

Реализация теста Graph500 для параллельной СУБД PargreSQL

Научный руководитель:
кандидат физ.-мат. наук, доцент
М.Л. Цымблер

Автор работы:
Студент группы ВМИ-456
А.Ю. Сафонов

Рецензент:
кандидат пед. наук
А.Ю. Эвнин


[Home](#)
[Complete Results](#)
[Benchmarks](#)
[Green Graph 500](#)
[Log In](#)

The Graph 500 List

Top 10 (November 2013)

Rank	Machine
1	DOE/NNSA/LLNL Sequoia - Lawrence Livermore National Laboratory (65536 nodes, 1048576 cores)
2	DOE/SC/Argonne National Laboratory Mira - Argonne National Laboratory (49152 nodes, 786432 cores)
3	JUQUEEN - Forschungszentrum Juelich (FZJ) (16384 nodes, 262144 cores)
4	K computer - RIKEN Advanced Institute for Computational Science (AICS) (65536 nodes, 524288 cores)
5	Fermi - CINECA (8192 nodes, 131072 cores)
	Tianhe-2

November 2013

No.	Rank	Machine	Installation Site	Number of nodes	Number of cores	Problem scale	GTEPS ▼
1	1	DOE/NNSA/LLNL Sequoia (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Lawrence Livermore National Laboratory	65536	1048576	40	15363
2	2	DOE/SC/Argonne National Laboratory Mira (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Argonne National Laboratory	49152	786432	40	14328
3	3	JUQUEEN (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	Forschungszentrum Juelich (FZJ)	16384	262144	38	5848
4	4	K computer (Fujitsu - Custom supercomputer)	RIKEN Advanced Institute for Computational Science (AICS)	65536	524288	40	5524.12
5	5	Fermi (IBM - BlueGene/Q, Power BQC 16C 1.60 GHz)	CINECA	8192	131072	37	2567
6	6	Tianhe-2 (MilkyWay-2) (National University of Defense Technology - MPP)	Changsha, China	8192	196608	36	2061.48
		Turing (IBM -					

Цель и задачи исследования

Цель:

Оценка эффективности параллельной СУБД PargreSQL на задачах интенсивной обработки данных с помощью сравнительного теста Graph500

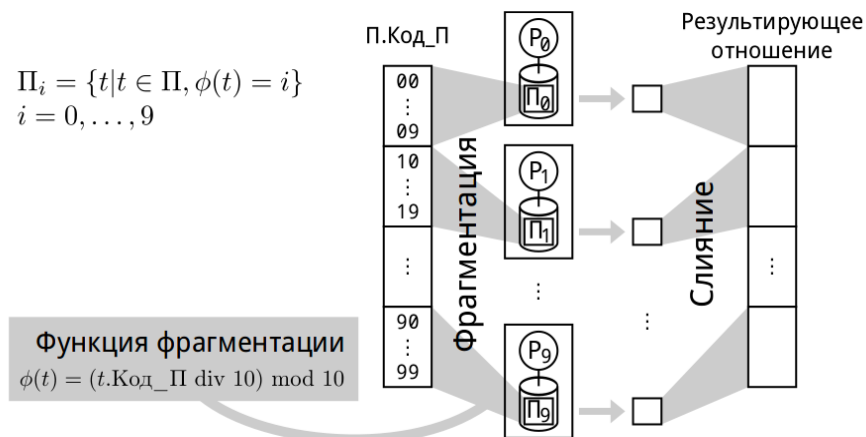
Задачи:

- ▶ Изучить архитектуру параллельной СУБД PargreSQL и спецификацию теста Graph500
- ▶ Разработать схему базы данных для хранения графа и промежуточных данных в соответствии со спецификацией теста Graph500
- ▶ Выполнить проектирование и разработку алгоритмов на языке SQL, реализующих тест Graph500 для параллельной СУБД PargreSQL
- ▶ Провести вычислительные эксперименты на суперкомпьютере «Торнадо ЮУрГУ», исследующие эффективность параллельной СУБД PargreSQL на тесте Graph500

