РАЗРАБОТКА ВЕБ-ПРОЕКТОВ

7-я Международная конференция эффективной разработки на PHP

29-30 мая 2008
http://PHPConf.ru
2008@phpconf.ru
+7 (495) 585-92-61

XML Native Database Sedna 3.0

Климов Евгений (Slach)

Александр Календарев (Alexandre)

Вступительное слово

Сразу после phpConf2007, работаю в Start-Up'e ;)

Есть приложение, в котором необходимо получать из внешних БД (1С 7.7, 8.0 и 1С8.1, Axapta или любого другого источника) данные об ассортименте, складском остатке, ценовой и скидочной политике
Источников много и предполагается что одновременно большое кол-во данных будет заливаться в систему по расписанию или хаотично

100 источников = (~20-25 Gb данных + 6 Gb индексов) = 1 Shard,
хотим около 100 Шардов =)
Много трудностей при масштабировании:

- PRIMARY KEY - GUID CHAR(36) vs ID auto_increment (сдох InnoDB)
- тормозит ПОИСК+UPDATE (MyISAM table lock)
- MySQL Cluser не годится потому что IN MEMORY, а у нас не так много серверов и денег ;)
- Sphinx Fulltext search требует ID auto_increment
- Нельзя кешировать результаты поиска и т.п.

Видимо мы плохо знаем MySQL, ЧТО ДЕЛАТЬ?? ;-)

Существующие XML- native DB:

- **Apache XIndice** (XML:DB XUpdate, JDCB, XML_RPC Работает под с Apache Tomcat)
- **eXist** (Реализована на Java)
- **MonetDB** (API: ODBC, PHP, Phyton, XQuery processor поверх SQL)
- **MySQLDB** (Работает поверх MySQL)
- **Sedna** (ACID) API : C/C++, Perl, Java, PHP, Python
  mod_sedna (apache) XQuery triggers, updates,
  Tamino коммерч Россия, API все яз.
- **XStreamDB** коммерч, Java

Основные черты XML- native DB:

- Определена логическая модель XML документа
- XML документ – представлен, как фундаментальная часть хранилищ
- для доступа к хранилищу информации должен использоваться язык запросов XQuery

возможное применение XML- native DB :

- использование в Web-службах
- генерация суммарных отчетов данных из XML
- поиск релевантных документов в слабоструктурированных данных
- публикация данных для WEB непосредственно в XHTML
- формирование сводных XML документов

Sedna XMLDB что это?

Полноценная СУБД (а не надстройка над SQL), специально спроектированная для XML представления данных и работы с ними на языке запросов XQuery

Аналоги: Oracle Berkeley XMLDB
http://www.oracle.com/technology/tech/xml/xmldbАналоги: Oracle Berkeley XMLDB
http://www.oracle.com/technology/tech/xml/xmldb, IMB DB2 PureXML
http://www-306.ibm.com/software/data/db2/xml/, MSSQL XML тип данных

Субъективные преимущества Sedna перед аналогами

- **GNU GPL лицензия, win32 порт**
- **php5 extension (в том числе для win32), Python, C++/Java Client API**
- **СДЕЛАНО в России!! в ИСП РАН, релиз 0.1 в 2004 году, текущая версия 3.0.xx**

Кардинальное отличие от SQL

- **Документы / Коллекции документов вместо таблиц – как следствие нет жесткой структуры данных и ее можно изменять на лету,**
- **XML Узлы - вместо строк и столбцов таблицы**
- **XQuery, XPath и XUpdate вместо SQL – перебор узлов вместо поиска пересечения множеств строк в SQL (очень грубое сравнение)**
- **Мечта восторженного неофита – весь сайт в «одном XML файле» ;)**

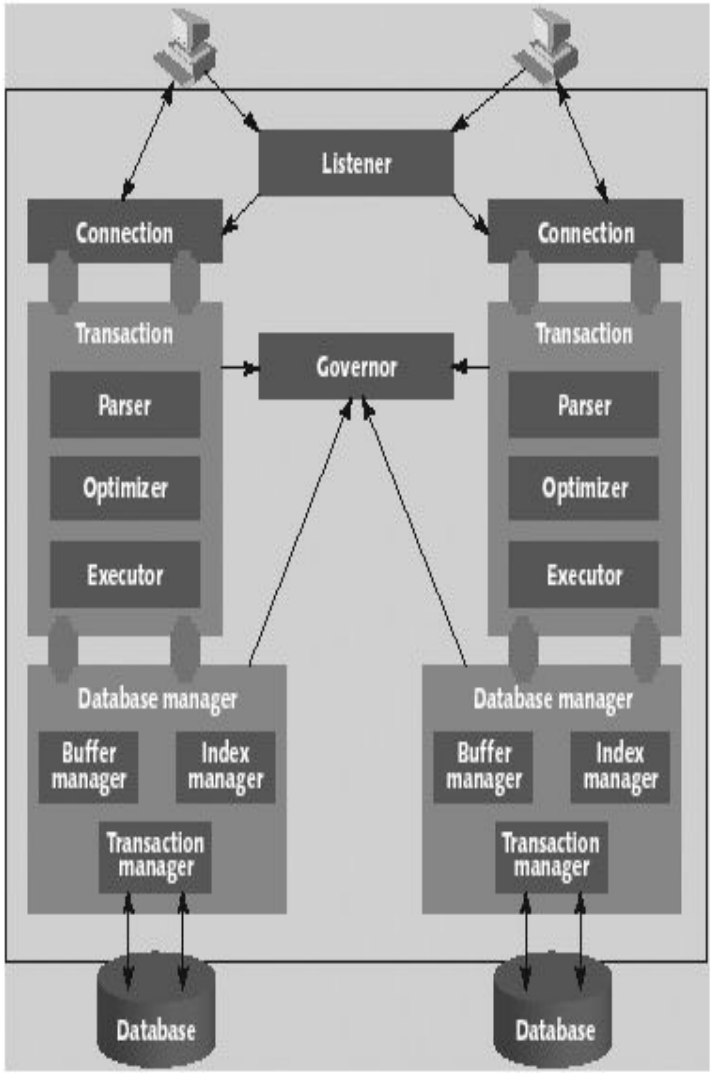
SednaXMLDB основные возможности

- **транзакционные операции с данными (ACID модель);**
- **резервирование данных;**
- **авторизация и разделение доступа;**
- **оптимизация запросов;**
- **управление внешней памятью;**
- **индексирование документов;**
- **полнотекстовый поиск (сделан на основе dtSearch, планируется свой);**
- **возможности создания триггеров на определенные узлы XML-документа**;
- **возможности создания модулей с набором пользовательских ф-ций;**

SednaXMLDB - Поддерживаемые платформы:

- Linux x86 kernel version 2.4 or higher;
- Windows 2000/XP/2003/Vista;
- Mac OS X 10.4 and higher (PPC/Intel);
- PowerPC at IBM RS6000 running Debian Sarge.

SednaXML – общая архитектура (1/3)



Governer - регулятор, центральный компонент все остальные компоненты регистрируются у него, следит за выполнением транзакций

Listener - слушает сокеты, авторизует пользователя и создает Connection

Connection – получение XQuery запросов из сокета и отдача XML ответов, инициализация транзакций

Transaction – атомарная операция, locking данных на уровне Collection, Document или Nodeset'а, может быть несколько транзакций в рамках одного Connection, есть Rollback, нет вложенных транзакций? ;) состоит из нескольких компонентов, см. далее

SednaXML – общая архитектура (2/3)

Компоненты, инкапсулированные в Transaction
Parser – парсинг текстовых XQuery запросов в логическое представление, которое является деревом операций
Optimizer – составляет план запроса на основе логического представления, создавая дерево низкоуровневых операций для манипуляции над физическими структурами данных
Executor – осуществляет выполнение низкоуровневых операций скармливая их Database Manager'у

Компоненты, инкапсулированные в Database Manager
Один Database manager на КАЖДУЮ базу данных, состоит из четырех компонентов
Buffer Manager - управление пулом буфферов как в RAM так и на диске, чем больше буфферов в RAM тем быстрее, Sedna прожорлива до памяти
Index Manager – управление обновлением и поиском по value и fulltext индексам
Transaction (Trigger) Manager – обеспечивает корректность обработки параллельных транзакций

