



Журнал  
для  
пользователей  
компьютеров

#9 (42)  
сентябрь 2001

Издательство "Техно-ПРЕСС", С.-Петербург

## Энергетика системного блока

Bluetooth многоликий и  
многообещающий

Ворота для ваших  
мегабайтов

ADSL сегодня

Умные  
интерфейсы

Интернет и usability

Сотворение мира



# WEB-камеры — ВСЕВИДУЩЕЕ ОКО ИНТЕРНЕТ

**КОМПЬЮТЕРЫ**

Pentium 4 и чипсеты для него .....	2
Ворота для ваших гигабайтов.....	6
Энергетика системного блока.....	9
Дитя войны.....	10

**НАЧИНАЮЩИМ**

Тайны системного реестра.....	12
-------------------------------	----

**ФОТОИСКУССТВО И ПК**

Очарование шара.....	16
----------------------	----

**МУЗЫКАЛЬНЫЙ ПК**

Cool Edit Pro — космические технологии в музыке.....	18
--	----

**ПЕРИФЕРИЯ**

Bluetooth, многоликий и многообещающий.....	23
Планшетные сканеры: ПЗС против КДИ.....	27

**ВАШЕ ЗДОРОВЬЕ**

Мир олову, война свинцу!.....	30
-------------------------------	----

**ИНТЕРНЕТ**

Окно в Европу.....	32
Сделай сам: гостевая книга.....	35
Интернет и usability.....	36
PHP — мощь Pearl и гибкость Javascript.....	40
Копилка веб-мастера.....	41

**ЦИФРОВАЯ СВЯЗЬ**

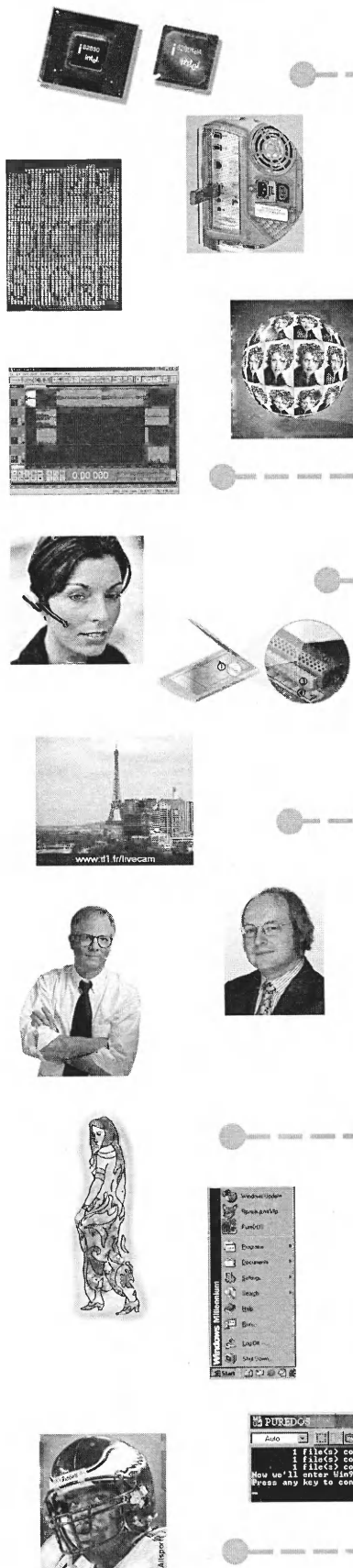
ADSL сегодня.....	45
-------------------	----

**НОМО COMPUTERUS**

Сто выдающихся деятелей IT ушедшего столетия.....	48
Кризис технологии потребления информации.....	50
Умные интерфейсы.....	53
Сотворение мира.....	54

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ**

Windows Millennium — DOS возвращается!.....	58
Загрузочные диски — в поисках компромисса.....	58
Распишитесь, пожалуйста!.....	61
"Кофейный" язык.....	65





Виктор Куц

## Pentium 4 и чипсеты для него

**К**ак быстро идет время... Вот уже и платформа Pentium III для настольных компьютеров, еще вчера казавшихся для многих пределом мечтаний, семимильными шагами уходит в прошлое, несмотря на агонию вроде поддержки перспективной памяти типа DDR SDRAM в последних чипсетах. Впрочем, эта поддержка мало чем может помочь исчерпывающей себя технологии. Выпущенный в конце прошлого года процессор Pentium 4 позиционируется Intel как основа для нового поколения высокопроизводительных настольных ПК на ближайшие несколько лет. Но для успешной работы любой компьютерной системы необходимы соответствующие чипсеты — специализированные наборы контроллеров, служащих для связи процессора с другими устройствами. Вот о них-то мы сегодня и поговорим.

### Немного о процессоре Pentium 4

Процессор Pentium 4 (фирменное обозначение ядра — Willamette) производится по технологическому процессу 0,18 мкм и содержит 42 млн транзисторов, занимающих площадь 217 мм<sup>2</sup>. Размеры кристалла, вдвое превышающие размеры бли-

жайших аналогов, обусловлены тем, что Pentium 4 создавался в первую очередь с большим заделом на будущее, в том числе и на применение в процессе производства более тонкой технологии. Напряжение питания ядра процессора 1,7 В, конструктивное исполнение на сегодня — только Socket 423, хотя еще до конца года планируется начать переход на новый процессорный корпус mPGA 478.

Pentium 4 имеет новую микроархитектуру, получившую специальное название NetBurst. В числе наиболее значимых новшеств, появившихся в NetBurst, — 400-мегагерцовая процессорная шина, технологии Hyper-Pipelined Technology, Rapid Execution Engine и новый набор команд SSE2.

**400-мегагерцовая системная шина**, иногда называемая Quad Pumped Bus, у Pentium 4 имеет тактовую частоту 100 МГц. Однако благодаря использованию технологии Quad Pumping по этой шине передается четыре блока данных за один такт, что дает эффективную рабочую частоту 400 МГц, а пропускную способность — 3,2 Гб/с. Потенциал у шины очень высок, и вскоре возможно появление модификации процессора Pentium 4, поддерживающего системную шину 533 МГц (133 x 4, что дает пропускную способность 4,26 Гб/с!).

**Hyper-Pipelined Technology.** У Pentium 4 очень длинный конвейер, состоящий из 20 стадий, тогда как конвейер у процессоров семейства P5 всего 5 стадий, семейства P6 — 10, а AMD Athlon/Duron — 12 стадий. Более длинный конвейер, с одной стороны, позволяет достичь высоких тактовых частот даже при существующем уровне технологии, но, с другой стороны, при обнаружении неправильно предсказанного перехода весь конвейер останавливается, и его перезагрузка требует значительного времени. Поэтому преимущества технологии Hyper-Pipelined сегодня не столь очевидны и в полной мере проявятся только при значительном увеличении тактовой частоты процессора. Кроме того, в Pentium 4 интегрирован более совершенный механизм предсказания переходов, и количество ошибочно предсказанных переходов у него в среднем на 33% меньше, чем у процессоров семейства P6.

**Rapid Execution Engine** — блок выполнения арифметико-логических операций, причем состоит он из двух ALU-модулей, работающих параллельно, да еще и на удвоенной тактовой частоте по отношению к частоте процессора. Таким образом, весь Rapid Execution Engine способен выполнять до четырех це-

лочисленных операций за один рабочий такт процессора.

**Streaming SIMD Extensions 2 (SSE2)** содержит 144 новых SIMD-инструкций, предназначенных для увеличения производительности системы при обработке аудио- и видеоданных, которые добавлены к базовому набору SSE-инструкций, реализованному ранее в процессоре Pentium III. Из них 68 расширяют возможности старых SIMD-инструкций по работе с целыми числами, а остальные 76 являются совершенно новыми, позволяющими оперировать со 128-разрядными числами (как целыми, так и вещественными с двойной точностью).

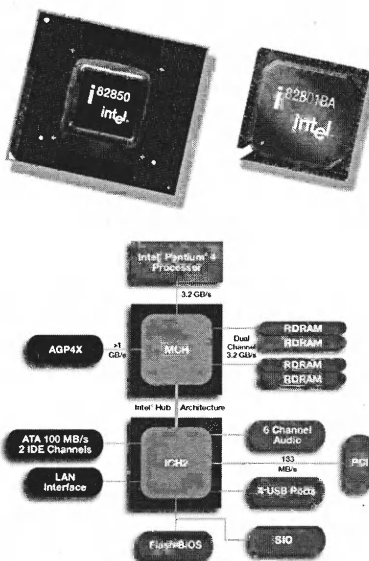
### Чипсет Intel i850

Первым чипсетом, поддерживающим Pentium 4, стал Intel 850, работающий только с относительно дорогой памятью фирмы Rambus. Он создан на базе серверного чипсета i840 и состоит из трех основных микросхем (хабов): Intel 82850 — Memory Controller Hub (MCH), Intel 82801BA — I/O Controller Hub (ICH2) и Intel 82802 — Firmware Hub (FWH).

Аналогом северного моста, использующегося во всех классических чипсетах, является хаб MCH. В его составе имеются контроллер системной шины (Quad Pumping Bus, тактовая частота 100 МГц), двухканальный контроллер памяти Direct Rambus (поддержка модулей памяти PC600/800 RDRAM, максимальная пропускная способность 3,2 Гб/с, максимальный поддерживаемый объем памяти — 2 Гб, модули устанавливаются в слоты попарно), контроллер шины AGP (поддерживает режимы работы AGP 4x и Fast Write). Для связи между MCH и ICH2 используется шина HI8 (Hub Interface) с пропускной способностью 266 Мб/с. Микросхема MCH выполнена в корпусе типа OLGA с 615 выводами и устанавливается в разъем, напоминающий Socket 7 для процессоров Pentium.

Южный мост ICH2 (Intel 82801BA) является универсальным и используется в составе всех современных чипсетов Intel, начиная с i810. В ICH2

встроены контроллер PCI-шины (поддерживает до шести PCI Bus Master-устройств), двухканальный IDE-контроллер (поддерживает до четырех устройств ATA/100), двухканальный USB-контроллер (4 USB-порта), 6-канальный звуковой контроллер AC97 (с цифровыми выходами типа SP/DIF) и интерфейс локальной сети (LCI), обеспечивающий поддержку 10/100 Мб/с Ethernet и Home PNA. Поддерживается коммуникационный интерфейс CNR (Communications Network Riser). FWH предназначен в первую очередь для размещения системного BIOS, содержит также генератор случайных чисел (Random Number Generator, RNG), который можно использовать для построения систем защиты или для других системных нужд. Хотя чипсет i850 и не поддерживает двухпроцессорные конфигурации, что значительно ограничивает применение систем на его основе в серверах, имеется возможность использования в его составе дополнительного хаба P64H для подключения 64-разрядной 66-мегагерцовой шины PCI.



Внешний вид и блок-схема чипсета i850

Один из самых значительных недостатков, помимо высокой стоимости самого i850, — сложность разводки соединений элементов чипсета на плате. Это вынуждает разработчиков использовать дорогой 6-слойный дизайн системных плат, что

еще больше увеличивает их стоимость. Правда, в последнее время стали говорить о появлении отдельных системных плат на базе чипсета i850, имеющих более дешевый 4-слойный дизайн.

### Чипсет Intel i845A

В самом начале июля на выставке Computex Taipei 2001 был анонсирован новый чипсет Intel, положивший конец гегемонии памяти Rambus на платформе Pentium 4. Главная отличительная особенность чипсета i845, разработавшегося под кодовым наименованием Brookdale, — поддержка самого дешевого на сегодня типа памяти PC133 SDRAM, что, по мнению руководства Intel, должно помочь процессору Pentium 4 завоевать рынок недорогих компьютерных систем. Кроме того, i845 поддерживает процессоры Pentium 4 только в новом малогабаритном корпусе mPGA 478 (в отличие от выпускаемых в настоящее время PGA 423). MCH нового чипсета аналогичен северному мосту i850, за исключением, естественно, контроллера памяти — он позволяет работать с 3 Гб PC133 SDRAM. Однако в Сети постоянно циркулируют слухи о встроенной в него поддержке DDR, которая просто временно отключена. В качестве южного моста i845 до середины 2002 года будет использоваться микросхема ICH2, поддерживающая протокол IDE ATA/100.

Честно говоря, чипсет i845A своим появлением вызывает больше вопросов, чем дает ответов. Например, i845A, изначально планировавшийся как дешевое решение, требует дорогого 6-слойного дизайна системной платы. Плюс к этому уже сейчас отпускная цена чипсета значительно выше, чем у его основного конкурента VIA P4X266, который, к тому же, поддерживая быструю память DDR, имеет гораздо более высокую производительность (согласно первым тестам). Очень жаль, что разработчики чипсета для расширения изначально узкого места новой системы не сочли нужным использовать хотя бы два канала шины памя-

ти: похоже, медленная память не даст возможности процессору Pentium 4 проявить все достоинства новых технологий.

### Перспективные чипсеты Intel

В перспективных планах Intel на ближайший год значатся i845D, представляющий собой тот же i845A, но с включенной поддержкой памяти DDR SDRAM. К большому разочарованию многих, будет поддерживаться только самая медленная ее разновидность, PC1600, уже сейчас считающаяся устаревшей. Видимо, принудительное ограничение производительности было сделано с тем расчетом, чтобы не создавать конкуренцию горячо любимой Intel памяти RDRAM. Кстати, для любителей еще более медленных решений сохраняется обратная поддержка обычной SDRAM. i845D должен появиться в начале 2002 года, хотя не исключено, что сложная ситуация на компьютерном рынке заставляет Intel ускорить его выход.

Чуть позднее, во втором полугодии следующего года, должны появиться еще два чипсета, пока известные под именами Tullloch и Brookdale-G. Оба они рассчитаны на новую процессорную шину Pentium 4 с тактовой частотой 133 МГц, а значит, с учетом технологии Quad Pumping, общая частота шины составит 533 МГц. Для них будет разработан новый южный мост ICH4, поддерживающий три канала высокоскоростной шины USB 2.0 (всего 6 портов, максимальная скорость обмена 480 Мб/с) и новый последовательный интерфейс Serial ATA для связи с жесткими дисками. Различаться новые чипсеты будут типом поддерживаемой памяти. Tullloch будет работать с упрощенной, а значит, и более дешевой 4-банковой разновидностью RDRAM PC1066 4i (сейчас — 16 банков). Неизбежное при уменьшении количества банков памяти снижение производительности планируется компенсировать увеличением частоты ядра памяти до 133 МГц, что позволит сохранить пропускную способность памяти на уровне 2 Гб/с на

канал. А Brookdale-G наконец-то обретет нормальную поддержку DDR SDRAM, как PC1600, так и PC2100, плюс к этому PC133 SDRAM.

### Чипсеты VIA

Из всех конкурентов интеловских чипсетов раньше всего появится на свет VIA P4X266, который, впрочем, уже сейчас готов к массовому производству, несмотря на явно негативное отношение к этому Intel, которая пока еще, из боязни потерять значительную долю рынка для своего i845, не лицензировала VIA системную шину Pentium 4. Тем не менее, VIA громко заявляет о своем праве производить P4X266 благодаря наличию кросс-лицензионного соглашения между Intel и S3, которая была в свое время куплена VIA.

Согласно перспективным планам VIA, P4X266 станет ее основным продуктом для сегмента высокопроизводительных компьютерных систем, причем компания собирается сразу установить цены на чипсет P4X266 на 25—30% ниже, чем цены Intel на i845A.

Чипсет P4X266, выполненный по хабовой архитектуре, содержит северный мост на микросхеме VT8753, который, в отличие от своего главного конкурента i845A, поддерживает двухпроцессорную конфигурацию, что наряду с использованием отдельного чипа-компаньона VPX-64 для организации 64-разрядной шины PCI может сыграть существенную роль при продвижении его на серверный рынок. В традициях VIA контроллер памяти обеспечивает поддержку широкой номенклатуры типов памяти: помимо основной PC1600/2100 DDR SDRAM возможно использование P100/133 SDRAM и VCM SDRAM. Для связи с южным мостом VT8233 используется двойная шина V-Link с общей пропускной способностью 533 Мб/с. VT8233 содержит контроллер шины PCI, двухканальный IDE-контроллер (ATA/100), три канала USB (6 портов), 6-канальный звуковой контроллер AC97 (с цифровыми выходами типа SP/DIF) и LAN-контроллер 10/100 Мб/с Ethernet и Home PNA. Наряду с

коммуникационным интерфейсом CNR поддерживается и ACR (Advanced Communications Riser).

В дальнейших планах VIA — выпуск и P4M266 с интегрированным графическим ядром Savage 4, который, в отличие от своего базового варианта, будет поддерживать только память DDR SDRAM.

В первой половине будущего года VIA планирует выпустить чипсет P4X333, поддерживающий PC2700 DDR SDRAM. В новом наборе логики помимо поддержки нового типа памяти VIA собирается ввести поддержку графической шины AGP 8x. Вместе с P4X333 должен появиться и новый южный мост VT8235, который будет соединяться с северными мостами по ускоренной шине V-Link с пропускной способностью 533 Мб/с. Новый южный мост будет поддерживать интерфейсы USB 2.0 и IEEE1394. Свято блюдя традицию, вслед за обычным чипсетом VIA готовит его интегрированный вариант, который обретет новое графическое ядро UltraSavage, содержащее аппаратный блок T&L и увеличенное число конвейеров по сравнению с Savage 4.

### Чипсеты SiS

Компания SiS для платформы Pentium 4 предполагает до конца этого года выпустить чипсет SiS645 с поддержкой памяти новой спецификации DDR2700. В этом чипсете фирма вернулась к двухчиповой конфигурации, причем связь между северным (SiS645) и южным (SiS961) мостами осуществляется по 16-битной 266 МГц шине собственной разработки MuTIOIOL с пропускной способностью до 533 Мб/с.

Интегрированный контроллер памяти нового чипсета обеспечивает поддержку до 3 Гб памяти PC2700 (работающей с тактовой частотой 166 МГц)/2100 DDR SDRAM или обычной PC133 SDRAM.

В первом квартале 2002 года планируется выпустить вариант SiS645 — SiS650 с интегрированным графическим ядром SiS330, полностью совместимым с DirectX 8.0. Южный мост SiS961 по своим

возможностям практически не отличается от VIA VT8233. Поддержка USB 2.0 должна появиться только в 2002 году в новом чипе SiS962.

### Чипсеты Ali

К четвертому кварталу 2001 года компания ALi, для которой, как и для SiS, проблем с лицензированием технологий Intel никогда не было, намерена выпустить чипсет M1671. Он практически полностью, за исключением процессорной шины, повторяет не сыскавшую особого признания модель Aladdin Pro 5 для Pentium III (AGP 4x, поддержка DDR SDRAM и PC100/133 SDRAM, шина PCI). Классический южный мост ALi M1535D+ в качестве отличительной

особенности имеет полнофункциональный звуковой контроллер.

В следующем чипсете M1681, запланированном на 2002 год, закладывается поддержка шины AGP 8x, а для связи с южным мостом M1563 предполагается использовать новую высокоскоростную магистраль High Speed Link Bus. От своих собратьев, выпускаемых другими фирмами, M1563 будет отличаться поддержкой USB 2.0 и интерфейса флэш-памяти Memory Stick.

Таким образом, самая большая интрига последнего времени, похоже, подходит к своему финалу: в перспективных планах основных производителей чипсетов четко просматривается тенденция к поддержке памяти типа DDR. Для памяти PC133

SDRAM остается небольшая ниша самых дешевых систем начального уровня, а вот у памяти RDRAM перспективы совсем туманные, ведь ее в следующем году будет поддерживать только один интеловский чипсет Tuloch, а i850, похоже, сойдет со сцены окончательно. Что касается перспектив Tuloch, то не мешает вспомнить, что отрицательного опыта создания чипсетов под RDRAM у Intel гораздо больше, чем положительного. А ведь в лагере сторонников DDR скоро должно произойти пополнение: переговоры между Intel и NVIDIA по поводу лицензирования последней системной шины процессоров Pentium 4 ведутся уже давно, и рано или поздно новый чипсет этой компании займет свое место.

## О чем писала компьютерная пресса...

### ...20 лет назад:

1981, сентябрь

- Microsoft начала работу над графическим интерфейсом пользователя для MS-DOS, который первоначально был назван Interface Manager.

- Компания Osborne Computer достигла уровня продаж 1 млн долларов в месяц.

- Apple Computer представила свой первый жесткий диск ProFile объемом 5 Мб и ценой \$3500.

- IBM начала продажи IBM PC с опережением графика.

- Компания Acorn Computer Ltd представила компьютер BBC с процессором 6502A, RAM до 48 Кб, 73-клавишной клавиатурой и 16-цветной графикой.

### ...15 лет назад:

1986, сентябрь

- Малоизвестная компания Advanced Logic Research объявила о создании первого PC-компьютера Access 386 на базе процессора Intel 386.

- Compaq Computer представила свой первый PC-компьютер Compaq Deskpro 386 на базе процессора Intel 80386 частотой 16 МГц.

- Apple Computer представила компьютер Apple IIGS на базе процессора Western Digital W65C816

(65816) с частотой 1 или 2,8 МГц. Компьютер был впервые оснащен 3,5-дюймовым дисководом собственной разработки Apple. Цена \$1000.

- IBM объявила о выпуске компьютера IBM PC-XT 286, оснащенного RAM объемом 640 Кб, дисководом для дискет на 1,2 Мб, жестким диском объемом 20 Мб, последовательным и параллельным портами. Цена \$4000.

- Стив Джобс принял решение использовать в первом компьютере NeXT привод для работы с перезаписываемым оптическим диском.

### ...10 лет назад:

1991, сентябрь

- Intel представила микропроцессор i486SX с частотой 16 МГц и производительностью 13 MIPS. В том же месяце появилась версия этого процессора с частотой 25 МГц и производительностью 20 MIPS, затем процессор 80386SL с частотой 25 МГц и производительностью 5,3 MIPS.

- Компания MIPS Technologies начала поставки процессоров R4000.

- Digital Research выпустила первую операционную систему для PC-компьютеров DR-DOS 6.0 по цене \$100.

- Выпущена карта PCMCIA v2.0, ориентированная на работу с модемами, сетевыми картами, внешними накопителями и иным периферийным оборудованием.

- Компания Chips & Technologies выпустила интегрированный процессор F8680 для ноутбуков и карманных компьютеров, совместимый с Intel 8086 и 80186. Чип содержал универсальный асинхронный приемопередатчик, контроллер CGA-дисплея, поддерживал карту PCMCIA. Цена \$45.

### ...5 лет назад

1996, сентябрь

- Hewlett-Packard объявила о разработке компьютеров Vectra XM и Vectra VL на базе процессоров Pentium.

- Motorola сообщила о выпуске клонов компьютеров Power Macintosh — StarMax (Mac-совместимый) и PowerStack II (Windows NT) на базе процессоров PowerPC 603e и 604e с частотой 166 и 200 МГц.

- Microsoft объявила о том, что ведет разработку операционной системы Windows CE для карманных компьютеров. Кодовое наименование проекта — Pegasus. Аббревиатура "CE" расшифровывается как Consumer Electronics (бытовая электроника).

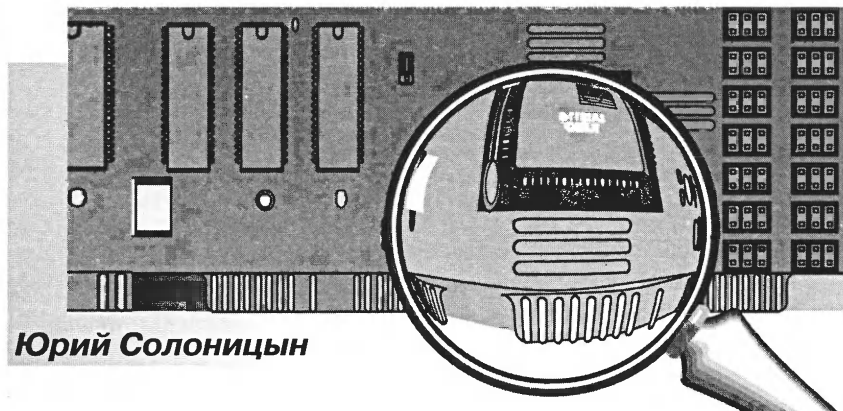
**Ч**асто в компьютерных магазинах можно услышать фразу вроде "Возьмите лучше этот сканер, он не на Centronics, а на USB подключается". Человека, плохо знающего компьютер изнутри, такое выражение может поставить в тупик. В самом деле, что такое Centronics и что такое USB, понять еще можно, но чем один лучше или хуже другого? Эта статья поможет вам избавиться от таких сомнений, раз и навсегда уяснить, какие бывают интерфейсы периферийных устройств и зачем они нужны.

Периферийные устройства — принтер, сканер, цифровая камера или, например, MP3-плеер — существенно различаются по конструкции, принципам работы и назначению, но объединяет их то, что они используются совместно с компьютером. Интерфейс периферийных устройств — это и есть то, что соединяет их с ПК. Термин "аппаратный интерфейс" можно толковать двояко. В широком смысле это система, позволяющая соединить два или несколько устройств между собой и передавать информацию, а в узком — разъемы для подключения кабелей, соединяющих эти устройства.

По способу передачи данных, представленных группами битов (например, байтами) аппаратные интерфейсы делятся на две большие категории: последовательные и параллельные.

В последовательных интерфейсах биты данных передаются один за другим. Обычно такой способ передачи используется там, где возможно или выгодно использовать только одну линию связи. При этом каждая группа битов сопровождается особыми признаками начала и конца передачи. Это нужно для отделения их друг от друга при приеме. Синхронизация передачи данных отсутствует, поэтому данный тип интерфейса называют также асинхронным.

Последовательная передача битов одной группы значительно снижает общую скорость передачи данных. Разного рода меры по повышению скорости обмена делают такие системы более дорогими. Поэтому в



Юрий Солоницын

## Ворота для ваших гигабайтов

большинстве случаев интерфейсы этого типа используются для низкоскоростных систем. Лишь в последнее время стали распространяться относительно дешевые конструкции последовательных интерфейсов (USB, FireWire) с высокими скоростями передачи данных. Недостаток этих систем — ограниченная длина линий передачи (при увеличении длины проводов возрастает уровень помех, начинают сильнее сказываться частотные свойства кабеля).

В параллельных интерфейсах все биты группы одновременно передаются по параллельным линиям. Разумеется, необходимость прокладки нескольких линий усложняет конструкцию системы и увеличивает ее стоимость, особенно при большой длине линий. Однако такой способ передачи дает существенно большую скорость, чем последовательная схема.

Если для надежной одновременной передачи данных по всем линиям используется синхронизация, интерфейс называют синхронным.

### Интерфейс RS-232C

Интерфейс стандарта RS-232C более известен пользователям под названиями "последовательный порт" или "COM-порт". Назначение этого интерфейса — последова-

тельная передача данных при связи компьютера с с внешними устройствами, такими как модемы. Первые модификации интерфейса допускали передачу информации со скоростями менее 5 Кб в секунду.

Физически интерфейс RS-232C, устанавливаемый на материнской плате компьютера или плате-адаптере, состоит из кабельного разъема, микросхем драйвера линии и микросхемы UART (Universal Asynchronous Receiver/Transmitter, универсальный асинхронный приемопередатчик).

Кабельный разъем служит для присоединения кабеля, соединяющего ПК с каким-либо устройством. Используются разъемы с 9 или 25 контактами. В 25-контактном разъеме часто задействованы только 9 линий, остальные применяются только в расширенных версиях порта. Прием и передачу сигналов в линию, их преобразование в форму, принятую внутри компьютера, и интерпретацию выполняют три микросхемы: два драйвера линии и UART.

Драйверы преобразуют сигналы от логических уровней TTL, принятых в компьютере (логической 1 соответствует напряжение выше 2,4 В, логическому 0 — меньше 0,7 В) в форму, используемую в канале связи (логической 1 соответствуют напряжения от -3 до -25 В, а логическому 0 — от +3 до +25 В). Большой раз-

брос напряжений позволяет повысить устойчивость к помехам малой амплитуды.

Данные по последовательному интерфейсу передаются в следующем виде: первым передается так называемый старт-бит (признак начала передачи слова данных), затем биты слова данных, а за ними следуют бит четности (служит для контроля правильности передачи) и стоп-бит, обозначающий конец передачи данных.

Микросхема UART преобразует параллельный код в последовательный, и наоборот. Она же управляет работой интерфейса. Эта микросхема может содержать буферную память для повышения скорости и надежности передачи данных. В настоящее время драйверы линии и схема UART выполняются на одном кристалле, содержащем все или большинство необходимых для работы ПК устройств (контроллеры, шинные формователи и т. д.).

Работу с последовательным портом обеспечивают стандартные функции, встроенные в BIOS. Для доступа к портам выделено специальное пространство памяти. BIOS позволяет установить до четырех адаптеров последовательных портов. Наличие подключенных адаптеров определяется во время самодиагностики ПК при включении питания.

В настоящее время интерфейс RS-232C постепенно вытесняется такими, как USB и FireWire, обеспечивающими большую скорость передачи.

### Интерфейс Centronics

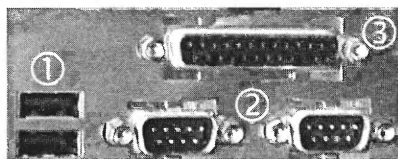
Большинство компьютеров оснащается адаптерами параллельного интерфейса, совместимыми со стандартом Centronics (назван по имени компании, разработавшей этот интерфейс для работы с принтерами) или его расширениями.

Разъем интерфейса Centronics имеет 25 контактов, из которых 8 предназначены для одновременной передачи восьми битов одного байта, а 9 обслуживают сигнальные линии, служащие для управления принтером и приема данных о его состо-

янии. Данные на информационных линиях считаются действительными только при наличии сигнала синхронизации.

В первоначальных версиях стандарта Centronics предусматривалась исключительно однонаправленная передача данных (от ПК к принтеру). Данное ограничение можно было обойти, используя для передачи данных в компьютер линии контроля. Байт при этом передавался в два "захода" по 4 бита.

В более поздних модификациях этого интерфейса (стандарт IEEE 1284) было введено несколько новых режимов, позволяющих ускорить передачу данных и организовать двунаправленную связь.



Задняя панель корпуса ATX со стандартным набором разъемов: 1 — два разъема USB; 2 — два классических 9-контактных разъема последовательного интерфейса; 3 — 25-контактный разъем параллельного интерфейса Centronics.

Режимы работы интерфейса стандарта IEEE 1284:

*Centronics* — режим, обеспечивающий совместимость со стандартным принтерным портом.

*Nibble* и *Byte* — режимы для передачи данных в компьютер. В режиме *Nibble* передача данных производится по 4 бита (за 2 такта), в режиме *Byte* — за один такт.

*EPP (Enhanced Parallel Port)* — режим повышенного быстродействия за счет обоюдного подтверждения приема. К интерфейсу, реализующему этот режим, предъявляются более строгие требования, что позволяет увеличить длину линии передачи (при использовании специального кабеля). Возможна двунаправленная передача данных.

*ECP (extended capabilities port)* — мало отличается от режима EPP, однако порт, работающий в этом режиме, обладает возможностью аппаратного сжатия данных по алгоритму RLE (Run-Length Encoding, кодирова-

ние повторов), что существенно увеличивает скорость передачи.

То, что современные параллельные интерфейсы способны производить не только вывод, но и ввод информации, позволяет использовать их для подключения различных устройств ввода данных, например, сканеров или внешних накопителей.

Режим работы адаптера параллельного интерфейса, как правило, может быть определен с помощью программы настройки BIOS. При включении компьютера BIOS производит поиск и тестирование установленных адаптеров параллельного интерфейса.

### Высокоскоростная шина USB

В 1995 году группа компаний в составе Compaq, DEC, IBM, Intel, NEC, Northern Telecom и Microsoft разработала спецификацию интерфейса USB (Universal Serial Bus, универсальная последовательная шина). Этот интерфейс имеет ряд особенностей:

- Горячее подключение устройств.
- Автоматическое распознавание и настройка подключаемого оборудования (Plug & Play).
- Легкое ветвление линий передачи при помощи разветвителей, которыми могут служить сами подключаемые устройства.
- Питание маломощных устройств (до 2,5 Вт) непосредственно от линии шины.

Все эти качества в сочетании со скоростью передачи до 12 Мбит/с (спецификация USB 1.1) или 240 Мбит/с (спецификация USB 2.0) значительно упрощают конструкцию и эксплуатацию различного периферийного оборудования.

При помощи разветвителей к одному контроллеру USB может быть подключено до 128 устройств. Сами устройства USB часто снабжают дополнительными разветвителями на 2 или 4 линии.

Кабель USB содержит четыре линии: две образуют витую пару для передачи данных, одна служит для подачи питания (+5 В) устройствам,



не имеющим собственных источников энергии, и один провод общий.

### Высокоскоростная шина FireWire

Интерфейс FireWire был разработан фирмой Apple в 1986 году для унификации интерфейсов различных периферийных устройств и высокоскоростного обмена данными. Однако стандартизирован он был только в 1995 году, стандарт получил наименование IEEE 1394.

Линии связи на основе FireWire первых спецификаций имеют пропускную способность до 100 Мбит/с, более поздние версии — до 400 Мбит/с. Это позволяет использовать их для передачи видеоданных в реальном времени, что с успехом используется в системах цифрового видеомонтажа.

Как и шина USB, FireWire обеспечивает горячее подключение устройств и подачу питания маломощным устройствам непосредственно через линию связи. Для передачи данных используются две витые пары (прямого и обратного обмена информацией), что повышает помехозащищенность линии. К одному контроллеру FireWire можно подключить только 63 устройства, но с помощью концентраторов и мостов можно строить сложные системы, содержащие до 1000 устройств.

Для повышения скорости пере-

дачи данных до 1 Гбит/с применяют оптоволоконные линии связи.

### Инфракрасный порт SIR

Последовательный интерфейс SIR (Serial InfraRed port) был разработан ассоциацией IrDA. Первые версии этого стандарта были опубликованы в 1994 году. Для передачи данных в этом интерфейсе используются инфракрасный излучатель (светодиод) и фотоприемник.

Сравнение пропускной способности портов

Порт	Пропускная способность, Мбит/с
LPT (Centronics)	1,2
LPT (ECP/EPP)	24
SCSI (8 бит)	40
Fast SCSI (16 бит)	80
Ultra2 Wide SCSI (16 бит)	640
Ultra3 SCSI (16 бит)	1280
USB 1.1	12
USB 2.0	120 – 240
FireWire	100 – 400

По своим свойствам интерфейс SIR близок к интерфейсу стандарта RS-232C. Скорость передачи по инфракрасному каналу составляет до 115 Кбит/с (в последних спецификациях — до 4 Мбит/с). Обмен данными — последовательный. Для устранения ошибок используются алгоритмы проверки контрольной суммы пакетов данных (CRC, cyclical redundancy check).

Существенный недостаток инфракрасного порта — ограниченный

радиус действия (скорость передачи 4 Мбит/с достигается на расстоянии около 1 м). Кроме того, между приемником и передатчиком не должно быть посторонних предметов. Поскольку инфракрасный порт не требует дополнительных кабелей и разъемов, он часто используется в мобильных устройствах.

### SCSI-интерфейс

Интерфейс SCSI (Small Computer System Interface, интерфейс малых вычислительных систем) позволяет объединять до 7, а в последствии модификациях — до 15 устройств. Обеспечивает высокоскоростной обмен данными между устройствами и центральным процессором, а также между самими устройствами (без дополнительной загрузки процессора и памяти). Первые модификации интерфейса SCSI были разработаны в 1986 году.

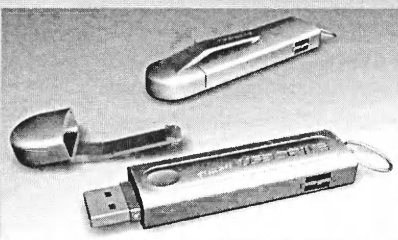
Устройства соединяются между собой при помощи экранированного кабеля, состоящего из витых пар проводов. Максимальная длина линии связи — 25 метров. Работа шины обеспечивается специальным контроллером, устанавливаемым на слот расширения компьютера. Интерфейс SCSI используется для подключения устройств, требующих высокоскоростного доступа или оперирующих большими объемами данных (например, дисковых накопителей или сканеров).

## Переносной накопитель в форме авторучки...

...Может стать заветной мечтой любого пользователя, деятельность которого связана с частой транспортировкой данных большого объема. И не только, поскольку компактное хранилище данных, уместяющееся в нагрудном кармане, нужно практически всем.

Разработчики JMtek Inc. порадовали пользователей персоналок выпуском ряда накопителей Flash USBDrive размером с авторучку. Емкость модулей памяти на базе стандарта flash — 16, 32, 64, 128, 256 и 512 Мб при гарантированном сроке сохранения

данных не менее 10 лет. Flash USBDrive можно подключить к настольному ПК, ноутбуку или карманному компьютеру.



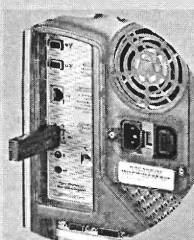
Накопитель подключается к компьютеру через интерфейс USB-1 и распознается операционной средой (Windows 98/ME/2000) как дополни-

тельный винчестер, обращение к которому производится автоматически. Максимальные скорости обмена данными — 800 Кб/с (чтение) и 500 Кб/с (запись).

Для защиты данных в накопитель встроены аппаратный ключ. Пользователь может применить любой имеющийся в его распоряжении криптопакет.

Трехцветовой светодиод информирует пользователя о режимах записи данных, чтения или idle-mode, когда он готов к "горячему отключению".

*Алексей Смирнов*





Николай Богданов-Катков

# Энергетика системного блока

**П**роблема охлаждения компьютера последнее время стала очень популярна в печати. В ведущих рекламных изданиях даже появилась отдельная рубрика "Системы охлаждения". Да и в сервисных центрах в недавний период летней жары инженеры постоянно сталкивались с проблемами, вызванными перегревом. Вот и мы решили еще раз вернуться к этой теме.

В "Магии ПК", №4/2001 подробно рассказывалось о принципах работы разнообразных систем охлаждения, применяемых в современной технике. Рассмотрим подробнее некоторые устройства.

## Что и как греется

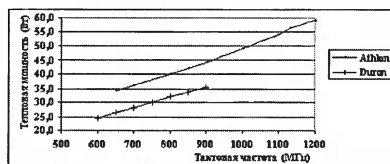
Все электрические и электронные устройства нагреваются проходящим током. Механические узлы, содержащие движущиеся части, также нагреваются. То и другое относится к внутреннему нагреву — тепло выделяется внутри самого устройства. Для многих деталей, например, конденсаторов, внутренний нагрев невелик, но они могут нагреваться внешним теплом — при повышении температуры внутри системного блока.

На первом месте, разумеется, процессор. На рисунках приведены значения выделяющейся тепловой

мощности для разных процессоров с разными тактовыми частотами.

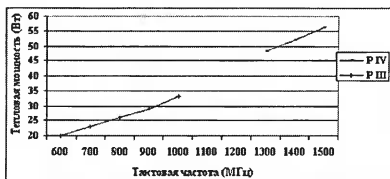
Следует иметь в виду, что для процессоров установлены разные максимально допустимые температуры ядра:

- Athlon 1100, 1133, 1200 МГц — 95°C,
- Все остальные Athlon и Duron — 90°C,
- Pentium III 1000 МГц — 70°C,
- Остальные Pentium III и Pentium IV — 80°C.



Тепловые характеристики процессоров AMD (максимальная температура ядра 90°C).

Может показаться, что процессоры AMD значительно "горячее". Действительно, Athlon греется сильнее, чем Pentium с той же частотой. Но если учесть, что Athlon'ы значительно шустрее, то окажется, что



Тепловые характеристики процессоров Intel (максимальная температура ядра 80°C).

сопоставимые по производительности процессоры выделяют практически одинаковую мощность. Другое дело, что для Athlon'a допустима более высокая температура ядра. Не снизит ли это надежность? Пока массовых отказов процессоров AMD не наблюдалось, а вот с процессорами Pentium III бывали проблемы. Фирме Intel пришлось даже отзывать целую партию 933-мегагерцовых...

Винчестеры нагреваются при работе меньше чем процессоры, но все же очень заметно. Известно, что "тихие" и "холодные" винчестеры надежнее "шумных" и "горячих". Выделяемая тепловая мощность зависит от многих параметров — от скорости вращения и количества магнитных дисков (трение дисков о воздух), от объема буфера (выделение джоулевого тепла) и т. п. Но нагрев совершенно не зависит от главного потребительского качества — форматируемой емкости винчестера. Поэтому все фирмы стараются максимально увеличить плотность записи и обойтись одним диском, тогда как раньше ставили по три-четыре.

Анализ данных ряда фирм-производителей по различным сериям выпускаемых винчестеров позволяет сделать следующие выводы:

1. Обычно винчестер со скоростью вращения шпинделя 5400 об./

мин и одним магнитным диском выделяет 4,0—4,5 Вт.

2. При увеличении скорости вращения до 7200 об./мин мощность возрастает в 1,8—2,0 раза, а при увеличении до 10000 об./мин — в 2,5—3 раза.

3. Каждый следующий диск добавляет по 0,7—1,2 Вт в зависимости от скорости вращения.

4. При увеличении объема буфера с 512 Кб до 2 Мб мощность возрастает на 0,7—1,0 Вт.

Эти величины применимы к винчестерам всех фирм: никакие конструкционные ухищрения не позволят обойти законы физики. Однако здесь важна не только и не столько мощность, сколько температура, до которой может нагреться винчестер.

Всего может получиться ватт 12—18. Это кажется незначительным, но, во-первых, максимально допустимая температура винчестеров обычно не выше 55—60°C, то есть ниже, чем для процессоров. Кроме того, винчестер, в отличие от процессора, не снабжен специальным вентилятором и крепится в таком месте, где скорость воздуха при обычном расположении вентилятора минимальна. Поэтому тем, кто польстился на современный скоростной винчестер, следует позаботиться о его охлаждении.

Известно, что одни винчестеры на ощупь греются сильнее, другие слабее. Причина в том, что некоторые производители делают корпуса из более теплопроводных материалов именно для улучшения теплоотвода.

### Железо и воздух

Какие кулеры лучше использовать для охлаждения самых "горячих" процессоров AMD, писали уже не раз. Достаточно сказать, что обычный радиатор с вентилятором может "раскидать" 35—40 Вт, если процессор допускает нагрев до 90°C, и около 30 Вт при допустимой температуре 80°C. Следовательно, для Athlon, начиная с 800 МГц, необходимы более эффективные охладители — Chrom Orb или Golden Orb.

В мае в продаже появился первый охладитель для процессора, ос-

нованный на эффекте Пельтье. Пластина толщиной в несколько миллиметров вставляется между процессором и радиатором и обеспечивает разницу температур между ними 40°C.

Но кроме процессора надо охлаждать еще видеокарту и винчестер. К тому же никакое охлаждение не будет эффективным, если температура воздуха внутри корпуса поднимется до 50—60°C. А если воздух из компьютера будет выкачивать только вентилятор, установленный на блоке питания, то так и выйдет.

Дело даже не в том, что мощность вентилятора мала. Как бы он ни трудился, скорость воздухообмена будет ограничиваться скоростью поступления воздуха в корпус через имеющиеся отверстия, а их никогда не бывает много. Для эффективного охлаждения нужно как минимум два вентилятора — нагнетающий и вытяжной.

В современных корпусах есть возможность установить один или два вентилятора в передней стенке и один в задней. Если вентиляторы в

передней стенке будут нагнетающими, а вентилятор блока питания и дополнительный — вытягивающими, то ток воздуха от передней стенки к задней будет равномерно обдувать все платы, которые расположены большей частью вдоль корпуса и содержат не так много выступающих деталей.

Производительность вентилятора характеризуется объемом воздуха, прокачиваемого в минуту.

$$V = 56 \times \frac{W}{\Delta t}, \text{ литров в минуту,}$$

где  $W$  — мощность, а  $\Delta t$  — разность температуры внутри корпуса и температуры окружающего воздуха. Последнюю величину обычно задают равной 10°C.

