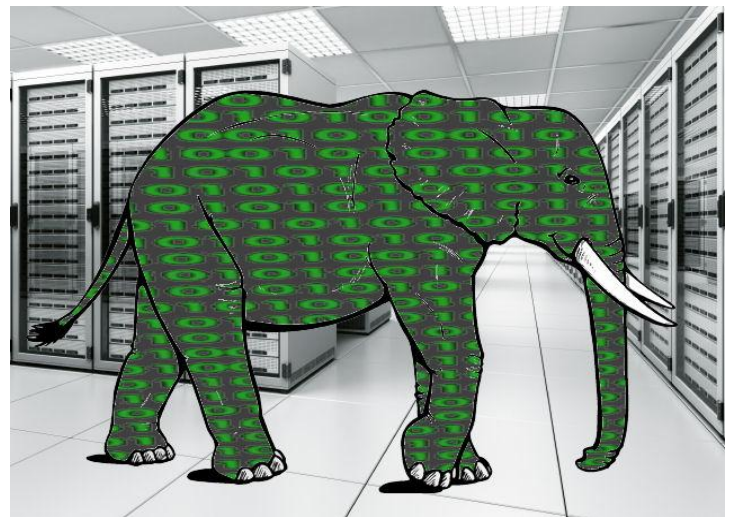
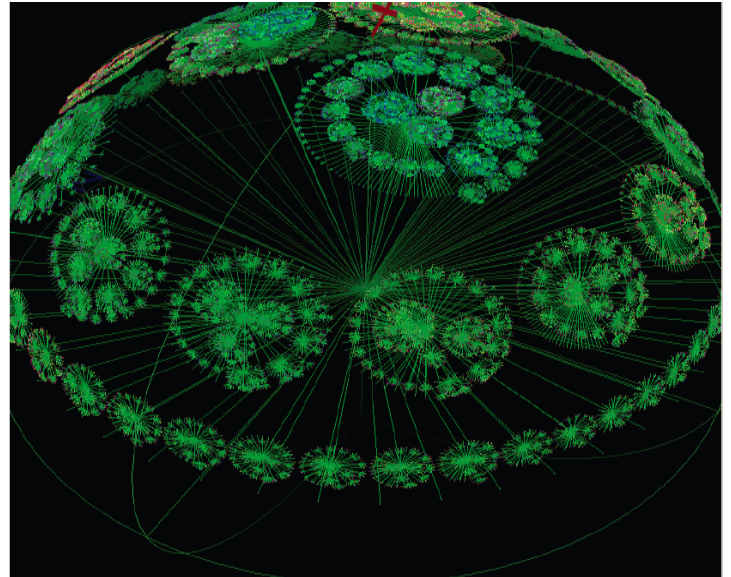


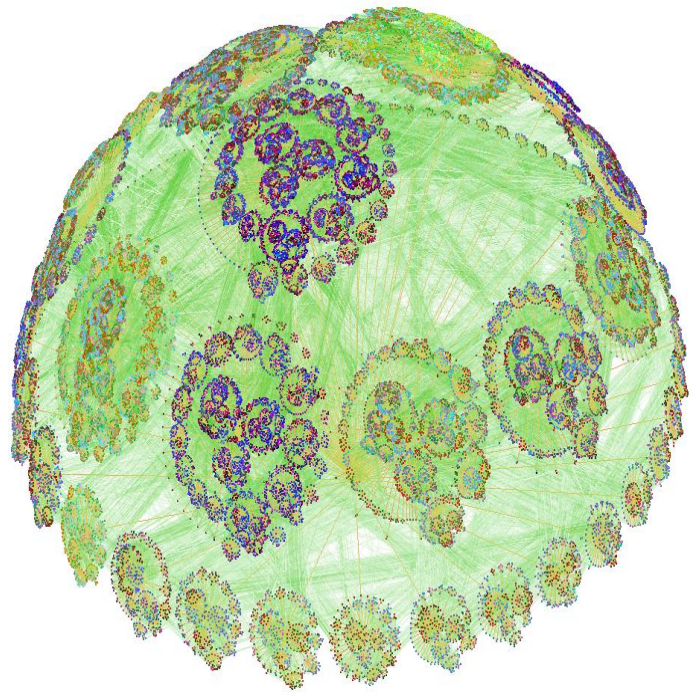
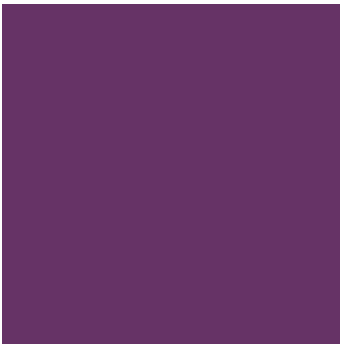


