



> Конспект > 5 урок > Методология Anchor Modeling: Практикум

> [Преобразование Data Vault в Якорную модель](#)

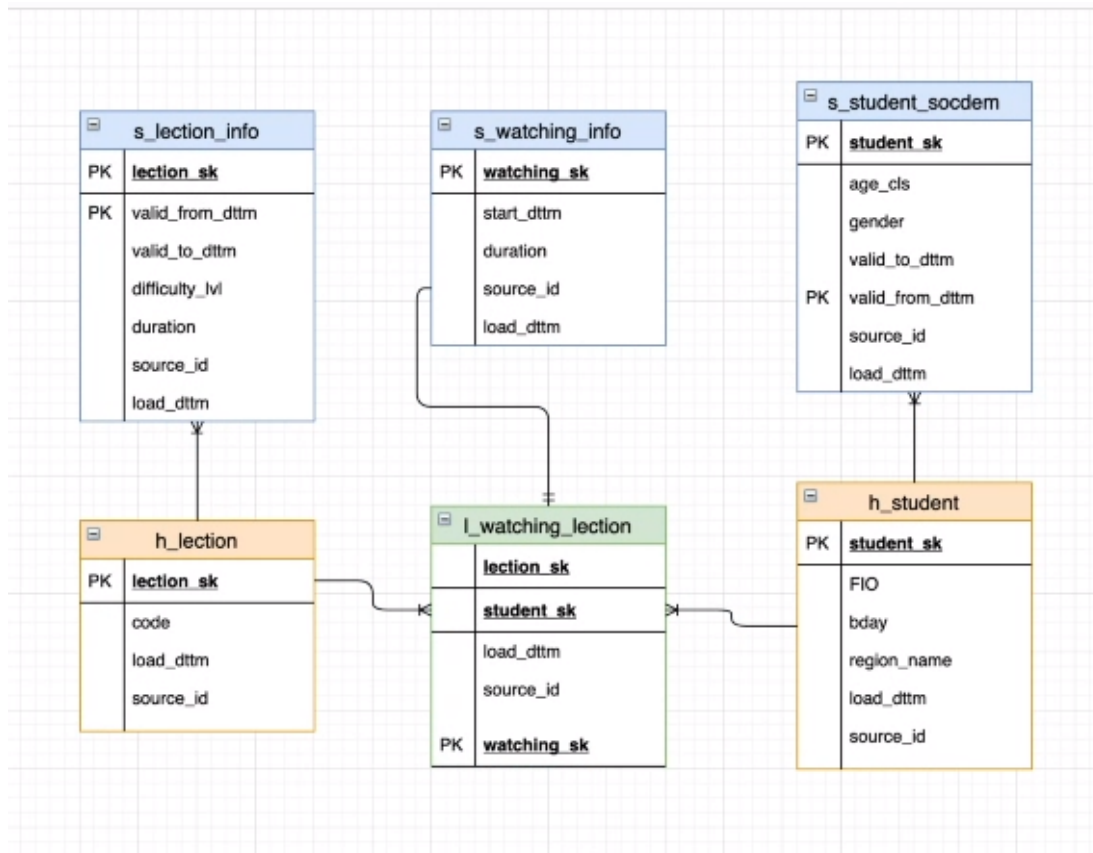
> [Якорь просмотров лекций](#)

> Преобразование Data Vault в Якорную модель

Сегодня мы продолжим моделировать DWH по Data Vault и начнем проектировать по якорному подходу. Начнем работу с финальной схемы, полученной в практике по Data Vault, так как подходы с точки зрения алгоритмов похожи.