SednaXML – общая архитектура (3/3)

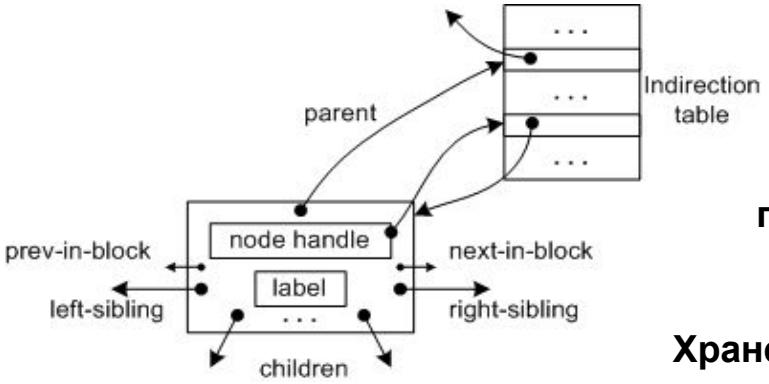


Рисунок (1)
Структура Node Descriptor внутреннее представление узла

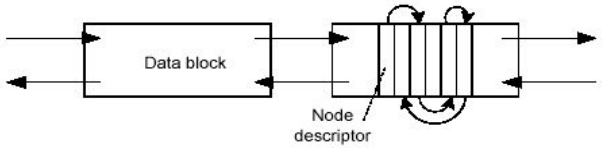


Рисунок (2)
Хранение дескрипторов узла в блоках «памяти»

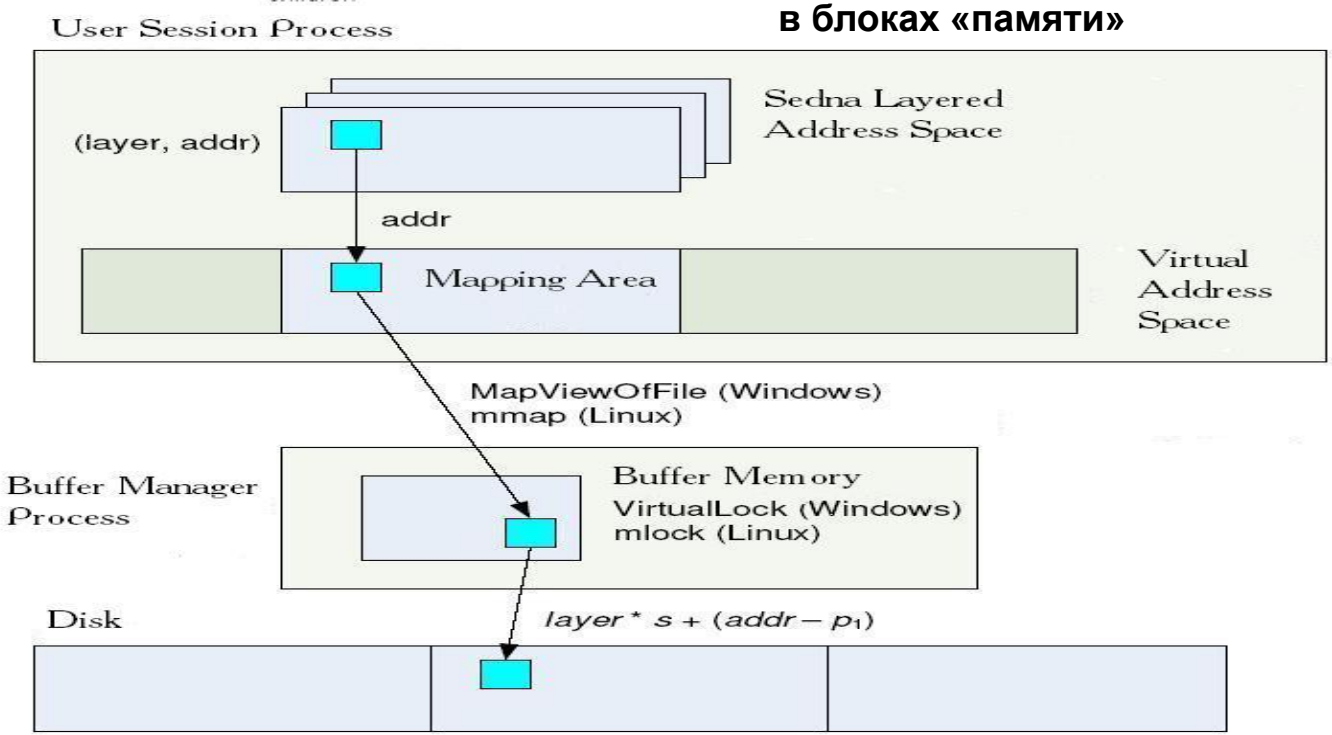


Рисунок (3)
Общая схема работы buffer manager

SednaXML – установка и администрирование (1/2)

Где скачать - http://modis.ispras.ru/sedna/sedna_download_register.htmlГде скачать - http://modis.ispras.ru/sedna/sedna_download_register.html и http://modis.ispras.ru/FTPContent/ (тут много вкусного ;)

Как ставить – под win32 просто распаковать, под Linux \ MacOSX make clean, make install ;) внешних библиотек использует мало, отдельных пакетов под Debian и портов под FreeBSD пока нет

Структура папок и файлов

cd /[senda-install-dir]; ls –la

/bin/ - исполняемые файлы

/etc/ - файлы конфигурации всего Sedna сервера

/cfg/ - конфигурация БД в виде [имя_базы]_cfg.xml

/share/ - конфигурация пользователей, ролей и прав, метаданные по всему серверу

/data/[имя_базы]_files/ - файлы данных отдельной БД, содержит следующие файлы

- [имя_базы].sedata – данные, текстовые значения из коллекций для узлов и атрибутов
- [имя_базы].setmp – временные данные, используемые во время обработки XQuery запроса
- [имя_базы].*.seph – файлы используемые для хранения узлов для документов и коллекций
- [имя_базы].*llog – лог транзакций

SednaXML – установка и администрирование (2/2)

Утилиты:

se_gov \ se_stop – запуск\остановка самого сервера (Governer)
se_rc – просмотр списка запущенных компонентов

se_cdb \ se_ddb [имя бд] – создание\удаление БД (конфигурация БД записывается в /sedna/cfg/имя_бд.xml), есть очень много параметров тонкой настройки как размера БД, так и менеджера буфферов

se_sm \ se_smsd – запуск\остановка менеджера БД (по умолчанию СРАЗУ выделяется 100Mb RAM, но надо больше, зато потом потребление не сильно растет ;)

se_term – терминал для исполнения XQuery команд и *.xq файлов
se_exp – экспорт\импорт\бекап данных в Sedna в чистом XML (!!!) формате

Создание пользователей и раздача прав:
se_term –name SYSTEM –pswd MANAGER [имя_базы]
CREATE USER "test" WITH PASSWORD "test“ &
GRANT ALL ON DATABASE TO "test” &
GRANT ALL ON COLLECTION “westtorg” TO "test” &
&\n – разделитель команд в se_term
Остальное смотрим в [sedna-install-dir]/doc/AdminGuide.pdf ;-)

Использование Sedna PHP Extension

Где скачать - http://modis.ispras.ru/FTPContent/api/
Как поставить
Win32: добавить в php.ini extension=php_sedna.5.2.2.dll (работает под 5.2.3),
Linux \ OSX \ FreeBSD: ./configure --with-sedna=shared,${PATH-TO-SEDNA}; make
отдельных пакетов под Debian и портов под FreeBSD пока нет