# CLOUD MSW



## Облачная СУБД с поиском по ПОХОЖЕСТИ

Распределенное хранение больших данных  
Многопользовательский доступ  
Неограниченное масштабирование  
Кастомизируемый поиск по похожести



## CLOUD MSW:

### ключевой компонент приложений для поиска данных по схожести

Эта СУБД предлагается тем, кто разрабатывает приложения для работы с большими данными и нуждается в организации распределенного хранения, множественного доступа и быстрого поиска данных по критериям, которые принято называть «схожестью». Сегодня построено уже немало такого рода приложений для поиска похожих изображений, нуклеотидных цепочек и протеинов, лекарственных веществ, музыкальных записей и отпечатков пальцев. Они являются важной частью систем распознавания лиц и поведения, вещей, плагиата в литературе и научных публикациях. Системы классификации объектов и сцен для анализа уровня безопасности и систем оружия также используют поиск по схожести в больших базах данных сигнатур. Фокус этих разработок сегодня лежит на нахождении вычислимых и содержательных признаков схожести, характерных для каждого

из целевых классов объектов. Для хранения данных используются как правило стандартные СУБД, которые построены для хранения данных без учета специфики поиска по схожести, из которых данные извлекаются в специальную структуру для повышения эффективности поиска. В некоторых системах данные хранятся в приложениях непосредственно как объекты, образуя проприетарную объектную СУБД. Такие подходы оказываются неприемлемыми в тех случаях, когда предмет поиска находится среди огромного хранилища данных (big data case), которое невозможно за приемлемое время выгрузить в специальную ориентированную на поиск структуру, а написание приложения, в котором корректно написан код распределенного хранения и поиска с масштабированием делает его труднореализуемым и неприемлемо дорогим.



# CLOUD MSW – это платформа для организации распределенного хранения и поиска данных



CLOUD MSW – программная платформа, позволяющая объединить в единую базу данных неограниченное количество файлов, для которых пользователем определяется одна или несколько метрик – то есть числовые характеристики похожести одного файла на другой. Файлы могут быть различного типа и определять текстовые документы, изображения, аудио или видео записи и т.п. Такие файлы являются минимально адресуемыми единицами (MAE) информации в «CLOUD MSW». API СУБД поддерживает следующие операции над MAE :

- запись MAE;
- чтение по точному совпадению с заданным фрагментом или метаданным файла;
- чтение множества MAE похожих на заданный файл или его метаданные;
- удаление MAE.

Файлы могут быть размещены как на единичном сервере, так и на многих серверах объединенных IP-сетью, в том числе удаленно, что не нарушает единства базы данных. По мере необходимости платформенные компоненты СУБД могут развертываться на вновь подключаемых к сети серверах, обеспечивая неограниченное масштабирование. Операции с данными могут производиться независимо, асинхронно с различных серверов, поддерживающих данную СУБД. Поиск данных по похожести производится без построения индекса содержимого файлов, что делает латентность СУБД весьма малой: Файл становится доступным для поиска немедленно после завершения его присоединения к базе данных. СУБД CLOUD MSW может функционировать в роли интегратора файловых каталогов одного или многих компьютеров в единую базу данных с поиском по похожести. Методы определения похожести могут быть как внешними и