В иностранных технических документах производительность вентилятора часто указывают в кубических футах в минуту (CFM); один CFM равен 32 л/мин. Если мерить в кубических футах, коэффициент в формуле будет составлять 1,76.

Желательно, чтобы суммарная производительность всех нагнетающих вентиляторов не намного отличалась от производительности всех

### Звездные часы человечества — 2

## ДИТЯ ВОЙНЫ

*21 июня 1948 года англичане Том Килбурн и Фредди Уильямс в Манчестерском университете запустили первую программу на компьютере Baby. "Малыш" весил одну тонну, состоял из 600 радиоламп и имел память 1024 бита, набор команд состоял из 7 инструкций.*

В чем же значимость этого события для истории компьютеростроения?

Еще в 1942 году в США в Пенсильванском университете под руководством Джона Моучли и Джона Эккерта начались работы над проектом ЭВМ ENIAC (Electronic Numerical Integrator and Computer), которая предназначалась для решения задач баллистики. Весной 1945 года ЭВМ была построена. Она содержала 17468 электронных ламп, 7200 кристаллических диодов, 4100 магнитных элементов, занимала площадь 300 м<sup>2</sup> и весила 30 тонн, однако в 1000 раз превосходила по быстродействию релейные вычислительные машины: за 1 секунду она производила 360

операций умножения или 5000 сложений многозначных чисел.

Память ENIAC состояла из 20 аккумуляторов, соединенных между собой и с другими устройствами шинами данных и так называемыми "программными линиями" для синхронизации данных. Каждый аккумулятор мог хранить двоичное число: 10 бит на каждую цифру и один знаковый бит. Для хранения констант было предусмотрено 104 12-разрядных регистра, формирующих функциональную таблицу. Тактовая частота — 100 кГц. Однако ввод программы осуществлялся через специальную панель переключателей и занимал порой около недели (!).

Одновременно с постройкой

вытяжных вентиляторов. Только тогда система будет работать эффективно.

Одна из конструкций вентилятора для охлаждения винчестера — блок, вставляемый в трехдюймовый отсек. Поток воздуха обдувает винчестер только сверху.

Другой более совершенный охладитель для винчестера выполнен заодно с салазками и устанавливается в пятидюймовое гнездо, а винчестер — уже в него. Он имеет два небольших вентилятора и амортизаторы и обеспечивает равномерный обдув корпуса винчестера со всех сторон, одновременно защищая его от вибрации. Это очень удачная конструкция, а лучше всего то, что он может служить нагнетающим вентилятором для охлаждения всего системного блока.

Однако только его недостаточно. И он, и вентилятор блока питания расположены в верхней части корпуса. Основной поток воздуха будет идти выше процессора и видеокарты.

Еще один вентилятор выполнен в форме карты расширения. Он рабо-

тает на вытягивание, и если установить его в нижнее гнездо, сможет обеспечить наиболее равномерный ток воздуха по всему объему системного блока. Пожалуй, можно считать наилучшим решением следующее сочетание: охладитель винчестера и вентилятор на передней панели работают на нагнетание, а вентилятор блока питания и дополнительный на задней панели — на вытягивание. Получается дороговато, но продвинутый компьютер с мощными процессором и видеокартой, скоростным винчестером нуждается в более интенсивном охлаждении, чем то, которое способен обеспечить стандартный корпус.

Кстати, здесь понадобится и более мощный блок питания. Некоторые фирмы по требованию покупателя могут установить в любой корпус блок питания, предназначенный для Pentium IV (250 Вт). Это несколько выше мощности стандартных блоков питания для корпусов ATX Minitower (200—235 Вт). Вентилятор на нем стоит тоже значительно более мощный.

ENIAC, также в обстановке секретности, но с гораздо меньшим бюджетным финансированием, создавалась ЭВМ в Великобритании. В течение 1943 года в Лондоне была построена машина Colossus на 1500 электронных лампах для дешифровки кодов, которыми пользовались вооруженные силы Германии в период второй мировой войны. Хотя и ENIAC, и Colossus работали на электронных лампах, они по существу копировали электромеханические машины: новое содержание (электроника) было втиснуто в старую форму — структуру доэлектронных машин.

Концепцию хранимой в памяти компьютера программы впервые выдвинул Эккерт в 1944 году. В начале 1945 года под руководством Моучли и Эккерта был разработан проект первого компьютера EDVAC (Electronic Discrete Variable Computer) с вводимыми в память программами и числами.

Тогда же Джон фон Нейман, присоединившийся к группе разработчиков ENIAC, описал в докладе уст-

ройство будущего компьютера EDVAC и дал детальное определение концепции хранимой программы, правда, забыв при этом упомянуть имена своих коллег — Моучли и Эккерта. В результате концепция получила название "архитектура фон Неймана".

Концепция концепцией, однако сама американская ЭВМ была завершена лишь в 1950 году.

Между тем, первый универсальный компьютер с хранимой программой был создан в старушке Англии несколькими годами раньше. Создан тихо, можно сказать "по-английски". Лишь в сентябре 1948 года, чтобы "застолбить" приоритет, акт рождения Baby (к сожалению, не сохранившейся к настоящему времени) был упомянут в коротенькой заметке в чисто академическом издании "NATURE". В нескольких строчках этой заметки было сформулировано главное, что отличает Baby от всех предыдущих ЭВМ:

"Машина универсальна в том смысле, что может использоваться

### Что почем

Вот некоторые цены на охладители.

Обычный кулер для процессоров AMD — 100—140 р.

Кулеры Golden Orb, Chrom Orb — 300—450 р.

Дополнительный вентилятор для передней панели — 100—150 р.

Дополнительный вентилятор для видеокарты — 50—120 р.

Вентилятор для винчестера — 400—450 р.

Вентилятор для задней панели в форме карты расширения — 150—200 р.

Блок вентиляторов для пятидюймового слота — 400—600 р.



для решения любых задач, выраженных в виде программы с элементарными инструкциями, причем программу можно менять без каких-либо механических или электромеханических манипуляций".

#### Electronic Digital Computers

A small electronic digital computing machine has been operating successfully for some weeks in the Royal Society Computing Machine Laboratory, which is at present housed in the Electrical Engineering Department of the University of Manchester. The machine is purely experimental, and is on too small a scale to be of mathematical value. It was built primarily to test the soundness of the storage principle employed and to permit experience to be gained with this type of machine before embarking on the design of a full-size machine. However, apart from its small size, the machine is, in principle, 'universal' in the sense that it can be used to solve any problem that can be reduced to a programme of elementary instructions; the programme can be changed without any mechanical or electro-mechanical circuit changes.

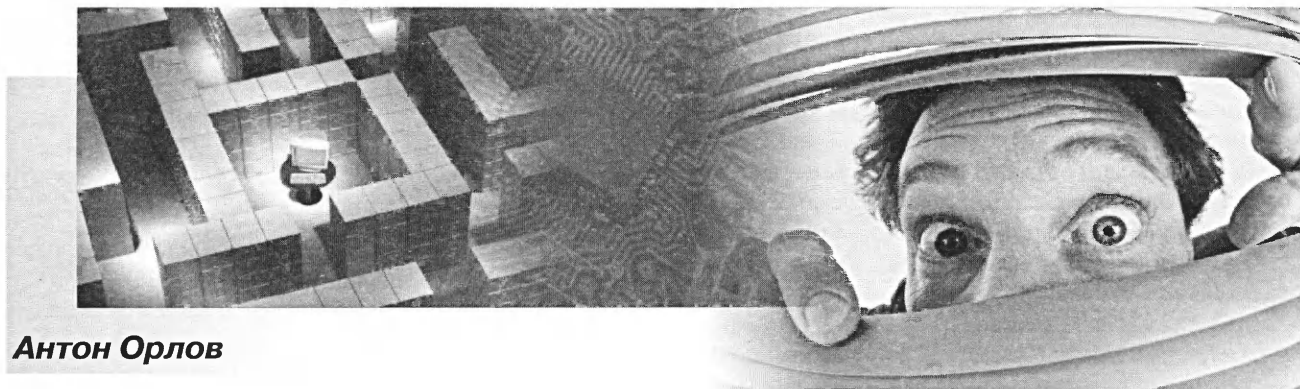
The development of this machine has been very actively supported by the Telecommunications Research Establishment, Great Malvern.

F. C. WILLIAMS  
T. KILBURN

Electrical Engineering Laboratories,  
University, Manchester 13.  
Aug. 3.

September 25, 1948 NATURE

Практически единоличным автором изобретения был Фредди Уильямс, а его энергичным помощником и составителем первой программы для факторизации (разложения на



Антон Орлов

# Тайны системного реестра

Продолжение. Начало см. "Магия ПК" №5/2001

## Системный блок

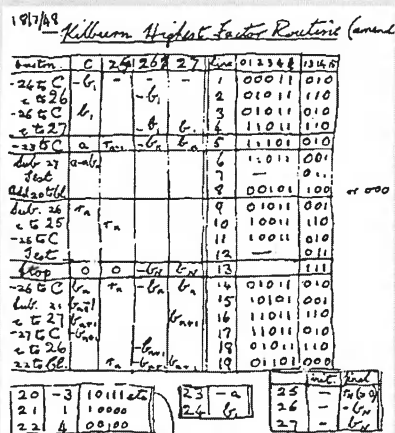
Следующий раздел реестра — HKEY\_LOCAL\_MACHINE, хранилище информации о вашем компьютере и о независимых от наличия различных пользователей настройках программ. Он — самый крупный и важный из всех разделов реестра, со-

держит и весь раздел HKEY\_CLASSES\_ROOT, который вынесен в отдельный ключ реестра лишь для удобства редактирования, и все возможные варианты содержимого раздела HKEY\_CURRENT\_CONFIG, в который помещается информация из HKEY\_LOCAL\_

MACHINE в зависимости от текущей аппаратной конфигурации.

Как HKEY\_LOCAL\_MACHINE в Редакторе реестра отображается файл System.dat, то есть вся информация этого ключа физически содержится именно в этом файле. Таким образом, HKEY\_LOCAL\_MACHINE — первый из описанных "реальный" раздел

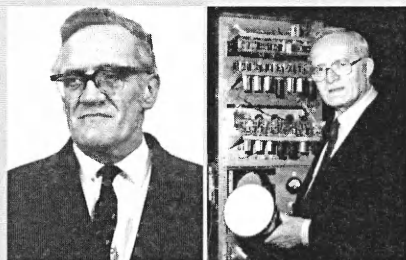
множители) большого целого числа ( $2^{30}-1$ ) стал Том Килбурн.



Программа Килбурна

По своим вычислительным ресурсам Baby была действительно малышкой: операционное поле машины ограничено 128 байтами, отведенными для хранения данных и программы, хотя занимала она целую комнату и выглядела весьма уг-

ловато в силу несоразмерности комплектующих: их пришлось собирать из кабин поврежденных во время войны английских и американских самолетов.



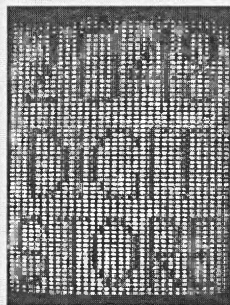
Ф. Уильямс

Т. Килбурн

По словам авторов, принципиальной трудностью в реализации компьютера было создание быстрого оперативного запоминающего устройства. Магической палочкой-выручалочкой для проекта Baby стала электронная лучевая трубка (как ее теперь

называют), которая была названа трубкой Уильямса, или CRT-памятью. В основу ОЗУ был положен эффект остаточного заряда на экране ЭЛТ после попадания на нее электронного пучка. Такая память была не совсем надежной, но зато по тем временам очень быстрой и дешевой. Да и ввод данных с клавиатуры позволил существенно ускорить процесс.

Фактически это был вариант первого быстродействующего ОЗУ (время доступа — доли миллисекунд) с последовательным доступом (из-за построчной развертки луча). В числе прелестей нового ОЗУ было совмещение функций: сам экран электронной трубки исполнял роль дисплея, на котором контролировались и данные, и записанная в память компьютера программа.



ОЗУ на ЭЛТ

реестра, то есть имеющий под собой "физическую природу" — файл System.dat (HKEY\_CLASSES\_ROOT и HKEY\_CURRENT\_CONFIG — его подразделы, а HKEY\_CURRENT\_USER — подраздел HKEY\_USERS, второго "реального" раздела реестра). Подраздел Config содержит в себе данные о зарегистрированных в системе аппаратных конфигурациях. Windows позволяет при регулярных изменениях состава устройств в компьютере (например, при работе с ноутбуком, который имеет "док" — присоединяемую часть с CD-ROM, модемом и др.) не устанавливать и удалять каждый раз драйверы для всех этих устройств, а создать фиксированный набор драйверов для каждого из этих комплектов устройств и менять эти наборы при необходимости. Такой набор называется аппаратной конфигурацией и устанавливается через окно Панель управления — Система — Конфигурация.

В настоящее время в связи с развитием технологий USB и других средств "горячего подключения" необходимость в различных аппаратных конфигурациях почти отпала, но

От версии к версии размеры ОЗУ увеличивались за счет увеличения как диаметра ЭЛТ (до 15 дюймов), так и их количества, что привело к естественному формированию концепции "страничной" организации архитектуры ОЗУ, со временем достигшего объема 2 Кб (!). Это не было пределом для увеличения размера ОЗУ, но здесь плодотворная концепция ОЗУ зашла в тупик в силу естественного человеческого фактора: считать глазами данные с такого дисплея стало просто невозможно. Это привело к разработке в 50-е годы ОЗУ на магнитных сердечниках, а представление данных на экране дисплея стало символьно-цифровым.

Что же касается Baby, то на ее основе была создана более мощная ЭВМ "Mark I". Вскоре к проекту присоединился Алан Тьюринг и написал один из первых, еще примитивных вариантов ассемблера.

*Алексей Смирнов*

их поддержка в Windows присутствует. В ключе Config под своими номерами располагаются записи о всех созданных конфигурациях, а при загрузке ПК или изменении конфигурации соответствующий раздел "отражается" в реестре под видом раздела HKEY\_CURRENT\_CONFIG. Если аппаратная конфигурация одна, то и в разделе Config есть только один подраздел, совпадающий с ключом HKEY\_CURRENT\_CONFIG.

В разделе HKEY\_LOCAL\_MACHINE\Config находятся и некоторые настройки операционной системы. Например, в ключе Config\0001 (номер конфигурации)\display\settings указываются шрифты, используемые Windows по умолчанию, например, в "Блокноте". А в параметре Resolution этого раздела расположено указание на экранное разрешение. Можете, кстати, его поменять на любые цифры, хоть на 1000x200, и если ваша видеокарта позволит, то именно таким разрешением экрана и будет.

Подраздел DesktopManagement посвящен работе системы с библиотекой deskmgmt.dll из каталога Windows\System, отвечающей за работу операционной системы с процессором, памятью и другими устройствами материнской платы. Подраздел Enum содержит информацию о распознанных операционной системой устройствах ПК. В Windows есть несколько важных программных компонентов, отвечающих за обнаружение устройств и предоставление их для работы другим компонентам системы. Основными среди них являются два — диспетчер конфигурации и эnumератор. При загрузке ПК диспетчер конфигурации опрашивает BIOS и получает от него информацию о всех устройствах, не поддерживающих стандарт Plug&Play, а затем заносит ее в реестр, в этот самый раздел Enum. Затем в дело вступает программа-эnumератор, которая считывает из реестра информацию о не-Plug&Play устройствах, а затем опрашивает имеющиеся в компьютере Plug&Play-устройства и составляет их список, назначив каждому свой идентификатор — HardwareID. Затем

основная роль вновь переходит к диспетчеру конфигурации, который на основании информации эnumератора о наличии устройств и их характеристике (Plug&Play или нет) распределяет между устройствами прерывания, адреса ввода-вывода и другие ресурсы системы. В этом ему помогают специальные подпрограммы-арбитры, которые изменяют выделяемые ресурсы так, чтобы не было используемых одновременно двумя устройствами. Вначале распределяются ресурсы между не-Plug&Play устройствами (информация о которых пришла еще из BIOS), а затем — между Plug&Play, так как последние умеют работать с разными настройками распределения ресурсов, и переназначить им, например, прерывания можно программно. Все внесенные изменения фиксируются в разделе Enum, а также в разделе HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\Class, содержащем информацию о распознанных устройствах и о загруженных для них драйверах.

Подраздел HKEY\_LOCAL\_MACHINE\Hardware хранит в себе информацию о последовательных портах и установленных на них модемах (на уровне перечисления имеющихся в ПК портов и разъемов для модемов).

Подразделы Network и Security содержат малозначимые параметры сети Microsoft Network.

И вот, наконец, второй подраздел Software, на этот раз уже состоящий из универсальной для всех пользователей данного компьютера информации! Он состоит из разделов с именами компаний-производителей с подразделами внутри них, посвященных отдельным программам. В разделе HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft расположены настройки самой Windows и ее компонентов, а также всех программ Microsoft. Например, в подразделе Active Setup этого раздела находятся параметры системы установки компонентов Windows через Интернет — Active Setup. В его подразделе Installed Components вы увидите все компоненты Internet Explorer, которые вы выбирали (или

нет) при установке этого браузера. В подразделе Shared Tools раздела HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft находится список программ и библиотек пакета MS Office, которыми могут пользоваться другие компоненты этого пакета (скажем, "галерея клипов", которая может быть вызвана из любого приложения MS Office), с указанием пути к каждому из них. Но, конечно, самым интересным местом здесь будет подраздел Windows\CurrentVersion, относящийся к операционной системе. В самом CurrentVersion, даже не в его подразделах, есть весьма полезные параметры. Скажем, именно здесь находится информация, отображающаяся в верхней части окна Панель управления — Система — Общие.

Подраздел App Paths раздела CurrentVersion — это аналог параметра Path из файла autoexec.bat. Можно назвать его реестровым собранием ярлыков программ для Windows. Здесь можно указать пути к различным программам и имена для их запуска, а затем набирать эти имена в окне "Выполнить" для запуска этих программ, не заботясь об указании пути к ним.

Подраздел Explorer опять-таки посвящен программе "Проводник", но уже ее общим настройкам. В подразделах Desktop\NameSpace и MyComputer\NameSpace находятся идентификаторы тех объектов, которые должны быть расположены на Рабочем столе и в папке "Мой компьютер", например, "Корзина" или "Удаленный доступ к сети". Вы можете удалить отсюда ненужные объекты или, наоборот, добавить сюда нужные, создав вначале для них идентификаторы в разделе HKEY\_CLASSES\_ROOT.

В параметрах подразделов LastCheck и LastOptimize (подраздел Explorer подраздела CurrentVersion) содержится информация о последней

дефрагментации или проверке жестких дисков ПК.

Подраздел Fonts хранит в себе информацию об именах шрифтов и их соответствии названиям файлов. Вы, кстати, никогда не удивлялись, почему в "Проводнике" и в Norton Commander содержимое папки Windows\Fonts выглядит по-разному? Да потому, что Norton Commander берет информацию об именах файлов в этой папке из FAT, а "Проводник" — вот из этого раздела реестра. Почему копирование шрифтов в папку Fonts

длится значительно дольше, чем копирование в любую другую папку? А потому, что еще в реестре их все надо прописать, а перед этим из каждого извлечь полное имя!

В подразделе MS-DOS Emulation (подраздел AppCompat) находится список программ MS-DOS, которые могут работать только в "Режиме MS-DOS", а не в окне "Сеанса MS-DOS", и требуют для запуска выгрузки Windows.

Из следующего подраздела реестра, MS-DOSOptions, берутся параметры для диалоговых окон при настройке ярлыков программ MS-DOS для особой конфигурации "Режима MS-DOS". Именно с помощью этих параметров можно удобно настроить файлы autoexec.bat и config.sys для каждого ярлыка программы, работающей в "Режиме MS-DOS".

Подразделы Run, RunServices, RunOnce, RunOnceEx, RunServicesOnce содержат в себе указания на программы, которые должны запускаться автоматически при запуске Windows. Почаще навещайте этот

раздел — здесь могут прятаться трюны! Три последних раздела из перечисленных запустят указанную в них программу лишь один раз. В разделе HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies (подраздел Ratings) есть параметр

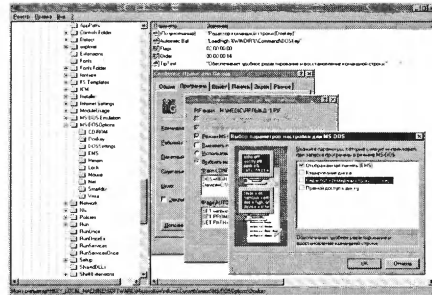
Key. Сотрите его, если вы случайно в процессе исследования MS Internet Explorer задали пароль на ограничение доступа к отдельным ресурсам Интернета (Свойства обозревателя — Содержание — Ограничение доступа)

и забыли его. Если вы хотите изменить путь к установочному файлу Windows (например, поставили Windows с компакт-диска, а затем поместили ее дистрибутив на винчестер и хотите, чтобы при любой установке драйверов или своих компонентов она обращалась именно туда), то введите в параметр SourcePath раздела HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup нужный путь.

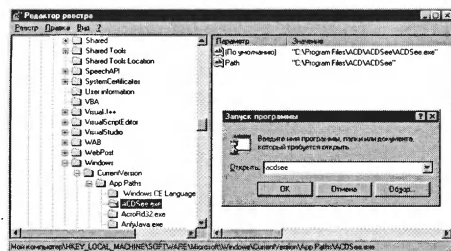
В подразделе Time Zones приведен список настроек всех возможных часовых зон из меню настройки местного времени (параметры установленной временной зоны размещаются ниже, в разделе HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\control\TimeZoneInformation).

Подраздел Uninstall (подраздел Explorer подраздела CurrentVersion) содержит в себе пути и параметры деинсталляторов для всех программ, которые допускают такую возможность. Именно отсюда берется информация для представления в окне Панель управления — Установка и удаление программ. Параметр DisplayName определяет название программы, которое в этом окне будет отображаться.

Если вы хотите напугать какого-нибудь любителя "Lines" или "Сапе-



Отсюда берут свои параметры диалоговые окна настройки ярлыков режима MS-DOS



Укажите в этом разделе имя для запуска программы и путь к ней, и можете не думать о задании правильного пути в окне "Выполнить"

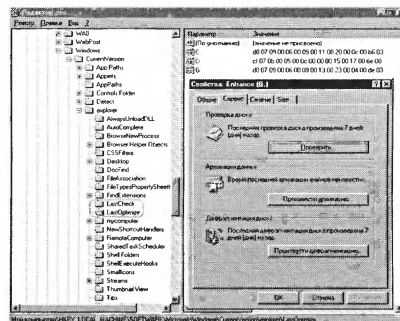
ра", включившего компьютер без вашего ведома, вставьте в раздел `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Winlogon` параметры `LegalNoticeCaption` и `LegalNoticeText` с неким устрашающим текстом. Тогда при загрузке Windows будет отображаться окошко с этими данными.

Ну, а теперь перейдем к самому последнему подразделу ключа `HKEY_LOCAL_MACHINE\System`, к его единственному подразделу `CurrentControlSet`. В этом подразделе содержатся два ключа: `Control` и `Services`. Параметры этой части реестра не менее важны, чем, скажем, подраздела `Enum`, так как содержат в себе многие настройки системы. Но большая их часть — чисто служебная, и изменять их через редактор реестра не стоит. Впрочем, можно слегка пробежаться по наиболее интересным разделам.

В подразделе `FileSystem` записаны некоторые параметры работы операционной системы с носителями, имеющими другую файловую систему, вроде CD-ROM.

Подраздел `Keyboard layouts` содержит список возможных раскладок клавиатуры и имена соответствующих им файлов с раскладками (установлены могут быть отнюдь не все эти файлы). Подраздел `Nls` посвящен кодовым страницам, которые поддерживаются системой. В его подразделе `Codepage` каждому номеру кодовой страницы (например, "1251" — кодировка Windows-1251) поставлен в соответствие файл, содержащий информацию об этой кодовой странице, а в подразделе `Locale` — имена этих страниц. Иногда проблемы с отображением русских символов в некоторых программах можно устранить, поставив значение параметра "1252" в разделе `HKEY_LOCAL_MACHINE\System\CurrentControlSet\control\Nls\Codepage` в "CP\_1251.nls". Подраздел `PerfStats` посвящен состоянию системы, подраздел `SessionManager` — старым приложениям MS-DOS. В подразделе `CheckBadApps` содержится список программ MS-DOS, которые могут быть несовместимы с Windows. Наверное, вы пробовали

из-под Windows запустить Norton Commander 5.0 и получали сообщение, что программа некорректно работает в этой операционной системе? Для каждой программы MS-DOS, при попытке запуска которой выдается такое сообщение, есть свой подраздел с именем ее запускающего файла в подразделе `CheckBadApps`, а в нем — параметр с именем запускающего файла, в котором указано число. Это число — указатель на номер справочного раздела в файле `Apps.hlp` из папки `Windows\Help`, в котором рассказано, почему данная программа некорректно работает в Windows.



*Вот откуда компьютер знает, сколько дней назад вы последний раз дефрагментировали винчестер*

Подраздел `CheckBadApps400` имеет то же предназначение, что и предыдущий, но в нем размещены имена программ для Windows 3.x, а не для MS-DOS. Подразделы подраздела `SessionManager` с буквами "DLL" в имени хранят в себе имена системных библиотек, с которыми работает Windows, в зависимости от имени раздела — 16- или 32-рядных. А в подразделе `WarnVerDLLs` приводятся имена библиотек, совместимость которых с Windows вызывает у нее "подозрения".

Подраздел `TimeZoneInformation` (подраздел `System\CurrentControlSet\control`) хранит в себе информацию о текущей временной зоне. В следующем большом подразделе `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services` содержится в основном служебная системная информация. Некоторые из его подразделов очень важны для системы.

Подраздел `Arbitrators` содержит

информацию программ-арбитров, о которых шла речь выше. Подраздел `Class` переключается по содержанию с описанным выше разделом `HKEY_LOCAL_MACHINE\Enum`: частично берет свои данные из него и содержит информацию об установленных и распознанных устройствах. Именно из него отображается информация в окне Панель управления — Система — Устройства. Фактически этот подраздел является "реестром в реестре" — базой данных по устройствам, установленным в ПК. В него включена информация, поставляемая эnumerатором и диспетчером конфигурации и записанная в разделе `HKEY_LOCAL_MACHINE\Enum`, а также данные о загруженных драйверах устройств и их параметрах (к примеру, строка инициализации модема помещается в ключе `Class\Modem\0000\Init`). Вспомните, кстати, что в `Enum`'е были ссылки на драйверы устройств вида `Driver = DiskDrive\0001`. Так вот, это были ссылки именно на подразделы данного раздела `Class`. Таким образом, раздел `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class` — основной держатель информации об устройствах ПК, назначенных им ресурсах и загруженных для них драйверах. Раздел `HKEY_LOCAL_MACHINE\Enum` тесно связан с ним и хранит "технические" данные для каждого устройства. В разделе `HKEY_DYN_DATA` располагаются данные о текущем состоянии каждого из устройств.

В разделах `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class` и `HKEY_LOCAL_MACHINE\Enum` содержится информация лишь об имеющихся устройствах, их характеристиках и назначенных ресурсах. Информация о конкретной работе устройств, об их исправности, находится в разделе `HKEY_DYN_DATA`, который постоянно находится в оперативной памяти. Остальная часть раздела `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services` содержит данные некоторых программ и особого интереса для пользователя не представляет.



**Б**еру на себя смелость утверждать, что шар является основой мироздания. И не только потому, что наша планета — шар. Миллионы шаров неустанно крутятся в шарикоподшипниках, светильники в виде шаров освещают улицы и интерьеры, в аэропортах возвышаются громадные шары, в которых спрятаны радары. А сколько видов спорта заканчивается на "ball"! Шары кидают, бросают, толкают, бьют руками, ногами, головой, клюшкой, ракеткой, битой.

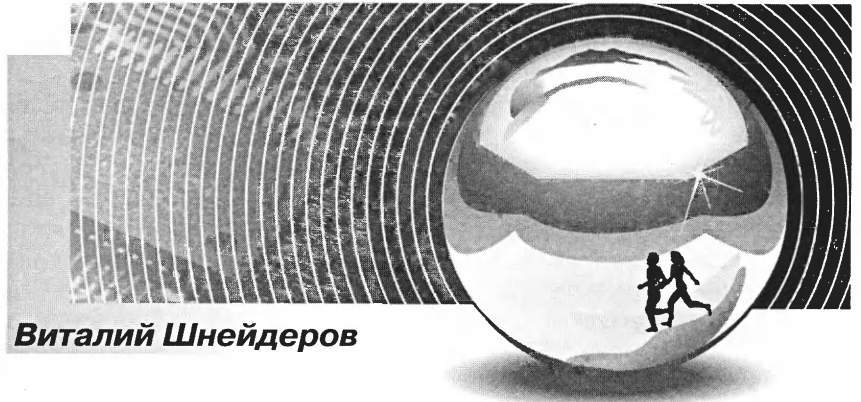
Но шары не только труженики, шары — это праздник! Девушки носят бусы с шариками из камня, стекла, металла или дерева. Праздничное шествие на улицах невысказанно без надувных шариков, а если праздник масштабный, то тысячи шариков отпускают в небо. На воздушных шарах люди отправляются в небо. А с чем может сравниться красота переливающихся красок мыльных пузырей? Елочные игрушки зеркальными шарами преломляют окружающее, придавая всему необычный блеск.

Но шары не только радуют глаз, их идеальная форма вызывают приятные тактильные ощущения. Перебирая пальцами четки-шарики, человек успокаивается. Шары — элементы психотерапии.

Наконец, в 1957 году шар был запущен в космос.

Да, при таких заслугах шару давно пора поставить памятник. Кстати, скульптур, изображающих шар, очень много. Там, где скульптура сторожевого льва, рядом тоже шар! Аналогия очевидна: лев — царь зверей, а шар — царь объектов!

Простота вычисления кривизны сферической поверхности привела к тому, что шар стал тем "пробным шаром" (вот видите, даже в лингвистике без этого слова не обойтись), с которого начиналась трехмерная машинная графика, на чем оттачивались формулы рендеринга. Сейчас можно управлять широким набором оптических характеристик материалов, таких как отражение, преломление, непрозрачность, самосвечение, диффузное отражение, рельефность, зеркальное отражение и др.



Виталий Шнейдеров

## Очарование шара

Как же можно построить шар средствами машинной графики? Есть несколько способов, наиболее простые и доступные мы и рассмотрим.

### Градиентная радиальная заливка

Пример шара, полученного при градиентной радиальной заливке круглой маски, приведен на рисунке 1. Центр заливки выбирается в том месте, где должно быть самое светлое место. В данном случае параметры смещения блика такие: Center Offset, Horizontal — минус 25%, Vertical — 25%. Можно воспользоваться и готовой заливкой, то есть приведенный ниже рисунок получить с помощью заливки Circle grey1. К градиентной заливке можно добавить и текстуру в виде случайно распределенных точек, которые создают эффект вкраплений. Для этого нужно добавить шум — Add Noise More spike. Изменяя размер точек с помо-

щью фильтров Blur и Sharpen, можно менять вид вкраплений (рис. 2).

Для усиления объема воспользуемся таким эффектом. По периметру шара текстура кажется нерезкой. Для размытки шума применим фильтр Radial Blur. Использование этого фильтра позволяет размыть точки по краям шара, оставив их резкими в центре. Таким образом, фильтр Radial Blur имитирует расфокусировку по глубине. Следующим шагом повышения реализма шара является добавление блика и тени. Блик можно создать с помощью фильтра Light Effects, а тень — фильтром Drop of shadow, но для этого шар должен быть оформлен в виде объекта.

Методом градиентной заливки можно получать не только выпуклости, но и углубления. В углублении от шара все оказывается наоборот: самое светлое место на шаре становится самым темным в углублении, а самое темное — самым светлым. Ло-

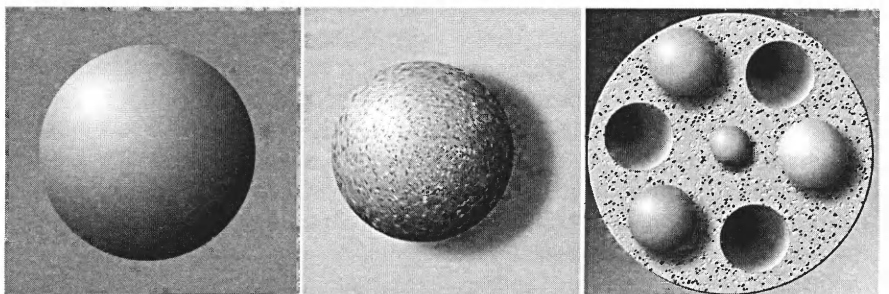


Рис 1, 2, 3

гика подсказывает, что если мы проинвертируем шар фильтром Invert, то получим углубление (рис. 3).

### Фильтр Sphere

Фильтр Sphere создает впечатлительное выпучивание изображения в сферу. Известный голландский график Эшер рисовал картины, которые как будто бы продавлены кулаком. Для этого требовалось незаурядное воображение. Теперь же фильтр Sphere позволяет любое изображение преобразовать (натянуть) на сферу. Как он работает, наглядно демонстрирует рисунок 4. Фоновый текст после применения фильтра оказался на сфере.

С помощью этого фильтра можно "оклеить" шар, например, фотографиями (рис. 5). Для этого к исходной фотографии сначала применим фильтр Tile (черепица). Затем фон градиентной радиальной заливкой, имитирующей контровое освещение, отбросим тень от шара. Вот вам и шар, как настоящий.

Фильтр Sphere лучше применять не ко всему изображению, а только к его фрагменту. Для выбора фрагмента используются маски. Чтобы шар производил впечатление воздушного или стеклянного, нужно выбрать из фотографии наиболее светлый фрагмент, например, облака (рис. 6). Здесь шар кажется реальным своей фактурой и, в то же время, нереальным своими размерами и положением в пространстве. Огромный шар, висящий в небе, создает напряжение. Мы восхищаемся красотой этого шара, удивляемся его размерам. Шар, созданный из "воды и неба", как бы несет в себе их стихию, прозрачность неба и вес огромной массы воды.



Рис. 4. Шар-текст

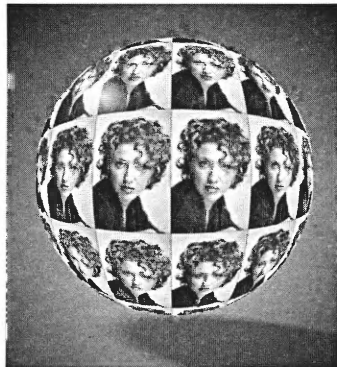


Рис. 5. Шар-фото

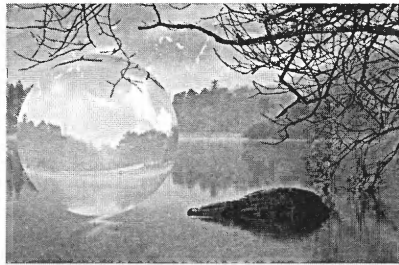


Рис. 6

Аналогичным образом шар, созданный из кирпича, несет в себе его твердость и вес (рис. 7). Здесь для усиления объема с помощью фильтра Drop of Shadow на стену отбросена тень от шара.

Да, по этим шарам мы видим, что фильтр Sphere открывает возможности для реализации любых самых фантастических проектов. Обратимся к космосу. Космические сюжеты требуют абстрагирования от действительности. Для этой цели воспользуемся фильтром Extrude. Он создает абстрактный фон. Космическую станцию получим из фона путем предварительного маскирования квадратной области. Применим к ней соларизацию для усиления эффекта металлической поверхности. Чтобы показать, что эта станция совершила далекое межпланетное путешествие, нужно обозначить следы ее встреч с микрометеоритами. Для этого на "космическую" конструкцию (маскированную поверхность) до запуска фильтра Sphere нанесем шум (фильтр Add Noise).

Ну вот, космический странник готов (рис. 8). На приведенных выше рисунках шары получены, можно сказать, прямо по месту, за счет применения фильтра Sphere.

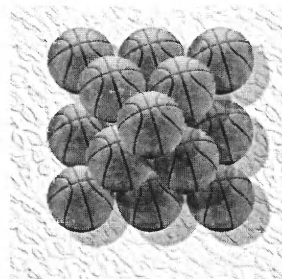


Рис. 9



Рис. 7

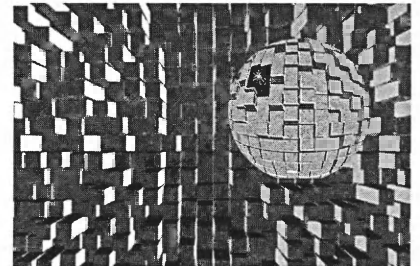


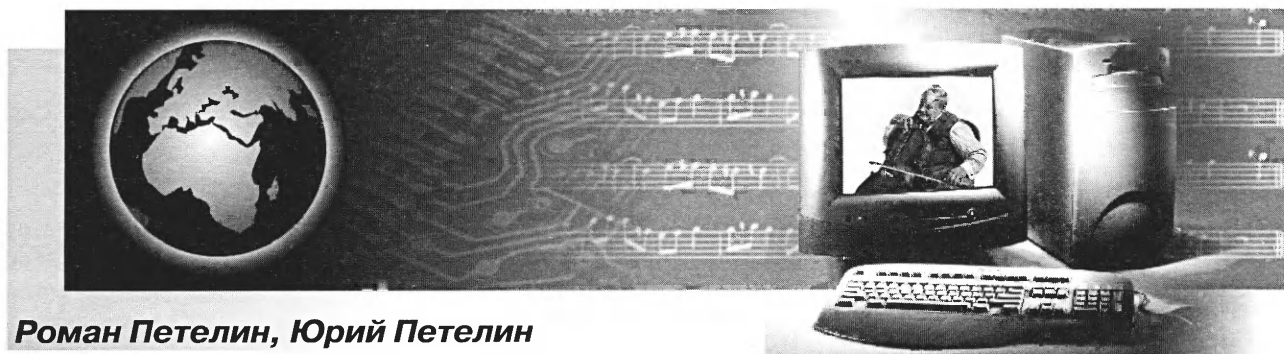
Рис. 8

### Готовые шары

В PhotoPaint имеется набор шаров-примитивов в инструменте Image Sprayer. Это шары из цветного стекла, гранита, жевательной резинки, спортивные мячи. Работа в Image Sprayer напоминает процедуру рисования: что нарисовано первым, то оказывается снизу. На рисунке 9 получена пирамида из 14 баскетбольных мячей. Ее построение имитирует работу с реальными шарами: начинать нужно с нижнего яруса.

"Укладываем" в основание девять мячей. Затем в ямки, образованные каждым из четырех мячей, "кладем" еще по одному мячу. Таким путем строится второй ярус. Наконец, осторожным движением кладем наверх последний мяч. И хотя мы работаем мышкой, возникает полная иллюзия, что мы строим пирамиду из шаров, — одно неосторожное движение, и она рассыплется. Добавив тени, усилим объем пирамиды.

В принципе, перечисленных выше основных приемов вам будет достаточно, чтобы создать свой неповторимый шар. Тем же, кому этих возможностей окажется мало, рекомендую фильтр KPT Spheroid Designer. Его возможности по созданию шаров просто ошеломляют.



Роман Петелин, Юрий Петелин

## Cool Edit Pro — космические технологии в музыке

**П**рограммное обеспечение, предназначенное для обработки звука и редактирования музыки, стремительно развивается, функциональные возможности программ расширяются. Математики и инженеры, разрабатывающие программы, все глубже проникают в тайны музыкального творчества и предлагают новые технические и математические способы решения проблем как чисто музыкальных, так и находящихся на стыке музыки и акустики. Постепенно дело идет к тому, что человеку, использующему в своем творчестве компьютер, уже недостаточно будет оставаться только музыкантом и пользователем персонального компьютера. Ему придется овладеть новыми техническими терминами, понять сущность операций со звуковыми данными и алгоритмов обработки музыкального материала, смоделированных в программах. Задача для музыканта не из легких. Ведь к началу XXI века в музыкальных программах нашли применение многие результаты исследований в области математики, технической кибернетики и теории сигналов. Вот только некоторые из них:

- Оптимальная обработка, корреляционный и спектральный анализ сигналов на фоне помех с различными статистическими свойствами.

- Синтез звуков с заданным частотно-временным распределением в различных ортогональных базисах.
- Адаптивные методы аналого-цифрового преобразования.
- Полиномиальная аппроксимация амплитудно-частотных характеристик фильтров.
- Устранение искажений на основе интерполяции сигнала по методу максимального правдоподобия.

Даже достижения таких отраслей знаний, как психоакустика, поставлены на службу музыке.

### **Cool Edit Pro — это очень сложно**

Лучшей иллюстрацией применения в программах для записи музыки и редактирования звука достижений современной науки может служить Cool Edit Pro — звуковой редактор, в котором воплощено все самое передовое из области обработки сигналов. "Магия ПК" уже не впервые обращается к Cool Edit Pro. Мы вполне согласны с автором статьи "Cool Edit Pro — это очень просто" (№ 4, 2000) в том, что интерфейс программы достаточно прост и интуитивно понятен. Но за этой внешней простотой скрываются сложные и эффективные алгоритмы, методы, технологии. Подобные технологии принято называть космическими. И

это не только образное выражение. В самом деле, оптимальная фильтрация сигнала на фоне помех, которая применялась при радиолокационном исследовании Венеры, используется в Cool Edit Pro для устранения шума, сопровождающего фонограмму, записанную с микрофона. Или возьмем алгоритм интерполяции, способный сотворить чудо — восстановить первоначальную форму клипированного сигнала. Этот алгоритм подобен тем, что позволяют воссоздать фрагменты переданных с борта космического аппарата на Землю сигналов, когда они оказываются утраченными, например, из-за возмущений в ионосфере.

Только радиолюбитель сумеет оценить по достоинству возможность несколькими легкими движениями руки с мышью синтезировать фильтр Бесселя 30-го порядка с идеально прямоугольной амплитудно-частотной характеристикой.

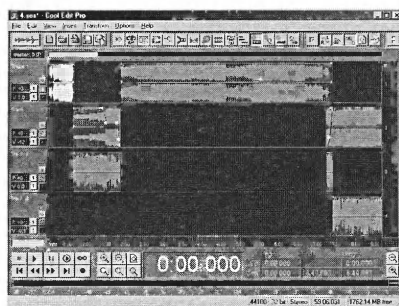
В общем, более наукоемкого звукового редактора, чем Cool Edit Pro, пока нет. А между тем прототип этого редактора создан всего одним человеком — Дэвидом Джонсом. Права на распространение программы принадлежат фирме Syntrillium Software Corporation. Популярность Cool Edit Pro отчасти объясняется тем, что более ранняя версия редактора, Cool Edit 96, была свободно

распространяемой (Share Ware). Да и сейчас его демоверсии можно найти в Интернете на сайте [www.syntrillium.com](http://www.syntrillium.com). На протяжении ряда лет автор программы непрерывно улучшал свое детище и практически достиг совершенства. Самая свежая версия программы Cool Edit Pro 1.2.a представляет собой звуковой редактор, обладающий всеми возможностями, какие только могут быть у программ такого класса.

### Два мира — две системы

Итак, Cool Edit Pro — это программа для работы со звуком, представленным в цифровой форме. Это означает, что аналоговые звуковые волны должны быть предварительно преобразованы (сэмплированы) в последовательность двоичных цифровых отсчетов. Такое преобразование осуществляется в аналого-цифровом преобразователе (АЦП). В результате работы АЦП получается цифровой образ звука, в английском языке именуемый Waveform. Некоторые пользуются буквальным переводом этого термина: волновая форма. Мы же предпочитаем именовать объект, с которым оперирует звуковой редактор, сэмплом. Сэмплы хранятся на жестком диске в файлах, имеющих различные форматы. Чаще всего это файлы с расширением .WAV. Таким образом, когда вы "собираете" в Cool Edit Pro аудиокомпозицию (например, песню), WAVE-файлы служат стандартными элементарными блоками, кирпичиками. В текущей версии Cool Edit Pro существуют два принципиально различных режима работы: редактирование отдельных сэмплов и совместное мультитрековое редактирование группы сэмплов. Для каждого из ре-

жимов предусмотрено свое главное меню и свое главное окно: Edit Waveform View и Multitrack View.



