

Казанский национальный исследовательский технический университет
им. А.Н.Туполева (КНИТУ-КАИ)

**МЕТОДЫ И АЛГОРИТМЫ ОРГАНИЗАЦИИ
АССОЦИАТИВНО-ПОИСКОВОЙ СИСТЕМЫ АНАЛИЗА
ПОЛНООБЪЕКТНЫХ КАРТОГРАФИЧЕСКИХ СЦЕН НА
ПЛАТФОРМЕ ВЫЧИСЛИТЕЛЬНЫХ КЛАСТЕРОВ**

05.13.17 – Теоретические основы информатики
Диссертация на соискание ученой степени кандидата
физико-математических наук

Сергей Васильевич Пыстогов

Научный руководитель:
Райхлин Вадим Абрамович,
доктор физ.-мат. наук, профессор

Челябинск 2017

АКТУАЛЬНОСТЬ ТЕМЫ

Картография.

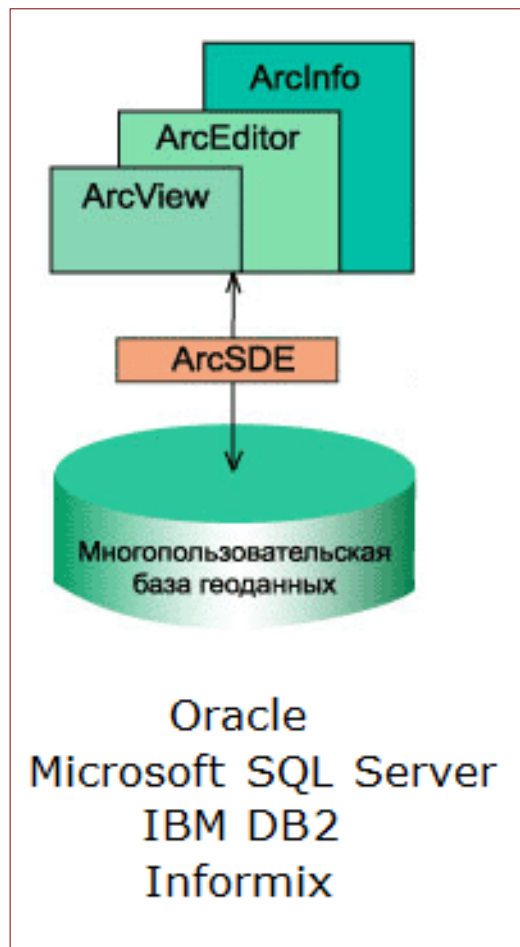
Средства анализа пространственных данных и ГИС вышли за рамки использования военными или промышленными ведомствами и широко применяются в повседневной деятельности. (Google Maps, 2Gis). Особенностью ГИС является возможность разделения информации об объектах на *тематические слои* в соответствии с той или иной предметной областью. Тематические слои ГИС могут содержать конфиденциальные сведения, требующие защиты.

Высокопроизводительные вычисления.

Основной тенденцией последнего времени в сфере обработки данных является использование высокопроизводительных параллельных СУБД. Рост объемов БД КС приводит к необходимости организации *параллельной обработки*.

Известные параллельные СУБД с поддержкой пространственных данных

ArcSDE



PostgreSQL – СУБД, поддерживающая *пространственные данные* и имеющая встроенные элементы защиты.

ArcSDE – серверное ПО для организации хранения и управления *пространственными данными*.

Имеется ряд исследовательских и коммерческих проектов параллельных СУБД:

DB2 Parallel Edition, MySQL Cluster, Oracle EXADATA, PG Cluster.

Большинство из них являются *универсальными* и характеризуются высокой стоимостью.

Повышение эффективности управления объемами БД КС с элементами маскирования данных может быть достигнуто построением специализированных СУБД.

ЦЕЛЬ ДИССЕРТАЦИОННОЙ РАБОТЫ

Выработка практических рекомендаций по построению высокоэффективных ассоциативно-поисковых систем анализа полнообъектных картографических сцен на платформе вычислительных кластеров

1. ФОРМУЛИРОВКА ЗАДАЧИ

Разработка методов и алгоритмов организации систем управления базами данных полнообъектных картографических сцен с ассоциативной защитой.

Вопросы теории ассоциативной стеганографии, в частности, связанные с удовлетворением требований конфиденциальности и помехозащищенности, в диссертации не рассматриваются. Эти вопросы детально исследованы в монографии* и на них получены положительные ответы.

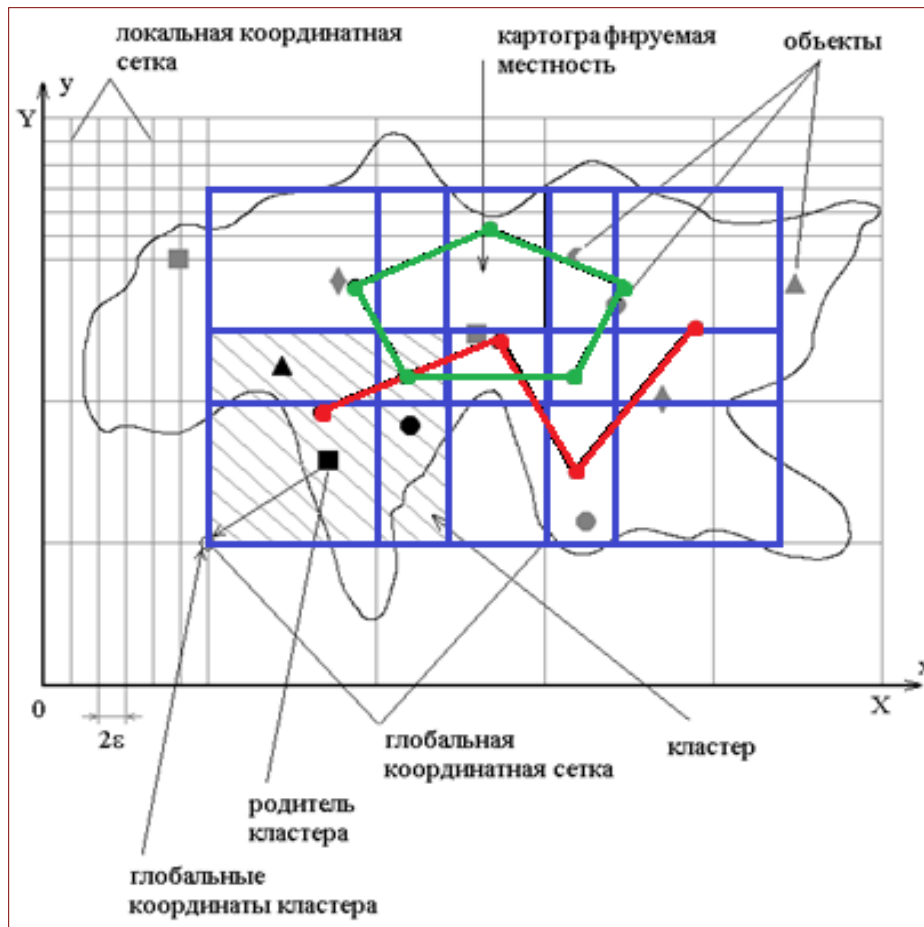
С решением сформулированной задачи связывается разработка и исследование близких к натурным моделей серверной и клиентской частей ассоциативно-поисковой программной системы анализа замаскированных картографических сцен и их тестирование в различных режимах с целью получения действенных практических рекомендаций по построению подобных систем.

* Райхлин, В.А. Конструктивное моделирование систем информатики /В.А.Райхлин, И.С. Вершинин, Р.Ш. Минязев, Р.Ф. Гибадуллин /Под ред. В.А. Райхлина – Казань: Изд-во «ФЭН» («Наука»), 2016. – 312 с. ISBN 978-5-9690-0303-3

2. КОНСТРУКТИВНЫЕ МОМЕНТЫ АССОЦИАТИВНОЙ СТЕГАНОГРАФИИ В ПРИЛОЖЕНИИ К АНАЛИЗУ СЦЕН

Используемое представление сцены*

Исходная информация : **ИМЯ ОБЪЕКТА_КООРДИНАТА x_КООРДИНАТА y**



- $X = Y = A$
- $\Gamma_{x,y}$ – число градаций.
- $\Gamma_{x,y} \geq \sqrt{A/(2\epsilon)}$
- $C = (2\epsilon) \Gamma_{x,y}$ – линейные размеры клубстера
- ϵ – погрешность определения координат объектов (в метрах).

*Вершинин И.С., Гибадуллин Р.Ф., Пыстогов С.В., Райхлин В.А. Ассоциативная стеганография (Приложение к анализу сцен) /Под ред. В.А. Райхлина. – Казань: Изд-во Казан. ун-та, 2014. – 132 с.

- Используется **кодирование** информации о типах объектов и их **координатах** трехразрядными числами – от 000 до 999.
- При генерации БД КС выполняются операции по созданию **тематического слоя**. Объекты тематического слоя подвергаются **кластеризации**. **Глобальные координаты** точечных объектов и узловых точек линейных и площадных объектов переводятся в **локальные координаты**.
- В БД **вводятся пустые** (незначащие) **объекты** и узловые точки линейных / площадных объектов. Записи в таблицах **перемешиваются**.
- Используемые десятичные символы стилизуются как **бинарные изображения** в виде двоичных матриц-эталонов фиксированных размеров $m \times n$, $m=2n-1$.

1	1	1	1	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1
0	0	0	1	0
0	0	1	0	0
0	1	0	0	0
1	0	0	0	0

Пример представления символа 9 для $n=5$.

Принятый алгоритм маскирования*

A^t - множество матриц эталонов, M^t - множество матриц масок, X^t - множество троичных эталонов

$$A^t = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}, \overline{M}^t = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \Rightarrow X^t = \begin{bmatrix} - & 0 & 1 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & 1 \\ - & - & 0 & - & 0 & - \end{bmatrix}.$$

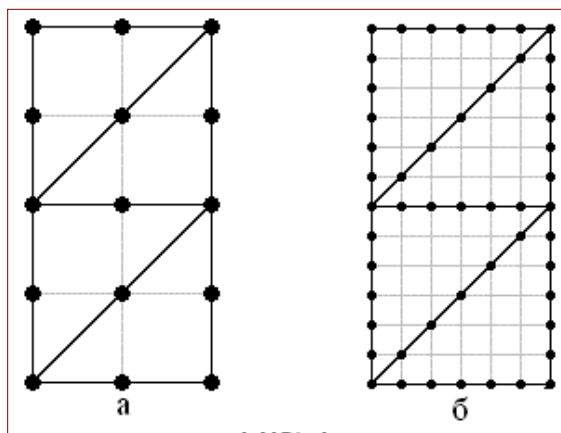
В работе использован базовый алгоритм двумерно-ассоциативного механизма маскирования (АЛГОРИТМ)

Ниже представлен полученный программным путем по АЛГОРИТМу вариант маскирования для одной из исходных перестановок десятичных цифр в случае $m = 5$, $n = 3$. Под каждой цифрой приводится соответствующая инверсная матрица маски.

0	1	9	6	7	8	2	5	4	3
100	100	100	100	100	100	000	000	100	100
000	000	001	000	000	000	000	000	000	001
010	000	000	000	001	010	000	000	001	000
100	000	010	100	010	100	110	110	010	010
010	010	010	010	010	010	010	010	010	010

* К решению задачи маскирования стилизованных двоичных изображений / Райхлин В.А., Вершинин И.С., Глебов Е.Е. // Вестник КГТУ им. А.Н. Туполева. 2001. №1. С. 42–47.

Представление передаваемых по сети стегоданных от сервера к клиенту



Информация по каждой точке – три контейнера :

ИМЯ ОБЪЕКТА, КООРДИНАТА x , КООРДИНАТА y .

Каждый контейнер – из трех секций – по числу разрядов кода. Длина каждой секции **$L=9n-12$** определяется числом **существенных элементов**, расположенных по **совокупному контуру** бинарных матриц используемых символов (точки на рис а, б для случаев $n=3, 7$. Секция заполняется **отрезком ПСП** длиной **L** . Затем в нее **по маске** внедряются **значащие биты** соответствующего символа.

