

## Теоритический минимум. Базы данных

1. *Тип данных* – определяется аналогично типу данных в любом ЯП.
2. *Домен* – понятие, определяемое путем задания некоторого базового типа и произвольного логического выражения. (Допустимое потенциальное ограниченное подмножество данного типа)
3. *Заголовок отношения* – конечно множество пар вида  $\langle A, T \rangle$ , где  $A$  – имя атрибута,  $T$  – либо тип, либо домен.
4. *Кортеж* – множество упорядоченных триплетов вида  $\langle A, T, v \rangle$ , где  $v$  – допустимое значение домена  $T$ .
5. *Тело отношения* – произвольное множество кортежей.
6. *Значение отношения* – пара множеств  $Hr, Vr$ .
7. *Переменная отношения* – именованный контейнер, содержащий допустимое значение отношения.
8. *Степень (3-7)* – мощность заголовка отношения.
9. *Схема реляционной БД* – набор пар, включающий имена и заголовки всех переменных отношения, которые определены в БД.
10. *Первичный ключ переменной отношения* – минимальное подмножество множества атрибутов заголовка данного отношения, составное значение которых уникально определяет кортеж отношения.
11. *Фундаментальные свойства отношений*:
  - 1) Отсутствие кортежей-дубликатов в теле отношения
  - 2) Наличие у каждого значения отношения первичного ключа
  - 3) Отсутствие упорядоченности кортежей
  - 4) Отсутствие упорядоченности атрибутов
  - 5) Значения всех атрибутов являются атомарными (1NF)
12. *Возможный ключ* – минимальный набор атрибутов, обладающий свойством уникальности, но не являющийся первичный ключом.
13. *Модель данных* – модель, описывающая некий набор родовых понятий и признаков, которыми должны обладать СУБД и БД, основанные на этой модели.
14. *Реляционная модель*:
  - 1) Структурная часть
  - 2) Манипуляционная часть
  - 3) Целостная часть
15. *Целостность сущности* – у любой переменной отношения должен существовать первичный ключ, и никакое значения первичного ключа в кортежах переменной не должно содержать неопределенного значения.
16. *Целостность по ссылкам* – для каждого значения внешнего ключа ссылающейся переменной либо должен найтись кортеж с таким же значением первичного ключа, либо значение первичного ключа должно быть неопределенным.
17. *Операции алгебры Кодда*:
  - 1) Объединение отношений (UNION)
  - 2) Пересечение отношений (INTERSECT)
  - 3) Взятие разности (MINUS)
  - 4) Взятие расширенного декартова произведения (TIMES)
  - 5) Соединение (JOIN)
  - 6) Ограничение (WHERE)
  - 7) Проекция (PROJECT)
  - 8) Деление (DIVIDE BY)
  - 9) Переименование (RENAME)
  - 10) Присваивание (:=)
18. *Совместимость отношений по объединению* – два отношения совместимы по объединению в том и только в том случае, когда их заголовки совпадают

19. Совместимость по взятию расширенного декартова произведения – два отношения совместимы по взятию расширенного декартова произведения в том и только том случае, когда пересечение множеств имен их атрибутов пусто
20. Базис алгебры  $A$  -  $\langle \text{NOT} \rangle$  ,  $\langle \text{AND} \rangle$  ,  $\langle \text{OR} \rangle$  , дополнение базиса -  $\langle \text{RENAME} \rangle$  ,  $\langle \text{REMOVE} \rangle$

