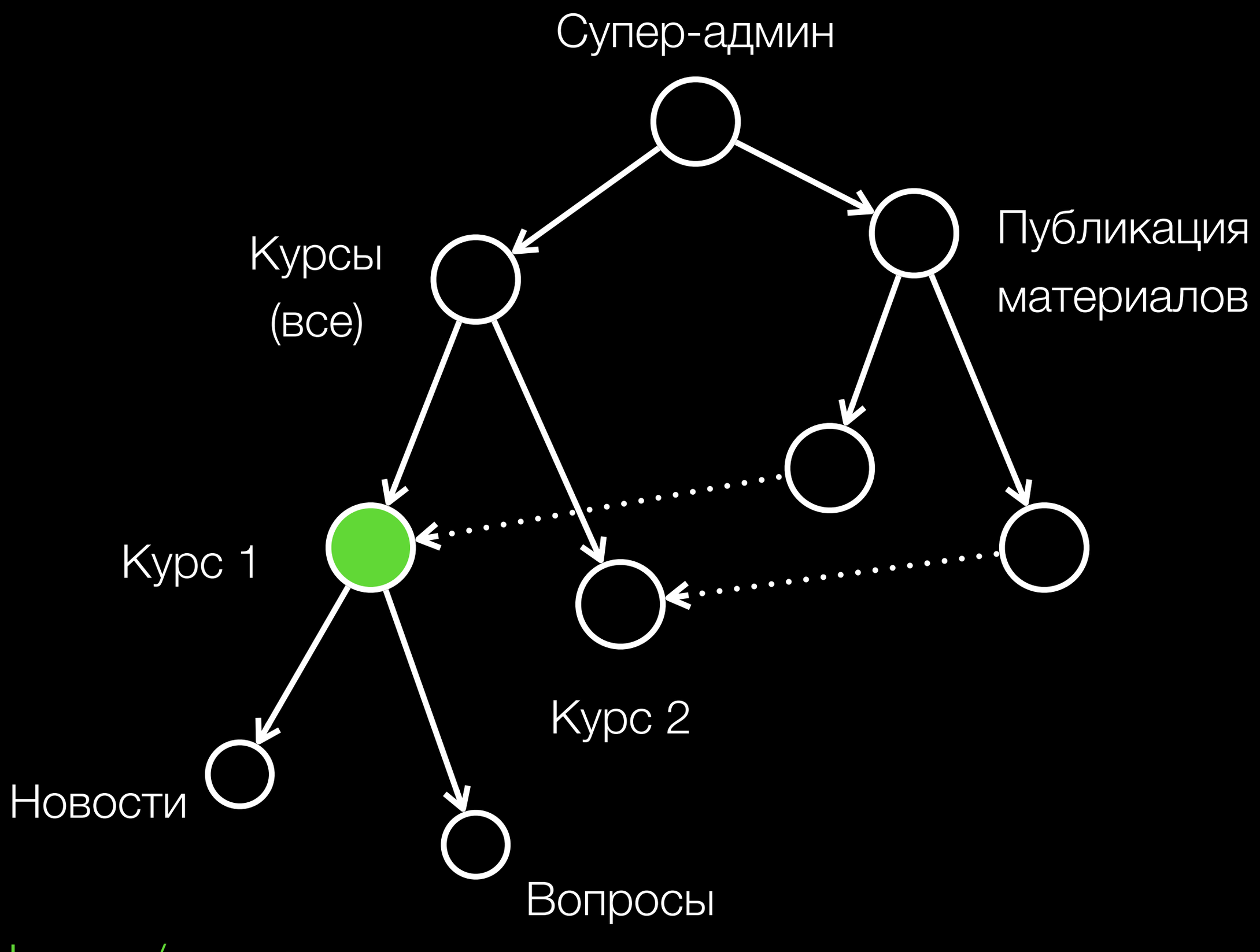
Курс 2

Новости

Вопросы
-ответы

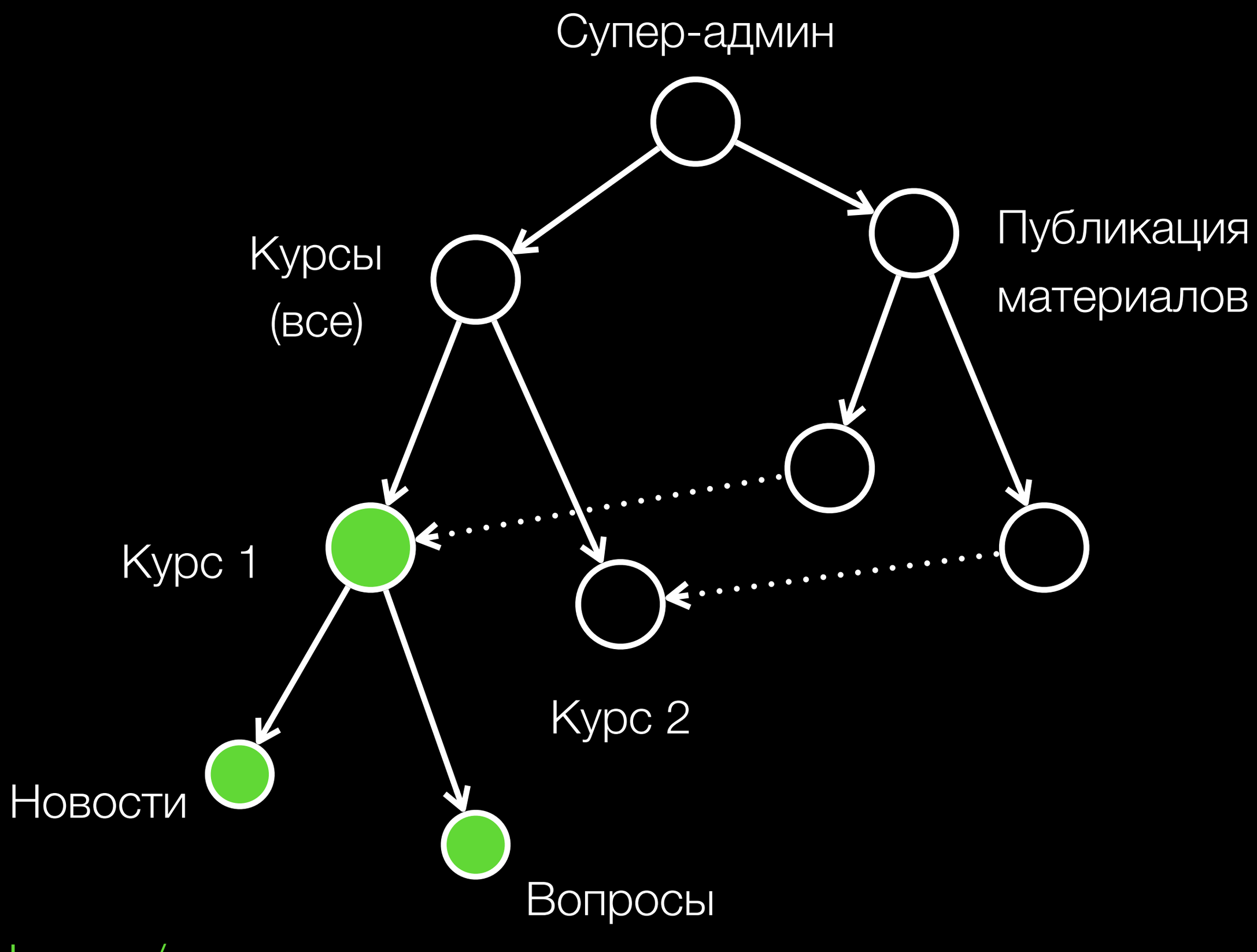
Чтение / запись

Только чтение



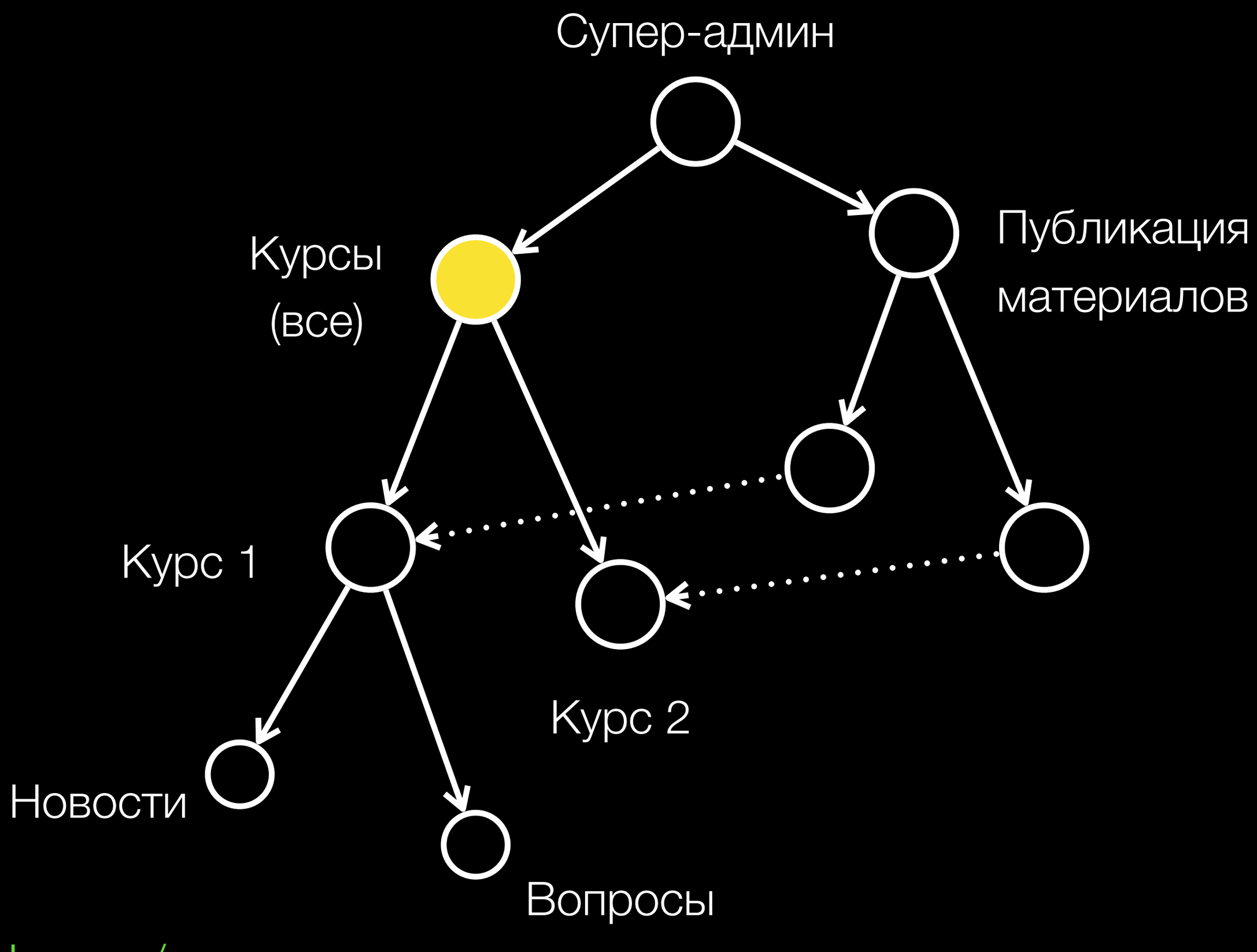
Чтение / запись

Только чтение



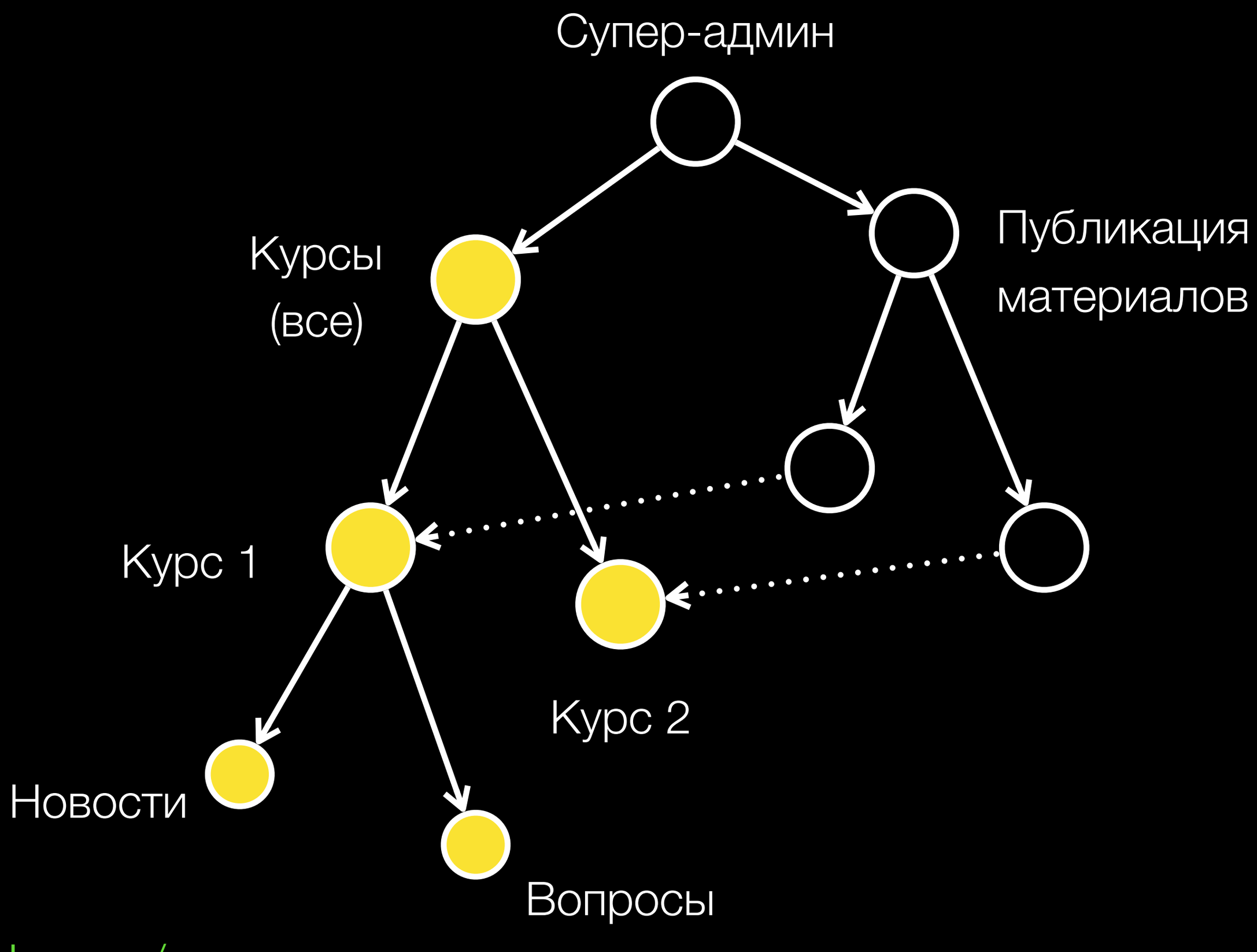
Чтение / запись

Только чтение



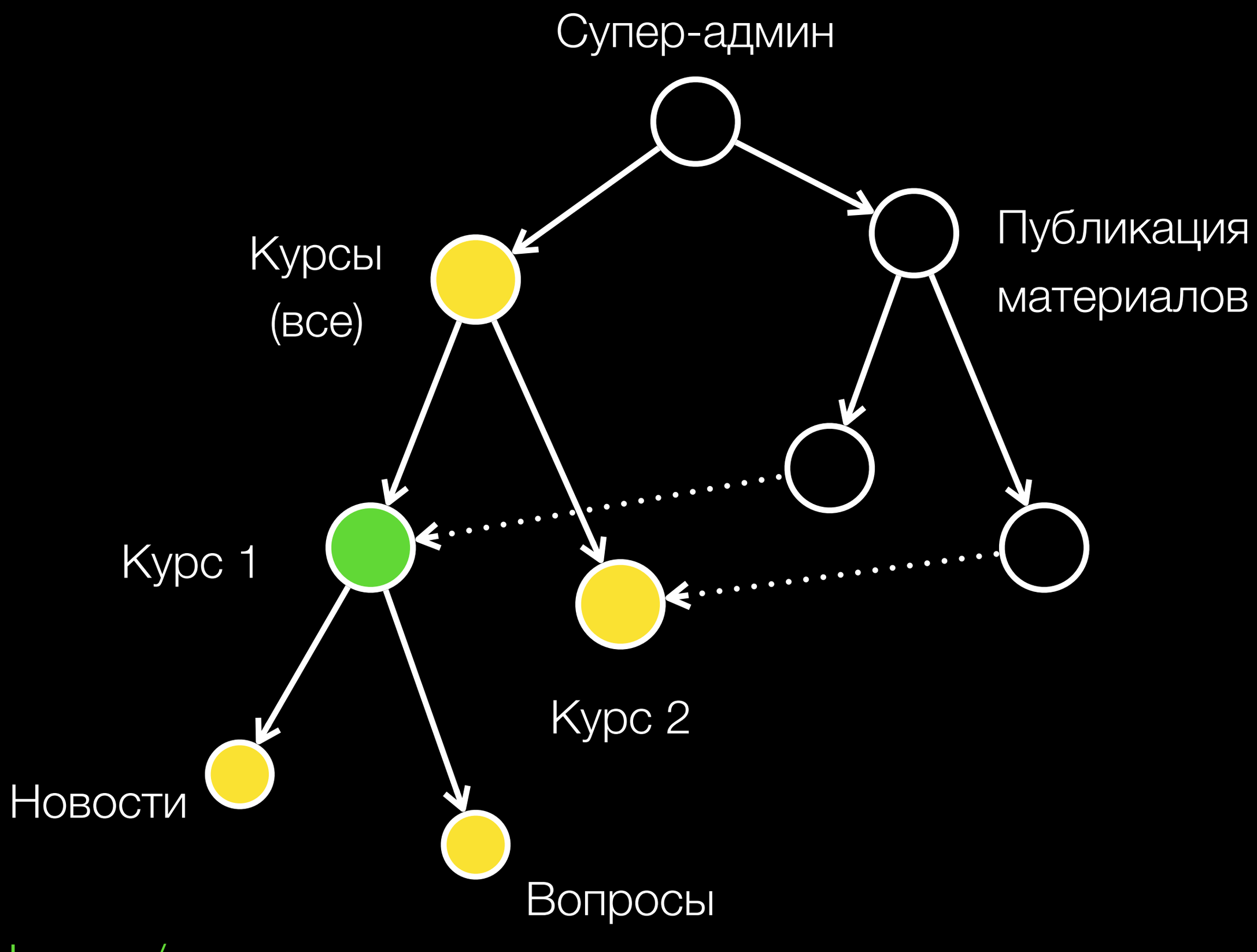
Чтение / запись

Только чтение



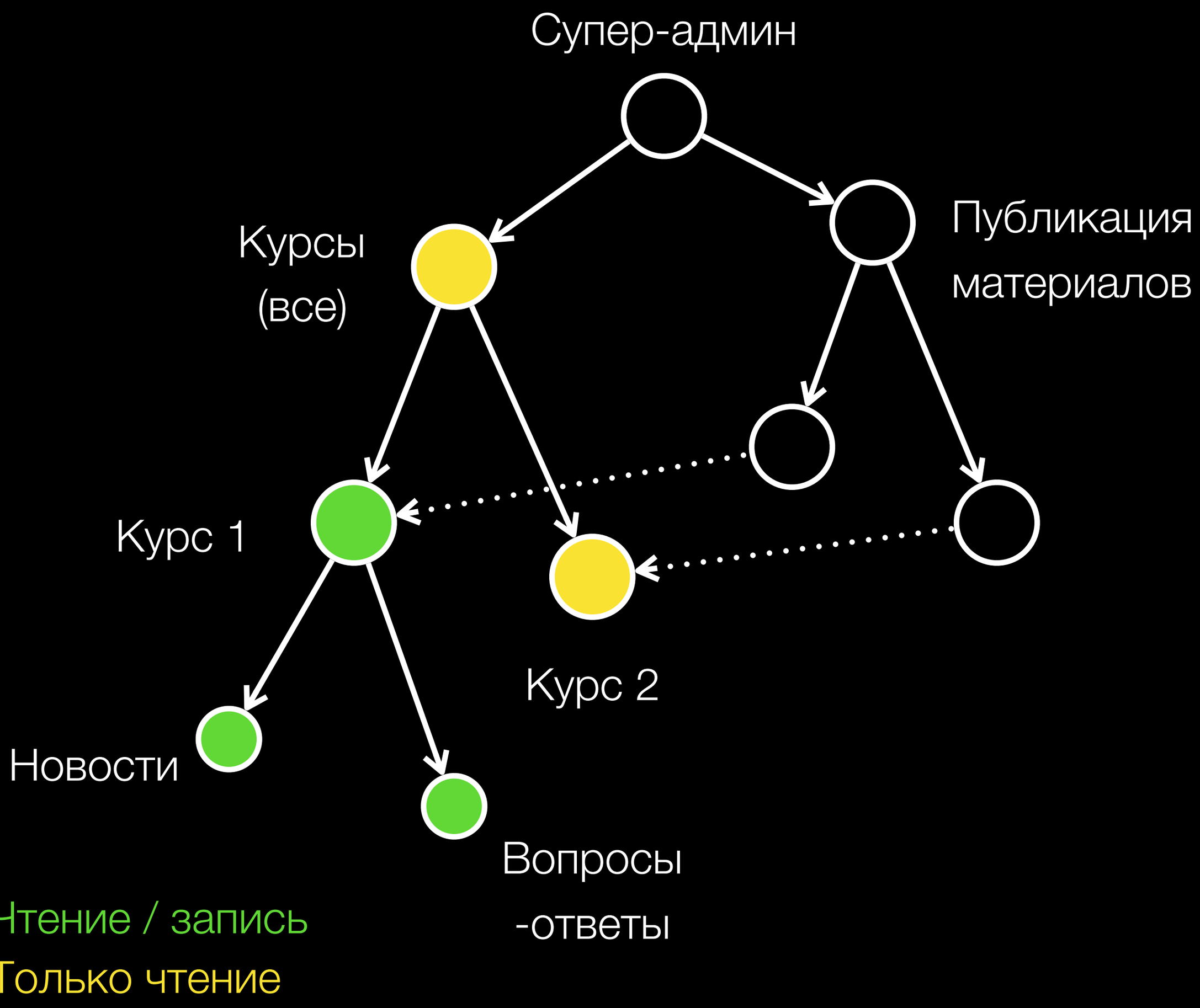
Чтение / запись

Только чтение



Чтение / запись

Только чтение



Чтение / запись

Только чтение

https://lab.sirius.online/#/course/5/report

Cheops

Ольга Большакова

Олимпиады

Курсы

Пространства

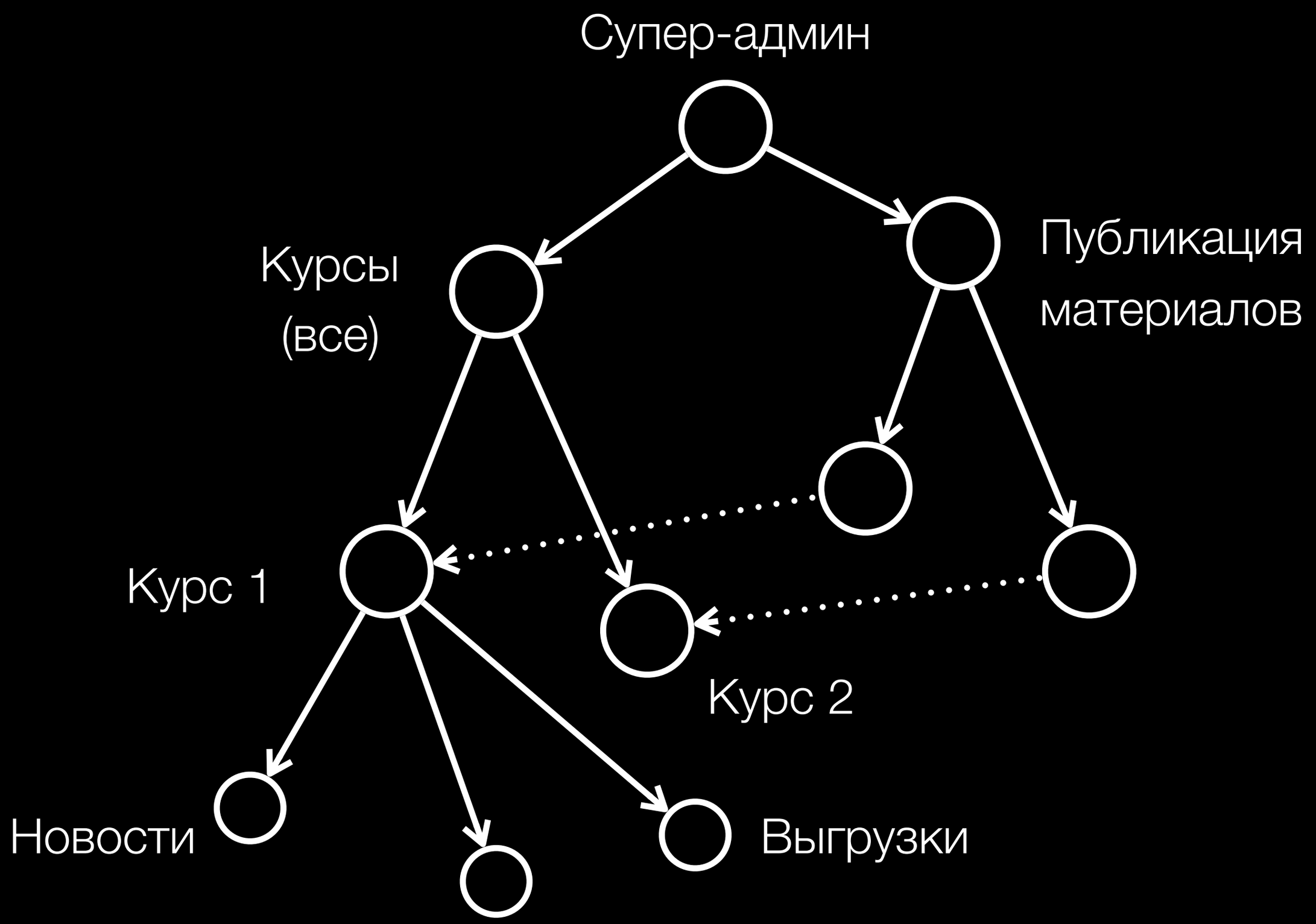
Дополнительные главы геометрии. 9 класс

Заказать выгрузку

Карта | Настройки | Новости | Вопросы-ответы | Выгрузки

1 апреля 2019 г.	10:52 0 сек.	Горбачев А.
29 марта 2019 г.	11:44 0 сек.	Гусев А.
28 марта 2019 г.	23:30 0 сек.	Гусев А.
28 марта 2019 г.	11:19 0 сек.	Гусев А.
27 марта 2019 г.	10:20 0 сек.	Горбачев А.
20 марта 2019 г.	12:01 0 сек.	Гусев А.
19 марта 2019 г.	10:06 0 сек.	Горбачев А.
18 марта 2019 г.	17:05 0 сек.	Горбачев А.
