Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Базы данных (БД)

ОТЧЕТ

по лабораторной работе № 1

Тема работы: Анализ модели базы данных

Выполнил

студент: гр. 151003 Барановский Р.А.

Проверил: Фадеева Е.Е.

Минск 2023

Инфологическая модель:

* Account – *Учётная запись (описывает учётную запись)*:
  + id *(идентификатор аккаунта)*;
  + balance (*баланс счёта*, MONEY);
  + account\_owner (*владелец счёта*, FK);
  + system\_account (*флаг, указывающий, что эта учётная запись не принадлежит человеку*).
* Status – *Статус (статус аккаунта, например, «Активен», «Заблокирован» и т.д.)*:
  + id *(идентификатор статуса)*;
  + name (*название статуса*).
* Transaction operational – *Текущие транзакции (для транзакций в текущем месяце)*:
  + id (*идентификатор транзакции*);
  + source\_account (*исходный счёт*, FK);
  + destination\_account (*целевой счёт,* FK);
  + date\_and\_time (*дата и время транзакции*);
  + sum (*общая сумма транзакции*).
* Transaction archive – *Архив транзакций (для транзакций до текущего месяца)*:
  + id (*идентификатор транзакции*);
  + source\_account (*исходный счёт*, FK);
  + destination\_account (*целевой счёт,* FK);
  + date\_and\_time (*дата и время транзакции*);
  + sum (*общая сумма транзакции*).
* Account owner – *Владелец счёта (клиент банка)*:
  + id (*идентификатор владельца аккаунта*);
  + name (*имя владельца аккаунта*).
* Site page – *Страница сайта (страница сайта банка)*:
  + id (*идентификатор страницы*);
  + parent\_page (*родительская страница,* FK);
  + name (*название страницы*).
* Office – *Офис* (*офис банка*):
  + id (*идентификатор офиса*);
  + city (*местонахождение офиса*);
  + name (*название офиса*);
  + total\_sells\_sum (*сумма прибыли офиса*, MONEY).

Даталогическая модель:

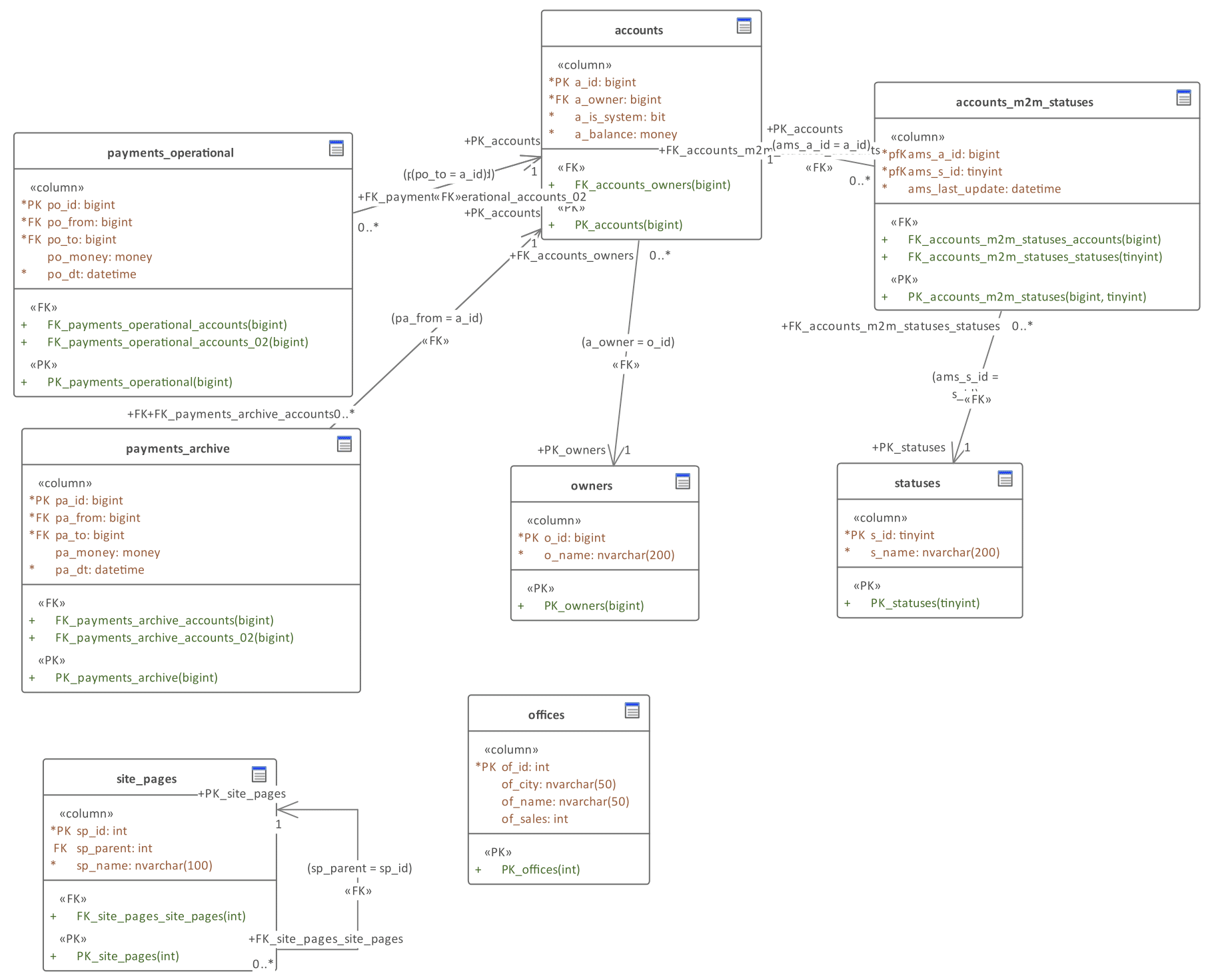


Рисунок – Даталогическая модель

Вариант 1: Работа банка

1. Видите ли вы какие-либо недостатки на инфологической и даталогической моделях? Если да, запишите их.

**Ответ:**

В сущности «Account» следует убрать атрибут «system\_account» и создать отдельную таблицу «System\_account», которая будет иметь связь один-к-одному с таблицей «Account». К данному решению стоит пребегнуть, так как подавляющее большинство аккаунтов не относится к системным, что делает наличие в таблице «Account» атрибута «system\_account» избыточным.

Сущность «site\_pages» следует удалить из модели, так как она не относится к банковской системе. Также в сущности «Account\_owner» не хватает атрибутов, так как банку нужно явно больше информации о клиенте, чем просто его имя. Сущности «Transaction\_operational» и «Transaction\_archive» можно объединить в одну, так как простейшим запросом можно извлечь данные как за последний месяц, так и более старые. Назовем новую сущность «Transaction».

1. Какие вопросы вы хотите задать заказчику для улучшения модели? Запишите эти вопросы и дайте на них ответ от лица заказчика.

**Ответ:**

Вопрос: Для каких целей используется атрибут «system\_account» в сущности «Account», возможно его стоит убрать?

Ответ: Возможны два варианта:

1. Если данный атрибут действительно нужен для неких целей, то прибегнуть к решению из пункта один.
2. Если же нет, то убрать его из списка атрибутов.
3. Какие отношения отсутствуют в базе данных? Добавьте эти отношения.

**Ответ:**

Как упоминалось выше, при необходимости сохранения атрибута system\_account в сущности Account, лучше создать отдельную таблицу со связью One-To-One.

1. Какие атрибуты отсутствуют в существующих отношениях? Добавьте недостающие атрибуты.

**Ответ:**

Помимо нехватающей информации о владельце счета, стоило бы добавить побольше информации об офисе.

1. Все ли объекты в базе данных соответствуют единым соглашениям об именовании объектов? Исправьте те, которые нарушают такие соглашения.