Простейший пример остальное см. документацию идущую с модулем

```
<?php
//установка соединения к sedna
$host='localhost'; $db='westtorg'; $user='test'; $passwd='test';
$sedna=sedna_connect($host,$db,$user,$passwd)
    or die('Sedna error:'.sedna_error().' '.sedna_ercls());
//указываем sedna что узлы в результате не должны идти в том же порядке,
// как они идут в исходном документе, часто это ускоряет выборки
$query = 'declare ordering unordered;'."\n\n";
//выполнение простейшего XQuery запроса к Sedna
$query .= 'collection("westtorg")/WT_DOCUMENT/WT_BODY/WT_GOODS/ROW';
$q = sedna_execute($query) or die(`Error:'.sedna_error().' '.sedna_ercls());
//фетчинг Nodeset, данные сериализуются на сервере в строку и бьются в массив
по корневым узлам
while($q && $node=sedna_result_array()) var_dump($node);
sedna_close($sedna) or die('Sedna error:'.sedna_error().' '.sedna_ercls());
?>
```

XQuery - **язык, который**

- **полезен для работы со структурированной и мало структурированной информацией**
- **независим от платформы**
- **строго типизированный**
- **совместим со стандартами:**

W3C NameSpace, XML Schema,

XML 1.1 , Xpath 1.0

W3C **Спецификации** XQuery

- **XML Path Language (XPath) 2.0**
- **XQuery and XPath Data Model**
- **XQuery and XPath Functions and Operators**
- **XQuery Formal Semantics**
- **XML Query Requirements**
- **XML Query Use Cases**
- **XSLT and XQuery Serialization**
- **XML Syntax for XQuery (XQueryX)**
- **XQuery and XPath Full-Text Requirements**
- **XQuery and XPath Full-Text Use Cases**

SednaXML возможности XQuery

- Выборка элементов или атрибутов из документов, коллекций и последовательностей узлов;
- Объединение данных из документов, коллекций и последовательностей узлов;
- Добавление новых элементов или атрибутов в результирующий набор узлов;
- Группировка и сортировка результирующего набора

SednaXML – основы XQuery (1/3)

Язык XQuery рекомендован W3C Консорциумом, как язык запросов для доступа к XML данным в XML ориентированных базах данных и является такой же неотъемлемой частью, как SQL в реляционных СУБД.

Отличие XQuery от XSLT в том что XQuery не является XML языком и служит именно для выборки и отображения, а не для трансформации, это язык формирующий XML, во много построенных на принципах XPath

XQuery много использует XPath выражения и функции, дополняя их своими конструкциями языка

XQuery может работать со строгой типизацией
XQuery умеет работать с датами
XQuery поддерживает работу с агрегирующими функциями

Для формирования выходного XML в XQuery активно используются конструкторы элементов
<tag attribute="{expression}">{ expression }</tag>
фигурные скобки отделяют XQuery выражение от конструктора элементов и аналогичны вставке Xpath выражений в XSLT

SednaXML – основы XQuery (2/3)

Основные конструкции – FLWOR (for, let, where, order, return)

```
(: комментарий, записывается так :)
for $n1 in Expression1,$n2 in Expression2
let $v2 := Expression3, $v3 in Expression4
where WhereExpression1, WhereExpression2
order by SortExpression
return Expression
(: частный случай FLWR :)
let $varname := Expression
```

Для последовательности выбранных узлов существуют операции комбинирования - intersect, except, union (можно заменить на XPath |)

```
<all_beer>
{ collection(“westtorg")//WT_GOODS/ROW[CODE=1000]
intersect
collection(“westtorg")//WT_GOODS/ROW[contains(NAME,”пиво”)] }
</all_beer>
```

Важно помнить о том, что intersect и except бессмысленно использовать для комбинирования узлов разных документов, поскольку узлы в разных документах никогда не могут быть идентичными.

SednaXML – основы XQuery (3/3)

Условное выражение IF

Требуется наличие всех трех условий (if, then и else), а выражение в условии if должно быть заключено в скобки.

```
<WT_PRICE>
{
 for $p in $westtorg/WT_PRICE/ROW
 if ($p/@discounted)
   then $p/BASE_PRICE
   else $p/RETAIL_PRICE
}
</WT_PRICE>
```

Кванторные выражения

```
some $n in (5,7,9,11) satisfies $n > 10
every $n in (5,7,9,11) satisfies $n > 10
```

SednaXML – основы XUpdate (1/2)

Вставка узлов в заданное место
UPDATE insert Expr1 (into|preceding|following) Expr2

Пример
Добавляем ROW в WT_GOODS перед строкой ROW для которой тег CODE=180

UPDATE insert <ROW><GUID>…</GUID></ROW>
preceeding collection(“westtorg")//WT_GOODS/ROW[CODE=180]

-Удаление узлов в заданном месте
UPDATE delete|delete_undeep Expr

delete_undeep удаляет только узел, а его потомков добавляет к родительскому узлу

Пример
Удаляем строку из WT_GOOS для которой тег CODE=180
UPDATE delete collection(“westtorg")//WT_GOODS/ROW[CODE=180]

SednaXML – основы XUpdate (2/2)

Замена узлов

UPDATE replace $var [as type] in Expr1 with Expr2($var)

Удвоение остатка для списка товаров

```
UPDATE replace
$g in collection("westtorg")/WT_GOODS_AMOUNT/ROW[CODE=(100,180,220)]
with
<ROW>
{
      (
      $g/@*,
      $g/node()[not(self::AMOUNT)],
      for $a in $g/AMOUNT
        return <AMOUNT>{$a*2}</AMOUNT>
      )
}
</ROW>
```

Переименование узла

UPDATE rename Expr on NewNodeName

```
UPDATE raname collection("westtorg")//WT_GOODS/ROW on NEW_ROW
```

SednaXML – работа с индексами (1/2)

Создание VALUED индекса

```
CREATE INDEX “goods_by_code”
ON collection(“westtorg”)/WT_DOCUMENT/WT_BODY/WT_GOODS/ROW
BY CODE AS xs:integer
```

Удаление

```
DROP INDEX “goods_by_code”
```

Выборка с использованием индекса

```
for $i in index-scan(“goods_by_code”,54888,EQ|LT|GT|GE|LE)
return parent::ROW
```

Выборка с использованием индекса по диапозону

```
for $i in index-scan-between(“goods_by_code”,54888,113856,EQ|LT|GT|GE|LE)
return parent::ROW
```

SednaXML – работа с индексами (2/2)

Создание FULL-TEXT Indexes используется с dtSearch

```
CREATE FULL-TEXT INDEX “goods_name_fulltext”
ON collection(“westtorg”)/WT_DOCUMENT/WT_BODY/WT_GOODS/ROW/NAME
AS “string-value”
```

Полнотекстовое сканирование fulltext индекса

```
ftindex-scan(
collection(“westtorg") /WT_DOCUMENT/WT_BODY/WT_GOODS/ROW/NAME,
«пиво and not балтика»)/parent::ROW
```

Полнотекстовое сканирование raw XML, без использования индекса, но с использованием dtSearch

```
ftscan(
collection(“westtorg") /WT_DOCUMENT/WT_BODY/WT_GOODS/ROW/NAME,
“пиво and not балтика”,
“xml")/parent::ROW
```

SednaXML – модули и функции

Модули представляют собой текстовые файлы с набором ф-ций под определенным namespace например - wt.xqlib

```
module namespace wt = "http://www.host.ru";
declare ordering unordered;

declare function wt:search_group_by_seller($query as xs:string) {
  <SEARCH_RESULT> { XQuery код ф-ции пропущен }</SEARCH_RESULT>
};
```

Загрузка и использование модуля очень просты

```
LOAD OR REPLACE MODULE "wt.xqlib"
import module namespace wt = "http://www.host.ru";
wt:search_group_by_seller("test")
```

Удаление модуля

```
DROP MODULE “http://www.host.ru”
```

SednaXML – триггеры (1/2)

Общий вид выражения для создания триггера

```
CREATE TRIGGERtrigger-name
 (BEFORE|AFTER)(INSERT|DELETE|REPLACE)
ON XpathExpression
 (FOR EACH NODE|FOR EACH STATEMENT)
DO
{
Update-statement($NEW,$OLD,$WHERE);
...
Update-statement($NEW,$OLD,$WHERE);
XQuery-statement($NEW,$OLD,$WHERE);
}
```

$NEW,$OLD,$WHERE используются только для триггеров узла (each node)

SednaXML – триггеры (2/2)

