

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА ПРИЛОЖЕНИЯ KVAZARONLINE С
ВЫЧИСЛИТЕЛЬНЫМ МОДУЛЕМ НА C++
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Аношкина Андрея Алексеевича

Научный руководитель

зав. каф. техн. пр., к. ф.-м. н., доцент _____

И. А. Батраева

Заведующий кафедрой

к. ф.-м. н., доцент _____

С. В. Миронов

ВВЕДЕНИЕ

Kvazar — это программный комплекс, разработанный научной группой «Glukhova Research Group» для решения задач молекулярного моделирования, состоящих из вычислительного модуля с одноименным названием — Kvazar, и визуализатора — KView, изначально написанных на языке Python2 под ОС Ubuntu.

Ввиду ряда ограничений, таких как привязка к конкретной операционной системе определенной версии и скорость работы программ, написанных на языке Python, научная группа «Glukhova Research Group» поручила обновить программный комплекс, сделав его более доступным для пользователей, а также разработать исполняемый файл для операционной системы Windows. Поэтому на данном этапе было принято решение перенести программный комплекс в формат web-приложения, разделив визуализатор KView и вычислительный модуль Kvazar на клиентскую и серверную части соответственно.

Причиной такого выбора стала распространённость сети Интернет и сервиса World Wide Web (WWW). Несмотря на то что изначально данный сервис и использовался только для связывания и предоставления статической информации, теперь он стал платформой для удалённого использования специальных прикладных программ — *web-приложений*. Так, ранее, прежде чем использовать приложение на локальном компьютере, было необходимо его предварительно устанавливать, а сейчас приложение можно запускать с помощью *web-браузера*: достаточно указать его адрес. При этом само приложение выполняется на удалённом ПК (*сервере*), а пользователь может работать с ним на своём компьютере с помощью web-браузера.

Таким образом, преобразование программного комплекса Kvazar в web-приложение сделает его доступным для эксплуатации с любого устройства, имеющего выход в Интернет и способного открывать любой существующий браузер.

Целью данной работы является разработка ядра web-приложения KvazarOnline на C++.

Ввиду объема работы, было принято решение собрать команду разработчиков и разделить обязанности по разработке.

Для упрощения командной разработки было решено использовать систему контроля версий GitHub.

Для достижения поставленной цели были выявлены следующие задачи.

1. Проанализировать структуру программного комплекса Kvazar и разделить проект на модули.
2. Подобрать стек технологий для разработки отдельных модулей.
3. Разделить обязанности по разработке каждого модуля среди членов команды разработки.
4. Разработать модули ядра проекта на C++, сохранив их функциональность и улучшив скорость, предусмотрев дальнейшую эксплуатацию в качестве серверного приложения.

Структура и объём работы. Выпускная квалификационная работа состоит из введения, 2 разделов, заключения, списка использованных источников и 2 приложений. Общий объем работы — 51 страница, из них 33 страницы — основное содержание, включая 5 рисунков, USB-носитель в качестве приложения, список использованных источников информации — 23 наименования.

Первый раздел «Особенности Kvazar и выбранного стека технологий» посвящен описанию основных определений, используемых в работе, а также теоретической информации, касающейся технологического стека и структуры вычислительного модуля Kvazar.

Второй раздел «Разработка web-приложения» посвящен детальному описанию кодовой базы инструментов разработки.

Особенности Kvazar и выбранного стека технологий

Вычислительный модуль Kvazar

Kvazar — открытый многопроцессорный программный комплекс молекулярного моделирования, который сочетает в себе преимущества наиболее успешных современных пакетов молекулярного моделирования, но вместе с тем обладает гибкой и легко модифицируемой архитектурой.

Задачей программного комплекса «Kvazar» является предоставление набора инструментов, позволяющих исследователю решать конкретную научную задачу на всех её этапах:

1. проектирование геометрии исследуемых объектов;
2. задание периодических граничных и внешних условий;
3. проведение численных экспериментов с помощью различных математических моделей.

Согласно официальному методическому пособию, в программном пакете Kvazar реализованы такие методы молекулярного моделирования, как метод сильной связи (ТВ), метод молекулярной динамики, силовые поля REBO, AIREBO, MARTINI.

Язык C++ для вычислительного модуля

C++ является компилируемым языком программирования высокого уровня со статической типизацией, который подходит для создания самых различных приложений.

Предшественником C++ является язык C, благодаря которому C++ получил одну из его главных особенностей — возможность напрямую работать с памятью, с помощью чего разработчики получают огромные возможности в оптимизации и улучшении производительности кода, написанного на C++.

Так как C++ является компилируемым, компилятор транслирует исходный код на C++ в исполняемый файл, содержащий в себе набор машинных инструкций. Ввиду особенностей различных операционных систем и платформ, исполняемый файл, скомпилированный под одну платформу, не может быть запущен на другой, но в то же время исходный код C++ может быть скомпилирован на любой операционной системе, если не содержит в себе использования библиотек, присущих только определенным платформам, что дает языку C++ переносимость между различными платформами.

