



Тайм-менеджмент

для системных администраторов

НАПИСАНИЕ на ЗАКАЗ:

1. Дипломы, курсовые, чертежи...
2. Диссертации и научные работы.
3. Школьные задания.

Онлайн-консультации.

ЛЮБАЯ тематика,
в том числе ТЕХНИКА.

Приглашаем авторов.

УЧЕБНИКИ, ДИПЛОМЫ, ДИССЕРТАЦИИ:

полные тексты в электронной библиотеке

www.учебники.информ2000.рф.

Тайм-менеджмент

для системных администраторов

Томас Лимончелли



Санкт-Петербург — Москва
2007

Томас Лимончелли

Тайм-менеджмент для системных администраторов

Перевод С. Иноземцева

Главный редактор
Зав. редакцией
Научный редактор
Редактор
Корректор
Верстка

А. Галунов
Н. Макарова
О. Цилюрик
А. Кузнецов
О. Макарова
Д. Орлова

Лимончелли Т.

Тайм-менеджмент для системных администраторов. - Пер. с англ. - СПб: Символ-Плюс, 2007. - 240 с, ил.
ISBN 5-93286-090-1

По тайм-менеджменту изданы сотни книг, но только эта написана сисадмином для сисадминов. Автор учитывает специфику их труда: работая над долгосрочными проектами, сисадмины вынуждены постоянно прерываться, чтобы наладить технику, помочь пользователям. И даже овладев всеми тонкостями профессий, сисадмин задерживается по вечерам и работает по выходным.

Основное внимание уделяется приемам, которые помогут сисадминам не только вести повседневные дела, но и справляться с неизбежными критическими ситуациями. Вы научитесь управлять отвлекающими факторами, исключать непроизводительные затраты времени, вести список дел, разрабатывать процедуры для регулярно совершаемых действий, сосредотачиваться на текущей задаче, расставлять приоритеты в соответствии с ожиданиями клиентов, документировать и автоматизировать рабочие процессы для ускорения их выполнения. Издание насыщено примерами, взятыми автором из его долгой карьеры, на протяжении которой он занимался обслуживанием рабочих станций и серверов, а также разработкой ПО и систем безопасности. А это значит, что читатель получит советы бывалого человека. Томас давно занимается проблемой тайм-менеджмента, ведет обучающие семинары. Приемы, которыми он делится с коллегами, проверены на практике. Они работают!

ISBN-13:978-5-93286-090-8

ISBN-10:5-93286-090-1

ISBN 0-596-00783-3(англ)

© Издательство Символ-Плюс, 2007

Authorized translation of the English edition © 2006 O'Reilly Media, Inc. This translation is published and sold by permission of O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

Все права на данное издание защищены Законодательством РФ, включая право на полное или частичное воспроизведение в любой форме. Все товарные знаки или зарегистрированные товарные знаки, упоминаемые в настоящем издании, являются собственностью соответствующих фирм.

Издательство «Символ-Плюс». 199034, Санкт-Петербург, 16 линия, 7,
тел. (812) 324-5353, edit@symbol.ru. Лицензия ЛП N 000054 от 25.12.98.

Налоговая льгота - общероссийский классификатор продукции

ОК 005-93, том 2; 953000 - книги и брошюры.

Подписано в печать 26.04.2007. Формат 70x100'Дв. Печать офсетная.

Объем 15 печ. л. Тираж 2000 экз. Заказ N 1230

Отпечатано с готовых диапозитивов в ГУП «Типография «Наука»
199034, Санкт-Петербург, 9 линия, 12.

www.natahaus.ru
Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

Оглавление

Об авторе	8
Вступительное слово	9
Предисловие	12
1. Принципы тайм-менеджмента	23
Трудности тайм-менеджмента	25
Принципы тайм-менеджмента для системных администраторов	26
Будет нелегко	31
Резюме	31
2. Сосредоточенность как средство против прерываний	34
Сосредоточенный мозг	35
Обстановка, способствующая сосредоточенности	37
Прерывания	43
Перенаправление прерываний	43
Как прогнать клиента, не нахамив ему	45
Резюме	51
3. Рутинные процедуры	54
Примеры процедур	55
Как разрабатывать собственные процедуры	66
Удаление старых процедур	69
Резюме	69
4. Система Цикл	71
Не доверяйте своему мозгу	72
Почему другие системы не работают	74
Успешные системы	76
Цикл	76
Резюме	79

5. Система Цикл: списки дел и расписания	81
Типичный день	82
Полезные советы	95
Настройка бумажного органайзера для системы Цикл	97
Настройка портативного компьютера для системы Цикл	98
Резюме	101
6. Система Цикл: работа с календарем	103
Как пользоваться календарем	104
Один календарь для деловой и личной жизни	107
Повторяющиеся задачи	107
Узнайте свой личный ритм	110
Узнайте ритм своей фирмы	111
Резюме	113
7. Система Цикл: долгосрочные цели	115
Секретный прием	117
Постановка целей	118
Планирование последующих шагов	120
Занесение шагов в расписание	121
Регулярный просмотр списка целей	122
Резюме	123
8. Расстановка приоритетов	124
Расстановка приоритетов в списке дел	124
Приоритеты проектов	130
Запросы, поступающие от начальства	132
Резюме	137
9. Как справиться со стрессом	138
Перегруженность и конфликтующие указания	139
Отпуск	142
Йога, медитация и массаж	144
Резюме	146
10. Управление электронной почтой	148
Обработка вашей электронной почты	148
Приступайте, не медля	155
Резюме	156
11. Исключение непроизводительных затрат времени	158
Что такое непроизводительные затраты времени?	159
Как избежать соблазнительных затрат времени	159

Распространенные причины непроизводительных затрат времени	160
Непроизводительные совещания	162
Стратегия и тактика	168
Резюме	169
12. Документация	171
Документация, которая нужна вам	172
Технология Wiki	178
Резюме	181
13. Автоматизация	183
Что автоматизировать?	184
Процесс автоматизации	186
Простые задачи, выполняемые часто	189
Сложные задачи, выполняемые однократно	201
Предоставление другим пользователям прав суперпользователя	214
Резюме	217
Эпилог	219
Алфавитный указатель	223

Об авторе

Томас Лимончелли (Thomas A. Limoncelli) - признанный во всем мире автор и лектор, специалист во многих областях, включая системное администрирование, компьютерные сети и компьютерную безопасность. Работает системным администратором с 1988 года, периодически выступает с докладами на конференциях, посвященных самым разным темам - от брандмауэров и безопасности до тайм-менеджмента. Томас работал в Cibernet, Dean For America, Lumeta, Bell Labs/Lucent, AT&T и Mentor Graphics. В соавторстве с Кристиной Хоган (Christine Hogan) написал книгу «The Practice of System and Network Administration» (Практика системного и сетевого администрирования), Addison Wesley. Закончил Университет Дрю (Drew University) в Мэдисоне, штат Нью-Джерси, по специальности «Информатика». Томас Лимончелли ведет блог на сайте www.EverythingSysadmin.com.

Вступительное слово

Письмо самому себе:

Дорогой Я, (а как еще начать?)

Не забудь модернизировать сервер LDAP. Не забудь поставить заплату на брешь в безопасности zlib и на каждый пакет, использующий эту библиотеку. (Впрочем, если задуматься, есть ли пакеты, которые ее *не* используют?) Не забудь запланировать очередное 10-кратное увеличение объема внешней памяти. Не забудь решить проблему, возникшую у шефа с Outlook, или, по крайней мере, подготовить все необходимое для ее решения. Не забудь переустановить Oracle. Проверь, есть ли реализация Wiki-технологии лучше той, которую вы используете. Переделай систему регистрации пользователей и на этот раз убедись, что она корректно ведет себя в ситуациях, которые кажутся тебе физически невозможными. Убедись, что она не нарушает закон Sarbanes-Oxley, удовлетворяет требованиям ISO9000 и соблюдает кашрут. Проверь, как идет работа над проектами твоих подчиненных. Прочитай LISA за последние два года, чтобы убедиться, что ничто полезное для вашей инфраструктуры не упущено. Потом, если у тебя останется время, начни составлять план на следующую неделю.

Не стоит беспокоиться из-за того, что пункту «уйти в отпуск» вот уже 73 недели подряд нет места в твоем списке. Также не следует обижаться на жену, которая буквально покатила со смеху, когда услышала, что ты собираешься написать вступительное слово к книге по тайм-менеджменту. Или все-таки обидеться?

Возможно, тебя вдохновят слова Генри Киссинджера: «На следующей неделе кризиса быть не может. В моем расписании уже нет места».

Ладно, чего там. Возвращайся к работе.

Искренне твой,

я

Вам знакома эта картина (ну, кроме эпизода с моей женой, действительно имевшего место)?

Первая книга Тома Лимончелли, написанная им в соавторстве с Кристиной Хоган (Christine Hogan) (теперь Лир (Leag)), стала ключевой работой в области системного администрирования. «The Practice of System

and Network Administration» (Практика системного и сетевого администрирования) учит читателя строить разумную и организованную инфраструктуру, следуя отличным практическим рекомендациям. Но только в одной главе той книги говорится, как самому при этом остаться разумным и организованным. Книга, которая перед вами, как раз об этом.

Но почему системным администраторам нужна собственная книга по тайм-менеджменту? Думаю, за все эти годы я прочитал слишком много общих работ по этой теме. В данном издании Том блестяще раскрывает те грани системного администрирования, которые делают тайм-менеджмент в нашей профессии особенно сложным. В добавление к этому мне хотелось бы привести несколько черт типичного системного администратора, еще больше усложняющих ситуацию.

Во-первых, большинство системных администраторов проявляют исключительное упорство в решении проблем. Они вцепляются в проблему, как бульдоги, и не бросают ее, пока не добьются своего. Другие дела, такие как запланированные встречи или поддержание жизнедеятельности организма (например, еда и сон), отходят на второй план, а работа над проблемой, реальная или мысленная, продолжается, не считаясь ни с какими временными рамками. Для человека, который постоянно твердит «Еще секундочку, я почти разобрался», управление временем является почти непосильной задачей.