Таблица условий при которых стартует триггер

	update: INSERT	update: DELETE	update: REPLACE
Trigger event INSERT	**Длина пути trigger >= Длина пути update**		**Длина пути trigger >= Длина пути update**
Trigger event DELETE		**Длина пути trigger >= Длина пути update**	**Длина пути trigger >= Длина пути update**
Trigger event REPLACE			**Длина пути trigger >= Длина пути update**

SednaXML – практическое использование (1/7)

table_name	table_rows	avg_row_length
wt_barcode	81573	1232
wt_category_compare	5	150
wt_common_categories	20	393
wt_company	3	238
wt_contractor_compare	15	233
wt_discount_rule	5797	631
wt_goods	47126	273
wt_goods_attribute	0	0
wt_goods_characteristic	0	0
wt_goods_compare	2	730
wt_goods_into_organization	24544	590
wt_goods_quality	1	272
wt_organization	6	232
wt_price	41115	564
wt_price_type	4	87
wt_units	46815	690
wt_units_type	4	617
wt_warehouse	74	87

Как сейчас: SQL схема для поиска и импорта данных

18 таблиц, по 8-9 столбцов в каждой
Основных таблиц 5

Кол-во строк приведено для 1 из 100 компаний соответственно в реальности будут таблички по миллиону строк

Большая длинна записей, много CHAR полей

Связи через комбинацию полей
GUID CHAR(36) + company_id (int)

SednaXML – практическое использование (2/7)

Как сейчас: Типичный SQL запрос для поиска

```
SELECT SQL_CALC_FOUND_ROWS
  o.company_id,o.guid, o.name, MIN(p.price) AS min_price, COUNT(DISTINCT go.guid) AS
goods_count
FROM wt_goods AS g
INNER JOIN wt_goods_into_organization AS go
        ON (g.guid=go.goods_guid AND g.company_id = go.company_id AND (go.availability=1))
INNER JOIN wt_price AS p
        ON ( go.goods_guid=p.goods_guid AND go.company_id = p.company_id AND
go.goods_characteristic_guid=p.goods_characteristic_guid )
INNER JOIN wt_price_type AS pt
        ON (pt.guid=p.price_type_guid AND p.company_id = pt.company_id AND pt.is_default=1)
INNER JOIN wt_organization AS o
        ON ( o.guid=go.organization_guid AND o.company_id = g.company_id )
WHERE ((((g.name LIKE '%пиво%' OR g.fullname LIKE '%пиво%') AND (g.name LIKE '%туборг%' OR
g.fullname LIKE '%туборг%'))) AND g.company_id IN (2) AND g.is_group=0)
GROUP BY o.guid, o.company_id
ORDER BY min_price ASC LIMIT 0, 20;
```

EXPLAIN Выдает все select_type=SIMPLE, и type=ref, er_req, но при этом using temporary и using filesort
тут 5 JOIN и этот запрос отрабатывается около 1-1.5 sec всего на 40 тысячах в wt_goods, а что будет когда их станет 400 тысяч записей? Что

SednaXML – практическое использование (3/7)

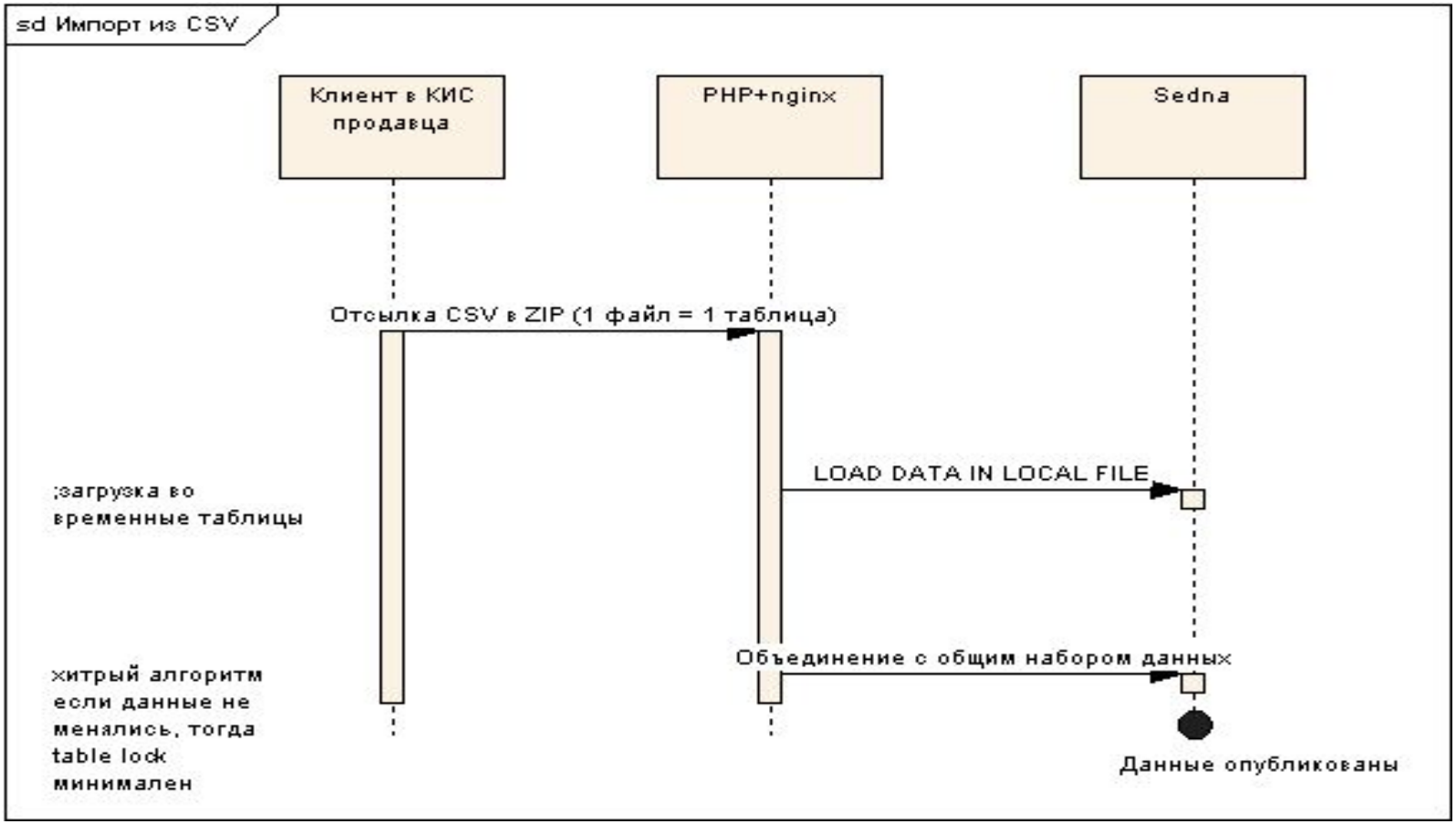
Как могло бы быть: Переделка запроса под XQuery

```
<SEARCH_RESULT> {
let $goods := (ftindex-scan("goods_fullname_fulltext", "пиво туборг")/parent::ROW
          union ftindex-scan("goods_name_fulltext", "пиво туборг")/parent::ROW),
   $organization_guid := distinct-values(
      $goods/WT_GOODS_INTO_ORGANIZATION/ROW/ORGANIZATION_GUID )
   for $og in $organization_guid
   let $org_name :=
distinct-values($goods/WT_GOODS_INTO_ORGANIZATION/ROW[ORGANIZATION_GUID
eq $og]/ORGANIZATION_NAME),
      $go := for $i in $goods/WT_GOODS_INTO_ORGANIZATION/ROW
where ($i/ORGANIZATION_GUID eq $og) and ($i/GOODS_GUID = $goods/GUID) return $i,
      $p := for $i in $go return $i/../../WT_PRICE/ROW
   return
    <ROW>
     <ORGANIZATION_GUID>{ $og }</ORGANIZATION_GUID>
     <ORGANIZATION_NAME>{ $org_name }</ORGANIZATION_NAME>
     <GOODS_COUNT>{ count($go) }</GOODS_COUNT>
     <MIN_PRICE>{ min($p/PRICE) }</MIN_PRICE>
    </ROW>
} </SEARCH_RESULT>
```

