

Разработка пользовательских интерфейсов

Назначение:

Пользовательский интерфейс предназначен для организации диалога между человеком и ЭВМ.

Основные понятия:

Диалог представляет собой регламентированный обмен информацией между пользователем и программной системой, направленный на совместное решение конкретной задачи.

Сообщение - порция информации, участвующая в диалоговом обмене.

➤ **Входное сообщение** (порождается пользователем):

- запрос информации;
- запрос помощи;
- приказ;
- ввод или изменение информации;
- выбор поля кадра.

➤ **Выходное сообщение** (формируется системой):

- подсказки;
- информационные сообщения, не требующие ответа или действия;
- вопросы, требующие ответа;
- приказы, требующие действий;
- сообщения об ошибках, нуждающиеся в ответных действиях;
- изменения формата кадра.

Рекомендуется группировать сообщения в отдельном модуле (будет легче переводить на другой язык, меньше избыточность, выше технологичность. Также сообщения можно разбить на виды – будет легче разбираться).

Кадр- совокупность сообщений на экране дисплея. Может иметь фиксированную или гибкую структуру, изменяющуюся с учетом потребностей пользователя или системы.

Также выделяют:

➤ ***Внутренний формат сообщения***

(определяет синтаксические свойства, т.е. грамматические правила, ограничения на длину, порядок спецификации элементов и т.д.).

➤ ***Внешний формат сообщения***

(описывает расположение на экране и способы его оформления - яркость, цвет и т.д.).

Шаг диалога

- это элементарная единица диалога.

Состоит из:

- действия первого участника;
- и следующей за ним **реакции** второго участника.
(**реакция** определяет входное или выходное сообщение).

Шаги диалога можно указать, например, в таблице **типичного хода событий** (см. объектный подход).

Интерфейс классифицируется:

По типу диалога (этапы разработки интерфейса):

- **Абстрактный** - определяет основные отличительные свойства диалога.

Определяются:

- *действия и соответствующие реакции на них;
- *инициаторы действия;
- *возможности пользователя по выбору функций;
- *однозначность формулировки задачи и др.

- **Конкретный** - связан с характеристикой сообщений.

Рассматриваются:

- *элементы представления сообщений

(алфавит, словарь);

- *структура представления сообщений

(внутренние и внешние форматы);

- *структура содержания сообщений

(т.е. семантические свойства - связаны с допустимой неопределенностью сообщений и их содержательностью, обеспечивается однозначная интерпретация с учетом предыдущего контекста или нет).

- **Технический** - обусловлен выбором основных и дополнительных устройств диалога (средства ввода, вывода).

По виду управления:

1. Управляемые системой

(т.е. система определяет сценарий);

2. Управляемые пользователем

(т.е. пользователь определяет сценарий).

В сложных системах можно встретить **гибридное управление**, например, **в интеллектуальных**.

По формам диалога.

1.Открытая

(выбор и/или манипулирование элементами на экране)

- элементы структурированы на экране (меню-вектор, меню-таблица и др.);
- элементы не структурированы.

2.Закрытая

(ввод элементов)

- ввод по шаблону согласно некоторой структуры (шаблон-вектор, шаблон-таблица, и др.);
- свободный ввод (директивная, фразовая).

Элементы на экране бывают: *на формализованном языке; *на естественном языке; *в виде пиктограмм.

Директивная форма

Основные свойства:

- ✓ при вызове исполнения операции применяется директива (команда)
- ✓ инициатором обмена является пользователь
- ✓ возможности выбора операций не ограничены
- ✓ интерпретация запроса однозначная
- ✓ словарь данной формы состоит из ключевых слов на естественном языке, сокращений, чисел и мнемокодов.

Основное достоинство - большая гибкость.

Основной недостаток - требует повышенной квалификации.

Меню и шаблоны

Основные свойства:

- Ориентированы на пользователя непрограммиста;
- Простое взаимодействие;
- Легкость обучения
(например, меню является по существу подсказкой и представляет собой перечень элементов).

Недостаток - менее гибкая по сравнению с директивной формой.

* **Например**, выделяют три группы дизайна меню сайта (навигация по меню): горизонтальные, вертикальные и раскрывающиеся (выпадающие). Можно одновременно использовать несколько видов меню. Это зависит от степени разветвленности структуры сайта.

Фразовая

Основные свойства:

- Использует ограниченно естественный язык;
- Инициатором шага может быть как пользователь, так и система;
- Для уменьшения времени набора фраз применяют: сокращения; шаблоны фраз; программируемые клавиши ключевых слов; меню и др.

Недостатки - большие затраты, не гарантируют однозначность формулировок, необходимость ввода грамматически правильных фраз.

