OLAP-анализ данных: решаемые задачи, технологии, актуальные проблемы

Кудрявцев Юрий, ВМиК МГУ
mail@ykud.com
22 февраля
ACM SIGMOD Москва

Организационное

- http://ykud.com/sigmod/2007
- 1,5 часа Вашего времени
- Вопросы приветствуются

План

- Задачи анализа, определение OLAP
- OLAP-расширения SQL
- MOLAP продукты
- Вопросы, перспективные технологии

Появление термина OLAP

Статья Кодда “Providing OLAP for End-User Analysis” отосланная в IEEE Computer – 12 признаков OLAP-системы

12 признаков OLAP.

1. **Многомерная концепция данных**. OLAP оперирует CUBE данными, которые являются многомерными массивами данных. Число измерений OLAP кубов не ограничено.
2. **Прозрачность.** OLAP системы должны опираться на открытые системы, поддерживающие гетерогенные источники данных.
3. **Доступность.** OLAP системы должны представлять пользователю единую логическую схему данных.
4. **Постоянная скорость выполнения запросов.** Производительность не должна падать при росте числа измерений.
5. **Клиент\сервер архитектура.** Системы должны базироваться на открытых, модульных системах.
6. **Различное число измерений.** Системы не должны ограничиваться 3хмерной моделью представления данных. Причем измерения должны быть эквивалентны по применению любых функций.

12 признаков OLAP.

7. **Динамическое представление разреженных матриц.** Идея относится к «нулям» в реляционных базах данных и сжатию больших файлов, «разреженная матрица» - матрица, не каждая ячейка которой содержит данные. OLAP системы должны содержать средства хранении и обработки больших объемов данных.
8. **Многопользовательская поддержка.** OLAP системы должны поддерживать многопользовательский режим работы.
9. **Неограниченные многомерные операции.** Аналогично, требованию о различном числе измерений : все измерения считаются равными и многомерные операции не должны накладывать ограничений на отношения между ячейками.
10. **Интуитивно понятные инструменты манипулирование данными.** В идеале, пользователи не должны пользоваться различными усложненными меню и прочим, чтобы сформулировать многоуровневые запросы.
11. **Гибкая настройка конечных отчетов.** Пользователи должны иметь возможность видеть только то, что им необходимо, причем все изменения данных должны немедленно отображаться в отчетах.
12. **Отсутствие ограничений на количество измерений и уровней агрегации данных**

НО

- Последние 4 страницы статьи посвящены Essbase – проверка соответствия OLAP критериям
- Жена Кодда в это время работает в Arbor Software (разработчик Essbase)
- Arbor Software спонсировало написание статьи

Результат

Журнал Computer после публикации официально изымает статью Кодда из своих архивов

Простое определение OLAP

- Nigel Pendse -- olapreport.com
- FASMI
  - FAST
  - Analysis
  - Shared
  - Multidimensional

Для задач анализа мы вводим «многомерность» данных

В SQL измерения – обычно аргументы запроса с Group By

Drill-up\down, slice&dice

2 задачи для примера:

- Кросс-таблица
- Нарастающий итог за квартал

OLAP-расширения SQL

- Группировка данных
  - Grouping Set
  - Rollup
  - Cube
- Row_Number(), Rank
- Window By

SQL-1999

На чем запускать запросы

- MySQL
- Microsoft SQL Server
- Oracle
- IBM DB2
- Postgres

Нужен
ORACLE 10.2.0.1.0 + OLAP Option (EE)

Таблицы не создаются – достаточно прав на TEMP

Grouping Set (grouping_sets.sql)

```
select dept,job_title, count(*) as
staff_quantity
   from
     emp_data
   group by grouping sets (dept,job_title)
```

EMPID	DEPT	JOB_TITLE
1	hr	manager
2	it	sysadmin
3	it	dba
4	hr	clerk
5	it	networkadmin
6	hr	clerk
7	it	networkadmin
8	it	clerk

DEPT	JOB_TITLE	STAFF_QUANTITY
hr		3
it		5
	networkadmin	2
	dba	1
	clerk	3
	sysadmin	1
	manager	1

Rollup (rollup.sql)

Group By Rollup (a,b,c) == Group by grouping sets (a,b,c)(a,b)(a)()

```
select dept,job_title, count(*) as staff_quantity
  from
    emp_data
  group by rollup (dept,job_title)
```

EMP_Data

EMPID	DEPT	JOB_TITLE
1	hr	manager
2	it	sysadmin
3	it	dba
4	hr	clerk
5	it	networkadmin
6	hr	clerk
7	it	networkadmin
8	it	clerk