Рекомендуемые размеры бинарной матрицы символов

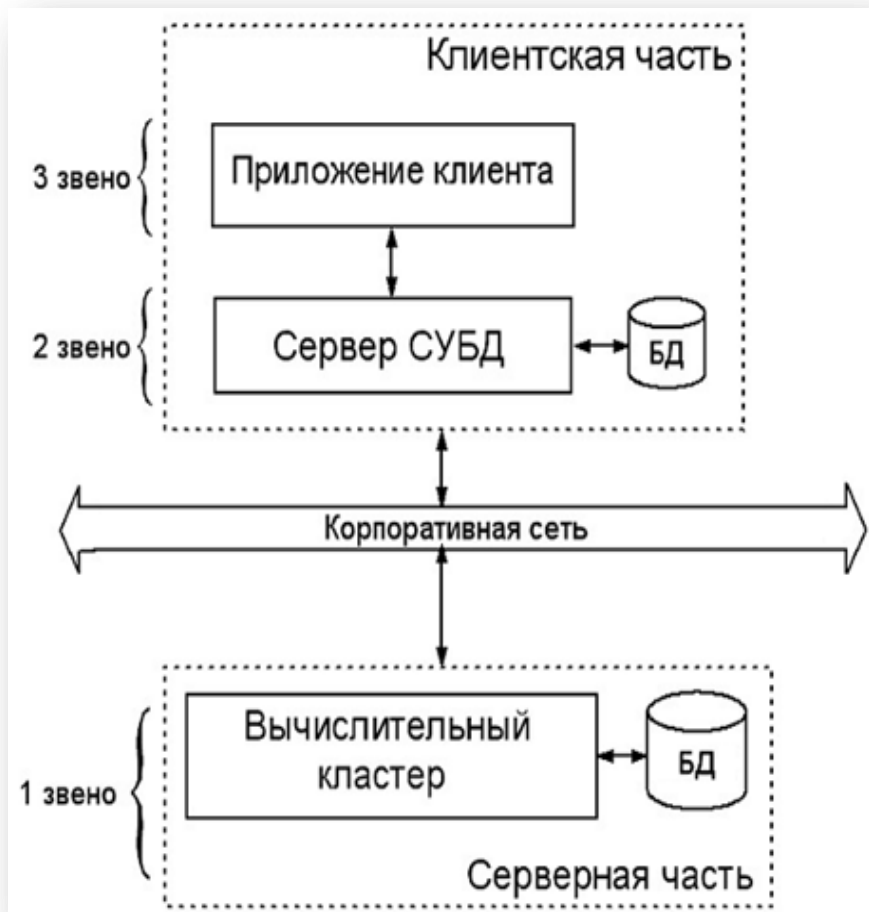
При выборе **$n = 40-60$** и использовании генератора ПСП «Вихрь Мерсенна» подходящая рандомизация (заполнение **несущественных бит**) находится с первой попытки [**Р.Ф. Гибадуллин**], а несанкционированное (без знания маски) распознавание объектов изображения оказывается практически невозможным –

доказуемая стойкость стеганографии.

Велика и устойчивость к действию случайных и преднамеренных помех при хранении и передаче замаскированных данных [**И.С. Вершинин**].

3. ПРИНЯТЫЕ ОГРАНИЧЕНИЯ

Архитектура системы управления ПБД КС



Задачи клиента:

1. Формирование и отправка запросов на сервер.
2. Обработка полученных результатов и визуализация данных.

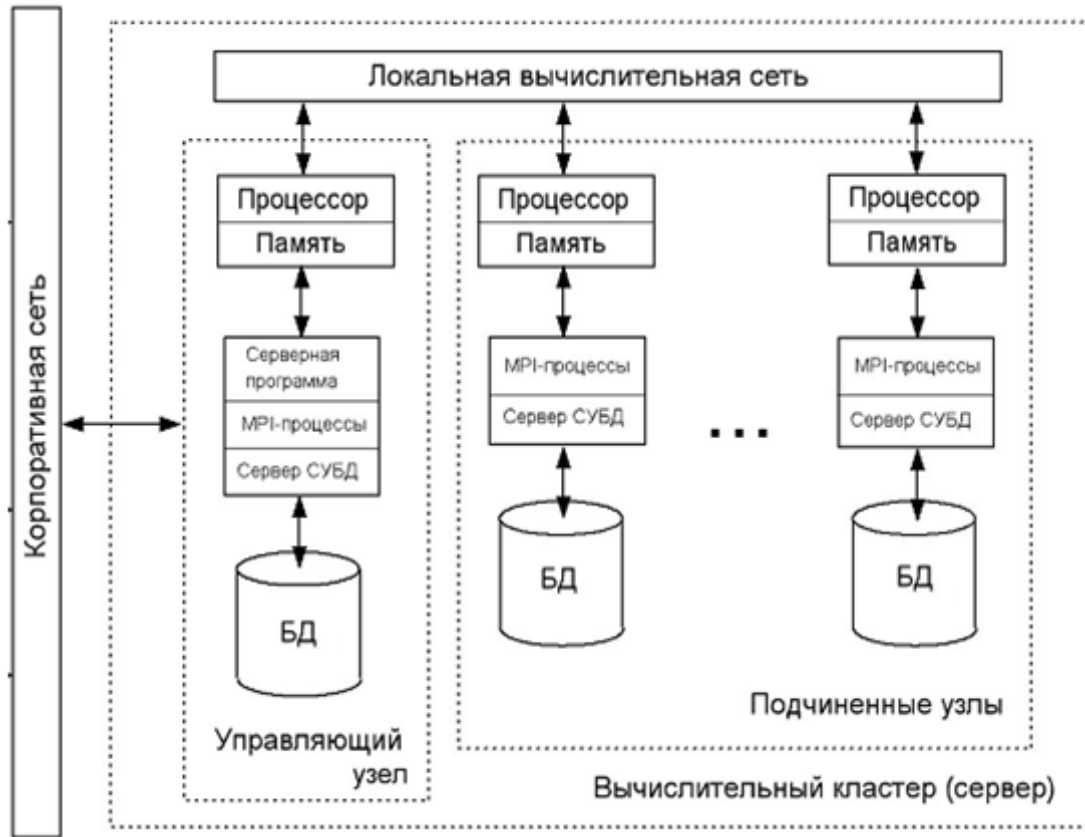
Задачи сервера:

1. Синтез ПБД КС.
2. Распределение по узлам кластера.
3. Прием клиентских запросов.
4. Обработка запроса (в параллельном режиме).
5. Подготовка файлов результата.
6. Отправка результата клиентам.

Задачи вычислительных узлов:

1. Выполнение запросов на узле в параллельном режиме.

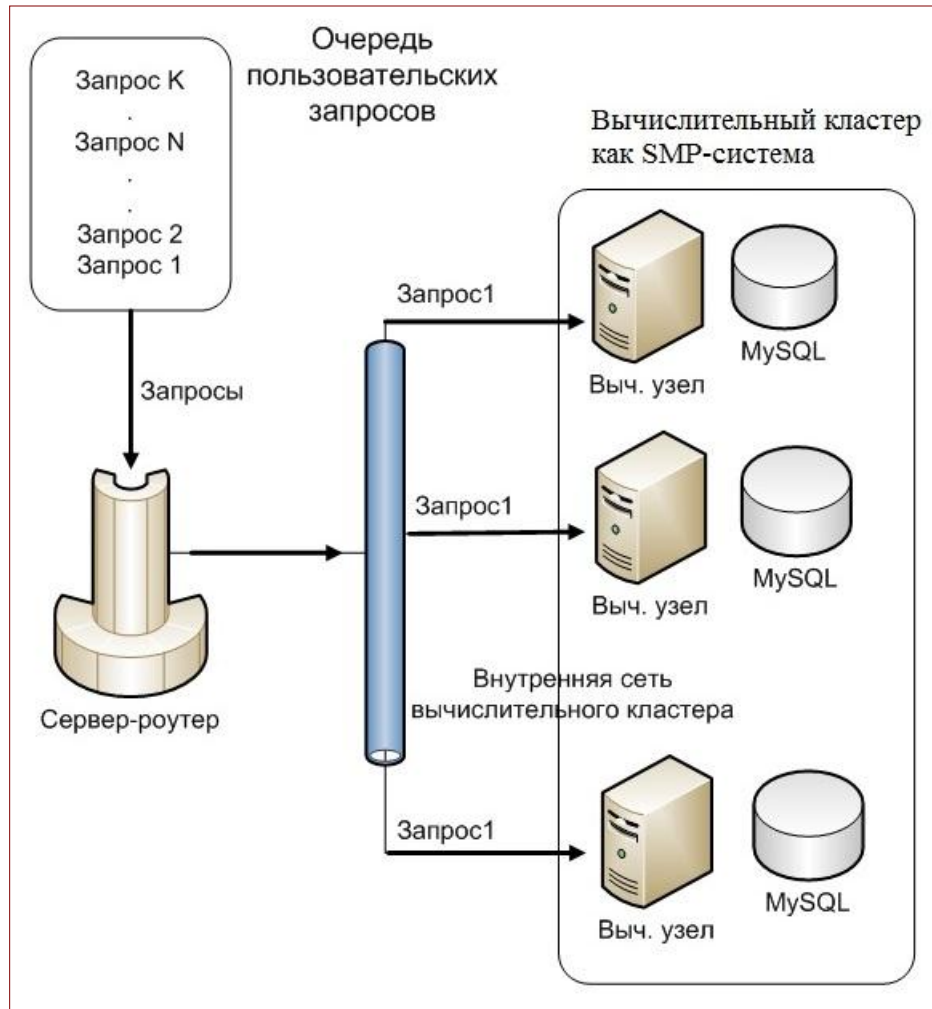
Серверная часть системы



Рассматриваемые варианты сервера

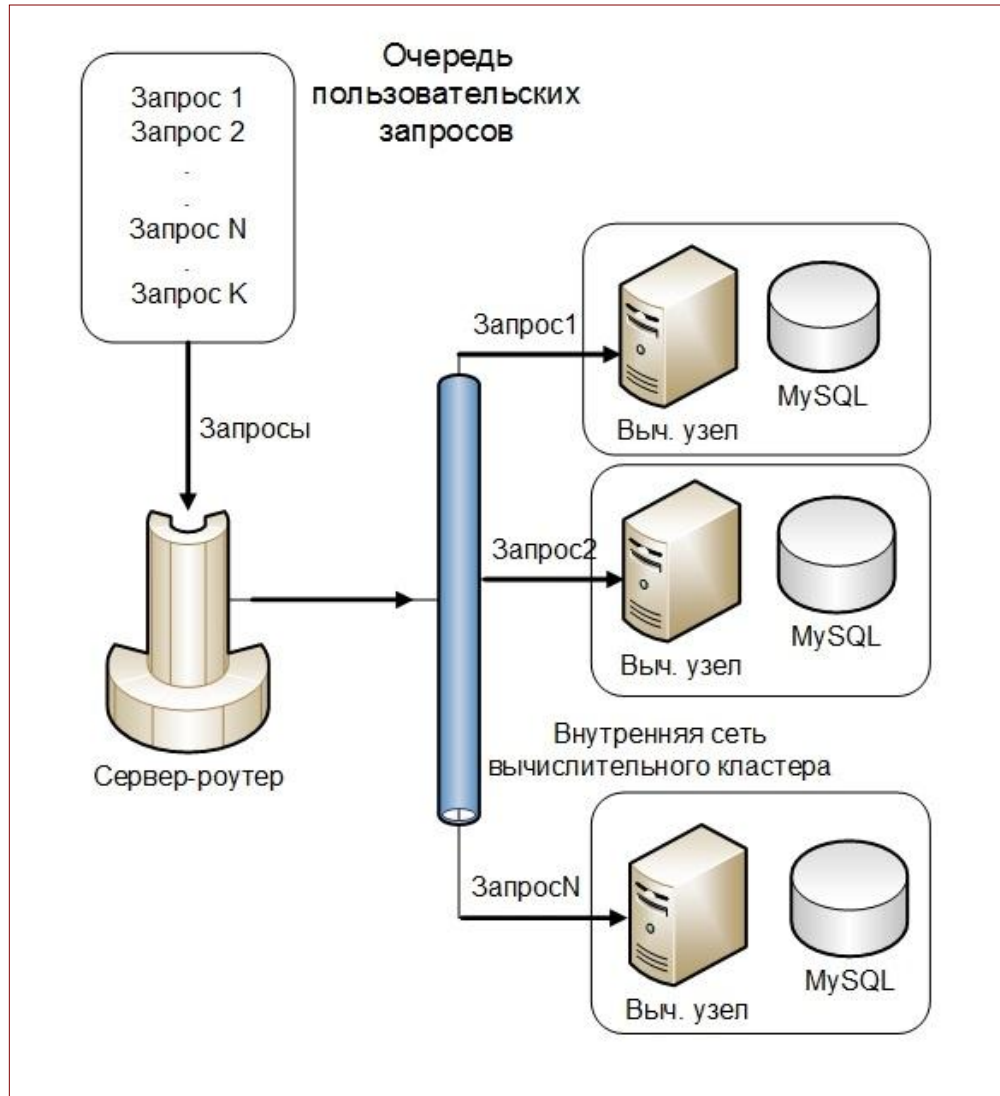
1. **Монокластер** – «весь кластер на запрос». Все вычислительные узлы серверной части в один момент времени обрабатывают только один запрос.
2. **Мультикластер** – «один узел на один запрос». Параллельное выполнение множества клиентских запросов на кластере. Система способна обрабатывать одновременно до N запросов, где N – число вычислительных узлов кластера.