21. *Реляционное дополнение* – в тело результата входят все кортежи, соответствующие заголовку и не входящие в тело отношения:
- 1)  $H_s = H_r$  (заголовок результата совпадает с заголовком операнда);
  - 2)  $B_s = \{t_s : \text{exists } t_r (t_r \text{ Br and } t_s = t_r)\}$
22. *Удаление атрибута* - заголовок результата получается из заголовка операнда изъятием атрибута, в тело результата входят все кортежи операнда, из которых удалено значение атрибута.
23. *Реляционная конъюнкция (<AND>)*
- 1)  $H_s = H_{r1} \text{ union } H_{r2}$
  - 2) Тело принимает три разных формы в зависимости от значений заголовков:
    - I. *Схемы отношений имеют непустое пересечение* – операция работает как естественное соединение
    - II. *Пересечение схем отношений пусто* – операция работает как расширенное декартово произведение
    - III. *Схемы отношений совпадают* – операция работает как пересечение
24. *Реляционная дизъюнкция (<OR>)*
- 1)  $H_s = H_{r1} \text{ union } H_{r2}$
  - 2) Тело принимает три разных формы в зависимости от значений заголовков:
    - I. *Пересечение схем отношений пусто* – тело результата содержит все кортежи, которые являются объединением кортежей  $tr_1$  и  $tr_2$ , соответствующих заголовкам отношений-операндов, и хотя бы один из этих кортежей принадлежит телу одного из операндов
    - II. *Схемы отношений имеют непустое пересечение* - тело результата содержит все кортежи, которые являются объединением кортежей  $tr_1$  и  $tr_2$ , соответствующих заголовкам отношений-операндов, если хотя бы один из этих кортежей принадлежит телу одного из операндов, и значения общих атрибутов совпадают
    - III. *Схемы отношений совпадают* – тело результата является объединением тел операндов
25. Алгебра  $A$  является полной:
- 1) PROJECT ~ <REMOVE>
  - 2) UNION ~ <OR>
  - 3) TIMES, INTERSECT, JOIN ~ <AND>
  - 4)  $R_1 \text{ MINUS } R_2 = R_1 \text{ <AND> <NOT> } R_2$
  - 5) WHERE:  $R_1 \text{ <AND> } R_2$ , где  $R_2$  – тело, содержащее условие
  - 6)  $R_1\{A,B\}, R_2\{B\}, R_1 \text{ DIVIDE BY } R_2 = (R_1 \text{ PROJECT } A) \text{ MINUS } (((R_2 \text{ TIMES } (R_1 \text{ PROJECT } A) \text{ MINUS } R_1) \text{ PROJECT } A)$
26. *Функциональная зависимость*. В значении переменной отношения  $r$  атрибут  $Y$  функционально зависит от атрибута  $X$  в том и только в том случае, если каждому значению  $X$  соответствует в точности одно значение  $Y$ . Здесь  $X$  является детерминантом  $Y$ , а  $Y$  является зависимым от  $X$ .  $r.X \rightarrow r.Y$
27. *Тривиальная функциональная зависимость*. FD  $A \rightarrow B$  называется тривиальной, если  $B$  является подмножеством  $A$ .
28. *Замыкание множества FD*. Замыканием множества FD  $S$  является множество FD  $S^+$ , включающее все FD, логически выводимые из FD множества  $S$ .
29. *Транзитивная функциональная зависимость*. FD  $A \rightarrow C$  называется транзитивной, если существует такой атрибут  $B$ , что имеются функциональные зависимости  $A \rightarrow B$  и  $B \rightarrow C$  и отсутствует зависимость  $C \rightarrow A$ .
30. *Аксиомы Армстронга*.
- 1) *Рефлексивность*. Если  $B$  – подмножество  $A$ , то  $A \rightarrow B$
  - 2) *Пополнение*. Если  $A \rightarrow B$ , то  $AC \rightarrow BC$
  - 3) *Транзитивность*. Если  $A \rightarrow B$  и  $B \rightarrow C$ , то  $A \rightarrow C$
31. *Расширения аксиом Армстронга*
- 1) *Самодетерминированность*.  $A \rightarrow A$
  - 2) *Декомпозиция*. Если  $A \rightarrow BC$ , то  $A \rightarrow B$  и  $A \rightarrow C$
  - 3) *Объединение*. Если  $A \rightarrow B$  и  $A \rightarrow C$ , то  $A \rightarrow BC$
  - 4) *Композиция*. Если  $A \rightarrow B$  и  $C \rightarrow D$ , то  $AC \rightarrow BD$
  - 5) *Накопление*. Если  $A \rightarrow BC$  и  $B \rightarrow D$ , то  $A \rightarrow BCD$

32. Замыкание множества атрибутов. Пусть заданы отношение  $r$ , множество  $Z$  атрибутов этого отношения и некоторое множество  $FD\ S$ , выполняемых для  $r$ . Тогда замыканием  $Z$  над  $S$  называется наибольшее множество  $Z^+$  таких атрибутов  $Y$  отношения  $r$ , что  $FD\ Z \rightarrow Y$  входит в  $S^+$

```
K := 0; Z[0] := Z;
DO
  K := K+1;
  Z[K] := Z[K-1];
  FOR EACH FD A→B IN S DO
    IF A ⊆ Z[K] THEN Z[K] := (Z[K] UNION B) END DO;
  UNTIL Z[K] = Z[K-1];
Z* := Z[K];
```

33. Алгоритм вычисления  $Z^+$ .

