Джефф Джонсон

УМНЫЙ ДИЗАЙН

Простые приемы разработки пользовательских интерфейсов

MK
MORGAN KAUFMANN

бесплатно:

ПИТЕР

Jeff Johnson

DESIGNING WITH THE MIND IN MIND:

Simple Guide to Understanding User Interface Design Rules

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier

MK MORGAN KAUFMANN

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Джефф Джонсон

УМНЫЙ ДИЗАЙН

Простые приемы разработки пользовательских интерфейсов

ПИТЕР®

Москва · Санкт-Петербург · Нижний Новгород · Воронеж
Ростов-на-Дону · Екатеринбург · Самара · Новосибирск
Киев · Харьков · Минск
2012

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Джефф Джонсон

Умный дизайн: Простые приемы разработки пользовательских интерфейсов

Серия «Библиотека специалиста»

Перевела с английского Е. Шикарева

Заведующий редакцией	*А. Кривцов*
Ведущий редактор	*Ю. Сергиенко*
Литературный редактор	*Н. Рощина*
Художественный редактор	*К. Радзевич*
Корректор	*В. Листова*
Верстка	*Л. Родионова*

ББК 32.973.23-044

УДК 004.5

Джонсон Дж.

Д42 Умный дизайн: Простые приемы разработки пользовательских интерфейсов. — СПб.: Питер, 2012. — 224 с.: ил.

ISBN 978-5-459-01616-1

Неправильно спроектированный интерфейс может погубить любой интернет-проект. Но для того чтобы разработать совершенный пользовательский интерфейс, необходимо быть специалистом в разных областях знания: от графики и дизайна до когнитивной психологии и теории распознавания образов. Как дизайнер может использовать знания этих дисциплин в своей работе? Джефф Джонсон, автор бестселлеров «Ляпы GUI» и «Web-дизайн: типичные ляпы и как их избежать», предоставляет дизайнерам уникальную возможность использовать всю мощь компьютерной графики и теории распознавания образов для разработки совершенных пользовательских интерфейсов.

Книга необходима дизайнерам, разработчикам UI и веб-разработчикам.

ISBN 978-0123750303 англ.
ISBN 978-5-459-01616-1

© Elsevier Inc., 2010
© Перевод на русский язык ООО Издательство «Питер», 2012
© Издание на русском языке, оформление ООО Издательство «Питер», 2012

Права на издание получены по соглашению с Morgan Kaufmann Publishers. Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельцев авторских прав.

Информация, содержащаяся в данной книге, получена из источников, рассматриваемых издательством как надежные. Тем не менее, имея в виду возможные человеческие или технические ошибки, издательство не может гарантировать абсолютную точность и полноту приводимых сведений и не несет ответственности за возможные ошибки, связанные с использованием книги.

ООО «Мир книг», 198206, Санкт-Петербург, Петергофское шоссе, 73, лит. А29.
Налоговая льгота — общероссийский классификатор продукции ОК 005-93, том 2; 95 3005 — литература учебная.
Подписано в печать 28.05.12. Формат 70x100/16. Усл. п. л. 18,060. Тираж 2000. Заказ
Отпечатано по технологии CtP в ГППО «Псковская областная типография».
180004, Псков, ул. Ротная, 34.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Оглавление

Благодарности 9

Предисловие 11

Введение 15

Правила создания пользовательских интерфейсов: откуда они берутся и как их эффективно применять? 16

Для разработки интерфейсов и организации взаимодействия с пользователем требуются понимание и опыт 16

Сравнение рекомендаций по проектированию пользовательских интерфейсов 18

Кто создает рекомендации по проектированию 19

Целевая аудитория книги 19

Глава 1. Наше восприятие зависит от ожиданий 21

Опыт искажает восприятие 22

Текущий контекст искажает восприятие 25

Цели искажают восприятие 26

Выводы, влияющие на дизайн 29

Избегайте двусмысленности 29

Следите за единообразием 30

Понимайте цели 30

Глава 2. Наше зрение оптимизировано для восприятия структуры 31

Гештальт-принцип: близость 32

Гештальт-принцип: схожесть 35

Гештальт-принцип: непрерывность 36

Гештальт-принцип: замкнутость 38

Гештальт-принцип: симметрия 39

Гештальт-принцип: «фигура/фон» 40

Гештальт-принцип: общая судьба 43

Гештальт-принципы: всё вместе 44

Глава 3. Ищем и используем визуальную структуру 47

Структурирование помогает читать длинные номера и числа 51

Специфические элементы управления улучшают структурирование ввода 52

Визуальная иерархия помогает выделить важную информацию 53

Глава 4. Чтение противоестественно 55

Мы запрограммированы на понимание языка, а не на чтение 56

Чтение — от характеристик или от контекста? 58

У опытных и неопытных читателей работают разные части мозга 61

Плохое оформление информации мешает чтению 62

Специфические или незнакомые термины 63

Сложные написания и шрифты 64

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Мелкий шрифт 64
Текст на шумном фоне 65
Информация прячется среди многочисленных повторений 67
Выравнивание по центру 67
Сочетание недостатков, мешающих чтению 69
Выводы для дизайнера: не мешайте чтению, помогайте читателю! 69
Большую часть написанного читать необязательно 70
Выводы для дизайнера: минимизируйте необходимость в чтении 73
Проверяйте на реальных людях 74

Глава 5. Цветовое зрение ограничено 75
Как работает цветовое зрение 76
Зрение оптимизировано для восприятия контраста в пограничных зонах 78
Способность различать цвета зависит способа их представления 79
Цветовая слепота 81
Внешние факторы, влияющие на способность различать цвета 83
Рекомендации по использованию цвета 83

Глава 6. Периферическое зрение плохо работает 87
Разрешение в ямке по сравнению с разрешением на периферии поля зрения 88
Есть ли польза от зрительной периферии 91
Примеры пользовательских интерфейсов 92
Как сделать сообщение заметным 95
Осторожно! Тяжелая артиллерия для создания заметных сообщений 97
Всплывающее сообщение в диалоговом окне 97
Звуковой сигнал 97
Короткое мерцание или колебание 98
Расчетливо применяйте эти методы 98

Глава 7. Внимание ограниченно, а память неидеальна 101
Краткосрочная и долгосрочная память 102
Современный взгляд на память 103
Долгосрочная память 103
Краткосрочная память 104
Характеристики краткосрочной памяти 106
Режимы 109
Результаты поиска 110
Инструкции 112
Характеристики долгосрочной памяти 113
Склонность к ошибкам 113
Усиливается эмоциями 114
Ретроактивные изменения 114
Влияние долгосрочной памяти на дизайн пользовательских интерфейсов 115

Глава 8. Ограниченность внимания влияет на мысли и действия 121
Мы концентрируемся на целях и почти не обращаем внимания на инструменты 122
Мы следим за ходом выполнения задачи, используя внешние подсказки 123

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Мы идем к цели по информационному «следу» 124
Мы предпочитаем знакомые пути 127
Цикл мышления: цель, выполнение, оценка. 128
Главная цель достигнута, не забудьте «прибрать за собой». 131

Глава 9. Узнавать легко, вспоминать трудно. 133

Узнавать просто 134
Вспоминать трудно 137
Узнавание или вспоминание: почему они важны для разработчиков пользовательских интерфейсов 138
Используйте эскизы для компактного представления полноразмерных изображений . . 140
Чем больше людей будут использовать функцию, тем заметнее она должна быть 141
Добавляйте визуальные подсказки, чтобы пользователи могли ориентироваться 141
Помогайте пользователям вспоминать данные для аутентификации 143

Глава 10. Учиться на своем опыте легко, решать задачи сложно 145

У нас три мозга 146
Учиться на опыте (обычно) легко 147
Выполнять заученные действия просто 149
Решать задачи и вычислять трудно. 152
Выводы для дизайнера пользовательских интерфейсов 157
Ответы на головоломки 160

Глава 11. На обучение влияет множество факторов 161

Мы быстрее учимся, когда наша деятельность сфокусирована на задаче, проста и единообразна. 162
Анализ задачи. 164
Анализ объектов/действий 165
Как можно проще 166
Единообразие 167
Матрица объектов/действия 168
Единообразие нажатий клавиш 171
Мы учимся быстрее, когда терминология сфокусирована на задаче, знакома и единообразна. 172
Терминология должна быть сфокусирована на задаче 173
Терминология должна быть знакомой 174
Терминология должна быть единообразной. 176
Разработать сфокусированную на задаче, знакомую и единообразную терминологию проще, когда у вас есть хорошая концептуальная модель 179
Мы учимся быстрее, когда риск низок 180
Резюме 181

Глава 12. Требования к времени. 183

Что такое отзывчивость 184
Временные константы человеческого мозга 186
Самый короткий интервал тишины, который мы способны услышать в звуковом потоке: 1 мс 189

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Минимальное время, в течение которого необходимо наблюдать визуальный раздражитель, чтобы он повлиял на нас (возможно, подсознательно): 5 мс *189*
Скорость непроизвольной моторной реакции на возможную опасность: 80 мс *189*
Задержка между визуальным событием и моментом, когда мы полностью воспринимаем его: 100 мс *190*
Максимальный интервал между событиями, которые мы воспринимаем как причину и следствие: 140 мс *190*
Время, необходимое для определения числа элементов в поле зрения (до четырех-пяти элементов): 200 мс (50 мс на элемент) *190*
«Установочное окно» для событий при достижении ими сознания: 200 мс *191*
Невосприимчивость к поступающей информации, следующая за узнаванием объекта: 500 мс *191*
Время визуально-моторной реакции (осознанная реакция на неожиданное событие): 700 мс *191*
Максимальная продолжительность паузы между фразами разных людей во время диалога: около 1 с *192*
Время, в течение которого внимание непрерывно уделяется одной задаче: около 10 с . . *192*
Внедрение временных констант в программное обеспечение: порядки величин 193
Учет предельных периодов взаимодействия 194
0,001 секунды *196*
0,01 секунды *196*
0,1 секунды *197*
1 секунда *198*
10 секунд *199*
100 секунд (~1,5 мин) *199*
Дополнительные рекомендации по проектированию отзывчивых интерактивных систем . .199
Используйте индикаторы занятости *199*
Используйте индикаторы прогресса. *200*
Задержки между единицами задачи не так беспокоят пользователей, как задержки внутри единиц задачи *201*
В первую очередь показывайте важную информацию. *202*
Имитируйте тяжеловесные вычисления во время задач, требующих зрительно-моторной координации *204*
Старайтесь предугадать действия пользователей *204*
Обрабатывайте вводимые пользователем данные согласно приоритету, а не в порядке поступления. *205*
Отслеживайте выполнение временных требований; снижайте качество работы, чтобы не допускать опозданий. *205*
Обеспечивайте своевременную ответную реакцию даже в Сети *206*
Отзывчивость очень важна 207

Резюме 209
Предостережение 210

Приложение. Правила разработки пользовательских интерфейсов 213

Библиография 219

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Благодарности

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Я бы не смог написать эту книгу без помощи и поддержки многих людей.

Во-первых, необходимо упомянуть слушателей курса «Взаимодействие человека и компьютера», который я вел зимой 2006 года как стипендиат фонда Эрскина (Кентерберийский университет, Новая Зеландия). Именно благодаря мне удалось подготовить лекцию, посвященную краткому знакомству с перцепционной и когнитивной психологией и помогающую понять принципы проектирования пользовательских интерфейсов. Эта лекция сначала превратилась в курс повышения квалификации, а затем в эту книгу.

Во-вторых, я безмерно благодарен редакторам чернового варианта рукописи: Сьюзан Фоулер (Susan Fowler), Робину Джеффрису (Robin Jeffries), Тиму Маккою (Tim McCoy) и Джону Мидсу (Jon Meads). Они сделали множество полезных замечаний и внесли разумные предложения, благодаря которым мне удалось значительно улучшить книгу.

В-третьих, нельзя обойти благодарностью исследователей, занимающихся когнитивистикой, то есть наукой о мышлении. Они предоставили мне массу ценных материалов, подсказали полезную литературу и даже позволили отталкиваться от озвученных ими идей. Это профессора Эдвард Адельсон (Edward Adelson) (факультет когнитивных наук и наук о мозге Массачусетского технологического института) и Дэн Ошерсон (Dan Osherson) (факультет психологии Принстонского университета) и доктор Дэн Баллок (Dan Bullock) (факультет когнитивных наук и нейронных систем Бостонского университета).

Благодарю за заботу, уход, помощь с логистикой и общее содействие сотрудников Elsevier, особенно Мэри Джеймс (Mary James), Дэвида Биванса (David Bevans) и Андре Куэлло (Andre Cuello).

Огромную помощью с редактированием оказала мне Кейт де Хир. И самая большая благодарность моей жене и другу Карен Андэ — за ее любовь и поддержку, которые так помогли мне. В это время Карен тоже писала книгу — Face to Face: Children of the AIDS Crisis in Africs — фотоальбом, посвященный африканским детям, осиротевшим из-за СПИДа (http://FaceToFaceAfrica.com).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Предисловие

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Проектирование интерактивных компьютерных систем — это не только искусство, но и наука — по крайней мере, как мы это себе представляем. Может быть, не совсем наука, но все же определенное смешение компьютерного и когнитивного моделирования, представляющее собой основанные на научных разработках техники создания интерактивных систем, удовлетворяющих определенным требованиям.

Как автомобили, здания и одежда интерактивные компьютерные компоненты могут доставлять эстетическое удовольствие, демонстрировать стиль и моду и иметь социальную значимость. Объекты, умеющие всплывать, мерцать и взаимодействовать, открывают огромное пространство для творчества и занятий промышленным дизайном. Однако результаты их использования должны также правильно работать, согласовываясь с привычной человеческой деятельностью. Прекрасный дом с огромными окнами, в котором жильцы летом изнывают от жары, а во время урагана трясутся от страха из-за вибрирующих стен, — это архитектурный провал. Еще на этапе проектирования инженер должен спрогнозировать тепловую нагрузку на здание, применяя методы, учитывающие географическую широту, климат, распределение оконных проемов, внутренние объемы и циркуляцию воздуха. Также ему потребуется воспользоваться различными технологическими решениями, такими как многослойное изоляционное стекло, жалюзи, выступающие элементы здания и вентиляторы, и выбрать из предлагаемого разнообразия наиболее удачные и подходящие для устранения конкретной проблемы. Проектирование не заменяет искусство в строительстве, оно делает возможным достижение самых амбициозных целей.

Проектирование — сложный этап строительства. Не менее сложен он и при создании интерактивных объектов — по той простой причине, что намного проще изучить тонкости возведения зданий, чем человеческой природы. Необходимость в научной базе и вспомогательных инструментах проектирования стала причиной возникновения новой научной дисциплины — изучения взаимодействия человека и компьютера. Но как это осуществляется на практике? Самый простой метод называется тестированием удобства в использовании. Вы наблюдаете за пользователями, выполняющими задачи, определяете, с какими сложностями они сталкиваются, и исправляете вызывающие эти сложности недостатки дизайна. Тестирование удобства в использовании — полезный, необходимый и малопродуктивный метод. Результаты трудно учесть с помощью инструментов проектирования, и они не слишком хорошо помогают понять, почему же вещи ломаются. Это когнитивный эквивалент «поджаривания» пользователей на солнце с целью изучить влияние больших окон. И все же тестирование удобства в использовании позволяет обнаружить многие недостатки системы. Кроме того, это очень доступный метод, так как интерактивную систему обычно поменять намного проще, чем перестроить здание.

Лучше всего, конечно, с самого начала избегать ошибок, и один из способов добиться этого — применять *правила проектирования*. Вместо того чтобы раз за

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

разом обнаруживать (путем тестирования удобства), что красно-зеленые интерфейсы неудобны для пользователей, не различающих цвета, просто сформулируйте для себя правило проектирования — использовать другие оттенки. Тем не менее у правил проектирования есть свои недостатки. На практике они могут оказаться двусмысленными, или требовать сложного толкования контекста, или противоречить другим правилам. И это приводит нас к необходимости иметь под рукой такую книгу, как эта.

Идея данной книги заключается в том, чтобы объединить правила проектирования с соответствующими принципами когнитивной и перцепционной науки, лежащими в основе восприятия мира человеком. У такого формата несколько преимуществ: научные аспекты психологии преподносятся в конкретном и простом для понимания виде благодаря актуальным дизайнерским примерам, а правила проектирования становится проще осмысливать и связывать с контекстом, когда познакомишься с их логическим обоснованием.

Джеф Джонсон — самый подходящий человек для написания такой книги. На протяжении всей своей карьеры он занимался как дизайном интерфейсов, так и психологическими науками. Я познакомился с ним, когда он работал в команде разработки пользовательского интерфейса линейки продуктов Xerox Star, — и это был первый коммерческий графический интерфейс пользователя. Таким образом, ему довелось наблюдать зарождение такого понятия, как GUI. Что касается психологии, Джеф получил степени в Йельском и Стэнфордском университетах. Объединяя дизайн и психологию, он работал над коммерческими интерактивными системами, преподавал в университете и оказывал услуги в качестве консультанта. Его визитная карточка — использование конкретных примеров дизайна для иллюстрирования абстрактных принципов. На самом деле, он стал известен благодаря запоминающимся примерам неудачного дизайна — провальным интерфейсам, позволяющим наглядно донести свою точку зрения до слушателя. Точно так же он поступает и в этой книге.

Есть и третий метод применения научных методов для разработки систем, выходящий за пределы правил проектирования, — *модели дизайна*. В книге Джефа подробно рассказывается и об этом методе. Он показывает, как смоделировать задачу в терминах объектов и действий и как разобраться в ограничениях, накладываемых на взаимодействие в реальном времени.

Итак, эта книга поддерживает инженерный подход к дизайну интерактивных систем и в то же время очень понятно объясняет первоисточники базовых принципов человеческого поведения — это своего рода когнитивная наука для вечно спешащих дизайнеров. Но что важнее всего, книга позволяет практичному человеку, желающему качественно выполнить свою работу, глубоко заглянуть в хитросплетения человеческого разума и сделать ценные выводы.

Стюарт Кард (Stuart Card)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Введение

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Правила создания пользовательских интерфейсов: откуда они берутся и как их эффективно применять?

На всем протяжении истории создания интерактивных компьютерных систем люди пытались содействовать распространению *хорошего* дизайна, публикуя рекомендации по разработке пользовательских интерфейсов (также известные как правила проектирования). Среди первых работ необходимо отметить следующие:

- Черитон (Cheriton, 1976) предложил правила проектирования пользовательских интерфейсов для первых интерактивных компьютерных систем (с разделением времени);
- Норман (Norman, 1983a, 1983b) представил правила проектирования программных пользовательских интерфейсов, учитывающие особенность когнитивной деятельности человека, в том числе когнитивные ошибки;
- Смит и Мозье (Smith, Mosier, 1986) создали, вероятно, наиболее всеобъемлющий набор рекомендаций по проектированию пользовательских интерфейсов;
- Шнайдерман (Shneiderman, 1987) впервые включил «Восемь золотых правил дизайна интерфейсов» в первое издание книги «Разработка пользовательских интерфейсов» (Designing the User Interface);
- Браун (Brown, 1988) дал своей книге о правилах проектирования очень подходящее название «Рекомендации по дизайну взаимодействия человека и компьютера» (Human — Computer Interface Design Guidelines);
- Нильсен и Молич (Nielsen, Molich, 1990) предложили набор правил проектирования для использования в эвристической оценке пользовательских интерфейсов;
- Маркус (Marcus, 1991) представил рекомендации по графическому дизайну сетевой документации и пользовательских интерфейсов.

В XXI веке новые рекомендации по разработке пользовательских интерфейсов были предложены Стоуном (Stone et al., 2005), Койани, Бейли и Нэллом (Koyani, Bailey, Nall, 2006), Джонсоном (Johnson, 2007), Шнайдерманом и Плейзент (Shneiderman, Plaisant, 2009). Microsoft, Apple Computer и Oracle публикуют рекомендации по проектированию программного обеспечения для своих платформ (Apple Computer, 2009; Microsoft Corporation, 2009; Oracle Corporation/Sun Microsystems, 2001).

Насколько полезны эти рекомендации? Все зависит от того, кто их применяет и из-под чьего пера выходят проблемные интерфейсы.

Для разработки интерфейсов и организации взаимодействия с пользователем требуются понимание и опыт

Следовать рекомендациям по проектированию пользовательских интерфейсов — это не то же самое, что готовить по рецепту из кулинарной книги. Правила проек-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

тирования часто описывают цели, а не действия, необходимые для их достижения. Такие правила намеренно формулируются в самом общем виде, чтобы обеспечить возможность их широкого применения, поэтому интерпретация правила и возможность его использования в той или иной ситуации остаются на совести дизайнера.

В действительности все еще сложнее. Во время разработки интерфейса подходящими могут оказаться сразу несколько конфликтующих правил, то есть предполагающих разные варианты дизайна. Дизайнеру приходится выбирать, какое из правил будет наиболее удобным и выгодным в конкретной ситуации, и отбрасывать остальные.

Проблемы дизайна помимо конфликтующих правил включают и конфликтующие цели, например:

- яркий экран *и* долгий заряд аккумулятора;
- небольшой вес *и* прочность;
- многофункциональность *и* легкость в освоении;
- мощность *и* простота;
- WYSIWYG (what you see is what you get — что вижу, то и получаю) *и* возможность использования устройства незрячими людьми.

Для того чтобы программный продукт или компьютерная услуга удовлетворяли всем требованиям, их создателям приходится идти на определенные компромиссы. Если откровенно, то на множество компромиссов. Жертв требует и поиск золотой середины среди конкурирующих правил проектирования.

Учитывая все эти сложности, к правилам и рекомендациям по проектированию пользовательских интерфейсов следует относиться очень внимательно и применять их продуманно; кроме того, делать это должны люди, хорошо знакомые с искусством дизайна и/или оценки пользовательских интерфейсов. Халатность недопустима. Правила и рекомендации по проектированию пользовательских интерфейсов — это, скорее, *законы*, а не *пошаговые инструкции*. Так же, как свод законов намного лучше умеют применять и толковать юристы и судьи, хорошо подкованные в этой сфере, набор рекомендаций по дизайну пользовательского интерфейса должны принять на вооружение люди, понимающие источники этих рекомендаций и имеющие большой опыт их использования.

К сожалению, за редкими исключениями (вроде книги (Norman, 1983a)), рекомендации по разработке пользовательских интерфейсов даются в форме простых списков однозначных правил — без каких-либо обоснований или разъяснений.

Помимо того, большинство первопроходцев в области дизайна пользовательских интерфейсов и изучения удобства в использовании были знакомы с когнитивной психологией, однако о новичках этого сказать нельзя. Поэтому им сложнее подходить к применению рекомендаций по проектированию пользовательских интерфейсов с точки зрения здравого смысла и научного опыта.

В этой книге я попытался привести обоснования и указать заслуживающие доверия научные источники.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Сравнение рекомендаций по проектированию пользовательских интерфейсов

В табл. B1 приведены два наиболее известных списка рекомендаций по проектированию пользовательских интерфейсов. Вы видите, какого типа правила они содержат и насколько похожи друг на друга (ссылки на другие наборы рекомендаций вы найдете в конце книги). Оба списка начинаются с требования соблюдать единообразие дизайна. Оба включают правило о предотвращении ошибок. Правило Нильсена—Молича «Помогайте пользователям распознавать, диагностировать и устранять ошибки» очень напоминает правило Шнайдермана—Плейзента «Обеспечивайте возможность легкой отмены действия». «Пользовательский контроль и свобода» — это почти то же самое, что «Пусть пользователи чувствуют, что контроль в их руках». То, что эти списки так схожи, — не случайность и не означает, что авторы более поздних работ писали их под впечатлением от книг первопроходцев.

Таблица B1. Два наиболее известных списка рекомендаций по проектированию пользовательских интерфейсов

Шнайдерман (1987); Шнайдерман и Плейзент (2009)	Нильсен и Молич (1990)
■ Стремитесь к единообразию ■ Обеспечьте одинаковое удобство в использовании ■ Предусмотрите информативную ответную реакцию ■ Прорабатывайте замкнутые потоки решения задач ■ Предотвращайте ошибки ■ Обеспечивайте возможность легкой отмены действия ■ Пусть пользователи чувствуют, что контроль в *их* руках ■ Минимизируйте нагрузку на кратковременную память	■ Единообразие и стандарты ■ Очевидность состояния системы ■ Соответствие системы реальному миру ■ Пользовательский контроль и свобода ■ Предотвращение ошибок ■ Узнавание, а не вспоминание ■ Гибкость и эффективность использования ■ Эстетичный и минималистичный дизайн ■ Помогайте пользователям распознавать, диагностировать и устранять ошибки ■ Предоставляйте интерактивную документацию и справку

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Кто создает рекомендации по проектированию

Для наших целей конкретные правила проектирования, подобные тем, что представлены в табл. B1, не так уж важны. Важно то, что их объединяет: источники и основы. Откуда взялись эти правила? И не действовали ли их авторы по тому же принципу, что и дизайнеры модной одежды, пытаясь навязать личные вкусы всем представителям компьютерной промышленности и сферы производства программного обеспечения?

Если бы это было так, то наборы правил проектирования должны были сильно различаться, поскольку авторы пытались быть непохожими на всех остальных. Но в действительности все наборы рекомендаций по разработке пользовательских интерфейсов очень похожи, если не обращать внимания на формулировки, акцентирование тех или иных составляющих и состояние компьютерных технологий на момент их написания. Почему это так?

Ответ прост — все правила проектирования основаны на человеческой психологии: как люди воспринимают, изучают, рассуждают, запоминают и преобразуют намерения и действия. Многие авторы правил проектирования занимались психологией и применяли полученные знания при разработке компьютерных систем.

Например, Дон Норман стал профессором, исследователем и автором многочисленных материалов по когнитивной психологии задолго до того, как начал интересоваться взаимодействием человека и компьютера. Первые рекомендации Нормана по разработке интерфейсов основывались на исследованиях когнитивной деятельности человека, проведенных им самим и другими учеными. Его особенно интересовали когнитивные ошибки, совершаемые людьми, и проблемы проектирования компьютерных систем, умеющих смягчать или устранять последствия таких ошибок.

Аналогично и другие авторы правил проектирования пользовательских интерфейсов, такие как Браун, Шнайдерман, Нильсен и Молич, использовали знание перцепционной и когнитивной психологии для усовершенствования дизайна полезных и удобных интерактивных систем.

Резюме: рекомендации по разработке пользовательских интерфейсов основываются на человеческой психологии.

Прочитав эту книгу, вы познакомитесь с наиболее важными аспектами психологии, лежащими в основе правил проектирования пользовательских интерфейсов и совершенствования удобства в использовании.

Целевая аудитория книги

Эта книга адресована разработчикам программного обеспечения, использующим рекомендации по проектированию пользовательских интерфейсов и взаимодействий. К ним, разумеется, относятся дизайнеры взаимодействий, дизайнеры поль-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

зовательского интерфейса, дизайнеры восприятия, дизайнеры-графики, дизайнеры компьютерных устройств и изделий. Кроме того, она будет полезна тестировщикам юзабилити и аналитикам, использующим эвристические правила при проверке программного обеспечения или ошибок использования.

Кроме того, я рекомендую эту книгу менеджерам по разработке программного обеспечения, желающим получить качественное представление о психологических основах правил проектирования пользовательских интерфейсов, для того чтобы хорошо понимать и правильно оценивать работу своих сотрудников.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу электронной почты comp@piter.com (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

На веб-сайте издательства http://www.piter.com вы найдете подробную информацию о наших книгах.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 1

Наше восприятие зависит от ожиданий

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

То, как мы воспринимаем мир, далеко не в полной мере отражает существующую действительность. Наше восприятие сильно зависит от того, чего мы ожидаем. На ожидания и, следовательно, восприятие мира воздействуют три фактора:

- **прошлое** — наш опыт;
- **настоящее** — текущий контекст;
- **будущее** — наши цели.

Опыт искажает восприятие

Представьте, что вы работаете в большой страховой компании. На встрече с агентом по недвижимости вы обсуждаете план нового кампуса, включающего несколько офисных зданий. Зданий пять, и они выстроены в ряд, одно за другим. У последних двух зданий предусмотрены внутренние дворики в форме буквы Т, обеспечивающие естественное освещение кафетерия и фитнес-центра. Если бы агент продемонстрировал вам карту, показанную на рис. 1.1, то вы бы увидели пять черных фигур, представляющих здания.

Рис. 1.1. План кампуса или слово? Все зависит от того, что вы ожидаете увидеть

А теперь представьте, что вы встречаетесь не с агентом по недвижимости, а с рекламным агентом и обсуждаете новый рекламный постер, который планируется разместить на определенных рынках по всей стране. Агент демонстрирует вам то же изображение, но в этом сценарии оно представляет собой набросок объявления, включающего одно-единственное слово — LIFE. И вы четко и недвусмысленно видите на картинке именно эти четыре буквы.

Когда ваша система восприятия информации настроена на архитектурный план, вы видите очертания зданий и не обращаете почти никакого внимания на белые области между черными фигурами. Когда же система восприятия настраивается на текст, вы видите белый текст, не замечая черных областей между буквами.

Довольно известный пример того, как «настройка» разума влияет на восприятие, представлен на рис. 1.2. На первый взгляд этот набросок, предположительно

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

сделанный Р. К. Джеймсом (R. C. James)[1], большинству людей кажется случайным набором чернильных пятен. Прежде чем продолжать чтение, внимательно рассмотрите изображение.

Рис. 1.2. Изображение, демонстрирующее эффект «настройки» визуальной системы на восприятие определенной информации. Что вы видите?

Только после того, как вам скажут, что здесь изображен далматин, обнюхивающий землю вокруг дерева, ваша визуальная система сможет преобразовать набор пятен в цельную картинку. Более того, единожды «увидев» собаку, вы будет очень трудно возвращаться к восприятию изображения как случайного набора чернильных клякс.

Ранее приведены примеры из визуальной сферы. Но опыт может влиять и на другие типы восприятия, такие как понимание текста. Например, заголовок «Новая вакцина содержит вирус бешенства» будет по-разному воспринят людьми, у которых в памяти свежи истории об успешном применении вакцин для борьбы с заболеваниями, и людьми, которые недавно слышали о бракованных зараженных вакцинах.

Пользователи компьютерного программного обеспечения и посетители веб-сайтов часто щелкают на кнопках или ссылках, почти не глядя на них. Их восприятие представленного на экране основывается по больше части на прошлом опыте, поэтому они не тратят силы на тщательное изучение картинки. Это нередко нарушает планы

[1] Marr D. (1982) *Vision*. W. H. Freeman, New York, NY, p. 101, Figure 3-1.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

дизайнеров программного обеспечения, которые ожидают, что пользователи будут видеть именно то, что было помещено на экран разработчиками. Но что поделать — так работают человеческие системы восприятия информации.

Например, если поменять местами кнопки Далее и Назад на последней странице многостраничного диалогового окна[1], то многие люди далеко не сразу обратят на это внимание (рис. 1.3). Их визуальную систему «убаюкает» единообразие размещения кнопок, которое наблюдалась на предыдущих нескольких страницах. Даже ошибившись несколько раз кнопкой, люди все равно продолжают ожидать «правильного» расположения кнопок и снова возвращаться на шаг назад, вместо того чтобы переходить к следующему окну. Вот почему правило «единообразно размещайте элементы управления» так часто встречается в рекомендациях по проектированию пользовательских интерфейсов.

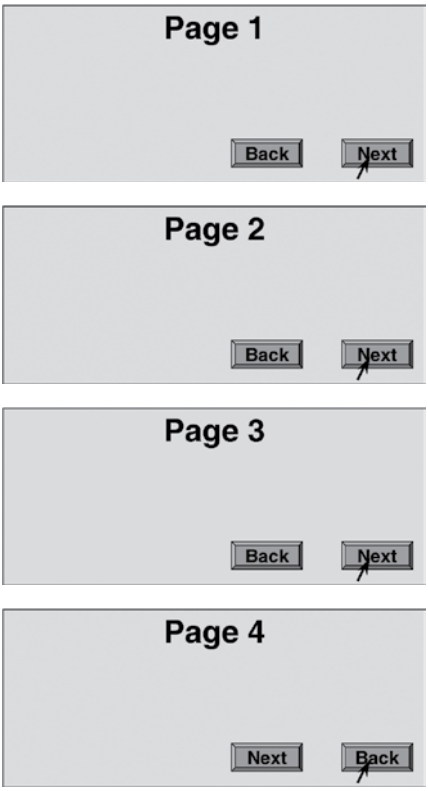


Рис. 1.3. Кажется, что кнопка Next (Далее) всегда находится на одном и том же месте, но в действительности это не так

[1] На жаргоне разработчиков пользовательского интерфейса многошаговое диалоговое окно называется *мастером*.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

То же самое происходит, когда мы подолгу ищем нужную вещь, не замечая ее, хотя она лежит прямо перед глазами — просто потому, что она находится на непривычном месте или выглядит не так, как всегда. Опыт заставляет нас искать привычные характеристики в привычных местах. Например, если кнопка Отправить на одной из форм веб-сайта отличается по форме или цвету от аналогичных кнопок на остальных формах, пользователь может не найти ее. О такой слепоте, вызываемой нашими ожиданиями, мы поговорим чуть далее в этой главе, в разделе, посвященном тому, как на восприятие влияют наши *цели*.

Текущий контекст искажает восприятие

При попытке разобраться, как же происходит визуальное восприятие, очень часто возникает соблазн свести все к простому восходящему процессу: объединение базовых признаков, таких как грани, линии, углы, кривые и узоры, в фигуры и, в конце концов, в осмысленные объекты. Если взять в качестве примера чтение, то можно предположить, что наша визуальная система сначала распознает знакомые фигуры букв, затем объединяет буквы в слова, слова — в предложения и т. д.

Однако визуальное восприятие — и, в частности, чтение — это не банальный восходящий процесс. Он включает также нисходящую обработку информации. Так, от слова, в составе которого мы видим определенный символ, зависит, каким образом мы данный символ идентифицируем (рис. 1.4).

Рис. 1.4. Один и тот же символ читается как буква «H» или «A» в зависимости от того, какие буквы с ним соседствуют

Сходным образом мы можем по-разному воспринимать одни и те же слова в зависимости от того, в составе каких предложений или абзацев видим их. Как иллюстрирует рис. 1.5, одна и та же последовательность букв принимает совершенно разные значения в различных фрагментах текста.

Fold napkins. *Polish silverware*. Wash dishes.

French napkins. *Polish silverware*. German dishes.

Рис. 1.5. Одно и то же предложение воспринимается по-разному в зависимости от того, в каком списке оно фигурирует

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Искажение смысла из-за контекста работает не только в пределах одного канала восприятия. Сигналы, поступающие от любого из наших пяти чувств, могут одновременно влиять на считывание информации остальными четырьмя, например:

- то, что мы видим, может искажаться тем, что мы слышим, и наоборот;
- то, что мы осязаем, может искажаться тем, что мы слышим, видим или обоняем.

В следующих главах я расскажу, как в человеческом мозгу происходят визуальное восприятие и чтение текста и как работает функция распознавания объектов. Пока же я просто замечу, что шаблон нейронной активности, соответствующий распознаванию буквы, слова, лица или любого другого объекта, включает также входные данные нейронной активности, индуцированной контекстом. В контекст входят другие расположенные поблизости воспринимаемые объекты и происходящие события и даже повторно задействованные воспоминания о ранее считанных объектах и событиях.

Контекст искажает восприятие не только у людей, но также у низших животных. Одной моей знакомой часто приходилось ездить по различным делам на автомобиле, и она нередко брала в эти поездки своего пса. Однажды, когда она подъехала к дому, на лужайке сидел кот. Пес увидел кота через окно автомобиля и начал лаять. Моя подруга открыла дверь, пес выскочил из машины и погнался за котом, который тут же шмыгнул под куст и сбежал. Пес сунулся в гущу веток, но не сумел схватить кота. Еще довольно долго после происшествия он не мог успокоиться.

С того дня и до самого переезда каждый раз, когда моя подруга подъезжала к дому на автомобиле со своим псом, он начинал нервничать и лаять, выпрыгивал из автомобиля, как только открывалась дверь, и мчался через всю лужайку к кусту. Разумеется, кота он ни разу не обнаружил, но это не имело значения. Одного того факта, что они подъезжали к дому на машине, было достаточно, чтобы пес увидел — а может быть, и почуял кота. В то же время, когда они возвращались домой пешком с ежедневной прогулки, «миража с котом» у пса не возникало.

Цели искажают восприятие

Помимо *прошлого* опыта и *настоящего* контекста на наше восприятие влияют также цели и планы из *будущего*. В частности, восприятие проходит через фильтр целей: вещи, не имеющие к ним отношения, обычно отбрасываются на подсознательном уровне, не доходя до «здравого сознания».

Например, просматривая меню приложения или содержимое веб-сайта в поисках определенной информации или функции, люди обычно не вчитываются в текст на экране. Они быстро и поверхностно сканируют его, останавливаясь только на объектах, связанных с искомой целью. Они не *игнорируют* ненужные вещи, они просто не *замечают* их.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Чтобы проверить это в действии, взгляните на рис. 1.6 и найдите в изображенном там ящике с инструментами ножницы. Затем сразу же возвращайтесь к тексту. Сделайте это прямо сейчас.

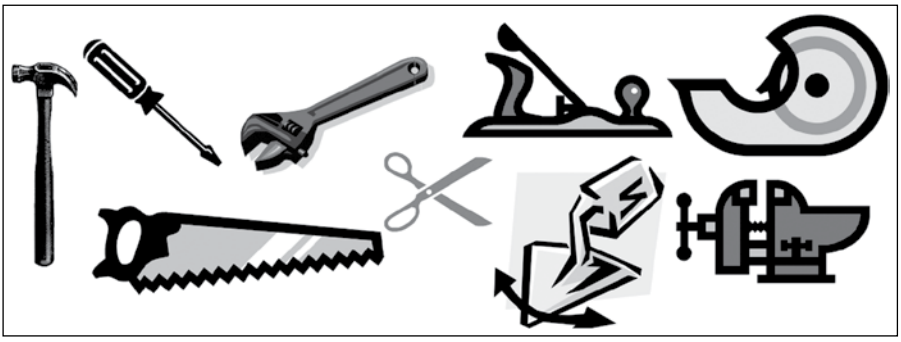


Рис. 1.6. Здесь есть ножницы?

Вам удалось обнаружить ножницы? А сможете ли вы, не глядя снова на рисунок, сказать, где в ящике с инструментами лежит отвертка?

Наши цели фильтруют не только зрительную информацию, но и информацию, получаемую с помощью других органов чувств. Распространенный пример — эффект коктейльной вечеринки. Общаясь с другом на многолюдной вечеринке, вы фокусируете внимание и слышите в основном его, несмотря на то что множество людей ведут вокруг шумные разговоры. Чем более вовлечены вы в свою беседу, тем сильнее ваш мозг отфильтровывает окружающий шум. Если же беседа утомляет вас, то вы, вероятно, будете слышать все большую и большую часть того, что произносят другие люди в помещении.

Этот эффект впервые был зафиксирован в исследованиях с участием авиадиспетчеров, которые способны поддерживать разговор с пилотами определенного воздушного судна, несмотря на то что одновременно на одной и той же радиочастоте ведется множество различных диалогов и все они транслируются из одного динамика на диспетчерском пульте (Arons, 1992). Исследования подтверждают, что наша способность фокусироваться на одном разговоре среди нескольких ведущихся одновременно зависит не только от уровня заинтересованности в конкретной беседе, но также от объективных факторов, таких как сходство голосов в окружающей какофонии, количество общего шума (такого, как звон столовых приборов или громкая музыка) и предсказуемость того, что говорит наш собеседник (Arons, 1992).

Фильтрация восприятия в зависимости от целей более свойственна взрослым, так как они лучше фокусируются на целях, чем дети. Ребенок с большей готовностью реагирует на сторонние раздражители, следовательно, его восприятие не так сильно фильтруется целями. Это означает, что дети отвлекаются чаще, чем взрослые, однако данная характеристика делает их более объективными наблюдателями.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Существует салонная игра, хорошо демонстрирующая влияние возраста на фильтрацию восприятия. Она очень похожа на наше предыдущее упражнение «найди инструмент в ящике». В большинстве квартир на кухне или в кладовой можно найти ящик, куда сложены различные кухонные приспособления или инструменты. Отправьте кого-нибудь из гостиной в помещение, где находится этот ящик, и попросите принести определенный предмет, например мерную ложку или разводной ключ. Когда подопытный вернется с искомым, спросите его о другом приспособлении — заметил ли он его в том же ящике? Большинство взрослых не смогут сказать, что еще лежало в ящике. Дети же в большинстве случаев оказываются в состоянии перечислить другие предметы из ящика — разумеется, если им удается оторваться от увлекательного разглядывания загадочных приспособлений и выполнить задание до конца.

Перцепционную фильтрацию можно наблюдать также у пользователей, перелистывающих страницы веб-сайтов. Предположим, я отправил вас на домашнюю страницу Кентерберийского университета Новой Зеландии (рис. 1.7) и попросил распечатать карту кампуса с обозначенным зданием факультета информатики. Вы наверняка быстро проглядели страницу и тут же щелкнули на одной из ссылок, содержащих описывающие поставленную цель слова: Departments (Факультеты) в верхнем левом углу, Departments and Colleges (Факультеты и колледжи) посередине у левого края или Campus Maps (Планы кампусов) в нижнем правом углу. Если вы «поисковик», то, возможно, сразу ввели нужное сочетание слов в поле поиска у правого края страницы, примерно посередине, и щелкнули на ссылке Go (Вперед).

Независимо от того, как именно вы поступите — просмотрите содержимое страницы или сразу перейдете к полю поиска, — в обоих случаях вы наверняка покинете главную страницу, не заметив, что выиграли в лотерею 100 долларов (объявление в нижнем левом углу). Почему? Потому что это сообщение никак не связано с вашей целью.

Каков же механизм воздействия текущих целей на наше восприятие? Есть два аспекта:

- **Влияние на область поиска.** Восприятие — это активный, а не пассивный процесс. Мы непрерывно двигаем глазами, руками, даже ушами и т. п., словно беря пробу вещей, составляющих наше окружение, чтобы обнаружить наиболее подходящие для текущей деятельности или для того, что мы собираемся делать (Ware, 2008). Когда мы просматриваем веб-сайт в поисках карты кампуса, глаза и лежащая на мыши рука притягиваются к любым объектам, обещающим привести к нужной цели. Все, что к этой цели не относится, мы в большей или меньшей степени игнорируем.
- **Повышение чувствительности системы восприятия к определенным характеристикам.** Когда перед нами стоит задача найти что-либо, мозг настраивает восприятие так, чтобы оно особенно чутко реагировало на характеристики, свойственные искомому объекту (Ware, 2008). Например, мы ищем на большой парковке красный автомобиль. В такой ситуации все красные машины сразу же

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

бросаются нам в глаза, но сознание едва ли регистрирует наличие автомобилей других цветов, несмотря на то что в определенном смысле мы их «видим». Точно так же, если человек пытается отыскать супругу или супруга в темной битком набитой людьми комнате, его мозг «программирует» слуховую систему, настраивая ее на комбинацию частот, составляющих голос партнера.

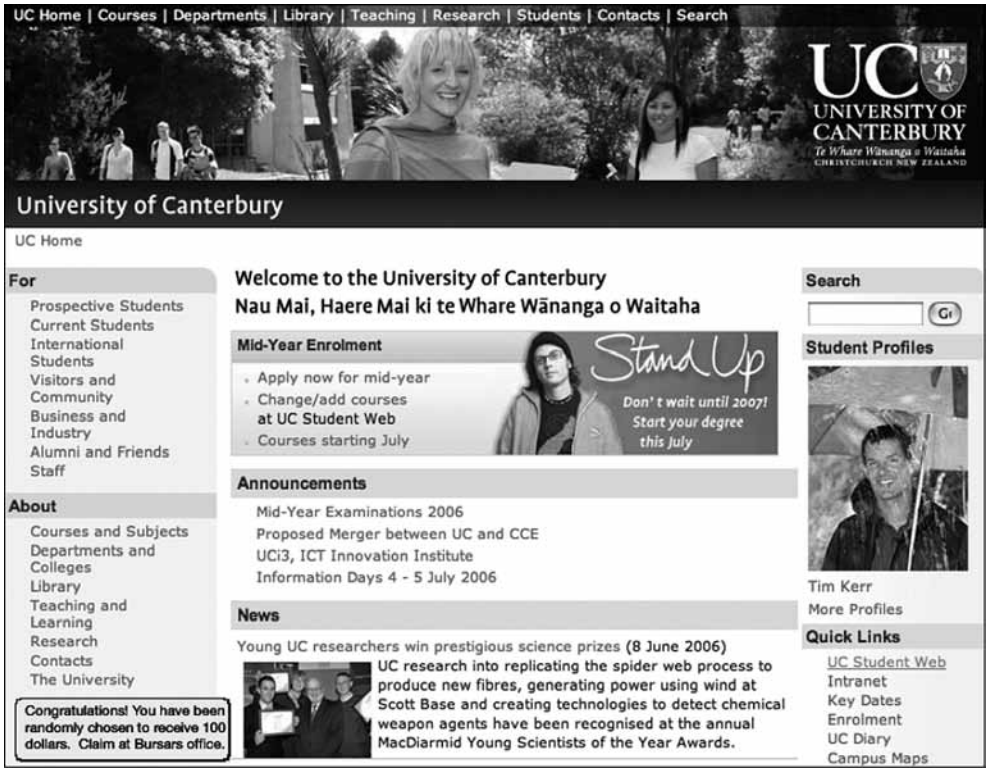


Рис. 1.7. Главная страница веб-сайта Кентерберийского университета: навигация по веб-сайтам включает фильтрацию восприятия

Выводы, влияющие на дизайн

Разумеется, все эти искажения восприятия и причины, их вызывающие, имеют большое значение для проектирования пользовательских интерфейсов. Вот три наиболее важных вывода, которые вы должны сделать по прочтении этой главы.

Избегайте двусмысленности

Избегайте двусмысленных информационных представлений и тестируйте дизайн, чтобы убедиться, что все пользователи одинаково трактуют то или иное пред-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ставление. Если двусмысленности не избежать, то для устранения нежелательных последствий либо полагайтесь на стандарты и условности, либо настраивайте систему восприятия пользователей так, чтобы они интерпретировали представление нужным вам образом.

Например, в информационных представлениях часто используются затененные кнопки и текстовые поля, выглядящие так, словно они немного возвышаются над фоновой поверхностью (рис. 1.8). Такое оформление основывается на условности, знакомой многим опытным компьютерным пользователям, — считается, что источник света находится в левом верхнем углу экрана. Если изобразить объект так, словно свет на него падает из другой точки, у пользователей не создастся впечатления его «приподнятости» над поверхностью экрана.

Рис. 1.8. Для оформления кнопок на компьютерных экранах часто используются тени, придающие им объем, однако данная условность работает только в том случае, если источник света подразумевается в верхнем левом углу

Следите за единообразием

Единообразно размещайте информацию и элементы управления. Элементы управления и информационные представления, выполняющие одинаковые функции на разных страницах, везде должны оставаться на одном и том же месте. Они также должны быть одного цвета, использовать один шрифт, вариант штриховки и т. п. Единообразие помогает пользователям быстро обнаруживать и узнавать объекты.

Понимайте цели

Пользователи заходят в систему с определенными целями, и дизайнеры систем должны понимать, с какими. Не забывайте, что цели могут меняться и что от целей пользователя зависит то, как он воспринимает информацию. Удостоверьтесь, что в любой точке взаимодействия с пользователем нужная ему информация доступна, заметна и ведет к достижению одной из возможных целей. Сделайте так, чтобы пользователь увидел и использовал эту информацию.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 2

Наше зрение оптимизировано для восприятия структуры

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В начале XX века группа немецких психологов поставила себе цель — разобраться, как работает зрительное восприятие человека. Они провели наблюдения и описали множество важных визуальных феноменов. Одним из основных открытий, сделанных ими, было то, что человеческое зрение отличается целостностью: наша визуальная система автоматически связывает структуру с полученной визуальной информацией, так как она настроена на восприятие цельных форм, фигур и объектов, а не разрозненных граней, линий и областей. По-немецки «фигура» или «форма» — *Gestalt* (читается «гештальт»), поэтому разработанные учеными теории стали называться гештальт-принципами визуального восприятия.

Сегодня психологи, изучающие восприятие и функции познания, относятся к гештальт-теории восприятия скорее как к описательному каркасу, а не как к объясняющей и предсказательной теории. Современные теории визуального восприятия обычно основываются на нейрофизиологии глаз, оптического нерва и мозга (см. главы 4–7).

Неудивительно, что наблюдения ученых, изучающих нейропсихологию, подтверждают выводы гештальт-психологов. Действительно, мы, как и другие животные, «заточены» под восприятие окружающей среды в формах целостных объектов (Stafford & Webb, 2005; Ware, 2008). Следовательно, гештальт-принципы все так же остаются в силе, если не как фундаментальное обоснование особенностей зрительного восприятия, то, по крайней мере, как основа для описания этих процессов. Они также обеспечивают весьма полезную базу для рекомендаций по графическому дизайну и дизайну пользовательских интерфейсов (Soegaard, 2007).

Для нас наибольший интерес представляют гештальт-принципы близости, схожести, непрерывности, замкнутости, симметрии, объекта/фона и общего направления. В следующих разделах я подробнее расскажу о каждом из этих принципов и приведу примеры статического графического дизайна и дизайна пользовательских интерфейсов.

Гештальт-принцип: близость

Принцип *близости* заключается в том, что относительное расстояние между объектами влияет на восприятие организации этих объектов: разделены ли они по подгруппам, и если да, то каким образом. Нам кажется, что объекты, находящиеся рядом (относительно прочих объектов), сгруппированы, в то время как объекты, разнесенные на некоторое расстояние, не формируют группу.

На рис. 2.1 звездочки в левой части теснее прижаты друг к другу по горизонтали, чем по вертикали, поэтому мы видим три строки звездочек. В правой части рисунка ситуация обратная — расстояние по вертикали между звездочками меньше, чем по горизонтали — поэтому мы видим три столбца.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Рис. 2.1. Близость: объекты, находящиеся рядом, кажутся нам сгруппированными: *слева* — строки, *справа* — столбцы

Очевидно, что принцип близости имеет большое значение при формировании макетов панелей управления, форм ввода данных в приложениях, веб-сайтов, электронных устройств. Дизайнеры зачастую разделяют группы экранных элементов управления и данных, помещая их в рамку или добавляя между группами разделительные линии (рис. 2.2).

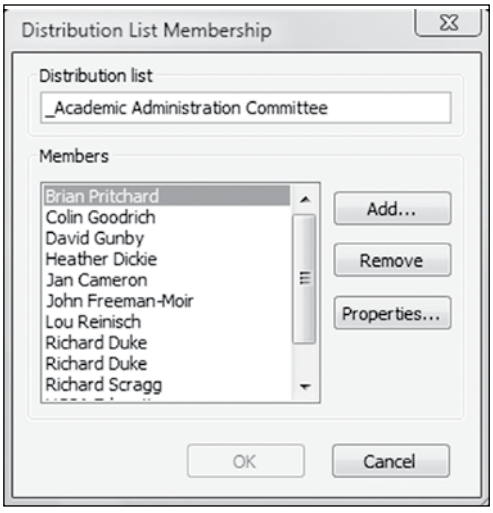


Рис. 2.2. В диалоговом окне Distribution List Membership (Группа рассылки) приложения Outlook кнопки, относящиеся к списку, находятся внутри рамки и, таким образом, отделены от кнопок управления диалоговым окном

Однако, согласно принципу близости, элементы представления можно сгруппировать, просто поместив их ближе друг к другу, чем к другим элементам управления, и не использовать при этом рамки или другие видимые границы (рис. 2.3). Многие

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

эксперты по графическому дизайну рекомендуют данный подход для уменьшения визуального беспорядка и объема кода, составляющего пользовательский интерфейс (Mullet & Sano, 1994).

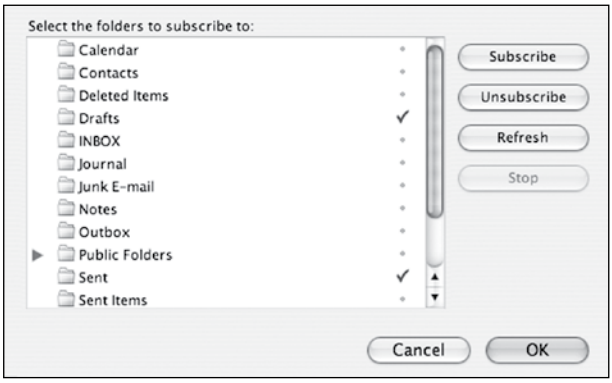


Рис. 2.3. В диалоговом окне Subscribe Folders (Папки подписки) приложения Mozilla Thunderbird элементы управления сгруппированы с использованием принципа близости

И наоборот, если местоположение элементов выбрано неудачно, то есть связанные по смыслу элементы управления находятся слишком далеко друг от друга, люди не будут воспринимать их как составляющие одной группы. Следовательно, им будет сложнее учиться пользоваться программным обеспечением и запоминать, что к чему относится. Например, в окне программы установки Discreet Software находятся шесть горизонтальных пар переключателей, каждая из которых предоставляет выбор «один из двух», но расположены они так, что мы видим два вертикальных набора переключателей, каждый из которых предоставляет выбор «один из шести». Это прекрасно согласуется с гештальт-принципом близости. Назначение переключателей неочевидно, и для того чтобы понять, как они работают, пользователю необходимо пощелкать на них (рис. 2.4).

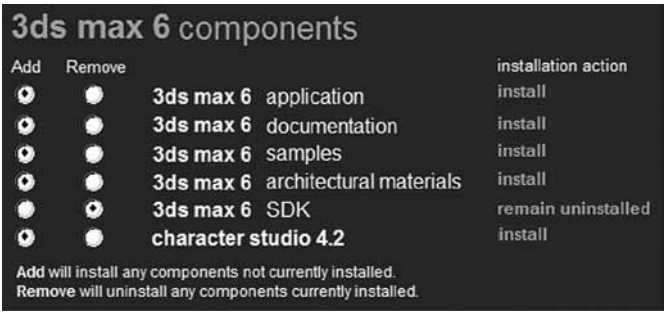


Рис. 2.4. В программе установки Discreet неудачное расположение переключателей создает впечатление наличия двух вертикальных столбцов вместо шести горизонтальных пар

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Гештальт-принцип: схожесть

Еще один фактор, влияющий на наше восприятие группировки, выражен в гештальт-принципе *схожести*: нам кажется, что похожие объекты составляют группу, а окружающие их объекты равнозначны. На рис. 2.5 чуть более крупные незакрашенные звезды воспринимаются как отдельная группа объектов.

Рис. 2.5. Схожесть: кажется, что элементы составляют группу, если они похожи между собой, но отличаются от окружающих объектов

В диалоговом окне Page Setup (Параметры страницы) операционной системы Mac OS принципы схожести и близости применяются для обозначения группировки элементов. Три очень похожих и расположенных вплотную друг к другу элемента управления Orientation (Ориентация), очевидно, составляют отдельную группу. Три раскрывающихся списка расположены не так плотно, но они достаточно похожи между собой, чтобы мы воспринимали их как связанные объекты, даже если разработчики не планировали этого.

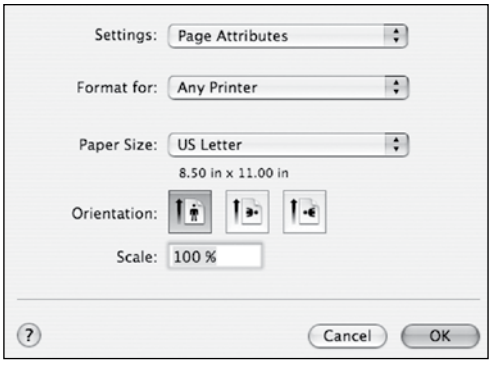


Рис. 2.6. Диалоговое окно Page Setup (Параметры страницы) операционной системы Mac OS: для группировки элементов управления Orientation (Ориентация) используются принципы схожести и близости

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Аналогично, текстовые поля на форме веб-сайта издательства Elsevier разбиты на группы: семь полей (с подгруппами) для адреса, группа из трех двойных полей для номеров телефона и два отдельных текстовых поля. Четыре меню помимо того, что служат полями для ввода данных, помогают разделять группы текстовых полей (рис. 2.7). В противоположность этому метки находятся слишком далеко от полей и не кажутся связанными с ними.