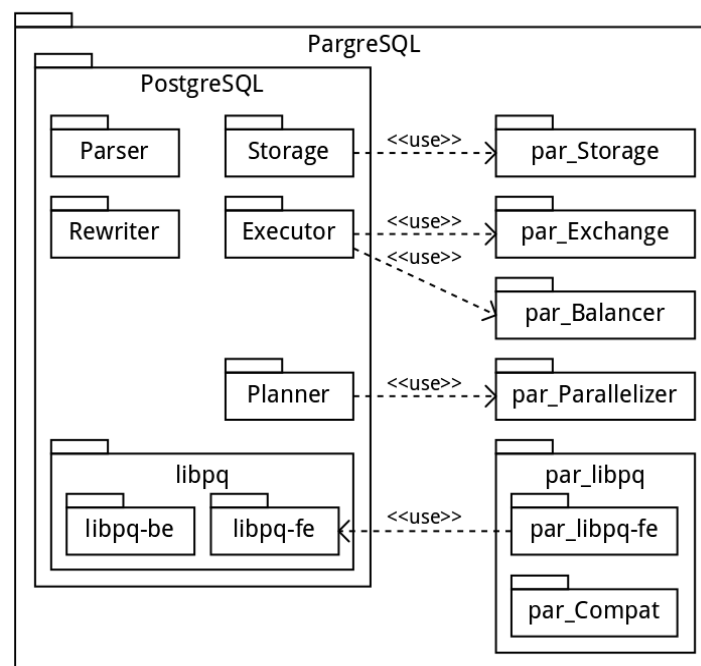


Параллельная СУБД PargreSQL

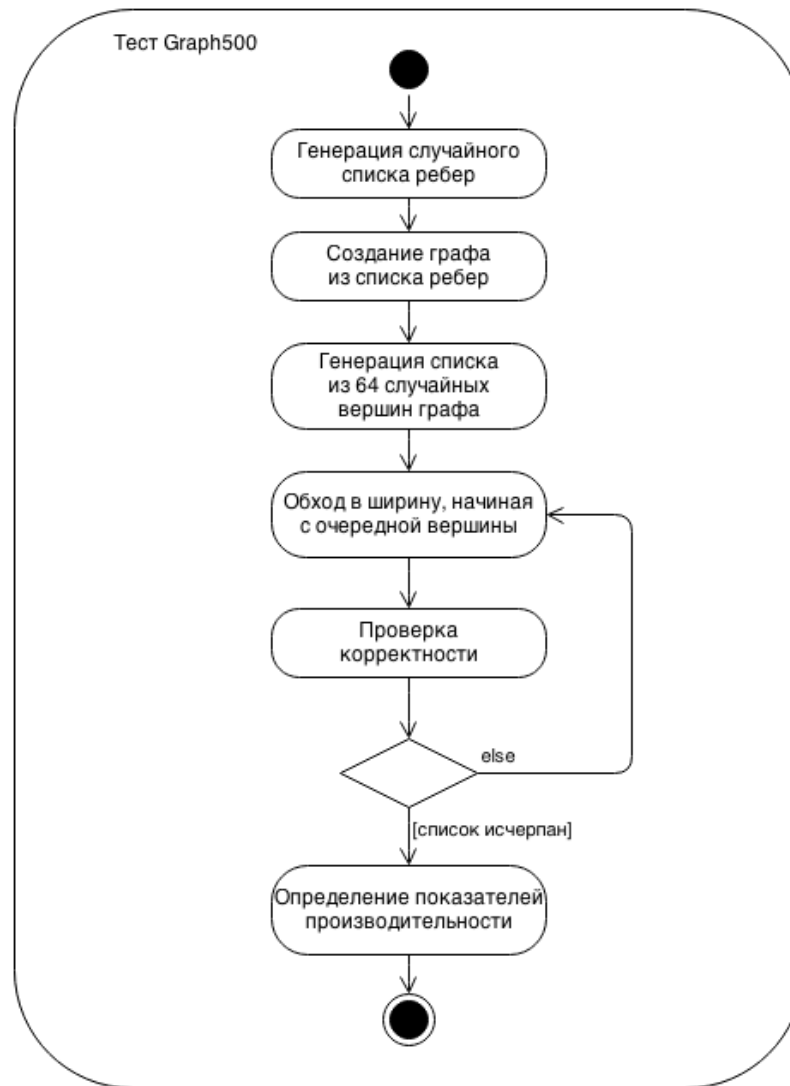
► Фрагментный параллелизм



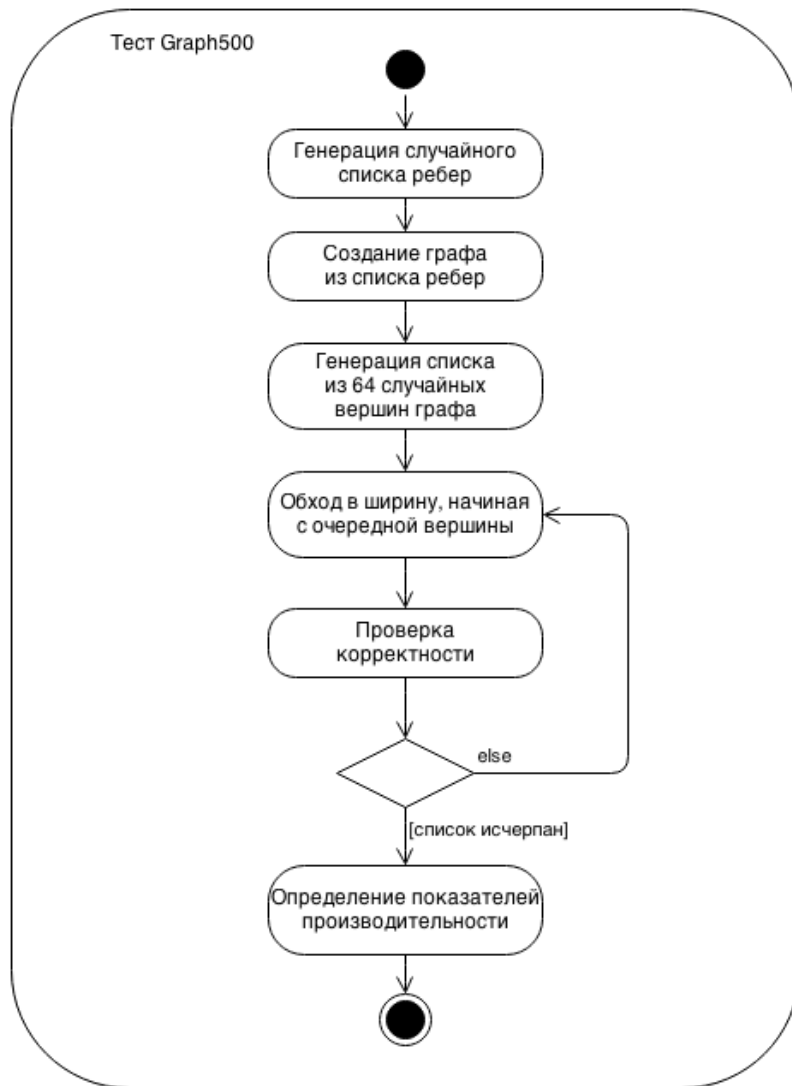
► Архитектура PargreSQL



Тест Graph500



