

Содержание:

Введение

С каждым годом информационные технологии играют все большую роль в жизни людей. Все предприятия, образовательные и государственные учреждения, имеют свою собственную систему, несущую в себе весь объем информации для поддержки прямой деятельности данной организации. В настоящее время существует множество различных информационных систем, каждая из которых имеет свои собственные особенности. Для того чтобы облегчить и ускорить работу людей, данные системы и приложения должны быть построены на одинаковых принципах. При этом временные затраты на адаптацию к работе в данных информационных системах для неподготовленного человека должны быть минимальны.

Актуальность данной темы состоит в том, что именно интерфейс видит пользователь и именно по интерфейсу, он оценивает и само приложение, а иногда и весь продукт.

Целью данной работы является создание наглядного представления о том, что такое информационные системы, как они строятся в организациях, что такое пользовательский интерфейс и его виды. Объектом работы являются информационные технологии, а предметом – пользовательский интерфейс информационных технологий.

В ходе работы решаются следующие задачи:

1. Раскрытие понятия «информационные технологии», изучение принципов создания информационных систем и их структуры
2. Раскрытие понятия «пользовательского интерфейса» и его типы, требования к созданию пользовательского интерфейса
3. Изучение пользовательских интерфейсов операционных систем Apple и Windows 7

В процессе работы были использованы метод научного анализа и синтеза и сравнительный метод. Работа состоит из введения, трех глав, заключения, списка

использованной литературы и приложений.

ГЛАВА 1. Информационные технологии и принципы их создания

1.1. Информационные технологии и их классификация

Информационная технология - это совокупность методов, производственных процессов и программно-технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, обработку, хранение, передачу и отображение информации [2].

Цель функционирования этой цепочки, т. е. информационной технологии, - это снижения трудоемкости процессов использования информационного ресурса и повышение их надежности и оперативности.

Эффективность информационной технологии определяется, в конечном счете, квалификацией субъектов процессов информатизации. При этом технологии должны быть максимально доступны потребителям.

Существует несколько способов классификации информационных технологий:

1. По типу обрабатываемой информации. Данное разделение достаточно условное, в виду того, что большинство информационных технологий позволяет поддерживать и другие виды информации. В текстовых процессорах возможна и несложная расчетная деятельность, а табличные процессоры обрабатывают не только цифровую информацию, но и могут генерировать графики.

Каждый вид ориентирован на работу с информацией определенного вида. Модификация элементов, составляющих информационные технологии, дает возможность образования новых технологий в различных компьютерных средах.

2. Обеспечивающие и функциональные информационные технологии.

Обеспечивающие технологии (ОИТ) - это технологии обработки информации, которые могут использоваться как инструментарий в различных предметных

областях.

При этом они могут обеспечивать решение задачи разного плана и разной степени сложности. ОИТ могут быть разделены по классам задач, в зависимости от класса ОИТ используют разные виды компонентов и программных средств. При объединении ОИТ по предметному признаку возникает проблема системной интеграции, т. е. приведение различных технологий к единому стандартному интерфейсу.

Функциональные информационные технологии (ФИТ) - это модификация обеспечивающих технологий для задач определенной предметной области, т. е. реализуется предметная технология. Предметные технологии и информационная технология влияют друг на друга.

Например, появление пластиковых карточек как носителей финансовой информации принципиально изменила предметную технологию. При этом пришлось создавать совершенно новую информационную технологию. Но, в свою очередь, возможности, представленные новой ИТ, повлияли на предметную технологию пластиковых носителей (в области их защиты, например).

3. По типу пользовательского интерфейса.

Прикладной интерфейс связан с реализацией функциональных информационных технологий.

Системный интерфейс - это набор приемов взаимодействия с компьютером, который реализуется операционной системой или ее надстройкой.

Большинство обеспечивающих и функциональных технологий используются пользователем без посредников - программистов. Пользователь может самостоятельно изменять последовательность применения тех или иных технологий [3].

4. По участию пользователя в информационном процессе, технологии можно разделить на пакетные и диалоговые.

Задачи, решаемые в пакетном режиме, характеризуются следующими свойствами:

- алгоритм решения задачи формализован, процесс не требует вмешательства человека.

- имеется большой объем входных и выходных данных; значительная их часть хранится на магнитных носителях.
- большое время решения задач, обусловленное объемами данных.
- регламентность, т. е. задачи решаются с заданной периодичностью.

Диалоговый режим это не альтернатива пакетному режиму, а его развитие. Диалоговый режим позволяет пользователю вмешаться в процесс решения задачи, он отпускает пользователя, отменяет жестко закрепленную последовательность обработки данных. Применение режимов зависит в первую очередь от предметной технологии.

5. По степени их взаимодействия между собой. Например, дискретное и сетевое взаимодействие; взаимодействие с использованием различных вариантов обработки и хранения данных; распределенная информационная база и распределенная обработка данных.

Информационная технология тесно связана с информационными системами, которые являются для нее основной средой.

Информационная технология является процессом, состоящим из четко регламентированных правил выполнения операций, действий, этапов разной степени сложности над данными, хранящимися в компьютерах.

Основная цель информационной технологии - в результате целенаправленных действий по переработке первичной информации получить необходимую для пользователя информацию[5].

Информационная система является средой, составляющими элементами которой являются компьютеры, компьютерные сети, программные продукты, базы данных, люди, различного рода технические и программные средства связи и т. д.

Основная цель информационной системы - организация хранения и передачи информации. Информационная система представляет собой человеко-компьютерную систему обработки информации.

Реализация функций информационной системы невозможна без знания ориентированной на нее информационной технологии. Информационная технология может существовать и вне информационной системы.

1.2. Принципы создания информационных систем

Пользователям компьютеров достаточно часто приходится сталкиваться с ситуациями, когда программное обеспечение хорошо работает на одном устройстве, а при установке его на другое такое же устройство, возникают различные проблемы.

Проблема совместимости – это проблема совмещения различных вычислительных, телекоммуникационных и информационных устройств.

Развитие систем и средств вычислительной техники, расширенное их внедрение во все сферы науки, техники, сферы обслуживания и быта привели к необходимости объединения конкретных вычислительных устройств и реализованных на их основе информационных систем в единые информационно-вычислительные системы и среды, при этом разработчики столкнулись с рядом проблем.

Разнородность технических средств вычислительной техники с точки зрения организации вычислительного процесса, архитектуры, системы команд, разрядности процессора и шины данных потребовала создания физических интерфейсов, реализующих, как правило, взаимную совместимость устройств. С увеличением числа типов интегрируемых устройств возрастала и сложность организации физического интерфейса между ними.

Многообразие операционных систем, различия в разрядности и многие другие особенности привели к созданию программных интерфейсов между устройствами и системами. При этом достижение полной совместимости между программными продуктами и программными средами происходило не всегда.

Любая информационная система строится на нескольких основных принципах:

1. Принцип системности, который предполагает учет всех взаимосвязей, анализ отдельных частей системы, как самостоятельных составляющих и выявление роли каждой из них, в процессе функционирования всей системы. Этот принцип реализует процессы анализа и синтеза, смысл которых состоит в разложении целого на части и воссоединение целого из частей.
2. Принцип развития (открытости). Заключается в том, что внесение изменений в систему, обусловленных самыми различными причинами должно осуществляться только путем дополнения системы без переделки уже созданного.

Термин открытая система сегодня можно определить как "исчерпывающий и согласованный набор международных стандартов на информационные технологии и профили функциональных стандартов, которые специфицируют интерфейсы, службы и поддерживающие их форматы, чтобы обеспечить взаимодействие и мобильность программных приложений, данных и персонала"[19].

Все открытые системы имеют следующие общие свойства:

- Расширяемость/масштабируемость, что обеспечивает добавление новых функций для информационных систем, или изменение уже существующих, без вреда для оставшейся части системы.
- Мобильность/переносимость обеспечивает возможность для переноса данных или программ при модернизации или замене аппаратных платформ информационных систем.
- Взаимодействие – это способность информационных систем взаимодействовать с другими системами.
- Стандартизуемость – информационные системы разрабатываются на основании международных стандартов и предложений, а реализация открытости осуществляется на базе функциональных стандартов (профилей) в области информационных технологий.
- Интуитивно понятный интерфейс дает возможность работать пользователю, который не имеет специальной подготовки.

Черты открытости рассматриваются и реализуются в совокупности, что позволяет решать проблемы проектирования, разработки и внедрения современных информационных систем.