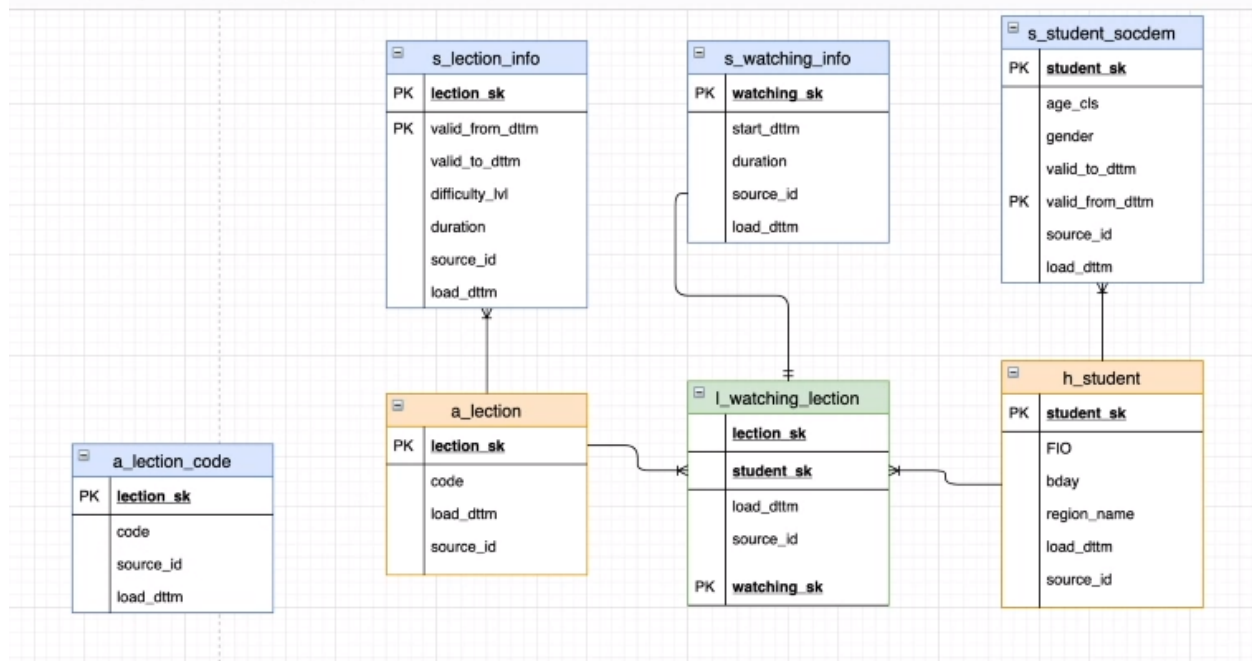
Мы проектировали таблицу факта измерения, мы проектировали наше ядро по третьей нормальной форме (сверху вниз). На этом практическом занятии мы попробуем сконвертировать Data Vault в якорную модель.

Наше ядро не отрицает таблицу факта измерений, оно остается и желательно, чтобы оно не менялось. Мы рассматриваем блок "студенты - лекции".