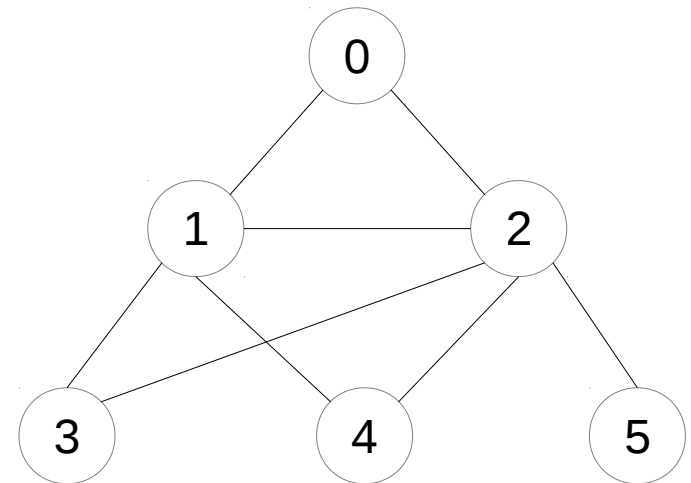
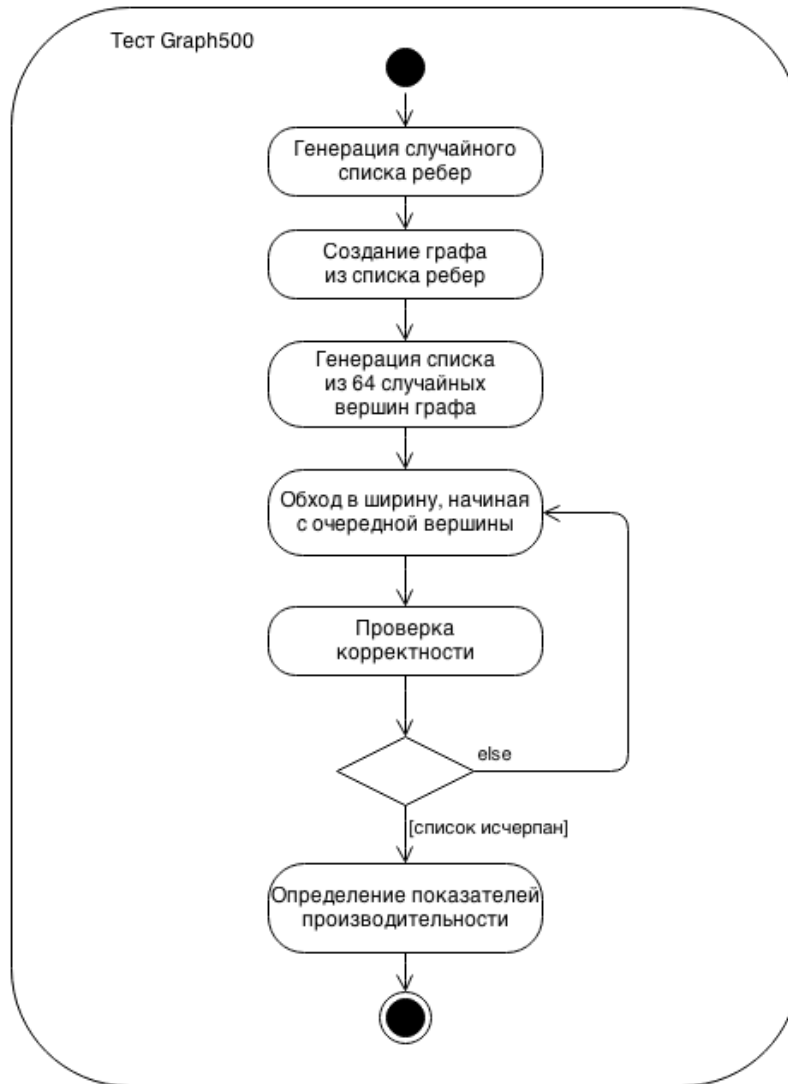
Тест Graph500. Генерация ребер



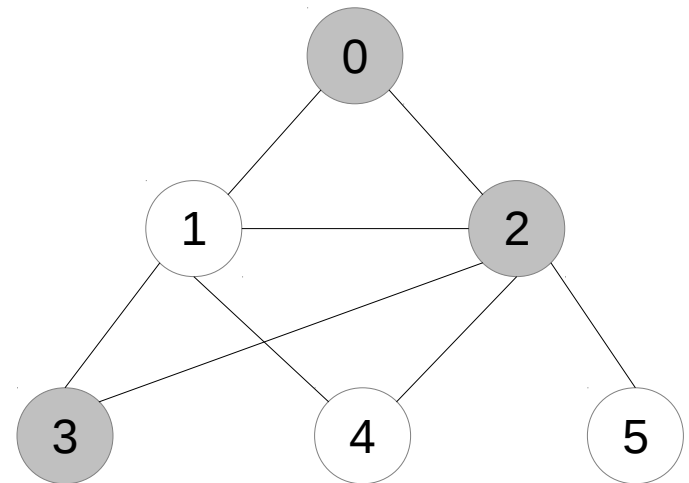
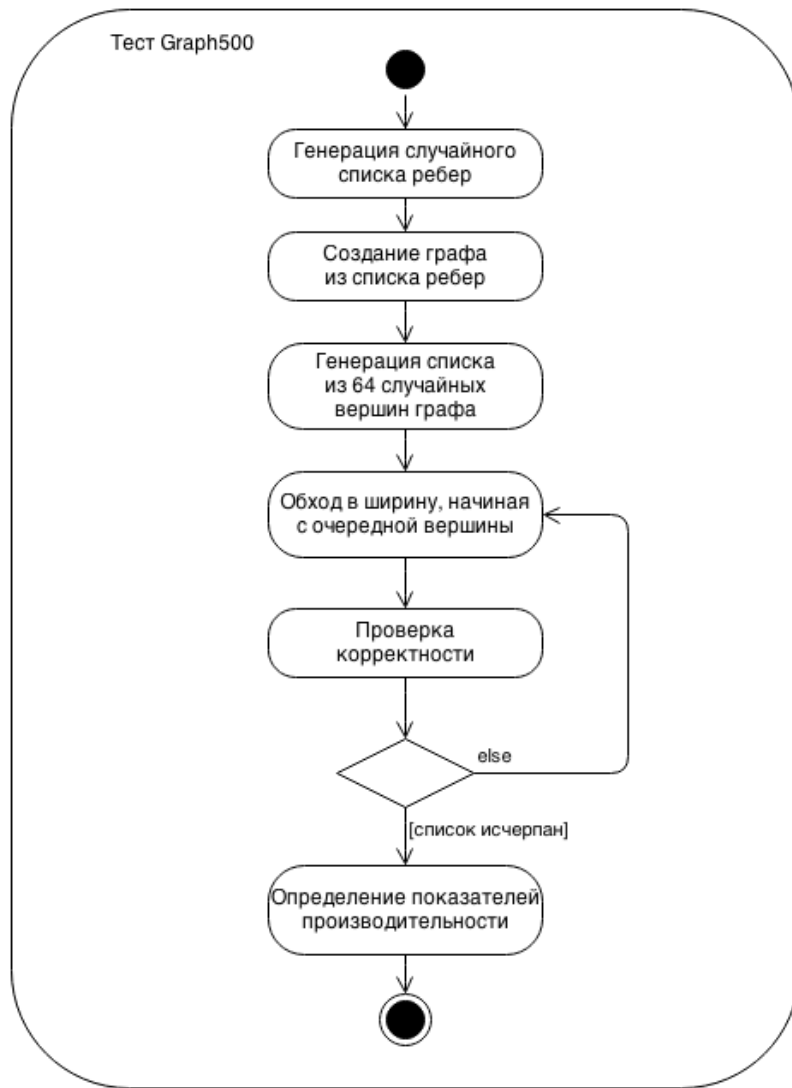
Начальная вершина	Конечная вершина
0	1
2	0
1	2
3	1
4	2
4	1
2	5
3	2