Вторая характерная особенность, которую я подметил в себе и моих коллегах, - искреннее желание помочь людям, поддержать их в общении с недружелюбной и не прощающей ошибок техникой и заставить ее работать так, чтобы у них все получалось. Черта сама по себе похвальная, но, заметив, что вы хотите и можете помочь, вас будут просить еще и еще. Люди всегда идут по пути наименьшего сопротивления, и жизнь в конце концов становится тем, что я называю «сплошная техподдержка». Пока была жива моя бабушка, я время от времени приезжал к ней во Флориду. Всякий раз, когда бы я ни появлялся, она и все ее друзья приносили мне свои электронные часы, чтобы установить точное время. И знаете, мне это нравилось. Все-таки жизнь не всегда идет только по плану, раз к вам в любой момент могут обратиться за помощью. Держу пари, у Супермена тоже были проблемы с тайм-менеджментом.

С готовностью сисадминов прийти на помощь тесно связано их желание поучаствовать в разрешении любого кризиса и спасти положение. Они не способны пробить лбом стену, но если бы могли, то обязательно сделали бы это. Идеализированный образ рыцаря-спешащего-на-помощь, безусловно, весьма привлекателен для новичков, но не является достаточно убедительной причиной оставаться на этой работе долгие годы.

Последняя черта типичного сисадмина, которую я хочу упомянуть, тоже похвальна сама по себе, но вызывает раздражение у его родных и близких и обращает в прах все его попытки применить тайм-менедж-

мент. Большинство системных администраторов получает удовольствие от своей работы. Монтаж оборудования, установка и переустановка программ, построение сетей, «копашение» в компьютере и т. д. - развлечение для них. Они занимаются этим целый день на работе, продолжая делать то же самое дома.

Я как-то ехал в междугороднем автобусе, и моей соседкой оказалась шеф-повар какого-то ресторана. Она заявила, что ненавидит готовить дома. Еще она сказала, что почтальоны, будучи в отпуске, не любят долго гулять пешком. Большинство системных администраторов, которых я знаю, подобные идеи покажутся дикими. Придя к ним домой, вы застанете их (и меня, как справедливо заметила моя жена) склонившимися над ноутбуком, и так они проводят практически все свободное время. Для системного администратора понятия «развлечения» и «работа» больше всего похожи на расплывчатую квантовую суперпозицию. Это замечательно, что мы любим свою работу, но это и ужасно, потому что мы не можем (и не хотим) прерывать ее. Трудно управлять своим временем, когда оно до такой степени неструктурировано.

Значит, надежды нет? К счастью, это не так. Тайм-менеджмент для системных администраторов был бы пустой затеей, но их поддерживают два решающих фактора:

1. Они сами.
2. Том Лимончелли.

Как я уже говорил, системные администраторы обожают чинить, организовывать, встраивать и оптимизировать. Однажды я наблюдал, как мой друг системный администратор стоял в очереди в кассу в супермаркете и переключал покупки в корзине. Он внимательно осматривал каждый предмет и клал его на подходящее место, словно играл в трехмерный Тетрис. Если бы мы только могли обернуть это мастерство себе на пользу, чтобы преодолеть трудности в организации своего времени, с которыми мы сталкиваемся...

А ведь мы это можем! Том поможет нам. Он знает, как это сделать. Том много лет работает над проблемой, связанной с тем, как организовать время и преуспеть в нашей профессии. С тех пор как мы с ним познакомились лет десять назад на моей первой конференции LISA, я имею честь наблюдать, как он решает этот вопрос в самых разных контекстах - от разбиения сети AT&T Bell Lab на две части до поддержки технологической инфраструктуры избирательного штаба одного из кандидатов на политическую должность. В любой ситуации он способен внести в работу многолетний опыт системного администрирования, прекрасное знание людей и здоровое чувство юмора.

А теперь сядьте поудобнее, пристегните ремни и радуйтесь - Том поможет привнести тайм-менеджмент и здравый смысл в вашу жизнь.

- Дэвид Н. Бланк-Эдельман (David N. Blank-Edelman)
Сентябрь, 2005 (сисадмин с 20-летним стажем)

Предисловие

- «Тайм-менеджмент для системных администраторов»?
 - Угу.
 - Будете рассказывать о том, как пользоваться карманными компьютерами, электронным календарем vCal, серверами календарей и прочей дребеденью?
 - Вообще нет. В подобных вещах системный администратор разберется и без книг.
 - Тогда почему бы мне просто не купить одну из 10 миллионов других книг по тайм-менеджменту?
 - Потому что они не годятся. Ладно, они *годятся*. Только не «нам». Они обращаются к некоему обобщенному читателю, которым мы с вами не являемся. Я - профи. Системный администратор. Эксперт по сетям. После работы я живу почти так же, как на работе, - посмотрите, какой убийственный сервер я установил дома. Как только доведу его до ума, на работе сделаю то же самое. Других таких профессий поискать. Нейрохирург, придя домой, не пытается испробовать новую методику на своем коте в надежде на то, что при удачном исходе ее можно будет применять на пациентах.
 - (Выгоняя кота из комнаты.) Я больше не подпущу вас к своему коту!
 - Послушайте, я лишь пытаюсь сказать, что системное администрирование не работа. Это стиль жизни. Нам нужны книги по тайм-менеджменту, в которых говорится о нашем стиле жизни, используется привычная нам терминология и решаются наши проблемы.
 - Стиль жизни?
 - Стиль жизни, стиль работы, как вам больше понравится. Ни в какой другой профессии человеку не приходится буквально разрываться на части. Пользователи непрерывно отвлекают нас просьбами, не давая ни одно дело довести до конца. У компьютеров свои запросы, их мы тоже должны удовлетворять. Наше начальство хочет, чтобы мы реализовали долгосрочные проекты, но одновременно заваливает нас требованиями устранить мелкие неполадки, так что работать над долгосрочными проектами абсолютно невозможно!
- В нашей профессии трудно встретить хорошего наставника. Если шеф технарь, он может помочь в решении технических вопросов, но не в об-

ласти тайм-менеджмента. Если же он далек от техники, то не может быть наставником, потому что «не въезжает» в требования, предъявляемые нашей работой.

- А как вы достигли такой высокой квалификации?

- Ну, во-первых, когда-то я прослушал кучу курсов по тайм-менеджменту и понял, что 80% того, чему меня учили, не имеет отношения к сисадминам. Однако я не растерял остальные 20%. С годами я усовершенствовал эти приемы, разработал много собственных и даже стал вести семинары по этой теме. Содержание семинаров и обобщено в этой книге.

- Вы меня все еще не убедили.

- Позвольте привести пример. Вы ведь знаете, в чем разница между интерпретируемым и компилируемым языками.

- Конечно! Интерпретируемые языки медленнее, потому что каждая строка кода заново интерпретируется всякий раз, когда интерпретатор ее встречает. В компилируемых языках много времени уходит на предварительную обработку программы и ее перевод на машинный язык, зато потом она выполняется быстрее аналогичной интерпретируемой программы.

- Совершенно верно.

- Так вы хотите, чтобы я откомпилировал свою жизнь?

- Нет, хотя это был бы идеальный вариант. Впрочем, у компиляторов мы можем научиться главному - потратить немного времени сначала, чтобы много раз не тратить его потом. Например, на одном из предыдущих мест работы я должен был периодически заменять ленты для резервного копирования. Тогда еще не было недорогих кассетных магнитофонов, облегчающих эту работу. У нас было три главных сервера в машинном зале и восемь серверов поменьше в разных концах здания. Можно было не менять ленту, на которой еще достаточно свободного места, но мне всегда было трудно принять решение, могу ли я не менять ленту на конкретном сервере. Если я ошибался в оценке свободного места на ленте, то операция резервного копирования на нее могла сорваться. Естественно, я хотел избежать подобных сбоев! Эта ситуация держала меня в состоянии постоянного стресса. В один прекрасный день я осознал, что действую как интерпретатор, ежедневно повторяя каждый свой шаг и обдумывая каждую деталь. Мне понадобилось проанализировать процесс всего один раз и затем всегда придерживаться принятых решений.

Первое решение, которое я принял, звучало так: «Лента гораздо дешевле моего времени». Поэтому вместо экономии каждого бита я стал жертвовать местом на ленте ради выигрыша во времени.

Следующее решение: «Не нужно напрягаться по поводу несущественных данных». Данные на восьми малых серверах, установленных в дру-

гих частях здания, были гораздо менее важными, чем данные на главных серверах в машинном зале. Тем не менее я беспокоился и об этих не столь важных данных. Я должен был перестать волноваться (и испытывать стресс) по поводу несущественной информации. Сисадмины не всегда правильно расставляют приоритеты.

Я решил, что должен провести анализ один раз и пользоваться его результатами каждый день. Я должен был из интерпретатора превратиться в компилятор: один раз откомпилировать решение, а затем применять его снова и снова. Мой анализ показал, что ленты для серверов в машинном зале нужно менять почти ежедневно. Следовательно, я должен был менять их каждый день независимо от того, сколько на них остается свободного места.

А на малых серверах ленты надо было менять гораздо реже. Я должен был менять их каждый понедельник и на следующий день после неудачного резервного копирования по причине переполнения ленты.

- Вы приняли решение, что периодические сбои допустимы?

- Да. Я перестал стремиться к идеалу там, где это необязательно. Перфекционизм нередко излишен и приводит к непроизводительным затратам времени.

Создатели Интернета прекрасно это понимали. Они осознавали, что никогда ничего не добьются, если будут ждать появления безупречных коммуникаций, и разработали протоколы, работавшие в неидеальной среде.

- Но мой шеф ждет от меня идеальной работы.