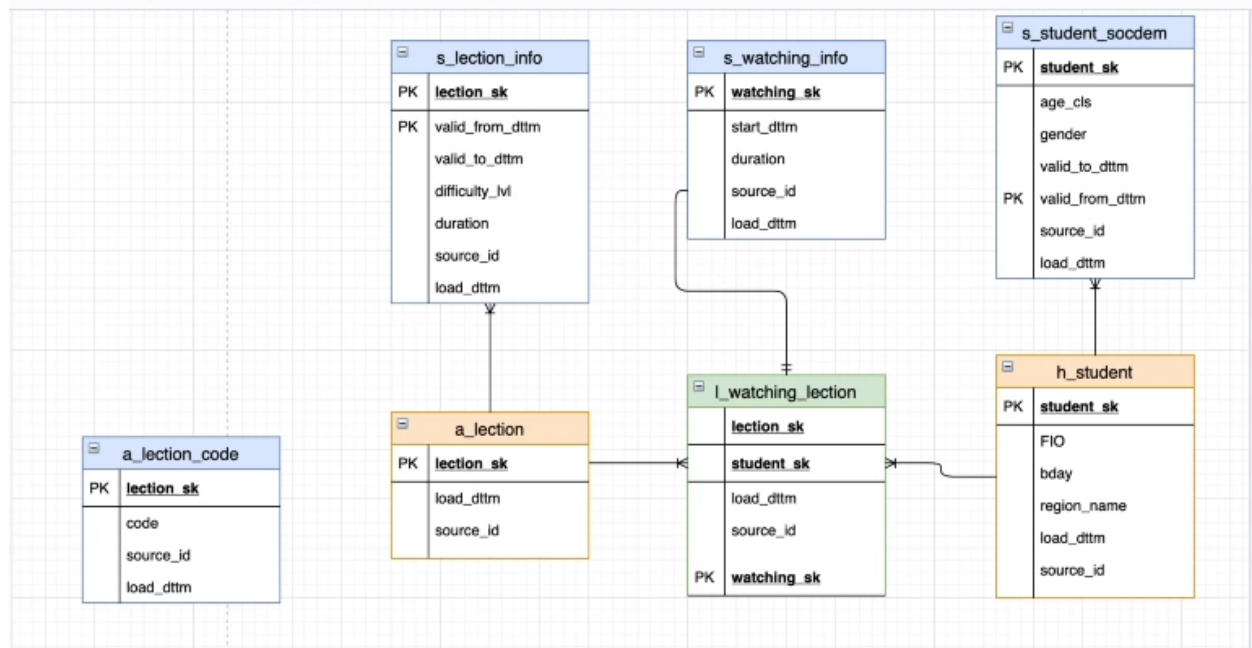


Якорная модель нам говорит, что не может существовать бизнес-ключ в нашем хабе. Переименуем hub **h_lection** в **a_lection**. Из этой таблицы **code** необходимо вынести в отдельную таблицу, для этого дублируем таблицу **s_lection_info**.

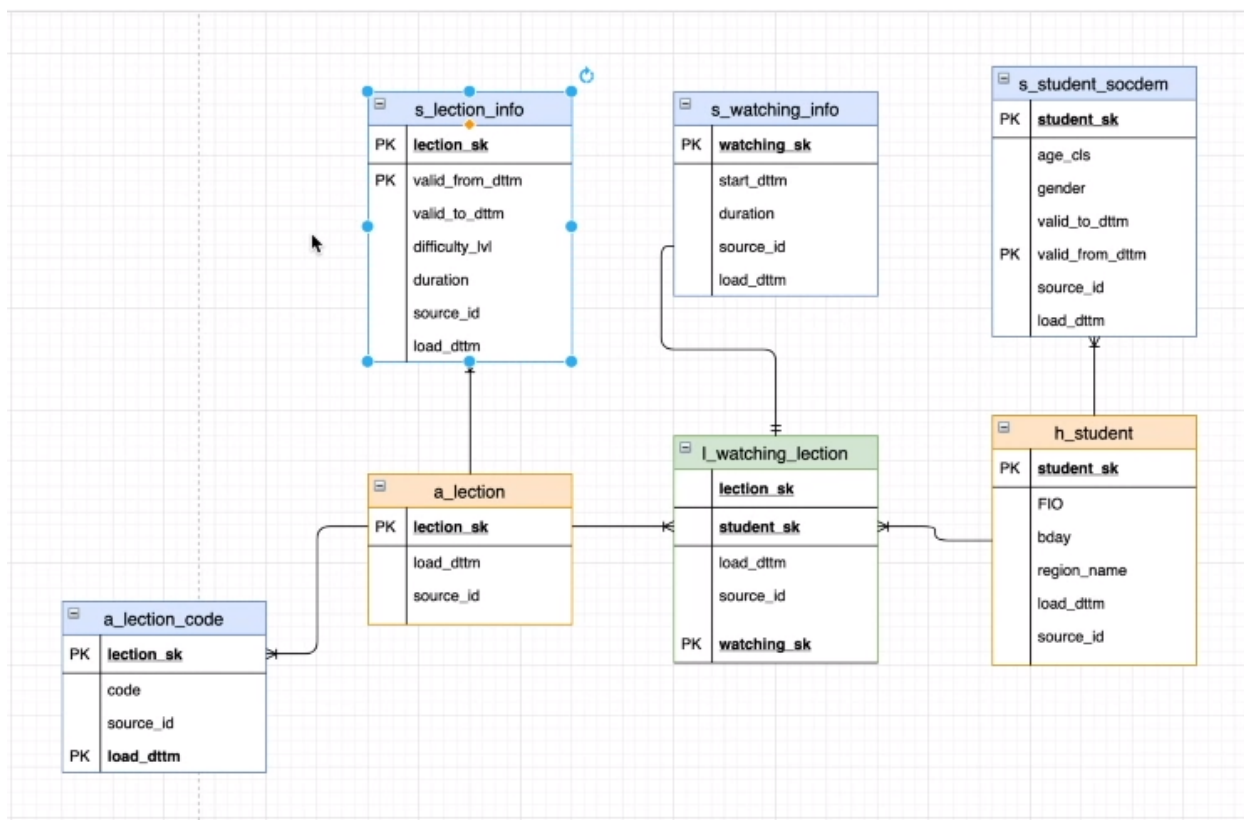
Переименовываем таблицу **s_lection_info** в **a_lection_code**, и оставляем в таблице **code**, **source_id**, **load_dttm**.