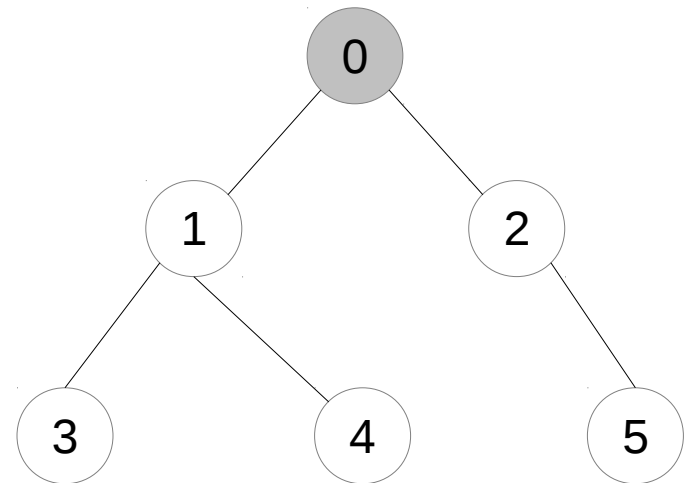
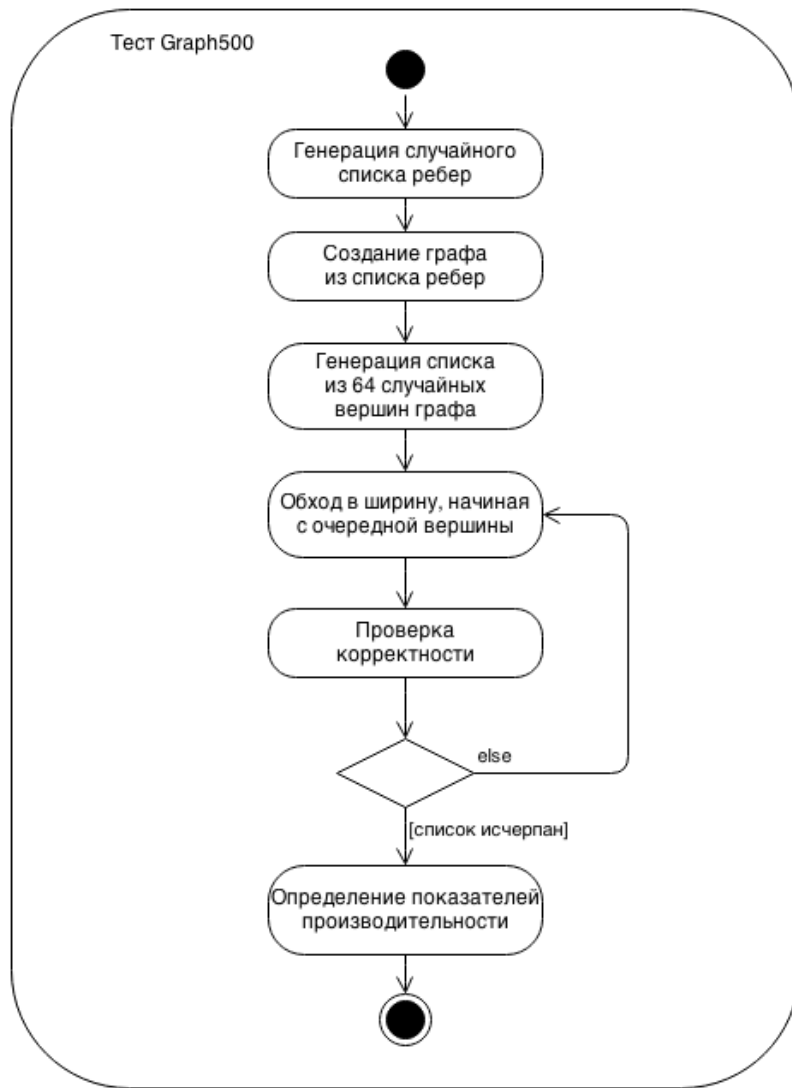
Тест Graph500. Первое ядро



