

Порядок построения простого запроса на языке РЯ

Шаг 1. Определить источник данных.

Шаг 2. Определить подмножество кортежей источника, содержащих требуемую информацию.

Шаг 3. Определить подмножество атрибутов источника, значения которых запрашиваются.

Шаг 4. Описать последовательность операций РЯ над отношениями источника, формирующих требуемое отношение.

7) Получить значения номеров поставщиков, поставляющих детали для изделий J1 и J2.

(SPJ WHERE Jnum = 'J1')[Snum] INTERSECT (SPJ WHERE Jnum = 'J2')[Snum];

8) Получить значения номеров поставщиков, поставляющих для изделия J1 красную деталь.

а) Получить множество поставок для J1 -- **SPJ WHERE Jnum = 'J1'**

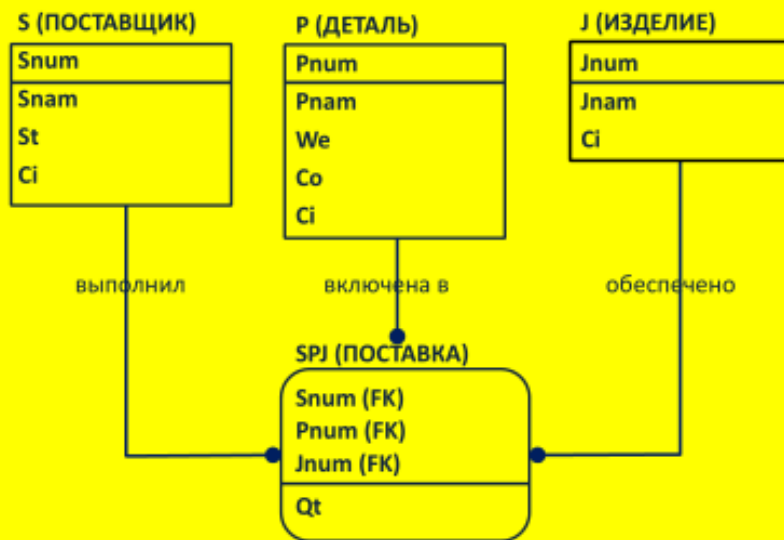
б) Получить множество красных деталей -- **P WHERE Co = 'красный'**

с) Получить соединение этих отношений

(SPJ WHERE Jnum = 'J1') JOIN (P WHERE Co = 'красный')

д) Выполнить проекцию соединения на атрибут Snum

((SPJ WHERE Jnum = 'J1') JOIN (P WHERE Co = 'красный'))[Snum];



9) Получить значения номеров деталей, поставляемых **для каждого** изделия, производимого в Томске.

Источник данных: отношения **SPJ** и **J**

1. Получить множество **PJ** пар номеров деталей и изделий, встречающихся в **SPJ**.

SPJ[Pnum, Jnum]

Обозначим: PJ

2. Получить множество **JT** номеров изделий, производимых в Томске.

(J WHERE Ci = 'Томск')[Jnum] *Обозначим: JT*

3. Сформировать множество номеров деталей по правилу:
если Pnum встречается в **PJ** в паре с каждым значением Jnum из **JT**,
то Pnum принадлежит целевому множеству.

Т.Е. : PJ DIVIDEBY JT

ОКОНЧАТЕЛЬНО: **SPJ[Pnum, Jnum] DIVIDEBY (J WHERE Ci = 'Томск')[Jnum];**

10) Получить значения номеров поставщиков, поставляющих красные детали для изделий, производимых в Томске или Яе.

а) Получить множество значений номеров красных деталей.

б) Получить полную информацию о поставках красных деталей (множество **Red**).

с) Получить множество значений номеров изделий, производимых в Томске или Яе (множество **JTY**).

д) Получить множество значений номеров поставщиков, поставляющих детали из множества **Red** для изделий из множества **JTY**.

```
(  
    (P WHERE Co = 'красный')[Pnum] JOIN SPJ  
        JOIN (J WHERE Ci = 'Томск' OR Ci = 'Яя')[Jnum]  
    ) [Snum];
```

Можно и так:

```
( (P WHERE Co = 'красный') JOIN SPJ  
    JOIN (J WHERE Ci = 'Томск' OR Ci = 'Яя')  
)[Snum];
```

11) Получить значения номеров изделий, снабжаемых хотя бы одним поставщиком, расположенным не в том же самом городе.

```
(  
  ( J RENAME Ci AS JCi) JOIN  
    SPJ JOIN  
  ( S RENAME Ci AS SCi)  
)  
  WHERE NOT (JCi = SCi)  
)[Jnum];
```

Отношение со схемой:
{ Jnum, Jnam, JCi, Snum, Snam, St, SCi, Pnum, Qt }

Другой вариант:

```
(( SPJ[Snum, Jnum] MINUS ( S JOIN J ) [ Snum, Jnum] ) [Jnum];
```

12) Получить значения номеров изделий,
для которых не поставляется ни одной красной детали из Томска.