3. Принцип современности. При создании системы должны быть реализованы информационные интерфейсы, благодаря которым она может взаимодействовать с другими системами согласно установленным правилам. В современных условиях это особенно касается сетевых связей локального и глобального уровней.

Если в локальных сетях относительно несложно установить и соблюдать стандарты "общения" отдельных бизнес-процессов между собой и со смежными системами, то выход в глобальные сети требует дополнительных ужесточенных мер по защите информации, знания и соблюдения различного рода протоколов.

4. Принцип стандартизации (унификации) содержит в себе рациональное использование типовых, унифицированных и стандартизированных элементов. Задачи необходимо разрабатывать таким образом, чтобы они подходили к возможно более широкому кругу объектов.

В современных разработках пакетов прикладных программ рассматриваемый принцип задействован. Однако при знакомстве с конкретным пакетом необходимо обращать внимание на сущность реализации типовых решений, поскольку каждый разработчик по-своему "видит" такие решения.

5. Принцип эффективности предусматривает достижение рационального соотношения между затратами на создание системы и целевыми эффектами, включая конечные результаты, отражающиеся на прибыльности и получаемые по окончании внедрения автоматизации в управленческие процессы.

1.3. Структура среды информационной системы

Любая информационная системы состоит из двух взаимосвязанных частей:

- функциональная часть (прикладные программы, реализующие функции прикладной области)
- среда/системная часть (обеспечивает исполнение прикладных программ)

Исходя из структуры информационной системы можно выделить следующие вопросы стандартизации:

1. Прикладной программный интерфейс (Application Program Interface – API)
2. Взаимодействие информационной системы с внешней для неё средой (External Environment Interface – EEI)[15]

API и EEI определяют спецификации внешнего описания среды информационной системы или её архитектуру с точки зрения конечного пользователя и проектировщика системы.

В 1974 году IBM предложила Reference Open System Model – эталонную модель открытых систем, данная модель используется более 20 лет (Рис.1).

Вычислительная среда разбивается на 7 уровней, для описания взаимодействия между которыми используются соответствующие стандарты. Модель обеспечивает

связь уровней вне зависимости от построения уровня в каждой конкретной реализации. С одной стороны, эталонная модель позволяет детально описать связи в среде с точки зрения технических устройств и коммуникационных взаимодействий, однако, с другой стороны, она не учитывает взаимосвязь в соответствии с мобильностью прикладного программного обеспечения.



Рисунок 1. Уровни моделей информационных систем

Эталонная модель среды открытых систем отражает разделение любой информационной системы на приложения и среду, в которой эти приложения функционируют. Между приложениями и средой определяются стандартизированные интерфейсы (API), которые являются необходимой частью профилей любой открытой системы. В информационных системах так же могут быть определены унифицированные интерфейсы взаимодействия функциональных частей друг с другом и интерфейсы взаимодействия между компонентами среды[19].

1.4. Модель создания информационных систем

В настоящее время существует достаточно много стандартов и методологий, освещающих вопросы проектирования информационных систем. Все эти подходы предполагают использование декомпозиции.

При разработке информационной системы следует определить:

1. Сущности (объекты информационной системы)
2. Атрибуты (свойства объектов)
3. Связи между объектами и информационные потоки

Разработка информационной системы проходит в несколько этапов, на каждом из которых происходит большая детализация элементов системы.

Для того, чтобы построить эффективную информационную систему, позволяющую качественно выполнять функции управления, необходимо начать с анализа предприятия и создания его схематической модели (Рис.2).



Рисунок 2. Онтологическое поле современной компании

Компания – это сложная понятийная структура, объединяющая в себе совокупности сущностей и взаимосвязей.

Информационная структура должна отражать логику данной организации, автоматизировать и упрощать процесс работы с информацией, а так же наглядно отображать результаты в виде наборов отчетных форм.

Во время создания модели предприятия происходит процесс создания «языка», который будет понятен руководителям предприятия и будущим пользователям системы и выработает единое представление для создателей системы.

Параметры проекта внедрения информационной системы управления на предприятие:

1. Цели, которые необходимо достичь при помощи информационной системы
2. Фактическая потребность в программном и аппаратном обеспечении
3. Реальные оценки сроков проектирования и запуска системы
4. Список ключевых пользователей
5. Соответствие программного обеспечения и особенностей компании
6. Последовательность внедрения элементов системы

После построения бизнес-модели происходит формирование модели проектирования, реализации и внедрения самой ИС.

Построение информационных систем происходит в несколько этапов:

1. Определение задач и функций системы и их последующий анализ. На этом этапе разработчики тесно сотрудничают с руководителями компании, для того, что бы четко определить какими свойствами должна обладать информационная система, какие задачи она должна выполнять, каким функциями обладать. Затем все эти данные обрабатываются, документируются и формализуются.
2. Проектирование – определение элементов системы, их спецификаций и способов взаимодействия между ними.
3. Разработка – создание отдельных элементов системы и объединение их в единое целое
4. Тестирование – определение соответствия системы требованиям, выявленным на первом этап
5. Внедрение – ввод системы в действие на предприятии
6. Функционирование и сопровождение – процесс эксплуатации системы на предприятии и его поддержка со стороны разработчиков



Рисунок 3. Модель информационной системы

Этапы разработки, тестирования, внедрения, эксплуатации и сопровождения системы составляют понятие реализации, которая в свою очередь является сложным многоаспектным процессом. Реализация происходит на основании международных стандартов, спецификаций и соглашений, что позволяет создать «открытую» систему.

Жизненный цикл ИС формируется в соответствии с принципом нисходящего проектирования и, как правило, носит спирально-итерационный характер. Реализованные этапы, начиная с самых ранних, циклически повторяются в соответствии с изменениями требований и внешних условий, введением дополнительных ограничений и т. п.

На каждом этапе жизненного цикла порождается определенный набор технических решений и документов, при этом для каждого этапа исходными являются документы и решения, принятые на предыдущем этапе. Жизненный цикл системы завершается, когда прекращается ее программное и техническое сопровождение[14].

Выводы по 1 главе:

1. Существует большое количество разнообразных информационных технологий, каждая из которых имеет отличительные особенности
2. Качественная информационная система строится на принципах системности, открытости, современности, унификации и эффективности
3. Любая информационная система состоит из 2 взаимосвязанных частей: функциональной (прикладные программы) и системной (среды).

4. Разработка информационной системы начинается и основывается на изучении системы компании.

ГЛАВА 2. Пользовательский интерфейс информационных технологий

2.1. Требования к пользовательскому интерфейсу информационных технологий

В условиях использования компьютерных информационных технологий актуальны вопросы организации взаимодействия человека с техническими и программными средствами. Такое взаимодействие обеспечивает пользовательский интерфейс.

Интерфейс с общих позиций определяется как определенная стандартами граница раздела двух систем, устройств или программ.

Применительно к информационным технологиям интерфейс определим как совокупность средств и правил, обеспечивающих взаимодействие устройств, программ.

Интерфейс пользователя – это элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением.

Пользовательский интерфейс состоит из 3 частей:

1. общение приложения с пользователем;
2. общение пользователя с приложением;
3. язык общения;

Язык общения определяется разработчиком программного приложения.

Интерфейс пользователя компьютерного приложения включает:

1. средства отображения информации, отображаемую информацию, форматы и коды;

2. командные режимы, язык «пользователь — интерфейс»;
3. устройства и технологии ввода данных;
4. диалоги, взаимодействие и транзакции между пользователем и компьютером, обратную связь с пользователем;
5. поддержку принятия решений в конкретной предметной области;
6. порядок использования программы и документацию на неё.

Пользовательский интерфейс часто понимают только как внешний вид программы. Однако на деле пользователь воспринимает через него всю программу в целом, а значит, такое понимание является слишком узким.

В действительности ПИ объединяет в себе все элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением (ПО), это не только экран, который видит пользователь.

К этим элементам относятся:

1. набор задач пользователя, которые он решает при помощи системы;
2. используемая системой метафора;
3. элементы управления системой;
4. навигация между блоками системы;
5. визуальный (и не только) дизайн экранов программы;
6. средства отображения информации, отображаемая информация и форматы;
7. устройства и технологии ввода данных;
8. диалоги, взаимодействие и транзакции между пользователем и компьютером;
9. обратная связь с пользователем;
10. поддержка принятия решений в конкретной предметной области;
11. порядок использования программы и документация на нее.