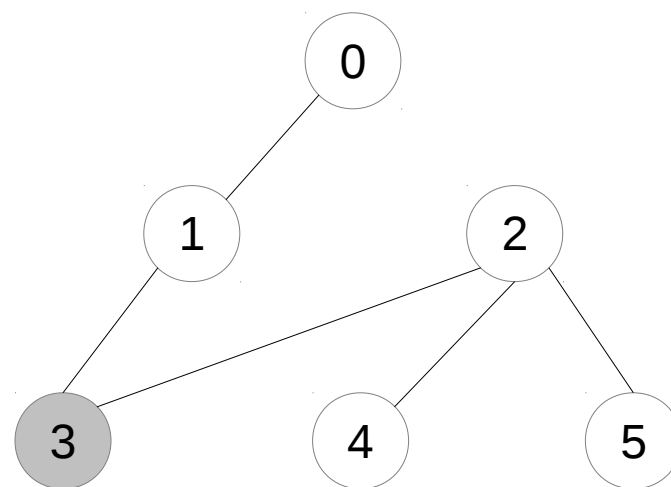
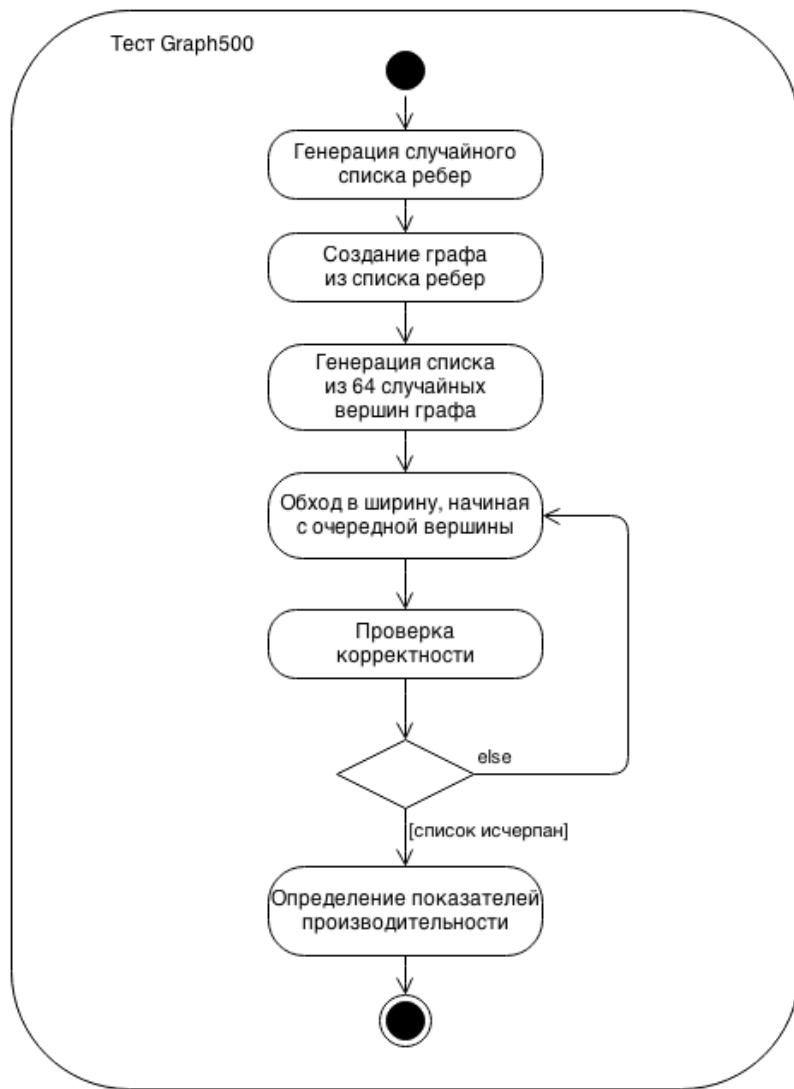
Тест Graph500. Второе ядро



Тест Graph500. Второе ядро



Тест Graph500. Второе ядро



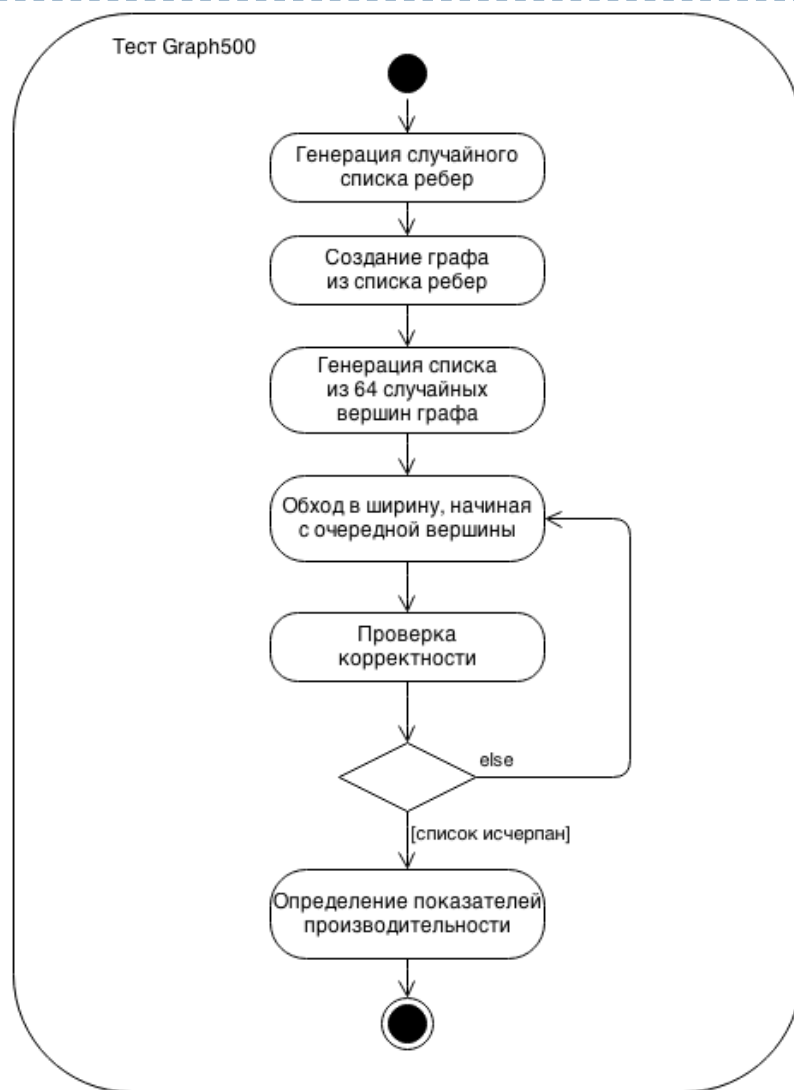
Проверка корректности

Для каждого ключа поиска после завершения обхода графа в ширину нужно проверить, что соблюдены условия:

1. Полученное BFS-дерево является деревом и не содержит циклов.
2. Каждое ребро BFS-дерева соединяет вершины, чей уровень при обходе в ширину различается ровно на 1.
3. Каждое ребро из входного списка ребер соединяет вершины, уровень которых в BFS-дереве различается не более чем на единицу или же обе эти вершины не включены в BFS-дерево.
4. BFS-дерево связывает все вершины данной компоненты связности.
5. Каждая вершина и ее родитель соединены ребром в исходном графе.



