

A dramatic image featuring a large, glowing nuclear mushroom cloud rising from the ocean. The scene is set against a sunset sky with vibrant orange and red clouds. The sun's reflection is visible on the water's surface. The text "БЕЖЕНАРЬ МИХАИЛ" is superimposed over the base of the mushroom cloud.

БЕЖЕНАРЬ МИХАИЛ

DB - GRAPH
DBMS - TinkerGraph

- Развитие идеи использовать graph db 1960-2000, НО проблемы с ACID гарантиями

К примеру: навигационные бд IBM IMS имели древовидные структуры в своей иерархической модели, но строгую древовидную структуру можно было обойти с помощью виртуальных записей

- Первая компания использующая такой тип БД и дающая гарантии ACID
Neo4j - 2007



Developer(s) Neo4j

Initial release 2007; 17 years ago^[1]

Stable release 5.19^[2] / 12 April 2024; 12 days ago

Repository github.com/neo4j/neo4j

Written in Java

Type Graph database

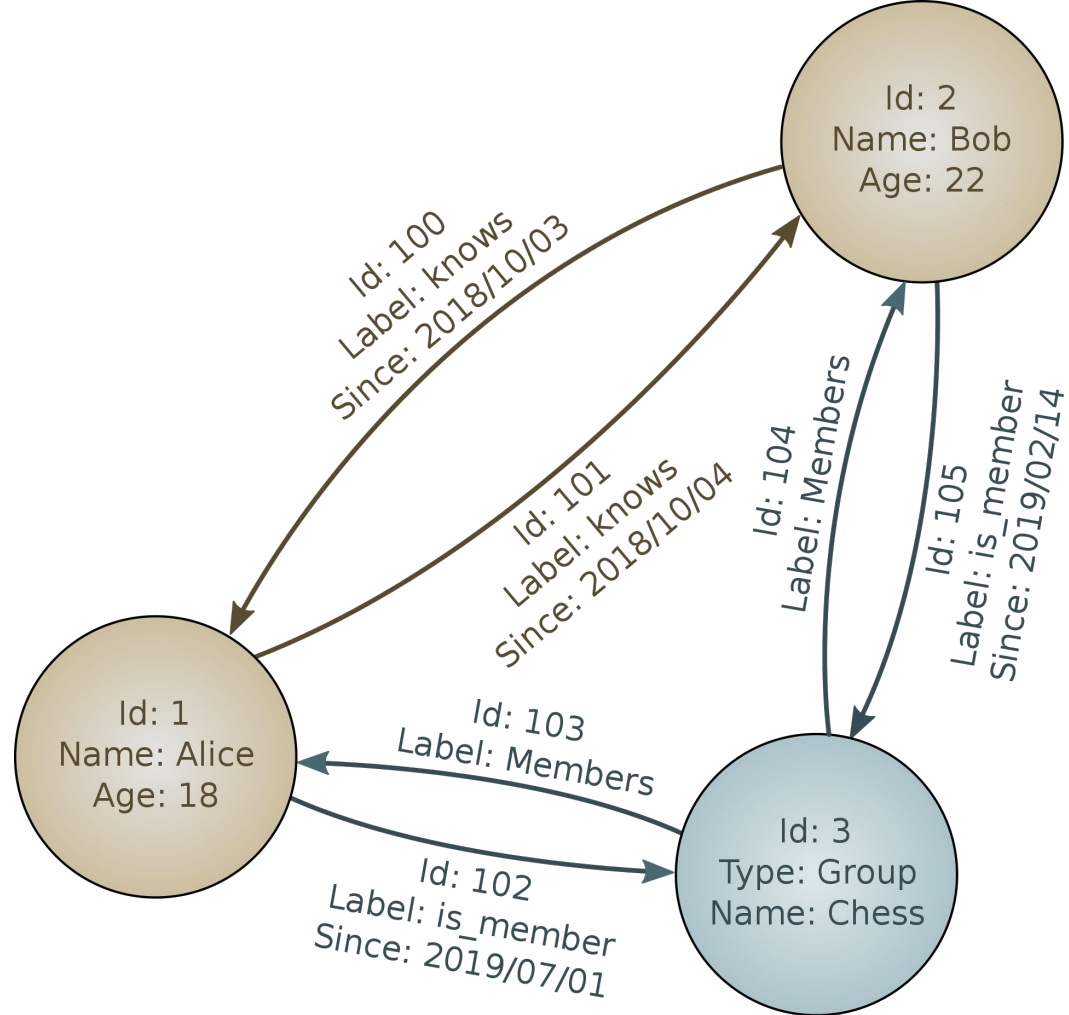
License **Source code:** GPLv3 and AGPLv3 with Commons Clause
Binaries: Freemium registerware

Website neo4j.com

Небольшой пример графовой
БД.