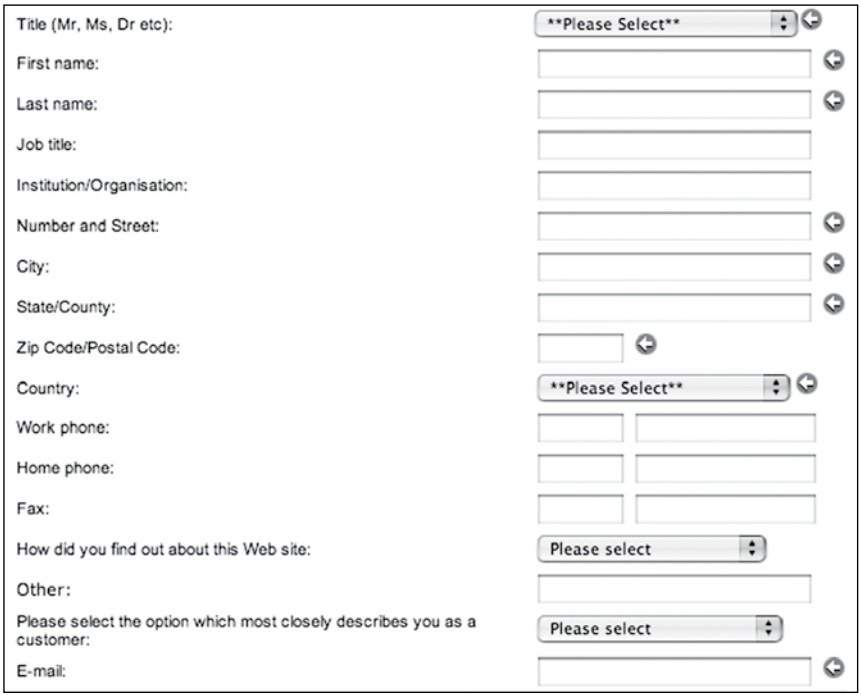


Рис. 2.7. Интерактивная форма на веб-сайте Elsevier.com: благодаря схожести текстовые поля выглядят сгруппированными

Гештальт-принцип: непрерывность

В дополнение к гештальт-принципам, касающимся стремления организовывать объекты в группы, несколько других гештальт-принципов описывают тенденцию человеческой визуальной системы устранять двусмысленность или восстанавливать отсутствующие данные так, чтобы в результате получился целостный объект. Первый из таких принципов, принцип *непрерывности*, утверждает, что наше визуальное восприятие склонно видеть непрерывные формы, а не разрозненные сегменты.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Например, в левой части рис. 2.8 мы автоматически видим две пересекающиеся линии. Мы видим не два более темных и два более светлых отрезка и не составленную из двух отрезков разных оттенков V наверху и такую же перевернутую V внизу. Справа же на рис. 2.8 мы видим морское чудище, частично скрытое водой, а не три его отдельных куска.

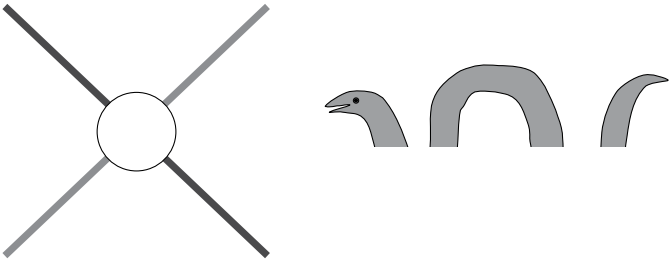


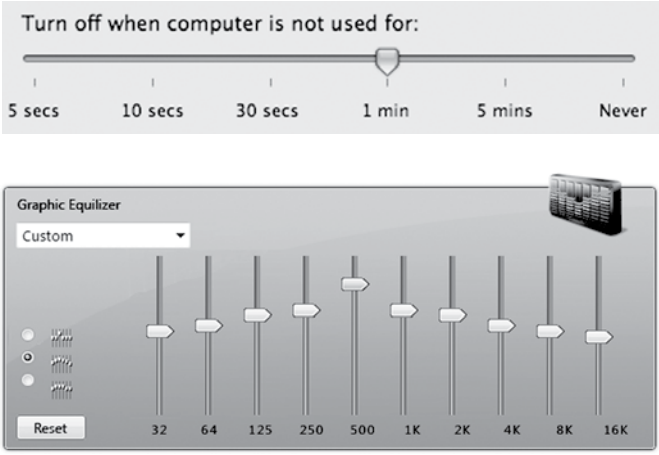
Рис. 2.8. Непрерывность: человеческое зрение настроено на то, чтобы видеть непрерывные формы, и наш мозг при необходимости добавляет недостающие данные

Хорошо известный пример использования принципа непрерывности в графическом дизайне — логотип корпорации IBM. Он состоит из несвязанных полосок, однако никакой двусмысленности в данном представлении нет. Мы с легкостью узнаем три буквы, написанные жирным шрифтом, — например, такими мы увидели бы их через венецианские жалюзи (рис. 2.9).

Рис. 2.9. В логотипе компании IBM благодаря принципу непрерывности из отдельных полосок формируются буквы

Элемент управления «ползунок» — это пример применения принципа непрерывности в пользовательском интерфейсе. Для нас ползунок — это единый диапазон, которым мы управляем с помощью бегунка, способного перемещаться по всему диапазону. Мы не воспринимаем этот элемент управления как два отдельных диапазона, разделенных бегунком (рис. 2.10, *а*). Даже окрашивание диапазона в разные цвета по разные стороны бегунка не «ломает» наше восприятие ползунка как единого объекта. Хотя, разумеется, выбор ComponentOne — сильно контрастирующие цвета (серый вверху и красный внизу) — в некоторой степени напрягает (рис. 2.10, *б*).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks



а

б

Рис. 2.10. Непрерывность: мы воспринимаем ползунок как одну бороздку с перемещающейся по ней рукояткой, а не как две бороздки с рукояткой в качестве разделителя: *а* — Mac OS; *б* — ComponentOne

Гештальт-принцип: замкнутость

С принципом непрерывности во многом схож гештальт-принцип *замкнутости*, который гласит, что наша визуальная система автоматически пытается замкнуть открытые фигуры, для того чтобы увидеть в них целостные объекты, а не отдельные фрагменты. Таким образом, на рис. 2.11 в несвязанных дугах мы видим окружность.

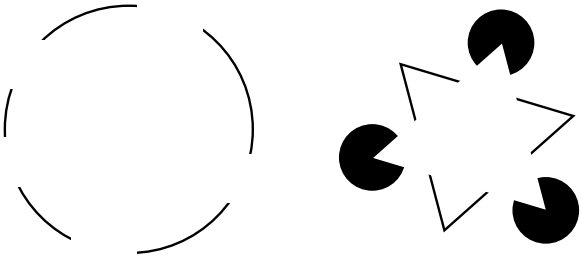


Рис. 2.11. Замкнутость: человеческое зрение настроено на восприятие законченных объектов, даже если в действительности они не завершены

Наша визуальная система так сильно настроена на то, чтобы видеть объекты, что даже абсолютно пустую область она может интерпретировать как объект. Комбинацию фигур справа на рис. 2.11 мы воспринимаем как белый треугольник, наложенный на другой треугольник и три черных круга, несмотря на то что

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

в действительности здесь присутствуют только три фигуры в форме буквы V и три фигуры, напоминающие Пакмана из знаменитой компьютерной игры.

Принцип замкнутости часто применяется в графических пользовательских интерфейсах. Например, наборы объектов, таких как документы или сообщения, в интерфейсах часто изображаются в виде пачек (рис. 2.12). Демонстрации одного целого объекта и краев остальных «позади» него достаточно для того, чтобы пользователи увидели целую пачку завершенных целостных объектов.

Рис. 2.12. Значки с изображением пачек объектов иллюстрируют принцип замкнутости: частично видимые объекты воспринимаются как целые

Гештальт-принцип: симметрия

Третий факт, описывающий наше стремление видеть целостные объекты, сформулирован в гештальт-принципе *симметрии*. Он утверждает, что мы разбираем сложные сцены на фрагменты так, чтобы минимизировать сложность. Чаще всего данные в визуальном поле можно интерпретировать несколькими способами, но наше зрение автоматически организует и интерпретирует их с позиции простоты и симметрии.

Например, сложную фигуру слева на рис. 2.13 мы воспринимаем как два пересекающихся ромба, а не как два соприкасающихся уголка или сдавленный посередине восьмиугольник с квадратом в центре. Пара пересекающихся ромбов проще, чем две другие интерпретации, показанные справа на рис. 2.13: в варианте с ромбами меньше сторон и больше симметрии, чем в сложных интерпретациях.

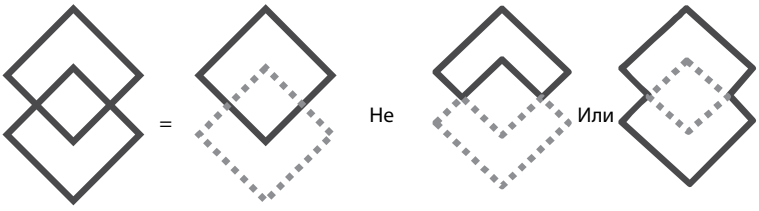


Рис. 2.13. Симметрия: человеческая визуальная система старается разбить сложные сцены на комбинации простых симметричных фигур

В печатных графических материалах и на компьютерных экранах — везде для представления трехмерных объектов на двумерном носителе можно пользоваться зависимостью визуальной системы от принципа симметрии. Иллюстрируют этот подход обложка книги Пола Тагарда «Согласованность мыслей и действий» (Thagard, 2002) (рис. 2.14) и трехмерное изображение городского ландшафта (рис. 2.15).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Рис. 2.14. На обложке книги «Согласованность мыслей и действий» благодаря принципам симметрии, замкнутости и непрерывности мы видим куб

Рис. 2.15. Симметрия: человеческая визуальная система раскладывает очень сложные двумерные изображения в трехмерные сцены

Гештальт-принцип: «фигура/фон»

Следующий гештальт-принцип, описывающий структурирование нашей визуальной системой получаемых данных, называется «*фигура/фон*». Он гласит, что разум разделяет визуальное поле на фигуру (передний план) и фон (задний план).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Передний план включает элементы сцены, привлекающие основное внимание, а на задний план относится все остальное.

Также согласно принципу «фигура/фон» разделение визуальной системой сцены на объект и фон в большой степени зависит от характеристик сцены. Например, когда небольшой элемент или цветное пятно находится поверх более крупного, мы воспринимаем этот элемент как фигуру, а более крупную составляющую сцены — как фон (рис. 2.16).

Рис. 2.16. Когда элементы пересекаются, мы видим в этой сцене небольшой объект на более обширном фоне

Однако наше восприятие фигуры и фона определяется не только характеристиками сцены. Оно зависит также от фокуса внимания зрителя. Нидерландский художник М. К. Эшер (M. C. Escher) использовал этот феномен для создания двусмысленных изображений, на которых объект и фон при смене фокуса внимания меняются местами (рис. 2.17).

Рис. 2.17. М. К. Эшер в своих работах использовал двусмысленность толкования фигуры и фона

В дизайне пользовательских интерфейсов и веб-дизайне принцип «объект/фон» применяется довольно часто, например «позади» основного содержимого поме-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

щают фон, создающий определенную атмосферу или впечатление (рис. 2.18). Фон может содержать информацию, например указывать текущее местоположение пользователя, подразумевать тему, сообщать торговую марку, воздействовать на настроение зрителя и, следовательно, интерпретацию содержимого.

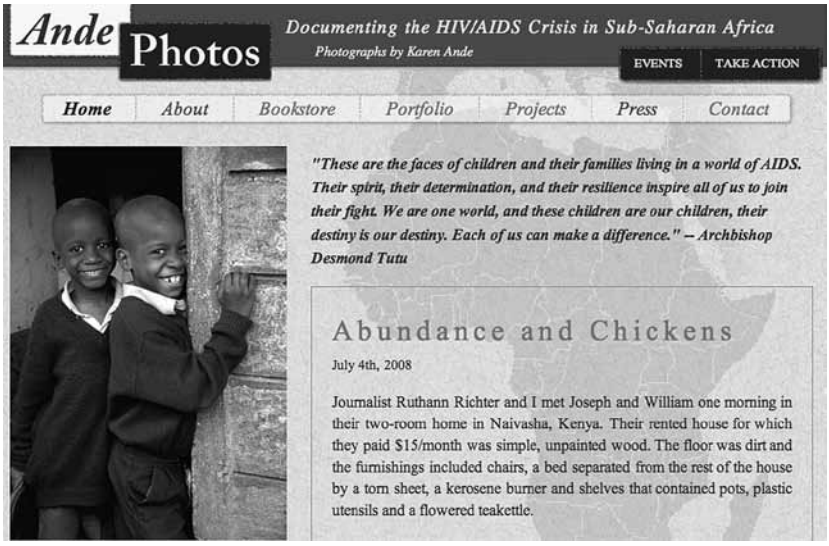


Рис. 2.18. Принцип «объект/фон» используется на веб-сайте AndePhotos.com — «позади» содержимого отображается тематический водяной знак

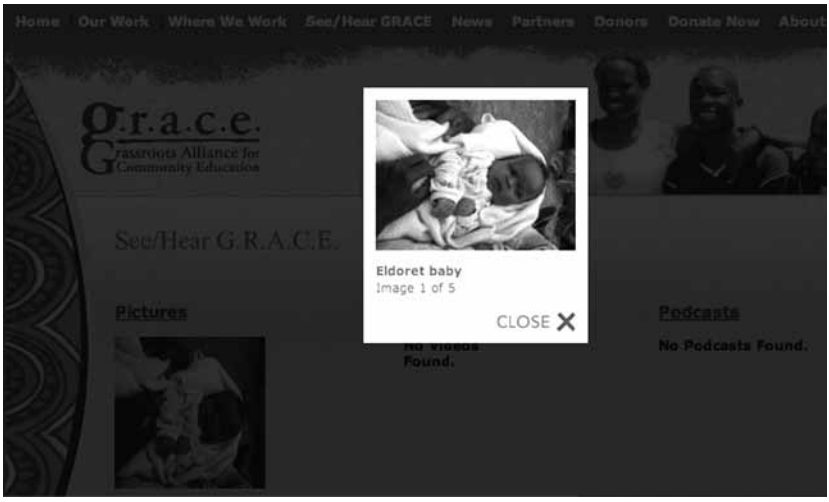


Рис. 2.19. Принцип «фигура/фон» используется на веб-сайте GRACEUSA.org — фотография всплывает поверх содержимого страницы

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Кроме того, принцип «фигура/фон» часто применяется для вывода важной информации поверх остального содержимого. Таким образом, содержимое, которое ранее играло роль объекта — фокуса внимания пользователей, — временно становится *фоном* новой информации, которая на короткое время отображается в качестве новой *фигуры* (рис. 2.19). Данный подход обычно дает лучшие результаты, чем временная замена старой информации новой, так как обеспечивает контекст, помогающий людям не потеряться, то есть понимать, в какой точке взаимодействия они находятся в каждый конкретный момент.

Гештальт-принцип: общая судьба

Предыдущие шесть гештальт-принципов описывали восприятие статических (неподвижных) фигур и объектов. Последний рассматриваемый принцип — гештальт-принцип *общей судьбы* — описывает движущиеся объекты. У него много общего с принципами близости и сходства: он также определяет, воспринимаем ли мы объекты как группу. Принцип общей судьбы утверждает, что объекты, движущиеся вместе, воспринимаются как сгруппированные или связанные.

Например, в представлении, включающем несколько десятков пятиугольников, только семь фигур синхронно колеблются взад-вперед. В таком случае люди воспринимают их как связанную группу, даже если колеблющиеся пятиугольники находятся на достаточном расстоянии друг от друга и внешне не отличаются от неподвижных (рис. 2.20).

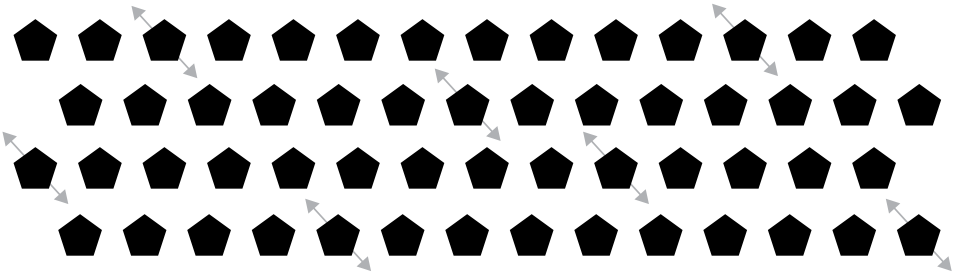


Рис. 2.20. Общая судьба: элементы кажутся сгруппированными или связанными, если они одинаково движутся

Общее движение и общее направление используются в анимированных представлениях для демонстрации взаимоотношения между сущностями. Например, на графиках GapMinder страны представлены анимированными точки, которые изменяются во времени под влиянием различных факторов экономического развития. Если точки двигаются одновременно, это означает, что история развития этих стран в соответствующие периоды была сходной (рис. 2.21).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

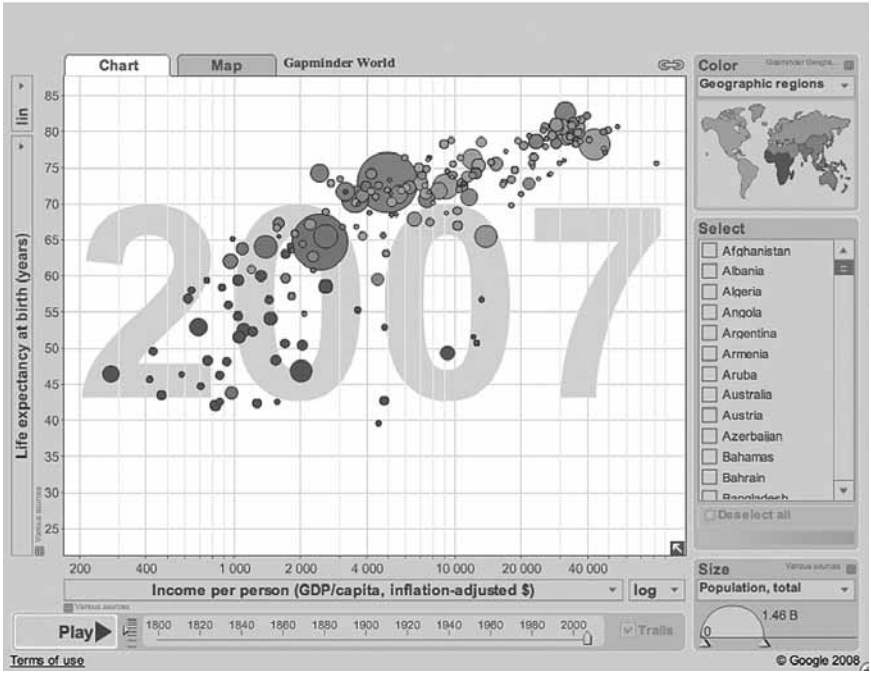


Рис. 2.21. Общая судьба: на графиках GapMinder точки движутся, демонстрируя страны со схожими историями развития

Гештальт-принципы: всё вместе

Конечно же, в любой визуальной сцене из реального мира гештальт-принципы работают совместно, а не по отдельности. Например, типичный рабочий стол в операционной системе Mac OS обычно воплощает шесть из семи описанных принципов (исключением является «общая судьба»): близость, схожесть, непрерывность, замкнутость, симметрия и «фигура/фон» (рис. 2.22). На типичном рабочем столе общая судьба (вместе со схожестью) используется, когда пользователь выделяет несколько файлов или папок и перетаскивает группу на новое место (рис. 2.23).

Когда все гештальт-принципы работают одновременно, между элементами дизайна могут появляться незапланированные визуальные взаимоотношения. После того как дизайн представления завершен, рекомендуется проверить его с точки зрения каждого из гештальт-принципов: близость, схожесть, непрерывность, замкнутость, симметрия, «фигура/фон», общая судьба, — для того чтобы убедиться, что среди элементов не появилось взаимоотношений, которые вы добавлять *не* намеревались.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

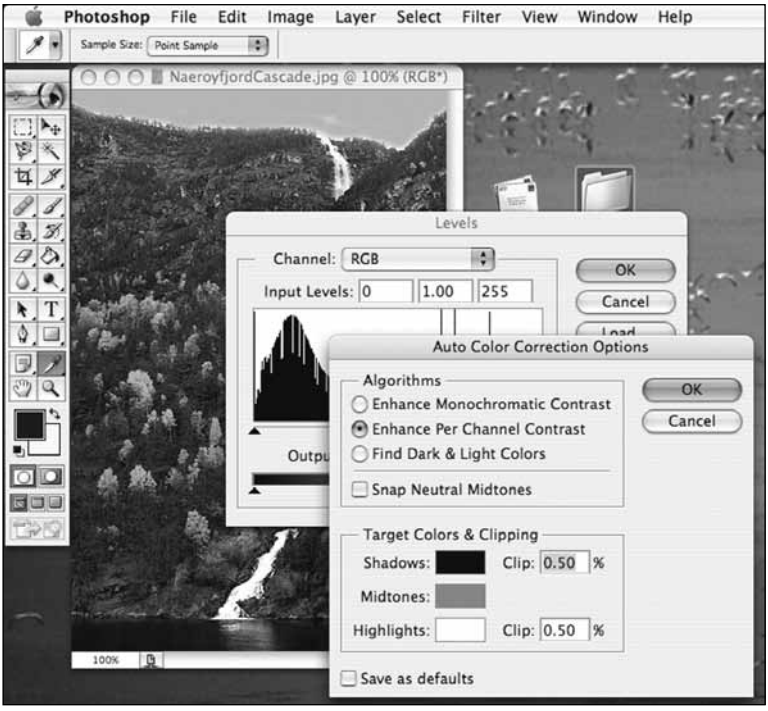


Рис. 2.22. Все гештальт-принципы, за исключением общей судьбы, работают на этом фрагменте рабочего стола Mac OS

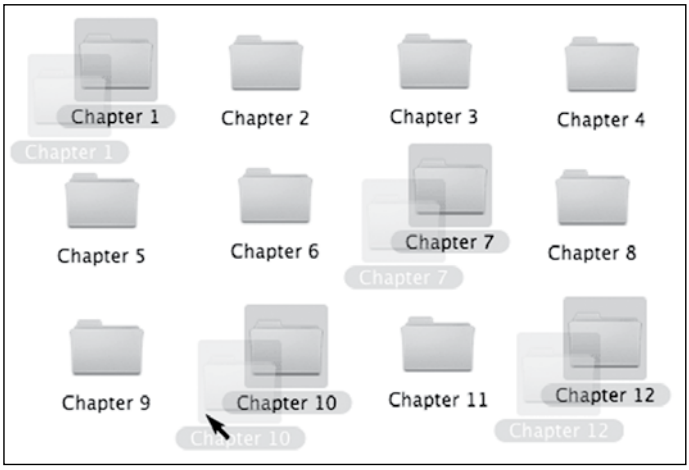


Рис. 2.23. Схожесть и общая судьба: когда пользователь перетаскивает выделенные папки, они кажутся единой группой за счет общей подсветки и движения

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 3

Ищем и используем визуальную структуру

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В главе 2 я на примере гештальт-принципов визуального восприятия показал, как сильно наша визуальная система оптимизирована для обнаружения структуры. Восприятие структуры в визуальном окружении помогает нам быстро распознавать объекты и события. Я также упомянул, что когда пользователи просматривают экран приложения или страницу веб-сайта, они не прилагают усилий для тщательного изучения и не читают каждое слово. Они быстро просматривают представление в поисках нужной информации. В этой главе вы найдете примеры, подтверждающие, что людям гораздо проще просматривать и понимать информацию, представленную лаконичным структурированным образом.

Рассмотрим два представления одной и той же информации о бронировании авиабилетов. Первое представление — это неструктурированный сплошной текст, второе — иерархическая структура (рис. 3.1). Структурированное представление данных бронирования можно просмотреть и понять намного быстрее, чем представление в форме сплошного текста.

Неструктурированное:

Вы забронировали место на рейсе 237 авиакомпании United, который отправляется из Окленда в 14:30 во вторник 15 октября и прибывает в Сан-Франциско в 11:40 во вторник 15 октября.

Структурированное:

Рейс: United 237, Окленд → Сан-Франциско

Отправление: 14:30 Вт 15 окт.
Прибытие: 11:40 Вт 15 окт.

Рис. 3.1. Структурированное представление данных бронирования куда проще понять

Чем более структурировано и сжато представлена информация, тем быстрее и эффективнее человек изучает и осмысливает ее. Взгляните на страницу Contents (Содержание) веб-сайта Департамента автотранспорта штата Калифорния (рис. 3.2). Многословные повторяющиеся ссылки затрудняют восприятие и «хоронят» важные слова, на которые ориентируется пользователь.

Сравните этот вариант с более сжатым, структурированным гипотетическим дизайном той же страницы. Я исключил ненужные повторения и выделил ссылками только те слова, которые представляют варианты действия (рис. 3.3). Все варианты, имеющиеся на исходной странице, доступны и на отредактированной, однако она занимает меньше экранного пространства и ее гораздо проще прочитать.

Отображение результатов поиска — это еще одна ситуация, в которой структурирование данных и исключение повторяющегося «шума» способно помочь людям быстрее просканировать список и найти искомое. В 2006 году результаты поиска по веб-сайту HP.com включали так много повторяющихся навигационных данных и метаданных для каждого элемента списка, что все представление полностью те-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ряло смысл. К 2009 году дизайнеры HP избавились от повторений и структурировали результаты, сделав их куда более читаемыми и полезными (рис. 3.4).

Renewals, Duplicates, and Information Changes for Driver Licenses and/or ID Cards

- How to renew your driver license in person
- How to renew your driver license by mail
- How to renew your driver license by Internet
- How to renew your instruction permit
- How to apply for a duplicate driver license or identification (ID) card
- How to change your name on your driver license and/or identification (ID) card
- How to notify DMV of my change of address
- How to register for the organ donor gift of life program

Рис. 3.2. На странице содержания веб-сайта Калифорнийского Департамента автотранспорта важная информация погребена под повторяющимися малозначительными словами

Licenses & ID Cards:	**Renewals, Duplicates, Changes**		
• Renew license:	in person	by mail	by Internet
• Renew:	instruction permit		
• Apply for duplicate:	license	ID card	
• Change of:	name	address	
• Register as:	organ donor		

Рис. 3.3. Страница содержания веб-сайта Калифорнийского Департамента автотранспорта с улучшенной визуальной структурой после устранения повторяющихся слов

Разумеется, для того чтобы информационное представление было легко и удобно просматривать, недостаточно сделать его лаконичным, структурированным и не содержащим излишних повторений. Оно также должно соответствовать правилам графического дизайна, о части которых я рассказал в главе 2. Например, в предварительной версии калькулятора ипотечного кредита на веб-сайте агентства недвижимости данные выводились в таблице, нарушающей по меньшей мере два важных правила графического дизайна (рис. 3.5, *слева*). Во-первых, люди обычно читают (как на бумаге, так и на экране) сверху вниз, однако метки рассчитанных сумм находились *под* числами. Во-вторых, метки располагались на одинаковом расстоянии как от «своих» значений, так и от следующих, находящихся ниже, поэтому невозможно было понять, исходя из принципа близости, каким образом метки группировались со значениями (см. главу 2). Для того чтобы вникнуть в таблицу с результатами ипотечных расчетов, пользователям приходилось тщательно и очень медленно изучать ее, пытаясь понять, к какому числовому значению относится та или иная метка.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

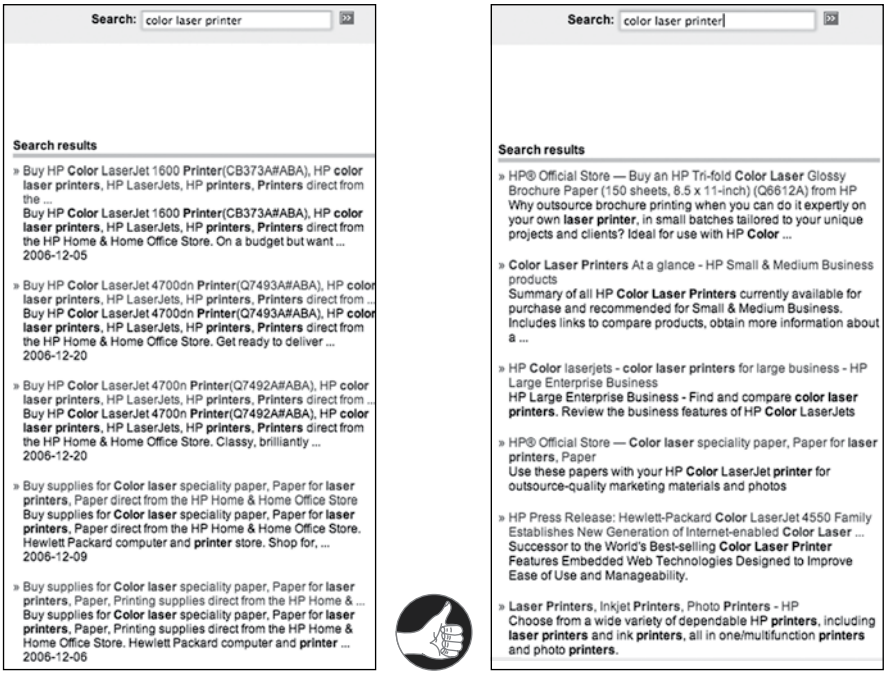


Рис. 3.4. В 2006 году функция поиска по веб-сайту HP.com возвращала дублирующиеся, «шумные» результаты (*слева*), но к 2009 году ситуация исправлена (*справа*)

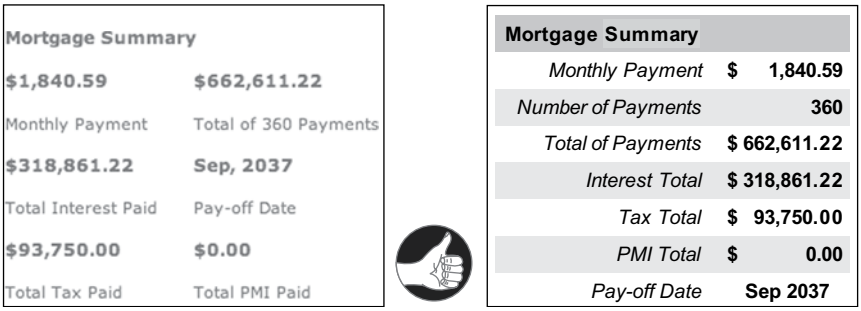


Рис. 3.5. Сводка по ипотечному кредиту: *слева* — в программном ипотечном калькуляторе; *справа* — улучшенный дизайн

В противоположность этому исправленный дизайн позволяет зрителю увидеть соответствие между метками и значениями, не задумываясь специально об этом (рис. 3.5, *справа*).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Структурирование помогает читать длинные номера и числа

Чтение информации даже небольшого объема можно упростить, структурировав ее. Два примера подобной информации: телефонные номера и номера кредитных карт (рис. 3.6 и 3.7). Такие номера традиционно разбивают на части, для того чтобы их было проще воспринимать и запоминать.

Легко:	(415) 123-4567
Сложно:	4151234567
Легко:	1234 5678 9012 3456
Сложно:	1234567890123456

Рис. 3.6. Телефонные номера и номера кредитных карт проще читать и воспринимать, когда они разбиты на сегменты

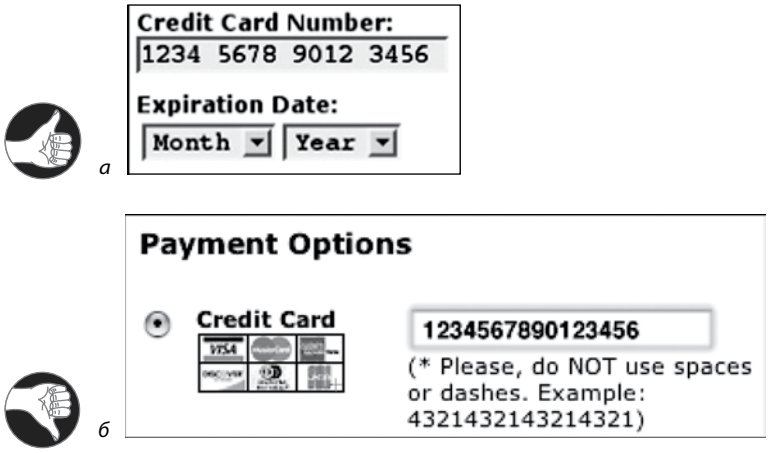


Рис. 3.7. На веб-сайте Democrats.org номер кредитной карты допускает пробелы (*а*); на веб-сайте Stufflt.com (*б*) в номере не должно быть пробелов, из-за чего введенную информацию сложнее проверить

Длинный номер можно разбить на части двумя способами: либо явно, создав в пользовательском интерфейсе отдельные поля для каждой части номера, либо просто позволив вводить число в единственное текстовое поле с использованием пробелов или знаков препинания (см. рис. 3.7, *а*). Тем не менее во многих современных представлениях телефонных номеров и номеров кредитных карт эти числа не сегментируются и пользователям не разрешается добавлять пробелы при вводе (см. рис. 3.7, *б*). Такое ограничение затрудняет восприятие номера и проверку правильности введения.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Сегментирование полей данных хорошо иллюстрирует визуальную структуру даже в том случае, когда вводимые данные, строго говоря, не являются числом. Например, сегментированные поля значительно улучшают удобочитаемость дат и помогают предотвращать ошибки при вводе информации, что иллюстрирует веб-сайт Bank of America (рис. 3.8).

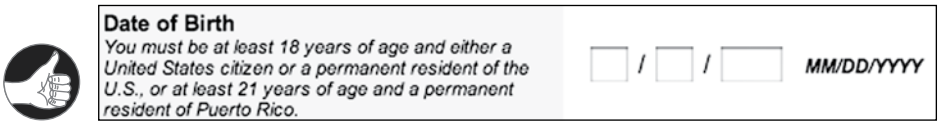


Рис. 3.8. Веб-сайт BankOfAmerica.com: сегментированные поля данных демонстрируют ожидаемую структуру данных

Специфические элементы управления улучшают структурирование ввода

Что касается структурирования вводимой информации, уровнем выше сегментированных полей данных находятся элементы управления для ввода данных конкретного типа. Вместо простых текстовых полей — неважно, сегментированных или нет — дизайнеры могут применять элементы управления, специально предназначенные для отображения (и считывания) значений определенного типа. Например, даты можно представлять (и считывать) в форме, в которой меню комбинируются с всплывающим элементом управления типа «календарь» (рис. 3.9).

Рис. 3.9. Веб-сайт NWA.com: даты отображаются и вводятся с помощью элемента управления, специально разработанного для календарных данных

Можно выделить визуальную структуру, совместив сегментированные текстовые поля с элементами управления для конкретных типов данных, как демонстрируют поля ввода адреса электронной почты на веб-сайте Southwest Airlines (рис. 3.10).

Рис. 3.10. Веб-сайт SWA.com: адрес электронной почты вводится в поля, структурированные специально для считывания его составляющих

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Визуальная иерархия помогает выделить важную информацию

Одна из наиболее важных задач структурированного представления информации — демонстрация визуальной иерархии, то есть системы структурирования информации, в которой:

- информация разбивается на отдельные секции, а большие секции подразделяются на подсекции;
- каждой секции и подсекции присваивается заметное обозначение, ясно идентифицирующее ее содержимое;
- секции и подсекции представляются в иерархической форме, и секции более высокого уровня выделяются сильнее, чем секции низких уровней.

Визуальная иерархия помогает людям при просмотре информации отделять важные данные от того, что в настоящий момент для них не имеет значения, и сосредоточиваться только на релевантной информации. Зритель быстрее обнаруживает искомое, так как может с легкостью пропустить все остальное.

Вы можете испытать это на себе. Взгляните на два информационных представления (рис. 3.11) и попробуйте найти сведения о выделении (prominence). Насколько больше времени вам потребовалось на поиск в неиерархическом представлении?

Create a Clear Visual Hierarchy

Organize and prioritize the contents of a page by using size, prominence, and content relationships. Let's look at these relationships more closely. The more important a headline is, the larger its font size should be. Big bold headlines help to grab the user's attention as they scan the Web page. The more important the headline or content, the higher up the page it should be placed. The most important or popular content should always be positioned prominently near the top of the page, so users can view it without having to scroll too far. Group similar content types by displaying the content in a similar visual style, or in a clearly defined area.

Create a Clear Visual Hierarchy

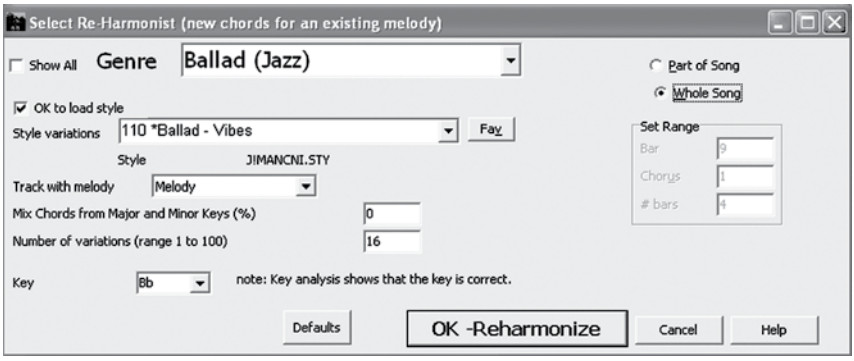
Organize and prioritize the contents of a page by using size, prominence, and content relationships.

Let's look at these relationships more closely:

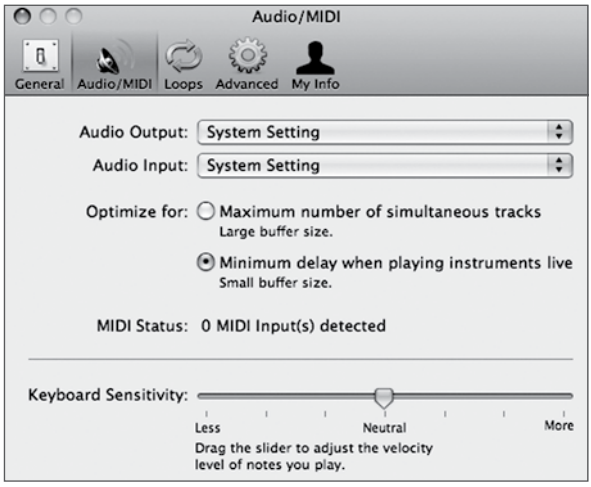
- **Size.** The more important a headline is, the larger its font size should be. Big bold headlines help to grab the user's attention as they scan the Web page.
- **Prominence.** The more important the headline or content, the higher up the page it should be placed. The most important or popular content should always be positioned prominently near the top of the page, so users can view it without having to scroll too far.
- **Content Relationships.** Group similar content types by displaying the content in a similar visual style, or in a clearly defined area.

Рис. 3.11. Найдите совет относительно выделения информации в каждом из двух представлений. Вам приходится полностью читать представление в форме сплошного текста (*слева*), однако визуальная иерархия (*справа*) позволяет проигнорировать информацию, не касающуюся вашей цели

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks



а



б

Рис. 3.12. Визуальная иерархия на интерактивных панелях управления и формах помогает пользователям быстро находить нужные настройки: *а* — плохой пример — Band in a Box; *б* — хороший пример — GarageBand

Примеры на рис. 3.11 демонстрируют значимость иерархии в текстовом информационном представлении, предназначенном только для чтения. Но визуальная иерархия точно так же — а может быть, и больше — важна на интерактивных панелях управления и формах. Сравните диалоговые окна двух музыкальных приложений (рис. 3.12). Диалоговое окно Reharmonize (приложение Band in a Box) имеет плохую визуальную иерархию, из-за чего пользователям трудно находить нужные настройки. В противоположность этому на панели управления Audio/MIDI (приложение GarageBand) хорошая визуальная иерархия, и пользователи быстро обнаруживают необходимые настройки и параметры.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 4

Чтение противоестественно

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Большинство людей в развитых странах выросли в окружении — как домашнем, так и школьном, — которое делало особый упор на образовании и чтении. Они научились читать в раннем детстве и к юности читали уже очень хорошо или отлично. У нас, взрослых людей, большая часть ежедневной активности включает чтение. Процесс чтения — интерпретации слов с целью понять их значение — у образованных людей происходит автоматически, то есть наш разум не фиксируется на этом процессе, и мы одновременно обдумываем значимость того, что читаем, и делаем выводы. Исходя из этого, умеющие хорошо читать люди полагают чтение такой же естественной человеческой деятельностью, как и умение разговаривать.

Мы запрограммированы на понимание языка, а не на чтение

Разговор и понимание устного языка — *действительно* естественные человеческие способности, однако к чтению это *не* относится. Сотни тысяч, а может, миллионы лет в человеческом мозге развивались нейронные структуры, необходимые для поддержки устной речи. В результате нормальные люди обладают врожденной способностью с младенческого возраста безо всяких систематических тренировок изучать тот язык, на котором говорят окружающие. Начиная приблизительно со школьного возраста, наша врожденная способность изучать язык на основе устной речи быстро угасает. В юности изучение нового языка ничем не отличается от получения любого другого навыка: требуются руководство и практика, а изучение и переработка знаний выполняются другими областями мозга — не теми, которые отвечали за освоение языка в раннем детстве (Sousa, 2005).

В отличие от устной речи, *письменная речь* и *чтение* появились лишь за несколько тысяч лет до Рождества Христова, а *широкое распространение* получили только четыре или пять веков назад — спустя долгое время после того, как человеческий мозг развился до его теперешнего состояния. Ни в каком возрасте ребенок не демонстрирует особых врожденных способностей к чтению. Нет, чтение — это искусственный навык, для обучения которому нужны систематические занятия и практика, и в этом смысле оно не отличается от игры на скрипке, жонглирования или чтения нот (Sousa, 2005).

Поскольку люди от природы не запрограммированы на чтение, дети, которым родители или воспитатели не читают вслух или которые не получают качественного образования в школе, могут не научиться читать. И таких людей в мире много, особенно в развивающихся странах. А людей, не научившихся устной речи, крайне мало.

По различным причинам даже люди, научившиеся читать, могут до конца жизни делать это плохо, с большим трудом. Возможно, их родители не говорили о пользе чтения. Может быть, они посещали плохую школу или вообще не получили школь-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

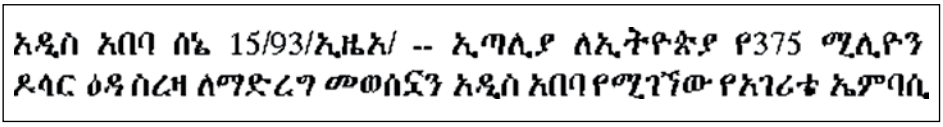
ного образования. Возможно, они овладели вторым языком, но так и не смогли научиться хорошо читать на нем. И наконец, люди с нарушениями когнитивных и познавательных функций, например страдающие дислексией, могут так никогда и не научиться бегло читать.

Для того чтобы научиться читать, нужно натренировать мозг, включая визуальную систему, на распознавание шаблонов. Наш мозг должен обрести навык восприятия целого диапазона шаблонов — от самого низкого до самого высокого уровня:

- линии, контуры и фигуры — это базовые визуальные характеристики, которые мозг умеет распознавать с рождения. Для того чтобы начать видеть их, специального обучения не требуется;
- объединение базовых характеристик создает объекты, в которых мы учимся распознавать символы — буквы, цифры и другие стандартные письменные элементы. В пиктографическом письме, например в китайском языке, символы представляют целые слова или концепции;
- в буквенной письменности шаблоны символов формируют морфемы, в которых мы учимся видеть единицы содержания. Например, в английском языке морфемами являются farm, tax, -ed, -ing;
- морфемы при объединении формируют шаблоны, которые для нас являются словами. Например, фрагменты farm, tax, -ed и -ing можно скомбинировать и получить слова farm, farmed, farming, tax, taxed, taxing. Даже пиктографические языки включают символы, служащие морфемами или модификаторами значения, а не отдельными словами или концепциями;
- сочетание слов формирует такие шаблоны, как словосочетания, идиоматические выражения, предложения;
- предложения объединяются в абзацы.

Для того чтобы понять, как текст выглядит для человека, не умеющего читать, просто взгляните на абзац текста, написанного на незнакомом вам языке или диалекте (рис. 4.1, *а* и *б*).

а



б

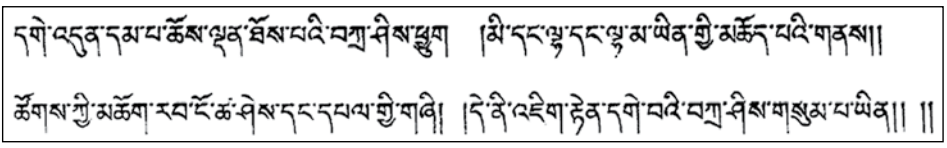


Рис. 4.1. Для того чтобы понять, что значить быть неграмотным, взгляните на текст на иностранном языке: *а* — амхарском; *б* — тибетском

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Помимо этого приблизительно понять впечатления неграмотного человека можно, взяв страницу текста, написанного на знакомом языке или диалекте, например страницу этой книги, и перевернув ее вверх ногами. Переверните эту книгу и постарайтесь прочитать следующие несколько абзацев. К слову, это упражнение создает лишь отдаленное впечатление о неграмотности — вы обнаружите, что перевернутый текст выглядит непонятным и нечитаемым лишь в первые несколько мгновений, но совсем скоро вы уже сможете читать его, правда, гораздо медленнее и с большим трудом.

Чтение — от характеристик или от контекста?

Как упоминалось ранее, чтение включает распознавание характеристик и шаблонов. Распознавание шаблонов и, следовательно, чтение может представлять собой либо восходящий процесс, отталкивающийся от характеристик, либо нисходящий, основывающийся на контексте.

Чтение «от характеристик» означает, что визуальная система начинает с обнаружения на странице или экране простых элементов — отрезков, линий под определенным углом или кривых определенного радиуса. Затем она объединяет их в более сложные характеристики, такие как углы, составные кривые, шаблоны. После этого мозг узнает в определенных фигурах образы или символы, представляющие буквы, цифры и, в идеографическом письме, слова. В буквенной письменности группы букв воспринимаются как морфемы или слова. Во всех видах письма последовательности слов складываются в осмысленные синтаксические конструкции, предложения и абзацы.

Чтение «от характеристик» иногда называют восходящим или контекстно-свободным процессом. Мозг человека умеет распознавать базовые характеристики — линии, грани, углы и т. п. — автоматически с самого рождения, это встроенный навык. Но распознаванию морфем, слов и синтаксических конструкций приходится учиться. Все начинается с неавтоматического, сознательного процесса, требующего осознанного анализа букв, морфем и слов, однако при наработке достаточного объема практики сознательный процесс превращается в автоматический (Sousa, 2005). Очевидно, что чем чаще встречаются морфема, слово или абзац, тем быстрее их распознавание переходит на автоматический уровень. Если речь идет об идеографическом письме (например, в китайском языке), включающем во много раз больше символов, чем языки с буквенной письменностью, то людям приходится потратить намного больше времени для того, чтобы научиться бегло читать.

Чтение «от контекста», или нисходящий процесс чтения, происходит одновременно с процессом чтения «от характеристик», но в противоположном направлении: от целых предложений или сути абзаца *вниз* до слов и символов. Сначала визуальная система распознает высокоуровневые шаблоны, такие как слова, обороты и предложения, возможно, благодаря тому, что мы заранее знаем, о чем говорится в данном тексте. Мозг использует это знание для того, чтобы вычислить — или догадаться,

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

какие компоненты составляют увиденный высокоуровневый шаблон (Boulton, 2009). Вероятность того, что чтение «от контекста» превратится в полностью автоматический процесс, разумеется, намного ниже: большинство шаблонов уровня синтаксических конструкций и предложений и соответствующих контекстов встречаются слишком редко, для того чтобы зафиксировать их в шаблонах нейронного возбуждения. Однако существуют исключения, такие как идиоматические выражения.

Для того чтобы получить представление о чтении «от контекста», быстро взгляните на рис. 4.2, а затем сразу же вернитесь к этому абзацу и продолжите чтение. Попробуйте сделать это прямо сейчас. О чем говорится в тексте?

The rain in Spain falls
manly in the the plain

Рис. 4.2. Нисходящее «распознавание» этой фразы может нарушить восприятие ее реального смысла

Теперь снова взгляните на это предложение и внимательно прочитайте каждое слово. Результат получился таким же?

Еще несколько десятилетий назад было установлено, что чтение включает обработку информации как «от характеристик» (восходящую), так и «от контекста» (нисходящую). Человек способен понять значение предложения путем анализа составляющих его букв и слов, но также умеет распознавать слова в предложении, исходя из смысла предложения, и буквы в слове на основе значения этого слова (рис. 4.3). Вопрос в следующем: беглое чтение — это в основном восходящий или нисходящий процесс? Или же ни одно из этих умений нельзя назвать доминирующим? Какой тип чтения предпочтительнее?

а Mray had a ltilte lmab, its feclee was withe as sown. And ervey wehre taht Mray wnet, the lmab was srue to go.

б Twinkle, twinkle little star, how I wonder what you are

Рис. 4.3. Нисходящее чтение: большинство читателей, если они знают песню, из которых взяты эти слова, могут без труда прочитать представленные предложения, несмотря на то, что: *а* — в словах переставлены все буквы, кроме первой и последней; *б* — большая часть символов скрыта

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В 1970-х годах ученые, занимающиеся исследованием проблем педагогики, применили к чтению информационную теорию и предположили, что вследствие избыточности письменного языка нисходящее чтение (чтение «от контекста») должно происходить быстрее, чем восходящее («от характеристик»). Это предположение привело их к гипотезе о том, что у людей, умеющих хорошо (быстро) читать, доминирует контекстная (нисходящая) обработка информации. Вероятно, как раз эта теория отвечает за появление в 1970–80-х годах множества методик быстрого чтения — предполагалось, что, применяя их, люди начинают быстрее читать, «впитывая» сразу целые фразы и предложения.

Однако эмпирические исследования процесса чтения, проведенные позже, убедительно доказали истинность противоположной теории. Кейт Станович (Keith Stanovich), исследователь, занимающийся проблемами чтения, объясняет: «...Контекст важен, но в основном он имеет значение для неопытных читателей, у которых еще не сформировалось автоматическое контекстно-свободное восприятие информации» (Boulton, 2009).

Другими словами, самый эффективный способ чтения — чтение «от характеристик» с помощью контекстно-свободных восходящих процессов, которые практиковались достаточно часто для того, чтобы стать автоматическими. Сегодня чтение «от контекста» в основном рассматривается как резервный метод, который хотя и работает одновременно с чтением «от характеристик», реальное значение имеет только в ситуациях, когда чтение «от характеристик» затруднено или еще не доведено до автоматизма.

Опытные читатели полагаются на контекст в случаях, когда чтение «от характеристик» невозможно из-за плохого представления информации (см. примеры далее в этой главе). Кроме того, в соревновании между этими двумя типами чтения контекстные подсказки порой побеждают характеристики представленной информации. Рассмотрим такой пример контекстного чтения: американцы во время путешествий по Англии часто неправильно читают знаки «To let» («Сдается») — вместо этой надписи они видят «Toilet» («Уборная»). Причина в том, что в США они часто видят слово toilet, однако почти никогда не встречают фразу «to let» — американцы на вывесках с предложением аренды помещений пишут «for rent».

У не слишком опытных читателей восприятие характеристик происходит не автоматически — это сознательный и трудоемкий процесс. Таким образом, у них контекстное чтение преобладает. Однако непроизвольное применение контекстного чтения и неумение автоматически распознавать характеристики морфем приводят к усилению кратковременной когнитивной нагрузки, что снижает степень понимания текста[1]. Таким людям приходится сосредоточиваться на том, чтобы

[1] О различиях между автоматической и контролируемой когнитивной обработкой информации говорится в главе 10. Пока же достаточно будет сказать, что контролируемые процессы, в отличие от автоматических, сильно нагружают рабочую память.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

расшифровать поток слов, и у них не остается ресурсов на то, чтобы постичь смысл предложений и абзацев. Вот почему неумелый читатель, вполне способный прочитать вслух целую фразу, иногда понятия не имеет о том, что же именно он только что прочитал.

Но почему же не все взрослые люди могут похвастаться навыком автоматического контекстно-свободного (восходящего) чтения? Кто-то не получил достаточного опыта чтения в детстве, для того чтобы процессы распознавания характеристик развились до нужного уровня. По мере взросления чтение таким людям начинает казаться слишком тяжелым, дающим большую психическую нагрузку занятием, и они избегают его. Это лишь закрепляет неумение и умножает трудности (Boulton, 2009).

У опытных и неопытных читателей работают разные части мозга

До 1980-х годов ученые, желавшие понять, какие части мозга вовлечены в обработку языка и чтения, ограничивались в основном исследованиями людей, переживших травмы мозга. Например, в середине XIX века обнаружили, что люди, мозг которых поврежден в области левого виска (сейчас эта область называется *полем Брока* в честь врача, обнаружившего ее), понимают речь, но с трудом могут говорить, а люди с повреждениями мозга в области левого уха (*зона Вернике*) не способны понимать речь (Sousa, 2005) (рис. 4.4).

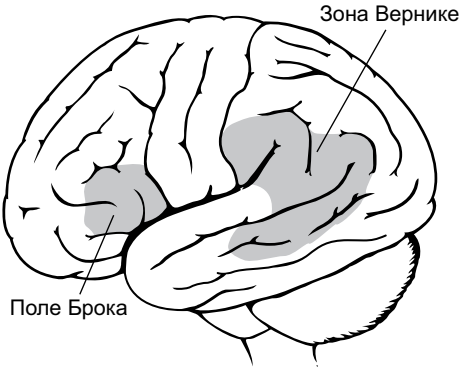


Рис. 4.4. Рисунок человеческого мозга с обозначенными полем Брока и зоной Вернике

В последние десятилетия были разработаны новые способы наблюдения за деятельностью функционирующего мозга живого человека, совершенствующие методы неинвазивного сканирования с помощью техник компьютерного анализа: электроэнцефалография (ЭЭГ), функциональная магнитно-резонансная томография (ФМРТ) и функциональная магнитно-резонансная спектроскопия (ФМРС). Эти методы позволяют исследователям наблюдать за реакциями различных областей

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

человеческого мозга, в том числе фиксировать последовательность ответов этих зон в то время, когда человек подвергается действию различных раздражителей или выполняет определенные задачи.

Применяя данные методы, исследователи обнаружили, что у неопытных и опытных читателей во время чтения задействуются разные нейронные проводящие пути. Разумеется, первой реагирует затылочная зона коры головного мозга, отвечающая за визуальное раздражение. Это происходит у всех людей независимо от того, насколько хорошо они умеют читать. После этого пути расходятся (Sousa, 2005):

- **неопытные читатели:** сначала активизируется область мозга чуть выше и позади зоны Вернике. Исследователи пришли к выводу, что именно в этой области слова «озвучиваются» и «собираются», то есть буквы анализируются и сопоставляются со звуками (по крайней мере, в языках с буквенной письменностью, таких как английский и немецкий). Затем область анализа слов обменивается информацией с полем Брока и фронтальной долей, где распознаются морфемы и слова — единицы смысла, а также извлекается общий смысл прочитанного. Что касается идеографических языков, где символы представляют целые слова, а их графическое начертание зачастую соответствует смыслу, то «озвучивание» не входит в процесс чтения;
- **опытные читатели:** область анализа слов не задействуется. Вместо этого активизируется затылочно-височная область (за ухом, рядом со зрительной зоной коры головного мозга). Преобладает мнение, что данная область распознает слова как единое целое, без «озвучивания и сборки», а вслед за этим активируются проводящие пути к фронтальной доле мозга, где извлекается смысл и ментальный образ слова. Поле Брока принимает в этом процессе незначительное участие.

Открытия, сделанные с помощью методов сканирования мозга, конечно же, не могут однозначно обозначить используемые процессы, однако они подтверждают теорию о том, что в мозге опытных читателей и новичков информация, получаемая с помощью чтения, обрабатывается по-разному.

Плохое оформление информации мешает чтению

Из-за небрежного написания или представления текста опытным читателям, у которых этот процесс происходит автоматически и не зависит от контекста, приходится возвращаться к осознанному, основанному на контексте чтению. Это нагружает рабочую память и, следовательно, снижает скорость чтения и уровень понимания. Если же человек не владеет навыком беглого чтения, то, возможно, плохо оформленный текст ему вообще не удастся прочитать.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Специфические или незнакомые термины

Одна из причин, по которой пользователей может сбивать с толку программное обеспечение, — использование незнакомых терминов, то есть слов, которых человек не понимает или в значении которых сомневается.

