



Федеральное государственное автономное образовательное
учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет ПИ и КТ

Лабораторная работа №3
по дисциплине: «Базы данных»

Выполнил:

Болорболд Аригуун,

группа Р3111

Преподаватель:

Николаев Владимир Вячеславович

Практик:

Чупанов Аликылыч Алибекович

it's *more than a*
UNIVERSITY

Санкт-Петербург

2023

1. Задание:

Для отношений, полученных при построении предметной области из лабораторной работы №1, выполните следующие действия:

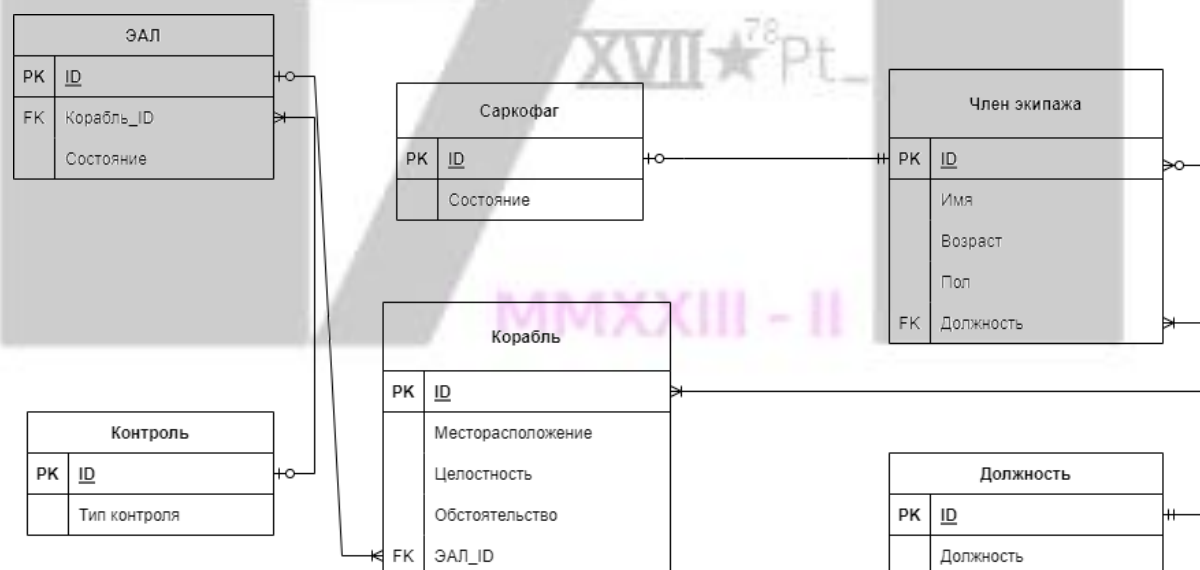
- опишите функциональные зависимости для отношений полученной схемы (минимальное множество);
- приведите отношения в 3NF (как минимум). Постройте схему на основе NF (как минимум). Постройте схему на основе полученных отношений;
- опишите изменения в функциональных зависимостях, произошедшие после преобразования в 3NF (как минимум). Постройте схему на основе NF;
- преобразуйте отношения в BCNF. Докажите, что полученные отношения представлены в BCNF;