18 марта 2019 г.	17:05 0 сек.	Горбачев А.
18 марта 2019 г.	11:05 0 сек.	Горбачев А.
18 марта 2019 г.	11:05 0 сек.	Горбачев А.
17 марта 2019 г.	12:22 0 сек.	Горбачев А.
16 марта 2019 г.	23:09 0 сек.	Гусев А.
16 марта 2019 г.	18:53 0 сек.	Горбачев А.

Сириус.Курсы — Открыта

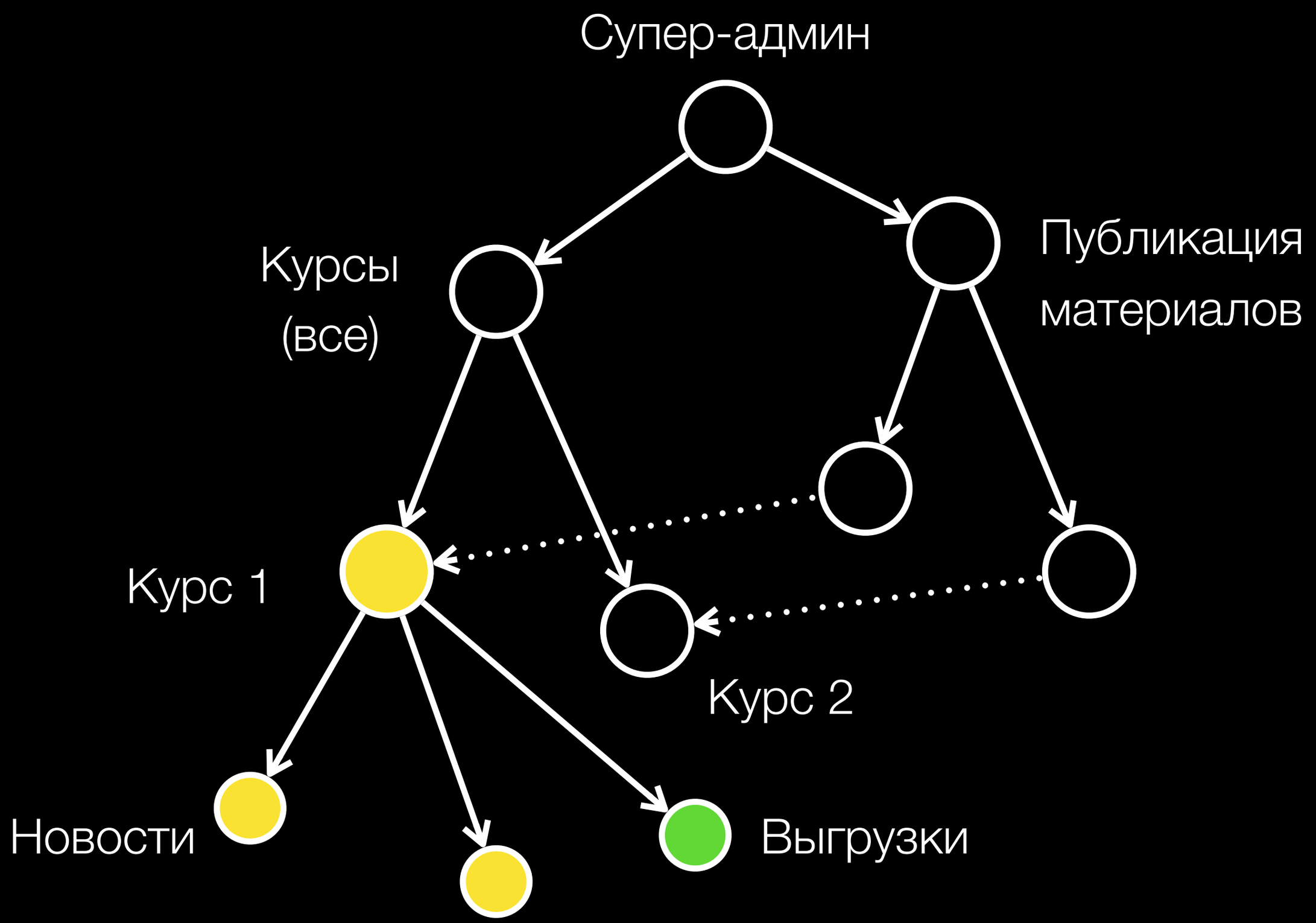


Чтение / запись

Только чтение

Вопросы

-ответы



Чтение / запись

Только чтение

Вопросы

-ответы

Handle pattern

Внешний код

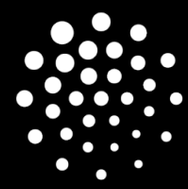
Обобщенный интерфейс

Конкретные реализации

Auth

Auth.Impl.Database

Auth.Impl.Remote



```haskell
module Auth (Handle(..)) where

data Handle = Handle {

      hasPermissionsFor :: UserId
                        -> PermissionTarget
                        -> IO Permissions

    , createObject      :: PermissionTarget
                        -> ObjectAddr
                        -> IO PermissionObjectId

    , ...
  }
```

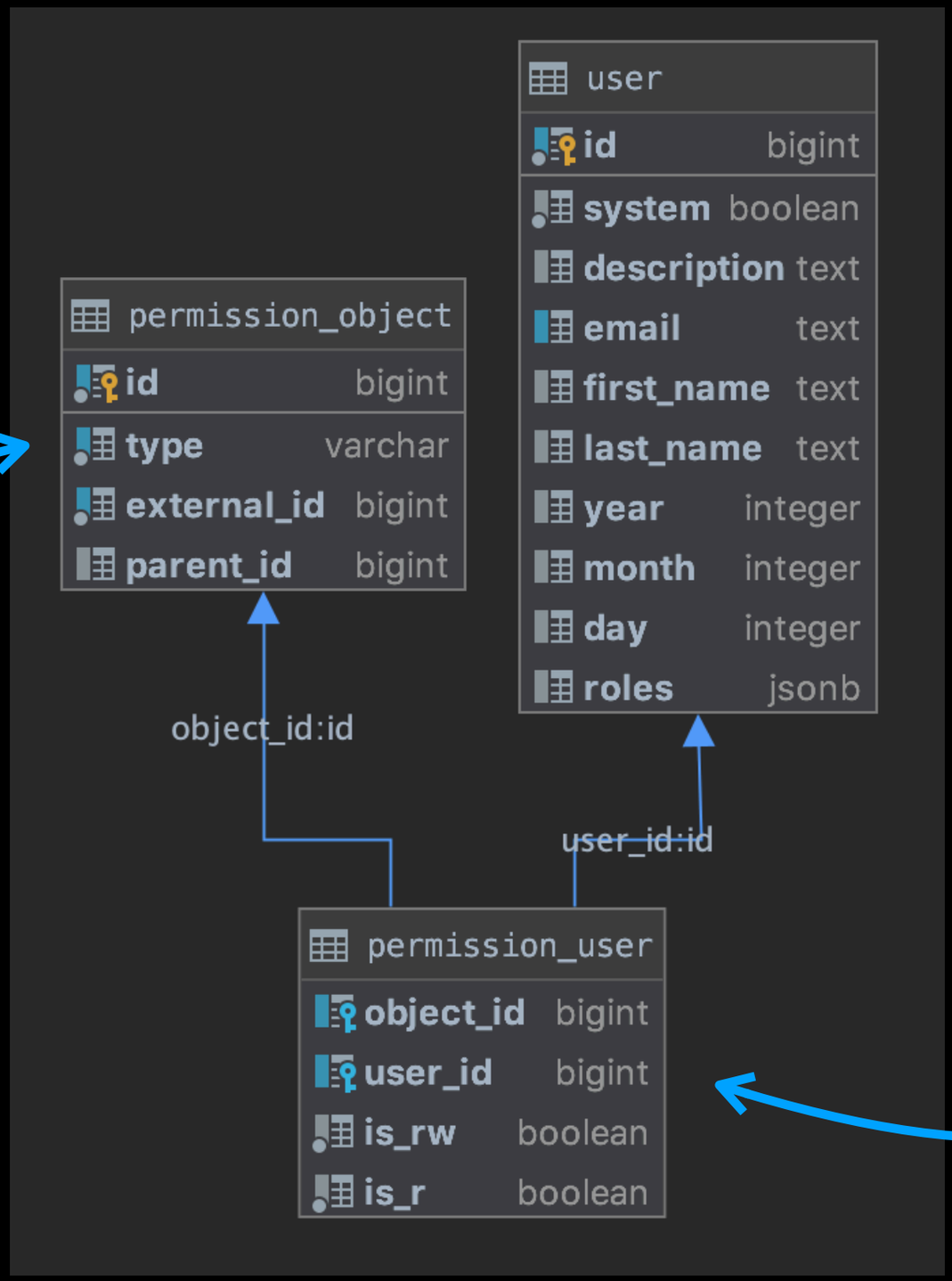
```
module Auth.Db (new) where

import Db

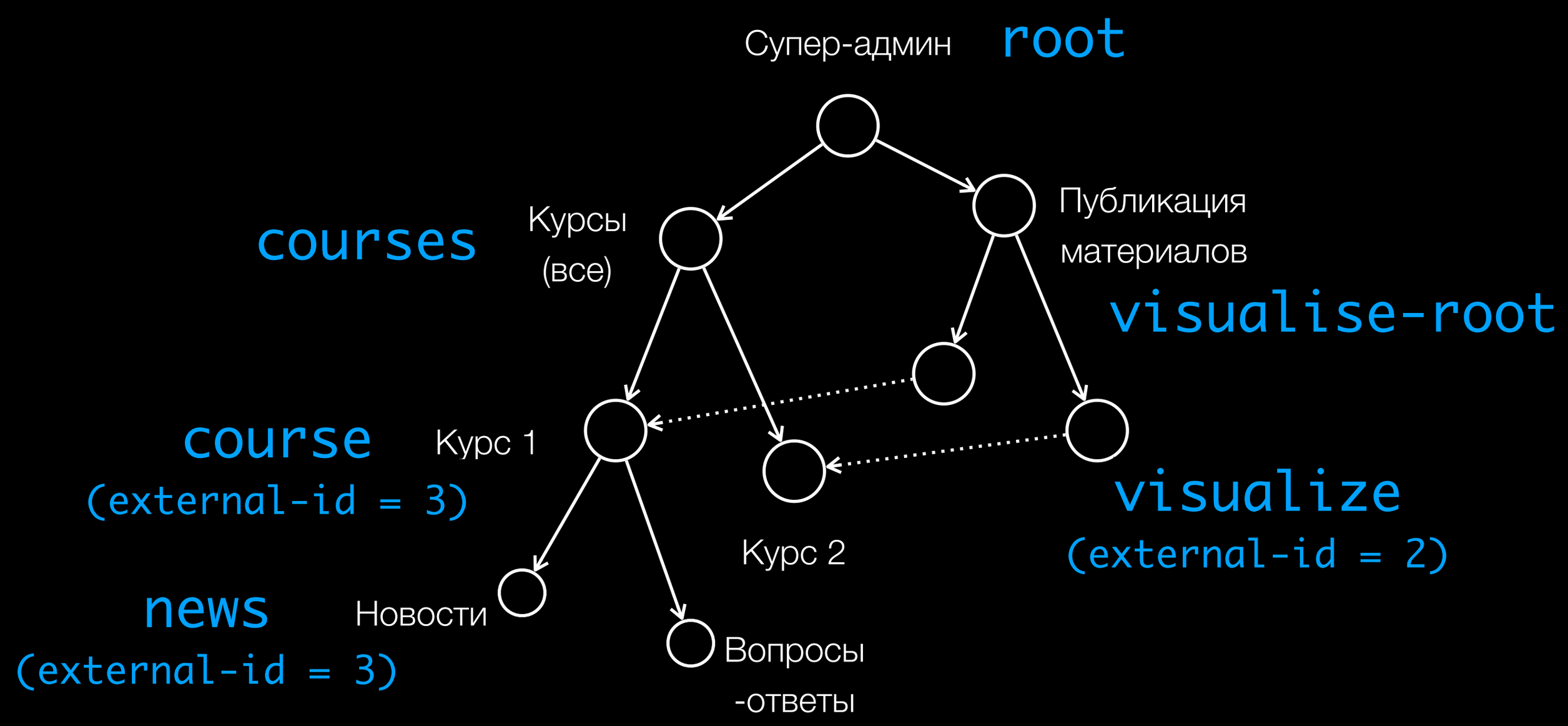
new :: Db.Handle -> Log.Handle -> Handle
new = ...
```