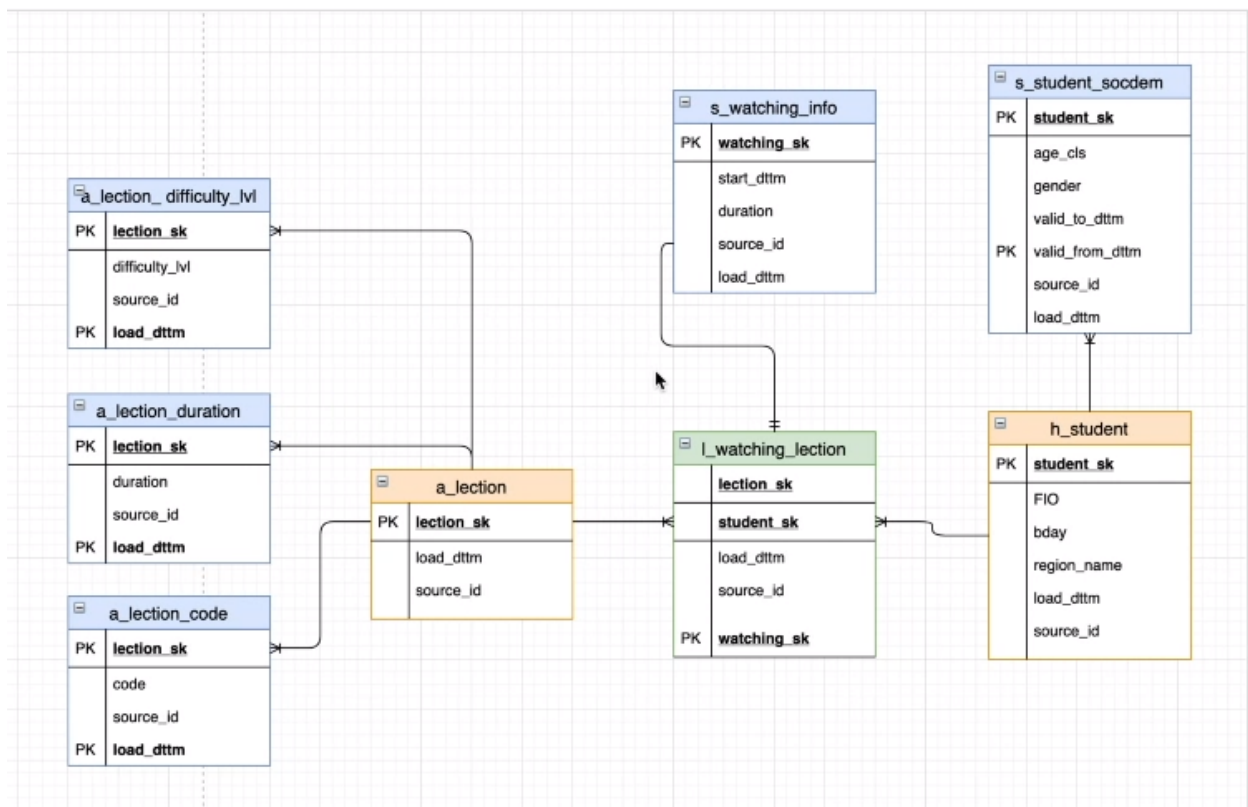
В таблице **a_lesson** удаляем **code**. Мы получили якорь и первый атрибут.



В **a_lesson_code** дата загрузки **load_dttm** можем дублироваться, и фактически она входит в первичный ключ. Поэтому для неё прописываем первичный ключ PK и добавляем связи. У нас есть заготовка, с якорем и первым атрибутом.