Тест Graph500



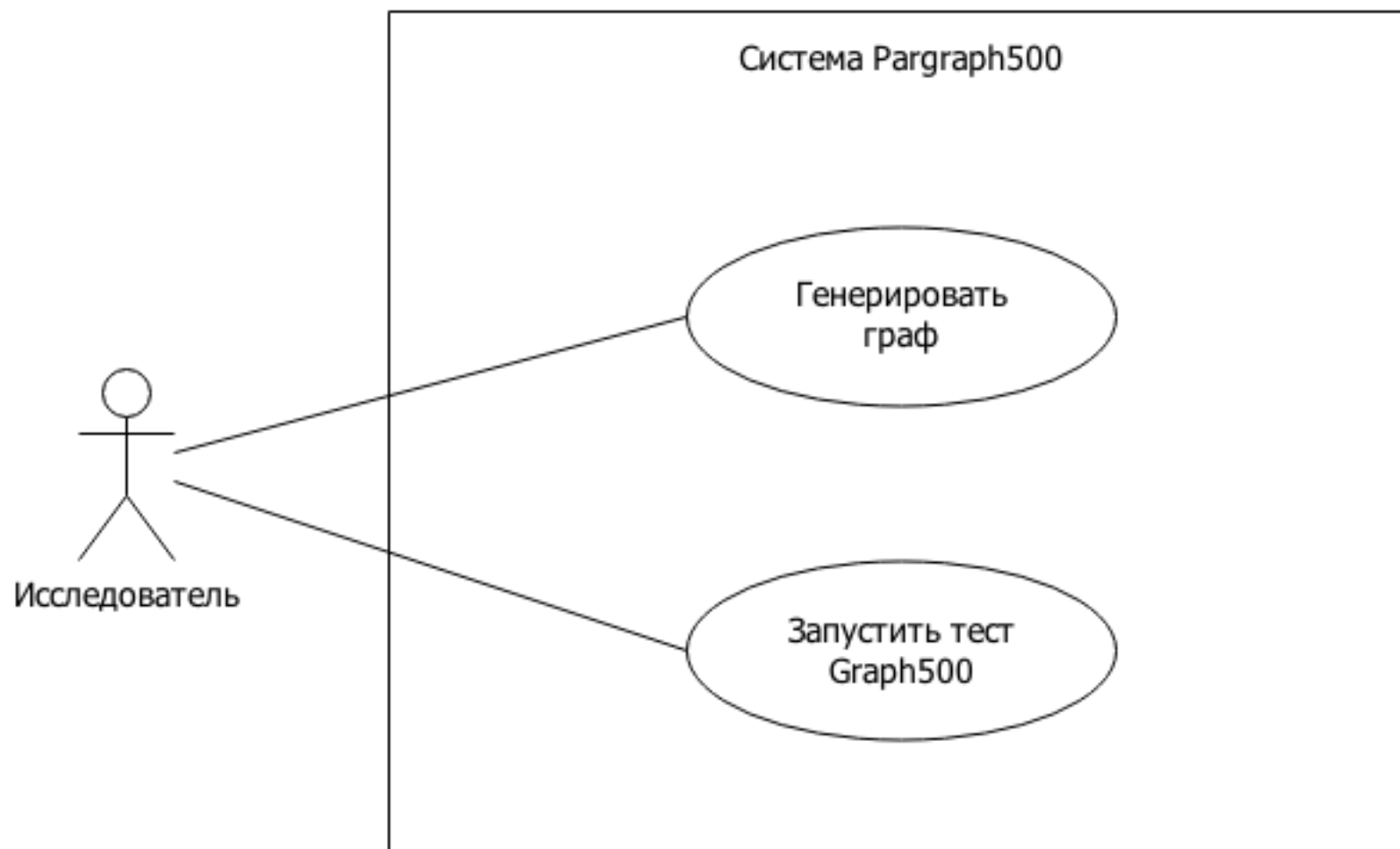
- TEPS (traversed edges per second) — количество пройденных ребер за секунду:

$$TEPS = \frac{m}{time_{K2}}$$

- m — количество ребер в итоговом дереве обхода
- $time_{K2}$ — время работы алгоритма обхода в ширину