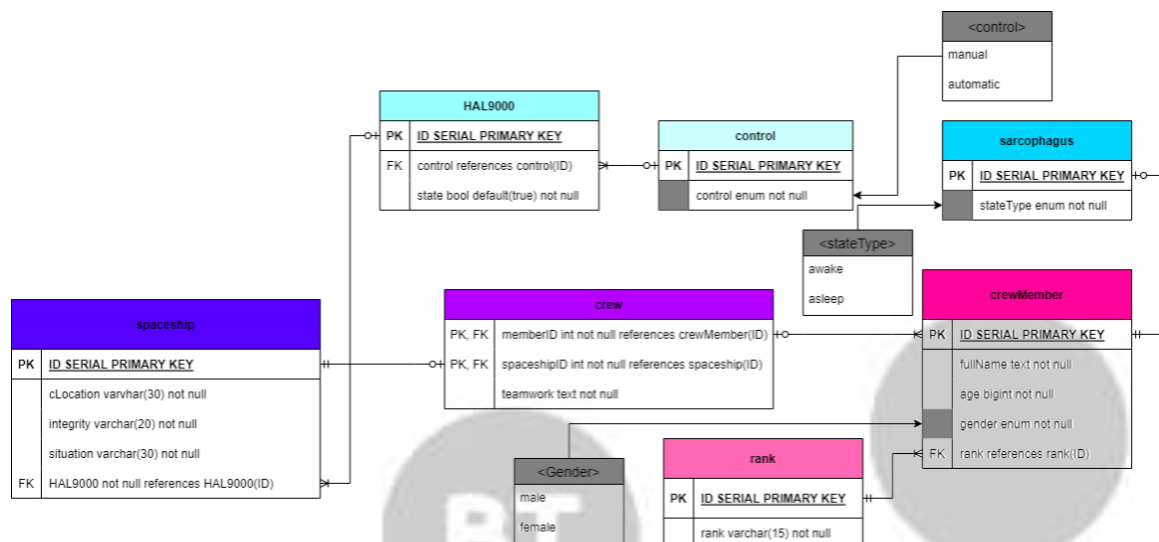
Если ваша схема находится уже в BCNF, докажите это.

Какие денормализации будут полезны для вашей схемы? Приведите подробное описание;

Придумайте функцию, связанную с вашей предметной областью, согласуйте ее с преподавателем и реализуйте на языке PL/pgSQL.

2. Выполнение:





Типы сущностей:

Стержневые:

Член экипажа — имя, возраст, должность, пол, пробуждение;

Корабль — месторасположение, целостность, обстоятельство;

Саркофаг — состояние;

ЭАЛ — состояние;

Ассоциативные:

Экипаж — командная работа;

Характеристические:

Контроль — тип контроля, описание.

Должность — тип должности.

Минимальное количество функциональных зависимостей:

crewMember:

id → fullName

id → age

id → gender

id → rank

spaceship:

id → cLocation

id → integrity

id → situation

id → HAL9000

sarcophagus:

id → stateType

HAL9000:

id → state

crew:

memberID, spaceshipID → teamwork

control:

id → control

rank:

id → rank

Первая нормальная форма:

Для каждой ячейки таблицы всегда одно значение — условие нормализации выполняется.

Вторая нормальная форма:

Все сущности, кроме **crew**, имеют один первичный ключ, хотя там присутствует составной ключ. Значит — условие нормализации выполняется.

Третья нормальная форма:

Между таблицами отсутствуют любые транзитивные зависимости. Значит — условие нормализации выполняется.

Нормальная форма Бойса-Кодда:

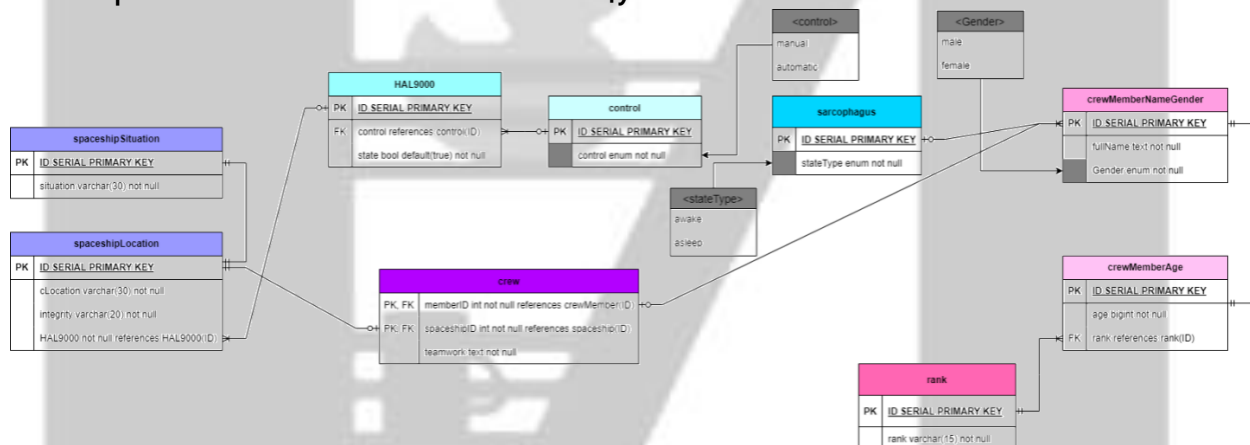
Все детерминанты (атрибут, который можно использовать для определения значений, назначенных другим атрибутам в той же строке) являются первичными и потенциальными ключами. Значит – условие нормализации выполняется.