Auth.Db

Дерево объектов

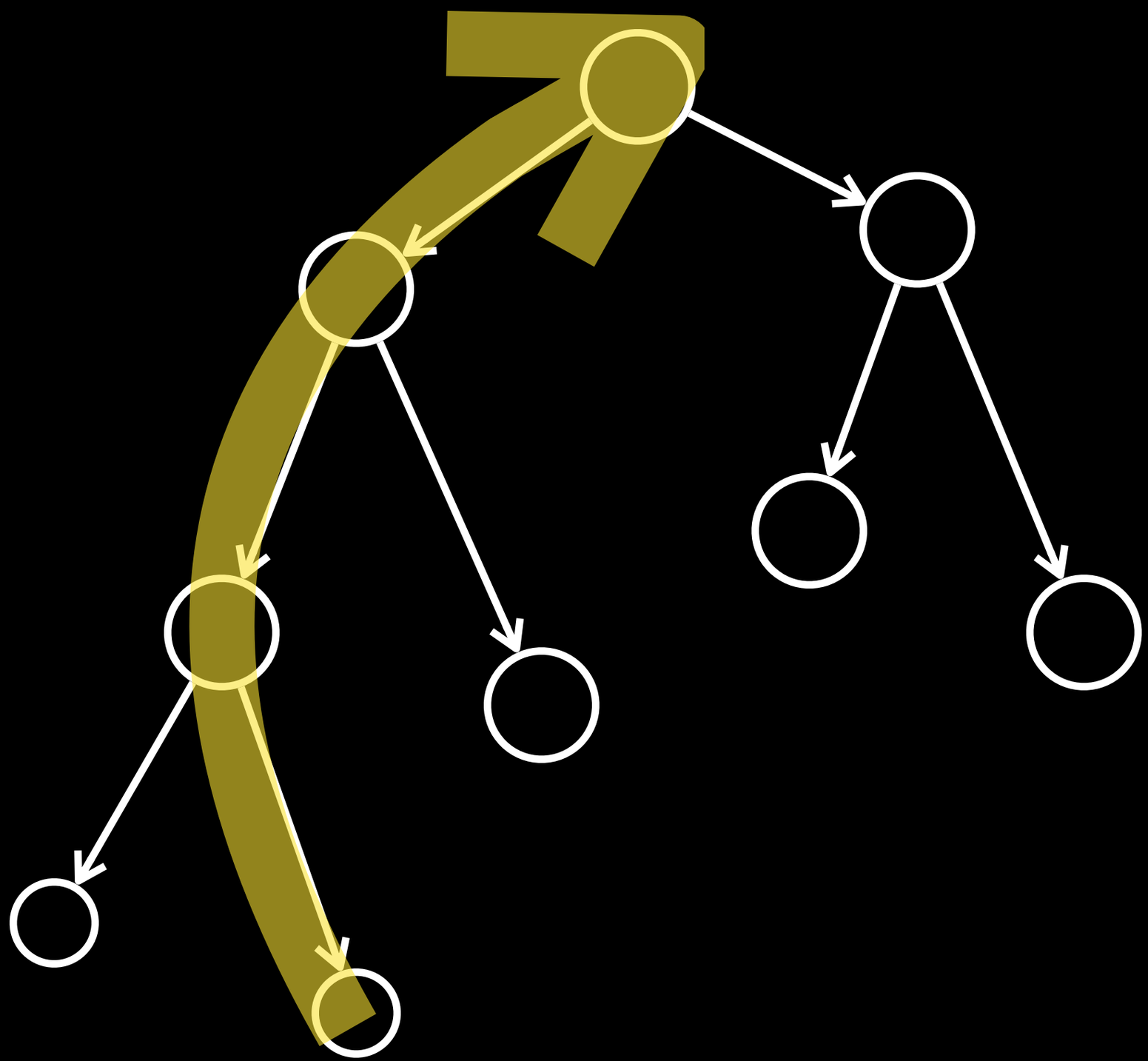


Права пользователей



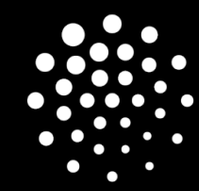
id	type	external_id	parent_id
1	root	0	<null>
2	environments	0	1
6	courses	0	1
7	course	3	6
8	news	3	7
274	visualize-root	0	1
275	visualize	1	274
276	visualize	2	274

Получение прав



Получение прав

```
WITH RECURSIVE
  perms AS (SELECT * FROM permission_user WHERE user_id = $1),
  base AS
     (SELECT apo.parent, apu.is_rw, apu.is_r
        FROM permission_object AS apo
           LEFT JOIN perms AS apu ON apu.object_id = apo.id
        WHERE apo.type=$2 AND external_id=$3
     UNION ALL
        SELECT apo.parent_id, apu.is_rw, apu.is_r
           FROM base
              LEFT JOIN permission_object as apo ON base.parent_id = apo.id
              LEFT JOIN perms as apu on app.object_id = apo.id
          WHERE base.parent_id IS NOT NULL)
SELECT
    COALESCE(bool_or(is_rw), FALSE),
    COALESCE(bool_or(is_r), FALSE)
 FROM base WHERE is_rw IS NOT NULL
```



Получение прав

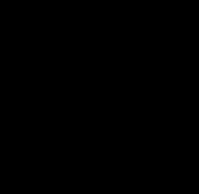
```
WITH RECURSIVE
  perms AS (SELECT * FROM permission_user WHERE user_id = $1),
  base AS
     (SELECT apo.parent, apu.is_rw, apu.is_r
        FROM permission_object AS apo
           LEFT JOIN perms AS apu ON apu.object_id = apo.id
        WHERE apo.type=$2 AND external_id=$3
     UNION ALL
        SELECT apo.parent_id, apu.is_rw, apu.is_r
           FROM base
              LEFT JOIN permission_object as apo ON base.parent_id = apo.id
              LEFT JOIN perms as apu on app.object_id = apo.id
          WHERE base.parent IS NOT NULL)
SELECT
    COALESCE(bool_or(is_rw), FALSE),
    COALESCE(bool_or(is_r), FALSE)
 FROM base
```

Находим нужный объект

Получение прав

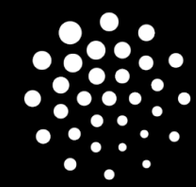
```
WITH RECURSIVE
  perms AS (SELECT * FROM permission_user WHERE user_id = $1),
  base AS
     (SELECT apo.parent, apu.is_rw, apu.is_r
        FROM permission_object AS apo
           LEFT JOIN perms AS apu ON apu.object_id = apo.id
        WHERE apo.type=$2 AND external_id=$3
     UNION ALL
        SELECT apo.parent_id, apu.is_rw, apu.is_r
           FROM base
              LEFT JOIN permission_object as apo ON base.parent_id = apo.id
              LEFT JOIN perms as apu on app.object_id = apo.id
          WHERE base.parent_id IS NOT NULL)
SELECT
    COALESCE(bool_or(is_rw), FALSE),
    COALESCE(bool_or(is_r), FALSE)
FROM base
```

Поднимаемся по дереву

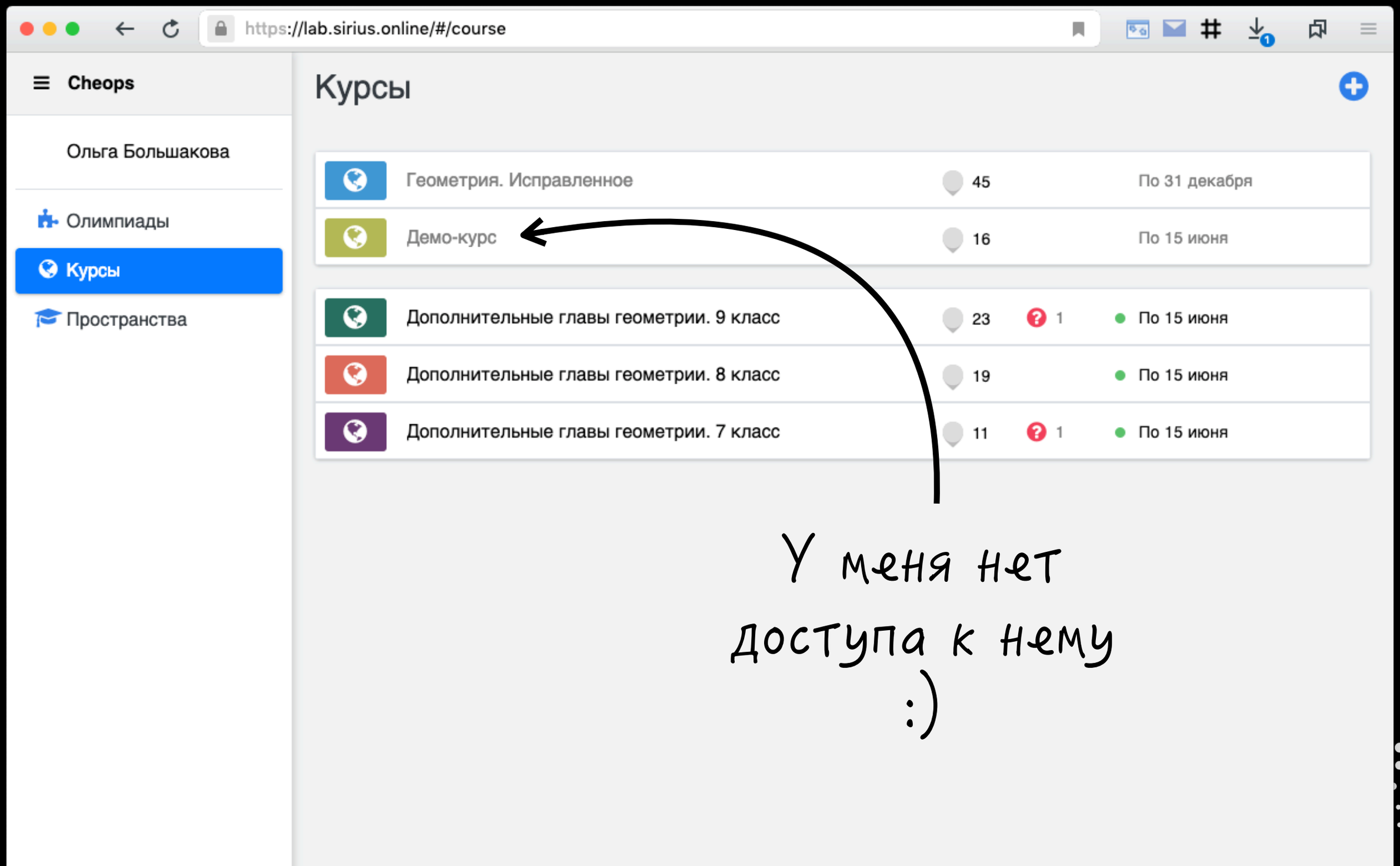


Получение прав

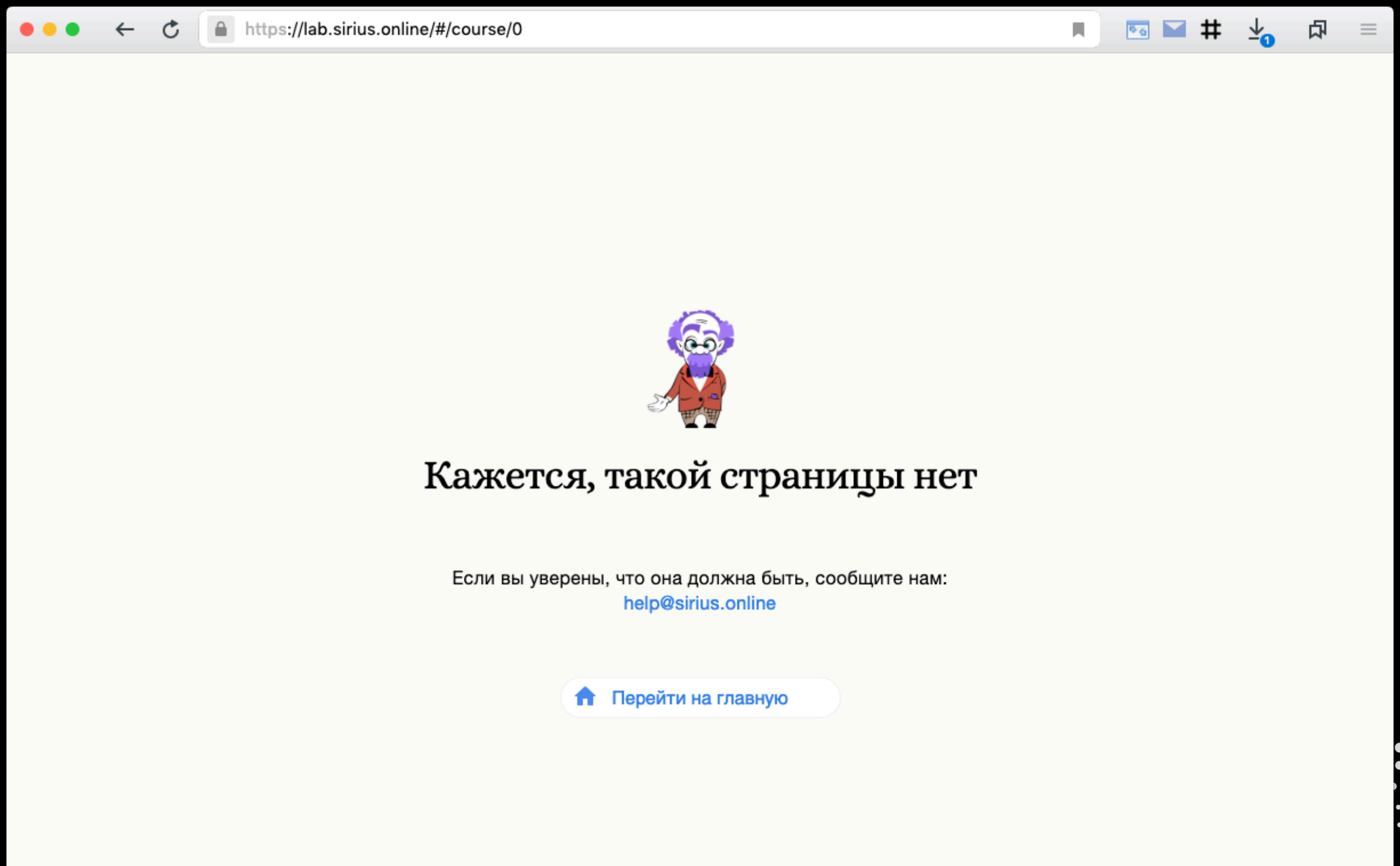
```
WITH RECURSIVE
  perms AS (SELECT * FROM permission_user WHERE user_id = $1),
  base AS
     (SELECT apo.parent, apu.is_rw, apu.is_r
        FROM permission_object AS apo
           LEFT JOIN perms AS apu ON apu.object_id = apo.id
        WHERE apo.type=$2 AND external_id=$3
     UNION ALL
        SELECT apo.parent_id, apu.is_rw, apu.is_r
           FROM base
              LEFT JOIN permission_object as apo ON base.parent_id = apo.id
              LEFT JOIN perms as apu on app.object_id = apo.id
          WHERE base.parent_id IS NOT NULL)
SELECT
    COALESCE(bool_or(is_rw), FALSE),
    COALESCE(bool_or(is_r), FALSE)
FROM base
```