**Ответ:**

Соглашение может меняться от команды к команде, например, в одной действует соглашение именовать таблицы в единственном числе, а в другой в множественном. Также имя первичного ключа может начинаться с имени соответствующей таблицы. В данном случае мне показалось странным, что имя каждого атрибута начинается с сокращенного названия имени таблицы, возможно, это следует убрать.

1. Все ли атрибуты в базе данных имеют оптимальные типы данных? Предложите оптимизацию.

**Ответ:**

В сущности «Transaction» следует поменять тип первичного ключа id с bigint на varchar(n), где будет храниться строковый идентификатор транзакции, так как для крупной платежной системы может не хватить имеющихся целочисленных идентификаторов и может произойти переполнение.

Также для хранения денежных сумм зачастую используют тип Decimal, однако учитывая, что money является decimal(19, 4), можно оставить в изначальном виде, но понимать, что это дает нам меньшую гибкость, так как, например, если потребуется хранить деньги с точностью до 6 знаков после запятой, то необходимо будет изменить тип на decimal(n, 6)

1. Все ли отношения в базе данных имеют оптимальные ключи? Если нет, предложите оптимизацию.

**Ответ:**

Достаточно большое количество PK и FK соответсвенно можно заменить иными типами, например, как описывалось выше, для «Transaction» можно заменить тип PK на varchar(n) для уменьшения вероятности переполнения в случае проектирования действительно большой банковской системы. Наоборот, например, для PK сущности «accounts» с огромной вероятностью было бы достаточно типа int, так как даже в очень крупной системе очень врядли будет зарегестрировано более 2 147 483 647 пользователей. В свою очередь, например, для «offices», оптимальным был бы тип smallint, так как едва ли найдется компания, которой необходимо более чем 32767 офисов.

1. Есть ли в базе данных избыточные или отсутствующие связи? Внесите необходимые исправления.

**Ответ:**

Как было указано выше, лишней является сущность «site\_pages» и ее отношения. Также странным является наличие атрибута «sales» у сущности «offices», т.е. получается, что на каждый офис приходится единственная финансовая отчетность о продажах, лучше создать отдельную сущность «sale» с отношением ManyToOne к «office».

Измененная даталогическая модель:

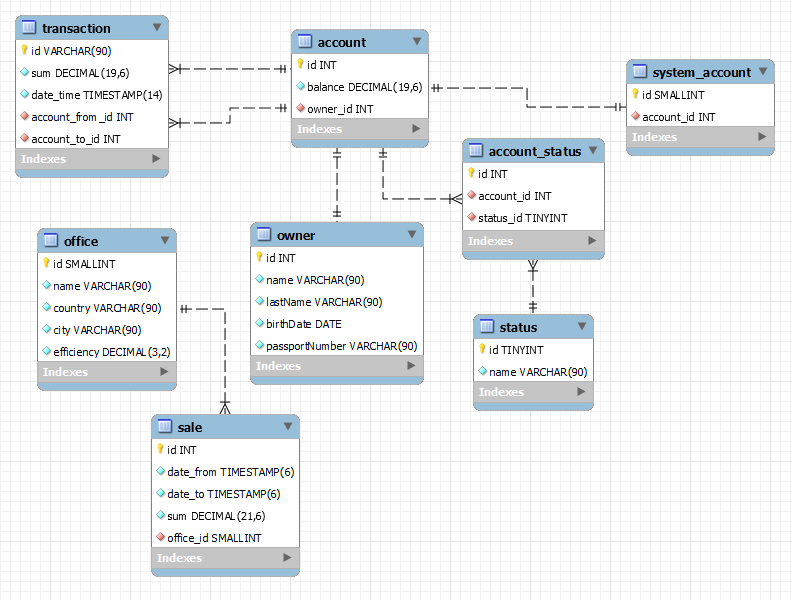


Рисунок 2 – Измененная даталогическая модель