Монокластер



- Все базы данных равномерно распределяются по узлам кластера.
- Каждый вычислительный узел выполняет поиск удовлетворяющих условиям запроса элементов в своей локальной части БД.
- Конечный результат получается путем конкатенации результатов со всех узлов, где были получены промежуточные результаты.
- Серверу нет необходимости выполнять функцию роутера. Запросы обрабатываются в порядке очереди их поступления на сервер.
- Возможна задержка барьерной синхронизации.

Мультикластер



- Репликация баз данных по узлам. Каждый вычислительный узел хранит полную копию всех ПБД КС.
- На управляющий узел возлагаются дополнительные функции распределения запросов между узлами – функция роутера.
- Необходимость контролирования когерентности данных в ПБД КС между узлами при изменении информации.

Аппаратная платформа

Исследования проводились на вычислительном кластере SUN
КНИТУ-КАИ.

20 вычислительных узлов, 160 ядер.

Характеристики узлов системы:

CPU 2 Quad-Core Intel(R) Xeon(R) E(5450) (2*6 MB L2 cache, 3.00
Ghz, 1333 MHz FSB)

32 GB Memory DDR2-667 MHz

Жесткий диск SAS 10000 RPM

Сеть Ethernet 10/100/1000 BASE-T / InfiniBand с пропускной
способностью до 20 Gbit/s)

ОС – Linux SLES.

Подход к сравнительной оценке производительности

Для сравнения по производительности моно- и мультикластерных вариантов в обоих случаях выявлялось суммарное время обработки пакета запросов $T_{\text{общ}}$ и для каждого клиентского запроса – время задержки $T_{\text{зд}}$ от момента его поступления в систему до момента отправки ответа.

$$T_{\text{зд}} = T_{\text{обр}} + T_{\text{ожд}} + T_{\text{пробр}}$$

$T_{\text{обр}}$ – время обработки запроса сервером,

$T_{\text{ожд}}$ – время простоя запроса в очереди запросов сервера,

$T_{\text{пробр}}$ – время предобработки запроса, включающее раскрытие сокрытых в запросе параметров, синтаксический, семантический анализ, права пользователя на исполнение запроса и т.п.

Полученные данные используются для оценки *математического ожидания*

$$M(T_{\text{зд}}) = (\sum T_{\text{зд}}) / n,$$

n – число обработанных запросов,

и *среднеквадратического отклонения*

$$\sigma(T_{\text{зд}}) = \sqrt{M[T_{\text{зд}}^k - M(T_{\text{зд}}^k)]^2}$$

Программные инструментальные средства

СУБД MySQL – основа системы.

ГПСЧ «Вихрь Мерсенна» - при его использовании подходящая рандомизация для стегаконтейнера находится с первой попытки, что существенно ускоряет работу системы при маскировании данных в БД.

Программный модуль маскирования – основан на базовом АЛГОРИТМе *.
(Реализация – Вершинин И.С., Гибадуллин Р.Ф.)

Программный модуль поиска в ПБД КС без раскрытия – позволяет выполнять работу с маскированными данными в БД «на лету». Реализовано с помощью внедрения в MySQL пользовательских функций UDF.
(Идея – Гибадуллин Р.Ф.**, Адаптированная реализация – Пыстогов С.В.)

ГИС MapInfo – инструментальное средство для визуализации и обработки картографических сцен. Используется на стороне клиента.

* К решению задачи маскирования стилизованных двоичных изображений / Райхлин В.А., Вершинин И.С., Глебов Е.Е. // Вестник КГТУ им. А.Н. Туполева. 2001. №1. С. 42–47.

** Обработка зашифрованных данных посредством СУБД MySQL / Гибадуллин Р.Ф., Пыстогов С.В. // Туполевские чтения: Материалы 17-й Междунар. молод. научн. конф. – Казань: КГТУ, 2009. – Т. 4. – С. 60-62.

4. ПОСТАНОВОЧНЫЕ РАЗРАБОТКИ

Полнообъектная база данных картографических сцен

Присутствие на тематическом слое карты **различных типов данных** (точечных, линейных и площадных объектов) и применение к ним **единого механизма защиты** требует разработки структуры **универсальной** полнообъектной БД КС.

База данных картографических сцен со свойствами полноты и маскирования — база данных, описывающая в терминах **«объект-координаты»** точечные, линейные и площадные объекты **векторизованных участков местности**, в которой согласно специфике предметной области конфиденциальная информация об объектах сцены может **храниться в сокрытом виде**.

Предлагаемая схема полнообъектной базы данных картографических сцен

ПБД КС для системы Map Cluster состоит из *вспомогательных* и *рабочих* отношений.

Вспомогательные отношения:



Рабочие отношения:

Описание кластеров в каждом тематическом слое

[Наименование слоя]
Cluster_num *
Global_X *
Global_Y *

Описание точек в каждом кластере

[Наименование кластера]
Obj_code (Point_num) *
Local_X *
Local_Y *

Подход к формированию ПБД КС

- В любом тематическом слое точечных объектов фигурирует *множество типов объектов*.
- Любой *слой линейных/площадных* объектов представляет *единственный* такой объект.
- Порядок записей в рабочих отношениях *случаен*.
- Узловые точки линейного объекта нумеруются от *начала* объекта к его *концу*.
- Узловые точки площадного объекта нумеруются в *порядке его обхода по часовой стрелке*.
- *Пустые точечные объекты* и *пустые узловые точки* каждого отношения имеют номера, *превышающие* число записей.
- Число записей для всех отношений *одинаково*.

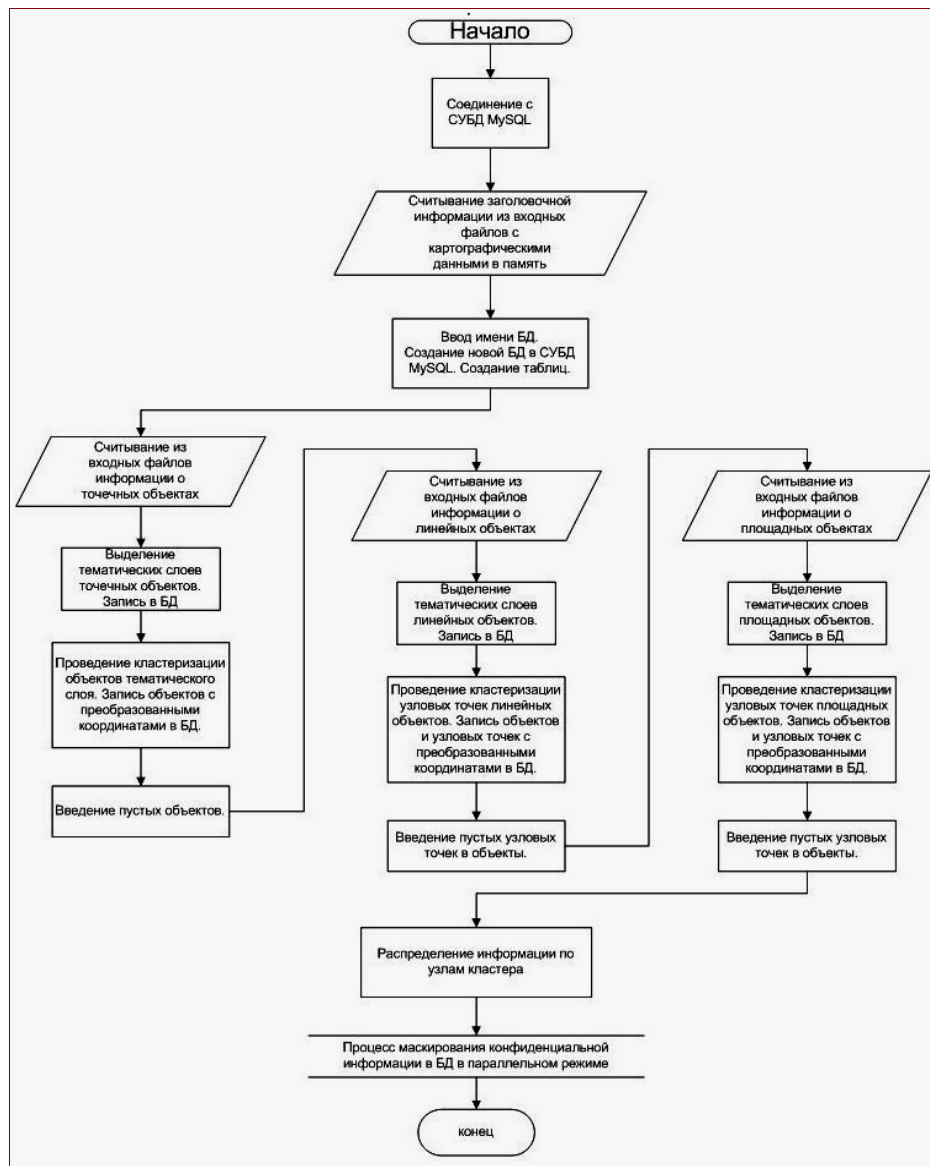
Особенности кластеризации, перемешивания и выравнивания размеров кластеров в случае ПБД КС

Формирование фрагментов (кластеров). Фрагменты сцены формируются случайным образом на основе выбора точечного объекта или узловой точки линейного/площадного объекта, нумерация последовательная. Координаты кластера (левый нижний угол) есть координаты ячейки глобальной координатной сетки, где расположен выбранный объект. Все непокрытые ранее созданными фрагментами точки входят в новый фрагмент. Формирование фрагментов продолжается до покрытия всех объектов сцены.

Добавление пустых точечных объектов и узловых точек. В рамках каждого тематического слоя выполняется выравнивание фрагментов сцены путем добавления во все фрагменты пустых точечных объектов до числа $N_{\max}^{\text{points}}$ – числа реальных объектов в самом насыщенном фрагменте. Коды пустых точечных объектов слоя выбираются случайным образом на множестве кодов объектов соответствующего типа, не задействованных в этом слое. Аналогично для несущественных узловых точек линейных и площадных объектов – их коды (номера) выбираются случайным образом из незадействованных номеров узловых точек объектов соответствующего типа на данном тематическом слое.

Перемешивание. Порядок следования строк в табличных отношениях, описывающих информацию об объектах внутри фрагментов картографической сцены для существенных и несущественных узловых точек и объектов делается случайным.