Достоинства - более свободное общение с машиной, рассчитана на пользователя непрограммиста.

Обработка фраз

Основные понятия и методы:

Словоформа - отрезок текста между двумя соседними пробелами.

Морфологический анализ - обработка словоформ вне связи с контекстом.

Рассмотрим два метода:

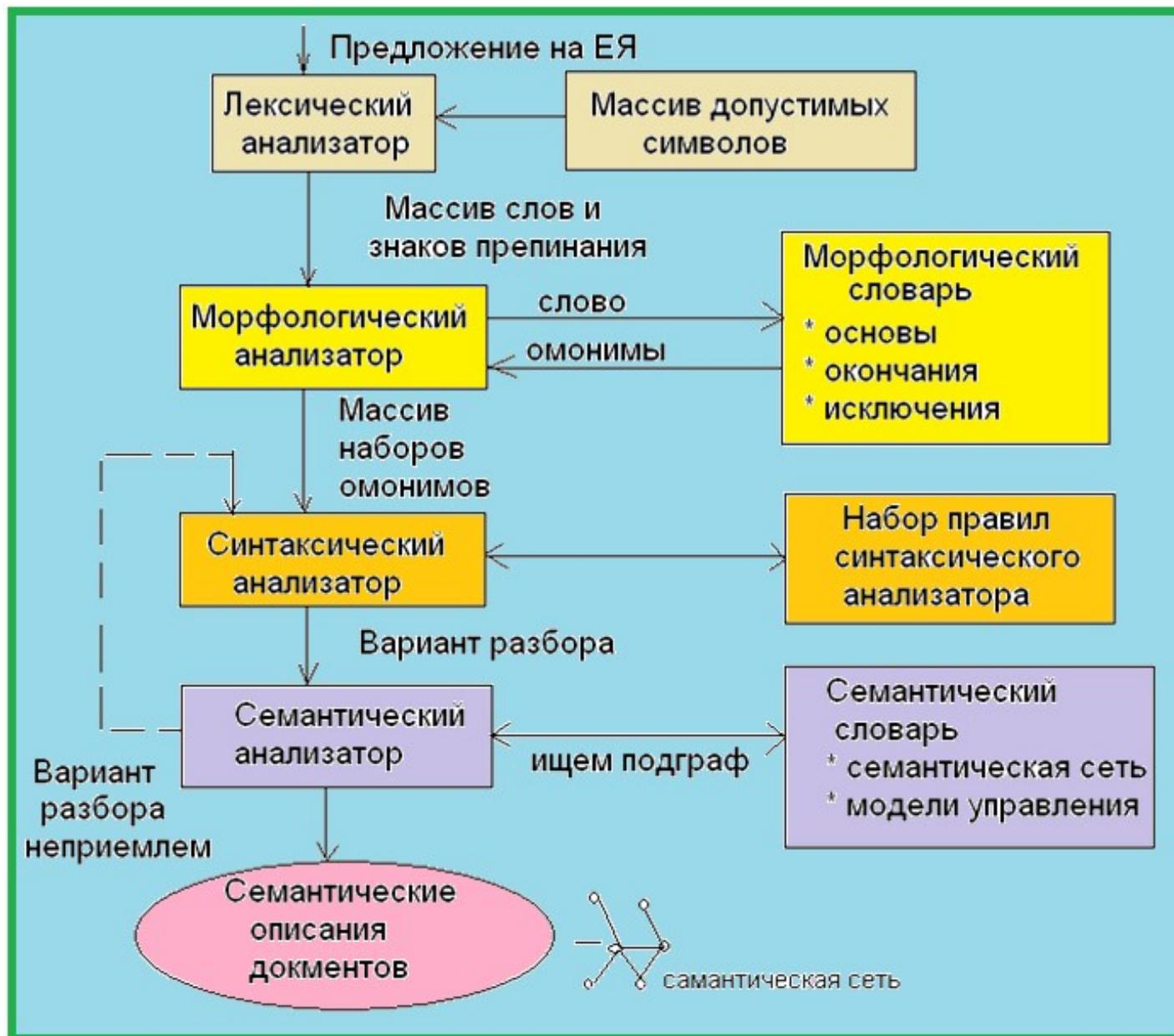
-декларативный - в словаре находятся все возможные словоформы каждого слова и анализ сводится к поиску словоформы в словаре.

-процедурный - выделяется в текущей словоформе основа, которая затем идентифицируется.

Синтаксический анализ - на основе информации полученной после морфологического анализа, строится синтаксическая структура входного сообщения, т.е. осуществляется **разбор предложения.**

Семантический анализ - определение смысловых отношений между словоформами (выявляются **главные предикаты**).