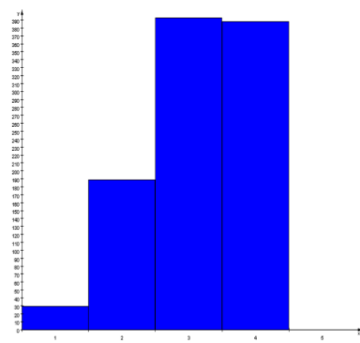
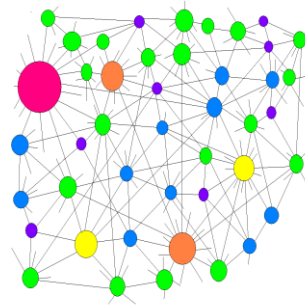
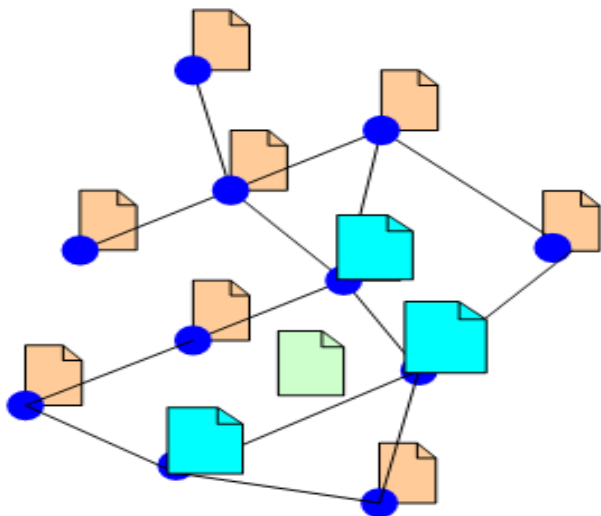
создаваться пользователем, так и выбираться из типовой библиотеки метрик близости файлов различных данных, входящей в состав поставляемой СУБД. В составе библиотеки реализованы следующие алгоритмы определения метрики (похожести):

- для векторов - Alignment, Cosine, Dice, Euclidean, Jaccard, Manhattan, Overlap, Pearson;
- для строк или последовательностей строк (текст) – Jaro, TFIDF, Averaged String Matching;
- для множеств - Jaccard, Loss of Information, Resemblance;
- для последовательностей - Levenstein Edit Distance;
- для деревьев - Bottom-up/Top-down Maximum Common Subtree, Tree Edit Distance;
- для графов - Conceptual Similarity, Graph Isomorphism, Subgraph Isomorphism, Maximum Common Subgraph Isomorphism, Graph Isomorphism Covering.

Для многих из этих алгоритмов имеется имплементация для CUDA платформ на базе GPU вычислителей Tesla от Nvidia. Использование таких аппаратных платформ позволяет многократно ускорить вычисление метрик, а значит и процесс поиска данных.

СУБД CLOUD MSW развертывается на отдельных серверах с виртуальной машиной Java SE 6.0 или на кластерах, а также в рамках одного или нескольких ЦОДов, объединенных IP-сетью.

API CLOUD MSW позволяет интегрировать СУБД в различные пользовательские приложения, используя стандартные средства Java SE.



# CLOUD MSW изнутри:

## Графовая, распределенная, гетерархическая

CLOUD MSW может быть представлена как графовая СУБД, вершины графа ассоциируются с хранимыми файлами (MAE), а ребра графа – определяют отношение соседства, максимальной близости для пары хранимых файлов. Отличительной чертой CLOUD MSW является возможность создания нескольких графов, соответствующих различным критериям схожести – метрикам. Поиск по схожести в этом случае может осуществляться как по локальным, так и по глобальным критериям с использованием методов многокритериальной оптимизации. Логическая структура базы данных определяется набором файлов, хранящихся в произвольном месте и привязанных к каждому файлу набора ссылок на своих соседей, подобно тому как это делается для XML+XLink файлов. Благодаря тому, что физическое и логическое местоположение хранимых файлов не имеет значения для работы СУБД, система поддерживает полную распределенность без необходимости построения таблиц хэш-кодов или других централизованных индексных структур.