Для успешного взаимодействия пользователя с приложением, необходимо, чтобы приложение обладало следующими чертами:

1. Доступность — наиболее важный элемент дизайна. По сути, вся цель пользовательского интерфейса состоит в том, чтобы дать возможность пользователям взаимодействовать с системой. Если человек не сможет понять, как приложение работает, он будет только запутан и в итоге разочарован. Вот почему, разрабатывая интерфейс приложения или веб-сайта, необходимо учесть, что он должен быть интуитивно понятен пользователю.

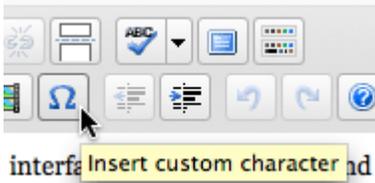


Рисунок 4. Реализация подсказок в MS Word

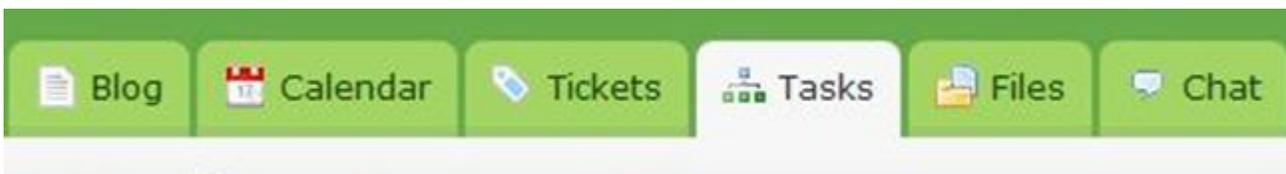
2. Минимализм. Большая загруженность — враг хорошего пользовательского интерфейса. Легко попасть в ловушку избыточной доступности — добавляя все больше и больше управляющих элементов, так как это сильно загромождает интерфейс. Он растет, и пользователь будет вынужден много читать, чтобы понять что, где и для чего располагается.



Рисунок 5. Панель регулировки уровня звука в OS X

3. Уверенность. Многие дизайнеры стремятся сделать интерфейсы «интуитивно понятными». Это означает, что пользователи должны инстинктивно понимать и осмысливать возможности проекта. Но так как приложения делают программисты, то что интуитивно знакомо для них, может быть чрезмерно сложным для обычных пользователей.

Поэтому необходимо проводить тестирование для каждого вида техники, интерфейса или приложения, чтобы еще на начальном этапе выяснить все его преимущества и недостатки.



Рисунок

6. Интуитивно понятный интерфейс GoPlan

4. Отзывчивость означает несколько вещей. Интерфейс веб-сайта должен работать очень быстро. Длительное ожидание загрузки страницы раздражает. Так же отзывчивость означает некоторую постоянную форму взаимодействия с пользователем.

Интерфейс должен информировать пользователя о происходящем. Например, вы нажимаете кнопку отправки сообщения. Если сообщение отправляется посредством AJAX, было бы разумно выводить состояния отправки, например «Отправка...», «Сообщение отправлено» или «Ошибка отправки сообщения». Когда пользователь видит процесс выполнения, он чувствует себя спокойнее. Особенно это заметно на медленных интернет-каналах.

Loading dmitry@usabilitypost.com...



Рисунок 7. Полоса загрузки Gmail

5. Соответствие контексту. При выборе определенных решений при создании дизайна необходимо брать в расчет тип содержимого страницы. Разные страницы могут содержать контент разного типа. Адаптация каждой страницы под соответствующий ей контент позволит сделать работу пользователя проще и удобнее.

6. Привлекательность. Хороший интерфейс должен быть привлекательным. Привлекательный пользователю интерфейс делает работу с ним приятной. Невозможно сделать интерфейс, который будет нравиться всем, но тем не менее, можно провести определенную градацию пользователей и на основании этого создавать интерфейс.

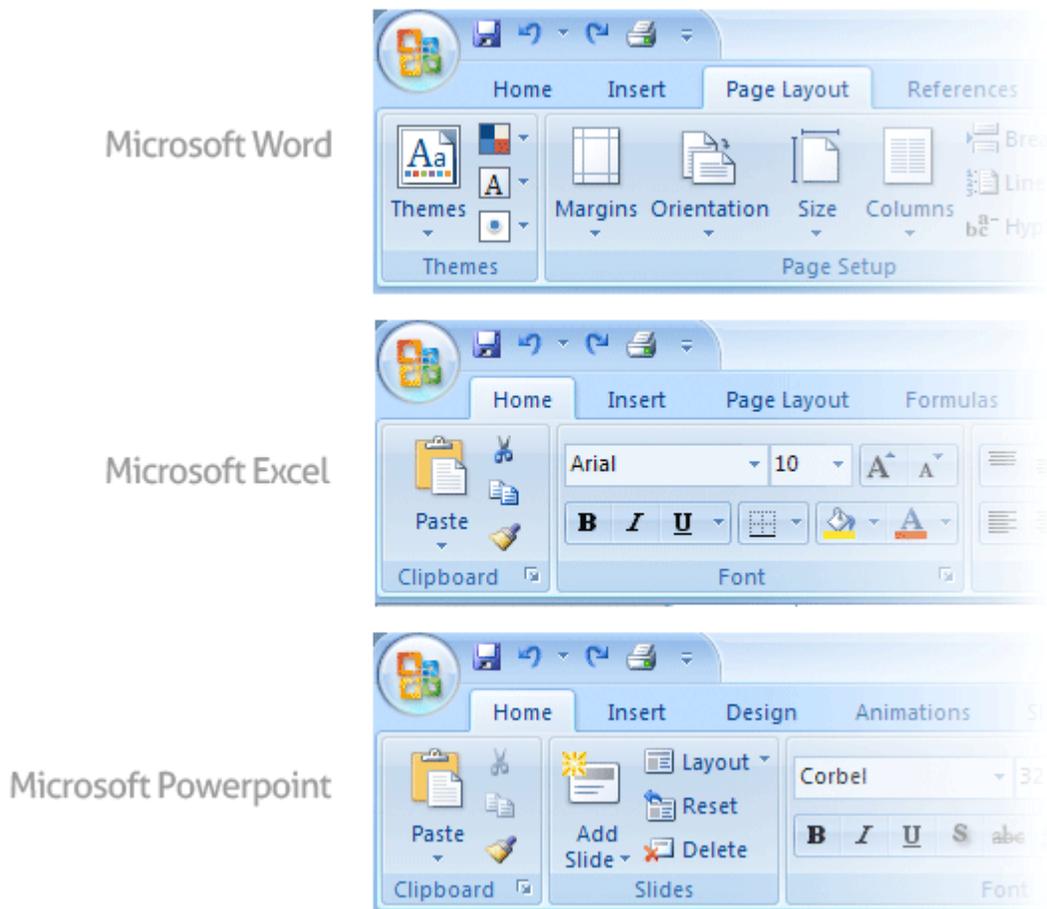


Рисунок 8. Элементы управления MS Office, различные для каждого типа контента

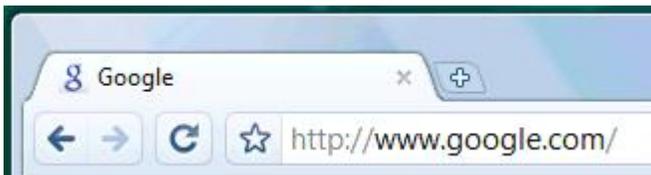


Рисунок 9. Интерфейс Google Chrome.

7. Эффективность. Пользовательский интерфейс — это инструмент управления. Он предоставляет доступ к различным функциям приложения или веб-сайта. Хороший интерфейс должен давать возможность пользователю с наименьшими усилиями выполнить интересующее его действие.

Очень важно понять, что пользователь чаще всего хочет выполнить на определенной странице. Не нужно выводить списком все возможности приложения, чаще всего пользователю интересна только небольшая часть этого списка. При этом пользователь должен моментально найти наиболее полезные и самые требуемые функции, это очень упростит его общение с проектом.

