**{{{Создание системы для OLAP-кубов**

Миронов Дмитрий Сергеевич, студент магистратуры

МИРЭА — Российский технологический университет (г. Москва)}}}

*В статье автор описывает построение системы, позволяющей быстро внедрять аналитические методы и принимать решения на основе данных.*

***Ключевые слова:*** *OLAP, куб, аналитика, индекс, данные.*

При создании системы для OLAP-кубов [1] нужно учитывать две основные проблемы: потребляемая память при расчете каждой агрегации куба и скорость расчета всех агрегаций, поскольку при каждом последующем расчете агрегаций объем куба увеличивается в несколько раз.

На текущий момент есть два основных решения в области BI: Hyperion planning и Qlik sense. Оба решения имеют ограниченный функционал и высокую стоимость владения. Детально прописывая план внедрения платформы, многие компании сходятся во мнении [2], что быстрее и дешевле создать свою платформу, используя более современные инструменты анализа данных, разработанные для DS (data science), и открытые библиотеки для визуализации для современных фреймворков JavaScript.

В данной статье описывается построение системы, позволяющей быстро внедрять аналитические методы и принимать решения на основе данных. Ввиду того, что в создаваемой системе основной функционал будет построен на создании OLAP-куба.

**Пример функционального решения**

Перед началом формирования OLAP-куба необходимо создать его структуру (рис. 1), то, из чего он будет состоять. Основой, конечно же, являются данные и стороны. Сторона — это измерение куба, то, что будет группироваться при формировании куба.

Необходимо указать столбцы основного файла и их иерархию. Пример иерархии или же одной стороны: «месяц — неделя — год». Данная сторона будет называться в структуре, например, «дата». Таких сторон в кубе может быть неограниченное количество, но с каждой добавляемой стороной и глубиной иерархии увеличивается объем куба и сложность при его расчете. В качестве данных необходимо указать столбцы с числовыми значениями, на основе которых будут проводиться расчеты.

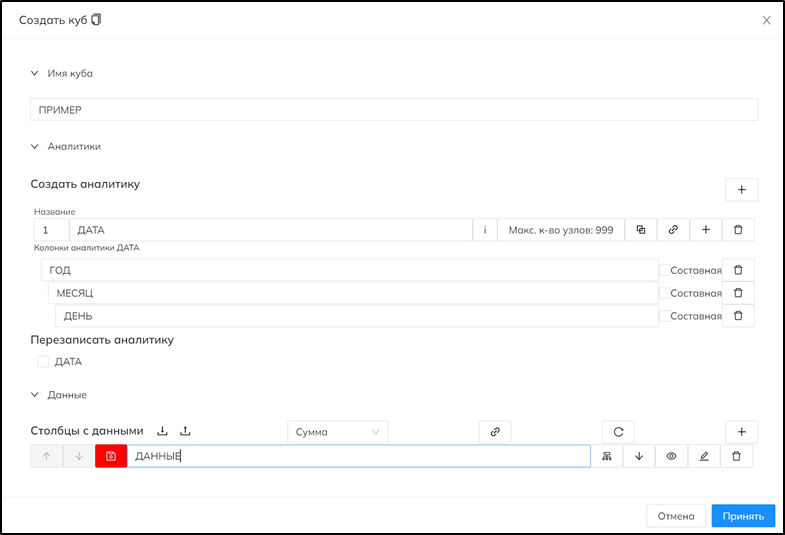


Рис. 1. Формирование структуры

*Источник: разработано автором*

После описания сторон и данных куба формируются параметры каждой стороны и столбцов с данными, для того чтобы эффективно хранить полученные значения и быстро выводить данные при запросе.

Для каждой стороны в структуре должна находиться следующая информация: название стороны, названия столбцов иерархии стороны, порядковый номер стороны, длина индекса для данной стороны.