Каждый из хранимых файлов (MAE) вместе с его ссылками и некоторым специальным кодом определяют так называемую активную единицу хранения (ADU-Active Data Unit). ADU взаимодействуют между собой, обеспечивая процесс traversal с помощью специальных сообщений, поддерживаемых платформой. В силу полной децентрализованности архитектура СУБД CLOUD MSW относится к гетерархическим. Ключевым преимуществом такой децентрализованности является высокая, практически неограниченная масштабируемость и устойчивость к нагрузкам. При этом CLOUD MSW обеспечивает высокую скорость поиска, близкую к поиску по дереву, благодаря применению специальных алгоритмов traversality, основанных на патентованном решении. В основе этого решения лежит организация хранения файлов в виде графа со свойствами метризованного тесного мира (Metrized Small World - MSW), подобного графам социальных сетей. В таких сетях любая пара вершин соединена между собой весьма коротким путем.

# CLOUD MSW API

Прикладной интерфейс (API) СУБД CLOUD MSW состоит из двух частей – интерфейс спецификации предметной области (App Domain API) и интерфейс доступа к данным (Data Access API).

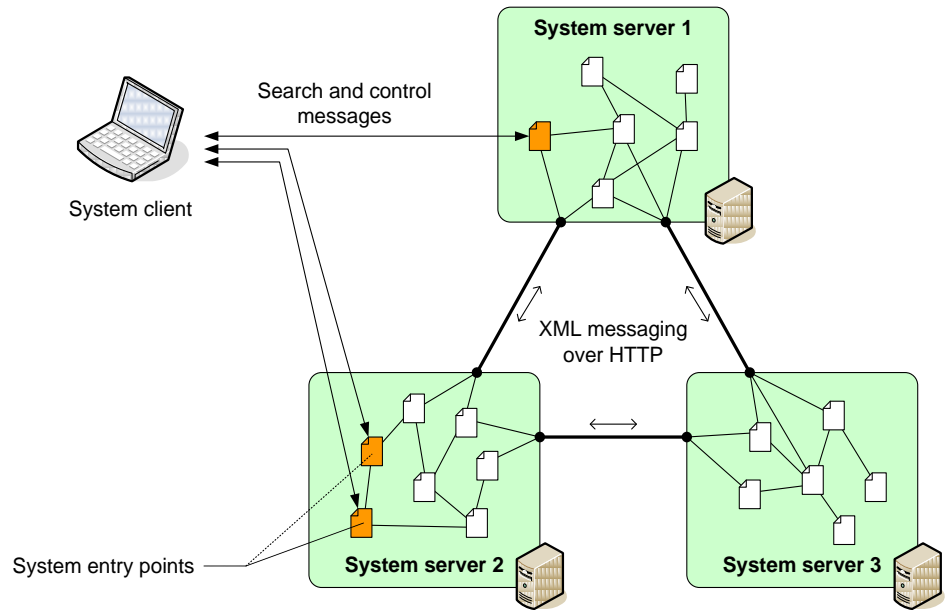
App Domain API предоставляет интерфейс для подключения программного модуля на языке Java, который задает следующие компоненты предметной области:

1. Средства извлечения поисковых данных из сохраняемых в системе файлов.
2. Средства постоянного сохранения и чтения поисковых данных.
3. Метрики (критерии близости), применяемые к поисковым данным.

