|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01** Информатика и вычислительная техника

**Отчет**

**по лабораторной работе № 2**

**Название:** Построение IDEF0-модели AS-IS функционирования системы

**Дисциплина:** Теория систем и системный анализ

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | ИУ6-73Б |  | В.К. Залыгин |
|  | (Группа) | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |
| Преподаватель |  |  | Д.А. Миков |
|  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2025

**Цель работы**

Овладение методологией IDEF0 для функционального моделирования сложных систем (система планирования путешествий с персональными рекомендациями).

**Задание**

Разработать структурно-функциональную модель системы на основе методологии IDEF0.

**Исходные данные**: исследуемая система выбирается индивидуально, по согласованию с преподавателем. Рекомендуется выбрать систему, которая исследовалась при выполнении лабораторной работы №1.

**ХОД РАБОТЫ**

**1 Выбор системы, описание предметной области**

Система: оптимизирующий компилятор (далее – “Система”).

Предметная область: приложения для сборки кодов программ в исполняемые файлы.

Субъект анализа: оптимизирующий компилятор.

Объект анализа: пользователь ПО.

Точка зрения: разработчик.

Цель: оптимизация и компиляция программ, составление ассемблерных файлов.

1. **Моделирование контекстной диаграммы**

Была построена таблица 1, содержащая выявленные недостатки.

Таблица 1 – Выявленные недостатки

|  |  |
| --- | --- |
| Узкое место | Способ исправления |
| Оптимизация не учитывает возможность многократных проходов по АСТ | Сделать обратную связь для многократных проходов при оптимизации |
| Входной файл может содержать некорректную программу | Добавить проверку правильности входного файла с программой. |

Контекстная диаграмма, представленная на рисунке 1, позволяет выявить ключевые потоки данных и определить возможные узкие места, связанные с производительностью и полнотой информации.

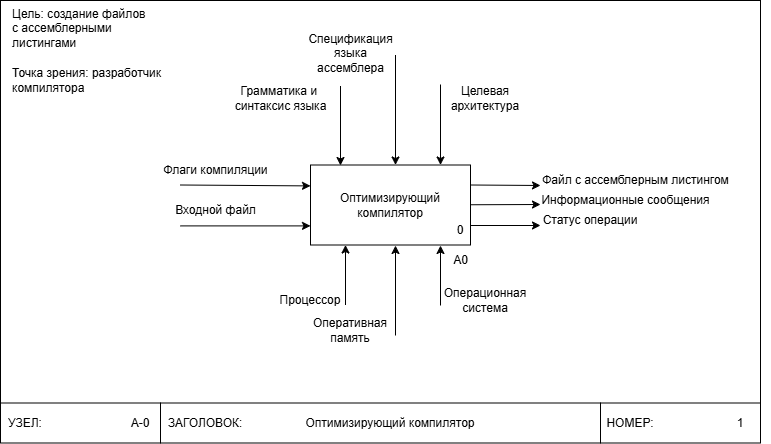


Рисунок 1 – Контекстная диаграмма

1. **Моделирование диаграмм декомпозиции**

Целью моделирования является формализация и описание процессов функционирования оптимизирующего компилятора.

Построение иерархии диаграмм позволяет глубже понять последовательность действий, выявить зависимые компоненты и определить возможные узкие места, влияющие на точность рекомендаций и производительность системы.

На рисунке 2 представлена диаграмма декомпозиции контекстной диаграммы A0.