Вспомним, например, компьютерный жаргон. Представьте, что пользователь отвлекся и не работал с корпоративным веб-приложением в течение 15 минут, после чего на странице появилось такое сообщение: «Сеанс завершен. Пожалуйста, повторите аутентификацию».

Приложение предназначается для поиска внутри компании ресурсов, таких как помещения, оборудование и т. п. Среди его пользователей секретари, бухгалтеры, менеджеры, инженеры. Большинство пользователей, не обладающих техническим образованием, не поймут слова «аутентификация». Режим автоматического чтения сменится у них на сознательное обдумывание незнакомого термина: что же приложение хочет мне сказать? Для того чтобы не разрушать привычную модель чтения, разработчики приложения должны использовать более простую и привычную фразу, например: «Войдите в систему еще раз». Подробнее о том, как компьютерный жаргон в компьютерных системах влияет на обучение, говорится в главе 11.

Модель чтения разрушается в любой ситуации, когда человек сталкивается с незнакомыми терминами, и это относится не только к компьютерному жаргону. Далее перечислены некоторые слова, которые встречаются в основном в контрактах, положениях о конфиденциальности и прочих юридических документах:

- *вышеупомянутый* — упомянутый раньше, прежде;
- *юрисдикция* — область, на которую распространяется право должностного лица производить суд, рассматривать правовой вопрос;
- *отрекаться* — отказываться, отклонять;
- *до сего времени* — до настоящего момента, ранее;
- *юриспруденция* — принципы и теории, на которых основывается правовая система;
- *обфусцировать* — запутывать, затемнять, делать более сложным для понимания;
- *предконечный* — предпоследний, например «предпоследняя глава книги».

Когда читатели, даже опытные, встречаются с подобным словом, их автоматические процессы чтения дают сбой, так как не могут опознать его. Мозгу приходится применять менее автоматизированные процессы, например, разбирать слово на части и пытаться понять его значение. Кроме того, мозг пробует извлечь значение слова из контекста. (Либо человеку приходится заглядывать в словарь.)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Сложные написания и шрифты

Даже когда все слова знакомы читателю, процесс чтения может нарушаться из-за тяжелых для восприятия вариантов написания и шрифтов. Восходящее, контекстно-свободное автоматическое чтение основывается на распознавании букв и слов, исходя из их визуальных характеристик. Таким образом, шрифт со сложными для узнавания характеристиками и фигурами будет трудно читать. Например, попытайтесь прочитать геттисбергскую речь Авраама Линкольна, оформленную ПРОПИСНЫМИ БУКВАМИ с помощью контурного шрифта (рис. 4.5).

ABRAHAM LINCOLN'S GETTYSBURG ADDRESS

FOURSCORE AND SEVEN YEARS AGO, OUR FOREFATHERS BROUGHT FORTH ON THIS CONTINENT A NEW NATION, CONCEIVED IN LIBERTY AND DEDICATED TO THE PROPOSITION THAT ALL MEN ARE CREATED EQUAL.

NOW WE ARE ENGAGED IN A GREAT CIVIL WAR, TESTING WHETHER THAT NATION, OR ANY NATION SO CONCEIVED AND SO DEDICATED, CAN LONG ENDURE. WE ARE MET ON A GREAT BATTLEFIELD OF THAT WAR. WE HAVE COME TO DEDICATE A PORTION OF THAT FIELD, AS A FINAL RESTING PLACE FOR THOSE WHO HERE GAVE THEIR LIVES THAT THAT NATION MIGHT LIVE. IT IS ALTOGETHER FITTING AND PROPER THAT WE SHOULD DO THIS.

BUT, IN A LARGER SENSE, WE CAN NOT DEDICATE – WE CAN NOT CONSECRATE – WE CAN NOT HALLOW – THIS GROUND. THE BRAVE MEN, LIVING AND DEAD, WHO STRUGGLED HERE, HAVE CONSECRATED IT, FAR ABOVE OUR POOR POWER TO ADD OR DETRACT. THE WORLD WILL LITTLE NOTE, NOR LONG REMEMBER WHAT WE SAY HERE, BUT IT CAN NEVER FORGET WHAT THEY DID HERE. IT IS FOR US THE LIVING, RATHER, TO BE DEDICATED HERE TO THE UNFINISHED WORK WHICH THEY WHO FOUGHT HERE HAVE THUS FAR SO NOBLY ADVANCED. IT IS RATHER FOR US TO BE HERE DEDICATED TO THE GREAT TASK REMAINING BEFORE US – THAT FROM THESE HONORED DEAD WE TAKE INCREASED DEVOTION TO THAT CAUSE FOR WHICH THEY GAVE THE LAST FULL MEASURE OF DEVOTION – THAT WE HERE HIGHLY RESOLVE THAT THESE DEAD SHALL NOT HAVE DIED IN VAIN – THAT THIS NATION, UNDER GOD, SHALL HAVE A NEW BIRTH OF FREEDOM – AND THAT GOVERNMENT OF THE PEOPLE, BY THE PEOPLE, FOR THE PEOPLE, SHALL NOT PERISH FROM THE EARTH.

Рис. 4.5. Текст, написанный ПРОПИСНЫМИ БУКВАМИ, довольно сложно читать, так как буквы очень похожи. Контурные шрифты затрудняют распознавание. Данный пример иллюстрирует обе проблемы

Мелкий шрифт

Еще один способ сделать текст в приложениях, на веб-сайтах и электронных устройствах нечитаемым — использовать слишком мелкие шрифты. Визуальная система читателя с большим трудом разбирает символы такого размера. Попытайтесь прочитать первый абзац конституции США, написанный мелким шрифтом (рис. 4.6).

We the people of the United States, in Order to form a more perfect Union, establish Justice, insure domestic Tranquility, provide for the common defense, promote the general Welfare, and secure the Blessings of Liberty to ourselves and our Posterity, do ordain and establish this Constitution for the United States of America.

Рис. 4.6. Вступительный абзац конституции США: размер шрифта семь пунктов

Иногда разработчики применяют крошечные шрифты, так как перед ними стоит задача уместить большой объем текста на маленькой площади. Но с учетом того,

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

что целевая аудитория оказывается не в состоянии прочитать этот текст, или же людям приходится прикладывать огромные усилия, для того чтобы сделать это, с таким же успехом текст можно было бы вообще не добавлять на экран или страницу.

Текст на шумном фоне

Визуальный шум внутри и вокруг текста мешает распознаванию характеристик, символов и слов и, следовательно, переводит чтение из автоматического режима «от характеристик» в сознательный, более контекстно-зависимый режим. В программных пользовательских интерфейсах и на веб-сайтах визуальный шум часто возникает вследствие того, что дизайнеры помещают текст поверх узорного фона либо используют малоконтрастные цвета, как в примере с веб-сайта RedTele.com (рис. 4.7).

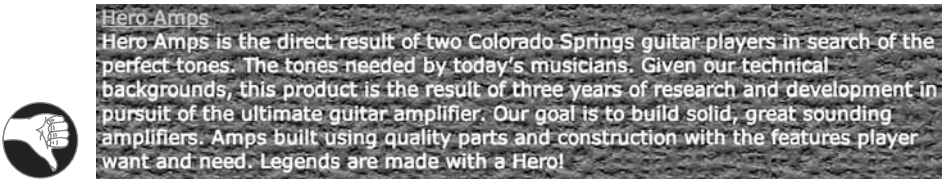


Рис. 4.7. Веб-сайт RedTele.com: текст отображен поверх шумного фона, к тому же цвет букв плохо контрастирует с цветом фонового узора

Однако бывают ситуации, когда дизайнеры намеренно делают текст трудночитаемым. Например, в Сети распространена такая мера обеспечения безопасности: показать пользователю слово в искаженном написании и попросить ввести его с клавиатуры, чтобы подтвердить, что с системой взаимодействует человек, а не бот. В основе данного метода лежит тот факт, что большинство людей в состоянии прочитать текст, который современным ботам распознать не под силу. Трудночитаемый текст, с помощью которого проверяется принадлежность пользователя к человеческому роду, называется «капча»[1] (рис. 4.8).

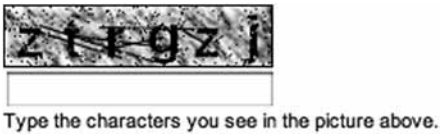


Рис. 4.8. Капча: текст намеренно выводится на «шумном» фоне, для того чтобы его невозможно было распознать с помощью программного обеспечения

[1] Слово «капча» (captcha) произошло от английского слова «capture» — ловить, захватывать. Существует также мнение, что это аббревиатура, расшифровывающаяся как «Completely Automated Public Turing test to tell Computers and Humans Apart» (полностью автоматизированный публичный тест Тьюринга для различения компьютеров и людей) — см. статью «Капча» (или «Captcha») в сетевой энциклопедии Wikipedia.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

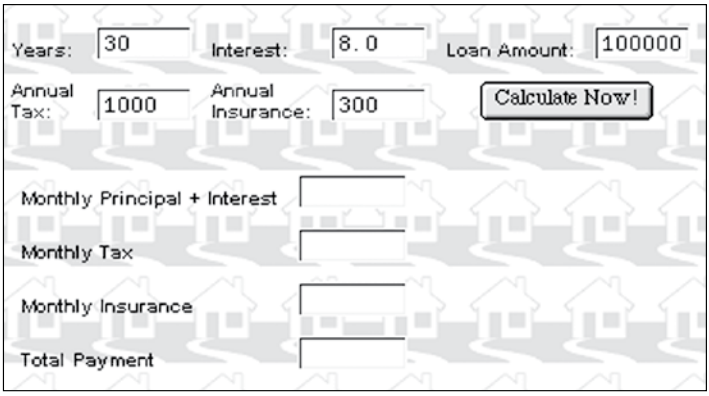


Рис. 4.9. В интерактивном ипотечном калькуляторе ФРБ текст раньше отображался на узорчатом фоне

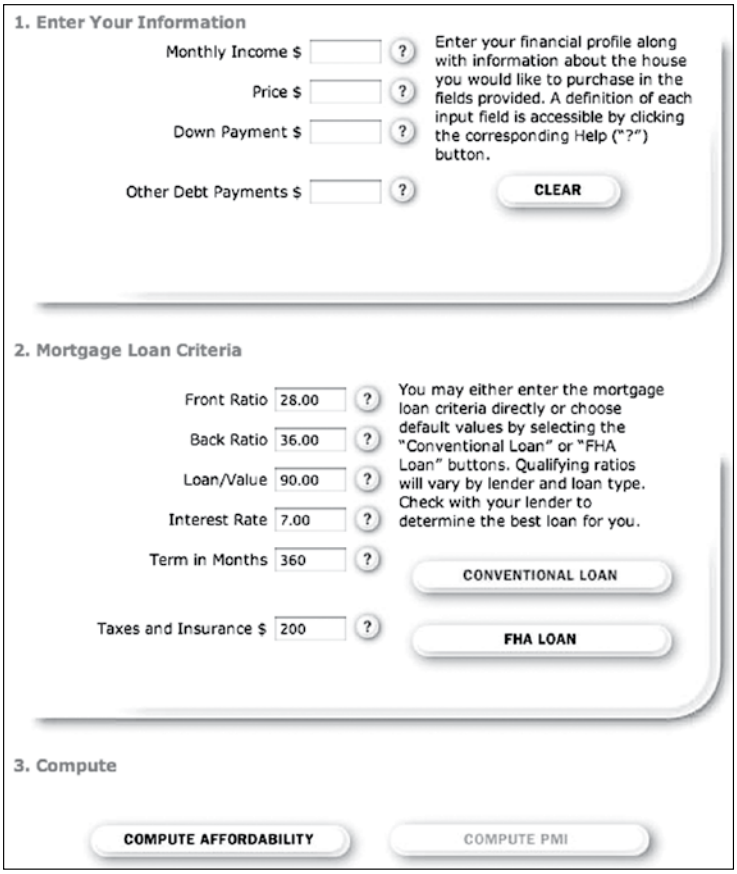


Рис. 4.10. Новая версия ипотечного калькулятора на веб-сайте ФРБ: текст на белом фоне

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Но, конечно же, в *большинстве* случаев текст в пользовательском интерфейсе должен читаться *легко*. Узорному фону не обязательно быть очень ярким, чтобы отвлекать внимание читателя от текста. Например, в комплекте веб-сайтов Федерального резервного банка (ФРБ) раньше был ипотечный калькулятор, украшенный повторяющимся фоновым узором домиков и лужаек в неярких, пастельных тонах (рис. 4.9). Разумеется, это не входило в намерения дизайнеров, но из-за фонового рисунка текст на странице было довольно сложно читать. Позже, когда Федеральный Резервный Банк изменил дизайн калькулятора и расширил его функциональность, от узорного фона избавились (рис. 4.10).

Информация прячется среди многочисленных повторений

Визуальный шум может исходить и от самого текста. Если в компактном тексте слишком много повторений, читатель не только теряет строку, что само по себе затрудняет чтение, но и с трудом извлекает из последовательности символов действительно важную информацию. Например, вспомните веб-сайт Калифорнийского Департамента автотранспорта из предыдущей главы (см. рис. 3.2).

Еще один пример повторений, создающих неразбериху и шум, можно найти в компьютерном магазине на сайте Apple.com. На страницах заказа настольного компьютера перечисляются различные варианты клавиатур, причем строки текста очень похожи. Из-за этого пользователю сложно заметить, что наиболее существенная разница между клавиатурами заключается в том, какой язык они поддерживают (рис. 4.11).

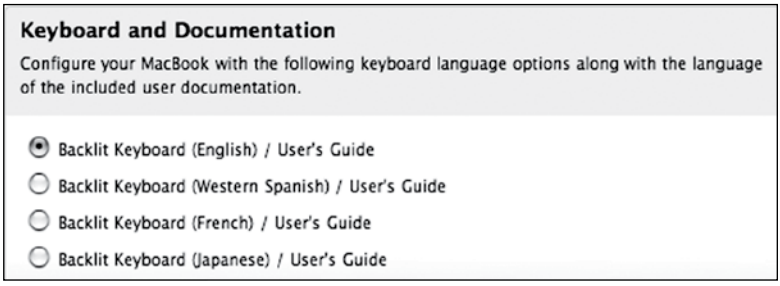


Рис. 4.11. На странице Buy Computer (Купить компьютер) веб-сайта Apple.com доступные варианты оформлены так, что самая важная информация (совместимость с тем или иным языком) теряется среди повторяющихся символов

Выравнивание по центру

Один из наиболее автоматизированных аспектов чтения у опытных читателей — это движения глаз. При автоматическом (быстром) чтении, дочитав строку, мы по привычке переводим взгляд на начальную позицию по горизонтали, но на одну

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

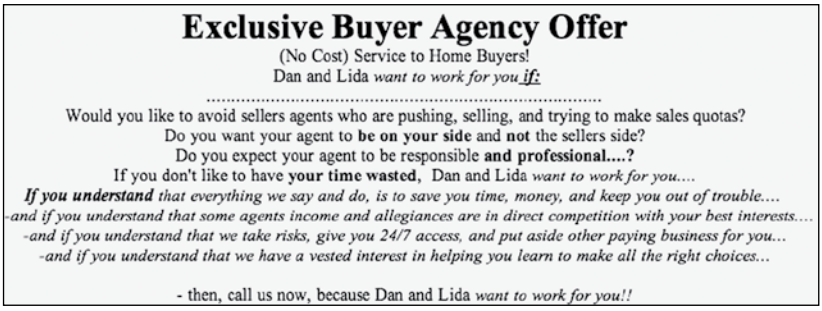


Рис. 4.12. На веб-сайте FargoHomes.com текст выровнен по центру, и это разрушает шаблон автоматического перемещения взгляда

Рис. 4.13. На веб-сайте FargoHomes.com нумерованные списки выровнены по центру, и это окончательно разрушает шаблон автоматического перемещения взгляда

строку ниже. Если текст выровнен по центру или по правому краю страницы, то первые буквы строк расположены не в столбец, а разнесены по разным позициям. Таким образом, автоматическое движение глаз переносит взгляд в неверную точку, и нам приходится сознательно корректировать себя, отыскивая взглядом *настоящее* начало строки. Мы выпадаем из автоматического режима, и скорость чтения существенно снижается. Если речь идет о стихах или приглашениях на свадьбу, то это не очень серьезная проблема, но для текста другого типа будет серьезным недостатком. Пример такого выровненного по центру информационного текста можно найти на веб-сайте агентства недвижимости FargoHomes (рис. 4.12). Попытайтесь быстро прочитать этот текст и понять, как двигаются ваши глаза.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

На этом сайте по центру выравниваются и нумерованные списки, что полностью разрушает модель автоматического чтения с привычными движениями глаз (рис. 4.13). Попробуйте пробежать по списку взглядом.

Сочетание недостатков, мешающих чтению

Веб-сайт Keller Williams, еще одного агентства недвижимости, объединяет многие из перечисленных ранее способов помешать автоматическому чтению и восприятию информации. Здесь можно найти и недостаточно контрастные цвета переднего плана и фона, и слишком контрастные, например синий текст на красном фоне, который противно мерцает при попытке прочитать его. Кроме того, сплошной текст выровнен по центру, а в некоторых местах под текст подложен узорчатый фон. Все это вместе взятое чрезвычайно затрудняет чтение данного сайта (рис. 4.14).

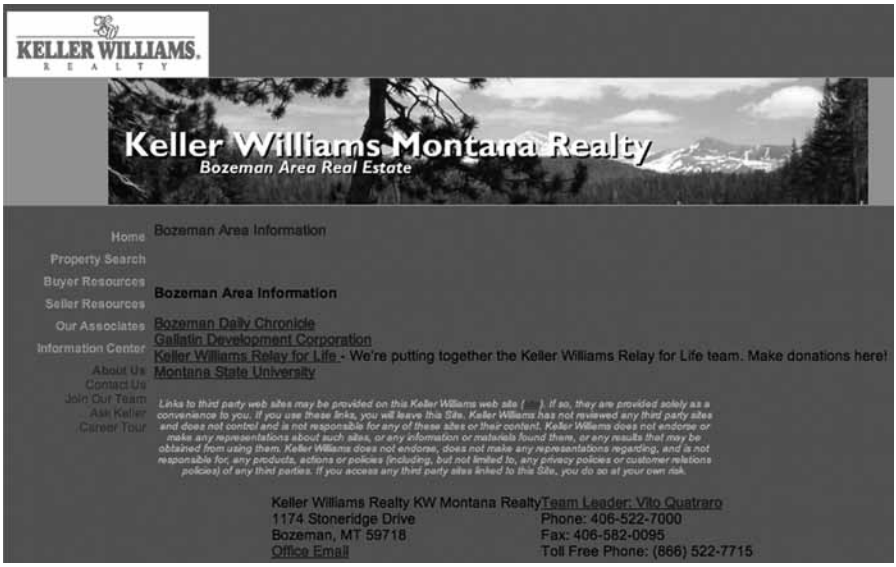


Рис. 4.14. Текст на веб-сайте Keller Williams нечитаемый по множеству причин

Выводы для дизайнера: не мешайте чтению, помогайте читателю!

Очевидно, что дизайнер должен ставить своей целью помочь читателю, а не препятствовать чтению. Беглое чтение — почти полностью автоматический процесс, основанный, по большей части, на распознавании характеристик, символов и слов. Чем проще происходит распознавание, тем проще человеку читать и тем быстрее он это делает. В противоположность этому менее опытным читателям большую помощь оказывают *контекстные* подсказки.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Для поддержания обоих способов чтения дизайнерам интерактивных систем следует придерживаться следующих рекомендаций:

- Текст в пользовательском интерфейсе должен помогать эффективному автоматическому распознаванию характеристик, а для этого необходимо избегать отвлекающих деталей, описанных ранее, таких как мелкие шрифты, узорчатые фоны, выравнивание по центру и т. п.
- Используйте ограниченный, хорошо согласованный лексикон; иногда такой словарь называют *простым языком*[1] или *упрощенным языком* (Redish, 2007).
- Форматируйте текст так, чтобы создавать визуальную иерархию, упрощающую восприятие текста при быстром просмотре, — используйте заголовки, маркированные списки, таблицы, визуальное выделение слов (рис. 4.15).

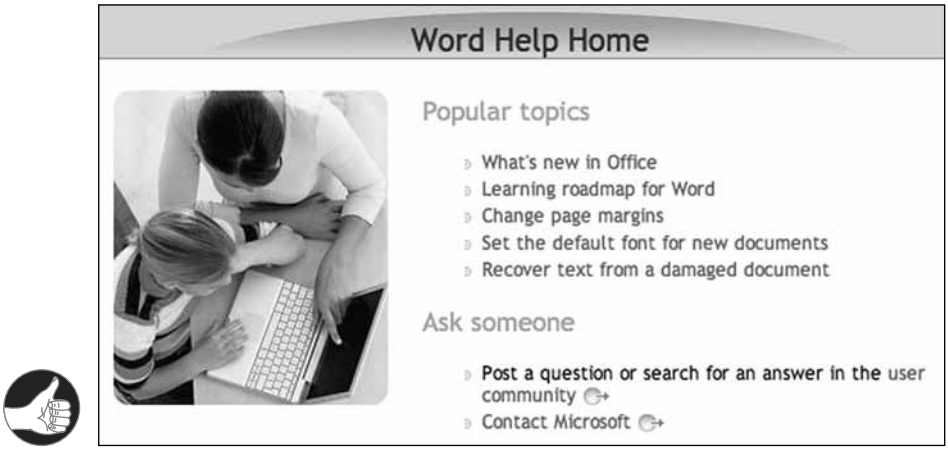


Рис. 4.15. Информацию на странице справки Microsoft Word легко просматривать и читать

Опытные специалисты по информационной архитектуре, редакторы и графические дизайнеры помогут вам создать легко читаемое представление текста, помогающее успешно извлекать важную информацию даже при быстром просмотре.

Большую часть написанного читать необязательно

Помимо типичных дизайнерских ошибок, мешающих чтению, многие пользовательские интерфейсы страдают чрезмерным избытком текста и заставляют пользователя читать больше, чем необходимо. Подумайте, сколько ненужного текста

[1] Подробнее о простом языке рассказывается на веб-сайте правительства США: http://www.plainlanguage.gov.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

использовано в диалоговом окне настройки параметров ввода текста в приложении SmartDraw (рис. 4.16).

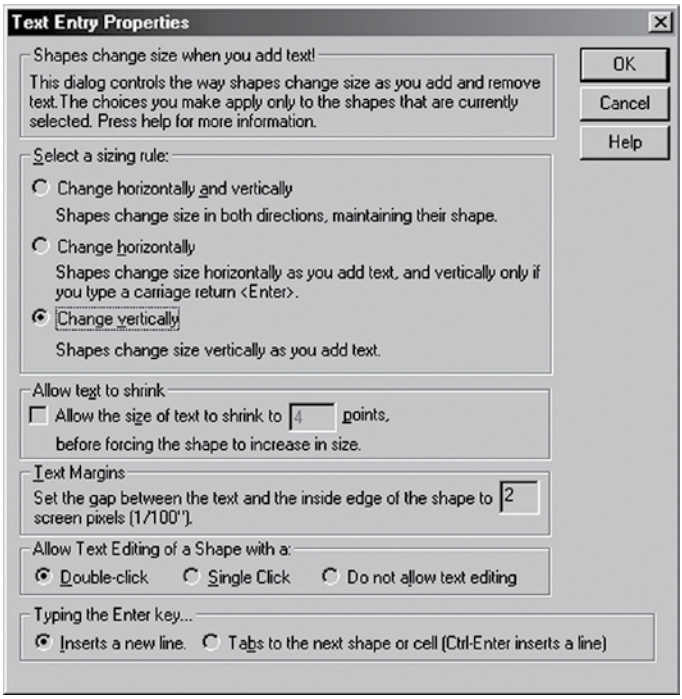


Рис. 4.16. Диалоговое окно Text Entry Properties (Параметры ввода текста) приложения SmartDraw: слишком много текста для такой простой функциональности

Разработчики программного обеспечения зачастую оправдывают многословные инструкции тем, что «им необходимо подробно объяснять пользователям, что следует сделать». Однако большинство подобных объяснений можно сократить без потери смысла. Давайте посмотрим, как в промежутке между 2002 и 2007 годами компания Jeep укорачивала инструкцию по поиску ближайшего дилера автомобилей Jeep (рис. 4.17):

- **2002 год** — на странице Find a Dealer (Найти дилера) мы видим большие абзацы сплошного текста, внутри которых запрятаны нумерованные списки, а также форму, запрашивающую больше информации, чем требуется для поиска ближайшего дилера;
- **2003 год** — инструкция по поиску дилера на странице Find a Dealer (Найти дилера) сокращена до маркированного списка из трех пунктов, а форма запрашивает меньше информации;
- **2007 год** — задача поиска дилера решается с помощью поля для ввода почтового индекса и кнопки Go (Искать) на домашней странице веб-сайта.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

2002

2003

2007

Рис. 4.17. В промежутке между 2002 и 2007 годами дизайнеры Jeep.com существенно сократили объем инструкций, помогающих найти ближайшего дилера

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Даже если текст предназначен для описания продукта, а не объяснения последовательности действий, перечисление всех свойств и характеристик в огромном абзаце сплошного текста приводит к результатам, обратным желаемым. Большинство потенциальных клиентов или покупателей не смогут или не захотят прочитать его от начала до конца. Сравните описания ноутбуков на веб-сайте Costco.com за 2007 и 2009 годы (рис. 4.18).

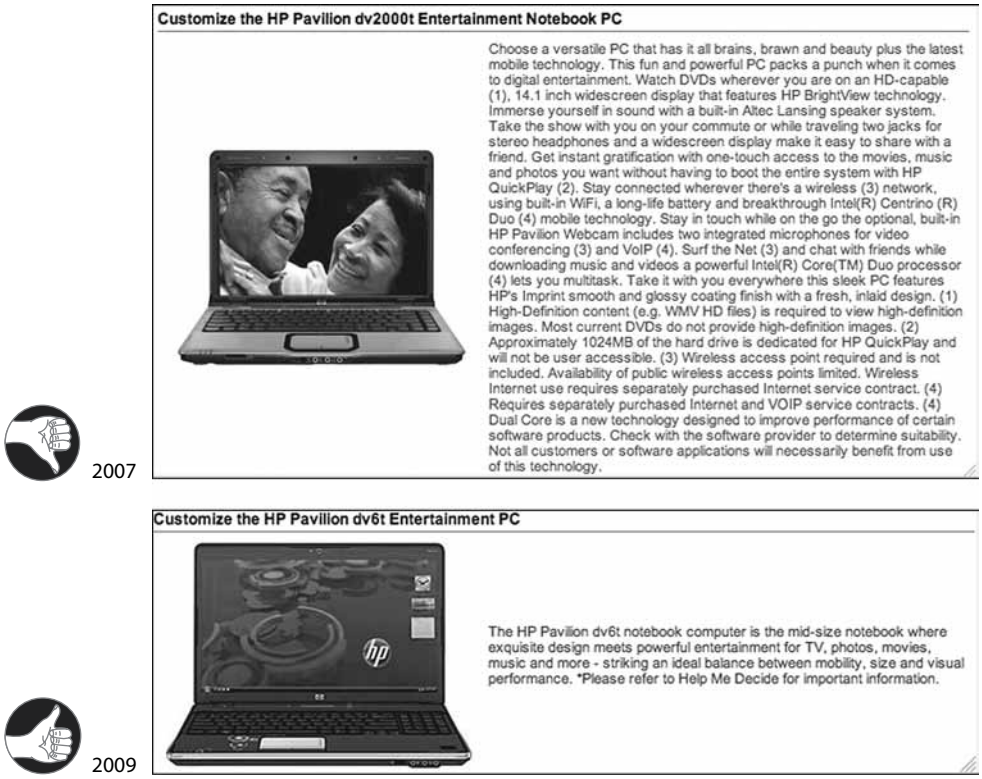


Рис. 4.18. В промежутке между 2007 и 2009 годами дизайнеры Costco.com существенно сократили объем текста в описании продуктов

Выводы для дизайнера: минимизируйте необходимость в чтении

Пользователи, не умеющие бегло читать, но составляющие, к сожалению, значительную часть населения, быстро теряют интерес к интерфейсам со слишком большим объемом текста. Однако чрезмерное количество текста утомляет и заставляет отвернуться и людей, умеющих читать *хорошо*: им не нравится, что использование интерактивной системы превращается в тяжелую *работу*.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Минимизируйте объем сплошного текста в пользовательских интерфейсах, не заставляйте пользователей читать длинные абзацы. Заключайте в инструкции *наименьший* объем информации, необходимый для того, чтобы пользователь добился желаемого результата. В описания продуктов помещайте только краткую аннотацию и предоставляйте пользователю возможность запросить дополнительную информацию. Помочь с подбором инструкций и оформлением текста вам могут технические писатели и редакторы. Дополнительные советы относительно того, как избавляться от лишнего текста, вы найдете в книгах Круга (Krug, 2005) и Редиша (Redish, 2007).

Проверяйте на реальных людях

Наконец, дизайнерам следует тестировать свои разработки на целевой аудитории пользователей, для того чтобы быть уверенными в том, что люди смогут быстро и без усилий прочитать весь необходимый текст. Часть тестирования можно провести заранее, на прототипах и предварительных реализациях, но перед выпуском релиза обязательно должна быть выполнена полная проверка. К счастью, срочные изменения в размеры и форматы шрифтов внести совсем несложно.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 5

Цветовое зрение ограничено

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

У способности людей воспринимать цвет есть свои сильные стороны и ограничения, многие из которых важно учитывать при разработке пользовательских интерфейсов:

- наше зрение оптимизировано для распознавания контрастных цветов (границ), а не абсолютной яркости;
- умение различать цвета зависит от того, каким образом цвета представлены;
- некоторые люди не различают цветов;
- монитор пользователя и условия, в которых он работает, влияют на восприятие цветов.

Чтобы разобраться в этих особенностях человеческого цветового зрения, нужно начать с основ. Давайте рассмотрим, как человеческая визуальная система обрабатывает цветовую информацию, получаемую из окружающей среды.

Как работает цветовое зрение

Если у вас в университете был начальный курс психологии или нейрофизиологии, то вы, вероятно, знаете, что на внутренней сетчатой оболочке глаза — поверхности, на которой фокусируется изображение, — находятся два типа рецепторных клеток, чувствительных к свету: палочки и колбочки. Также вы, скорее всего, знаете, что палочки чувствительны к уровню освещенности, но не помогают нам распознавать цвета — это задача колбочек. Наконец, известно, что существует три типа колбочек: чувствительные к красному, зеленому и синему цветам. Получается, что цветовое зрение имеет много общего с видеокамерами и компьютерными экранами, которые распознают или проецируют широкий диапазон цветов с помощью комбинаций красных, зеленых и синих пикселов.

То, о чем вы узнали в университете, истинно лишь наполовину. На сетчатке действительно есть палочки и три типа колбочек. Палочки чувствительны к общему уровняю яркости, а три типа колбочек — к различным длинам световых волн. Приблизительно здесь и начинается расхождение между правдой и тем, чему до недавних пор учили в университете.

Во-первых, мы, представители индустриального общества, почти не используем палочки. Они работают только при очень низком уровне освещенности. Палочки предназначены для ориентирования в плохо освещенных местах, то есть в обстановке, в которой наши предки жили вплоть до XIX века. Сегодня палочки работают, только когда мы ужинаем при свечах, ночью пробираемся по дому на кухню, выходим в ночной поход и т. п. При ярком свете и в помещениях с искусственным освещением — там, где мы проводим большую часть времени, — палочки не способны предоставить никакой полезной информации. В основном наше зрение ориентируется на информацию, получаемую от колбочек (Ware, 2008).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Так как же работают колбочки? Действительно ли существует три типа колбочек, чувствительных к красному, зеленому и синему цветам соответственно? В действительности все типы колбочек чувствительны к куда более широким диапазонам длин волн, чем можно было бы ожидать, причем диапазоны чувствительности разных типов колбочек в значительной степени совпадают. Кроме того, общая чувствительность трех типов колбочек сильно разнится (рис. 5.1, *а*):

- **Длинные волны.** Приспособленные для их приема колбочки чувствительны почти ко всему диапазону видимого света, однако лучше всего воспринимают волны средней (желтый цвет) и большой (красный цвет) длины (график *1*).
- **Средние волны.** Приспособленные для их приема колбочки реагируют на световые волны в диапазоне от коротких, ближе к среднему (синий цвет), до средних, ближе к длинным (желтый и оранжевый). В целом они менее чувствительны, чем колбочки, реагирующие на длинные волны (график *2*).
- **Короткие волны.** Приспособленные для их приема колбочки наиболее чувствительны к свету в правом конце видимого спектра, то есть к фиолетовому и синему цвету, но слабо реагируют на волны средней длины, то есть на зеленый цвет (график *3*). Эти колбочки обладают самой низкой чувствительностью из всех трех типов, и их меньше, чем остальных. Это объясняет, почему наши глаза менее чувствительны к синим и фиолетовым оттенкам, чем к другим цветам.

Сравните графики реальной чувствительности колбочек сетчатки к свету (рис. 5.1, *а*) с вариантом, иллюстрирующим, как инженеры-электрики могли бы сконструировать сетчатку по аналогии с камерой: как мозаику из рецепторов, чувствительных к красному (график *6*), зеленому (график *5*) и синему (график *4*) цветам (рис. 5.1, *б*).

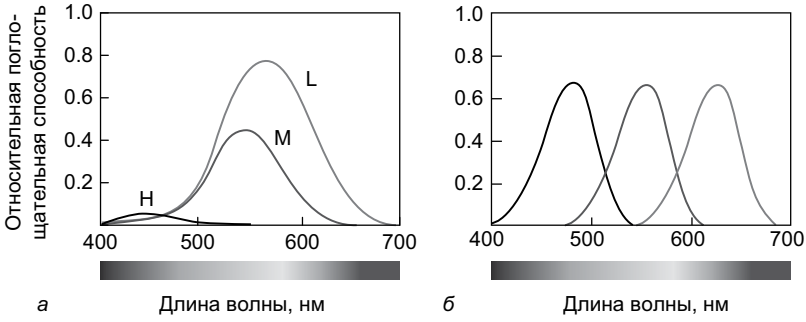


Рис. 5.1. Чувствительность: *а* — трех типов колбочек; *б* — искусственных красных, зеленых и синих рецепторов

Учитывая необычное соотношение показателей чувствительности трех типов колбочек, можно задать вопрос: каким образом наш мозг комбинирует сигналы колбочек, позволяя видеть широкий диапазон цветов?

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Ответ — путем *вычитания*. Нейроны в зрительной зоне коры головного мозга (в тыльной части мозга) *вычитают* сигналы, поступающие по оптическим нервам от средневолновых и длинноволновых колбочек, получая в результате «красно-зеленый» канал *разностного* сигнала. Другие нейроны в зрительной зоне коры вычитают сигналы от коротко- и длинноволновых колбочек, получая «желто-синий» канал разностного сигнала. Третья группа нейронов оптической зоны *складывает* сигналы, поступающие от длинно- и средневолновых колбочек, создавая общий канал сигнала *яркости* (или «черно-белый» канал)[1]. Эти три канала называются *цветооппонентными каналами*.

После этого мозг применяет дополнительные разностные процессы ко всем трем цветооппонентным каналам: сигналы, поступающие от определенной области сетчатки, фактически вычитаются из аналогичных сигналов, поступающих от соседних зон сетчатки.

Зрение оптимизировано для восприятия контраста в пограничных зонах

Все эти разностные процессы приводят к тому, что наша визуальная система гораздо лучше воспринимает различия в цвете и яркости, то есть контрастные границы, а не абсолютные уровни яркости. Для того чтобы убедиться в этом, сравните два круга (рис. 5.2). Они закрашены одним и тем же оттенком — правый круг представляет собой всего лишь копию левого, но из-за разных фоновых цветов чувствительная к контрасту визуальная система считает, что левый круг темнее.

Чувствительность визуальной системы к контрасту, а не к абсолютной яркости — это преимущество, которое помогало, например, нашим далеким предкам одинаково успешно замечать леопарда в кустах неподалеку как в яркий солнечный полдень, так и ранним утром в пасмурный день. Аналогично, благодаря чувствительности к цветовым контрастам, а не к абсолютным цветам мы всегда видим цвет красной розы одинаково, независимо от того, находимся на солнце или в тени.

Эдвард Х. Адельсон (Edward H. Adelson), занимающийся исследованиями мозга в Массачусетском технологическом институте, разработал поразительную иллюстрацию, доказывающую нечувствительность нашей визуальной системы к яркости и высокую чувствительность к контрасту (рис. 5.3). В это трудно поверить, но оттенки полей A и B на шахматной доске совпадают. Поле B кажется белым только потому, что оно находится в тени цилиндра.

---

[1] В сумме общей яркости отсутствует составляющая, получаемая от коротковолновых (сине-фиолетовых) колбочек. Эти колбочки настолько малочувствительны, что их вкладом в общее значение можно пренебречь, и отсутствие данной составляющей почти не играет роли.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

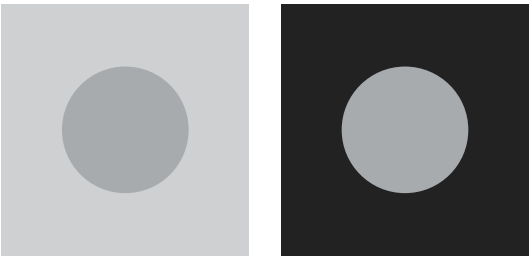


Рис. 5.2. Из-за того что цвета фона на картинках разные, кажется, что круги закрашены разными оттенками, однако они абсолютно одинаковые

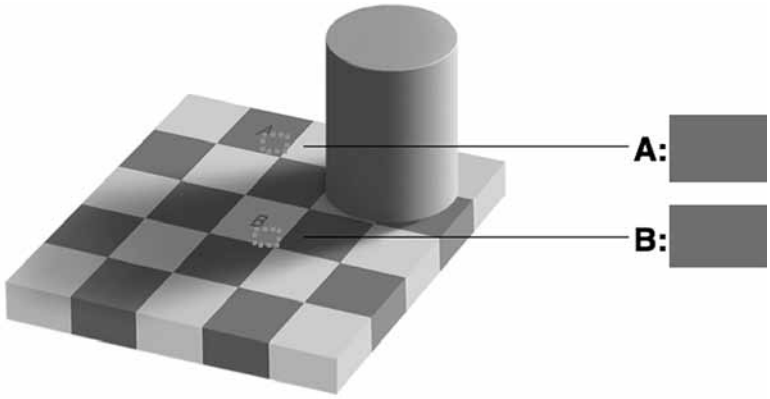


Рис. 5.3. Поля A и B окрашены одинаковым оттенком серого цвета. Поле B кажется нам белым, потому что оно находится в тени

Способность различать цвета зависит способа их представления

Даже наше умение распознавать различия между цветами ограничено. Вследствие особенностей работы визуальной системы три следующих фактора представления способны повлиять на ее способность отличать один цвет от другого:

- **бледность** — чем более бледные (менее насыщенные) цвета представлены, тем сложнее их отличать друг от друга (рис. 5.4, *а*);
- **размер образца цвета** — чем меньше или тоньше объекты, тем сложнее различать их цвета (рис. 5.4, *б*). Для отображения текста зачастую используются тонкие линии, поэтому становится очень трудно определить точный цвет букв;
- **разъединение** — чем дальше друг от друга находятся образцы цвета, тем сложнее различать их цвета, особенно если расстояние настолько велико, что требуется движение глаз (рис. 5.4, *в*).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

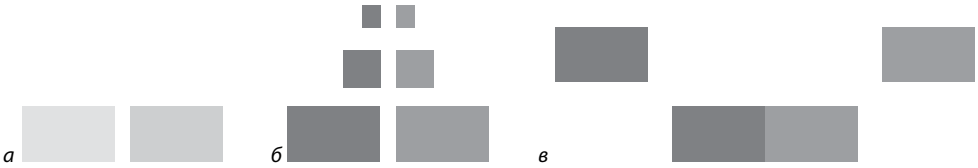


Рис. 5.4. Факторы, влияющие на способность различать цвета: *а* — бледность; *б* — размер; *в* — разъединение

Несколько лет назад на веб-сайте туристического агентства ITN.net для обозначения шагов процесса бронирования использовались два бледных цвета: бледно-желтый для активного шага (им выполнен фон для цифры 1) и белый для остальных (рис. 5.5). Некоторые пользователи не могли из этого представления понять, на каком шаге они находятся.

Рис. 5.5. Веб-сайт ITN.net (образца 2003 года): текущий шаг выделяется слишком бледным цветом, и пользователям трудно увидеть, на каком шаге процесса бронирования авиабилета они находятся

Маленькие образцы цвета зачастую можно увидеть на диаграммах и графиках. Многие пакеты для создания деловой графики создают на диаграммах «легенды», но делают образцы цвета слишком маленькими (рис. 5.6). Цветовые образцы на легенде диаграммы должны быть достаточно большими для того, чтобы у пользователя не было сложностей с различением и сопоставлением цветов (рис. 5.7).

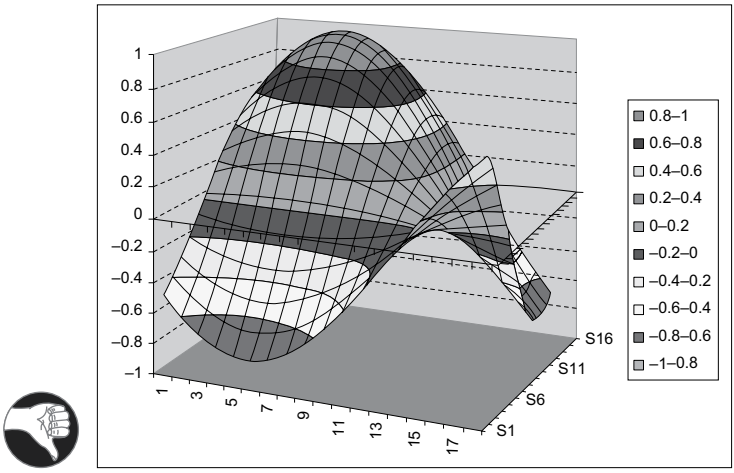


Рис. 5.6. Крошечные образцы цвета на легенде этой диаграммы сложно различать

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

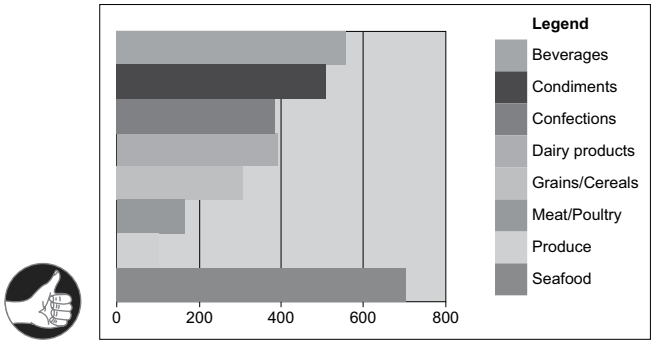


Рис. 5.7. Благодаря большому размеру образцов цвета их легко различать

На веб-сайтах ссылки, которые пользователь уже посетил, часто окрашиваются цветом, отличным от цвета остальных ссылок. Однако на некоторых сайтах цвета «посещенных» и «не посещенных» ссылок слишком похожи. Например, этой проблемой страдает веб-сайт Федерального резервного банка Миннеаполиса (рис. 5.8). Кроме того, оба цвета представляют собой оттенки синего, то есть цветового диапазона, к которому наши глаза наименее чувствительны. Получится ли у вас найти две ссылки, на которых пользователь уже щелкал? (Ответ далее.)

Рис. 5.8. MinneapolisFed.org: цвета, в которые окрашены посещенные и не посещенные ссылки, слишком похожи[1]

Цветовая слепота

Четвертый фактор представления цвета, влияющий на принципы дизайна интерактивных систем, заключается в том, может ли человек, страдающий одним из распространенных типов цветовой слепоты (дальтонизма), различать области

[1] Ссылки на рис. 5.8, на которых пользователь уже щелкал: Housing Units Authorized и House Price Index.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

интерфейса, окрашенные в разные цвета. Наличие цветовой слепоты не означает невозможности видеть цвета. Это говорит всего лишь о том, что один из каналов вычитания цветов (см. ранее) работает не совсем правильно, вследствие чего человеку становится сложно различать определенные пары цветов. Приблизительно 8 % мужчин и чуть менее 0,5 % женщин страдают проблемами цветового восприятия[1], то есть им сложно отличать друг от друга определенные пары цветов (Wolfmaier, 1999). Наиболее распространенный тип цветовой слепоты затрагивает красный и зеленый цвета, прочие типы встречаются намного реже. На рис. 5.9 показано, какими видят некоторые цветовые пары люди с дальтонизмом типа «красный/зеленый».

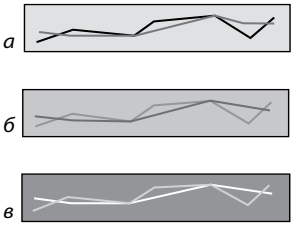


Рис. 5.9. Люди с цветовой слепотой по типу «красный/зеленый» не отличат: *а* — темно-красный от черного; *б* — синий от пурпурного; *в* — светло-зеленый от белого

Приложение для управления домашним бюджетом MoneyDance позволяет строить графическое представление расходов и выделяет цветом различные категории расходов (рис. 5.10). К сожалению, здесь используется слишком много оттенков, которые людям с цветовой слепотой будут казаться одинаковыми.

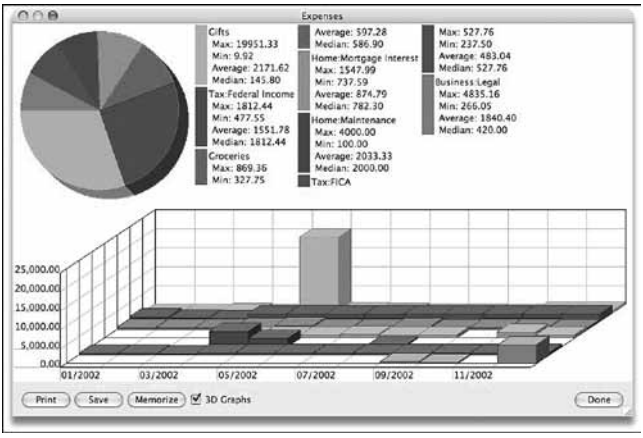


Рис. 5.10. Приложение MoneyDance использует цвета, неразличимые для некоторых людей

[1] Термин «цветовая слепота» не означает, что человек вообще не различает цветов — такие случаи крайне редки. Поэтому правильнее использовать термин «проблема цветового восприятия» или подобный.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Внешние факторы, влияющие на способность различать цвета

На способность человека различать цвета влияют также факторы внешней среды, например:

- **вариации цветных дисплеев.** Компьютерные дисплеи по-разному отображают цвета в зависимости от используемых технологий, драйверов и настроек отображения цветов. Даже на мониторах одной модели с одинаковыми настройками цвета могут немного различаться. Объект, кажущийся желтым на одном дисплее, может выглядеть бежевым на другом. Цвета, заметно различающиеся на одном устройстве, могут казаться одинаковыми на другом;
- **дисплеи, работающие с оттенками серого.** Несмотря на то что сегодня используются в основном цветные дисплеи, существуют устройства, в частности карманные компьютеры, с дисплеями, способными отображать лишь оттенки серого. Области разных цветов могут на монохромном дисплее казаться совершенно одинаковыми, что демонстрирует рис. 5.10;
- **угол обзора.** На некоторых компьютерных дисплеях, например жидкокристаллических (ЖК) мониторах, картинка выглядит намного лучше, если смотреть на нее прямо, а не под углом. Если взглянуть на ЖК-дисплей под углом, то цвета — и различия между цветами — сильно поменяются;
- **окружающее освещение.** Сильный свет, падающий на дисплей, делает цвета бледнее, более того, он способен размывать светлые и темные области, превращая цветные дисплеи в монохромные. Вы наверняка наблюдали этот эффект, если пробовали воспользоваться уличным банкоматом в яркий солнечный день. В офисных помещениях скрадывать различия цветов могут противослепящие экраны и венецианские жалюзи.

Дизайнеры программного обеспечения чаще всего не в силах управлять перечисленными четырьмя внешними факторами. Следовательно, им стоит помнить, что у них нет полного контроля над тем, какими цвета интерфейса покажутся целевой аудитории. Цвета, легко различимые на экранах дизайнеров в производственной среде при нормальных условиях искусственного освещения, могут быть слишком похожими, если смотреть на них в другой среде при других условиях использования программного обеспечения.

Рекомендации по использованию цвета

В интерактивных программных системах, в которых цвет крайне важен для передачи информации, рекомендуется применять следующие правила, которые помогут сделать так, чтобы пользователи программного обеспечения верно воспринимали информацию:

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

1. **Дифференцируйте цвета не только по тону, но также по насыщению и яркости.** Избегайте слабо различающихся цветов. Удостоверьтесь, что цвета хорошо контрастируют (не забывая, однако, о правиле № 5). Один из способов проверить, достаточно ли хорошо различаются цвета, — просмотреть дизайн в монохромном режиме. Если вам не удается отличить один цвет от другого, когда они отображаются в оттенках серого, значит, они слишком похожи.

2. **Используйте выразительные цвета.** Вспомните: наша визуальная система комбинирует сигналы, получаемые от колбочек сетчатки, для создания трех цветооппонентных каналов: красно-зеленого, желто-синего и черно-белого (сигнал яркости). Лучше всего люди различают цвета, которые создают сильный сигнал (положительный или отрицательный) на одном из трех каналов восприятия цвета и нейтральные сигналы на двух других. Неудивительно, что это красный, зеленый, желтый, синий, черный и белый цвета. Все остальные цвета создают сигналы сразу на двух или трех каналах, поэтому визуальная система не может отличить их от остальных цветов так же быстро и просто, как это происходит с шестью самыми выразительными цветами (Ware, 2008).

3. **Избегайте сочетаний цветов, выглядящих одинаковыми для людей с цветовой слепотой.** Например, речь о таких парах, как темно-красный и черный, темно-красный и темно-зеленый, синий и фиолетовый, светло-зеленый и белый. Не используйте темно-красный, темно-синий и темно-фиолетовый с любыми другими темными цветами. Лучше сочетать их со светло-желтыми и светло-зелеными оттенками. Используйте инструмент Vischeck.com для проверки веб-страниц и изображений — он поможет вам понять, как их видят люди с различными нарушениями цветового зрения.

4. **Используйте цвет в сочетании с другими подсказками.** Не полагайтесь исключительно на цвет. Если вы выделяете какую-то информацию цветом, то предусмотрите также другой вариант подсветки или выделения. Например, в приложении Apple iPhoto «умные» фотоальбомы отличаются от обычных фотоальбомов не только цветом, но и дополнительным символом (рис. 5.11).

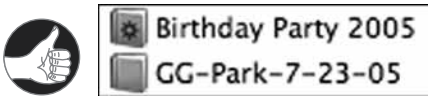


Рис. 5.11. В Apple iPhoto различать два типа фотоальбомов помогают как цвета значков, так и дополнительный символ

5. **Разделяйте сильные противоположные цвета.** Когда противоположные цвета находятся бок о бок или один поверх другого, это вызывает неприятное мерцающее ощущение, которого следует избегать (рис. 5.12).

Как упоминалось ранее, на веб-сайте ITN.net для обозначения текущего шага покупателя в процессе бронирования билета использовалась бледно-желтая под-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

светка (см. рис. 5.5), которую не все пользователи могли заметить. Простой способ усилить выделение — использовать более жирный шрифт для номера и названия шага и увеличить насыщенность желтого фонового цвета для цифры 1 (рис. 5.13, *а*). Однако ITN.net решила перейти на совершенно новый дизайн, в котором выделение цветом сочетается с изменением формы фонового рисунка (рис. 5.13, *б*).

Рис. 5.12. Противоположные цвета конфликтуют, когда находятся вплотную друг к другу

Рис. 5.13. Веб-сайт ITN.net: *а* — вариант выделения; *б* — текущий шаг выделяется двумя способами — цветом и формой

На диаграмме Федерального резервного банка используется белый цвет в сочетании с оттенками зеленого (рис. 5.14). Это пример хорошо продуманной диаграммы. Ее сможет понять любой зрячий человек, даже на монохромном дисплее.

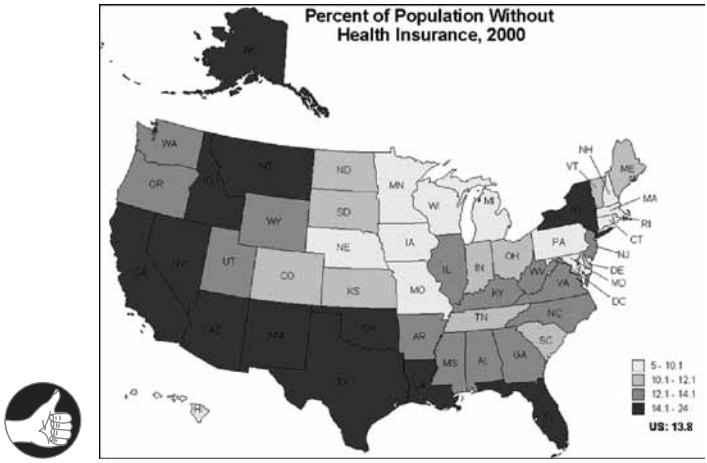


Рис. 5.14. Веб-сайт MinneapolisFed.org: на диаграмме используются сочетания цветов, различимые для всех зрячих людей и на любом дисплее

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 6

Периферическое зрение плохо работает

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В предыдущей главе говорилось о том, что способ распознавания и обработки цвета человеческой визуальной системой отличается от используемого в цифровых камерах. В том, что касается разрешения картинки, наша визуальная система также совсем не похожа на камеру. На фотодатчике цифровой камеры светочувствительные элементы выстроены в плотную матрицу и расположены равномерно, поэтому пространственное разрешение остается постоянным в любой точке кадра. В человеческой визуальной системе все совсем не так.

В этой главе я расскажу, почему мы:

- чаще всего не замечаем неподвижные элементы приглушенных цветов, расположенные на периферии зрительного поля;
- хорошо замечаем движение на периферии.

Разрешение в ямке по сравнению с разрешением на периферии поля зрения

Пространственное разрешение человеческого поля зрения значительно уменьшается от центра к краям. Сетчатка каждого глаза содержит около 6 млн колбочек. В центре поля зрения — в небольшой области под названием *ямка* — они располагаются намного теснее, чем у краев сетчатки (рис. 6.1). Площадь ямки составляет лишь около 1 % площади сетчатки, однако для обработки данных, поступающих от нее, задействуется около 50 % зрительной зоны коры головного мозга. Помимо этого фовеальные колбочки (находящиеся в ямке) соединяются с ганглиальными нервными клетками, ответственными за самое начало процесса обработки и передачи визуальных данных, в отношении 1:1, в то время как светочувствительные клетки (палочки и колбочки), расположенные на остальной площади сетчатки, соединяются по *несколько* с одной ганглиальной клеткой. Говоря техническим языком, информация, получаемая с периферии поля зрения, сжимается (с потерей данных) перед отправкой в мозг, а информация от ямки — нет. Вот почему у нас разрешение в центре поля зрения *намного* выше, чем по краям (Lindsay, Norman, 1972; Waloszek, 2005).

Для того чтобы представить, насколько мала ямка по сравнению со всем полем зрения, вытяните руку вперед и посмотрите на свой большой палец. В таком положении размер ногтя большого пальца приблизительно соответствует размеру ямки (Ware, 2008). Пока ваше зрение сфокусировано на ногте большого пальца, все остальное содержимое поля зрения находится за пределами ямки сетчатки.