- На самом деле у вашего шефа есть какая-то своя система приоритетов, и он понимает, что компромиссы неизбежны. В главе 8 мы поговорим, как следует управлять шефом.

- Пожалуйста, пообещайте, что ваши советы не будут иметь отношения к компиляторам и интерпретаторам.

- Обещаю. Не все поддается этой аналогии. Мы обсудим следующие важные темы:

- Содержание всех инструментов тайм-менеджмента в одном месте.
- Использование мозга для текущей работы, а внешних носителей - для хранения остальной информации.
- Разработка процедур для дел, которые вы совершаете регулярно.
- Предварительная компиляция решений путем выработки привычек и заклинаний.
- Поддержание в себе состояния сосредоточенности во время работы над проектом.
- Повышение качества личной жизни за счет применения описанных приемов и вне работы.

- И вы придумали для этого какое-то хитроумное сокращение?

- Нет. Сейчас вам важно знать, что я построил каждую главу так, что в ней собраны конкретные проблемы системных администраторов. Они, так сказать, наслаиваются друг на друга.

Глава 1 «Принципы тайм-менеджмента»

Что отличает нас от других? В первую очередь, количество прерываний, отвлекающих нас, и множество проектов, над которыми мы должны работать одновременно. Но есть и другие факторы. В этой главе представлены основные принципы, которые получают развитие на протяжении книги.

Глава 2 «Сосредоточенность как средство против прерываний»

В этой главе обсуждается, как корректно обращаться с клиентом, прерывающим вашу работу. Вы мало чего достигнете, если не будете управлять прерываниями.

Глава 3 «Рутинные процедуры»

В этой главе показано, как превратить хаос в рутинную процедуру. Наша работа имеет высокую степень хаотичности, а любая удачная процедура немножко уменьшает этот хаос и значительно понижает наш стресс. Когда мы разрабатываем процедуры, наши задачи становятся делом привычки, и мы с меньшей вероятностью забываем о них.

Глава 4 «Система Цикл»

Эта глава является введением в мою систему Цикл, которая представляет собой технику работы со списком дел. Вы научитесь «жонглировать» многочисленными делами, ничего не роняя. Даже если на выполнение всех ваших заданий требуется 100 часов, вы сумеете справиться с ними, не выходя за рамки 8-часового рабочего дня.

Глава 5 «Система Цикл: списки дел и расписания»

В этой главе подробно описана ежедневная практическая работа по системе Цикл, которая заключается в составлении списка дел и расписания на текущий день. В главе также приводятся тактические и стратегические советы на случай, когда вы чрезмерно завалены работой.

Глава 6 «Система Цикл: работа с календарем»

У системных администраторов всегда много деловых встреч, и они должны участвовать во многих совещаниях. При рациональном подходе вы перестанете пропускать встречи и совещания и у вас еще останется время на личную жизнь и развлечения. В этой главе я расширяю систему Цикл так, что в нее включается календарь.

Глава 7 «Система Цикл: долгосрочные цели»

Эта глава научит вас ставить долгосрочные цели - как личные, так и профессиональные - и достигать их. Как вы представляете свою жизнь через десять лет? Ваши представления с большей вероятностью

стью превратятся в реальность, если вы уже сейчас займетесь планированием.

Глава 8 «Расстановка приоритетов»

Хороший жонглер может подбрасывать одновременно много предметов, но будет вынужден остановиться, если допустит ошибку. Великий жонглер умеет растянуть жонглирование, специально уронив один или два шарика ради того, чтобы остальные продолжали летать. В этой главе я обсуждаю несколько разных способов расстановки приоритетов, позволяющих системе Цикл работать еще эффективнее.

Глава 9 «Как справиться со стрессом»

Короткая глава о том, как я научился справляться со стрессом.

Глава 10 «Управление электронной почтой»

Все мы получаем слишком много электронной почты. В этой главе я привожу несколько реалистичных советов по поводу того, как справиться с этим бурным потоком.

Глава 11 «Исключение непроизводительных затрат времени»

Одним из способов иметь больше времени является исключение его непроизводительных затрат. Здесь я говорю о том, как распознать потери времени и исключить их из своей жизни.

Глава 12 «Документация»

Эта глава посвящена безболезненному составлению документов. Документируя свою работу, мы облегчаем себе тайм-менеджмент двумя способами. Во-первых, мы тратим меньше времени на попытки вспомнить, как выполняется та или иная задача, потому что можем обратиться к документации. Во-вторых, нам легче будет поручить задачу другому сотруднику, полностью исключив ее из своего списка дел. Проблема в том, чтобы найти эффективный способ, позволяющий вам выработать привычку документировать процессы. Решение состоит в использовании технологии Wiki.

Глава 13 «Автоматизация»

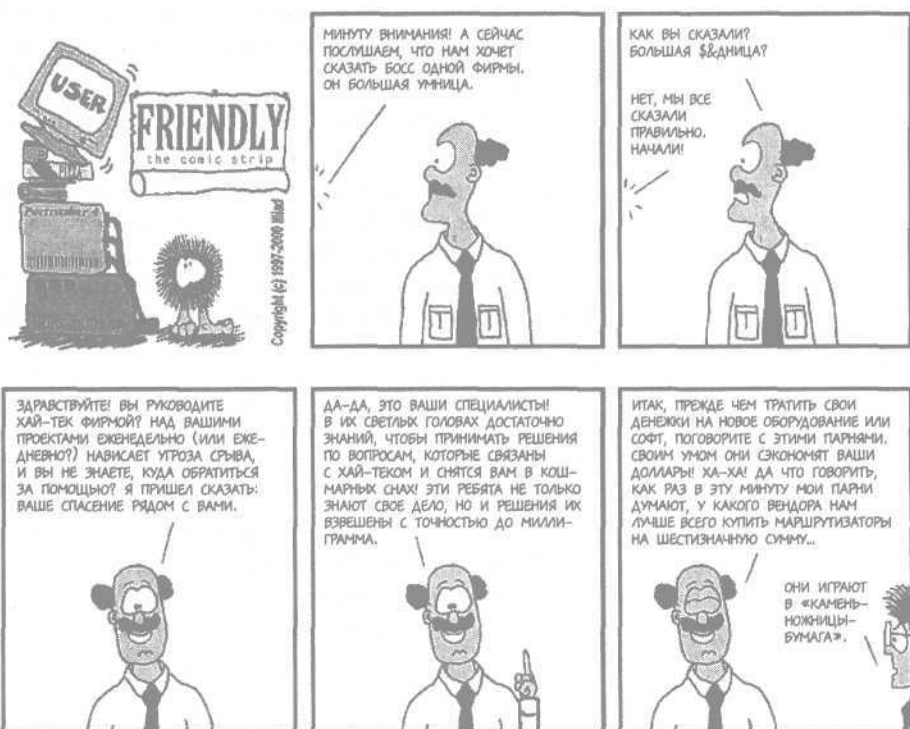
Что может быть лучше компьютера, выполняющего за вас вашу работу? Есть множество современных и простых способов, с помощью которых вы сможете приступить к автоматизации своей работы прямо сегодня, даже если вы не сильны в программировании. В этой главе приводятся методы, позволяющие приложить минимум усилий, чтобы автоматизировать максимум процессов.

Эпилог

Несколько заключительных замечаний с предложениями, как потратить свободное время, появившееся у нас после применения приемов, описанных в этой книге.

Как читать эту книгу

- И это все мне поможет?
- Безусловно! Когда вы прочтаете эту книгу, вся ваша жизнь удивительным образом мгновенно преобразится и все ваши проблемы будут решены. Кроме того, вы будете лучше выглядеть!
- Правда?
- Нет. Эта книга не решит все ваши проблемы, тем более мгновенно. Я думаю, вы встречали достаточно торговых агентов и понимаете: любой, кто обещает, что его товар моментально избавит вас от всех проблем, говорит неправду.
- Так чего мне ждать от этой книги?
- Эта книга даст вам технологию организации времени. Здесь описана система, которая подходит мне и другим. Она может подойти и вам. Эта технология позволит вам избавиться от старых плохих привычек и выработать хорошие. Грустная правда заключается в том, что вы всю жизнь выработывали дурные привычки, и потребуются определенное время, чтобы заменить их хорошими. Психологи говорят, что нужно вести себя по-новому 21 день, чтобы это поведение перешло в привычку.



- То есть через 21 день я...

- Возможно, у вас уйдет больше времени. Я не рассказывал вам, как впервые попробовал заняться тайм-менеджментом?

- Нет.

- Я посетил двухдневный семинар. В течение следующего месяца я не применил ни одного приема из числа упомянутых на семинаре. Мне казалось, что на изменение привычного стиля работы уйдет слишком много сил! А потом наступил действительно напряженный период, в течение которого я должен был сделать гораздо больше дел, чем мог успеть. И вот я «откопал» органайзер в кожаном переплете, который входил в комплект учебных пособий, выданных на семинаре, и стал лихорадочно вспоминать, чему меня учили. С помощью учебника я кое-как восстановил то, что проходил.

- И что дальше?

- В тот день я успел сделать столько, сколько раньше не успел бы за сто лет. Кроме того, я совсем не беспокоился по поводу дел, отложенных на потом.

Через несколько месяцев я снова открыл учебник, чтобы освежить в памяти методы тайм-менеджмента и воспользоваться какими-нибудь еще. Было нелегко, пришлось преодолевать себя, но в конце концов эти приемы стали моей второй натурой. И главное - я нашел методы, подходящие именно системным администраторам. Вскоре я стал обучать этим методам своих коллег, затем стал вести семинары сам, а теперь весь мой опыт здесь, в этой книге.

- Сколько времени прошло, прежде чем вы стали применять эти методы на практике?

- От месяца до года. Это зависело от конкретного метода. Я полагаю, нечто подобное произойдет и с этой книгой. Вы прочитаете ее, проигнорируете половину прочитанного и затем вернетесь к ней, чтобы поискать «новый» совет. Несколько десятков лет я вырабатывал плохие привычки. Чтобы отказаться от них и выработать новые, потребовалось немало времени.