Окно Multitrack View

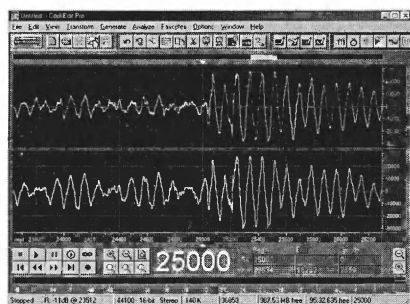
По сути дела это два различных по назначению звуковых редактора, объединенных в единый комплекс. В первом из этих окон вы работаете с каждым фрагментом будущего микса в отдельности. Здесь вырезаются ненужные части фонограммы, удаляются щелчки и шум, формируется тембр, корректируется динамический диапазон, применяются всевозможные эффекты. По мере готовности отдельных фрагментов вы экспортируете их в мультитрековый редактор Multitrack View. В нем фрагменты можно перемещать, располагая в необходимой последовательности. Например, если сначала должна идти заставка, затем дикторский текст, музыкальная композиция и рекламный модуль, то эти фрагменты должны быть расположены по времени один за другим. В принципе, в данном случае монтаж можно было бы выполнить и на одном треке. Но если дикторский текст должен звучать на фоне музыки или нужно, чтобы перечисленные фрагменты собирались "внахлест" (диктор еще говорит, а музыка уже начинается), то без нескольких треков обойтись будет трудно. Перемещая фрагменты во времени и с трека на трек, можно добиться, чтобы фонограмма укладывалась в отведенный хронометраж. Для получения необходимого баланса регулируются уровни громкости фрагментов, размещенных на отдельных треках. Можно нарисовать графики изменения громкости. Положение источников звука на стереопанораме и их перемещение также задается графическим способом независимо для каждого из треков.

Операции редактирования, выполняемые в Edit Waveform View (вырезка, вставка, обработка эффектами и др.), по сути являются разрушающими, и когда вы сохраняете файл, изменения применяются непосредственно к сэмплу. Редактирование, выполненное в Multitrack View (такие, как состыковка сэмплов, изменение громкости), является неразрушающим.

Во многих музыкальных редакторах, имеющих средства обработки аудиоданных, предусмотрены два варианта использования аудиоэффектов: применение эффекта в реальном времени и пересчет сэмпла. Первый вариант для пользователя удобнее: регулируете параметры эффекта и тут же слышите результат. Однако при этом требуется компьютер большой производительности. Применение эффекта путем пересчета сэмпла позволяет обрабатывать звуковые данные с помощью относительно слабого компьютера, но процесс обработки может растягиваться на десятки минут или даже часы. Поневоле не захочешь лишний раз вносить улучшения, пробовать что-нибудь новенькое.

Оптимальная стратегия использования эффектов может состоять в следующем. Сначала, включив циклический режим воспроизведения сэмпла, вы в реальном времени подбываете параметры эффекта, наилучшим образом соответствующие вашему замыслу. После этого единственный раз вы пересчитываете сэмпл, применив к нему эффект с уже выверенными параметрами.

В Cool Edit Pro вы можете использовать как собственные, встроенные в программу эффекты, так и эффекты, подключаемые посредством DirectX. Основной режим их применения — пересчет сэмплов. Полноценный режим реального времени в программе не предусмотрен, что создает определенные неудобства. Однако в ряде случаев замена режиму реального времени все же имеется. Если в окне того или иного эффекта есть кнопка Preview (предварительное прослушивание), то вы сможете подобрать параметры эффекта. Результаты изменений пара-



Окно Edit Waveform View

метров эффекта с помощью регуляторов в окне данного эффекта будут слышны немедленно.

Cool Edit Pro является многоканальной средой, то есть дает возможность поместить любое число сэмплов на различные треки как для одновременного воспроизведения, так и для последующего объединения всех треков в один. Процесс микширования включает в себя объединение всех сэмплов, помещенных на треках, в два (или больше) каналов вывода. Вы можете редактировать, добавлять на треки и убирать с них сэмплы. А Cool Edit Pro будет постоянно отслеживать все изменения (к примеру, изменение громкости), происходящие в ходе многоканального сеанса. Как только что-либо изменено, Cool Edit Pro немедленно вносит необходимые коррективы в микс, поступающий на выход программы (например, в ЦАП звуковой карты). Эти изменения выполняются программой в фоновом режиме.

В дополнение к WAVE-файлам Cool Edit Pro позволяет также использовать файлы, именуемые файлами сеанса, или сессионными (Session) файлами (файлами с расширением .SES). Если считать WAVE-файлы стандартными блоками, из которых строится любая аудиокomпозиция (например, песня), то сессионный файл — это и есть собственно песня, хотя SESSION-файл фактически не содержит никаких звуковых данных. Размер SESSION-файла очень мал. В нем содержатся только подробные указания программе Cool Edit Pro:

- Какие WAVE-файлы используются и где они хранятся.
- Какие имена присвоены трекам.
- В какой момент времени включить воспроизведение определенного WAVE-файла, а в какой — выключить.
- Какой уровень громкости и какую панораму установить перед началом воспроизведения каждого из файлов и по каким законам менять эти параметры в процессе воспроизведения.

Можно сравнить SESSION-файл с дирижером, а WAVE-файлы — с

оркестрантами. Дирижер дает знать каждому исполнителю, когда он должен вступить и с какими нюансами вести свою партию. Но дирижер без оркестра ничего не значит. Аналогично и SESSION-файл имеет смысл только тогда, когда и он, и WAVE-файлы, участвующие в сессии, находятся в определенных и взаимобусловленных каталогах. Вы не можете просто скопировать на гибкий диск и перенести его на компьютер друга, не можете произвольным образом переименовать или переписать в другой каталог какой-либо из WAVE-файлов, задействованных в сессии. Если, совершив любое из этих действий, вы попытаетесь открыть SESSION-файл и воспроизвести композицию, ничего из этого не выйдет. Дирижер не обнаружит музыкантов на положенных местах, и концерт не состоится. Это не означает, что SESSION-файлы и все внедренные в них WAVE-файлы нельзя перемещать из каталога в каталог. Можно. Но для этого в Cool Edit Pro существуют специальные операции сохранения сессии на диске.

### **Чтобы душа сначала развернулась...**

Очень много интереснейших средств преобразования звуковых данных сосредоточено в меню Transform. Вы можете:

- Заменить звук абсолютной тишиной.
- Поменять местами "начало" и "конец" звукового фрагмента.
- Подключить внешние эффекты.
- Изменить знаки отсчетов звуковых волн на противоположные (положительные полуволны станут отрицательными, и наоборот).

Но и это еще не все. Как вы относитесь к возможности применения: 7 средств амплитудной обработки; 9 эффектов, основанных на задержке сигнала; 7 разновидностей частотных фильтров; 4 модулей подавления шумов, устранения щелчков и искажений; 2 алгоритмов сдвига тона по высоте и сжатия-растяжения звукового фрагмента во времени; 4 специальных эффектов? Под специаль-

ными эффектами разработчики программы понимают очень интересные вещи:

- Свертку двух сигналов (свертка — термин теории сигналов, операция интегрирования произведения двух сигналов при всех возможных взаимных сдвигах). Один из свертываемых сигналов может представлять собой запись "короткого" звука (скажем, одиночного хлопка в ладоши), выполненную в вашем любимом концертном зале. Тогда ощущение присутствия именно в нем даст свертка музыкальной композиции с этим сигналом.

- Дисторшн (внесение нелинейных искажений, ограничение амплитуды аудиосигнала для обогащения тембра).

- Исполнение заданной мелодии произвольным "голосом" (можно заставить петь любой звучащий объект — от скрипящей двери до плачущего младенца).

- Внедрение в записанную фонограмму колебаний, синхронизирующихся с ритмами активности головного мозга. Эти колебания практически незаметны на слух, однако воздействуют на подсознание человека, усиливая эмоциональную реакцию на обработанную таким образом музыку (см. "Магия ПК" № 10, 1999). Иными словами, для того чтобы у вас появилась толпа поклонников, ревушая от восторга, не обязательно писать гениальную музыку. Достаточно подобрать колебание, совпадающее с подходящей волной активности мозга.

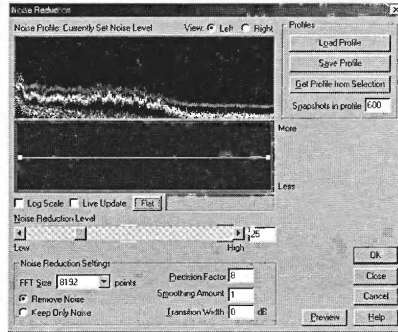
### **Рай в шалаше**

Среди алгоритмов шумоподавления отметим самый, на наш взгляд, интересный из них, основанный на обучении по шумовой выборке. Незаменимая для домашней студии вещь. Раньше или позже от музыкальных композиций вы перейдете к песням и обязательно станете с помощью микрофона записывать вокал. А в этом деле главное — тишина в студии (в комнате). Но, увы, тишина дорого стоит, значительно дороже самого крутого компьютера и самого фирменного микрофона. Дело в

том, что для достижения студийной тишины нужна не просто большая комната, а помещение с двойными стенами, полами и потолками, сделанными из дорогих материалов и подвешенными особым образом на специальных амортизаторах.

Если же вы ведете запись с микрофона, рассчитывая на ее обработку в Cool Edit Pro, то достаточно будет самых элементарных мер снижения уровня шума помещения: закройте окно шторами, отгородите микрофон от компьютера занавеской из плотной ткани (сойдет покрывало). Можно задрапировать также дверь в комнату и полированные поверхности мебели. Желательно использовать микрофон с кардиоидной диаграммой направленности. "Ноль" диаграммы нужно ориентировать в направлении наиболее сильного источника шума.

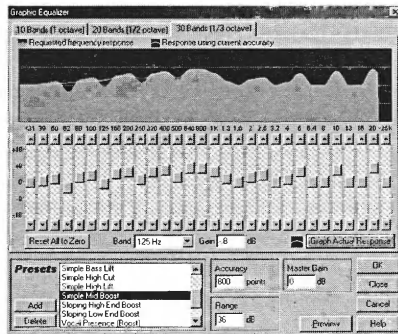
Конечно, если за стеной работает электродрель, то получить качественную запись не удастся. А вот с менее интенсивными шумовыми помехами Cool Edit Pro справится. Причем ничего особенного вам делать не придется. Просто включите запись на несколько секунд раньше, чем нужно будет запеть или заговорить. После завершения сеанса записи выделите этот фрагмент, не содержащий полезного сигнала, и откройте окно Noise Reduction. Нажав кнопку Get Profile from Selection, вы приведете в действие алгоритм анализа параметров шума. Вскоре в окне появятся разноцветные графики спектра шума и амплитудно-частотной характеристики фильтра, синтезированного программой. Результат можно сохранить в файле и использовать потом для обработки записей, выполненных в идентичных акустических условиях. Но это в будущем, а сейчас нужно вернуться в главное окно, выделить всю фонограмму и вновь открыть окно Noise Reduction. Движковым регулятором вы можете выбрать степень подавления шума. При полном подавлении шум в паузах исчезнет совершенно, но голос певца может приобрести "роботоподобные" оттенки. Как правило, оптимальная степень подавления шума составляет 50—70%.



Окно анализа и подавления шума

### Спектры бывают разные

В Cool Edit Pro широко используются спектральные представления и спектральные преобразования аудиосигналов. В качестве иллюстрации мы выбрали окно виртуального графического эквалайзера.

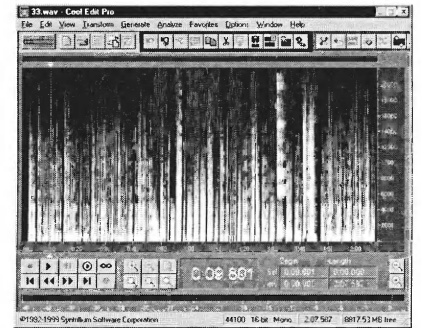


30-полосный графический эквалайзер

На самом деле значительно более тонкие, ювелирные частотно-временные преобразования сигнала можно выполнить с помощью параметрического эквалайзера, аналитических фильтров, а также фильтров, основанных на быстром преобразовании Фурье. В программе имеется два принципиально разных средства анализа спектра. Окно Analysis позволяет только наблюдать спектр. Здесь при вычислении спектра используется классический алгоритм Фурье, дающий характеристику сигнала, осредненную за время его существования.

Другое дело — главное окно программы в режиме Spectral View. В

нем для каждой временной позиции грациями яркости отображается мгновенный спектр. Такое представление сигнала позволяет легко обнаружить различные аномалии, вызывающие скачки ширины мгновенного спектра. Мало того, найденные щелчки или места некорректной стыковки фрагментов можно отредактировать (например, удалить), работая непосредственно в режиме Spectral View.

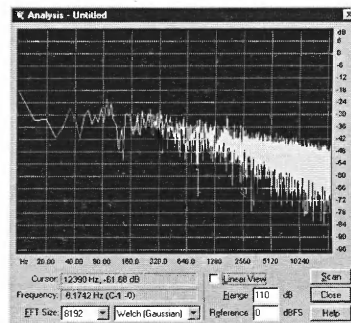


Главное окно программы в режиме Spectral View

Теперь вы, наверное, убедились в том, что для работы с программой действительно нужно иметь некоторые представления о теории сигналов. Их можно почерпнуть из нашей последней на данный момент книги "Музыкальный компьютер. Секреты мастерства". В ней с минимальным количеством формул излагаются основные и принципиально необходимые положения теории. Кстати, там же детально описывается работа с Cool Edit Pro. Полное содержание книги вы найдете на сайте [www.musicalpc.com/](http://www.musicalpc.com/).

### К сведению консерваторов

Из опыта общения с компьютерными музыкантами мы знаем, что существуют две категории пользователей, придерживающихся крайних позиций в отношении освоения программных продуктов. Одни находятся в непрерывном поиске, расходуя почти все свое время на апробирова-



Окно Analysis

ние новинок. Для применения программ по назначению и творчества у таких людей почти не остается времени. Гонка за новым софтом превращается в самоцель. Редкие из них создают что-либо свое. Как правило, такие люди становятся ценителями или критиками чужих работ.

Но встречаются и представители другой точки зрения. Для них однажды освоенный инструмент остается лучшим, если он достаточен для достижения поставленных целей. В эпоху Cubase VST и Sonar можно привести не один пример поклонения программам Sakewalk 3.0 и Midisoft Studio. Оптимальная стратегия следования за прогрессом находится, наверное, где-то посередине. Для тех, кто все еще продолжает применять Cool Edit 96 и не видит смысла в переходе к работе с профессиональной версией любимой программы, отметим, что мультитрековое редактирование — не единственная новинка Cool Edit Pro. В сравнении с Cool Edit 96 появились новые принципиально важные возможности, некоторые функции реализованы более удобно, алгоритмы и интерфейс оптимизированы. Перечислим основные новинки программы:

- Новый эффект Full Reverb, позволяющий строить детальную акустическую модель помещения с учетом свойств среды распространения звука, поглощающих и отражающих свойств материалов, из которых сделаны пол, потолок, стены, их амплитудно-частотных характеристик.
- Новый эффект Hard Limiter, реализующий жесткое ограничение амплитуды аудиосигнала, что в ряде случаев позволяет увеличить субъективную громкость звучания композиции.
- Новый эффект Pitch Bender, с помощью которого можно сместить по высоте как целую партию, так и ее отдельные фрагменты, подправить звучание нот, неверно взятых певцом.
- Новый фильтр DTMF/Notch Filter, предназначенный для удаления из фонограммы не только звуков, сопровождающих работу телефона с тональным набором номера,

но и любых узкополосных помех. Например, можно удалить наводку от сети питания с частотой 50 Гц и ряд связанных с этой частотой гармоник (в общей сложности до 6 частот), совершенно не затронув полезные составляющие звука на частотах 49 Гц и 51 Гц.

- Больше количество эффектов можно использовать в режиме предварительного прослушивания.
- Появилась возможность автоматически обнаруживать и удалять "тишину" между словами или отдельными звуками. Это позволяет, например, не воздействуя на звуки переднего плана, очищать фонограммы от посторонних звуков и шумов.
- Расширились возможности функции автоматического определения и разметки границ фраз в записях речи, границ тактов и долей в записях музыки (функция Auto-Cue).
- Наряду с графическими средствами управления динамической обработкой появились новые (хорошо забытые традиционные) средства, основанные на вводе числовых значений параметров, более привычные для звукооператоров с большим стажем работы.
- Улучшены алгоритмы реализации эффектов Chorus, Flanger, Sweeping Phaser, Brainwave Synchronizer, Clip Restoration, добавлены новые пресеты.
- Теперь вы можете записывать ваши любимые радиопередачи в автоматическом режиме, указав время старта и продолжительность сеанса записи.
- Подготовлены все необходимые предустановки для частот дискретизации до 192 кГц. Разработчики утверждают, что Cool Edit Pro может работать даже и с частотами дискретизации порядка 10 МГц, лишь бы были аппаратные средства, способные на это!

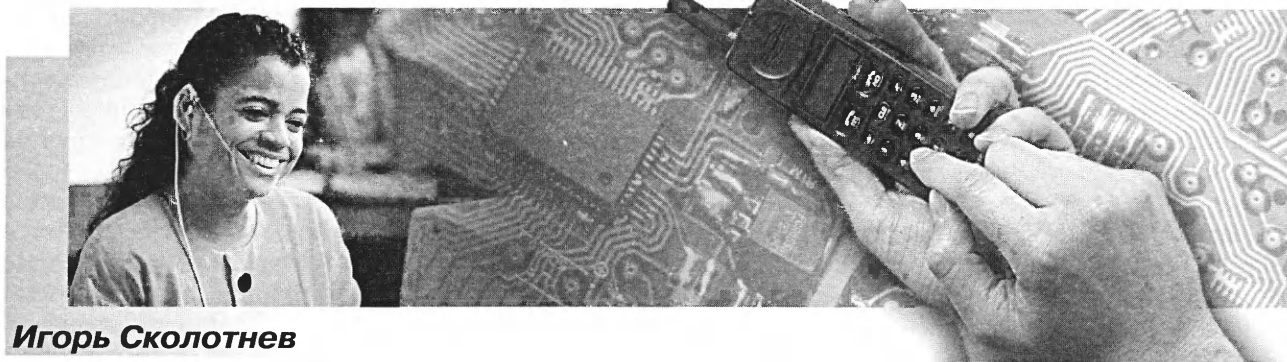
Обо всем интересном, что воплощено в Cool Edit Pro, в краткой статье рассказать, конечно, невозможно, да мы и не ставили перед собой такой цели. Хотелось одного — чтобы вы почувствовали магию программы и со временем взяли ее на вооружение.

**И**стория развития Bluetooth насчитывает уже около семи лет, началась она в 1994 году. Именно тогда инженеры исследовательского центра компании Ericsson в городе Лунд (Швеция) начали заниматься решением проблем, связанных с организацией соединенной между сотовым телефоном и периферийными устройствами. В результате была предложена технология маломощного и дешевого радиointерфейса, которая выдвигалась в качестве альтернативы традиционным соединительным кабелям.

В начале 1998 года по инициативе компании Ericsson и с участием IBM, Intel, Nokia и Toshiba был образован консорциум, а 20 мая того же года союзники объявили о формировании специальной рабочей группы для разработки, унификации и продвижения технологии Bluetooth на рынок. Войти в эту группу может любая компания, которая планирует разрабатывать устройства и ПО на основе спецификаций Bluetooth. Просто подписав соответствующее соглашение, компания становится членом сообщества Bluetooth и получает бесплатную лицензию на производство изделий, а также доступ к спецификациям Bluetooth. В результате такой открытой политики в последующие годы состав группы непрерывно увеличивался, и сейчас в нее входят уже более 2000 компаний, включая таких гигантов компьютерного и телекоммуникационного рынка, как 3Com, Compaq, Dell, Lucent Technologies, Microsoft, Motorola, Qualcomm, а также многие компании из таких, например, отраслей, как автомобильная промышленность, автоматизация производства, индустрия бытовой электроники и домашней техники и др.

Факт объединения столь большого числа компаний уже является уникальным и позволяет говорить о технологии беспроводной связи Bluetooth как о самом быстроразвивающемся коммуникационном стандарте де-факто.

В декабре 1999 года было опубликовано первое описание новой технологии, причем сразу объявля-



**Игорь Сколотнев**

# Bluetooth, многоликий и многообещающий

*Радиоинтерфейс обмена данными Bluetooth, по замыслу создателей, скоро объединит самые различные устройства: от персональных компьютеров и мобильных телефонов до бытовых приборов.*

но, что созданный вариант стандарта Bluetooth 1.0 не является окончательным, и финальная спецификация должна выйти только через три года после дополнительных проверок и доработок.

Специально уполномоченные центры проводят контроль соблюдения требований стандартов членами группы путем проверки соответствия параметров Bluetooth-интерфейсов различных производителей и тестирования их фактической совместимости с другими устройствами.

## **Характеристики Bluetooth**

Итак, технология беспроводной связи Bluetooth представляет собой недорогой радиоинтерфейс с низким энергопотреблением (мощность передатчика всего порядка 1 мВт) для организации персональных сетей (Personal Area Networking, PAN), обеспечивающий передачу в режиме реального времени как циф-

ровых данных, так и звуковых сигналов. Другими словами, Bluetooth позволяет связывать друг с другом любые электронные устройства с помощью небольших приемопередатчиков, либо встроенных в эти устройства, либо подключаемых к ним через свободный порт или PC-карту, что избавляет от необходимости применять для этих целей соединительные кабели.

Изначально дальность действия радиоинтерфейса закладывалась равной 10 метрам (примерно в границах одной комнаты), однако сейчас спецификация Bluetooth определяет уже и вторую зону — около 100 м (для покрытия стандартного дома). При этом нет необходимости в том, чтобы соединяемые устройства находились в зоне прямой видимости. Их даже могут разделять "радиопрозрачные" препятствия (стены, мебель и т. п.). К тому же приборы могут находиться в движении.

Для работы радиоинтерфейса Bluetooth используется так называемый нижний (2,45 ГГц) диапазон ISM (Industrial, Scientific, Medical), предназначенный для работы промышленных, научных и медицинских приборов. Особенность данного диапазона в том, что почти во всех странах мира, включая Россию, он свободен от лицензирования: для использования сертифицированного передаю-

щего оборудования, работающего на этих частотах, не требуется дополнительного разрешения.

Радиоканал обладает полной пропускной способностью 1 Мбит/с, что обеспечивает создание асимметричного канала передачи данных на скоростях 723,3/57,6 Кбит/с или полнодуплексного канала на скорости 433,9 Кбит/с. Через Bluetooth-соединение можно передавать до 3 дуплексных аудиоканалов по 64 Кбит/с в каждом направлении. Возможна также и комбинированная передача данных и звука.

В части организации обмена данными Bluetooth соответствует спецификации стандарта локальных сетей IEEE 802 и использует сигналы с расширением спектра путем скачкообразной перестройки частоты (frequency-hopping spread-spectrum, FHSS) по псевдослучайному закону со скоростью 1600 переключений в секунду в полосе 2400—2483,5 МГц.

Выбор подобного режима работы радиоканала был сделан не случайно — он позволяет решить сразу несколько проблем, и прежде всего это относится к взаимным помехам с другими устройствами. Дело в том, что в диапазоне частот, выбранном для Bluetooth, уже работает большое количество самых разных промышленных, медицинских и бытовых устройств, включая микроволновые



печи, автомобильные сигнализации и устройства для открывания дверей гаража, коммерческие системы передачи данных и беспроводные локальные сети в стандарте IEEE 802.11. Поскольку использование всех этих устройств разрешено на безлицензионной основе (а значит, и без выполнения соответствующего частотного и территориального планирования их размещения), то выбор любой фиксированной частоты неизбежно привел бы к "частотному конфликту" с другими устройствами и невозможности работы из-за взаимных радиопомех. Благодаря методу быстрой смены частот поражение отдельных частот помехами будет приводить к потерям только небольших фрагментов данных, которые могут быть легко восстановлены путем применения помехозащищенного кодирования. Кроме того, смена частот по псевдослучайному закону снижает влияние интерференционных замираний сигналов за счет преотражений от окружающих предметов, а также затрудняет перехват передаваемой информации.

### Взаимодействие устройств

Bluetooth работает как многоточечный радиоканал, управляемый многоуровневым протоколом, аналогичным сотовому протоколу GSM.

Топология локальной радиосети организована по принципу множественных пикосетей (piconet), взаимодействующих между собой по стандартному радиоканалу. В пикосетях устройства Bluetooth взаимодействуют по методу "ведущий — ведомые". Статус "ведущий" (master) всегда имеет наиболее мощное устройство, устанавливающее соединения с несколькими другими. "Ведущий" координирует посылку и прием данных в рамках образованной пикосети. Число активных, то есть обменивающихся данными "ведомых" (slaves) устройств в пикосети, может достигать до 7. Кроме активных "ведомых" может существовать и множество неактивных устройств, которые не могут обмениваться данными с "ведущим", пока заняты все каналы, но, тем не менее, синхронизированы

с ним. Если в радиусе действия "ведущего" оказывается более 7 активных устройств, формируется вторая пикосеть, управляемая первой, где роль "ведущего" выполняет одно из "ведомых" устройств первой сети. Такое наращивание сети может идти до бесконечности. Множество пикосетей, способных взаимодействовать друг с другом, формирует распределенную сеть (scatternet). В рамках такой сети разные устройства могут быть одновременно "ведущими" и "ведомыми" для разных пикосетей. Более того, при необходимости любой "ведомый" в пикосети всегда может стать "ведущим" (естественно, старый "ведущий" при этом станет одним из "ведомых"). Таким образом, в распределенную сеть можно объединять любое количество устройств Bluetooth, а логические связи — образовывать и менять, как это требуется.

Благодаря работе радиоканала со скачками частоты пикосети могут взаимодействовать друг с другом с минимальным риском взаимных помех. Единственным условием здесь является то, что различные пикосети, входящие в одну распределенную сеть, должны работать на разных частотах и иметь различный порядок смены каналов, определяемый в стандарте параметрами hopping sequence — последовательность скачков. Существующий вариант стандарта предусматривает 10 вариантов этого параметра: 5 с циклом 79 смен частот и 5 — с циклом 23 смены.

Важно то, что соединения в пикосетях осуществляются автоматически, как только различные устройства оказываются в пределах досягаемости. Для этого работа любых Bluetooth-устройств в незнакомом окружении начинается с режима поиска других устройств (device discovery): посылается специальный запрос, ответ на который зависит не только от наличия в радиусе связи других активных Bluetooth-устройств, но и от режима их работы. На этом этапе возможно три основных режима: discoverable mode (устройства в этом режиме всегда отвечают на запросы), limited

discoverable mode (устройства отвечают на запросы только ограниченное время или при соблюдении определенных условий) и non-discoverable mode (устройства не отвечают на новые запросы). Кроме того, все обнаруженные устройства могут быть в non-connectable или в connectable mode. В первом случае найденное устройство не позволяет настроить параметры соединения, и, таким образом, обмен данными с ним невозможен. Во втором случае связавшиеся Bluetooth-устройства "договариваются" между собой о параметрах соединения: используемый диапазон частот и порядок их смены, размер страниц данных и т. п.

По окончании процесса обнаружения новое Bluetooth-устройство получает набор адресов (каждое устройство для идентификации имеет уникальный 48-битный сетевой адрес) и имен всех доступных устройств, после чего происходит определение предоставляемых ими услуг (service discovery).

В качестве мер защиты в Bluetooth предусмотрены кодирование передаваемых данных и процедура авторизации устройств. Возможны три уровня защиты:

- минимальная — данные кодируются общим ключом и могут приниматься любыми устройствами без ограничений;
- защита на уровне устройств — непосредственно в чипе прописывается уровень доступа, в соответствии с которым устройство может получать определенные данные от других устройств;
- защита на уровне сеанса связи — данные кодируются 128-битными случайными числами, хранящимися в каждой паре устройств, участвующих в данном сеансе связи.

### Какие устройства уже есть?

Сами Bluetooth-чипы выпускают уже несколькими десятками компаний. Первое такое изделие создала фирма Ericsson еще в 1999 году, а в октябре 2000 года Matsushita-Kotobuki Electronic Industries разработала весьма компактный Bluetooth-модуль, минимизация раз-

меров которого была достигнута за счет помещения всех компонентов, включая антенну приемопередатчика (!), на единую керамическую подложку. Вообще же на разных стадиях освоения массового производства находятся такие компании, как Atmel, Broadcom, Conexant Systems, Intel, Motorola, National Semiconductor, NEC, Samsung Electro-Mechanics, Silicon Wave и др.

Особая цель разработчиков — максимальное снижение цены чипов, до уровня порядка \$5 за кристалл. Наиболее близко к этому уровню (\$8) подошла английская компания Cambridge Silicon Radio, также объединившая все компоненты Bluetooth в одной стандартной микросхеме.

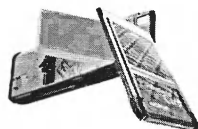


Что касается устройств Bluetooth, то их число и ассортимент сейчас растет буквально с каждым днем. Первое такое изделие — комплект беспроводной связи с сотовым телефоном в виде миниатюрной hands-free гарнитуры, содержащей наушник и микрофон, — был создан компанией Ericsson и продемонстрирован в ноябре 1999 года. А в 2000 году свои версии Bluetooth-наушников для сотовых телефонов создали компании Japan Total Design Communication Co. (модель "i2me", включающая адаптер весом 45 г и наушники весом 35 г) и Plantronics (гарнитура наушник-микрофон весом 26 г).



В это же время появились и первые сотовые телефоны со встроенной платой Bluetooth (Ericsson T36s, Ericsson R520, Siemens S42, Nokia 6310 и др.). По оценкам экспертов, уже в самом ближайшем будущем подобные устройства будут встраиваться в 80% сотовых телефонов. Создаются и специальные адаптеры для телефонов, не содержащих Bluetooth-плату (например, для

Ericsson T28s). Оригинальный вариант подобного адаптера предложила компания Nokia, создав Bluetooth Connectivity Kit для телефона Nokia 6210, представляющий собой комплект из аккумулятора и коммуникационной карты. Здесь приемопередатчик Bluetooth для сотового телефона встроен в аккумулятор специальной модификации, а второй элемент комплекта — коммуникационная карта, похожая на CompactFlash, — имеет PC-адаптер для установки в ноутбук. Благодаря этому комплекту владельцы Nokia 6210 получают возможность пользоваться своим телефоном и как беспроводным GSM-модемом для компьютера.



Вообще же Bluetooth-адаптеры для работы с компьютерами, принтерами и другими устройствами также разрабатываются или выпускаются сейчас уже целым рядом компаний: Axis (Bluetooth-модуль для принтеров), IBM (PC-карта и Bluetooth-модем для органайзеров типа Palm), Motorola (плата формата Mini PCI, поддерживающая как Bluetooth, так и стандарт модемной связи V.90), NEC (коммуникационный модуль для ноутбуков), Siemens (Bluetooth-адаптер, подключаемый к USB-порту настольного ПК или ноутбука, способный работать в радиусе 100 м), Sharp (Bluetooth-модуль в виде CompactFlash-карты), Toshiba (Bluetooth PC Card), TDK (Bluetooth USB Adapter) и др.

Универсальный вариант Bluetooth-адаптера под названием Infostick, внешне напоминающий Memory Stick, в январе 2001 года представила компания Sony. Благодаря ему теперь все устройства Sony, оснащенные слотом Memory Stick, смогут "общаться" между собой без проводов.

Но ассортимент изделий, использующих технологию Bluetooth для обмена данными, отнюдь не ограничивается только наушниками и коммуникационными адаптерами. Например, финская компания Nokian Tyres разработала технологию, позволяющую вести непрерыв-

ное автоматическое наблюдение за состоянием колес автомобиля. Встроенный в шину чип будет измерять параметры колеса и, в случае необходимости, через Bluetooth-модуль свяжется, например, с мобильным телефоном водителя. А вот компания Motorola с использованием технологии Bluetooth создала свой автомобильный hands-free комплект. Это устройство, взаимодействующее с бортовым компьютером машины и обеспечивающее возможность дистанционного управления дверьми автомобиля. Еще один вариант сходной автомобильной системы контроля на основе Bluetooth, подключающейся к внутренней локальной сети автомобиля, разработала японская компания Yazaki. Многие специалисты вообще уверены, что технология Bluetooth благодаря своей простоте и изяществу просто не имеет конкуренции в области создания автомобильных локальных сетей.

Но не только автомобилисты обратили свое внимание на Bluetooth. Так, компания BlueTags предложила встраивать интерфейс Bluetooth... в сумки и чемоданы, либо оснащать им ярлыки, наклеиваемые на багаж при его транспортировке. По мнению компании, такое нововведение существенно упростит жизнь работников отелей, вокзалов и аэропортов, которые смогут автоматически сортировать ручную кладь, а также сократить число потерь вещей.

А вот IBM создала Bluetooth-часы, которые можно не только носить на руке, но и использовать в качестве своеобразной выносной клавиатуры для дистанционного управления компьютером, например, во время различных докладов и презентаций.

Еще одно любопытное устройство под названием Chatpen создали шведские компании Anoto и Ericsson. Данная разработка представляет собой авторучку, позволяющую передавать сделанные ею записи в компьютер или сотовый телефон. Для этого в Chatpen, кроме пишущего узла, содержится миниатюрная видекамера, микропроцессор и Bluetooth-модуль. Процесс отлича-

ется от обычного только тем, что писать надо на специальной бумаге с сеткой из малозаметных точек. Относительно этих точек фиксируются все перемещения ручки по бумаге, а результат после соответствующей обработки пересылается через Bluetooth-интерфейс в расположенные поблизости компьютеры или на мобильный телефон для дальнейшей передачи. По замыслу разработчиков, Chatpen может использоваться для передачи "электронных" вариантов традиционной подписи или, например, при заказе товаров по печатному каталогу: пользователю в этом случае достаточно будет лишь "указать" ручкой нужный товар.

В другом направлении работают специалисты фирмы Philips, создающие телеприемник с беспроводными колонками пространственного звучания, подключаемыми по технологии Bluetooth. И это, похоже, только первая "ласточка". Дело в том, что различные исследовательские группы уже долгое время ищут простую и максимально универсальную технологию объединения многообразной домашней техники в общую сеть. Особую привлекательность здесь имеет именно технология Bluetooth, позволяющая максимально просто соединять друг с другом практически любые устройства (от компьютера, цифрового фотоаппарата и телефона до холодильника, микроволновой печи, кондиционера и т. п.), имеющие встроенный микрочип Bluetooth. Например, беспроводная сеть на базе Bluetooth уже развернута американской компанией Red-M в клинике немецкого города Майнца. Эта сеть позволяет врачам оперативно вводить информацию о пациентах в базу данных клиники, используя карманные компьютеры.

Но, как ни разнообразны существующие устройства с Bluetooth-модулями, будущее этой технологии представляется еще более удивительным.

### Перспективы

Похоже, что у Bluetooth они действительно огромны. И причин здесь сразу несколько: совершенство со-

зданной технологии, ожидаемая низкая стоимость устройств и заложенная в технологию простота их использования, отсутствие лицензионных ограничений, широкий диапазон потенциальных областей применения, поддержка подавляющим большинством ведущих производителей. По имеющимся оценкам, объемы продаж всего через 4—5 лет могут дойти до миллиарда Bluetooth-устройств в год.

Уже сейчас начинают выпускаться принтеры с Bluetooth, которые не надо подключать к компьютеру. Достаточно просто внести его в комнату, включить питание, и можно начинать печатать. Появляются и беспроводные мыши, клавиатуры, джойстики и т. п., способные начать работу с любым компьютером, достаточно только к нему подойти. Но главное, технология Bluetooth может обеспечить пользователям не просто возможность отказаться от жгутов соединительных кабелей у себя дома или в офисе, а открыть множество совершенно новых возможностей.

Вот только некоторые из фантастических картин завтрашних перспектив этой технологии, нарисованные разработчиками.

Для начала — совсем простая "картинка": триединый мобильный телефон. Когда его владелец находится на работе, он работает по каналу Bluetooth и обеспечивает полноценную связь через офисную мини-АТС. Аналогично он работает и дома — через домашний стационарный телефон, а в дороге он переходит в режим сотовой связи, обеспечивая в итоге своему владельцу заметную экономию расходов на мобильную связь.

А вот — "Wireless Wallet" (Беспроводной Кошелек). Здесь принцип действия тоже достаточно прост. В кошелек вмонтирован чип Bluetooth, который взаимодействует с вашими смарт-картами, мобильным телефоном, карманным (портативным, настольным) компьютером, кассовыми аппаратами и т. д. Кошельку соответствует сайт, на котором хранится и постоянно обновляется информация о всех картах, находящихся в кошельке. Когда вы что-то покупаете,

вам не надо доставать кредитку из кошелька: ваш Bluetooth-коммуникатор покажет вам сайт вашего кошелька, где вы сможете выбрать карту, подходящую для оплаты (например, кредитку). Карта связывается с кошельком через Интернет, и тот отправляет платеж по назначению. В случае с билетом он записывается на смарт-карту и автоматически считывается Bluetooth-устройствами на вокзале, в аэропорту, театре.

А теперь другой сценарий: вы сидите на совещании, в портфеле у вас сотовый телефон, а на столе перед вами ноутбук. Не соединяя никаких разъемов, вы непрерывно можете получать электронные сообщения из любой части мира (для этого ваш ноутбук связывается с сотовым телефоном, а тот — с Интернетом) и данные от других участников совещания (ноутбук напрямую обменивается информацией с портативными компьютерами сидящих за столом). Кроме того, в конференц-зале стоят принтеры, поддерживающие Bluetooth, и поэтому пользователи всегда могут сразу распечатать необходимый документ. И все это работает без какой-либо предварительной настройки: программа обнаружения Bluetooth-сервисов узнала о наличии принтеров и ноутбуков других сотрудников в помещении, как только вы в него вошли, и сама установила с ними контакты.

Не хуже обстоят дела и в вашем "цифровом доме". У вас есть брелок или браслет с Bluetooth-устройством, способным принимать ваш голос и передавать его на центральный компьютер в доме. Мощный вычислитель идентифицирует голос, распознает звуковую команду и тоже по Bluetooth-интерфейсу передает ее необходимому домашнему устройству. Вы сможете просто голосом управлять яркостью лампочек, работой телевизора, шторами, кондиционером и всем прочим — от входного замка до крана в ванной!

Подобные примеры можно продолжать еще долго, но главное — по мнению специалистов, все это станет реальностью уже в ближайшие пять лет!



Алексей Гинзбург

## Планшетные сканеры: ПЗС против КДИ

Сегодня сканер становится одним из тех периферийных устройств, без которых не может обойтись практически ни один офис, а частью и домашний пользователь. И самым распространенным видом сканеров является планшетный. Популярность эта вполне заслуженна: устройство таких сканеров дает все удобства при сканировании любых оригиналов. Оригинал неподвижно лежит на стекле, а считывание в большинстве случаев происходит в отраженном от него свете. Высокие скоростные характеристики таких сканеров также являются несомненным преимуществом. Достигается оно за счет того, что фотоэлементом в планшетных сканерах является не единственный фотоэлемент, а считывающая линейка фотоэлементов.

На рисунке 1 изображена схема устройства обычного планшетного ПЗС-сканера. Полоса света от источника освещения попадает на оригинал. Отразившись, свет попадает в систему зеркал. Зеркала расположены так, чтобы отраженный свет попадал на собирающую линзу. Линза проецирует свет на линейку фотоэлементов (с увеличением). В фотоэлементах свет трансформируется в аналоговый сигнал, который далее попадает в аналогово-цифро-

вой преобразователь (АЦП). В некоторых сканерах между фотоприемником и АЦП находятся промежуточные ступени, работающие с аналоговым сигналом. Они предназна-

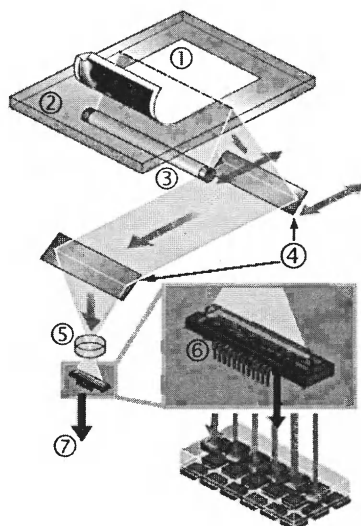


Рис. 1. Устройство планшетного сканера: 1 — оригинал; 2 — стекло; 3 — источник света; 4 — система зеркал; 5 — линза; 6 — линейка фотоэлементов; 7 — АЦП

ны для аппаратного исправления погрешностей сканирования и, иногда, самого изображения. В результате на выход, то есть в компьютер (после АЦП) идет полоска изображения исходного оригинала. Такая процедура сканирования охватывает только одну строку изображения. Поэтому

для полного сканирования и используется головка. После того как отсканированная строка пикселей попадет в компьютер, каретка сдвигается на один шаг (величина шага фиксирована и от нее зависит механическое разрешение сканера). После этого вся процедура повторяется до тех пор, пока заданная область не будет считана полностью.

Самой существенной деталью сканера — фотопринимающая матрица. От нее зависят оптическое разрешение (этот параметр дает оценку минимальной детали изображения, воспринимаемой сканером), динамический диапазон (грубо говоря, диапазон воспринимаемых сканером оттенков цвета) и почти все остальные характеристики (за исключением разве что рабочей области сканера). На сегодня наиболее распространены два типа фотопринимающей матрицы:

- ПЗС-матрицы (прибор с зарядовой связью — CCD, Couple-Charged Device);
- КДИ-матрицы (контактный датчик изображения, в английских обозначениях — CIS, Contact Image Sensor).

Технология использования КДИ-фотоприемника в планшетных сканерах, изначально разработанная фирмой Canon (LIDE — LED InDirect Exposure, не прямое светодиодное

экспонирование) была взята на вооружение другими фирмами, а сканеры, работающие на ее основе, стали называться КДИ-сканерами.

КДИ-матрицы применяются уже долгое время в устройствах типа факса. Однако в сканерах их начали использовать совсем недавно. Первые серийные планшетные сканеры, основанные на этой технологии, были выпущены в 1998 году.

Принципиального различия между КДИ- и ПЗС-матрицами нет. КДИ-сканеры отличаются от ПЗС-сканеров тем, что в них матрица растянута на всю ширину рабочей области, поэтому в них отсутствует оптическая система.

Однако от технологии фотопринимающей матрицы зависит устройство многих других узлов, так что следует говорить не о различиях в сканирующей матрице, а о различиях в сканерах.

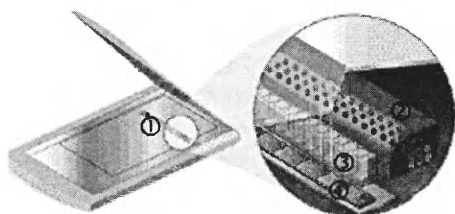


Рис. 2. Устройство КДИ-сканера: 1 — сканирующая каретка; 2 — источник освещения (светодиоды); 3 — линзы; 4 — КДИ-матрица

В КДИ-сканерах источник освещения заменяется светодиодами (рис. 2). При этом для цветного сканирования возникает необходимость в трех светодиодах на пиксел, в соответствии со стандартным разложением цвета RGB (Red, Green, Blue). Любой другой цвет можно представить как наложение в некоторой пропорции этих основных цветов. Каждая составляющая будет аналогична градациям серого цвета, поэтому фотоэлементы в состоянии их воспринимать.

Зеркала и объектив в КДИ-сканерах отсутствуют, так как эта технология обеспечивает прямую проекцию полной поверхности рабочей области прямо на считывающую матрицу.

