

ОТЗЫВ

официального оппонента Соколинского Леонида Борисовича на диссертационную работу Бучацкого Рубена Артуровича «Метод динамической компиляции SQL-запросов для реляционных СУБД», представленную на соискание ученой степени кандидата технических наук по специальности 2.3.5 – «математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей»

Диссертационная работа Р. А. Бучацкого посвящена исследованию вопросов, связанных с применением методов динамической компиляции для ускорения выполнения аналитических SQL-запросов в реляционных СУБД. Во многих современных реляционных СУБД для исполнения запросов используется итераторная модель, в которой каждый физический оператор реализуется в виде итератора, состоящего из трех методов: open, next и close. Модель итераторов имеет недостатки при выполнении сложных запросов, связанные с неэффективным расходованием ресурсов современных процессоров из-за невозможности применения оптимизаций, в частности оптимизации по встраиванию функций, так как метод next является виртуальным. Современным способом повышения эффективности использования процессора при обработке запросов является динамическая компиляция. Однако существующие методы динамической компиляции не применимы или ограниченно применимы к исполнителю запросов с итераторной моделью. Во многих коммерческих СУБД в оперативной памяти динамический компилятор запросов реализуется на основе модели явных циклов выполнения, который позволяет представить код запроса в виде вложенных циклов. В связи с этим является важной разработка метода динамической компиляции запросов, который будет применим к исполнителю запросов с итераторной моделью. Отсюда вытекает **актуальность** диссертационной работы Р. А. Бучацкого.

Основными результатами диссертационной работы являются следующие:

1. Предложен метод динамической компиляции SQL-запросов в модели явных циклов в оптимизированный машинный код для реляционной СУБД с итераторной моделью.
2. Описаны алгоритмы реализации операторов физической алгебры на базе модели явных циклов, заменяющие соответствующие алгоритмы, реализованные на базе итераторной модели.

3. Предложен метод динамической компиляции арифметических выражений, используемых в запросах.
4. Предложены эвристики стратегии выполнения запроса и метод кэширования скомпилированного кода.
5. Все разработанные методы реализованы в виде программного расширения для открытой объектно-реляционной СУБД PostgreSQL.

Все научные положения, выводы и рекомендации, выносимые на защиту диссертационной работы Р. А. Бучацкого, являются **обоснованными** в полной мере. **Достоверность полученных результатов обеспечивается** практической программной реализацией разработанного метода динамической компиляции и результатами вычислительных экспериментов.

Научная новизна работы заключается в разработке автором оригинального метода трансформации операторов плана запроса из модели итераторов в модель явных циклов с применением динамической компиляции. Кроме того, оригинальным является метод динамической компиляции выражений в SQL-запросах с применением открытой вставки предварительно скомпилированных встроенных функций СУБД.

В качестве **замечаний** к работе, не снижающего ее общего высокого уровня, следует отметить следующее:

1. Термин «резидентная СУБД», введенный диссертантом на стр. 5 диссертации, в русскоязычной научной литературе не используется. В англоязычной научной литературе термин «resident DBMS» также отсутствует. Вместо него используются термины «memory-resident DBMS» и «in-memory DBMS». Следовательно, термин «резидентная СУБД» является жаргонизмом. Вместо него в научных трудах следует использовать термин «СУБД в оперативной памяти».
2. Модель явных циклов предполагает направление процедурных вызовов снизу-вверх по дереву запроса, однако в предложенной реализации в бинарной операции соединения метод `finalize()` для синхронизации входных потоков данных вызывает функцию `main()` дочернего узла, что нарушает концептуальную стройность.