Чем раньше вы начнете, тем скорее произойдут какие-то сдвиги. Начните сегодня.

- Легко сказать. Пользователи заходят в мою комнату каждые пять минут. У меня не будет времени на чтение.

- Да, это так. Давайте заключим сделку. Я в первой же главе расскажу, как реагировать на отвлекающие факторы, а вы пообещаете попробовать на практике каждый метод хотя бы один раз.

- По рукам!

Для кого эта книга

Эта книга адресована инженерам, работающим в области информационных технологий (ИТ), системным администраторам, администраторам сетей, операторам, персоналу, обслуживающему вычислительную технику, и представителям многих других компьютерных профессий. Она написана для тех, кто стоит на первых ступеньках своей карьеры, но будет столь же полезной и «ветеранам». Если вы считаете, что вам некогда читать эту книгу, то она вам просто необходима.

Книга не рассчитана на программистов. «Бета-читатели» говорили мне, что программистам она будет очень полезна, но я чувствую, что у программистов несколько иные проблемы, и они заслуживают, чтобы для них написали отдельную книгу. Если вы программист, купите эту книгу и подарите ее своему системному администратору. Если вы прочтете ее до того, как подарите, я никому не скажу.

О книге

Эта книга описывает определенную технологию. Методы тайм-менеджмента могут быть реализованы при помощи карандаша и бумаги либо современного портативного компьютера (PDA). Первая часть книги поможет вам освоить основы тайм-менеджмента: управление прерываниями, отвлекающими вас от работы, и ведение списка дел, позволяющего вам не забыть о поступивших запросах и выполнить их в срок или хотя бы в соответствии с вашими приоритетами. Эта книга поможет превратить хаотичную непланируемую деятельность в набор простых процедур, о которых вы не забудете. Затем я расширю область применения описываемых методов и научу вас пользоваться ими для организации календаря встреч, обработки электронной почты, борьбы со стрессом и достижения долгосрочных целей. Наконец, я коснусь приемов, которые могут ускорить ваш карьерный рост, таких как исключение непроизводительных затрат времени, ведение экономящей время документации и автоматизация повседневной деятельности, позволяющая избежать технических ошибок и опять-таки сэкономить время.

Теперь, когда вы знаете, о чем эта книга, я должен объяснить, чего в ней нет. Здесь нет инструкций по использованию портативного компьютера или какого-либо программного пакета. Вы не найдете в ней сравнительных характеристик полусотни программ, позволяющих вести списки дел. Это не руководство по PalmOS или Microsoft Windows Mobile 2003 Second Edition Operating System. Эта книга о вас и о том, как вы можете повысить качество своей жизни, правильно организовав свое время.

Предположение, сделанное в этой книге

Излагая материал, я не делаю никаких предположений о профессиональном опыте и технической квалификации читателя. Однако читате-

лю, имеющему некоторый опыт системного администрирования, она, возможно, покажется более полезной. Чем большее напряжение вы испытываете в своей профессиональной деятельности, тем более ценной окажется эта книга.

В главе 13 приведены реальные фрагменты кода, поэтому для их понимания и применения на практике требуются определенные знания. Впрочем, в этих фрагментах нет ничего непонятного для среднестатистического системного администратора.

Соглашения, используемые в книге

В книге используются следующие издательские соглашения:

Специальный шрифт

Шрифтом *OfficinaSans* выделяются элементы интерфейса (пункты меню, кнопки) и клавиатурные сокращения (например, Alt или Ctrl).

Курсив

Курсивом выделяются новые термины, URL, адреса электронной почты, имена файлов, их расширения, пути к файлам, имена каталогов и названия утилит UNIX.

Моноширинный шрифт

Этим шрифтом выделяются команды, опции, переключатели, переменные, атрибуты, ключи, функции, типы, классы, пространства имен, методы, модули, свойства, параметры, значения, объекты, события, обработчики событий, теги XML и HTML, макросы, содержимое файлов и вывод команд.

Моноширинный полужирный шрифт

Этим шрифтом выделяются команды и другой текст, которые должен ввести пользователь.

Моноширинный курсив

Этим шрифтом выделяется текст, который должен быть заменен на значения, подставляемые пользователем.



Этой пиктограммой обозначается совет, предложение или общее замечание.



Этой пиктограммой обозначается предупреждение или предостережение.

Использование фрагментов кода

Эта книга призвана помочь вам в работе. В общем случае вы можете использовать код, приведенный в этой книге, в своих программах и доку-

ментах. Вы не обязаны получать наше разрешение, за исключением тех случаев, когда воспроизводите значительные фрагменты кода. Например, для написания программы, в состав которой входят отрывки кода, приведенные в этой книге, разрешение не требуется. Продажа или распространение CD-ROM с примерами из книг O'Reilly требует разрешения. Когда вы отвечаете на чей-то вопрос, цитируя эту книгу и фрагменты кода из нее, вы не нуждаетесь в разрешении. Если вы включаете значительный объем кода из этой книги в документацию к своему продукту, вы должны, получить разрешение.

Мы не требуем обязательных ссылок на книгу при цитировании, но будем благодарны, если вы их приведете. Ссылка обычно содержит название книги, автора, издательство и ISBN. Например: «Time Management for System Administrators by Thomas A. Limoncelli. Copyright 2006 O'Reilly Media, Inc., 0-596-00783-3».

Если вы сомневаетесь в корректности использования вами примеров кода, обратитесь за разъяснениями по адресу permissions@oreilly.com.

Контакты

Пожалуйста, направляйте свои комментарии и вопросы по этой книге издателю:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
(800) 998-9938 (США и Канада)
(707) 829-0515 (международный и местный)
(707) 829-0104 (факс)

Для этой книги издательство O'Reilly поддерживает веб-страницу со списком опечаток, примерами и другой дополнительной информацией. Адрес:

<http://www.oreilly.com/catalog/timemgmt>

Электронный адрес для комментариев и технических вопросов по этой книге:

bookquestions@oreilly.com

Обратиться к автору можно на его веб-странице или по электронной почте:

Веб-страница: <http://www.everythingsysadmin.com>

E-mail: timebook@everythingsysadmin.com

Дополнительная информация о наших книгах, конференциях, центрах ресурсов и портале O'Reilly Network представлена на нашем веб-сайте по адресу:

<http://www.oreilly.com>

Safari® Enabled



Если на обложке технической книги есть пиктограмма «Safari® Enabled», это означает, что книга доступна в Сети через O'Reilly Network Safari Bookshelf.

Safari предлагает намного лучшее решение, чем электронные книги. Это виртуальная библиотека, позволяющая без труда находить тысячи лучших технических книг, вырезать и вставлять примеры кода, загружать главы и находить быстрые ответы, когда требуется наиболее верная и свежая информация. Она свободно доступна по адресу <http://safari.oreilly.com>.

Благодарности

Эта книга не смогла бы появиться на свет, если бы не Крис Полк (Chris Polk), которая поддерживала меня эмоционально и технически на протяжении всего проекта. Ее замечания учтены в каждой главе.

Спасибо Дэвиду Бланк-Эдельману (David Blank-Edelman) за его трогательное вступительное слово. Я также благодарен Иллиаду (Illiad) за серию комиксов «User Friendly». Я благодарю судьбу за участие этих людей в создании книги.

Я хотел бы поблагодарить сотрудников издательства O'Reilly за их помощь, особенно Майка Лукидеса (Mike Loukides) за то, что он принес этот проект в издательство O'Reilly и помог мне определиться с книгой, а также Дэвида Брикнера (David Brickner), который доработал книгу так, что она могла быть опубликована. Дэвид взял неплохую книгу и превратил ее в замечательную. Без него я бы не справился. Благодаря моему редактору, Марлоу Шэффер (Marlowe Shaeffer), эти страницы увидели свет. Спасибо всем, кто работает в O'Reilly!

Если я вижу дальше других, то лишь потому, что стою на плечах гигантов. Вот лишь некоторые из них: Мэри Кларк (Mary Clark), Бенджи Фин (Benji Feen), Дуг Ферлонг (Doug Furlong), Трей Харрис (Trey Harris), Дженнифер Джой (Jennifer Joy), Энди Лестер (Andy Lester), Р. А. Лихтенштайгер (R. A. Lichtensteiger), Джон Линдерман (John Linderman), Лес Ллойд (Les Lloyd), Ральф Лоура (Ralph Loura), Тина Манкузо (Tina Mancuso), Клифф Миллер (Cliff Miller), Адам Москович (Adam Moskowitz), Дэйзи Нгайн (Daisy Nguyen), Кэт Окита (Cat Okita), Джей Пи (JP), Виктор Рэймонд (Victor Raymond), Том Рейнголд (Tot Reingold), Майкл Ричичи (Michael Richichi), Страта Роуз-Чалап (Strata Rose-Chalup), Гленн Сейб (Glenn Seib), Франк Дж. Т. Войчик (Frank J. T. Wojcik). Прощу прощения, если кого-то не упомянул.

1

Принципы тайм-менеджмента

Погодите! Прежде чем начать, давайте кое-что сделаем для уверенности, что мы действительно закончим.

Я прекрасно понимаю, что вы как системный администратор постоянно подвергаетесь прерываниям. То телефон зазвонит, то клиент¹ обратится с вопросом, то почтовая программа просигналит о новом сообщении, а то интернет-пейджер попытается отвлечь ваше внимание. Держу пари, пока вы читали этот абзац, вас уже кто-то прервал.

Я отложу обсуждение прерываний до следующей главы (надеюсь, вы не обидитесь), но все же опасаясь, что вам не удастся добраться так далеко. Чтобы решить эту проблему, я приведу здесь совет из главы 2, который, если вы ему последуете, оградит вас от прерываний с данного момента вплоть до того времени, когда мы научимся правильно обращаться с прерываниями.