Алгоритм генерации ПБД КС



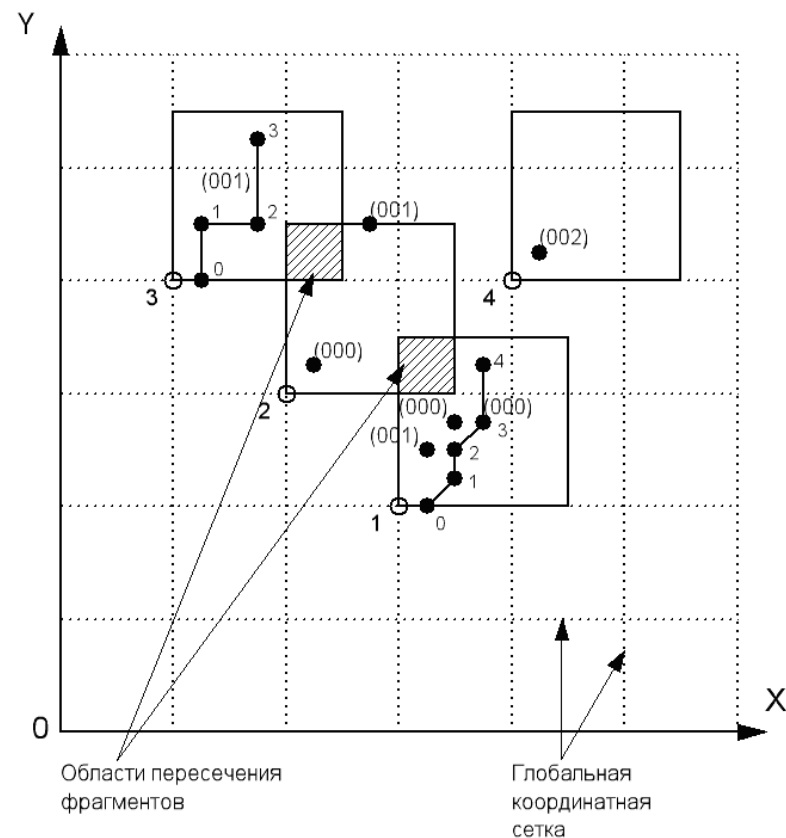
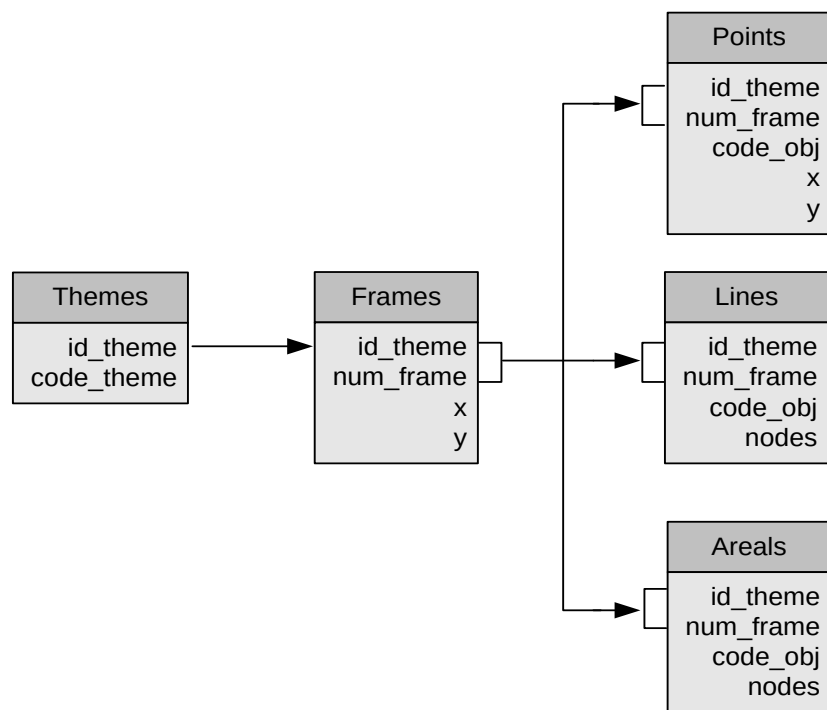
1. В качестве *входных данных* для программы синтеза используются *векторизованные участки картографических сцен*.

2. *Маскирование* данных в полях БД проводится в *параллельном режиме*. Для маскирования применяется разработанная ранее и встроенная в MySQL *библиотека UDF*, что позволяет *сократить время сокрытия* полей в БД. Параллельность маскирования данных обеспечивается применением технологии MPI.

3. Семантическая информация не хранится в БД. Все семантические данные об объектах выносятся в отдельный файл, который хранится на жестком диске. Для указания соответствия в БД сервера вводится таблица *db_semantics* (<имя_бд> <путь_к_файлу>).

Сравнение с ранее предложенной схемой БД *

Схема полнообъектной базы данных и размещение объектов на местности

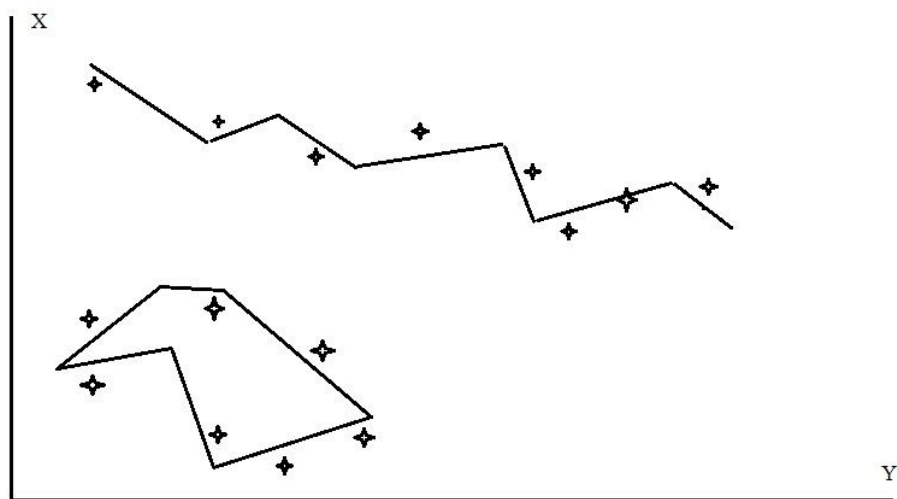


Например, пусть размер участка местности $X = Y = A = 10^6$ м, установленная погрешность $\varepsilon = 1$ м. Тогда линейный размер фрагментов сцены $C=2000$ м.

* - Приводятся результаты, полученные Р.Ф. Гибадуллиным в его кандидатской дис.

В разработанной системе **Map Cluster** нет привязки размеров создаваемых кластеров тематического слоя к размерам линейных или площадных объектов. Поэтому нет ограничений на размеры таких объектов.

Система адаптирована на работу с протяженными линейными и крупномасштабными площадными объектами. Система разработана для возможности работы с протяженными линейными и крупномасштабными площадными объектами: железные дороги, ЛЭП, нефте- и газопроводы, минеральные месторождения и пр.



Погрешность определения координат ϵ при таком расположении объектов крайне важна.

Предложенный ранее подход к построению БД КС не годится для таких ситуаций.

Возможные в системе запросы

Select – простая выборка объектов из картографического слоя. Возможно указание границ по координатам и ограничение по коду объекта.

Delete – удаление объектов из картографического слоя. Также возможно указание границ и кодов объектов.

Insert – добавление объектов в картографический слой.

Update – изменение существующего объекта. Для реализации в системе в чистом виде не рассматривается. В качестве альтернативы предлагается последовательность операций удаления и добавления.

Тестовые ПБД КС. Представительский тест.

Представительский тест (ПТ) для разработанной системы со специфической схемой ПБД КС – это набор (20 шт.) простых **запросов селективного типа**.

Пример запроса:

select * from kbd1.theme_name where X<20000 and Y>5000;

На основе параметров, указанных в этом запросе, формируется перечень подзапросов адаптированных к языку MySQL и схеме ПБД КС, выполняемых последовательно этой СУБД:

A. SELECT Code from Point_Themes where Name= "theme_name";

Результатом данного запроса станет получение кода тематического слоя, в рамках которого ведется поиск.

B. SELECT Cluster_num, Global_X, Global_Y from Theme_N;

Результатом запроса станет список кадров (кластеров) слоя *N* с их глобальными координатами. Здесь *Theme_N* – табличное отношение, описывающее слой *N*, где *N* = Code (результат запроса A).

C. Множество подзапросов

SELECT Obj_code, local_X + Global_X*A, local_Y + Global_Y*A from Theme_N, Cluster_(i) where Cluster_num=i, (local_X + Global_X*A) < 20000 and (local_Y + Global_Y*A)> 5000);

Для проведения экспериментальных исследований на инструментальном средстве с использованием **ПТ** потребовались соответствующие предложенной схеме полнообъектные базы данных.

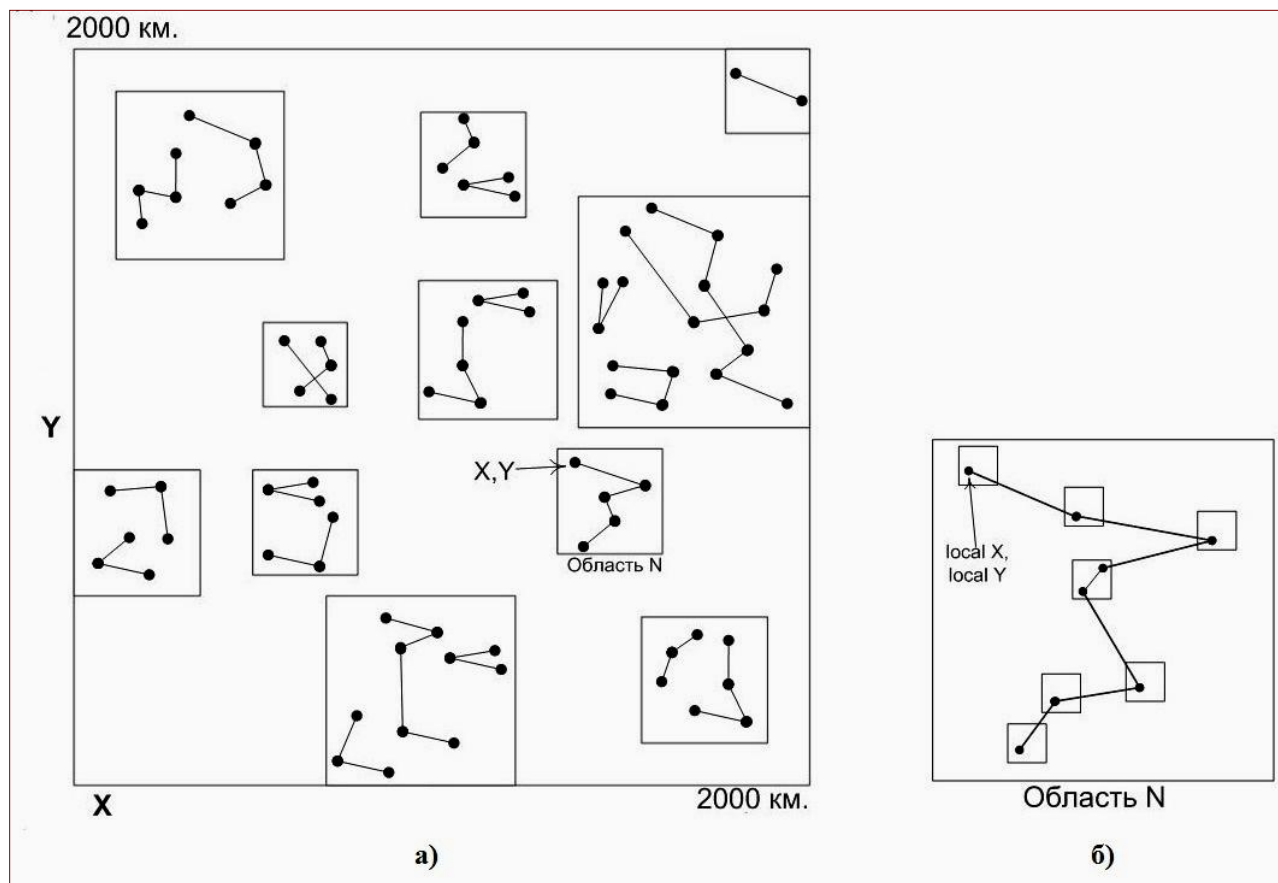
Генерируются по три **ПБД КС** разных размеров для каждого типа объектов (**всего 9**) на области размером 2000 км × 2000 км.

Базы данных генерируются в прямой зависимости от сформированного **ПТ**.