В вершинах хранятся объекты

Ребра это связи, отношения,
между объектами.



Преимущества:

1. Гибкость и масштабируемость
2. Быстрый доступ к данным
3. Анализ связей
4. Гибкая структура данных

Инструменты для взаимодействия с СУБД:

БАЗА Apache TinkerPop is a graph computing framework and top level project hosted by the Apache Software Foundation

ВОТ В НЕМ ЕСТЬ Язык запросов Gremlin:

```
(base) moisha@MacBook-Pro-Mihail-2 bin % ./gremlin.sh
```

```
  \,,,/
  (o o)
-----o00o-(3)-o00o-----
plugin activated: tinkerpop.server
plugin activated: tinkerpop.utilities
plugin activated: tinkerpop.tinkergraph
[gremlin> graph = TinkerGraph.open()
==>tinkergraph[vertices:0 edges:0]
gremlin> █
```



Лого ↑

Какой database engine используется в вашей СУБД?

TinkerGraph использует собственный графовый движок для обработки операций с данными, основанных на графах

Name	TinkerGraph
Description	A lightweight, in-memory graph engine that serves as a reference implementation of the TinkerPop3 API

Далее рассмотрим **CRUD** —>

Разверните БД с данными и выполните ряд запросов:

Создание графового объекта, а также объекта Traversal(обход), который манипулирует нашим графовым объектом.

```
[(base) moisha@MacBook-Pro-Mihail-2 bin % ./gremlin.sh
```

```
  \,,,/
  (o o)
```

```
-----o00o-(3)-o00o-----
```

```
plugin activated: tinkerserver
```

```
plugin activated: tinkertools
```

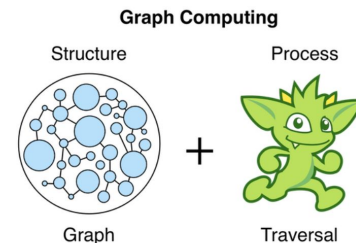
```
plugin activated: tinkerg
```

```
[gremlin> g = TinkerGraph.open().traversal()
```

```
==>graphtraversalsource[tinkerg[vertices:0 edges:0], standard]
```

```
gremlin> █
```

Graph Computing



Добавление вершины и добавление “свойств”

```
gremlin> g.addV('person').property('name', 'Alice').property('age', 30)
==>v[0]
gremlin> g.addV('person').property('name', 'Bob').property('age', 25)
==>v[3]
gremlin> g.addV('person').property('name', 'Adolf').property('age', 4)
```

Список всех вершин

```
gremlin> g.V().toList()
==>v[0]
==>v[3]
==>v[6]
gremlin> █
```

Добавление ребер

```
gremlin> g.V().has('name', 'Alice').addE('knows').to(__.V().has('name', 'Bob')).next()  
==>e[9][0-knows->3]  
gremlin> g.V().has('name', 'Alice').addE('mother').to(__.V().has('name', 'Adolf')).next()  
==>e[10][0-mother->6]  
gremlin> █
```

Список ребер

```
gremlin> g.E().toList()  
==>e[9][0-knows->3]  
==>e[10][0-mother->6]  
gremlin> █
```

Поиск связанных вершин

```
gremlin> g.V().has('name', 'Alice').out('knows').values('name')  
==>Bob  
gremlin> g.V().has('name', 'Alice').out('mother').values('name', 'age')  
==>Adolf  
==>4  
gremlin> █
```

Количество ребер

```
gremlin> g.E().count().next()  
==>2  
gremlin> █
```

Поиск с условием

```
[gremlin> g.V().has('age', gt(20)).toList()  
==>v[0]  
==>v[3]
```

Список вершин, у которых ключ age имеет значение больше 20

Обновление вершины

```
[gremlin> g.V().has('name', 'Adolf').property('age', 56)  
==>v[6]
```

Удаление вершины

```
[gremlin> g.V().count()  
==>8  
[gremlin> g.V().has('name', 'Adolf').drop()  
[gremlin> g.V().count()  
==>7  
gremlin> █
```

Какие типы индексов поддерживаются в БД

ВИДИМО НЕТ

Secondary indexes	no
-------------------	----

Но при этом во фреймворке tinkrPop такая возможность, вроде, есть и там можно создавать индексы по вершинам и ребрам

На каком языке программирования написана СУБД?