Предположим, в вашей фирме два системных администратора. Вы можете заключить с вашим коллегой соглашение о взаимной защите от прерываний. До обеда на все прерывания реагируете вы, чтобы ваш коллега мог поработать над проектами. После обеда все прерывания берет на себя он, позволяя работать над проектами вам. Разумеется, если возникнет проблема, решить которую можете только вы, вы займетесь ею. Однако вы обнаружите, что в результате такой организации рабочего дня вы успеваете больше сделать для проектирования. Возможно, у вас появится время почитать эту книгу.

1 В этой книге «клиентом» я называю любого внутреннего или внешнего пользователя ваших компьютеров, сетей, приложений и т. д. Я предпочитаю его термину «пользователь», поскольку так лучше отражается специфика взаимоотношений между системными администраторами и теми, кого они обслуживают.

Этот метод хорошо работает, если в фирме много системных администраторов. Мне как-то довелось быть членом весьма многочисленной команды сисадминов, и нам удалось распределить время «на обработку прерываний» так, что вся группа могла сосредоточиться на проектировании в течение всего рабочего дня за исключением одного часа.

Этот метод можно адаптировать и для случая с одним системным администратором. Если вы единственный сисадмин в своей фирме, поговорите с начальством о реализации подобной системы. Например, руководство фирмы может оповестить всех пользователей, что время после обеда отводится для работы над проектами и несрочные запросы следует отправлять вам (или вашей системе регистрации запросов) по электронной почте, с тем чтобы они рассматривались на следующий день. Это вполне может соответствовать обычному распорядку дня в офисе. Скажем, если большинство прерываний происходит в первой половине дня, имеет смысл отводить вторую половину под проектирование. Конечно, это следует делать только при наличии конкретного осязаемого проекта. Например, ваш босс поручает вам проект, от которого дела в фирме пойдут гораздо лучше. Это удобная возможность попросить его о поблках, позволяющих ускорить работу над этим проектом.

Существуют и физические средства защиты во время вашей работы над проектом (в «проектное время»). Очевидно, что если у вас отдельный кабинет, вы можете закрыть дверь, чтобы оградить себя от случайных и неслучайных посетителей. Более эффективный способ заключается в следующем. Заставьте клиентов проходить мимо системных администраторов первого уровня (сидящих лицом к клиенту), чтобы добраться до администратора второго уровня (то есть до вас). Если вы старший системный администратор, передвиньте свой рабочий стол так, чтобы идущему к вам клиенту пришлось пройти мимо младшего системного администратора. Роль младшего сисадмина состоит в том, чтобы обрабатывать 80% прерываний, передавая вам 20%, которые никто кроме вас обработать не сможет. В этом случае физическое местоположение является ключевым. Отойдите на 100 метров от своего рабочего стола, развернитесь и идите к нему, как будто вы типичный клиент. Кого вы



видите? Разумеется, сотрудника, работающего с клиентами и решающего их проблемы на первом уровне.

Организуйте взаимную защиту от прерываний прямо сейчас. Я подожду.

Эй, которое из слов «прямо сейчас» вам непонятно? Вы ведь не двинулись с места, не так ли? Пожалуйста, сделайте это и не читайте дальше. Я очень хочу, чтобы вы смогли дочитать книгу до конца.

Трудности тайм-менеджмента

Вот теперь можно начинать!

Тайм-менеджмент труден для сисадминов в первую очередь потому, что нашу работу постоянно прерывают. Как довести дело до конца, если нам все время приходится бросать его, чтобы устранить проблему или ответить на вопрос, с которым клиент обратился лично, по почте или с помощью новейшего средства прерывания работы - интернет-пейджера? Сколько раз вы говорили шефу, что на проект потребовалось бы два дня, если бы вас не отвлекали, а фактически работа над ним тянется целый месяц? Возвращение в работу требует много времени. Если прерывание займет одну минуту, а возврат к проекту еще две, то вы фактически будете двигаться во времени в обратном направлении. Вот удивился бы Герберт Уэллс! Хуже всего то, что возвращение к работе после прерывания чревато ошибками. При решении той или иной проблемы я нередко обнаруживал, что «ошибка» заключалась в том, что я пропустил очередной шаг после прерывания работы!

Руководство судит о работе сисадмина по тому, завершен ли проект. Однако клиенты оценивают нас по тому, насколько мы доступны. Эти приоритеты конфликтуют, а вы находитесь между ними. Если вы все время доступны клиентам, вы не сможете закончить проект, который вам поручило руководство. И кто тогда согласится повысить вам зарплату?

Почему книга, посвященная тайм-менеджменту, адресована только системным администраторам? Эта книга по необходимости отличается от «средней» книги по тайм-менеджменту, потому что системные администраторы не такие, как все. В частности:

- **У нас другие проблемы.** У системных администраторов необычайно много прерываний в работе, не позволяющих им завершать проекты.
- **У нас другие решения.** Системные администраторы в состоянии применять высокотехнологичные решения, такие как система регистрации запросов, фильтрация электронной почты с помощью утилиты *procmail*, сценарии автоматизации и иные средства, не доступные тому, кто далек от техники.
- **У нас нет системы наставничества.** Системным администраторам, как и другим менеджерам, необходимо изучить основы управления

списками дел, календарем и жизненными целями. Однако типичная трудовая биография системного администратора не включает в себя специальное обучение таким вещам. Наши наставники - коллеги-технари, разбросанные по всему свету и порой знакомые только по электронной почте. У нас мало возможностей учиться у вышестоящего, как, бывает, руководитель среднего звена учится у директора.

Принципы тайм-менеджмента для системных администраторов

Есть шесть принципов, на которых я основываю все свои приемы тайм-менеджмента. Не утверждаю, что какой-либо из них открыт мною, но я определенно причастен к их развитию. Вы легко проследите эти принципы на протяжении всей книги:

- Ведите единую «базу данных» по тайм-менеджменту (пользуйтесь одним органайзером).
- Храните в голове только важную информацию (экономно расходуйте оперативную память).
- Разрабатывайте процедуры и придерживайтесь их (используйте библиотеки кода; не изобретайте велосипед).
- Выработывайте привычки и заклинания (вместо вычислений во время выполнения используйте готовые ответы).
- Сохраняйте сосредоточенность в «проектное время» (работайте как семафор монопольного доступа ядра операционной системы).
- В повседневной жизни пользуйтесь теми же приемами, что и на работе (жизнь вне работы не является чем-то второстепенным).

Рассмотрим каждый из этих принципов более подробно.

Единая «база данных» по тайм-менеджменту

Главным инструментом тайм-менеджмента является портативный компьютер или органайзер, который позволяет вам хранить список дел, календарь и списки жизненных целей. Вы, конечно, знаете, что такое портативный компьютер; это устройство вроде Palm Pilot или Zaurus. Органайзер - его бумажный эквивалент. Вам он встречался во всем многообразии форм и марок канцелярской продукции.

Независимо от того, какой инструмент вы выберете, он станет платформой для почти любого метода, описанного в этой книге. Держа всю информацию в одном месте, вы избавите себя от необходимости переключаться с одной системы хранения на другую. При недостатке организованности это поможет вам внести порядок в свои действия. Если, напротив, вы уделяете организационному процессу слишком много внимания, этот инструмент позволит вам остановиться на простой базовой системе, не занимаясь организацией своей организованности.

В дальнейшем я буду использовать термин *органайзер* для обозначения как портативного компьютера, так и бумажного органайзера. Неважно, пользуетесь ли вы достижениями высоких технологий. Если понадобится, в конкретном примере я уточню, какой инструмент имеется в виду.

Мы будем исходить из предположения, что вы можете доверять своему органайзеру. Информация, которую вы в него запишете, не будет забыта, удалена или потеряна, а также не исчезнет, подобно невидимым чернилам. Только доверяя своему органайзеру, вы сможете свободно применять другие приемы тайм-менеджмента. Воспитание в себе такого доверия, как и выработка любой привычки, приходит со временем.

Храните в голове только важную информацию

Очень важно, чтобы ваш мозг не был «засорен». Засоренный мозг постоянно отвлекается и не позволяет вам сохранять сосредоточенность. Вы должны научиться хранить во внешней памяти все, на чем не сконцентрированы в данный момент.

Не принимайте это как личный выпад, но ваш мозг не такое надежное средство хранения информации, как бумажный листок или компьютер. Не используйте мозг для запоминания заданий и встреч. Пользуйтесь органайзером, базой данных, Wiki - чем угодно, но только не мозгом.

Емкость вашего мозга ограничена. Он работает либо как ОЗУ, либо как ЦПУ, в зависимости от ваших представлений о нем. Если вы засорите свой мозг сведениями обо всех задачах, которые вам предстоит решить в будущем, то не хватит места для задачи, над которой вы работаете сейчас.

В самом деле, если я работаю над проектом А, но постоянно думаю о проекте В, то лучшее, что я могу сделать, - это занести проект В в список моих дел и постараться выкинуть его из головы. Тогда я смогу сосредоточиться на проекте А. Я поручаю списку дел «помнить» о проекте В вместо меня, и мне не приходится непрерывно терять на этом умственную энергию.

По недостоверным источникам, у Альберта Эйнштейна в шкафу висело семь одинаковых костюмов на каждый день недели. Согласно этой легенде он так сохранял умственную энергию для физики, не тратя ее на размышления о том, что сегодня надеть. Возможно, по аналогичной причине руководитель компании Apple Computer Стив Джобс (Steve Jobs) всегда носит черные свитера. (У меня самого много одинаковых пар носков, но результат только в том, что после стирки не остается больше одного непарного носка.)

Эта книга позволит вам исключить из своего лексикона отговорку «Я забыл». Возможно, вы будете нарушать сроки по другим причинам, но уж точно не потому, что вы пытались запомнить так много всего, что о чем-то позабыли.