Рассмотрим преимущества C++:

— *Высокая производительность*

Ключевым преимуществом C++ является его высокая производительность, высочайшая среди языков высокого уровня. Так, обеспечивается возможность разработки быстрого и эффективного кода. Такая высокая производительность достигается посредством низкоуровневого доступа к памяти и аппаратным ресурсам, которые и дают возможность максимально оптимизировать код и добиться пика производительности.

— *Широкие возможности управления памятью*

Как уже было сказано, C++ предоставляет полный контроль над памятью. Данная особенность дает возможность оптимизировать использование ресурсов и избегать проблем с производительностью, связанных с автоматическим управлением памятью, подобно реализованной в других языках высокого уровня.

— *Кроссплатформенность*

Благодаря возможности компиляции исходного кода практически на любой платформе C++ поддерживает разработку кроссплатформенных приложений.

— *Поддержка многопоточности*

C++ предоставляет различные инструменты для работы с многопоточностью, что в купе с полным контролем над памятью позволяет эффективно распределять задачи между процессорными ядрами и улучшать производительность.

Таким образом, учитывая предъявленные к конечному продукту требования, C++ является наиболее подходящим языком программирования для разработки вычислительного модуля Kvazar.

Библиотека Eigen для операций линейной алгебры

Для реализации некоторых функций в вычислительном модуле Kvazar изначально использовались функции линейной алгебры, предоставляемые библиотекой Python numpy. Для успешного перенесения функционала и экономии времени на разработке, было решено найти подходящую для этого библиотеку, написанную на C++. Такой библиотекой оказалась библиотека Eigen.

Eigen — это библиотека шаблонов C++ для линейной алгебры: матриц, векторов и связанных с ними алгоритмов.

К преимуществам библиотеки относятся:

— *Универсальность*

- Поддерживает матрицы всех размеров, от небольших фиксированного размера до произвольно больших плотных и даже разреженных матриц.
- Поддерживает все стандартные числовые типы, включая комплексные, целые числа, и легко расширяется до пользовательских числовых типов.
- Поддерживает различные матричные разложения и геометрические особенности.

— *Скорость*

- Шаблоны выражений позволяют разумно удалять временные выражения и выполнять отложенное вычисление, когда это уместно.
- Матрицы фиксированного размера полностью оптимизированы: исключается динамическое выделение памяти, а циклы разворачиваются, когда это имеет смысл.

— *Надежность*

- Алгоритмы тщательно отбираются с точки зрения надежности.
- Eigen тестируется с помощью собственного набора тестов (более 500 исполняемых файлов).

Таким образом, библиотека Eigen была взята на замену библиотеке numru, используемой в исходной версии вычислительного модуля Kvazar.

Разработка web-приложения

Был проведен анализ существующей структуры вычислительного модуля. Вычислительный модуль состоит из следующих компонентов:

1. Класс Structure — класс, отвечающий на хранение и операции с некоторыми данными молекулярной структуры, не зависящих от силового поля или метода вычислений.
2. Класс Fileman — класс, отвечающий за обработку файлов.
3. Класс Diesel — класс, отвечающий за выполнение симуляции в режиме вычислений молекулярной динамики.
4. Классы TB, MARTINI, AIREBO и др. — классы, отвечающие за вычисление энергии и действующих сил при определенных условиях, таких как силовое поле и т. д.
5. main — основной файл программы, отвечающий за считывание файла конфигураций и установки параметров отдельных классов.

Более наглядно структура вычислительного модуля представлена на рисунке 1.

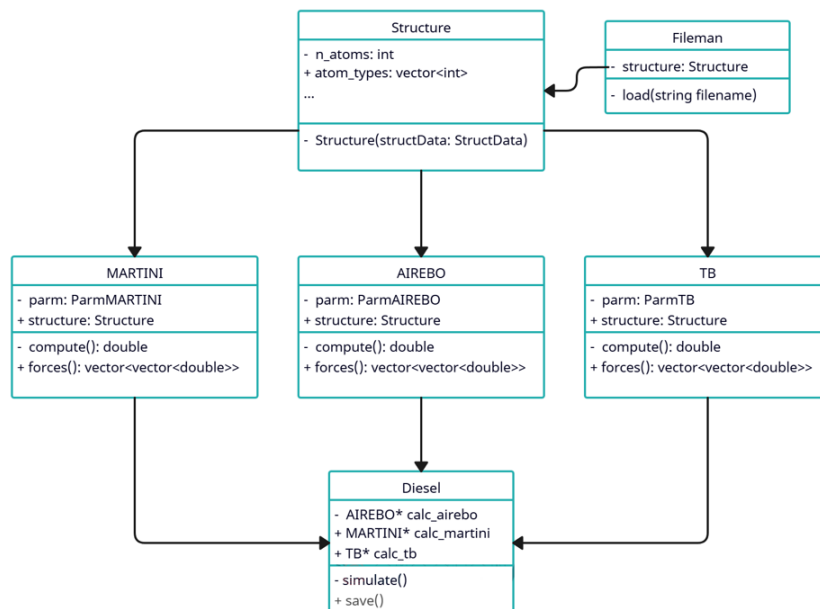


Рисунок 1 – Структура вычислительного модуля Kvazar

Класс Fileman

Для того, чтобы начать работу, необходимо реализовать считывание данных из поступающих файлов, за которое отвечает класс Fileman.