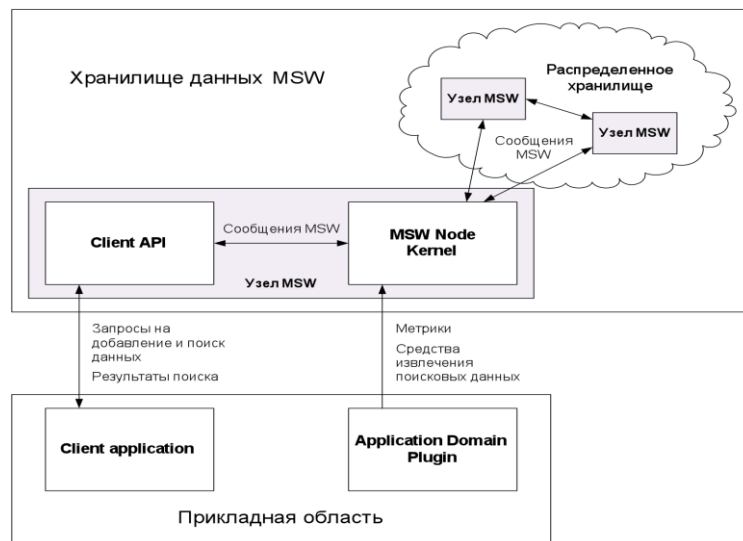
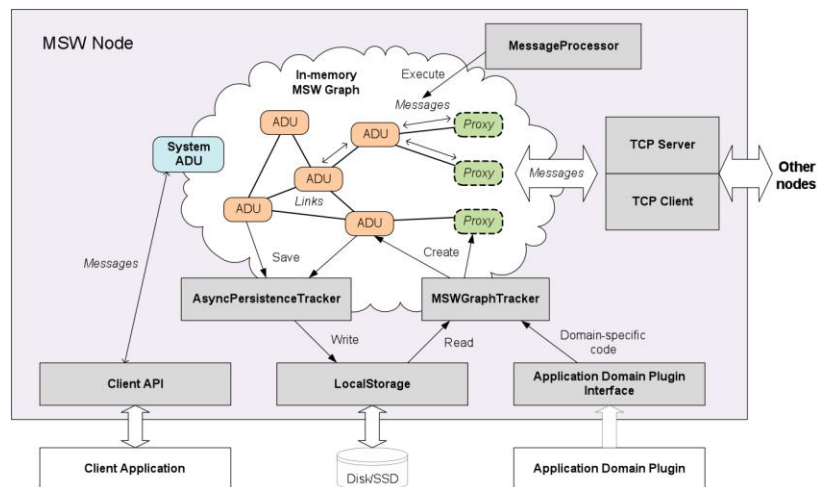
Data Access API (библиотека на языке Java) используется клиентским приложением, предоставляя следующую функциональность по доступу к данным в СУБД:

1. Добавление новых файлов.
2. Извлечение сохраненных файлов по уникальному идентификатору.
3. Извлечение сформированных связей между сохраненными в системе файлами.
4. Извлечение набора файлов по заданному критерию близости к запросу.
5. Удаление сохраненных файлов.

Примеры можно посмотреть ниже.