Пропорция объемов баз данных объектов одного типа примерно такова:

$$\mathbf{БД1 : БД2 : БД3 = 1 : 2 : 4.}$$

Метод формирования тестовой картографической сцены.



Объемы полученных тестовых ПБД КС АЗ:

БД1 ≈ 80 Мб.

БД2 ≈ 150 Мб.

БД3 ≈ 300 Мб.

Пусть k – число точечных объектов (узловых точек), выбираемых в результате выполнения селективного запроса m из области N для БД1. Тогда число объектов, выбираемых из области N тем же запросом m для БД2 должно быть примерно $2k$. Для БД3 – $4k$.

- Объекты картографических сцен располагаются на прямоугольной области размерами **2000x2000 км** в декартовой системе координат.
- Присвоение координат **X,Y** для точечных объектов и узловых точек линейных и площадных объектов происходит из диапазона **{0,1999999}** в результате работы ГПСЧ (функция **rand()**).
- Для упрощения генерации было сделано **отступление**. Линейные и площадные объекты могут пересекаться сами с собой. Допускается дублирование узловых точек в рамках одного объекта.
- При формировании отношений из сгенерированного набора данных применялись все основные принципы согласно структуре специализированной **ПБД КС** и подходу к представлению картографической сцены.
- Размеры кластеров – **2000 м**, $\epsilon = 1\text{м}$.

Подход к обработке запросов без раскрытия всей ПБД КС

Для работы с сокрытыми данными объектов «на лету» используется программный модуль с разработанной библиотекой расширения функционала СУБД MySQL. Библиотека содержит две функции для сокрытия и раскрытия данных в табличных полях ПБД КС: *udf_cipher()* и *udf_decipher()*.

Пример запроса, генерируемого системой при работе с ПБД КС. Запрос выполняет поиск объектов с кодами < 20 и считывание информации о них из табличного отношения Cluster_N. **Выбранные данные остаются в сокрытом виде.**

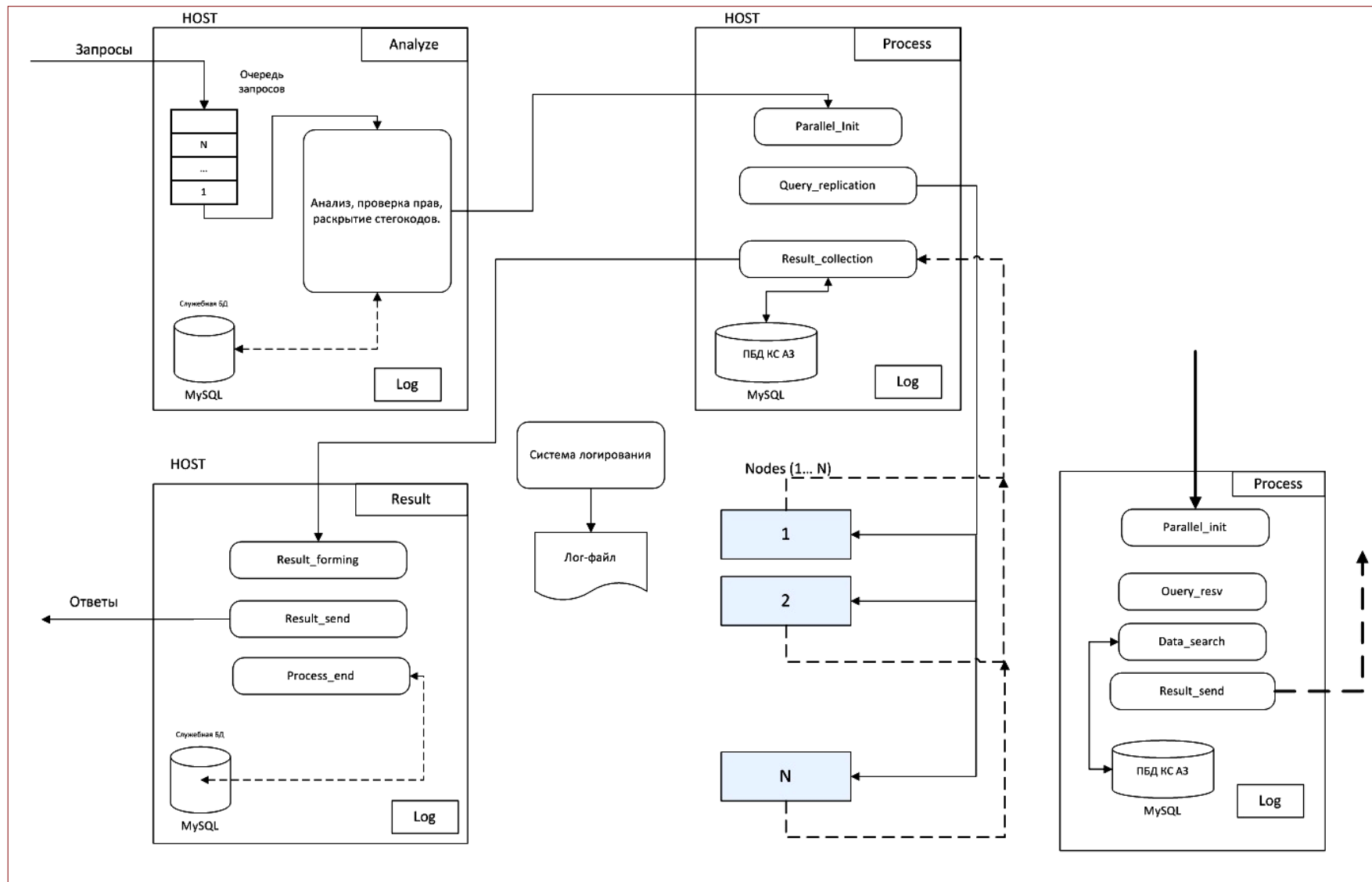
Cluster_N
Obj_code *
local_X *
local_Y *

```
select obj_code, local_X, local_Y from Cluster_N  
where udf_decipher(obj_code) < 20;
```

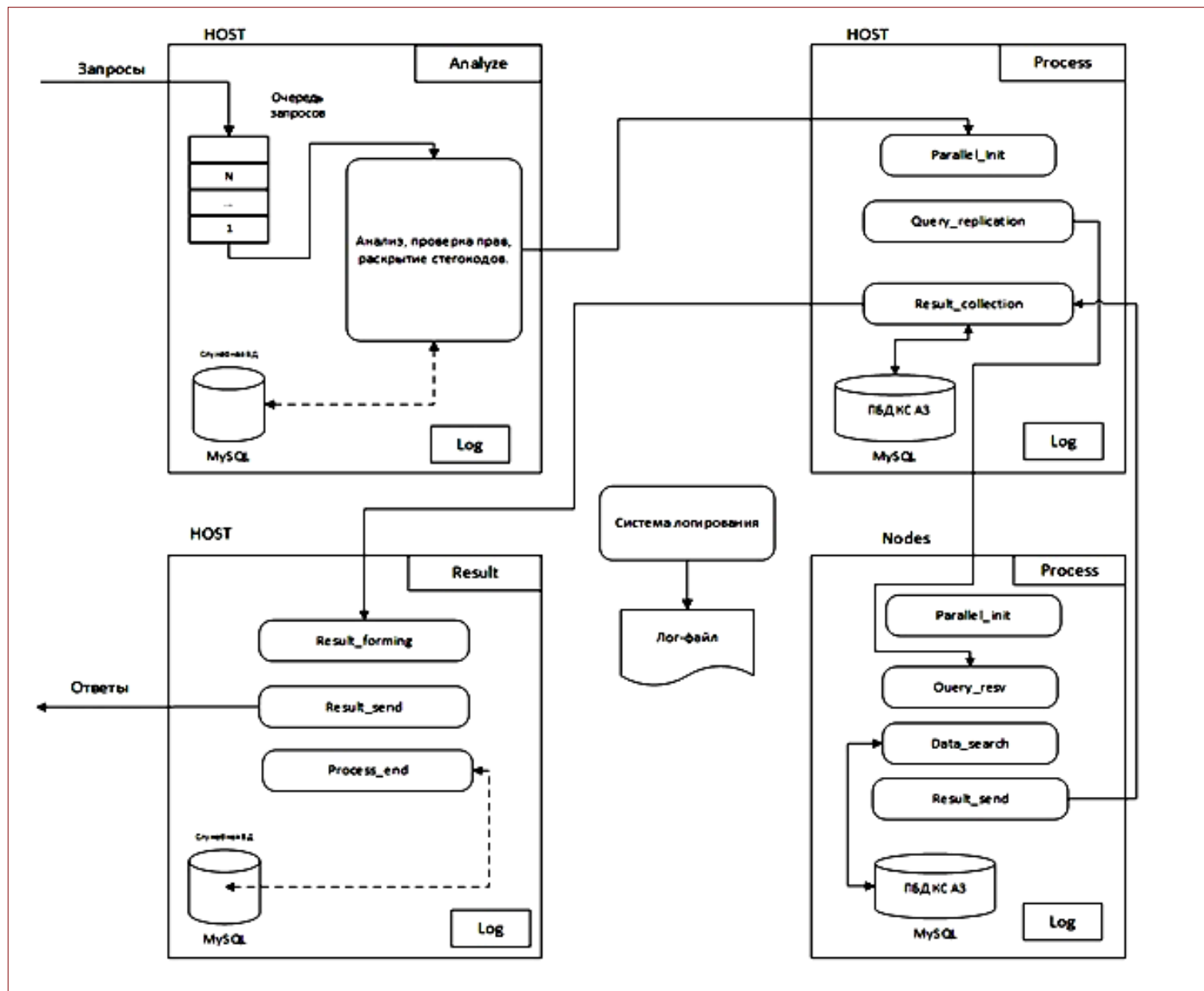
* - маскированные данные

1. ПРОГРАММНАЯ СИСТЕМА СЕРВЕРНОЙ ЧАСТИ

Монокластер. Взаимодействие элементов системы



Мультикластер. Взаимодействие элементов системы

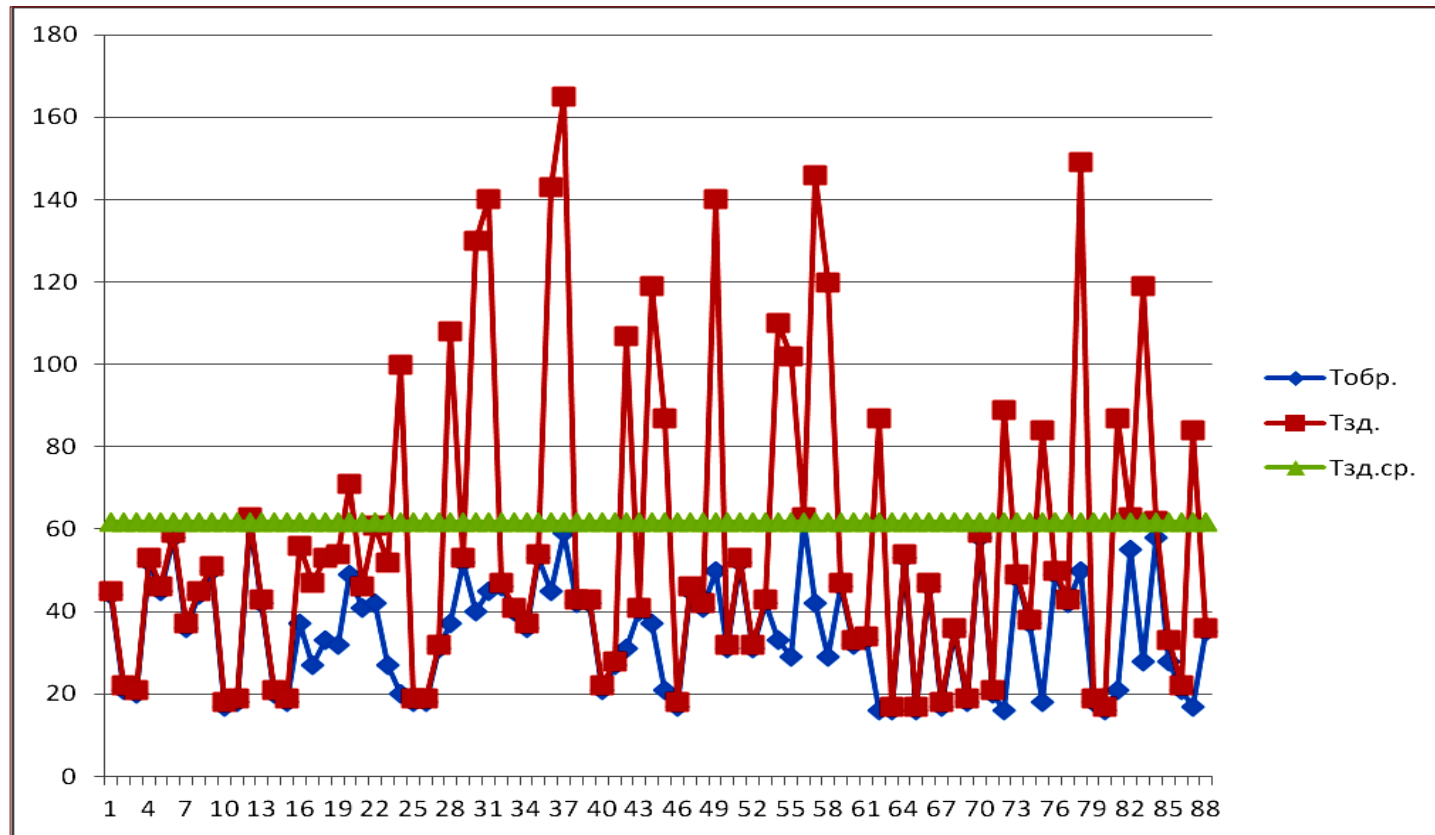


2. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Проведено **три эксперимента** на исследовательском прототипе системы в обеих архитектурах:

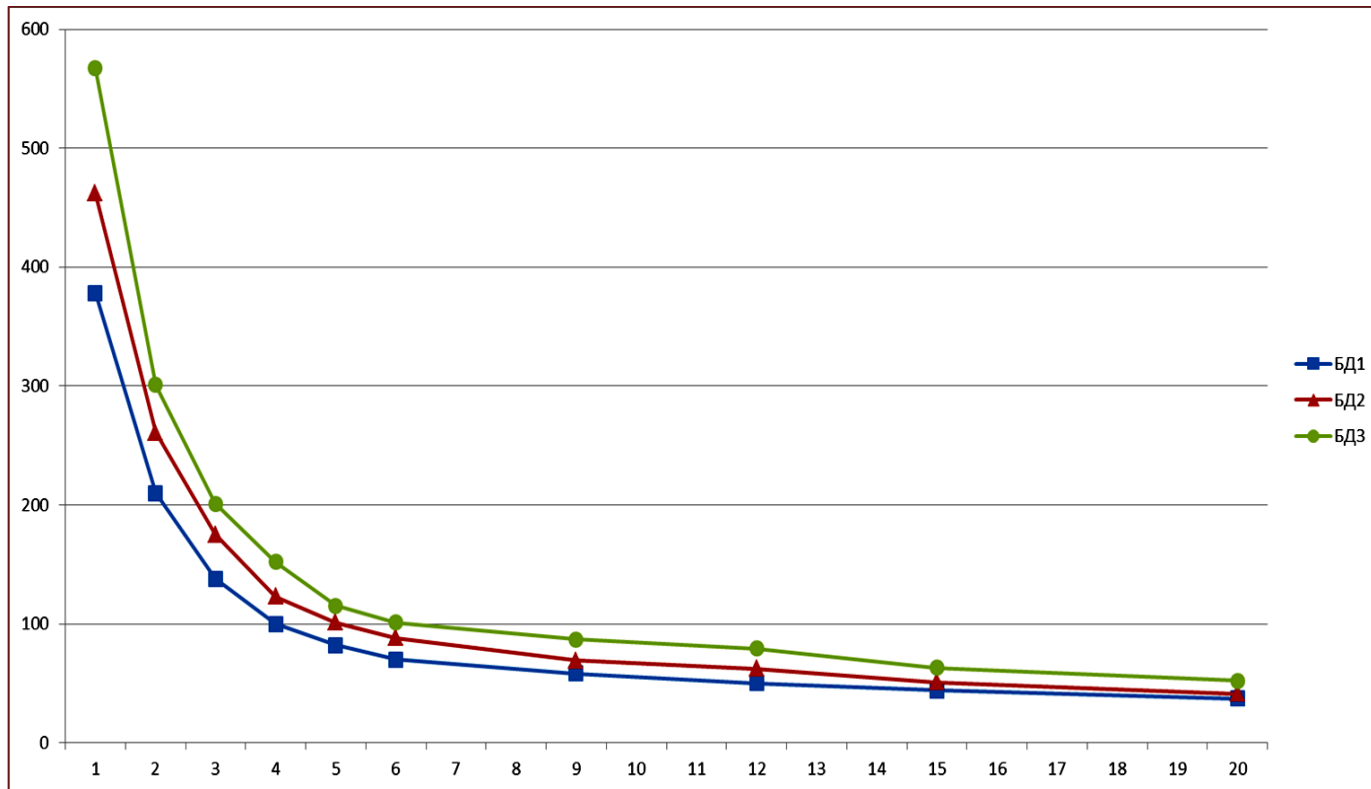
1. Установление границы масштабируемости и характера изменения параметров $M(T_{зд})$ с ростом N для архитектуры *монокластера* на **ПТ** из **20** запросов на всех сгенерированных **ПБД КС** для каждого типа объектов. Оценка значений M проводилась для последней секции потока, полученного конкатенацией трех одинаковых **ПТ**.
2. То же для архитектуры *мультикластера* на тех же **ПТ** и **ПБД КС**.
3. Сравнение производительности и характеристик $M(T_{зд})$ обеих архитектур на непрерывном потоке из **1000** запросов.

Вводимое ограничение



1. Исследование системы в целом было ограничено случаем селективных запросов.
2. Исследование мультикластера ограничивалось вариантом 1-узловых компонентов-монокластеров.

Эксперимент 1 (монокластер)

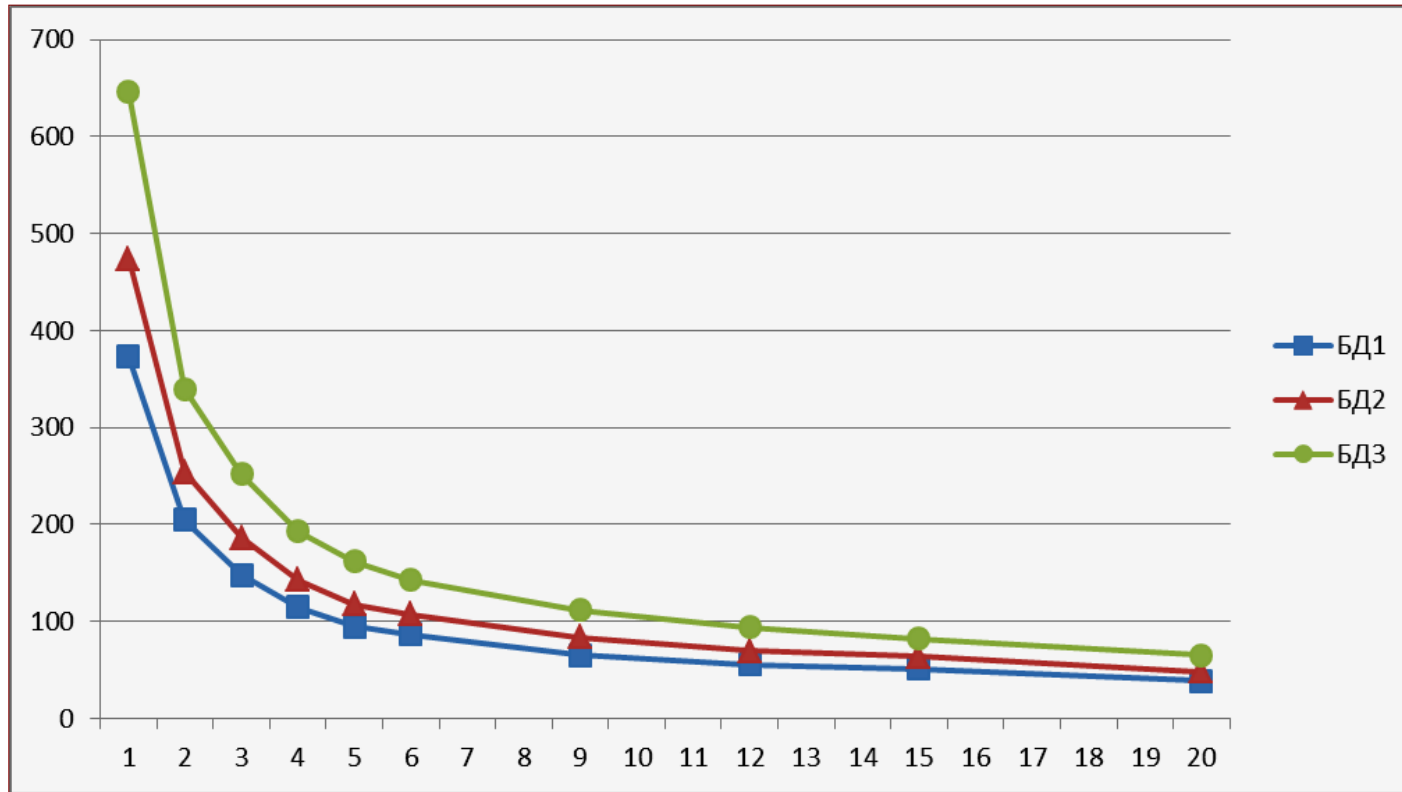


По оси абс-
цисс число
узлов в кла-
стере.

По оси орди-
нат — сум-
марное вре-
мя обработ-
ки пакета в
сек. для трех
БД.

Приведенные графики имеют гиперболический характер. Они отражают изменение производительности системы для случая точечных объектов. В случаях линейных и площадных объектов характер графиков аналогичен.

Эксперимент 2 (мультикластер)



По оси абс-
цисс число
узлов в кла-
стере.

По оси орди-
нат — суммар-
ное время об-
работки паке-
та в сек. для
трех БД.

По той же причине, что и ранее, приведены графики только для случая точечных объектов. Для сравнительно малого числа узлов мультикластер несколько уступает по производительности монокластеру.

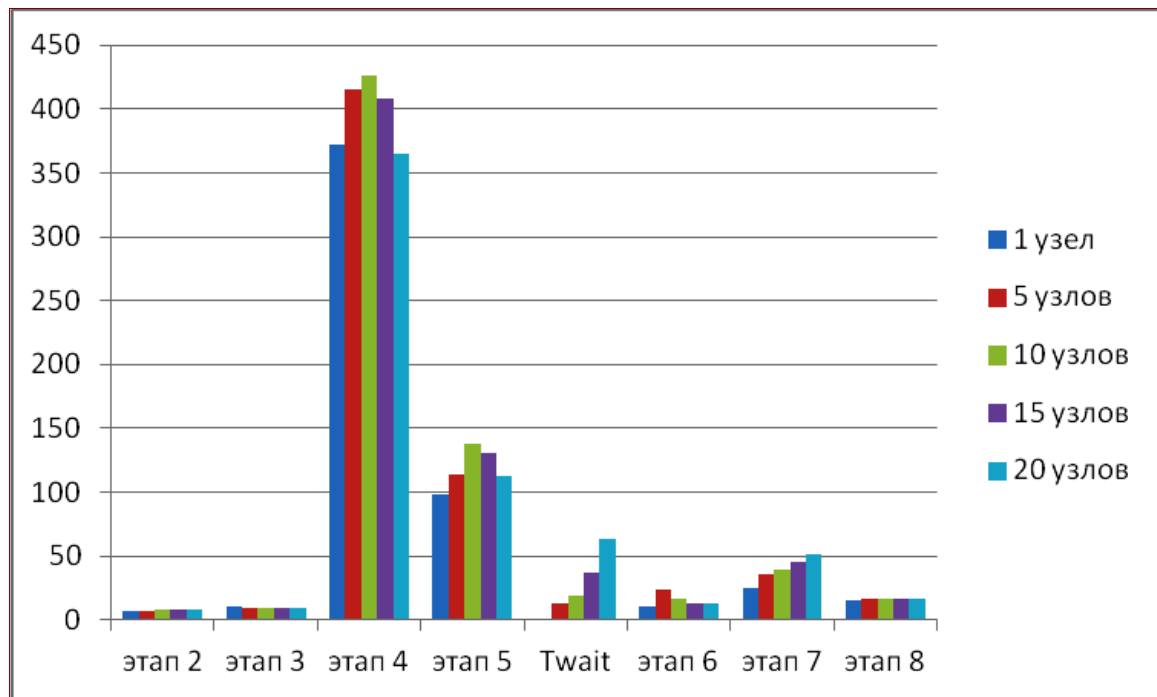
Эксперимент 3

Обработка потока из 1000 запросов, случайно сформированного на множестве 80 запросов не обязательно отвечающих условию попадания в области с объектами (более реальный сценарий).

