

Пользовательский интерфейс: назначение, принципы разработки и практические рекомендации

(подготовлено И.В.Горячей)

Предлагаемые учебно-методические указания адресованы для студентов первого курса факультета ВМК МГУ имени М.В.Ломоносова и могут быть для них полезны при выполнении практических заданий на ЭВМ [4]. Кроме того, данные материалы могут быть интересны всем начинающим программистам. В материалах объясняется назначение пользовательского интерфейса, принципы его разработки и даются практические рекомендации по некоторым аспектам его реализации.

1. Назначение пользовательского интерфейса

Ценность любой программы определяется не только её способностью безошибочно и качественно решать поставленную задачу, но и тем, насколько легко и удобно этой программой можно пользоваться. Даже самая простая, и в то же время, функциональная программа окажется невостребованной, если пользователь (т.е. конечный потребитель программы) при работе с ней быстро устаёт, раздражается, чувствует себя некомфортно. В этом случае пользователь обычно совершает больше ошибок и работает менее продуктивно (т.е. неэффективно). Для получения действительно удобной программы, с которой бы было приятно работать, необходим хорошо продуманный дружественный интерфейс с пользователем.

Под **пользовательским интерфейсом** (англ. user interface) будем понимать своеобразный коммуникационный канал (связующее звено), посредством которого пользователь взаимодействует с компьютером. Пользовательский интерфейс является важнейшей частью любой программы, так как именно с интерфейсом пользователь соприкасается в процессе своей работы. С точки зрения пользователя именно интерфейс является конечным продуктом. **Интерфейс** представляет собой совокупность используемых в программе средств для ввода данных, способов отображения информации на экране дисплея, элементов для управления различными этапами решения задачи. Основное **назначение (главная цель) пользовательского интерфейса** – обеспечить короткий и удобный путь к получению результата.

2. Основные принципы разработки пользовательского интерфейса

Сформулируем **принципы разработки пользовательского интерфейса**, соблюдение которых повышает его удобство и делает его дружественным по отношению к пользователю.

1) **Принцип структуризации.** Пользовательский интерфейс должен быть структурирован так, чтобы близкие по смыслу, родственные его части были связаны видимым образом, а независимые – разделены; похожие элементы должны выглядеть похоже, а непохожие – различаться.

2) **Принцип простоты.** Наиболее распространённые действия должны выполняться максимально просто, чтобы существенно снизить затрачиваемые усилия.

3) Принцип наблюдаемости. Все операции и данные, необходимые для решения определённой задачи, должны быть хорошо видны, когда пользователь пытается её решить. Излишняя и ненужная информация не должна отвлекать внимание пользователя.

4) Принцип обратной связи. Пользователь должен получать сообщения о действиях программы и о важных событиях внутри неё. Сообщения должны быть ясными, краткими, недвусмысленными и написанными на языке, понятном пользователю.

5) Принцип толерантности. Интерфейс должен быть гибким и терпимым к ошибкам пользователей. Ущерб от ошибок должен снижаться за счёт возможности отмены или повтора действий, а также за счёт разумной интерпретации любых разумных действий пользователя и введённых им данных (например, допускать различную последовательность выполнения этапов задачи и различные формы ввода исходных данных).

6) Принцип повторного использования. Чтобы поведение программы было предсказуемо, полезно повторно использовать известные приёмы и решения, применяемые при разработке интерфейсов для других программ.

3. Практические рекомендации по некоторым аспектам реализации пользовательского интерфейса

Неудачные интерфейсы сильно замедляют работу пользователей, способствуют быстрой утомляемости, большому числу ошибок и, как следствие, отбивают у пользователей дальнейшее желание работать с программой. В связи с этим разработчику программы следует учитывать известные рекомендации ведущих специалистов в области создания пользовательского интерфейса [1,2]. Приведём здесь лишь некоторые наиболее полезные, с нашей точки зрения, рекомендации.

Распределение информации на экране дисплея.

Логичнее и естественнее всего начинать взаимодействие с пользователем с **левого верхнего угла экрана**, постепенно переводя внимание пользователя на **правый нижний угол**. Тяжелее всего работать с интерфейсами, отдельные части которых расположены на экране в хаотическом, непредсказуемом порядке. В этом случае пользователь вынужден постоянно переключать свой взгляд на разные, не связанные между собой участки экрана для поиска нужной информации. Особенно плохо, если элементы (визуализируемые на экране объекты) при этом отличаются миниатюрными размерами.

Не следует также перегружать экранное пространство различными второстепенными элементами (объектами). Пользователи в этом случае будут постоянно путаться в интерфейсе, теряться среди нагромождения различных (как основных, так и второстепенных) элементов (объектов). Не надо впадать и в другую крайность, когда на экране остаётся огромное количество свободного места. Всё это приводит к неэффективному использованию экранного пространства.