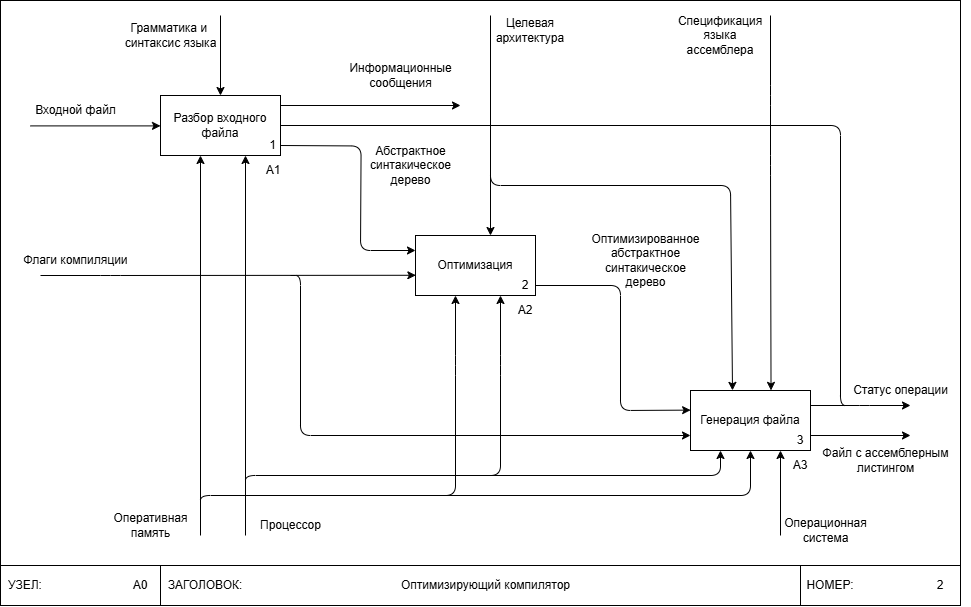


Рисунок 2 – Диаграмма декомпозиции контекстной диаграммы А0

На рисунке 3 представлена диаграмма декомпозиции A1 — “Разбор входного файла”, где решена проблема отсутствия проверок – добавлен блок, осуществляющий проверки файла.

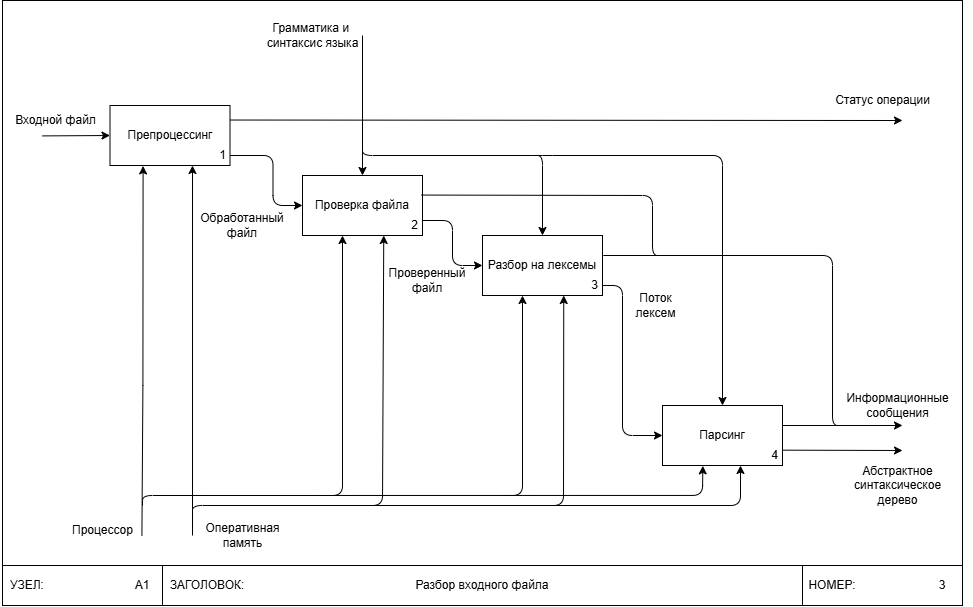


Рисунок 3 – Диаграмма декомпозиции А1

На рисунке 4 представлена диаграмма декомпозиции A2 — “Оптимизация”. Для устранения узкого места добавлены обратные связи между различными блоками оптимизаций, которые помогают достичь наибольшей оптимизации программы путем многократной синергии.