	Среднее время задержки $M(T_{зд})$, сек	Время обработки пакета $T_{общ}$, сек
Монокластер, 4 узла	122	10610
Мультикластер, 4 узла	76	8172
Монокластер, 9 узлов	55	4875
Мультикластер, 9 узлов	37	4152

При числе узлов $N=4$ мультикластер имеет существенный выигрыш по среднему времени задержки запроса $M(T_{зд})$, суммарное время обработки пакета лучше примерно на 23%. При $N=9$ мультикластерная архитектура по-прежнему лучше по параметрам $T_{общ}$ и $M(T_{зд})$. Однако улучшение параметра $T_{общ}$ относительно монокластера несколько ниже, чем для 4 узлов – порядка 17%.

Эксперимент 4



1. Считывание запроса из очереди, обработка скрытых параметров (Этап 2).
2. Тиражирование запроса по вычислит. узлам (Этап 3).
3. Поиск объектов в БД на вычислительных узлах согласно запросу (Этап 4).
4. Формирование промежуточного результата на вычислителях для отправки на управляющий узел (Этап 5).
5. Сбор результатов с вычисл. узлов (Этап 6).

6. Формирование конечного результата (Этап 7).

7. Отправка результатов пользователю (Этап 8).

Для этапов 2-3 и 6-8 выполнялось суммирование времени по следующей формуле:

$$T_i = \sum t_{ij}, j = 1 \dots 20,$$

где t_{ij} – время выполнения, i – номер этапа, j – номер запроса из пакета.

Для этапов 4-5 и T_{wait} суммирование выполнялось по каждому этапу и по всем узлам по формуле:

$$T_i = \sum t_{ij}^k, j = 1 \dots 20, k = 1 \dots N,$$

где t_{ij}^k – время выполнения, i – номер этапа, j – номер запроса из пакета, k – номер исполнительного узла, n – число исполнительных узлов в системе.

3. ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТОВ

1. Независимо от типа объектов, характер изменения производительности системы с ростом числа узлов кластерной платформы неизменен.
2. Обе архитектуры серверной части показали близкие параметры коэффициента ускорения с ростом числа узлов (не хуже 0,5).
3. Большее $M(T_{зд.})$ и $T_{общ}$ для архитектуры монокластер по сравнению с мультикластерной архитектурой на *непрерывном потоке запросов* (1000 шт.) показывает преимущество мультикластера в данном случае.
4. Монокластерная архитектура может эффективно использоваться в случаях:
 - обработки редко поступающих одиночных запросов на выборку больших областей с множеством объектов;
 - частого появления запросов изменения;
 - непрерывного потока запросов при отсутствии жестких требований ко времени ожидания ответа на запрос;
 - генерации новой ПБД КС или изменения защитного ключа в системе

Таким образом, *архитектура монокластера предпочтительна для практического применения.*

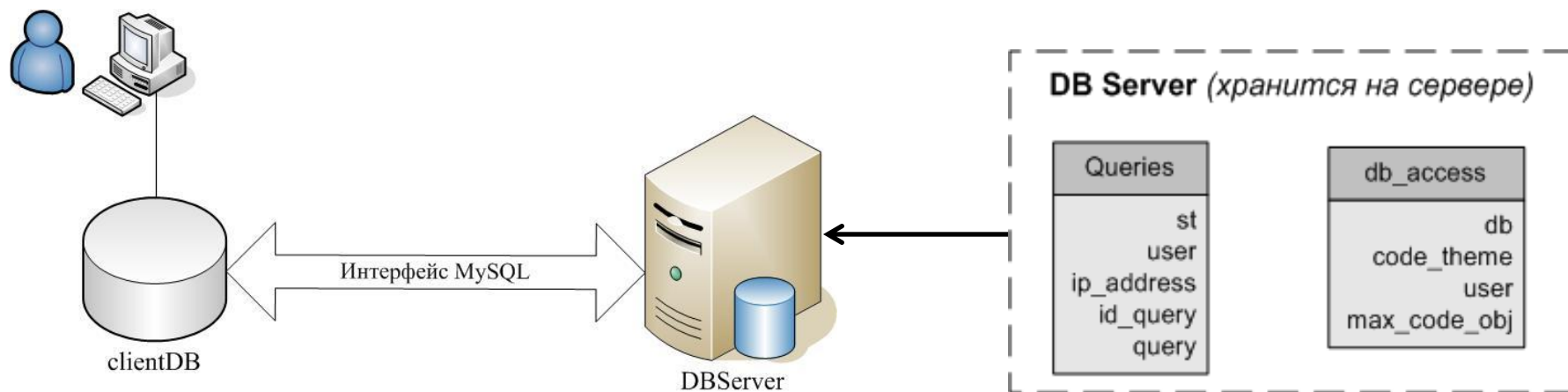
Взаимодействие клиента с сервером

Взаимодействие клиента с сервером происходит посредством интерфейса MySQL. Запросы от пользователя заносятся непосредственно в таблицы базы данных удаленного сервера.

Результаты обработки запроса также посылаются клиенту через интерфейс СУБД MySQL и записываются в связанные таблицы Themes-Frames-Objects в сокрытом виде.

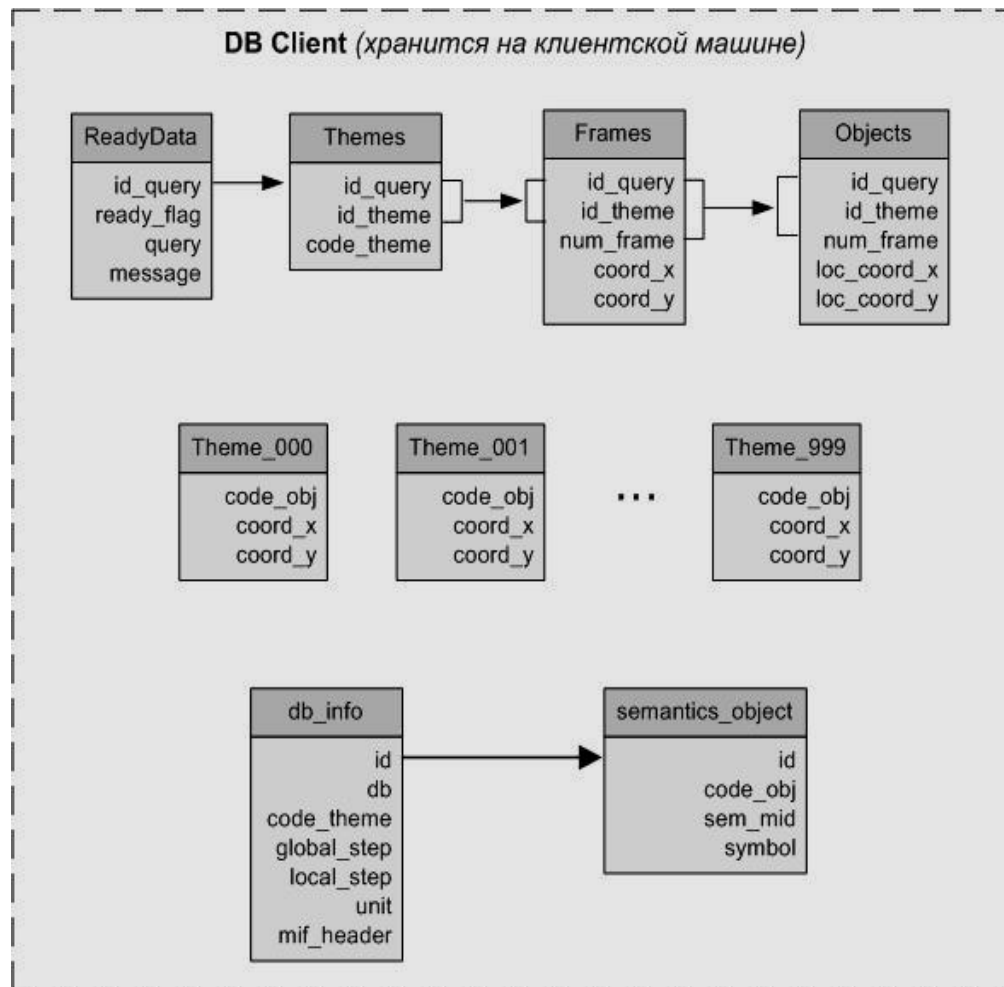
Для взаимодействия с удаленным сервером MySQL каждый пользователь обязательно должен обладать парой индивидуальных атрибутов:

- имя пользователя (**username**)
- пароль (**password**)



Структура БД на стороне клиента

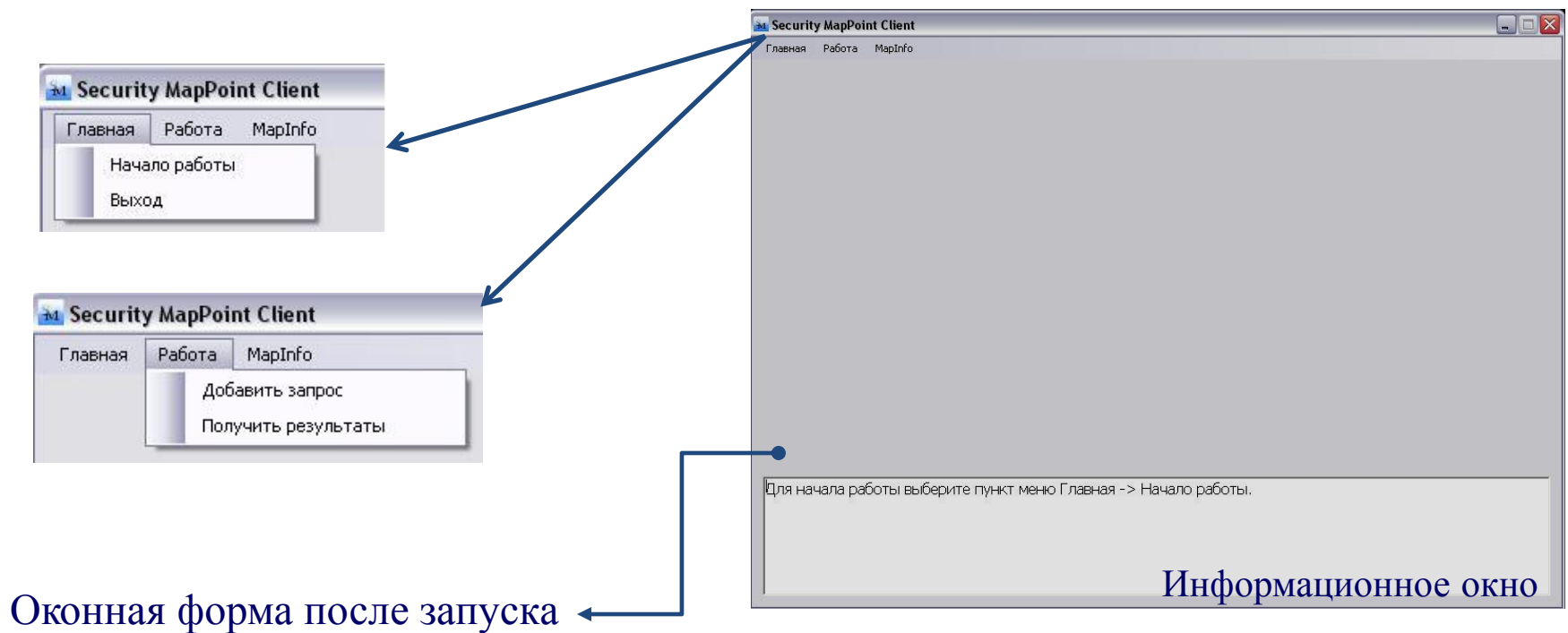
- Таблицы **Themes – Frames – Objects** хранят результаты запросов в сокрытом виде.
- Таблица **ReadyData** содержит запись о каждом введенном пользователем запросе.
- Таблицы **Theme_000... Theme_999** служат для **синтаксического анализа** вводимых запросов.
- Пара таблиц **db_info** и **semantics_object** используются при формировании открытых результатов.