Во многих профессиональных

планшетных ПЗС-сканерах установлен не один, а несколько объективов, что позволяет переключаться с одного на другой и, таким образом, значительно повысить оптическое разрешение, хотя при этом уменьшается рабочая область. Реализована эта возможность следующим образом: при сканировании полной рабочей области объектив проецирует ее полную ширину на полную длину ПЗС-линейки. Например, если при фокусировке один дюйм проецируется на 600 элементов ПЗС-линейки, то разрешение будет 600 ppi (пикселей на дюйм). Если на те же 600 элементов с помощью другого объектива сфокусировать полдюйма, то разрешение увеличится вдвое — до 1200 ppi. Такая конструкция дает очень высокое оптическое разрешение, однако при максимальном разрешении рабочая область становится очень малой. Например, модель ScanMate F8 Plus с разрешением 4000 ppi дает возможность сканировать оригинал, ширина которого не больше 54,9 мм. Для этой же модели разрешение на полной рабочей области (279,4x430,8) — 700 ppi.

Линза ПЗС- и КДИ-сканерах играет несколько разные роли. В ПЗС-сканерах свет, идущий от оригинала, попадает с помощью зеркал на линзу-объектив (или прямо на призму), которая проецирует ширину рабочей области сканера на ширину ПЗС-матрицы. При этом к качеству и точности настройки линзы (призмы) предъявляются весьма высокие требования, особенно при проецировании краев рабочей области сканера. В наиболее мощных моделях сканеров объективов два и более, причем режимы их использования можно переключать. Помимо обычного режима в них доступны усиленные режимы, когда один из дополнительных объективов проецирует на полную поверхность ПЗС-матрицы только часть полной рабочей поверхности сканера. Таким образом, для изображения меньшего размера используется такое же количество элементов ПЗС-матрицы, как и для полного, то есть повышается качество сканирования (оптическое разре-

шение). В КДИ-сканерах используется не одна линза, а много — по количеству элементов линейки фотоприемника. Каждая из них играет роль маленького проектора, который фокусирует попадающий свет на соответствующий фотоэлемент.

Главной деталью КДИ-сканера, как и любого планшетного, является сканирующая головка. Сканирующая головка состоит из источников света (светодиодов), цилиндрической линзы (набора линз) и печатной платы, на которой находятся чувстви-

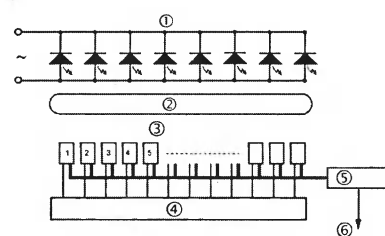


Рис. 4. Устройство сканирующей головки КДИ-сканера: 1 — набор светодиодов; 2 — линзы; 3 — фотоэлементы; 4 — регистр сдвига; 5 — видеоусилитель; 6 — выход (АЦП)

тельные элементы (датчики), видеоусилитель, регистры и буферы.

Излучение светодиодов отражается от оригинала и, пройдя через линзу, фокусируется на датчике изображения (фототранзисторе). В результате получается аналоговый сигнал, который усиливается на видеоусилителе и идет в АЦП.

Датчик изображения обычно состоит не из одной, а из нескольких линеек, последовательно соединенных в одну. Отсутствие оптической системы налагает свои ограничения на такую технологию. Если, например, полный датчик изображения длиной 216 мм (формат А4) состоит из 27 меньших датчиков, каждый из которых имеет 96 светочувствительных элементов, то в результате получится 12 элементов на миллиметр, или 300 элементов на дюйм.

Для сканирования полноцветного изображения используются три светодиода на один элемент датчика: красный, зеленый и синий, которые при сканировании включаются по очереди. Правда, в некоторых моделях сканеров встречаются го-

ловки, выполняющие полноцветное считывание за один раз. Достигается это за счет использования специальных светофильтров на интегрированных схемах. В качестве источника освещения в данном случае используются не светодиоды, а лампы с холодным катодом.

Если сравнить основные характеристики, результат явно оказывается в пользу ПЗС-технологий. Однако и у КДИ-сканеров есть свои преимущества. Среди положительных сторон этих сканеров необходимо отметить следующие.

- Меньшие габариты. КДИ-сканеры имеют меньшие размеры и вес, чем сканеры на основе ПЗС;
- Меньшая стоимость. Вместо объектива, зеркал, призмы и самого фотоэлемента в этих сканерах используется только КДИ-линейка, что позволяет значительно снизить стоимость сканеров такого типа.
- Меньшая потребляемая мощность. Это достигается за счет применения светодиодов вместо лампы с холодным катодом. Если для ПЗС-сканера нормальная потребляемая мощность 12 Вт, то для КДИ-сканера — 2,5 Вт.
- Равномерность качества. Для ПЗС-сканеров из-за недостатков фокусировки оптической системы качество сканирования может существенно различаться в центре рабочей области и по краям. Этот эффект неизбежен для любых приборов, в которых используется объектив. В КДИ-сканерах качество сканирова-

ния абсолютно равномерно, так как оптическая система отсутствует.

- Работа в экстремальных условиях. КДИ-сканеры гораздо менее чувствительны к внешним условиям. Сканер с подставкой может работать даже в вертикальном положении.



Рис. 6. Отсканированный КДИ-сканером тубик диаметром 3 см

ПЗС-сканеры явно выигрывают в отношении качества изображения:

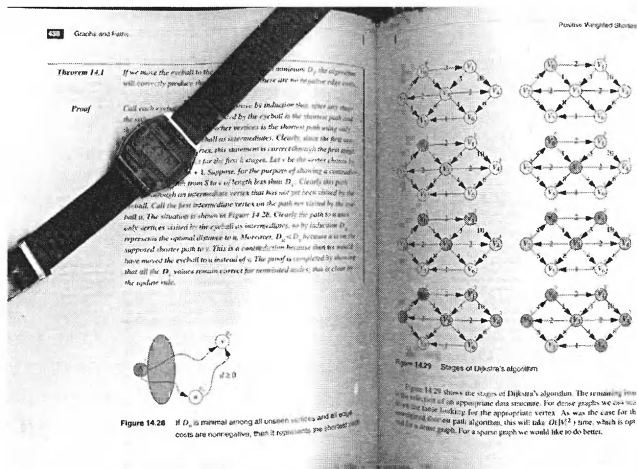
- Лучшая глубина резкости. Глубина резкости КДИ-сканеров 0,3 мм, тогда как для сканеров с ПЗС она равна 3 мм. Это означает, что трехмерные предметы, находящиеся на расстоянии 3 мм от общего уровня, будут нормально отсканированы ПЗС-сканером, а изображение, полученное КДИ-сканером, будет нерезким и размытым.
- Долше срок службы. ПЗС-сканер обеспечивает стабильное и неизменное качество в течение 10000 часов работы, тогда как у КДИ-сканеров после 500 часов происходит падение яркости на величину до 30%.
- Лучшая чувствительность к оттенкам. ПЗС-сканеры различают уровни оттенков с погрешностью 20%, в то время как КДИ-сканеры — 40%. Соответственно, передача деталей у ПЗС-сканеров будет значительно лучше.

- Меньшая чувствительность к посторонней засветке. Это преимущество связано с тем, что ПЗС-линейка невелика по длине, и благодаря системе зеркал “лишнего” света на нее не проецируется. В КДИ-сканерах линейка значительно больше, оптическая система практически отсутствует, поэтому любое дополнительное освещение значительно влияет на результат сканирования.

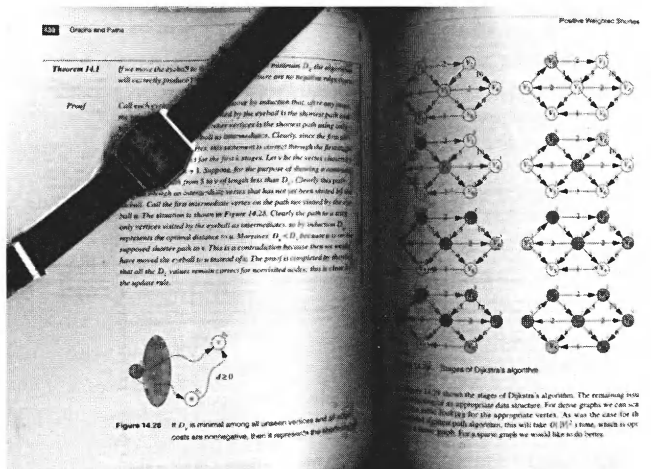
- Разрешение сканера. Максимальное разрешение профессиональных ПЗС-сканеров на данный момент 5000 ppi, тогда как для КДИ-сканеров верхний предел — 600 ppi.

Для наглядного сравнения качества сканирования приведу результаты работы КДИ- и ПЗС-сканера (рис. 7 и 8). На рабочую поверхность сканеров были помещены часы и книга, при этом прижим обеспечивался только крышками сканеров. Сканирование производилось с разрешением 300 ppi и с использованием установок сканирования, принятых по умолчанию. Как видно из иллюстраций, общая четкость изображения намного выше у картинке, отсканированной ПЗС-сканером. На развороте книги (прижим здесь менее эффективен) при сканировании КДИ-сканером изображение получилось значительно темнее, а в некоторых местах и вообще черное, чего нет при сканировании ПЗС-сканером.

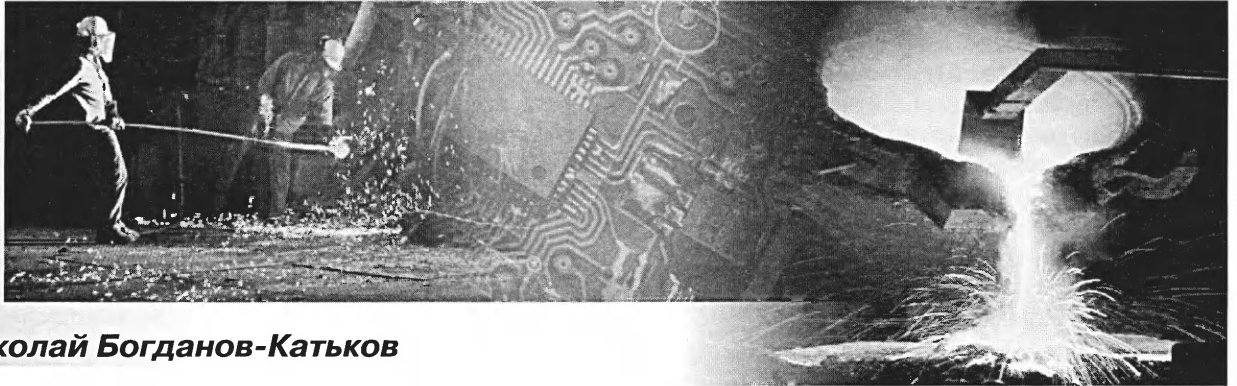
Из всего этого следует, что КДИ-сканеры пока еще не достигли уровня качества ПЗС-сканеров.



Отсканировано ПЗС-сканером



Отсканировано КДИ-сканером



Николай Богданов-Катков

# Мир олову, война свинцу!

*Охрана природы — штука выгодная... для тех, кто ее охраняет.*

**Н**едavno в новостных каналах Интернета появилось примечательное сообщение. Ведущие европейские производители электронных компонентов решили всемерно сокращать использование оловянно-свинцовых припоев. Несколько фирм предполагают к концу года выработать единый стандарт на бессвинцовые припои и надеются, что их разработка будет принята в качестве стандарта для всех стран—членов ЕС. Кстати, начиная с 1 января 2006 года в Европе будет полностью запрещено использование свинца, кадмия, ртути и некоторых других металлов при производстве электронного оборудования. А в мягких (легкоплавких) припоях, которые широко используются при монтаже печатных схем, разрешается применять олово, висмут, медь, цинк, иридий.

Последнее утверждение оставим на совести составителей сводок новостей. Разумеется, имеется в виду не иридий (очень твердый тугоплавкий металл), а легкоплавкий индий, до сих пор находивший лишь очень ограниченное применение в технологии пайки.

Казалось бы, чем реже применяются токсичные вещества, тем лучше. Во всяком случае экологические движения ряда стран выступают за

максимальное ужесточение всех мыслимых норм безопасности, за исключение контакта человека со всем, что может представлять хоть какой-то риск для его здоровья.

Совсем недавно "зеленые" начали придираться к Hewlett-Packard. Дескать, новые чернила для струйников пахнут каким-то органическим растворителем, а из тонера для лазерных принтеров выделяется "страшный канцероген" бензол.

Но тут ситуация иная. Зачинателями новой экологической инициативы выступают не экологические организации, не официальные органы Европейского Сообщества, а сами фирмы-производители. Представьте себе, что директора химических предприятий по доброй воле предлагают всемерно ужесточить нормативы загрязнения окружающей среды!..

## **+\$20. За что?**

Лейбл TCO'99 во всем мире считают надежнейшей гарантией безопасности. Мониторы, стандартизованные по этому нормативу, стоят на 15—25 долларов дороже, чем соответствующие более ранним стандартам TCO'95 и TCO'92. Тем не менее их охотно покупают. Два года назад, как только в продаже появились первые модели ново-

го стандарта, некоторые западные фирмы, заботясь о здоровье персонала, сменили весь свой парк мониторов.

А что, собственно, дает новый стандарт рядовому пользователю? Чрезвычайно низкий допустимый уровень электромагнитного излучения зафиксирован еще в стандарте TCO'92. Стандарт TCO'95 предписал измерять излучение не только перед экраном, но и вокруг всего прибора (если в офисе находятся несколько мониторов, то на каждого работающего оказывает влияние не только тот, за которым он сидит, но и соседние).

В стандарте TCO'99 добавлены требования эргономичности и экологической чистоты. С первым все ясно, любой прибор должен быть удобен, работа с ним не должна приводить к быстрому утомлению зрения, рук и т. п. Хотя и прежние мониторы особых проблем не вызывали, все же четкие нормы не помешают.

Но что такое экологическая чистота? Ни один монитор, ни один компьютер при работе не выделяет в воздух пары ртути или, скажем, иприта. Имеются в виду прежде всего требования к производству.

Чтобы аттестовать монитор на соответствие TCO'99, производитель должен доказать, что не только

сам монитор хорош по всем параметрам, но и завод экологически чистый, и рабочие получают витаминизированные обеды...

### Ложка дегтя в окружающую среду

Но это еще не все. Рано или поздно монитор отслужит свой срок и его придется выбросить или отправить на переработку. В первом случае важно, чтобы конструкционные материалы, компоненты печатных плат и электронные устройства не содержали веществ, способных оказать вредное воздействие на окружающую среду. Для переработки важнее другое: легкость разборки, сортировки материалов и самой переработки. Все это оговорено в стандарте.

Поскольку во многих странах предпочитают сжигать отходы, а не перерабатывать, стандарт запрещает использование пластмасс, содержащих хлор. При их сгорании образуются одни из самых опасных веществ — полихлордиоксины и полихлордибензофураны. Образуются они в ничтожных количествах и только при неправильном режиме сжигания, но все же...

Точно так же ни на свалку, ни в атмосферу не должны попадать тяжелые металлы. Но тут все определяется техническими возможностями. Пока еще никто не придумал, как обойтись без меди и цинка.

Многие детали для увеличения коррозионной стойкости покрывают никелем, хромом или кадмием. Впрочем, последний попадает под запрет — его токсичность очень высока. Однако кадмиевые покрытия наиболее стойки именно в тропическом климате, то есть уместнее всего как раз там, где расположено большинство современных электронных производств — в Юго-Восточной Азии.

### Из пушки по воробьям

Можно ли ожидать, что все принимаемые меры дадут ощутимый результат? Нет, по крайней мере, запрет на оловянно-свинцовые припои ничего не изменит. Большая

часть свинца используется не в припоях, а для производства свинцовых кислотных аккумуляторов, применяемых почти во всех автомобилях и многих где еще. По экономическим соображениям крупные аккумуляторные батареи могут быть изготовлены только из относительно дешевых материалов, а нетоксичных среди них нет.

Из всех типов щелочных аккумуляторов реальную альтернативу кислотным представляют никель-кадмиевые и никель-железные. Кадмий несравненно токсичнее свинца, но и никель занимает одно из первых мест по канцерогенности. К тому же никелевые аккумуляторы значительно дороже свинцовых.

Еще дороже современные высокоэнергетичные источники тока, литий-ионные и литий-полимерные. Те и другие находят применение лишь в дорогостоящих устройствах — ноутбуках, мобильных телефонах.



После нескольких тысяч циклов заряд-разряд аккумуляторы приходят в негодность. По данным международных экологических организаций, именно так на свалки попадает до 80% свинца и 30—35% никеля. Отработавшие свой срок электронные устройства дают не более 5% отходов тяжелых металлов. Так что запрет на использование оловянно-свинцовых припоев в электронике практически не скажется на состоянии окружающей среды.

То же можно сказать и про все остальное. TCO '99 запрещает применять поливинилхлоридные пластики в мониторах? Но их гораздо шире используют в химической и кабельной промышленности, там их просто нечем заменить. Обычный линолеум, кстати, тоже содержит

хлор и его категорически не рекомендуется сжигать.

Общий итог таков: платя лишние 15—20 долларов за монитор стандарта TCO '99, мы, потребители, оплачиваем хорошие условия труда корейских и тайваньских рабочих, да еще, пожалуй, беспричинные опасения западного обывателя, запуганного Гринписом, а сами ничего не выигрываем. Даже наоборот. При соблюдении всех природоохранных норм в производстве потребительские качества продукции неизбежно окажутся ниже.

Олово и висмут на порядок дороже свинца, индий еще дороже, а технологические свойства припоев на их основе существенно хуже. Если исключить некоторые типы пластмасс, придется применять либо более дорогие, либо менее прочные.

То же самое и с чернилами для принтеров. Вы не хотите, чтобы распечатки "текли" при попадании капли воды? Тогда краситель не должен растворяться в воде, а значит, чернила должны содержать органические растворители, причем быстро испаряющиеся. Ну, давайте и растворители запретим...

Но... Если стандарт будет принят, то выполнять его придется всем без исключения, причем не только европейцам, но и азиатам, не то они потеряют европейский рынок. Значит, стоимость всей продукции возрастет примерно одинаково у всех производителей. Скажем, системные платы подорожают на два-три доллара, из которых несколько десятков центов составит прибыль. А фирмы, разработавшие и запатентовавшие "экологически чистые" припои, конечно же, какое-то время будут и производить их, да еще получать лицензионные отчисления от всех остальных производителей.

Зато на задней панели монитора будет гордо красоваться значок EPA (Environmental Protection Agency, Министерство охраны природы США) или какой-нибудь европейский "голубой ангел". Охрана природы — штука выгодная. Правда, не столько для природы, сколько для тех, кто ее охраняет...

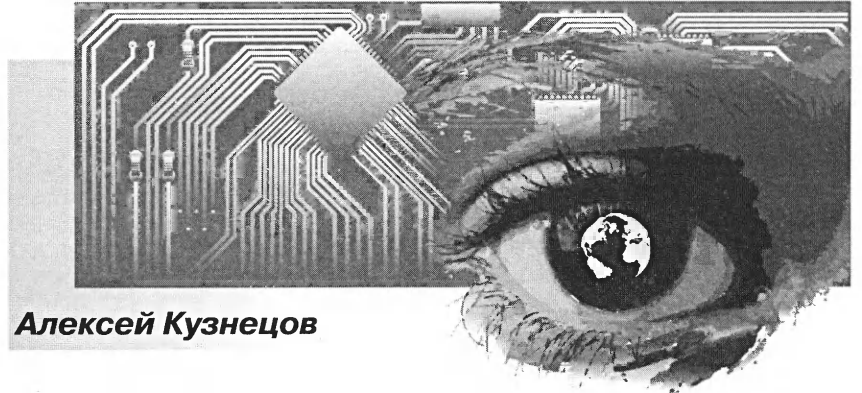


Данная статья посвящена не российской истории, а такому явлению Сети, как веб-камеры. Последние годы это направление усиленно развивается во всем мире и у нас благодаря повышению пропускной способности каналов связи, скоростей подключения и, особенно, благодаря удешевлению самих камер.

Идея проста: устройство, внешне похожее на камеру видеонаблюдения, передает картинку в Сеть. Объектами для камер обычно выбирается что-либо примечательное вообще или по мнению создателей камеры. В Питере первая камера показывала и показывает Мойку (<http://www.livacam.ru>), есть камеры, показывающие Петропавловскую крепость (<http://www.fortress.spb.osi.ru>) и Исаакиевский собор, Невский проспект (<http://www.metrocom.ru>). Домашние камеры отдельных граждан показывают интерьеры их домов.

Классическая веб-камера работает в режиме слайд-шоу, являясь по сути фотокамерой. Рисунок в формате JPG, получаемый с камеры, периодически обновляется на сервере. Сама страничка тоже часто делается самообновляющейся, например, введением тега `<META content=120 http-equiv=REFRESH>`, в данном случае дающего обновление каждые 2 минуты. В остальном HTML-код никаких особенностей обычно не имеет. Период обновления составляет от 1 минуты до получаса и даже суток.

Очень немногие камеры дают живое видео, хотя скорости соединения 28K вполне достаточно для передачи живого видео, конечно, при условии компрессии. Причины чисто экономические: живая камера дает большой трафик на сервере провайдера. Обычно только первое время камеры работают в живом режиме в целях раскрутки странички, а затем их "замедляют". Насколько в реальном времени работает та или иная камера, определить сложно, а иногда и просто невозможно. Сомневаюсь, конечно, что на камерах изображена ретроспектива за



Алексей Кузнецов

## Окно в Европу

последние несколько лет, но иметь период обновления, скажем, раз в сутки очень выгодно: не надо постоянного подключения к Сети, можно пользоваться сеансовым доступом через модем и передавать историю последнего дня на сервер в удобное время суток.



Петербург

На питерских камерах погода соответствует тому, что я вижу за окном. Но этого "открытия" мне показалось мало, и я стал ждать плохого прогноза по всей Европе. Дождавшись мощного циклона, я стал смотреть по камерам. Везде погода соответствовала ожиданиям: небо Женева было сродни питерскому в ненастный день.

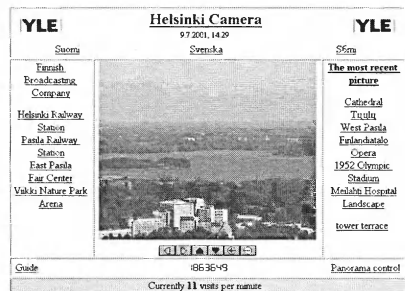
В отдельную категорию следует выделить камеры, работающие в режиме Real Video (для просмотра их изображений необходим Real Player

G2). Такая камера работает на Невском (<http://www.metrocom.ru>). Для кодирования данных на передающей стороне используется Real Encoder, программа-кодировщик реального времени, поддерживающая получение картинки с видеокamеры. Для нормальной работы в таком режиме ей необходим процессор не ниже PII и программное обеспечение Real Server. Часто статическое изображение в виде того же JPG на страничке такой камеры является ссылкой на файл \*.ram, формируемый кодировщиком. На таких страничках часто имеется ссылка на <http://www.real.com>, где предлагается скачать Real Player G2.

Некоторые камеры предлагают выбрать картинку или видео с несколькими режимами того и другого (для каждого режима на сайте своя страничка). Таких камер много в Германии, некоторые из них даже предлагают функцию Zoom, правда, программную (программа работает на сервере). Аппаратно Zoom реализован на камере в Хельсинки (<http://www.zorro.org/scripts/camera.dll?US>). Камера, укрепленная, как я понял, на телебашне, может поворачиваться вверх-вниз, вправо-влево и давать неплохое увеличение без потери четкости.

После нажатия на любую кнопку из шести страничка полностью об-

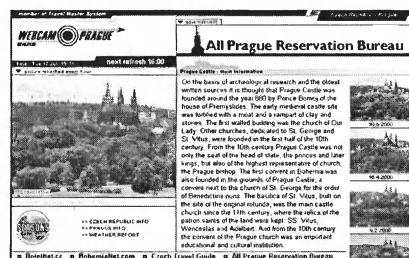
новляется с новой картинкой, при этом камера действительно поворачивается по вашей команде (кнопки являются простыми ссылками с соответствующими атрибутами на модуль camera.dll, управляющий камерой и генерирующий новый текст странички). Камера доступна даже для посетителя с модемом 9600, так как картинка и текст странички занимают всего 20 Кб (отклик на команду — примерно через 15 секунд).



Хельсинки

К сожалению, когда на камере больше двух посетителей в минуту, камера уже не слушается, а картинка дополняется сообщением о том, что камера занята. 15 ссылок на тот же модуль camera.dll по обе стороны от картинки настраивают камеру на показ таких достопримечательностей, как Олимпийский стадион 1952 года, собор, опера, вокзал и др. Вообще это наиболее функционально оснащенная камера из известных мне, если не считать ее программно-аппаратного близнеца в Лаhti (<http://masto.lahti.fi/scripts/camera.dll?ShowLast>), но панорама там менее интересна.

Мне лично больше всего понравился дизайн странички камеры в Праге (<http://www.ctg.cz/webcam>). Картинка там обновляется каждые полчаса, причем имеются часы, а рядом с ними время очередного обновления (генерируются Java-

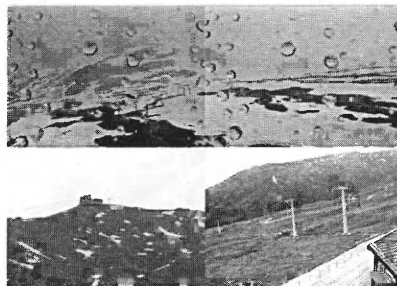


Прага

скриптом, опорное время берется по часам вашего компьютера). Рядом с основной картинкой — текст, поясняющий по-английски, что на картинке вы видите Пражский замок, построенный в 880 году.

Кроме того, есть маленькие картинки лучших видов на камере за прошедшие два года, являющиеся ссылками на своих собратьев с большим разрешением (ссылка дается на файл \*.html, в который картинка вставлена как фоновый рисунок, а благодаря все тому же Java-скрипту она открывается в новом окне IE, не имеющем элементов управления вообще). В целом все выглядит как страничка из энциклопедии.

По адресу [http://www.alpenarena.ch/weather/live\\_cams](http://www.alpenarena.ch/weather/live_cams) погода меняется постоянно (камера установлена на опоре фуникулера высоко в горах), и через сутки густой туман сменяется проливным дождем. Камера весьма экстремальная и по содержанию, и по условиям работы, в нее интересно смотреть каждый день в любое время года. Приятно также то, что страничка камеры не обременена примочками, грузится быстро (несмотря на то, что изображение содержится в файле формата BMP) и дает изображение телевизионного качества — своего рода эталон хорошего тона при оформлении веб-камер.



Альпы

По ссылке со странички можно посмотреть на фотографию камеры со стороны (метров с 500) и на панораму гор, даваемую сразу двумя камерами (картинка совмещена). Тему горных пейзажей продолжает



Ламбург

камера по адресу [www.topin.ch/be/druelles/default.asp](http://www.topin.ch/be/druelles/default.asp), правда, камера там всего одна.

Еще одной разновидностью камер являются такие, в которых основное изображение обновляется Java-апплетом (подпрограммой Java, загружаемой в папку Downloaded Program Files в каталоге Windows при первом входе на страничку).

Страничка при этом не регенерируется полностью, а лишь обновляется изображение с камеры. Приятно посмотреть, например, на [www.earthcam.com/usa/newyork/timesquare/cam6.html](http://www.earthcam.com/usa/newyork/timesquare/cam6.html). Кроме того, на этой же страничке предлагается потоковое видео для Windows Media Player (в файле формата \*.asx). Вообще эта страничка — чемпион по сложности программных решений, в ней применены почти все основные возможности DHTML (код, ответственный непосредственно за показ изображения, занимает 3%, а до 70% занимает переключатель камер, скрипты, отслеживающие версию браузера и прочие спецэффекты).



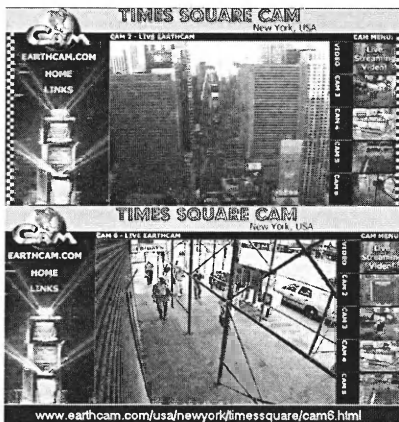
Аляска

Иногда такие Java-камеры интерактивны ([www.capitolium.org/eng/virtuale/liveview.htm](http://www.capitolium.org/eng/virtuale/liveview.htm)). По сути здесь HTML-страничка лишь образует окно для выполнения Java-программы, которая передает при нажатии кнопки Start Control новые координаты камеры и получает новую картинку. Zoom работает великолепно, а вот поворачивается камера как-то неожиданно резко. Нет "наводки" на достопримечательности, зато телевизионная антенна видна во всех деталях.

К сожалению, некоторые камеры

настолько обвешаны баннерами, что ждать загрузки изображения приходится довольно долго, а сама картинка маленькая и какая-то бесцветная, да и с композицией неважно. Иногда объект, указанный в пояснении, едва виден на горизонте, а то и заслонен чем-либо громоздким. Таких примеров, к счастью, не много.

Каждая камера по-своему уникальна, ведь если она не просто стоит за окном квартиры, в теплой комнате, и общается с Сетью по модему раз в месяц, то задача инженера, создающего ее, достаточно трудна. Часто надо совместить несовместимое: саму камеру как аппаратное средство, существующий канал связи, возможности сервера и новые идеи создателя. Задача имеет много переменных и поэтому часто порождает новшества технического и программного плана. Поэтому каждая камера — это индивидуальный проект, новая разработка.



Нью-Йорк

Камеру надо поддерживать, и достаточно большой трафик, создаваемый ею на сервере провайдера, иногда ставит посещение камеры на коммерческую основу или заставляет размещать на ее страничке рекламу. Да и место установки часто приходится согласовывать с кем-либо, ведь камера не бесплотный дух и ее надо где-то закрепить, кабель к ней проложить от провайдера или радиомодем поставить, вопрос электропитания



Париж

решить. К тому же надо написать программу, обновляющую картинку на сервере, и, наконец, все это собрать и наладить.



Лондон

Камеры редко содержат историю своего создания, а жаль. История создания многих камер не менее занимательна, чем картинка на них, и заслуга инженеров и менеджеров проекта уходит в тень незаслуженно. У ландшафтных веб-камер титры пока делать не принято. Но художник подписывает свою картину, почему не подписывают веб-камеры? Изредка можно лишь узнать название фирмы-создателя. Об инженерном решении камеры в большинстве случаев можно судить лишь по косвенным признакам.

В этом рассказе я сознательно обошел перечисление ссылок на домашние камеры (там, кстати, все подписи есть) — тесные обиталища европейцев достаточно однообразны и днем совершенно пустынно, а ночью в них темно или картинка не обновляется с вечера. Ну, а любителям камер для взрослых в большинстве случаев придется запастись банковской карточкой.

Камеры, конечно, пока еще далеки от совершенства. Причина этого — низкая скорость каналов связи, дорогой трафик. К тому же HTML еще недостаточно подстроен под живое видео, поэтому создать по-настоящему живую камеру очень не просто, особенно интерактивную — часто приходится обновлять не только изображение с камеры, но и всю ее страничку. Это ограничивает количество кадров, так как страничка обновляется полсекунды и более, независимо от скорости канала пользователя. Конечно, Real Player — выход из

положения, но им создатели камер пользуются не очень охотно, поскольку это ограничивает круг посетителей камеры (Real Player должен быть установлен на компьютере пользователя, а "модемные" скорости все же несколько маловаты для Real Video). С технологией Java проблем меньше, но и здесь все упирается в скорость каналов передачи данных.



Киль

Качество живого изображения пока хуже телевизионного даже при скорости 128K, поэтому на камерах господствует GIF, JPEG и даже BMP с большим разрешением. Но даже то, что существует сейчас, — огромный технологический прорыв, и я надеюсь, что повсеместное живое видео не за горами. Поживем, увидим. Пока просмотрите то, что уже есть (по ссылкам из статьи и на путеводителях [www.utis.com.ua/run/data/index.htm](http://www.utis.com.ua/run/data/index.htm), [www.spg.ru/connect/camera.htm](http://www.spg.ru/connect/camera.htm), [www.supertv.boom.ru](http://www.supertv.boom.ru)). А потом — вперед, на поиски новых веб-камер. На Yandex по запросу "live cam" я получил сотни ссылок, многие из которых вели на путеводители, причем тематика была самая разнообразная.

**Успехов!**



Владислав Гуськов

# Сделай сам: гостевая книга

**П**рограммирование для Сети обычно начинается с языка HTML, создания собственной странички. К сожалению, технология гипертекста не обладает динамичностью, поэтому счетчики, голосования, гостевые книги остаются только в мечтах веб-мастеров. Для расширения возможностей HTML служит язык JavaScript, способный выводить на экран анимационную "мишуру", перемещать объекты. Главным недостатком здесь считается выполнение скрипта на стороне клиента, отсюда — слишком узкая специализация.

Гораздо более широкими возможностями для создания динамических страниц обладают скрипты, выполняемые на сервере: от простого вывода текста до работы с файлами. С помощью шлюзового интерфейса (CGI) скрипт получает данные от клиента (например, браузера) и приступает к их обработке. Сценарии можно писать на многих языках, но лично мне более симпатичен Perl, мощное средство для создания отчетов с довольно простым синтаксисом.

Чаще всего веб-мастерам хочется иметь собственную гостевую книгу. Устройство ее не такое уж и сложное, так что давайте попробуем создать собственную гостевую книгу, а заодно и повторим синтаксис.

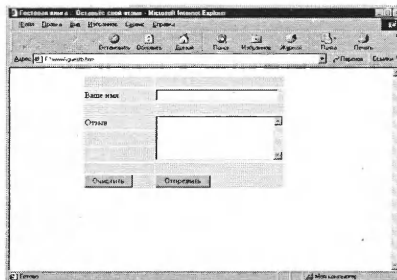
## Как это работает

Итак, наша гостевая книга будет состоять из трех файлов:

1. Файл .htm, пользователю выводится форма, в которой он заполняет поля (имя и отзыв) и нажимает кнопку "Отправить".

2. Сам скрипт с расширением .cgi, он отвечает за обработку запроса, записи результатов в файл и вывод всех отзывов.

3. Небольшая база данных, содержащая имена и отзывы посетителей. Под запись выделяется одна строка, причем имена отделяются от текста отзыва спецсимволом | (в принципе, можно любым другим).



Выглядит это так:

ИМЯ ПОСЕТИТЕЛЯ | ОТЗЫВ  
ИМЯ ПОСЕТИТЕЛЯ | ОТЗЫВ

Создать форму на языке HTML не представит особого труда, БД скрипт будет заполнять сам, остается написать его. Сначала прикинем объем работ и составим алгоритм:

1. Пользователь обращается к файлу gb.htm, а в ответ получает форму.

2. После ее заполнения он нажимает кнопку "Отправить", и все данные отправляются на сервер.

3. CGI формирует среду переменных для скрипта и вызывает его для дальнейшей работы.

4. Скрипт получает данные и декодирует их в переменные.

5. Добавляет имя посетителя и отзыв в базу данных.

6. Считывает всю информацию из БД в массив.

7. Организует цикл и выводит HTML-таблицу, содержащую все отзывы.

## Пляшем от печки

Даже увлекаясь CGI-программированием, забывать язык HTML не рекомендуется, поскольку весь результат работы скриптов выводится в гипертексте. Еще бы, как же тогда браузеры смогут прочитать полученную информацию? Первым делом необходимо создать форму для ввода данных посетителем. Проводить экскурс в HTML нет смысла, поэтому ограничусь короткими замечаниями. Содержание, оформление документа может быть любым, но в нем обязательно должен присутствовать тег <form>, обозначающий способ отправки запроса и расположение скрипта, которому предназначается вся информация.

В нашем случае для ввода имени посетителя используется текстовое поле (textfield), его ширина регулируется опцией size. Отзыв может не уместиться в такое поле, поэтому используется текстовая область (textarea). Количество и ширину рядов задают параметры rows и cols. После ввода текста посетитель нажимает на кнопку (submit), и данные отправляются на сервер. За надпись на кнопке отвечает опция value. На рисунке я привел форму, созданную с помощью таблиц. Еще раз повторю, дизайн странички может быть ЛЮБЫМ, главное — теги, обозначающие форму.

Примерно такой блок должен присутствовать в любой форме:

```
<form method=get action=/cgi-bin/example.cgi>
```

```
<input type=text name=textfield size=32>
```

```
<textarea name=textarea rows=5 cols=28></textarea>
```

```
<input type=reset name=Reset value=Очистить>
```

```
<input type=submit name=Submit value=Отправить>
```

```
</form>
```

Однако, если вы не до конца понимаете устройство форм, используйте веб-редакторы, такие как Frontpage или Dreamweaver.

### Запрос: между сервером и клиентом

В основу World Wide Web положен принцип деления всех компьютеров в сети на серверы и клиенты, общающиеся по единому протоколу HTTP. Клиент (браузер) передает серверу сведения о том, какой ему необходим документ, а сервер в свою очередь высылает его клиенту. Обычно запрос выглядит так:

```
GET <название файла> HTTP/1.0
```

Параметр GET указывает способ отправки данных (еще бывает POST, но о нем поговорим позднее), а HTTP/1.0 — версию протокола, с которой работает браузер. Обычному пользователю незачем знать такие премудрости, поскольку от него требуется только вводить адреса сайтов в адресной строке, а браузер сам составляет запрос к серверу. Но CGI-программисты о параметрах запросов должны знать побольше.

После того как пользователь нажал на кнопку "Отправить", браузер обрабатывает данные полей, кодирует все кириллические символы в шестнадцатеричной системе, формирует запрос и передает его на сервер.

```
GET /cgi-bin/example.cgi?textfield=%D3%E8%E2&textarea=%CD%F0%E3%F2
```

```
HTTP/1.0
```

Структура запроса несколько не изменилась, только что это за абракадабра из символов процента, амперсанта? А это и есть закодированные поля формы. Знак вопроса пос-

ле файла означает, что дальше идут данные. Передаются они парами имя=значение и разделяются амперсантом (&). Кириллические символы не могут корректно передаваться от браузера серверу, поэтому они кодируются знаком процента и следующей за ним двухбайтовой комбинацией. Пробел заменяется знаком плюса.

Вот в таком ужасном виде запрос попадает к скрипту. Теперь надо декодировать запрос и распределить его по переменным.

### На пути к гостевой книге

Вот исходный код гостевой книги:

```
#!/usr/local/bin/perl
#Настройки
$path='/cgi-bin';
$width='width=90%'; #Общая ширина таблицы
@TableWidth=('width=10%', 'width=15%', 'width=35%', 'width=10%');
# Блок №1
print "Content-type: text/html\n\n";
$query=$ENV{'QUERY_STRING'};
```

## Интернет и usability

Случалось ли вам, уважаемые читатели, поселившись в гостинице, вдруг обнаружить, что душа в номере нет, а единственный санузел находится на этаже? Или в кафе с ужасом осознать, что стул привинчен к полу и его невозможно придвинуть к столику, чтобы не тянуться к тарелке за три метра? Думаю, что в таких ситуациях оказывался каждый. Несомненно, отсутствие элементарных условий комфорта — большой минус для тех, кто предоставляет те или иные услуги, тем более — платные. Понятно, что любые услуги, даже элитные, должны соответствовать представлениям потребителя об удобстве, практичности, скорости, результативности и прочих качествах, отсутствие которых мы называем дискомфортом, а услуги, явно не соответствующие нашим запросам — халтурой и надувательством.

Сегодня Интернет — это огромный кладезь самых разнообразных услуг. Соответственно, раз на данном рынке нет дефицита тех или иных услуг, они должны четко соответствовать запросам пользователя. И пользователь, естественно, надеется получить качественное обслуживание и полный комфорт. Именно эти два компонента успеха поставщика услуг и составляют систему знаний о параметрах оценки и критериях эффективности оказываемых услуг, которая называется usability.

В переводе с английского это слово можно перевести как "удобство", "полезность" или, что ближе по смыслу, — "удобство пользования". За рубежом (особенно в США) понятием usability оперируют самые разнообразные науки: психология, социология, менеджмент, маркетинг и др. Разумеется, в каждой области есть своя специфика, однако суть

понятия usability остается неизменной — систематизированный методологически и технологически комплексный подход к предоставлению качественных услуг. В области информационных технологий и, в частности, Интернета, такой подход также необходим.

Сегодня система usability получает все большее распространение в процессе создания интернет-проектов, изучения потребительского поведения и т. д. Как и любая система, usability (далее будем говорить только об Интернете) включает в себя отдельные направления. Всего в специализированных источниках указывается около 100 направлений usability, но наиболее важными и интересными, на мой взгляд, являются дизайн, реклама, навигация, человеческий фактор, графика. На них я и остановлюсь подробнее.

### Дизайн

Вопросам дизайна отводится одно из главных мест в системе

```

@fields=split(/&,$query);
foreach (@fields) {
$_ =~ s/\+//g;
$_ =~ s/%([a-fA-F0-9][a-fA-F0-9])/;
pack("C", hex($1))/eg;
if (/^(.*)=(.*)/)
{push(@FieldsDecoded,$2);}
}
# Блок №2
$NewRecord=$FieldsDecoded
[0].'. '$FieldsDecoded[1];
if ($FieldsDecoded[0]!~/\S/ ||
$FieldsDecoded[1]!~/\S/) {
print "He все поля заполнены";
exit;
}
open (file,">>$path/records.txt");
print file "$NewRecord\n";
close(file);
# Блок №3
open (file,"$path/records.txt");
@Records=<file>;
close(file);
#Блок №4
$count=2;
foreach (@Records) {$count++;}
print qq {
<html>

```

```

<head>
<title>Гостевая книга :: Просмотр
отзывов</title>
</head>
<body bgcolor="#FFFFFF">
<table $width cellspacing=1
cellpadding=3 border="0"> <tr>
<td @TableWidth[0]
rowspan=$count>&nbsp;&nbsp;&nbsp;</td>
<td @TableWidth[1]>&nbsp;&nbsp;&nbsp;</td>
<td @TableWidth[2]>&nbsp;&nbsp;&nbsp;</td>
<td @TableWidth[3]
rowspan=$count>&nbsp;&nbsp;&nbsp;</td>
</tr>
<tr bgcolor="#E1E1E1">
<td colspan=2
align=center><b>Получены следующие отзывы:</b>
</td>
</tr>;
foreach (@Records) {
@temp=split(/|,$_);
print "<tr bgcolor=#E1E1E1>\n";
print "<td @TableWidth[1]>$temp[0]</td>\n";
print "<td @TableWidth[2]>$temp[1]</td>\n";
print "</tr>";
}

```

```

print "</table>\n";
print "</body>\n";
print "</html>\n";

```

Теперь разберемся в каждой строчке. Согласно нашему алгоритму скрипт разбит на четыре блока: получение и декодирование запроса, запись отзыва в БД, получение всех записей из БД, опубликование документа в формате HTML.

Назначение первой строчки, надеюсь, понятно всем: указание на расположение интерпретатора. Заметьте, в начале строки стоит символ #, обычно использующийся в качестве комментария, но в данном случае его наличие необходимо.

Программист, создавая скрипт, не знает, где он будет использоваться. Поэтому скрипт должен быть максимально гибким к изменениям. Пути ко всем файлам задаются в переменной \$path, чтобы при необходимости можно было легко изменить их. Если дизайн страничек поручен другому человеку, имеющему иные вкусы, то, вполне возможно, его не удовлетворят размеры таблицы.

Целесообразнее всего сделать

usability. И это не удивительно, если учесть, что первое впечатление о вашем интернет-проекте у посетителя будет складываться именно на основе визуальной оценки сайта. Как говорится, "встречают по одежке". Однако рассуждать о том, что правильно в дизайне, а что нет, — задача не из легких, так как любой подход к оценке дизайнерского решения в той или иной мере субъективен, что не дает возможности судить о соответствии сайта критериям usability. Для достижения максимальной объективности были разработаны специальные методики оценки сайта по критериям usability. Одна из наиболее простых и результативных — методика "Success Rate: The Simplest Usability Metric", разработанная основоположником системы usability Якобом Нильсеном. Однако к методикам мы вернемся чуть позже, а сейчас — мнение профессионалов по поводу usability в дизайне интернет-проектов.