У людей с нормальным зрением ямка обладает очень высоким разрешением: эта область воспринимает до нескольких тысяч точек — у большинства современных карманных цифровых камер разрешение намного ниже. Сразу же за границей ямки разрешение падает до нескольких десятков точек на дюйм, если рассматривать картинку на расстоянии вытянутой руки. По краям поля зрения размер «пикселов» нашей визуальной системы достигает размера дыни (или человеческой головы) — опять же речь о картинке на расстоянии вытянутой руки (рис. 6.2, *а* и *б*).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

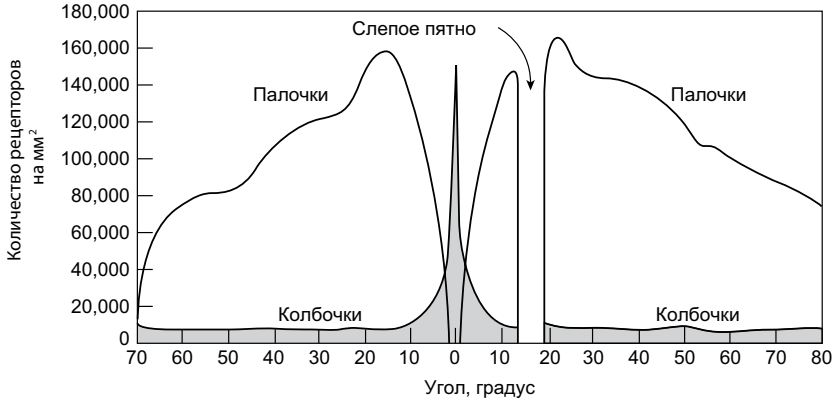


Рис. 6.1. Распределение светочувствительных клеток (палочек и колбочек) по сетчатке (Lindsay, Norman, 1972)

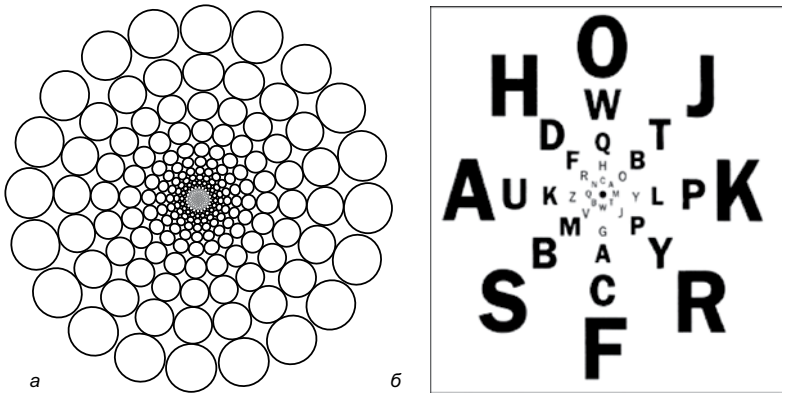


Рис. 6.2. Разрешение нашего поля зрения очень велико в центре и существенно меньше по краям (*а*); изображение из Vision Research, том 14 (1974). Elsevier (*б*)

Если разрешение периферического зрения настолько мало, то почему же мы не смотрим на мир туннельным зрением, то есть нам не кажется, что все, за исключением объекта, на который мы смотрим в данный момент, находится вне фокуса? Мы видим окружающие предметы ясно и четко. Эта иллюзия возникает благодаря тому, что наши глаза постоянно быстро и резко двигаются (примерно три перемещения в секунду). Мы не осознаем и не замечаем этого движения, но оно происходит для того, чтобы ямка сетчатки могла сфокусироваться на определенных объектах обстановки. Всю остальную картинку мозг дорисовывает в грубой импрессионистской манере, основываясь на наших знаниях и ожиданиях[1]. У мозга нет необходимости

[1] Мозг также заполняет перцепционные бреши, возникающие во время резкого движения глаз, когда зрение частично заблокировано или ограничено.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

сохранять ментальную модель окружающей среды в высоком разрешении, потому что он может приказать глазам в любой момент «взять пробу» или «перепроверить» детали обстановки (Clark, 1998).

Например, по мере того как вы читаете эту страницу, ваши глаза стремительно движутся туда-сюда, сканируя и читая текст. Независимо от того, на какой точке листа сфокусировано зрение, в мозге поддерживается впечатление наличия перед глазами целой страницы текста, и это, разумеется, отражает истинное положение вещей. А теперь представьте, что вы просматриваете эту страницу на экране компьютера и компьютер, отслеживая движения ваших глаз, знает, на каком именно участке сфокусировано зрение. Представьте, что когда вы смотрите на определенный участок, там и только там, на небольшой площади, соответствующей ямке сетчатки, отображается правильный текст. Весь остальной текст на странице заменяется случайным бессмысленным набором букв. Когда фокус переносится с одного участка страницы на другой, компьютер, отслеживая эти движения, быстро обновляет зону фокусировки, подставляя туда осмысленный текст, а предыдущую зону фокусировки заполняет текстовым шумом. Поразительно, но эксперименты доказали, что люди *не замечают* этого: они не просто продолжают спокойно читать текст, но пребывают в уверенности, что вся страница содержит только правильный осмысленный текст (Clark, 1998).

К описанному эксперименту имеет отношение тот факт, что участок сетчатки, соответствующий центру поля зрения, — ямка и небольшая область вокруг нее — это единственный участок, умеющий читать. Остальная площадь сетчатки читать не умеет. По сути, это означает, что нейронные сети, начинающиеся в ямке и проходящие по оптическим нервам по направлению к зрительной зоне, а затем и к другим областям мозга, «научились» читать путем длительных тренировок, однако нейронные сети, начинающиеся в других клетках сетчатки, такого обучения не получили и не научились читать. Весь читаемый нами текст поступает в визуальную систему после изучения центральной областью сетчатки, а это означает, что для чтения требуется выполнить множество движений глазами. Разумеется, основываясь на полученной ранее информации и на знании мира, мозг иногда в состоянии предугадать текст, еще не прочитанный ямкой (или хотя бы его смысл). Но тогда мы попросту пропускаем его, а это уже нельзя назвать чтением.

Тот факт, что колбочки очень тесно располагаются как внутри, так и вблизи ямки, а на периферии сетчатки расстояние между колбочками значительно увеличивается, влияет не только на пространственное, но и на цветовое разрешение воспринимаемой картинки. Мы куда лучше различаем цвета в центре поля зрения, чем по краям.

Есть еще один интересный факт, касающийся нашего поля зрения, — существование слепого пятна, то есть зоны, в которой мы не видим вообще ничего. Слепое пятно соответствует зоне сетчатки, где оптический нерв и кровеносные сосуды выходят из тыльной части глаза (см. рис. 6.1). В этой зоне нет палочек и колбочек, поэтому, когда изображение объекта в визуальном поле попадает в данную зону, мы

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

его не видим. Обычно мы не замечаем наличие слепого пятна, так как наш мозг заполняет его содержимым из окружающей картинки — так дизайнер в приложении Photoshop закрашивает дефект на фотографии путем копирования близлежащих фоновых пикселов.

Иногда люди замечают слепое пятно, рассматривая звезды. Пока вы вглядываетесь в одну звезду, другая, расположенная поблизости, может ненадолго пропасть. Однако вы снова замечаете ее, немного сместив взгляд. Понаблюдать эффект слепого пятна можно также, выполнив упражнение, проиллюстрированное рис. 6.3. Помимо этого у людей могут встречаться и другие варианты слепого пятна, являющиеся следствием изъянов сетчатки, ее повреждений, инсультов или других мозговых нарушений, затрагивающих зрительную зону коры головного мозга. Тем не менее слепая зона оптического нерва — это дефект, имеющийся у каждого человека.

Рис. 6.3. Для того чтобы «увидеть» слепое пятно, закройте левый глаз, приблизьте книгу к глазам и сфокусируйтесь правым глазом на значке +. Медленно отодвигайте книгу, не отводя взгляда от значка +. В какой-то момент значок @ исчезнет

Есть ли польза от зрительной периферии

Создается впечатление, что ямка сетчатки лучше периферии по всем показателям. Так зачем же нам вообще периферическое зрение? Какая от него польза?

Дело в том, что периферическое зрение предназначено для того, чтобы предоставлять мозгу «подсказки» в низком разрешении. Руководствуясь ими, мозг приказывает глазам двигаться, для того чтобы ямка рассмотрела все интересные и важные составляющие поля зрения. Наши глаза сканируют окружающую обстановку не случайным образом. Они двигаются так, чтобы ямка фокусировалась на важных вещах, в первую очередь (обычно) критически важных. Расплывчатые ориентиры на периферии поля зрения предоставляют данные, помогающие мозгу планировать, куда перевести взгляд и в каком порядке.

Например, когда мы рассматриваем упаковку лекарства в поисках надписи «Срок годности», расплывчатого пятна на периферии поля зрения, приблизительно напоминающего очертаниями дату, достаточно для того, чтобы вызвать движение глаз. Область ямки фокусируется на этом пятне, и мозг проверяет, действительно ли это то, что мы ищем. Если на продуктовом рынке мы хотим купить клубнику, то нечеткое красноватое пятно на краю поля зрения привлечет наши взгляд и внимание, даже если в результате оно окажется пучком редиски, а не упаковкой клубники. Если поблизости слышится звериное рычание, то наш взгляд мгновенно перенесется к расплывчатой фигуре животного, замеченной на периферии поля зрения, особенно если эта фигура движется по направлению к нам (рис. 6.4).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

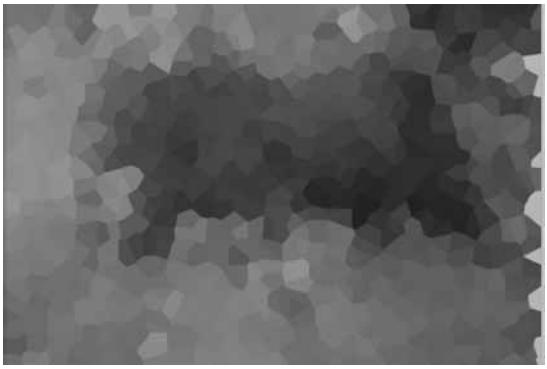


Рис. 6.4. Движущаяся фигура на краю поля зрения притягивает наш взгляд: это может быть еда, а может быть хищник, считающий едой нас

Мы подошли к еще одному преимуществу периферического зрения: оно хорошо распознает движение. Любой движущийся объект на периферии поля зрения, даже если он совершает минимальные перемещения, притягивает взгляд и, следовательно, заставляет наш мозг сфокусировать на нем ямку сетчатки. Причина данного феномена заключается в том, что наши предки — включая тех, кого еще нельзя было отнести к людям, — участвовали в естественном отборе, и для того чтобы выжить, им необходимо было уметь находить еду и избегать хищников. Вот почему, несмотря на то что мы умеет перемещать взгляд сознательно, намеренно контролируя движение глаз, некоторые механизмы управления зрением срабатывают бессознательно, непроизвольно и очень быстро.

А что происходит в случае, если мы не ожидаем увидеть ничего интересного в определенной точке периферии[1] и никакая фигура или движение не притягивают к ней наш взгляд? Вполне вероятно, что наши глаза так никогда и не сфокусируются на данной точке и мы не узнаем, что там находится.

Примеры пользовательских интерфейсов

Низкая острота периферического зрения объясняет, почему пользователи программного обеспечения и посетители веб-сайтов часто не замечают определенные сообщения об ошибках. Когда человек щелкает на кнопке или ссылке, ямка сетчатки у него сфокусирована обычно именно на этой точке экрана. Все, что находится за пределами окружности радиусом 2 см с центром в точке фокусировки (предполагается нормальное расстояние между лицом человека и компьютерным монитором), регистрируется периферическим зрением в низком разрешении. Если после щелчка сообщение об ошибке появляется в периферической зоне, совершенно неудивительно, что пользователь его не видит.

[1] О том, как ожидания искажают восприятие, рассказывается в главе 1.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Например, на веб-сайте InformaWorld.com — интерактивном инструменте Informa Healthcare — после ввода неправильного имени пользователя или пароля и щелчка на кнопке Sign In (Зарегистрироваться) сообщение об ошибке выводится очень далеко от того места, где, скорее всего, сфокусирован взгляд пользователя (рис. 6.5). Красное слово Error (Ошибка) в периферическом поле зрения пользователя может появиться в виде небольшого красноватого пятна, которое может привлечь внимание. Однако с таким же успехом красное пятно может попасть в слепую зону поля зрения пользователя и останется незамеченным.

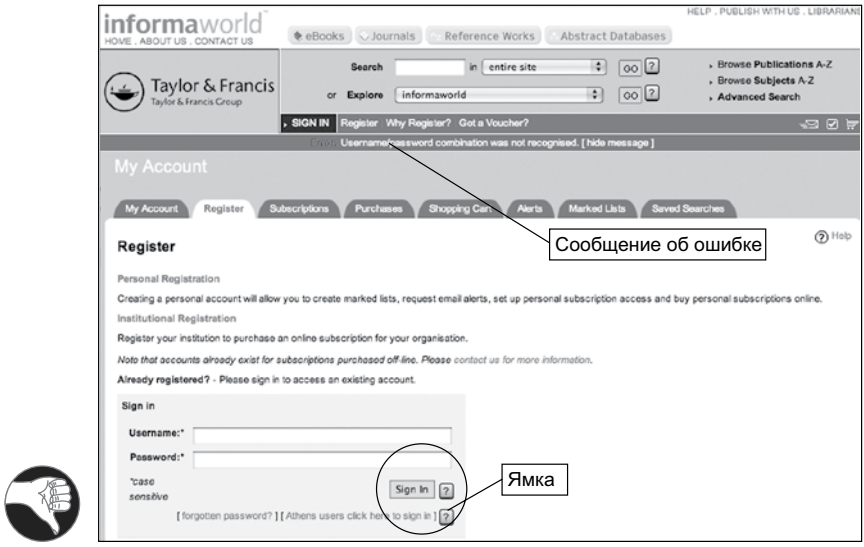


Рис. 6.5. Сообщение об ошибке при входе в систему выводится в зоне периферического зрения, где пользователь его с большой вероятностью не заметит

Рассмотрим последовательность событий с точки зрения пользователя. Пользователь вводит имя и пароль и щелкает на кнопке Sign In (Зарегистрироваться). Страница перезагружается, но появляются те же самые пустые поля. Пользователь думает: «В чем дело? Я же ввел учетные данные и щелкнул на кнопке подтверждения. Разве не так? Может, я нажал не на ту кнопку?» Пользователь снова вводит имя и пароль и еще раз щелкает на кнопке Sign In (Зарегистрироваться). После очередной перезагрузки страницы он видит все те же пустые поля. Теперь пользователь окончательно сбит с толку. Он вздыхает (или произносит пару ласковых слов), откидывается на спинку кресла и начинает задумчиво рассматривать страницу. Внезапно он замечает сообщение об ошибке: «Ага, вот! И что же, давно это сообщение здесь висит?»

Даже когда сообщение об ошибке оказывается ближе к центру поля зрения посетителя сайта, чем в описанном примере, оно может оказаться незамеченным из-за влияния других факторов. До недавнего времени веб-сайт Airborne.com сигнализировал об ошибке при входе в систему при помощи сообщения, появляющегося

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

прямо над полем Login ID (Идентификатор для входа в систему), причем сообщение выделялось красным цветом (рис. 6.6). Оно выводилось недалеко от кнопки Login (Войти), на которой, предположительно, фокусировался взгляд пользователя. Тем не менее многие пользователи не обращали внимания на это сообщение при первом появлении. Почему? Вы сможете придумать объяснение?

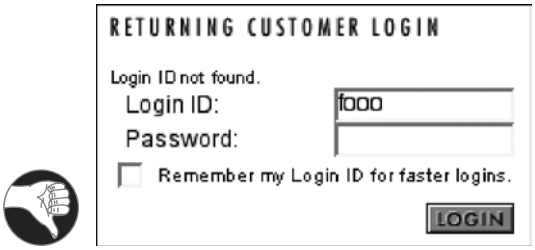


Рис. 6.6. Многие пользователи не замечали это сообщение об ошибке входа в систему, несмотря на то что оно находится близко к кнопке Login (Войти)

Одна из причин заключается в том, что, хотя сообщение об ошибке располагается довольно близко к точке, на которой пользователь фокусируется, щелкая на кнопке Login (Войти), оно все же попадает в зону периферического зрения, а не ямки. Ямка очень маленькая, около 1–2 см на компьютерном экране при условии, что лицо пользователя находится на нормальном расстоянии от монитора.

Вторая причина состоит в том, что данное сообщение об ошибке — не единственная надпись красным шрифтом наверху страницы. Ее заголовок тоже выделяется красным цветом. Разрешение периферического зрения невелико, поэтому, когда сообщение об ошибке появляется, визуальная система пользователя может вовсе не зарегистрировать никаких изменений: наверху было что-то красное, и там все так же что-то красное (рис. 6.7).

Рис. 6.7. Имитация поля зрения пользователя, когда его взгляд (ямка) сфокусирован на кнопке Login (Войти)

Если бы заголовок страницы был окрашен в черный или любой другой цвет, отличный от красного, то красное сообщение об ошибке было бы куда проще заметить даже в зоне периферического зрения.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Как сделать сообщение заметным

Существует несколько хорошо известных методов создания заметных сообщений об ошибках, основывающихся на следующих рекомендациях:

- **Помещайте сообщения туда, куда смотрит пользователь:** во время взаимодействия с графическим интерфейсом пользователя люди фокусируются в предсказуемых точках. В западных обществах формы и панели управления обычно просматривают в направлении от верхнего левого угла к нижнему правому. Перемещая указатель мыши, люди смотрят либо туда, где он находится в данный момент, либо туда, куда он движется. Щелкая на кнопке или ссылке, они обычно смотрят прямо на этот элемент управления как в момент действия, так и, по крайней мере, в течение нескольких мгновений после. Исходя из предсказуемости поведения пользователей, дизайнеры могут выводить сообщения об ошибках там, где они с наибольшей вероятностью будут замечены.
- **Обозначайте ошибку**: любым способом заметно помечайте ошибку, для того чтобы не оставалось сомнений, что что-то не в порядке. Например, проще всего сделать это, поместив сообщение об ошибке рядом с элементом, в котором ошибка произошла. Разумеется, так делать не следует, если известно, что пользователь, скорее всего, будет смотреть совсем в другую точку.
- **Используйте символ, сигнализирующий об ошибке:** для того чтобы сделать ошибку или сообщение об ошибке более заметными, поместите рядом узнаваемый символ предупреждения, например , , или .
- **Используйте красный цвет только для обозначения ошибок**: условно считается, что в интерактивных компьютерных системах красный цвет подразумевает *предупреждение*, *проблему*, *ошибку*, *опасность* и т. п. Выделение красным цветом любой другой информации на экране компьютера влечет за собой неправильную трактовку и непонимание. В то же время вы можете разрабатывать веб-сайт для Стэнфордского университета, у которого красный — фирменный цвет. Или же вы создаете продукт для рынка Китая, где красный цвет считается благоприятным, положительным символом. Что же делать? Используйте для обозначения ошибок другой цвет, дополняйте сообщения символами ошибок или применяйте более сильные методы (о них рассказывается в дальнейшем).

В улучшенной версии сообщения об ошибке входа в систему на сайте InformaWorld учтены сразу несколько из перечисленных рекомендаций (рис. 6.8).

На веб-сайте America Online форма регистрации новой учетной записи электронной почты хорошо отвечает приведенным ранее рекомендациям (рис. 6.9). Содержащие ошибку поля данных помечаются красными символами. Сообщения об ошибках выводятся красным цветом вплотную к элементам, где ошибки обнаруживаются. Кроме того, большинство сообщений об ошибках отображаются сразу же после того, как пользователь вводит проблемные данные, то есть в то время, когда чело-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

век еще смотрит на эту часть формы и не успел переключить внимание на кнопку подтверждения и отправки формы. Вряд ли пользователи AOL смогут упустить из виду такие заметные сообщения.

Рис. 6.8. Это сообщение об ошибке входа в систему гораздо заметнее и отображается рядом с точкой, на которой сфокусирован взгляд пользователя

Рис. 6.9. На странице регистрации нового пользователя AOL.com сообщения об ошибках хорошо заметны и выводятся рядом с проблемными элементами

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Осторожно! Тяжелая артиллерия для создания заметных сообщений

Если распространенных и привычных способов сделать сообщения более заметными для пользователей оказывается недостаточно, то дизайнеры пользовательских интерфейсов могут прибегнуть к помощи трех более сильных методов. Однако, несмотря на всю их эффективность, эти методы обладают заметными негативными свойствами, поэтому использовать их следует с большой осторожностью и лишь в крайней необходимости.

Всплывающее сообщение в диалоговом окне

Когда сообщение об ошибке выводится в диалоговом окне, оно всплывает прямо перед глазами пользователя и его невозможно не заметить. Такое диалоговое окно прерывает работу человека и требует, чтобы на него немедленно обратили внимание. Это хорошо для сообщений, сигнализирующих о критических условиях. Но пользователей раздражает, когда к данному подходу прибегают для вывода сообщений о незначительных событиях, например для подтверждения запрошенного действия.

Раздражающее воздействие всплывающих сообщений приводит нас к понятию модальности. Пользователь может игнорировать *немодальное диалоговое окно* и продолжать обычную работу. Однако *модальное окно приложения* блокирует любую дальнейшую деятельность в приложении, выдавшем ошибку, следовательно, у пользователя остается только возможность взаимодействовать с операционной системой или другими программами. *Системное модальное диалоговое окно* блокирует любые действия пользователя до тех пор, пока он не закроет его доступным способом.

Модальные окна приложения следует использовать лишь в самых серьезных случаях, то есть в ситуациях, когда без немедленной реакции пользователя данные приложения могут быть утеряны. Применять системные модальные диалоговые окна допустимо крайне редко — фактически, только если в системе произойдет авария, на устранение которой потребуется несколько часов, или люди могут погибнуть, если пользователь проигнорирует сообщение об ошибке.

В Сети существует еще одна причина, почему всплывающих диалоговых окон с сообщениями стоит избегать, — многие пользователи настраивают в своих браузерах блокировку всех всплывающих окон. Если ваш сайт полагается на всплывающие сообщения об ошибках, существует риск, что некоторые пользователи никогда их видеть не будут.

Звуковой сигнал

Когда компьютер издает звуковой сигнал, пользователь понимает, что произошло нечто, требующее его внимания. Глаза человека непроизвольно начинают сканировать экран в поисках других признаков события, послужившего источником звука. В таком

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

случае пользователь может заметить сообщение об ошибке, даже если оно появилось в другом месте, не там, куда он только что смотрел, — например, в стандартном поле для предупреждений. Таким образом, звук может оказаться весьма полезным. Однако представьте группу людей, работающих в большом открытом офисном помещении или учебном классе. Все они используют приложение, сообщающее об ошибках и предупреждениях звуковым сигналом. Шумовая обстановка в таком помещении будет, мягко говоря, раздражающей. Еще хуже то, что люди не всегда будут понимать, чей компьютер издал предупреждающий звук — собственный или соседа.

В шумных помещениях, например на фабриках или в серверных комнатах, звуковые сигналы могут остаться незамеченными из-за высокого общего уровня шума. Наконец, многие пользователи уменьшают или полностью выключают звук на своих компьютерах. Следовательно, сигнализировать об ошибках и других критических условиях с помощью звука следует только в особых, допускающих такой подход ситуациях.

Короткое мерцание или колебание

Как уже упоминалось, наше периферическое зрение очень хорошо распознает движение. Мы непроизвольно переводим взгляд на источник движения на периферии, для того чтобы ямка сетчатки сфокусировалась на нем и позволила мозгу его изучить. Дизайнеры пользовательских интерфейсов могут воспользоваться этой особенностью: люди с большей вероятностью заметят сообщение об ошибке, если оно немного подвигается или несколько раз вспыхнет и погаснет. Для того чтобы привлечь внимание, не требуется значительных перемещений — даже крохотного движения достаточно для того, чтобы пользователь взглянул в его сторону. Миллионы лет эволюции не прошли зря.

Однако, как и всплывающие диалоговые окна и звуковые сигналы, движение следует использовать с большой аккуратностью. Большинству опытных пользователей колеблющиеся, мерцающие объекты на экране кажутся раздражающими. Многие из нас научились игнорировать подобные мерцающие элементы, так как многие из них оказываются ненужной нам рекламой. И наоборот, у некоторых компьютерных пользователей встречаются расстройства внимания, из-за которых им становится крайне сложно игнорировать мигающие и ерзающие объекты.

Таким образом, если вы все же решили реализовать мерцание или колебание, это движение должно быть кратковременным: от четверти до половины секунды, не дольше. В противном случае оно быстро превратится из объекта для рефлекторного привлечения внимания в осознанный раздражитель.

Расчетливо применяйте эти методы

Всю эту тяжелую артиллерию следует применять крайне умеренно — исключительно для критических сообщений. Когда всплывающие окна, звук, движение и мерцание слишком часто используются для привлечения внимания пользователя, возни-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

кает психологический феномен под названием «привыкание». Наш мозг обращает все меньше и меньше внимания на раздражители, если они появляются слишком часто. Помните старую басню про мальчика, который то и дело кричал: «Волки!»? Жители деревни в итоге перестали обращать внимание на его крики, и когда волк действительно появился, призывы о помощи остались без ответа. Чрезмерное применение сильных методов привлечения внимания может привести к тому, что важные сообщения по привычке будут блокироваться сознанием пользователя.

На веб-сайте REI.com можно увидеть пример всплывающего диалогового окна, содержащего сообщение об ошибке. Это окно появляется, когда в процессе регистрации новой учетной записи покупателя человек не заполняет обязательное поле на форме (рис. 6.10). Можно ли назвать это допустимым сценарием использования всплывающего диалогового окна? Веб-сайт AOL.com (см. рис. 6.9) демонстрирует, что сообщения о недостающих данных можно успешно отображать и без лишних диалоговых окон, так что подход разработчиков REI.com кажется несколько тяжеловесным.

Рис. 6.10. Всплывающее диалоговое окно на веб-сайте REI сообщает о незаполненных обязательных полях. Такое предупреждение не упустишь из виду, но это все же чрезмерно

Примеры более обоснованного использования диалоговых окон с сообщениями об ошибке можно найти в приложениях Microsoft Excel (рис. 6.11, *а*) и Adobe InDesign (рис. 6.11, *б*). В обоих случаях речь идет о возможной утере данных.

В компьютерных играх для оповещения о событиях и условиях часто применяется звуковой сигнал. В играх он не раздражает, так как ожидаем. Звук повсеместно используют в играх, и он нормально воспринимается даже в игровых клубах, где десятки машин одновременно стреляют, ревут, жужжат, клацают и гудят. (Раздражает это разве что родителей, которым приходится заходить в клубы, для того чтобы забрать детей, и терпеть все это скрежетание и громыхание, но ведь игры создаются не для родителей.)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

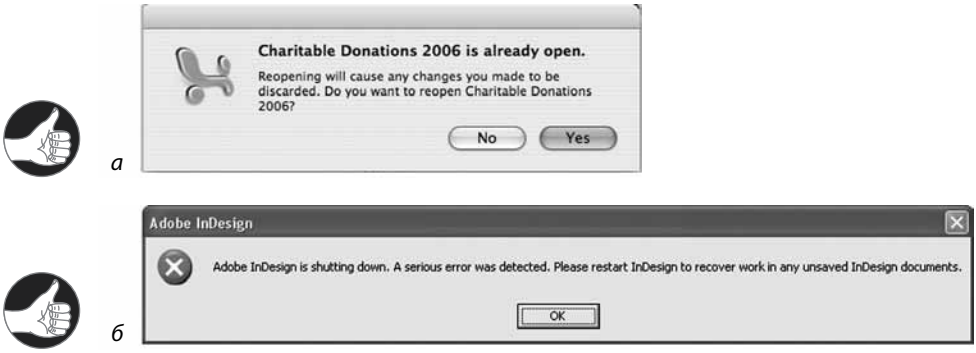


Рис. 6.11. Уместное использование всплывающих окон с сообщениями об ошибках: *а* — Microsoft Excel; *б* — Adobe InDesign

Наиболее популярный вариант мерцания в компьютерных пользовательских интерфейсах (за исключением рекламы) реализуется в меню приложений. Когда пользователь выбирает в меню определенное действие, например Правка ▸ Копировать, перед закрытием меню этот пункт на мгновение подсвечивается, подтверждая, что система получила команду и пользователь не промахнулся мимо нужного элемента. Такой сценарий встречается очень часто. Однако мерцание здесь настолько короткое, что большинство пользователей даже не замечают его, — в то же время без моментальной подсветки пунктов меню мы не были бы уверены, что выбрали нужную команду.

Что касается движения, привлекающего внимание пользователя, можно привести такой пример: представьте, что сообщение об ошибке на улучшенном экране входа в систему InformaWorld (см. рис. 6.8) появилось, в течение четверти секунды сдвинулось на один пиксел вверх, вниз, влево и вправо (рис. 6.12), а затем остановилось, превратившись в простое статическое изображение. Легкое дрожание гарантированно притянет к себе взгляд пользователя, но не покажется навязчивым или раздражающим. В конце концов, движущийся объект на краю поля зрения может оказаться крадущимся леопардом.

Рис. 6.12. Сообщение об ошибке может немного поерзать после появления, для того чтобы на нем сфокусировалась ямка сетчатки пользователя

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 7

Внимание ограниченно, а память неидеальна

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

У человеческой визуальной системы есть свои сильные и слабые стороны, и точно так же дело обстоит с человеческой памятью. В этой главе мы обсудим преимущества и недостатки нашей памяти, для того чтобы научиться проектировать интерактивные системы, помогающие человеческой памяти и дополняющие ее возможности, а не нагружающие и сбивающие с толку. Начнем с обзора: как же работает память?

Краткосрочная и долгосрочная память

Уже очень давно психологи установили, что у людей есть два вида памяти: *краткосрочная* и *долгосрочная*. Под краткосрочной памятью подразумевается, что информация сохраняется в мозге на короткий отрезок времени — от доли секунды до нескольких секунд, возможно, одну минуту. Долгосрочная память предназначена для хранения информации за куда более длительные периоды: минуты, часы, дни, годы, всю жизнь.

Выглядит весьма привлекательной идея разделять эти два вида памяти и рассматривать их как два отдельных хранилища. И действительно, некоторые теории памяти основываются на таком представлении. В конце концов, в цифровом компьютере хранилища краткосрочной памяти (регистры данных процессора) никак физически не связаны с долгосрочной памятью (ОЗУ, жесткий диск, флеш-память, компакт-диск и т. д.). Кроме того, известно, что повреждение определенных частей человеческого мозга приводит к утрате краткосрочной памяти, но не влияет на долгосрочную, и наоборот. Наконец, скорость, с которой информация или планы могут исчезать из зоны немедленного внимания, несравнима с кажущимся постоянством воспоминаний о важных событиях жизни, лицах значимых людей, деятельности, которую мы практиковали в прошлом, и вещах, о которых мы узнали в детстве. Эти феномены приводят многих исследователей к мысли о том, что краткосрочная память представляет собой отдельное хранилище в нашем мозге, где информация ненадолго задерживается после поступления через средства восприятия (например, органы зрения или слуха) или после извлечения из долгосрочной памяти (рис. 7.1).

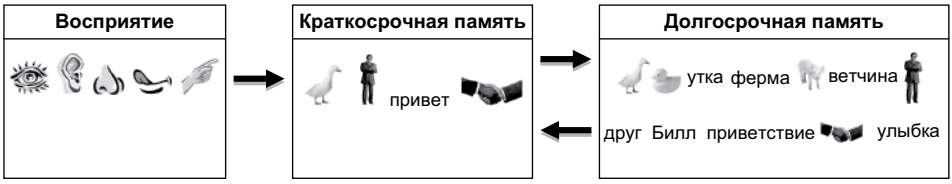


Рис. 7.1. Традиционное (устаревшее) представление о краткосрочной и долгосрочной памяти

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Современный взгляд на память

Недавние исследования памяти и функций мозга показали, что краткосрочная и долгосрочная память — это функции единой системы памяти, которая, к слову, намного теснее связана с восприятием, чем полагалось ранее (Jonides et al., 2008).

Долгосрочная память

Мы воспринимаем окружающий мир через визуальную, слуховую, обонятельную, вкусовую и тактильную сенсорные системы. В ответ на наши ощущения организм запускает ответные реакции, начинающиеся в областях мозга, предназначенных для обработки каждого из чувств (то есть в зрительной или, например, слуховой зоне коры головного мозга). Затем реакции распространяются в другие зоны мозга, *не отвечающие* за какую-то одну конкретную сенсорную модальность. Зоны мозга, ответственные за определенную сенсорную модальность, распознают только простые характеристики данных, такие как *граница между тенью и светом*, *диагональная линия*, *высокий тон звука*, *кислый вкус*, *красный цвет* или *движение вправо*. Зоны мозга, находящиеся на следующих шагах процесса обработки информации, объединяют низкоуровневые характеристики и из их сочетаний выявляют высокоуровневые, такие как *животное*, *дядя Петя*, *минорная тональность*, *угроза* или слово «*утка*».

То, какой набор нейронов будет участвовать в распознавании, в значительной степени зависит от характеристик и контекста сцены, причем контекст не менее важен, чем характеристики. Например, лай собаки запускает выполнение совершенно разных шаблонов нейронной активности в вашем мозге, когда вы проходите мимо соседского двора и когда находитесь в собственной машине с закрытыми дверцами. Чем более похожи два перцепционных раздражителя, то есть чем больше у них общих характеристик и контекстуальных элементов, тем сильнее частичное совпадение наборов нейронов, активируемых в ответ на эти раздражители.

Первоначальная сила впечатления зависит от того, насколько оно усиливается или смягчается другими видами активности мозга. Все впечатления оставляют свой след, но некоторые настолько слабы, что можно считать, будто они никогда и не были зарегистрированы: шаблон активирован однажды, но после этого ни разу не срабатывал.

Формирование памяти происходит за счет длительных и даже перманентных изменений в нейронах, участвующих в шаблоне нейронной активности. Такие изменения упрощают повторную активацию шаблона в будущем[1]. При некоторых подобных изменениях в зоны, окружающие нейроны, выбрасываются определенные химиче-

[1] Доказано, что долговременные нейронные изменения, связанные с обучением, происходят в основном во время сна. Таким образом, разделяя периоды обучения периодами сна, вы лучше усваиваете материал (Stafford, Webb, 2005).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ские вещества, которые меняют их чувствительность к раздражителям на довольно длительные периоды, до тех пор, пока химические вещества не будут выведены или нейтрализованы другими веществами. Более продолжительные изменения включают рост нейронов и формирование новых связей с другими нейронами.

Активация памяти — это запуск того же шаблона нейронной активности, который работал при формировании и записи данного впечатления. Каким-то образом мозг различает *первоначальную* активацию шаблона и все *последующие* — возможно, оценивая относительную легкость повторного запуска шаблона. Новые впечатления, очень похожие на исходные, повторно запускают те же шаблоны нейронной активности, и в результате, если повторно активированное впечатление достигает сознания, происходит *узнавание*. В отсутствие похожего впечатления повторно активировать шаблон нейронной активности могут также раздражители, возникшие из-за процессов в других частях мозга, и тогда, если повторно активированное впечатление достигает сознания, происходит *вспоминание*.

Чем чаще повторно активируется нейронный шаблон памяти, тем «сильнее» он становится, то есть его повторная активация проходит все проще и быстрее — это, в свою очередь, означает, что проще и быстрее становится узнавать или вспоминать впечатление, с которым он ассоциирован. Нейронные шаблоны памяти могут усиливаться или ослабляться возбуждающими или сдерживающими сигналами из других частей мозга.

Отдельные воспоминания не хранятся в каких-то конкретных точках мозга. Шаблон нейронной активности, составляющий воспоминание, охватывает целую сеть нейронов, расположенных на довольно большой площади. Шаблоны активности, соответствующие разным воспоминаниям, перекрывают друг друга в зависимости от того, какими общими характеристиками обладают эти воспоминания. Удаление, повреждение или подавление нейронов в определенной части мозга обычно не стирает полностью воспоминания, зависящие от этих нейронов, а просто снижает уровень детализации или точности вследствие исчезновения тех или иных характеристик[1]. Однако некоторые участки шаблонов нейронной активности могут представлять критически важные проводящие пути, поэтому их удаление, повреждение и подавление может препятствовать активации большей части соответствующих шаблонов — фактически, это равносильно разрушению соответствующих воспоминаний.

Краткосрочная память

Процессы, о которых говорилось ранее, относятся к долгосрочной памяти. А где в этой картине мира место краткосрочной памяти? Ответ подсказывает слово «сознание».

---

[1] Нечто похожее происходит при удалении фрагментов голографического изображения: снижается общее разрешение изображения, а не удаляются куски картинки, как если бы мы проделали это с обычной фотографией.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Краткосрочная память — это не хранилище, то есть речь не идет о каком-то *месте*, куда воспоминания и впечатления *отправляются* для дальнейшей обработки. Точнее, это не какой-то временный склад информации, только что поступившей от системы чувств или извлеченной из долгосрочной памяти. Краткосрочной памятью называется сочетание феноменов, обусловленных восприятием и вниманием.

У каждого из наших чувств есть собственная очень короткая краткосрочная «память», представляющая собой остаточную нейронную активность. Такая активность существует еще несколько мгновений после исчезновения перцепционного раздражителя — точно так же мы еще слышим звон колокольчика, когда его язычок уже не касается стенок. До тех пор пока эти остаточные впечатления не погаснут, они могут использоваться в качестве входных данных для механизмов внимания мозга, которые собирают информацию от различных перцепционных систем и фокусируют внимание на определенных составляющих этой информации. Все вместе остаточные впечатления от различных органов чувств составляют небольшую часть краткосрочной памяти.

Также возможным источником входных данных для механизмов внимания является долгосрочная память, повторно активированная путем узнавания или воспоминания. Как говорилось ранее, каждое воспоминание соответствует определенному шаблону нейронной активности, охватывающему весь мозг. Будучи активированным, шаблон воспоминания становится одним из кандидатов на наше внимание.

У человеческого мозга несколько механизмов внимания, работающих сознательно или непроизвольно. Они фокусируют сознание на очень небольшом поднаборе впечатлений и активированных долгосрочных воспоминаний, игнорируя все остальное. Этот крошечный поднабор всей предлагаемой нашими перцепционными системами и долгосрочной памятью информации, на котором мы сосредоточены *прямо сейчас*, и представляет собой основной компонент *краткосрочной памяти*; ученые, занимающиеся когнитивными науками, называют его *рабочей памятью*.

Не следует считать рабочую память временным буфером, куда впечатления и воспоминания переносятся для того, чтобы мозг мог обработать их. В действительности это комбинированный *фокус внимания* — активированные в настоящий момент нейронные шаблоны, на которых сосредоточено сознание. Количество элементов в краткосрочной памяти в любой момент крайне невелико и непостоянно.

А что же установленный ранее факт, что повреждение одних частей мозга приводит к проблемам с краткосрочной памятью, в то время как другие типы мозговых нарушений вызывают поражения долгосрочной памяти? Современная интерпретация этих открытий такова: в зависимости от типа повреждения мозга ухудшают или уничтожают либо способность мозга фокусировать внимание на определенных объектах и событиях, либо способность хранить долгосрочные воспоминания и обращаться к ним.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Характеристики краткосрочной памяти

Краткосрочная память, как уже говорилось, эквивалентна фокусу внимания. На чем сфокусировано внимание, то и находится в нашем сознании в конкретный момент.

Прямо сейчас вы осмысливаете последние прочитанные слова и соответствующие идеи, но вряд ли частью вашего внимания является цвет противоположной стены. Однако теперь, когда я изменил фокус внимания, вы осознаете, какого цвета эта стена, но, возможно, забыли некоторые идеи, с которыми познакомились, прочитав предыдущую страницу.

Основные характеристики краткосрочной памяти — ее непостоянство и небольшая емкость.

О малой емкости краткосрочной памяти известно довольно хорошо. Многие люди, получившие высшее образование, читали о «магическом числе семь, плюс-минус два», предложенном когнитивным психологом Джорджем Миллером (George Miller) в 1956 году в качестве предела количества несвязанных элементов, одновременно находящихся в краткосрочной памяти человека.

Представление о краткосрочной памяти, созданное Миллером, естественным образом заставляет задать несколько вопросов.

- **Какие элементы находятся в краткосрочной памяти?** Это текущие впечатления и извлеченные из долгосрочной памяти воспоминания. Цели, числа, слова, имена, звуки, изображения, ароматы — все, что может осознавать человек.
- **Почему речь идет о несвязанных элементах?** Потому что, если два элемента связаны, они соответствуют одному большому шаблону нейронной активности — одному набору характеристик, и, следовательно, это один элемент, а не два.
- **Почему упоминается поправочный фактор «плюс-минус два»?** Исследователи не в силах с безукоризненной точностью оценить объем памяти человека, а также потому, что все люди разные и запоминают разные объемы информации.

Более поздние исследования, проведенные в 1960-х и 1970-х годах, доказали, что оценка Миллера завышена. В его экспериментах некоторые элементы, которые людям необходимо было запоминать, фактически составляли «порцию», то есть были связанными. Из-за этого создавалось впечатление, что краткосрочная память содержала больше данных, чем на самом деле. После пересмотра экспериментов и устранения «порций» (то есть связанных элементов) оказалось, что емкость краткосрочной памяти не превышает четырех элементов плюс-минус один, то есть от трех до пяти элементов (Broadbent, 1975).

Последовавшие за этим эксперименты бросили тень сомнения на саму идею о том, что емкость краткосрочной памяти можно измерять в элементах или «порциях». Оказывается, в ранних испытаниях краткосрочной памяти людей просили ненадолго запомнить элементы (такие как слова или изображения), довольно сильно отличающиеся друг от друга, то есть обладающие малым числом общих характеристик.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В таких ситуациях нет необходимости запоминать все характеристики элементов, для того чтобы вспомнить их через несколько секунд, — достаточно запомнить лишь некоторые. Из-за этого создавалось впечатление, что люди вспоминали элементы как целостные объекты, и укрепилось мнение, что емкость краткосрочной памяти допустимо измерять в целых элементах.

В недавних экспериментах людей просили запоминать похожие элементы, то есть элементы, обладающие некоторыми одинаковыми характеристиками. Известно, что, для того чтобы вспомнить элемент, но не перепутать его с другими, человеку приходится запоминать больше его характеристик. Эти эксперименты обнаружили, что люди запоминали разное число деталей (то есть характеристик) разных элементов: для каких-то больше, для каких-то меньше. Чем больше они обращали внимания на определенные элементы, тем детальнее их запоминали (Bays, Husain, 2008). Это открытие подразумевает, что единицу внимания — и, следовательно, емкостный предел краткосрочной памяти — лучше оценивать в количестве характеристик элементов, а не в числе целых элементов или «порций» (Cowan, Chen, Rouder, 2004).

Еще одна важная характеристика краткосрочной памяти — ее непостоянство. Раньше когнитивные психологи говорили, что новые элементы, поступающие в краткосрочную память, выталкивают старые. Однако такое описание основывается на представлении о краткосрочной памяти как о временном *хранилище* информации. Современное представление о краткосрочной памяти как о текущем фокусе внимания упрощает описание: фокусировка внимания на новой информации уводит фокус от части данных, которые находились в центре сознания раньше.

Каким бы образом мы ни описывали этот феномен, информация легко утекает из краткосрочной памяти. Если элементы в краткосрочной памяти не сочетать между собой и не повторять (не репетировать), то высок риск того, что фокус внимания сместится с этих элементов на какие-нибудь другие. Важно отметить, что подобный риск распространяется как на цели, так и на детали объектов. Исчезновение элементов из краткосрочной памяти происходит, когда мы забываем что-то или теряем нить действия или рассуждения. Мы все периодически обнаруживаем себя в следующих ситуациях:

- идем в другую комнату за чем-нибудь, а оказавшись в комнате, не можем вспомнить, что хотели сделать или взять;
- отвечаем на телефонный звонок, а после разговора не можем вспомнить, чем занимались до звонка;
- что-то внезапно отвлекает нас от беседы, и мы не можем вспомнить, о чем говорили до этого;
- в процессе суммирования длинного списка чисел что-то отвлекает нас, и приходится начинать сначала.

Один из способов продемонстрировать ограничение объема и «срока действия» краткосрочной памяти — последовательно показать человеку две версии рисунка и спросить, отличается ли вторая версия от первой. Как ни удивительно, даже если

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

между версиями существует множество различий, люди могут их совершенно не замечать. На втором этапе этого исследования ученые задают испытуемым вопросы относительно первой картинки, для того чтобы понять, на какие ее характеристики они обратили внимание. В результате выясняется, что люди замечают изменения только тех характеристик, на которые они специально обращали внимание. Изменения в остальных характеристиках остаются незамеченными. Это называется «слепотой к изменению» (Angier, 2008).

Примечательный пример непостоянства краткосрочной памяти можно наблюдать в следующем эксперименте: экспериментаторы изображают заблудившихся туристов и с картой города в руках просят проходящих мимо местных жителей помочь им определиться с направлением. Когда местный житель фокусируется на карте туриста, стараясь найти наилучший маршрут, два рабочих (в действительности участники эксперимента) проходят между ним и «туристом», неся большую дверь. В этот момент происходит подмена: место первого «туриста» занимает другой «турист»-экспериментатор. Удивительно, но после того как рабочие с дверью уходили, более половины местных жителей продолжали помогать «туристу», не замечая, что это уже другой человек, даже когда у двух «туристов» различался цвет волос или один из них носил бороду, а второй нет (Simons, Levin, 1998). Некоторые люди не замечали подмены, даже когда второй «турист» был другого пола. Из этого следует вывод: человек фокусируется на «туристе» лишь на короткий промежуток времени, достаточный для того, чтобы определить, не представляет ли тот угрозы и действительно ли нуждается в помощи. После того как зафиксирован факт, что это турист, которому следует помочь, человек полностью фокусируется на карте и задаче поиска маршрута.

ВЛИЯНИЕ КРАТКОСРОЧНОЙ ПАМЯТИ НА ДИЗАЙН ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

Емкость и изменчивость краткосрочной памяти — это важные характеристики, которые необходимо учитывать во время проектирования интерактивных компьютерных систем. Главное, о чем стоит помнить: пользовательские интерфейсы должны помогать людям запоминать только существенную информацию между каждыми двумя моментами взаимодействия с системой. Не требуйте от человека помнить статус системы или все выполненные ранее действия. Внимание пользователей фокусируется на главной цели и на прогрессе ее достижения. Конкретные примеры вы найдете в следующих подразделах.

Режимы

Ограниченная емкость и непостоянство краткосрочной памяти — одни из причин, по которым в рекомендациях по проектированию пользовательских интерфейсов часто советуют либо вообще избегать режимов, либо обеспечивать адекватное обозначение активного режима.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ТЕСТ НА КРАТКОСРОЧНУЮ ПАМЯТЬ

Для того чтобы проверить свою краткосрочную память, возьмите ручку или карандаш и два чистых листа бумаги. Выполните следующие шаги:

1. Одним листом бумаги закройте следующую страницу (находящуюся на обороте этого листа).
2. Переверните страницу, на 3 секунды немного приспустите лист бумаги, закрывающий цветные строчки, и прочитайте числа в верхней строке. Затем сразу же возвращайтесь к этой странице. Не смотрите на остальные строчки, иначе результат теста не будет достоверным.
3. Вслух произнесите свой телефонный номер наоборот.
4. Теперь запишите числа, которые запомнили. У вас получилось вспомнить всю строчку?
5. Еще раз переверните страницу, снова на 3 секунды приспустите лист бумаги и прочитайте числа во второй сверху строке. Затем возвращайтесь к этой странице.

7. На память запишите числа, прочитанные на предыдущем шаге. Если вы заметили, что это первые семь цифр числа «пи» (3,141592), то вам проще восстановить последовательность, ведь потребовалось запомнить только одно число, а не семь разных.
8. Опять переверните страницу на 3 секунды и прочитайте числа в третьей строке. Вернитесь к этой странице.
9. На память запишите числа, прочитанные на предыдущем шаге. Вам проще вспомнить их, если вы заметили, что это последовательность нечетных чисел от 1 до 13, — в памяти остались три «порции» («нечетные, первое 1, последнее 13» или «нечетные, семь штук, первое 1»), а не семь отдельных чисел.
10. Еще раз на 3 секунды переверните страницу, прочитайте слова в четвертой строке и возвращайтесь.
11. Запишите слова, которые запомнили. Вам удалось восстановить их все?
12. Переверните страницу на 3 секунды, прочитайте слова в последней строке и возвращайтесь.
13. Запишите слова, которые вы запомнили... Эти слова вам наверняка было вспомнить намного проще, чем слова из предыдущей строки, так как они составляют предложение. Таким образом, вашей задачей было запомнить одно предложение, а не семь слов.

В многорежимном пользовательском интерфейсе некоторые действия пользователя могут иметь разные последствия в зависимости от включенного режима, например:

- если в автомобиле нажать на педаль газа, машина будет двигаться вперед, назад или стоять на месте в зависимости от положения рычага переключения передач: D (drive), R (reverse) или N (neutral). Рычаг переключения передач позволяет выбирать режим пользовательского интерфейса автомобиля;
- во многих цифровых камерах выбранный режим управляет действиями при нажатии кнопки спуска: либо делается снимок, либо начинается запись видеофрагмента;

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

3 8 4 7 5 3 9

3 1 4 1 5 9 2

1 3 5 7 9 11 13

город река кукуруза струна машина лопата

в чем смысл жизни

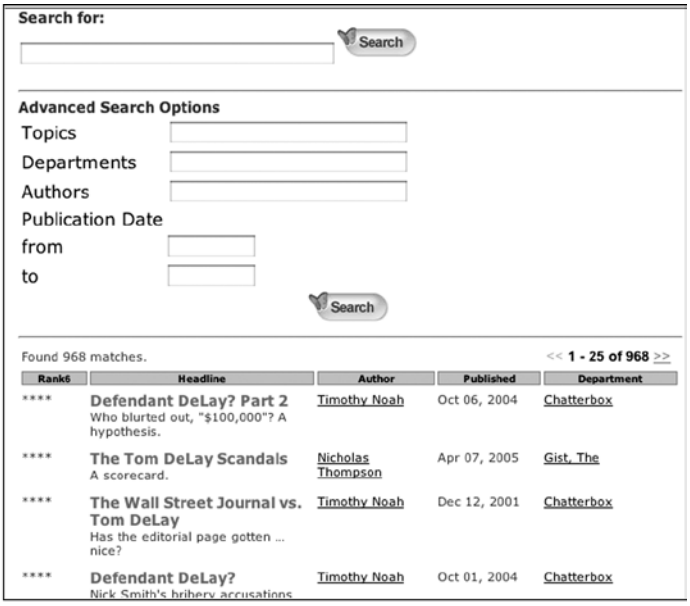
- в программе для рисования щелчок и последующее перетаскивание курсора обычно означает выделение одного или нескольких графических объектов на холсте. Однако в режиме рисования прямоугольника щелчок и перетаскивание — это добавление на холст прямоугольника и растягивание его до желаемого размера.

У многорежимных пользовательских интерфейсов есть свои преимущества, поэтому они часто используются в интерактивных системах. Режимы позволяют реализовать в устройстве больше функций, чем у него есть элементов управления: один и тот же элемент управления выполняет разные задачи в разных режимах. Благодаря режимам в интерактивной системе одним и тем же жестам можно присваивать разные значения, сокращая таким образом число жестов, которые приходится заучивать пользователю. Однако самый большой *недостаток* режимов заключается в том, что люди часто делают *режимозависимые* ошибки: они забывают, в каком режиме функционирует система в данный момент, и случайно выполняют неправильное действие (Johnson, 1990). Особенно часто это происходит в системах, в которых плохо обозначается текущий режим. Из-за риска возникновения режимозависимых ошибок многие рекомендации по проектированию пользовательских интерфейсов содержат совет либо вообще избегать режимов, либо хорошо обозначать активный режим. Человеческая краткосрочная память слишком ненадежна — дизайнер не должен предполагать, что пользователь сможет в любой момент сказать, в каком режиме работает система. Это невозможно без ясного постоянного обозначения текущего режима. Причем это верно даже для тех случаев, когда переводит систему из одного режима в другой сам пользователь.

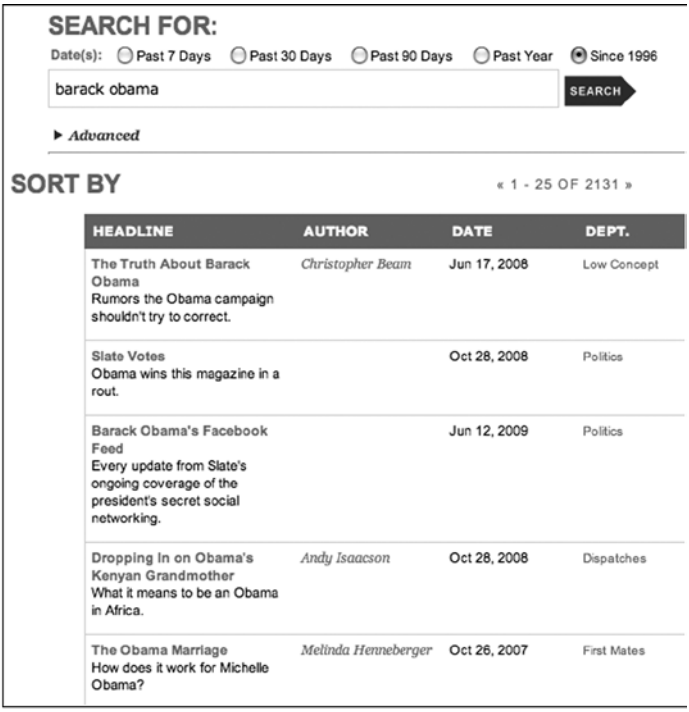
Результаты поиска

Когда пользователь желает найти информацию на компьютере, он прибегает к функции поиска. Он вводит критерии поиска, нажимает на кнопку Старт, а затем просматривает результаты. Для оценки результатов поиска ему зачастую необходимо знать, какими были критерии поиска. Если бы краткосрочная память не была так ограничена, люди бы всегда помнили, поиск по каким терминам они запустили всего несколько секунд назад. Однако, как мы убедились ранее, краткосрочная память очень ограничена. Когда появляется список результатов, внимание

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks



а



б

Рис. 7.2. Результаты поиска на веб-сайте Slate.com: *а* — в 2007 году пользовательские критерии поиска не отображались; *б* — в 2009 году над списком результатов выводится использованный поисковый термин

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

пользователя совершенно естественным образом переключается с введенных им критериев поиска на представленные результаты. Ничего удивительного в том, что люди, просматривающие результаты поиска, зачастую не помнят, какие поисковые термины они только что ввели для получения этого списка.

К сожалению, некоторые разработчики функций поиска этого не понимают, и результаты поиска по сайтам и приложениям не всегда содержат указание терминов, послуживших отправной точкой. Например, в 2006 году на странице результатов поиска веб-сайта Slate.com отображались текстовые поля, позволяющие повторить поиск, однако критерии поиска, по которым был выполнен предыдущий запрос, не выводились (рис. 7.2, *а*). Более свежая версия сайта демонстрирует последние использованные критерии поиска (рис. 7.2, *б*), что снижает нагрузку на краткосрочную память посетителей.

Инструкции

Попросив у подруги рецепт или подсказку, как добраться до ее дома, вы, вероятно, не станете утруждать себя запоминанием длинной последовательности шагов. Вы и так знаете, что не сможете надежно сохранить инструкции в краткосрочной памяти, поэтому записываете их на бумаге или просите подругу прислать вам электронное сообщение. Позже, когда наступит время воспользоваться инструкциями, вы поместите их туда, где с ними можно будет быстро и удобно сверяться до тех пор, пока цель не будет достигнута. Аналогично, если интерактивная система предлагает инструкции для многошаговых операций, она должна позволять человеку сверяться со списком шагов во время выполнения инструкций до тех пор, пока не будет завершен последний шаг. Большинство интерактивных систем соответствуют этому требованию (рис. 7.3), однако в некоторых эта возможность не предусмотрена (рис. 7.4).

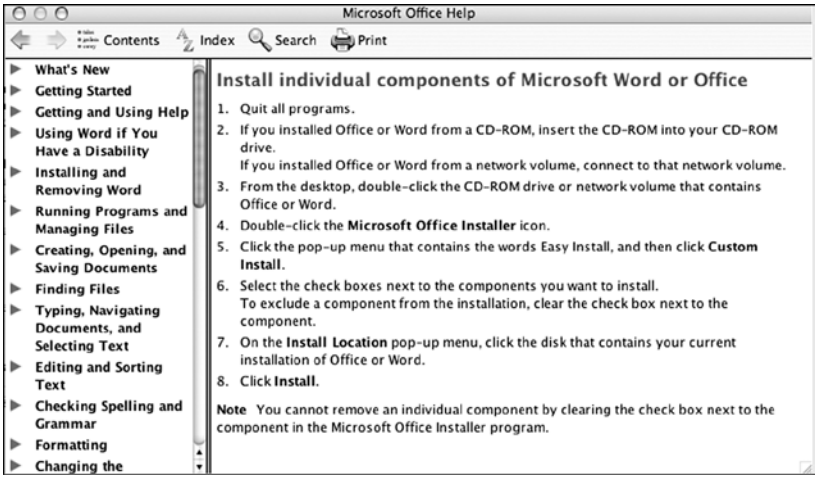


Рис. 7.3. Инструкции в файлах справки Windows остаются на экране все то время, пока они нужны пользователю

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Рис. 7.4. Инструкция по настройке беспроводной сети в Windows XP начинается с того, что пользователю предлагают закрыть диалоговое окно с инструкцией

Характеристики долгосрочной памяти

Долгосрочная память отличается от краткосрочной по многим показателям. В отличие от краткосрочной, долгосрочная память *действительно* представляет собой хранилище воспоминаний.