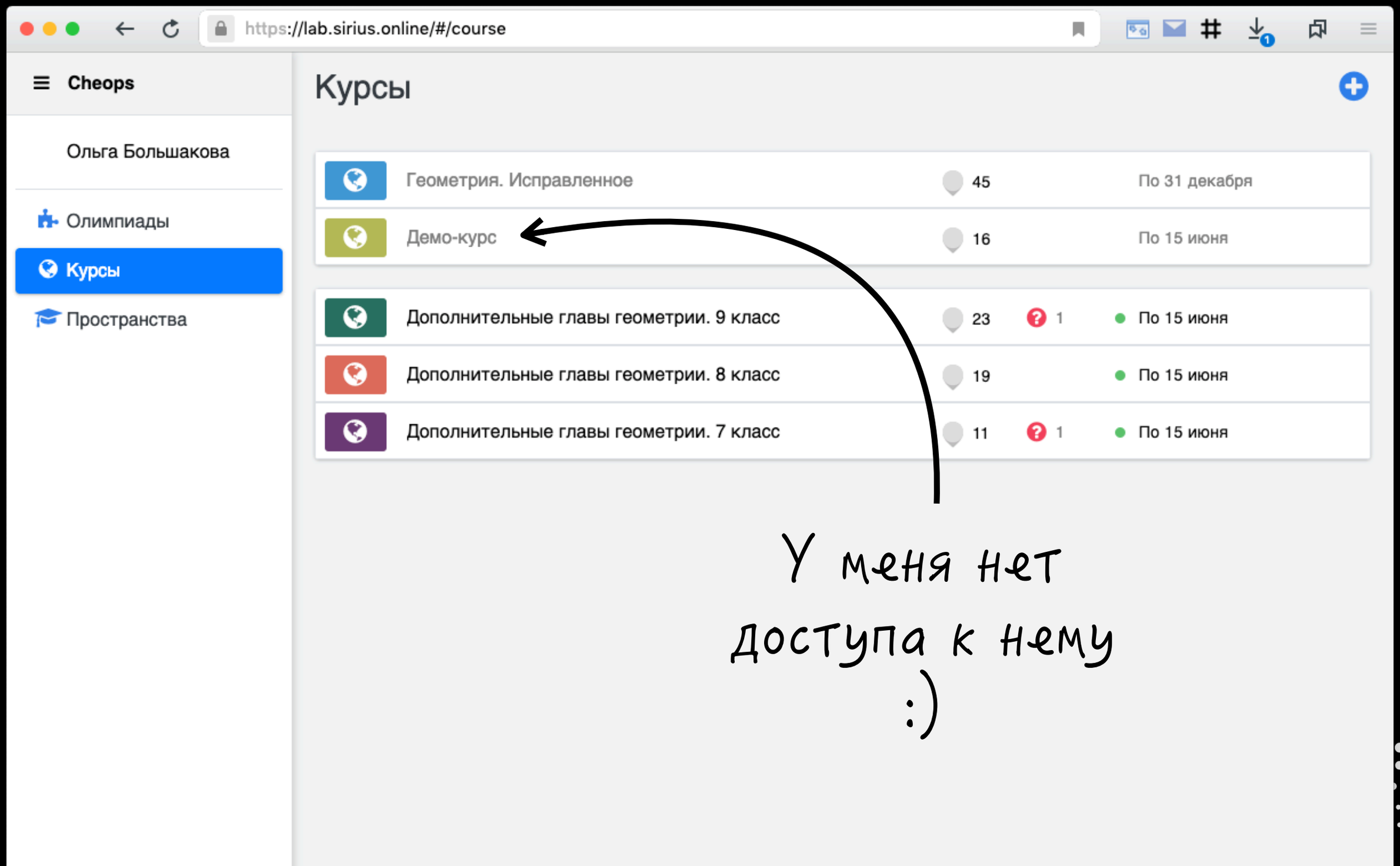
Новые требования



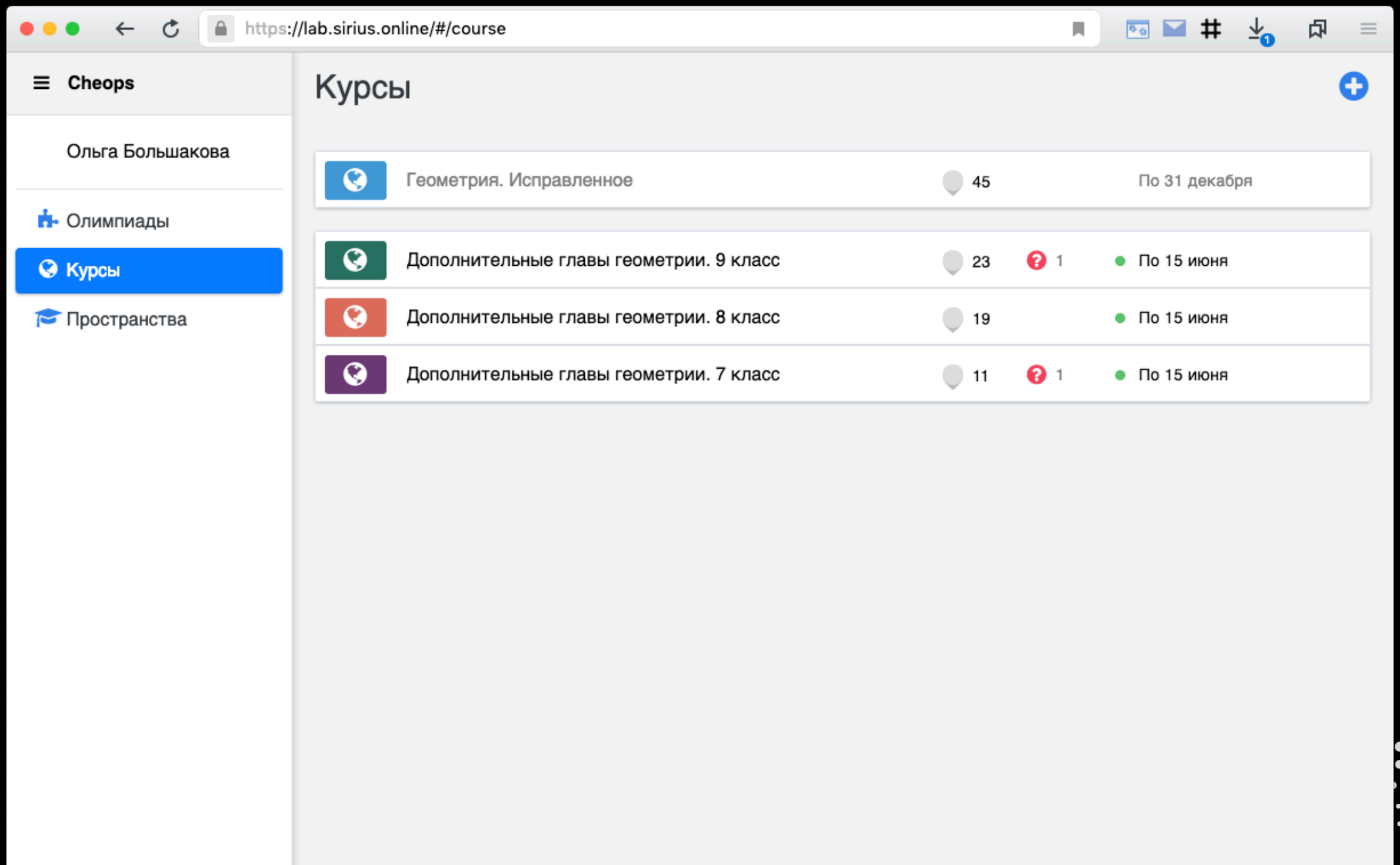
Новые требования



Новые требования



Новые требования



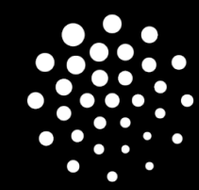
```haskell
module Auth (Handle(..)) where

data Handle = Handle {

    hasPermissionsFor :: UserId
                      -> PermissionTarget
                      -> IO Permissions

  , createObject      :: PermissionTarget
                      -> ObjectAddr
                      -> IO PermissionObjectId

  , ...
}
```



```haskell
module Auth (Handle(..)) where

data Handle = Handle {
  ...

  , getObjectsByPermission :: UserId
                           -> PermissionObject
                           -> Permissions
                           -> IO [PermissionObjectId]
}
```

Вводим
новый метод

```
new txn = Auth.Handle {
  ...
  , Auth.getObjectsByPermission =
      \uid typ b -> Db.run txn $
        Db.statement
          txn
          Permissions.getObjectsByPermission
          (b, typ, uid)
  ...
}
```

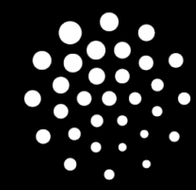
Реализуем новый метод в конкретном обработчике

```
loadCourses auth exec uid = do
    rowIds <- getObjectsByPermission auth
                                     uid
                                     “course”
                                     ReadOnly
                  <&> coerce
    listReports (FilterIds rowIds) ()
```

Итого

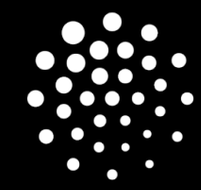
- Спроектировано решение, которое удовлетворяет всем требованиям
- Решение можно расширять, не затрагивая остальные части проекта

В таком виде решение использовалось в первый месяц жизни проекта



Но не все так радужно…

- Нас не заставляют проверять права
- Нам дают работать напрямую с базой, игнорируя сервис
- Не унифицирован возврат ошибок
- Не унифицировано создание прав
- Сложно тестировать



Люди и команды

Вики

Трекер

CHEOPSDEV-431: Метод, возвращающий новость по id

Протестировано. Обновлен 25 февраля 2019 г.. Создал Ольга Большакова 18 февраля 2019 г..

3. Не отображается новость для пользователя **с правами** только на новости этого курса - v6.95 Screenshot .

Лаборатория. Ноополис, Доделать

CHEOPSDEV-361: Метод для получения структуры модуля в курсе

Тестирование отложено. Обновлен 27 февраля 2019 г.. Создал Ольга Большакова 9 февраля 2019 г..

3. 401 вместо 200 для пользователя **с правами** courses(is_r) - v6.94 Screenshot . Толь, ещё у нас появилась следующая проблема: в API Лаборатории нет метода...

Лаборатория. Ноополис, Доделать

CHEOPSDEV-352: Метод, возвращающий список новостей по курсу

Тестирование отложено. Обновлен 26 февраля 2019 г.. Создал Ольга Большакова 7 февраля 2019 г..

2. НЕ отображается список новостей для пользователя **с правами** только на чтение - v6.95 Screenshot Отдавать 200 для прав: root - r courses - r course{cid} - r news{cid} - r . 3...

Лаборатория. Ноополис, Доделать

CHEOPSDEV-339: Метод, возвращающий вопросы-ответы по курсу

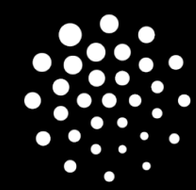
Тестирование отложено. Обновлен 12 марта 2019 г.. Создал Ольга Большакова 1 февраля 2019 г..

2. 403 вместо 200 для пользователя **с правами** root-r - v6.104.1 Screenshot . Svetlana Bersneva, [12 марта 2019 г., 14:29:51]: создала новый курс с тремя модулями в одном из модулей задала...

Лаборатория. Ноополис, Смоук

Что делать?

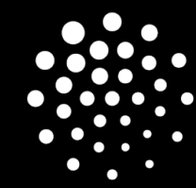
1. Спрятать конкретные реализации в internal модули, не экспортировать их



Что делать?

1. Спрятать конкретные реализации в internal модули, не экспортировать их

2. Высокоуровневый API, специфичный для проекта

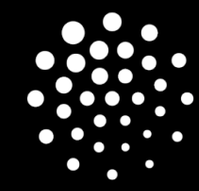


Permissions

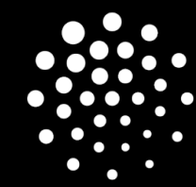
Высокоуровневый проектно-специфичный интерфейс

Хотим получить

- Форсирование проверки наличия прав в API
- Единую логику обработки ошибок
- Единый интерфейс создания новых прав и объектов



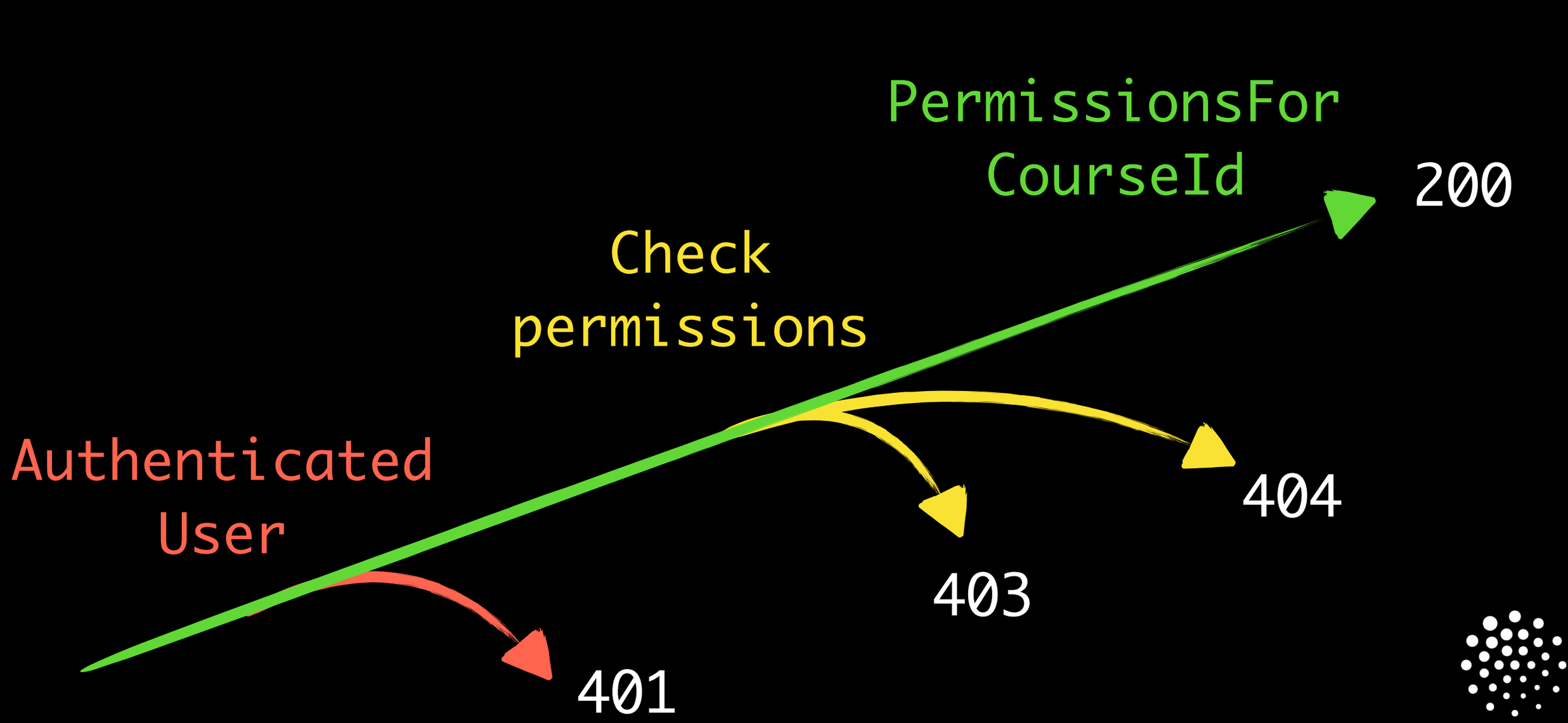
```
newtype PermissionsFor (t :: Type)
  = PermissionsFor UserId
```



Проверка токена | Проверка прав | Безопасная работа

Servant | Handler | Application logic

`Получаем ID` | `ID -> PermissionsFor t` | `PermissionsFor t`

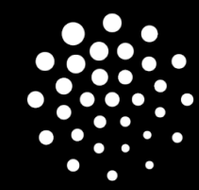


Определяет требуемые права

```
mkPermissionsFor :: MkPermissionsFor a
                 => Auth.Handle
                 -> UserId
                 -> MkPermissionsInput a
                 -> Handler (PermissionsFor a)
```

Управляет роутингом

Гарантирует проверку прав



```haskell
{-# LANGUAGE TypeFamilies #-}

class MkPermissionsFor a where

  type MkPermissionsInput a :: Type

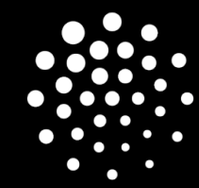
  mkTarget      :: MkPermissionsInput a
                -> PermissionTarget

  mkPermissions :: MkPermissionsInput a
                -> RequiredPermission
```

Что нам необходимо, чтобы найти узел в дереве прав

Узел в дереве

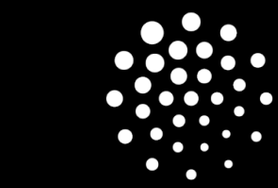
Требуемые права



Примеры

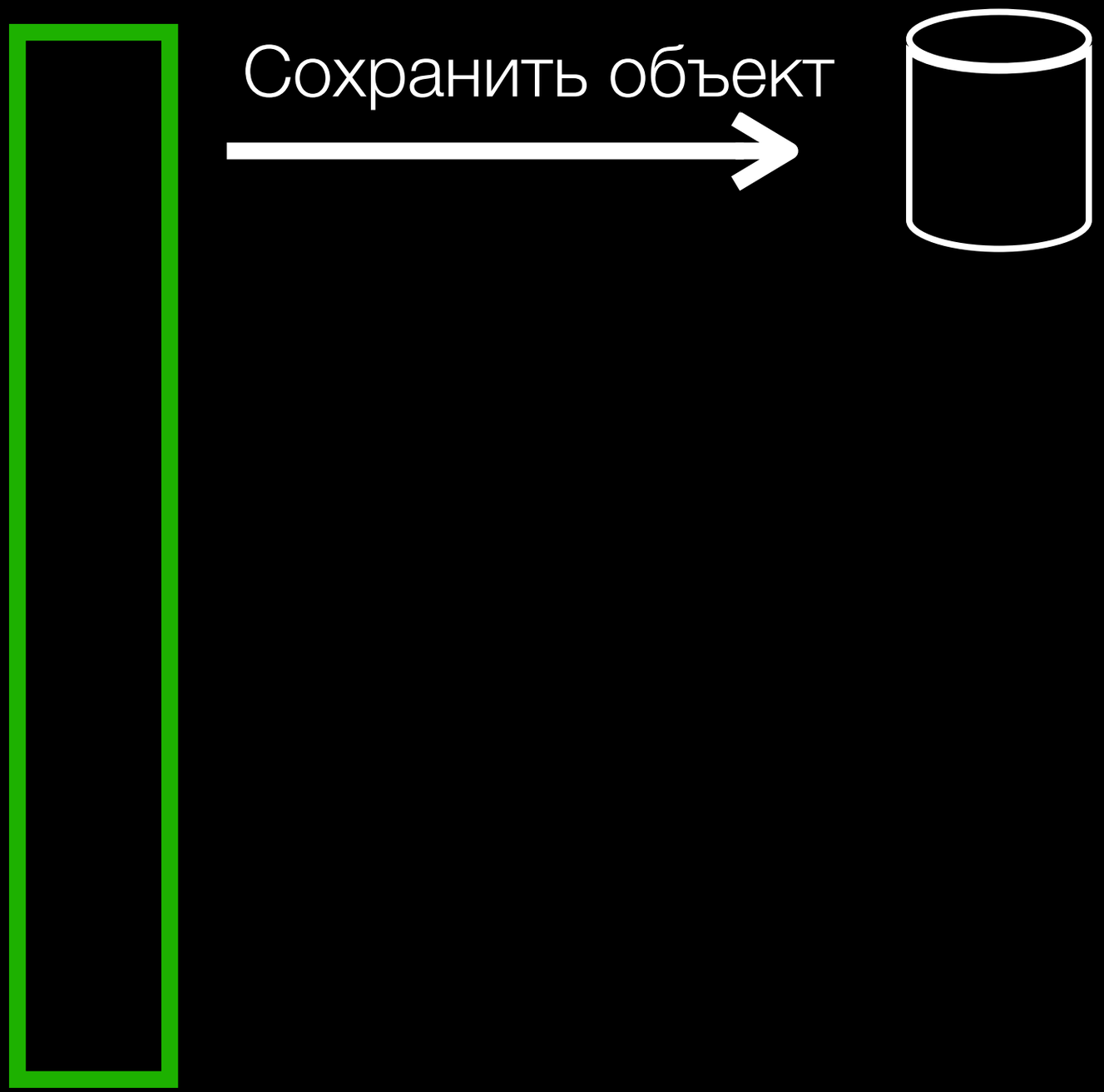
```
instance MkPermissionsFor ReadCourse where
  type MkPermissionsInput ReadCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
  mkPermissions _ = RequiredPermissionsRead


instance MkPermissionsFor EditCourse where
  type MkPermissionsInput EditCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
  mkPermissions _ = RequiredPermissionsWrite
```