Крис Пол, креатив-директор IBM WebSphere: "...Если вы озадачены

тем, что ситуация на сайте не соответствует вашим ожиданиям, проблема, возможно, кроется не в том, что предлагаемые вашей компанией продукты или услуги не востребованы на рынке, и не в плохом подходе к информационному наполнению сайта. Причина может заключаться в недостаточном внимании к пользовательским предпочтениям. Мало использовать последние достижения в области веб-технологий, активно применять системы электронной коммерции и размещаться на сверхскоростных серверах. Если вы хотите добиться успеха, необходимо тщательно изучить предпочтения пользователя ваших услуг... Понятие "пользовательские предпочтения" можно рассматривать как суммарную степень удовлетворения человека от пользования вашим сайтом... И внешнее оформление здесь играет одну из ключевых ролей..."

Другими словами, посетителя в последнюю очередь волнует, на какой аппаратной платформе работает сайт и какие в нем используются ноу-

хау. Первое, на что он обращает внимание, — это внешний вид. В зависимости от того, удовлетворен ли посетитель увиденным, зависит его дальнейшее поведение как потребителя.

В качестве альтернативы предыдущему мнению привожу доводы Питера Сибаха, независимого эксперта по вопросам usability. Он во главу угла ставит понятие удобного дизайна:

"...Веб-сайты разрабатываются людьми, имеющими мощные компьютеры, современные браузеры, полную документацию и быструю локальную сеть... А просматривают эти веб-сайты люди с самыми разными по своим возможностям компьютерами, модемами, браузерами и пр. Первым гораздо проще давать советы вторым относительно того, как нужно обустроить свои рабочие места, чем устранить барьер несовместимости... В результате складывается впечатление, что разработчики стремятся к тому, чтобы сделать свои веб-сайты как можно менее доступными для пользователя..."

параметр width легко изменяемым. Конечно, еще много чего можно изменить, но пора приступать к делу.

```
print "Content-type: text/html\n\n";
```

Первым делом выводим строку, означающую тип документа, в нашем случае это формат HTML. Значения Content-type могут быть различными, в зависимости от целей: если выводится картинка, то используется image/gif или image/jpeg. Но в любом случае эта строчка должна обязательно присутствовать, иначе вывести результаты будет просто невозможно.

```
$query=$ENV{'QUERY_STRING'};
```

В CGI переменную QUERY\_STRING помещаются данные формы. Чтобы начать с ними работу, присваиваем ее значение \$query.

Запрос состоит из пары имя=значение и разделен знаком амперсанта (&). Функция split (/ЗНАК/,ПЕРЕМЕННАЯ) осуществляет поиск заданного знака в переменной и помещенные найденные объекты в массив.

Нас интересует ЗНАЧЕНИЕ, и для этого мы должны декодировать каж-

дый элемент массива, отбросив ненужную часть ИМЯ=. Цикл foreach наиболее удобен для этих целей, поскольку на каждом шаге выбирается следующий элемент массива. Значение элементов попадает во временную переменную \$\_, с ней мы и работаем. Используя регулярные выражения, заменяем знак + на символ пробела, а функцией hex() декодируем символы. Как же отбросить ненужную часть? Мне нравится такой способ:

```
if (/^(.*)=(.*)/) {push
(@FieldsDecoded,$2);}
```

Представляем выражение как два набора символов, разделенных знаком равенства. Значения, помещенные в скобки (.\*), присваиваются временным переменным \$1 и \$2. Поскольку нам нужна только вторая часть, вставляем ее в массив @FieldsDecoded. Рекомендую воспользоваться для этого функцией push(@МАССИВ,\$ПЕРЕМЕННАЯ), добавляющей значение переменной в конец массива. Если ее не применять, то последующие значения будут записываться в один и тот же

элемент, стирая предыдущие. После выполнения цикла мы получаем массив @FieldsDecoded, содержащий имя и отзыв посетителя. Меня иногда упрекают: "Зачем изобретать велосипед, можно же выполнить проще?". Не спорю, есть способы упростить себе работу по декодированию запросов, например, используя модуль CGI.pm. После строки Content-type вставьте следующий код, и результат получится аналогичным, а экономия строк налицо:

```
use CGI;
$query=new CGI;
$name=$query->param('name');
$respond=$query->param('respond');
```

### Запись в БД

Перед тем как перейти ко второму пункту задачи, необходимо проверить, введены ли имя и отзыв посетителя. Легче всего использовать регулярные выражения. Символ '\S' означает любой непробельный символ, а '||' — логическое ИЛИ.

```
if ($FieldsDecoded[0]!~/\S/ ||
$FieldsDecoded[1]!~/\S/
```

зайн станет исключением из правил только если получит оценку usability на 100% выше стандартной.

- Если 60—90% разработчиков создают оригинальный дизайн, то он становится устоявшейся условностью, и вам приходится уступать перед ней до тех пор, пока ваш альтернативный дизайн не получит 50-процентной оценки степени usability.

- Если меньше 60% разработчиков создают оригинальный дизайн, пока единой доминанты в дизайне не существует, вы имеете возможность создавать альтернативный дизайн. При этом, если по меньшей мере 20% разработчиков используют несколько оригинальных элементов дизайна, вам следует ограничиться применением одного из таких элементов до тех пор, пока ваш альтернативный дизайн не получит оценки usability на 25% выше, чем сайты других разработчиков, сделавших свой выбор наилучших элементов дизайна.

Таким образом, можно выделить еще один подход к оценке usability в

Здесь мы имеем дело с неким интегрированным направлением usability, получившим название "accessibility" (доступность). Конечная цель в данном случае — определение степени общей доступности ресурса его потребителям. В паре "разработчик—посетитель сайта" акцент сделан именно на функциональной стороне. За счет снижения требований к визуальной привлекательности проекта достигается полная совместимость, а с ней — accessibility и usability. Ядром проектирования веб-сайтов становится дизайн, подогнанный под некий усредненный стандарт, определенный путем статистических исследований по используемому ПО, операционным системам, периферии и т. д.

Но в современном Интернете чаще бывает наоборот — никакие стандарты не принимаются, а проектирование строится по принципу "кто в лес, кто по дрова". В результате веб-продукт корректно отображается у половины пользователей, у 25% он требует дополнительного ПО, а

последние 25% вообще лишены возможности с ним ознакомиться. И это еще не это самое плохое. Печально то, что такое вполне сознательное неприятие основ usability для многих веб-разработчиков стало входить в норму. Вот что думает по этому поводу Якоб Нильсен:



*"Веб-дизайн — вещь простая. Если вам нужно разработать веб-сайт или какой-то отдельный элемент, все, что вам потребуется для этого, — посмотреть, как это было реализовано на двух десятках других, наиболее популярных сайтов..."*

Из этого утверждения Нильсен выводит закономерность — "плохой дизайн становится стандартом":

- Если 90% разработчиков или более создают оригинальный дизайн, то он становится общепризнанным стандартом, которому вы вынуждены подчиняться. Ваш ди-

Если хотя бы одна из этих переменных не содержит ни одного символа, то выводится текст о незаполненных полях. Команда 'exit' прерывает работу скрипта.

Чтобы было легко отделить имя посетителя от отзыва, поставим между ними знак ( | ), используя конкатенацию слов.

```
$NewRecord=$FieldsDecoded[0].
'|'. $FieldsDecoded[1];
```

В результате у нас одна переменная, содержащая ИМЯ | ОТЗЫВ.

Файл открывается с помощью функции open(ДЕСКРИПТОР, "ФАЙЛ"), где дескриптор — любое символическое имя для представления файла в программе. Знак ">>" перед именем файла называется флагом, определяющим, в каком режиме будет открыт файл. В нашем случае это режим добавления новых записей.

Символ флага	Значение
<	Чтение (по умолчанию)
>	Запись
+=	Чтение/Запись
+<	Стирает содержимое, запись в начало
>>	Добавление

Обычно функцию print использу-

ют для вывода информации на экран, но если после нее поставить дескриптор, то данные будут записываться в файл. Обратите внимание, после переменной ставится знак \n (переход на новую строку). Это обеспечивает распознавание скриптом разных записей, отделенных друг от друга новой строкой. Не забудьте взять символ в кавычки.

После добавления отзыва надо вывести все имеющиеся записи в нашей БД на печать. Для этого снова открываем файл, но уже с флагом '<' (можно и не использовать, поскольку он стоит по умолчанию) и присваиваем массиву @Records все содержимое файла. Считывание идет построчно, поэтому каждому элементу массива будет присвоена строка (ИМЯ | ОТЗЫВ).

### Вывод информации

Переходим к последнему шагу алгоритма. Вся работа скрипта до этого была скрыта от глаз пользователя, но теперь необходимо выдать результат. Самый простой способ —

организовать цикл и разделять имя и отзывы символом пустой строки, например, следующим образом:

```
foreach (@Records) {
    @temp=split(/|/, $_);
    print "$temp[0]\n";
    print "$temp[1]\n";
}
```

Сразу же бросаются в глаза недостатки такого решения: дискомфорт в просмотре документа, ведь он оформлен без единого HTML-тега. Мнение посетителя для программиста является главенствующим, поэтому на разработку удобного интерфейса сделаем главный упор. Я предлагаю выводить текст в таблице из четырех колонок: первая и последняя — пустые, дающие позиционирование текста по центру, а во второй и в третьей будут располагаться соответственно имя и отзыв посетителя. Как это будет выглядеть, показано на рисунке.

Получены следующие отзывы:	
Евася	Все круто
Петя	Отстой
Женя	Полный Failure!!!

Пустые колонки создаются пара-

дизайне, условно обозначив его "альтернативным дизайном". В основе такого подхода лежит стратегия умелого маневрирования между устоявшимися стандартами и оригинальными разработками. В процессе создания чего-либо нового, необычного, оперируя показателями usability по отношению к стандарту, нельзя слишком близко приближаться к нему. Именно такая балансировка, по мнению Нильсена, и дает разработчикам возможность создавать оригинальный дизайн.

### Резюме

В конечном счете мы имеем три подхода к оценке usability в дизайне веб-сайтов:

#### 1. "Красивый дизайн":

Основная задача — создание привлекательного внешнего вида проекта, ориентируясь на конкретные визуальные предпочтения пользователей.

*Минусы:* достижения ИТ отходят на второй план и мало соотносятся с общей концепцией usability; удов-

летворенность внешним видом конфликтует с недоработками в плане совместимости и технической базы проекта.

*Плюсы:* запоминающееся оформление веб-сайта.

#### 2. "Удобный дизайн":

Основная задача — достижение максимального уровня accessibility веб-сайта за счет устранения барьеров несовместимости между технической комплектацией разработчиков и потребителей.

*Минусы:* Вынужденное ограничение дизайнерских решений.

*Плюсы:* Максимальный охват пользовательской аудитории, доступность и простота использования.

#### 3. "Альтернативный дизайн":

Основная задача — разработка оригинального дизайна посредством маневрирования между стандартом и альтернативой.

*Минусы:* Угроза попадания "в зависимость" от устоявшихся стандартов в дизайне или необходимости постоянно создавать альтернативный дизайн.

*Плюсы:* Получение наглядного представления о ситуации за счет постоянного изучения сложившихся стандартов, адаптация к конкурентной среде.

### Ссылки по теме:

Jacob Nielsen's UseIt.com — <http://www.useit.com/>

Usability First — <http://www.usabilityfirst.com/>

IBM DeveloperWorks Usability Column — <http://www-106.ibm.com/developerworks/usability/>

UIWEB.COM — Web and interaction design — <http://uiweb.com/>

School of Computing and Information Technology, Griffith University — <http://www.gu.edu.au/>

ClickZ — Site Design Column — <http://clickz.com/column/sd.html>

User Interface Engineering — <http://world.std.com/~uiweb>

Usability в России — <http://www.usability.ru/>

*Продолжение следует.*

Алексей Петюшкин



метром rowspan в первом ряду. Его значение — это количество рядов в таблице. Их будет столько же, сколько и отзывов посетителей плюс два (первый ряд — пустой, второй содержит заголовок "Получены следующие отзывы"), следовательно, появляется необходимость их изменить. Применяем все тот же универсальный цикл foreach. Переменная \$count будет содержать количество рядов.

```
$count=2;
foreach (@Records) {$count++;
```

Самое время вспомнить HTML-теги, без них — никуда. Выводить их можно несколькими способами функцией print.

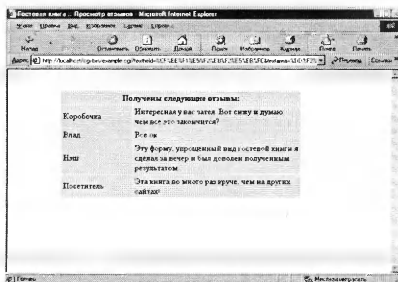
```
print "<ter>\n";
либо:
print { <ter>
      <ter>};
```

Первая форма используется при выводе одной строки. Если же имеется целый блок текста, целесообразнее применить вторую форму. При этом не нужно ставить кавычки и писать в начале каждой строки "print".

При печати отзывов каждый элемент массива @Records разбиваем с помощью функции split и загоняем в массив @temp. Теперь \$temp[0]

содержит имя пользователя, а \$temp[1] — отзыв. Не забудьте, что при обращении к определенному элементу массива ставится знак скалярной переменной.

```
<tr bgcolor=#E1E1E1>
<td $TableWidth[1]>$temp[0]</td>
<td $TableWidth[2]>$temp[1]</td>
</tr>
```



На каждом следующем шаге цикл создает новый ряд, используя парные теги <tr> и </tr>.

Внутри ряда обозначаем две колонки: для имени и отзыва. Еще две указывать не надо, параметр rowspan в начале таблицы избавляет нас от рутинной работы. В массиве @TableWidth хранится информация о ширине колонок, соответственно, мы используем только первый и вто-

рой элементы. В конце документа осталось дописать закрывающие теги, и гостевая книга готова!

**Мы ее слепили из того, что было...**

Теперь у вас есть собственная гостевая книга, и нет необходимости искать по каталогам бесплатных скриптов. Устройство ее не такое уж и сложное, поэтому без труда можно внести некоторые поправки, улучшить функциональность. Правда, за простотой исполнения скрываются некоторые недостатки: отсутствие проверки корректности информации, введенной посетителем, записи IP-адреса или даты внесения отзыва. Среди особых "примочек" я бы отметил скрипт для морфологической проверки слов в отзыве, например, для исключения нецензурных выражений.

Улучшать книгу можно до бесконечности, в зависимости от задач, которые вы перед собой ставите. Правда, чем сложнее система, тем больше вероятность появления "дыр" в безопасности. Созданием удобного, функционального и, главное, безопасного скрипта мы займемся в следующих статьях.

# PHP — мощь Perl и гибкость JavaScript

Последнее время в компьютерной прессе большое внимание уделяется CGI- и JavaScriptам, и не удивительно: данные средства дают много возможностей для разработки интерактивных элементов веб-страниц, тем самым делая мир WWW более разнообразным и увлекательным. Тем не менее, это не означает, что веб-мастер должен ограничиваться использованием только данных средств. В этой статье я расскажу еще об одном средстве создания интерактивных веб-страниц — это PHP (Personal Home Page).

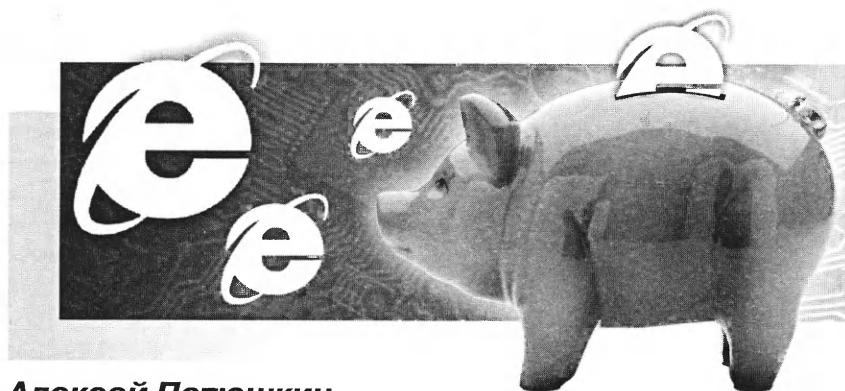
**Язык сценариев PHP**

PHP — это язык серверных скриптов, похожий на JavaScript. Опытный веб-мастер сразу возразит: "Что же общего может быть между этим примитивным JavaScript и PHP?". Действительно, PHP — это межплатформенный язык, действующий на стороне сервера (в отличие от JavaScript), а синтаксис языка по большей части похож на синтаксис Perl и C, да и возможностей для создания интерактивных страниц у него больше. Тем не менее PHP, как и JavaScript, является языком программирования, который встраивается непосредственно в документы

HTML, что заметно упрощает разработку веб-приложений. Таким образом, PHP соединил в себе мощь таких языков программирования, как C и Perl в сочетании с простотой и гибкостью JavaScript.

**Хочу PHP**

Поскольку PHP действует на стороне сервера, для его использования необходим интерпретатор (его можно взять на <http://www.php.net>) и соответствующая настройка сервера, где, в частности, нужно указать расширение для файлов, использу-



Алексей Петюшкин

## Копилка веб-мастера

**М**ногие из вас, дорогие читатели, уже хорошо отдохнули, погрелись на горячем солнышке, набрались свежих сил и полны различных творческих идей и замыслов. В этом выпуске "Копилки" мы постараемся помочь вам в реализации ваших планов и поделимся новой порцией "примочек" для веб-

мастера. Итак, продолжаем знакомиться с интересными и полезными приемами в области веб-дизайна.

### Пользовательские настройки

Допустим, вы создали некий веб-сайт, который оформили в определенной цветовой гамме, по-вашему, наиболее подходящей, не режущей глаз и не раздражающей посетите-

лей. Однако, как известно, сколько людей, столько и мнений, поэтому первый посетитель может быть в восторге от "боевой раскраски" вашего интернет-детища, второй болезненно сморщится, а третий того и гляди сбежит на другой сайт, так как хронически не переваривает фоновый цвет страниц. Возникает классический вопрос: что делать? Как же угодить всем посетителям? В принципе, абсолютно всем угодить невозможно, но кое-что сделать вполне реально с помощью модного сегодня инструментального средства под названием кастомизация.

Слово "кастомизация" (англ. customization) означает процесс, систему действий по сопоставлению определенных объектов конкретным потребительским предпочтениям. Другими словами, это возможность сделать процесс потребления (услуг, товаров) удобным для каждого конкретного пользователя. Самый близкий каждому пример — настройка интерфейсов программного обеспечения.

Данное понятие применимо и к нашей ситуации с цветовой гаммой сайта: необходимо предоставить возможность каждому посетителю быстро поменять цветовые настрой-

ющих PHP-инструкции (обычно это `html`, `php`, `php3` и др.). Итак, интерпретатор установлен, сервер настроен, тогда вперед — напишем свой первый PHP-скрипт.

```
<?php
$ip = $REMOTE_ADDR;
$txt1 = "Добрый день, Хозяин";
$txt2 = "Добро пожаловать,
пользователь сайта";
?>
<html><head>
<title>Мой первый PHP-скрипт</
title>
</head><body>
<?php if ($ip == "127.0.0.1") {echo
$txt1;}
else {echo $txt2; ?><br> ваш IP-
<?php echo $ip; ?>
</body></html>
```

Данный скрипт проверяет IP-адрес посетителя и, если он совпадает с заданным, выводит первое со-

общение, а во всех остальных случаях выводится второе сообщение и IP-адрес пользователя. Этот небольшой пример иллюстрирует следующие основы PHP:

1. Код PHP помещается между тегами `<?php` и `?>`, при этом его можно размещать в любом месте HTML-документа.

2. Имена переменных начинаются со знака доллара (\$).

3. Несколько команд следует разделять точкой с запятой.

4. `echo` является основным оператором языка, с помощью которого выводятся данные.

5. Самое главное: можно в любой момент переключаться между PHP и HTML.

6. Синтаксис языка достаточно прост и очень похож на синтаксис Perl, C и JavaScript.

Важной особенностью PHP явля-

ется возможность включения файлов. Например:

```
<?php
$title="Мой второй PHP-скрипт";
include "head.inc";
?>
<body>А здесь само BODY</
body></html>
```

В файле `head.inc` напишем:

```
<html><head>
<title><?php echo $title; ?>
</title>
</head>
```

Теперь, если вам нужно поменять заголовок документа, например, добавить JavaScript или изменить подключаемый файл CSS, достаточно внести соответствующие изменения только в один файл. Обратите внимание на то, что переменная, установленная во включающем файле, доступна и во включаемом файле.

ки оформления веб-страниц. Одно из наиболее простых решений — встроенный в код каждой страницы инструмент изменения фонового цвета. Реализуется этот инструмент с помощью следующего JavaScript-сценария (вставляется в раздел HEAD):

```
<script language="JavaScript">
<!--
function changeBG(hexNumber){
document.bgColor=hexNumber
}
prefix="#"
rnum1=0
bnum1=0
gnum1=0
rnum2=0
bnum2=0
gnum2=0
hexNumber2="#000000";
rcount=0;
bcount=0;
gcount=0;
function num2hex(num)
{
if (num==15) return "f";
else if (num==14) return "e";
else if (num==13) return "d";
else if (num==12) return "c";
else if (num==11) return "b";
else if (num==10) return "a";
```

```
else if (num==9) return "9";
else if (num==8) return "8";
.....
else if (num==1) return "1";
else return "0";
}
// -->
</script>
```

Теперь в нужном месте веб-страницы (раздел BODY) вставляем элементы переключения цветов и связываем их с указанным выше скриптом с помощью события onClick:

```
<form name="background"
method="post">


```

где "changeBG" — функция смены цвета фона, "#C0C0C0" и др. — шестнадцатеричный цветовой код заднего фона.

Ну вот, теперь посетитель сможет сам выбрать цвет фона по душе и наслаждаться содержанием вашего сайта. Правда, у данного инструмента есть один недостаток: скрипт не запоминает указанную посетителем цветовую настройку и при переходе на следующую страницу загружается цвет, определенный по умолчанию. Устранить этот минус можно при помощи дополнительного скрипта, написанного на языке Perl или PHP (в следующих выпусках "Копилки" я расскажу о том, как это сделать).

### Обязательные поля формы

Бьюсь об заклад, что вы не раз сталкивались с необходимостью обязательного ввода информации, например, при регистрации, анкетировании, голосовании, авторизации, на конференциях, форумах и пр. Иногда это раздражает (мало кто горит желанием сообщать свой почтовый адрес или девичью фамилию своей бабушки), но порой такое требование просто необходимо.

Например, на сайте юридической фирмы организована возможность онлайн-заказа бесплатной консультации и выставлена форма:

дарту отвечают почти все популярные браузеры). Вы можете полностью контролировать процесс загрузки файлов и определять, какие действия должны быть выполнены с файлом после загрузки.

Очень часто PHP используют для интеллектуальной обработки форм. Например, при заполнении анкеты пользователь может пропустить часть важных ее элементов, и тогда после обработки формы ему придется заново вводить все данные, но благодаря PHP можно повторно вывести форму с теми значениями, которые пользователь уже ввел. После того как форма будет обработана, можно, например, средствами PHP организовать аутентификацию зарегистрированного пользователя.

### Что мы имеем

А имеем мы очень мощное сред-

При просмотре содержимого выведенной веб-страницы (Вид —> В виде HTML) вы не увидите ни малейшего упоминания о PHP, что является немаловажным достоинством этого языка, так как обеспечивает дополнительный уровень секретности.

### Что умеет PHP

Итак, мы рассмотрели простейшие примеры использования PHP, но этим, естественно, его возможности не ограничиваются. Как еще можно использовать PHP? Очень часто PHP применяют для работы с базами данных: именно высокий уровень интеграции PHP с различными их типами, наряду с простотой написания веб-страницы, работающей с БД, является неотъемлемым достоинством этого языка серверных скриптов. На сегодняшний день поддерживаются следующие базы

данных: Oracle, Adabas D, Sybase, FilePro, mSQL, Velocis, MySQL, Informix, Solid, dBase, ODBC, Unix dbm, PostgreSQL.

Помимо работы с HTML язык PHP может использоваться для динамического создания файлов GIF или потоков изображений GIF. Для этого необходимо скомпилировать PHP с библиотекой функций изображения — GD. Благодаря этому свойству можно, например, создать красивую диаграмму, иллюстрирующую результат опроса, которая будет динамически изменяться при изменении в голосовании, или свой графический счетчик посещений. В общем, предел применения этой замечательной возможности ограничен лишь вашей фантазией.

Благодаря PHP можно принимать файлы, загруженные из любого браузера, отвечающего стандартам RFC-1867 (на сегодня этому стан-

Ф.И.О., e-mail, статус и суть вопроса. Все поля помечены особым символом, означающим обязательное заполнение (чаще всего это "звездочка"). Если в страницу с формой заказа не вставить специальный сценарий, контролирующий заполнение обязательных полей, к специалистам фирмы может прийти "пустое" письмо: нет имени или электронного адреса, не указан статус или суть задаваемого вопроса. Если не будет указан e-mail, вы не сможете получить ответ; если не сообщите свой социальный статус, консультанту будет гораздо сложнее дать вам исчерпывающий ответ; ну а если вы вдруг "забудете" задать свой вопрос...

От долгого лирического отступления переходим к самому скрипту (раздел HEAD):

```
<script>
function checkrequired (which){
var pass=true
if (document.images){
for (i=0;i<which.length;i++){
var tempobj=which.elements[i]
if (tempobj.name.substring(0,8)=="required"){
if(((tempobj.type=="text"||tempobj.type=="textarea")&&tempobj.value=="")||
```

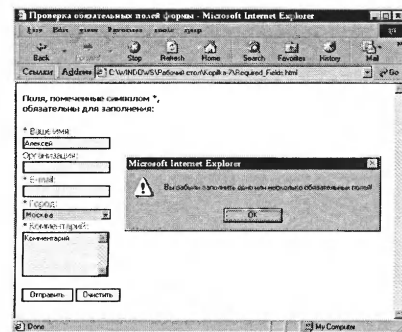
```
(tempobj.type.toString().charAt(0)=="s"&&tempobj.selectedIndex===-1)){
pass=false
break
}
}
}
if (!pass){
alert("Вы забыли заполнить одно или несколько обязательных полей!")
return false
}
else
return true
}
}</script>
```

где "function checkrequired (which)" — функция проверки конкретного элемента формы (в данном примере проверяются текстовые поля TEXT и TEXTAREA, а также ниспадающее меню SELECT); "alert("Вы забыли заполнить одно или несколько обязательных полей!")" — вывод диалогового окна с сообщением о том, что не все обязательные поля заполнены.

Далее необходимо пометить нужные элементы формы как обяза-

тельные для заполнения. Делается это просто: впереди значения атрибута NAME каждого обязательного элемента добавляется приставка required (не забудьте указывать символ \* для таких полей), например:

```
* Ваше имя:<br>
<input type="text" name="requiredname" size="25" maxlength="50">
```

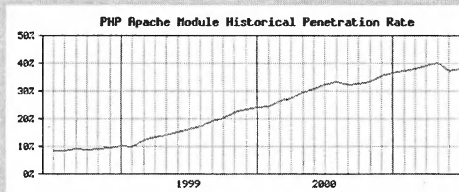


### Сообщение об ошибке

Язык JavaScript на сегодняшний день весьма популярен как у профессиональных разработчиков, так и у начинающих веб-мастеров. Некоторые документы могут включать в себя десятки различных сценариев (взаимосвязанных и автономных), и бывают случаи, когда работа того или иного скрипта дает сбой. Что в итоге имеет пользователь? Простое и лаконичное сообщение об ошибке JavaScript-сценария. Результаты таких ошибок — от мелких сбоев в работе сайта до полного отказа его механизмов. И что же делать в такой ситуации пользователю, особенно неопытному? Предлагаю вам аналог вывода сообщения об ошибке, более приятный по восприятию и исчерпывающий по содержанию:

```
<script>
function JSerror(){
errorwindow=window.open(
"","","width=300,height=200")
errorwindow.document.write(
<title>Сообщение об ошибке JavaScript</title>
<strong>Произошла ошибка JavaScript.</strong><br>
Закройте это окно и попробуйте перезагрузить требуемую страницу. Если ошибка повторится, пожалуйста, сообщите об этом разработчику.
<br><form><input type="button"
```

ство для разработки веб-приложений, которое, несомненно, способно сделать из вашего сайта целое интерактивное веб-приложение, а также существенно облегчить разработку и поддержку сложных проектов, основанных на работе с базами данных и интерактивными элементами. Так что, создавая свой веб-проект, подумайте, возможно, вместо стандартных расширений html или shtml сразу стоит использовать для ваших файлов расширение php или phtml.



Самые ранние версии PHP появились еще в 1994 году. Создатель Расмус Ледорфом использовал их

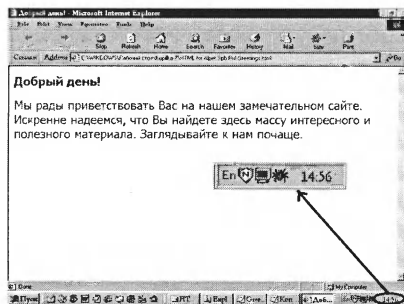
для того, чтобы следить за тем, кто просматривал его личную страничку. Первая версия, состоявшая из простейшего движка синтаксического анализатора, стала доступна в начале 1995 года под названием Personal Home Page Tools. Уже к 1996 году PHP был использован по крайней мере на 15000 веб-сайтах во всем мире. В середине 1997 года эта цифра превысила 50000 сайтов. Стало ясно, что из персонального проекта PHP превратился в серьезный и, самое главное, перспективный продукт. Синтаксический анализатор был заново переписан Зевом Сураски и Анди Гутмансом и стал основой для PHP Версии 3. Сегодня, по данным интернет-агентства Netcraft, рост использования PHP по сравнению с другими языками оказался рекордным!

Алексей Бараношников

```
value="Закрыть окно" onClick=
"window.close()"></form>")
    errorwindow.document.close()
    errorwindow.document.bgColor=
"#FFFFFF"
    return true
}
window.onerror=JSerror
</script>
```

где "function JSerror()" — функция вывода сообщения об ошибке; "errorwindow=window.open("", "", "width=300,height=200")" — настройки окна с сообщением об ошибке: наличие панели инструментов, размеры и т. д.); "errorwindow.document.write" — содержание окна (допускает свободное форматирование, включая таблицы, изображения, CSS и пр.).

Если вставить этот код в каждую страницу, содержащую JavaScript-сценарий, то при возникновении ошибок в работе скриптов появится окно с сообщением об ошибке (там же можно порекомендовать перезагрузку страницы, попросить сообщить об ошибке авторам сайта и пр.)



**Доброе утро, товарищи!**

Персонализированное обращение к пользователю — чрезвычайно важный элемент в общем подходе к разработке интернет-проектов любой сложности и направленности. Следующий пример напрямую с данным понятием не связан, однако во многом определяет отношение посетителей как к сайту в целом, так и к его автору. Речь идет о соответствующем приветствии в зависимости от времени суток. Согласитесь, что фраза типа "Доброе утро, дорогой посетитель!" в 12 часов ночи часто вызывает недоумение и даже ироническую улыбку. Поэтому, если на вашем сайте диалог с посетите-

лем играет важную роль, используйте следующий скрипт:

```
<script language="JavaScript">
<!--
    var time = new Date();
    var hour = time.getHours();
    if(hour >= 0 & hour <= 5)
document.write("Доброй ночи!");
    if(hour >= 6 & hour <= 11)
document.write("Доброе утро!");
    if(hour >= 12 & hour <= 18)
document.write("Добрый день!");
    if(hour >= 19 & hour <= 23)
document.write("Добрый вечер!");
// -->
</script>
```

Код вставляется в тело документа, после чего идет стандартный, неизменяемый работой скрипта текст.

**Соглашение сторон**

Следующий пример будет также исключительно практического плана и в какой-то мере поможет вам уберечь свои или чужие авторские права, если на своем сайте вы распространяете авторские материалы. Например, вы хотите организовать на своем сайте свободное распространение некоего материала, защищенного законом об авторском праве. Но одно дело предупредить посетителей об ответственности, которую они могут понести в случае нарушения упомянутых прав, а другое — иметь прямое доказательство того, что люди, получившие в пользование материал, дали свое согласие на сохранение авторского права.

Для этого нужно дать посетителям возможность ознакомиться с выдержками из "Закона РФ об авторском праве и смежных правах" (или его краткими тезисами), после чего попросить указать, согласны они с ними или нет. Если согласны, будьте любезны — говорите e-mail и ждите

обещанный материал. Не согласны, уж не обессудьте — ввести адрес в поле формы вам не удастся.

Делается это следующим образом. Сначала в раздел HEAD документа вписывается маленький код:

```
<script language="JavaScript">
agree = 0;
</script>
```

Далее идут текст соглашения, элементы согласия/несогласия, поле ввода электронного адреса и кнопка подтверждения:

```
<form name="enableform">
    Ознакомьтесь с нашим
    Соглашением:<br><br>
    <textarea cols="25" rows="7">
    Здесь идет текст соглашения, тезисов Закона или его выдержек...
</textarea>
```

```
<br><br>
    <input type="radio" name=
"enable" value="agree" onClick=
"agree=1; document.
enableform.box.focus();">Я
согласен<br>
```

```
<input type="radio"
name="enable" value='disagree'
onClick="agree=0;
document.enableform.box.value='';">Я
несогласен
```

```
<br><br>
```

Введите E-mail, на который<br>будет выслан материал:

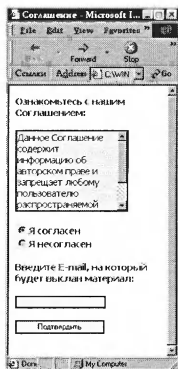
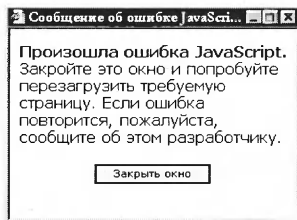
```
<br><br>
```

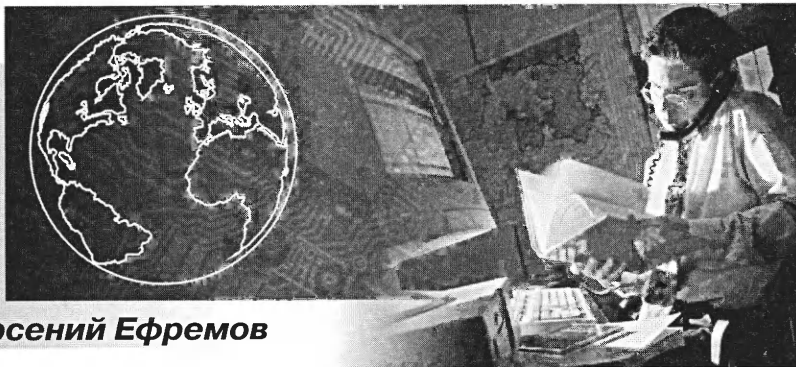
```
<input type="text"
name="box" onFocus="if
(!agree)this.blur();"
onChange="if
(!agree)this.value='';" size="25">
```

```
<br><br>
<input type="submit"
value="Подтвердить">
</form>
```

где name="enableform" — атрибут NAME и его значение (обязательны!); "document.enableform.box.focus()" — становление фокуса формы на поле ввода e-mail (при согласии); document.enableform.box.value="" — блокировка поля ввода e-mail (при несогласии).

На этом пока все. До встречи в следующем месяце (полные листинги и примеры этого выпуска можно найти на <http://www.alpet.spb.ru/surveys/kopilka7.shtml>).





Арсений Ефремов

# ADSL сегодня

**В** 1989 году человечеству пришла в голову замечательная идея, как использовать телефонные провода для передачи данных. Казалось бы, что тут нового? Обычные модемы давно принимают и отправляют данные по этим самым линиям. Но изюминка изобретения состояла в том, чтобы выделить для передачи данных отдельные частоты.

Принципы телефонии основаны на технике частотного уплотнения каналов (FDM, Frequency Division Multiplexing). Речевые сигналы имеют спектр шириной примерно 10000 Гц, однако основные гармоники укладываются в диапазон от 300 до 3400 Гц. Следовательно, для качественной передачи речи достаточно канала связи с полосой пропускания 3100 Гц. Сигналы от абонентов телефонной сети поступают на входы FDM-коммутатора, который переносит частоты каждого канала в свой диапазон по всей полосе пропускания общего канала. На первом уровне уплотнения 12 сигналов от абонентов уплотняют по частотам для передачи по одному каналу связи и так далее, до уровня каналов междугородной и международной связи.

Но это уплотнение происходит уже на телефонном узле, которые связаны между собой каналами свя-

зи. А линия, ведущая к абоненту от телефонной станции (где происходит уплотнение первого уровня) задействована только на низкой частоте 4 кГц. Используется именно 4 кГц, которые взяты с запасом в 900 Гц от наложения частот друг на друга при модуляции. Получается, что возможности обычной телефонной линии используются лишь на 1%.

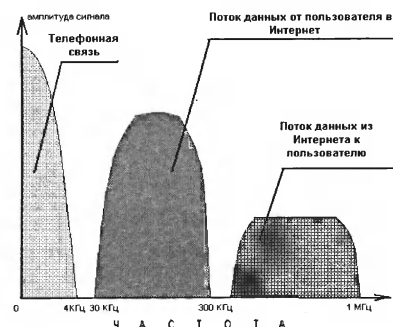
Потенциально частота передачи сигнала зависит прежде всего от оборудования, которым этот сигнал передается и принимается. Выше частота, выше скорость, выше стоимость. По этим критериям полоса пропускания 1 МГц считается сегодня оптимальной. Чтобы волшебным образом преобразить телефонные провода в конкурента выделенного канала с пропускной способностью 1 Мб понадобится дополнительное оборудование — специальные ADSL-модемы и не только. В результате — значительный выигрыш по скорости в сравнении с обыкновенными модемами, всегда свободный телефон и постоянное соединение a-la выделенный канал.

Открываются розовые перспективы просмотра видеофильмов прямо в Интернете, выкачивания мегабайтов музыки, телеконференций. И все это без дорогой прокладки отдельного кабеля. По прогнозам западных аналитиков, к 2004 году в

США 36% пользователей будут предпочитать ADSL, лишь 12% отдадут предпочтение ISDN, 28% будут рады своим кабельным модемам и 8% будут получать Интернет со спутников. Согласитесь, неплохо для технологии, реально увидевшей свет и опробованной на практике лишь в 1998 году. Но все это теория, слова, которые можно прочитать в рекламных проспектах провайдеров. Информация, конечно, интересная, но не слишком подробная.

## Технические нюансы

Начнем с самого простого. ADSL расшифровывается как Asymmetric Digital Subscriber Line — асимметричная цифровая абонентская линия. Возникает первый вопрос: в чем асимметрия ADSL? Оказывается, для максимально эффективной работы реализованы различные скорости передачи данных от пользователя в Интернет и из Интернета к пользователю. Решение разумное — многие замечали, что их модем передает в Сеть куда меньше данных, чем принимает из нее. Ширина соответствующей частотной полосы много больше, чем частотный диапазон аналоговой телефонной линии, где в оба направления трудится обыкновенный модем. На рисунке приведено деление частот в пределах 1 МГц на обычную телефонную связь и потребности ADSL.



Таким образом, благодаря частотному уплотнению линии связи задействуется вся полоса частот. В каждом потоке данных может быть несколько каналов передачи данных. Иногда для увеличения скорости соединения применяют еще и эхокомпенсацию. В этом случае частоты

потоков от пользователя к Интернету и от Интернета к пользователю перекрываются и разделяются уже на стороне пользователя специальными средствами эккомпенсации. Такой прием позволяет расширить полосу частот и, следовательно, скорость передачи данных. Увы, подобное частотное разделение требует много большего, чем один простой модем.

### Сага о девайсах

Попытаемся представить себе механизм работы системы ADSL. Итак, произошло разделение между обычной телефонной связью и передачей данных. Разделение это должно проследиваться и на стороне отправки данных (то есть на АТС, откуда передаются данные на ваш телефон), и у конечного пользователя. Данные от одного из Интернет-серверов проходят по оптоволоконным магистралям и попадают на специальное коммутационное оборудование. Это оборудование провайдера, обеспечивающего ADSL-доступ. После этого потоки направляются в сети ADSL, связывающие оптоволоконно с телефонными сетями.

На телефонной станции тот же самый провайдер должен установить стационарное оборудование ADSL (фактически это модем), которое "запаковывает" данные в частоты городской телефонной сети, оставляя пользователям возможность вести обычные телефонные разговоры. Дальше — телефонный коммутатор на ADSL-модем пользователя. Именно к нему и будут подключены ваш телефон и компьютер. Остальное очевидно: телефонные разговоры "в телефон", данные из Сети — в компьютер. Итак, мы насчитали на пути следования два новых устройства — стационарное оборудование ADSL (иногда его еще называют провайдерскими модемами) и клиентские модемы ADSL.

Начнем с первого. Это серьезные многоканальные устройства, выпускающиеся в промышленном исполнении. Обычно строятся на интерфейсе Ethernet на 100 или 1000 Мбит, имеют встроенные маршрути-

заторы. Заблуждением будет считать, что ADSL-сигнал проходит по телефонным проводам без дополнительного оборудования (вся аппаратура АТС работает на низких частотах), хотя абсолютно верно, что линия, на которой работает ADSL, доступна для телефонии. Важная составляющая стационарного оборудования ADSL — частотный разделитель. Его часто встраивают в ADSL-модем. Частотный разделитель должен уметь различать полосы частот телефонных разговоров и поток данных в разных направлениях и модулировать данные. Фактически это обычный трехпортовый фильтр частот. Вернее, два фильтра — низкочастотный для телефонии и высокочастотный для данных.

Точно такой же частотный разделитель должен быть установлен и на стороне пользователя. Кстати, здесь лучше отдать предпочтение пассивным разделителям, поскольку активные, встроенные в модем, будут зависеть от электропитания, и при аварии вы лишитесь телефона. Пассивные же разделители подведут сигнал к телефонному аппарату и при неисправном модеме. Хороший частотный фильтр должен отсеивать импульсные помехи, создаваемые телефоном и АТС. Но создаются и обратные помехи — высокочастотные, сказывающиеся на телефонном соединении. Поэтому при выборе ADSL-оборудования стоит уделить частотному фильтру особое внимание. Он не должен иметь слишком низкую граничную частоту, которая может напрочь отсечь возможность передачи факсимильных сообщений.

Исходя из сказанного, не стоит паниковать при ухудшении качества телефонной связи. Вызовите специалиста, который даст совет по изменению той или иной характеристики разделителя. Впрочем, уважаемые провайдеры должны сами предоставить вам наилучший разделитель, исходя из параметров линии связи. Заметим, что устройства, использующие уплотнение частот, такие как охранная сигнализация, блокираторы, АБУ, ВЧ/НЧ, работать совместно с ADSL не будут. Необходимо также понимать, что клиентский ADSL-мо-

дем и провайдерский — совершенно разные устройства. Два клиентских модема не способны установить между собой связь. Самые дешевые версии клиентских модемов — подключаемые через порт USB или просто внутренние. Нужна и интерфейсная плата Ethernet 10baseT.

### Что может быть лучше ADSL?

Теоретический предел скорости передачи данных от Сети к абоненту — до 8 Мбит/с, обратно — до 1 Мбит/с. Неплохие цифры, особенно если сравнить их с другими средствами связи. Обычный модем поставлен в пределы 56 Кбит/с. Выделенный канал — 64 Кбит/с. Даже если их два, то это 128 Кбит/с. Спутниковый вид связи работает на скорости от 157 Кбит/с до 1 Мбит/с. Кабельные модемы развивают скорость до 10 Мбит/с. Где-то рядом находятся максимальные мощности радиоинтернета — 11 Мбит/с.

Но... Все это с оговоркой "теоретически". А на практике максимальная скорость передачи данных зависит от таких факторов, как состояние телефонной линии, ее протяженность, диаметр проводов, наличие кабельных отводов, термических катушек. Длина линии связи не должна превышать 5,5 км, но желательно и не меньше 2,5 км, а толщина провода не менее 0,5 мм, иначе происходит искажение сигнала. Например, при превышении длины линии высокочастотный сигнал ослабевает на 90 дБ. Важна даже система изоляции, которой оснащена телефонная линия: бумага может внести серьезные задержки в работу частотного разделителя.