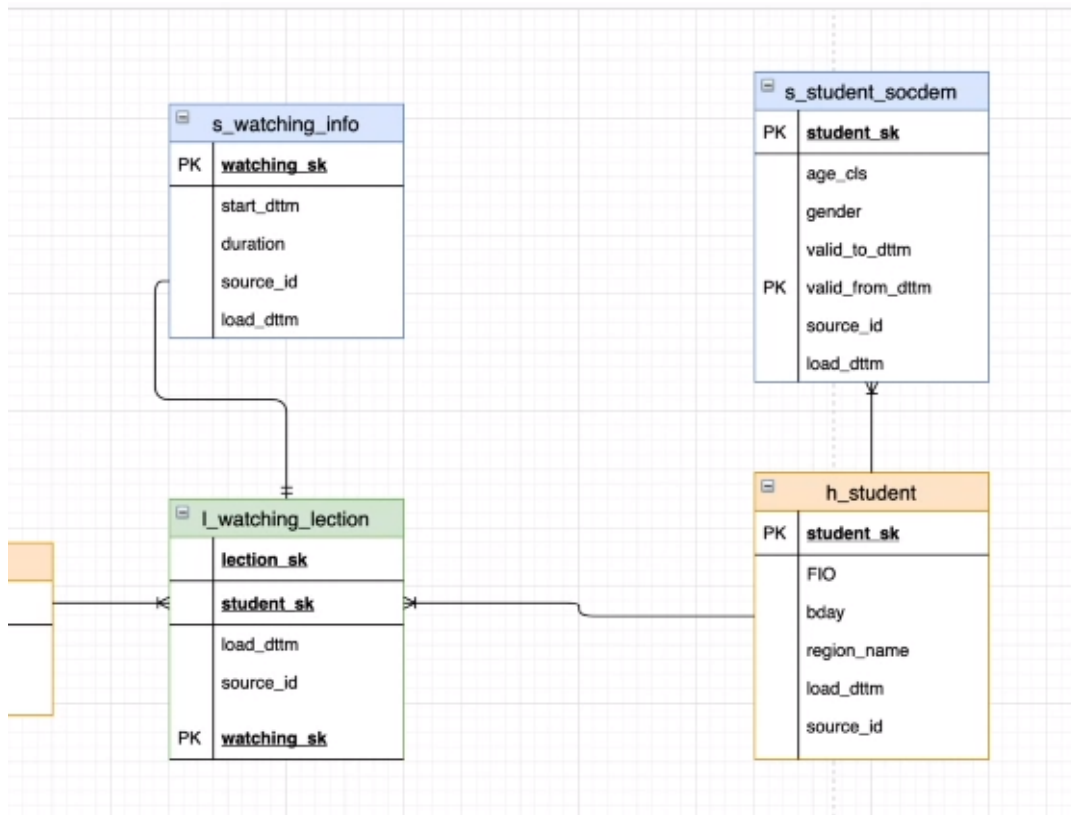
В ядре отдельный атрибут - отдельная таблица. Поэтому всю информацию по лекции нам надо разобрать на отдельные таблицы. У нас есть `difficulty` и `duration`. Удаляем сателлит и добавляем связи к двум новым атрибутам. Пример вы можете увидеть ниже



По аналогии проводим такую же работу по студентам и по социально-демографическим признакам (крайние правые таблицы на слайде сверху).

Переименовываем `h_student` в `a_student`, создавая якорь. Добавляем атрибуты: `a_student_gender`, `a_student_age_cls`, `a_student_name`, `a_student_bday`, `a_student_FIO`.

Добавляем связи и оцениваем работу, которую вы провели.



Необходимо из сущности `a_student` удалить бизнес-ключи.