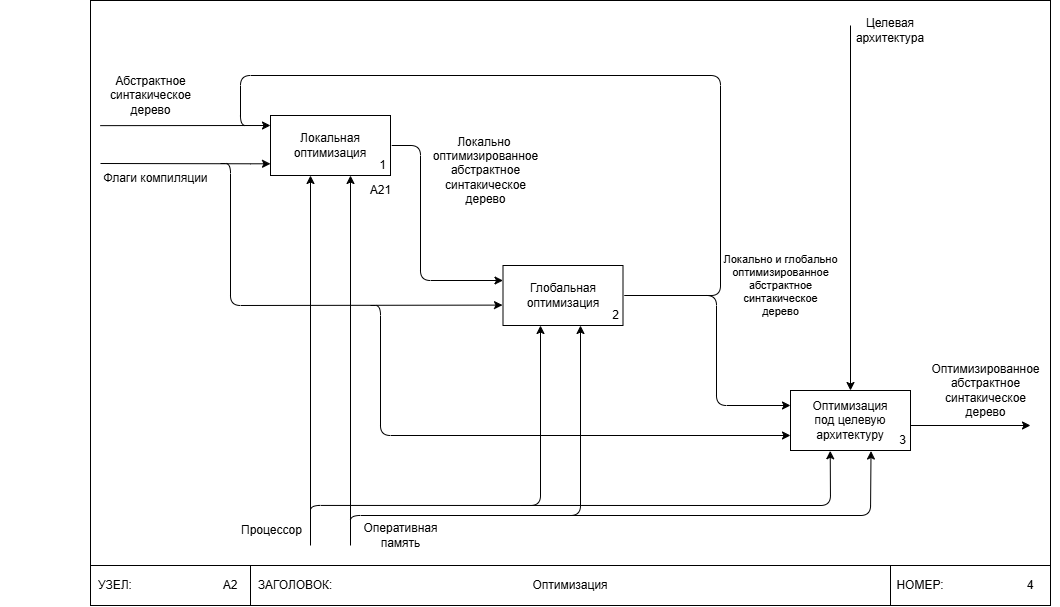


Рисунок 4 – Диаграмма декомпозиции А2

На рисунке 5 представлена диаграмма декомпозиции A3 — “Генерация файла”.

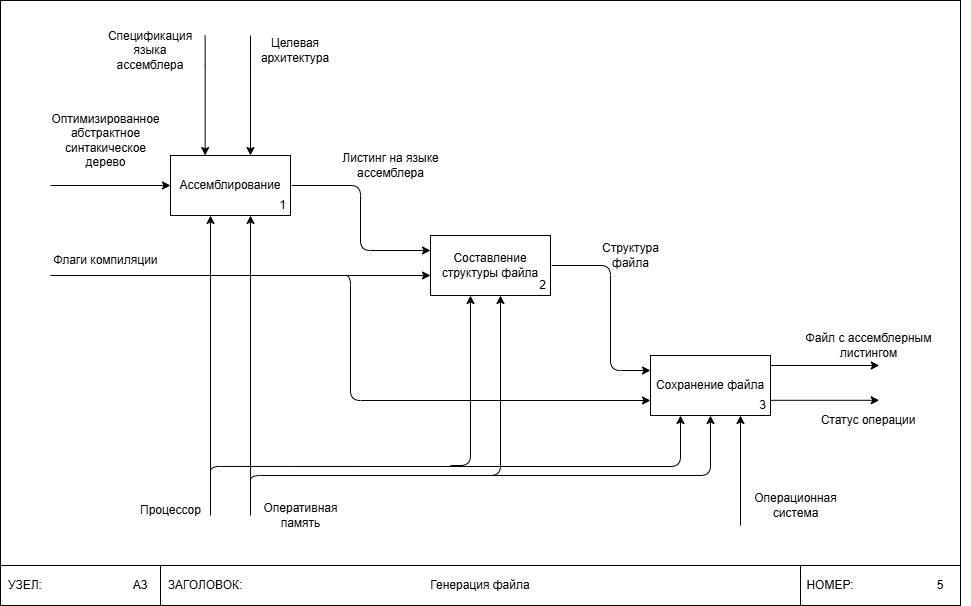


Рисунок 5 – Диаграмма декомпозиции А3

На рисунке 6 показана диаграмма декомпозиции A21 — “Локальная оптимизация”. Диаграмма позволяет уточнить способ проведения локальных оптимизаций. В данной диаграмме также произведены изменения для увеличения степени оптимизации путем многократных проходов (обратная связь от блока 3 до блока 1).