Основными методами класса Fileman являются:

- load
- set_struct
- get_struct

Метод load принимает на вход путь к файлу формата .ACD, после чего считывает данные из него и помещает в класс структуры.

Класс Structure

Класс Structure состоит из следующих основных методов:

- Конструктор класса
- get_full_links
- calc_tors_sequence
- tors_dfs
- get_tors_seq
- get_neighbours
- get_neighbours_
- update_links_params

Конструктор класса представляет собой перенесение информации из заполненных в результате работы функции parse_acd полей структуры StructData и постобработку некоторых из них, а также проверку на корректность полученных параметров. Так, был реализован конструктор класса Structure.

Класс AIREBO

Класс AIREBO отвечает за силовые поля REBO и AIREBO.

Основными методами класса AIREBO являются:

- Конструктор класса.
- preprocessing
- update_neighbours
- init_cutoff
- forces
- getAIREBO_parm
- u_rebo
- u_tors
- u_lj
- compute

Конструктор класса принимает класс `Structure` и список параметров для вычислений. Список параметров различается для `AIREBO` и `REBO`. Конструктор объявляет ряд констант и применяет метод `getAIREBO_parm`, заполняющий параметры для силового поля `AIREBO`, записанные в файл формата `json`.

Сбор данных из файла формата `json` был произведен с помощью библиотеки `nlohmann::json`.

Также были реализованы методы `preprocessing`, запускаемый в конструкторе класса и вычисляющий массы атомов, `init_cutoff`, инициализирующий механику отсечения, и `update_neighbours`, обновляющий данные о соседях.

Одним из основных методов класса является метод `compute`, вычисляющий энергию молекулярной структуры.

Класс TB

Класс `TB` отвечает за метод сильной связи.

Основными методами класса `TB` являются:

- Конструктор класса.
- `preprocessing`
- `getTB_parm`
- `forces`
- `compute_erep`
- `compute_ebond`
- `compute`

Конструктор класса инициализирует структуру, параметры и запускает `preprocessing`.

Класс MARTINI

Класс `MARTINI` отвечает за крупнозерновое моделирование.

Основными методами класса `MARTINI` являются:

- Конструктор класса
- `preprocessing`
- `getMARTINI_parm`
- `compute`
- `u_angle`
- `u_charge`
- `u_lj`

- `u_tors`
- `u_bond`
- `forces`

Конструктор класса инициализирует список параметров, получает параметры MARTINI, обновляет конфигурацию связей и запускает preprocessing.

Метод preprocessing вычисляет массу отдельных атомов и проверяет, являются ли они белком.

Был реализован метод forces для вычисления сил, действующих на атомы молекулярной структуры, а также вспомогательные методы.

Метод compute вычисляет энергию молекулярной структуры как сумму составляющих энергий молекулярной структуры, зависящих от параметров, полученных при инициализации класса.

Класс Diesel

Класс Diesel предназначен для вычисления в режиме молекулярной динамики.

Основными методами класса Diesel являются:

- Конструктор класса.
- Группа методов инициализаторов.
- Группа методов-геттеров.
- Группа методов, отвечающих за интеграторы движения.
- `save`
- `get_forces`
- `simulation`

Сам класс Diesel содержит среди своих полей один из классов: AIREBO, MARTINI, TB — для проведения вычислений в соответствии с определенными конфигурациями.

Был разработан метод save для сохранения данных о состоянии молекулярной структуры в процессе симуляции.

Были разработаны интеграторы движения leapfrog_berendsen_step, leapfrog_nosehoover_step, dissipative_step, leapfrog_step, velocity_verlet_step, velver_nosehoover_step, leapfrog_lincs_step.

Был реализован метод get_forces, позволяющий получать значения сил, действующих на атомы молекулярной структуры, в зависимости от конфигураций.

Был реализован метод `simulation` для процесса симуляции.

Главный файл `main.cpp`

Для реализации основного файла программы, необходимо реализовать считывание файла конфигураций. Файл конфигурации содержит данные, записанные в формате `yaml`. Был разработан соответствующий считыватель.

Была разработана функция `single_point` для реализации соответствующего метода.

Была разработана функция `molecular_dynamic` для реализации метода молекулярной динамики с использованием одного из силовых полей.

Интеграция с серверной частью

С целью объединения с серверной частью были разработаны функции, принимающие данные, приходящие с клиентской части и возвращающие отдельную структуру.

ЗАКЛЮЧЕНИЕ

В процессе выполнения выпускной квалификационной работы были:

- Проанализирована структура программного комплекса Kvazar и разделена на модули.
- Подобран стек технологий.
- Проанализирована структура вычислительного модуля Kvazar.
- Реализованы отдельные модули ядра проекта на C++ и проведена интеграция с серверной частью.

Таким образом, все задачи были выполнены. В результате работы было разработано ядро web-приложение KvazarOnline на C++.