> Якорь просмотров лекций

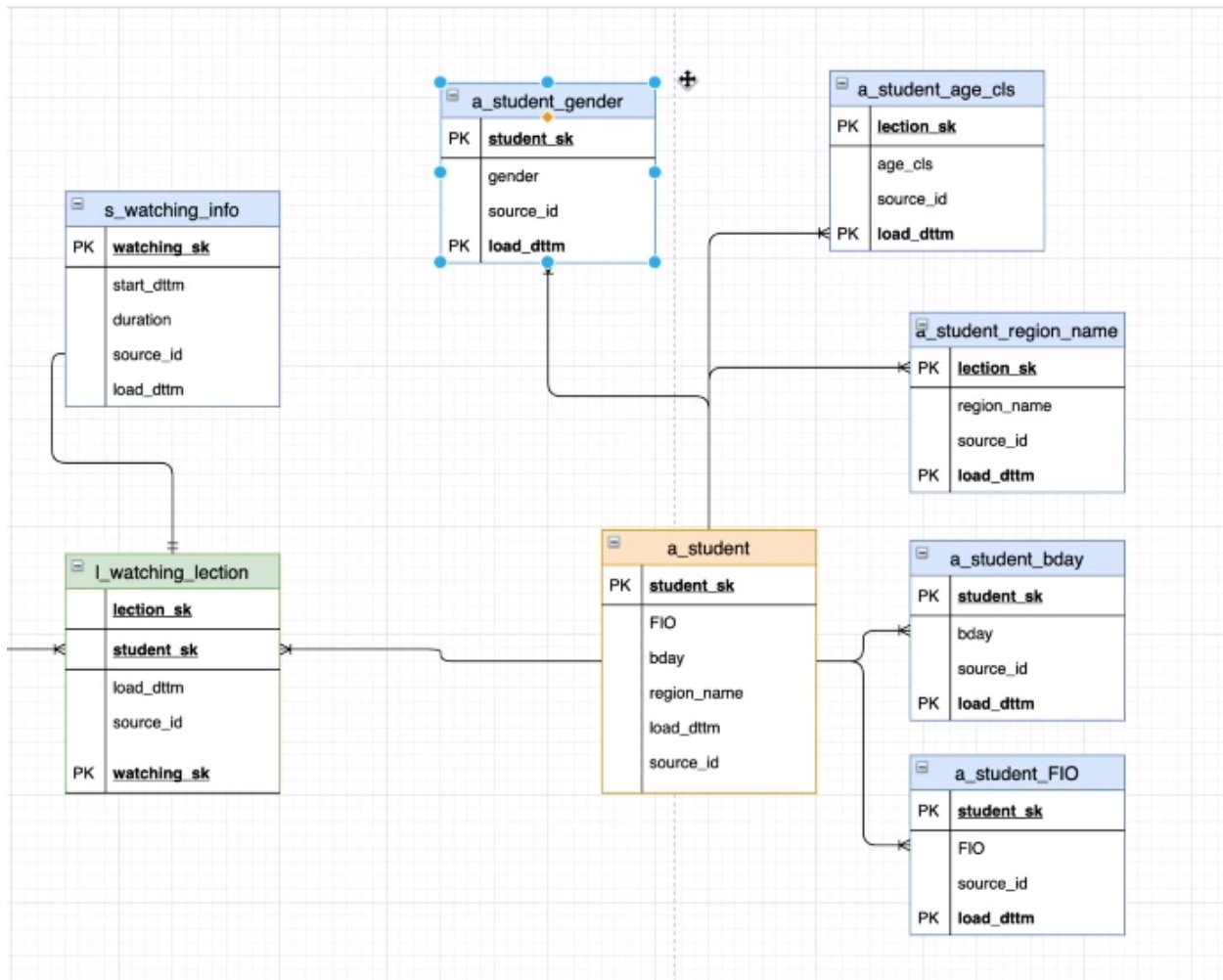
Добавим изменения в просмотр лекций. Якорь не позволяет вешать атрибуты на link. Поэтому нужно добавить две таблички связи и сделать якорь. Назовем этот якорь - `a_watching`.

Рядом расположены наши блоки, один блок сателлита превращается в 4 блока:

`a_watching_duration`, `a_watching_start_dttm`, `l_student_watching` и `l_lection_watching`.

Скорее всего они не будут меняться и это значит, что мы можем не добавлять поля для SCD2. Добавим в каждый блок поля `load_dttm` и `source_id`. Из якоря удалим суррогатные ключи `course_sk` и `student_sk`, а затем перекрасим якорь.