выполняется 0.2-0.6 sec!! , но надо переделать XML структуру документа

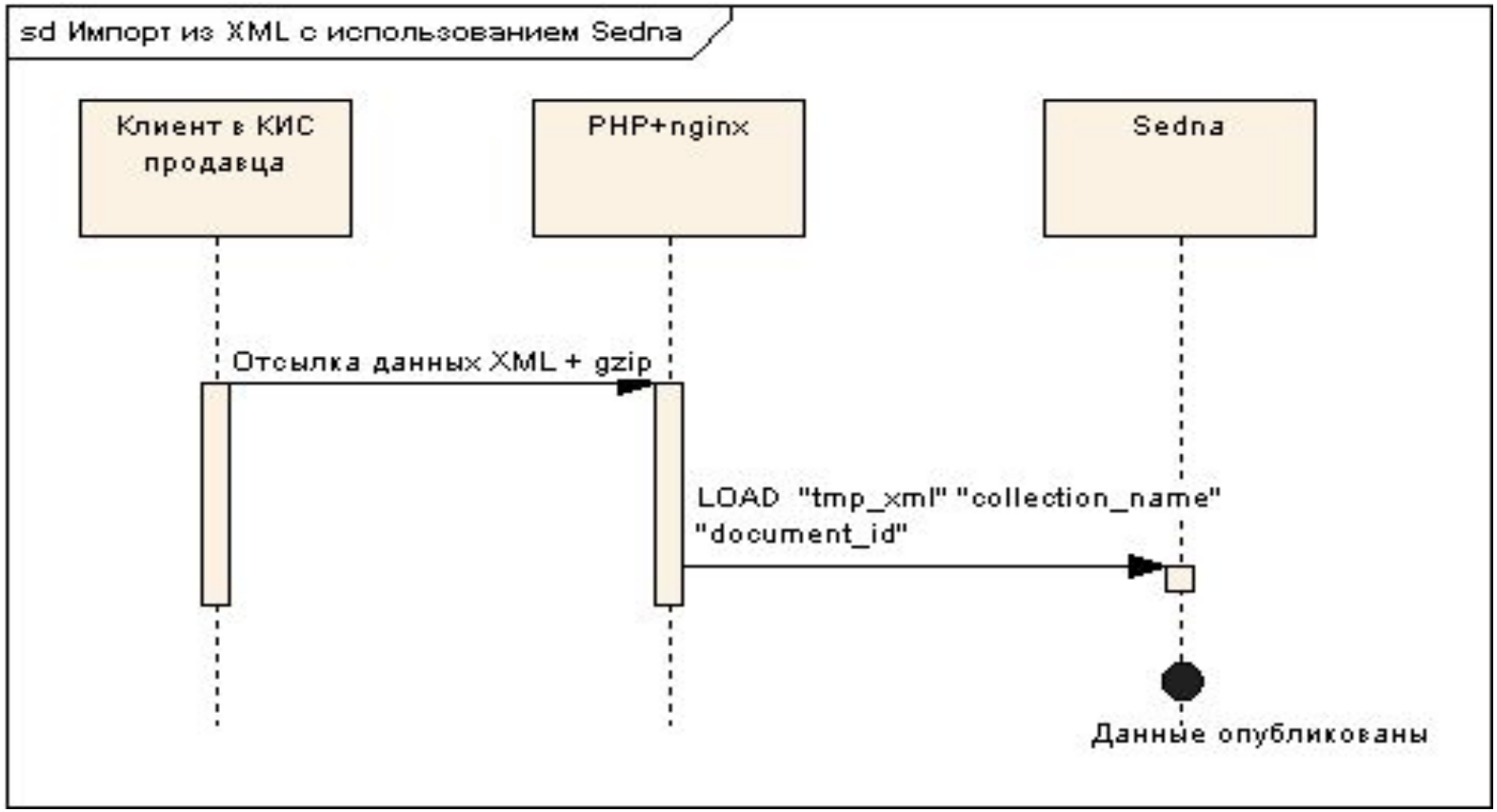
SednaXML – практическое использование (4/7)

Как есть:
схема импорта данных в MySQL с использованием LOAD DATA IN FILE – table lock для всех Engine, на Slave SELECT выстраиваются в очередь



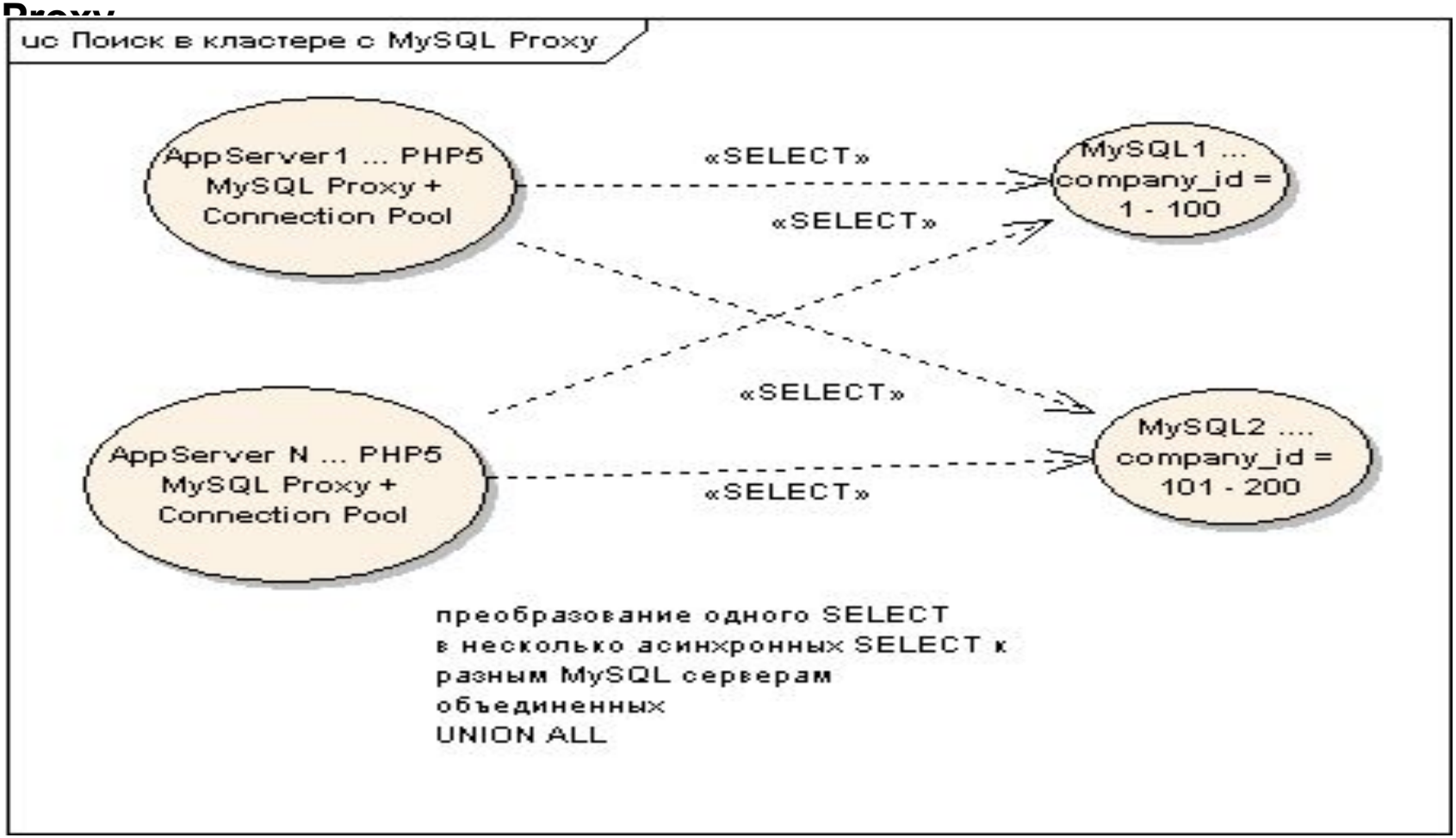
SednaXML – практическое использование (5/7)

Как можно сделать:
схема импорта данных в Sedna с использованием BULK LOAD
При BULK LOAD не блокируются транзакции на чтение, если сделают LOAD OR REPLACE то будет еще надежнее