DEPT	JOB_TITLE	STAFF_QUANTITY
hr	clerk	2
hr	manager	1
hr		3
it	dba	1
it	clerk	1
it	sysadmin	1
it	networkadmin	2
it		5
		8

Cube (cube.sql)

Group By Cube == Group by grouping sets (a,b,c)(a,b)(b,a)(b,c)(a)(b)(c)()

```
select dept,job_title, count(*) as staff_quantity
  from
    emp_data
  group by cube (dept,job_title) ;
```

EMP_Data

EMPID	DEPT	JOB_TITLE
1	hr	manager
2	it	sysadmin
3	it	dba
4	hr	clerk
5	it	networkadmin
6	hr	clerk
7	it	networkadmin
8	it	clerk

DEPT	JOB_TITLE	STAFF_QUANTITY
		8
	dba	1
	clerk	3
	manager	1
	sysadmin	1
	networkadmin	2
hr		3
hr	clerk	2
hr	manager	1
it		5
it	dba	1
it	clerk	1
it	sysadmin	1
it	networkadmin	2

Row_Number

- Возвращает номер кортежа
- Варианты определения:
  - IDENTITY (MicroSoft) – колонка в таблице
  - ROWID (ORACLE) – физический номер в сегменте
  - ROW_NUMBER() – функция (Sybase WatCom SQL)

Row_Number (rownum.sql)

```
select dept,job_title, row_number() over (order by empid) as row_num
  from
    emp_data;
```

DEPT	JOB_TITLE	ROW_NUM
hr	manager	1
it	sysadmin	2
it	dba	3
hr	clerk	4
it	networkadmin	5
hr	clerk	6
it	networkadmin	7
it	clerk	8

Ранжирование

- RANK RANK | DENSE_RANK RANK | DENSE_RANK | PERCENT_RANK RANK | DENSE_RANK | PERCENT_RANK | CUME_DIST -- разные типы ранжирования по значению меры

Window By (window_by.sql)

```
select region,month, sales, sum(sales)
   over (partition by region
         order by month asc
         rows 2 preceding)
   as moving_average
  from
   sales_data
```

Moving_Total – нарастающий итог за квартал

REGION	MONTH	SALES
south	1	20
south	2	30
south	3	20
south	4	40
south	5	50
south	6	60
north	1	5
north	2	7
north	3	10
north	4	20
north	5	5
north	6	10

REGION	MONTH	SALES	MOVING_AVERAGE
north	1	5	5
north	2	7	12
north	3	10	22
north	4	20	37
north	5	5	35
north	6	10	35
south	1	20	20
south	2	30	50
south	3	20	70
south	4	40	90
south	5	50	110
south	6	60	150

Oracle Model By

http://www.oracle.com/technology/products/bi/db/10g/model_examples.html

```
<prior clauses of SELECT statement>
MODEL [main]
[reference models]
    [PARTITION BY (<cols>)]
DIMENSION BY (<cols>)
MEASURES (<cols>)
    [IGNORE NAV] | [KEEP NAV]
[RULES
    [UPSERT | UPDATE]
    [AUTOMATIC ORDER | SEQUENTIAL ORDER]
    [ITERATE (n) [UNTIL <condition>] ]
    ( <cell_assignment> = <expression> ... )
```

Oracle Model By (model_by_simple.sql)

```
select region,month, sales
  from sales_data
   model
   partition by (region)
   dimension by (month)
   measures (sales)
   rules (sales[7] = (sales[5]+sales[6])/2)
   order by region, month;
```

Считаем продажи в 7ом месяце

REGION	MONTH	SALES
south	1	20
south	2	30
south	3	20
south	4	40
south	5	50
south	6	60
north	1	5
north	2	7
north	3	10
north	4	20
north	5	5
north	6	10

REGION	MONTH	SALES
north	1	5
north	2	7
north	3	10
north	4	20
north	5	5
north	6	10
north	7	7,5
south	1	20
south	2	30
south	3	20
south	4	40
south	5	50
south	6	60
south	7	55

Oracle Model By (model_by_running_total.sql)

Running_Total (Sales_RT) – накопленный итог продаж

```
select region,month, sales,sales_rt
  from
    sales_data
   model
   partition by (region)
   dimension by (month)
   measures (sales,0 sales_rt)
   rules
   (sales_rt[any] = case
                    when cv(month) = 1 then (sales[cv(month)])
                    else (sales_rt[cv(month)-1] + sales[cv(month)])
                    end
   )
   order by region, month;
```

Oracle Model By (model_by_running_total.sql)