JAVA

```
(base) moisha@MacBook-Pro-Mihail-2 lib % ls -l
total 55768
-rw-rw-r--@ 1 moisha  staff   337868 25 anp   2023  antlr4-runtime-4.9.1.jar
-rw-rw-r--@ 1 moisha  staff   968137 25 anp   2023  caffeine-2.3.1.jar
-rw-rw-r--@ 1 moisha  staff   246918 25 anp   2023  commons-beanutils-1.9.4.jar
-rw-rw-r--@ 1 moisha  staff   353793 25 anp   2023  commons-codec-1.15.jar
-rw-rw-r--@ 1 moisha  staff   588337 25 anp   2023  commons-collections-3.2.2.jar
-rw-rw-r--@ 1 moisha  staff   644714 25 anp   2023  commons-configuration2-2.9.0.jar
-rw-rw-r--@ 1 moisha  staff   577742 25 anp   2023  commons-lang3-3.11.jar
-rw-rw-r--@ 1 moisha  staff    61829 25 anp   2023  commons-logging-1.2.jar
-rw-rw-r--@ 1 moisha  staff   238400 25 anp   2023  commons-text-1.10.0.jar
-rw-rw-r--@ 1 moisha  staff    46942 25 anp   2023  exp4j-0.4.8.jar
-rw-rw-r--@ 1 moisha  staff   281817 13 май 2023  gremlin-console-3.6.4.jar
-rw-rw-r--@ 1 moisha  staff   2144892 13 май 2023  gremlin-core-3.6.4.jar
-rw-rw-r--@ 1 moisha  staff   219275 13 май 2023  gremlin-driver-3.6.4.jar
-rw-rw-r--@ 1 moisha  staff   723144 13 май 2023  gremlin-language-3.6.4.jar
-rw-rw-r--@ 1 moisha  staff   2602884 13 май 2023  gremlin-shaded-3.6.4.jar
-rw-rw-r--@ 1 moisha  staff   5664564 25 anp   2023  groovy-2.5.22-indy.jar
-rw-rw-r--@ 1 moisha  staff    65008 25 anp   2023  groovy-cli-picocli-2.5.22.jar
-rw-rw-r--@ 1 moisha  staff   749942 25 anp   2023  groovy-console-2.5.22.jar
-rw-rw-r--@ 1 moisha  staff   443401 25 anp   2023  groovy-groovysh-2.5.22-indy.jar
-rw-rw-r--@ 1 moisha  staff   128557 25 anp   2023  groovy-json-2.5.22-indy.jar
-rw-rw-r--@ 1 moisha  staff    17319 25 anp   2023  groovy-jsr223-2.5.22-indy.jar
-rw-rw-r--@ 1 moisha  staff   374398 25 anp   2023  groovy-swing-2.5.22.jar
-rw-rw-r--@ 1 moisha  staff   101107 25 anp   2023  groovy-templates-2.5.22.jar
-rw-rw-r--@ 1 moisha  staff   226377 25 anp   2023  groovy-xml-2.5.22.jar
-rw-rw-r--@ 1 moisha  staff   1140290 25 anp   2023  hppc-0.7.1.jar
-rw-rw-r--@ 1 moisha  staff   780321 25 anp   2023  httpclient-4.5.13.jar
-rw-rw-r--@ 1 moisha  staff   328593 25 anp   2023  httpcore-4.4.13.jar
-rw-rw-r--@ 1 moisha  staff   1398591 25 anp   2023  ivy-2.5.1.jar
-rw-rw-r--@ 1 moisha  staff   283858 25 anp   2023  jansi-1.17.1.jar
-rw-rw-r--@ 1 moisha  staff    65507 25 anp   2023  javatuples-1.2.jar
```

IN-MEMORY

TinkerGraph

- A small in-memory graph implementation that is great for learning.

на сколько понял, по дефолту хранится в оперативке, но есть способы, используя сторонние инструменты, хранить в диске.