Схема функционирования лингвистического процессора



Способы взаимодействия человека и ЭВМ

При организации диалога для одного пользователя возможно **синхронное** и **асинхронное** взаимодействие сторон.

Синхронный способ характеризуется:

- ✓ партнеры диалога активизируются поочередно;
- ✓ общение четко регламентировано;
- ✓ ни один из партнеров не может прервать другого;
- ✓ широко распространен;
- ✓ реализуется достаточно просто.

Асинхронный способ обеспечивает:

- ✓ возможность выдачи экстренных сообщений от системы;
- ✓ ввод экстренных запросов пользователя.

Отдельно выделяют:

Сканирующий вывод системы с оперативным вмешательством пользователя.

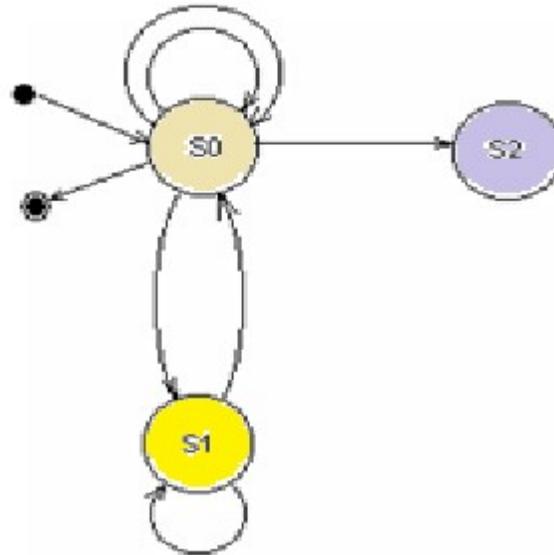
(Характерно наличие периодического изменения содержимого экрана в темпе машинной обработки, отсутствуют фазы ожидания и ввода данных)

Графическое отображение результатов проектирования интерфейса

При разработке интерфейса определяют:

- структуру диалога;
- формы интерфейса;
- иерархию меню и др.

Структуру диалога можно представить с помощью диаграммы состояний (граф диалога).



Каждая вершина соответствует определенному состоянию диалога.

(Например, S0 - начальное состояние, S1 - состояние ввода данных, S2 - конечное состояние.).

Дуга определяет смену состояния.

Основные структуры: линейная, древовидная , сетевая

В технологии визуального программирования

Классификация форм диалога:

- Основная и вторичная форма;
- Модальное и немодальное диалоговое окно.

Например, в Delphi и др. форма может быть **основной** и являться **модальным** диалоговым окном, иметь один управляющий элемент, например Close.

Модальное означает, что

такое окно должно быть закрыто прежде, чем пользователь может обратиться к другому окну (например, должны определить обязательно данные, а потом только двигаться дальше).

Например, форма может быть вторичной и являться модальным диалоговым окном (иметь кнопку ОК и выдаваться после основного окна).

Аналогично форма может быть и немодальным диалоговым окном, т.е. их можно не закрывать и при этом обратиться к другим окнам.

Если несколько форм, то необходимо выбрать вид интерфейса

- **Однодокументный (SDI – Single Document Interface).**
Содержит элементы: главное меню; панель инструментов; окна для размещения элементов управления данными; элементы управления и строки состояния.
- **Многодокументный (MDI – Multiple Document Interface).**
Каждый документ в своем собственном окне.

В обоих типах интерфейса взаимодействие происходит посредством обработки событий, поступающих от элементов управления.

Рекомендации при создании меню:

- Пункты меню следует группировать в логическом порядке и по содержанию.
- Следует избегать избыточных меню (информационно перегруженных).
- Пункты меню верхнего уровня должны содержать раскрывающиеся меню.
- Используйте клавиатурные эквиваленты команд меню («горячие» клавиши).
- Помещайте на панель инструментов часто используемые команды меню («частотный» принцип проектирования, т.е. часто используемые команды должны быть под «рукой»).
- Информируйте пользователя о ходе процесса (см. способы взаимодействия).

Способы ускорения ввода данных:

- Использование эквивалентов команд (кроме «мышки» использовать программируемые клавиши).
- Согласование расположения элементов с учетом специфики задачи, например, учет последовательности ввода т.п.
- Должна быть учтена зависимость по данным (пользователь не должен заполнять второстепенные поля, если главные не определены).
- Использование устоявшихся стандартов (обеспечить преемственность) при выполнении операций, т.е. использовать способы, к которым привыкли пользователи.
- По возможности вместо ввода обеспечить выбор данных из заранее заданного множества (**кроме ускорения ввода исключаются неправильные данные**).