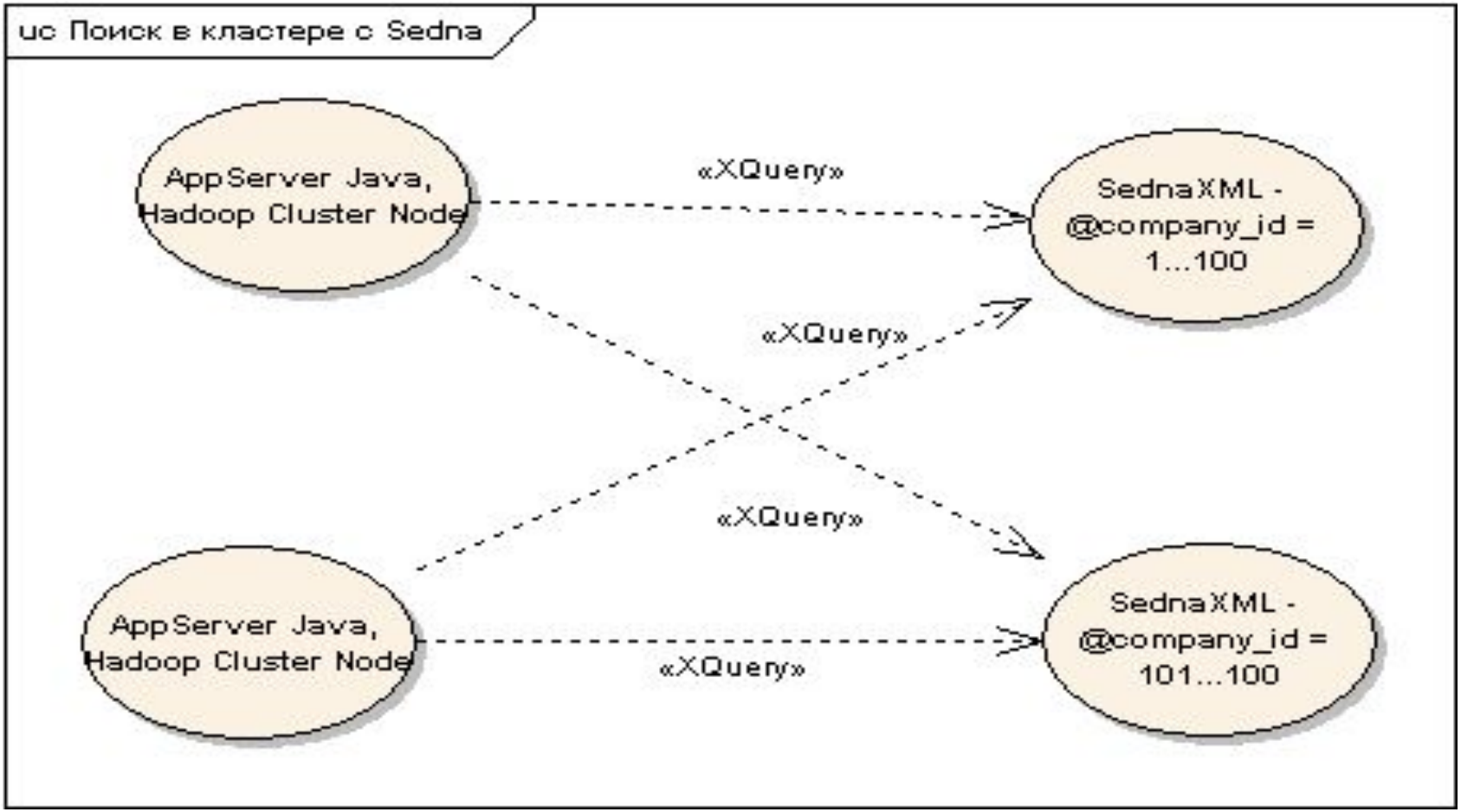
SednaXML – практическое использование (6/7)

Хочется: схема кластерного поиска в MySQL с использованием MySQL Proxy



SednaXML – практическое использование (7/7)

Хочется: схема кластерного поиска в Sedna с использованием Hadoop



Шаблоны проектирования XML - данных

- Базовые соглашения

- Отношение иерархии

- Отношение один ко многим

- Отношение многие ко многим

Шаблоны проектирования XML – данных

Базовые соглашения

Каждая таблица – это отдельный документ

Имя таблицы – имя root элемента в мн. числе
Строка – имя тбл. в ед числе
Имя столбца – имя аттрибута, если короткие данные

Исключение: колонки типа VARCHAR/TEXT
Которые имеют длинное текстовое значение
Лучше представлять в качестве вложенного тега. Имя колонки – имя тега, значение колонки – текстовое содержание тега.

Шаблоны проектирования XML – данных

Базовые соглашения - пример

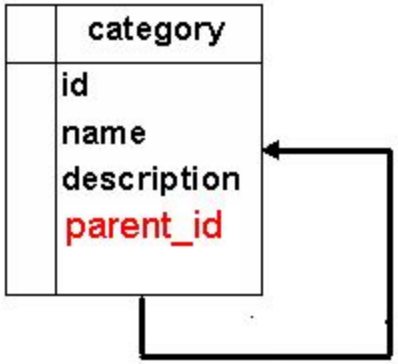
Пример:

	good
	id title description

```
<?xml version="1.0" ?>
- <goods>
  - <good id="6924">
      <title>Epson EMP-7950NL</title>
      <description>Новый проектор Epson EMP-7950NL обладает
        мощным световым потоком 4000 ANSI-лм при массе всего 5,8
        кг и оснащен беспроводным интерфейсом Wi-Fi. Оптическая
        система проектора создана на основе 1-дюймовых LCD-матриц
        и использует микролинзовые массивы, благодаря чему
        проектор обеспечивает столь мощный световой поток и
        высокую контрастность 700:1.</description>
    </good>
  - <good id="6925">
      <title>Epson EMP-760</title>
      <description>ультрапортативный проектор LCD, разрешение XGA
```

Шаблоны проектирования XML - данных

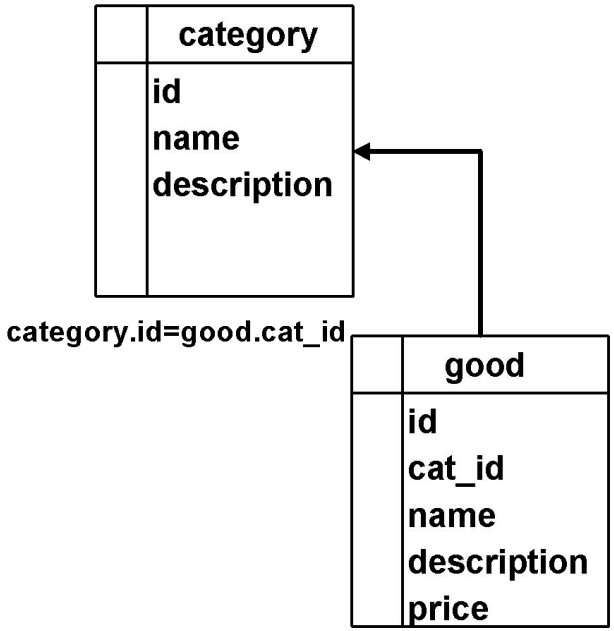
Отношение иерархии



<category >
<category>
<category>

Шаблоны проектирования XML - данных

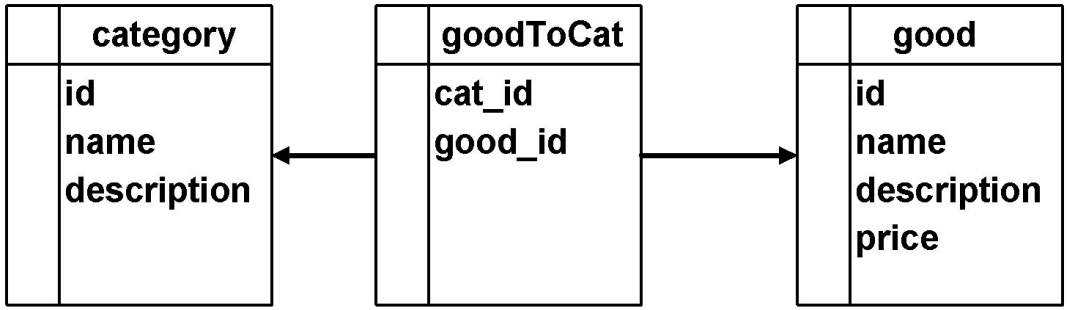
Отношение один ко многим




```
<category>
  <good @name @id @price >
    <description >
```