Running_Total (Sales_RT) – накопленный итог продаж

REGION	MONTH	SALES
south	1	20
south	2	30
south	3	20
south	4	40
south	5	50
south	6	60
north	1	5
north	2	7
north	3	10
north	4	20
north	5	5
north	6	10

REGION	MONTH	SALES	SALES_RT
north	1	5	5
north	2	7	12
north	3	10	22
north	4	20	42
north	5	5	47
north	6	10	57
south	1	20	20
south	2	30	50
south	3	20	70
south	4	40	110
south	5	50	160
south	6	60	220

Oracle Model By (model_by_iterate.sql)

Прогноз продаж считаем на базе предыдущего прогноза и факта

```
select region,month, sales, sales_forecast
  from
    sales_data
   model return updated rows
   partition by (region)
   dimension by (month)
   measures (sales, 0 sales_forecast)
   rules ITERATE(100) UNTIL (ABS((PREVIOUS(sales_forecast[6]) - sales_forecast[6]) ) < 0.001 )
   (   sales_forecast[any] = case
                when sales_forecast[cv(month)-1] > 0 then (sales[cv(month) -1] + sales_forecast [cv(month)-1])/2
-- (sales_forecast[cv(month)] +
                else (1.5 * sales[cv(month)-1])
                end
   )
   order by region, month
```

Oracle Model By (model_by_iterate.sql)

Прогноз продаж считаем на базе предыдущего прогноза и факта

REGION	MONTH	SALES
south	1	20
south	2	30
south	3	20
south	4	40
south	5	50
south	6	60
north	1	5
north	2	7
north	3	10
north	4	20
north	5	5
north	6	10

REGION	MONTH	SALES	SALES_FORECAST
north	1	5	
north	2	7	7,5
north	3	10	7,25
north	4	20	8,625
north	5	5	14,3125
north	6	10	9,65625
south	1	20	
south	2	30	30
south	3	20	30
south	4	40	25
south	5	50	32,5
south	6	60	41,25

Проблемы ROLAP

- Хранение агрегатов (материализация) или вычисление на лету
- Моделирование измерений и вычислений
- Схемы хранения «снежинка» и «звезда» (Кимбалл и Инмон)

Достаточно ли подобных расширений SQL?

4 типа OLAP систем по Кодду

- Categorical – простые запросы
- Exegetical – многомерный анализ, drill-up\down
- Contemplative – изменение расчетных результатов, при изменении входных параметров
- Formulaic – задание правил поведения системы и цель, сценарное моделирование

Задачи 3его, 4го типа

- Goal-Seeking, BackSolving – многомерные обратные расчеты уравнений

Вводим данные в ячейку, являющуюся пересечением формул по двум измерениям (например, Продажи по всем продуктам (сумма), в Год (сумма))

Пересчет по профилям

Многомерные Базы Данных

- Статисческие базы данных (SBD)
- Модель данных изначально включающая измерения (с иерархиями) как объекты
- Особая роль измерения Время
- Использование многомерных формул