Опытные разработчики пользовательских интерфейсов считают, что нужно стремиться к сбалансированности при выборе местоположения для управляющих элементов и видимых объектов: слева и справа, сверху и снизу их количество должно быть примерно одинаковым. Не стоит во что бы то ни стало добиваться при этом идеальной симметрии. Однако слишком перекошенный интерфейс вряд ли может способствовать созданию у пользователя ощущения комфорта. Не последнюю роль в эстетической привлекательности интерфейса играет **выравнивание** элементов (объектов) **по верхним и левым границам экрана**, соблюдение пропорций при выборе размеров для изображения этих элементов (объектов).

Управление вниманием пользователя (использование цвета, анимации, звука, шрифтов, курсора).

Как правило, интерфейс имеет одну точку, в которой должен **сейчас** происходить процесс взаимодействия между пользователем и программой – так называемую **точку фокуса**. Если процессом взаимодействия управляет **пользователь**, то в фокусе обычно находится текущая выборка (например, альтернатива меню) или мигающий курсор (при наборе данных с клавиатуры). Если на действия пользователя отвечает **программа**, то в фокусе обычно находится результат действия.

Хороший интерфейс должен сразу привлекать (приковывать) внимание пользователя к точке фокуса, к самой важной (наиболее существенной в данный момент работы с программой) информации. Пользователь должен чётко понимать, где на экране размещена наиболее значимая для него информация, а где – второстепенная. В связи с этим отметим **основные приёмы для управления вниманием пользователя**.

Установлено, что наибольшее внимание пользователя привлекают **анимированные** (то есть движущиеся или мигающие) **визуальные объекты**. Однако злоупотребление этим приёмом может привести к негативному эффекту: усталости глаз, чувству раздражения (т.е. в этом надо знать меру!)

Управлять вниманием пользователя можно также и с помощью **цветового выделения** (кодирования). Но и в этом необходимо знать чувство меры. Большие области ярких, интенсивных цветов могут сбивать пользователей с толку и вызывать быструю утомляемость при длительной работе с программой. Некоторые сочетания цветов, такие как зелёный на красном или красный на зелёном, особенно негативно влияют на человека. Свойство цвета привлекать внимание пользователя вовсе теряется, если в яркие кричащие тона раскрашены все элементы интерфейса. Не стоит также одновременно использовать большое количество цветов; практический предел – не более 4-7 цветов.

Существует договорённость, согласно которой так называемыми **тёплыми цветами** (*розовым, жёлтым, оранжевым*) обозначаются наиболее **активные** части интерфейса; **холодными цветами** (*зелёным, синим, белым*) – обозначаются более **пассивные** части интерфейса.

Следует иметь в виду и то, что встречается немало людей, страдающих той или иной формой дальтонизма (цветовой слепотой). Это примерно одна двенадцатая часть

всех мужчин и чуть меньшая доля женщин. Многие из них, например, не могут отличить зелёный цвет от красного. Это надо учитывать при подборе цветов для изображения интерфейсных элементов.

Цветовое выделение, как правило, сочетается с другими изобразительными приёмами. Например, часто, чтобы подчеркнуть важность некоторой информации, её заключают в **рамки** (одинарные, двойные, анимированные). Иногда, для привлечения внимания к некоторому заголовку или названию соответствующий текст выводит только **заглавными буквами**. Используются также и **звуковые эффекты**, однако, в некоторых ситуациях (например, при работе в коллективе) звуки могут вызывать раздражение. Поэтому звуковыми эффектами злоупотреблять не нужно, пользоваться ими весьма осторожно и иметь возможность отключать звук в любой необходимый момент.

Наконец, традиционным элементарным средством для привлечения внимания пользователя является **курсор**. Курсор – это мигающее изображение на экране, чаще всего, представленное в виде чёрточки или прямоугольника. Обычно (при вводе текста), появление курсора рассматривается пользователем как приглашение к вводу данных и указывает на позицию, начиная с которой набираемые с клавиатуры символы будут отображаться на экране дисплея. **Форма курсора** часто говорит о текущем режиме ввода символов (нижнее подчёркивание – режим **вставки**, прямоугольник – режим **наложения**).

4. Заключение

Тщательно продуманный, лёгкий в изучении и понимании, учитывающий человеческие возможности и ограничения интерфейс гарантирует удобство работы с программой и, в конечном итоге, её коммерческий успех.

Приведём, в заключение, слова из книги [3] Станислава Жаркова: “Лучший пользовательский интерфейс – это такой интерфейс, которому пользователь не должен уделять много внимания, почти не замечать его. Пользователь просто работает, вместо того, чтобы размышлять, какую кнопку нажать... Такой интерфейс называется **прозрачным** – пользователь как бы смотрит сквозь него на свою работу”. Именно к такому интерфейсу следует стремиться при написании программ.

Литература

1. Джеф Раскин. Интерфейс: новые направления в проектировании компьютерных систем. Символ-Плюс, 2004, 272 стр.
2. Л.Константайн, Л.Локвуд. Разработка программного обеспечения. Изд. Питер, 2004, 592 стр.
3. С. Жарков. Shareware: профессиональная разработка и продвижение программ. БХВ-Петербург, 2002, 320 стр.
4. Трифонов Н. П., Пильщиков В. Н. Задания практикума на ЭВМ (первый курс): Учебное пособие для вузов. – М.: МАКС-Пресс, 2001.