34. Суперключ отношения  $r$  – любое подмножество  $K$  заголовка  $r$ , включающее, по меньшей мере, хотя бы один возможный ключ  $r$ .
35. Покрытие множества  $FD$ . Множество  $S_2$  называется покрытием множества  $S_1$ , если любая  $FD$ , выводимая из  $S_1$ , выводится так же и из  $S_2$ .
36. Эквивалентные множества – множество, каждое из которых является покрытием другого.
37. Минимальное множество  $FD$ . Множество  $FD\ S$  называется минимальным в том и только в том случае, когда оно удовлетворяет следующим свойствам:
- 1) Правая часть любой  $FD$  из  $S$  является множеством из одного атрибута.
  - 2) Детерминант каждой  $FD$  из  $S$  обладает свойством минимальности: удаление любого атрибута из детерминанта приводит к изменению замыкания.
  - 3) Удаление любой зависимости из  $S$  приводит к изменению замыкания.
38. Для любого множества  $FD\ S$  существует эквивалентное ему минимальное множество.
39. Минимальное покрытие множества  $FD\ S$  – любое минимальное множество  $FD\ S_1$ , эквивалентное  $S$ .
40. Декомпозиция без потерь – декомпозиция отношения, которая обратима.
41. Теорема Хита. Пусть задано отношение  $r\ \{A, B, C\}$  ( $A, B$  и  $C$ , в общем случае, являются составными атрибутами) и выполняется  $FD \rightarrow AB$ . Тогда  $r = (r\ \text{PROJECT}\ \{A, B\})\ \text{NATURAL JOIN}\ (r\ \text{PROJECT}\ \{A, C\})$ .
42. Минимально зависимые атрибуты. Атрибут  $B$  минимально зависит от атрибута  $A$ , если выполняется минимальная слева  $FD\ A \rightarrow B$ .
43. Свойства нормальных форм:
- 1) Каждая следующая  $NF$  устраняет проблемы предыдущей
  - 2) В каждой следующей  $NF$  все свойства предыдущей сохраняются
44. Аномалии обновления – трудности при выполнении операции добавления, удаления и модификации кортежей.
45. Вторая нормальная форма ( $2NF$ ). Переменная отношения находится во второй нормальной форме тогда и только тогда, когда она находится в первой нормальной форме, и каждый неключевой атрибут минимально зависит от первичного ключа.
46. Третья нормальная форма ( $3NF$ ). Переменная отношения находится в третьей нормальной форме тогда и только тогда, когда она находится во второй нормальной форме, и каждый неключевой атрибут нетранзитивно функционально зависит от первичного ключа.
47. Независимые проекции отношений – проекции, которые могут обновляться независимо.
48. Теорема Риссанена. Проекции  $r_1$  и  $r_2$  отношения  $r$  являются независимыми тогда и только тогда, когда:
- 1) Каждая  $FD$  в отношении  $r$  логически следует из  $FD$  в  $r_1$  и  $r_2$ .
  - 2) Общие атрибуты  $r_1$  и  $r_2$  образуют возможный ключ хотя бы для одного из этих отношений.
49. Атомарное отношение – отношение, которое нельзя декомпозировать на независимые проекции.
50. Нормальная форма Бойса-Кодда ( $BCNF$ ). Переменная отношения находится в нормальной форме Бойса-Кодда тогда и только тогда, когда любая выполняемая для этой переменной отношения нетривиальная и минимальная  $FD$  имеет в качестве детерминанта некоторый возможный ключ данного отношения.