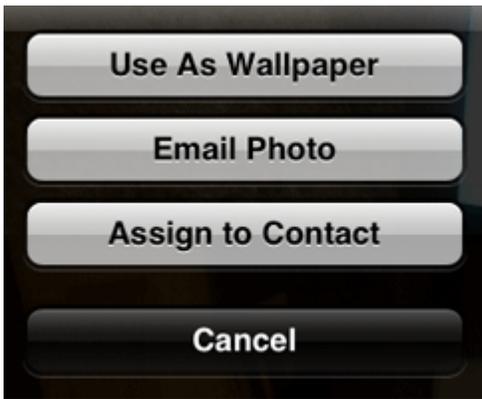


Рисунок 10. Три самых часто выполняемых действий над фотографиями в Apple iPhone объединены в общий список с моментальным доступом

8. Снисходительность. Программист должен быть готов к тому, что пользователи будут делать ошибки при работе с проектом. Это может происходить как по вине разработчика, так и по вине пользователя. Необходимо грамотно обрабатывать все возможные ошибки — это является одним из главных показателей качества проекта. Важным показателем снисходительности интерфейса, является возможность беречь данные от случайных действий пользователя. Например, если кто-то удаляет важную информацию, необходима возможность ее восстановления.



Рисунок 11. «Случайно удалена важная информация в Gmail»

Работая над достижением одной из этих характеристик, можно создать проблемы для достижения другой. Например, стараясь сделать интерфейс более понятным, добавляется много описаний и объяснений, что в конечном итоге делает интерфейс еще более громоздким и неудобным. Или урезая материал для достижения минимализма, можно сделать вещи непонятными рядовому пользователю. Что актуально для одного, для другого может быть не допустимо.

Таким образом, обладая всеми вышеперечисленными свойствами, информационная система сможет оптимально работать на любом предприятии.

2.1. Типы пользовательского интерфейса

Пользовательский интерфейс — это набор приемов и средств взаимодействия пользователя с приложением.

С точки зрения пользователя операционная система формирует удобный пользовательский интерфейс, программное окружение, на фоне которого выполняется разработка и осуществляется исполнение прикладной программы пользователя.

Здесь пользовательский интерфейс можно рассматривать как командный язык для управления функционированием компьютера и набор сервисных услуг, освобождающих пользователя от выполнения рутинных операций.

Интерфейс пользователя — элементы и компоненты программы, которые способны оказывать влияние на взаимодействие пользователя с программным обеспечением.

В зависимости от типа пользовательского интерфейса информационные технологии имеют соответствующую классификацию. При этом выделяется системный и прикладной интерфейс.

1. Прикладной интерфейс связан с реализацией некоторых функциональных информационных технологий.
2. Системный интерфейс — это набор приемов взаимодействия с компьютером, который реализуется операционной системой или ее надстройкой.

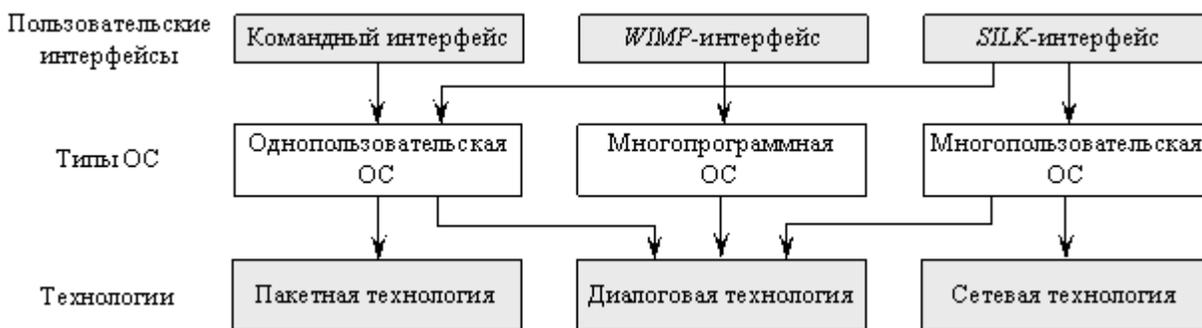


Рисунок 11. Типы пользовательского интерфейса

1) Командный интерфейс. Командный интерфейс называется так по тому, что в этом виде интерфейса человек подает "команды" компьютеру, а компьютер их выполняет и выдает результат человеку. Командный интерфейс реализован в виде пакетной технологии и технологии командной строки.

2) WIMP - интерфейс (Window - окно, Image - образ, Menu - меню, Pointer - указатель). Характерной особенностью этого вида интерфейса является то, что диалог с пользователем ведется не с помощью команд, а с помощью графических

образов - меню, окон, других элементов. Хотя и в этом интерфейсе подаются команды машине, но это делается "опосредственно", через графические образы. Этот вид интерфейса реализован на двух уровнях технологий: простой графический интерфейс и "чистый" WIMP - интерфейс [12].

3) SILK - интерфейс (Speech - речь, Image - образ, Language - язык, Knowledge - знание). Этот вид интерфейса наиболее приближен к обычной, человеческой форме общения. В рамках этого интерфейса идет обычный "разговор" человека и компьютера. При этом компьютер находит для себя команды, анализируя человеческую речь и находя в ней ключевые фразы. Результат выполнения команд он также преобразует в понятную человеку форму. Этот вид интерфейса наиболее требователен к аппаратным ресурсам компьютера, и поэтому его применяют в основном для военных целей.

Пакетная технология. Исторически этот вид технологии появился первым. Она существовала уже на релейных машинах Зюса и Цюзе (Германия, 1937 год). Идея ее проста: на вход компьютера подается последовательность символов, в которых по определенным правилам указывается последовательность запущенных на выполнение программ.

После выполнения очередной программы запускается следующая и т. д. Машина по определенным правилам находит для себя команды и данные. В качестве этой последовательности может выступать, например, перфолента, стопка перфокарт, последовательность нажатия клавиш электрической пишущей машинки (типа CONSUL). Машина также выдает свои сообщения на перфоратор, алфавитно-цифровое печатающее устройство (АЦПУ), ленту пишущей машинки. Такая машина представляет собой "черный ящик" (точнее "белый шкаф"), в который постоянно подается информация и которая также постоянно "информирует" мир о своем состоянии.

Человек здесь имеет малое влияние на работу машины - он может лишь приостановить работу машины, сменить программу и вновь запустить ЭВМ. Впоследствии, когда машины стали помощнее и могли обслуживать сразу нескольких пользователей, вечное ожидание пользователей типа: "Я послал данные машине. Жду, что она ответит. И ответит ли вообще?" - стало, мягко говоря, надоедать. К тому же вычислительные центры, вслед за газетами, стали вторым крупным "производителем" макулатуры. Поэтому с появлением алфавитно-цифровых дисплеев началась эра по-настоящему пользовательской технологии - командной строки.

Технология командной строки. При этой технологии в качестве единственного способа ввода информации от человека к компьютеру служит клавиатура, а компьютер выводит информацию человеку с помощью алфавитно-цифрового дисплея (монитора).

Эту комбинацию (монитор + клавиатура) стали называть терминалом, или консолью. Команды набираются в командной строке. Командная строка представляет собой символ приглашения и мигающий прямоугольник - курсор. При нажатии клавиши на месте курсора появляются символы, а сам курсор смещается вправо. Это очень похоже на набор команды на пишущей машинке. Однако, в отличие от нее, буквы отображаются на дисплее, а не на бумаге, и неправильно набранный символ можно стереть.

Команда заканчивается нажатием клавиши Enter (или Return) После этого осуществляется переход в начало следующей строки. Именно с этой позиции компьютер выдает на монитор результаты своей работы. Затем процесс повторяется. Технология командной строки уже работала на монохромных алфавитно-цифровых дисплеях. Поскольку вводить позволялось только буквы, цифры и знаки препинания, то технические характеристики дисплея были не существенны. В качестве монитора можно было использовать телевизионный приемник и даже трубку осциллографа.

Обе эти технологии реализуются в виде командного интерфейса - машине подаются на вход команды, а она как бы "отвечает" на них.

Преобладающим видом файлов при работе с командным интерфейсом стали текстовые файлы - их и только их можно было создать при помощи клавиатуры. На время наиболее широкого использования интерфейса командной строки приходится появление операционной системы UNIX и появление первых восьмиразрядных персональных компьютеров с многоплатформенной операционной системой CP / M.

2.3. Графический интерфейс. Wimp- и Silk-интерфейсы

Идея графического интерфейса зародилась в середине 70-х годов, когда в исследовательском центре Xerox Palo Alto Research Center (PARC) была разработана концепция визуального интерфейса. Предпосылкой графического интерфейса

явилось уменьшение времени реакции компьютера на команду, увеличение объема оперативной памяти, а также развитие технической базы компьютеров [14].

Аппаратным основанием концепции, конечно же, явилось появление алфавитно-цифровых дисплеев на компьютерах, причем на этих дисплеях уже имелись такие эффекты, как "мерцание" символов, инверсия цвета (смена начертания белых символов на черном фоне обратным, то есть черных символов на белом фоне), подчеркивание символов. Эти эффекты распространились не на весь экран, а только на один или более символов.