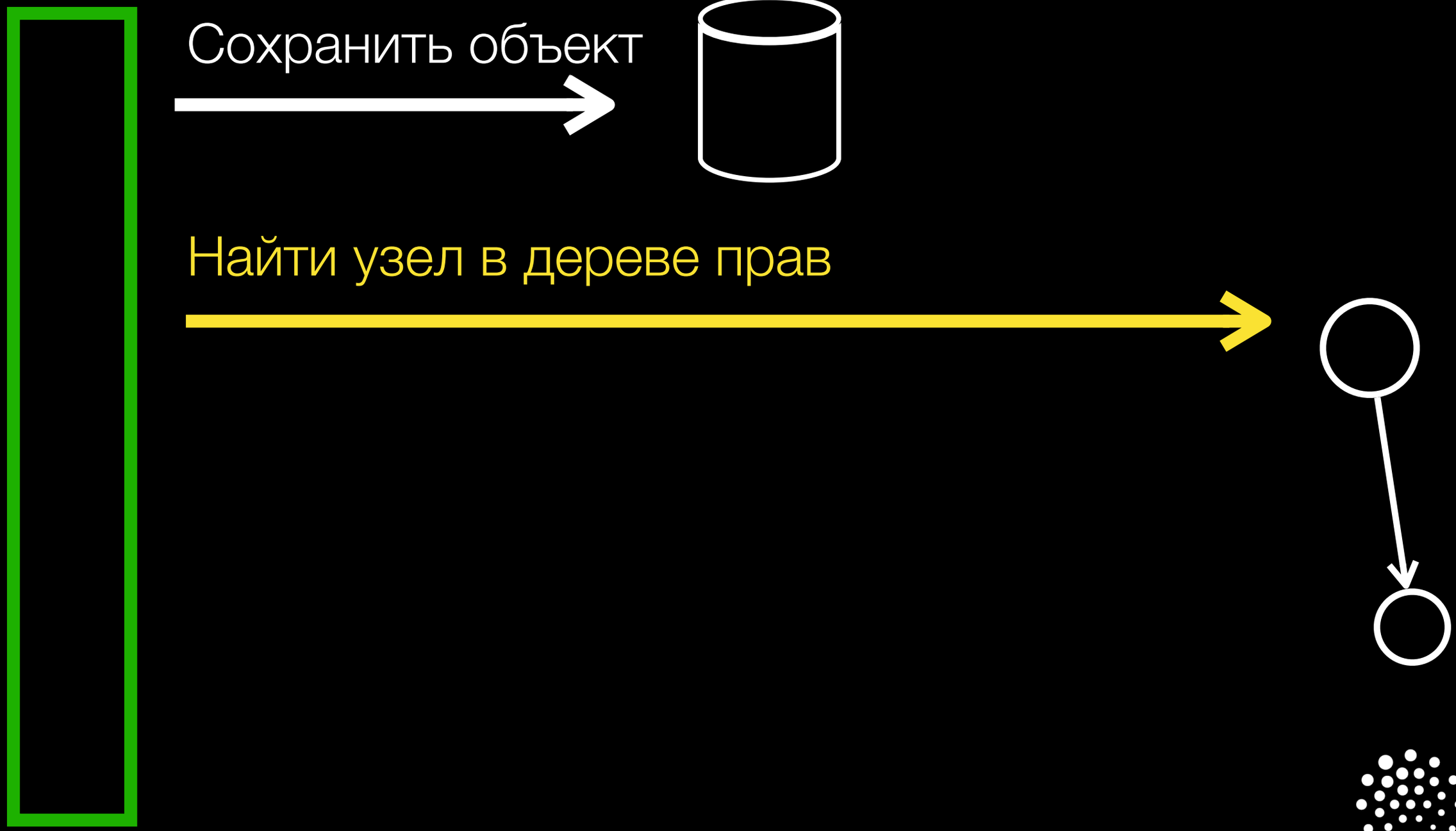


Создание прав

Создание прав

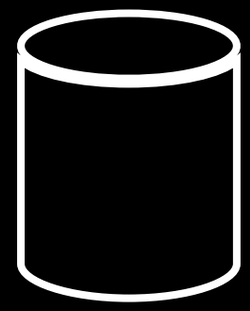


Создание прав



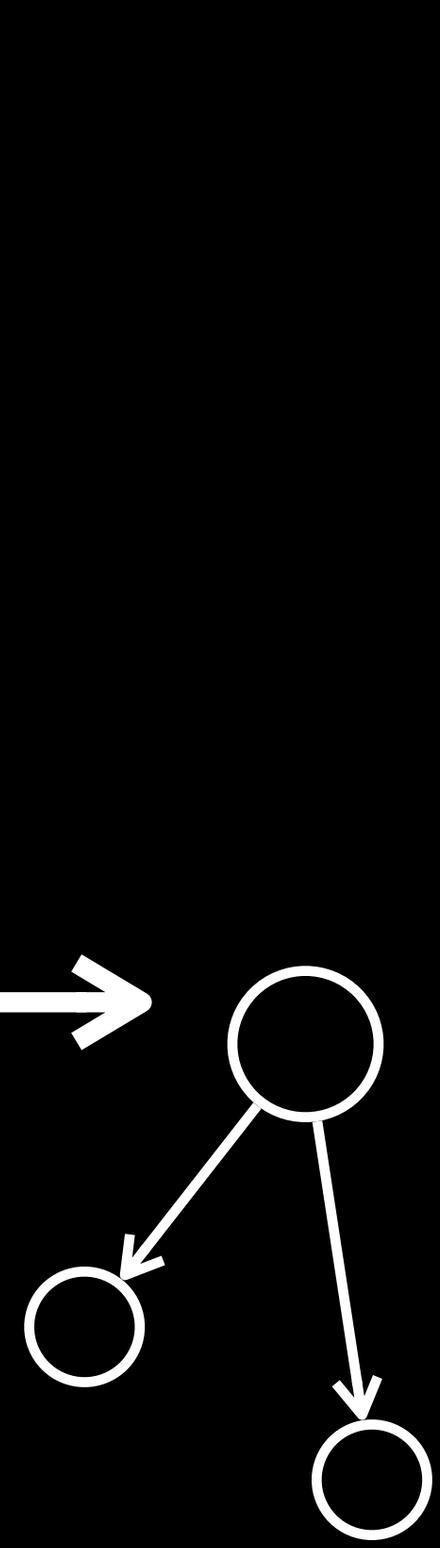
Создание прав

Сохранить объект



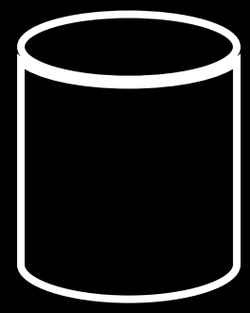
Найти узел в дереве прав

Сохранить новый дочерний узел

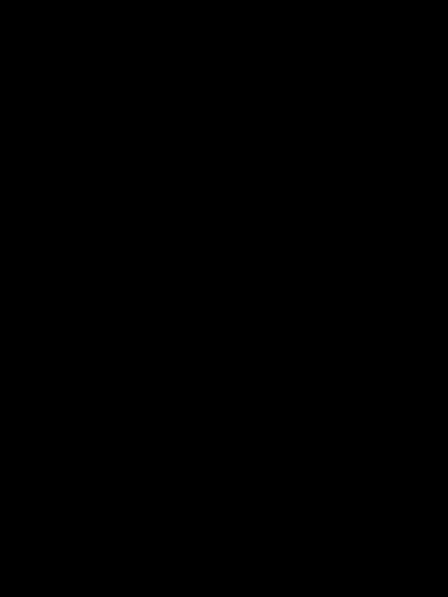


Создание прав

Сохранить объект



Найти узел в дереве прав

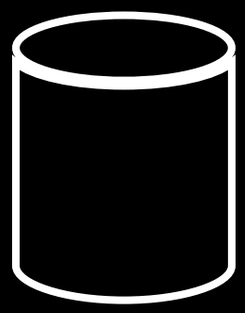


Сохранить новый дочерний узел

Обновить права пользователя

Создание прав

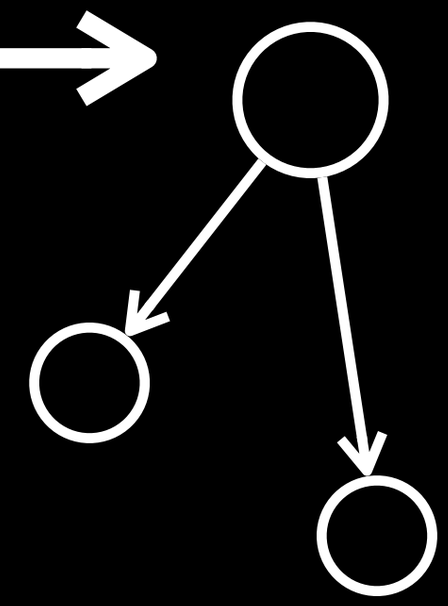
Сохранить объект



Для каждого объекта!

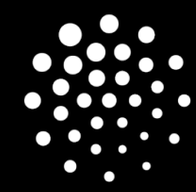
А хочется ...

Найти узел в дереве прав



Сохранить новый дочерний узел

Обновить права пользователя



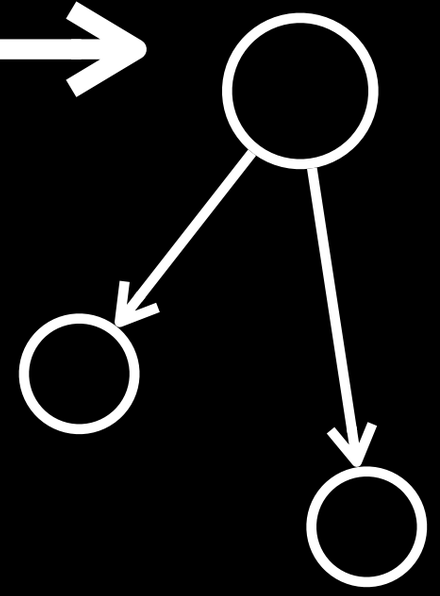
Сохранить объект

Событие "Новый объект"

Найти узел в дереве прав

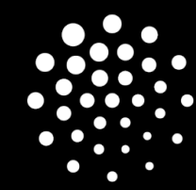
Сохранить новый
дочерний узел

Обновить права
пользователя



Построение дерева прав

```
class IsPermissionsTarget c where
    permissionsTarget :: c -> PermissionsTarget
    permissionsParent :: c -> PermissionsTarget
```

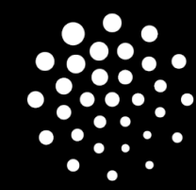


Пример. Создание новости

```
instance IsPermissionsTarget NewsPerm where

    permissionsTarget (NewsPerm cid) =
                PermissionsTarget “news”
                                  (coerce cid)

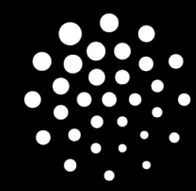
  permissionsParent (NewsPerm cid) =
            PermissionsTarget “course”
                                  (coerce cid)
```



Permissions. Итоги

- Проблемы ушли, новые права и проверки прав добавлялись без труда
- Добавление новых прав не затрагивает основной код проекта

С момента внедрения эта версия успешно работает в продакшене

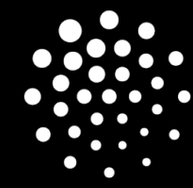


Что ещё можно улучшить?

Не дублировать типы

```
instance MkPermissionsFor ReadCourse where
  type MkPermissionsInput ReadCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
  mkPermissions _ = RequiredPermissionsRead


instance MkPermissionsFor EditCourse where
  type MkPermissionsInput EditCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
  mkPermissions _ = RequiredPermissionsWrite
```



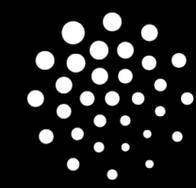
Хранить доступ на уровне типов

```haskell
class MkPermissionsFor a where

  type MkPermissionsInput a :: Type

  mkTarget       :: MkPermissionsInput a
                 -> PermissionTarget

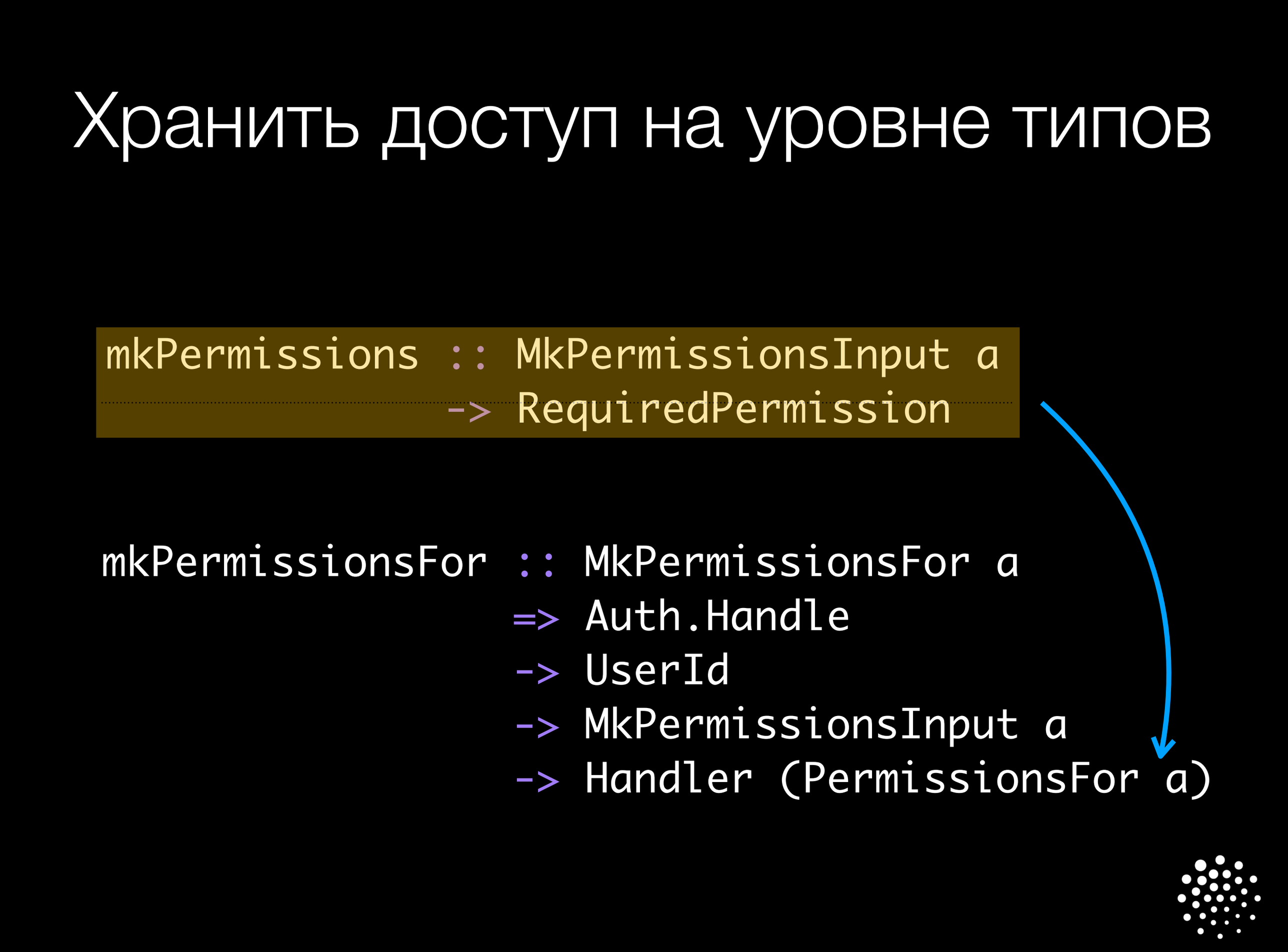
  mkPermissions :: MkPermissionsInput a
                -> RequiredPermission
```



Хранить доступ на уровне типов

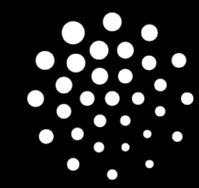
```
mkPermissions :: MkPermissionsInput a
              -> RequiredPermission
```

```
mkPermissionsFor :: MkPermissionsFor a
                 => Auth.Handle
                 -> UserId
                 -> MkPermissionsInput a
                 -> Handler (PermissionsFor a)
```

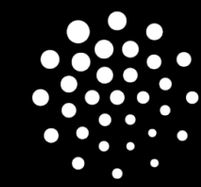


Хранить доступ на уровне типов

```
mkPermissionsFor
  :: (MkPermissionsFor a, MkPermission t)
  => Auth.Handle
  -> UserId
  -> MkPermissionsInput a
  -> Handler (PermissionsFor t a)
```