Варианты использования системы



Модульная структура

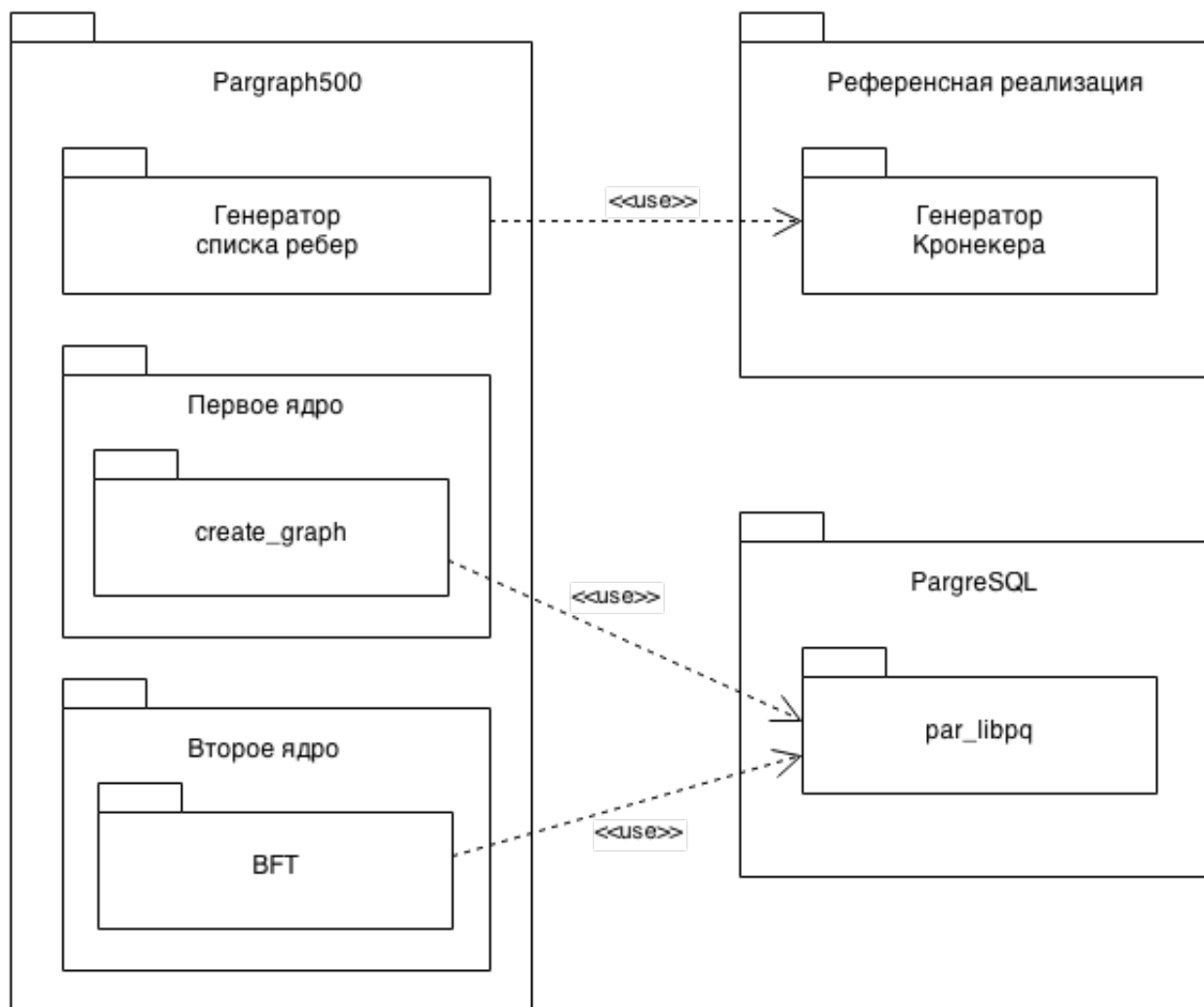


Схема классов

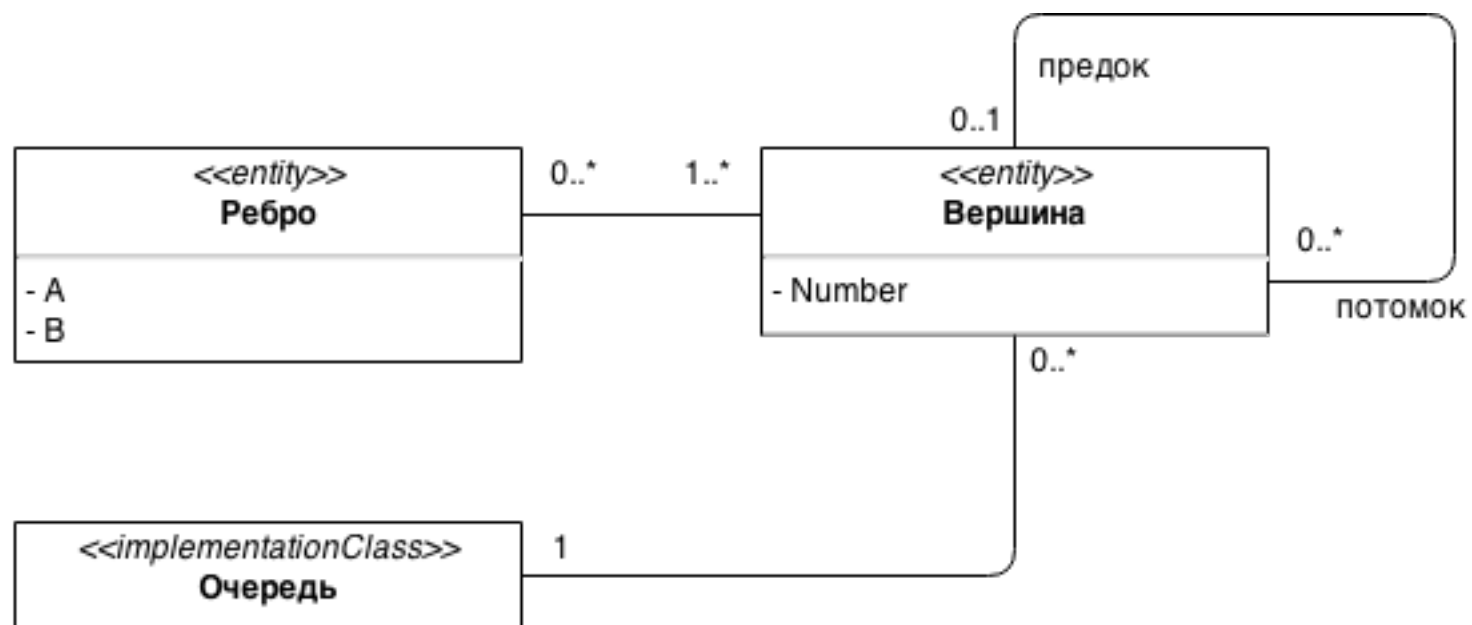
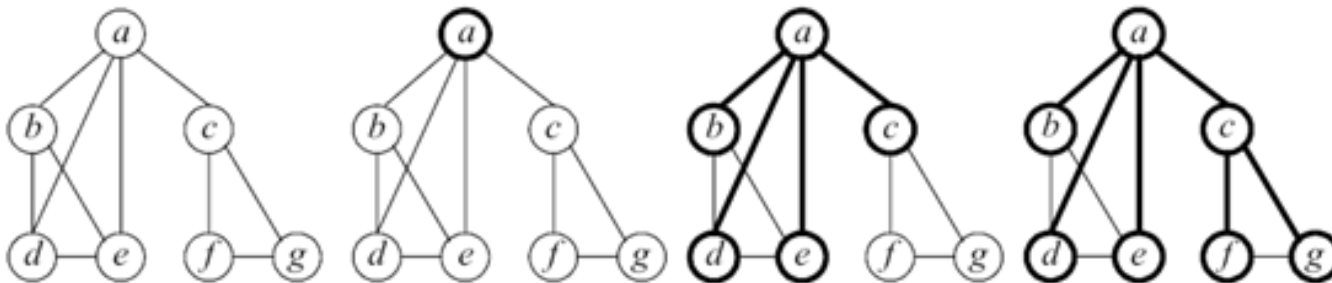