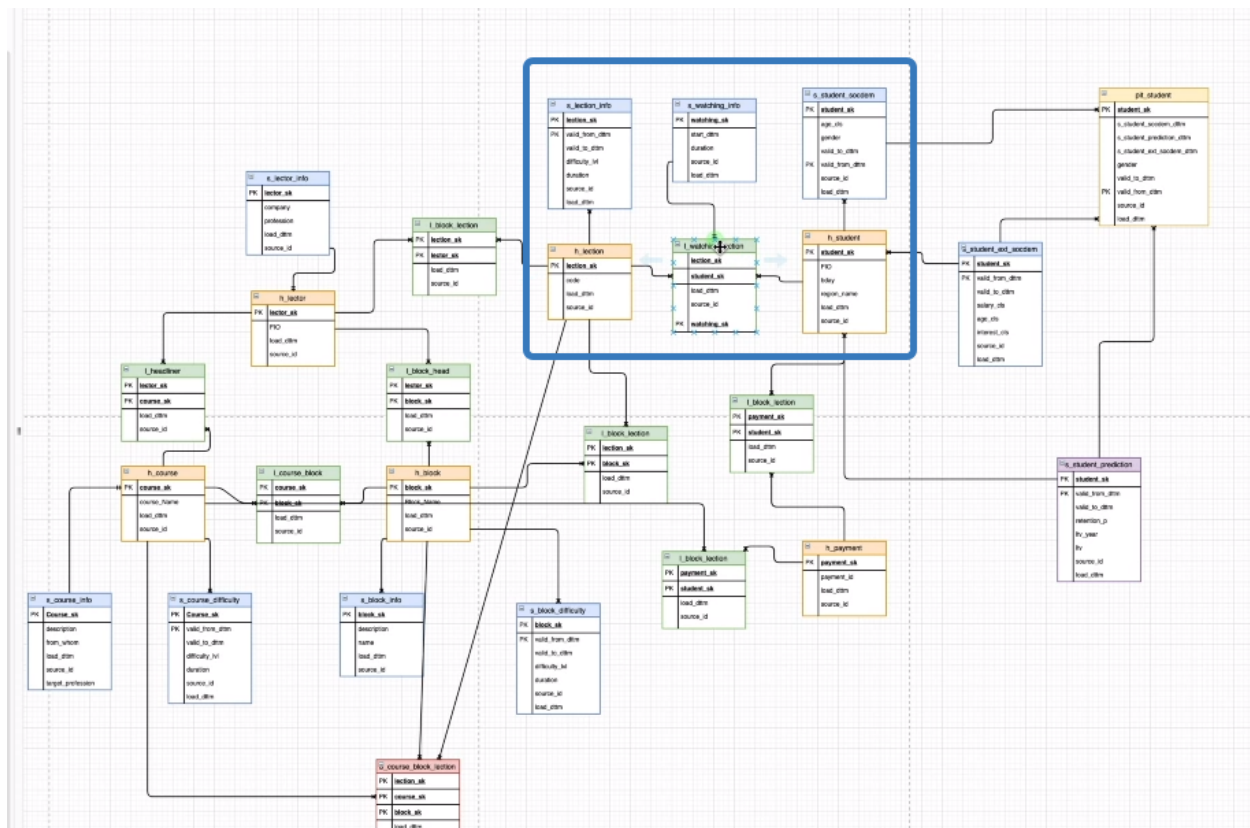
Добавляем связи для полноты картины.