А теперь вспомните, в каком году прокладывались наши телефонные сети. Один участок провода проложен при царе Горохе, второй позавчера привезли из Германии. И все они разного диаметра и длины. Именно эта лоскутная система пагубно сказывается на качестве связи. Провода разного диаметра могут не повлиять на низкие частоты телефона, но при этом напрочь убить высокие частоты передачи данных.

Ясно, что оговорка провайдеров "максимальная скорость передачи данных не ограничивается, но зависит от качества абонентской линии и загрузки сети ADSL" существенно снижает пределы трафика. Хотя, допустим, "Россия-Он-Лайн" гарантирует и минимальную скорость передачи данных: 128 Кбит/с из Интернета и 64 Кбит/с в Интернет. Это скорость выделенной линии.

Кажется, что по своей сути ADSL ориентирован на домашних пользователей. Этакими студентами и школьниками, постоянно вступающими в конфликт с родителями из-за занятого модемом телефона. Но цена на ADSL пока явно выше возможностей этого контингента. Поэтому российские ADSL-провайдеры ориентируются на небольшие компании, которым необходим Интернет. Выигрыш достигается в том, что не требуется никаких коммутационных работ, так как используются обычные телефонные провода. Добавьте к этому полную совместимость ADSL с самыми старыми моделями мини-АТС, хорошие возможности по работе в сети.

### ADSL двух столиц

Пока ADSL для России новинка. В столице она появилась в мае прошлого года, в Санкт-Петербурге — в начале 2001-го. Состояние рынка сегодня можно охарактеризовать простым словом "монополия". А как иначе, ведь мало приобрести оборудование и установить его. Провайдер должен предоставлять своим пользователям услуги телефонии. Понятно, что такими организациями в столицах являются МГТС и ПТС.

В Москве ADSL-доступ предоставляет компания "ПТТ Телепорт" (проект точка.ру), являющаяся в свою очередь... дочерним предприятием АО МГТС. Можно предположить, что монополия будет держаться долго, ибо она выгодна самим телефонистам. Тем не менее, уже около сотни телефонных узлов оснащены современным ADSL-оборудованием. А вот цены, которые предлагает точка.ру, явно ориентированы на корпоративного клиента. Всего существует четыре тарифных плана, но

в любом случае прежде всего — установка и подключение оборудования стоимостью 750 американских рублей. Взимается ежемесячная абонентская плата от \$60 до \$270 в зависимости от лимита бесплатного трафика (от полной оплаты по \$0,16 за мегабайт до оплаченного трафика 2 Гб). Цифры просто пугающие.

Но в Петербурге провайдерам "Web Plus" и "Россия-Он-Лайн" удалось установить ADSL-оборудование на городские АТС. Правда, до сих пор остается загадкой то соглашение, которое они между собой заключили. На презентации ADSL, устроенной "Россия-Он-Лайн", представители компании так и не смогли дать вразумительный ответ на вопрос о границах сотрудничества с коллегами из "Web Plus". Есть предположение, что это все дело рук "Телекоминвест", которому принадлежит "Web Plus" и треть акций "ПетерСтар" (эта компания предоставила свои АТМ-коммутаторы).

Работы по установке ADSL в Питере велись с февраля 2000 года и, по непроверенным данным, первый этап обошелся инвесторам в 1,8 миллиона долларов. И, тем не менее, тарифы петербургских компаний выглядят значительно привлекательнее. "Россия-Он-Лайн" предлагает всего два тарифа: "Оптима" и "Профи". "Оптима" рассчитана на 7 постоянно работающих в Сети пользователей, трафик не оплачивается, а заменяется ежемесячной абонентской платой в размере \$59. Скорость передачи данных — 64 Кбит/с и 16 Кбит/с. Вариант "Профи" выглядит серьезнее, у него по соответствующим позициям: \$299, 128 Кбит/с и 64 Кбит/с. Рассчитан он на 15 пользователей.

Но главное, в менее денежном Питере провайдеры не продают ADSL-оборудование пользователям, а сдают его в аренду — на время пользования услугой и всего за \$199. Сравните это с 750 московскими! "Web Plus" пошел по несколько иному пути: снизил абонентскую плату и установил лимит по трафику. "Оптима" и "Профи" у него соответствуют "Домашний" и "Бизнес". При тех же скоростях и стоимости под-

ключения в месяц придется платить всего 29 и 99 у. е., но в "Домашнем" тарифе за месяц можно извлечь из Сети 500 Мб данных, оплачивая каждый последующий по 8 центов, а в "Бизнес" это 1000 Мб и 6 центов.

Вот так идет становление высоких технологий доступа к Интернету в России. В Екатеринбурге ADSL налаживает ОАО "Уралтелеком". "Россия-Он-Лайн" намерена развить эту услугу в Нижнем Новгороде. Аналитики считают, что массовый рынок ADSL в России сформируется лишь к 2003 году. О количестве своих клиентов компании умалчивают. Первоначально сообщалось, что питерское оборудование рассчитано на 2640 клиентов. Не окажутся две тысячи шестьсот сорок первым!

Для сравнения приведу тарифы на ADSL в Великобритании. Там две крупнейшие корпорации Telewest и British Telecommunications начали ценовую войну. Первая снизила абонентскую плату до 33 фунтов стерлингов (около \$50). Клиенты British Telecommunications пока платят 50 фунтов. Конечно, \$50 для англичанина — это даже не 29 для россиянина. Но совершенно ясно, что нам предлагают эту услугу по мировым ценам и даже ниже.

Стоит ли ADSL этих денег? Безусловно, по цене это следующая ступенька после обычного модемного соединения, а вот по качеству — существенно отличается в лучшую сторону. Новинка должна быстро обрести своих почитателей. Это люди, для которых скорость имеет значение. Если вы из их числа, не желаете довольствоваться малым и готовы платить за качество, то ADSL может стать вашим выбором.



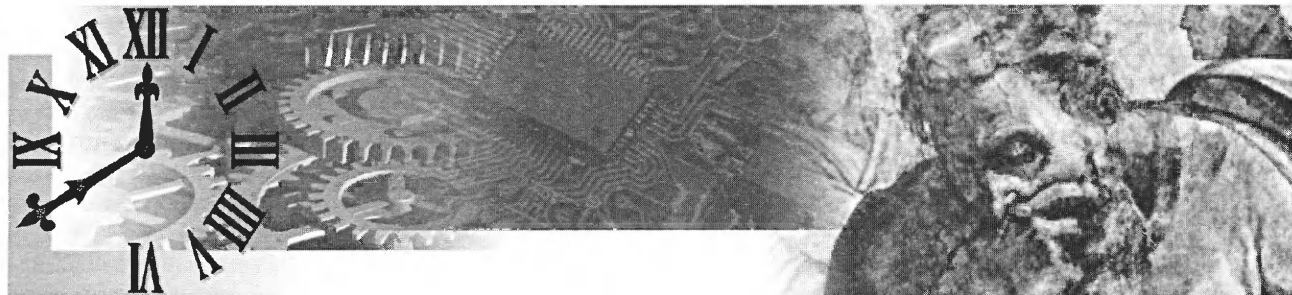
магия  
ПК

"Магия ПК"  
- в сети  
Интернет

- <http://www.magicpc.spb.ru>



Продолжение. Начало см. "Магия ПК" №1/2000



# Сто выдающихся деятелей IT ушедшего столетия

## Нейман Джон фон (von Neumann, John, 1903—1957)

Американский математик, оставивший значительный след в математике, экономике, физике, кибернетике, лингвистике. Уроженец Будапешта (сын будапештского банкира). Уже в восьмилетнем возрасте владел основами высшей математики и несколькими иностранными языками. Закончив Будапештский университет (1926), преподавал в Германии, а в 1930 г. эмигрировал в США и стал сотрудником Принстонского института перспективных исследований. В 1928 г. сформулировал основы теории игр, ныне широко применяемой в машинном моделировании сложных ситуаций, позже выдвинул идею создания математической машины, способной реализовать некоторые принципы этой теории. В 40-х годах входил в Манхеттенскую группу, работавшую над созданием первой атомной бомбы. В 1940 г. руководил разработкой ЭВМ MANIAC (Mathematical Analyzer Numerical and Computer). В 1944 г. в соавторстве с экономистом О. Моргенштерном написал книгу "Теория игр и экономическое поведение". В книге опи-



сана не только математическая теория игр, но и возможности ее применения к экономическим, военным и другим наукам. В январе 1945 г. присоединился к группе разработчиков ENIAC и на основе критического анализа ее конструкции в 1946 г. в предварительном докладе о машине EDVAC (Electronic Discrete Variable Computer) описал устройство будущего компьютера, предложив ряд новых идей организации ЭВМ, в том числе концепцию хранимой программы (хранения программы в запоминающем устройстве). В результате реализации этих идей была создана архитектура ЭВМ, во многих чертах сохранившаяся до настоящего времени — так называемая "архитектура фон Неймана". По окончании работ над EDVAC фон Нейман вернулся в Манхеттенскую группу, но не оставлял идеи об использовании ЭВМ как универсального инструмента для решения сложных математических и логических задач. Незадолго до смерти выпустил книгу, в которой показал, что ЭВМ не только позволяет эффективно решать сложные математические задачи, но и стимулирует дальнейшее развитие самой математики. В 1957 г. покончил жизнь самоубийством. Уже после смерти фон Неймана по оставшимся после него рукописям и на броскам А. Беркс выпустил книгу "Теория самовоспроизво-

дующих автоматов", послужившую основой для возникновения нового научного направления — теории клеточных автоматов, переросшей позже в теорию однородных структур. Эта теория лежит в основе создания ЭВМ с новой архитектурой (нейрокомпьютеров, сетевых машин и ряда других).

## Нойс Роберт (Noyce Robert Norton, 1927—1990)

Американский инженер, изобретатель интегральной схемы (1959), один из основателей корпорации Intel. Уроженец штата Айова, в 1949 г. окончил местный колледж со степенью бакалавра, а в 1953 г. стал доктором философии Массачусетского технологического института. В 1956—1957 годах работал в полупроводниковой лаборатории изобретателя транзисторов Уильяма Шокли, а затем вместе с коллегами уволился и основал Fairchild Semiconductor, одну из первых фирм по производству кремниевых полупроводников. Одновременно (но независимо) с Дж. Килби изобрел интегральную микросхему. В 1958 г. разработал промышленную технологию тиражирования при производстве интегральных схем (запатентована в 1959 г.),



что дало старт гонке за уменьшение размеров активных элементов микросхем и увеличение плотности интеграции их активных элементов. В 1968 г. вместе с давним коллегой Г. Муром основал корпорацию Intel. Спустя два года они создали микросхему из кремния и поликремния, которая заменила прежние малоэффективные керамические сердечники в запоминающих устройствах компьютеров. В 1971 г. Intel представила микропроцессор, объединяющий в одной микросхеме функции запоминающего устройства и процессора. Вскоре корпорация стала лидером по производству микропроцессоров. В 1980 г. Нойс был награжден Национальной медалью за научные достижения. В 1988 г. стал президентом корпорации Sematech, исследовательского консорциума, финансируемого промышленным капиталом и правительством США и занимающегося развитием передовых технологий в полупроводниковой промышленности. Скончался 3 июня 1990 г. от инфаркта.

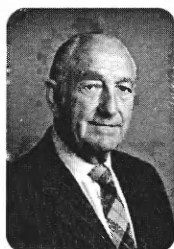
### **Нортон Питер (Norton, Peter, p. 1952)**



Американский программист, бакалавр математики. Окончил Ридоновский колледж (Портленд, шт. Орегон), затем Калифорнийский университет в Беркли. Программированием занимался с 1969 года. В 1981 г. приобрел один из только что появившихся тогда в продаже персональных компьютеров. В 1982 г. уволился из аэрокосмической промышленности и основал фирму Peter Norton Computing, которая стала заниматься поставкой обслуживающих программ (утилит) для IBM-совместимых ПК. Капитал фирмы составлял 30 тыс. долларов, штат состоял из одного человека — основателя. Первым и самым популярным ее продуктом был пакет Norton Utilities, позволявший восстанавливать потерянные данные. Через некоторое время

имя Нортон стало известно миллионам пользователей. К концу 1982 г. он продал утилит на \$68000, в 1983 г. на \$600000, а в 1984 г. — более чем на 1 млн долларов. Со временем в пакет были добавлены и другие функции — контроль качества дисков, повышение скорости работы с ними, тестирование компьютера, настройка параметров Windows и др. В 1986 г. фирма выпустила на рынок операционную оболочку Norton Commander (ее автор с 1-й по 3-ю версии — программист Джон Соча), и вскоре "синий экран" покорила мир. К 1990 г. фирма насчитывала свыше сотни сотрудников, имела годовой оборот 15 млн долларов. В том же году Peter Norton Computing объединилась с компанией Symantec. Компания получила право использовать имя Нортон в будущих продуктах, а он получал 1% с каждой продажи. Сегодня общее число легальных пользователей Norton Utilities в мире превышает 5 млн человек, нелегальных — еще 40 миллионов. Помимо прочего Norton Utilities применялись для восстановления потерявшихся секретных данных в ЦРУ и для восстановления запорченных данных, моделирующих ситуацию на Чернобыльской АЭС. Общий тираж книг Нортон, переведенных на 14 языков, включая русский, составил более 1 млн экземпляров. Разбогатев, Нортон стал меценатом. Каждый год он выделяет примерно \$350000 на поддержку искусства, образования и борьбу со СПИДом, а также \$50000 различным музеям США на закупку образцов современного искусства. Журнал ARTNEWS включил Нортон в список 200 крупнейших коллекционеров мира.

### **Паккард Дэвид (Packard, David, 1912—1996)**



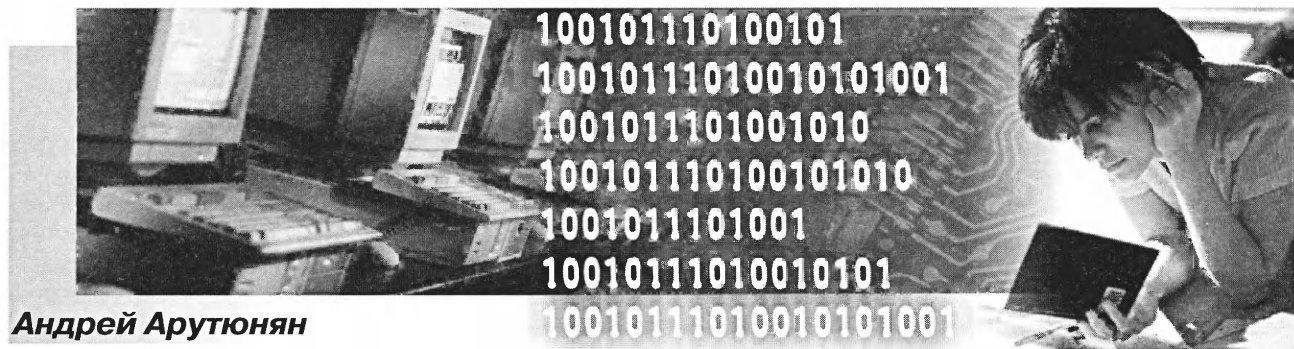
Уроженец штата Колорадо, по окончании Стэнфордского университета в 1936—1938 гг. работал инженером в компании General Electric, а в 1938 г. вместе со своим однокашником по

университету У. Хьюлеттом создал в Пало-Альто (шт. Калифорния) компьютерную компанию Hewlett-Packard, ныне известную благодаря своим принтерам. На момент основания фирма имела капитал размером \$538 и размещалась в небольшом гараже. В 1947—1964 гг. Паккард занимал должность президента фирмы, в 1964—1968 и 1972—1993 гг. был председателем правления, в 1993—1996 гг. — почетный председатель в отставке. В 1968 г. назначен министром обороны США, занимал эту должность до 1971 г., после чего вернулся в родную компанию. Возглавлял многие общественные фонды США.

### **Патерсон Тим (Pateson, Tim, p. 1956)**



Родоначальник дисковой операционной системы DOS. По окончании школы в Сиэтле (США) работал техником в компьютерной фирме Seattle Computer Products, вскоре начал проектировать собственные платы. В июне 1978 г. закончил университет со степенью бакалавра и уехал на семинар для ознакомления с первыми процессорами Intel 8086, а в апреле 1979 г. приступил к работе над операционной системой для компьютеров, оснащенных новым процессором. Хотя операционная система Патерсона, названная QDOS (быстрая операционная система), была в значительной степени основана на CP/M, имелись и существенные различия, прежде всего в управлении файлами. Несколько месяцев спустя Microsoft приобрела права на QDOS и на ее базе начала разработку своей операционной системы PC 86-DOS, к работе над которой в апреле 1981 г. подключился и Патерсон, ставший к тому времени штатным сотрудником Microsoft. А течение следующих полутора лет он принимал участие в создании MS DOS версий 1.1 и 2.0. В 1982 г. вернулся в свою прежнюю фирму, но в начале 90-х годов опять перешел в Microsoft.



Андрей Арутюнян

# Кризис технологии потребления информации

*Предоставлять информацию человечество уже научилось. Осталось научиться потреблять ее.*

**Н**ачну с тезиса, который многим может показаться спорным или, по крайней мере, неожиданным. Современное развитие ИТ идет прежде всего в направлении технологий передачи данных, тогда как технологии потребления информации значительно отстают в развитии.

## Предложение информации

Все мы наблюдаем стремительный рост пропускной способности линий связи. Если всего несколько лет назад 28800 Кбит/с были вполне удовлетворительной величиной, то потребности многих индивидуальных пользователей сегодня уже далеко перешагнули за 128 Кбит/с. И технологии, предоставляющие такую скорость соединения, уже доступны многим.

Пропускная способность оптоволоконных линий последнее время удваивается за период от полугода до года. В частности, активно обсуждаются перспективы революционного скачка технологий передачи данных за счет внедрения полого оптического волокна с зеркальной поверхностью, что, по заявлению авторов технологии, позволит увеличить пропускную способность в сотни и тысячи раз при снижении затрат на промежуточные устройства усиле-

ния сигнала, стоимость которых при значительной длине линий составляет десятки миллионов долларов.

Не отстает по темпам и рост объемов накопителей информации.

Так что же будет передаваться по сверхмощным каналам связи завтрашнего дня и храниться на сверхъёмких накопителях? На сегодня выбор не так уж велик. Несомненно, будут востребованы услуги видеосвязи, в первую очередь в сферах образования и территориально распределенной работы, телевидение "по требованию", но что еще?

Естественно, спрос рождает предложение, но в области информационных технологий, похоже, в настоящее время предложение очень уж лихо обгоняет спрос. Характерный пример — ставшие чуть ли не эталоном престижности приложения к печатным журналам в виде лазерных дисков, зачастую набитые откровенным хламом или заполненные всего на треть от фактической емкости носителя.

Таким образом, уже начал ощущаться некоторый уровень насыщения, за которым рост ИТ значительно замедлится. Или изменит вектор развития. Какое-то время этот дисбаланс будет компенсировать естественное стремление людей копить информацию впрок (запас не ноша, рук не оттянет), но долго этот "период накопления" продлится не мо-

жет. Избыточность информации рано или поздно перерастет в отдельную проблему.

## Потребление информации

Однако не этот дисбаланс — самое страшное. Страшно то, что при очевидном взрывном росте технологий создания, хранения и передачи информации практически не изменилась технология ее потребления: остается тем же время, необходимое человеку для прочтения и усвоения всех тех гигантских объемов информации, к которым он может получить доступ. Хуже того, практически никаких технических решений этой проблемы и не предлагается. Если и существуют технологии так называемого управления знаниями, направленные на корпоративный уровень, очень громоздкие и дорогие, то для индивидуального пользователя, к сожалению, они неприменимы.

Еще менее управляемой величиной остается способность человека усваивать информацию. Технологий, меняющих способность к восприятию и усвоению информации, тоже немного. К тому же большинство из них требует специальной обработки информации (25-й кадр, метод Эланы Давыдовой). Естественно, обрабатывать таким образом массу информации в Сети никто не будет: и дорого, и долго, и нет единых стан-

дартов, да и весомых доказательств безопасности для здоровья пока тоже нет.

Лишь частично решению проблемы создания новых технологий потребления информации и ее производства помогают всевозможные мобильные технологии — мобильная связь, мобильный интернет-доступ, всевозможные PDA, все уменьшающиеся по весу и цене портативные компьютеры. Роль, которую они выполняют, практически сводится к доступу и хранению информации. А много ли толку в том, что в вашем ноутбуке, с которым вы не расстаетесь ни на секунду, хранятся десятки тысяч книг, если вы не успеваете за месяц прочесть даже две из них?

Чтение как технология потребления информации, использующаяся уже несколько тысяч лет, увы, безнадежно устарела. Технологии быстрого чтения также не вполне применимы для восприятия информации из Сети, так как в ней быстро растет доля мультимедийной информации. Аудио- и видеоинформация, несомненно, гораздо удобнее для восприятия, однако не стоит их переоценивать. Технологии видеозаписи могут использоваться только в отношении очень узкого спектра информации. Ведь письмо и чтение — прежде всего способ передачи мыслей человека, а видеозапись и телевидение — лишь средство передачи видимого отображения, и то только с определенной точкой.

Современные технологии подготовки информации также меняются крайне медленно. Если на подготовку информации и далее будет затрачиваться месяц, а передаваться она будет за 1 секунду, то налицо еще один явный дисбаланс. И экономически оправданной деятельностью по созданию или подготовке информации будет только в редких случаях.

Наконец, очень существенная проблема — деградация качества информации. Стремительный рост ее количества (число документов, доступных во Всемирной сети, по некоторым оценкам достигает десятков миллиардов и растет лавинообразно), увы, не сопровождается качественным ростом, а поэтому

эффективность работы с информацией опять-таки быстро падает. Сегодня внимания к качеству информации в Интернете уделяется даже меньше, чем несколько лет назад. Устаревшие и не представляющие практической ценности материалы приобретают все больший удельный вес (я лично к мусору отношу информацию по критерию "4 НЕ": невостребованную, неактуальную, не достоверную и некачественную). При этом они исправно индексируются поисковыми системами, заметно снижая скорость поиска информации действительно ценной и нужной пользователям.

Результаты исследований, проведенных Европейской Комиссией, впечатляют: затраты, которые несут пользователи только на чтение и удаление ненужной почты, составляют порядка 10 миллиардов евро в год.

Неудивительно, что ценность Интернета как источника знаний быстрыми темпами падает. Многие организации и частные лица сокращают использование Интернета, в первую очередь, не желая мириться с огромными потерями времени на поиск информации.

Это ударит прежде всего по тем, кто сегодня проявляет полнейшее равнодушие к проблемам качественного контента — крупнейшим провайдерам, фирмам, предоставляющим услуги хостинга, одним словом, тем, кто получает львиную долю прибылей от интернет-технологий. Решение проблемы качества информации, размещаемой в Интернете, возможно только в результате консолидации усилий основных участников этого рынка, что в данный момент практически нереализуемо — слишком уж различаются сферы их интересов.

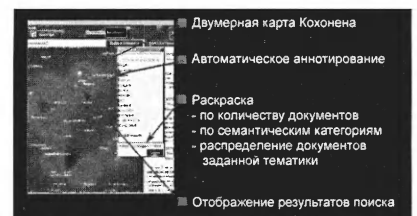
Так что же, кризис интернет-индустрии, первые признаки которого проявились в начале этого года, неизбежен?

Надо полагать, кризис не только неизбежен, но и в некоторой степени необходим, ведь именно в кризисных условиях рождаются действительно революционные технологии и решения, способные вывести на новый уровень все общество в целом.

## Главный человеческий ресурс — это время

Какие же сегодня существуют технологии, направленные на снижение затрат времени при потреблении информации? Если создание известных всем поисковых систем, основанных на индексации документов в Сети, на какой-то период и смогло снять проблему чрезмерных затрат времени на поиск документа, то сегодня, увы, они практически не удовлетворяют потребностям пользователей. По некоторым оценкам, до 75% имеющихся в Интернете документов не подвергаются индексации, и эта "terra incognita" ширится с каждым днем. К тому же список из нескольких тысяч найденных в результате запроса документов оставляет мало надежды найти нужный в течение короткого времени.

Весьма интересны в этом плане разработки российской фирмы "НейрОК" (<http://www.neurok.ru/>), в частности, программный продукт под названием "Semantic Explorer". Его отличительная черта в том, что навигация осуществляется в соответствии с семантикой документов и поиск возможен по смысловым и тематическим ассоциациям. Основным отличием от традиционных поисковых систем является векторное представление документов.



Интерфейс пользователя программного комплекса Semantic Explorer

Для приближения навигации в массивах документов к способностям восприятия человека используются двумерные самоорганизующиеся карты (карты Кохонена), что позволяет сохранять локальную топологию векторного пространства: близость документов на карте подразумевает их близость в исходном многомерном пространстве. Основан продукт на нейротехнологии.

Самоорганизующиеся карты строятся путем обучения нейросети, в которой узлы карты представлены отдельными связанными между собой нейронами.

В результате работы с базой данных документов пользователь получает "галактическую" карту этой базы данных, на которой каждый документ имеет свое уникальное положение. Причем близкие по смыслу документы располагаются рядом. Насколько это удобно, поймет не только тот, для кого поиск информации — основная работа, но и каждый, кто провел десятки часов в поисках крупиц информации.

Однако поиск документа — это еще не все затраты времени на работу с ним. Мне довелось познакомиться с несколькими разработками, направленными на активизацию восприятия, в частности, с продуктами фирмы MIMEX. Надо признаться, впечатления, что это прорыв в технологиях восприятия информации, у меня не было. Не секрет, что техно-

логии такого рода входят в сферу интересов разнообразных силовых ведомств, отчасти этим объясняется периодические слухи, что очередное "закрытое" КБ достигло прорыва в данном направлении. К сожалению, информация эта так и остается на уровне слухов.

Потребности рынка в передовых технологиях восприятия информации в настоящий момент огромны, и, как уже показала новейшая история ИТ, те компании, которые первыми выйдут на рынок с работоспособными продуктами, в корне меняющими нынешние технологии, не только с лихвой окупят затраты на исследования, но и получат огромные прибыли. Это оставляет надежду на то, что в относительно близкое время мы все же сможем потреблять гораздо больше информации за меньшее время. А значит, больше времени появится для общения с друзьями, спорта, рыбалки. Человечество готово хорошо заплатить за это. Вот только пока некому.

Современный программный интерфейс, конечно, проделал в своем развитии громадный путь. И это вызывает восхищение, особенно у тех, кто помнит интерфейсы первых лет, когда компьютер не был рассчитан на частное лицо, пользователя. Интерфейсы первых компьютеров были местом встречи не "двух лиц", а двух машин. А программист или оператор имел возможность вмешаться в это общение, что-то подправить так, чтобы это общение шло нужным человеку образом. Иначе говоря, исходный интерфейс компьютера — машинный.

Затем, с появлением DOS, Хортона и Окна, хозяину становилось все легче вмешиваться в работу машины. И все же интерфейс оставался на уровне интерактивного листка из записной книжки оператора. Но и сейчас это в большинстве случаев таблица с кнопками. Прорыв за рамки таблиц был сделан в мультимедийных программах, где интерфейсы имитируют настоящие приборы, в играх и рекламе. Однако в целом отношение к интерфейсам до сих пор несерьезное.

### Чем мы думаем

Все табличные интерфейсы каким-то образом воздействуют на человеческое сознание, меняют его. Это многие замечают, но природа воздействия остается скрытой. Можно много говорить о необходимости нового подхода к интерфейсам, но пока природа воздействия интерфейса на сознание человека не будет понятна, все новшества останутся одной лишь косметикой. Суть проблемы явно лежит вне программирования и дизайна. Она в психологии и физиологии человека, и именно эту возможность просмотрели творцы современных интерфейсов. Между тем, только на базе знания психофизиологии восприятия интерфейс можно сделать не "дружественным", а простым и человеческим.

Для начала — эксперимент. Откройте на экране монитора любой табличный интерфейс, которым ча-

## Хэлло, Долли!

### Опыт поздравительной эпитафии

*Едва ли не на следующий день после решения правительства РФ о запрете клонирования живых организмов начались эксперименты по клонированию российских компьютерных журналов.*

**Ж**урнал "Домашний компьютер" за последние два года достиг уровня почти идеала при традиционной компоновке (подвожу к выводу: важна не форма, а содержание). От первой до последней страницы — Стиль с большой буквы. К сожалению, неизвестно, какой ценой достигалось совершенство, — со ссылкой на решение издателя в восьмом номере нам сообщили о практически полной замене коллектива редакции.

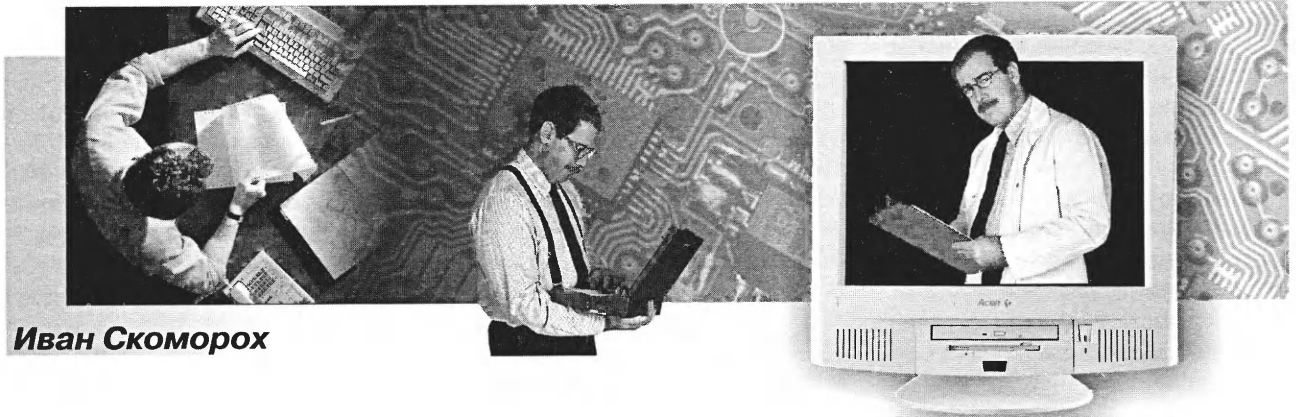
Возможно, я ненаблюдательный читатель, поскольку не заметил никаких признаков убыточности: цена, например, практически не менялась. Больше озадачивает другое — потеряв ДК, взамен мы не получили новый журнал. Нам предложили всего лишь сильно разбухшую "Компьютеру".

Что ж, поздравим новый коллектив старой редакции. Или старый коллектив новой редакции? А может, новую редакцию старого коллектива? Совсем запутался, но искренне желаю успеха всем причастным к журналу в деле его возрождения и преобразования.

А теперь — обещанная поздравительная эпитафия:

*В пути наверх, за гребнем века  
Он сбит машиною хайтека,  
И в день двухсотый Новой эры  
Его зарыли в недра Терры.  
Из пепла на его могиле  
Создали клон иной богини:  
Желаем ей счастливой доли —  
В безводье выжить, и в грозу,  
Но, главное, овечка Долли, —  
Смотри, не вырасти в козу!*

Сергей Вардин



Иван Скоморох

# Умные интерфейсы

*Нам нужен новый мир, и мы его творим!*

сто пользуетесь, — браузер или текстовый редактор, неважно. Всмотритесь в него, а затем резко переведите взгляд в окно и посмотрите куда-нибудь вдаль, на верхушки деревьев и крыши домов. Позвольте зрению рассеяться, потом снова резко окунитесь в табличный интерфейс, проделайте несколько действий. Сохраняя состояние лица неизменным, доберитесь до зеркала, и вы тут же заметите, что ваше лицо изменилось. Что с глазами? Они словно собрались для этой работы.

Работая в сложном режиме даже привычного табличного интерфейса, чтобы удержать зрение и внимание собранными, вы хмурите брови и собираете складки вокруг глаз, как это делают близорукие люди. Это явно говорит о том, что машинные интерфейсы делают человечество близоруким!

Какова природа воздействия машинного интерфейса на человека? Психофизиология отвечает однозначно: это воздействие связано с напряжениями мышц, управляющих глазами яблоками. Но для того, чтобы это понять, никакой психофизиологии не нужно, достаточно просто понаблюдать за собой. Более того, если вы проделаете упражнение с зеркалом несколько раз, то заметите, что все ваше лицо

распускается, когда вы расслабляетесь. Это значит, что напряжения, необходимые для собранного зрения, без которого с табличным интерфейсом не поработаешь, охватывают гораздо больший объем мышц, чем можно предположить. Напряжения, которыми мы регулируем зрение, передаются на все мышцы тела, хотим мы того или нет. И это настолько привычно, что никто не замечает, а значит, и не учитывает. В том числе и при создании интерфейсов.

Возьмем, например, тягу программистов писать все свои тексты в интерфейсах самыми мелкими шрифтами, которые не рассмотреть на экране, иначе как засунув голову в ящик монитора. По сути это не программисты, а сам старый подход к созданию интерфейсов как зон машинного взаимодействия заставляет людей служить себе и делает их частью машины. Иначе говоря, это псевдоживое существо само проводит над нами эксперименты и пытается сростить человека с машиной, сделать его своим биологическим придатком. А с помощью сети рождается огромное суперсущество — Машинный Бог, состоящий из множества человеко-компьютеров. Так что вопрос о психологии внимания и напряжений при работе с интерфейсами на деле оказывается вопросом

о свободе человека и вообще о его будущем существовании как вида.

Теперь зададимся таким вопросом: если ты, чтобы прочитать мелкий текст или разобраться в непонятных хитросплетениях кнопок и меню, собрал внимание и напряг управляющие мышцы лица, то где заканчивается это напряжение? Ясно, что напряжение должно или встретить равное сопротивление оболочки (подобно воздушному шару), или же найти опору, например, костяк.

Но костяк — это внутренняя опора, он лишь дает возможность управлять мышцами. Для внешних действий, то есть для работы и движения, нужны еще и внешние точки опоры. Например, для ходьбы нужно опираться на землю, для отодвигания тяжелого шкафа — опираться спиной в стену. Это значит, что любая наша деятельность требует одновременного использования минимум двух точек опоры — внешней и внутренней. Но самое главное — это то, что в любой привычной деятельности мы используем всегда один и тот же рисунок взаимодействия мышц с опорами. А это означает, что у любой работы есть свой опорный рисунок, который после множества повторений как бы впечатывается в тело в виде образа мышечных напряжений! Причем

слесарь носит в себе один образ мышечных напряжений, машинистка другой, а бизнесмен — третий. Что еще важнее — этот образ и есть та основа, отталкиваясь от которой человек думает!

Возникает вопрос: а где была основная опорная точка, когда программисты думали над машинными интерфейсами? Правильно, в заднице. И не просто в заднице, а расположенной так, чтобы засунуть голову как можно глубже в монитор. Исходный образ мышечных напряжений, нужный программисту, чтобы думать, был впечатан в его тело так, что строился по направлению от задницы к машине. Это и есть основная поза верного почитателя Машинного Бога. Поза для поклонения, она же поклонная. Так что заявления, будто ту или иную программу задом делали, отнюдь не издевка, а психофизиологический факт.

Можно ли тут что-то поменять, сделать более человеческим? Не так-то просто. Обратите внимание,

поклонная поза есть детище жесткого стула с прямой спинкой. Проведите опыт, пересядьте в мягкое кресло и попробуйте работать с привычным машинным интерфейсом. Вы почувствуете, что или не можете собрать для этого внимание, или же непроизвольно в мягком кресле попытаетесь усесться, как на жестком стуле.

Это означает, что стул вписан в образ той основы, отталкиваясь от которой программист думает и привыкает думать всех, кто пользуется компьютером. Компания Майкрософт будет крепко стоять на ногах, пока пользователи сидят на жестком стуле.

Кстати, художники от компьютерного дизайна предпочитают вращающиеся полукресла. А вот игрокам гораздо удобнее делать свое дело в креслах, подобных креслам пилотов космолетов. Если находится способ, как установить управление на кресле, то любой геймер радостно изменит жесткому стулу и Машинному

Богу. У них свои Миры и свои Боги. А компьютер для них не алтарь, а виртуатор, машина-телепорт, позволяющая начать волшебное путешествие.

Не буду разбирать, чей Бог лучше. Скажу как человек науки: сложившаяся на сегодня работа за компьютером с головой, засунутой в ящик, вредна для здоровья. Воздействие, которое оказывает компьютер на человека, разрушительно, и одно это оправдывает любые попытки сделать управление этой машиной уютным и свободным. Сам бог велел при таком положении дел искать какие-то новые пути в создании интерфейсов пользовательских программ.

Вот эта мысль и была для нашего НИИ "Дорога Домой" исходной при разработке SIT — Smart Interface Technology, технологии, имеющей своей целью создать сообразительные интерфейсы. Не искусственный интеллект, а интерфейс-помощник, думающий и заботящийся о своем хозяине. Думающий о нем, а не за него! Работа эта далеко не законче-

## Сотворение мира

*"Для них Я сотворил век; но когда Адам нарушил Мои постановления, определено быть тому, что сделано"*

*Третья Книга Эзры, 7:11*

**М**еня зовут Альберт. Я вольный программист, художник виртуального пространства. Признаюсь сразу: характер у меня не сахар. Живу один. Пес и компьютер скрашивают одиночество. Я, впрочем, не ропщу. Такая жизнь по мне. Ненавижу нахальных дамочек и зеленые бумажки. Неудивительно, что у меня бывают проблемы с этими бумажками. От дамочек я отказываюсь сам.

Но Сеть люблю. Она мне открывает мир, и в этом мире больше ничего не нужно. С моей квалификацией я мог бы многого достичь. И я достигну.

Вчера случайно прочитал в Сети рекламу корпорации Universal Matrix Communications Group (UMCG). Сначала я не понял, чего хотят от нас, от вольных программистов. По-

том проникся. И родилась идея у меня обставить всех. Я написал по электронной почте в UMCG и получил пакет-задание.

Им надо сделать мир. Почти что настоящий. Не игровой, а именно реальный. То есть реально-виртуальный. Чтобы существовал по собственным законам и не нуждался в помощи извне. Мир самоуправляющийся и саморегулирующийся. Пока что лишь в компьютере.

Они объявили конкурс. Обещают 7508 баксов в качестве приза за лучший из миров. Сумма, конечно, аховая. Я думаю, таков и был расчет: крутые программы и веб-дизайнеры, работающие только на заказ, ею не соблазняются. Иное дело мы, свободные энтузиасты. Нам главное идея и — творить! И чтобы никто, конечно, над душой не стоял.

Если за творчество еще и баксами заплатят... тогда я сразу сделаю апгрейд, давно пора.

Идея увлекла меня. Вы понимаете, им нужен настоящий мир! Как в "Матрице", и даже лучше. Мир не заезженный, именно новый. И чтобы никаких "игрушечных" заморочек. Очевидно, хотят с нас, вольных программистов, сливки собрать. Хм! Увидите, я их обставлю. Я им скриэйчу то, чего они не ждут.

Нужно начать с азав. Нет, даже не с ассемблера. Специалистов по ассемблеру и без меня хватает. Я сделаю свой собственный язык, специально для моего мира.

Готово. Сделал. Есть у меня такой язык. Не верите? Ну и не надо, дело ваше. Мне главное, что я этот язык создал.

Черт... Пока я занимался языком, прошли все сроки. Опоздал! На сайте UMCG уже полно анонсов. Ну кто бы мог подумать... Есть! Они продлили сроки конкурса. Всего на неделю. И говорят, что больше продлевать не будут. Я должен успеть. Если нужно, буду работать днем и ночью, но успею.

на. Но если на смену машинным интерфейсам придет думающий компьютер, то произойдет коренное улучшение софта.

Когда институт приступал к разработке своего подхода, первыми встали вопросы отнюдь не новых алгоритмов для нейронных сетей. Все гораздо проще. Настолько проще, что на этом не докажешь всему миру, что ты самый умный. Вопрос всего лишь в том, как убрать жесткие напряжения, произвольно возникающие у человека при восприятии современных машинных интерфейсов, как сделать зрение пользователя расслабленным, а думание — не зависящим от задницы. Решить эту задачу — значит сохранить зрение и снять хронические мигрени у миллионов людей. К тому же перевод человека с жесткого стула в мягкое кресло, где он не имеет опоры, а значит, может менять напряжения по своему усмотрению, сделает человека независимым от внешних условий. А это значит, подготовит к грядущим

интерфейсам, которые будут речевыми и виртуальными.

### Речевой интерфейс

Проблема речевого управления, которая скоро обрушится на компьютерный мир, поскольку технология быстро создает для этого условия, существенно поменяет нынешнюю картину. Подавляющее большинство современных профессионалов окажутся выкинутыми за борт и сильно отстанут, как это было с программами первого поколения, привыкшими работать с большими машинами. Многие из них так и не перестроились на ПК. Точнее, молодежь, сразу приходившая на ПК, их обогнала. То же самое будет и с приходом речевого управления. Так что к этому стоит готовиться.

Но, как ни странно, именно эта слегка пугающая мысль поможет вам понять, что такое СИТ. Представьте себе, как будет происходить ваше общение с компьютером, ког-

да он заговорит. Это уже не раз показывали в фильмах. Вы входите в свой дом и начинаете раздавать приказы. А компьютер их слышит и выполняет. Это уже как-то делается. Но мысль идет дальше, и вы можете вообразить, что садитесь в кресло и начинаете работать с компьютером как с секретарем, а он все понимает и выполняет потом за вас всю заданную работу. Эта часть уже фантастична. Все знают, что современные средства распознавания речи пока еще не достигли такого уровня. Но знают и то, что прогресс очень скор и уже в недалеком будущем мы получим и такую возможность.

### Искусственный Идиот

Впрочем, сильно не разгоняйтесь. В середине прошлого века таким же близким казалось создание искусственного интеллекта, однако воз и ныне там. Кроме развития технологии, нужно еще и понимание. Технологию создают профессиона-

Все. Я отключился от Сети, снес операционную систему, оставил только свой язык. Я должен сделать этот мир.

\*\*\*

#### День первый

С чего начать? Задача. Гляжу на монитор и вижу черное пространство. Чистейший вакуум. Tabula rasa. Мне следует ее заполнить. Н-да... Легче сказать, чем сделать. Весь день прошел точно в тумане. А я всего только успел создать посреди мрака шар, базовую сферу мира. Похоже на планету? Не знаю... Подсветку удалось добавить и зациклить это дело. Теперь у них там тоже сутки: свет днем, тьма ночью. Еще там будет вечер, будет утро. И все пока.

#### День второй

Планета странная какая-то скриэтилась. Одна вода на поверхности. Очевидно, я второпях напутал в операторах элементарного состава. Пришлось лезть туда снова и кое-что менять. Работа затянулась. Но к концу дня появилась у меня суша. Еще и облака добавить удалось: вода в них

собирается и падает дождем на сушу, а далее стекает в океан. Поэтому, это логично. Хотя и тривиально. Но все-таки оно работает!

#### День третий

Сегодня производил отладку. Сушу закрепил, а воду разделил на океаны и моря. И дальше двинулось, как по маслу. Язык мой — друг мой! Посеянные модули мгновенно дали всходы. Теперь имеются деревья, травы, ну, и прочие зеленые насаждения, как же без них. Мой мир из серого стал разноцветным. Нет, не рискну сказать, что это выглядит отлично, но вижу, что неплохо, даже хорошо.

#### День четвертый

Он начался с досадных траблов. Совершенно разладилось освещение, и посеянные вчера ростки увяли. Долго думал, в чем проблема. Так и не нашел. Стал думать над альтернативным решением. Чертовски разболелась голова. Ага, придумал. Пускай мои светила вращаются сами: те вокруг этих, а эти тоже вокруг чего-нибудь, как это принято везде. Перестроил

алгоритм. Теперь мой мир вращается вокруг большой желтой звезды, она отвечает за управление днем. А за управление ночью ответственна планета меньшего размера, спутник моей. Пришлось повозиться, конечно, но вижу, что это хорошо.

#### День пятый

Сегодня я в ударе. Травы и прочих насаждений мне уже мало. Как понесло творить с утра, так не могу остановиться. Сколько скриэтил рыб, зверей и птиц, сам сбился со счета. Да, и еще рептилий сотворил, на всякий случай. Все-таки очень работоспособные модули.