Четвёртая нормальная форма:

В таблицах все нетривиальные многозначные зависимости являются функциональными. Есть независимые подмножества, но они не зависят от ключевого подмножества. Значит – условие нормализации выполняется.

Пятая нормальная форма:

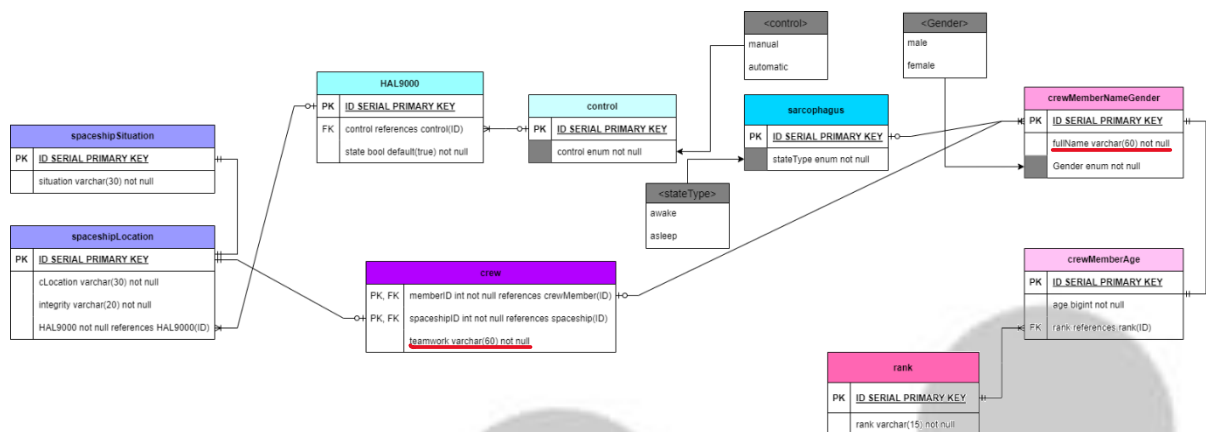
5НФ – нормальная форма, когда каждая нетривиальная зависимость соединения в ней определяется потенциальным ключом (ключами) этого отношения. Для удовлетворения 5НФ мне пришлось обособить таблицу на несколько:



Только после такой операции данная область представления находится в 5НФ. Значит – условие нормализации выполняется.

Доменно-ключевая нормальная форма:

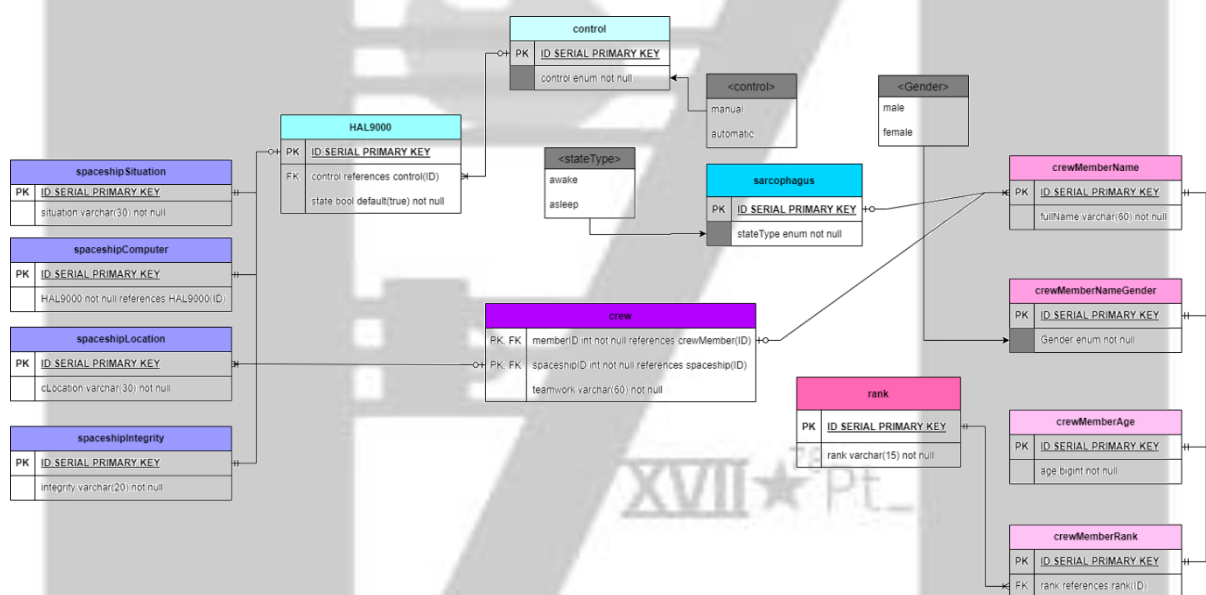
Для удовлетворения ДКНФ необходимо, что все столбцы (в этом случае домены) имели наложенные ограничения (по типу данных, по длине данных, по значению данных):