**Как формируется куб**

Основная идея формирования куба заключается в том, чтобы формировать каждый последующий разворот куба на основе предыдущего. Так, идя от начальной таблицы, мы постепенно будем ее увеличивать, добавляя сгруппированные значения, и из полученной таблицы будем рекурсивно группировать её по каждой иерархии стороны (рис. 2).

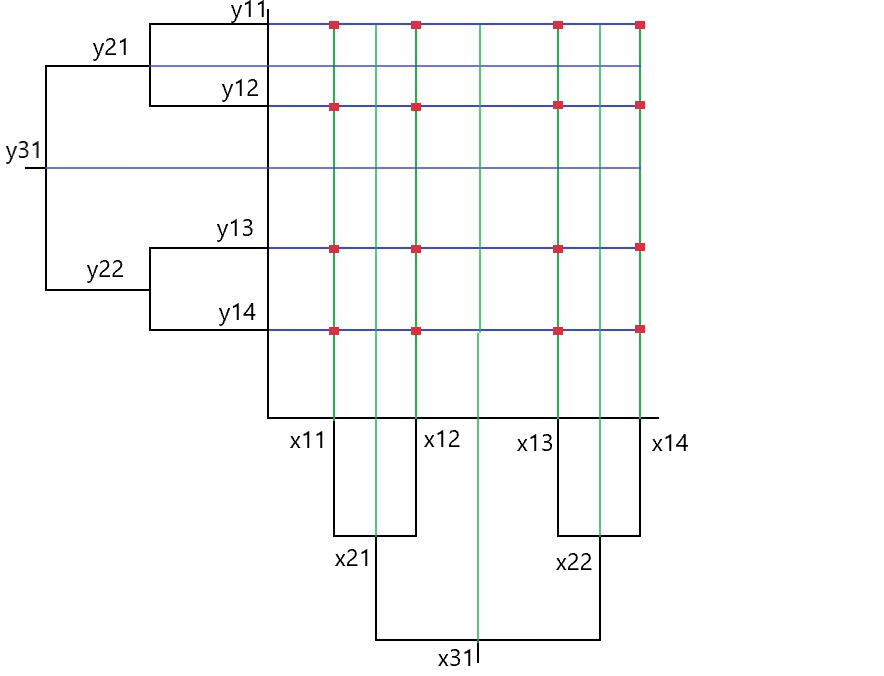


Рис. 2. Агрегация в двумерной плоскости

*Источник: разработано автором*

На рис. 2 представлена первая агрегация, при которой формируются значения на листьях. Добавляя полученные значения к основным, мы можем делать группировку следующего уровня иерархии стороны. Нам не приходится каждый раз рассчитывать куб до нужного уровня, поскольку при каждой агрегации уровня иерархии стороны сформированные данные уже будут, но используя больше памяти при каждой последующей агрегации над кубом. Данную проблему можно решить следующим образом: каждый столбец данных считать отдельно, последовательно. Таким образом, если в изначальных столбцах данных находятся 100 столбцов, мы соберем куб 100 раз для каждого столбца; следовательно, мы уменьшим объем потребляемой памяти при формировании куба, но скорость формирования всего куба увеличится.

**Как формируются индексы**

С точки зрения математики для того чтобы хранить плоскость на прямой, необходимо каждой точке дать свой уникальный индекс (номер). Таким образом, если у нас многомерная плоскость, то каждой из плоскостей нужно присвоить свой индекс. Сформированный куб — это и есть многомерное пространство плоскостей. Каждый индекс будет состоять из индексов стороны.

Для того чтобы сформировать индекс стороны, нужно подготовить таблицу, в которой они будут храниться. Данная таблица будет состоять из уникальных значений каждого уровня иерархии стороны; в столбце «имя» будут находиться имена всех уникальных значений стороны, а в столбце «индекс» — число для каждого уникального значения из столбца «имя».

Перед подстановкой индексов нужно отредактировать агрегированные данные. Для этого нужно создать столбцы с названиями всех сторон и подставить первое непустое значение из названий столбцов иерархии сторон. Таким образом, в итоговой таблице количество столбцов окажется равным количеству сторон.

