

ЛЕКЦИЯ 18

Тема: Типы информационных потоков. Case-средства проектирования информационных потоков данных

Цель: изучить основные методы проектирования потоков.

1. SADT (Structured Analysis and Design Technique) – моделированию процессов в различных нотациях IDEF0

2. DFD (Data Flow Diagrams) – диаграммы потоков данных- используются для описания структуры проектируемой системы

ERD (Entity-Relationship Diagrams) – диаграммы "сущность-связь" описания модели данных логического и физического уровней.

Результат структурного проектирования — иерархическая структура ПС.

Действия структурного проектирования зависят от типа информационного потока в модели анализа.

ТИПЫ ИНФОРМАЦИОННЫХ ПОТОКОВ

Типы информационных потоков бывают 2 типов:

1) ПОТОК ПРЕОБРАЗОВАНИЙ;



Рис. 5.1. Элементы потока преобразований

2) ПОТОК ЗАПРОСОВ.



Рис. 5.2. Структура потока запроса

1. в потоке преобразований выделяют 3 элемента:

- Входящий поток,

- Преобразуемый поток

- Выходящий поток.

2. Потоки запросов имеют в своем составе особые элементы — запросы. Назначение элемента-запроса состоит в том, чтобы запустить поток данных по одному из нескольких

путей. Анализ запроса и переключение потока данных на один из путей действий происходит в центре запросов

Проектирование для потока данных типа преобразование

Шаг 1. Проверка основной системной модели. Модель включает: контекстную диаграмму IDEF0, словарь данных и спецификации процессов. Оценивается их согласованность с системной спецификацией.

Проверяем каркас входных и выходных данных которые мы хотим получить в результате ,анализируя IDEF0

Шаг 2. Проверки и уточнения диаграмм потоков данных уровней 1 и 2. Оценивается согласованность диаграмм, достаточность детализации преобразователей.

Проверяем достаточность детализации 1-го и 2-го уровня IDEF0, если плохая детализация расширяем функционал

Шаг 3. Определение типа основного потока диаграммы потоков данных. Основной признак потока преобразований — отсутствие переключения по путям действий.

это тот тип потоков, которой у нас переходит из диаграммы в диаграмму детализации

Шаг 4. Определение границ входящего и выходящего потоков, отделение центра преобразований.

Входящий поток — отрезок, на котором информация преобразуется из внешнего во внутренний формат представления.

Выходящий поток обеспечивает обратное преобразование — из внутреннего формата во внешний. Границы входящего и выходящего потоков достаточно условны.

Мы должны определиться с результатом того, что мы хотим видеть на выходе,каеи ф-и, данные, объекты.

Шаг 5. Определение иерархическая структура ПС формируется нисходящим распространением управления.

В иерархической структуре: 1

1. модули верхнего уровня принимают решения; 1
2. модули нижнего уровня выполняют работу по вводу, обработке и выводу; 1
3. модули среднего уровня реализуют как функции управления, так и функции обработки.

Каждый модуль имеет контролирующий блок

1. Контроллер входящего потока (контролирует получение входных данных).
2. Контроллер преобразуемого потока (управляет операциями над данными во внутреннем формате).
3. Контроллер выходящего потока (управляет получением выходных данных).

Шаг 6. Детализация структуры ПС. После прохождения преобразуемого поток слева направо. Возможны следующие варианты отображения: »

---1 преобразователь отображается в 1 модуль; »

Один главный блок 2-го уровня декомпозируется в один блок нижнего уровня детализации

--- 2-3 преобразователя отображаются в 1 модуль;

Если у нас 3 основных блока 2-го уровня взаимосвязаны так, что выходящий поток имеет только 1 блок декомпозицию нижнего уровня

---1 преобразователь отображается в 2-3 модуля.

Если у нас диаграмма 2-го уровня состоит из одного основного блока, а декомпозируется на 3 ф-и

Шаг 7. Уточнение иерархической структуры ПС. Модули разделяются и объединяются для:

1) повышения связности и уменьшения сцепления;

2) упрощения реализации;

3) упрощения тестирования;