Процесс выполнения запросов в TinkerGraph

- 1) Инициализации графа
- 2) Запрос
- 3) Gremlin зовет TinkerPop, который умеет обходить графы, нужным нам способом
- 4) Ответ

(кажется, это требовалось:)


Какие методы восстановления поддерживаются в вашей СУБД

Partitioning methods 	none
Replication methods 	none

по сколько все хранится в оперативке, то нужно как-то самому думать об том, чтобы в случае чего все не удалилось

Расскажите про шардинг в вашей конкретной СУБД

TinkerGraph не поддерживает шардинг данных, так как данные хранятся в памяти и не требуют разделения на различные узлы.

Partitioning methods 	none
--	------

Транзакции в вашей TinkerGraph

НЕТ

Transaction concepts 

no

Возможно ли применить термины Data Mining, Data Warehousing и OLAP

- 1) Data Mining: В TinkerGraph можно использовать алгоритмы анализа графов, такие как PageRank, Community Detection, Path Finding..., чтобы находить интересные закономерности.
- 2) Data Warehousing: В TinkerGraph можно создать граф, в который будут включены данные из различных источников, чтобы анализировать их вместе и получать ценные инсайты.
- 3) OLAP: В графовых базах данных OLAP может быть использован для анализа данных в реальном времени с учетом графовой структуры.

(мне этот пункт не нравится)

Есть ли для TinkerGraph понятие «план запросов»?

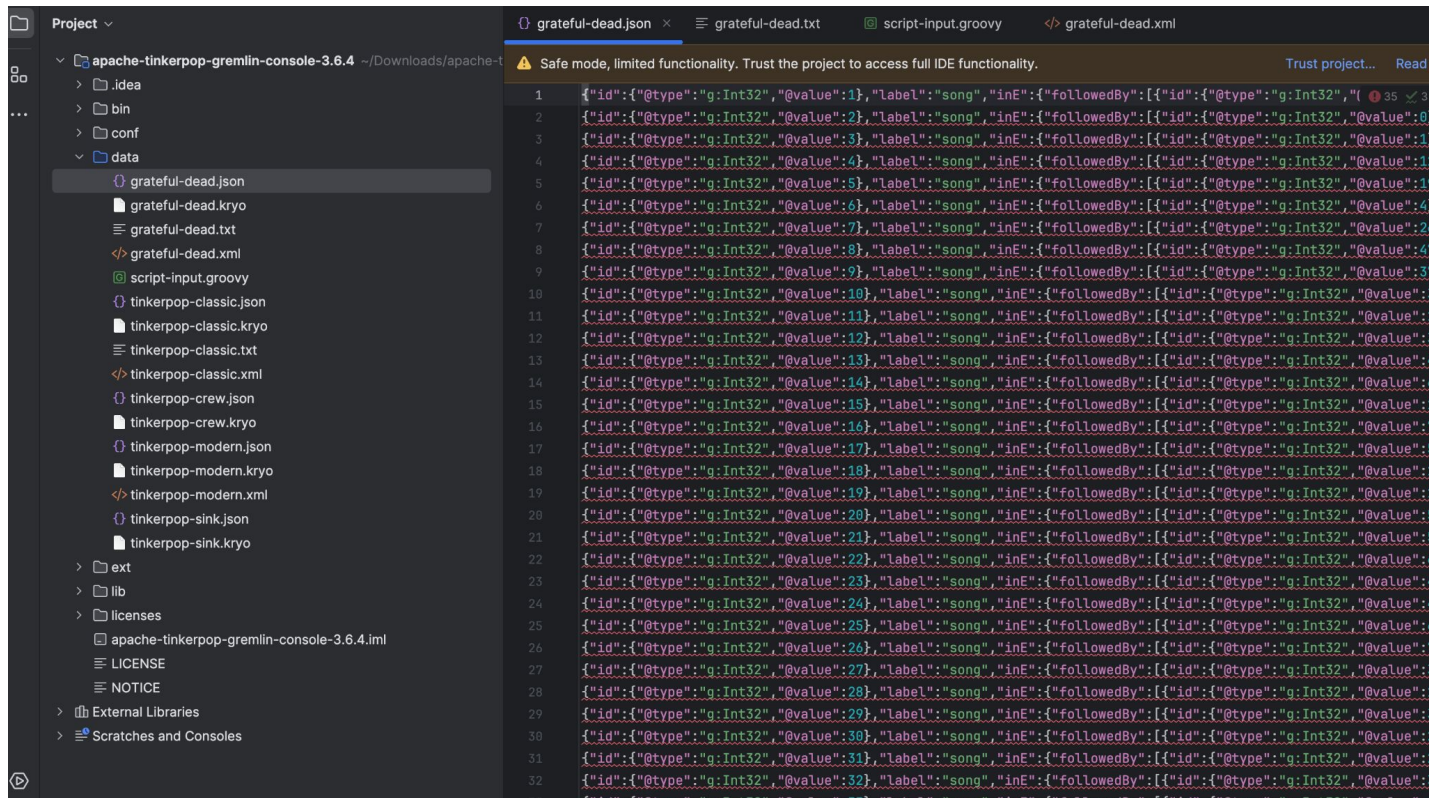
Опять же для самой СУБД нет, но для расширения Gremlin, который является частью TinkerPop, query plan существует