MOLAP-продукты

- Essbase
- Express
- Ms Analysis Services

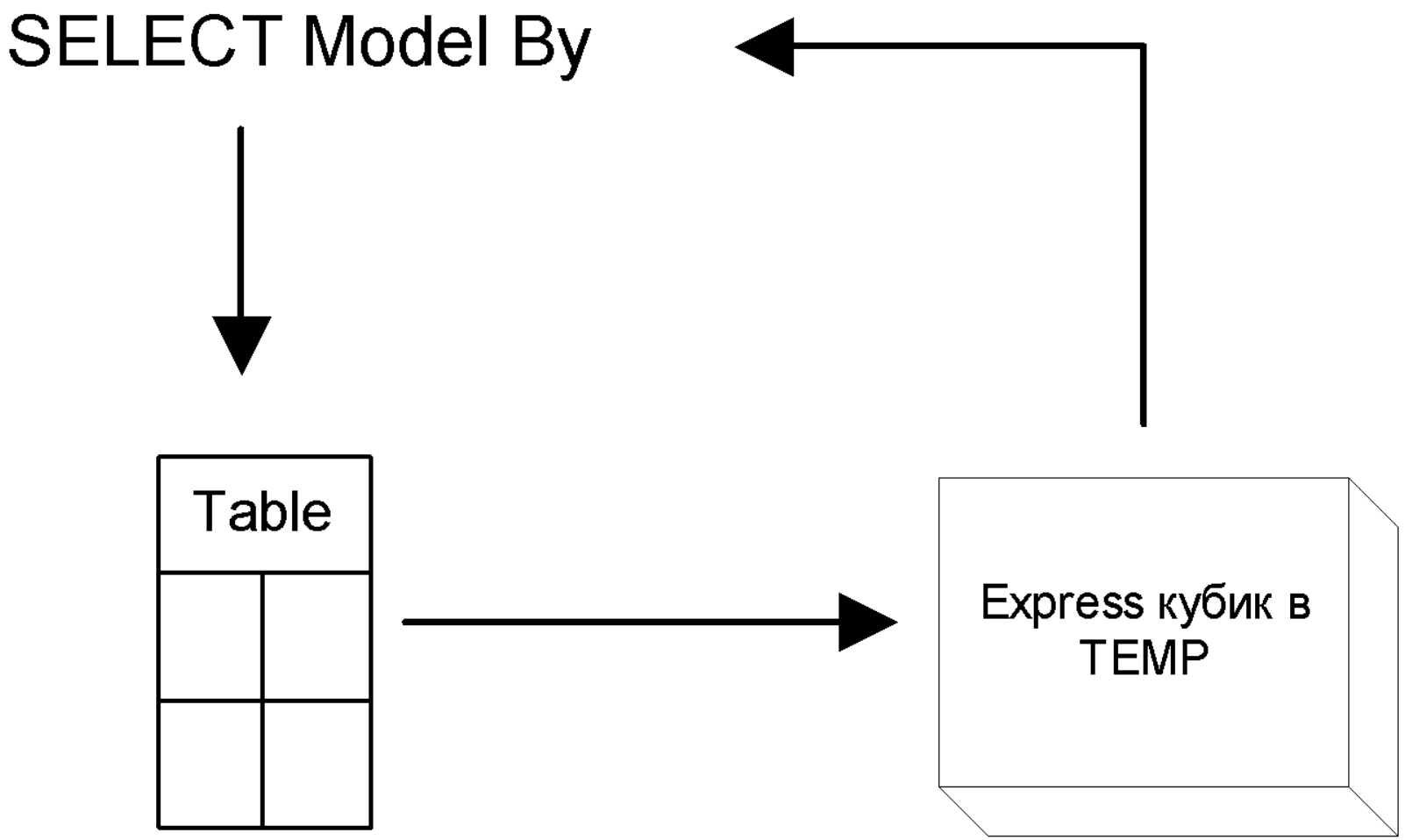
Hyperion Essbase

- Extended Spread Sheet database
- Arbor Software
- Роберт Эйрль – column-based storage

Oracle Express

- MIT (1975) ->…->IRI Software->Oracle (1995)
- Express 4GL
- Схема работы Model By

Схема работы Model By



MicroSoft

- Analysis Services – HOLAP движок
- Новая разработка
- Panorama -> Microsoft
- MDX – новый язык запросов к многмерным данным (поддерживается Hyperion)

MDX (пример синтаксиса)

```
SELECT
{ [Measures].[Dollar Sales], [Measures].[Unit Sales] }
on columns,
{ [Time].[Q1, 2005], [Time].[Q2, 2005] }
on rows
FROM [Sales]
WHERE ([Customer].[MA])
```

А что же IBM?

- Перепродавали Essbase, как IBM OLAP Server, прекратили в 2006
- Сделали IBM Cube Views – не продавался

Open-Source

- Mondrian – ROLAP, поддерживает MDX
- PALO – memory-based MOLAP, новая разработка

Стандарты OLAP

- OLAP Council.
  - JOLAP – поддерживался Hyperion и Mondrian => мертв
  - APB-1 Benchmark – набор тестов для определения производительности OLAP-движка
- XML\A – стандарт взаимодействия с MS Analysis Services (описание сервиса). Использует MDX.

Выводы и замечания

- Что такое OLAP?
- Нет стандартов, ни в модели данных, ни в языках
- Статистические пакеты сближаются с OLAP-приложениями
- Сервера отчетности не используют возможности SQL

Новые решения

- Языки векторного программирования (APL, K)
- In-memory базы данных (TimesTen, Applix, KX) как буфера для хранения агрегатов в СУРБД.
- Streaming OLAP.

Рекомендуемая Литература

- Codd E.F. Providing OLAP for end-user analysys: An IT mandate.
- Thomsen E. OLAP Solutions: Building Multidimensional Information Systems. Second Edition. Wiley, 2002.
- Rafanelli M. Multidimensional Databases — Problems and Solutions. Idea Grouping Publ., Hershey, London, Melbourne, Singapore, Beijing, 2003.
- Celko J. Analytics and OLAP in SQL. Morgan Kaufmann, 2006