J[Jnum] **MINUS**

((P **WHERE** Co = 'красный' **AND** Ci='Томск') [Pnum] **JOIN** SPJ) [Jnum];

13) Получить имена поставщиков, поставляющих деталь **P2**.

((SPJ **JOIN** S) **WHERE** Pnum = 'P2') [Snam];

14) Получить имена поставщиков, поставляющих все детали.

((SPJ[Snum, Pnum] **DIVIDE**BY P[Pnum]) **JOIN** S)[Snam];

15) Получить имена поставщиков, поставляющих все поставляемые детали.

((SPJ[Snum, Pnum] **DIVIDE**BY SPJ[Pnum]) **JOIN** S)[Snam];

Реляционное исчисление с переменными-кортежами

Реляционное исчисление

Формальный язык,
предназначенный для точного описания
целевого множества кортежей реляционной БД,
содержащих всю запрашиваемую информацию

Определение РИ см. Основы технологии БД, пп. 6.4.2

Основные понятия

Окно, перемещающееся по строкам таблицы

Переменная-кортеж

Имя, принимающее значения на некотором множестве кортежей.

Область определения

Отношение, на котором переменная-кортеж принимает значения.

Целевая схема

Перечень атрибутов области определения, образующих схему отношения-результата.

Условие (предикат)

Логическое выражение, содержащее ссылки на переменные-кортежи, логические операции и кванторы.

Определяет ограничения на значения переменных-кортежей.

Синтаксис РИ с переменными-кортежами

*область-определения ::= **RANGE OF** переменная **IS** список-элементов-области;*

*выражение ::= (список-целевых-элементов) [**WHERE** ппф] ;*

список-элементов ::= элемент [, список-элементов] ;

элемент-области ::= отношение | выражение ;

целевой-элемент ::= переменная |

*переменная.атрибут [**AS** атрибут] ;*

ппф ::= условие | **NOT** ппф | условие **AND** ппф | условие **OR** ппф |
 IF условие **THEN** ппф |
 EXISTS переменная (ппф) |
 FOR ALL переменная (ппф) | (ппф);

условие ::= (ппф) | сравнение;
сравнение ::= символ θ символ;
символ ::= переменная.атрибут | константа;
 θ ::= < | > | = | <>;
отношение, переменная, атрибут -- идентификаторы

Примеры запросов на языке РИ с переменными-кортежами

1) Получить полные сведения обо всех производимых изделиях.

RANGE OF JX IS J;
(JX);

2) Получить номера и названия изделий, производимых в Томске

RANGE OF JX IS J;
(JX.Jnum, JX.Jnam) WHERE JX.Ci = 'Томск';

Далее считается, что переменные-кортежи **SX, PX, JX, SPJX**
определены на отношениях **S, P, J, SPJ**, соответственно.
Ненужные скобки опускаются.

3) Получить значения номеров поставщиков, выполняющих поставки для изделия J1.

SPJX.Snum WHERE SPJX.Jnum = 'J1';

4) Получить значения номеров поставщиков,
поставляющих деталь P1 для изделия J1.

SPJX.Snum **WHERE** SPJX.Jnum = 'J1' **AND** SPJX.Pnum = 'P1';

5) Получить значения цветов деталей, поставляемых поставщиком S1

PX.Co **WHERE EXISTS** SPJX (PX.Pnum = SPJX.Pnum **AND** SPJX.Snum = 'S1');

6) Получить значения наименований изделий,

для которых выполняет поставки поставщик S1.

JX.Jnam **WHERE EXISTS** SPJX (SPJX.Snum = 'S1' **AND** SPJX.Jnum = JX.Jnum);

7) Получить значения номеров поставщиков,

поставляющих детали для изделий J1 и J2.

SPJX.Snum **WHERE** SPJX.Jnum = 'J1' **AND** ~~SPJX.Jnum = 'J2'~~

EXISTS SPJY (SPJY.Snum = SPJX.Snum **AND** SPJY.Jnum = 'J2');

8) Получить значения номеров поставщиков, поставляющих для изделия J1 красную деталь.

```
SPJX.Snum WHERE SPJX.Jnum = 'J1' AND  
          EXISTS PX ( PX.Co = 'красный' AND PX.Pnum = SPJX.Pnum );
```

9) Получить значения номеров деталей, поставляемых для каждого изделия, производимого в Томске.

```
SPJX.Pnum WHERE FORALL JX  
          ( IF JX.Ci = 'Томск' THEN EXISTS SPJY  
            ( SPJY.Jnum = JX.Jnum AND  
              SPJY.Pnum = SPJX.Pnum  
            )  
          );
```

Другая формулировка запроса 9:

```
SPJX.Pnum WHERE NOT EXISTS JX  
          ( JX.Ci = 'Томск' AND NOT EXISTS SPJY  
            ( SPJY.Jnum = JX.Jnum AND  
              SPJY.Pnum = SPJX.Pnum  
            )  
          );
```