При написании запроса на Gremlin для TinkerGraph, TinkerPop оптимизирует выполнение запроса путем расстановки операций в оптимальном порядке и минимизации прохождения по граф

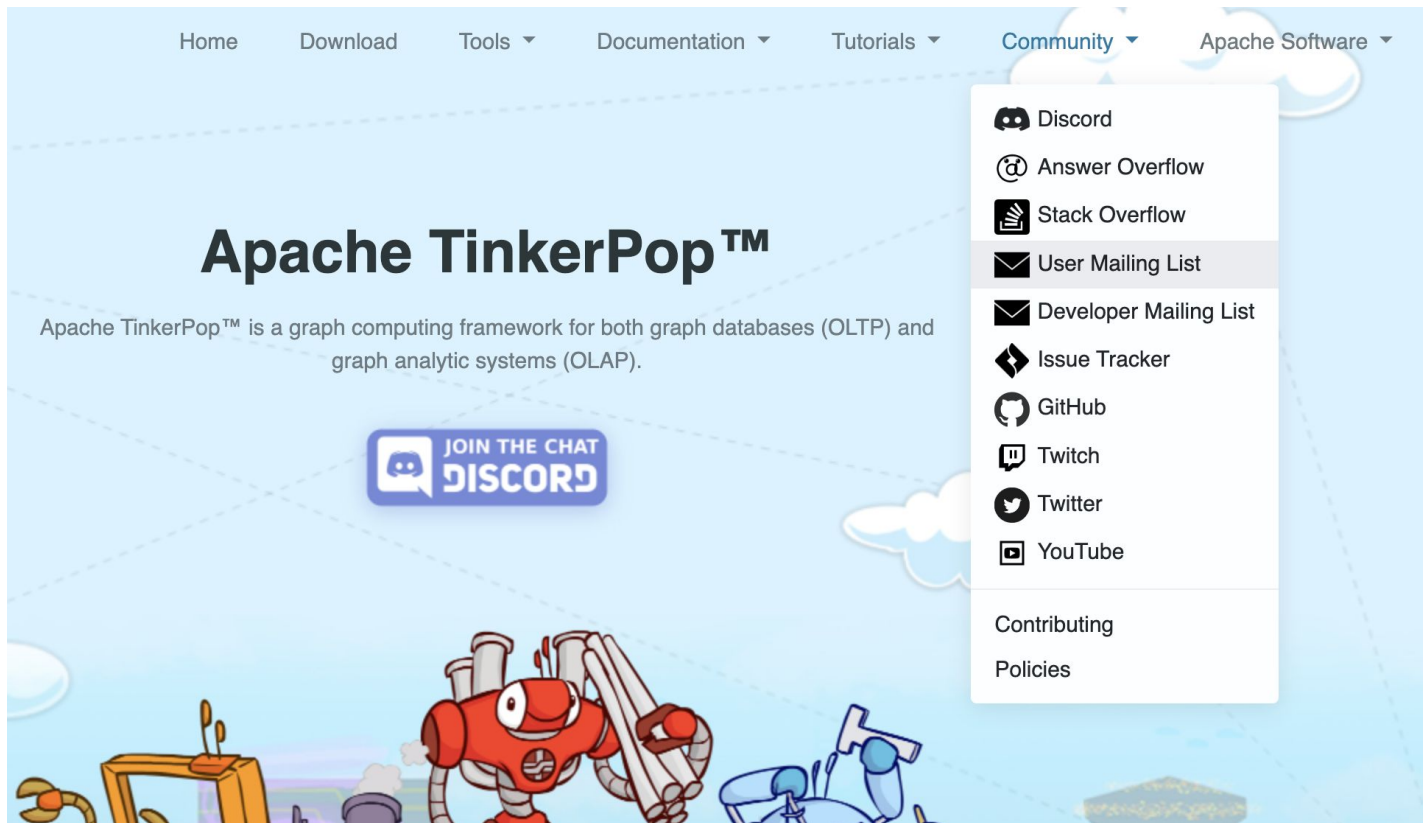
БЕЗОПАСНОСТЬ

TinkerGraph может обеспечивать защиту данных через механизмы доступа к данным, такие как модели авторизации и аутентификации, но шифрование трафика или хранения данных может потребовать сторонних инструментов или настройки на уровне приложения.