Перенос заданий, инструкций и прочей информации из головы на бумагу или в память компьютера - первый шаг к получению помощи в решении ваших задач. В то время как наш мозг является однопользовательской системой и никто, я надеюсь, не читает наши мысли, внешние носители информации являются многопользовательскими и позволяют другим оказывать нам помощь в нашей работе. Например, документирование процедуры и размещение ее на веб-сайте позволит другим изучить эту процедуру и, возможно, следовать ей. Использование базы данных с запросами клиентов позволяет другим системным администраторам реагировать на запросы, клиентам - видеть, что их проблемы решаются, а руководству - собирать статистику. Зачем запоминать список служб, которые надо протестировать после установки новой версии операционной системы? Храните этот список в виде электронной таблицы, а не в голове. Вы сможете показать эту таблицу коллегам и проверить, не упустили ли чего.

Разрабатывайте процедуры и придерживайтесь их

Хороший разработчик программного продукта старается кодировать как можно реже. Если фрагмент кода работает, программист использует его всюду, где только возможно. Я пишу очень много кода на языке Perl. Окружающие считают меня экспертом по этому языку. На самом деле я знаю около десятка действительно хороших идиом языка и использую их снова и снова. Программы редко создаются с нуля. Вам следует организовать свое время аналогичным образом. Превратите повторяющиеся действия в процедуры, чтобы тратить меньше времени на планирование событий, которые все равно произойдут.

Раньше я тратил несколько часов в неделю, пытаюсь «поймать» шефа по телефону, чтобы договориться с ним о встрече. Теперь мы еженедельно в одно и то же время проводим короткое совещание. Больше нет неопределенности. Это наша процедура. Установив эту процедуру, мы оба сокращаем время на раздумья о том, что надо сделать на этой неделе.

Чтобы не тратить много времени на организацию своих приемов тайм-менеджмента, я выработал процедуру и для этого вида деятельности. В начале каждого рабочего дня перед проверкой электронной почты я просматриваю список дел, чтобы установить приоритеты на сегодня.

Важно, чтобы вы придерживались установленных процедур, поскольку ваши коллеги привыкают рассчитывать на них. Это помогает им планировать свое время. Все в выигрыше.

Вырабатывайте привычки и заклинания

Привычки - это процедуры, которые вы выполняете, не задумываясь. Заклинания — это мысленные «включатели» тех или иных правил.

Я заправляю машину каждое воскресенье. Это хорошая привычка. Я даже не могу вспомнить, откуда она взялась, но полагаюсь на нее. Вспом-

ни! Я часто опаздывал на работу по понедельникам, потому что был вынужден заезжать на заправочную станцию. Однажды я решил, что хватит опаздывать. Теперь это привычка. Если мой бензобак не полон доверху, я всегда заправляюсь в воскресенье.

Правила аналогичны привычкам, но не обладают регулярностью. Они являются способом запоминания вашей оптимальной реакции на конкретные обстоятельства. Активизируя правило, я произношу заклинание. Для мелких дел, которые я склонен откладывать на потом, у меня есть такое заклинание:

Лучше раньше, чем позже.

Например, как-то я был вынужден периодически выливать воду из резервуара переносного кондиционера, установленного в маленькой комнате с компьютерами. (Конечно, по большому счету, следовало провести слив или установить охладитель, испаряющий воду. Это было временное решение.) На резервуаре было две отметки уровня: «Пора вылить воду» и «Идиот! Ты уже должен был вылить воду». Вода могла не доходить до первой отметки целую неделю, и несколько дней подряд мне вроде бы не требовалось выливать ее. Это приводило к трем проблемам. Во-первых, вылить воду из полного резервуара гораздо труднее, чем из полупустого, - она просто расплескивается. Во-вторых, как назло, чаще всего я замечал, что уровень воды приблизился к критичной отметке, в конце того рабочего дня, когда меня ждали другие важные дела. И мне приходилось опаздывать, потому что слив воды занимал минут пятнадцать. В-третьих, и это самое важное: все время был риск переполнения резервуара. Конечно, у кондиционера был детектор для автоматического отключения при переполнении, но температура в помещении могла так повыситься, что перегорели бы все компьютеры.

Как вы можете догадаться, ситуация заметно улучшилась, когда я стал применять заклинание «Лучше раньше, чем позже». Замечая, что вода достигла первой отметки, я сразу говорил себе «Лучше раньше, чем позже» и выливал воду. Проблема была решена. Это заклинание близко к английской пословице «Вовремя сделанный стежок стоит девяти».

Со временем привычки и заклинания становятся вашей второй натурой. Выработка привычек и заклинаний окупается тем, что ваш мозг, в конечном счете, не засоряется ненужной информацией. Став второй натурой, они не вмешиваются в ход наших обычных мыслей, следовательно, не противоречат принципу сбережения оперативной памяти.

Вот еще одно заклинание, которое меня всегда выручало:

Доверяй процессу.

Через несколько глав я порекомендую вам каждое утро отводить пять минут на планирование дня. Бывают суматошные дни, когда возникает соблазн не делать этого, но по злой иронии именно в эти дни планирование особенно необходимо. Я говорю себе «Доверяй процессу» и занимаюсь планированием. Потом я всегда радуюсь, что сделал это.

Если ваша голова полна категоричных или вкрадчивых мыслей вроде «Я сделаю это позже» или «Я слишком занят, чтобы тратить время на планирование дня», заклинание способно заполнить ее позитивными мыслями, вытесняя отрицательные. Если вы приступите к действиям, пока ваш мозг ненадолго пропитался заклинанием, то сделаете что-то позитивное раньше, чем отрицательные мысли смогут вернуться.

Это немаловажный момент. Вы можете научиться тому, как перехитрить свой мозг!

Сохраняйте сосредоточенность в «проектное время»

Ранее в этой главе я говорил о важности управления прерываниями. Это одна из основ сохранения сосредоточенности. Прерывания - истинный враг сосредоточенности. Вам следует научиться пользоваться органайзером и другими инструментами и приемами поддержания сосредоточенности.

Отвлекающие факторы отвлекают (прошу прощения за тавтологию) от работы! Посмотрите, как работает операционная система. Для выполнения неотложных операций ядро блокирует все остальные задачи и монополично выполняет одно действие, пока оно не будет завершено. Например, когда какому-то приложению выделяется память, ядро блокирует все другие запросы к памяти, чтобы данная операция прошла корректно и другие процессы не могли обращаться к таблицам распределения памяти в это же время. Как системный администратор вы должны точно так же сосредотачиваться на одной задаче.

Разработчики операционных систем всячески стараются сделать так, чтобы после прерывания процесс возобновлялся быстро (особенно при постоянном переключении между процессами в многозадачной операционной системе). Они добиваются этого, потому что знают, что время возврата после прерывания - потерянное время, и его следует минимизировать. Вы должны поступать так же.

В повседневной жизни пользуйтесь теми же приемами, что и на работе

И последнее (по порядку, но не по сути). На работе свет клином не сошелся. Те же приемы, которые позволяют нам рационально расходовать время на важные проекты, помогут нам жить полноценной общественной и личной жизнью.

Никто не молится о том, чтобы проводить в офисе как можно больше времени.

Применяя одни и те же методы в работе и в повседневной жизни, вы оттачиваете свою технику! Чем больше практики, тем быстрее у вас выработаются полезные организационные навыки. Вы будете применять хорошие апробированные приемы, и вам не придется изобретать велосипед.

Сказанное не означает, что ваша личная жизнь должна быть структурирована и расписана по минутам. Нет ничего плохого, если вы запланируете вечером подурочиться!

Будет нелегко

Мне говорили, что, начиная кого-то учить, лучше сразу предупредить учеников о предстоящих трудностях во избежание разочарования, когда они увидят, что не везде их ждут молочные реки с кисельными берегами (или мармелад в шоколаде). Мне говорили, что это гораздо лучше, чем обещать им «результат легко и сразу», чтобы при первых серьезных трудностях они не пали духом, думая, что не добились обещанного скорого результата по собственной вине.

Итак, позвольте мне быть предельно откровенным: это, скорее всего, будет самое трудное дело, за которое вы когда-либо брались. Всю жизнь вы привыкали неправильно распоряжаться временем и не сможете сразу преодолеть инерцию. Потребуется многочасовая практика. Вы будете спотыкаться и падать, а когда через месяц перечитаете какую-нибудь главу, то обнаружите, что опять все делали неправильно. Временами вам будет казаться, что нет никакого просвета, что все эти методы - пустая трата времени и что применять их гораздо труднее, чем так же бестолково вести дела, как прежде.

Поверьте, все это ждет вас, потому что я и сам это испытал.

Однако сейчас я пишу эту книгу. Значит, я выжил. Уцелеете и вы.

Каждый раз, погружаясь во мрак и безнадежность, вспоминайте, что изменения происходят постепенно. Продолжайте попытки. Придерживайтесь программы. Вытесняйте из головы негативные мысли, сказав себе «Доверяй процессу», и пробуйте снова.

В самый неожиданный момент кто-то скажет вам: «Вы такой организованный человек! Хотел бы я знать, как это у Вас получается!», и вы поймете, что эта книга вам больше не нужна. Желаю успеха!

Резюме

- Тайм-менеджмент представляет особую трудность для системных администраторов, поскольку наши проблемы уникальны (смесь проектов и прерываний), наши технические наставники не особо сильны в тайм-менеджменте, а наши далекие от техники начальники ничего не смыслят в нашей работе. Однако у нас в активе то, что мы сами технари и для организации своего времени можем применять технологичные решения.
- Внешние прерывания (клиенты) и прерывания, которые мы сами себе навязали (интернет-пейджер, уведомления о новых электронных письмах и т. д.), губительны для производительности. Возврат от прерывания к работе требует времени и чреват ошибками.