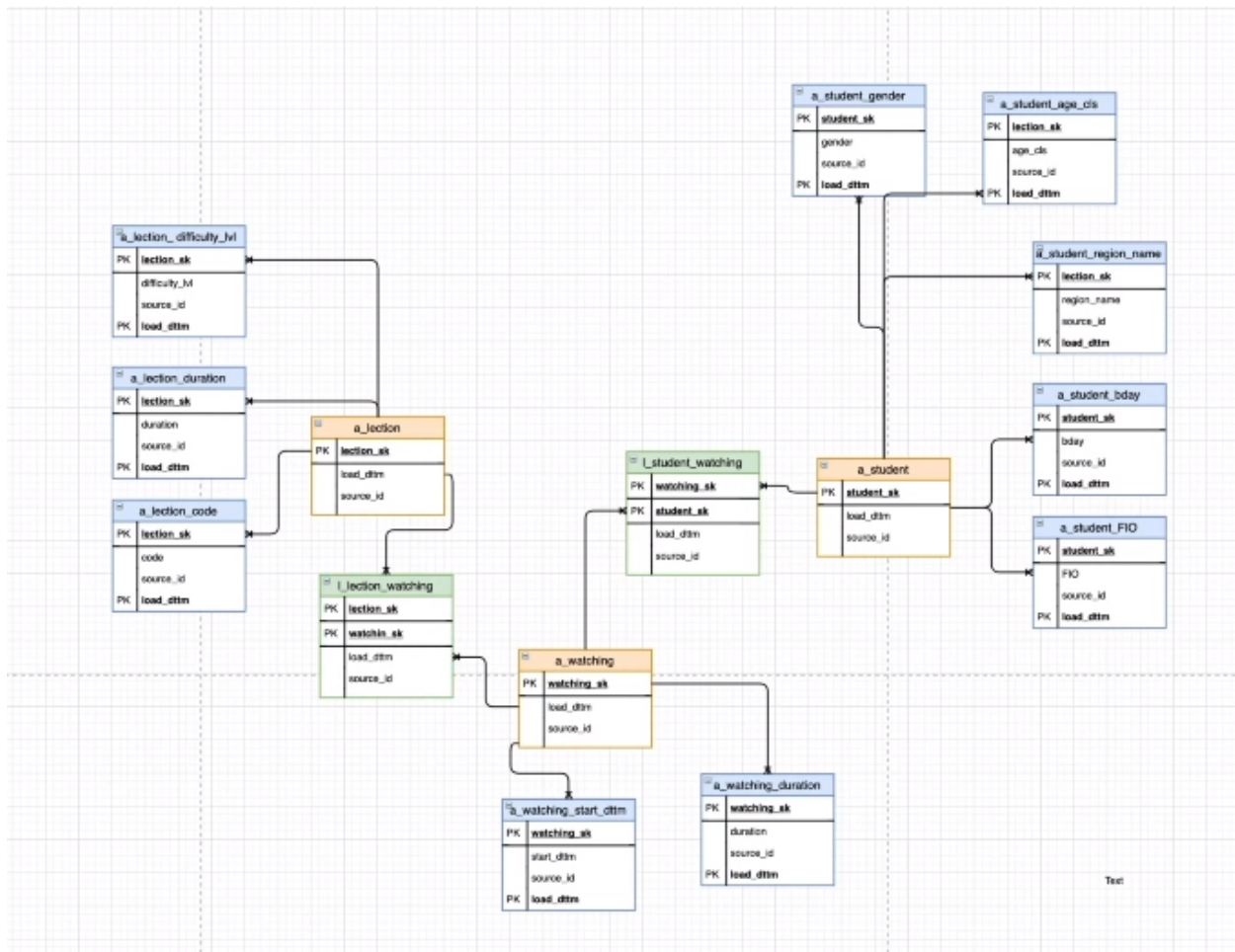
В яоре есть дополнительная табличка, которая называется узел.

Можно добавить например справочник полов или добавить специальное поле тип просмотра. Мы не можем навесить на справочник много полей, поэтому каждый справочник норовит превратиться в якорь. Это приведет к переделыванию, поэтому со справочниками нужно быть аккуратными.

Давайте оценим проделанную работу. Мы взяли шесть таблиц в выделенной области.



По итогам работы у нас появилась вот такая россыпь ключей и атрибутов к ним.



Что вам необходимо сделать, чтобы наработать практические навыки:

- из каждого хаба необходимо доставать бизнес-информацию и вкидывать в атрибуты;
- каждый отдельный спутник разделить на отдельные таблицы;
- все транзакционные линки также нужно будет переделать и обновить.

В чем удобство якоря - это гибкость. Для работы с якорной моделью необходимы средства автоматизации. Существуют инструменты, которые позволяют автоматизировать создание якорной модели, ER-диаграммы и DDL-сценарии.

Самая читабельная схема, с точки зрения сущности и связей, это третья нормальная форма.