10) Получить значения номеров поставщиков, поставляющих
красные детали для изделий, производимых в Томске **ИЛИ** в Яе.

```
SPJX.Snum WHERE EXISTS PX EXISTS JX  
(  
    PX.Pnum = SPJX.Pnum AND PX.Co = 'красный' AND  
    JX.Jnum = SPJX.Jnum AND ( JX.Ci = 'Томск' OR JX.Ci = 'Яя' )  
);
```

10-б) Получить значения номеров поставщиков, поставляющих
красные детали для изделий, производимых в Томске **И** в Яе.

```
SPJX.Snum WHERE  
    EXISTS PX EXISTS JX  
    ( PX.Pnum = SPJX.Pnum AND PX.Co = 'красный' AND  
      JX.Jnum = SPJX.Jnum AND JX.Ci = 'Томск'  
    )  
AND  
    EXISTS SPJY EXISTS PX EXISTS JX  
    ( SPJY.Snum = SPJX.Snum AND  
      PX.Pnum = SPJY.Pnum AND PX.Co = 'красный' AND  
      JX.Jnum = SPJY.Jnum AND JX.Ci = 'Яя'  
    );
```

11) Получить значения номеров изделий, снабжаемых хотя бы одним поставщиком, расположенным не в том же самом городе.

```
JX.Jnum WHERE EXISTS SPJX EXISTS SX  
    (  
        JX.Jnum = SPJX.Jnum AND  
        SX.Snum = SPJX.Snum AND NOT ( JX.Ci = SX.Ci )  
    );
```

12) Получить значения номеров изделий, для которых не поставляется ни одной красной детали из Томска.

```
JX.Jnum WHERE NOT EXISTS SPJX EXISTS PX  
    (  
        SPJX.Pnum = PX.Pnum AND SPJX.Jnum = JX.Jnum  
        AND PX.Co = 'красный' AND PX.Ci = 'Томск'  
    );
```

13) Получить имена поставщиков, поставляющих деталь P2.

```
SX.Snam WHERE EXISTS SPJX  
    ( SPJX.Pnum = 'P2' AND  
      SPJX.Snum = SX.Snum  
    );
```

14) Получить имена поставщиков, поставляющих **все** детали.

```
SX.Snam WHERE FOR ALL PX EXISTS SPJX  
    ( SPJX.Pnum = PX.Pnum AND  
      SPJX.Snum = SX.Snum  
    );
```

15) Получить имена поставщиков, поставляющих **все поставляемые** детали.

```
SX.Snam WHERE FOR ALL PX  
    ( IF EXISTS SPJX  
      SPJX.Pnum = PX.Pnum  
    THEN EXISTS SPJY  
      ( SPJY.Pnum = PX.Pnum AND  
        SPJY.Snum = SX.Snum  
      )  
    );
```


16) Получить номера деталей, поставляемых для какого-либо изделия, производимого в Яе поставщиком из Яи.

РА Вариант 1.

1. Получить множество всех пар (Jnum, Snum) таких, что изделие и поставщик размещены в Яе:

T1 := (J JOIN S WHERE Ci = 'Яя')[Jnum, Snum];

2. Отобрать все кортежи SPJ, содержащие пары из T1.

T2 := T1 JOIN SPJ;

3. Выполнить проекцию T2 на атрибут Pnum.

Окончательно: ((J JOIN S WHERE Ci = 'Яя')[Jnum, Snum] JOIN SPJ)[Pnum];

РА Вариант 2.

((J JOIN SPJ JOIN S) WHERE Ci = 'Яя')[Pnum];

РИ (вариант не единственный!)

SPJX.Pnum WHERE EXISTS JX EXISTS SX

(JX.Ci = 'Яя' AND SX.Ci = 'Яя' AND

SPJX.Snum = SX.Snum AND SPJX.Jnum = JX.Jnum

);

17) Получить номера поставщиков, поставляющих хотя бы одну деталь которая поставляется хотя бы одним поставщиком, поставляющим красную деталь.

PA: T1 := (SPJ JOIN P WHERE Co = 'красный')[Snum];
T2 := (SPJ JOIN T1)[Pnum];
T3 := (SPJ JOIN T2)[Snum];

Окончательно PA:

```
( SPJ JOIN  
  ( SPJ JOIN  
    ( SPJ JOIN P WHERE Co = 'красный' )[ Snum ]  
  )[ Pnum ]  
)[ Snum ];
```

ПИ: SPJX.Snum WHERE EXISTS SPJY EXISTS SPJZ EXISTS PX
 (PX.Co = 'красный' AND
 SPJZ.Pnum = PX.Pnum AND
 SPJZ.Snum = SPJY.Snum AND
 SPJY.Pnum = SPJX.Pnum
);