

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по научно-исследовательской работе
Тема: Описание предполагаемого способа решения

Студентка гр. 4303

Полушина Ю.И.

Преподаватель

Фомичёва Т.Г.

Санкт-Петербург

2019

СОДЕРЖАНИЕ

1.	ОПИСАНИЕ МЕТОДА РЕШЕНИЯ	3
1.1.	Алгоритм преобразования в модель TR	3
1.2.	Описание используемых технологий	5
	Заключение	6
	Список использованных источников	7

2. ОПИСАНИЕ МЕТОДА РЕШЕНИЯ

2.1. Алгоритм преобразования в модель TR

Допустим, что r — запись некоторого файла на файловом уровне. В таком случае справедливо утверждение: хранимая форма записи r состоит из двух логических различных частей — множества значений полей и множества «связующих» данных, позволяющих связать друг с другом эти значения полей, причем для физического хранения каждой из этих частей можно воспользоваться широким спектром возможностей [2].

В системах с непосредственным отображением эти две части хранятся вместе. В таких системах связующая информация задается по принципу физической близости. В отличие от этого, в модели TR эти две части хранятся отдельно — значения полей находятся в таблице значений полей, а связующая информация — в таблице реконструкции записей. А именно такое разделение является основным источником многочисленных преимуществ, которые способна предоставить модель TR.

На рис. 2.1. приведен пример таблицы с данными, ниже приведено преобразование данной таблицы в таблицу значений полей и таблицу реконструкции записей.

Последовательность полей:		1	2	3	4
Последовательность записей:		#	Name	Status	City
	1	S4	Clark	20	London
	2	S5	Adams	30	Athens
	3	S2	Jones	10	Paris
	4	S1	Smith	20	London
	5	S3	Blake	30	Paris

Рисунок 2.1 – Исходная таблица с данными

На рис. 2.2 показана таблица TR, называемая таблицей значений полей, которая соответствует файлу на рис. 2.1. Стоит обратить внимание на то, что ее строки не соответствуют каким-либо очевидным образом тем записям, которые показаны на рис. 2.1.

Последовательность столбцов:		1	2	3	4
Последовательность строк:		#	Name	Status	City
	1	S1	Adams	10	Athens
	2	S2	Blake	20	London
	3	S3	Clark	20	London
	4	S4	Jones	30	Paris
	5	S5	Smith	30	Paris

Рисунок 2.2 – Таблица значения полей

Для того чтобы иметь возможность реконструировать показанный на рис. 2.1 файл из таблицы значений полей на рис. 2.2, требуется еще одна таблица — таблица реконструкции записей (рис. 2.3). Следует отметить, что в ячейках этой таблицы теперь находятся номера строк.

Последовательность столбцов:		1	2	3	4
Последовательность строк:		#	Name	Status	City
	1	5	4	4	5
	2	4	5	2	4
	3	2	2	3	1
	4	3	1	1	2
	5	1	3	5	3

Рисунок 2.3 – Таблица реконструкций записей

Таблица значений полей — это единственная таблица TR, которая содержит пользовательские данные как таковые. Все остальные таблицы содержат внутреннюю информацию, которая имеет смысл только для модели TR, но непосредственно не касается пользователя. Формируется таблица значений полей с помощью сортировки: каждый столбец таблицы содержит значения из соответствующего поля файла, отсортированные в порядке возрастания. Стоит отметить, что независимо от первоначального расположения записей в файле всегда формируется одна и та же таблица значений полей.

Формирование таблицы реконструкций записей происходит с помощью более сложного алгоритма чем сортировка, который описан ниже:

- Каждый столбец сортируется по возрастанию.
- Для каждого столбца выполняется обратная перестановка. Например, столбец 43512 преобразуется в 45213.

```
fun reversePerm(p : int[], rep : int[])  
    for i = 1 to n  
        rep[p[i]] = i;
```

- Для каждого столбца, поочередно для каждой строки необходимо выполнить следующее: перейти в ячейку [i, 1] таблицы обратных перестановок. Пусть эта ячейка содержит значение r; кроме того, допустим, что следующая ячейка справа [i, 2] содержит значение r1. Перейти в r-ю строку таблицы реконструкций записей и поместить значение r1 в ячейку [r, 1].

2.2. Описание используемых технологий

Инструмент преобразования представляет собой настольное приложение с запуском из командной строки, которое на вход будет получать SQL скрипты.

В качестве используемых технологий разработки предполагается следующий набор:

- Python
- SQL

ЗАКЛЮЧЕНИЕ

В результате проведенной работы был выполнен начальный этап разработки инструмента для позволяющего работать с моделью TransRelational. Проведен анализ алгоритма преобразования в TR модель, выбраны инструменты для решения поставленной задачи.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. U.S. Patent and Trademark Office: Value-Instance-Connectivity Computer-Implemented Database. U.S. Patent No. 6,009,432. - December 28, 1999.
2. Дейт, К. Дж. Введение в системы баз данных, 8-е издание/ М.: Издательский дом "Вильямс", 2005. — 1328 с.
3. Date C. J. Go Faster! The TransRelational™ Approach to DBMS Implementation