Далее необходимо заменить значения в столбцах на значение в таблице с индексами и соединить каждую строку в одну ячейку. Таким образом, получатся индексы для каждого значения OLAP-куба.

**Получение разворота OLAP-куба по индексам**

Для того чтобы получить запрашиваемый разворот, необходимо сделать запрос, в котором указан индекс запрашиваемой стороны; если запрашиваемый разворот включает в себя две стороны, то в запросе нужно указать два индекса.

Приведем пример (рис. 3). Каждой стороне присвоен индекс, его максимальная длина — два знака, количество символов зависит от количества уникальных значений в столбце стороны: «ID чел» — 1, «дата» — 2, «данные» — 0 и 1. Столбец с данными в индексе всегда стоит последним и должен иметь минимум два значения индекса: для суммы всех столбцов с данными (это индекс 0) и для каждого столбца (индекс становится плюс 1). Так как в примере всего лишь один столбец с данными, индекс будет выглядеть следующим образом: 010100. На рис. 3 представлен разворот с индексами 000100 и 010100.

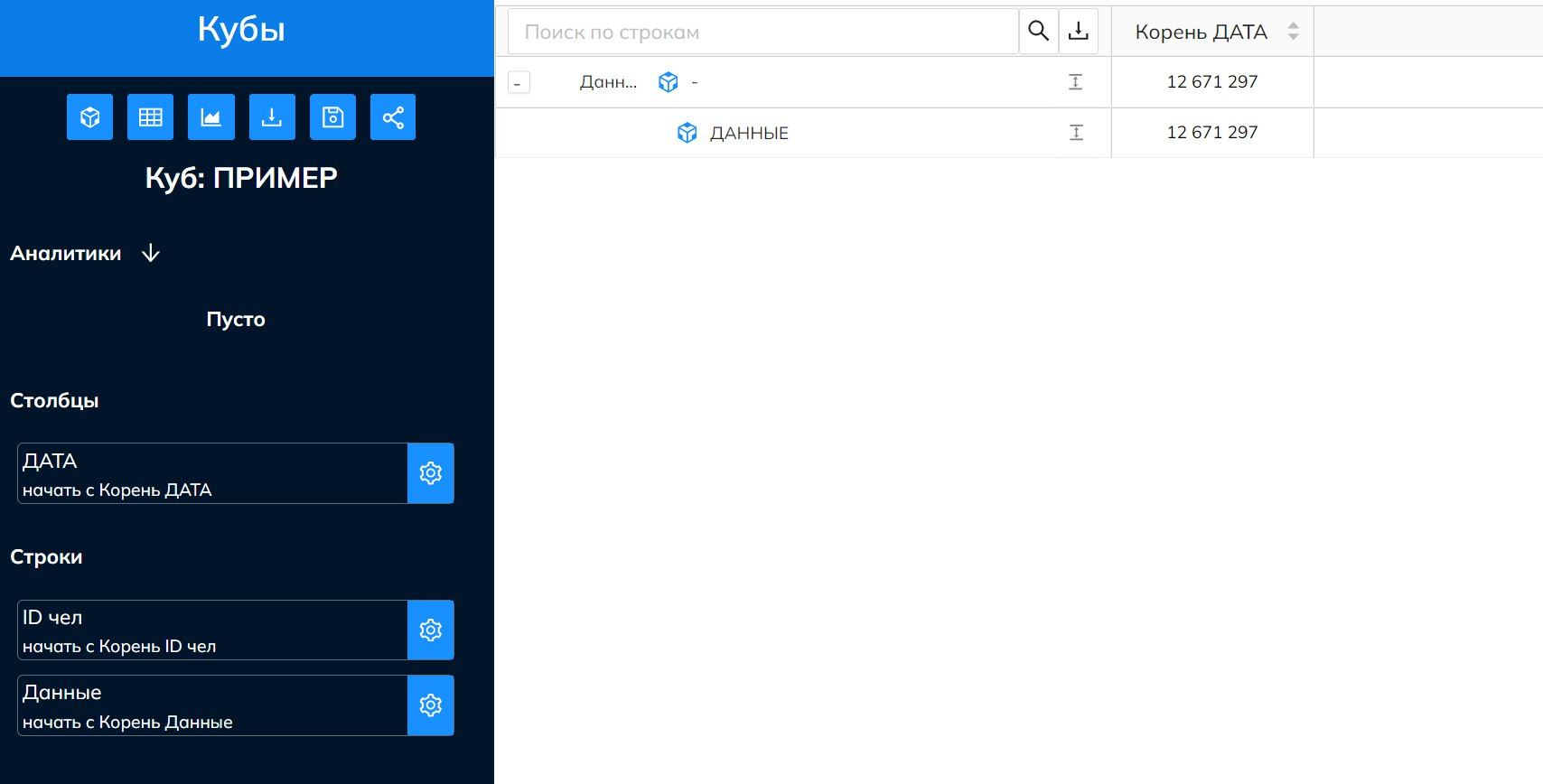


Рис. 3. Разворот куба

*Источник: разработано автором*

Если продолжить получать развороты, то мы увидим все больше данных (рис. 4).