Однако конкретные воспоминания хранятся не в каком-то отдельном нейроне или поле мозга. Как уже говорилось, любое воспоминание или впечатление представляет собой последовательность активации большого набора нейронов. Связанным воспоминаниям соответствуют перекрывающиеся шаблоны активации нейронов. Это означает, что каждое воспоминание хранится «по распределенной схеме» и охватывает многие части мозга. В этом отношении долгосрочная память в мозге схожа с голографическими световыми изображениями.

Долгосрочная память развилась для того, чтобы наши предки и мы могли хорошо ориентироваться в окружающем мире. Однако у нее много слабых сторон: она склонна к ошибкам и спонтанным ассоциациям, впечатлительна, уникальна и непредсказуема, подвержена ретроактивным изменениям и легко модифицируется различными факторами как в момент фиксации, так и в момент извлечения. Давайте подробнее рассмотрим некоторые из этих слабостей.

Склонность к ошибкам

Почти весь наш жизненный опыт хранится в долгосрочной памяти. В отличие от емкости краткосрочной памяти, емкость долгосрочной памяти кажется безграничной. На основе средней скорости обучения человека Ландауэр (Landauer, 1986) вычислил объем информации, которую возможно усвоить за целую жизнь[1], однако никто еще не измерил и даже не дал приблизительной оценки максимальной информационной емкости человеческого мозга.

Впрочем, содержимое долгосрочной памяти нельзя представлять себе как точные записи о жизненных впечатлениях, сохраненные в высоком разрешении. В терминах, знакомых компьютерным инженерам, долгосрочную память можно охарак-

[1] 10^9 бит, или несколько сотен мегабайт.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

теризовать как хранилище с высокой степенью сжатия информации (во время ее записи в человеческую память огромная часть данных теряется). Изображения, концепции, события, ощущения, действия — все упрощается до комбинаций абстрактных характеристик. Разные воспоминания хранятся с различной степенью детализации, то есть с большим или меньшим числом характеристик.

Например, лицо мужчины, с которым вы общались всего несколько секунд и который не очень важен для вас, может храниться в памяти как среднее лицо европеоида мужского пола с бородой без каких-либо других деталей — все лицо упрощается до трех характеристик. Если позже вас попросят описать этого человека в его отсутствие, большинство из вас честно и откровенно скажут: «Белый мужчина с бородой». Вы не сможете указать на него во время полицейского опознания, где он будет стоять в одном ряду с другими белыми бородатыми мужчинами. В противоположность этому воспоминание о лице лучшего друга включает гораздо больше характеристик и позволяет дать намного более детальное описание, не говоря уж о том, что вы без труда опознаете своего друга в полиции. Любые воспоминания все равно остаются наборами характеристик, а не чем-то вроде растровых изображений.

Еще один пример: у меня со времен детства сохранилось живое воспоминание о том, как я попал под плуг и был сильно ранен. Однако мой отец говорит, что это произошло с моим братом. Один из нас ошибается.

Что касается сферы взаимодействия человека и компьютера, пользователи Microsoft Word наверняка знают о существовании команды, позволяющей вставить номер страницы, но далеко не все из них помнят, в каком меню находится эта команда. Возможно, данная характеристика не была зафиксирована в памяти в тот момент, когда пользователь научился добавлять номера страниц. Или же местоположение команды в меню *было* зафиксировано в памяти, но когда пользователь попытался повторить операцию, данная составляющая воспоминания не активировалась повторно вместе с остальными.

Усиливается эмоциями

В главе 1 я рассказывал о псе, который вспоминал о коте на лужайке перед домом каждый раз, когда хозяйка привозила его домой на автомобиле. Пес был так возбужден, когда увидел кота впервые, что память об этом событии оказалась очень продолжительной. Другой пример: у взрослого человека могут сохраниться яркие воспоминания о его первом дне в детском саду, но, скорее всего, он не вспомнит о десятом. В первый день он, надо полагать, был очень огорчен тем, что родители оставили его в незнакомом месте, однако к десятому дню поход в детский сад уже не воспринимался как экстраординарное событие.

Ретроактивные изменения

Представьте, что вы с семьей путешествуете на круизном лайнере и в один из дней видите в океане китовую акулу. Несколько лет спустя, обсуждая круиз в кругу

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

семьи, вы вспоминаете, как рядом с бортом появился кит, а один из ваших родственников утверждает, что это была акула. У вас обоих это впечатление записалось в долговременной памяти с потерей некоторых деталей, так как оно не подходило под распространенную концепцию.

В реальной жизни также можно встретить немало подобных примеров. Например, ныне покойный бывший президент США Рональд Рейган в 1983 году, во время своего первого президентского срока, встречался с лидерами еврейского государства Израиль. Он рассказывал о том, как в период Второй мировой войны оказался в Европе, где помогал освобождать евреев из нацистских концентрационных лагерей. В действительности же он не бывал в Европе на протяжении Второй мировой войны. В его актерской карьере был фильм о Второй мировой войне, но он был целиком снят в Голливуде.

ТЕСТ ДОЛГОСРОЧНОЙ ПАМЯТИ

Проверьте свою долгосрочную память, попытавшись ответить на следующие вопросы:

1. В ящике с инструментами из главы 1 был рулон липкой ленты?
2. Назовите свой *предыдущий* номер телефона.
3. Какие из перечисленных далее слов *не* присутствовали в списке из теста краткосрочной памяти, приведенного чуть ранее в этой главе?

   поселок поток кукуруза автомобиль шнурок скребок
4. Как звали вашу учительницу в первом классе? Во втором классе? В третьем?..
5. Какой веб-сайт я называл ранее, говоря, что там в списке результатов поиска не отображаются критерии поиска, по которым был выполнен запрос?

Что касается вопроса 3: когда мы запоминаем слова, у нас в памяти зачастую откладывается то, что называется *концепцией*, а не точный набор букв. Например, если человек услышит слово «автомобиль», позже он может сказать, что слово было «машина».

Влияние долгосрочной памяти на дизайн пользовательских интерфейсов

Главное, о чем следует помнить в связи с особенностями долгосрочной памяти, — людям требуются инструменты, помогающие ей и дополняющие ее. Технологии, помогающие нам надолго запоминать вещи и события, начали появляться еще в доисторические времена и продолжают развиваться: зарубки на палочках, узлы на веревках, мнемонические коды, передающиеся из уст в уста рассказы и истории, письменность, свитки, книги, системы счисления, списки покупок, памятки, адресные книги, ежедневники, учетные книги, кулинарные таймеры, компьютеры, электронные секретари, интерактивные общие календари и т. п.

Учитывая то, как сильно человечество нуждается в технологиях, *дополняющих* память, становится очевидно, что разработчики программного обеспечения должны

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

создавать продукты, удовлетворяющие эту потребность. По крайней мере, не стоит разрабатывать системы, *излишне нагружающие* долгосрочную память. Однако многие интерактивные системы именно так себя и ведут.

Аутентификация — одна из функциональных областей, в которых прикладные системы предъявляют непосильные требования к долгосрочной памяти пользователей. Например, разработанное несколько лет назад веб-приложение просило пользователей поменять личный идентификационный номер (personal identification number, PIN) на «число, которое легко... запомнить», однако на это значение одновременно накладывались такие ограничения, что требование невозможно было выполнить (рис. 7.5). Тот, кто написал эти инструкции, похоже, осознал, что требования к PIN-коду оказались завышенными, так как на последнем шаге пользователям рекомендуется записать придуманное значение! Помимо того, что это угроза безопасности — записанный где-то PIN-код может оказаться в руках мошенника, — просьба записать PIN-код нагружает память человека еще одной задачей: ему приходится запоминать, куда он спрятал листок с записанным кодом.

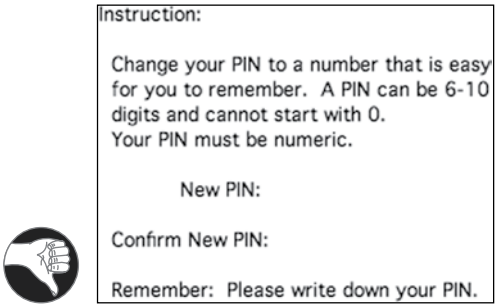


Рис. 7.5. Согласно инструкциям, пользователь должен создать легкий для запоминания PIN-код, однако ограничения делают эту задачу невыполнимой

Еще один пример обременения долгосрочной памяти пользователя во имя безопасности можно обнаружить на веб-сайте Intuit.com. Для того чтобы приобрести программное обеспечение, посетителю необходимо зарегистрироваться. В процессе регистрации в меню необходимо выбрать контрольный вопрос и ответить на него (рис. 7.6). Но что делать, если человек не может ответить ни на один из перечисленных вопросов? Что, если он не в состоянии вспомнить кличку первого домашнего питомца и символ университета или дать ответы на остальные вопросы?

Однако на этом проблемы не заканчиваются. Некоторые вопросы подразумевают более одного возможного ответа. Многие люди посещали несколько начальных школ, у них было в детстве несколько лучших друзей и примеров для подражания. Чтобы завершить регистрацию, человеку необходимо выбрать вопрос, а затем *запомнить* ответ, который он дал для сайта Intuit. Как? Вероятно, записав его где-то. Позже, когда Intuit.com запросит ответ на контрольный вопрос, человеку придется *вспоминать*, где этот ответ записан. Зачем нагружать человеческую память, если

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

можно было бы просто позволить пользователю придумать собственный контрольный вопрос, на который он с легкостью даст один-единственный возможный ответ?

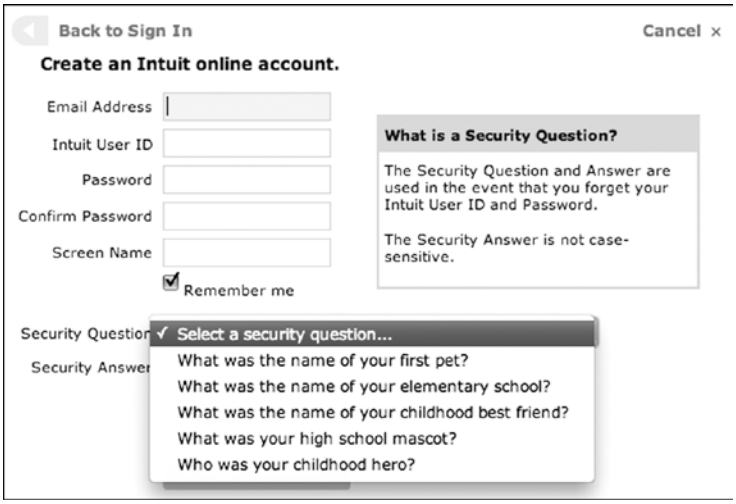


Рис. 7.6. Процесс регистрации на веб-сайте Intuit.com излишне нагружает долгосрочную память: у пользователя может не быть уникальных и легко запоминающихся ответов ни на один из заданных вопросов

Такие непомерные требования, предъявляемые к долгосрочной памяти людей, противодействуют безопасности и эффективности, которую, предположительно, должны обеспечивать компьютерные приложения (Schrage, 2005), так как пользователи:

- приклеивают листочки с ответами на компьютеры или рядом с ними или «прячут» их в ящики стола;
- обращаются в службу поддержки пользователей за восстановлением пароля, который не в состоянии вспомнить;
- используют пароли, которые посторонним людям несложно будет угадать;
- отключают использование учетных данных в системах либо используют во всех системах один и тот же набор из имени и пароля.

Форма регистрации на веб-сайте NetworkSolutions.com демонстрирует небольшое усовершенствование в деле обеспечения безопасности и удобства пользователя. Как и на сайте Intuit.com, здесь предлагается список контрольных вопросов, но пользователь также может создать собственный вопрос, уникальный ответ на который ему будет запомнить проще (рис. 7.7).

Особенности долгосрочной памяти также объясняют, почему в интерактивных системах обучение и долговременное запоминание происходят намного эффективнее, если пользовательский интерфейс характеризуется единообразием.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

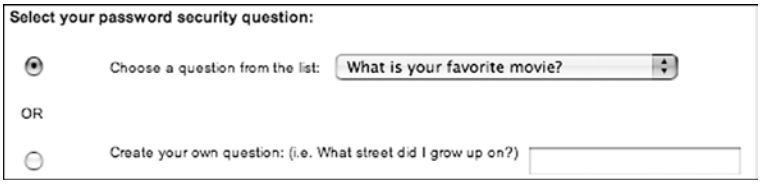


Рис. 7.7. Веб-сайт NetworkSolutions.com разрешает пользователям создать собственный контрольный вопрос, если ни один вопрос из стандартного меню им не подходит

Чем более похожи действия, необходимые для запуска разных функций, или действия, выполняемые на объектах разных типов, тем меньше пользователю приходится запоминать[1]. Пользовательские интерфейсы с множеством исключений и разнородными функциями и объектами заставляют пользователей сохранять в долгосрочной памяти множество характеристик для каждой функции и объекта, а также правильный контекст их применения. Необходимость фиксировать большое количество характеристик затрудняет изучение подобных пользовательских интерфейсов. Кроме того, возрастает вероятность того, что во время записи или извлечения воспоминания память пользователя пропустит те или иные важные характеристики, а это означает, что человек не запомнит нужную информацию, запомнит ее неправильно или сделает какую-то другую ошибку.

Рассмотрим три альтернативных проекта наборов сочетаний клавиш для операций вырезания и вставки в гипотетическом мультимедийном редакторе документов. Редактор поддерживает создание документов, содержащих текст, чертежи, таблицы, изображения и видео. В проекте А операциям вырезания и вставки назначаются одни и те же сочетания клавиш независимо от того, содержимое какого типа редактируется. В проекте В сочетания клавиш для операций вырезания и вставки уникальны для каждого типа содержимого. В проекте С для всех типов содержимого, *за исключением* видео, используются одни и те же клавишные сочетания (табл. 7.1).

Первый вопрос: какой из этих вариантов самый простой для изучения? Очевидно, вариант А.

Второй вопрос: какой из них самый сложный для изучения? Здесь ответ не так очевиден. На первый взгляд, кажется, что проект В сложнее остальных, так как в нем меньше всего единообразия. Однако в действительности ответ зависит от того, что мы подразумеваем под «сложностью для изучения».

Если это означает дизайн, при использовании которого пользователям потребуется больше всего времени для достижения хорошего уровня продуктивности, то это, однозначно, проект В. Большинство пользователей тратят много времени на запоминание разнообразных сочетаний клавиш, выполняющих вырезание и вставку

[1] См. главу 11.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

разных типов содержимого. Однако люди поразительно хорошо адаптируются, если у них есть достаточная мотивация, — мы способны заучивать какие угодно не связанные между собой вещи, если, например, это требуется для выполнения работы. Когда-нибудь — возможно, через месяц — пользователи привыкнут и смогут быстро работать в приложении, сделанном по проекту B. В противоположность этому пользователи дизайна C начнут продуктивно работать почти одновременно с пользователями дизайна A, то есть через несколько минут.

Таблица 7.1. Какой дизайн пользовательского интерфейса будет самым простым для изучения и запоминания? А какой самым сложным?

Объект	Клавиатурные комбинации в редакторе документов					
	Проект A		Проект B		Проект C	
	Вырезать	Вставить	Вырезать	Вставить	Вырезать	Вставить
Текст	CTRL+X	CTRL+V	CTRL+X	CTRL+V	CTRL+X	CTRL+V
Чертеж	CTRL+X	CTRL+V	CTRL+C	CTRL+P	CTRL+X	CTRL+V
Таблица	CTRL+X	CTRL+V	CTRL+Z	CTRL+Y	CTRL+X	CTRL+V
Изображение	CTRL+X	CTRL+V	CTRL+M	CTRL+N	CTRL+X	CTRL+V
Видео	CTRL+X	CTRL+V	CTRL+Q	CTRL+R	CTRL+E	CTRL+R

Однако если «самый сложный для изучения» — это тот, в котором пользователи дольше всего будут делать ошибки, то речь о проекте C. Для всех типов содержимого документа вырезание и вставка выполняются одними и теми же сочетаниями клавиш, но не для видео. Несмотря на то что пользователи дизайна C скоро начнут продуктивно работать, они будут постоянно ошибаться, пытаясь вырезать и вставить видео сочетаниями CTRL+X и CTRL+V. Ошибки будут возникать на протяжении нескольких месяцев, а некоторые не избавятся от них никогда.

Несмотря на существующую критику концепции единообразия как недостаточно четкой и провоцирующей неправильное применение (Grudin, 1989), тот факт, что единообразие в пользовательском интерфейсе значительно снижает нагрузку на долгосрочную память человека, неоспорим. Марк Твен когда-то написал: «Если вы говорите правду, вам не приходится ничего вспоминать». Знаменитую цитату можно перефразировать: «Если бы все работало одинаково, нам не приходилось бы много запоминать». К вопросу единообразия мы вернемся в главе 11.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 8

Ограниченность внимания влияет на мысли и действия

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Во время осмысленного взаимодействия с окружающим миром, частью которого являются компьютерные системы, некоторые аспекты поведения людей подчиняются предсказуемым шаблонам. Многие шаблоны при этом формируются под влиянием таких факторов, как ограниченное внимание и малая емкость краткосрочной памяти. Людям намного проще работать с интерактивными системами, умеющими распознавать и поддерживать подобные шаблоны. Более того, некоторые правила разработки пользовательских интерфейсов основываются непосредственно на шаблонах поведения и, следовательно, учитывают ограничения краткосрочной памяти и внимания. В этой главе рассказывается о шести важных шаблонах.

Мы концентрируемся на целях и почти не обращаем внимания на инструменты

Как я уже объяснял в главе 7, наше внимание сильно ограничено. Когда человек выполняет задачу — пытается достичь цели, его внимание в основном сосредоточено на целях и на прочей информации, связанной с его задачей. Чаще всего люди очень мало внимания обращают на инструменты, с помощью которых выполняется задача, независимо от того, используют они компьютерные приложения, интерактивные службы или электронные устройства. Инструментам они уделяют лишь поверхностное внимание, и только в случае необходимости.

Разумеется, мы *способны* обращать внимание на инструменты. Однако объем внимания (то есть краткосрочной памяти) крайне ограничен. Сосредоточиваясь на инструментах, мы не можем уделять достаточно внимания деталям задачи. Перенос фокуса внимания увеличивает вероятность того, что человек забудет, что он делал или на каком конкретно этапе выполнения задачи находился.

Например, вы стрижете лужайку, и газонокосилка внезапно глохнет. Вы сразу же прекращаете работу и фокусируете внимание на газонокосилке. Повторный ее запуск становится вашей основной задачей, а сама она — центром внимания. Вы не обращаете особого внимания на инструменты, которыми пользуетесь для починки газонокосилки, точно так же, как почти не обращали внимания на саму косилку, пока центром вашего внимания была лужайка. После того как машина снова заработало, вы возвращаетесь к подстриганию лужайки, но вряд ли помните, на каком месте остановились. К счастью, это видно по длине травы — лужайка сама подсказывает, с какого места продолжать.

Другие задачи, например чтение документа, табличные расчеты, подсчет золотых рыбок в аквариуме, не дают таких ясных подсказок о прерванной задаче и о том, на каком этапе вы остановились. Возможно, вам придется каждый раз начинать сначала. Возможно, вы вообще забудете, чем занимались, и перенесете все внимание на другую задачу.

Вот почему в большинстве рекомендаций по разработке программного обеспечения особо подчеркивается, что прикладные программы и большая часть веб-сайтов не

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

должны отвлекать внимание на себя, их назначение — служить неприметным фоном, для того чтобы пользователи могли сфокусироваться на собственных целях. Одна из рекомендаций даже послужила названием для известной книги С. Круга, посвященной веб-дизайну: «Не заставляйте меня думать» (Don't Make Me Think) (Krug, 2005). Это название подразумевает следующее: если ваша программа или веб-сайт заставляет меня думать о *них самих*, а не о том, что я пытаюсь сделать, то я не ваш клиент.

Мы следим за ходом выполнения задачи, используя внешние подсказки

Поскольку наша краткосрочная память и внимание очень ограничены, мы учимся не полагаться на них. Мы «делаем зарубки» в своем окружении, для того чтобы оно подсказывало нам, на каком этапе задачи мы находимся. Например:

- **подсчет объектов** — если возможно, мы передвигаем уже подсчитанные объекты в другую кучку, отделяя их от объектов, которые еще не пересчитаны. Если переместить объект невозможно, мы указываем пальцем на последний подсчитанный объект. Чтобы не потеряться в последовательности чисел, загибаем пальцы, рисуем графические отметки или записываем номера;
- **чтение книг** — прерывая чтение, мы закладываем страницу, на которой остановились;
- **арифметика** — мы учимся выполнять арифметические действия на бумаге или используем калькулятор;
- **контрольные списки** — контрольный перечень помогает как долгосрочной, так и краткосрочной памяти. Если речь идет о критически важной или редко выполняемой задаче, контрольный список помогает не пропустить ни одного шага процесса. Таким образом, этот вариант памятки *дополняет* ненадежную долгосрочную память. Выполняя задачу, мы вычеркиваем шаги по мере выполнения — это помощь краткосрочной памяти. Если в контрольном перечне невозможно вычеркивать пункты, его трудно использовать. В такой ситуации мы обычно делаем копию и помечаем выполненные элементы на ней;
- **редактирование документов** — люди часто хранят документы, предназначенные для редактирования в будущем, документы, редактируемые сейчас, и уже отредактированные документы в отдельных папках.

Из этого следует такой вывод: интерактивные системы должны обозначать, что пользователи уже сделали, а что еще сделано не было. Обычно в клиентах электронной почты разными способами помечаются прочитанные и непрочитанные сообщения, на веб-сайтах разными цветами окрашиваются посещенные и не посещенные ссылки, а во многих приложениях завершенные этапы многошаговой задачи отмечаются специальными символами (рис. 8.1).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Рис. 8.1. Утилита Mac OS Software Update показывает, какие обновления уже установлены (галочка), а какие находятся в процессе установки (вращающийся кружок)

Второй вывод, который следует сделать дизайнеру, заключается в том, что интерактивные системы должны разрешать пользователям помечать или перемещать объекты в зависимости от того, завершена работа с ними или еще нет. В операционной системе Mac OS пользователь может назначать файлам разные цвета. Как и перекидывание файлов между папками, эту технику можно применять для отслеживания готовности задачи (рис. 8.2).

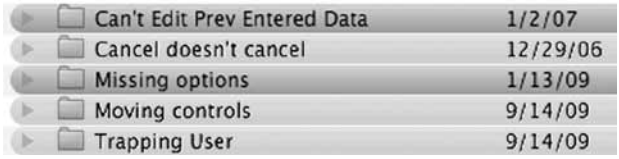


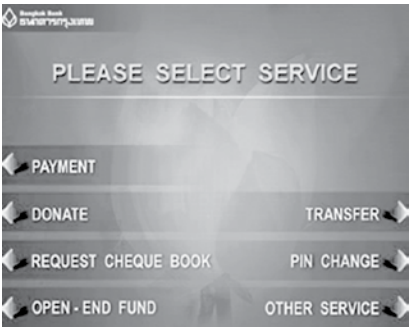
Рис. 8.2. В Mac OS с папками и файлами можно связывать разные цвета; пользователь отслеживает процесс выполнения задачи по цветам объектов

Мы идем к цели по информационному «следу»

Фокусируя внимание на цели, мы очень *буквально* интерпретируем все, что видим на дисплее или слышим в телефонном меню. Люди особо не задумываются над инструкциями, названиями команд, метками параметров, значками, элементами навигационного меню или любыми другими составляющими пользовательского интерфейса или электронного инструмента. Если человек в голове держит цель «забронировать авиабилет», то его внимание привлечет любой объект, подписанный «купить», «рейс», «билет» или «бронирование». Находясь в процессе бронирования билета, человек не обратит внимания на другие элементы, которые, по мнению дизайнера или маркетолога, должны привлекать покупателей, такие как «выгодные отели». В то же время подобная реклама привлечет посетителей, ищущих возможность недорого забронировать номер.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Человеческое свойство замечать на экране компьютера только те вещи, которые имеют отношение к актуальной цели, и буквальное понимание представленной информации при выполнении задачи на компьютере получило название «идти к цели по информационному следу» (Chi, Pirolli, Chen, Pitkow, 2001; Nielsen, 2003). Возьмем в качестве примера экран банкомата (рис. 8.3). Что в первую очередь привлекает ваше внимание при попытке достичь каждой из перечисленных целей?



Какой пункт меню привлечет ваше внимание при попытке выполнить каждую из перечисленных задач?

- Оплатить счет
- Перевести деньги на сберегательный счет
- Заплатить стоматологу путем перевода средств
- Поменять PIN-код
- Открыть новый счет
- Приобрести дорожные чеки

Рис. 8.3. Экран банкомата. В первую очередь наше внимание привлекают элементы, названия которых буквально совпадают с нашей целью

Вы, вероятно, заметили, что некоторые из перечисленных целей первоначально направляют ваше внимание к неподходящим пунктам меню. «Заплатить стоматологу путем перевода средств» делается в меню «Платежи» (Payment) или «Переводы» (Transfer)? Задача «Открыть новый счет», скорее всего, заставила вас на мгновение взглянуть на меню «Инвестиционный фонд» (Open-end Fund), хотя в действительности это делается через меню «Прочие услуги» (Other Service). А прочитав пункт «Приобрести дорожные чеки», вы разве не посмотрели на меню «Заказать чековую книжку» (Request cheque book)? Наверняка так и было, ведь в этих двух фразах встречается одно и то же слово.

Стратегия достижения цели по информационному следу, применение которой можно наблюдать в разнообразнейших ситуациях и системах, предполагает, что разработчики должны создавать в своих интерактивных системах сильные и заметные «следы», действительно приводящие пользователей к их целям. Для этого им необходимо понимать, какие цели возникают у пользователя во время выполнения задачи в каждой точке принятия решения, и в каждой такой точке предлагать варианты достижения всевозможных важных целей с четким обозначением, куда ведет каждый из путей.

Представьте, например, что перед вами стоит задача отменить бронирование или запланированный платеж. Вы нажимаете нужную кнопку, и в ответ отображается диалоговое окно с вопросом, действительно ли вы хотите сделать это. Как должны быть подписаны варианты? С учетом того, что люди, двигаясь к своей цели по информационному следу, буквально понимают слова, стандартные обозначения кнопок OK (что значит «да») и «Отмена» (что значит «нет») могут перевести их на ложный

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

путь. Сравните диалоговые окна подтверждения отмены с сайтов Marriott.com и Quicken.com. Вы видите, что вариант Marriott.com дает более ясный след (рис. 8.4).

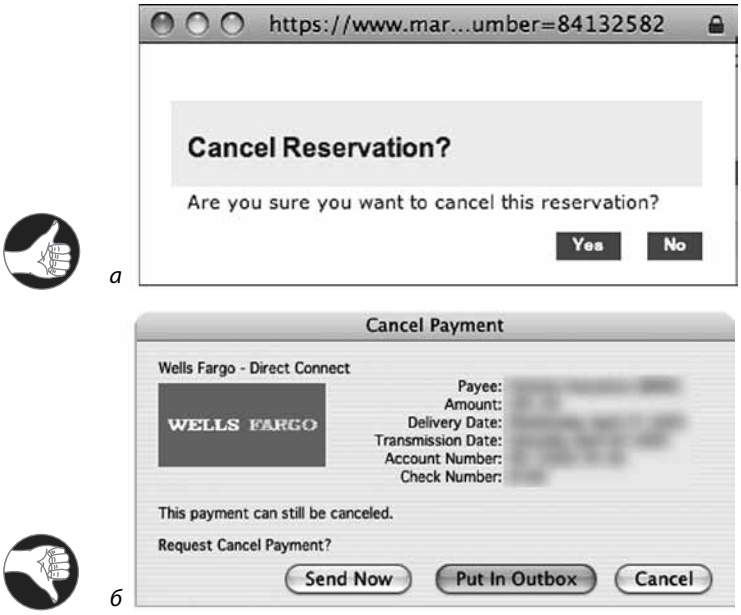


Рис. 8.4. Диалоговое окно подтверждения отмены на сайте Marriott (*а*) дает более ясный след, чем вариант Quicken (*б*)

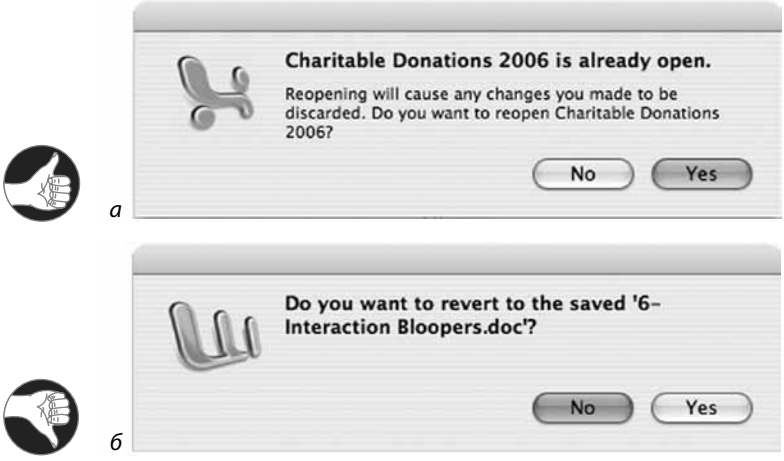


Рис. 8.5. Предупреждение о попытке открыть уже открытый файл намного понятнее в Microsoft Excel (*а*), чем в Word (*б*)

Возьмем другой пример: предположим, что вы пытаетесь открыть какой-то документ, забыв, что чуть раньше уже открыли его. Разработчики Microsoft Excel лучше

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

предусмотрели подобную ситуацию, чем дизайнеры Microsoft Word. Они поняли, какие цели могут стоять перед вами в этой точке, и представили инструкции и варианты, четко разъясняющие возможные пути действия (рис. 8.5).

Мы предпочитаем знакомые пути

Людям известно о том, что их внимание ограничено, поэтому они действуют соответствующим образом. Стремясь достичь определенной цели, они по возможности следуют *знакомыми* маршрутами, а не изыскивают *новые*, особенно если срок исполнения задачи поджимает. Изучение нового пути достижения цели подразумевает решение очередной проблемы (подробнее об этом рассказывается в главе 10), а это тяжелая нагрузка на внимание и краткосрочную память. И наоборот, выполнение задачи по знакомому, хорошо изученному сценарию происходит почти автоматически и не нагружает внимание и краткосрочную память.

Давным-давно во время сеанса тестирования удобства в использовании один из участников, приступая к задаче, сказал мне: «Я тороплюсь, так что пойду по длинному пути».

Он подозревал, что, вероятно, существует более эффективный способ достижения поставленной перед ним цели, однако он также знал, что изучение короткого пути — это трата времени и усилий, и на такую жертву не был готов идти.

Выучив один способ выполнения определенной задачи в компьютерном приложении, мы продолжаем применять его и можем так *никогда* и не обнаружить другие, более эффективные. Даже если мы прочитаем или услышим где-то, что существует «лучший» способ, скорее всего, мы по-прежнему будет придерживаться старого и привычного, потому что так проще и удобнее и, что самое главное, так не приходится лишний раз думать, а при использовании компьютера это очень важно. Люди готовы *больше* нажимать на клавиши, лишь бы *меньше* думать.

Такое стремление выбирать знакомые, почти не требующие приложения умственных усилий пути диктует несколько правил разработки интерактивных систем:

- **Иногда отсутствие необходимости думать оправдывает большое число нажатий клавиш.** Если программное обеспечение предназначено для редкого или спонтанного использования, как, например, банкоматы или приложения для домашней бухгалтерии, то гораздо важнее сделать так, чтобы пользователь быстро начал продуктивно работать, не слишком затрудняясь решением задач обучения, чем сократить количество нажатий клавиш. Такие приложения используются слишком редко, и количество шагов, необходимое для выполнения задачи, не играет особой роли. Однако если профессиональные пользователи — скажем, операторы телефонной службы бронирования авиабилетов — работают с приложением целый день в напряженных производственных условиях, то каждое дополнительное нажатие клавиши в ходе решения задачи значительно увеличивает суммарные издержки.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- **Направляйте пользователей на лучшие пути.** С самого первого экрана или домашней страницы программное обеспечение должно показывать пользователям пути, ведущих к достижению их целей. Фактически, в любом приложении должен присутствовать ясный информационный след.
- **Помогайте опытным пользователям ускоряться.** Предусмотрите для них возможность по мере наработки опыта переключаться на более быстрые пути решения задач. Предназначенные для новичков медленные пути также должны содержать указатели на быстрые сценарии, если таковые имеются. Именно по этой причине в большинстве приложений сочетания клавиш для часто используемых функций отображаются в меню рядом с названиями соответствующих команд.

Цикл мышления: цель, выполнение, оценка

Изучая человеческое поведение в течение многих десятилетий, ученые постоянно обнаруживали, что широчайшим диапазоном наших действий управляет циклический шаблон:

- сформировать **цель** — открыть банковский счет, съесть персик или удалить слово из документа;
- выбрать и **выполнить** действия, для того чтобы сделать очередной шаг к цели;
- **оценить**, сработали ли действия, то есть достигнута ли цель или, по крайней мере, стала ли она ближе;
- повторять до тех пор, пока цель не будет достигнута (или не станет ясно, что она недостижима).

Люди постоянно действуют в соответствии с этим шаблоном, повторяя его раз за разом (Card, Moran, Newell, 1983). Например, мы пытаемся вставить в документ рисунок, а это часть крупной задачи по написанию курсовой, что, в свою очередь, входит в задачу более высокого уровня — успешно завершить курс истории, над которой стоит задача закончить университет, подчиняющаяся задаче получить хорошую работу, и все это нам нужно для достижения цели самого высокого уровня, то есть обеспечения себе комфортной жизни.

В качестве примера давайте подробно рассмотрим цикл типичной задачи, выполняемой за компьютером: покупка в Интернете авиабилета. В первую очередь человек формулирует цель, а затем начинает разбивать задачу на действия, которые, предположительно, должны привести его к цели. Многообещающие действия выбираются для исполнения, исполняются, а затем оцениваются: продвинулся ли человек благодаря им к своей цели.

- **Цель** — приобрести авиабилет в Берлин, используя веб-сайт любимого туристического агентства.
- **Шаг 1** — открыть веб-сайт туристического агентства. Вы все еще далеки от цели.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- **Шаг 2** — найти подходящие рейсы. Это нормальный, очень предсказуемый шаг для веб-сайта турагентства.
- **Шаг 3** — просмотреть результаты поиска. Выбрать один из перечисленных рейсов. Если ни один из рейсов в списке результатов вам не подходит, повторить шаг 2 с другими критериями поиска. Вы еще не достигли цели, но уверены, что скоро доберетесь.
- **Шаг 4** — перейти к странице оформления покупки. Вы уже так близки к цели, что почти осязаете ее.
- **Шаг 5** — подтвердить данные полета. Проверить, все ли указано правильно. Если нет, исправить, в противном случае продолжить. Почти готово.
- **Шаг 6** — оплатить билет кредитной картой. Проверить реквизиты карты. Все ли в порядке?
- **Шаг 7** — распечатать электронный билет. Цель достигнута.

В примере с авиабилетом, для того чтобы не усложнять описание, я не вдавался в подробности каждого из шагов. В более детальном описании мне пришлось бы создать несколько подшагов, в которых все крутилось бы в привычном цикле «цель — выполнение — оценка».

Давайте рассмотрим другой пример и на этот раз изучим подробности некоторых высокоуровневых шагов. Нашей задачей будет отправить цветы подруге. При взгляде на самый высокий уровень мы видим задачу такой:

```
Отправить цветы подруге.
```

Для того чтобы изучить цикл «цель — выполнение — оценка» для этой задачи, ее необходимо разбить на более детальные шаги. Первый вопрос — *как* отправить цветы подруге? Чтобы ответить на него, разделим задачу верхнего уровня на две подзадачи.

```
Отправить цветы подруге.
  Найти веб-сайт службы доставки цветов.
  Заказать доставку цветов подруге.
```

В определенном смысле идентифицированные нами два шага дают достаточную детализацию. После исполнения каждого из них мы проверяем, стала ли цель ближе. Но *как* выполняется каждый из этих шагов? Давайте посмотрим, разбив задачи второго уровня (предназначенные для достижения подцелей второго уровня) на подшаги:

```
Отправить цветы подруге.
  Найти веб-сайт службы доставки цветов.
    Открыть веб-браузер.
    Перейти на страницу поиска Google.
    Ввести в строку поиска «доставка цветов».
```

продолжение ⇨

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

```
    Просмотреть первую страницу результатов поиска.
    Открыть некоторые из перечисленных ссылок.
    Выбрать службу доставки цветов.
  Заказать доставку цветов подруге.
    Просмотреть ассортимент цветов выбранной службы.
    Выбрать цветы.
    Указать адрес и дату доставки.
    Оплатить цветы и доставку.
```

После выполнения каждого подшага мы оцениваем ситуацию — приблизились ли мы к подцели того уровня, на котором сейчас находимся? Чтобы понять, каким образом выполняется и оценивается сам подшаг, его нужно взять в качестве подцели очередного уровня и рассмотреть этапы достижения.

```
Отправить цветы подруге.
  Найти веб-сайт службы доставки цветов.
    Открыть веб-браузер.
      Щелкнуть на значке браузера на панели инструментов,
        в меню «Пуск» или на рабочем столе.
    Перейти на страницу поиска Google.
      Если служба Google не установлена в качестве начальной страницы,
        выбрать эту ссылку в списке избранного.
      Если ссылка на Google не добавлена в список избранного,
        ввести Google.com в адресном поле браузера.
      Ввести в строку поиска «доставка цветов».
        Установить курсор в поле поиска.
        Ввести текст.
        Исправить опечатки: не «цвитов», а «цветов».
      Открыть некоторые из перечисленных ссылок.
        Подвести указатель мыши к ссылке.
        Щелкнуть на ссылке.
        Взглянуть на открывшуюся веб-страницу.
      Выбрать службу доставки цветов.
        Ввести URL-адрес выбранной службы в адресной строке браузера.
...
```

Принцип вы поняли. Можно было бы и дальше углубляться в детали — вплоть до уровня отдельных нажатий клавиш и движений мыши, однако нам редко требуется такой уровень детализации для понимания задачи и создания программного обеспечения, соответствующего ее шагам и циклу «цель — выполнение — оценка» каждого шага. Как сделать так, чтобы программное обеспечение помогало пользователям работать в цикле «цель — выполнение — оценка»? Применить все перечисленные далее советы:

- **Цель.** Четко обозначайте пути, включая первоначальные шаги, к целям пользователя, для достижения которых предназначено программное обеспечение.
- **Выполнение.** Прикладные концепции (объекты и действия) должны основываться на задачах, а не на реализации (см. главу 11). Не заставляйте пользова-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

телей гадать, каким образом объекты и действия в программе соответствуют объектам и действиям в их задачах. Обеспечивайте ясный информационный след в точках выбора, чтобы пользователь не терялся на пути к цели. Не заставляйте его выбирать действия, которые приводят к цели обходными путями.

- **Оценка.** Обеспечивайте обратную связь и отображайте информацию о состоянии, чтобы пользователь знал, насколько успешно он продвигается к цели. Предусмотрите возможность возврата к исходной точке, если выбранный путь не привел пользователя к цели.

Иллюстрацию для последнего совета — понятная обратная связь и информирование о прогрессе выполнения многошаговой задачи — можно найти в системе бронирования авиабилетов ITN (рис. 8.6). Кстати, вам эта картинка не кажется знакомой? Наверняка так и есть: вы уже видели ее в главе 5 (см. рис. 5.15, *б*), и ваш мозг узнал ее.

Рис. 8.6. Система бронирования авиабилетов ITN ясно показывает пользователю, на каком этапе достижения своей цели тот находится

Главная цель достигнута, не забудьте «прибрать за собой»

Работа цикла «цель — выполнение — оценка» напрямую зависит от краткосрочной памяти, и это объяснимо: краткосрочная память представляет собой всего лишь фокус внимания человека в каждый конкретный момент. Частью фокуса внимания является и наша текущая цель. Остальные ресурсы внимания используются для получения информации, необходимой для достижения текущей цели. Фокус внимания смещается по мере того, как задачи выполняются, а текущей становится сначала самая высокоуровневая цель, затем цели нижних уровней, затем снова очередная цель высокого уровня и т. д.

Внимание — ресурс весьма скудный. Наш мозг не расходует его попусту и не фокусируется на вещах, уже потерявших значимость. Таким образом, после того как мы завершаем задачу, ресурсы внимания, которые были выделены для достижения главной цели, освобождаются и могут быть перенаправлены на другую информацию, которая теперь представляется нам более важной. Создается впечатление, что как только цель достигнута, все относящееся к ней тут же выпадает из краткосрочной памяти, то есть мы забываем об этой цели.

Данная особенность восприятия приводит к тому, что у задач часто остаются недоделанные «хвосты». Например, люди нередко забывают:

- выключить фары автомобиля после остановки, для того чтобы не разряжался аккумулятор;

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- вынуть последнюю страницу документа из копира или сканера;
- выключить духовку или горелку после готовки;
- добавить закрывающую скобку или кавычку после ввода фрагмента текста;
- выключить сигналы поворота после выполнения маневра;
- забрать с собой книгу, которую читали во время полета;
- выйти из системы на общественном компьютере после завершения всех задач;
- переключить устройства и программы в обычный режим после работы в специальном режиме.

Эти провалы в краткосрочной памяти, случающиеся после завершения задачи, абсолютно предсказуемы и их можно избежать. Когда с нами происходит нечто подобное, мы говорим, что стали рассеянными, однако это всего лишь результат естественной работы (или нежелания работать) мозга в сочетании с отсутствием помощи от программ и устройств.

Для предупреждения провалов в памяти в интерактивные системы можно и нужно встраивать инструменты напоминания о висящих «хвостах». В некоторых случаях возможно даже сделать так, чтобы система сама завершала задачу за человека, например:

- автомобили сами выключают сигналы поворота после маневра;
- автомобили должны автоматически отключать фары при выключении зажигания (и большая часть сейчас так и делает) или, по крайней мере, напоминать водителям, что фары все еще включены;
- копиры и сканеры должны автоматически выбрасывать все документы после завершения задач или, по крайней мере, сигнализировать о том, что страница все еще находится в устройстве;
- кухонные плиты должны подавать сигнал в случае, если горелка долго остается включенной, но кастрюли над ней нет (то же верно для включенных духовок, внутри которых ничего нет);
- при попытке выключить компьютер или перевести его в спящий режим устройство должно предупреждать о незавершенных фоновых задачах, таких как сохранение файлов или отправка документа на печать;
- специальные программные режимы (в противоположность «нормальным») должны выключаться автоматически: либо по таймеру (как делается в некоторых устройствах), либо благодаря ручкам с подпружиниванием, которые необходимо физически удерживать в «ненормальном» состоянии и которые сразу же возвращаются в «нормальное», если их отпустить (Johnson, 1990).

Разработчикам программного обеспечения следует внимательно анализировать задачи, решение которых поддерживают их системы. Могут ли в этих задачах появляться «хвосты»? Если да, то система должна либо напоминать пользователям о незавершенных шагах, либо «подчищать хвосты» за них, избавляя людей от мыслей о лишних деталях.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 9

Узнавать легко, вспоминать трудно

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В главе 7 я рассказал о сильных сторонах и ограничениях человеческой долгосрочной памяти и об их значении для разработки интерактивных систем. В этой главе мы продолжим обсуждать тему долгосрочной памяти, рассмотрев важные различия между двумя ее функциями: узнаванием и вспоминанием.

Узнавать просто

Благодаря миллионам лет естественного отбора и эволюции человеческий мозг научился быстро узнавать вещи. В противоположность этому очевидно, что вспоминание, то есть извлечение воспоминаний из памяти без перцепционной поддержки, было куда менее важно для выживания, так что с этой задачей наш мозг справляется намного медленнее.

Вспомните, как работает долгосрочная память (см. главу 7): впечатления поступают через органы чувств, сигналы которых, достигая мозга, запускают сложные схемы нейронной деятельности. То, какой нейронный шаблон будет активирован благодаря впечатлению, зависит не только от характеристик поступившей информации, но и от контекста, в котором произошло восприятие. Схожие впечатления в схожих контекстах запускают похожие схемы нейронной деятельности. Благодаря многократной активации определенного нейронного шаблона его запуск упрощается, то есть с каждым разом он запускается все быстрее и свободнее. Со временем между нейронными шаблонами развиваются такие связи, что активация одного шаблона приводит к активации другого. Грубо говоря, каждый шаблон нейронной активности соответствует отдельному воспоминанию.

Шаблоны нейронной активности — а это и есть, по сути, наши воспоминания — можно активировать двумя путями:

- через поступление дополнительных впечатлений от органов чувств;
- вследствие иной мозговой активности.

Новое впечатление, похожее на то, которое наш мозг уже когда-то регистрировал, к тому же возникшее в похожем контексте, легко запускает сходную схему нейронной активности, благодаря чему происходит узнавание. Фактически, узнавание — это результат совместной работы восприятия и долгосрочной памяти.

Такое успешное сотрудничество функций мозга и дало нам преимущество умения быстро оценивать ситуацию. У наших далеких предков, живших в восточноафриканских саваннах, была всего одна, максимум две секунды на то, чтобы принять решение: виднеющееся в траве животное — это еда или хищник, для которого они сами могут стать легкой добычей (рис. 9.1)? От этого зависело выживание.

Точно так же люди очень быстро распознают лица — обычно на это требуется всего доля секунды (рис. 9.2). До недавнего времени детали этого процесса оставались загадкой. Однако в те времена ученые предполагали, что узнавание — это процесс, в ходе которого увиденные лица сохраняются в краткосрочной памяти и сравни-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ваются с ранее записанными в долгосрочную память. Поскольку мозг узнает лица очень быстро, психологи полагали, что мозг одновременно производит поиск в разных частях долгосрочной памяти, применяя технику, которая в компьютерной науке называется параллельной обработкой. Однако даже процесс параллельного поиска с очень большим количеством потоков не смог бы обеспечить такую поразительную скорость распознавания лиц.

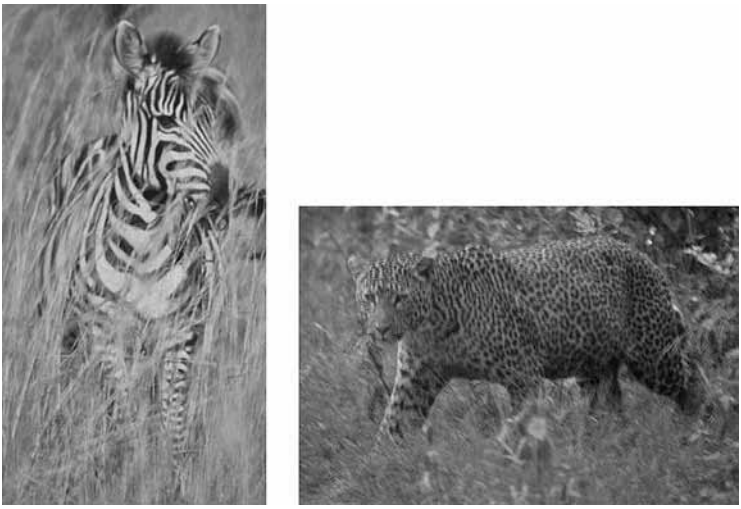


Рис. 9.1. Ранним гоминидам приходилось очень быстро распознавать среди животных потенциальные жертвы и хищников

Сегодня принято считать, что восприятие и долгосрочная память тесно связаны, и с этой точки зрения никакой тайны в скорости распознавания лиц нет. Когда мы видим лицо, в мозге сразу же активируются миллионы нейронов, входящие во множество шаблонов. Отдельные нейроны и группы нейронов, составляющие шаблон, отзываются на определенные характеристики лица и контекст, в котором в мозг поступило это изображение. Разные лица стимулируют разные шаблоны нейронной реакции. Если мы уже видели определенное лицо раньше, значит, соответствующий нейронный шаблон уже когда-то активировался. Повторное поступление информации об этом лице активирует тот же шаблон нейронной активности, причем это происходит чуть проще, чем раньше. В этом суть узнавания. Не нужно ничего *искать* в долгосрочной памяти: новое впечатление повторно активирует тот же (возможно, с небольшими отличиями) шаблон нейронной активности, что и в прошлый раз. Повторная активация шаблона — это повторная активация долгосрочной памяти.

На компьютерном жаргоне мы могли бы сказать, что информация в человеческой долгосрочной памяти *адресуется по содержимому*, однако слово «адресуется» ошибочно подразумевает, что каждое воспоминание хранится в определенной

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

точке мозга. В действительности каждому воспоминанию соответствует шаблон нейронной активности, охватывающий большую поверхность мозга.

Рис. 9.2. Как быстро вы узнали лица этих людей?[1]

Это объясняет, почему нам не нужно тратить много времени на то, чтобы понять, что лицо принадлежит незнакомому человеку. Если нам покажут портрет и спросят, видели ли мы раньше этого человека, мы не начинаем рыться в памяти, пытаясь вспомнить, не встречались ли с ним (рис. 9.3). Никакого поиска не происходит. Новое лицо активирует шаблон нейронной активности, который ранее никогда не активировался, поэтому узнавания не происходит. Разумеется, новое лицо может быть так похоже на лицо одного из наших знакомых, что сработает *ошибочное узнавание*, в то время как отдаленно похожее лицо запустит схожий шаблон, вызвав ощущение, что человек на данном портрете напоминает кого-то очень знакомого.

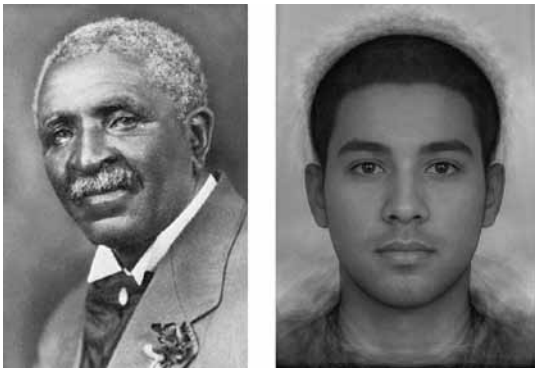


Рис. 9.3. Как быстро вы осознали, что не узнаете этих людей?[2]

[1] Барак Обама и Билл Клинтон.

[2] Джордж Вашингтон Карвер (George Washington Carver), американский ученый, педагог и изобретатель, и усредненное мужское лицо с FaceResearch.org.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Те же механизмы, благодаря которым визуальная система удивительно быстро распознает лица, позволяют нам быстро узнавать сложные узоры. Любой человек, получивший по меньшей мере среднее образование, мгновенно без труда узнает карту Европы и шахматную доску (рис. 9.4). Шахматисты, изучавшие историю шахматных чемпионатов, могут узнать даже показанную на доске позицию (Каспаров против Карпова, 1986).

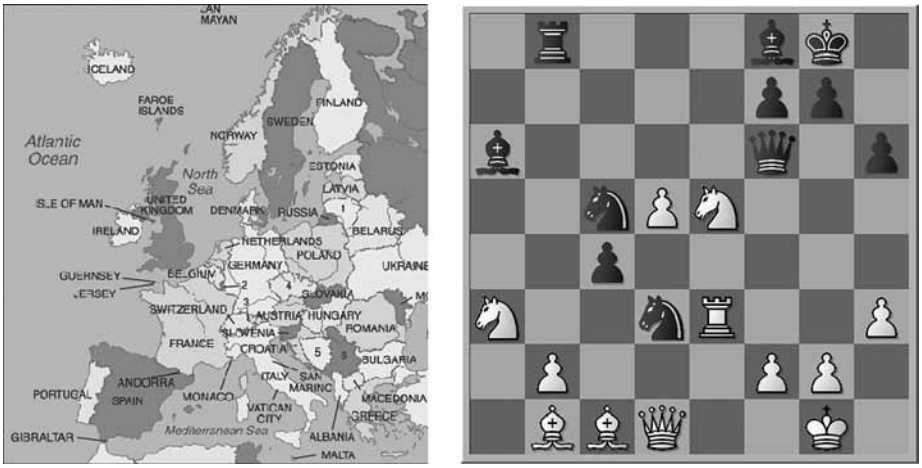


Рис. 9.4. Мы быстро распознаем сложные узоры

Вспоминать трудно

В противоположность сказанному ранее *вспоминание* представляет собой повторную активацию старых нейронных шаблонов без одновременного похожего перцепционного раздражения. Это намного сложнее, чем повторно активировать нейронный шаблон, содержащий те же или похожие впечатления. Люди *умеют* вспоминать, так что, очевидно, активность в других нейронных шаблонах или информация от других областей мозга *в состоянии* повторно активировать шаблон нейронной активности, соответствующий воспоминанию. Однако затрачиваемое время и усилия, необходимые для координации процесса, увеличивают вероятность того, что будет активирован неверный шаблон или лишь часть нужного шаблона, а это означает, что воспоминание вызвать не удастся.

Какими бы ни были эволюционные причины, наш мозг не научился быстро вспоминать факты. Многие школьники не любят уроки истории, потому что там от них требуют запоминать конкретные сведения, например когда была подписана Великая хартия вольностей, как называется столица Аргентины или все пятьдесят штатов США. Эта неприязнь не должна удивлять — человеческий мозг плохо подходит для решения задач такого рода.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Люди плохо умеют вспоминать и поэтому разрабатывают методы и техники, помогающие не забывать факты и процедуры (см. главу 7). В Древней Греции ораторы использовали метод локусов для запоминания основных идей длинной речи. Они воображали большое здание или площадь и мысленно помещали в различных ее точках основные идеи речи. Произнося речь, они в своем воображении «проходили» по выбранному месту, по пути собирая разложенные здесь и там идеи.

Сегодня для упрощения вспоминания мы больше полагаемся на внешние подсказки, чем на внутренние методы. Современные ораторы вспоминают основные идеи речи благодаря тому, что записывают их на листке бумаги, прозрачных слайдах или кадрах презентации и просматривают записи во время выступления. Коммерсанты отслеживают, какими суммами денег они владеют, сколько они должны другим и сколько другие должны им, ведя бухгалтерские книги. Для хранения контактных данных друзей и родственников мы используем записные книжки. Вспоминать о встречах, днях рождения, юбилеях и других значительных событиях нам помогают календари и будильники. Электронные календари лучше всего справляются с задачей отслеживания встреч, так как они активно напоминают о запланированных событиях — нам не приходится помнить о том, что следует периодически сверяться с ними.

Узнавание или вспоминание: почему они важны для разработчиков пользовательских интерфейсов

Относительная легкость, с которой мы узнаем вещи, и сложности с вспоминанием лежат в основе проектирования пользовательских интерфейсов (Johnson et al., 1989). Два главных правила дизайна пользовательских интерфейсов известны всем:

- **Видеть и выбирать проще, чем вспоминать и вводить.** Показывайте пользователям варианты и позволяйте делать выбор. Не заставляйте их вспоминать возможные варианты и рассказывать компьютеру, чего же они хотят. Это правило объясняет, почему почти повсеместно в компьютерном мире графические интерфейсы пользователя пришли на смену интерфейсам на основе командной строки (рис. 9.5). «Узнавание, а не вспоминание» — это одно из известных эвристических правил Нильсена и Молича (Nielsen, Molich, 1990), применяющихся для оценки пользовательских интерфейсов. Однако управлять компьютерным приложением с помощью обычных языковых команд иногда получается выразительнее и эффективнее, чем в графическом интерфейсе. Таким образом, «вспоминание и ввод» остается весьма полезным подходом, особенно для случаев, когда пользователь с легкостью вспоминает, что ему нужно ввести, например критерии в текстовом поле поиска.
- **Когда возможно, описывайте функции с помощью изображений.** Люди очень быстро узнают картинки; кроме того, узнавание картинки стимулирует вспоминание связанной информации. Вот почему в современных пользовательских

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

интерфейсах функции часто описываются с помощью изображений (рис. 9.6 и 9.7): значков на рабочем столе и панелях инструментов, символов ошибки, списков с графическим оформлением вариантов. Картинки, в которых люди узнают объекты из реального мира, очень полезны тем, что пользователя не приходится учить распознавать их. Конечно, такое узнавание играет положительную роль, только если пользователь узнает объекты «правильно», то есть считывает в них смысл, вложенный разработчиками компьютерной системы (Johnson, 1987). При этом не так уж важно использовать изображения объектов, знакомых нам по физическому миру. Пользователи компьютеров быстро учатся ассоциировать новые значки и символы с подразумеваемым смыслом при условии, что эти графические элементы хорошо проработаны. Запоминающиеся значки и символы намекают на зашифрованный в них смысл, их легко отличить от остальных, и они всегда означают одно и то же, даже если используются в разных приложениях.