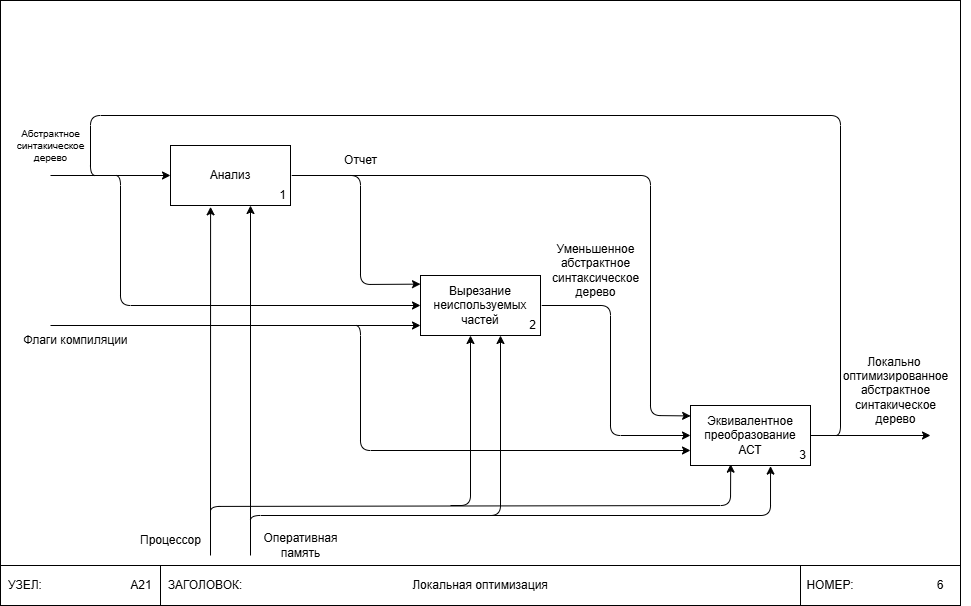


Рисунок 6 – Диаграмма декомпозиции А21

**ВЫВОД**

В ходе выполнения лабораторной работы была разработана усовершенствованная структурно-функциональная модель системы рекомендаций туристических предложений, встроенной в мобильное приложение для планирования путешествий, в нотации IDEF0 (версия TO-BE). Модель отражает оптимизированную архитектуру системы, построенную на основе анализа текущего состояния (AS-IS) и устранения выявленных недостатков.

В процессе работы были выполнены следующие шаги:

— проведён анализ исходной модели и определены ключевые проблемы, связанные с избыточной сложностью входных потоков и недостаточной интеграцией обратной связи;

— построена обновлённая контекстная диаграмма A-0, в которой объединены информационные потоки и укрупнены механизмы обработки данных, что позволило повысить наглядность и сократить дублирование функций;

— выполнена декомпозиция уровня A0 на четыре оптимизированных процесса: сбор данных пользователя, обработка данных и формирование рекомендаций, предоставление рекомендаций пользователю, сбор обратной связи и адаптивное обучение модели;

— построены диаграммы первого уровня (A1–A4) и уточнённая диаграмма второго уровня (A22) с учётом изменений в логике взаимодействия модулей и автоматического обновления пользовательских профилей;

— добавлен блок проверки корректности входного файла;

— предложены меры по повышению эффективности оптимизаций, в частности добавлены дополнительные обратные связи в рамках диаграмм “Оптимизации” для повышения качества оптимизации.

Разработанная модель TO-BE позволила формализовать улучшенную структуру системы, повысить связность процессов и адаптивность алгоритмов рекомендаций.

Результаты моделирования демонстрируют повышение логической согласованности и управляемости системы, а также создают основу для дальнейшей автоматизации и масштабирования функционала рекомендательного модуля.

**КОНТРОЛЬНЫЕ ВОПРОСЫ**

**1. В чем преимущество структурного подхода при исследовании**

**сложных систем?**

Структурный подход позволяет рассматривать сложную систему как совокупность взаимосвязанных функциональных блоков.  
Каждый блок выполняет определённую функцию и имеет чётко определённые входы, выходы, управляющие воздействия и механизмы.  
Такой подход облегчает анализ, проектирование и оптимизацию системы, так как позволяет:

* разбить сложную задачу на более простые части;
* выявить связи между элементами;
* понять, где находятся узкие места и дублирование функций;
* упростить модификацию и документирование.