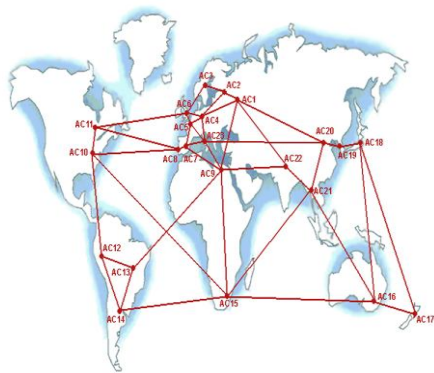
## Функциональная диаграмма платформы





# Интеграция с приложением

## Развертывание, наполнение, поиск

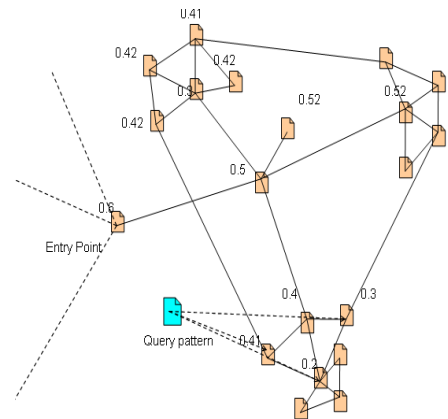


*Распределенное развертывание*

```
package com.meralabs.msw.server;
import java.io.File;
import java.io.InputStream;
import java.io.FileOutputStream;
import java.io.FileInputStream;
import java.io.InputStreamReader;
import java.io.IOException;
import org.apache.commons.cli.Option;
import org.apache.commons.cli.OptionBuilder;
import org.apache.commons.cli.Options;
import org.apache.commons.cli.CommandLine;
import org.apache.commons.lang3.SystemUtils;
import org.json.JSONObject;
import org.json.JSONTokener;
import com.meralabs.msw.Node;
import com.meralabs.msw.Util;
import com.meralabs.msw.cli.App;
public class Server extends App
{
    private static final String
    OPT_CREATE_CONFIG_LONG =
    "create-config";
    private static final String
    OPT_CREATE_CONFIG_SHORT = "n";
    private static final String
    OPT_CREATE_SCRIPTS_LONG =
    "create-scripts";
```

*Наполнение данными*

```
// adding new application
data items
Node clientNode = new
Node(Node.DEFAULT_CLIENT_CO
NFIG); // emdedded client
node
String[] serverNodeIDs =
{"msw:server1.example.com:8
080",
"msw:server2.example.com:80
82"}; // server node IDs
for data storage
Loader dataLoader = new
Loader(clientNode,
serverNodeIDs);
byte[] applicationData1 =
readFromFile("example1");
byte[] applicationData2 =
readFromFile("example2");
Future<AduAddress>
newAduAddressFuture1 =
dataLoader.load(application
Data1, "exampleContentID");
Future<AduAddress>
newAduAddressFuture2 =
dataLoader.load(application
Data2, "exampleContentID");
```



*Быстрый поиск по похожести*

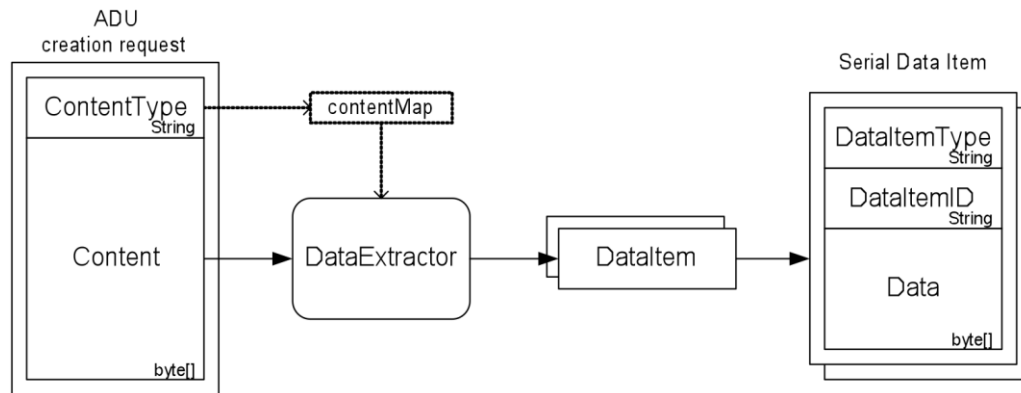
```
//searching for data items
Node clientNode = new
Node(Node.DEFAULT_CLIENT_CO
NFIG);
String serverNode =
"msw:server1.example.com:80
80";
SimilaritySearch
similaritySearch = new
SimilaritySearch(clientNode
, serverNode);
byte[] query =
```

Обмен сообщениями между узлами осуществляется с помощью кросс-платформенного протокола Google Protocol Buffers, поэтому можно осуществлять доступ к данным из программ, написанных на других языках программирования (C++, Python), и даже создавать реализации самих узлов на этих языках