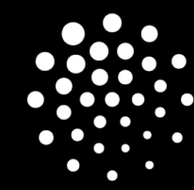


```
newtype PermissionsFor
  (a :: AccessLevel) (t :: Type)
  = PermissionsFor UserId
```

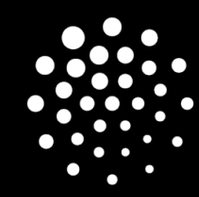


```haskell
data AccessLevel =
        ReadOnly | ReadWrite

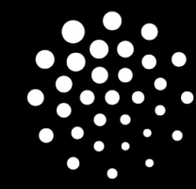
newtype PermissionsFor
  (a :: AccessLevel) (t :: Type)
  = PermissionsFor UserId
```



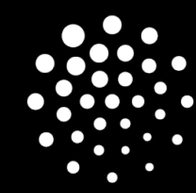
```haskell
instance MkPermissionsFor ReadCourse where
  type MkPermissionsInput ReadCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
  mkPermissions _ = RequiredPermissionsRead
```



```haskell
instance MkPermissionsFor ReadCourse where
  type MkPermissionsInput ReadCourse = CourseId
  mkTarget cid     = PermissionsTarget “course”
                                       (cid^.coerced)
mkPermissions _ = RequiredPermissionsRead
```



```haskell
instance MkPermissionsFor CourseId where
  type MkPermissionsInput CourseId = CourseId
  mkTarget cid    = PermissionsTarget “course”
                                     (cid^.coerced)
```



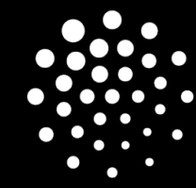
Права к нескольким объектам

```
handle :: CourseId
       -> UserId
       -> PermissionFor ‘ReadOnly CourseId
       -> PermissionFor ‘ReadOnly NewsId
       -> Handler Reply
```

Повтор

Права к нескольким объектам

```
handle :: CourseId
       -> UserId
       -> PermissionFor [(‘ReadOnly, NewsId),
                         (‘ReadOnly, CourseId)]
       -> Handler Reply
```

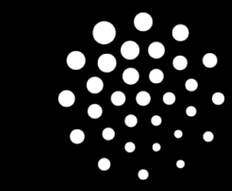


Типы пишут код за нас

```
class MkRequest xs where
  type MkRequestParams xs = *
  mkRequest :: MkRequestParams xs
            -> [PermissionTarget]

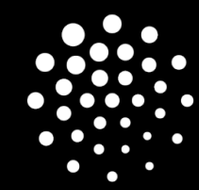
instance MkPermissionsFor x => MkRequest [x] where
  type MkRequestParams [x] = PermissionsInput x
  mkRequest x              = [mkPermissionObject x]

instance MkRequest xs => MkRequest (x ': xs) where
  type MkRequestParams (x ': xs) =
         (PermissionsInput x, MkRequestParams xs)
  mkRequest (x, xs)              =
         mkPermissionObject x    : mkRequest xs
```



Другие улучшения

- Хранить `MkPermissionsInput` внутри структуры `MkPermissionsFor`
- Использовать расширение `DerivingVia` для уменьшения количества boilerplate-кода



Проект растёт и развивается. Сервис должен быть к этому готов

Скоро

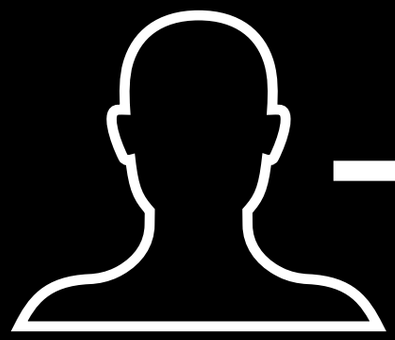
Завтра

Сегодня

Методистов	10^{1}	10^{2}	10^{3}
Объектов	10^{3}	10^{5}	10^{6}
Курсов	10^{1}	10^{2}	10^{4}

Нужно думать о производительности уже сейчас

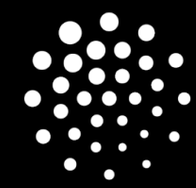
DB



Действие

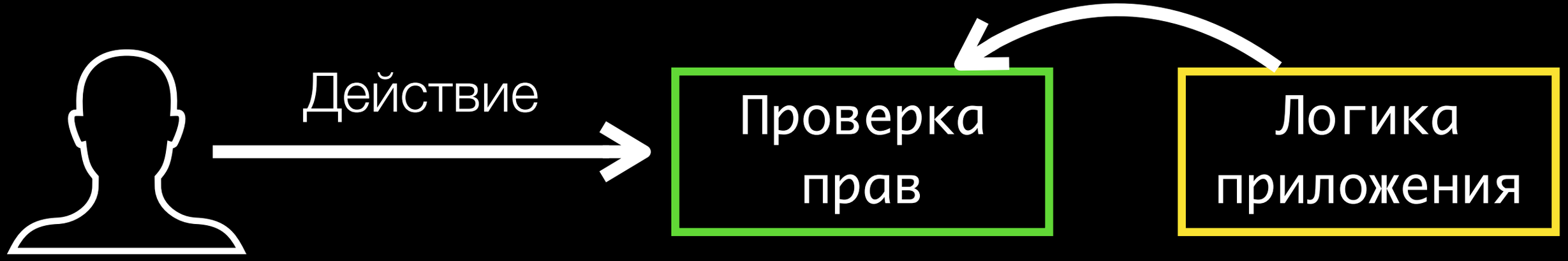
Проверка прав

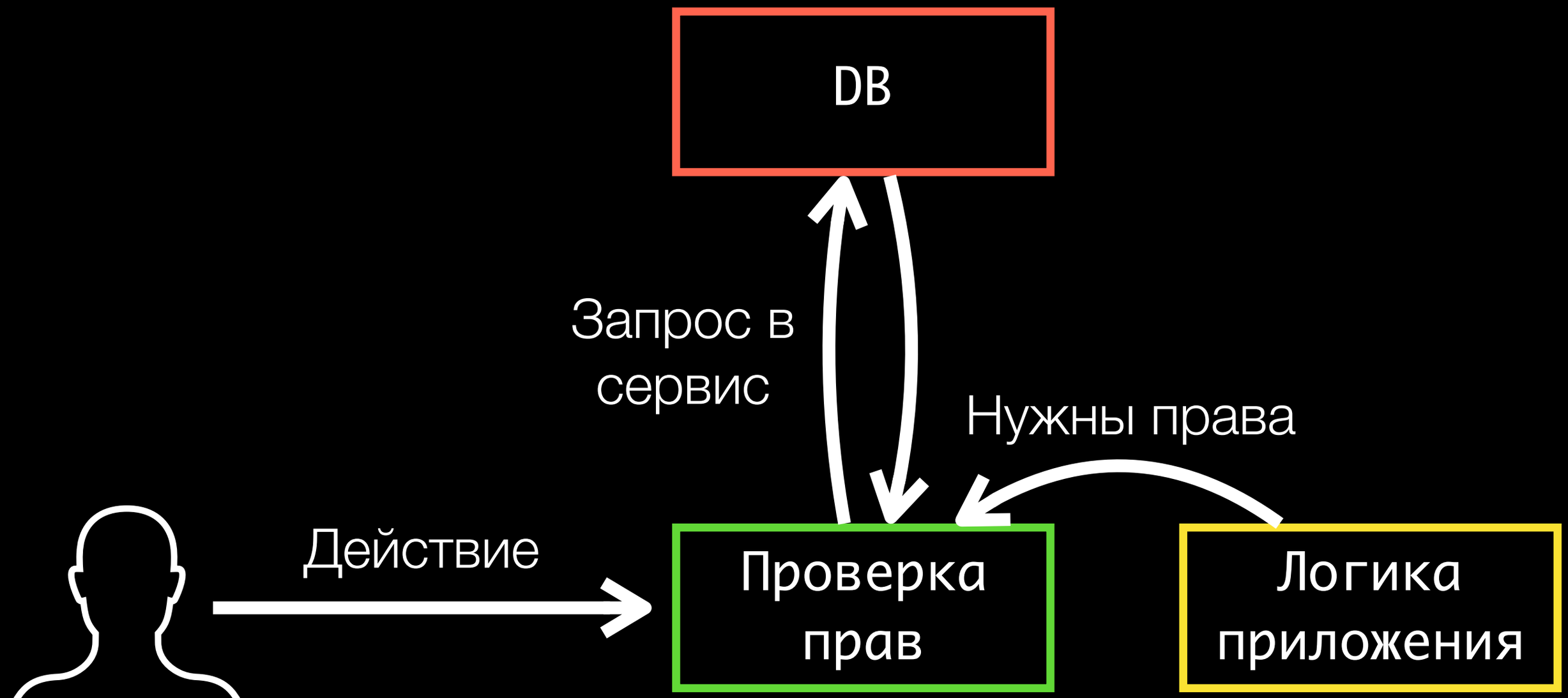
Логика приложения

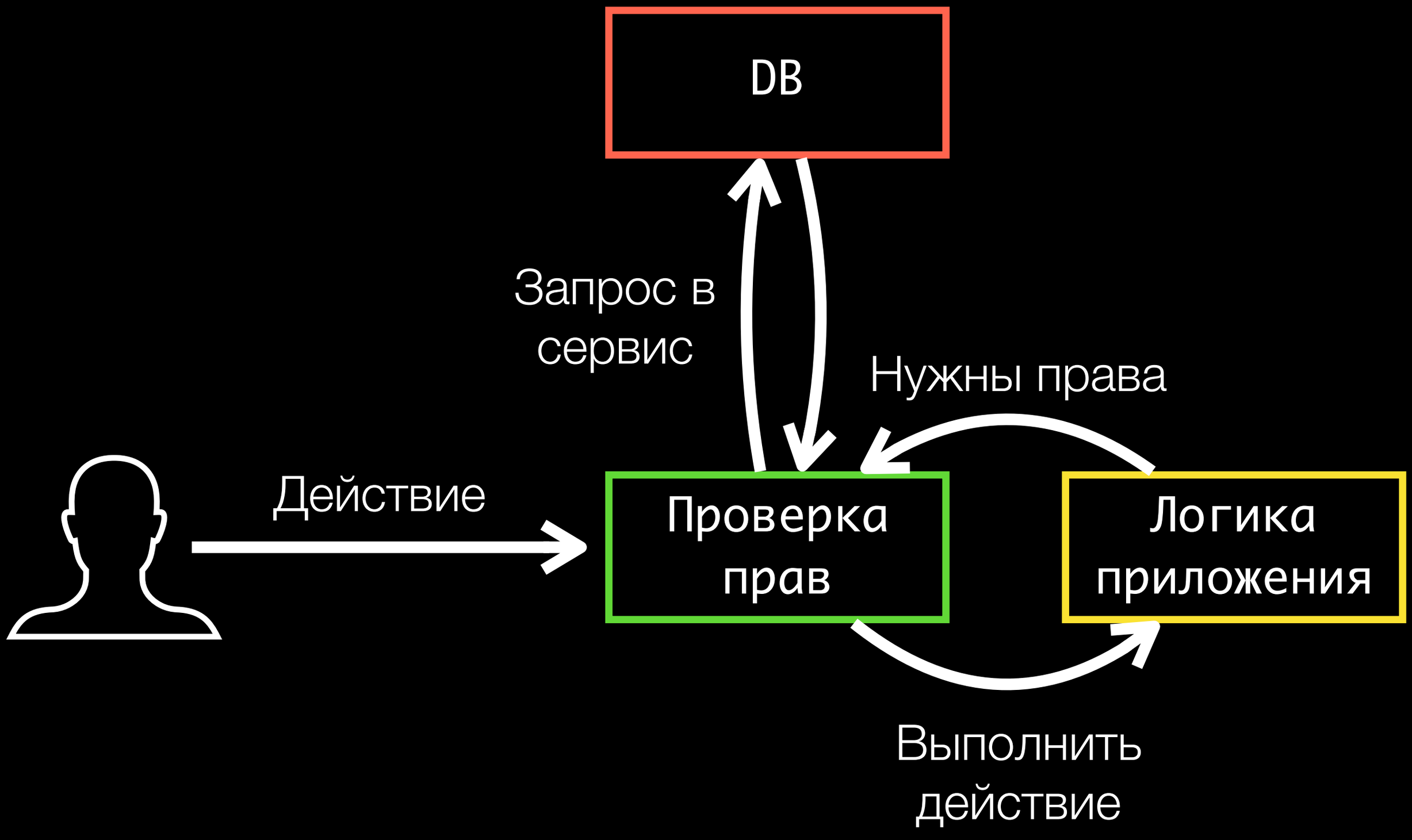


DB

Нужны права




DB
Запрос в сервис
Нужны права
Действие
Проверка прав
Логика приложения


DB
Запрос в сервис
Нужны права
Действие
Проверка прав
Логика приложения
Выполнить действие

DB

На каждый запрос!

Запрос в сервис

Нужны права

Действие

Проверка прав

Логика приложения

Выполнить действие

Explain sql

Operation	Params	Rows	Total Cost
Select		1	176.01
Union (Aggregate)		1	176.01
Index Scan	table: permission_user; ind...	2	4.25
Union (Recursive Union)		101	169.23
Nested Loops (Nested Loop)		1	2.56
Index Scan	table: permission_object; in...	1	2.5
Access (CTE Scan)		2	0.04
Hash Join		10	16.46
Nested Loops (Nested Loop)		10	16.35
Access (WorkTable Scan)		10	0.2
Index Scan	table: permission_object; in...	1	1.62
Transformation (Hash)		2	0.04
Access (CTE Scan)		2	0.04
Access (CTE Scan)		100	2.02

Что можно изменить?

DB

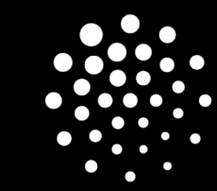
Вот мои права

Нужны права

Действие

Проверка прав

Логика приложения



JWT-токен

DB

Вот мои права

Нужны права

Действие

Проверка прав

Логика приложения



JWT-токен

DB

Вот мои права

Нужны права

Действие

Проверка прав

Логика приложения

OK!

JWT-токен

DB

Вот мои права

Нужны права

Действие

Проверка прав

Логика приложения

OK!