- Совместно с коллегами организуйте взаимную защиту от прерываний, чтобы кто-то из них обрабатывал прерывания в то время, когда вам необходимо работать над проектом.
- Вся информацию по тайм-менеджменту храните в единой базе данных. Когда все находится в одном месте, организованность дается легче.
- Берегите свой мозг для важной информации. Используйте его непосредственно для работы, а списки дел, встреч и просто заметки записывайте в органайзер.
- Разрабатывайте процедуры и придерживайтесь их. Вместо того чтобы периодически изобретать велосипед или повторять процесс принятия решений, организуйте рабочие процедуры.
- Выработывайте привычки и заклинания. Они позволят вам повторно применять удачные решения.
- Сохраняйте сосредоточенность во время работы над проектом. Вы будете работать лучше, если сосредоточитесь на чем-то одном.
- В повседневной жизни пользуйтесь теми же приемами, что и на работе. Ваша жизнь вне офиса безусловно важна, и ее следует организовывать теми же методами, что и профессиональную деятельность, а то вы забудете про отдых и развлечения.

Покерные фишки

Каждый готов дать совет, как избежать откладывания дел в долгий ящик. В Google вы найдете десятки тысяч ссылок. Вы найдете таблицы, составленные в университетских консультационных центрах, помогающие вам разобраться с факторами, мешающими вашей работе. Вы найдете книги и статьи. Вы найдете списки с десятками причин, заставляющих человека изо дня в день откладывать какое-то дело, и узнаете, как с этим бороться. Вы найдете профессиональных «учителей жизни», которые (лично или по телефону) научат вас преодолевать эту и другие жизненные проблемы (за плату). Выбирайте, что вам больше подходит. Сделайте это прямо сейчас.

Я думаю, вы должны прежде всего запомнить, что откладывать дело «на потом» естественно. Такое поведение происходит из боязни и неуверенности в себе. Мы все боимся перемен. Мы все сомневаемся в своей способности достичь результата.

Вместо того чтобы концентрироваться на неуверенности в себе, перенесите внимание на противоположный объект - ваше *самоуважение*. Самоуважение подобно стопке покерных фишек. Если у вас мало фишек, вы можете делать лишь маленькие ставки.

Это означает, что вы не сможете выиграть кучу денег. Вам придется часто пасовать, чтобы не потерять последнюю фишку. Когда у вас много фишек, вы можете делать большие ставки, способные принести крупный выигрыш. Вы можете рисковать. Вы можете делать то, на что никогда бы не решились с небольшим количеством фишек. Вы можете выиграть!

При низкой самооценке мы с большей вероятностью бросим начатое дело или даже не приступим к нему. Не пытаюсь, ничего не добьешься. Вот мы и не добиваемся. При высокой самооценке мы легче идем на риск и ставим себя в положение, в котором можем выиграть.

Теперь понятно, почему я сравнил самоуважение с большой стопкой фишек?

Магия в том, что фишки самоуважения существуют только в вашем сознании, и вы можете создать их сколько угодно!

В покере фишки являются реальными физическими объектами. Вы не можете щелкнуть пальцами так, чтобы появились новые фишки. Зато вы можете выполнить некий ритуал, чтобы волшебным образом увеличилось количество фишек вашего самоуважения. Мое заклинание «Лучше раньше, чем позже» дает мне фишки, помогающие не откладывать дела на потом. Проявления любви любимым вами человеком дает вам новые фишки. Молчаливая поддержка коллег, позволяющая вам читать эту книгу, порождает новые фишки. Помогает все, от чего возрастает количество ваших фишек. Если громкие выкрики «Я могу!» добавляют вам фишек самоуважения, кричите, сколько хотите.

Очень просто, да?

Специалисты сходятся во мнении: покупка этой книги, так сказать, «автоматически» увеличивает количество имеющихся у вас фишек. Переверните страницу и приступайте к работе.

2

Сосредоточенность как средство против прерываний

Сколько раз вы говорили шефу, что можете сделать работу за день, если вас не будут прерывать, а в реальности тратили на нее месяц? Системные администраторы говорят это, потому что их работу над проектом постоянно прерывают как клиенты, так и начальство.

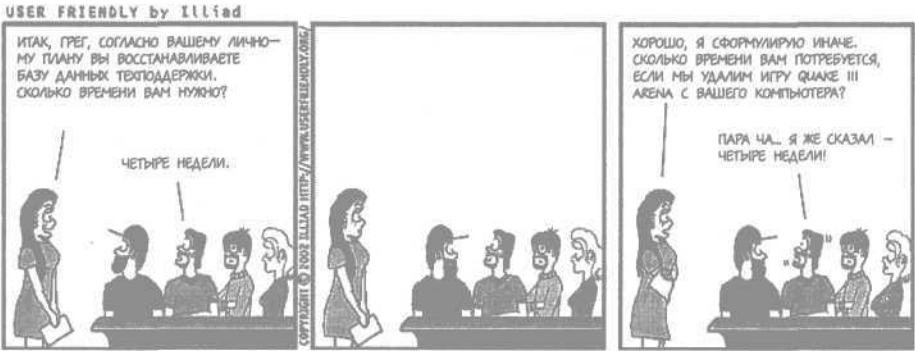
Когда системный администратор говорит, что его постоянно дергают пользователи, это означает, что в действительности он хочет сосредоточиться на своих проектах.

Когда мы сосредоточены и можем работать не отрываясь, мы способны выполнить любую работу. Сосредоточенность требует усилий. Сосредоточившись, мы справляемся с работой быстрее, а высвободившееся время можем потратить на другие дела или на личную жизнь. Это все равно что отключить от ноутбука лишнюю периферию. Аккумулятор дольше протянет, и вы больше поработаете или поиграете в любимую игру.

Прерывания - истонные враги сосредоточенности. Они прямо и косвенно воруют наше время. Прямые потери времени очевидны: прерывание, отвлекающее нас на t минут, оттягивает завершение работы на t минут. Здесь все ясно. Однако прерывания коварно воруют у нас время косвенным образом. Возвращаясь к работе, вы тратите p минут на то, чтобы сообразить, на каком этапе вы оставили работу. Если вас прервали на третьем шаге многошагового процесса, с какого шага вы должны продолжить - с третьего или четвертого? Выяснение этого является дополнительным занятием, крадущим время у проекта. Признаюсь, что за мою бытность сисадмином самые серьезные технические ошибки возникали, когда из-за прерывания я пропускал какой-то шаг или забывал довести предыдущий шаг до конца. Я, увы, возвращался к четвертому шагу вместо третьего. Если время, необходимое

для устранения этих ошибок, обозначить через s , то суммарная задержка в результате прерывания составит $t+p+s$ минут и, вполне возможно, превысит время выполнения задания!

К сожалению, системным администраторам никуда не деться от прерываний. Мы должны решать проблемы наших клиентов - это часть нашей работы. Но попытка сбалансировать решение этих проблем с нашими проектами может привести к возникновению барьеров и напряженности в отношениях с коллегами. Можно сказать, что эта глава учит вас, как сохранять сосредоточенность на проекте и спокойно обрабатывать прерывания.



Сосредоточенный мозг

Сосредоточиться - значит максимально нацелить свой мозг на решение конкретной задачи. Мозг разделен на несколько частей. Передняя часть обрабатывает ваши сиюминутные мысли (если хотите, это ЦПУ и кэш уровня L1/L2), задняя часть содержит то, что вы помните (ОЗУ), а за ней находится часть, хранящая долговременную информацию (ваш жесткий диск). Сосредоточенность относится к той части, которую я ненаучно называю передней частью мозга.

Когда вы сосредоточены, вы пытаетесь использовать 100% передней части вашего мозга. Чтобы лучше разобраться в этом, рассмотрим мозг, который ни на чем не сосредоточен. Представьте, что вы пытаетесь сосредоточиться на какой-то задаче, например на написании программы на языке Perl, автоматизирующей некую процедуру. Но в вашем мозгу роятся и мысли о совещании, которое начнется через час, о трех других задачах, которые надо решить сегодня, о молоке, которое надо купить по дороге домой, и о том, что ваш шеф сказал вам утром. Все эти мысли занимают место в передней части вашего мозга, отбирая его ресурсы у программы, которую вы пишете! Как вы думаете, легко ли ей ужиться с тем, чем забита ваша голова?

Вы, вероятно, не думаете, что, стараясь не забыть о покупке молока после работы, отнимаете ресурсы у задачи, над которой работаете, но дело

обстоит именно так. Какие-то участки вашего мозга заняты поддержкой этой информации. Аналогичным образом работает динамическая оперативная память. Информация в ней непрерывно регенерируется, иначе она будет утрачена. (Интересно, что при использовании статической оперативной памяти регенерация информации не происходит, но стоит такая реализация намного дороже.) Поддержание памяти передней части мозга в оперативном состоянии - такая же работа мозга, как и любая другая его деятельность.

Очистите ваш мозг от всех этих «не забыть бы», делегировав ответственность за их запоминание другой системе. Поставьте будильник, чтобы он зазвонил незадолго до совещания, занесите три задачи в список дел (см. главу 5), а «молоко» - в список покупок и пометьте в ежедневнике, что завтра утром надо спросить у шефа, что же он имел в виду (см. главу 8). Теперь вы избавились от посторонних мыслей и освободили голову для задачи, над которой работаете. Не бойтесь забыть о том, что надо было помнить; доверяйте системам, которым вы делегировали эту функцию.

Конечно, вы умный человек. Вы *можете* помнить обо всех этих вещах и работать одновременно, но, спрашивается, ради чего? По сравнению со многими своими коллегами я туп как дуб, но с помощью этих методов я расчищаю игровое поле. Если вы умный человек, то сможете работать с эффективностью гения. А если вы гений, то зачем вам эта книга? Дайте и нам какой-то шанс!

Вам трудно заснуть?

Заснуть - все равно что дать мозгу расслабиться. Как вы можете расслабиться, если вам надо помнить о завтрашних делах? У вас не получится и то и другое одновременно.

Положите рядом с кроватью ручку и блокнот. Если что-то не дает вам заснуть, сделайте запись в блокноте и снова попытайтесь заснуть. Готов поспорить, вы скоро заснете.