Следующим шагом явилось создание цветного дисплея, позволяющего выводить, вместе с этими эффектами, символы в 16 цветах на фоне с палитрой (то есть цветовым набором) из 8 цветов. После появления графических дисплеев, с возможностью вывода любых графических изображений в виде множества точек на экране различного цвета, фантазии в использовании экрана вообще не стало границ.

Первая система с графическим интерфейсом 8010 Star Information System группы PARC, таким образом, появилась за четыре месяца до выхода в свет первого компьютера фирмы IBM в 1981 году. Первоначально визуальный интерфейс использовался только в программах. Постепенно он стал переходить и на операционные системы, используемых сначала на компьютерах Atari и Apple Macintosh, а затем и на IBM - совместимых компьютерах.

С более раннего времени, и под влиянием также и этих концепций, проходил процесс по унификации в использовании клавиатуры и мыши прикладными программами. Слияние этих двух тенденций и привело к созданию того пользовательского интерфейса, с помощью которого, при минимальных затратах времени и средств на переучивание персонала, можно работать с любыми программным продуктом. Описание этого интерфейса, общего для всех приложений и операционных систем, и посвящена данная часть.

Вторым этапом в развитии графического интерфейса стал "чистый" интерфейс WIMP.

WIMP-интерфейс – это наиболее часто используемый интерфейс в настоящее время. Все продукты компании «Microsoft» основаны на нем (все версии Windows, в том числе и для операционной системы UNIX, и многие системы для Макинтошей).

Аббревиатура WIMP – это Windows (окна), Icons (значки, иконки), Menus (меню), Pointers (указатели мыши). Иногда – Windows, Icons, Mice (мышь), Pull-down menus (выпадающие меню).

Отметим, что кроме этих элементов широко используются кнопки, инструментальные панели, палитры, диалоговые окна и др. Их принято называть widgets, что лучше всего перевести, как «прибамбасы». Наборы таких «прибамбасов» образуют инструментарий взаимодействия человека с компьютером.

Современные оконные системы используют одинаковые наборы «прибамбасов» для реализации основного принципа – видеть и чувствовать.

Windows. Окна могут по-разному располагаться, быть наложенными друг на друга или располагаться каскадом, менять размеры, разворачиваться во весь экран, свертываться в иконку. Важная часть окна – наличие скролл-бара для перетаскивания содержимого в видимую часть (что делает окно аналогом реального окна).

Icons. Маленькая картинка служит «представителем» закрытого окна, позволяя развернуть его, когда потребуется. Такая картинка – напоминание о существовании окна, экономит место на экране. Иконки также служат для отображения других элементов системы, являясь их «сверткой», например мусорная корзина, разные диски, папки и пр. Варианты изображения иконки могут быть самыми разнообразными и служат мнемоническими правилами, позволяющими по виду иконки понять, что за ней скрывается.

Pointers. Указатели – основа функционирования WIMP-интерфейса. Большое значение имеет форма курсора. Скажем, стрелка, либо вертикальная полоска, либо курсор меняют форму при изменении функции (стрелка заменятся крестом при рисовании линий). Форма указателя может также информировать о состоянии системы: иметь вид песочных часов, если выполняется программа и надо подождать.

Форма указателя подобна иконке, но в дополнение к ней указатель имеет так называемую hot-spot (активную область); если указатель в форме стрелки (любого вида), то центр этой области – на острие стрелки, если в виде указывающего пальца – на кончике этого пальца, но есть формы, когда активная область не столь очевидна (и это плохо). При выборе формы указателя следует всегда заботиться об очевидности активной области.

Используются и другие элементы, например диалоговые окна (они могут быть также всплывающими или нет, иметь текст, иконку или и то и другое), наборы инструментальных кнопок (toolbars), при нажатии на которые происходят какие-то часто встречаемые действия.

Этот подвид интерфейса характеризуется следующими особенностями:

1. Вся работа с программами, файлами и документами происходит в окнах - определенных очерченных рамкой частях экрана.
2. Все программы, файлы, документы, устройства и другие объекты представляются в виде значков - иконок. При открытии иконки превращаются в окна.
3. Все действия с объектами осуществляются с помощью меню. Хотя меню появилось на первом этапе становления графического интерфейса, оно не имело в нем главенствующего значения, а служило лишь дополнением к командной строке. В чистом WIMP - интерфейсе меню становится основным элементом управления.
4. Широкое использование манипуляторов для указания на объекты. Манипулятор перестает быть просто игрушкой - дополнением к клавиатуре, а становится основным элементом управления. С помощью манипулятора указывают на любую область экрана, окна или иконки, выделяют ее, а уже потом через меню или с использованием других технологий осуществляют управление ими.

Следует отметить, что WIMP требует для своей реализации цветной растровый дисплей с высоким разрешением и манипулятор. Также программы, ориентированные на этот вид интерфейса, предъявляют повышенные требования к производительности компьютера, объему его памяти, пропускной способности шины и т. п.

Однако этот вид интерфейса наиболее прост в усвоении и интуитивно понятен. Поэтому сейчас WIMP - интерфейс стал стандартом де-факто.

С середины 90-х годов, после появления недорогих звуковых карт и широкого распространения технологий распознавания речи, появился так называемый "речевая технология" SILK - интерфейса. При этой технологии команды подаются голосом путем произнесения специальных зарезервированных слов - команд.

Слова должны выговариваться четко, в одном темпе. Между словами обязательна пауза. Из-за неразвитости алгоритма распознавания речи такие системы требуют индивидуальной предварительной настройки на каждого конкретного пользователя. "Речевая" технология является простейшей реализацией SILK - интерфейса.

Таким образом, для того, чтобы создать оптимальный пользовательский интерфейс, который будет удобен для программиста, пользователя и организация, необходимо следовать международным стандартам и определенным требованиям, которые предъявляются к пользовательскому интерфейсу практически любого приложения.

2.4. Трехмерный (пространственный) интерфейс и виртуальная реальность

Такой тип интерфейса все более востребован последнее время. Самая убедительная иллюстрация этого – виртуальная реальность, но этим отнюдь не исчерпывается круг возможных приложений 3D (3-Dimensions, т. е. трехмерных образов). В целом же пространственный интерфейс сочетается с другими возможностями, особенно в WIMP-интерфейсах [13].

Простейший случай – использование в WIMP-интерфейсе элементов объемности, к примеру затенения (для кнопок, полос прокрутки и др.). Разумно используемый эффект трехмерности позволяет подсветить активные (или какие-то нужные) области.

К сожалению, во многих интерфейсах злоупотребляют этими эффектами, скульптурно выделяя каждый фрагмент текста, меню, границы, в результате чего теряется смысл это мощного средства. В пространственных интерфейсах объекты обычно плоские, но путем изменения их размера создают ощущение удаленности-приближенности. Для создания ощущения трехмерности пользуются углом освещения, размерами (о которых мы сказали) и наложением одних изображений на другие.

Трехмерные интерфейсы в большинстве случаев относятся к интерфейсам прямой манипуляции. Смысл в том, что пользователь не отдает команды системе, а манипулирует объектами, что более естественно. Первым популярным применением этого метода была корзина для удаления файлов на Macintosh

(начиная с Windows 95 такая корзина стала стандартом и в мире Windows, хотя существовала и раньше).

Для достижения достаточной эффективности не следует стараться наиболее реалистично отражать действие, значительно важнее более реалистично отразить объект, над которым это действие совершается. Например, компьютерную панель управления работой осветительных приборов не обязательно снабжать точными изображениями выключателей.

Главное – реалистично отразить на ней план помещения и расположение источников света, равно как и показать прямую (читай – непосредственную) связь между этой информацией и собственно выключателями. Во-вторых, бывают ситуации, когда эффективность непосредственного манипулирования уравнивается неэффективностью физических действий пользователя.

Термин «виртуальная реальность», или «виртуальное окружение», означает, что создается такое компьютерное изображение, которое, находясь в согласии с прошлым опытом человека, интуитивно воспринимается им как естественное.