51. Многозначная зависимость. В переменной отношения  $r$  с атрибутами  $A, B, C$  (в общем случае, составными) имеется многозначная зависимость  $B$  от  $A$  ( $A \twoheadrightarrow B$ ) в том и только в том случае, когда множество значений атрибута  $B$ , соответствующее паре значений атрибутов  $A$  и  $C$ , зависит от значения  $A$  и не зависит от значения  $C$ .
52. Лемма Фейджина. В отношении  $r\{A, B, C\}$  выполняется MVD  $A \twoheadrightarrow B$  в том и только в том случае, когда выполняется MVD  $A \twoheadrightarrow C$ .

53. Теорема Фейджина. Пусть имеется переменная отношения  $r$  с атрибутами  $A, B, C$  (в общем случае, составными). Отношение  $r$  декомпозируется без потерь на проекции  $\{A, B\}$  и  $\{A, C\}$  тогда и только тогда, когда для него выполняется  $MVD A \twoheadrightarrow B \mid C$ .
54. Четвертая нормальная форма (4NF). Переменная отношения  $r$  находится в четвертой нормальной форме в том и только в том случае, когда она находится в BCNF, и все MVD  $r$  являются FD с детерминантами – возможными ключами отношения  $r$ .
55. Тривиальная многозначная зависимость. В переменной отношения  $r$  с атрибутами  $A$  и  $B$  (в общем случае, составными)  $MVD A \twoheadrightarrow B$  называется тривиальной, если либо  $B$  есть подмножество  $A$ , либо  $A \cup B = r$ .
56. Зависимость проекции \соединения. Пусть задана переменная отношения  $r$  с подмножествами заголовка  $A, B, \dots, Z$  (составными, перекрывающимися). В переменной отношения  $r$  выполняется зависимость проекции \соединения  $*(A, B, \dots, Z)$  тогда и только тогда, когда любое допустимое значение  $r$  можно получить путем естественного соединения проекций этого значения на атрибуты  $A, B, \dots, Z$ .
57. PJD, подразумеваемая возможными ключами. В переменной отношения  $r$  PJD  $*(A, B, \dots, Z)$  называется подразумеваемой возможными ключами в том и только в том случае, когда каждый составной атрибут  $A, B, \dots, Z$  является суперключом  $r$ .
58. Тривиальная PJD. В переменной отношения зависимость проекции \соединения называется тривиальной, если хотя бы один из составных атрибутов  $A, B, \dots, Z$  совпадает с заголовком отношения.
59. Пятая нормальная форма (5NF, PJ\NF). Переменная отношения  $r$  находится в пятой нормальной форме в том и только в том случае, когда каждая нетривиальная PJD в  $r$  подразумевается возможными ключами  $r$ .
60. Ограниченность реляционной модели:
- 1) Модель не обеспечивает достаточных средств для представления смысла данных
  - 2) Во многих прикладных областях трудно моделировать предметную область на основе плоских таблиц
  - 3) Реляционная модель не представляет какие-либо формализованные средства для представления зависимостей
  - 4) Реляционная модель данных не предлагает какого-либо механизма для разделения сущностей и связей.
61. Сущность – это реальный или представляемый объект, информация о котором должна сохраняться и быть доступной.
62. Связь – это графически изображаемая ассоциация, устанавливаемая между двумя типами сущностей.
63. В месте «стыковки» связи с сущностью используются:
- 1) трехточечный вход в прямоугольник сущности, если для этой сущности в связи могут (или должны) использоваться много экземпляров сущности
  - 2) одноточечный вход, если в связи может (или должен) участвовать только один экземпляр сущности.
64. Обязательный конец связи изображается сплошной линией, а необязательный – прерывистой линией.
65. Уникальным идентификатором сущности может быть атрибут, комбинация атрибутов, связь, комбинация связей или комбинация связей и атрибутов, уникально отличающая любой экземпляр сущности от других экземпляров сущности того же типа.
66. Первая нормальная форма ER-диаграммы – в первой нормальной форме устраняются атрибуты содержащие множественные значения, т.е. производится выявление неявных сущностей, «замаскированных» под атрибуты
67. Вторая нормальная форма – во второй нормальной форме устраняются атрибуты, зависящие только от части уникального идентификатора. Эта часть уникального идентификатора определяет отдельную сущность.
68. Третья нормальная форма – в третьей нормальной форме устраняются атрибуты, зависящие от атрибутов, не входящий в уникальный идентификатор. Эти атрибуты являются основой отдельной сущности.
69. Типы и подтипы: Если у типа сущности  $A$  имеются подтипы  $B_1, B_2, \dots, B_n$ , то:
- 1) любой экземпляр типа сущности  $B_1, B_2, \dots, B_n$  является экземпляром типа сущности  $A$  (включение)