Выполнить действие

JWT-токен

DB

Вот мои права

Нужны права

Действие

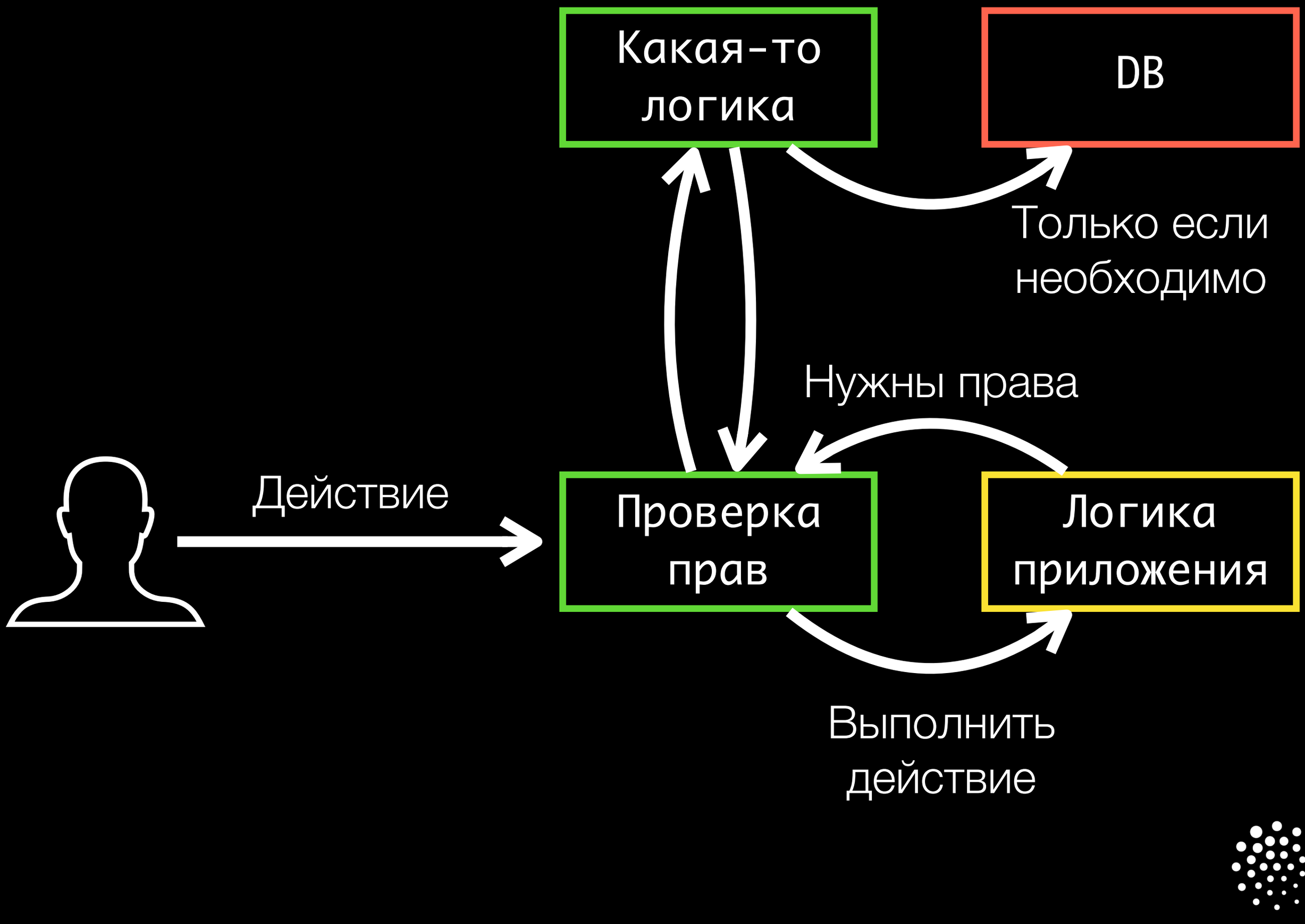
Проверка прав

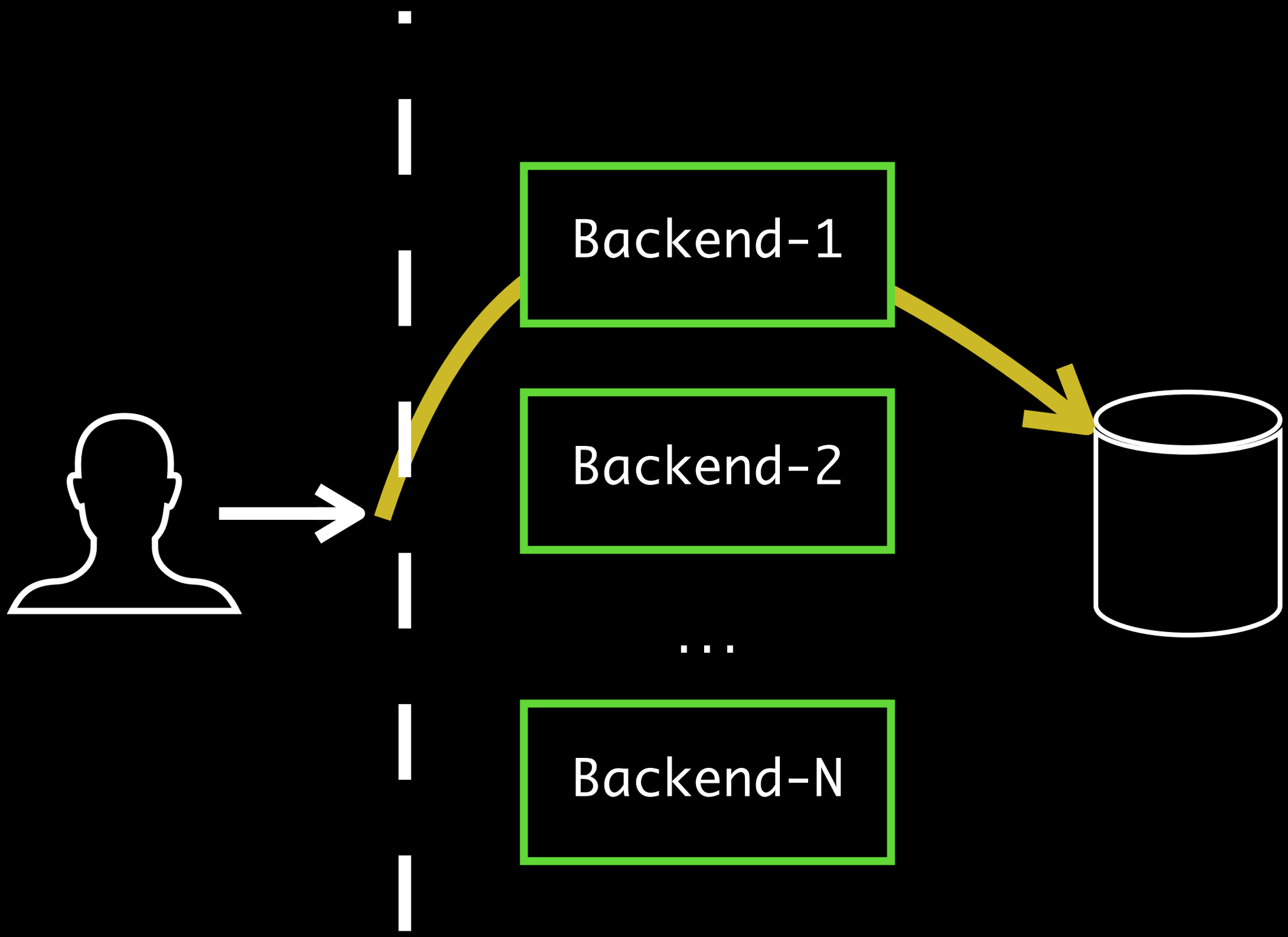
Логика приложения

OK!

Выполнить действие

Размер токена vs сложность протокола


Какая-то логика
DB
Только если необходимо
Нужны права
Действие
Проверка прав
Логика приложения
Выполнить действие

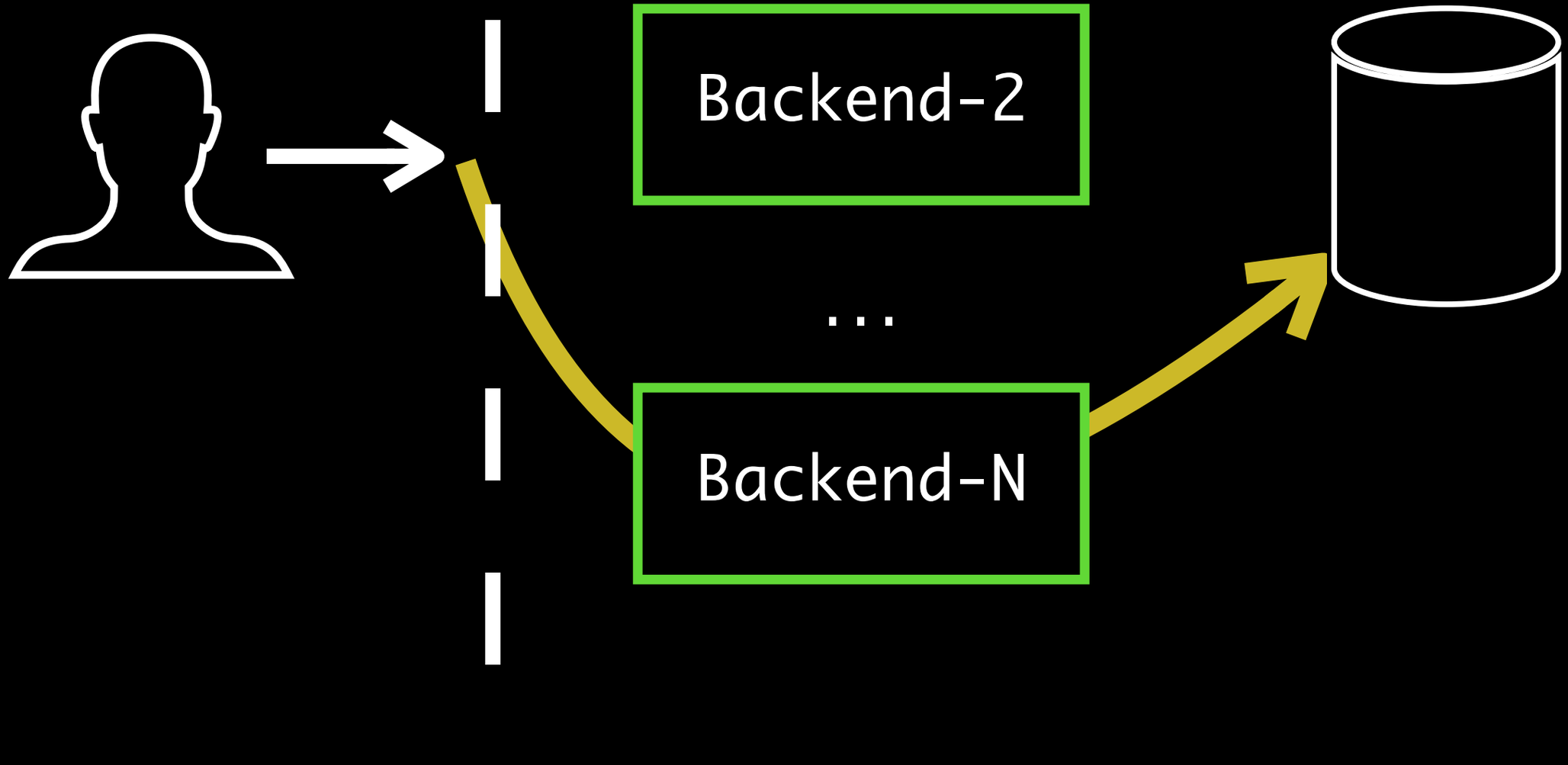

Backend-1
Backend-2
...
Backend-N

Backend-1

Backend-2

...

Backend-N

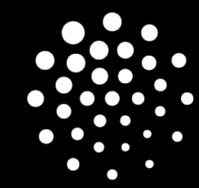


Допущения

1. Мы готовы ждать, пока права пользователя "прорастут"

2. Мы готовы ждать, пока отнятые права пользователя "отвалятся"

3. Мы НЕ готовы ждать, пока появятся права на только что созданный объект

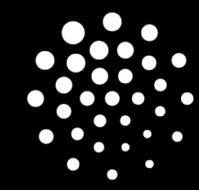


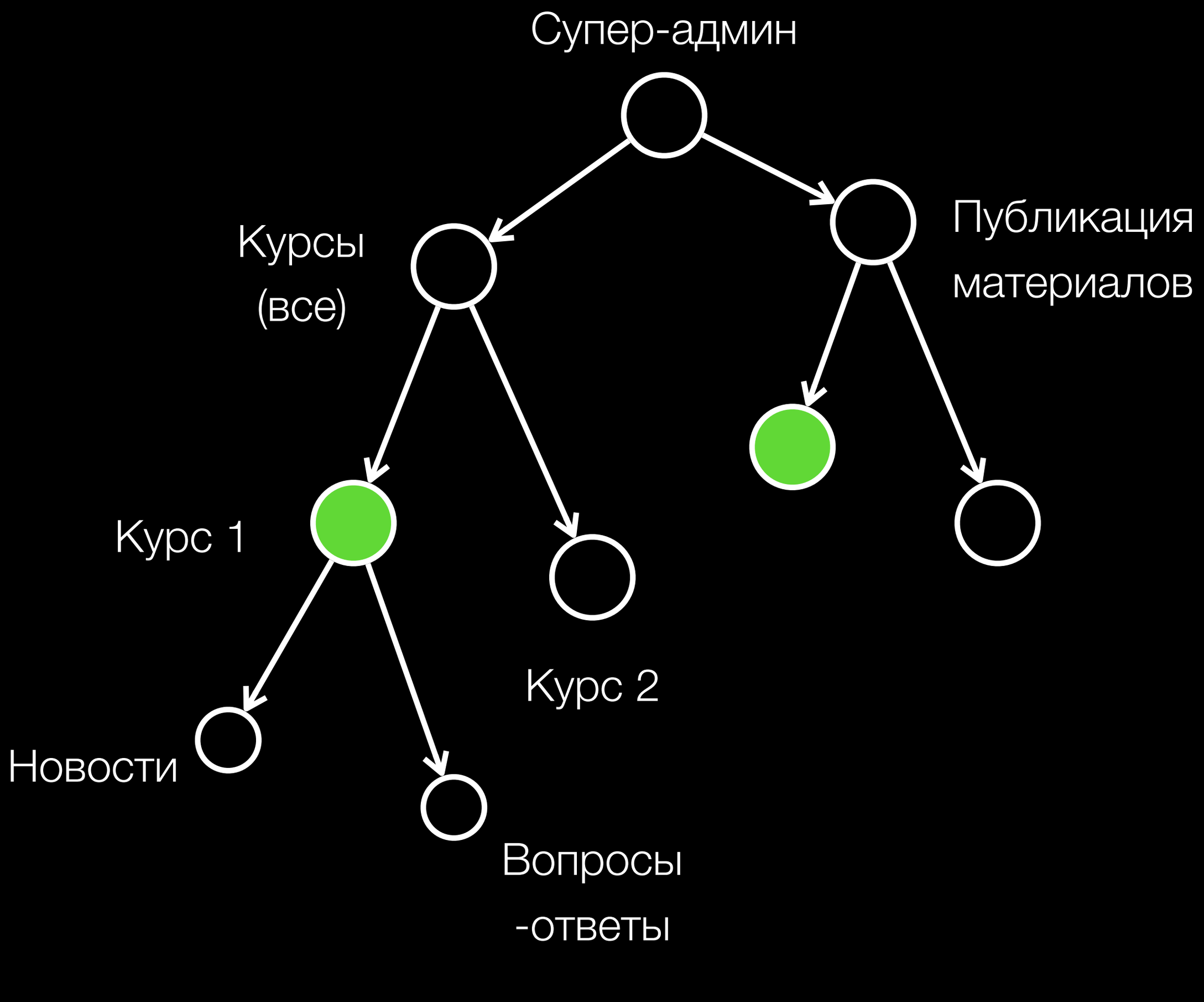
Простое кеширование (LRU)

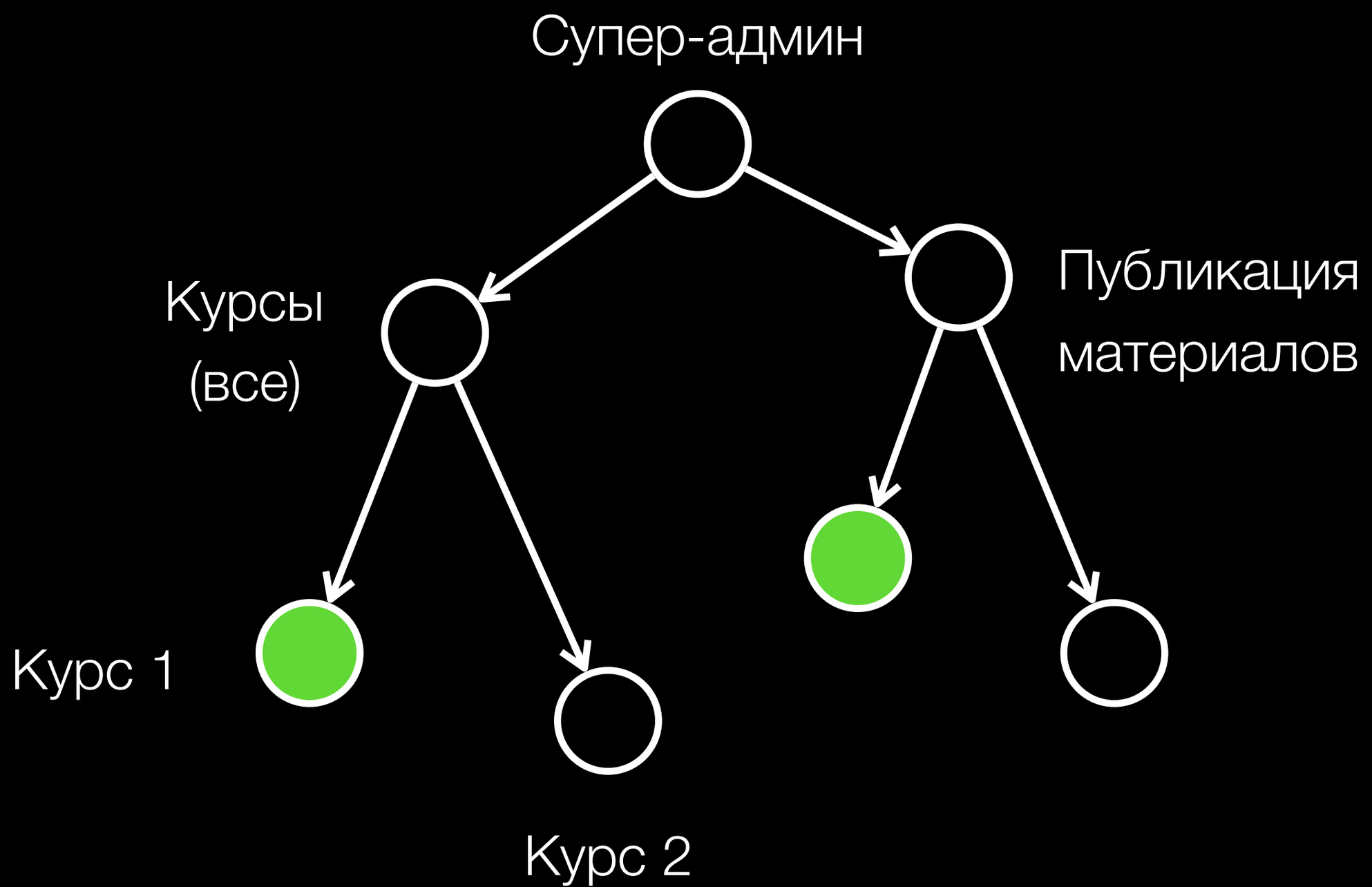
- Если ответа на запрос нет в кеше, то отправляем запрос в db-реализацию
- Нельзя хранить негативные ответы

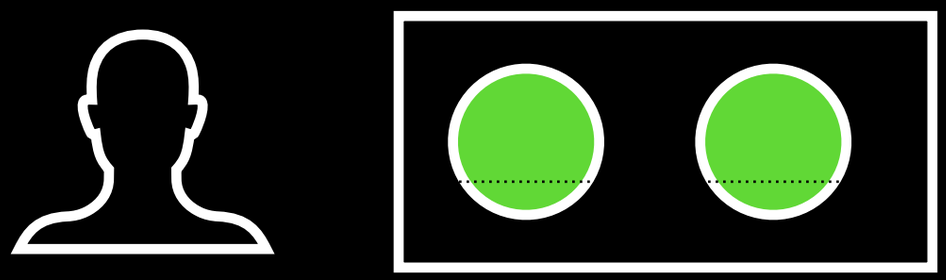
Сложное кеширование

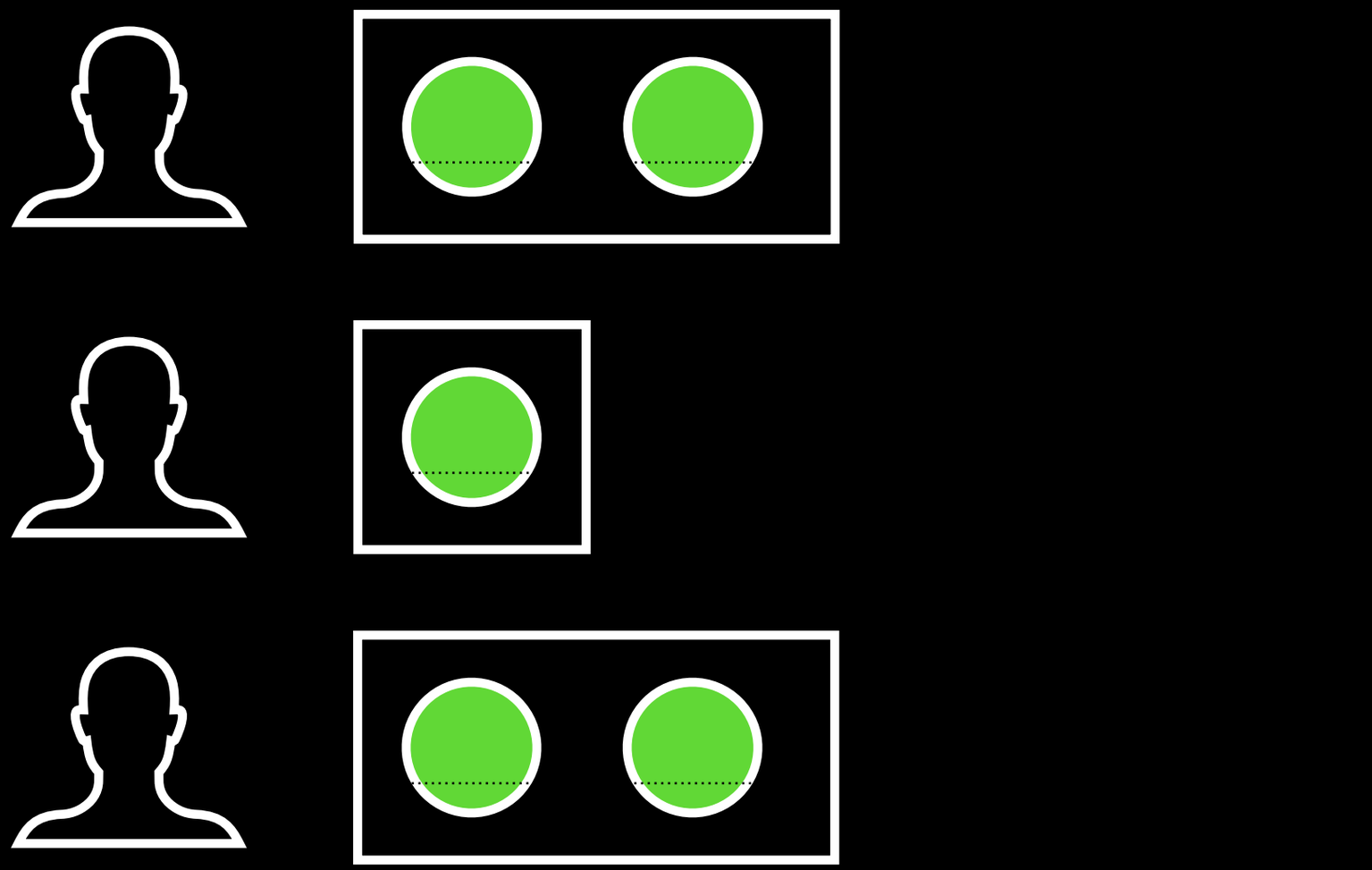
- Храним функции `UserId -> PermissionTarget -> Permissions`
- В основе — сжатое префиксное дерево
- Периодическое обновление