Виртуальная реальность отличается от множества других компьютерных образов выполнением следующих условий:

- графическое изображение должно позволять выполнять пространственные преобразования, зависящие, скажем, от пространственного положения пользователя;
- все три оси, описывающие положение объектов в пространстве, должны быть активны, т. е. задействованы;
- изменения объектов на экране должны соответствовать их реальным изменениям в трехмерном пространстве;
- должно быть реализовано большинство интуитивно возможных взаимодействий с предъявляемым объектом в реальном мире;
- должны быть возможны преобразования объектов по всем реальным или требуемым степеням свободы;
- реакция объектов на воздействие должна происходить в квазиреальное время.

Виртуальная реальность есть результат отделения (можно даже сказать, отрыва) логического представления процессов от их физической сути. Для человеко-компьютерного взаимодействия под логическим представлением понимается представление пользователя о системе.

Виртуализация требует наличия средств преобразования логического представления в физическое и обратно. В качестве ресурсов, подвергающихся виртуализации, могут выступать как программные, так и аппаратные средства.

Главное – возможность разделения объекта виртуализации на отдельные блоки, каждый из которых можно сравнительно легко преобразовать в требуемую форму.

На уровне приложений, например, виртуализация существует давно: еще в гг. ею пользовались для создания псевдонногозадачных сред. Примерно тогда же появились виртуальные машины в мэйнфреймах IBM. Это позволило перейти от пакетной обработки данных к интерактивным приложениям и обеспечить параллельную работу нескольких пользователей или приложений.

Сегодня большинство операционных систем поддерживает параллельную обработку приложений и может обслуживать одновременно несколько пользователей, а средства виртуализации продолжают развиваться, например, в сторону обеспечения одновременной работы нескольких операционных систем на одной машине. Современный уровень развития средств виртуализации позволяет временно не задействованным ресурсам приносить реальную пользу.

Виртуализация дает конкурентные преимущества, именно поэтому она столь популярна.

Выводы по 2 главе:

1. Пользовательский интерфейс должен обладать следующими чертами: доступность, минимализм, уверенность, отзывчивость, соответствие контексту, привлекательность, эффективность и снисходительность
2. Информационные технологии делятся на прикладные и системные, в свою очередь прикладной интерфейс делится на командный, Wimp - и Silk-интерфейсы
3. Командный интерфейс был первым, затем появились графический интерфейс и звуковой (речевой) интерфейс, в настоящее время все больше применяется трехмерный интерфейс

4. Трёхмерный интерфейс относится к интерфейсам прямой манипуляции, то есть пользователь не отдаёт команды системе, а манипулирует объектами

ГЛАВА 3. Практическое приложение требований к пользовательским интерфейсам

3.1. Инновации пользовательского интерфейса от Apple

В настоящее время Apple является крупнейшей мировой компанией, которая занимается выпуском компьютеров, сотовых телефонов и сопровождающей продукции. Основным отличием Apple от других компаний является их узнаваемость по всему миру, высокое качество и соответственно – высокая цена.

Пользовательский интерфейс во всех видах техники, которые выпускает Apple, идеально вписывается в требования к удачному интерфейсу. К примеру, все продукты Apple обладают достаточно похожими внешними характеристиками и интерфейсом, что позволяет предлагать пользователю интуитивно понятный интерфейс.

Компания Apple не стоит на месте, а постоянно предлагает пользователю все более новые решения его проблем. Единственным их недостатком является сложная совместимость с устройствами других компаний, что создает определенные трудности для пользователей.

В настоящий момент, компания занимается следующими разработками в области пользовательского интерфейса.

Несколько месяцев назад Apple подала на рассмотрение в Патентное бюро США ещё один свой патент, на этот раз касающийся графических пользовательских интерфейсов. Всем пользователям продуктов Apple (и не только) известна вещь под названием Cover Flow - это особый способ организации элементов списка (чаще всего, обложек музыкальных альбомов) (Рис.12).



Рисунок 12. Cover Flow

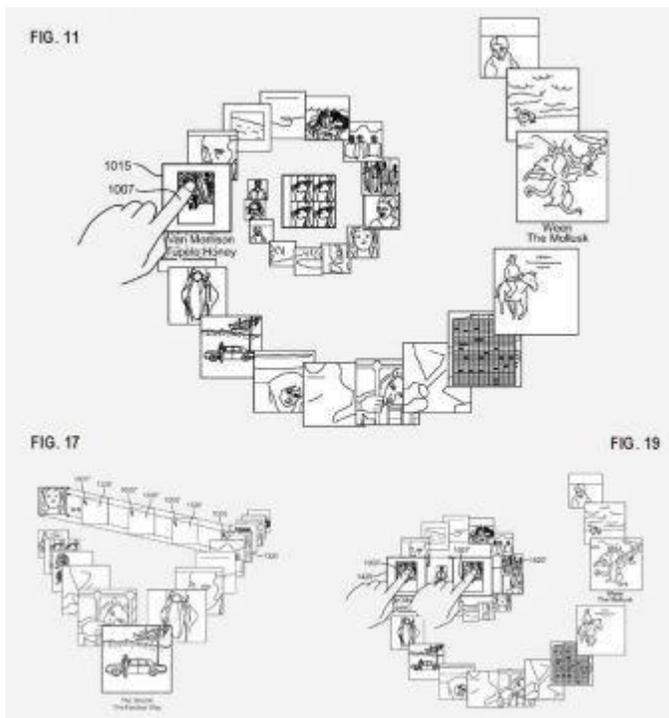


Рисунок 13. Spiral UI

Apple разработала замену Cover Flow. Именно эту замену они и пытаются запатентовать. Новая разработка называется Spiral UI (Рис.13), и главной основой данного способа организации списков является спираль. Благодаря Spiral UI представляется возможным вместить больше информации на меньший экран, что так актуально для мобильных устройств.

В ноябре 2011 года, компания Apple подала заявку на получение оригинального патента, предлагающего новый пользовательский интерфейс, объединяющий в себе компактный проектор, внутренние средства коммуникации и систему

распознавания жестов.

Новый патент Projected Display Shared Workspaces (Проецируемый дисплей разделяемого рабочего стола) позволяет пользователям взаимодействовать с изображениями, проецируемыми на стену компактными устройствами, такими как iPhone или iPad (Рис.3)

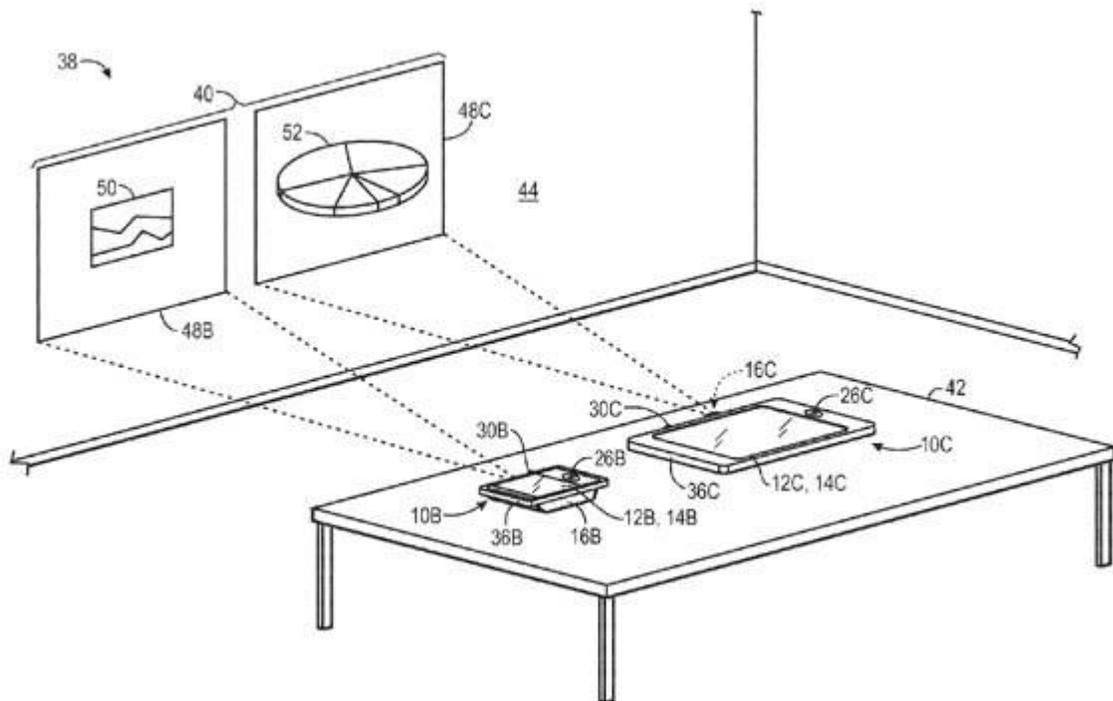


Рисунок 14. PDSW

В заявке на патент сказано, что уже существующие на рынке электронные устройства в теории могут работать с новой системой, однако здесь же замечается, что для больших групп зрителей такие решения вряд ли подойдут, так как проецируемое изображение не будет слишком большим, а для того, чтобы реализовать саму идею "разделяемости" рабочего стола между группой пользователей, устройство-излучатель придется передавать из рук в руки, что неудобно.