- 2) если  $a$  является экземпляром типа сущности  $A$ , то  $a$  является экземпляром некоторого подтипа сущности  $B_i$  ( $i = 1, 2, \dots, n$ ) (отсутствие собственных экземпляров у супертипа сущности)
- 3) ни для каких подтипов  $B_i$  и  $B_j$  ( $i, j = 1, 2, \dots, n$ ) не существует экземпляра, типом которого одновременно являются типы сущности  $B_i$  и  $B_j$  (разъединенность подтипов)

70. Базовые приемы перехода в реляционную схему:

- 1) простой тип сущности → таблица
- 2) имя сущности → имя таблицы
- 3) экземпляры типа сущности → строки таблицы
- 4) атрибут → столбец таблицы
- 5) компоненты уникального идентификатора сущности → первичный ключ таблицы
- 6) если в состав уникального идентификатора входят связи, к числу столбцов первичного ключа добавляется копия уникального идентификатора сущности, находящегося на дальнем конце связи. Для именованых этих столбцов используется имена концов связей и/или имена парных типов сущностей.
- 7) Связи «многие к одному» (и «один к одному») → внешние ключи
- 8) Если между двумя сущностями А и В имеется связь «один к одному», то соответствующий внешний ключ может быть объявлен и в таблице А и в таблице В
- 9) Для поддержки связи «многие к многим» между типами сущности А и В создается дополнительная таблица АВ с двумя столбцами, один из которых содержит уникальные идентификаторы экземпляров сущности А, а другой В
- 10) Индексы создаются для первичного ключа (уникальный индекс), внешних ключей и тех атрибутов, на которых предполагается в основном базировать запросы.

71. Способы представления ER-диаграмм в реляционную схему. Если в концептуальной схеме присутствуют подтипы, то возможны для способа из представления в реляционной схеме:

- 1) Собрать все подтипы в одной таблице («+/-» - СТР. 178 )
- 2) Для каждого подтипа образовать отдельную таблицу («+/-» - СТР. 179)

72. Способы представления ER-диаграмм в реляционную схему при наличии взаимно исключающих связей.

- 1) Общее хранение внешних ключей
- 2) Раздельное хранение внешних ключей

73. Диаграмма классов (в терминологии UML) - называется диаграмма, на которой показан набор классов (и некоторых других сущностей не имеющих явного отношения к проектированию БД), а также связей между этими классами

74. Класс – это именованное описание совокупности объектов с общими атрибутами, операциями, связями и семантикой (графически класс изображается в виде прямоугольника)

75. Атрибутом класса – это именованное свойство класса, описывающее множество значений, которые могут принимать экземпляры этого свойства.

76. Операция класса – это именованная услуга, которую можно запросить у любого объекта этого класса

77. Сигнатура операции – это имена и типы всех параметров, а если операция является функцией, то и тип её значения.

78. В диаграмме классов могут участвовать связи трех различных категорий:

- 1) Связи-зависимости – это связь по применению, когда изменение в спецификации одного класса может повлиять на поведение другого класса, использующего первый класс. Зависимость показывается прерывистой линией со стрелкой, направленной к классу, от которого имеется зависимость
- 2) Связи-обобщения - это связь между общей сущностью, называемой суперклассом (или родителем), и более специализируемой разновидностью этой сущности, называемой подклассом (или потомком)
- 3) Связь-ассоциация – это структурная связь, показывающая, что объекты одного класса некоторым образом связаны с объектами другого или того же самого класса. С понятием ассоциации связаны четыре важных дополнительных понятия:
  - I. Имя – это имя характеризующие природу связи. Смысл имени уточняет с помощью черного треугольника, который располагается над линией связи справа или слева от имени ассоциации. Этот треугольник указывает направление чтения имя связи.
  - II. Другим способом именованья является задание роли. Роль задается именем, помещаемым под линией ассоциации ближе к данному классу.