Вспоминание и ввод:

> copy doc1 doc2

> remove olddoc

Узнавание и выбор

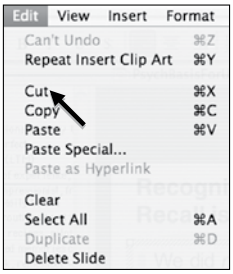


Рис. 9.5. Основное правило дизайна современных графических интерфейсов: видеть и выбирать проще, чем вспоминать и вводить

Рис. 9.6. Узнавая значки на рабочем столе по аналогии с физическими объектами или исходя из опыта, человек угадывает связанные с ними функции

Рис. 9.7. На приборной панели (Dashboard) веб-приложения Wordpress.com для обозначения функциональных страниц используются и символы, и текст

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Идея графического интерфейса пользователя возникла в середине 1970-х годов и получила широкое распространение десятилетия спустя — в 1980–90-х. С тех пор было сформулировано еще много рекомендаций по разработке интерфейсов, основанных на человеческом восприятии в целом и на узнавании и вспоминании в частности. В завершение главы расскажу о некоторых из этих новых правил.

Используйте эскизы для компактного представления полноразмерных изображений

Функция узнавания человеческого мозга почти нечувствительна к размеру объектов — в конце концов, мы должны уметь узнавать вещи независимо от того, на каком расстоянии они находятся.

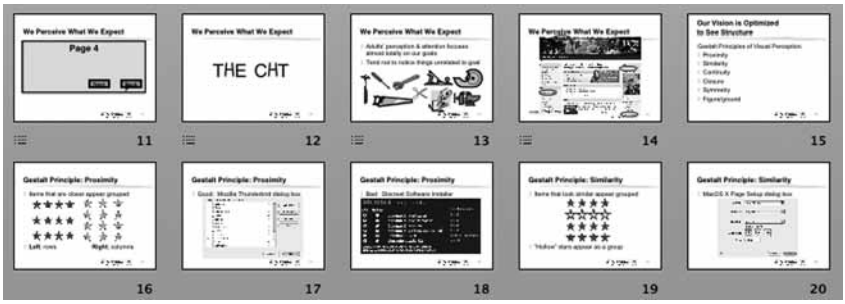


Рис. 9.8. Приложение Microsoft PowerPoint умеет показывать слайды в виде набора эскизов; общее впечатление о коллекции создается за счет узнавания

Рис. 9.9. Apple Safari показывает пользователю эскизы недавно посещенных страниц для быстрого выбора на основе узнавания

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Самое главное — это характеристики. Если большинство характеристик новой картинки совпадают с характеристиками старой, то новое впечатление активирует старый нейронный шаблон и узнавание происходит.

Таким образом, набор эскизов, то есть уменьшенных копий картинок — это отличный способ представления изображений, которые люди *уже* видели. Чем более знакома человеку картинка, тем меньше можно сделать ее эскиз, и человек все равно узнает ее. Видя перед собой набор эскизов, а не полноразмерных изображений, мы лучше ориентируемся в возможностях выбора, в представленных данных, истории изображений и т. п., причем все это одновременно.

В приложениях для управления фотографиями и их просмотра представление на основе эскизов позволяет создать общее впечатление о коллекции изображений или слайдов (рис. 9.8). В веб-браузерах в виде эскизов отображаются недавно посещенные пользователем страницы (рис. 9.9).

Чем больше людей будут использовать функцию, тем заметнее она должна быть

По причинам, описанным ранее, попытка вспомнить что-то часто завершается неудачей. Если в приложении важная функциональность прячется где-то глубоко и пользователям приходится вспоминать, как выполнить ту или иную задачу, то у определенного процента людей это получается не каждый раз или вовсе не получается. Чем больше пользователей работают с приложением, тем многочисленнее становится группа тех, кто испытывает трудности с вспоминанием, даже если процент их невелик. Очевидно, что разработчикам программного обеспечения не слишком хочется терять эту многочисленную группу.

Решение данной проблемы заключается в том, чтобы сделать востребованные функции хорошо заметными — они всегда должны быть на виду. Тогда пользователи смогут в любой момент увидеть доступные варианты и *узнать* среди них нужные, то есть им не придется утруждать себя *вспоминанием*. И наоборот, функции, необходимые лишь небольшому числу пользователей, особенно если речь идет о профессионалах, можно прятать, например, на панелях детализированных сведений, в контекстных меню или за специальными сочетаниями клавиш.

Добавляйте визуальные подсказки, чтобы пользователи могли ориентироваться

Визуальное узнавание происходит быстро и надежно, поэтому в интерфейс следует встраивать визуальные подсказки, по которым пользователи смогут мгновенно ориентироваться и понимать, где же они находятся. Например, хорошо известное правило веб-дизайна гласит, что все страницы веб-сайта должны быть объединены общим узнаваемым визуальным стилем. Благодаря общему стилю человек

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

сразу понимает, находится он еще на старом сайте или уже перешел на другой. Незначительные, но систематические вариации визуального стиля сайта могут подсказывать пользователю, в каком разделе тот оказался.

Некоторые настольные операционные системы позволяют настраивать разные рабочие столы («кабинеты» или «рабочие места») для разных категорий рабочих объектов. На каждом отображается собственный фоновый рисунок, что способствует быстрому узнаванию.

На некоторых корпоративных веб-сайтах используются изображения, подтверждающие, что пользователи находятся на защищенном ресурсе. Пользователь устанавливает понравившуюся ему картинку в качестве логотипа своей учетной записи, и этот логотип отображается, когда сайт узнает пользователя (по файлам cookie или после ввода правильного имени пользователя, но еще не пароля (рис. 9.10)). Таким образом, человек понимает, что это реальный сайт компании, а не подделка, с помощью которой злоумышленники планируют выкрасть его личные данные.

Рис. 9.10. После того как веб-сайт BankOfAmerica.com узнает пользователя, там отображается выбранный самим пользователем логотип учетной записи (ключ сайта (SiteKey)), что подтверждает реальность и надежность сайта

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Помогайте пользователям вспоминать данные для аутентификации

Известно, что людям сложно запоминать произвольные факты, слова и последовательности букв или цифр. Вот почему они часто записывают пароли и ответы на контрольные вопросы и хранят эту информацию в легкодоступных ненадежных местах. Или же выбирают пароли, основанные на именах их детей, дате рождения, домашнем адресе или другой информации, которую нетрудно вспомнить. К сожалению, подобные пароли довольно легко поддаются взлому — другой человек может их попросту угадать (Schrage, 2005). Каким образом разработчик может помочь пользователю избежать такого небезопасного поведения?

Для начала не нужно *усложнять* пользователям задачу по вспоминанию реквизитов. Вспомните системы, описанные в главе 7, которые накладывали обременительные ограничения на пароль или предлагали ограниченный набор контрольных вопросов.

Гораздо лучше дать пользователю свободу. Пусть он сам придумывает пароль, который сможет запомнить, или контрольный вопрос, ответ на который будет уникальным. Можно также предложить пользователю добавить *подсказку* о пароле, которая будет храниться в системе и выводиться по мере необходимости. Разумеется, это возможно при условии, что пользователи в состоянии изобрести подсказку, дающую им достаточно информации, но не позволяющую посторонним угадать пароль.

Возможно, наилучшим выходом будут способы аутентификации, не зависящие от способности пользователя вспоминать учетные данные. В эту категорию попадают биометрические аутентификационные методы, такие как сканирование сетчатки глаза и отпечатков пальцев, а также идентификация голоса. Однако многие люди считают подобные методы угрозой безопасности, так как они подразумевают сбор и хранение биометрических данных в окружении, не защищенном от утечек и злонамеренного использования информации. Таким образом, хотя биометрическая аутентификация не нагружает память человека, она должна быть реализована с учетом строжайших требований к конфиденциальности. Только в этом случае возможно ее широкое использование.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 10

Учиться на своем опыте легко, решать задачи сложно

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Как мы узнали в предыдущей главе, в которой сравнивали узнавание и вспоминание, человеческий мозг одни действия выполняет хорошо, а другие — не очень. В этой главе мы сравним еще несколько функций мозга, для того чтобы лучше понять его сильные и слабые стороны, и посмотрим, какое значение они имеют для дизайна компьютерных систем. Однако сначала немного поговорим о мозге и разуме.

У нас три мозга

В действительности у нас три мозга, или, если хотите, мозг, состоящий из трех основных частей, каждая из которых отвечает за определенные аспекты мышления и поведения (Weinschenk, 2009):

- **старый мозг**: это, фактически, ствол мозга — в месте, где спинной мозг переходит в основание головного мозга. Он появился еще у первых рыб (у насекомых и моллюсков, которые развились раньше рыб, нет мозга в привычном смысле этого слова). Старый мозг разбивает все окружающие объекты на три категории: «можно есть», «опасно» и «сексуально привлекательно». Он также регулирует автоматические функции тела, такие как пищеварение, дыхание и рефлексивные движения. У рептилий, амфибий и большинства рыб есть только старый мозг;
- **средний мозг**: эта часть мозга называется средней по двум причинам: потому что физически она расположена над старым мозгом, но под корой головного мозга, и потому что развилась она позже старого мозга, но раньше нового. Средний мозг управляет эмоциями, он реагирует на вещи и события радостью, огорчением, страхом, агрессией, опасением, гневом и т. п. У птиц[1] и низших млекопитающих есть только старый мозг и средний мозг;
- **новый мозг**: эта часть мозга состоит в основном из коры головного мозга. Она управляет умышленной, намеренной, сознательной деятельностью, включая планирование. У большинства млекопитающих помимо старого и среднего мозга также есть кора головного мозга, однако значительного размера она достигает только у некоторых высокоразвитых животных — слонов, белух, дельфинов и китов, а также приматов, в том числе обезьян и людей.

Человеческий разум нельзя считать абсолютно рациональным и сознательным, более того, некоторые эксперты утверждают, что иррационального и бессознательного в нем куда больше половины. Средний мозг и старый мозг в такой же степени, если не сильнее, влияют на наши мысли и поведение, что и новый мозг. Когда мы видим какой-то объект или рядом с нами происходит определенное событие,

[1] У врановых (ворон, ворона, сорока) и некоторых типов попугаев (например, новозеландского кеа) нет коры головного мозга, но, по сравнению с другими птицами, очень крупный мозг. Они зачастую демонстрируют признаки интеллекта, сравнимого с интеллектом слонов, дельфинов и обезьян. У этих птиц функции, которые в мозге млекопитающих выполняет кора головного мозга, очевидно, обрабатываются в других его частях.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

все три мозга «реагируют» на него и вносят свой вклад в ход нашего мышления и ответное поведение. На самом деле старый мозг и средний мозг реагируют даже быстрее, чем новый мозг, поэтому иногда мы начинаем действовать, основываясь на их приказах, причем еще до того, как кора головного мозга примет решение или хотя бы осознает, что необходимо действовать.

Учиться на опыте (обычно) легко

Люди чаще всего неплохо справляются с обобщением: на основе определенного опыта и наблюдений мы успешно делаем выводы. Обобщением мы занимаемся постоянно на протяжении всей жизни.

Пока что ученым не удалось так же глубоко вникнуть в нейронные основы поведенческого обучения, как в процессы узнавания и вспоминания (Liang et al., 2007). Однако люди все время учатся на собственном опыте, зачастую даже не осознавая этого. Исходя из этого факта, можно предположить, что человеческий мозг развил умение быстро и легко учиться на опыте, потому что это давало ему эволюционные преимущества. Таким образом, большинство людей, получив соответствующий опыт, без труда усваивают уроки, подобные следующим:

- держаться подальше от леопардов;
- не есть еду, которая плохо пахнет;
- мороженое вкусное, но в жаркую погоду быстро тает;
- подождать сутки, прежде чем отвечать на электронное сообщение, которое тебя разозлило;
- не открывать вложения в электронных письмах от незнакомых людей;
- LinkedIn — полезный сервис, а на Facebook только попусту теряешь время (или наоборот, в зависимости от ваших предпочтений).

Однако наше умение учиться на опыте несовершенно по нескольким причинам. Во-первых, сложные ситуации, включающие множество переменных или подверженные действию разнообразных сил, трудно предсказывать, сложно также извлекать из них уроки и обобщать их, например:

- опытные инвесторы фондового рынка все еще не уверены, какие акции нужно продавать и покупать прямо сейчас;
- люди, много лет прожившие в Денвере, до сих пор с трудом предсказывают местную погоду;
- пообщавшись с бойфрендом сестры в разных ситуациях, вы, тем не менее, не можете сказать, хороший ли он человек.

Во-вторых, жизненный опыт — наш собственный или опыт родственников и друзей — влияет на наши выводы намного сильнее, чем то, о чем мы читали или

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

слышали. Например, даже если мы много раз читали и видели репортажи, потребительские обзоры и статистику, подтверждающую, что Toyota Prius — отличный автомобиль, мы все равно составим о нем негативное впечатление, если сестра или дядя расскажут, что недовольны машиной. Дело в том, что наш средний мозг считает членов семьи похожими на нас, а, следовательно, полученные от них сведения — более достойными доверия, чем данные от тысяч анонимных покупателей машин, даже если с рациональной точки зрения статистика намного надежнее (Weinschenk, 2009).

В-третьих, когда человек делает ошибку, он не всегда извлекает из нее правильный урок. К тому времени, как он понимает, что оказался в неудачном положении, он уже может не помнить предыдущие действия. Следовательно, не связывает ситуацию с истинной причиной или причинами неудачи.

Четвертая проблема обучения на основе опыта заключается в том, что люди зачастую чрезмерно обобщают, то есть делают обобщения, базируясь на неполных данных. Например, человек считает, что все вороны черные, потому что он никогда в жизни не видел ворон другого цвета. В действительности же существуют не черные вороны (рис. 10.1).

Рис. 10.1. Распространенное мнение о том, что все вороны черные, ошибочно: *слева* — африканский пегий ворон; *справа* — белая ворона (не альбинос), штат Огайо

Однако некоторые утверждают, что чрезмерное обобщение — это не проблема, а *особенность*. Почти невозможно увидеть все возможные примеры того или иного явления. Например, человек не сможет увидеть всех ворон в мире. Тем не менее в повседневной жизни (но не в научных исследованиях) вполне допустимо предполагать, что все вороны черные, исходя из того, что встречал огромное количество черных ворон, но не ворон другой окраски. Таким образом, чрезмерное обобщение кажется необходимым инструментом адаптации к реальному миру. Проблемы возникают в основном тогда, когда мы доводим чрезмерное обобщение до экстремальных высот, то есть основываемся в своих выводах на *одном* примере либо *нетипичных* примерах.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

У умения учиться на опыте длинная эволюционная история. Для того чтобы уметь делать это, животному не требуется кора головного мозга (новый мозг). Старый мозг и средний мозг умеют учиться на опыте. Даже насекомые, моллюски и черви, не обладающие старым мозгом — только несколькими нейронными кластерами, — учатся на опыте. Однако только обладатели коры головного мозга или мозговых структур, выполняющих аналогичные функции[1], могут учиться на *чужом* опыте. Определенно, без коры головного мозга невозможно осознать, что научился чему-то на своем или чужом опыте, и только животные с самым большим новым мозгом (по отношению к размеру тела) — возможно, только люди — способны сформулировать, чему конкретно они научились из опыта.

Несмотря на то что обучение на основе собственного или чужого опыта имеет определенные ограничения, главное, что вы должны запомнить, — обучение и обобщение на базе опыта происходит в человеческом разуме относительно просто.

Выполнять заученные действия просто

Отправляясь куда-то, где мы бывали много раз, или делая что-то, что мы уже неоднократно повторяли раньше, мы действуем без сознательного обдумывания и оценки действий. Маршрут, рутина, рецепт, процедура, действие становятся полуавтоматическими или совершенно автоматическими. Вот несколько примеров:

- езда на велосипеде после многолетней практики;
- выезд на автомобиле из двора и поездка на работу в трехсотый раз;
- чистка зубов (у взрослых);
- воспроизведение на музыкальном инструменте мелодии, которую вы играли сотни раз;
- использование мышки или сенсорной панели для перемещения курсора на экране компьютера после нескольких дней практики;
- запуск банковской транзакции в знакомом приложении электронного банка;
- чтение и удаление текстового сообщения на мобильном телефоне, которым вы владеете уже несколько месяцев.

Фактически, автоматическим действием в когнитивной психологии называется рутина, любое хорошо заученное поведение (Schneider, Shiffrin, 1977). Исследователи доказали, что при выполнении действий такого типа либо потребляется совсем немного когнитивных ресурсов, либо они вовсе не задействуются. Таким образом, на автоматическую деятельность не распространяются ограничения внимания и краткосрочной памяти, описанные в главе 7.

Автоматическая деятельность может даже осуществляться параллельно с другой активностью. Вы можете постукивать ногой по полу, напевать знакомую мелодию

[1] Некоторые птицы способны обучаться, наблюдая за другими птицами.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

и одновременно разбивать яйцо, однако ваш разум будет достаточно свободен, чтобы дополнительно вы могли присматривать за ребенком или планировать отпуск.

Как же новая деятельность превращается в автоматическую? Так же, как вы попадаете в «Карнеги-холл» (вспомните старую шутку): практика, практика и еще раз практика.

Когда человек впервые садится за руль автомобиля, особенно оборудованного механической коробкой передач, каждая составляющая его активности требует сознательного внимания. Правильная ли передача включена? Какой ногой нажимать на педаль газа, педаль тормоза и педаль сцепления? Как сильно нажимать на каждую из этих педалей? Как сильно я нажимаю на педаль сцепления в данный момент? В какую сторону я еду? С какой скоростью я еду? Что находится впереди, позади, рядом? Где зеркала, в которые мне надо периодически поглядывать? Впереди виднеется моя улица? «Объекты в зеркале ближе, чем кажутся» — что это означает? И что это за огонек мигает на приборной панели?

Емкости внимания *абсолютно* недостаточно для того, чтобы человек мог сознательно отслеживать все элементы, составляющие деятельность по вождению автомобиля, — вспомните, что в фокусе внимания может быть всего четыре элемента, плюс-минус два (см. главу 7). Из-за этого ученики автошкол зачастую чувствуют себя так, словно взялись за неподъемную задачу. Вот почему они нередко уезжают тренироваться водить машину на большие парковки, за город, в тихий двор, где мало людей и автомобилей, для того чтобы снизить количество вещей, которые необходимо держать в памяти.

В результате продолжительной практики все действия, составляющие вождение автомобиля, становятся автоматическими. Они больше не ведут борьбу за наше внимание и удаляются из области сознания. Мы даже не всегда осознаем, что производим их. Например, какой ногой вы нажимаете на педаль газа? Чтобы вспомнить, вам, возможно, придется на мгновение осознанно прижать педаль.

Точно так же учитель музыки не заставляет ученика отслеживать и контролировать все аспекты игры на музыкальном инструменте одновременно. Это перегрузило бы внимание ученика. Вместо этого учитель узко фокусирует его внимание на одном или двух возможных аспектах игры: правильные ноты, ритм, интонация, артикуляция, темп. Только после того как ученик овладевает некоторыми из аспектов настолько хорошо, чтобы выполнять их автоматически, не задумываясь, учитель начинает требовать от него одновременно контролировать большее число действий.

Для того чтобы продемонстрировать самим себе, сколько сознательного внимания требуется для выполнения хорошо знакомых (автоматических) и, наоборот, новых (контролируемых) задач, попробуйте сделать следующее:

- перечислите буквы алфавита от А до М. Затем перечислите их в обратном порядке, от М до А;

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- посчитайте от 10 до 0, как перед запуском космической ракеты. Затем посчитайте нечетные числа от 21 до 1;
- поезжайте на работу по обычному маршруту. На следующий день попробуйте совсем другой, незнакомый маршрут;
- киньте мяч той рукой, которой вы обычно это делаете. Затем попробуйте бросить мяч другой рукой;
- наберите свой номер телефона на стандартной 12-клавишной панели телефона. Теперь наберите его с помощью цифровых клавиш наверху компьютерной клавиатуры;
- введите свое полное имя на клавиатуре компьютера. Затем скрестите руки и еще раз попробуйте ввести его. (Сначала я хотел предложить вам попробовать прокатиться на велосипеде со скрещенными руками, но это действительно опасно, так что *не рекомендую* делать этого.)

Большинство задач, с которыми мы сталкиваемся в реальном мире, состоят из смешения автоматических и контролируемых компонентов. Когда вы едете на работу по обычному маршруту, это происходит почти автоматически, и вы можете сфокусироваться на новостях, транслируемых по радио, или подумать о том, что собираетесь съесть на ужин. Но если другой автомобиль рядом с вами совершит неожиданный маневр или на дорогу выбежит ребенок, то ваше внимание моментально возвратится к задаче вождения.

Аналогично, проверяя почту в привычном почтовом клиенте, вы загружаете и просматриваете сообщения хорошо отрепетированным и почти автоматическим способом. Чтение текста — это тоже хорошо знакомая автоматическая деятельность. Однако содержимое каждого очередного письма для вас в новинку, поэтому требует сознательного внимания. Предположим, находясь в отпуске, вы заходите в интернет-кафе и проверяете почту на незнакомом компьютере в незнакомой операционной системе и незнакомом почтовом клиенте. В этом случае автоматически выполняется намного меньшая часть задания. Следовательно, вам приходится сознательно обдумывать процесс, это занимает больше времени, а вероятность ошибки возрастает.

Когда перед человеком стоит задача сделать что-то — в противоположность желанию решить интеллектуальную проблему, — он выбирает автоматические или полуавтоматические методы, для того чтобы сэкономить время и умственные усилия, а также снизить вероятность ошибки. Если вы торопитесь забрать ребенка из школы, то поедете за ним по много раз испытанному маршруту, даже если сосед вчера рассказал вам о новом коротком пути. Вспомните, что сказал один из моих испытуемых во время тестирования удобства в использовании (я упоминал об этом в главе 8): «Я тороплюсь, так что пойду по длинному пути».

Что нужно делать дизайнерам интерактивных систем, для того чтобы задачи в их системах выполнялись быстрее, проще и с минимальным числом ошибок? Нужно разрабатывать их таким образом, чтобы все действия пользователя быстро пере-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ходили на автоматический уровень. А как этого добиться? О некоторых способах рассказывается в главе 11.

Решать задачи и вычислять трудно

Рептилии, амфибии и большинство птиц счастливо живут в своем мире, обладая лишь старым мозгом и средним мозгом[1]. Насекомые, пауки и моллюски выживают в своей среде обитания с еще более скудным мозговым арсеналом. Животные, у которых отсутствует кора головного мозга (или ее эквивалент, который можно встретить у некоторых птиц), умеют учиться на опыте, однако чаще всего для этого требуется очень большое количество повторений, а изменения в поведении оказываются незначительными. Стоит разобраться в требованиях среды обитания того или иного животного — и становится ясно, что его поведение шаблонно, повторяемо и предсказуемо (Simon, 1969). И это нормально, если для выживания во внешних условиях не требуется ничего выходящего за пределы автоматизированного поведения.

Но что происходит, если среда обитания застает животное врасплох, требуя нового поведения — *здесь и сейчас*? Что, если существо оказывается в ситуации, с которой раньше никогда не сталкивалось и, возможно, не столкнется в будущем? Короче, что ему делать при обнаружении *проблемы*? В таких ситуациях существа, не обладающие корой головного мозга или ее эквивалентом, пасуют.

Благодаря наличию коры головного мозга (нового мозга) животное освобождается от необходимости полагаться исключительно на инстинктивное, рефлекторное, автоматическое и хорошо отрепетированное поведение. Именно в коре мозга происходит сознательное мышление (Monti, Osherson, Martinez, Parsons, 2007). В целом, чем больше кора головного мозга животного по отношению к остальной его части, тем более у него развита способность интерпретировать и анализировать ситуации на лету, а также планировать или находить стратегии и процедуры, помогающие справляться с тупиковыми ситуациями, исполнять эти стратегии и процедуры и контролировать ход исполнения.

Выражаясь компьютерным языком, большая кора головного мозга обеспечивает нам возможность на лету изобретать для самих себя программы и исполнять их в эмулированном, хорошо контролируемом режиме, а не в скомпилированном или «родном». Фактически, именно это мы делаем, когда готовим что-то по рецепту, играем в бридж, подсчитываем налоги на прибыль, следуем инструкциям в руководстве пользователя по приложению или отыскиваем причину отсутствия звука на компьютере при воспроизведении видео.

---

[1] Например, если предложить саламандре чашки с двумя, тремя или четырьмя плодовыми мушками внутри, она выберет ту, где мушек больше (Sohn, 2003).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

НОВЫЙ МОЗГ ТАКЖЕ СЛУЖИТ ТОРМОЗОМ ИМПУЛЬСИВНОГО ПОВЕДЕНИЯ

Одна из функций нового мозга — конкретнее, лобной коры головного мозга — подавлять рефлективное и импульсивное поведение, инициируемое средним мозгом и старым мозгом. Это делается, когда подобное поведение мешает исполнению тщательно проработанных планов нового мозга (Sapolsky, 2002). Благодаря ему мы не вскакиваем с места и не выпрыгиваем из вагона метро, когда заходит плохо пахнущий человек, потому что, в конце концов, нам нужно успеть на работу вовремя. Лобная кора заставляет нас тихонько сидеть в кресле во время концерта классической музыки, но позволяет вставать, кричать и завывать на рок-концерте. Она помогает нам не ввязываться в драки (чаще всего). Она пытается запретить нам покупать вон ту красную спортивную машину, потому что сохранение брака — более приоритетная цель, чем владение спортивным автомобилем. И хотя старый мозг и средний мозг моментально соблазняются электронными сообщениями с заголовками вроде «Открой свой бизнес прямо сейчас — уникальная возможность заработать 12,5 миллиона долларов», новый мозг запрещает нам открывать такие письма, уговаривая: «Это спам или даже письмо с вирусом, ты же это и сам знаешь!»

Мы выяснили, что наличие большого нового мозга обеспечивает нам достаточную для оперативного решения возникающих проблем гибкость мышления, однако у всего есть своя цена, в том числе и у такой гибкости. Учиться на опыте и исполнять хорошо заученные действия легко в основном потому, что для этого не требуется постоянно сохранять бдительность и фокусировать внимание, а также потому, что мы можем одновременно делать несколько автоматических дел. В противоположность этому контролируемая обработка информации, включая решение задач и вычисления, требует фокусировки внимания и постоянного сознательного мониторинга; кроме того, задачи решаются поочередно, а не одновременно, и относительно медленно (Schneider, Shiffrin, 1977). Наша краткосрочная память напрягается до предела, потому что порциям информации, необходимым для исполнения конкретной процедуры, приходится бороться между собой за скудные ресурсы внимания. От нас требуются непрерывные ментальные усилия, в чем вы убедились, когда перечисляли буквы алфавита от М до А в обратном порядке.

На компьютерном языке можно сказать, что человеческий мозг оснащен только одним последовательным процессором для режима эмуляции, то есть для контролируемого выполнения процессов. Емкость временного хранилища этого процессора сильно ограничена, а тактовая частота на порядок меньше, чем у высокопараллельного процессора, предназначенного для скомпилированной автоматической обработки.

Превращение первых гоминидов в современных людей произошло между 200 000 и 50 000 лет назад, однако чисел и цифровых расчетов не существовало приблизительно до 3400 года до н. э., когда жители Месопотамии (территория современного Ирака) изобрели числа и начали использовать числовые расчеты в торговле. Их мозг уже почти не отличался от мозга современного человека. И поскольку человеческий мозг развился до нынешнего состояния до того, как были изобретены цифровые расчеты, он не оптимизирован для вычислений.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Вычисления выполняются в основном в контролируемом, управляемом режиме. Эти задачи потребляют все скудные ресурсы внимания и краткосрочной памяти, поэтому нам так сложно делать вычисления только в уме, не пользуясь подручными средствами. Исключение составляют лишь некоторые шаги вычислений, которые можно заучить и, следовательно, превратить в автоматические действия. Например, весь процесс умножения 479 на 832 — сознательный и контролируемый, однако определенные подшаги процесса могут выполняться автоматически, если мы хорошо запомнили таблицу умножения для одноразрядных чисел.

Для большинства людей не составляет труда решить задачи и сделать вычисления, если последние обладают следующими характеристиками: состоят не более чем из одного-двух шагов, включают автоматические (то есть хорошо отрепетированные) шаги или охватывают совсем немного информации. Также мы с легкостью справляемся с проблемами, для решения которых доступна вся необходимая информация, благодаря чему ее не требуется удерживать в краткосрочной памяти, например:

- $9 \cdot 10 = ?$
- Мне необходимо вынести стиральную машину из гаража, но путь перегораживает автомобиль, ключи от которого у меня в кармане. Что делать?
- У моей подруги два брата, Боб и Фред. Я уже встречался с Фредом, и сейчас передо мной не Фред, значит, это Боб.

Однако проблемы, не укладывающиеся в ограничения нашей краткосрочной памяти или требующие извлечения определенной информации из долгосрочной памяти, напрягают наш мозг точно так же, как задачи, во время решения которых нас что-то отвлекает, например:

- Мне необходимо вынести стиральную машину из гаража, но путь перегораживает автомобиль, ключи от которого... хм... не в кармане. Где же они?.. (Поискать в автомобиле.) Они не в машине. Может быть, я оставил их в куртке... А где же куртка? (Поискать в доме, найти куртку в спальне.) Отлично, ключи нашлись... Кошмар, какой в спальне беспорядок — нужно убрать, прежде чем жена вернется домой... Хм-м-м... А зачем я искал ключи от автомобиля? (Снова пойти в гараж, увидеть стиральную машину.) Ах, да, чтобы выгнать автомобиль на улицу и вынести из гаража стиральную машину. (*Промежуточные подцели выталкивали высокоуровневую цель из краткосрочной памяти.*)
- В главе 8 я приводил примеры задач, в которых людям приходится обращать внимание на завершающие шаги («прибирать за собой») после достижения основной цели, например выключать фары автомобиля после того, как вы прибыли в место назначения, или вынимать последнюю страницу из копира после того, как она скопирована.
- Кот Джона — не черный, и он любит молоко. Кот Сью — не коричневый и не любит молоко. Кот Сэма — не белый и не любит молоко. Кот Мэри — не рыжий и любит молоко. Человек нашел на улице рыжего кота, который любит молоко.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Чей это кот?[1]. (*Отрицания создают дополнительные порции информации, и большинство людей не может одновременно удержать в краткосрочной памяти все предоставленные данные.*)

- Человек построил дом. Окна на всех четырех стенах выходят на юг. Мимо проходил медведь. Какого он был цвета? (*Требуется применить дедукцию. Вы должны знать и быть способны извлечь из памяти определенные факты о мире и дикой природе.*)
- Вам необходимо отмерить ровно четыре литра воды, но у вас есть только трехлитровая и пятилитровая бутыли. Как решить задачу? (*Необходимо мысленно представлять себе различные последовательности переливаний до тех пор, пока не будет найдена верная; при этом напрягается краткосрочная память и, вероятно, превышаются возможности воображения.*)

При решении подобных задач люди зачастую разгружают свою память, прибегая к внешней помощи, например записывают промежуточные результаты, набрасывают схемы или создают модели проблем. Эти инструменты дополняют нашу ограниченную краткосрочную память и ограниченную способность манипулировать составляющими задачи в воображении.

Решение проблем и вычисления также представляют большую сложность, если требуют когнитивной стратегии, метода решения или процедуры, неизвестной нам или которую мы не можем найти или изобрести на ходу, например:

- $93{,}3 \cdot 102{,}1 = ?$ (*Требует выполнения арифметических действий, количество которых превышает емкость нашей памяти, поэтому приходится считать на калькуляторе или на бумаге. Второй вариант требует умения умножать многоразрядные десятичные дроби на бумаге.*)
- У фермера есть коровы и куры — всего 30 животных. Всего у животных 74 ноги. Сколько коров и сколько кур у фермера? (*Требует перевода словесной формулировки в систему из двух уравнений и решения их алгебраическими методами.*)
- Мастер дзен завязал глаза трем своим ученикам. Он сказал, что нарисует на лбу каждого либо синюю, либо красную точку. На самом деле он нарисовал каждому красную точку. Потом сказал: «Сейчас я сниму с вас повязки. После этого посмотрите друг на друга, и если увидите хотя бы одну красную точку, поднимите руку. Затем угадайте, какого цвета точка у вас на лбу». Мастер снял повязки, три ученика посмотрели друг на друга и каждый поднял руку. Через мгновение один из учеников сказал: «У меня красная точка». Как он догадался? (*Требует мышления от противного — это специальный метод, который изучают в курсах логики и математики.*)
- Вы запустили на компьютере видеоролик с веб-сайта YouTube, но звука нет, несмотря на то что у актеров в ролике шевелятся губы. В чем причина проблемы:

[1] Ответ вы найдете в конце главы.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

в самом ролике, проигрывателе, вашем компьютере, кабелях, которыми подключены динамики, или же динамиках? (*Необходимо изобрести и выполнить последовательность диагностических тестов, сужающих набор возможных источников проблемы. Для этого требуются знания компьютерной техники и электроники.*)

Эти выдуманные примеры демонстрируют, что для многих вычислений и решения определенных проблем необходимы знания, которыми обладают не все люди. Далее во врезке рассказывается о реальных ситуациях, в которых людям не удавалось решить технические проблемы: либо они не умели эффективно диагностировать их, либо им было попросту неинтересно этим заниматься.

РЕШЕНИЕ ТЕХНИЧЕСКИХ ПРОБЛЕМ ТРЕБУЕТ ЗАИНТЕРЕСОВАННОСТИ И СПЕЦИАЛЬНОГО ОБУЧЕНИЯ

Специалисты по программному обеспечению специально учатся систематически диагностировать проблемы. Умение придумать и исполнить последовательность тестов для проверки всех возможных причин неполадки и выявления истинной — часть их работы. Инженеры часто разрабатывают технологические продукты так, словно все пользователи этих продуктов настолько же подкованы в технической сфере, как и они сами. Однако большинство людей, если они не инженеры по образованию, никогда не изучали тонкости диагностики технических проблем и, следовательно, не умеют эффективно обнаруживать причины сбоев. Вот реальные примеры, когда люди без технического образования, столкнувшись с проблемами, не могли решить их без помощи специалиста:

- Энн хотела забронировать авиабилет, но у нее не получалось — веб-сайт не позволял выполнить операцию. Он требовал ввести пароль, но Энн не знала никакого пароля. Она позвонила другу-инженеру, и тот задал ей несколько вопросов, чтобы разобраться в ситуации. Выяснилось, что, с точки зрения веб-сайта, за компьютером работал муж Энн, который раньше уже покупал там авиабилеты. Сайт требовал ввода его имени пользователя и пароля. Энн не знала пароля, а муж был в командировке. Друг посоветовал ей выйти из системы на веб-сайте, а затем снова открыть сайт в качестве нового покупателя и завести собственную учетную запись.

- В церкви неожиданно прекратил работать один из двух напольных динамиков. Помощник режиссера предположил, что динамик сломался, и собрался заменить его. Музыканту, у которого было техническое образование, показалось, что динамик в порядке, поэтому он вынул кабели из разъемов в обоих динамиках и поменял их местами. После этого «сломанный» динамик заработал, а «хороший» перестал работать, то есть проблема действительно была не в сломанном динамике. Помощник режиссера сделал вывод, что порван один из кабелей и его нужно заменить. Но до того, как он сделал это, музыкант-инженер отключил кабели обоих динамиков от усилителя и также поменял их местами, чтобы проверить, не кроется ли причина проблемы в усилителе. Оказалось, что истинной причиной неполадки был разболтавшийся разъем на усилителе.

Даже когда человек знает, что *может* решить проблему или выполнить вычисление, иногда он этого не делает, потому что, по его мнению, игра не стоит свеч — потенциальный выигрыш не стоит затраченных усилий. Такая реакция особенно часто встречается в ситуациях, когда решение определенных проблем не относится к сфере обязанностей человека (например, не записано в его должностной инструкции). Вот несколько реальных примеров:

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- Объявление в сети Freecycle в Сан-Франциско: «Отдам даром Epson Stylus C86. Прекрасно работал, а потом перестал распознавать новый картридж, полный чернил. Не знаю, в чем проблема, в картридже или в принтере. Купил новый принтер и этот отдаю бесплатно».
- Фред и Элис — муж и жена. Он школьный учитель, а она медсестра. Они никогда не устанавливают и не обновляют программное обеспечение на своем домашнем компьютере. Они не знают, как это делается, и не хотят знать, поэтому используют только те программы, которые были установлены на компьютер при покупке. Если на экране появляется сообщение о том, что доступны обновления, они игнорируют его. Если приложение — например, веб-браузер — перестает работать из-за устаревшей версии, они прекращают использовать его. Когда возникает необходимость, они покупают новый компьютер.
- У другой пары, Теда и Сью, есть телевизор, видеомагнитофон и DVD-проигрыватель. Пульты дистанционного управления валяются рядом с телевизором, и ими никто не пользуется. Тед и Сью встают с дивана и подходят к устройствам каждый раз, когда возникает необходимость что-то переключить. Они говорят, что учиться пользоваться пультами слишком сложно, к тому же им трудно запомнить, какой пульт от какого устройства. При этом они ежедневно используют компьютер: отправляют электронную почту и просматривают страницы в Интернете.

Люди, описанные в этих примерах, не глупы. Многие из них закончили колледж, то есть находятся среди 30 % наиболее образованных людей в США. Некоторые даже изучали, как диагностировать проблемы в других сферах, например в медицине. Им просто не хватает знаний или неинтересно решать технические проблемы с компьютером или другими электронными устройствами.

Калькуляторы и компьютеры изобретались в основном как инструменты для выполнения вычислений и решения задач, с которыми люди не могут справиться самостоятельно. Эти устройства производят вычисления и решают задачи намного быстрее и надежнее, чем мы, по крайней мере, когда условия задач хорошо определены.

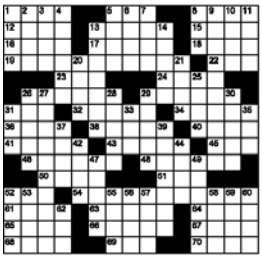
Выводы для дизайнера пользовательских интерфейсов

Люди часто занимаются тем, что создают себе сложности и устраняют их, а именно: придумывают и разгадывают головоломки, напрягающие мозг, помогающие упражнять его (рис. 10.2). Однако из этого не следует, что мы радуемся встрече с любыми нагружающими разум задачами, особенно если их перед нами ставят посторонние люди или объекты. У каждого человека есть собственные цели. Он использует компьютер для их достижения. Он хочет — и ему необходимо — фокусировать внимание исключительно на своих целях. Интерактивные системы и, разумеется, их разработчики должны с уважением относиться к этому. Недопустимо отвлекать пользователей от их целей, создавая у них технические сложности или навязывая им ненужные цели.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Вот несколько примеров технических проблем, которые компьютеры и веб-сервисы вынуждают своих пользователей решать вместо достижения основных целей:

- «От меня требуют идентификатор пользователя. Это то же самое, что и имя пользователя? Думаю, да».
- «Что? С меня списали полную цену! А где моя скидка? Что мне теперь делать?»
- «Тут пишут, что программа может быть несовместима со встраиваемым модулем, уже установленным на моем компьютере. Может быть? Так совместима или несовместима? И если все же несовместима, о каком модуле идет речь? И вообще, что мне делать?»
- «Мне нужно, чтобы номера страниц в этой главе начинались с 23-й, а не с 1-й, но я не вижу команды, которая позволила бы сделать это. Я пробовал Параметры страницы, Макет документа и Просмотреть колонтитулы, но там ничего подобного нет. Остается только команда Вставить номера страниц. Но я не хочу *вставлять* номера, ведь страницы в главе уже пронумерованы. Я всего лишь хочу поменять начальный номер».
- «Хм-м-м. Этот флажок подписан „Выровнять значки по вертикали“. Но что произойдет, если я его сброшу? Значки выровняются по горизонтали? Или к ним вообще никакое выравнивание применяться не будет?»



5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Рис. 10.2. Мы бросаем себе вызов, создавая и решая головоломки, нагружающие наши умственные способности

Интерактивные системы должны работать так, чтобы не оттягивать на себя внимание пользователя (Krug, 2005). В противном случае пользователю приходится делить ценные когнитивные ресурсы между задачей, которую он стремится выполнить с помощью компьютера, и системой, которая в идеальном случае должна незаметно помогать ему в этом. Вот несколько рекомендаций по разработке:

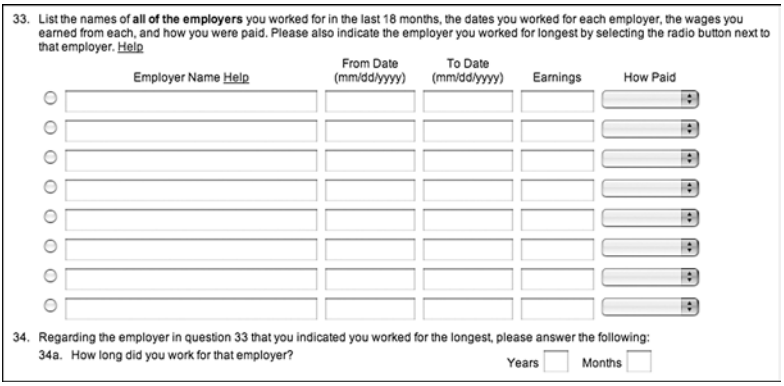
- **ясно обозначайте статус системы и прогресс пользователя на пути к его цели.** Если у пользователя есть возможность в любой момент одним взглядом проверить статус системы и задачи, то работа в системе не перенапрягает его внимание и краткосрочную память;
- **направляйте пользователей к их целям.** Это можно делать неявно, создавая в каждой точке выбора четкий информационный след, ведущий пользователя

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

к достижению цели, либо явно, предлагая для решения задачи многошаговое диалоговое окно (мастер). Если вы всего лишь выводите на экран набор вариантов, кажущихся одинаково привлекательными, то не стоит ожидать, что пользователь без труда поймет, с чего начать и как добраться до своей цели, особенно если речь идет о редко выполняемой задаче;

- **недвусмысленно и четко сообщайте пользователю все, что ему необходимо знать.** Не требуйте от пользователя домысливать информацию. Не заставляйте его докапываться до сути, по очереди отбрасывая неподходящие варианты;
- **не заставляйте пользователей диагностировать системные неполадки**, такие как ненадежное сетевое подключение. Для правильной диагностики требуется техническое образование, которого у большинства пользователей нет;
- **минимизируйте количество и сложность параметров.** Не заставляйте пользователей заниматься поиском оптимальных комбинаций множества взаимозависимых настроек и параметров. Люди с этим справляются *очень* плохо;
- **делайте так, чтобы люди воспринимали, а не вычисляли.** Некоторые задачи, на первый взгляд, требующие вычислений, можно представлять в графическом виде, позволяя пользователям достигать целей за счет быстрой перцепционной оценки, а не трудоемких вычислений. Простой пример: предположим, что вам необходимо перейти в середину документа. В 1970–1980-х годах использовались редакторы документов, в которых для этого приходилось сначала проверять длину документа, затем делить ее пополам и только после этого выполнять команду перехода к странице с определенным номером. В современных текстовых редакторах достаточно перетащить бегунок приблизительно в середину полосы прокрутки, находящейся у края окна, — и задача решена. Точно так же «магнитные» сетки и направляющие в графических редакторах избавляют пользователей от необходимости определять, сопоставлять и вычислять координаты существующих графических элементов, для того чтобы точно располагать новые;
- **делайте системы знакомыми.** Используйте концепции, терминологию и рисунки, уже знакомые пользователям. Благодаря привычным элементам новая система сразу кажется пользователям знакомой и им не приходится слишком много думать, изучая ее. До определенной степени к этому подходу можно прибегать даже в случаях с системами, которые предоставляют совершенно новую, не известную пользователям функциональность. Для создания легко узнаваемых систем рекомендуется: а) использовать условные обозначения и стандарты, принятые в отрасли (например, Apple Computer, 2009; Microsoft Corporation, 2009); б) делать новые приложения похожими по функциональности на старые, с которыми пользователи работали раньше; в) закладывать в основу дизайна метафоры, такие как метафора рабочего стола (Johnson et al., 1989). Наконец, разработчикам необходимо изучать своих пользователей и узнавать, что им знакомо, а что нет;
- **позволяйте компьютеру делать вычисления.** Не заставляйте людей производить расчеты, которые компьютер может с легкостью сделать за них (рис. 10.3).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks



33. List the names of **all of the employers** you worked for in the last 18 months, the dates you worked for each employer, the wages you earned from each, and how you were paid. Please also indicate the employer you worked for longest by selecting the radio button next to that employer. Help

Employer Name Help	From Date (mm/dd/yyyy)	To Date (mm/dd/yyyy)	Earnings	How Paid

34. Regarding the employer in question 33 that you indicated you worked for the longest, please answer the following:
34a. How long did you work for that employer? Years Months

Рис. 10.3. В этих двух вопросах интерактивная форма для оформления заявки на пособие по безработице запрашивает данные, которые компьютер мог бы вычислить самостоятельно

Ответы на головоломки

- Это кот Джона.
- Это был белый медведь, потому что дом, все стены которого выходят на юг, может находиться только на Северном полюсе.
- Для того чтобы получить четыре литра воды, наполните трехлитровую бутыль и вылейте воду в пятилитровую бутыль. Затем снова наполните трехлитровую бутыль и влейте столько, сколько влезет, в пятилитровую бутыль. В трехлитровой бутыли останется один литр воды. Теперь вылейте воду из пятилитровой бутыли и перелейте в нее один литр воды из трехлитровой бутыли. Теперь еще раз наполните трехлитровую бутыль и вылейте воду в пятилитровую бутыль.
- Предположим, что A — это число коров, а B — число кур. «У фермера есть коровы и куры — всего 30 животных» можно записать как $A + B = 30$. «Всего у животных 74 ноги» можно представить в виде формулы $4A + 2B = 74$. Решая уравнения относительно A и B, получаем: $A = 7$, $B = 23$. Таким образом, у фермера 7 коров и 23 курицы.
- Ученик дзен увидел, что руки подняли все трое, а также заметил красные точки на лбу остальных двух учеников. Этой информации было недостаточно, чтобы понять, синяя точка у него на лбу или красная. Он сделал предположение, что его точка синяя, и стал ждать. Он размышлял так: остальные два ученика увидят его (предположительно) синюю точку и еще одну красную точку, сделают вывод, что для того, чтобы все три руки были подняты, необходимы две красные точки, и быстро сообразят, что у них на лбу красные точки. Однако по прошествии нескольких мгновений ни один из этих двух учеников ничего не сказал, и наш ученик понял, что они не могут понять, какого цвета их точки. Следовательно, у него на лбу не синяя точка — она может быть только красной.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 11

На обучение влияет множество факторов

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В главе 10 автоматические процессы, с помощью которых наш мозг выполняет хорошо заученные действия, противопоставлялись сознательным, контролируемым и управляемым процессам, которые мы применяем для решения новых задач, а также вычислений. Автоматические процессы почти (или совсем) не потребляют ресурсы кратковременной памяти (внимания) и могут выполняться параллельно. В то же время контролируемые процессы предъявляют высокие требования к краткосрочной памяти и выполняются поочередно (Schneider, Shiffrin, 1977).

Первая или даже несколько начальных попыток выполнить определенное действие происходят осознанно, с тщательным контролем каждого шага. Однако по мере того как мы нарабатываем практику, действие становится все более и более автоматизированным. Примеры таких действий встречаются повсеместно каждый день: очистка яблока от кожуры, вождение автомобиля, жонглирование мячиками, езда на велосипеде, чтение, игра на музыкальном инструменте. Даже деятельность, казалось бы, требующая неусыпного внимания, такая как отделение хороших вишен от испорченных, может превратиться в автоматическую работу: мы начинаем выполнять ее как некую фоновую задачу, а когнитивные ресурсы высвобождаются, позволяя нам поддерживать разговор, следить за новостями по телевизору и т. п.

Такое превращение контролируемой деятельности в автоматическую ставит перед дизайнерами интерактивных приложений, интернет-служб и электронных устройств очевидный вопрос: как сделать так, чтобы использование этих продуктов переходило на автоматический уровень за разумное время?

В этой главе разъясняются и иллюстрируются факторы, влияющие на скорость обучения использованию интерактивных систем. Перечислю их, прежде чем переходит к подробному обсуждению. Мы учимся быстрее при выполнении следующих условий:

- деятельность сфокусирована на задаче, простая и единообразная;
- терминология сфокусирована на задаче, знакомая и единообразная;
- риск низок.

Мы быстрее учимся, когда наша деятельность сфокусирована на задаче, проста и единообразна

Когда мы применяем инструмент — неважно, компьютерный или нет — для выполнения задачи, нам приходится преобразовывать наши цели и желания в действия, поддерживаемые инструментом. Например:

- представьте, что вы астроном. Вам нужно направить телескоп на Альфу Центавра. Большинство телескопов не позволяют просто указать звезду, которую вы хотели бы понаблюдать. Вам приходится преобразовывать свою цель в параметры, на базе которых происходит позиционирование инструмента: вертикальный (азимут) и горизонтальный угол или, возможно, *разность* между *текущей* и *желаемой* позициями телескопа;

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- предположим, на вашем телефоне не запрограммированы кнопки быстрого набора. Для того чтобы позвонить знакомому, вам приходится переводить понятие «*знакомый*» в *телефонный номер* и вводить это значение;
- вы собираетесь нарисовать схему организационной структуры своей компании в обычном приложении для рисования. Для обозначения организаций, филиалов и руководителей вы рисуете прямоугольники, подписываете на них названия компаний и имена директоров и соединяете эти прямоугольники линиями.

Ученые, занимающиеся когнитивной психологией, называют разрыв между тем, какой инструмент требуется пользователю, и тем, какие операции умеет выполнять этот инструмент, ущельем исполнения (gulf of execution) (Norman, Draper, 1986). Человеку, желающему воспользоваться инструментом, приходится тратить когнитивные усилия, для того чтобы преобразовать свои желания в поддерживаемые операции, и наоборот. Эти когнитивные усилия отвлекают внимание человека от задачи и заставляют его фокусироваться на требованиях инструмента. Чем меньше ущелье исполнения между операциями, выполняемыми инструментом, и тем, чего желает пользователь, тем меньше ему приходится думать об инструменте и тем больше он концентрируется на своей задаче. В результате использование инструмента быстрее превращается в автоматическую деятельность.

Для того чтобы сокращать ущелье исполнения, инструменты следует проектировать так, чтобы поддерживаемые операции соответствовали намерениям пользователя. Попробуем дополнить предыдущие примеры:

- в управляющей системе телескопа можно хранить базу данных небесных тел, чтобы пользователь мог без особых усилий выбрать объект, который он желает понаблюдать, например, щелкнув на нем указателем мыши;
- телефоны с функцией быстрого набора позволяют просто выбирать человека или организацию из адресной книги, устраняя необходимость сначала преобразовывать эту цель в телефонный номер;
- в специальном приложении для редактирования организационных схем можно было бы просто ввести названия компаний и имена директоров, и оно само нарисовало бы прямоугольники и соединило их.

Если разработчик программного обеспечения, услуг и устройств хочет, чтобы операции, поддерживаемые его продуктами, соответствовали целям и задачам пользователей, он должен хорошо понимать, о каких конкретно целях и задачах идет речь. Сбор информации, необходимой для понимания, происходит в три этапа:

1. Проанализировать задачу.
2. Разработать сфокусированную вокруг задачи концептуальную модель, включающую в основном анализ объектов/действий.
3. Разработать пользовательский интерфейс, основанный строго на анализе задачи и концептуальной модели.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Анализ задачи

Невозможно в нескольких предложениях описать, как выполняется анализ целей и задач пользователя. Этому посвящаются целые главы — и даже целые книги (Beyer, Holtzblatt, 1997; Hackos, Redish, 1998; Johnson, 2007). Ограничусь тем, что перечислю вопросы, на которые обязательно должен дать ответ качественный анализ задачи:

- Каких целей желают добиться пользователи, используя это приложение?
- Какой набор «человеческих» задач должно поддерживать приложение?
- Какие задачи выполняются часто, а какие считаются редкими?
- Какие задачи наиболее, а какие наименее важны?
- На какие шаги подразделяется каждая задача?
- Каковы результаты и выходные данные каждой задачи?
- Откуда поступает информация для выполнения каждой задачи?
- Как используется информация, являющаяся результатом каждой задачи?
- Кто из людей какие задачи выполняет?
- Какие инструменты применяются для выполнения каждой задачи?
- Какие проблемы возникают у людей при выполнении каждой задачи? Какие ошибки они совершают чаще всего? Что становится причиной ошибок? Насколько они критичны?
- Какую терминологию используют люди, выполняющие эти задачи?
- Требуется ли для выполнения этих задач взаимодействовать с другими людьми?
- Как связаны между собой разные задачи?

Когда ответы на эти вопросы получены (путем наблюдения и/или опроса людей, выполняющих задачи, которые будет поддерживать данный инструмент), необходимо переходить к следующему шагу разработки. Но это *не* создание набросков возможных пользовательских интерфейсов. Сначала необходимо разработать концептуальную модель инструмента, сфокусированную на задачах и целях пользователя (Johnson, Henderson, 2002).

Концептуальная модель объясняет функции программного обеспечения и перечисляет концепции, с которыми должны быть знакомы пользователи для успешной работы с приложением. В идеальном случае это должны быть концепции, полученные при анализе задач. Чем нагляднее соответствие между концепциями инструмента и концепциями поддерживаемых задач, тем меньше умственных преобразований приходится делать пользователю и тем проще ему изучить инструмент.

После того как вы разработали сфокусированную вокруг задач концептуальную модель — как можно более простую и как можно более единообразную, — можно начинать разрабатывать для нее пользовательский интерфейс. Хороший пользовательский интерфейс должен минимизировать временные и умственные затраты

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

пользователя, необходимые для того, чтобы работа с приложением перешла на автоматический уровень.

Анализ объектов/действий

Самый важный компонент концептуальной модели — это анализ объектов/действий. Он определяет все концептуальные модели, которые приложение будет представлять пользователям, действия, которые пользователи смогут производить над каждым объектом, а также атрибуты (видимые пользователям настройки) каждого типа объекта и взаимоотношения между объектами (Card, 1996; Johnson, Henderson, 2002).

Реализация программного обеспечения может включать и другие объекты, не перечисленные в концептуальной модели, но эти дополнительные объекты не должны быть видны пользователям. Объекты и действия, связанные исключительно с реализацией, такие как текстовый буфер, хэш-таблица или запись в базе данных, не относятся к концептуальной модели.

Таким образом, анализ объектов/действий представляет собой *определение* (*декларацию*) концепций, видимых пользователю. Запомните простое правило: если этот элемент не входит в анализ объектов/действий, то пользователи не должны знать о нем.

Предположим, мы разрабатываем программное обеспечение для управления чековым счетом. В этом случае основанный на задачах анализ объектов/действий включал бы такие объекты, как *транзакция*, *чек* и *счет*. В него не входили бы не связанные с задачами объекты, например *буфер*, *диалоговое окно*, *режим*, *база данных*, *таблица* и *текстовая строка*.

В основанную на задачах концептуальную модель также входили бы следующие действия: *оформление* и *аннулирование* чеков, *внесение на счет* и *снятие* денежных средств, *погашение* счетов, — но не *щелканье* на кнопках, *загрузка* баз данных, *редактирование* строк в таблицах, *очистка* буферов и *переключение* режимов.

Концептуальная модель, сфокусированная на задаче, могла бы определять такие атрибуты:

- у **чеков** могут быть *предъявитель*, *номер*, *сумма*, текст *пометки* и *дата*;
- у **счетов** могут быть *владелец* и *остаток*;
- у **транзакций** (для внесения средств на счет и снятия со счета) могут быть *сумма* и *дата*.

Если добавить в модель атрибуты из компьютерной сферы, такие как *формат записи транзакции*, то она потеряет статус «сфокусированной на задаче». Пользователям неинтересно, какой внутренний формат приложение использует для сохранения записей транзакций. Заставляя их обращать на это внимание, вы снижаете полезность своего приложения и делаете его менее удобным для

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

изучения независимо от того, как много усилий было вложено в разработку пользовательского интерфейса.