Таблица в PargreSQL

```
create table graph (  
    id bigint,                -- порядковый номер дуги  
    a_id bigint,              -- номер начальной вершины  
    b_id bigint,              -- номер конечной вершины  
    parent bigint,            -- номер вершины родителя a_id  
    queue bigint               -- номер в очереди вершины a_id  
) with (fragattr = id);     -- функция фрагментации
```



Обход в ширину



```

BFS (start_node) {
    for (all nodes i) {
        parent[i] = -1;
    }

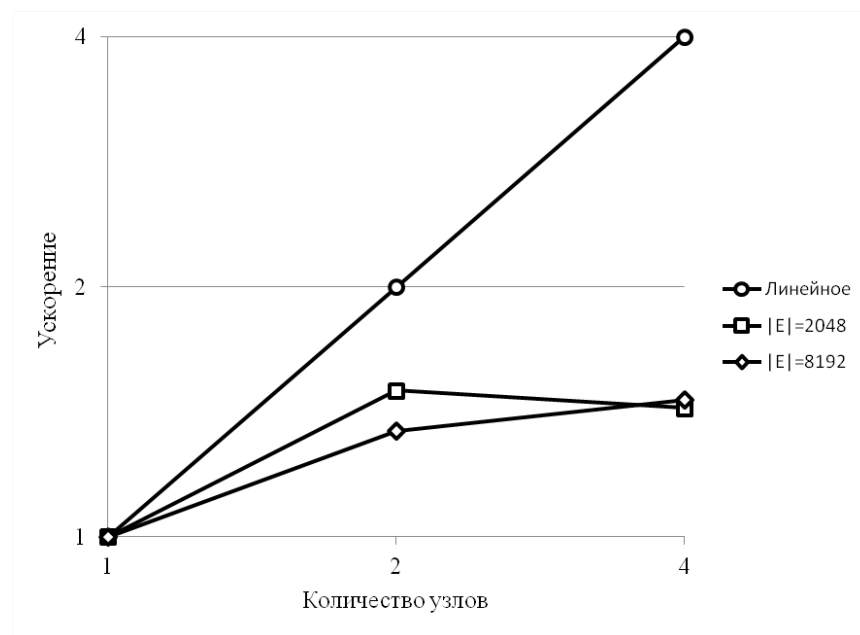
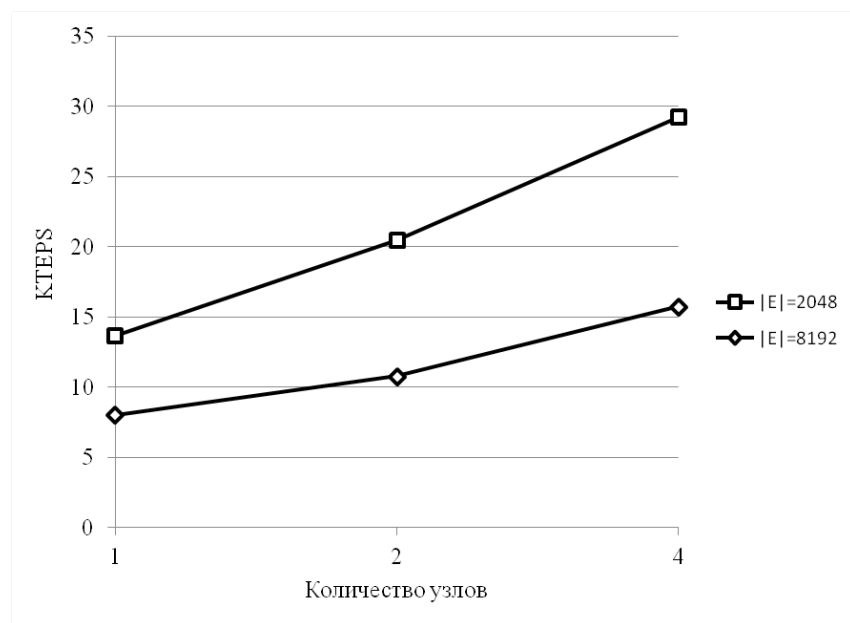
    queue.push(start_node);
    parent[start_node] = start_node;

    while( !queue.empty() ) {
        node = queue.pop();
        foreach(child in expand(node)) {
            if(parent[child] == -1) {
                queue.push(child);
                parent[child] = node;
            }
        }
    }
}

```



Эксперименты



Основные результаты

- ▶ Изучена архитектура параллельной СУБД PargreSQL и спецификация теста Graph500
 - ▶ Разработана схема базы данных для хранения графа и промежуточных данных в соответствии со спецификацией теста Graph500
 - ▶ Выполнено проектирование и разработка алгоритмов на языке SQL, реализующих тест Graph500 для параллельной СУБД PargreSQL
 - ▶ Проведены вычислительные эксперименты на суперкомпьютере «Торнадо ЮУрГУ», исследующие эффективность параллельной СУБД PargreSQL на тесте Graph500
-