Шаблоны проектирования XML - данных

Отношение многие ко многим



SednaXML – текущие недостатки

- **Нет нормальных инструментов написания, отладки и профилирования XQuery запросов, необходимо существенно перестраивать сознание разработчика для того чтобы он мыслил не множествами а узалми**

- **Пока можно писать и отлаживать на Stylus Studio + Saxon XQuery, Oxygen XML Editor собирается со следующей версии поддерживать Sedna**

- **Полнотекстовый поиск в текущей реализации сделан на dtSearch – платная (~1000$ на один Linux сервер), летом 2008 планируется выпуск собственной реализации full-text индексов**

- **Отсутствие возможности валидации XML данных через XMLSchema и DTD, отсутствие поддержки CDATA (планируется реализовать)**

- **Высокий коэффициент роста дискового пространства относительно сырых XML данных (600Mb sedna vs 100Mb pureXML)**

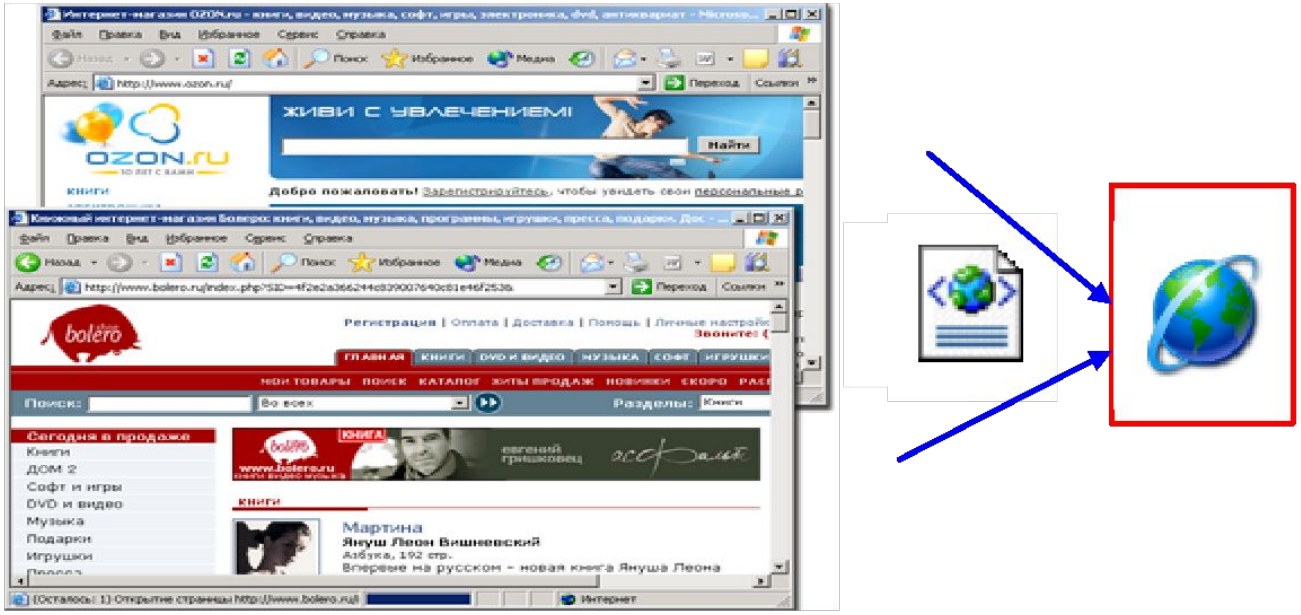
- **IMHO пока что очень слабые предпосылки для scale-out масштабирования и кластеризации**

Коротко о проекте

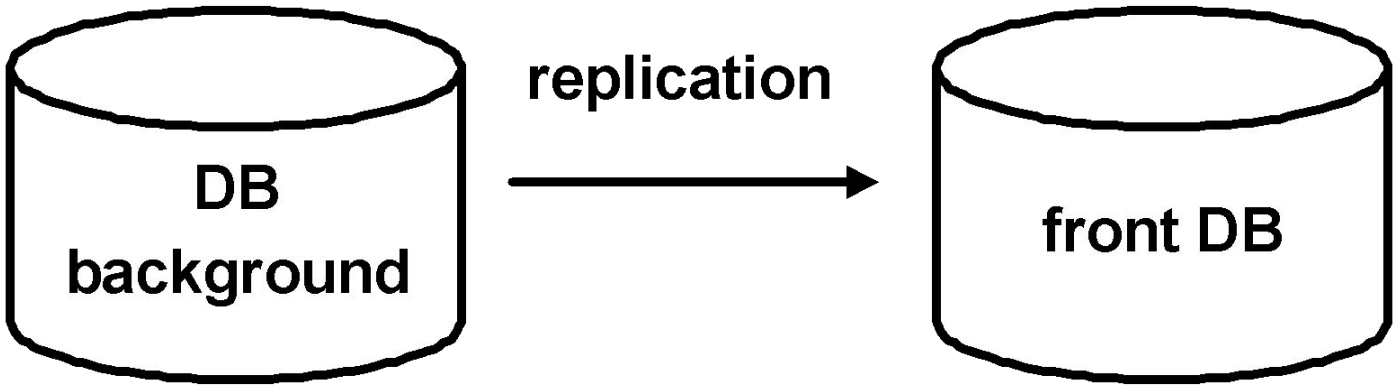
~ 1,5 **млн – предложений**

- 600 **магазинов**
- 700 **категорий товаров**

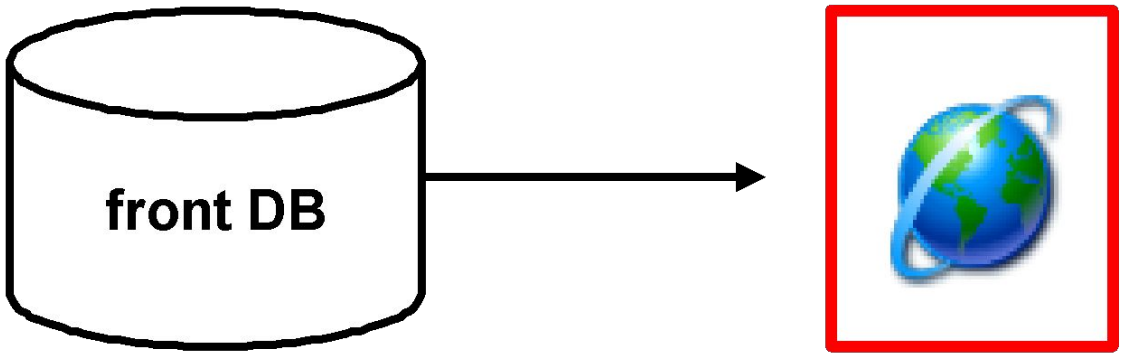
Проект – изнутри (1/3)



Проект – изнутри (2/3)