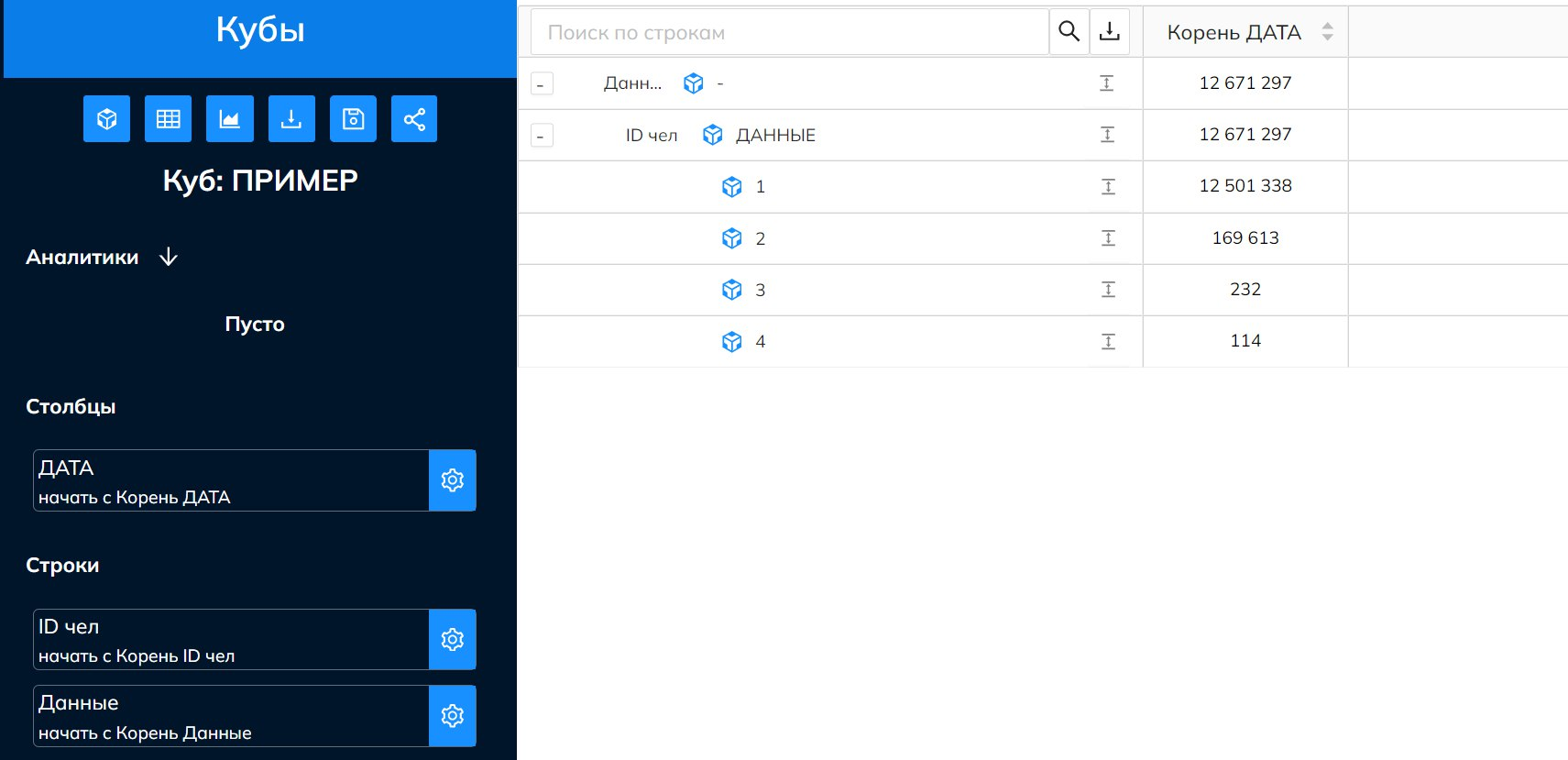


Рис. 4. Разворот куба

*Источник: разработано автором*

Если понадобится рассмотреть столбец «дата» более детально, то нужно нажать на соответствующий столбец; появятся уровни данной стороны (рис. 5).

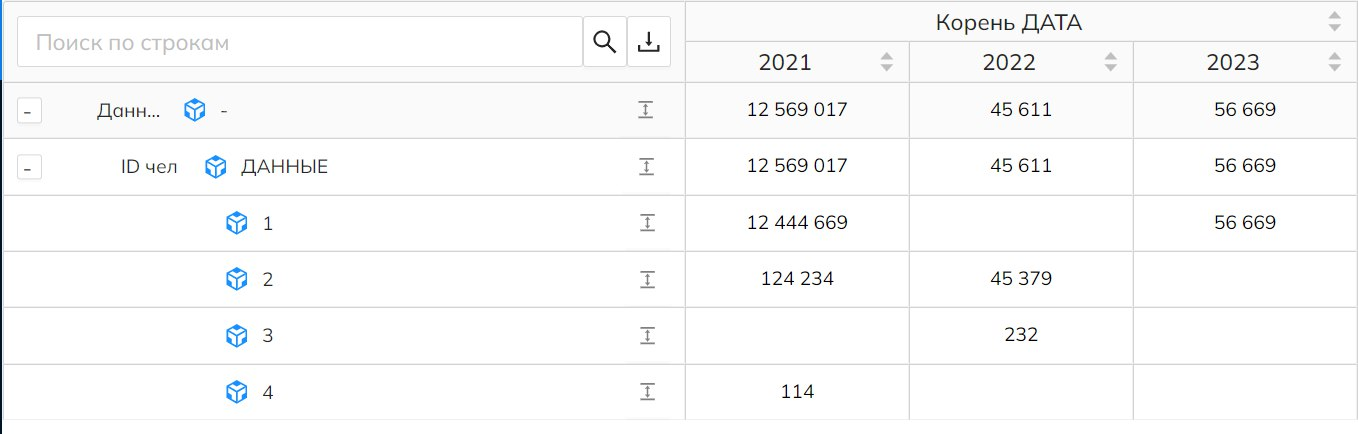


Рис. 5. Разворот куба с детальным разворотом столба

*Источник: разработано автором*

Литература:

# Введение в OLAP и многомерные базы данных — URL: http://www.olap.ru/basic/alpero2i.asp (дата обращения: 25.11.2023).

# «I Was Seduced By a Build Scenario»: 11 Ways to Avoid This Exec’s Greatest Tech Failure//Better Cloud. — URL: https://www.bettercloud.com/monitor/build-vs-buy/ (дата обращения: 23.11.2023).