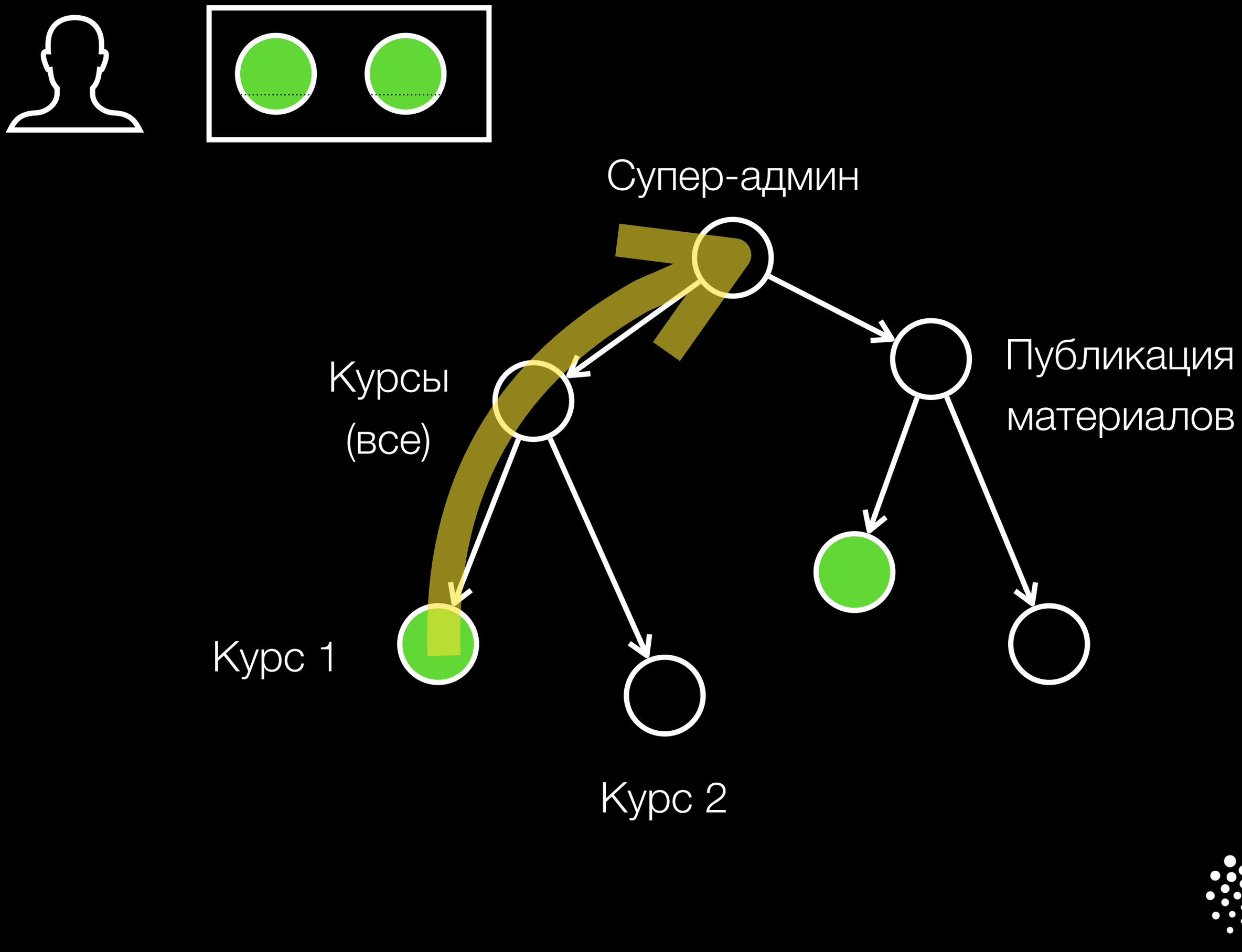


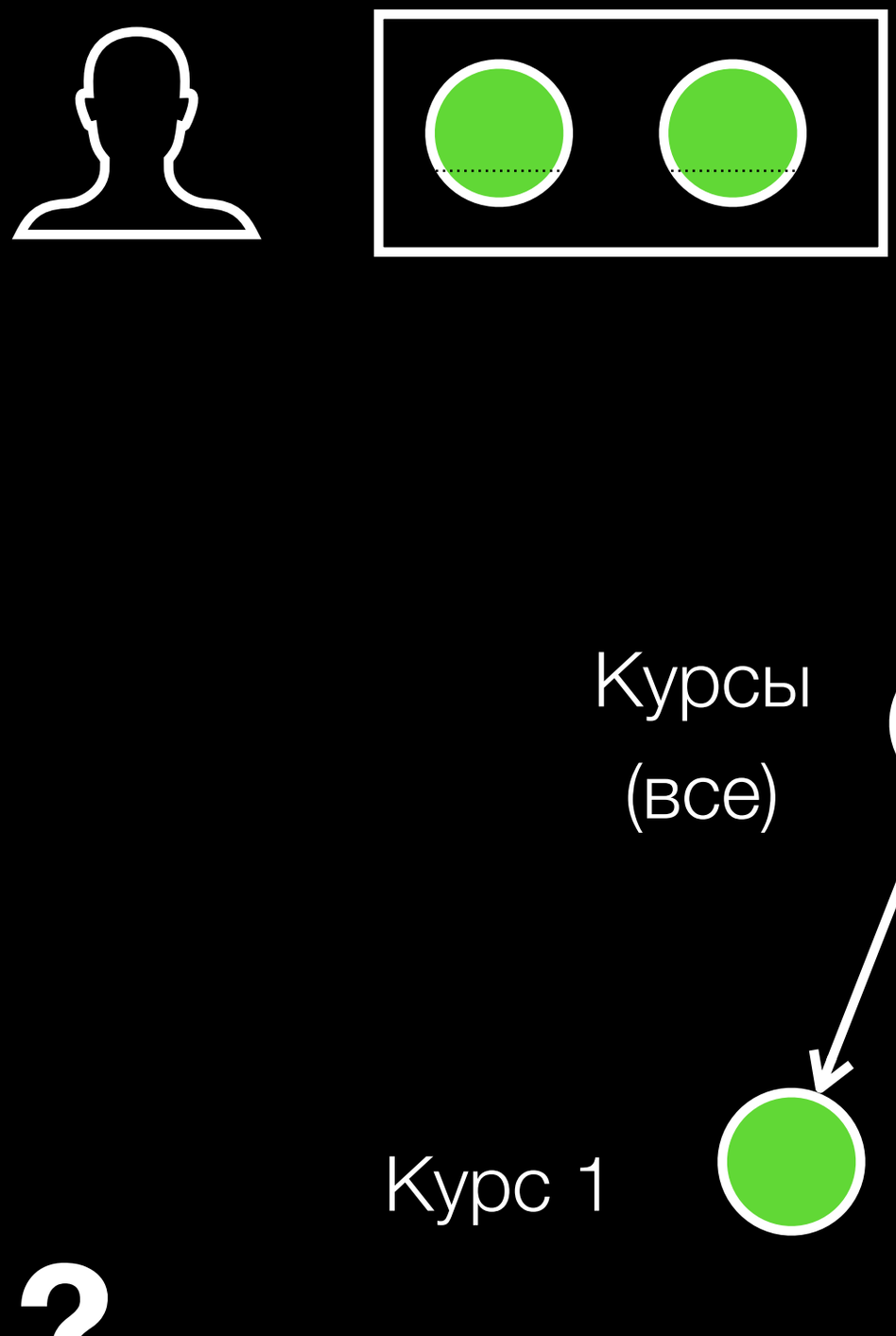

Супер-админ
Курсы
(все)
Публикация
материалов
Курс 1
Курс 2
Новости
Вопросы
-ответы

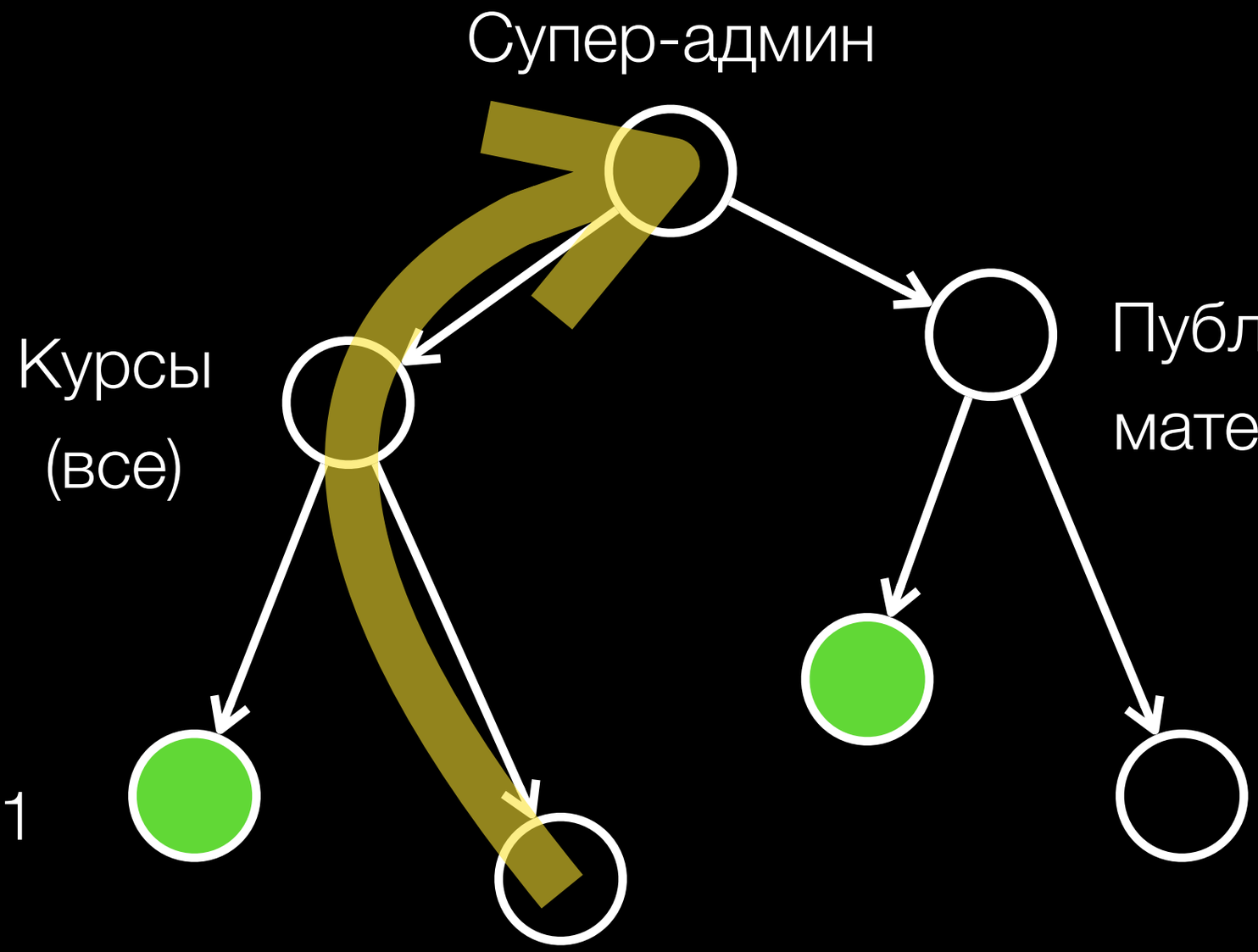

Супер-админ
Курсы
(все)
Публикация
материалов
Курс 1
Курс 2






Супер-админ
Курсы
(все)
Публикация
материалов
Курс 1
Курс 2

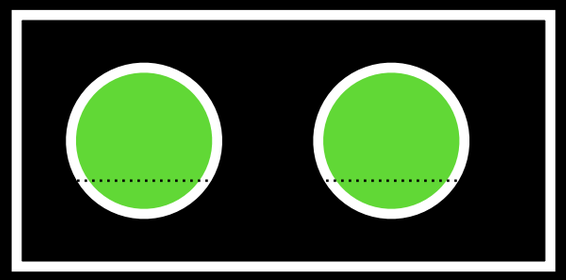




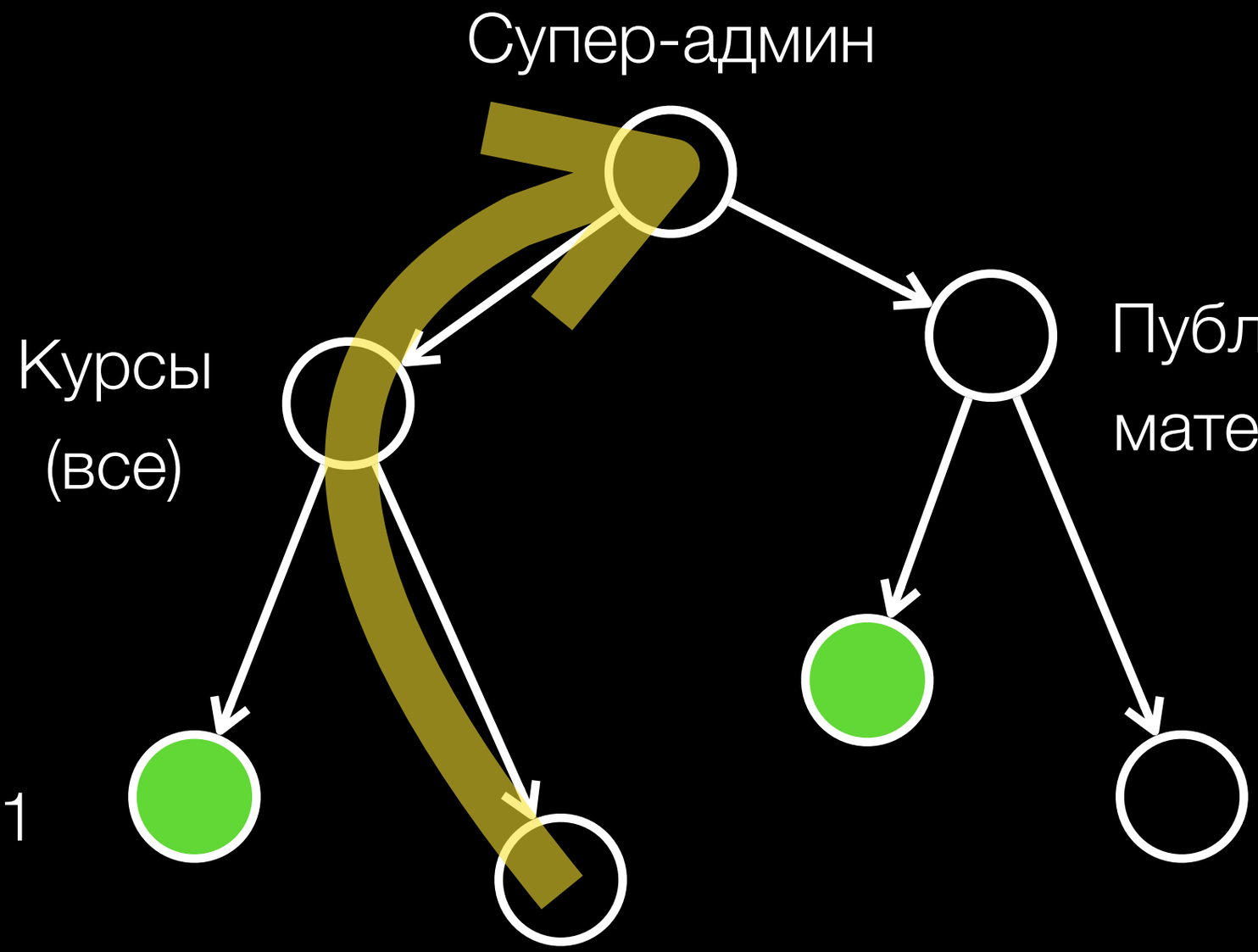
?

1. Нет доступа
2. Нет объекта в сохранённом дереве





\+ Известные объекты

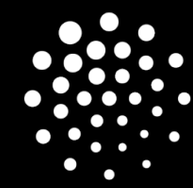


?

1. Нет доступа
2. ~~Нет объекта в сохранённом дереве~~

LRU vs Сложный кеш

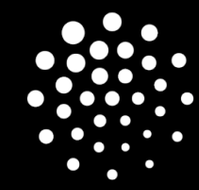
- Построение метрик
  - Cache-hit
  - Размер структуры
  - Время ребалансировки и модификации
- Настраиваемость и воспроизводимость подхода



Итоги

Требовалось разработать гибкую систему прав

1. Реализована простая версия, с которой можно работать
2. Введён высокоуровневый интерфейс, позволяющий облегчить работу
3. Что дальше — реализация с кешированием, которая позволит выдерживать нарастающую нагрузку



Спасибо!

alexander.vershilov@sirius.online