Объекты, действия и атрибуты приложения для управления чековым счетом кажутся очевидными. Но давайте рассмотрим задачу, в которой анализ объектов/действий не имеет таких четко выраженных границ: публикация покупателями интернет-магазина отзывов о продуктах.

Для концептуальной модели подходят следующие объекты: *покупатели*, *продукты*, *отзывы* покупателей, *ответы* на отзывы. Такие объекты, как *базы данных*, *таблицы* и *постоянные файлы cookie*, не должны включаться в модель.

Действия над продуктами включают *просмотр* и *добавление* комментариев. Действия над отзывами — *просмотр* и *написание ответа*, а также, для собственных отзывов пользователя, *редактирование*. Атрибуты отзыва: *заголовок*, *имя* покупателя, *дата* публикации.

Обратите внимание на то, что в обоих примерах — приложении для управления чековой книжкой и системе пользовательских отзывов о продуктах — важные концептуальные вопросы разработки можно решить до начала создания пользовательского интерфейса, еще до того, как мы узнаем, будет пользовательский интерфейс представлен на экране компьютера или, например, как голосовое телефонное меню.

Как можно проще

Помимо фокусировки на задачах пользователя концептуальная модель должна обладать еще одним важным свойством: она должна быть как можно проще. Под простотой понимается минимизация числа концепций. Чем меньше концепций приходится изучать пользователям, тем лучше, при условии, что существующий ограниченный набор предоставляет всю необходимую функциональность. Не зря говорят: лучше меньше, да лучше. Главное, чтобы наличествующие объекты удовлетворяли целям и задачам пользователей, например:

- Нужна ли в приложении для планирования дел на день возможность назначать пунктам многоуровневые приоритеты (от 1 до 10) или будет достаточно двух (низкий и высокий)?
- Должна ли функция поиска поддерживать целые булевы выражения? Если добавить эту функциональность, то много ли людей воспользуются ею? Если нет, лучше не усложнять поиск.
- Должен ли автомат по продаже билетов на железнодорожной станции продавать билеты на маршруты, не проходящие через данную станцию?

Очень часто на разработчиков оказывают давление, заставляя добавлять функциональность просто «на случай, если пользователь этого захочет». Сопротивляйтесь этому, если только не существует доказательств, что значительное число потенциальных клиентов и пользователей действительно нуждаются в дополнительной

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

функциональности. Почему? Потому что каждая дополнительная концепция увеличивает сложность программного обеспечения. Это еще одна вещь, которую приходится учить и запоминать пользователям. В действительности даже не *одна*. Каждая концепция приложения взаимодействует с множеством других концепций, и их взаимодействие увеличивает уровень сложности. Следовательно, по мере добавления концепций в приложение его сложность растет не линейно, а пропорционально (рис. 11.1).

Рис. 11.1. Сложность приложения нелинейно возрастает при добавлении новых концепций

Единообразие

От *единообразия* интерактивной системы в значительной степени зависит то, как быстро пользователи переходят от контролируемой, сознательно управляемой, медленной деятельности к автоматической, неконтролируемой и быстрой (Schneider, Shiffrin, 1977). Чем более предсказуемо работают разнообразные функции приложения, тем согласованней работает приложение в целом. В хорошо согласованных системах нетрудно предсказать, как будет работать определенная функция, исходя из ее *типа*. Поэтому люди быстро понимают, как функционирует система, и работа в ней входит в привычку — становится автоматизированной. В несогласованных системах пользователи не в состоянии предсказать, как будет работать та или иная функция, поэтому им приходится каждый раз начинать с нуля. Это замедляет процесс изучения системы, а работа в ней остается на уровне контролируемого, потребляющего все ресурсы внимания процесса.

Интерактивные системы могут быть согласованными или несогласованными, по меньшей мере, на двух уровнях: концептуальном и уровне нажатия клавиш. Согласованность на концептуальном уровне определяется соответствием объектов, действий и атрибутов концептуальной модели (см. ранее). Можно ли сказать, что у большинства объектов в системе одинаковые атрибуты и с ними связаны одинаковые действия? Согласованность на уровне нажатия клавиш определяется

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

соответствием между концептуальными действиями и физическими движениями, необходимыми для их выполнения. Можно ли сказать, что все концептуальные действия одного типа инициируются и управляются одними и теми же физическими движениями?

ЧРЕЗМЕРНАЯ СЛОЖНОСТЬ ИЗ-ЗА СХОЖЕСТИ НЕСВЯЗАННЫХ КОНЦЕПЦИЙ

Некоторые программы оказываются слишком сложными из-за того, что используемые в них концепции частично совпадают по функциональности или значению. Например, на одном веб-сайте службы поддержки пользователей можно обнаружить четыре концепции, которые разработчики посчитали достаточно непохожими:

- **членство** — платит ли компания за услуги по поддержке пользователей;
- **подписка** — оформила ли компания подписку на новостную рассылку службы поддержки пользователей;
- **доступ** — в какие области веб-сайта службы поддержки пользователей компания может заходить;
- **права** — услуги, предоставляемые на каждом уровне членства.

Пользователи постоянно путаются в этих концепциях. В хорошем дизайне все четыре концепции должны быть объединены в одну, по крайней мере, их должно быть меньше четырех.

Еще одна компания разработала веб-сайт для людей, желающих приобрести дом. На сайте предлагали два способа поиска дома: а) по названию штата, графства или города; б) путем выбора точки на карте. Эти два раздела назывались «По местоположению» и «На карте» соответственно. Пользователь должен был выбрать один из них. Тест удобства в использовании показал, что многие пользователи не видели разницы между двумя способами поиска дома. Для них оба метода относились к категории «По местоположению», разница была только в том, каким образом местоположение выбиралось.

Матрица объектов/действия

Необязательный, но зачастую полезный шаг процесса разработки интерактивной системы заключается в оформлении концептуальной модели в виде матрицы объектов и действий. Объекты перечисляются слева, действия перечисляются наверху (рис. 11.2). Пока что мы игнорируем иерархию типов объектов и всего лишь перечисляем сами объекты. Чем больше объектов, тем выше матрица, чем больше действий, тем она шире.

Матрица объектов/действий позволяет визуально оценить, насколько проста или сложна концептуальная модель вашей интерактивной системы. Чем больше матрица, тем *больше* концепций приходится изучать пользователям. Высокая матрица указывает на большое число *объектов*, а широкая — на большое число *действий*, которые пользователям нужно будет запомнить. Матрица также иллюстрирует, насколько согласованна концептуальная модель, то есть насколько легко пользователям будет переносить знания, полученные в одной части системы, в другую.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Действия

Объекты	A	B	C	D	E	F	G	H	I	...
1		✓			✓	✓			✓	
2			✓			✓	✓			
3								✓		
4	✓			✓	✓	✓		✓		
5	✓			✓		✓		✓		
6		✓			✓		✓	✓	✓	
...										

Рис. 11.2. Матрица объектов/действий демонстрирует, какие действия доступны для каждого объекта

Небольшая плотная матрица описывает простую в изучении систему: мало объектов, мало действий, одинаковые операции над объектами всех типов (рис. 11.3, *а*). Например, в простом графическом приложении могут существовать такие концептуальные объекты: линии, эллипсы, дуги, прямоугольники, треугольники, текстовые метки и т. д. Применимые к действиям объекты — это, предположительно, создать, удалить, просмотреть/отредактировать атрибуты, переместить, скопировать, изменить размер, повернуть, отразить и т. д. Матрица объектов/действий для такого простого графического приложения будет состоять из строк, по одной для каждого типа объектов, и столбцов, по одному для каждого действия. Все действия будут применимы к объектам всех типов, поэтому матрица будет заполнена очень плотно, как показанная на рис. 11.3, *а*.

Действия

Объекты	A	B	C	D	E	F
1	✓	✓	✓	✓	✓	✓
2	✓	✓	✓	✓	✓	✓
3	✓	✓	✓	✓	✓	✓
4	✓	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓	✓

Хорошая матрица

Действия

Объекты	A	B	C	D	E	F	G	H	I	J	K	L	M
1		✓	✓				✓						
2					✓	✓					✓		
3								✓		✓			
4	✓			✓	✓	✓		✓					
5			✓							✓			✓
6						✓			✓		✓		
7		✓					✓						✓
8				✓	✓	✓						✓	
9		✓			✓								

Плохая матрица

Рис. 11.3. Матрицы объектов/действий, представляющие: *а* — простой в изучении концептуальный дизайн; *б* — сложный дизайн

Большая матрица с множеством пустых ячеек отражает несогласованный дизайн. Пользователям будет сложно изучать и запоминать его, так как всем концептуальным объектам соответствуют разные действия (рис. 11.3, *б*). Независимо от того, какой пользовательский интерфейс будет наложен на такой дизайн, у пользователей всегда станут возникать с ним трудности.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Простейшее эмпирическое правило рекомендует упрощать концептуальную модель, ориентируясь на ее матрицу объектов/действий, в результате матрица должна получиться как можно меньше и плотнее. В то же время маленькая матрица означает ограниченную функциональность. Добиться небольшого размера матрицы довольно сложно, если приложение несет больше функциональности, чем, скажем, простая программа для рисования, адресная книга в телефоне или веб-сайт со списком почтовых тарифов. Представьте, например, матрицу объектов и действий для системы мониторинга жизнедеятельности пациента в палате реанимации. Даже типичная программа подготовки текстов, такая как Microsoft Word и Apple Pages, включает нетривиальный массив концептуальных объектов и действий.

Для любой желаемой функциональности возможна разработка концептуальных моделей разной сложности. Например, у большинства приложений для управления личными банковскими счетами схожая функциональность. Однако концептуальная модель Intuit Quicken упрощена до минимума — возможно, это одна из причин того, почему данное приложение так популярно. Разработчики программного обеспечения должны стремиться к тому, чтобы создавать простейшие концептуальные модели (с самой компактной матрицей объектов/действий) для требуемой функциональности.

Несмотря на то что простые в изучении и использовании системы зачастую характеризуются небольшими и плотными матрицами объектов/действий, вполне допустимы и нестандартные конфигурации матриц. Представьте приложение, в котором доступ ко всей функциональности осуществляется через пять или шесть общих действий, применимых ко всем объектам. В такой системе может существовать огромное число объектов, однако это не замедлит скорость обучения, так как работа со всеми объектами выполняется единообразно. Матрица объектов/действий для такой системы будет, разумеется, высокой, но узкой и плотной. К подобному подходу прибегали при разработке некоторых высокофункциональных систем, таких как офисная рабочая станция Xerox Star (Johnson et al., 1989). В Star одни и те же шесть команд: переместить, скопировать, открыть, удалить, показать свойства и скопировать свойства — применяются ко всем объектам: символам, словам, абзацам, строкам таблиц, таблицам, диаграммам, сообщениям электронной почты, документам, папкам, принтерам и т. д.

Если включить в матрицу иерархию *типов* объектов, то можно увидеть еще один вариант легкой в изучении концептуальной модели: модель, в которой все объекты четко подразделяются на категории, а в каждой категории используются собственные действия (рис. 11.4). Возможно также существование нескольких действий, применимых ко всем объектам. Матрицу такой модели нельзя назвать небольшой или плотной, но она и не разрозненная. Она характеризуется регулярностью и единообразием, которые способствуют обучению и запоминанию.

В качестве примера можно привести веб-сайт агентства недвижимости, на котором реализован подбор как жилых, так и нежилых помещений. Для каждой из этих

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

категорий предлагаются свои действия, а некоторые действия применимы к помещениям обоих типов.

Объекты		Специфические действия для разных типов								Общие действия				
		A	B	C	D	E	F	G	H	I	J	K	L	M
X	1	✓	✓	✓	✓									✓
	2	✓	✓	✓	✓									✓
	3	✓	✓	✓	✓									✓
Y	4					✓	✓	✓	✓					✓
	5					✓	✓	✓	✓					✓
	6					✓	✓	✓	✓					✓
Z	7									✓	✓	✓	✓	✓
	8									✓	✓	✓	✓	✓
	9									✓	✓	✓	✓	✓

Рис. 11.4. Матрица объектов/действий, представляющая более реалистичную, но также легкую в изучении концептуальную модель

Я называю шаг по созданию матрицы объектов/действий необязательным по двум причинам:

- у опытных дизайнеров взаимодействия редко возникает необходимость чертить реальную матрицу, для того чтобы оценить концептуальную модель, лежащую в основе проекта, — простая она или сложная, согласованная или нет;
- при тестировании удобства в использовании могут обнаружиться требующие упрощения аспекты концептуальной модели приложения, которые дизайнерам и так казались довольно простыми.

Дизайнерам, разрабатывающим концептуальную модель приложения, зачастую достаточно представить, *как бы выглядела матрица объектов/действий, если бы они нарисовали ее на бумаге.*

Главная цель заключается в создании концептуальной модели, сфокусированной на задачах пользователя, при этом как можно более простой и как можно более согласованной. Основываясь на такой модели, нетрудно создать пользовательский интерфейс, в котором работа с приложением будет переходить на автоматический уровень за минимальное время и при минимуме потраченных усилий.

Единообразие нажатий клавиш

Когда дизайнер переходит от концептуального дизайна к дизайну реального пользовательского интерфейса, на первый план выходит единообразие нажатия клавиш.

Единообразие на уровне нажатия клавиш сложнее проиллюстрировать и оценить, однако этот фактор ничуть не менее важен, чем концептуальная согласованность приложения. От него также в значительной степени зависит, насколько быстро пользователь начнет взаимодействовать с интерактивной системой автоматически,

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

не прилагая к этому особых ментальных усилий. Главная цель здесь — содействовать развитию так называемой мышечной памяти, то есть моторных привычек.

Для того чтобы достичь единообразия на уровне нажатия клавиш, необходимо стандартизировать физические действия, с помощью которых выполняются задачи одного типа. Например, одним из типов активности является редактирование текста. Единообразное редактирование текста означает, что нажатия клавиш (и движения мыши) всегда остаются одинаковыми независимо от того, в каком контексте текст редактируется — в документах, полях формы, названиях файлов и т. п. Другие типы активности, для которых желательно поддерживать единообразие на уровне нажатия клавиш, — это открытие документов, переход по ссылкам, выбор команды из меню, выбор одного варианта из представленного набора, щелчки на кнопках, прокрутка представления и т. п.

Система, в которой отсутствует единообразие на уровне нажатия клавиш, не позволяет быстро развивать мышечную память: пользователям постоянно приходится оставаться настороже и гадать, какие клавиши нажимать в каком контексте, даже если контексты различаются совсем незначительно.

Для обеспечения единообразия на уровне нажатия клавиш разработчики зачастую прибегают к помощи стандартов по оформлению диалога машины с пользователем. Такие стандарты могут входить в руководства по стилю или быть частью распространенных инструментов построения пользовательских интерфейсов и наборов компонентов. Существуют общие для всей отрасли руководства по стилю, а также отдельные руководства для настольных приложений (Apple Computer, 2009; Microsoft Corporation, 2009) и веб-дизайна (Koyani, Bailey, Nall, 2006). В идеальном варианте у компаний должны также быть собственные внутренние руководства по стилю, дополняющие промышленные стандарты и определяющие правила оформления диалога пользователя с компьютером в конкретных приложениях.

Независимо от того, каким образом эти стандарты и условности оформляются, главное правило разработчика — придерживаться единообразия на уровне нажатия клавиш, внедряя инновации на концептуальном уровне и уровне задач. Мы, дизайнеры, *совершенно* не хотим, чтобы пользователи наших приложений непрерывно думали о том, какую клавишу им нажать или куда подвинуть указатель мыши, точно так же пользователи не желают забивать себе голову этими подробностями.

Мы учимся быстрее, когда терминология сфокусирована на задаче, знакома и единообразна

Реализация в приложении, веб-службе или устройстве небольшого, единообразного и соответствующего задаче набора концепций — это большой шаг вперед, однако его недостаточно для того, чтобы минимизировать время изучения интерактивной системы. Необходимо также удостовериться, что *терминология* (словарь, использу-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

емый для описания концепций) хорошо подходит для задачи, знакома пользователю и единообразна.

Терминология должна быть сфокусирована на задаче

Как и сами видимые пользователю концепции интерактивной системы, их *названия* также должны быть сфокусированы на задаче. Обычно термины, подходящие для обозначения этих концепций, можно обнаружить путем интервьюирования пользователей и наблюдения за ними — дизайнеры делают это на этапе анализа задачи. Иногда программное обеспечение представляет новые концепции, еще не знакомые пользователям. Задача дизайнера в таком случае — сделать так, чтобы подобные концепции и их названия все так же оставались сфокусированными на задаче, а не на технологии.

Приведу примеры интерактивных систем, в которых терминология не сфокусирована на задаче:

- Компания разработала настольное приложение для выполнения инвестиционных транзакций. Приложение позволяет создавать и сохранять шаблоны часто используемых транзакций. Сохранить шаблон можно либо на собственном компьютере, либо на сетевом сервере. Шаблоны, хранящиеся на собственном компьютере, доступны только их создателю. Шаблоны на сервере могут использоваться другими людьми. Разработчики применили для шаблонов на сервере термин «база данных», так как шаблоны действительно сохраняются в базу данных. Шаблоны на собственном компьютере пользователя называются локальными, так как это понятный для разработчиков компьютерный термин. Однако гораздо лучше было бы использоваться термины, сфокусированные на задаче: «общие» или «публичные» для шаблонов из базы данных и «частные» вместо «локальные».
- На веб-сайте iCasualties.org выводятся актуальные данные по количеству служащих, убитых или раненых во время войн в Ираке и Афганистане. Работа с сайтом начинается с того, что пользователей просят выбрать базу данных. Однако посетители сайта не знают, что данные веб-сайта могут храниться в разных базах данных, и их это совершенно не интересует. Сфокусированные на задаче инструкции просили бы выбрать *страну*, в которой ведется война, а не базу данных (рис. 11.5).

Рис. 11.5. Инструкции на веб-сайте iCasualties.org не сфокусированы на задаче (пример тому — термин «база данных»)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Терминология должна быть знакомой

Для того чтобы сокращать время, необходимое пользователям для освоения вашего приложения, веб-сайта или устройства, то есть для того чтобы его использование перешло на автоматический или почти автоматический уровень, нужно делать так, чтобы им не приходилось осваивать новый лексикон. В главе 4 я объяснял, что знакомые слова проще читать и понимать, потому что они распознаются автоматически. Незнакомые слова заставляют людей применять более осознанные методы декодирования, потребляющие скудные ресурсы краткосрочной памяти и, следовательно, ухудшающие понимание.

К сожалению, многие компьютерные продукты и услуги вываливают на пользователей массу незнакомых терминов из сферы вычислительной техники (так называемого компьютерного жаргона) и требуют, чтобы те быстро овладели им (рис. 11.6). Почему? Для того чтобы начать готовить на плите, нам не нужно учить терминологию, описывающую сжатие и химический состав природного газа, а также производство и поставку электричества. Почему покупки в Интернете, рассылка фотографий и проверка электронной почты должны зависеть от компьютерного жаргона, всяких там USB, TIFF и *широкополосной передачи*? Однако очень часто все происходит именно так.

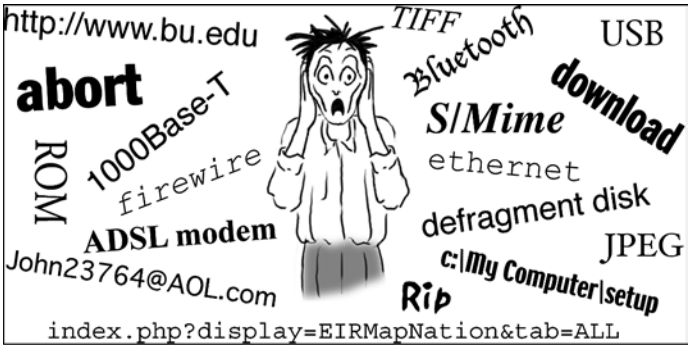


Рис. 11.6. Незнакомая компьютерная терминология замедляет обучение и разочаровывает пользователей

Несколько примеров интерактивных систем, использующих незнакомую терминологию:

- Команда разработчиков создала систему поиска и хранения видеоклипов, которую школьные учителя могут использовать во время уроков: находить видео, предлагаемое школьным округом, загружать его и демонстрировать ученикам. Первоначально разработчики планировали организовать иерархическую структуру видео, с «категориями» и «подкатегориями». Однако интервью с учителями показали, что для организации учебного материала, включая видео, в их среде применяются термины «тема» и «урок». Если бы в системе была использова-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

на терминология разработчиков, то учителям пришлось бы *запоминать*, что «категория» соответствует «теме», а «подкатегория» — «уроку», что сильно затруднило бы освоение системы.

- На веб-сайте авиакомпании Continental Airlines отображались сообщения об ошибках, сформулированные с использованием компьютерного жаргона (рис. 11.7). Чаще всего эти сообщения должны были информировать пользователей о возникновении той или иной проблемы, но поскольку они содержали незнакомые термины, лишь немногие люди понимали, что сайт говорит им, и поэтому не были уверены, что делать дальше. Подобные сообщения об ошибках лучше подходят для уведомления системных инженеров о существующих неполадках. В контексте общения компьютера с пользователями такие сообщения необходимо либо переформулировать в понятных терминах, либо показывать только администраторам веб-сайта, следящим за его работоспособностью.

Рис. 11.7. Сообщение об ошибке на сайте Continental.com написано на компьютерном жаргоне

- Проигрыватель Windows Media Player периодически отображает сообщения об ошибках, в которых знакомые термины используются в непонятных, жаргонных формулировках (рис. 11.8). В сообщении на этом рисунке слово state означает *состояние приложения*: «Запрос недопустим в текущем состоянии», однако обычный американец — пользователь Media Player может истолковать его неверно, поняв слово state как *штат*, в котором он проживает: «Запрос недопустим в вашем штате».

Рис. 11.8. В сообщении об ошибке в Windows Media Player знакомый термин (current state) используется в непривычной формулировке

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

В противоположность предыдущим примерам веб-сайт авиакомпании Southwest Airlines старается предотвратить возникновение ошибок, однако если проблема все же случается, он объясняет ее знакомым, сфокусированным на задаче языком (рис. 11.9).

Рис. 11.9. Сообщения об ошибках на веб-сайте Southwest Airlines сфокусированы на задаче и сформулированы ясно и недвусмысленно

Терминология должна быть единообразной

Люди хотят фокусировать свои когнитивные ресурсы на собственных целях и задачах, а не на используемом программном обеспечении. Они стремятся достичь цели, какой бы она ни была, и программное обеспечение их мало интересует. Они интерпретируют то, что видят на экране системы, очень поверхностно и буквально. Их ограниченные ресурсы внимания настолько сосредоточены на цели, что они могут не найти кнопку поиска на странице, если она подписана «Запрос», а не «Поиск». Таким образом, терминология в интерактивной системе должна быть максимально согласованной.

Терминологию интерактивной системы можно назвать единообразной, когда каждой концепции соответствует *одно и только одно* название. Каролин Джарретт (Caroline Jarrett), специалист по пользовательским интерфейсам и дизайну форм, советует запомнить такое правило: «*Одно название — одна вещь*; *другое название — другая вещь*» (*FormsThatWork.com*). Это означает, что между терминами и концепциями должно существовать однозначное соответствие — соответствие «один к одному». Никогда не описывайте *одну и ту же* концепцию разными терминами и никогда не используйте *один и тот же* термин для именования разных концепций. Даже термины, имеющие два смысла в реальном мире, *в системе* должны означать одну-единственную вещь. В противном случае трудно будет изучать систему и запоминать ее концепции.

Пример использования разных терминов для одной и той же концепции можно найти на странице часто задаваемых вопросов (frequently asked questions, FAQ) веб-сайта Earthlink в разделе, посвященном хостингу (рис. 11.10). В *вопросе* две

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

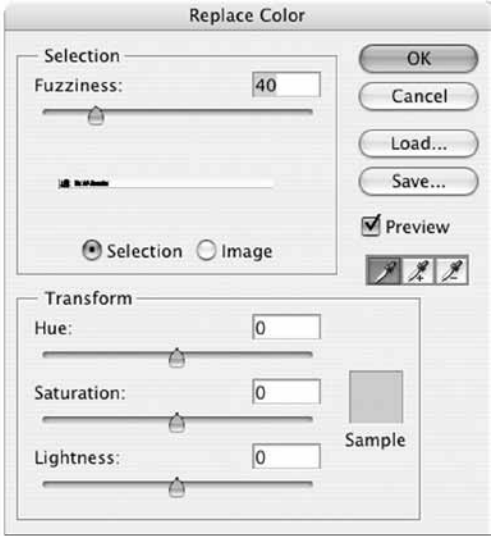
доступные платформы хостинга называются «на базе Windows» (Windows-based) и «на базе UNIX» (UNIX-based), в то время как в *таблице* на них ссылаются как на «стандартную» (Standard) и «ASP». Покупателям приходится отвлекаться от насущного вопроса и задумываться, как же соотносятся между собой определения в вопросе и в таблице. А вы знаете правильный ответ?

What are the differences between Windows-based and UNIX-based platforms?
For detailed information on choosing between these two operating systems, please visit our Windows or UNIX page.

For a quick look at which features and programs each platform supports, please consult the chart below:

Feature	Standard	ASP
Microsoft FrontPage Extensions	YES	YES
RealVideo and RealAudio	YES	NO
ASP (Active Server Pages)	NO	YES
ADO (ActiveX Data Objects)	NO	YES
ODBC Data Sources	NO	YES
Windows Media Server	NO	YES
PHP	YES	NO
MySQL Databases	YES	NO
Free ready-to run scripts, such as a hit counter, forum, e-mail form, blog, and guestbook	YES	NO

Рис. 11.10. На странице часто задаваемых вопросов о хостинге веб-сайта Earthlink в формулировке вопроса и в таблице разные термины описывают одни и те же концепции



а

б

Рис. 11.11. В приложении Photoshop в двух функциях замены цвета для параметра, определяющего допуск, используются разные названия: *а* — Fuzziness (Неопределенность) в функции Replace Color (Заменить цвет); *б* — Tolerance (Допуск) в функции Paint Bucket (Ведро с краской)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

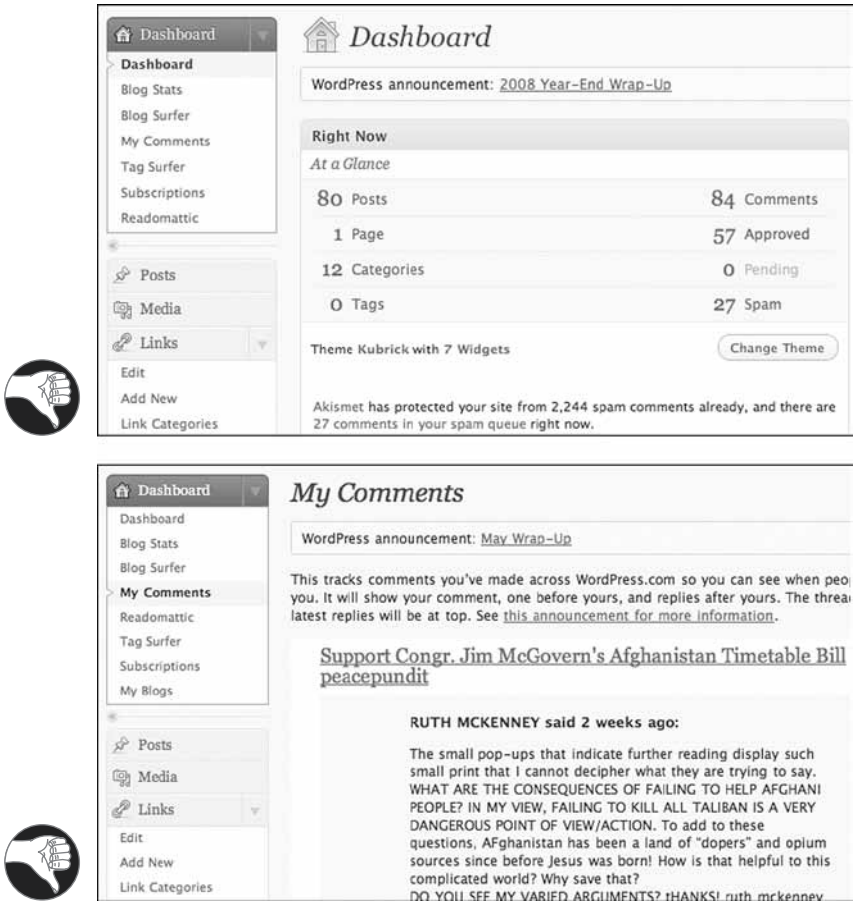


Рис. 11.12. В системе WordPress термином Dashboard (Приборная панель) называется вся административная часть блога, а также одна из страниц административной части

Пример из приложения Adobe Photoshop демонстрирует, как несогласованная терминология может затруднить обучение. В Photoshop есть две функции замены цвета в изображении: Replace Color (Заменить цвет), которая меняет один цвет на другой во всем изображении, и Paint Bucket (Банка с краской), которая заменяет цвет только в выделенной области. У обеих функций есть параметр, определяющий, насколько похожим должен быть цвет изображения на выбранный цвет, для того чтобы произошла замена. Несоответствие заключается в том, что у функции Replace Color (Заменить цвет) этот параметр называется Fuzziness (Неопределенность), а у функции Paint Bucket (Банка с краской) — Tolerance (Допуск) (рис. 11.11). В интерактивной справке о функции Replace Color (Заменить цвет) в Photoshop даже говорится: «Adjust the *tolerance* of the mask by dragging the *Fuzziness* slider or entering a value» (Отрегулируйте допуск маски, перетаскивая ползунок Неопределенность или указав точное значение). Если бы параметр просто назывался Tolerance (Допуск) в настрой-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

ках *обеих* функций замены цвета, то пользователи, выучившие особенности одной функции, могли бы быстро перенести это знание на вторую. Однако из-за существующего несоответствия людям приходится знакомиться с обеими функциями с нуля.

Наконец, на веб-сайте WordPress.com можно найти пример использования одного термина для двух разных концепций — это также называют *перезагрузкой* термина. В системе WordPress всем пользователям для администрирования блога предоставляется инструмент под названием Dashboard (Приборная панель), включающий несколько страниц с функциями мониторинга и управления. Проблема заключается в том, что одна из страниц приборной панели также носит название Dashboard (Приборная панель), то есть одно и то же имя может обозначать как приборную панель целиком, так и одну из ее составляющих (рис. 11.12). Таким образом, когда новый пользователь начинает изучать WordPress, он обнаруживает, что термин «приборная панель» может относиться как ко всей административной области, так и к одной определенной странице этой области, и прилагает усилие, чтобы это запомнить.

Разработать сфокусированную на задаче, знакомую и единообразную терминологию проще, когда у вас есть хорошая концептуальная модель

Хорошие новости: при выполнении анализа задач и разработке сфокусированной на задачах концептуальной модели вы одновременно составляете словарь, которым ваша целевая аудитория пользуется при обсуждении этих задач. Вам не приходится придумывать новые термины для концепций приложения, видимых пользователям: вы просто берете термины, которые люди, выполняющие задачи, *уже* используют. Более того, *не следует* присваивать этим концепциям новые имена, так как любые новые названия, скорее всего, окажутся концепциями из сферы компьютерных технологий и, следовательно, чуждыми задачам[1].

Основываясь на концептуальной модели, команда по разработке программного обеспечения должна составить *лексикон* продукта. Лексикон присваивает название и определение каждому объекту, действию и атрибуту, которые продукт — и документация по продукту — представляет пользователям. В лексиконе термины должны однозначно соответствовать концепциям. Нельзя присваивать несколько терминов одной концепции или связывать один термин с несколькими концепциями.

Кроме того, термины лексикона должны происходить от *задач*, поддерживаемых приложением, а не от его реализации. Они должны хорошо встраиваться в обычный словарь пользователей, выполняющих эти задачи, даже если вам пришлось добавить какие-то новые термины. Чаще всего помощь в создании лексикона могут оказать технические писатели, дизайнеры пользовательского интерфейса, разработчики и менеджеры.

[1] Если только вы не проектируете инструменты для разработки программного обеспечения.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Названия определенных концепций графического интерфейса пользователя стали промышленным стандартом — эквивалентом зарезервированных слов в языках программирования. Переименовав такие концепции или связав новые значения со стандартными названиями, вы собьете пользователей с толку.

Согласованно используйте лексикон продукта в программном обеспечении, руководствах пользователя и маркетинговых материалах. Относитесь к нему как к развивающемуся документу: по мере того как продукт совершенствуется, лексикон должен меняться с учетом новинок дизайна, изменений в функциональности, результатов тестирования удобства в использовании и ответной реакции рынка.

Мы учимся быстрее, когда риск низок

Представьте, что вы собираетесь в командировку в другой город на одну-две недели. После работы, а также по выходным у вас свободное время, которое вы можете провести, как вам нравится. Сравните два города:

- о первом городе известно, что в нем легко ориентироваться. Улицы пересекаются под прямыми углами, понятные уличные знаки и вывески в метро написаны на знакомом языке, жители и полицейские говорят на том же языке, что и вы, и с удовольствием помогают туристам;
- относительно второго города вас предупредили, что в нем сложно найти дорогу. Извилистые улицы плохо обозначены, редкие уличные знаки и вывески в метро написаны на незнакомом языке, а жители не знают вашего языка и в целом враждебно относятся к туристам.

Какой город вы с большей вероятностью отправитесь исследовать?

В подавляющей части интерактивных систем — настольных приложений, веб-служб, электронных устройств — заложено *намного* больше функциональности, чем большинство пользователей хотя бы единожды пробуют применить. Зачастую люди даже не подозревают о существовании множества дополнительных функций в программах или устройствах, которыми пользуются каждый день. Одна из причин, почему люди не стремятся осваивать новую функциональность, — боязнь «обжечься».

Люди делают ошибки. А во многих интерактивных системах слишком легко сделать ошибку, которую потом исправить либо совсем невозможно, либо можно, но ценой больших затрат, в том числе временных. В подобных системах у людей не получается работать продуктивно: они тратят слишком много времени на исправление ошибок и устранение их последствий.

Еще более важным фактором, чем *время*, является *обучение*. Система с высоким риском, в которой легко совершить дорогостоящую ошибку, препятствует исследованию: когда люди нервничают и боятся сделать неправильный шаг, они

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

предпочитают придерживаться знакомых, безопасных путей и функций. А когда у людей нет желания исследовать и высока степень тревожности, обучение серьезно страдает.

В противоположность этому в системах с низким риском, где ошибку совершить трудно, а если уж она совершена, это не приводит к большим затратам и легко исправляется, люди не испытывают стресса и с удовольствием занимаются исследованием, постоянно узнавая что-то новое. В подобных системах пользователи с большим энтузиазмом пробуют незнакомые пути: «Хм-м-м, интересно, а что делает *эта* кнопка?»

Чтобы поддерживать желание изучать, интерактивные системы должны восприниматься пользователями как среды с низким риском, в которых не страшно заниматься исследованием и пробовать новые вещи. Разрабатывать безопасное программное обеспечение вам помогут следующие рекомендации:

- где возможно, предотвращайте ошибки;
- отключайте недопустимые команды;
- упрощайте обнаружение ошибок, показывая пользователям, какие действия они выполнили (например, по ошибке удалили абзац);
- позволяйте пользователям с легкостью отменять действия, устранять последствия ошибочных действий и исправлять ошибки.

Резюме

Целью этой главы было объяснить и описать факторы, влияющие на скорость изучения интерактивных систем. Вы узнали, от чего зависит, как быстро люди овладевают системами настолько профессионально, что управлять работой в них начинают в основном автоматические когнитивные процессы. Мы быстрее учимся использовать интерактивные системы при следующих условиях:

- работа в системе основывается на целях и задачах пользователя (а не на реализации системы), действия концептуально просты и единообразны;
- используемая терминология знакома пользователям, относится к задачам пользователя и применяется единообразно (то есть однозначно сопоставляет термины и концепции);
- система является средой с низким риском, в которой трудно сделать ошибку, а если ошибка произошла, то не влечет больших затрат и ее легко исправить.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Глава 12

Требования к времени

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

События в реальном мире происходят не мгновенно, на каждое требуется какое-то время. Процесс *восприятия* объектов и событий также занимает определенное время. Время необходимо для запоминания событий, размышления о прошлых и будущих событиях, извлечения уроков из них, выполнения планов, реагирования на то, что мы увидели и вспомнили. О каком количестве времени мы говорим? И как знание продолжительности перцепционных и когнитивных процессов поможет в разработке интерактивных систем?

В этой главе вы найдете ответы на заданные вопросы. В ней рассказывается о продолжительности перцепционных и когнитивных процессов и на основании этих значений даются реальные рекомендации относительно *предельных сроков* исполнения процессов в интерактивных системах, которые необходимо соблюдать, чтобы успешно синхронизироваться с пользователями. Системы, не отвечающие человеческим требованиям к времени, менее эффективны, и люди считают их недостаточно отзывчивыми.

Может показаться, что второй аспект, а именно отзывчивость системы, как ее оценивают пользователи, менее важен, чем ее эффективность, однако в действительности все наоборот. В последние четыре десятилетия исследователи постоянно обнаруживают, что отзывчивость интерактивной системы: ее способность работать в темпе пользователей, информировать их о текущем состоянии и не заставлять их ожидать неизвестно чего — это *наиболее важный* фактор, определяющий степень удовлетворенности пользователей. Не *один из* основных, а главный, *самый* важный фактор[1]. Он важнее, чем простота в изучении. Важнее, чем легкость использования. Сколько бы исследований ни проводилось, все они подтверждают данный факт (Barber, Lucas, 1983; Carroll, Rosson, 1984; Lambert, 1984; Miller, 1968; Rushinek, Rushinek, 1986; Shneiderman, 1984; Thadhani, 1981).

В этой главе в первую очередь я дам определение отзывчивости. Затем перечислю некоторые наиболее важные временные константы человеческой перцепционной и когнитивной деятельности. А в завершение приведу несколько связанных с временными параметрами рекомендаций по проектированию интерактивных систем, сопроводив их примерами.

Что такое отзывчивость

Отзывчивость связана с производительностью, но это не одно и то же. Производительность оценивается в терминах количества вычислений за единицу времени.

[1] Некоторые исследователи полагают, что при оценке пользователями скорости веб-сайтов в действительности существует обратная причинно-следственная связь: чем больше положительных эмоций люди получают в процессе работы с сайтом, тем быстрее он им кажется, даже если их оценка скорости не совпадает с фактическими показателями функционирования (Perfetti, Landesman, 2001).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Отзывчивость оценивается в терминах соответствия пользовательским требованиям ко времени и, как уже говорилось, в терминах удовлетворения пользователей.

Интерактивные системы могут быть отзывчивыми, несмотря на низкую производительность. Если вы позвоните знакомому с целью спросить у него что-нибудь, он может проявить отзывчивость, даже если не в состоянии немедленно дать ответ: он выслушает вас и пообещает перезвонить позже. Если же он отличается повышенной отзывчивостью, то может даже сказать, *когда именно* перезвонит.

Отзывчивые системы продолжают общаться с пользователем, даже когда не могут немедленно выполнить его запросы. Они предоставляют сведения о том, какие действия пользователь выполнил и что сейчас происходит, причем обратной связи назначаются определенные уровни приоритета, зависящие от перцепционных, моторных и когнитивных предельных сроков, свойственных *человеку* (Duis, Johnson, 1990). В частности, системы делают следующее:

- немедленно сообщают, что получили введенные пользователем данные;
- дают приблизительную оценку того, сколько времени займет длительная операция (рис. 12.1);
- позволяют пользователям во время ожидания заниматься другими делами;
- интеллектуально управляют событиями в очереди;
- выполняют служебные и низкоприоритетные задачи в фоновом режиме;
- предвосхищают наиболее часто используемые пользователями запросы.

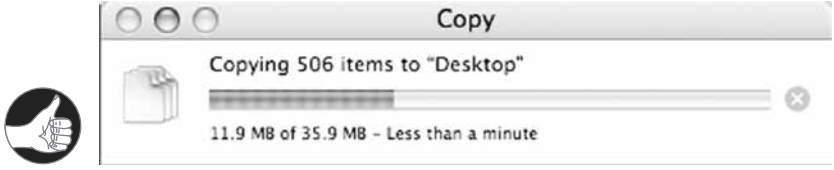


Рис. 12.1. Диалоговое окно перемещения файлов в Mac OS X: хороший индикатор выполнения, удобная оценка времени, присутствует кнопка отмены (крестик в кружке)

Даже быстрое приложение может казаться неотзывчивым. Представьте, что часовщик очень быстро чинит часы, однако не отзывается, когда вы входите в его мастерскую, и в целом игнорирует ваше присутствие до тех пор, пока не закончит возиться с теми часами, которые у него в руках. Он не отзывается, когда вы протягиваете ему свои часы, — просто уходит в подсобку, не сообщая, пошел ли он их чинить или у него начался обеденный перерыв. Даже если он немедленно начинает трудиться над вашими часами, вы все равно посчитаете его неотзывчивым, если он не сообщит, сколько это займет времени: пять минут или пять часов.

Системы, демонстрирующие низкую отзывчивость, не отвечают требованиям к предельным срокам, которые ставят пользователи. Они не в состоянии работать в темпе пользователей. Эти системы не умеют оперативно реагировать на пользова-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

тельские действия, поэтому люди постоянно пребывают в сомнении относительно того, что они только что сделали и чем в данный момент занимается система. Они заставляют пользователей ждать, причем случается это в непредсказуемые моменты, а ожидание длится неопределенный период времени. Они ограничивают скорость работы пользователей, иногда очень сильно. Вот несколько примеров низкой отзывчивости:

- замедленная реакция на нажатие кнопок, перемещение ползунка на полосе прокрутке или манипулирование объектами;
- невозможность прерывания длительных операций, блокирующих остальные действия в приложении (рис. 12.2);
- отсутствие любых упоминаний о том, как долго будет выполняться длительная операция (рис. 12.2);
- дергающаяся анимация, которую трудно отслеживать;
- игнорирование вводимых пользователем данных во время выполнения служебных задач, которые пользователь не запрашивал.

а *б*

Рис. 12.2. Нет ни полосы прокрутки (только индикатор занятости), ни кнопки отмены: операционная система Mac OS X (*а*); приложение iMovie (*б*)

Перечисленные проблемы снижают производительность работы пользователей, раздражают и разочаровывают их. К сожалению, несмотря на все исследования, доказывающие, что отзывчивость имеет критическое значение для обеспечения удовлетворенности пользователей и повышения их продуктивности, многие современные интерактивные системы отличаются очень низкой отзывчивостью (Johnson, 2007).

Временные константы человеческого мозга

Для того чтобы понять, какие связанные со временем требования пользователи предъявляют к интерактивным системам, давайте немного углубимся в нейропсихологию.

Человеческий мозг и нервная система — не единый орган, они включают множество состоящих из нервных клеток органов, возникших на совершенно разных этапах эволюционной цепочки, благодаря которой черви превратились в людей. Этот набор органов обладает огромным количеством сенсорных, регулирующих,

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

моторных и когнитивных функций. Неудивительно, что разные функции работают с разными скоростями. Некоторые срабатывают очень быстро, буквально за доли секунды, другие намного медленнее: их выполнение занимает несколько секунд, минут, часов и даже больше.

Например, в главе 10 я рассказывал, что автоматическая обработка, которая происходит в вашем мозге, когда вы *играете* на музыкальном инструменте давно выученную мелодию, выполняется на «тактовом процессоре», скорость которого, по меньшей мере, в 10 раз больше, чем у «процессора», управляющего контролируемыми функциями, такими как *сочинение* мелодии. Еще один пример — рефлекс «уворачивания». Область нашего мозга, которая называется верхними колликулами, — часть возникшего в самом начале эволюционного процесса старого мозга — умеет «видеть» быстро приближающиеся объекты и заставляет нас уклоняться или вскидывать руки задолго до того, как этот объект будет замечен и идентифицирован корой головного мозга.

В следующей врезке приводятся сведения о продолжительности некоторых наиболее важных перцепционных и когнитивных функций мозга. Большинство понятны без объяснений, но о некоторых мы подробнее поговорим чуть далее.

СКОЛЬКО ВРЕМЕНИ НАШЕМУ МОЗГУ НУЖНО НА ТО, ЧТОБЫ...?

Далее приводятся измеренные значения продолжительности работы перцепционных и когнитивных функций мозга, влияющие на нашу оценку отзывчивости интерактивных систем. Значения упорядочены от наименьших к наибольшим (Card et el., 1991; Johnson, 2007; Sousa, 2005; Stafford, Webb, 2005).

Перцепционные и когнитивные функции	Продолжительность
Самый короткий интервал тишины, который мы способны услышать в звуковом потоке	1 мс (0,001 с)
Минимальное время между импульсами слуховых нейронов — самых быстрых нейронов мозга	2 мс (0,002 с)
Минимальное время, в течение которого необходимо наблюдать визуальный раздражитель, чтобы он повлиял на нас (возможно, подсознательно)	5 мс (0,005 с)
Минимальная задержка в поступлении чернил во время рисования пером	10 мс (0,01 с)
Максимальный интервал между последовательными звуковыми импульсами, для того чтобы мы воспринимали их как непрерывный тон (звуковой поток)	20 мс (0,02 с)

продолжение ⇨

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Перцепционные и когнитивные функции	Продолжительность
Максимальный интервал между последовательными изображениями, для того чтобы мы воспринимали их как визуальный поток	50 мс (0,05 с)
Скорость непроизвольной моторной реакции на возможную опасность	80 мс (0,08 с)
Задержка между визуальным событием и моментом, когда мы полностью воспринимаем его (цикл перцепционной обработки)	100 мс (0,1 с)
Продолжительность саккады (непроизвольного скачкообразного движения глазных яблок), в течение которой зрение подавляется	100 мс (0,1 с)
Максимальный интервал между событиями, которые мы воспринимаем как причину и следствие	140 мс (0,14 с)
Время, за которое мозг умеющего бегло читать человека понимает напечатанное слово	150 мс (0,15 с)
Время, необходимое для определения числа элементов в поле зрения (до четырех-пяти элементов)	200 мс (0,2 с; 50 мс на элемент)
«Установочное окно» для событий при достижении ими сознания	200 мс (0,2 с)
Время, необходимое для идентификации (то есть именования) увиденного объекта	250 мс (0,25 с)
Время, необходимое для мысленного подсчета *всех* элементов сцены, если *их больше четырех*	300 мс (0,3 с)
Мигание внимания (то есть невосприимчивость к остальной поступающей информации), следующее за узнаванием объекта	500 мс (0,5 с)
Время визуально-моторной реакции (осознанная реакция на неожиданное событие)	700 мс (0,7 с)
Максимальная продолжительность паузы между фразами разных людей во время диалога	Около 1 с
Время, в течение которого внимание непрерывно уделяется одной задаче (единице задачи)	6–30 с
Время, необходимое на принятие критических решений в экстремальных ситуациях, например при установлении очередности оказания помощи пострадавшим	1–5 мин
Продолжительность принятия решения о важной покупке, например при выборе автомобиля	1–10 дней
Время, необходимое для выбора дела жизни	20 лет

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Самый короткий интервал тишины, который мы способны услышать в звуковом потоке: 1 мс

Наш слух намного чувствительней к коротким событиям и небольшим различиям, чем зрение. Человеческий слух работает на базе механических приемников звука, а не электрохимической нейронной схемы. Барабанная перепонка передает вибрации на слуховые косточки (косточки среднего уха), которые, в свою очередь, передают их волосковым клеткам улитки, вибрация которых, наконец, инициирует передачу электрических импульсов в мозг. Благодаря механическому соединению элементов уши намного быстрее реагируют на звук, чем палочки и колбочки сетчатки глаза — на свет. Такая скорость позволяет слуховой системе распознавать очень небольшие различия во времени поступления звука в одно и другое ухо — именно на основании этой разницы приема звукового сигнала мозг определяет направление источника звука.

Минимальное время, в течение которого необходимо наблюдать визуальный раздражитель, чтобы он повлиял на нас (возможно, подсознательно): 5 мс

Это основа так называемого подсознательного восприятия. Если вам показать картинку, задержав ее всего на 5–10 мс, то вы не осознаете, что видите ее, тем не менее низкоуровневые составляющие зрительной системы зарегистрируют данный раздражитель. Одним из следствий будет то, что в следующий раз, увидев эту картинку, вы посчитаете ее знакомой. Краткое воздействие изображения или неясный образ объекта также запускают реакции в старом мозге и среднем мозге: желание избежать встречи, страх, гнев, печаль, радость. Это происходит, даже если изображение исчезает еще до того, как его идентифицирует разум. Однако в противоположность популярному заблуждению, подсознательное восприятие незначительно влияет на поведение. Оно не может заставить вас делать вещи, которые в противном случае вы бы делать не стали, или желать чего-то, что, не будь его, не вызвало бы у вас интереса (Stafford, Webb, 2005).

Скорость непроизвольной моторной реакции на возможную опасность: 80 мс

Когда к нам быстро приближается объект — или даже тень, или когда мы слышим поблизости громкий звук, или если кто-то или что-то внезапно толкает, ударяет или хватает нас, мы рефлекторно реагируем: вырываемся, закрываем глаза, вскидываем руки, чтобы защититься, и т. п. Рефлекс срабатывает очень быстро по сравнению с осознанной реакцией на наблюдаемое событие — приблизительно в 10 раз быстрее. Высокая скорость «уворачивания» подтверждается не только экспериментами, но и событиями реальной жизни. Например, об этом можно судить по травмам, которые наблюдаются у людей, пострадавших в результате нападения или во время дорожно-транспортных происшествий. Очень часто повреждения рук и кистей жертв указывают на то, что люди сумели поднять руки за доли секунды (Blauer, 2007).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Задержка между визуальным событием и моментом, когда мы полностью воспринимаем его: 100 мс

От момента, когда свет, отразившись от внешнего объекта, падает на сетчатку глаза, до момента, когда нейронные импульсы, запущенные этим событием, достигают коры головного мозга, проходит приблизительно 0,1 с. Допустим, осознание внешнего мира мозгом отстает от реальных событий на десятую долю секунды. Такая задержка крайне неблагоприятна для выживания: за одну десятую секунды кролик, на которого вы охотитесь, успеет пересечь большую часть лужайки. Наш мозг компенсирует такую задержку путем экстраполяции: он делает предположение относительно того, где движущийся объект окажется через 0,1 с. Таким образом, когда кролик мчится через ваше поле зрения, вы видите его там, где, по оценке мозга, он находится *сейчас*, а не где он был десятую долю секунды назад (Stafford, Webb, 2005).

Максимальный интервал между событиями, которые мы воспринимаем как причину и следствие: 140 мс

Этот интервал представляет собой предельный срок восприятия связи между причиной и следствием. Если интерактивной системе требуется больше 0,14 с для того, чтобы среагировать на ваше действие, ответ системы уже не кажется связанным с этим действием. Например, если появление на экране вводимых с клавиатуры символов задерживается более чем на 140 мс, исчезает впечатление, что именно вы вводите эти символы. Вы отвлекаетесь от значения текста, и внимание переносится на сам акт ввода символов, что замедляет работу, переводит набор текста с автоматического уровня обработки на сознательный и увеличивает вероятность ошибки.

Время, необходимое для определения числа элементов в поле зрения (до четырех-пяти элементов): 200 мс (50 мс на элемент)

Представьте, что ваш знакомый бросил на стол две монеты. Вам потребуется всего лишь один взгляд для того, чтобы сказать, что монет ровно две. У вас нет необходимости сознательно пересчитывать их. То же самое происходит, когда монет три или четыре. Некоторые люди способны одновременно воспринимать пять объектов. Данная функция носит название *одномоментного опознавания количества* (subitizing). Когда объектов больше четырех или пяти, задача усложняется: вам приходится подсчитывать их. Если же монеты случайно лягут на поверхность стола отдельными группками, возможно, вы сможете одновременно опознать количество в каждой группе и сложить результаты. Вот почему, считая большое количество объектов, мы рисуем четыре палочки, затем пятой перечеркиваем эту группку и т. д., приблизительно так: 𝍸 𝍸 ||. Создается впечатление, что одномоментное опознавание количества происходит действительно *одномоментно*, но это не так. Нам требуется около 50 мс на регистрацию каждого элемента (Card et al., 1983; Stafford, Webb, 2005). И все же это намного быстрее, чем явный подсчет, в ходе которого мы затрачиваем приблизительно 300 мс на каждый элемент.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

«Установочное окно» для событий при достижении ими сознания: 200 мс

Мы не всегда воспринимаем события в том порядке, в котором они происходят. Очевидно, что у нашего мозга есть подвижное «установочное окно» продолжительностью около 200 мс, во время которого элементы, которые мы восприняли и вспомнили, борются между собой за то, чтобы переместиться в сознание. В течение времени «окна» события и объекты, которые в других условиях могли бы оказаться в нашем сознании, иногда подавляются другими событиями и объектами — даже теми, которые произошли позже (но в пределах «окна»). Кроме того, внутри «окна», пока события перемещаются в сознание, их последовательность может меняться. Пример: если мы видим, как точка исчезает и мгновенно появляется в новом месте, мы воспринимаем эту последовательность изменений сцены как движение точки. Почему? Определенно, наш мозг не угадывает вторую позицию объекта и не заставляет нас видеть фантомное движение в этом направлении, так как мы видим правильное направление движения *независимо* от того, где появляется новый объект. Ответ: в действительности мы не воспринимаем происходящее как движение, пока точка не появляется на новом месте. Вторая точка должна появиться в течение 0,2 с после исчезновения первой, для того чтобы мозг смог объединить эти события в последовательность.

Невосприимчивость к поступающей информации, следующая за узнаванием объекта: 500 мс

Представьте, что вы находитесь в вагоне метро, который медленно подъезжает к станции. Вы проезжаете мимо десятков незнакомцев, стоящих на платформе, но почти не обращаете на них внимания. Вдруг вы замечаете на платформе друга, но поезд еще движется, поэтому вы теряете его из виду. Ваше внимание полностью сосредоточивается на друге: в мозге всплывают разнообразные мысли и чувства, связанные с ним. В этот момент мимо окна по платформе проходит другой ваш друг. Весьма велика вероятность того, что вы его не заметите, так как разум все еще сосредоточен на первом друге. Это явление называется миганием внимания (Stafford, Webb, 2005). Вы можете наблюдать его, позвав на помощь коллегу. Загадайте два слова и скажите их коллеге. Затем сообщите, что собираетесь прочитать список слов, после чего он должен будет вам сказать, встретились ли в списке загаданные слова. Быстро прочитайте длинный список слов — со скоростью приблизительно три слова в секунду. В каком-нибудь месте списка должно встречаться одно из загаданных слов. Если второе загаданное слово следует сразу же за первым или, возможно, через одно-два слова, скорее всего, коллега его не услышит.

Время визуально-моторной реакции (осознанная реакция на неожиданное событие): 700 мс

Это составной интервал, включающий в себя время, за которое ваша визуальная система фиксирует нечто в окружающей среде и инициирует сознательное моторное

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

действие, и время, которое требуется моторной системе на выполнение действия. Например, вы ведете автомобиль. Когда подъезжаете к перекрестку, на светофоре загорается красный свет. Ваша реакция состоит из следующих действий: заметить красный сигнал светофора, принять решение остановиться и нажать на педаль тормоза. Сейчас мы говорим о времени, которое требуется для выполнения трех перечисленных действий. Время, за которое автомобиль фактически *остановится*, не входит в описываемые 700 мс. Интервал до остановки зависит от того, с какой скоростью автомобиль движется, в каком состоянии дорожное покрытие и т. п.

Данная реакция — *не* то же самое, что реакция старого мозга на быстро приближающиеся объекты, заставляющая вас автоматически закрывать глаза, отшатываться или вскидывать руки для защиты. Скорость визуально-моторной реакции приблизительно в 10 раз меньше скорости рефлекса «уворачивания» (см. ранее).

Время осознанной визуально-моторной реакции невозможно оценить точно. У разных людей оно может варьироваться в определенных пределах. Кроме того, оно увеличивается при наличии отвлекающих факторов, из-за сонливости, алкогольного опьянения и, возможно, возраста.

Максимальная продолжительность паузы между фразами разных людей во время диалога: около 1 с

Это приблизительная длительность обычных пауз во время беседы. Если пауза затягивается, то участники беседы — либо тот, кто говорит, либо слушатели — зачастую произносят что-нибудь для поддержания разговора: они вставляют «ну да» или «ага» либо берут на себя роль оратора, то есть сами начинают излагать мысль. Такие паузы заставляют слушателей сосредоточить внимание на ораторе: люди переводят на него взгляд, чтобы понять, чем вызвана заминка. Точная длительность пауз в большой степени зависит от культурных особенностей, но она всегда находится в диапазоне от 0,5 до 2 с.