- III. Кратность роли – это характеристика, указывающая, сколько объектов класса с данной ролью может или должно участвовать в каждом экземпляре ассоциации
- IV. Иногда в диаграмме классов требуется отразить тот факт, что ассоциация между двумя классами имеет специальный вид «часть-целое». В этом случае класс «целое» имеет более высокий концептуальный уровень, чем класс «часть». Ассоциация такого рода называется АГРЕГАТНОЙ.

79. В UML допускается два способа определения ограничений:

- 1) На естественном языке
- 2) На языке OCL

80. *Инвариант класса* – это логическое выражение, вычисление которого должно давать true при создании любого объекта данного и сохранять истинное значение в течение всего времени существования этого объекта.

81. *Операция Select*. результатом каждой операции является новое множество, мультимножество, соответственно, из тех элементов входной коллекции, для которых результатом вычисления логического выражения является true

82. *Операция Collect*. результатом является мультимножество для операции collect, определенных над множествами и мультимножествами для операции collect. При этом результирующая коллекция соответствующего типа (коллекция значений или объектов) состоит из результатов применения выражения к каждому элементу входной последовательности.

83. *Основные цели System R*:

- 1) Обеспечение ненавигационного интерфейса пользователя с базой, который обеспечивает независимость данных.
- 2) Обеспечение многообразия использования СУБД: программируемые транзакции, диалоговые транзакции, генерация отчетов.
- 3) Поддержание динамической изменяемости среды БД.
- 4) Обеспечение мультипользовательского режима
- 5) Обеспечение восстановления согласованного состояния БД при любом виде сбоя системы
- 6) Обеспечение механизма авторизации пользователей
- 7) Обеспечение производительности, сравнимой с низкоуровневыми БД

84. *Транзакция* – последовательность элементарных атомарных операций. При этом гарантируется выполнение следующих условий:

- 1) Эта операция успешно выполнится или не выполнится вовсе
- 2) Во время выполнения этой операции не будет выполняться никакая другая операция любой транзакции (строгая очередность)

85. В *System R* организовано (в основном на средствах *SQL*):

- 1) Точки сохранения
- 2) Определение условного воздействия - это каталогизированной операции модификации, для которой задано условие ее автоматического выполнения
- 3) Определение представления - запомненного именованного запроса на выборку данных
- 4) Авторизация доступа (с возможностью выдачи и изъятия у некоторых пользователей всех или нескольких прав)
- 5) Организация каталога БД в виде таблицы, к которой, в свою очередь, применимы SQL-запросы.
- 6) Обеспечение изолированности пользователей
- 7) Обеспечение отката транзакций
- 8) Организация журнала – файла, в котором запоминается информация об изменениях, выполненных транзакциями

86. *Структурная организация System R*:

- 1) Система управления памятью RSS, которая разбивается на 2 компонента:
  - I. Управление памятью
  - II. Управление синхронизацией
- 2) Компилятор запросов SQL

87. *Идентификатор кортежа tid* – пара <номер страницы, индекс описателя кортежа в странице>

88. *Индекс* – дополнительная управляющая структура в *System R*, определенная на одном или нескольких полях отношения, составляющих ключ отношения, и позволяющая производить прямой поиск по ключу кортежей (их tid'ов) и последовательное сканирование отношения по индексу, начиная с указанного ключа, в порядке возрастания или убывания значений ключа.

89. *B-дерево* - это сбалансированное сильно ветвистое дерево во внешней памяти, представляемое как мультисписочная структура страниц внешней памяти, т.е. каждому узлу дерева соответствует блок внешней памяти (страница).

90. Особенности физической организации System R:

- 1) Организация индексов в виде B-деревьев
- 2) Поддержка кластеризации связанных кортежей одного или нескольких отношений
- 3) В ранних версиях поддержание связей - физической ссылки (fid) из одного кортежа на другой (не обязательно одного отношения).
- 4) Наличие во внешней памяти списков - мгновенного снимка некоторой выборки с проекцией кортежей одного отношения, возможно, упорядоченный в соответствии со значениями некоторых полей
- 5) Наличие файлов данных и файлов индексов. В файлах данных могут храниться как кортежи одного отношения, так и различных.