Дабы компенсировать это неудобство, Apple заявляет, что новая система должна работать с многими более мощными системами проецирования, такими как голографические лазеры, LED-проекторы и другие.

Кроме того, система проецирования предусматривает объединение нескольких устройств, передающих на стену изображения. Передача может производиться как напрямую устройствами, так и через устройство-посредник.

Система позволяет ее пользователям работать с традиционным для Apple жестовым интерфейсом, перемещая объекты между устройствами. Фиксация перемещений ведется при помощи камеры. В случае перемещения объектов, пользователи могут выставлять права на передачу файлов. Перед передачей файла пользователь-получатель может ознакомиться с содержимым файла еще на устройстве-передатчике за счет встроенной системы предпросмотра.

В Apple признают, что предлагаемый интерфейс представляет собой совершенно новое решение и для того, чтобы его реализовать, в самих устройствах придется реализовать несколько технологических новинок, в частности, значительно переписать программное обеспечение и изменить файловую систему, которая сейчас в iPhone и iPad скрыта от прикладных приложений.

3.2. Пользовательский интерфейс Windows 7

Интерфейс Windows 7 создан с целью повышения производительности, но вначале он может вызывать определенные трудности. Пользовательский интерфейс Windows 7 довольно значительно отличается от интерфейса Windows XP. Рассмотрим основные отличительные функции этой операционной системы.

- Aero Snap

Функция Aero Snap является долгожданным дополнением для любого пользователя, который обычно работает со многими окнами одновременно, но при этом не использует конфигурацию нескольких мониторов. В основе этой функции лежит идея того, что она позволяет разбивать экран на две половины, чтобы можно было просматривать одно окно в одной части экрана, а другое – в другой. Это можно было делать и в Windows XP и в Vista, но это было несколько сложнее.

В предыдущих версиях Windows, при необходимости просматривать несколько окон одновременно, нужно было перетаскивать эти окна в определенное место, а затем изменять их размер вручную. Было очень сложно задать всем окнам одинаковый размер.

В Windows 7 функция Aero Snap выполняет все эти действия за пользователя, ему лишь нужно взять окно и перетащить его в правую или левую половину экрана. Когда курсор приближается к краю экрана, рабочий стол Windows затемняется. Это сигнал того, что работает Aero Snap. Если отпустить кнопку мышки, цвета рабочего стола вернуться в прежнее состояние, а окно закрепится с правой или с левой стороны экрана и автоматически изменит свой размер в соответствии с размерами этой половины рабочего стола. Функция Aero Snap позволяет автоматически изменять размер окон в соответствии с размерами половины экрана.

Функция Aero Snap позволяет не только разбивать экран. Если перетащить окно в верхнюю часть экрана, окно автоматически развернется по максимальному размеру. Подобно этому, перетаскивание окна в нижнюю часть автоматически свернет его.

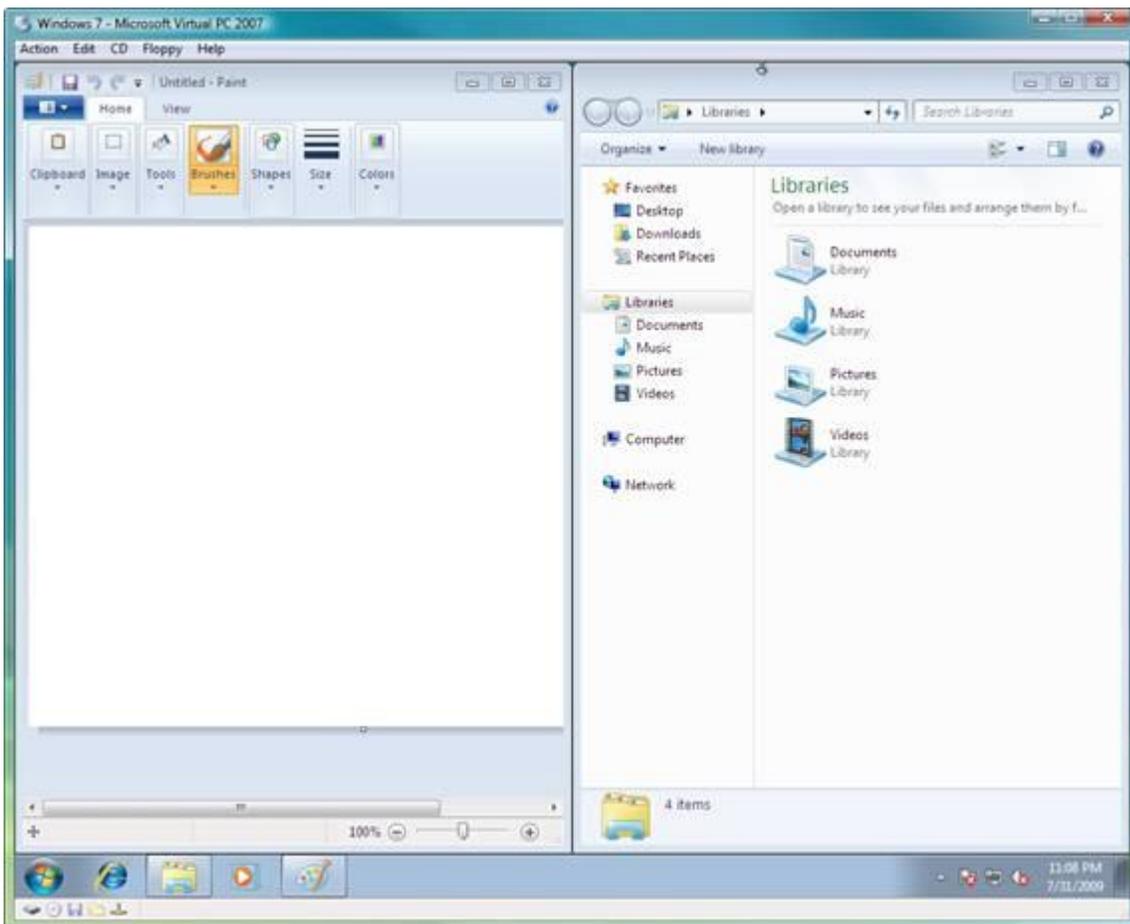


Рисунок 15. Расположение окон при помощи Aero Snap

- Функция Aero Shake

Она помогает ориентироваться во многих окнах, открытых на рабочем столе. Функция Aero shake создана для быстрого сворачивания всех окон за исключением того окна, которое в настоящее время используется. Для использования этой функции нужно просто взять окно приложения мышью и потрясти его несколько раз. После этого все окна других приложений будут свернуты. Если снова потрясти окно приложения, все окна приложений вернуться в исходное положение.

- Функция Aero Peek

Еще одной полезной функцией является функция под названием Aero Peek. В Vista если держать курсор мыши над открытым приложением в панели задач, предоставляется небольшой обзор этого приложения. Это отлично работало за исключением тех случаев, когда было открыто несколько окон, в результате чего обзоры окон наслаивались друг на друга. А это означало, что если открыто десять окон обозревателя, можно было видеть обзор только одного окна.

Aero Peek делает все немного по-другому. Если открыто несколько приложений, расположение курсора мыши над списком приложений в панели задач заставит Windows отобразить отпечаток обзора каждого приложения. Более того, можно нажать на этом обзоре, чтобы перейти непосредственно к этому окну. Эта функция экономит массу времени.