3. В разделе 2.1.3 диссертации описывается механизм прерывания обработки запроса в модели явных циклов. Термин «прерывание» в программировании предполагает наличие «обработчика прерываний», что не соответствует контексту использования термина «прерывание» в диссертации. В этом случае лучше использовать термин «завершение».
4. В разделе 2.1.4 диссертации описывается процесс декомпозиции алгоритмов реляционных операторов модели Volcano в модель явных циклов. Термин «декомпозиция» обычно подразумевает разделение алгоритма на некоторые части, которые затем включаются в соответствующий алгоритм модели явных циклов. На самом деле речь идет о написании этих алгоритмов заново. Поэтому в данном контексте следует вместо «декомпозиции» использовать термин «замена».
5. На стр. 77 диссертации читаем «...метаинформация загружается только в состоянии 1 — HEAVY_FILTER. Для достижения этого состояния запрос должен пройти два эвристических фильтра», однако на рис. 2.9 состоянию «1 HEAVY_FILTER» предшествует только один фильтр «0 LIGHT_FILTER».
6. Описание диаграммы состояний конечного автомата, приведенной в диссертации на стр. 77 (рис. 2.9) и в автореферате на стр. 17 (рис. 3), требует дополнительных разъяснений:
 - a. На стр. 76-77 указано «Пунктирные линии, соединяющие некоторые состояния, означают, что переход происходит только в случае ошибки». О какой ошибке идет речь?
 - b. В следующем предложении говорится, что «Состояния с пунктирными линиями указывают, что кэшированный запрос может продолжить выполнение из этих состояний после того, как его метаинформация была загружена из подсистемы кэширования планов», однако на диаграмме отсутствует путь из состояния с пунктирными линиями «5 ALWAYS_COMPILE» в состояние «6 CACHE_INIT», предусматривающее помещение плана запроса в кэш. Если же это ошибка, и линии у «5 ALWAYS_COMPILE» должны быть сплошными, возникает другой вопрос: почему планы, которые всегда компилируются, не помещаются в кэш?
 - c. Состояние «2 MEASURE_INTERPRETER» обведено пунктирными линиями и имеет циклическую ссылку, замыкающую это состояние на себя.

В соответствии с описанием, процитированным в отзыве выше (пункт «b»), это не поддается рациональному объяснению.

- d. Почему состояния «4 ALWAYS INTERPRETER» и «6 CACHE_INIT» обведены пунктирными линиями? Это не согласуется с описанием, процитированным в отзыве выше (пункт «b»).
7. На стр. 78 диссертации читаем «Если улучшение не достаточно, то при последующих выполнения *этот* запрос будет интерпретироваться» (курсив мой). Как формально определяется эквивалентность планов запросов в описываемом подходе?
8. При применении эвристик стратегии выполнения запроса (раздел 2.3 диссертации) сравниваются время выполнения динамически скомпилированного кода (T_E) со временем интерпретатора (T_I). Эти времена зависят не только от размеров входных отношений, но и от распределения значений атрибутов, которое может меняться со временем. Как этот фактор учитывается для кэшированных запросов?
9. На стр. 81 диссертации указано «Для управления структурой кэша запросов был выбран алгоритма кэширования (вытеснения кэша) LFU». Следует отметить, что при использовании стратегии вытеснения LFU кэш может засоряться ранними часто используемыми запросами при смене профиля нагрузки. Эти более неиспользуемые запросы могут оставаться в кэше сколь угодно долго. Следует обратить внимание на более продвинутые стратегии класса LRU-K и LFU-K.
10. Разделы 3.1 «Интерпретатор запросов СУБД PostgreSQL» и 3.2 «Компиляторная инфраструктура LLVM» следовало перенести в главу 1.
11. Разделы 3.3.1-3.3.5 перегружены трудночитаемыми псевдокодами. Следовало вынести эти псевдокоды в приложение.

Указанные замечания не носят принципиальный характер и не умаляют большой значимости результатов, полученных Р. А. Бучацким в ходе диссертационной работы.

Диссертационная работа в полном объеме соответствует требованиям ВАК РФ, предъявляемым к диссертациям на соискание учёной степени кандидата технических наук. Тематика и содержание диссертации соответствует паспорту научной специальности 2.3.5 – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей».

На основании анализа текстов диссертации и автореферата можно сделать следующее **заключение**. Диссертационная работа Р.А. Бучацкого представляет собой законченную научно-квалификационную работу, в которой решена задача применимости метода динамической компиляции к реляционной СУБД с итераторной моделью для ускорения выполнения запросов. Диссертация в полной мере отвечает требованиям, предъявляемым к кандидатским диссертациям Положением о порядке присуждения ученых степеней, а ее автор заслуживает присуждения учёной степени кандидата технических наук по специальности 2.3.5 – «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей».

Официальный оппонент
доктор физико-математических наук,
профессор, проректор по информатизации
ФГАОУ ВО «Южно-Уральский
государственный университет
(национальный исследовательский
университет)»,
Российская Федерация, 454080, г. Челябинск,
проспект им. В. И. Ленина, 76

Л. Б. Соколинский
23 ноября 2022 г.