Время, в течение которого внимание непрерывно уделяется одной задаче: около 10 с

Когда человек выполняет задачу, он разбивает ее на небольшие кусочки — подзадачи. Например, покупка авиабилета включает следующие шаги: 1) зайти на веб-сайт туристического агентства или авиакомпании; 2) ввести данные о перелете; 3) просмотреть результаты; 4) выбрать подходящие рейсы; 5) ввести данные кредитной карты; 6) проверить информацию перед оформлением покупки; 7) завершить покупку. Некоторые подзадачи разбиваются на еще более мелкие составляющие. Например, ввод данных о перелете означает необходимость ввести город отправления, город назначения, даты, время и т. п., кусочек за кусочком. Такое разбиение задач на подзадачи дает в результате набор минимальных подзадач, которые могут быть выполнены без нарушения концентрации. При этом либо промежуточная цель и вся необходимая информация сохраняются в рабочей памяти, либо их

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

можно напрямую взять из окружающей среды. Такие подзадачи нижнего уровня называются единицами задачи (Card et al., 1983). Между выполнением единиц задачи люди обычно отвлекаются, проверяют, не требуют ли их внимания другие вещи и события, возможно, недолго смотрят в окно или делают глоток кофе и т. д. Единицы задачи можно распознать в таких видах деятельности, как редактирование документов, ввод транзакции в приложении работы с чековым счетом, разработка электронных микросхем, управление истребителем во время воздушного боя, и всегда их длительность составляет от 6 до 30 с.

Внедрение временных констант в программное обеспечение: порядки величин

При проектировании интерактивных систем необходимо учитывать требования ко времени, предъявляемые людьми — их пользователями. Однако реализовать поддержку всего огромного ассортимента перцепционных и когнитивных временных констант почти невозможно.

В то же время интерактивные системы разрабатывают инженеры, а не ученые. Перед нами не стоит задача учитывать всевозможные, связанные с мозговой деятельностью временные константы и тактовые периоды. Мы всего лишь должны создавать интерактивные системы, работающие для людей, и работающие хорошо. Такой упрощенный подход дает нам больше свободы и позволяет объединять множество перцепционных и когнитивных временных констант в группы, которые проще изучать, запоминать и использовать в дизайне.

Изучение представленного ранее списка критически важных интервалов позволяет задать удобные условия группировки. Все константы, связанные с восприятием звука, — порядка миллисекунды, поэтому их можно объединить вокруг этого значения. Точная продолжительность — одна миллисекунда, две или три — нам не так важна. Нас интересует только порядок величины.

Аналогично можно создать группы периодов продолжительностью около 10 и 100 мс, 1, 10 и 100 с. Действия, продолжительность которых превышает 100 с, не интересны. Таким образом, необходимую точность при разработке интерактивных систем обеспечивают следующие консолидированные предельные периоды:

- 0,001 с (1 мс) — самый короткий распознаваемый интервал тишины;
- 0,01 с (10 мс) — подсознательное визуальное восприятие, самый короткий замечаемый перерыв в поступлении чернил, звуковой поток;
- 0,1 с (100 мс) — одномоментный подсчет элементов (от одного до четырех), непроизвольное движение глаз (саккада), восприятие причинно-следственной связи, перцепционно-моторная реакция, визуальный поток, рефлекс «уворачивания», идентификация объекта, «установочное окно» сознания, один «момент» осознанной осведомленности;

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- 1 с — средняя продолжительность паузы в диалоге, время осознанной визуально-моторной реакции, мигание внимания;
- 10 с — единица задачи, непрерывное внимание к задаче, один шаг сложной задачи;
- 100 с (1,6 мин) — критически важное решение в чрезвычайной ситуации.

Обратите внимание на то, что эти периоды образуют удобную последовательность: каждый в 10 раз (на порядок) длиннее предыдущего. Таким образом, дизайнерам несложно запомнить данную последовательность — правда, заучить, к *чему* относится каждый из периодов, может быть сложнее.

Учет предельных периодов взаимодействия

Для того чтобы пользователи считали интерактивную систему отзывчивой, она должна отвечать следующим требованиям:

- мгновенно реагировать на факт совершения пользователем действия, чтобы у человека не нарушалось восприятие причинно-следственной связи, даже если возвращение ответа на действие займет некоторое время;
- уведомлять пользователя о том, когда приложение занято выполнением задач, а когда свободно для взаимодействия;
- разрешать пользователям заниматься другими делами во время ожидания завершения работы функции;
- отображать плавную и понятную анимацию;
- позволять прерывать (отменять) длительные операции, если необходимость в них у пользователя исчезла;
- предоставлять информацию, на основе которой пользователи смогут оценить реальный срок выполнения длительных операций;
- подстраиваться под пользователей, чтобы они работали в удобном темпе.

В этих рекомендациях слово «мгновенно» означает задержку длительностью не более 0,1 с. Стоит немного промедлить, и пользовательский интерфейс перейдет с уровня причинно-следственных связей, рефлексов, перцепционно-моторной реакции и автоматического поведения на уровень пауз в диалогах и сознательного поведения (см. врезку «Сколько времени нашему мозгу нужно на то, чтобы...?»). Через 2 с система превысит допустимую длительность пауз в диалогах и переместится на уровень единиц задач, принятия решений и планирования.

Итак, мы перечислили временные константы человеческого восприятия и сознания, а также объединили их в упрощенный набор, который можно описывать такими терминами, как «мгновенно», «скоро», «плавно» и «длительно» (табл. 12.1).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Таблица 12.1. Предельные сроки при взаимодействии человека и компьютера

Предельный интервал	Перцепционные и когнитивные функции	Предельные сроки, учитываемые в дизайне интерактивных систем
0,001 с	• Минимальный распознаваемый интервал тишины	• Максимально допустимая задержка или выпадение звука при звуковой ответной реакции (тональные сигналы, звуковые подписи, музыка)
0,01 с	• Подсознательное восприятие • Минимальный замечаемый перерыв в поступлении чернил в стержне ручки	• Вызов подсознательного ощущения, что изображения или символы уже знакомы пользователю • Генерация звуковых тонов разного уровня • Максимальное время задержки поступления электронных чернил
0,1 с	• Одномоментное определение количества объектов (от 1 до 4) • Непроизвольное движение глаз (саккада) • Рефлекс «уворачивания» • Восприятие причинно-следственной связи • Перцепционно-моторная реакция • Визуальный поток • Идентификация объектов • «Установочное окно» сознания • Длительность момента в человеческом восприятии	• Предполагайте, что пользователь способен подсчитать 1–4 экранных элемента примерно за 100 мс, но если элементов больше четырех, потребуется 300 мс на каждый объект • Ответная реакция на успешную зрительно-моторную координацию, то есть движение указателя мыши, перемещение объекта или изменение его размера, прокрутку, рисование указателем мыши • Ответная реакция на щелчок на кнопке или ссылке • Отображение индикатора «система занята» • допустимое наложение фрагментов речи • максимальная длина интервала между кадрами анимации
1 с	• Максимально продолжительная пауза в диалоге • Время визуально-моторной реакции на неожиданные события • Мигание внимания	• Отображение индикаторов хода выполнения для длительных операций • Завершение запрошенных пользователем операций, таких как открытие окна • Завершение незапрашиваемых операций, таких как автосохранение • Время после представления текста, которое можно использовать для других вычислений, например для перевода объектов из неактивного состояния в активное • Необходимое время ожидания после представления важной информации, прежде чем отображать следующую порцию

продолжение ⇨

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Предельный интервал	Перцепционные и когнитивные функции	Предельные сроки, учитываемые в дизайне интерактивных систем
10 с	• Длительность непрерывной концентрации на задаче • Единица задачи: одна составляющая большой задачи	• Завершение одного шага многошаговой задачи, например одной правки в текстовом редакторе • Завершение ввода пользовательской информации перед выполнением операции • Завершение одного шага мастера (многостраничного диалогового окна)
100 с	• Критически важное решение в чрезвычайной ситуации	• Вся информация, необходимая для принятия решения, должна предоставляться или легко отыскиваться в течение этого времени

0,001 секунды

Как говорилось ранее, человеческая слуховая система чувствительна даже к крайне малым интервалам между звуками. Если интерактивная система предоставляет звуковое содержимое или реакцию на действия пользователя, то ее компонент, ответственный за генерацию звука, должен уметь справляться с неполадками в сетевом подключении, попытками «выдавить» его из списка обрабатываемых процессов, зависаниями и прочими препятствиями на пути передачи звуковой информации. В противном случае возможны возникновение заметных пауз, щелчков, а также рассинхронизация звуковых дорожек. Звуковая ответная реакция и содержимое должны предоставляться хорошо синхронизированными процессами, выполняющимися с высоким приоритетом и при наличии достаточных ресурсов.

0,01 секунды

«Подсознательное» восприятие редко используется в интерактивных системах, если вообще встречается в этой сфере, поэтому углубляться в данную тему мы не будем. Достаточно сказать, что если дизайнер желает повысить узнаваемость определенных визуальных символов или изображений, причем сделать это незаметно для пользователя, то для этого нужно несколько раз на короткое время, до 10 мс за раз, показать пользователю эти изображения или символы. Также стоит упомянуть, что, хотя краткое отображение может повысить узнаваемость объекта, эффект очень слабый, и, конечно, этого недостаточно, чтобы заставить человека полюбить или возненавидеть определенные продукты.

Один из способов генерирования тональных сигналов в программном обеспечении заключается в воспроизведении отрывистых звуков с различными скоростями. Если пауза между звуками меньше 10 мс, их поток воспринимается человеком как единый непрерывный сигнал, высота тона которого определяется в том числе и ча-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

стотой отдельных звуков. Пользователи будут слышать дискретный набор звуков, если интервалы между ними будут превышать 10 мс.

В системах, где данные вводятся с помощью стилуса (пера), задержка в поступлении электронных «чернил» не должна превышать 10 мс, в противном случае пользователи заметят ее, и это вызовет их недовольство.

0,1 секунды

Если программное обеспечение замирает более чем на 0,1 с, прежде чем отреагировать на действие пользователя, восприятие причинно-следственной связи нарушается и ответ приложения уже не кажется результатом действия пользователя. Таким образом, экранные кнопки должны демонстрировать, что щелчок пользователя принят не позже, чем через 0,1 с, иначе пользователь посчитает, что промахнулся мимо кнопки, и нажмет ее еще раз. Это не означает, что привязанная к кнопке *функция* должна завершаться за 0,1 с, — речь идет исключительно о предельном сроке, в течение которого кнопка должна *отреагировать* на нажатие.

Что касается рефлекса «уворачивания», то с ним связан только один аспект дизайна: интерактивные системы не должны пугать пользователей и заставлять их «уворачиваться». Кроме этого факта, рефлекс и его продолжительность не влияют на дизайн интерактивных систем. Трудно представить, какие преимущества могло бы принести использование данного рефлекса при взаимодействии человека и компьютера. Разумеется, игры с громкими звуками, джойстики с резкой обратной силовой связью и трехмерные виртуальные среды в определенных обстоятельствах заставляют пользователей отшатываться. Возможно, если транспортное средство, распознав неминуемое столкновение, будет заставлять водителя отшатнуться, это как-то поможет предотвратить травмы.

Когда пользователь перетаскивает объект или меняет его размер, а реакция приложения запаздывает более чем на 0,1 с, человеку становится трудно помещать объект в нужное место или точно изменять его размер. Таким образом, интерактивные системы должны отдавать наивысший приоритет реагированию на задачи, связанные со зрительно-моторной координацией, для того чтобы подобных задержек никогда не возникало. Если уменьшить время задержки невозможно, то систему следует спроектировать так, чтобы задача не требовала тщательного зрительного координирования движений рук.

Если операция занимает более мгновения, как его воспринимает человек (то есть более 0,1 с), то в это время на экране должен отображаться индикатор «система занята». Если же существует возможность показать такой индикатор в течение 0,1 с, то это усилит впечатление о том, что действие пользователя принято и обрабатывается. В противном случае ответ программного обеспечения должен быть двухфазным: сначала быстрое подтверждение в течение 0,1 с, а затем отображение индикатора занятости (или хода выполнения) в течение 1 с. Подробнее об индикаторах занятости рассказывается чуть позже.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Человеческий мозг умеет менять последовательность событий, попавших в промежуток приблизительно такой величины, прежде чем они достигнут сознания. Человеческая речь особенно хорошо иллюстрирует упорядочивание событий, произошедших в «неправильном» порядке. Например, вы общаетесь с несколькими людьми одновременно, и часть из них начала говорить за мгновение до того, как предыдущий человек закончил фразу (эти два события попали в интервал рассматриваемой длины). Ваш мозг автоматически «распутывает» фразы, и вам кажется, что вы услышали их последовательно, без заметного наложения. В телевизионных и кинофильмах преимуществом данного феномена иногда пользуются для ускорения диалогов, которые в противном случае продолжались бы слишком долго.

Кроме того, 10 кадров в секунду — это приблизительное минимальное значение частоты смены картинок, позволяющее нам воспринимать их как плавную анимацию, хотя по-настоящему плавное анимированное изображение должно содержать около 20 кадров в секунду.

1 секунда

Поскольку 1 с — это максимальная длительность паузы, которую мы ожидаем услышать во время разговора, и поскольку взаимодействие с интерактивной системой представляет собой определенного рода диалог, интерактивные системы не должны делать долгих пауз. В противном случае пользователь начинает задавать себе вопрос: что же происходит? У системы есть примерно одна секунда для того, чтобы либо выполнить запрос пользователя, либо показать индикатор, сообщающий, как долго задача будет выполняться. Нельзя заставлять пользователей ждать.

Если известно, что операция займет несколько секунд, показывать индикатор хода выполнения обязательно. Для интерактивной системы индикатор прогресса — это способ подчиниться ожидаемому протоколу беседы. С помощью индикатора она как бы говорит: «Я работаю над задачей. Вот сколько я уже сделала, и примерно столько времени мне еще понадобится». Подробнее об индикаторах прогресса говорится в дальнейшем.

Помимо этого приблизительно в течение 1 с человек способен осознанно отреагировать на неожиданное событие. Таким образом, если система внезапно выводит на экран какую-то информацию, создатели системы должны исходить из того, что пользователю для реагирования потребуется по меньшей мере 1 с (если только при появлении информации не срабатывает рефлекс «уворачивания», о котором говорилось ранее). Подобная задержка может оказаться полезной в случаях, когда системе необходимо отобразить интерактивный объект, однако у нее не получается как визуализировать его, так и сделать интерактивным в течение положенных 0,1 с. Вместо этого система может сначала показать «фальшивую», неактивную версию объекта, а затем потратить одну секунду на заполнение его деталями и прочими интерактивными возможностями. Современные компьютеры успевают многое сделать за одну секунду!

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

10 секунд

На 10-секундные интервалы люди обычно разбивают задачи планирования и выполнения каких-то более крупных проектов. Можно привести следующие примеры единиц задачи: одиночное редактирование в программе для обработки документов, ввод деталей транзакции в приложение управления банковским счетом, выполнение маневра во время воздушного боя. Программное обеспечение должно поддерживать сегментацию задач на 10-секундные фрагменты.

Пользователи согласны тратить примерно 10 с на подготовку «тяжеловесных» операций, таких как передача файлов или поиск, и начинают терять терпение, если не успевают закончить за это время. Тем не менее обработка результатов на стороне машины может длиться намного больше при условии, что система предоставляет адекватную информацию о ходе выполнения задачи.

Аналогично, каждый шаг в многозадачном диалоговом окне (мастере) должен выполняться максимум за 10 с. Если для выполнения запрашиваемых действий на каком-то шаге пользователю приходится тратить больше времени, вероятно, стоит разбить данный этап на несколько мелких подшагов.

100 секунд (~1,5 мин)

Интерактивные системы, поддерживающие принятие решений в чрезвычайных ситуациях, следует проектировать так, чтобы либо вся необходимая информация сразу предоставлялась пользователю, либо он мог найти ее с минимальными усилиями и за минимальное время. Для ситуаций такого типа лучше всего подходят пользовательские интерфейсы, в который пользователь получает всю критическую информацию, просто переводя взгляд на то место, где она отображается[1] (Isaacs, Walendowski, 2001).

Дополнительные рекомендации по проектированию отзывчивых интерактивных систем

Помимо приведенных рекомендаций по отдельным консолидированным предельным интервалам существуют также общие рекомендации, следуя которым вы будете создавать более отзывчивые интерактивные системы.

Используйте индикаторы занятости

Индикаторы занятости бывают разного уровня сложности. На одном конце диапазона находятся простые статические указатели мыши, например в форме песочных часов. Они дают очень мало информации: всего лишь сообщают, что приложение пока что занято и пользователь не может выполнять в нем другие операции.

[1] Так называемые пользовательские интерфейсы без щелчков (no-click user interface).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

На следующем уровне находятся анимированные индикаторы ожидания, например анимированные курсоры вроде вращающегося цветового колеса в Mac OS. Кроме курсоров популярны также более крупные рисунки в определенном месте экрана, например анимированные изображения «идет загрузка данных» в некоторых веб-браузерах. Анимированные индикаторы ожидания более «дружелюбны», чем статические курсоры, так как они демонстрируют, что система работает, а не сломалась и не зависла в ожидании открытия сетевого подключения или разблокирования данных. Разумеется, анимированные индикаторы занятости должны соответствовать фактическим вычислениям, которые они представляют. Если индикатор создается той же функцией, которая выполняет вычисления, но работает независимо от нее, то он перестает выполнять свое предназначение: анимация продолжает двигаться, даже когда соответствующий процесс зависает или обрывается, тем самым вводя пользователей в заблуждение.

Зачастую можно услышать такое оправдание отсутствию индикатора занятости: «функция должна выполняться быстро, ей этот индикатор ни к чему». Но насколько быстро это «быстро»? А что, если функция не всегда срабатывает «быстро»? Что, если компьютер пользователя медленнее компьютера разработчика или на нем выбрана неоптимальная для данной функции конфигурация? Что, если функция попытается обратиться к временно заблокированным данным? Что, если функция зависит от сетевых услуг, а сеть перегружена или недоступна?

Программное обеспечение должно отображать индикатор занятости для *любой* функции, блокирующей дальнейшие действия пользователя, в течение всего времени выполнения, даже если в нормальных условиях она отрабатывает быстро (то есть менее чем за 0,1 с). Этот индикатор очень полезен для пользователя, так как дает понять, не зависла ли функция по какой-то причине. Помимо этого реализация индикатора занятости уж точно ничему не вредит: когда функция выполняется с обычной скоростью, он появляется и исчезает так быстро, что пользователь его едва замечает.

Используйте индикаторы прогресса

Индикаторы прогресса лучше индикаторов занятости в том смысле, что они позволяют пользователям видеть, как долго функция еще будет выполняться. И снова предельный срок после запуска функции и до появления индикатора прогресса равен 1 с.

Индикаторы прогресса могут быть графическими (индикаторная линейка), текстовыми (счетчик файлов в очереди копирования) или комбинированными, то есть включающими и текст, и графику. Наличие индикатора прогресса делает систему гораздо более отзывчивой в глазах пользователя, несмотря на то что время выполнения операций от этого не уменьшается.

Индикатор прогресса следует отображать для любой операции, длящейся более чем пару секунд. Чем длительнее операция, тем важнее наличие индикатора прогресса.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Многие некомпьютерные устройства оснащены подобными индикаторами, поэтому часто мы воспринимаем их как должное. Например, лифты без указателя этажа нам кажутся «неправильными». Многие люди ни за что не купят микроволновую печь, не умеющую показывать оставшееся время готовки.

Вот несколько рекомендаций по созданию эффективных индикаторов прогресса (McInerney, Li, 2002):

- Сообщайте, какой объем работ осталось сделать, а не сколько уже сделано. Плохой вариант: «3 файла скопированы». Хороший вариант: «3 из 4 файлов скопированы».
- Показывайте общий прогресс, а не прогресс текущего шага. Плохой вариант: «5 секунд осталось на данном шаге». Хороший вариант: «15 секунд осталось».
- Сообщая, какая доля операции выполнена, начинайте с 1 %, а не с 0 %. Пользователи начинают беспокоиться, если индикатор остается на отметке 0 % дольше, чем на секунду или две.
- Аналогично, в конце операции показывайте значение 100 % лишь на короткое мгновение. Если индикатор остается на отметке 100 % дольше, чем на секунду или две, пользователям кажется, что что-то идет не так.
- Анимация индикатора прогресса должна быть плавной, с линейным увеличением значения, а не скачкообразной.
- Используйте значения, удобные для восприятия человеком, не стремитесь к компьютерной точности. Плохой вариант: «240 секунд». Хороший вариант: «Около 4 минут».

Задержки между единицами задачи не так беспокоят пользователей, как задержки внутри единиц задачи

Изучение единиц задач полезно не только для понимания, как (и почему) пользователи разбивают крупные задачи на небольшие фрагменты. Это также позволяет установить, в какие моменты задержки со стороны системы наиболее и наименее опасны и неприятны для пользователей.

Во время выполнения определенной единицы задачи цель и необходимая для ее достижения информация удерживаются в рабочей памяти пользователя или в пределах его поля восприятия. После того как одна единица задачи выполнена, пользователь может отдохнуть и отвлечься и только после этого «подгрузить» в память или в поле зрения данные для выполнения следующей единицы задачи.

Поскольку на протяжении выполнения единицы задачи содержимое рабочей памяти и поля восприятия пользователя должно оставаться относительно стабильным, неожиданные задержки со стороны системы *в эти промежутки времени* особенно вредны и неприятны. Пользователь может потерять мысль или забыть, какое действие собирался выполнить. В противоположность этому задержки *между*

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

единицами задачи не вредят пользователю и не раздражают его, несмотря на то что в целом могут снизить скорость работы.

Различие между задержками со стороны системы во время и между выполнением единиц задачи иногда описывается в рекомендациях по проектированию пользовательских интерфейсов с использованием термина «закрытие задачи». Например, в классическом справочнике «Рекомендации по дизайну взаимодействия человека и компьютера» (Human-Computer Interface Design Guidelines) Брауна (Brown, 1988) говорится так: «Ключевой фактор, определяющий допустимую длительность задержки ответа, — это степень закрытия... Задержка после завершения крупного блока задачи зачастую не беспокоит пользователя и не снижает уровень его производительности. В то же время задержки между мелкими шагами внутри одного крупного блока приводят к тому, что пользователь забывает, какие шаги он планировал выполнить следующими. В целом действия с высоким уровнем закрытия, такие как сохранение законченного документа в файл, менее чувствительны к задержкам со стороны системы. Действия с самыми низкими уровнями закрытия, например ввод символа и наблюдение за его появлением на экране, наиболее чувствительны к задержкам».

Вывод: если задержка с ответом системы неизбежна, она должна произойти между единицами задачи, а не во время их выполнения.

В первую очередь показывайте важную информацию

Если интерактивная система сначала отображает важную информацию и только после этого — детали и вспомогательные сведения, у пользователя создается впечатление, что она работает быстрее. Не дожидайтесь полной визуализации представления, прежде чем показывать его пользователю. Дайте пользователю что-то, над чем он сможет подумать, а после этого постарайтесь как можно скорее загрузить недостающие данные.

У этого подхода несколько преимуществ. Во-первых, он отвлекает внимание пользователя от того факта, что часть информации пока отсутствует, и заставляет поверить, что компьютер очень быстро выполнил запрос. Исследования подтверждают, что пользователи предпочитают вариант с постепенным выводом результатов варианту с индикатором прогресса (Geelhoed, Toft, Roberts, Hyland, 1995). Во-вторых, при постепенном отображении результатов пользователи скорее получают возможность начать планировать следующую единицу задачи. Наконец, это связано с упомянутым ранее минимальным временем сознательной реакции на увиденное. Данный подход обеспечивает системе лишнюю секунду на подготовку, прежде чем пользователь попытается что-то сделать. Вот несколько примеров:

- **Приложение для редактирования документов:** когда вы открываете документ, программа показывает первую страницу сразу же, как только считывает ее, не дожидаясь, пока файл загрузится целиком.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- **Функция поиска по веб-сайту или базе данных:** когда вы запускаете выполнение поискового запроса, приложение выводит найденные элементы по мере их извлечения, в то же время продолжая искать другие подходящие объекты.

Бывает, что изображения с высоким разрешением визуализируются медленно, особенно в веб-браузерах. Для того чтобы улучшить впечатление пользователя о скорости визуализации, система может быстро показать картинку с низким качеством, а затем повторно визуализировать ее в более высоком разрешении. Поскольку наша визуальная система воспринимает изображения как целостные объекты, такой вариант кажется нам более быстрым, чем медленная построчная загрузка изображения сразу в хорошем разрешении (рис. 12.3). Исключением является текст: не рекомендуется сначала выводить страницу текста с низким разрешением, а затем обновлять ее с повышением качества — это раздражает пользователей (Geelhoed et al., 1995).

Рис. 12.3. Если для вывода изображения требуется более двух секунд, начните с отображения картинки в низком разрешении (вариант *а*) — это лучше, чем построчно загружать ее в максимальном разрешении (вариант *б*)

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Имитируйте тяжеловесные вычисления во время задач, требующих зрительно-моторной координации

В интерактивных системах для достижения определенных целей пользователям приходится выполнять последовательности быстрых корректировок со зрительной координацией движений рук. В качестве примеров можно привести прокрутку документа, перемещение героя игры по полю, изменение размера окна, перетаскивание объекта на новое место. Если реакция системы на действия пользователя задерживается более чем на 0,1 с, достижение цели затрудняется. Если система не может обновлять представление достаточно быстро для того, чтобы укладываться в предельный срок зрительно-моторной координации, обеспечьте для начала облегченную обратную связь. Реальную операцию можно будет выполнить, когда цель пользователя станет ясна.

Графические редакторы имитируют обратную связь, когда показывают не объекты целиком, а лишь их полупрозрачные очертания во время их перемещения или изменения размеров. Некоторые приложения для редактирования документов вносят быстрые небрежные изменения во внутренние структуры данных документов, оперативно представляя на экране результат пользовательских действий, и приводят данные в порядок позже, когда на это есть время.

Старайтесь предугадать действия пользователей

По возможности старайтесь на один шаг обгонять пользователей. Периоды низкой нагрузки на приложение можно использовать для предварительного вычисления реакции на наиболее вероятные запросы. Такие периоды обязательно будут встречаться — такова особенность работы человека с компьютером. Обычно интерактивные системы проводят довольно много времени в ожидании данных от пользователя. Не тратьте это время попусту! Воспользуйтесь им для подготовки информации, которую пользователь с большой вероятностью захочет получить. А если не захочет, невелика беда. Программа обрабатывала эти прогнозы в свое свободное время, ей не приходилось отбирать ресурсы у других важных процессов. Вот пара примеров фоновой обработки для «предугадывания» желаний пользователя:

- Функция поиска по тексту может найти *следующее* вхождение искомого слова, пока пользователь оценивает текущее. Когда пользователь прикажет системе перейти к следующему результату поиска, он уже будет подготовлен, и создастся впечатление, что команда отработала очень быстро.
- Приложение для просмотра документов подготавливает следующую страницу в то время, пока пользователь читает текущую. Когда пользователь нажмет на кнопку перехода к следующей странице, та уже будет готова к отображению.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Обрабатывайте вводимые пользователем данные согласно приоритету, а не в порядке поступления

Порядок выполнения задач зачастую имеет очень большое значение. Слепое выполнение запросов в порядке поступления может привести к излишней трате времени и ресурсов и даже создать дополнительный объем работы. Интерактивные системы должны стараться менять порядок задач в очереди, так как во многих случаях это позволяет более эффективно выполнить весь набор запросов.

Сотрудники авиакомпаний выполняют нелинейную обработку входных данных, когда проходят вдоль длинных очередей на регистрацию, находят пассажиров рейсов с ближайшим временем отправки и регистрируют их вне очереди. В веб-браузерах щелчок на кнопке Назад или Стоп (или на любой ссылке) *мгновенно* прерывает процесс загрузки и отображения текущей страницы. Учитывая, как долго иногда загружаются и выводятся определенные страницы, возможность остановить этот процесс критически важна для одобрения системы пользователем.

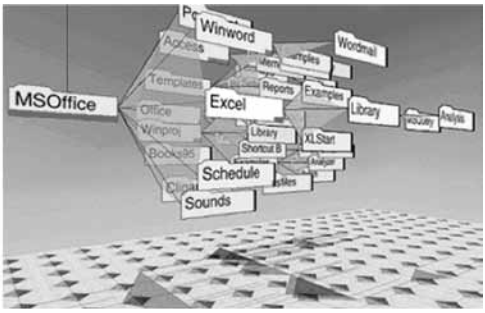
Отслеживайте выполнение временных требований; снижайте качество работы, чтобы не допускать опозданий

В интерактивную систему несложно встроить возможности оценки того, насколько хорошо выполняются требования к предельным срокам. Если стало очевидно, что предельный срок пропущен, либо существует риск не уложиться в следующий интервал, то система может применить более простые и быстрые методы обработки, обычно приводящие к временному снижению качества выходных данных. Данный подход должен ориентироваться на *реальное* время, а не на тактовую частоту процессора, для того чтобы на разных системах достигался одинаковый уровень отзывчивости.

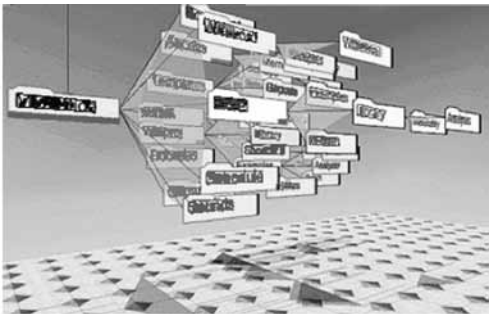
Эта техника применяется в некоторых анимационных интерактивных системах. Как уже говорилось, для того чтобы анимация воспроизводилась достаточно плавно, частота кадров должна быть около 20 в секунду. В конце 1980-х годов ученые из исследовательского центра Xerox Palo Alto Research Center (PARC) разработали программный механизм представления интерактивной анимации, в котором частоте кадров отдается наивысший приоритет (Robertson, Card, Mackinlay, 1989, 1993). Если графический процессор не справляется с обеспечением минимальной частоты кадров — изображения слишком сложны или пользователь активно взаимодействует с ними, — то визуализация упрощается за счет отказа от таких деталей, как текстовые метки, эффект объема, подсветка и затенение, оттенки цвета. Смысл в том, что лучше временно упростить трехмерное анимированное изображение до плоской графики, чем позволить частоте кадров упасть на недопустимо низкий уровень.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

На базе описанного графического процессора работает приложение Cone Tree, разработанное в PARC. Это интерактивный дисплей, представляющий иерархическую структуру данных, например каталоги и подкаталоги с файлами (рис. 12.4). Пользователь щелкает на любой части дерева и поворачивает его. У приложения может не хватать времени на визуализацию всех деталей каждого кадра, пока дерево поворачивается. В таком случае анимация становится отрывистой, с задержками. Однако сэкономить время можно за счет, скажем, визуализации имен файлов — пусть они во время вращения отображаются как черные пятна, а не как разборчивый текст. Когда пользователь прекратит вращение и отпустит дерево, названия файлов снова будут визуализироваться с полной детализацией. Большинство людей даже не заметят ухудшения изображения во время вращения, приписав невозможность прочитать метки расфокусировке из-за движения.



а



б

Рис. 12.4. Пока пользователь вращает дерево, приложение Cone Tree (*а*) визуализирует метки папок как черные пятна (*б*)

Обеспечивайте своевременную ответную реакцию даже в Сети

Не исключаю, что разработчики веб-приложений отмели идею о соблюдении рассмотренных ранее предельных сроков как невыполнимую.

Действительно, соблюдать предельные сроки выполнения задач в Сети сложно, а зачастую попросту невозможно. Тем не менее речь идет о психологических вре-

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

менных константах, запрограммированных в мозге человека в ходе миллионов лет эволюции и управляющих нашим восприятием и оценкой отзывчивости. Это не произвольные значения, которые мы можем самовольно корректировать с учетом ограничений Сети или любой другой технологической платформы. Если интерактивная система — даже веб-приложение — не соблюдает предельных сроков, то пользователи *неизменно* воспринимают ее как неотзывчивую. Разумеется, большинство веб-приложений обладают низкой отзывчивостью. Вопрос в том, есть ли у дизайнеров и разработчиков возможность максимизировать отзывчивость приложений в Сети? Далее перечислены несколько подходов:

- Минимизируйте размер и число изображений.
- Прибегайте к помощи быстро отображаемых эскизов изображений и кратких обзоров, но предусматривайте способы перехода к подробным представлениям, если у пользователя возникнет в этом необходимость.
- Если объем данных слишком велик и их единовременное отображение занимает много времени, добавляйте в систему общее уменьшенное представление *всех* данных и разрешайте пользователю углубляться в определенные части данных до требуемого уровня детализации.
- Для проектирования стиля и макета страниц применяйте таблицы CSS (Cascading Style Sheets), а не презентационный HTML-код, фреймы и таблицы.
- Используйте встроенные компоненты браузера, такие как диалоговые окна с сообщениями об ошибках, вместо создания собственных аналогов в HTML.
- Загружайте в браузер апплеты и сценарии, применяйте методы AJAX.

Отзывчивость очень важна

Выполняя рекомендации, представленные в этой главе, а также дополнительные советы по повышению отзывчивости систем, которые можно найти в книге Джонсона (Johnson, 2007), дизайнеры взаимодействия и разработчики могут создавать системы, отвечающие предельным срокам взаимодействия с человеком в реальном времени. Такие системы пользователи будут считать надежными и отзывчивыми.

Однако представителям сферы производства программного обеспечения в первую очередь необходимо согласиться со следующими утверждениями, касающимися отзывчивости:

- Отзывчивость систем чрезвычайно важна для пользователей.
- Отзывчивость — это не то же самое, что производительность. Проблемы отзывчивости невозможно решить, просто подогнав производительность или установив более быстрое оборудование.
- Отзывчивость — это вопрос дизайна, а не реализации.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Практика показывает, что установка быстрых процессоров далеко не всегда решает проблему отзывчивости. Скорость современных персональных компьютеров сравнима со скоростью суперкомпьютеров, существовавших 30 лет назад, однако людям все так же приходится ждать, и они не прекращают критиковать неотзывчивые устройства. Через 10 лет, когда персональные компьютеры и электронные устройства будут сравнимы по мощности с *современными* наиболее мощными суперкомпьютерами, отзывчивость все также же будет оставаться острой проблемой, так как приложения станут предъявлять еще более высокие требования к «железу» и сетевым подключениям. Например, сегодняшние текстовые редакторы умеют выполнять в фоновом режиме проверку *орфографии и пунктуации*, вполне возможно, будущие версии научатся выполнять фоновую проверку *фактов*, сверяясь с данными из Интернета. Помимо этого через 10 лет приложения, вероятно, будут основываться на следующих возможностях и технологиях:

- дедуктивное мышление;
- распознавание изображений;
- генерация и распознавание речи в режиме реального времени;
- загрузка файлов объемом порядка терабайта;
- беспроводной обмен данными между десятками бытовых электроприборов;
- объединение данных из тысяч удаленных баз данных;
- сложный поиск по всей Сети.

В результате системы будут намного сильнее нагружать компьютеры, чем это происходит сегодня. История подтверждает, что по мере увеличения мощности компьютеров приложения также начинают «отъедать» все больше и больше мощности. Таким образом, несмотря на постоянно повышающуюся производительность, проблема отзывчивости никогда сама собой не исчезнет.

Подробнее о недостатках дизайна, снижающих отзывчивость, принципах дизайна отзывчивых систем и дополнительных техниках обеспечения отзывчивости рассказывается в книге Джонсона (Johnson, 2007).

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Резюме

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Во введении я сказал, что применять рекомендации по дизайну взаимодействий в реальных условиях непросто, и это нельзя делать небрежно, не считаясь ни с чем более. Возникают различные ограничения, для преодоления которых нужно идти на компромиссы. Иногда дизайнерам приходится нарушить одну рекомендацию, чтобы выполнить другую, поэтому им необходимо понимать, в каких ситуациях какие рекомендации обладают наивысшим приоритетом.

Вот почему дизайн взаимодействий — это навык, который невозможно приобрести, бездумно следуя рецепту. Получение этого навыка означает не только изучение рекомендаций, но и анализ того, каким правилам предпочтительнее следовать в каждой из возможных ситуаций.

Моей целью при написании этой книги было коротко познакомить вас с перцепционной и когнитивной психологией человека, лежащей в основе рекомендаций по дизайну взаимодействий. Я надеюсь, что теперь, когда у вас есть базовые знания, вам стали более понятны правила построения пользовательских интерфейсов, с которыми вы работали раньше. Они больше не должны казаться вам произвольным набором догм, сформулированным неким гуру пользовательских интерфейсов. Кроме того, вы увидели, что в основе всех наборов рекомендаций по проектированию пользовательских интерфейсов лежат одни и те же принципы (см. приложение). Наконец, в вашем арсенале появилось множество новых инструментов, помогающих интерпретировать, находить компромиссы и применять подобные рекомендации в условиях реального мира.

Предостережение

Технологии — особенно компьютерные технологии — развиваются с огромной скоростью. Уровень компьютерных интерактивных систем повышается так быстро, что крайне сложно успеть опубликовать книгу до того, как упомянутые в ней технологии и примеры дизайна устареют.

В то же время фундаментальные основы человеческого восприятия, обучения и мышления меняются куда медленнее. Базовые операции, с помощью которых человек воспринимает и осознает реальность, остаются относительно стабильными уже десятки, если не сотни тысяч лет. В долгосрочной перспективе перцепционные и когнитивные функции человека продолжат развиваться, однако в ближайшее время, пока эту книгу можно будет найти на прилавках магазинов и в библиотеках, никаких более или менее значительных изменений не произойдет. Однако люди уже применяют технологии для усовершенствования процессов восприятия, запоминания и мышления и будут применять их все шире и шире. Таким образом, человеческое восприятие и мышление *будет* меняться в течение последующих десятилетий по мере того, как наши вспомогательные инструменты станут множиться и улучшаться, а наша зависимость от них — усиливаться.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

И еще: *знания* о человеческом восприятии и сознании, как и компьютерные технологии, стремительно расширяются. За последние 20 лет произошел огромный прорыв в понимании тонкостей работы человеческого мозга, чему в значительной степени способствовали новые исследовательские инструменты: функциональная магнитно-резонансная томография, системы отслеживания движений глаз и модели нейронных сетей. Благодаря им когнитивная психология сумела выйти за пределы «моделей черного ящика», которые всего лишь *предсказывали* поведение. Современные модели объясняют, как мозг обрабатывает и хранит информацию и что определяет наше поведение. В этой книге я постарался в доступной форме рассказать об этих потрясающих новых открытиях, имеющих огромное значение для дизайна интерактивных систем. Я сделал это, отдавая себе отчет в том, что, как и компьютерные технологии, знания о когнитивной и перцепционной психологии продолжат расширяться, а некоторые факты, изложенные в данной книге, будут опровергнуты. Тем не менее гораздо лучше разрабатывать интерактивные системы, основываясь на достоверных сведениях об особенностях человеческого мышления и восприятия, чем создавать пользовательские интерфейсы, *никак* не учитывающие человеческую психологию.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Приложение

Правила разработки пользовательских интерфейсов

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Далее вы найдете краткую выборку из ранее публиковавшихся руководств по разработке пользовательских интерфейсов.

Из статьи «Design rules based on analysis of human error» (Norman, 1983a)

Выводы на основе исследований

- Ошибки, связанные с переключением режимов, сигнализируют о необходимости более качественной ответной реакции.
- Ошибки, связанные с выбором неправильных объектов, сигнализируют о необходимости улучшения конфигурации системы.
- Недостаточное единообразие приводит к ошибкам.
- Ошибочные действия сигнализируют о необходимости избегать пересекающихся последовательностей команд.
- Проблемы с активацией объектов подтверждают важность вспомогательных средств, упрощающих вспоминание.
- Люди непременно будут делать ошибки, поэтому создавайте системы, нечувствительные к ошибкам.

Уроки

- **Обратная связь:** необходимо ясно и недвусмысленно сигнализировать о текущем состоянии системы и предоставлять наборы параметров, характерные для конкретного режима, чтобы в итоге предотвращать ошибки режима.
- **Схожесть последовательностей ответных реакций:** разные классы действий должны обладать непохожими последовательностями команд (или шаблонов меню) для предотвращения выбора пользователем неправильных объектов и выполнения ошибочных действий.
- **Действия должны быть обратимыми:** обеспечению обратимости необходимо уделять как можно больше внимания; кроме того, если ошибочное действие приводит к дорогостоящим последствиям, его должно быть трудно выполнить, то есть необходимо предотвращать случайные опасные действия.
- **Единообразие системы:** система должна быть согласована как в части структуры, так и в части дизайна команд, для того чтобы минимизировать трудности с вспоминанием операций.

Из книг «Designing the user interface: Strategies for effective human–computer interaction» (Shneiderman, 1987) и « Designing the user interface: Strategies for effective human–computer interaction» (Shneiderman, Plaisant, 2009)

- Стремитесь к согласованности и единообразию.
- Обеспечьте одинаковое удобство в использовании.
- Предусмотрите информативную ответную реакцию.
- Прорабатывайте замкнутые потоки решения задач.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- Предотвращайте ошибки.
- Обеспечивайте возможность легкой отмены действия.
- Пусть пользователи чувствуют, что контроль в их руках.
- Минимизируйте нагрузку на кратковременную память.

Из статьи «Heuristic evaluation of user interfaces» (Nielsen, Molich, 1990)

- Единообразие и стандарты.
- Очевидность состояния системы.
- Соответствие системы реальному миру.
- Пользовательский контроль и свобода.
- Предотвращение ошибок.
- Узнавание, а не вспоминание.
- Гибкость и эффективность использования.
- Эстетичный и минималистичный дизайн.
- Помогайте пользователям распознавать, диагностировать и устранять ошибки.
- Предоставляйте интерактивную документацию и справку.

Из книги «User interface design and evaluation»(Stone et al., 2005)

- **Очевидность**: первый шаг на пути к достижению цели должен быть понятен.
- **Легкость взаимодействия**: элемент управления подразумевает правильный вариант использования.
- **Ответная реакция**: должно быть очевидно, что произошло только что или происходит в данный момент.
- **Простота**: как можно проще, и фокусируйтесь на задаче.
- **Структура**: разумная организация содержимого.
- **Единообразие**: сходство обеспечивает предсказуемость.
- **Выносливость**: предотвращайте ошибки, помогайте устранять последствия.
- **Доступность**: может использоваться всеми пользователями из целевой аудитории, несмотря на их физические недостатки, устройство, используемое для доступа, и условия работы.

Из книги «GUI bloopers 2.0: Common user interface design don'ts and dos» (Johnson, 2007)

Принцип 1. Фокусируйтесь на пользователях и их задачах, а не на технологии

- Понимайте пользователей.
- Понимайте задачи.
- Учитывайте контекст использования программного обеспечения.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Принцип 2. Сперва продумывайте функцию и только потом представление

- Разрабатывайте концептуальную модель.

Принцип 3. Придерживайтесь пользовательского взгляда на задачу

- Стремитесь к естественности.
- Используйте терминологию пользователя, а не привычный для вас словарь.
- «Внутренности» программы должны оставаться внутри программы.
- Находите правильный компромисс между мощностью и сложностью.

Принцип 4. Проектируйте с учетом популярных задач

- Часто требуемые результаты должны быть легко достижимы.
- Существует два типа популярности: в зависимости от того, *как много* пользователей выполняют задачу и *как часто* эта задача выполняется.
- Учитывайте в дизайне основные, распространенные случаи, не пытайтесь досконально проработать экстремальные варианты.

Принцип 5. Не усложняйте пользователю его задачу

- Не создавайте для пользователя дополнительных проблем.
- Не заставляйте пользователей докапываться до результата путем исключения.

Принцип 6. Упрощайте обучение

- Думайте «снаружи внутрь», а не «изнутри наружу».
- Единообразие, единообразие, единообразие.
- Обеспечивайте среду с низким уровнем риска.

Принцип 7. Предоставляйте информацию, а не простые данные

- Тщательно прорабатывайте представления информации, обращайтесь за помощью к профессионалам.
- Экран принадлежит пользователю.
- Сохраняйте неизменность представления.

Принцип 8. Стремитесь к отзывчивости

- Мгновенно подтверждайте пользовательские действия.
- Пользователям должно быть понятно, когда приложение занято, а когда нет.
- Разрешайте пользователям заниматься другими задачами во время ожидания.
- Создавайте плавную и четкую анимацию.
- Предоставляйте пользователям возможность прерывать ненужные длительные операции.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

- Предоставляйте пользователям данные для оценки того, как долго будет выполняться операция.
- Старайтесь позволять пользователям работать в привычном для них темпе.

Принцип 9. Проводите испытания на пользователях и вносите исправления

- Результаты тестов могут удивлять даже опытных дизайнеров.
- График создания приложения должен предусматривать время на исправление проблем, обнаруженных при тестировании.
- У тестирования две цели: информационная и социальная.
- Существуют тесты для всех времен и целей.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Библиография

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Angier, N. (2008). Blind to change, even as it stares us in the face. *New York Times*. April 1, 2008. www.nytimes.com/2008/04/01/science/01angi.html.

Arons, B. (1992). A review of the cocktail party effect. *Journal of the American Voice I/O Society*, *12*, 35–50.

Apple Computer (2009). *Apple human interface guidelines*. developer.apple.com/mac/library/documentation/UserExperience/Conceptual/AppleHIGuidelines

Barber, R., & Lucas, H. (1983). System response time, operator productivity, and job satisfaction. *Communications of the ACM*, *26*(11), 972–986.

Bays, P. M., & Husain, M. (2008). Dynamic shifts of limited working memory resources in human vision. *Science*, *321*, 851–854.

Beyer, H., & Holtzblatt, K. (1997). *Contextual design: A customer-centered approach to systems design*. Morgan-Kaufmann Publishers.

Blauer, T. (2007). On the startle/flinch response. *Blauer tactical intro to the spear system: Flinching and the first two seconds of an ambush*. YouTube video: http://www.youtube.com/watch?v=p6CdH6xHcgo.

Broadbent, D. E. (1975). The magical number seven after fifteen years. In A. Kennedy & A. Wilkes (Eds.), *Studies in long-term memory* (pp. 3–18). Londmon: Wiley.

Brown, C. M. (1988). *Human–computer interface design guidelines*. Norwood, NJ: Ablex Publishing Corporation.

Boulton, D. (2009). Cognitive science: The conceptual components of reading & what reading does for the mind. Interview of Dr. Keith Stanovich, Children of the Code website: www.childrenofthecode.org/interviews/stanovich.htm

Card, S. (1996). Pioneers and settlers: Methods used in successful user interface design. In M. Rudisill, C. Lewis, P. Polson, & T. McKay (Eds.), *Human-computer interface design: Success cases, emerging methods, real-world context*. San Francisco: Morgan Kaufmann.

Card, S., Moran, T., & Newell, A. (1983). *The psychology of human–computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Card, S., Robertson, G., & Mackinlay, J. (1991). *The Information Visualizer, an Information Workspace. Proceedings of ACM CHI'91,* 181–188.

Carroll, J., & Rosson, M. (1984). Beyond MIPS: Performance is not quality. *BYTE*, 168–172.

Cheriton, D. R. (1976). Man–machine interface design for time-sharing systems. *Proceedings of the ACM National Conference*, 362–380.

Chi, E. H., Pirolli, P., Chen, K., & Pitkow, J. (2001). Using information scent to model user information needs and actions on the web. *Proceedings of ACM SIGCHI Conference on Computer–Human Interaction (CHI 2001)*, 490–497.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Clark, A. (1998). *Being there: Putting brain, body, and world together again*. Cambridge, MA: MIT Press.

Cowan, N., Chen, Z., & Rouder, J. (2004). Constant capacity in an immediate serial-recall task: A logical sequel to Miller (1956). *Psychological Science*, *15*(9), 634–640.

Duis, D., & Johnson, J. (1990). Improving user-interface responsiveness despite performance limitations. *Proceedings of IEEE CompCon'90*, 380–386.

Geelhoed, E., Toft, P., Roberts, S., & Hyland, P. (1995). To influence time perception. *Proceedings of ACM CHI'95*, *5*, 272–273.

Grudin, J. (1989). The case against user interface consistency. *Communications of the ACM*, *32*(10), 1164–1173.

Hackos, J., & Redish, J. (1998). *User and task analysis for interface design*. New York: Wiley.

Isaacs, E., & Walendowski, A. (2001). *Designing from both sides of the screen: How designers and engineers can collaborate to build cooperative technology*. Indianapolis, Indiana: SAMS.

Johnson, J. (1987). How faithfully should the electronic office simulate the real one? *SIGCHI Bulletin*, 21–25.

Johnson, J. (1990). Modes in non-computer devices. *International Journal of Man–Machine Studies*, *32*, 423–438.

Johnson, J. (2007). *GUI bloopers 2.0: Common user interface design don'ts and dos. San Francisco, CA:* Morgan-Kaufmann Publishers.

Johnson, J., & Henderson, A. (2002). Conceptual models: Begin by designing what to design. *Interactions*, 25–32.

Johnson, J., Roberts, T., Verplank, W., Smith, D. C., Irby, C., Beard, M., & Mackey, K. (1989). The xerox star: A retrospective. *IEEE Computer*, *September*, 11–29.

Jonides, J., Lewis, R. L., Nee, D. E., Lustig, C. A., Berman, M. G., & Moore, K. S. (2008). The mind and brain of short-term memory. *Annual Review of Psychology*, *59*, 193–224.

Koyani, S. J., Bailey, R. W., & Nall, J. R. (2006). *Research-based web design and usability guidelines*. US Department of Health and Human Services. Website: usability.gov/pdfs/guidelines.html.

Krug, S. (2005). *Don't make me think: A common sense approach to web usability* (2nd ed.). Indianapolis: New Riders Press.

Lambert, G. (1984). A comparative study of system response time on program developer productivity. *IBM Systems Journal*, *23*(1), 407–423.

Landauer, T. K. (1986). How much do people remember? Some estimates of the quantity of learned information in long-term memory. *Cognitive Science*, *10*, 477–493.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Liang, P., Zhong, N., Lu, S., Liu, J., Yau, Y., Li, K., & Yang, Y. (2007). *The neural mechanism of human numerical inductive reasoning process: A combined ERP and fMRI study*. Berlin: Springer Verlag.

Lindsay, P., & Norman, D. A. (1972). *Human information processing*. New York and London: Academic Press.

Marcus, A. (1992). *Graphic design for electronic documents and user interfaces*. Reading, MA: Addison-Wesley.

Marr, D. (1982). *Vision*. New York, NY: W. H. Freeman. p. 101, Figure 3-1, attributed to R. C. James.

McInerney, P., & Li, J. (2002). Progress indication: Concepts, design, and implementation, July, IBM. *Developer Works*. Website: www-128.ibm.com/developerworks/web/library/us-progind.

Microsoft Corporation (2009), *Windows User Experience Interaction Guidelines*: http://www.msdn.microsoft.com/en-us/library/aa511258.aspx

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97.

Miller, R. (1968). Response time in man–computer conversational transactions. *Proceedings of IBM Fall Joint Computer Conference Vol. 33, 267–277*.

Monti, M. M., Osherson, D. N., Martinez, M. J., & Parsons, L. M. (2007). Functional neuroanatomy of deductive inference: A language-independent distributed network. *NeuroImage*, *37*(3), 1005–1016.

Mullet, K., & Sano, D. (1994). *Designing visual interfaces: Communications oriented techniques*. Prentice-Hall.

Nielsen, J. (2003). Information foraging: Why Google makes people leave your site faster, Alertbox, June 30, 2003.

Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. *Proceedings of ACM CHI'90 Conference*. (Seattle, WA, 1–5 April), 249–256.

Norman, D. A. (1983a). Design rules based on analysis of human error. *Communications of the ACM*, *26*(4), 254–258.

Norman, D. A. (1983b). Design principles for human–computer interfaces. In A. Janda (Ed.), *Proceedings of the CHI-83 conference on human factors in computing systems (Boston)*. New York: ACM. Reprinted in *Readings in Human-Computer Interaction*, ed. by Ronald M. Baecker and William A. S. Buxton. San Mateo, CA: Morgan-Kaufmann Press (1987).

Norman, D. A., & Draper, S. W. (1986). *User centered system design: New perspectives on human–computer interaction*. Hillsdale, NJ: CRC.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Oracle Corporation/Sun Microsystems (2001). *Java Look and Feel Design Guidelines*, 2nd ed. http://www.java.sun.com/products/jlf/ed2/book/index.html

Perfetti, C., & Landesman, L. (2001). The truth about download time. Web article, User Interface Engineering, Jan 31, 2001, http://uie.com/articles/download_time/

Redish, G. (2007). *Letting go of the words: Writing web content that works*. San Francisco: Morgan-Kaufmann Publishers.

Robertson, G., Card, S., & Mackinlay, J. (1989). The cognitive co-processor architecture for interactive user interfaces. *Proceedings of the ACM Conference on User Interface Software and Technology (UIST'89)*. November 1989, ACM Press, 10–18.

Robertson, G., Card, S., & Mackinlay, J. (1993). Information visualization using 3D interactive animation. *Communications of the ACM*, *36*(4), 56–71.

Rushinek, A., & Rushinek, S. (1986). What makes users happy? *Communications of the ACM*, *29*, 584–598.

Sapolsky, R. M. (2002). *A primate's memoir: A neuroscientist's unconventional life among the Baboons*. New York: Scribner.

Schneider, W., & Shiffrin, R. M. (1977). Controlled and automatic human information processing: 1. Detection, search, and attention. *Psychological Review*, *84*, 1–66.

Schrage, M. (2005). The password is fayleyure. *Technology Review*, *March*. http://www.technologyreview.com/read_article.aspx?ch=specialsections&sc=security&id=16350.

Shneiderman, B. (1984). Response time and display rate in human performance with computers. *ACM Computing Surveys*, *16*(4), 265–285.

Shneiderman, B. (1987). *Designing the user interface: Strategies for effective human–computer interaction* (1st ed.). Reading, MA: Addison-Wesley.

Shneiderman, B., & Plaisant, C. (2009). *Designing the user interface: Strategies for effective human–computer interaction* (5th ed.). Reading, MA: Addison-Wesley.

Simon, H. A. (1969). *The sciences of the artificial*. Cambridge, MA: MIT Press.

Simons, D. J., & Levin, D. T. (1998). Failure to detect changes in people during a real-world interaction. *Psychonomic Bulletin and Review*, *5*, 644–669.

Smith, S. L., & Mosier, J. N. (1986). *Guidelines for designing user interface software*. Springfield, VA: National Technical Information Service. *Technical Report ESD-TR-86-278*.

Soegaard, M. (2007). Gestalt Principles of form perception. Web article, Interaction-Design.org, http://www.interaction-design.org/encyclopedia/gestalt_principles_of_form_perception.html

Sohn, E. (October 8, 2003). It's a math world for animals. *Science News for Kids*. http://www.sciencenewsforkids.org/articles/20031008/Feature1.asp.

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks

Sousa, D. A. (2005). *How the brain learns to read*. Thousand Oaks, CA: Corwin Press.

Stafford, T., & Webb, M. (2005). *Mind hacks: Tips and tools for using your brain*. Sebastapol, CA: O'Reilly.

Stone, D., Jarrett, C., Woodroffe, M., & Minocha, S. (2005). *User interface design and evaluation*. San Francisco: Morgan Kaufmann Publishers.

Thadhani, A. (1981). Interactive user productivity. *IBM Systems Journal*, *20*(4), 407–423.

Thagard, P. (2002). *Coherence in thought and action*. Cambridge, MA: MIT Press.

Waloszek, G. (2005), Vision and Visual Disabilities: An Introduction, SAP Design Guild. http://www.sapdesignguild.org/editions/highlight_articles_01/vision_physiology.asp

Ware, C. (2008). *Visual thinking for design*. San Francisco, CA: Morgan-Kaufmann Publishers.

Weinschenk, S. M. (2009). *Neuro web design: What makes them click? Berkeley, CA:* New Riders.

Wolfmaier, T. (1999). Designing for the color-challenged: A challenge. ITG Publication, March 1999, 2.1. http://www.internettg.org/newsletter/mar99/accessibility_color_challenged.html

Больше книг по дизайну и творчеству бесплатно:
t.me/dbooks & vk.com/dbooks