

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
Информатики и радиоэлектроники
Кафедра инженерной психологии и эргономики

На правах рукописи

УДК004.658

Козярский
Юрий Викторович

СИСТЕМА ОБРАБОТКИ И ХРАНЕНИЯ ДАННЫХ ПОВЫШЕННОЙ
ЭФФЕКТИВНОСТИ

Автореферат на соискание академической степени
магистра технических наук

1-23 80 08 Психология труда, инженерная психология, эргономика

Магистрант Ю.В. Козярский

Научный руководитель
В.В. Мазюк, кандидат
технических наук, доцент

Заведующий кафедрой ИПиЭ
К.Д. Яшин, кандидат
технических наук, доцент

Нормоконтролер
Т.В. Гордейчук,
ассистент кафедры ИПиЭ,
магистр технических наук

Минск 2016

ВВЕДЕНИЕ

Данная диссертация на соискание степени магистра технических наук посвящена оптимизации хранения данных уже существующей системы.

Наиболее важным при написании больших, долгосрочных продуктов является продумывание архитектуры. Грамотная реализация этого дает более гибкий функционал, возможность масштабирования и разрешения проблем с производительностью.

Но не всегда можно спрогнозировать проблемы, которые возникнут через годы, или при реализации продукта. Долгое время можно игнорировать их присутствие, но чем больше программный продукт развивается, тем четче видны дефекты архитектуры. И тогда программистам приходится признать и разрешать те ошибки, которые они допустили в проектировании. Понятно, что любой программист хочет взять и написать все заново, так как он уже больше знает как в недостатках текущей реализации, так и просто в понимании процесса программирования.

Это первое что хочется сделать и оно очень заманчиво, но редко это правильное решение. Поэтому самое главное в проектировании новой системы это найти середину между новыми плюсами и затраты времени программистов на реализацию всей системы в целом, с использованием уже имеющихся наработок.

Целью работы является исследование задач возникающих в процессе развития уже функционирующей системы обработки и сохранения данных, поступающих из многих источников с большой интенсивностью. В связи с возрастанием объема обрабатываемых данных и увеличением количества сервисов обрабатывающих эти данные прежняя система не справляется со своими задачами. В работе необходимо разрешить возникающие задачи по развитию системы.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Целью данного исследования является построение надёжной отказоустойчивой системы хранения данных, на основе текущей архитектуры. Задачи данного исследования включают следующее:

- обзор теории по выбранной теме исследования;
- исследование и поиск узких мест старой архитектуры;
- практическая реализация изменений с минимальными затратами;
- испытание внесенных изменений.

Актуальность исследования состоит в постоянно растущих объемах данных и необходимости их постоянной структуризации, повышений требований к производительности программных систем, а также с внедрением самых современных технологий, повышающих их эффективность.

Материалы настоящей магистерской диссертации были доложены на 51-й научно-технической конференции студентов, магистрантов и аспирантов БГУИР в апреле 2015 года.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Диссертация состоит из трёх глав.

В первой главе приведён обзор научно-технической литературы по предмету диссертации, рассматривается понятие архитектуры программы, а также основные принципы разработки и проектирования сложных компьютерных систем, приводятся наиболее известные способы командной разработки, обсуждается плюсы и минусы сервис-ориентированной архитектуры с точки зрения программиста. Также рассматриваются основные современные системы хранения данных на собственных серверных машинах под управлением Microsoft Server 2008 и выше (не рассматривается хранение данных с помощью облачных сервисов).

Поднимаются проблемы и преимущества хранения данных в классических реляционных базах данных (MySQL, Oracle, SQLite, PostgreSQL и т.д.) и в последнее время все более и более популярных нереляционных, или как их часто называют NoSQL (Memcached, LiveDB, Redis, Tokyo Cabinet и т.д.). В конце основе обзора литературы приводятся выводы о возможностях применения какой-либо из рассмотренных технологий в нашем проекте.

Вторая глава содержит описание и назначение системы предотвращения утечек, рассматривается разработка эргономичного интерфейса для системы. Рассматривается программная архитектура системы, исследуется каждый сервис программы, его требования к производительности и эргономичности, а также соответствия им. Производится поиск наиболее проблемных участков системы, улучшение производительности которых необходимо. Предпринимаются попытки оценить возможные способы изменения ошибочных архитектурных решений, а также рассматриваются вопросы планового переделывания модулей, без остановки выхода версий продукта на долгое время. В конце главы выработывается список необходимых изменений в продукте с выставленным приоритетом, которые сформулированы как отдельные задачи и не зависят друг от друга.

Третья глава описывает конкретные шаги, связанные именно с планом, который мы наметили в предыдущей главе. Естественно на реальном проекте разработки по улучшению архитектуры ведутся сразу в нескольких направлениях. Но поскольку проект является коммерческим, то не все части его изменений можно отобразить в данной работе. Поэтому приведен конкретный пример улучшения сделанного мной, состоящего в ускорении обработки некоторых запросов к базе данных в десятки раз, и построен на основе его графический интерфейс. Создано специальное тестовое приложения для доказательства проделанной оптимизации. Данная доработка будет добавлена в версию нашего продукта, которая выйдет в следующем году.

ЗАКЛЮЧЕНИЕ

В данной работе построена модель работы каждого компонента, рассматриваемой системы, а так же взаимодействия их друг с другом. Сформированы общие требования к каждому компоненту по функциональности и производительности. Найдены наиболее уязвимые места в существующей архитектуре и ее недочеты. В частности, указано на не правильную организацию хранения данных.

Показано, что независимость компонент в данном продукте сильно сказывается на производительности. Найдены возможные улучшения для достижения заложенных новых требований.

Так же показано направление на поэтапное движения к замене старой архитектуры, что бы изменения не были координальны и не приостановили выпуски новых версий продукта на длительный срок. Реализованы некоторые компоненты в рамках предложенной модели.

Исследовано возможное изменение архитектуры. Показанно что перенос атрибутов документов из поискового сервера в саму базу данных, дает значительный прирост в производительности и большие возможности для манипулирования данными. Данные документов переносятся в key-value хранилище, если их размер не превышает нескольких десятков килобайт и в файловую систему в обратном случае. Так же протестировано совмещения таблиц сервера юзеров и основного хранилища данных. Экспериментально показано, что при таком изменении можно значительно сократить количество информации хранящейся в сервере отчетов.

Созданы специальные тестовые данные для данного эксперимента. Показаны минусы и плюсы данноого изменения в архитектуре, и их преимущество над старым решением.. Так же были созданы специальные базы данных для последующего тестирования новых компонент, хотя эти базы еще не полны, но это уже рельные данные, которые можно считать универсальными. Они не сгенерированы, а созданы, благодаря чему можно на них тестировать не просто запросы к базе, а и полнотекстовые поиски, привязки и тд.

Мною была выполнена следующая работа: оптимизация структуры БД для ускорения запросов от сервера отчетов, библиотека обертка над SQL синтаксисом для сервера хранения.

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

[1-А.] Козьярский, Ю.В. Система обработки и хранения данных повышенной эффективности / Ю.В. Козьярский // Узкие места реляционных баз данных: Тезисы докл. к конф. – Минск, 2015.

Библиотека БГУИР