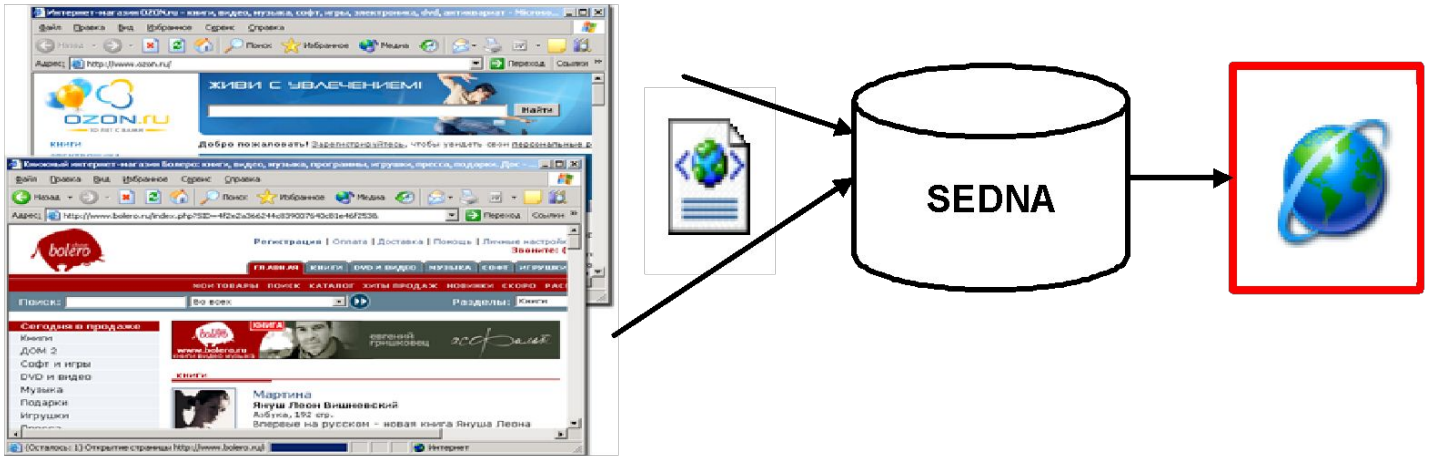


Проект – изнутри (3/3)



Publication

как могло бы быть:



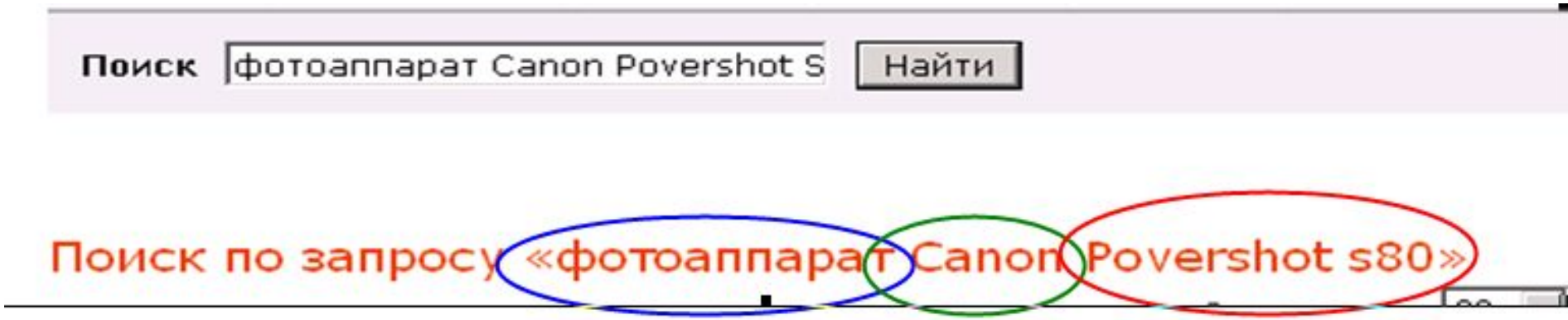
где использовать SednaXML в проекте

- Прием и хранение XML Документов
- Обобщение и обработка XML Документов
- Публикация
- Поиск Информации

Поиск информации

- Выбор шаблона поиска
- Выбор информации по шаблону

Шаблоны поиска



- Категория
- Брэнд
- Модель
- Свойства

Шаблоны поиска

<Категория> <Брэнд> <Модель>

<Категория>

<Категория> <Брэнд>

<Категория> <Модель>

<Брэнд> <Модель>

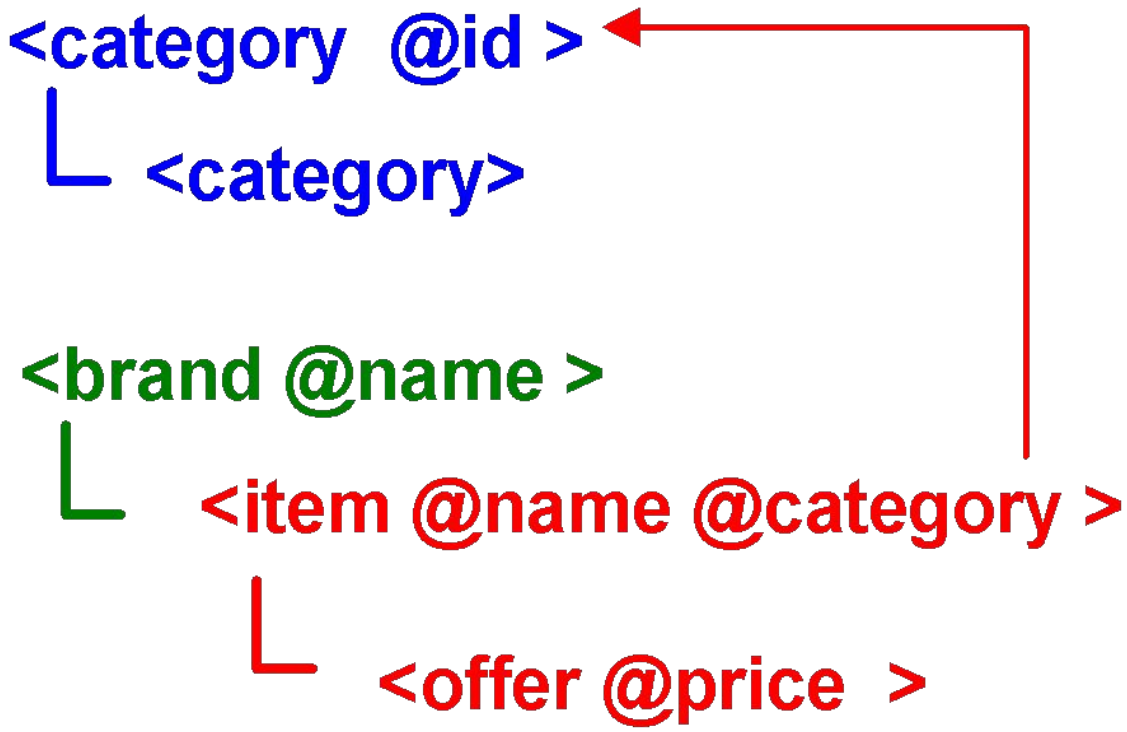
<Модель>

<части модели>

<Модель> <свойство>

<Категория> <свойство>

Структура данных



Запросы

XPath для шаблона <Категория> <Брэнд> <Модель>:
Brand[@name="Canon"]/model[@name="Povershort S80" and @category="**<category_id>**"]/offer

XQuery:

```
let  $model := doc("good")/Brand[@name="Canon"]/model;
for $m in $model
where @name="Povershort S80" and @category="<category_id>"
 {
return
 <offer id="{$m/offer/@id}" shop_id="{$m/offer/@shop_id}">
    {$m/offer/@price}
</offer>
}
```

Запросы (продолжение)

XQuery:

```
let  $cat_id := doc("catalog")/ Category[ @name="фотоаппарат"]/@id;
let  $model := doc("good")/ Brand[@name="Canon"]/model;
for $m in $model
where @name="Povershort S80" and @category=$cat_d
 {
return
 <offer id="{$m/offer/@id}" shop_id="{$m/offer/@shop_id}">
     {$m/offer/@price}
</offer>
}
```

Проблемы

- Составные имена модели
- Морфология
- Ошибки ввода

Используемая литература

http://www.sai.msu.su:7000/database/articles/sedna/index.shtml
XML-СУБД Sedna: технические особенности и варианты использования
Максим Гринев, Сергей Кузнецов, Андрей Фомичев
Статья была опубликована в журнале “Открытые системы”

http://www.w3.org/TR/xquery/
XQuery 1.0 w3c recommendation

http://www.w3.org/TR/xquery-use-cases/
XQuery Use Cases Tutorial

http://www.citforum.ru/internet/articles/xqlzxml.shtml
Основы XQuery на русском

https://msdb.ru/Downloads/Events/Materials/Platform2006/08122005_Blue/db07_MoscowXmlAndXQueryInSqlServer2005Rus.ppt
XML и MSSQL 2005 материалы с Платформы 2006