91. Группы операций в интерфейсе RSS:

- 1) операции сканирования отношений и списков;
- 2) операции создания и уничтожения постоянных и временных объектов базы данных;
- 3) операции модификации отношений и списков;
- 4) операция добавления поля к отношению;
- 5) операции управления прохождением транзакции;
- 6) операция явной синхронизации.

92. Сериальный план выполнения набора транзакций – план, в ходе которого результат совместного выполнения транзакций эквивалентен результату некоторого последовательного выполнения этих же транзакций.

93. Сериализация транзакций - это механизм их выполнения по некоторому сериальному плану

94. Виды конфликтов работы транзакций:

- 1) W-W - транзакция 2 пытается изменить объект, измененный не закончившейся транзакцией 1;
- 2) R-W - транзакция 2 пытается изменить объект, прочитанный не закончившейся транзакцией 1;
- 3) W-R - транзакция 2 пытается читать объект, измененный не закончившейся транзакцией 1.

95. Феномены, вызываемые конфликтами параллельной работы транзакций:

- 1) Проблема потери результатов обновления. (Две транзакции по очереди записывают некоторые данные в одну и ту же строку и фиксируют изменения. Транзакция, закончившая работу первой, теряет данные своей работы)
- 2) Проблема незафиксированной зависимости (чтение "грязных" данных, неаккуратное считывание): транзакция 1 меняет данные в строке; транзакция 2 считывает данные; транзакция 1 откатывается => 2 считала данные, которых нет в БД
- 3) Проблема несовместимого анализа:
  - I. Неповторяемое считывание: транзакция 1 читает строку; транзакция 2 изменяет ее значение; транзакция 1 повторно читает строку=> транзакция 1 работает с данными, которые самопроизвольно меняются
  - II. Фиктивные элементы (фантомы): транзакция 1 дважды выполняет выборку строк с одним и тем же условием; между выборками вклинивается транзакция 2, которая добавляет новую строку, удовлетворяющую условию отбора=> транзакция 1 на одной и той же выборке получила разные результаты
  - III. Собственно несовместимый анализ.

96. Основные режимы синхронизационных захватов:

- 1) совместный режим - S (Shared), означающий разделяемый захват объекта и требуемый для выполнения операции чтения объекта;
- 2) монополярный режим - X (exclusive), означающий монополярный захват объекта и требуемый для выполнения операций занесения, удаления и модификации.

97. Этапы выполнения транзакции на основе двухфазового протокола:

- 1) накопление захватов;
- 2) фиксация или откат - освобождение захватов;

98. Гранулированный синхронизационный захват – захват объектов разного уровня.(IS, IX, SIX)

99. *Предикатный захват* – захват условий, которым удовлетворяют нужные объекты. Решает проблему фантомов.

100. *Граф ожидания транзакций* - это ориентированный двудольный граф, в котором существует два типа вершин - вершины, соответствующие транзакциям, и вершины, соответствующие объектам захвата.

101. Методы сериализации транзакций:
- 1) Синхронизационный захват объектов
  - 2) Метод временных меток: если транзакция T1 началась раньше транзакции T2, то система обеспечивает такой режим выполнения, как если бы T1 была целиком выполнена до начала T2.
102. Мягкий сбой – потеря данных, которые к моменту сбоя находились в буферах оперативной памяти.
103. Жесткий сбой – потеря данных на внешнем носителе.
104. Ситуации, требующие восстановления состояния БД:
- 1) Индивидуальный откат транзакции
  - 2) Мягкий сбой
  - 3) Жесткий сбой
105. Варианты ведения журнальной информации:
- 1) Локальный для каждой транзакции и общий
  - 2) Только общий
106. Виды буферов в контексте БД:
- 1) Буфер журнала
  - 2) Буфер страниц ОП
107. Физически согласованное состояние внешней памяти БД – состояние, при котором наборы страниц всех объектов согласованы, т.е. соответствуют состоянию объекта либо до его изменения, либо после.
108. Точки физической согласованности БД – моменты времени, в которые во внешней памяти находятся согласованные результаты операций, завершившихся до этого момента времени, и отсутствуют результаты операций, которые еще не завершились, а буфер журнала вытолкнут во внешнюю память.
109. Способы установки точек физической согласованности:
- 1) Теневой механизм
  - 2) Журнализация постраничного изменения
- 110.