Программа пользователя

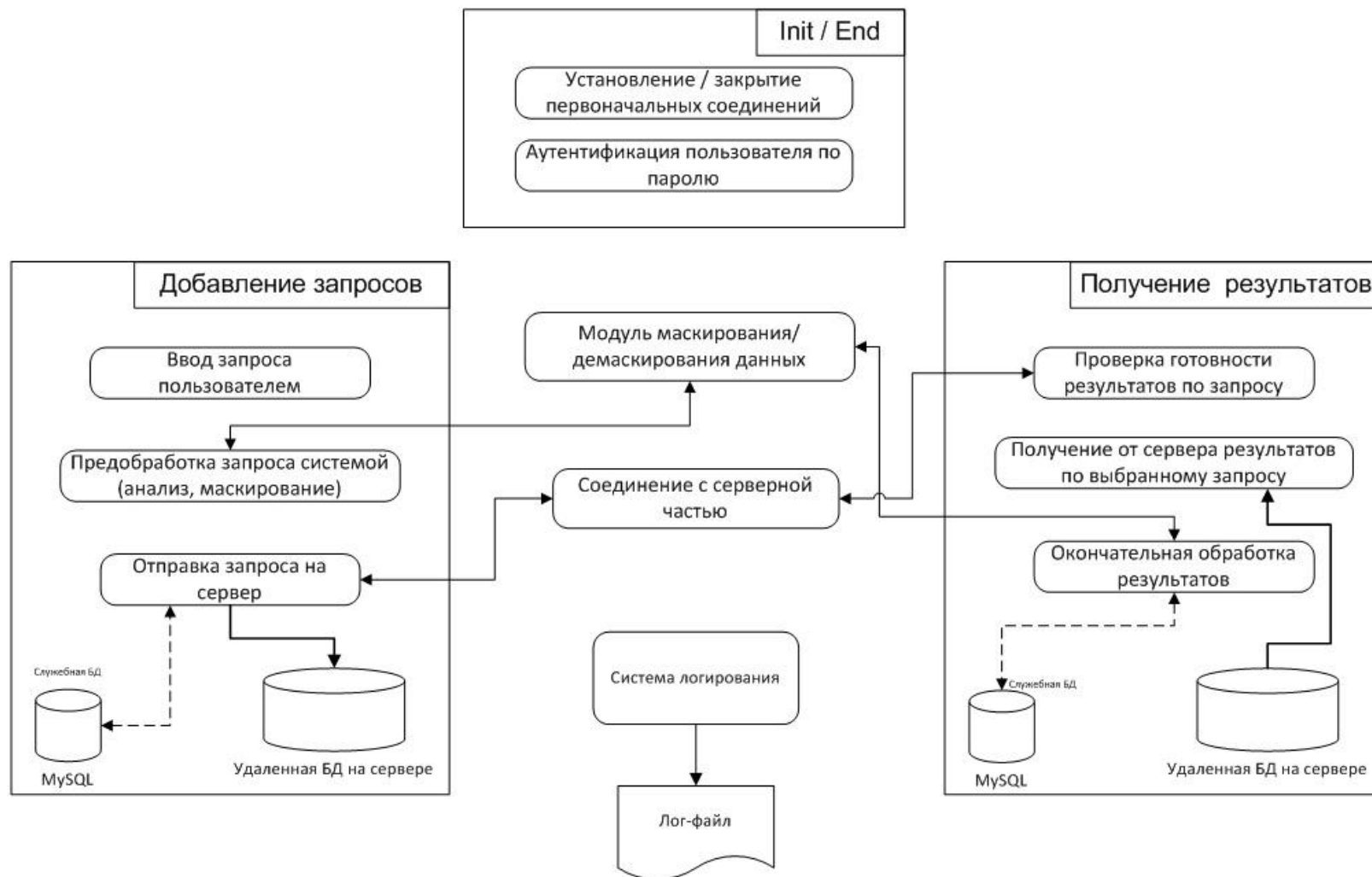
Программа пользователя (клиентская программа) является одной из важных частей системы Map Cluster, выполняя большую часть ее **клиент-серверных** функций. Данная программа располагается на каждом клиентском узле.

Разработка велась с применением современных языков программирования C++/C#. Для взаимодействия с серверами баз данных использовался программный интерфейс **MySQL C API**.



Программа пользователя.

Взаимодействие основных модулей.



Файлы с результатами

Запрос: select * from points.theme_002;

Результат: файлы points.theme_002.MIF, points.theme_002.MID

MIF

MID

Point 1293789.182 373933.2126

Symbol (33,0,14,"MapInfo Symbols",0,0) → "89700014","Сад фруктовый"

Point 1293839.182 373933.2126

Symbol (33,0,14,"MapInfo Symbols",0,0) → "89700014","Сад фруктовый"

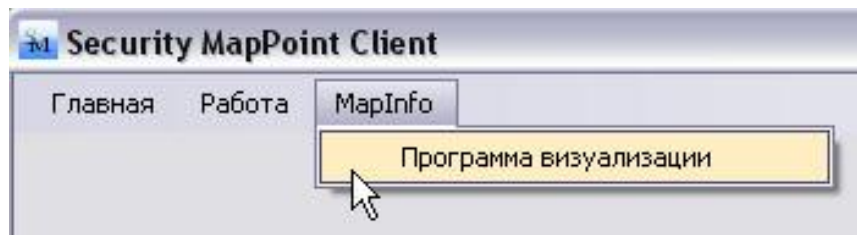
Point 1293789.182 373943.2127

Symbol (30,0,12,"MapInfo Symbols",0,0) → "89700014","Сад фруктовый"

Point 1293799.182 373943.2127

Symbol (30,0,12,"MapInfo Symbols",0,0) → "89700014","Лес естественный
высокоствольный"

Для визуализации полученных данных необходимо запустить ГИС MapInfo из программы пользователя.

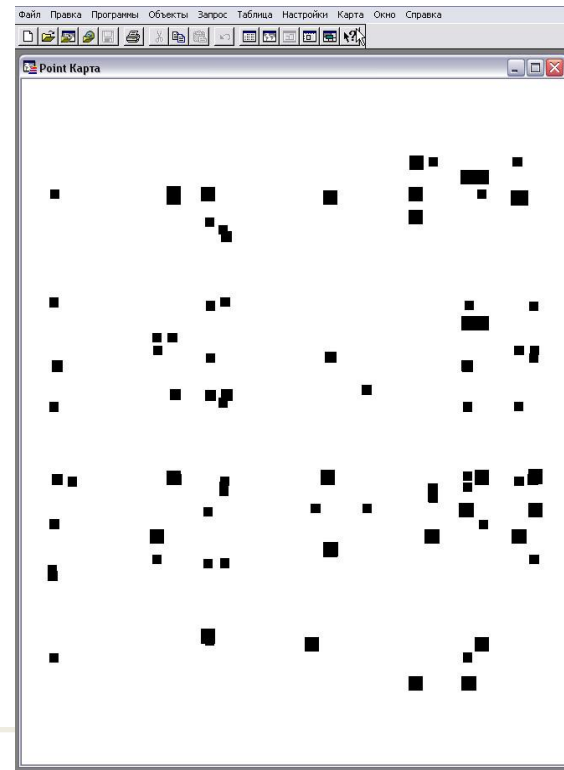


Визуализация данных

Визуализация точечных картографических объектов выполнялась в ГИС **MapInfo**. Визуализируется информация, содержащаяся в файлах импорта MID/MIF. Данные файлы формируются в результате работы программы пользователя.



Визуализация данных,
раскрытых на истинном ключе



Визуализация данных,
раскрытых на ложном ключе

ЗАКЛЮЧЕНИЕ

Основные результаты

1. Разработана схема специализированной полнообъектной базы данных картографических сцен. Предложенная схема ПБД КС, в отличие от предыдущего прототипа, позволяет обойти ограничение на размеры линейных и площадных объектов. Погрешность, завязанная с размером фрагмента картографической сцены, может быть минимальной. Разработаны методы, алгоритмы и программы формирования таких БД на вычислительном кластере на основе векторизованных данных.
2. Предложен подход к генерации тестовых ПБД КС для целей исследования. Метод генерации тестовых баз данных и релевантного им представительского набора запросов для специализированной СУБД позволяет проводить экспериментальные исследования подобной системы.
3. Разработаны методы, алгоритмы и программы организации серверной части ассоциативно-поисковой системы анализа ПБД КС.
4. Разработаны методы и алгоритмы организации клиентской части ассоциативно-поисковой системы анализа ПБД КС для случая точечных объектов.

5. Разработан исследовательский прототип серверной части ассоциативно-поисковой системы анализа ПБД КС на кластерной платформе как инструмента проведенных исследований.

На основе проведенных на разработанном исследовательском прототипе экспериментов было показано:

- обе архитектуры обладают приемлемым коэффициентом расширяемости;
- монокластерная архитектура, несмотря на присущие таким системам недостатки, имеет преимущество над мультикластером при работе с коротким потоком представительских запросов;
- мультикластерный подход предпочтителен по среднему времени ожидания ответа на запрос и по суммарному времени обработки пакета при работе с достаточно большим непрерывным потоком селективных запросов различной сложности;
- при практическом применении, с учетом необходимости генерации ПБД КС, предпочтение следует отдать монокластерной архитектуре.

Научная новизна

- Схема полнообъектной базы данных картографических сцен с ассоциативной защитой.
- Метод генерации тестовых баз данных картографических сцен для проведения исследований.
- Методы обработки запросов в серверной части без раскрытия всей ПБД КС.
- Методы организации серверной части СУ ПБД КС с ассоциативной защитой.

Научная значимость

- Разработка близкой к натурной модели серверной части полнообъектной СУБД картографических сцен с ассоциативной защитой для целей исследований.
- Установление по результатам экспериментов на этой модели предпочтительности выбора так называемого монокластерного режима работы СУПБД КС.

Практическая ценность работы

Разработка практических рекомендаций по построению высокоэффективных специализированных систем управления полнообъектными базами данных картографических сцен с ассоциативной защитой. Созданный исследовательский прототип системы (Map Cluster) может быть использован как платформа для дальнейших исследований вопросов защиты и обработки картографических данных и изучения этих вопросов в учебном процессе вузов.

ОСНОВНЫЕ ПУБЛИКАЦИИ

1. Reliable Recognition of Masked Binary Matrices. Connection to Information Security in Map Systems / V. A. Raikhlin, I. S. Vershinin, R. F. Gibadullin, S. V. Pystogov // Lobachevskii Journal of Mathematics, 2013, Vol. 34, No. 4, pp. 319–325. (*из списка ВАК / SCOPUS*)
2. Сравнение архитектурных решений для полнообъектной СУБД картографических сцен с ассоциативной защитой. / Пыстогов С.В. // Вестник КГТУ им.А.Н.Туполева. 2014. №3. (*из списка ВАК*)
3. Вершинин И.С., Гибадуллин Р.Ф., Пыстогов С.В., Перухин М.Ю. Импорт/экспорт ассоциативно защищенных картографических данных с их обработкой в системе Security Map Cluster // Вестник Казан. технол. ун-та, 2015. – № 10. – С. 174-180. (*из списка ВАК*)
4. Raikhlin V.A., Vershinin I.S., Gibadullin R.F., Pystogov S.V. Reliable Recognition of Masked Cartographic Scenes During Transmission over the Network // 2016 International Siberian Conference on Control and Communications (SIBCON-2016). (**SCOPUS**)

Дополнительно:

- 2 научные статьи;
- 11 тезисов докладов в материалах конференций;
- 1 совместная монография;
- 1 учебное пособие;
- Свидетельство о государственной регистрации программы для ЭВМ.