- Панель задач

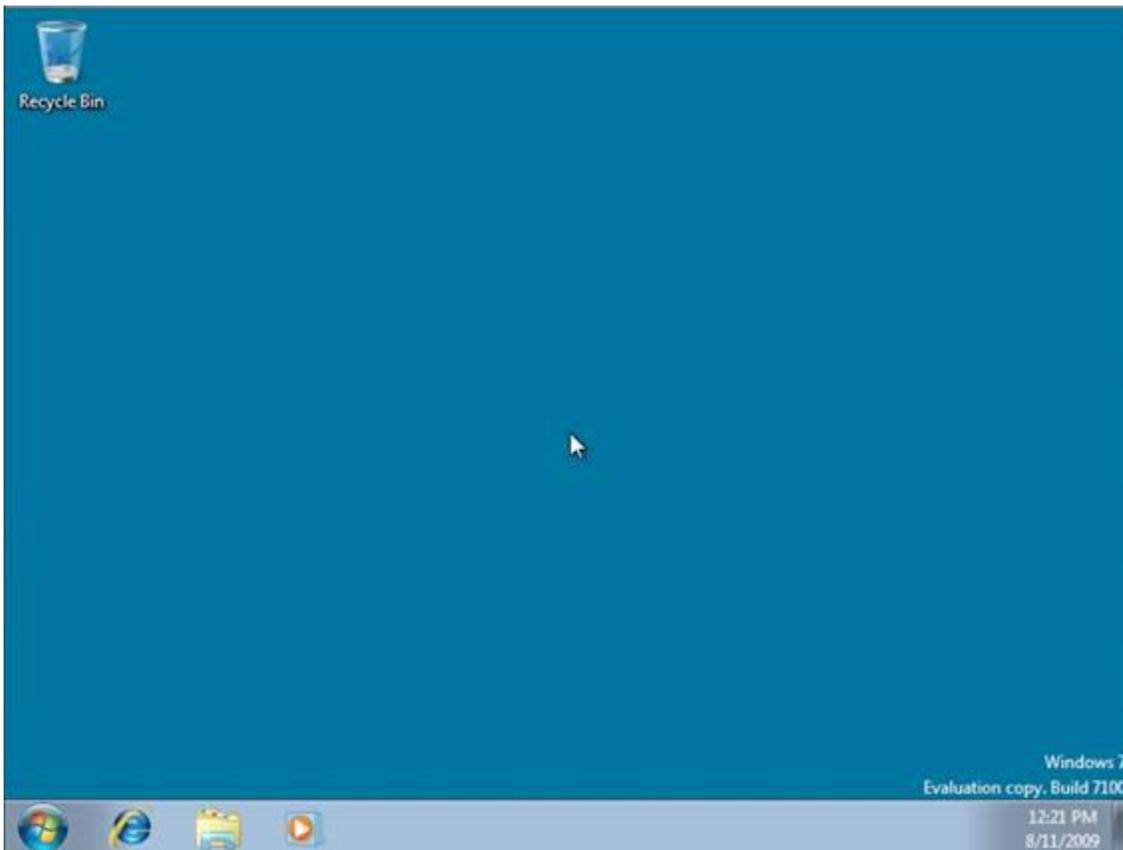


Рисунок 16. Панель задач Windows 7

Панель задач также была изменена в Windows 7, она немного шире панели задач в Vista и включает значки для Internet Explorer, Windows Explorer и Windows Media Player. Эти приложения закреплены в панели задач

Закрепление приложений в панели задач позволяет запускать их без необходимости использования меню Пуск или ярлыков на рабочем столе.

Если нужно закрепить приложения в панели задач, необходимо просто перетащить его из меню Пуск в панель задач.

- ○ Список перехода (Jump lists)

В Windows Vista меню Пуск содержало контейнер недавно использовавшихся документов (Recent Items), из которого был доступ к недавно использовавшимся документам. Проблема этого контейнера заключалась в том, что он не учитывал связи приложений с файлами, то есть могли быть MP3 файлы вместе с PDF файлами.

Функция списков перехода позволяет группировать последние документы по типу приложения. Расположение курсора на приложении в меню Пуск или панели задач

отобразит список документов, которые недавно открывались в этом приложении.

В случае с Internet Explorer эта функция отобразит список всех недавно посещенных веб страниц. Это означает, что теперь можно навести курсор на значок приложения в панели задач и посмотреть список недавно открывавшихся в нем файлов, а также перейти к этим файлам без необходимости предварительного запуска приложения. Эта функция избавляет от необходимости поиска списков открывавшихся файлов этого приложения.

Таким образом, пользовательский интерфейс Windows 7 отличается от интерфейсов Windows XP и Vista и сделает работу Windows более удобной даже на начальной стадии освоения этой ОС.

Выводы по 3 главе:

1. Apple предлагает все более новые «примочки» для пользовательских интерфейсов, например Spiral UI и проецируемый дисплей разделяемого рабочего стола
2. Пользовательский интерфейс Window 7 позволяет облегчить работу пользователя путем внедрения Aero Peek, Aero Snap, Aero Shake и Jump Lists.

Заключение

Таким образом, информационные технологии представляют собой совокупность методов, производственных процессов и программно-технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, обработку, хранение, передачу и отображение информации.

На данный момент существует множество видов информационных технологий, которые классифицируются в зависимости от типа обрабатываемой информации, от того, какие задачи они выполняют, по типу пользовательского интерфейса, по участию пользователя в информационном процессе и по степени их взаимодействия между собой.

Любая информационная система создается на основании определенных принципов, главным из которых является принцип «открытости» системы, что позволяет, с одной стороны, значительно облегчить работу разработчиков, а, с

другой стороны, упростить работу в системе для пользователя.

Каждая информационная система состоит из функциональной и системной частей. Эталонная модель «открытой» информационной системы состоит из 7 уровней. Информационную систему для любой компании создают на основании её онтологического поля и полного анализа компании, затем проектируется информационная система, далее осуществляется процесс её внедрения и затем отслеживается процесс её эксплуатации на предприятии и осуществляется консультативная помощь при возникновении неполадок.

Интерфейс - это определенная стандартами граница раздела двух систем, устройств или программ.

Пользовательский интерфейс создается с соблюдением определенных требований, он должен быть естественным, согласованным, интуитивно понятным, простым, гибким, эстетически привлекательным и обладать системой «обратной связи». При выполнении вышеперечисленных требований, интерфейс становится удобным в использовании.

По типам пользовательского интерфейса выделяют командный интерфейс, wimp - и silk - интерфейсы. Командный интерфейс реализован в виде пакетной технологии и технологии командной строки. Wimp-интерфейс представляет собой диалог пользователя с компьютером с помощью графических образов, а silk-интерфейс наиболее приближен к «человеческой» форме общения.

Трёхмерный интерфейс так же является разновидностью пользовательских интерфейсов, он используется в приложениях компании Apple – Spiral UI и проецируемый дисплей разделяемого рабочего стола и в Windows 7 (реализуется при помощи функций Aero Snap, Aero Shake, Aero Peek и Jump Lists).

Список используемой литературы

1. Пушкарев человеческими ресурсами на основе компьютерных технологий / Под ред. . – М: Финансы и статистика, 2009. – 176с.
2. Алиев технологии и системы финансового менеджмента: учебное пособие. - М.: «ФОРУМ»: ИНФРА-М. 2007

3. Информационные технологии (для экономиста): учебное пособие / ред. . – М.: ИНФРА-М, 2001. – 309 с.
4. Грабауров технологии для менеджеров. – М.: Финансы и статистика, 2001. – 368 с.
5. Кияев технологии в управлении. БИНОМ. Лаборатория знаний, Интернет-университет информационных технологий - ИНТУИТ. ру, 2008
6. Бондаренко технологии управления. – М.: ЮНИТИ-ДАНА, 2008
7. Плюснина информационной системы: учебное пособие. – Котлас, 2006 – 62.с
8. Козырев технологии в экономике и управлении: Учебник. - СПб.: Изд-во , 2008
9. Павленко технологии в экономике и управлении. – М.: КНОРУС, 2007
10. Информатика: учебник для вузов / и др. М.: Финансы и статистика, 2004. – 768 с.
11. Информатика Учебник. / Под ред . - 4-е изд. - М.: Финансы и статистика, 2007
12. Могилев и др. Информатика: Учебное пособие для вузов / , , ; Под ред. . - М.: Изд. центр "Академия", 2008
13. Советов технологии: Учеб. Для вузов. - М.: Высш. шк., 2005
14. Степанов : учебник для вузов. - СПб.: Питер, 2005. – 684 с.
15. Информационные технологии управления: учеб. Пособие для вузов / Под ред. Проф. . 2-е изд. доп. М.: ЮНИТИ-ДАНА, 2003
16. Информационные системы и технологии в экономике и управлении: Учебник / Под ред. проф. . – М.: Высшее образование, 2009. – 480с.
17. Черников технологии управления.– Изд: Форум, Инфра-М, 2008 гс.
18. Цветкова и информационные технологии. Конспект лекций./ – Изд: Эксмо, 2007 гс.