По сути, структурное моделирование помогает «разобрать систему на слои» и увидеть её логику без избыточных деталей.

**2. Укажите назначение методологии IDEF0.**

Методология IDEF0 предназначена для описания, анализа и документирования функций сложных систем. Она используется для формализации бизнес-процессов и технологических схем в удобном для анализа виде.

IDEF0 показывает, что система делает, что ей нужно для выполнения функций, что она производит и что управляет её поведением.

Основная цель — создать наглядную и логически непротиворечивую модель, которая облегчает анализ текущего состояния и разработку улучшенной версии.

**3. Поясните назначение моделей «как есть» и «как должно быть».**

Модель «как есть» (AS-IS) описывает текущее состояние системы — как она функционирует на данный момент, с существующими процессами, взаимодействиями и проблемами. Она нужна для выявления узких мест, избыточных потоков и неэффективных операций.

Модель «как должно быть» (TO-BE) представляет улучшенный вариант системы, в котором устранены недостатки и реализованы оптимизации. Эта модель используется для проектирования новой архитектуры, планирования автоматизации и внедрения изменений.

Построение обеих моделей позволяет провести анализ перехода от текущего состояния к целевому.

**4. Что представляет собой IDEF0-модель?**

IDEF0-модель — это графическое и текстовое представление функциональной структуры системы.  
Она состоит из набора взаимосвязанных диаграмм, каждая из которых описывает определённый уровень детализации.  
Каждый функциональный блок в IDEF0 имеет стандартную структуру:

* входы (Inputs) — что поступает на обработку,
* выходы (Outputs) — что создаётся в результате,
* управления (Controls) — что регулирует выполнение функции,
* механизмы (Mechanisms) — кто или что выполняет функцию.

Такая модель позволяет единообразно описывать процессы любой сложности.

**5. Зачем при построении IDEF0-модель необходимо указывать цель**

**моделирования и точку зрения?**

Цель моделирования определяет зачем создаётся модель — например, для оптимизации процессов, анализа узких мест или автоматизации функций.

Точка зрения задаёт границы и уровень детализации модели — с позиции кого рассматривается система (пользователя, разработчика, аналитика, руководства).

Без этого модель может оказаться избыточной или, наоборот, неполной:

* цель обеспечивает направленность и логическую завершённость описания;
* точка зрения помогает определить, какие функции и связи действительно важны для анализа.

**6. Перечислите и поясните ограничения сложности IDEF0-диаграмм.**

Методология IDEF0 предусматривает несколько ограничений для повышения читаемости и структурности диаграмм.

1. Количество блоков на диаграмме — от 3 до 6 (максимум 7), чтобы не перегружать схему.
2. Количество входящих и исходящих стрелок — должно быть разумным, чтобы диаграмма оставалась понятной.
3. Иерархичность — каждая диаграмма может быть декомпозирована на подуровни (A0, A1, A2…), но уровень детализации должен быть оправдан задачей.
4. Единый контекст — каждая диаграмма описывает только один аспект системы.
5. Однозначность связей — все стрелки и блоки должны иметь уникальные имена и понятные назначения.

Эти ограничения делают модель компактной, логичной и пригодной для анализа.

**7. Какие диаграммы входят в состав IDEF0-модели?**

Полноценная IDEF0-модель включает несколько типов диаграмм.

1. Контекстная диаграмма (A-0) — верхний уровень, показывающий систему в целом, её границы, входы, выходы, управляющие и механизмы.
2. Диаграмма первого уровня (A0) — раскрывает основную функцию на несколько крупных подпроцессов.
3. Диаграммы декомпозиции (A1, A2, A3, …) — детализируют каждый подпроцесс, показывая внутреннюю структуру и взаимодействия.
4. Диаграммы второго и последующих уровней (A21, A31, …) — применяются при необходимости более глубокого анализа конкретных функций.

Такое многоуровневое представление позволяет описывать систему «сверху вниз» — от общего к частному.