ПРИ СКАЧИВАНИИ с официального сайта сразу дается тестовый набор данных



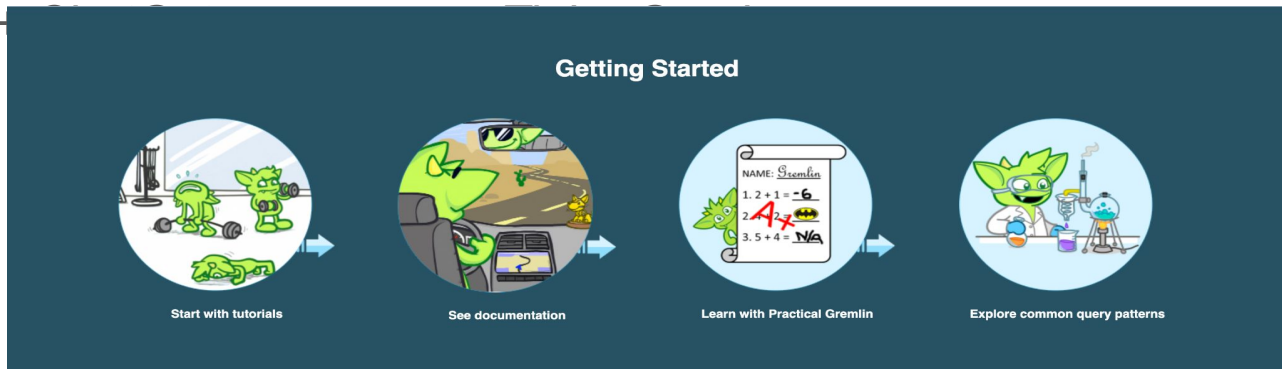
Развитое community



Где найти документацию и пройти обучение

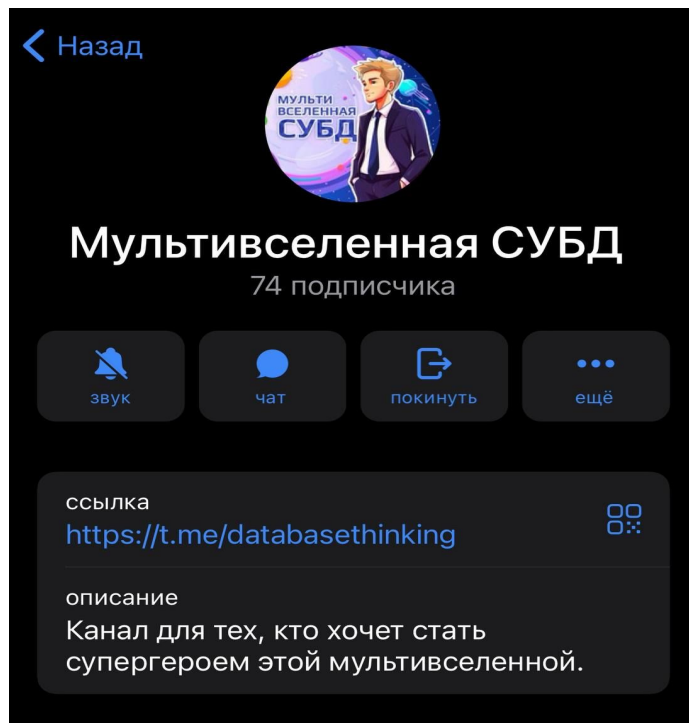
- 1) В интернете
- 2) Официальная документация:
<https://tinkerpop.apache.org/docs/current/reference/#traversalvertexprogram>
- 3) <https://www.kelvinlawrence.net/book/PracticalGremlin.html#csvair> -хорошая книга

В принципе, эта ссылка -



Как быть в курсе происходящего:

ТОЛЬКО ТАК! —>



(+ мониторить официальные ресурсы)