К концу дня надстроил программу самовоспроизводства. Теперь каждая тварь может без моего участия создать такую же. И не одну. Сидел всю ночь, отлаживал систему. Устал. Зато работает. Приз будет мой. Я уже почти сделал этот мир.

#### День шестой

Эх, видно, сглазил. Опять проблемы. Так вроде все готово, работает. Мир живой, саморегулирующийся. Твердь отделена от воды, день отделен от ночи, полно зеленых насаждений и всякой



лы, и свое дело они знают. Но они не очень хорошо представляют, как это использовать и для чего. Никто не подсказывает им, в каком направлении надо доводить их приборы. Это дело психологов, а психологи боятся программирования.

Так вот, главная сложность на пути речевого управления возникнет не с приборами и их чувствительностью, а с пониманием того, что есть образ, который хозяин хочет передать машине, то есть не с тем, как его расслышать, а с тем, как его понять. И это никак нельзя сделать, пока ты мыслишь в рамках таблично-кнопочного интерфейса. У таблиц и кнопок нет ни лица, ни образа. Лучшие пиктограммы, которыми снабжаются кнопки, — это убожество по сравнению с теми образами, что накопило человечество в живом языке за тысячи лет своего существования. А это значит, что таблично-кнопочные интерфейсы — тупик с точки зрения развития компьютерного мира. Это надо понять и принять как данность.

Но принять как идею — это хоро-

разной живности. Чего еще надо? Другой бы на моем месте почил на лаврах и побежал в UMCG за баксами. Я не таков. Мне хочется улучшить этот мир. Раз можно, почему бы не улучшить? Сижу и думаю, чего б еще туда добавить...

Что, если... Нет, нет! Это уж слишком. Такое даже у меня не выйдет. Лучше оставить все как есть, гран-при я получу и так. А с этим новшеством не знаю, выдержит ли вся система. Эх! Попытка — не пытка. Если средства языка позволяют, почему бы не сделать это? Короче: я перенес туда матрицу своего организма. "Се человек"? Хм. А черт его знает. Если смотреть с моей стороны экрана, на меня похож, определенно. На самом деле он эйон, конечно, — искусственный интеллект.

Э?... Да я тут, между делом, создал искусственный интеллект! Но если так, то мне, пожалуй, нобелевку надо... Ладно. Потом с премиями разберусь, никуда они не денутся. А сейчас добавлю своему эйону вспо-

шо, а как воплотить ее в жизнь? Как я уже сказал, главная сложность в общении компьютера с человеком не в том, чтобы слышать, а в том, чтобы понимать. Это значит, что речевое управление не сдвинется с мертвой точки без развития теории искусственного интеллекта. А он пока — Искусственный Идиот! Я не хочу сказать, что ИИ делали дураки, нет. Его делали очень умные люди, но сидевшие на жестких стульях, — технари, влюбленные в точные науки и презирующие психологию и обычных людей, не уче-



«— Сыграем в игру "Найди почтальона"? В этих рисунках скрыты инструменты управления наших интерфейсов. Они очевидны, но видны не больше, чем "почтальон в кустах". Желающим поразмышлять на эту тему я готов предложить, как минимум, переписку —>



могательный модуль. Зачем? Хочу посмотреть, как два эйона будут между собой взаимодействовать. Да, вот так... Тьфу! Что ж это такое?! Дамочка получилась... Жаль, нет времени разбираться, где трабл. Пусть остается, как есть. Так даже интереснее.

Надо торопиться. Поскольку в моем мире уже живут мои подобия, я вручил им всю власть над этим миром. Над сушей, и над морем, и над деревьями с травой, и над животными, конечно. Если какие будут траблы — пусть сами разбираются и исправляют, выказывают интеллект. Дарую им свободу.

Смотрю и радуюсь: мой мир, я его сделал! Без ложной скромности — да, хорошо весьма.

*День седьмой*

В общем, я сделал все. Анонс послал в UMCG по электронной почте. Ну, представляю, как изумятся эти чудики! А я буду сидеть и ждать, когда они переведут мне баксы. Вернее, я буду тут сидеть и любоваться своим миром.

Сегодня, кажется, суббота? Прошла неделя, как сижу, не отходя, у компа. Клонит в сон... Почти семь дней не спать — такое не каждый выдержит...

\*\*\*

...Хм, сон очень странный. Как будто и не сплю. Вижу своих эйонов. Молодцы, делом занимаются... Но что это?! Вижу какую-то зеленую нить... Нет, это существо живое, извивается. Змея? Наверное. Я разве всех могу припомнить, кого успел за эти дни создать?..

Змея общается с моим эйоном. Со вторым, с дамочкой. Плохо слышу, о чем они говорят... не пойму, ужасно клонит в сон.

Эй! Ты что делаешь, гад?! Он ведет ее в корневой каталог. Эти двое уже у основания Дерева. Проклятие! Я разве не предупреждал? Туда нельзя! Нет у них прав доступа, только я могу входить в систему с этого логина, только я...

Гад извивается вокруг Дерева, тут и второй эйон подходит... вернее, первый. Гад пытается передать им

го Идиота понимать себя. Но что такое этот ИИ? Пока это разум, но неполноценный. Подумаем, не пришлось ли нам уже решать подобные задачи? Нет ли в культуре готовых способов общения с неполноценным разумом?

Конечно, есть! И много. Все общение с детьми, домашними животными и есть управление неполноценным разумом. И самое ценное в этом то, что мы все знаем и умеем это с детства. И даже если не умеем, то узнаем и понимаем сходу. Эти знания живут в миллионах пользователей компьютеров, а рынок почему-то ими не пользуется!

Все эти знания SIT и делает той опорой, на которой выстраивает управление своими интерфейсами. Допустим, вы видите некий рисунок, содержащий в себе различные детали. Но представьте себе, что все, за что в нем цепляется ваш глаз, является деталями не просто рисунка, а скрытого пока интерфейса. Причем такого, который сам подсказывает, какое действие может для тебя выполнить.

доступ к Дереву. О, нет!.. Я понял. Это вирус. Кто-то запустил в мой мир вирус разрушения... или искушения? Должно быть, они успели зацепиться, пока я передавал в UMCG анонс. Кто "они"? Завистники? Конкуренты? Все может быть. Но я им не позволю разрушить этот мир. Кем бы они ни были — не позволю!

О, нет... НЕТ!! Они уже внутри Древа. Как сумели войти, понятия не имею. Наверное, я в спешке пропустил фатальный трагед, и этот вирус, змей, проклятый гад, моим эйонам показал, где дырка. Теперь они пытаются познать то, что недоступно им согласно логике Системы. Наивные кретины, жалкие гордецы, просто ничтожества! Так-то они благодарят меня за все, что я для них создал!? Ну ладно. Вот только я проснусь и сразу наведу порядок...

Дай бог добраться мне до них, я их отделаю, я покажу, кто там хозяин... О Боже, что они творят.. Господи! Прежде я не верил в тебя, но

Пока работы не закончены, я не раскрываю суть глубже. Но определенно могу сказать: главная забота при создании речевого управления — это использование в общении с ИИ таких образов, которые хорошо различает не машина, а сам хозяин! Именно он. Как гласит древняя истина, кто ясно мыслит, тот ясно излагает. Для того чтобы управлять Искусственным Идиотом, нужно мыслить очень ясно, как это нужно для управления лошадью, почтовым голубем, упрямым ослом, в конце концов.

SIT — это такой подход к созданию интерфейса, который позволяет сделать компьютер уютным домом, построив и обставив его по желанию хозяина и заселив живыми и сообразительными помощниками.

Дом же с помощью Сети легко становится частью большого мира. Но для того чтобы он этой частью стал, мир надо создать. Только в этом случае поведение помощников станет оправданным и понятным, а значит, доступным управлению. За этим скрывается психологи-

если ты на самом деле существуешь, яви мне чудо! Помоги войти туда.

Да, я хочу туда, в мой мир, который создал. Я должен навести порядок, я должен покарать измену, я должен... Какая боль... я понял. Да, это конец. Я проиграл тот конкурс.



Мой мир не может без меня. Но я выиграл свой конкурс. Меня ничто больше не держит здесь. Мой мир не здесь, а там... я ведь его создал, и это было хорошо!

Мне нужно только чуть

ческое ноу-хау. Мы понимаем только то, что считаем закономерным, то есть подчиняющимся определенным законам. Следовательно, для того чтобы Искусственный Идиот стал помощником, нужно, чтобы он вел себя по законам. Причем эти законы нужно определить, глядя на его поведение.

Для того чтобы интерфейсы стали понятны хозяину, ожили, стали сообразительными и уютными, нужно создать мир, в котором оправданы законы, определяющие все взаимодействия человека и машины. Пока подобный мир создали только игровики. Это действительно довольно цельный мир. Но его законы не позволяют взаимодействовать с настоящим миром, где ты зарабатываешь на жизнь. Главный недостаток игровых миров в том, что их задача — увести человека из настоящей жизни в мир мечты. В этом же и их психотерапевтическое значение. Но наша цель — дать людям помощников, облегчающих реальную жизнь и делающих ее уютнее. Поэтому нам нужен иной мир, и мы его творим!

подправить траблы. Яви мне чудо, отпусти, и я в Тебя поверю, Боже...

\* \* \*

Три полисмена высадили дверь и ворвались в квартиру. У порога лежал труп здорового пса. А дальше, в комнате, за монитором, сидел обглоданный скелет.

На мониторе бешено плескался океан, и огромный корабль диковинной конструкции, полный всякой живности, казалось, вот-вот исчезнет среди гигантских волн...

Когда убрали трупы, сержант, окинув взглядом монитор, спросил лейтенанта:

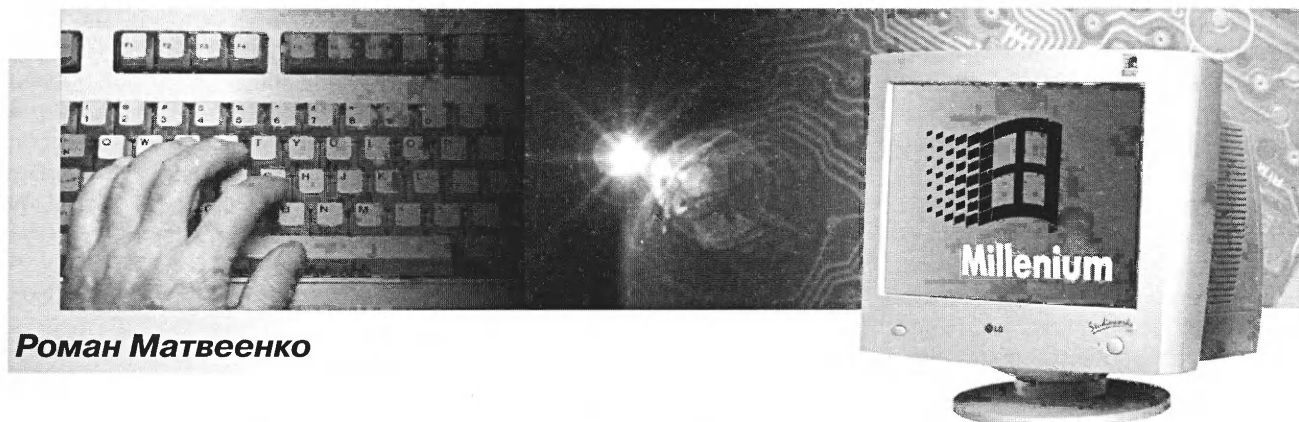
— А что с этим делать, сэр? С компьютером.

— Да просто выключи его. Кому он теперь нужен, обычный компьютер, — со вздохом ответил лейтенант. — Хозяин-то уже давным-давно на небесах...

— Глупая смерть, — кивнул сержант.

— Смерть всегда глупа, Джонни...

*Борис Толчинский*



Роман Матвеевко

## Windows Millennium — DOS возвращается!

**И**счезновение полноценной поддержки режима MS-DOS в Win Me для определенного контингента пользователей явилось достаточно огорчительным "нововведением" этой операционной системы. Пункт загрузочного меню, появляющийся при загрузке после нажатия кнопки F8 и предлагающий загрузить DOS-сессию, бесследно исчез. Точно так же исчез и соответствующий пункт из меню "Завершение работы Windows", оставив некоторым из нас лишь ностальгические воспоминания о былой возможности выбора. Не будем вступать в полемику по поводу "нужности" или "ненужности" этого режима и обсуждать степень его значимости в операционной системе Windows. В мире "drag-and-drop и plug-in-play" этот режим для кого-то представляет несомненную ценность, а кому-то вообще неизвестен, так сказать, за ненадобностью. Описанная ниже несложная процедура предназначена в основном для тех, кто так или иначе желает вернуть себе возможность переключения между "чистым" DOS и Win Me.

### Предыстория

Принцип, положенный в основу описанной ниже процедуры, чрезвычайно прост. Основывается он на

подмене (перезаписи) новых, не поддерживающих DOS-сессию модулей, содержащихся в Win Me, их аналогами, содержащимися в Win 98. Естественно, подобная перезапись обратима, то есть при выходе из DOS-сессии все измененные файлы Win Me возвращаются к своему первоначальному виду. Все необходимые подмены осуществляются при помощи обычных bat-овских файлов. Для реализации этой процедуры вам потребуется компьютер

с Win Me и установочный диск Windows 98/98SE.

Несколько слов об источнике. Отправной точкой для написания этой статьи послужил материал, авторами которого являются люди, имеющие самое непосредственное отношение к компании Microsoft. Если верить информации на их сайте, они являются сотрудниками этой компании (нынешними или бывшими, доподлинно неизвестно). Тем не менее, их работа по созданию меха-

## Загрузочные диски — в поисках компромисса

**З**агрузочные диски, или ESD (Emergency Startup Disk), относятся к числу тех носителей информации, вспоминать о которых приходится не слишком часто. Необходимость в использовании таких дисков в основном связана с ситуациями, в которых требуется перезагрузить компьютер без использования основного, жесткого диска. Создавать же такие загрузочные источники приходится еще реже, обычно непосредственно в момент установки ОС или позже, посредством специального встроенного в Windows мастера. Загрузочные диски, полученные та-

ким способом, выполняют свои функции вполне сносно.

И все бы ничего, но немного огорчает тот факт, что число доступных системных сервисов, получаемых при загрузке с таких дискет, крайне мало. По большому счету, к их числу можно отнести лишь командную строку DOS-shell, поддержку CD-ROM и виртуальный диск. Все остальные достижения компьютерной мысли остаются, так сказать, за бортом. Естественно, что в дальнейшем такие диски приходится слегка корректировать. Причем что-то приходится удалять, поскольку размер дискеты ограничен, а что-то и до-

низма возврата DOS в Win Me носит исключительно неофициальный характер, то есть изложенный ниже процесс не является официальным детищем компании Microsoft. Поэтому, если вы регулярно контактируете со службой технической поддержки этой компании или связаны ее гарантиями, подумайте, прежде чем проводить подобный эксперимент со своим компьютером. Для остальных пользователей на личном опыте могу сказать, что описанный на сайте [http://www.sgmvp.freewebsites.com/WinME\\_DOS/Win-ME.htm](http://www.sgmvp.freewebsites.com/WinME_DOS/Win-ME.htm) механизм работает, и ниже привожу краткое руководство по его реализации.

### **Шаг первый, не обязательный**

Данный шаг является подстраховочным и на всю дальнейшую процедуру особого влияния не окажет. Основная решаемая на этом шаге задача — сохранение резервных копий всех основных системных файлов Win Me. Если у вас имеется загрузочный диск Win Me, скопируйте на него файл Sys.com из папки Windows\Command. Если же такого диска у вас нет, то сначала создайте его, выбрав значок "Удаление-установка программ" в Контрольной па-

бавлять — прежде всего это относится к реализации более удобных способов навигации и управления файловой системой. Но в любом случае ручная правка доступна только тем, кто изначально знает, что и как нужно делать. А как быть тем, кто не знает? С целью расширения потенциальных возможностей тех, "кто знает" и помощи в создании универсальных загрузочных источников тем, "кто не знает", создан специализированный программный инструмент The Ultimate Boot Disk. С помощью этой программы задача создания загрузочных дисков может приобрести качественно иной характер.

Есть два варианта этой программы — для Windows 9x и Me. Поддержка Windows NT и 2000 также обеспечена, понадобится лишь небольшой модуль — plug-in. Для нормального запуска программы надо иметь

нели и, далее, закладку "Загрузочный Диск", а затем нажав кнопку "Создать Диск". После того как диск будет создан, скопируйте на него файл Sys.com из указанной директории. Не забудьте после этого удалить дискету из дисковода.

### **Шаг второй**

Создаем промежуточную рабочую директорию, скажем, C:\SetupDOS, хотя можно использовать и любое другое название. Главное, чтобы эта директория находилась в корневом каталоге системного диска. В дальнейшем эту директорию можно будет удалить. Помещаем в нее файл setupdos.bat (его можно загрузить с адреса <http://www.sgmvp.freewebsites.com/Files/setupdos.bat>). Далее помещаем установочный CD-диск для Win98 или Win98SE в привод CD-ROM. Теперь через строку "Выполнить" (Run), меню "Старт", набираем следующее: "C:\Setupdos\setupdos X", где X будет буквой, обозначающей привод CD-ROM (к примеру, C:\SetupDOS\setupdos E), и нажимаем Enter. После этого будет создана директория C:\PureDOS, в которую затем с установочного диска Windows будут скопированы необходимые DOS-файлы (всего их около

по меньшей мере 32 Мб ОЗУ — так, во всяком случае, заявляют авторы. Однако я выяснил, что это требование — не более чем подстраховка: программа нормально работает и с меньшим объемом ОЗУ, правда, медленнее. Размер дистрибутивов для Windows 9x и Me порядка 600 Кб, а модуля plug-in для Win NT/2000 — 260 Кб, все необходимое можно загрузить с домашней странички программы (<http://www.startdisk.com/Web2/ubd/ubd.htm>).

Итак, после того как вы загрузите и запустите на исполнение добытый из Интернета файл (для Windows 9x он будет называться UBD.EXE, для Windows Me — MEUBD.EXE), вам будет предложено разместить в дисковом диске формата 3,5", и процедура создания загрузочной дискеты начнется. В целом работа этой программы очень напоминает работу

30). Тут необходимо отметить следующее: setupdos.bat подразумевает, что дистрибутив Windows 98 содержится в директории X:\Win98, где X — литера, обозначающая привод CD-ROM. Если же используемый инсталляционный диск Windows 98 содержит дистрибутив в папке с названием, отличным от указанного, файл setupdos.bat потребует слегка подкорректировать (найти строчку, содержащую запись вида CD\WIN98, и отредактировать ее таким образом, чтобы название директории в точности соответствовало истинному расположению дистрибутива Windows 98).

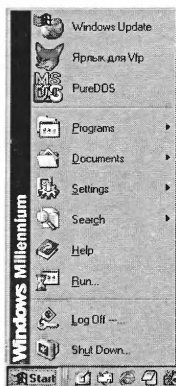
### **Шаг третий**

С помощью Windows Explorer запускаем на исполнение предварительно загруженный (<http://www.sgmvp.freewebsites.com/Files/Configuration.exe>) самораспаковывающийся zip-архив Configuration.exe. Если вы не захотите по понятным причинам запускать exe-файл, его можно распаковать вручную. По умолчанию содержащийся в нем bat-овский файл PureDOS.bat должен быть помещен в директорию C:\PureDOS (в случае ручной распаковки не забудьте об этом и поместите его в нужную ди-

встроенного в Windows мастера — несколько простых вопросов с последующим копированием файлов на дискету. Причем никаких предварительных инсталляционных действий для нормальной работы "The Ultimate Boot Disk" не потребуются — загруженный из Интернета файл является вполне автономной программой. Кстати, целевую дискету можно предварительно не форматировать: The Ultimate Boot Disk сделает это самостоятельно. Как говорится, мелочь, а приятно.

В результате у нас на руках окажется почти обычный загрузочный диск, и теперь, для того чтобы продолжить разговор в предметном русле, мы перезапустим компьютер с помощью только что созданной загрузочной дискеты и произведем краткий обзор ее функциональных возможностей.

ректорию, в C:\PureDOS). Теперь можно, но не обязательно, отредактировать ярлык этой программы. Для этого необходимо щелкнуть правой кнопкой мыши на значке C:\PureDOS\PureDOS.bat и, выбрав из появившегося меню пункт "Свойства", перейти в закладку "Программа", где отметить флажком строчку "Закрывать окно по завершении сеанса работы", нажать кнопку "Применить" и, наконец, закрыть окно свойств с помощью кнопки "ОК". Далее перетаскиваем ярлык PureDOS в "Стартовое" меню. Это делается исключительно с целью облегчить дальнейший доступ к данному файлу. В принципе, никаких перетаскиваний можно и не делать.



**Шаг четвертый**

Теперь загружаем и запускаем на исполнение файл Commands.exe (<http://www.sgmvp.freewebsites.com/Files/commands.exe>). Этот файл также является самораспаковываю-

После перезагрузки ПК мы попадаем в интерфейсную оболочку.

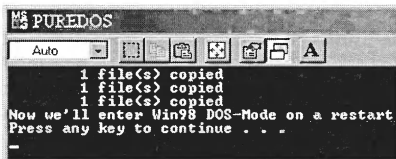


Графическое оформление программы выполнено довольно аскетично — используется символьный цветной режим 80x25. Впрочем, это не вредит общему качеству оболочки, скорее наоборот, псевдографика позволяет достичь большей системной экономичности. Кроме того, многие из возможностей этой оболочки реализуются не внутри нее самой, а с помощью подключаемых в момент вызова модулей. Напри-

щимся zip-архивом и содержит два файла: SetPath.bat и WinMe.bat. В процессе распаковки zip-архива по умолчанию эти файлы будут помещены в директорию C:\Windows\Command. При ручной распаковке архива не забудьте поместить указанные файлы в нужную директорию: C:\Windows\Command. По большому счету, самораспаковывающиеся zip-архивы только для того и предназначены — без лишних хлопот положить все файлы именно туда, куда нужно.

**Шаг пятый, завершающий**

Нажмите на кнопку "Старт", вызовите ярлык PureDOS — в этот момент происходит перезапись нескольких системных файлов Win Me их аналогами из Win 98.



Далее компьютер перезапускается, и мы попадаем в обычную DOS-сессию. После начала DOS-сессии наберите в командной строке Setpath и нажмите клавишу Enter.

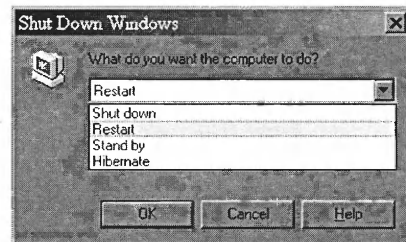
мер, для проверки аппаратной и программной конфигурации компьютера вызывается SysChk, а для работы с файловой системой — старый знакомый VC 4.0. И пусть вас не смущает полное отсутствие окон аля Windows, понятие GUI в данном случае трактуется довольно специфичным образом. Итак, бегло пройдемся по пунктам главного меню:

*Help* — название говорит само за себя; правда, предоставляемая здесь информация выглядит не слишком исчерпывающей, но в трудной ситуации это лучше, чем ничего.

*Fast Boot* — несколько опций, позволяющих сократить количество загружаемых при рестарте модулей; сюда отнесены драйверы и менеджеры памяти.

*Computer Info* — информация об аппаратной конфигурации вашего компьютера (с использованием программы SysChk).

Данный вызов устанавливает некоторые переменные окружения, в частности, прописывает путь к директории C:\PureDOS. Делать его, в принципе, не обязательно. Теперь вы работаете в среде DOS и делаете все, что считаете нужным. Для выхода из DOS-сессии и возврата в Windows набираете в командной строке WinMe и нажимаете клавишу Enter. В результате происходит восстановление модифицированных системных файлов Win Me с возвращением им исходного, пред-DOS-сессионного вида.



Теперь в любой момент вы вызываете ярлык PureDOS, перезапускаете компьютер и оказываетесь в DOS-сессии, а после набора WinMe снова возвращаетесь в Windows. Все как раньше, в Windows 9x. За исключением, пожалуй, благословения от Microsoft.

*Virus Scan* — антивирусное сканирование целевого диска (свежесть антивирусных баз не проверялась).

*Fix Registry* — сохранение и восстановление реестра Windows.

*Scan Drive for Errors* — сканирование структуры и поверхности дисков, аналог Scandisk.

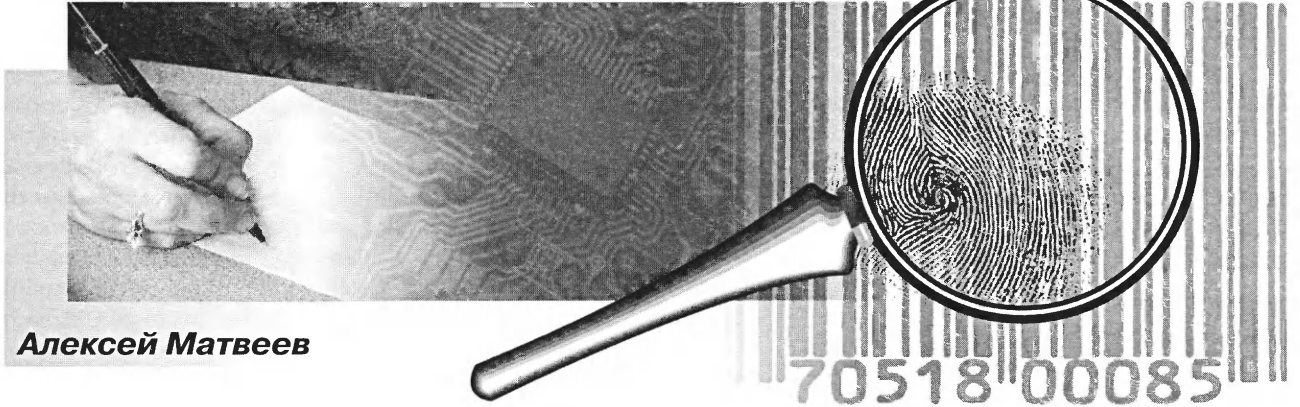
*Prepare Drive* — предварительная подготовка жестких дисков, разбиение на разделы и форматирование.

*Install Windows* — установка Windows.

*Find Files* — запуск файл-менеджера Volkov Commander.

*Documentaion* — документация к программе; после вызова этой опции соответствующие файлы распаковываются во временный каталог, созданный в корне первого раздела.

*More Tools* — опция, вызывающая меню дополнительных инструментов (распаковка CAB-файлов, манипуля-



Алексей Матвеев

# Распишитесь, пожалуйста!

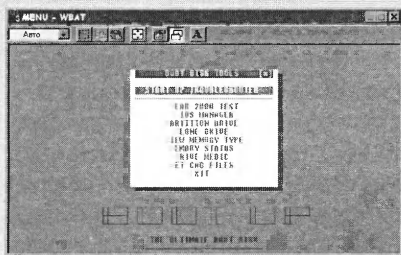
Продолжение. Начало см. "Магия ПК" №7-8/2001

**П**режде чем обратиться к рассмотрению некоторых из широко известных систем, необходимо сказать несколько слов об общей классификации систем ЭЦП.

## Классификация систем ЭЦП

Все существующие схемы ЭЦП делятся на два класса: схемы с ап-

пции с физической памятью, сохранение и восстановление наборов установок BIOS, разбиение или клонирование разделов жесткого диска и некоторые другие возможности.



Безусловно, созданная таким образом загрузочная дискета тоже не идеальна. По большому счету, идеальный загрузочный источник — это тот, который сделан самостоятельно и с оглядкой на конкретные конфигурационные и прикладные особенности компьютера. Но на созданной дискете есть многое из того, что нуж-

пендиксом и схемы с восстановлением сообщения.

Схемы с аппендиксом (пример такой схемы был приведен в прошлом номере) в качестве одного из элементов, необходимых для проверки подписи, используют собственно сообщение. ЭЦП сопровождает текст в виде аппендикса, "прицепляемого" к этому тексту, а сам текст не изменяется. К этому

но для автономной работы с компьютером в экстренном режиме. Правда, при начальном знакомстве возникает ощущение некоторой "примитивности", но к ней довольно быстро привыкаешь, тем более, что объясняется она довольно просто: на носителе с объемом 1,44 Мб особо не разгуляешься. Зато благодаря такому подходу на небольшом по объему носителе удастся разместить очень много полезных программных инструментов, а плюс к этому — снабдить пользователя хоть каким-то подобием интерфейса.

Одним словом, такие загрузочные дискеты являют собой некий компромисс: комфорт приносит в жертву универсальности. Хотя комфортность в данном случае — понятие относительное, вспомним хотя бы загрузочные дискеты, созданные с помощью встроенных средств Windows.

Роман Матвеевко

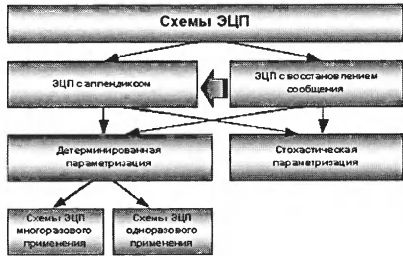
классу относятся, например, схемы DSA (Digital Signature Algorithm), Эль-Гамала, Шнорра, а также отечественный ГОСТ. В силу некоторых объективных причин, о которых скажу пару слов ниже, схемы ЭЦП с аппендиксом более распространены, чем схемы с восстановлением.

В схемах с восстановлением сообщения текст в исходном виде не передается. наоборот, передаваемое подписанное сообщение восстанавливается из подписи при ее проверке. Иными словами, весь пакет передаваемой информации суть ЭЦП, содержащая в себе сообщение. К этому классу относятся такие схемы, как RSA, Рабина, Найберга-Рюппеля.

Оба эти класса схем ЭЦП можно подразделить на схемы с детерминированными параметрами и схемы, в которых присутствуют случайный параметр (стохастическая параметризация). Случайный параметр выбирается из некоторого множества индексов ЭЦП. Мы это увидим, когда будем рассматривать конкретные схемы ЭЦП (в частности, схему ЭЦП Шнорра).

Схемы с детерминированными параметрами, в свою очередь, делятся на многоразовые и одноразовые. Последние я не буду рассматривать, ограничусь лишь замечанием, что это схемы, которые предпо-

лагают использование ключевых элементов только один раз, для единственного сообщения. В противном случае секретный ключ ЭЦП будет скомпрометирован.



**Функции хеширования**

Следует отметить один важный аспект. Любая схема ЭЦП с восстановлением сообщения может быть очень легко преобразована в схему ЭЦП с аппендиксом. Принципиально схему с аппендиксом от схемы с восстановлением сообщения отличает одна простая вещь, называемая хеш-функцией.

Поскольку схемы ЭЦП используют операции в некотором конечном поле, в котором число элементов имеет битовое представление длины  $N$  (как уже упоминалось, это достаточно длинные представления), для того чтобы сообщение участвовало в этих операциях, необходимо "загнать" его в это поле. Это можно сделать несколькими способами. Можно просто сузить область применения ЭЦП до множества сообщений ограниченной длины, которая гарантирует попадание сообщений в рабочее поле. Можно использовать сообщения любой длины, но при этом нарезать их на подходящие фрагменты и к каждому из них применять ЭЦП. Именно так работают схемы с восстановлением сообщения.

Схемы с аппендиксом решают проблему иначе. Пусть определена функция  $h: \{0,1\}^* \rightarrow P$

где  $\{0,1\}^*$  — множество битовых последовательностей произвольной длины, а  $P$  — рабочее поле, в котором работает алгоритм ЭЦП. Эта функция занимается тем, что "загоняет" битовые последовательности произвольной длины (произвольные сообщения) в битовую последовательность, которая соответствует некоторому элементу в рабочем ко-

нечном поле. Она-то и называется функцией хеширования.

Функция хеширования считается безопасной, если она свободна от коллизий и однонаправленна. Под свободой от коллизий подразумевается, что вычислительно сложно найти два текста, отображающихся в один и тот же элемент поля. Понятное дело, для нужд ЭЦП применяются безопасные хеш-функции. Существуют соответствующие алгоритмы построения надежных функций хеширования, но это отдельный большой разговор.

Так вот, если применять схему с восстановлением не к сообщению, а к его хеш-образу, мы получим схему с аппендиксом. Это очень существенно, поскольку алгоритмы подписания и проверки ЭЦП вычислительно весьма тяжеловесны, а посему применение схем с аппендиксом более экономично. Кроме того, разбиение сообщения на блоки с последующим применением к ним ЭЦП может быть небезопасным с точки зрения возможных атак с целью компрометации ЭЦП. Впрочем, слабая функция хеширования — тоже брешь в крепостной стене, и потому хеш-функции обязаны быть неотъемлемой частью схемы ЭЦП, чтобы избежать атаки на нее путем "подставки" слабой хеш-функции.

Теперь рассмотрим некоторые конкретные схемы ЭЦП.

**RSA**

Алгоритм RSA (начальные буквы фамилий — Rivest, Shamir, Adleman) базируется на том, что разложение большого числа на множители является нетривиальной вычислительной задачей.

Секретным элементом RSA-схемы является порядок мультипликативной группы, в которой производятся операции. Такая вот заковыка: операции в группе можно производить, а узнать ее порядок — трудно. Противнику, чтобы взломать схему, надо всего лишь найти элемент, обратный данному, но он не может этого сделать, не зная порядка группы. Чтобы найти порядок рабочей мультипликативной группы, ему требуется

разложить большое несекулярное число на два множителя, что достаточно сложно.

Чтобы сказанное не стало общим местом, вспомним чуть-чуть теории. В кольце вычетов по модулю  $n$ ,  $Z(n)$ , только те его элементы имеют обратные по умножению, которые взаимно просты с  $n$ . Понятно, что подмножество обратимых по умножению элементов кольца образуют мультипликативную группу. Если  $n$  — простое, то кольцо вычетов превращается в поле. Количество чисел, меньших  $n$  и взаимно простых с ним, называется функцией Эйлера  $\phi(n)$ . Очевидно, если  $n$  — простое, то  $\phi(n) = n - 1$ . Легко понять, что для взаимно простых чисел  $n$  и  $m$  функция Эйлера мультипликативна, то есть  $g.c.d(n, m) = 1 \Rightarrow \phi(nm) = \phi(n)\phi(m)$ . Здесь  $g.c.d$  — greatest common divisor, наибольший общий делитель (НОД). Таким образом, если  $n = pq$ , где  $p$  и  $q$  — простые,

$$\phi(n) = (p-1)(q-1)$$

Последнее выражение есть не что иное, как порядок мультипликативной подгруппы в кольце вычетов по модулю  $n$ . Чтобы найти его, надо разложить данное число  $n$  на множители  $p$  и  $q$ .

Классическая схема RSA реализуется как схема с восстановлением сообщения, но если применять хеширование сообщения, можно реализовать схему с аппендиксом. Схема RSA является схемой с детерминированной параметризацией, никаких дополнительных параметров из индексного множества в нее не привносится. Пространство подписываемых сообщений и пространство подписанных текстов — одно и то же,  $Z(n) = \{0, 1, 2, \dots, n-1\}$ , где  $n = pq$ , а  $p$  и  $q$  — большие простые числа приблизительно одного порядка. Если применяется схема с восстановлением, подписываемый текст режется на сегменты, попадающие в  $Z(n)$ . Если применяется схема с аппендиксом, подписываемое сообщение отображается в  $Z(n)$  с помощью хеш-функции  $h$ . Для удобства будем полагать, что владелец секретного ключа ЭЦП — некто Алиса (есть определенные традиции в этой тематике). Публичный ключ ЭЦП со-

ответственно привязан к Алисе путем его регистрации в каком-то центре сертификации. Пусть Боб — тот недоверчивый парень, который хочет убедиться в целостности принятого сообщения и в том, что его подписал не кто иной, как Алиса.

**Алгоритм генерации ключей в схеме RSA**

Каждый владелец ЭЦП генерирует публичный ключ и связанный с ним секретный, в том числе и Алиса выполняет следующее:

1. Генерирует два различных больших простых числа  $p$  и  $q$  приблизительно одного порядка; вычисляет  $n=pq$  и  $w=(p-1)(q-1)$ ,  $p$  и  $q$  — секретны;
2. Выбирает случайное целое  $e$ ,  $1 < e < w$ , такое, что  $\text{НОД}(e, w) = 1$ ;
3. С помощью расширенного алгоритма Евклида вычисляет уникальное  $d$ ,  $1 < d < w$ , такое, что  $ed = 1 \pmod{w}$ .
4. Публичный ключ Алисы — пара  $(n, e)$ ; ее секретный ключ —  $(d, p, q)$ . Вот то, о чем я сказал выше: противнику достаточно вычислить  $d$  — элемент, обратный  $e$ , — но он не знает  $w=(p-1)(q-1)$  — порядка группы.

**Применение ЭЦП**

Рассмотрим случай RSA с аппендиксом. Пусть  $m$  — сообщение, которое подписывает Алиса. Она выполняет следующее:

1. Вычисляет  $\tilde{m} = h(m)$  — образ ее сообщения в  $Z(n)$ .
2. Вычисляет  $s = \tilde{m}^d \pmod{n}$  ( $d$  — секретный ключ).
3. Подписью сообщения  $m$  является  $s$ .

**Проверка ЭЦП**

У Боба, проверяющего подпись Алисы, есть сообщение  $m$ , ее подпись  $s$  и ее публичный ключ  $(n, e)$ .

1. Боб тоже вычисляет  $\tilde{m} = h(m)$ , поскольку хеш-функция не секретна.
2. Далее он вычисляет  $\tilde{m}' = s^e \pmod{n}$
3. Если  $\tilde{m} = \tilde{m}'$ , Боб полагает, что подпись верна.

**Доказательство**

**состоятельности проверки**

$\tilde{m}' = s^e \pmod{n} = \tilde{m}^{de} = \tilde{m} \pmod{n}$  поскольку  $ed = 1 \pmod{w}$ .

Заметим, что каждый владелец ЭЦП по схеме RSA генерирует два больших (очень больших!) простых числа.

**Схема аутентификации по Шнорру**

Рассмотрим очень интересную схему аутентификации, построенную на схеме Шнорра, которая является модификацией схемы Эль-Гамала. Надо сказать, почему-то в работах на тему ЭЦП редко описывается базовая идея семейства схем Эль-Гамала.

Схема аутентификации по Шнорру — доказательство владения некоторым секретным ключом без его раскрытия. ЭЦП по Шнорру — это цифровая подпись с аппендиксом, со стохастической параметризацией (с привлечением некоторого случайного элемента из индексного множества).

Работа идет с полем  $Z(p)$ . Это поле вычетов  $\text{mod } p$ , где  $p$  — большое простое число. Есть также простое число  $q$ , меньшее  $p$  и делящее  $p-1$ . Число  $q$  определяет циклическую подгруппу порядка  $q$  в поле  $Z(p)$ , для которой нам потребуется генератор  $g$ . Генератор обладает свойством  $g^q \equiv 1 \pmod{p}$

По-прежнему мы собираемся эксплуатировать функцию  $f : s \rightarrow g^s \pmod{p}$ , которая при правильно выбранных  $p, q$  и  $g$  является однонаправленной, поскольку обратная ей — логарифмирование в конечном поле  $Z(p)$  — по-прежнему является сложной вычислительной задачей.

**Идея метода**

В основу семейства Эль-Гамала-подобных алгоритмов аутентификации и ЭЦП положена простая и изящная идея. Схема Шнорра, на мой взгляд, отражает эту идею наиболее наглядно. Пусть мы имеем прямую на плоскости, описываемую уравнением

$$y = mx + b \pmod{q} \quad (*)$$

над полем порядка  $q$  в поле  $Z(p)$ . Прямая на плоскости, очевидно, полностью описывается коэффициентом наклона (далее — просто наклоном)  $m$  и сдвигом  $b$ . Если мы хо-

тим засекретить прямую, мы можем спрятать определяющие параметры от любопытных глаз с помощью нашей функции  $f$ , поскольку она является однонаправленной:

$$c \equiv g^b \pmod{p} \quad n \equiv g^m \pmod{p}$$

Элементы  $c$  и  $n$  являются "тенью", отброшенной прямой на поле  $Z(p)$  в свете функции  $f$ . Как вы понимаете, знание  $c$  и  $n$  не дают непосредственного знания параметров прямой, но при этом они дают возможность проверить, лежит ли данная точка  $(x, y)$  на этой прямой. Действительно, если точка  $(x, y)$  удовлетворяет уравнению  $(*)$ , то эта же точка удовлетворяет и уравнению  $g^y \equiv g^{mx+b} \equiv (g^m)^x g^b \equiv n^x c \pmod{p}$  и, естественно, наоборот. Поскольку в уравнении

$$g^y \equiv n^x c \pmod{p} \quad (**)$$

нет ни одного явно указанного параметра секретного уравнения прямой, всякий может проверить равенство  $(**)$ , то есть проверить, лежит ли заданная точка  $(x, y)$  на заданной секретной прямой. Однако генерировать точки на этой прямой может только тот, кто знает ее параметры  $m$  и  $b$  в уравнении  $(*)$ . Отсюда уже ясен способ аутентификации.

**Протокол Шнорра для аутентификации**

Пусть теперь Алиса — владелец секретного ключа  $m$ , который соответствует публичному ключу  $n$ . Публичный ключ обычно привязывается к владельцу, в данном случае к Алисе. Пусть опять Боб — тот недоверчивый парень, который хочет убедиться, что на другом конце именно Алиса, а не самозванец, то есть тот, с кем он имеет дело, действительно знает секретное значение  $m$ , соответствующее данному публичному ключу  $n$ .

1. Боб знает публичный ключ  $n$  Алисы (это тень наклона прямой); Если Боб не знает публичного ключа Алисы, она может послать ему сертификат, связывающий публичный ключ  $n$  и ее персональные данные. Такой сертификат выдается третьей стороной, которой оба доверяют, — обычно центром сертификации и генерации ключей.





2. Алиса посылает Бобу значение  $s$  (тень сдвига прямой);

3. Боб посылает Алисе оклик (challenge), случайное число  $x$ ;

4. Алиса получает  $x$ , вычисляет  $u$  из уравнения (\*) и посылает его Бобу. В отличие от Боба и всех прочих она может это сделать легко, поскольку знает секретный коэффициент прямой;

5. Боб получает  $u$  и применяет уравнение (\*\*) для проверки, принадлежит ли  $(x, y)$  заданной прямой. Если (\*\*) выполняется, значит, Алиса знает  $m$  и, следовательно, является тем, за кого себя выдает.

Вы уже заметили один толстый нюанс этого протокола? Правильно, его нельзя применить два раза с одними и теми же  $(s, n)$  без того, чтобы немедленно не компрометировать секретное значение  $m$ . Действительно, если пара  $(s, n)$  использована дважды, Боб получит пару точек  $(x_0, y_0), (x_1, y_1)$  на одной прямой. Если только Боб не закоренелый двоечник, он немедленно вычисляет

$$m = \frac{y_0 - y_1}{x_0 - x_1} \pmod{q}$$

и дело в шляпе — секретный ключ Алисы у него в руках. Так что же делать? На самом деле в данном протоколе аутентификации уже содержится неявный ответ. Секретным является только наклон прямой, а сдвиг при каждом новом подписании выбирается случайно из некоторого индексного множества (поэтому такие алгоритмы и называются алгоритмами со стохастической параметризацией). Каждый раз, выполняя пункт 2 протокола, Алиса случайно выбирает сдвиг  $b$  и вычисляет для него тень  $s$ . Индексное множество  $\{b\}$  должно быть настолько велико, чтобы вероятность применения одного и того же индекса на практике была крайне мала. Множество индексов  $\{b\}$ , очевидно, суть подполе  $Z(q)$  в  $Z(p)$ , поэтому простое число  $q$ , определяющее это подполе, должно быть достаточно велико. В итоге Алисе как владельцу ЭЦП принадлежит множество из  $q$  параллельных прямых  $s$  с наклоном  $m$ .