4) повышения удобства сопровождения

CASE-СИСТЕМЫ

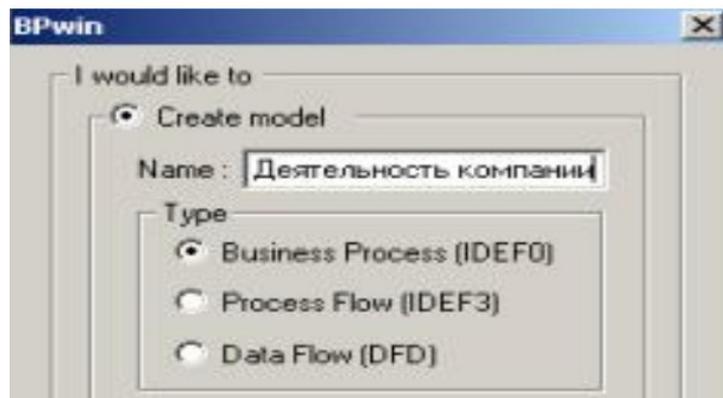
Имеющиеся на рынке программных продуктов CASE-системы для концептуального проектирования АИС чаще всего поддерживают методологию IDEF. В России широко известны продукты BPWin, ERWin, biasness modeler, OOWin фирмы Logic Works, Design/IDEF фирмы Meta Software, Silverrun фирмы CSA и др.

Например, пакет BPWin поддерживает работу с IDEF0, IDEF3, DFD моделями.

CASE- система(*Computer Aided Software/Sistem Engineering*) автоматизированное проектирование программ/ информационных систем.

ЭТО РАЗНОВИДНОСТИ БИЗНЕС МОДЕЛИРОВАНИЯ:

1. Моделирование бизнес процессов,
2. Технологического процесса ,
3. Поток данных



Два направления использования CASE-систем.

1. Как средство повышения эффективности в разработке сложного программного обеспечения, - соответствующие CASE-системы часто называют инструментальными средами разработки ПО.
2. Поддержка концептуального проектирования сложных систем.
моделирования деятельности предприятий, позволяя осуществлять проектирование и перепроектирование (реинжиниринг) бизнес-процессов. Такие CASE-системы часто называют системами BPR (Business Process Reengineering).

ХАРАКТЕРИСТИКИ CASE СРЕДСТВ

Основными характеристиками CASE средств:

1. Наличие графического интерфейса. CASE-технологии обеспечивают всех участников проекта, включая заказчиков, единым строгим, наглядным и интуитивно понятным графическим языком, позволяющим получать обозримые компоненты с простой и ясной структурой.

Понятный интерфейс позволяет заказчику участвовать в процессе разработки, а разработчикам - общаться с экспертами предметной области, разделять деятельность системных аналитиков, проектировщиков и программистов.

2. Наличие репозитория. Основа CASE-технологии - использование базы данных проекта (репозитория) для хранения всей информации о проекте, которая может разделяться между разработчиками в соответствии с их правами доступа.

позволяет раздавать права доступа разработчиками от БД и настраивать под каждого

3. Гибкость применения. CASE средства должны позволять проводить анализ процессов и создавать модели, сфокусированные на различных аспектах деятельности предприятия.

Мы можем видеть все процессы и анализируя их можем объединять в одну модель, а потом готовые модули использовать повторно в разработке

4. Возможность коллективной работы. поддерживает групповую работу над проектом, обеспечивая возможность работы в сети, экспорт-импорт любых фрагментов проекта для их развития и/или модификации, а также планирование, контроль, руководство и взаимодействие.

Разработчики могут параллельно смотреть разные части модели проектирования и выгружать себе отдельные ее части.

5. Построение прототипов. CASE-технология дает возможность быстро строить макеты (прототипы) будущей системы, что позволяет заказчику на ранних этапах разработки оценить, насколько она приемлема.

Мы можем быстро спроектировать любой процесс и оценить его стоимость и кадровую востребованность

6. Построение отчетов. Вся документация по проекту генерируется автоматически на базе репозитория (как правило, в соответствии с требованиями действующих стандартов).

При этом документация всегда отвечает текущему состоянию дел, поскольку любые изменения в проекте автоматически отражаются в репозитории.

Автоматически генерируется отчетность по добавлению каких либо моделей или процессов, внесение изменений в проектирование.

Контрольные вопросы:

1. Из чего состоит поток преобразований?
2. Что такое поток запросов?
3. Какие направления case систем вы знаете?
4. Назовите и опишите основные характеристики case средств
5. Что такое входящий и выходящий поток?
6. Из каких шагов состоит проектирование потоков данных.
7. Что такое case-система ?
8. Какие два направления case-система вы знаете?
9. Назовите и опишите характеристики case-средств

Список использованных источников:

1. Технологии разработки программного обеспечения С.А. Орлов
2. Технологии разработки программного обеспечения В.В. Бахтизин, Л.А. Глухова
3. Project Management For Dummies / Управление проектами для "чайников"
4. Л. Н. Боронина З. В. Сенук основы управления проектами
5. http://www.kpms.ru/Automatization/CASE_tools.htm
6. <https://studfiles.net/preview/3828360/page:10/>