Для их реализации ещё нужно использовать триггеры. Только после такого преобразования данная область представления находится в 5НФ. Значит — условие нормализации выполняется.

Шестая нормальная форма:

Для удовлетворения 6НФ требуется провести декомпозицию **до предела**:



Не произошла потеря данных. Значит — условие нормализации выполняется. Но так как эта база данных не является хронологической, приведение к 6НФ имеет даже отрицательные последовательности, так как оно приведёт к значительному снижению производительности. Кроме этого, такая нормализация сделает работу с базой данных очень сложной за счёт многократного увеличения количества таблиц.

Денормализация:

- 1) Можно внести control и rank в соответствующие зависящие таблицы. Значит, эти таблицы уже не нужны и благодаря этому уменьшается количество связей, и значит, количество запросов.
- 2) Можно внести HAL9000 в таблицу spaceship и sarcophagus в crewMember, то есть удаляются внесённые таблицы.

Триггер и функция, связанная с предметной областью:

```
-- Эта функция убивает всех членов экипажа при взрыве корабля.
CREATE OR REPLACE FUNCTION reflect_ship_condition()
  RETURNS TRIGGER LANGUAGE plpgsql AS $FUNC$
BEGIN
  DELETE FROM crewmember WHERE ID IN(
    SELECT ID FROM crewmember WHERE ID IN(
      SELECT ID FROM CREW WHERE spaceshipID IN(
        SELECT ID FROM spaceship WHERE spaceship.integrity ~*
        'взорван|exploded')));
  DELETE FROM spaceship WHERE spaceship.integrity ~* 'взорван|exploded';
  RETURN NEW;
END;
$FUNC$;
-- Этот триггер активируется при взрыве космического корабля.
CREATE OR REPLACE TRIGGER EXPLODED
  AFTER UPDATE ON spaceship
  FOR EACH ROW EXECUTE FUNCTION reflect_ship_condition();
```

Результат:

До:

```
studs=> select * from crewmember
studs-> ;
 id | fullname | age
----+-----+----
  1 | Боумен   | 41
  2 | Уайтхед  | 39
  3 | Камински | 35
  4 | Хантер   | 36
  5 | Аригуун  | 18
  6 | Дима     | 19
  7 | Камиль   | 19
  8 | Вика     | 18
(8 строк)
```

```
studs=> update spaceship
studs-> set integrity = 'взорван'
studs-> where id = 1;
```

После:

```
studs=> SELECT * FROM crewmember;  
id | fullname | age | gender  
----+-----+----+-----  
5 | Аригуун | 18 | male  
6 | Дима    | 19 | male  
7 | Камиль  | 19 | male  
8 | Вика    | 18 | female  
(4 строки)
```

Вывод: нормализация играет большую роль в теории реляционных баз данных, так как она способствует преследованию какого-нибудь стандарта. Без таких стандартов не было бы понятия о том, как должны быть устроены базы данные с точки зрения оптимизации и скорости выполнения запросов. Думаю, мне пора оптимизировать и запросы.

XVII★⁷⁸Pt

MMXXIII - II