### ЭЦП по схеме Шнора

Понятно, как легко трансформи-

ровать протокол аутентификации Шнора в схему ЭЦП. Боб не посылает Алисе случайный оклик  $x$ , вместо него оба используют  $h$ -образ сообщения  $M$ , где  $h$  — безопасная функция хеширования:

$h: \{0,1\}^* \rightarrow Z(q)$   $\{0,1\}^*$  — по-прежнему множество битовых последовательностей произвольной длины,  $Z(q)$  — подполе в рабочем поле.

Пусть  $p$  и  $q$  — большие простые числа,  $q$  делит  $p-1$ ;  $g$  — генератор подполя  $Z(q)$  в поле  $Z(p)$ , отыскиваемый следующим способом.

Выбираем произвольный элемент  $\alpha$  из поля  $Z(p)$ , вычисляем

$$g \equiv \alpha^{(p-1)/q} \pmod{p}$$

Если  $g=1$ , выбираем другой  $\alpha$ , и т.д. Выбранный таким образом генератор, очевидно, обладает свойством

$$g^q \equiv \alpha^{p-1} \equiv 1 \pmod{p}$$

поскольку последний переход является формулировкой малой теоремы Ферма.

### Генерация ключей

Алгоритм генерации ключей практически аналогичен для всего семейства Эль-Гамаль-подобных схем:

1. Выбирается  $m$  — произвольно выбранный элемент  $Z(q)$ ,  $0 < m < q$ .

2. Вычисляется

$$n = g^m \pmod{p}$$

3. Секретный ключ суть  $m$ ; публичный ключ — набор  $(p, q, g, n)$ .

Здесь  $p$  и  $q$  могут быть общими для всех пользователей, в отличие от схемы RSA, где они индивидуальны и секретны.

### Подпись

Пусть  $M$  — подписываемое Алисой сообщение произвольной длины из  $\{0,1\}^*$ .

1. Алиса выбирает случайное целое  $b$ ,  $0 < b < q$  (это и есть схема со стохастической параметризацией).

2. Алиса вычисляет  $c = g^b \pmod{p}$ ,  $x = h(M) \parallel c$  и  $mx + b \pmod{p}$ , где  $h$  — безопасная хеш-функция,  $m$  — секретный ключ ЭЦП Алисы;  $\parallel$  — конкатенация битовых последовательностей.

3. Цифровой подписью сообщения  $M$  является пара  $(x, y)$ .

### Проверка подписи

Чтобы проверить подпись  $(x, y)$  Алисы для сообщения  $M$ , Боб должен:

1. Иметь публичный ключ Алисы, набор  $(p, q, g, n)$ .

2. Вычислить  $c' = g^y n^{-x} \pmod{p}$ ,  $x' = h(M \parallel c')$

3. Боб считает подпись Алисы верной тогда и только тогда, когда  $x'=x$ .

### Доказательство состоятельности проверки

Если подпись создана Алисой, то  $c' \equiv g^y n^{-x} \equiv g^y g^{-mx} \equiv g^{y-mx} \equiv g^b \equiv c \pmod{p}$  следовательно,  $h(M \parallel c) = h(M \parallel c')$  и  $x=x'$ .

### Рекомендации по размерам рабочих полей

По состоянию на 1996 год существовали следующие рекомендации.

### Для RSA

Битовое представление  $n=pq$  должно иметь 768 разрядов для обычного применения, 1024 для коммерческого и 2048 — для особо важных систем.

### Для DSA и всего семейства схем Эль Гамала

Digital Signature Standard (DSS, иначе U.S. Federal Information Processing Standard, FIPS 186) устанавливает следующие ограничения на значения  $p, q$ :

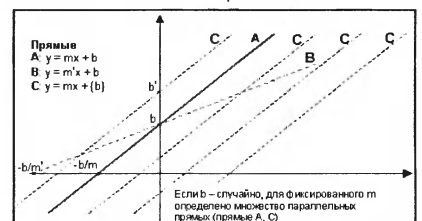
$$2^{159} < q < 2^{160},$$

$$2^{511+64t} < p < 2^{512+64t}, 0 \leq t \leq 8,$$

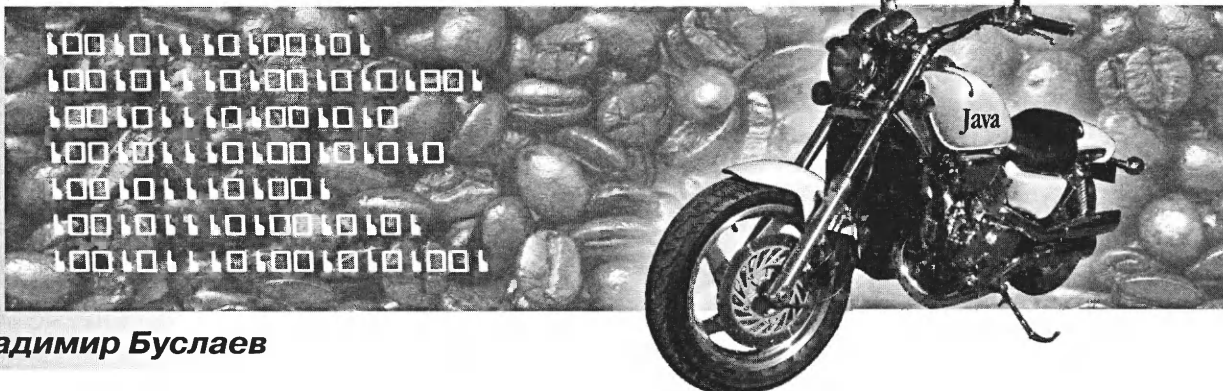
$$q \mid p-1$$

Рекомендованная длина битового значения  $p$  для семейства схем Эль-Гамала — не менее 768.

В следующий раз мы рассмотрим один из самых многообещающих на сегодня подходов к реализации ЭЦП и публичной криптографии вообще — к системам, построенным на эллиптических кривых.



История программирования



Владимир Буслаев

# “Кофейный” язык

**Я**зык Java сумел получить весьма заметную известность за последние несколько лет, так как ориентирован на самую популярную компьютерную среду — сеть Интернет и веб-серверы.

Если язык HTML обеспечивал “статическое” размещение страниц во Всемирной паутине, то появление Java привело к качественному скачку в создании интерактивных продуктов для сети Интернет.

Язык начал разрабатываться с 1991 года, однако рабочая версия (Java 1.0) появилась лишь в 1995 году. И в том же году на конференции в Бостоне о планах лицензирования Java объявили компании Lotus, Borland, Intuit, Macromedia, Spyglass и некоторые другие.

Технология Java (Кофе) родилась из проекта Oak (Дуб), основной целью которого была разработка объектно-ориентированных средств описания и коммуникации различного рода электронных устройств. Из-за неудачи этого проекта в 1994 году накопленный опыт было решено применить к интернет-ориентированным продуктам.

Рождению языка предшествовала довольно интересная история. В 1990 году один из разработчиков ПО компании Sun Microsystems Патрик Ноутон пришел к выводу о нецелесо-

образности поддержки сотни различных интерфейсов программ, используемых в компании, о чем и сообщил исполнительному директору Sun Microsystems Скотту МакНили.

МакНили попросил Ноутона составить список причин своего недовольства и выдвинуть любые разумные предложения. Ноутон, хотя и не рассчитывал на понимание, все же изложил свои претензии, беспощадно раскритиковав недостатки Sun Microsystems и, в частности, разрабатываемую в тот момент архитектуру ПО NeWS.

К удивлению Ноутона, его письмо возымело успех: оно было разослано всем ведущим инженерам Sun Microsystems, которые тут же откликнулись, выразив горячую поддержку своему коллеге и одобрение его взглядов на ситуацию в Sun Microsystems. Одобрило обращение и высшее руководство компании в лице ее основателя Билла Джоя и ведущего программиста Джеймса Гослинга.

Через некоторое время было принято решение о создании команды ведущих разработчиков с тем, чтобы они делали что угодно, но создали нечто необыкновенное.

Костяк команды составили Вейн Росинг, Майк Шеридан, Джеймс Гослинг и Патрик Ноутон. Первоначально группа, получившая кодовое наи-

менование Green, занималась исследованием бытовых устройств, устройств дистанционного управления, пыталась найти средство для взаимодействия между ними. Видеоманитофоны, проигрыватели лазерных дисков, стереосистемы и иные электроприборы были реализованы на разных процессорах, а это означало, что если производитель захочет добавить телевизору или видеоманитону дополнительные функции, он будет зажат в рамках средств, зашитых в аппаратное обеспечение.

В результате появилась идея нового подхода к разработке ПО для бытовой электроники. Команда тут же приступила к разработке нового объектно-ориентированного языка программирования, который был назван Oak (дуб), в честь дерева, росшего под окном Джеймса Гослинга. И уже в августе 1991 года группа продемонстрировала руководству компании основные идеи интерфейса пользователя и графическую систему.

В ноябре 1992 года компания Sun Microsystems преобразовала команду Green в компанию First Person.

Новая компания обладала интересной концепцией, но не могла найти ей подходящего применения. После ряда неудач ситуация нежи-

данно и резко изменилась: в апреле 1993 года был анонсирован Mosaic — первый "оконный" браузер для Интернета, с которого и началось бурное развитие WWW. Ноутон предложил использовать Oak в создании интернет-приложений. Так Oak стал самостоятельным продуктом. Вскоре были написаны Oak-компилятор и Oak-браузер WebRunner (1994 г.).

В 1995 году Sun Microsystems приняла решение объявить о новом продукте, переименовав его в Java (единственное разумное объяснение названию — любовь программистов к кофе). WebRunner был переименован в HotJava. Как только Java оказалась в лоне Интернета, компания Netscape перешла на поддержку Java-продуктов.

Три ключевых элемента объединились в технологии Java и сделали ее отличной от всего, существовавшего ранее:

- Java предоставляет для широкого использования свои апплеты — небольшие, динамичные, не зависящие от платформы активные сетевые приложения, встраиваемые в веб-страницы. Апплеты Java могут настраиваться и распространяться потребителям с такой же легкостью, как любые документы HTML;

- Java, будучи объектно-ориентированной технологией, сочетает в себе простой и знакомый синтаксис с надежной и удобной в работе средой разработки. Это позволяет широкому кругу программистов быстро создавать новые программы и апплеты;

- Java предоставляет программисту богатый набор классов объектов для ясного абстрагирования многих системных функций, используемых при работе с окнами, сетью и для ввода-вывода. Причем ключевая черта этих классов в том, что они обеспечивают создание независимых от используемой платформы абстракций для широкого спектра системных интерфейсов.

Программы, составленные на языке Java, по назначению можно разделить на две большие группы.

К первой группе относятся приложения Java, предназначенные для автономной работы под управлени-

ем специальной интерпретирующей машины Java. Реализации этой машины созданы для всех основных компьютерных платформ. Вторая группа — это апплеты.

**Апплет** — небольшая программа, динамически загружаемая по сети, точно так же, как картинка, звуковой файл или элемент мультипликации.

Главная особенность апплетов заключается в том, что они являются настоящими программами, а не очередным форматом файлов для хранения мультфильмов или какой-либо другой информации.

Апплет не просто проигрывает один и тот же сценарий, а реагирует на действия пользователя и может динамически менять свое поведение. Именно броские веб-страницы с анимацией привлекли большинство ранних приверженцев языка Java. Поскольку пользователи не сразу смогли полностью усвоить наиболее революционные аспекты Java, этот язык часто сравнивали с другими технологиями для загрузки динамических изображений и простого взаимодействия с веб-клиентами.

Приложения, относящиеся к первой группе, — это обычные автономные программы. Поскольку они не содержат машинного кода и работают под управлением специального интерпретатора, их производительность заметно ниже, чем у обычных программ, составленных, например, на языке C++. Однако не следует забывать, что программы Java без перетрансляции способны работать на любой платформе, что само по себе имеет большое значение в плане разработок для Интернета.

С помощью апплетов Java вы можете сделать HTML-страницы веб-сервера динамичными и интерактивными. Апплеты позволяют выполнять сложную локальную обработку данных, полученных от сервера или введенных пользователем с клавиатуры. Из соображений безопасности апплеты (в отличие от обычных приложений Java) не имеют никакого доступа к файловой системе локального компьютера. Все данные

для обработки они могут получить только от сервера.

Более сложную обработку данных можно выполнять, организовав взаимодействие между апплетами и веб-приложениями CGI и ISAPI. Для повышения производительности приложений Java в современных браузерах используется компиляция "на лету" — Just-In-Time compilation (JIT). При первой загрузке апплета его код транслируется в обычную исполняемую программу, которая сохраняется на диске и запускается. В результате общая скорость выполнения апплета Java увеличивается в несколько раз.

Язык Java объектно-ориентированный и поставляется с достаточно объемной библиотекой классов. Так же, как и в случае C++, библиотеки классов Java значительно упрощают разработку приложений, предоставляя в распоряжение программиста мощные средства решения распространенных задач. Поэтому программист может больше внимания уделить решению прикладных задач, а не таких, как, например, организация динамических массивов, взаимодействие с операционной системой или реализация элементов пользовательского интерфейса.

Впечатляет Java и сравнительной простотой (пожалуй, несколько проще, чем C/C++ или объектный Pascal), а описание языка занимает всего 60 страниц, его вполне можно прочитать за один вечер. Следует отметить, что простота программирования на этом языке во многом обусловлена особенностями среды Java, для которой и пишутся Java-программы.

В языке Java для решения задачи у вас будет совсем немного вариантов. Стремление к простоте зачастую приводило к созданию неэффективных и невыразительных языков типа командных интерпретаторов. Java к числу таких языков не относится.

Другой ключевой момент заключается в том, что Java дает защиту от несанкционированного доступа: программы на этом языке не могут вызывать глобальные функции и получать доступ к системным ресурсам.



# Письмо ведущему программисту из деревни

**З**дравствуй, сынок наш, Вася! Поздравляем тебя с днем рождения! Жаль, что ты опять не приехал. Сколько лет уж мы тебя не видели. Конечно, у тебя ответственная работа, высокий пост, мы тебя не упрекаем. Сегодня накрыли стол, посидели, вспомнили, каким ты был, как ты рос. Как будто это было вчера... Какими удивленными, широко раскрытыми глазами смотрел ты когда-то в монитор! Ты тогда даже еще говорить не умел, а уже хватал детальки ручками, жал на все кнопочки. Неумелый, неуклюжий, за тобой был нужен глаз да глаз. Все было в первый раз: первый удаленный файл, первая перезагрузка...

Потом ты научился говорить. А почему рабочий стол зеленый? А почему "мышь" назвали "мышью"? Милый, смешной почему-то, тебе все было интересно. Мы, бывало, так смеялись над твоими детскими вопросами...

Сегодня достали из чулана твои любимые игрушки. Вот пятидюймовая дискета, которую ты так любил форматировать. Вот твой старенький Courier на 2400, ты так его любил, все время слушал, как он шипит и курлычет, и даже на ночь клал с собой в постельку. А отец сидел рядом и читал вслух "Компьютер для чайников" — это была твоя самая любимая книжка...

Вспоминаются твои смешные детские страхи — как ты боялся черного экрана при загрузке, потому что не умел загружаться с дискеты... Твои маленькие победы — как ты прыгал и хлопал в ладоши, в первый раз установив Windows... Как ты капризничал и не хотел учить ассемблер — "не хочу, не буду". Да, мы заставляли тебя изучать скучные, но необходимые вещи, а тебе хотелось побегать с друзьями в Квейк, погонять мячик в

WorldSoccer. Ты тогда еще собирал скриншоты к Варкрафту, они до сих пор лежат в каком-то архиве, пароль от которого ты давно потерял...

Вот твои первые веб-странички на chat.ru: с орфографическими ошибками, неработающими скриптами и кучей баннеров. Ты про них давно позабыл, а мы часто заходим на них, они нам очень дороги...

Вот потерявший дистрибутив Linux — ты играл с ним, когда стал постарше... А вот и твоя первая программа: "Диагностика работы правой кнопки мыши v.1.0". Ты так гордился ею, все время носил на дискетке в нагрудном кармане. Ты тогда, глупенький, думал, что зарабатываешь на ней много-много денег...

А помнишь, как ты подрался с Вовой Булкиным в Mortal Combat и плакал, когда он тебя побил? А как таскал с ребятами провайдерские пароли у нашего соседа, дяди Кондрата, а он потом гонялся за вами по всему Интернету? Ты был шалуном, непоседой...

Твоя первая влюбленность, девочка, которой ты писал записочки на аську. Какая трагедия была для тебя, когда ты доверчиво открыл присланное ею письмо с темой "I love you", а потом... Это так больно, когда тебя заражает любимый человек... Но однажды настал момент, когда ты стал знать больше, чем мы. Ты так быстро вырос. Тебе становилось скучно с нами, мы ведь не знали, что такое Object PHP и Java, мы выросли во времена Бейсика и Коболы. Мы ничем не могли уже тебе помочь. И ты уехал...

Ты спрашиваешь, какие у нас новости? Все по-прежнему. Друзья



твои живы-здоровы, привет тебе передают. Булкин Володя работает в Управлении "Р" участковым милиционером. Катя вышла замуж, родила, сейчас устроилась веб-мастером на полставки.

Вот только Леночка... Она всегда была девочкой себе на уме, учиться не хотела, все искала, как половчее в жизни устроиться да легких денег заработать... Не знаем, правда или нет, сами не видели, но в чате говорили, что теперь у нее свой порносайт... Вот так. Ее дело, конечно, только родителей жалко.

Мы сами живем ничего. Цены, правда, растут. На e-mail еще хватает, а в музей лишний раз и не сходишь. Комплектующие берем на рынке — там дешевле. Хорошо еще за хостинг нам, пенсионерам, скидка, так что не жалуемся. Винчестер, правда, сыплется, ну да ничего. Одного нам хочется — на тебя посмотреть. У тебя, Васечка, конечно, и веб-камера хорошая, и канал мегабитный, только ты позабыл, что у нас-то связь модемная, окошечко в плеере маленькое, кадрики пропадают, так что тебя и не разглядишь толком. Ты бы приезжал сам? А не приедешь, пиши мылом, не забывай. Адрес все тот же: [derevnya@pochta.ru](mailto:derevnya@pochta.ru).

Целуем, твои любящие родители: мама aka Staruha и папа aka Starik.

*Олег Свешников*

# Max Payne

## Боль шла рядом...

Жил был на свете простой полицейский по имени Макс, по фамилии... Хм, не повезло ему с фамилией. Вроде бы ничего особенного, но очень уж она созвучна с английским словом Pain (боль). И где бы Макс ни был, всюду за ним следует боль. Женится, ребенка завел — вроде бы жить не тужить. Так нет, банда молодчиков, ворвавшаяся в дом Макса, отправляет жену и ребенка в мир иной. Следом за ними по идее должен был следовать Макс, но... ребята сами же получили предназначавшийся Максу билет на тот свет. И Макс начинает мстить мафии, которая тут же познает все прелести его фамилии. В комплекте с мафией поставляется куча оружия и мишеней для него и некая доля славы, проявляющаяся в том, что о Максе начинают говорить по всем телевизионным каналам (естественно, не в розовых красках)...

С врагами Макс у повезло. В самом начале своих приключений Макс выяснил, что этих мафиози вообще и убивать-то необязательно — достаточно дать им поругаться из-за детонатора, к примеру, и проблема решится сама собой в процессе дуэли. Остается только отправить победившего к проигравшему.

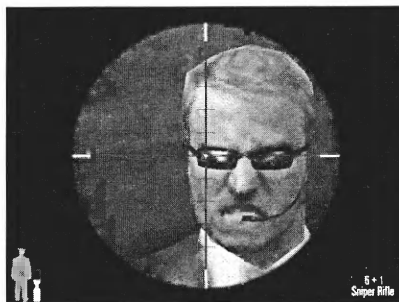
Море веселья в эту кашу добавляет русская мафия. Ее представители передвигаются на Мерседесе с номером vodka, говорят с акцентом, а в благодарность за оказанную им небольшую услугу снабжают Макса сумкой, полной оружия, что тут же делает нашего героя похожим на Нео (к "Матрице", кстати, мы еще вернемся).

Собственно, это и есть сюжет нового Third Person Shooter, который представили на наш суд финны из команды Remedy Entertainment. Сюжет, который преподносится нам в процессе игры с помощью великолепных комиксов, вставок (на движ-

ке игры), телефонных звонков и подслушанных разговоров, простой и стандартный, но в то же время захватывающий своей реализацией...

## Шоу во имя оригинальности

К сожалению, сегодня создать что-то в мире игр и не получить лейбл "клон" практически невозможно. Большинство жанров сложилось в середине 90-х, и шедевры тех лет помнят все. Но финны смогли сделать невозможное. То, что они создали, это даже не игра — это самое настоящее шоу. Красивое, жестокое, временами пугающее. Шоу, где нам предстоит не просто проходить уровни, убивать врагов, но делать все это красиво.



Достигается красота первым делом при помощи "slo-mo" — оригинального нововведения финнов, которое вскоре будет скопировано многими производителями игр. Вы "Матрицу" смотрели? Помните, как агент "уворачивался" от пуль? Вот и здесь нечто похожее. Игроку дана возможность замедлять течение времени. Стоит только нажать правую кнопку мыши, и мы видим, как неохотно двигаются враги, явно не успевая за Максом, как медленно пролетают пули разного калибра. И еще мы слышим монотонный стук сердца, обыденные звуки, только сильно замедлившись и ставшие от этого страшными. В общем, атмосфера еще та.

Поначалу "slo-mo" кажется немного неудобным и непривычным



из-за своей медлительности. Но потом начинаешь понимать, что враги еще медленнее, и что теперь можно спокойно решить, в кого стрелять и куда бежать, не торопясь прицеливаться и прикончить врага. А какие красивые получаются моменты! Прыжок с крыши дома на проезжающий мимо поезд в замедленном режиме не оставит никого равнодушным, как и полет вниз по лестнице наперегонки с гранатой.

Впрочем, за "slo-mo", как и за все на свете, приходится платить. Платить своеобразной энергией, которая дается за убитых врагов. Но не все так страшно: количество энергии, которое вам дают за одного врага, равняется тому количеству энергии, которое вам придется потратить на его убийство при самом худшем раскладе сил.

В сюжет вплетено достаточно много красивых и оригинальных элементов. К примеру, есть уровень, где Макса опаивают виски с наркотиком, и ему снится кошмар — интерактивный с явным уходом в аркаду. А когда Макс уключит наркотик, ему начнет казаться, что он — персонаж компьютерной игры, и что каждый его шаг кто-то контролирует (оригинальный у финнов юмор, правда?). Да и остальные уровни не отстают по насыщенности оригинальными эпизодами. К примеру, поездка на панорамном лифте, когда вас обстреливает вертолет...

Движок игры показывает нам во всей красе самые удачные моменты. При удачном выстреле камера преследует пулю до самого момента попадания во врага. Будьте уверены,

если вы красиво убили врага, то камера обязательно облетит вокруг него на прощание.

Финны четыре года вылизывали движок и довели его до совершенства. Текстуры, спецэффекты, анимация — все на высоте. Уровни интерактивны — кнопки нажимаются, телевизоры включаются, кассовые аппараты пищат и все это вместе взятое радостно гибнет под пулями "Беретты". Баллистика пуль просчитана до мелочей, так что будьте уверены: если выстрелили врагу в руку, то и след от попадания будет там же.

Единственное что разочаровало — это звук, вернее, музыка. Появляется она редко и ненадолго, все же остальное время приходится наслаждаться урбанистическим шумом и подслушанными диалогами бандитов. Несерьезно, господа.

### Нехорошие игрушки

Отдельная песня — оружие. Начинает игру Макс, имея обломок водопроводной трубы и верную "Беретту". Практически сразу можно обзавестись еще одной такой же, что позволит начать стрелять с двух рук и более мощным пистолетом Desert Eagle.

Впрочем, все перечисленное можно благополучно забыть, как только в ваших руках окажется стреляющий дробью помповик. Поверьте, именно он станет вашим самым главным другом первую половину игры, а затем уступит место Colt Commando. Почти сразу же после помпового товарища в руки Максy попадет сделанный из двустволки обрез — вещь неплохая, но скорострельность "два быстрых выстрела и долгая перезарядка" подходит только против одиночных врагов. Узиподобные "ингрэмы", которые

тоже можно использовать парами, появятся чуть позже. Хорошее, мощное оружие, но патроны к нему кончаются явно быстрее, чем Макс успевает подобрать новые. Следом появляется снайперская винтовка — оружие узкоспециализированное. А вот потом Максy достается Jackhammer — эдакий стреляющий очередями дробовик, поражающий своей разрушительной силой: убивает быстро и безболезненно. Так же быстро тратит патроны. Ну, а под конец игры, когда шинковать толпы врагов становится скучно, Максy подсовывают гранатомет M79 — ему уже все равно, сколько человек убивать, лишь бы все рядом стояли.



Не стоит забывать и о том, чем можно кидаться. В арсенале у Макса есть как обычные гранаты, так и хит сезона — Коктейль Молотов. Просто, сильно и со вкусом.

А вот с предметами, не предназначенными напрямую для убийства, у Макса туго. Только самое необходимое — аптечки (rainkillers по-ихнему — чувствуете юмор?), ключи и детонаторы.

### Жертвы

К сожалению (а может, и к счастью), чтобы сделать из игры шоу, программистам пришлось пойти на некоторые жертвы. Первое — это вид от третьего лица, что является явной жертвой удобства игры в пользу эффектности. Достаточно сказать, что стрелять прицельно в игре получается только при переходе в "slo-mo".

С "slo-mo" и прочими красотами связано еще несколько маленьких неудобств. Например, при переходе в "slo-mo" Макс совершает прыжок в ту сторону, куда он раньше бежал.

Это, конечно, красиво, но после прыжка он оказывается лежащим на полу, что влечет за собой утомительно долгий процесс принятия горизонтального положения на радость врагам, которые бегают вокруг и фаршируют беднягу свинцом. Ну, а если в "slo-mo" в Макса кинут гранату, так вообще ничего нельзя поделывать — остается только наблюдать замедленную смерть. Еще один момент связан со стрельбой из снайперской винтовки. После удачного выстрела камера обязательно следует за пулей, что очень раздражает, когда количество врагов больше одного. Ведь за эти секунды можно было бы спокойно прицелиться и убить второго врага. Что ж, красота требует жертв.

Ну, и самая главная жертва — отсутствие многопользовательского режима, что очень сильно сокращает срок жизни игры. Здесь все логично: реализовать игру нескольких человек, каждый из которых способен в любой момент замедлить время для себя, невозможно.

Концовку финны подпортили, но на это уже не обращаешь внимания. Max Payne — это такая игра, которую хочется проходить раз за разом. Этому способствует и наличие в игре нескольких уровней сложности, причем на самом последнем ограничивается время на прохождение уровня, а дополнительное время дается за убитых врагов.

В общем, радуется, что в игровой индустрии еще не перевелись люди, способные создать оригинальный, ни на что не похожий продукт.

*Андрей Александров*

### Хреновости

Из-за преступной халатности продавцов пиратских компакт-дисков едва не погиб 10-летний Вова Иванов, бесследно пропавший несколько дней назад. Только на четвертые сутки поисков спасателям удалось обнаружить его на 10-м уровне "Квейка" в критическом состоянии — у мальчика оставался один шотган с 4 патронами и всего 2 единицы жизни. Спасатели эвакуировали его с помощью команд "Save" и "Quit", о существовании которых продавцы не информировали ребенка.



# Американский футбол и российский Интернет

**М**ного воды утекло с тех пор, как Христофор Колумб открыл Америку. Янки "задвинули" испанцев на юг, французов на север, заставили британцев убраться восвояси на свой туманный Альбион. Где долларом, где хитростью, а где и силой США отобрали у Франции Луизиану, у Испании Флориду, у Мексики Калифорнию, а у России Аляску. Повоевав за независимость негров, Америка продолжала свое государственное самоутверждение, в котором оставался один большой пробел: ковбоям и зверобоям хотелось иметь национальные виды спорта. Американцы столь скоро адаптировали бейсбол на своих полях, что человечество уже забыло британские корни этого изобретения. Но и этого им показалось мало. Нужен был спорт, который бы соот-

ветствовал национальному духу Америки, был бы школой жизни, где преподают этику конкуренции и устремленность к успеху. Таким видом спорта и стал американский футбол.

Родиной американского футбола называют Гарвардский университет, где в 1874 году прошла первая игра. Дальше по нарастающей игра приобретает популярность и в 1920 году получает официальный статус и свою профессиональную лигу — National Football League (NFL). Она и сейчас является вершиной организации американского футбола. Чуть ниже стоит еще несколько американских лиг. В футбол играют в американских университетах, а начинается все обычно еще в средней школе. Наверняка в США нет ни одного



учебного заведения, у которого не было бы своей команды.

Лишь на первый взгляд игра кажется предельно примитивной. После ознакомления с правилами я насчитал 16 разных специализаций игроков — разыгрывающие, блокировщики, широкие принимающие, задние бегущие, бьющие. Сценарий игры, несмотря на кажущийся хаос, довольно строго расписан.

Первое, что привлекает сторон-

## Как я писал 300 дисков подряд

**К**ак сказал бы поэт, "CD — как много в этом слове...". Действительно: игры, музыка, видео, личные архивы и коллекции, все это можно записать на обычный CD. Чистый диск (иноземцы называют его "blank") стал привычным товаром и приобрел еще одно название — "болванка". Будучи счастливым обладателем CDR и коллекционером софта, я часто покупал столь нужные мне диски, причем делал это в первом попавшемся магазине, так как цена на них разнится на несколько десятков копеек в обе стороны. Поскольку по своей природе я еще и консерватор, всегда брал болванки по 650 Мб (74 минуты) как давно устоявшийся стандарт, не обращая внимания на новомодные 700-метровые творе-

ния лидеров дископроизводящей промышленности. Из фирм-производителей предпочитал только известных: Maxell, Memorex, TDK, Verbatim, ну, в крайнем случае, Philips. На всякое китайское чудо типа Mitsui и прочие "HoName" мною было наложено табу.

Как-то раз подарили мне упаковку из 3 дисков от фирмы Mirex (Gold, Silver и Pro). После экспериментов с ними я решил: "Золото детям, остальное СЕБЕ!"

При цене на несколько рублей ниже, чем, например, Verbatim или Memorex, Mirex оказался ничем не хуже их и занял почетное место в моей "табели о рангах". Особенно было приятно, что "Mirex" — это наш,

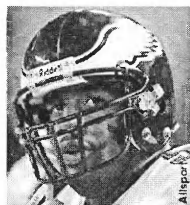


отечественный продукт: "Made in Е-бург". Все так продолжалось бы и далее, если бы я записывал на диски лишь скачанную из Интернета информацию.

Но как-то раз мне предложили скопировать для себя большую (более 300 наименований) коллекцию фильмов в новомодном формате

него наблюдателя в американском футболе, — это амуниция. Почти рыцарский шлем с забралом, огромный панцирь, делающий из самого хилого очкарика Геракла, обтягивающие штаны, яркая форма. Есть даже капа, которая как и у боксеров, защищает зубы игроков от преждевременного выпадения. Вторая составляющая успеха — бескомпромиссность этого спорта. Вот уж плацдарм для проявления мужской силы, храбрости и удали. Здесь к вам не подбегит какой-нибудь мерзавец в черном со свисточком и явным намерением удалить вас с поля за то, что вы ласково убрали соперника, корчащегося рядом от приступов боли.

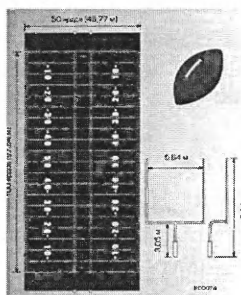
Задача — доставить кожаную дыню в зачетную зону команды соперника любой ценой. Правда, кадры из известного фильма с Брюсом Уиллисом — это, пожалуй, перебор. Любимое изобретение полковника Кольта принято оставлять дома. Наконец, третье проявление Америки — демократия. Количество замен по ходу матчей не ограничено: за одну игру на поле могут выйти до полусотни игроков. Это вам не европейский футбол, где 22 человека иг-



рают, а несколько тысяч смотрят. Как бы скверно ты ни играл, у тебя есть шанс выйти на поле, пусть и на несколько минут.

Любителям американского футбола и просто желающим проникнуться хотя бы частью того духа, который позволяет американцам забрасывать все дела и лхнуть к телевизорам с трансляциями американского футбола, одна дорога — в Интернет.

Начнем с лучшего: [www.nfl.com](http://www.nfl.com) — огромный ресурс, посвященный лучшей лиге американского футбола. На этом сайте для пропаганды игры используются все средства Интернета — чаты, видеоролики, прямые трансляции, фотографии и горячие новости. Даже сейчас, в период межсезонья, сайт обновляется несколько раз в день. Здесь можно купить билеты на игры в режиме онлайн, посетить Зал Славы, узнать о последних успехах американского футбола за рубежом, любую статистику за годы су-



ществования NFL и даже купить сувениры и личные вещи футболистов с виртуального аукциона.

Все 34 команды лиги имеют собственные веб-сайты: [www.giants.com](http://www.giants.com), [www.colts.com](http://www.colts.com), [www.buffalobills.com](http://www.buffalobills.com), [www.raiders.com](http://www.raiders.com), [www.azcardinals.com](http://www.azcardinals.com), [www.redskins.com](http://www.redskins.com), [www.jaguars.com](http://www.jaguars.com). Индустрия поставлена на широкую американскую ногу. Недаром последний финал Лиги, который называется "Супер Боул", собрал 125 миллионов зрителей. А это каждый второй житель США (!). Видимо, за исключением слепых, грудных младенцев, европейцев и женщин. В 1985 году, когда финал случайно совпал с инаугурацией Рональда Рейгана, в Белом

Доме не долго думая... перенесли на день инаугурацию. Следующий "Супер Боул" состоится 27 января 2002 года в Новом Орлеане. Желающие могут уже сейчас подумать о билетах, которые стоят от 150 долларов в кассах до десятков тысяч у "снимателей скальпов" — так

американцы именуют перекупщиков и спекулянтов билетами.

MPEG-4. Просмотрев принесенные диски, я пришел в ужас — только 10-12 фильмов было в пределах 650 мегабайт. 295 фильмов имели размеры от 680 до 700 мегабайт, и их можно было записать хотя бы на столь нелюбимые 700-метровые болванки (онье, впрочем, оказались ничем не хуже 650-метровых). Но вот оставшиеся фильмы были уж очень немаленькие — от 702 до 730 метров, и с ними можно было сделать только три вещи: 1) отдать обратно и забыть 2) списать их на HDD и оставить там до лучших времен и 3) обрезать в Адобовском "премьере" кусочек фильма в конце (титры). Я решил не уродовать фильмы и сохранить их пока на винчестере, благо его объем позволяет.

Конечно, вы можете спросить, а зачем все это было нужно, не проще ли было просто купить готовый диск с фильмом? Отвечаю: пиратский диск, в зависимости от места приоб-

ретения, стоит около 60 рублей, а чистые болванки при мелкооптовых закупках обходятся в 16—18 рублей. 300 шт. x 60 р. = 18000, 300 шт. x 18 р. = 5400, 18000—5400 = 12600 р. экономии! Комментарии, как говорится, излишни, особенно если учитывать, что CDRW (16x12x32x) стоит примерно 4800 рублей, а на перезапись одного диска на максимальной скорости уходит 4,5 минуты.

В ближайший выходной я поехал на рынок в Автово, где в среднем на 5 рублей дешевле, чем в центре города (17 рублей против 22), приобрел 300 штук 650- и 700-метровых болванок уважаемых мною фирм. В виде подарка у продавца был отобран "Техноподиум", который я стал изучать по дороге домой. Особое внимание привлек раздел, посвященный дискам для записи. В нем некая фирма, назовем ее "КардМедиа" предлагала мелким оптом диски неизвестной мне ранее фирмы

"Rostok Media". Предлагались диски как в коробке, так и без нее, TPEX типов — 650, 700 и 780 Мб по цене 35—37 центов. Позвонив им, я поинтересовался, во сколько мне обойдутся 10 дисков по 780 Мб. Названная цена хоть и была выше рекламной (в рекламе указана цена диска без упаковки), но ненамного — всего 61 цент, что в свою очередь было даже дешевле, чем 700-метровые диски на "Юноне". Но дешевое не всегда синоним хорошего, и я решил купить 10 дисков на пробу (10 дисков — это уже мелкий опт).

Дома я сел за компьютер с естественным желанием поскорее переписать столь нужные мне фильмы. Для начала решил протестировать новинку. Поставил диск на запись и пошел пить кофе. Вернувшись через десять минут, я обнаружил, что диск... умер в процессе финализации. Погоревав об утрате 17 рублей и целых 10 минут жизни, я повесил



Не стоит думать, что американский футбол есть только в США. В Европе тоже играют в сей вид спорта. Правда, куда менее успешно. Обычно дней за десять до "Супер Боула" американцы приглашают к себе в гости сборные Канады, Японии и Европы. На последний турнир в сборную Европы даже попали несколько русских игроков. Впрочем, янки обычно выносят своих конкурентов в прямом смысле этого слова. Однако подробности европейских и мировых дел можно узнать на сайте [www.football.com](http://www.football.com). Всего 9 раз разыгрывался европейский трофей World Bowl. Несколько лучших европейских команд на протяжении десяти недель играют между собой, определяя достойных финалистов. В этом году финал проходил 30 июня в Амстердаме, где победу праздновали испанцы. Российские клубы пока не участвуют в этом турнире.

В России американский футбол долгое время был под запретом как проявление буржуазного образа жизни. Единичные счастливицы смотрели его за границей и иногда привозили видеокассеты с запрещенной игрой, диссидентства в уз-

ком кругу на кухне. Лишь в 1989 году американцам удалось провести показательный футбольный матч в России. Десять тысяч зрителей стали свидетелями непонятого, но очень захватывающего спортивного спектакля. Но и это не стало толчком для развития массового американского футбола в России. Лишь в середине девяностых стали появляться первые ласточки-команды: "Добрые Молодцы", "Московские Орлы", "Каспийские Сфинксы", "Московские Богатыри". Эпидемия быстро распространилась по территории бывшего СССР, захватив Украину и Белоруссию. Обо всех встречах этих команд можно прочитать на крупном ресурсе <http://www.kulichki.com/claf/rusversion/4.html>. Информация копилась здесь два года. Имеется большое количество фотографий и пресс-релизов, подробный ликбез по правилам и организации американского футбола. Однако, похоже, что сейчас сервер заброшен и обновляться перестал. Единственное его предназначение — большой информационный архив.

Зато 2 августа в нашей стране появился по-настоящему професси-

ональный ресурс [www.americanfootball.ru](http://www.americanfootball.ru) — плод совместного творчества "Детской Лиги Американского Футбола" (ДЛАФ) и интернет-агентства [iwda.group](http://iwda.group). Одиннадцать тысяч посещений за десять дней — неплохой показатель. Сайтом занимаются настоящие профессионалы и энтузиасты этого дела, игроки, тренеры. Создана и работает рассылка (<http://subscribe.ru/catalog/sport.soccer.amfootball/>). Новости ежедневно пополняются.

Как ни странно, отдельных сайтов самих команд в Интернете нет. Видимо, американский футбол и российский Интернет пока еще далеки друг от друга. Или люди, которые ими занимаются, не могут совместить эти два явления. Судя по биографии, многие из них пришли в футбол из силовых видов спорта и очень редко с университетской скамьи. Но, думаю, долго ждать не придется, и в скором будущем фанаты смогут оставлять свои записи в гостевых книгах "Московских Патриотов" или "Минских Зубров".

*Арсений Ефремов*

убитый диск на стену и поставил на запись проверенный временем 650-метровый диск, который так же благополучно скончался. Первая мысль: мне подсунили "левые" диски?! Взяв несколько дисков, я отправился к знакомому, у которого все взятые диски были благополучно записаны. Мысль номер два: умирает рекордер. Дабы не паниковать понапрасну, я решил провести эксперимент: стал уменьшать скорость записи (постепенно доведя до первой скорости) и записывать не напрямую с диска на диск, а через промежуточный образ-диск. Результатом этих действий стала безвременная кончина еще нескольких болванок. Мдяя... рекордер. Пришлось еще раз потревожить знакомого и взять его CDR. Но каково же было мое удивление, когда и он отказался работать на моей машине. И тут меня посетила умная мысль — Windows

ключит! Через три часа я переустановил на свежееотформатированный винчестер систему с нуля, со всеми нужными мне приложениями (а их о-о-очень много). И все снова залетало — все без исключения диски (даже 780 метров) записались без малейшего сбоя на максимальной скорости.

Проведенное позже расследование показало, что переустановка системы привела к уменьшению размера папки "Winnt" на 180 Мб. Это с учетом того, что свап-файл у меня находится на другом диске. Когда выяснилось, что похудела папка System 32, я понял, в чем была причина сбоя: когда-то я ставил много разных программ, которые почти сразу и убивал. Но они убивались не полностью, а оставляли свои остатки в системной папке. И постепенно "трупный яд" отравил весь организм. Но на ошибках учатся, те-

перь перед инсталляцией всех программ я запускаю программу Fix-it, которая отслеживает ВСЕ изменения, вносимые устанавливаемой программой, а при необходимости удаляет все следы в соответствии с логом, что часто не под силу стандартному деинсталлятору.

Но вернемся к дискам. Записанные мною отечественные болванки живы и нормально читаются, что не может не радовать. К сожалению, их достаточно мало на рынке, а потому в магазинах их увидишь нечасто, но, как мне кажется, скоро они вытеснят дешевые (и часто невысокого качества) китайские подделки. И тогда, придя в магазин и обратившись к продавцу с просьбой показать вам самые дешевые и качественные диски, вы увидите стенд, заставленный дисками с надписью "Made in Russia".

*Георгий Баранов*



**М**обильные **К**омпьютеры

**Диалектика®**

[www.dialectica.ru](http://www.dialectica.ru)

ул. Ал. Невского, 6, т. 327-8228



Портативные компьютеры  
Панельные компьютеры  
Жидкокристаллические мониторы

Миникомпьютеры  
Цифровые фотокамеры  
Сотовые телефоны

В специализированном салоне портативной компьютерной техники

# Пора менять железо!



**КОМПЬЮТЕР ATLANT 3000 tm**

**Intel® Pentium®4 processor**

**1300-1700 MHz - от 749**

mb (ATX) с i850 chipset, DDRAM 128Mb, GeForce 32Mb, HDD 20 Gb, FDD 1,44, CD-ROM 48x, SB128, keyboard, mouse + Windows © ME + Русский Офис



На все компьютеры устанавливаются только лицензионные продукты Microsoft®

## ПОДАРКИ

**ВСЕМ ПОКУПАТЕЛЯМ**

компьютера с предустановленными:  
Microsoft® Windows® 95,  
Microsoft® Windows® ME,  
Microsoft® Windows® XP Home,  
Microsoft® Windows® 2000 Prof.,  
Microsoft® Windows® XP Prof.

**ATHLETE 3000 tm Intel® Pentium®III processor 667 - 1000 MHz - от 495**

mb Intel® (ATX), SDRAM 64Mb, RIVA TNT2-32Mb M 64, FDD 1,44, HDD 10,2 GB, CD-ROM 48x, SB 128, kb., mouse, Windows © 98

**KIDDY 2000 tm Intel® Celeron™ processor 633 - 800 MHz - от 279**

mb (ATX), SDRAM 64Mb, SVGA от 4Mb, HDD от 10 Gb UDMA66, FDD 1,44, CD-ROM от 48-x, Sb 16, kb., mouse.

при покупке с компьютером **СКИДКИ НА МОНИТОРЫ СТХ**

Настоящая  
двухлетняя гарантия  
Любое изменение  
конфигураций  
Бесплатная модернизация  
в течение гарантийного срока  
Доставка по  
Санкт Петербургу

интернет - магазин [www.svegaplus.ru](http://www.svegaplus.ru)

без обеда с 10 до 20 в субботу и воскресенье с 11 до 19

**9-я линия, 56 тел. 327-4630**

**наб.р. Фонтанки, 120**

тел. 259-9109, 259-9107, 251-1872, 251-1892

**НОВЫЙ МАГАЗИН Разъезжая, 36 тел. 112-3661**



**Свега+**



Авторизованный поставщик решений Intel®

Логотипы Intel Inside и Pentium являются зарегистрированными торговыми знаками, а Celeron торговым знаком Intel Corporation