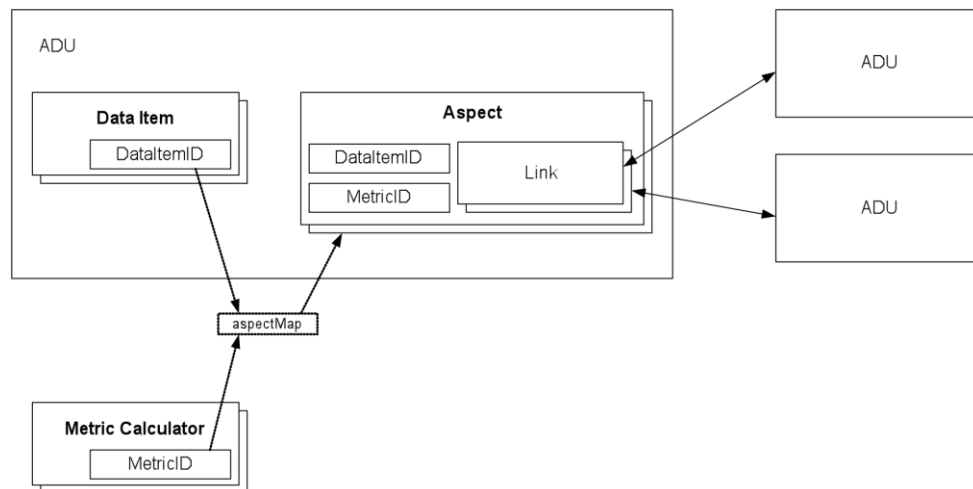


# CLOUD MSW работает с большими файлами

*CLOUD MSW может использоваться в качестве гигантского распределенного индекса: в графовую базу данных записываются только поисковые данные и для вычисления метрики нет необходимости извлекать весь файл*



При поступлении нового элемента прикладных данных (файла) происходит извлечение элементов поисковых данных (DataItem) активной единицы хранения (ADU), которые затем записываются в хранилище. Это позволяет избежать извлечения поисковых данных при каждом обращении к единице хранения.

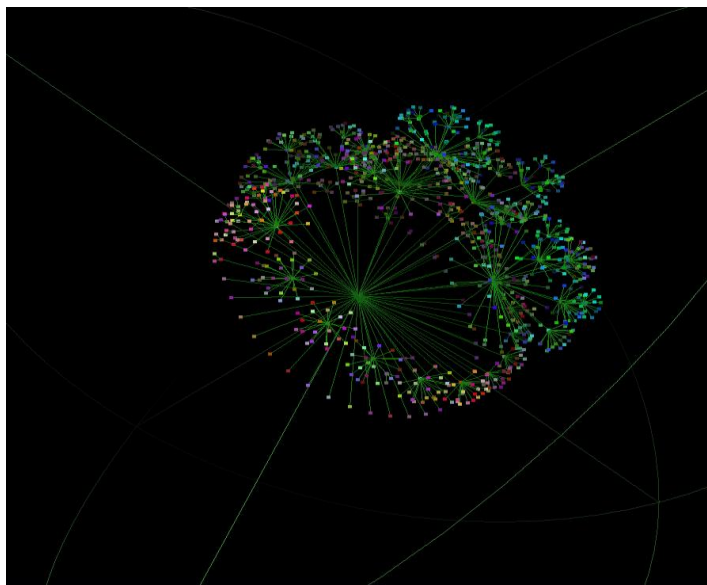


На основании данных о связях элементов поисковых данных и метрик формируются аспекты ADU, которые представляют элемент прикладных данных в контексте конкретного критерия близости. После этого для каждого из аспектов производится присоединение новой ADU к другим ADU, у которых имеется такой же аспект, путем заполнения списка ссылок, связанного с этим аспектом.



+

# CLOUD MSW



+

## MERA LABS L.L.C.

Ul. Delovaya, 13, Nizhny Novgorod, Russia, 603163

[www.meralabs.com](http://www.meralabs.com)