Этот прием работает и тогда, когда мы чем-то обеспокоены или рассержены. Беспокойство не дает нам заснуть, потому что мы стараемся не забыть сделать что-то, относящееся к предмету беспокойства. Гнев не дает нам заснуть, потому что мы стараемся не забыть о том, что рассердились. Если вы запишете свои беспокойные или гневные мысли, ваш мозг сможет расслабиться, потому что вы знаете: утром блокнот будет рядом.

Еще лучше, если вы позвоните себе на работу и оставите сообщение на автоответчике. Этот прием можно применить везде, где есть телефон. И он исключает для вас шанс забыть блокнот дома!

У многих сотовых телефонов и MP3-плееров есть встроенный диктофон. Заведите привычку пользоваться диктофоном, чтобы не забыть удачные идеи.

Обстановка, способствующая концентрации

Недостаточная сосредоточенность - результат не только внешних прерываний. Мы и сами виноваты - мы включаем музыку, на экране наших компьютеров пляшут анимированные изображения, прокручиваются сообщения из чата, интернет-пейджеры пытаются привлечь наше внимание. Все это отвлекает глаз и, в конечном счете, мозг. Отвлекающие факторы порождает и беспорядок на рабочем столе (как физическом, так и компьютерном).

Потратьте несколько минут, наведя порядок на рабочем месте. Лично для меня это очень трудная задача, так что я прибегаю к заклинанию:

Сомневаешься - выбрасывай.

Затем я применяю план из трех пунктов:

1. Подшить в скоросшиватель все, что удастся.
2. Незавершенные документы сложить в стопку, чтобы вскорости завершить работу над ними.
3. Все остальные бумаги положить в большой конверт, заклеить его и надписать: «Если я не открою конверт в течение трех месяцев с этого дня, его можно выбросить».

Через три месяца потребуется невероятное волевое усилие, чтобы выбросить конверт, не просмотрев его содержимое. Секрет в том, чтобы не тратить много времени на размышления об отдельных бумажках, беспокоясь о том, что они потом понадобятся. Решая выбросить конверт, я повторяю такое заклинание:

Сомневаешься - выбрасывай. Если потребуется, всегда можно достать копию.

Я обнаружил, что полезно убрать из поля зрения все постеры, календари и т. д. У меня много постеров, но они просто не висят перед глазами. Когда я сижу за компьютером, передо мной только голые стены, ничего отвлекающего.

Наконец, устранив неразбериху на рабочем месте, вы можете проделать то же самое на компьютере. Уберите пиктограммы с рабочего стола, закройте все интернет-пейджеры, плейеры, бегущие строки журналов сводок и новостей, а также почтовую программу. Я фанат электронной почты. Узнав, что пришло письмо, я сразу прочту его. Я могу целый день ждать очередное сообщение. Однако гораздо лучше открывать почтовую программу раз в два-три часа, читать письма и закрывать ее. Я не боюсь пропустить какое-то важное письмо. Если в нем будет сообщение настолько важное, как, скажем, конец света, то, я уверен, кто-нибудь зайдет в мой кабинет и скажет мне об этом (или, может, у меня будет видение насчет того, как себя вести).

Мою производительность повысили два обстоятельства: я теперь гораздо реже играю в компьютерные игры и игнорирую интернет-пейджер, когда работаю.

— Виктор Рэймонд (Victor Raymond)

<http://www.livejournal.com/users/badger2305>

Прямо сейчас потратьте несколько минут, чтобы сделать все, что описано в этом разделе.

Нет, в самом деле отложите книгу и сделайте это. Обещаю, что книга подождет, когда вы закончите. Я знаю, что вы любите вещи, которые вас отвлекают, и вам грустно от мысли, что их придется убрать. Они вас тоже любят. Вот почему они то и дело выскакивают и говорят: «Посмотри на меня! Посмотри на меня!» Избавьтесь от них.



Дон Аслетт (Don Aslett) написал несколько книг, посвященных устранению беспорядка дома и на рабочем месте. Больше других мне понравилась «Clutter's Last Stand: It's Time to De-Junk Your Life!» (Последний оплот беспорядка: пора убрать мусор из своей жизни!), Adams Media Corporation. Ее советы весьма практичны, а стиль изложения довольно живой.

Я встречал людей, которые говорили, что продуктивнее работают при отвлекающих факторах, например включенном телевизоре или радио. Когда мы были молодыми и не особо заботились о дисциплине, отвлекающие факторы не казались нам большой проблемой. На нас было меньше ответственности, не поджимали сроки и не давила необходимость завершить работу. С возрастом наши потребности меняются, как и представления о комфортной рабочей обстановке. Попробуйте избавиться от беспорядка на рабочем месте на одну неделю, и вы увидите, как это помогает. Вы страшнете с себя привычки, выработанные еще в те времена, когда вы были, в сущности, другим человеком.

Многозадачность

В профессии системного администратора многозадачность является нормой. Мы загружаем новые ISO-образы понравившихся нам дистрибутивов Linux и одновременно восстанавливаем файл из резервной копии; мы читаем электронную почту и одновременно реагируем на интернет-пейджер; при этом у нас открыто 15 окон, и в каждом что-то происходит. Настоящий рок-н-ролл!

На самом деле это хорошо. Если загрузка ISO-образа диска1 занимает целый час, то самое разумное использование этого времени - запустить ее выполнение и заняться чем-то другим.

После успешного начала загрузки она не требует нашего внимания. Мы проверим ее результат позже.

Проблема в том, что иногда мы взваливаем на себя слишком много. Мы запутываемся. Мы делаем ошибки, и нам приходится возвращать-

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

ся, чтобы их исправить. Я видел сисадминов, у которых открыто столько окон, что поиск нужного окна занимает больше времени, чем работа в нем.

Вот несколько приемов, которые мне помогают:

- **Четко разделять задачи, которые можно выполнять в многозадачном режиме, и те, которые нельзя.** Для параллельного выполнения подходят задачи типа «запустить и ждать», такие как загрузка большого файла, компиляция большой программы или восстановление файла из резервной копии. Все прочее не для многозадачного режима. Лучше хорошо выполнить одну задачу, чем одновременно выполнить несколько задач кое-как. Уделите все свое внимание задаче с наивысшим приоритетом. Чтобы не забыть о других задачах, занесите их в список дел (см. главу 5).
- **Следить за своим уровнем стресса.** Если вы устали или находитесь в состоянии стресса, не увлекайтесь многозадачностью. Бывают дни, когда я чувствую себя усталым, встал не с той ноги или чем-то удручен. Неожиданно я обнаруживаю, что одновременно делаю несколько дел и ничего не успеваю. Я закрываю все окна, интернет-пейджеры, плееры и т. д. Я делаю глубокий вдох. Затем я приступаю к задаче с наивысшим приоритетом и занимаюсь только ею. Самочувствие заметно улучшается.
- **Организовать окна с помощью виртуального диспетчера.** Виртуальный диспетчер окон позволяет вам сгруппировать окна в экраны вместо 50 одновременно открытых окон на одном экране. Например, я могу организовать шесть виртуальных экранов. Одним я пользуюсь для работы с электронной почтой, другим - для мониторинга системы, в третьем я решаю проблемы клиентов и т. д. Все необходимые окна не громоздятся на экране, а аккуратно организованы, и я не вижу те окна, которые мне в данный момент не нужны.
- **Организовывать окна на экране всегда одним и тем же способом.** Если окна на вашем экране (на виртуальных экранах) всегда расположены одинаково, вы потратите меньше времени на поиск нужного окна и с меньшей вероятностью введете команду не в том окне. Например:
 - При сравнении двух версий одного документа я всегда помещаю окно со старой версией слева, а с новой - справа.
 - Я работал некоторое время в Лондоне на компьютерах, имевших зеркальные пары в США. Я всегда размещал лондонское окно справа от американского (просто потому, что Лондон находится к востоку от Соединенных Штатов).
 - Когда я пишу код на языке Perl, я всегда организую три окна: текстовый редактор (широкое окно слева вверху), окно для тестирования и запуска программ и ввода данных (узкое окно справа вверху) и окно для вывода (широкое окно внизу экрана).

- Просматривая файлы журналов с нескольких машин, чтобы оценить из взаимодействия в комплексе, я всегда располагаю окна в направлении потока данных (сверху вниз).
- **Создавать окна для рациональной организации работы.** Командные окна (окна оболочки) создаются бесплатно, так что не жадничайте. Я выхожу из себя, если вижу начинающего системного администратора, работающего с недостаточным количеством окон. Типичный пример - решение проблем с электронной почтой в системе UNIX или Linux. Я видел начинающих администраторов, которые отправляют тестовое сообщение, а затем пытаются как можно быстрее ввести команду вывода журнала почты, чтобы успеть увидеть строчки, относящиеся к тестовому сообщению. После этого они проверяют, было ли сообщение доставлено, в результате чего журнал прокручивается за пределы окна и теряется важная информация. Потом они корректируют конфигурацию, сохраняют файл, закрывают редактор и отправляют новое тестовое сообщение. Прекратите это безумие! Не бросайтесь очертя голову в решение проблемы, сперва подготовьте рабочее место. Создайте четыре окна:
 - Широкое окно, в котором выполняется команда `tail -f /var/log/mail.log`. В нем в реальном времени будет выводиться журнал по мере того, как в нем появляются новые строчки. Пусть оно работает, пока вы выполняете другие шаги.
 - Маленькое окно с командой `echo test | mail -s test testuser@example.com`. Воспользовавшись историей команд, вы получите возможность многократной отправки тестового сообщения.
 - Следующее окно - для редактирования файла конфигурации почтовой системы. Вы сможете его периодически сохранять, но не закрывайте редактор. Оставьте его открытым.
 - Последнее окно для проверки того, получено ли сообщение.

Теперь вы одновременно видите всю необходимую информацию, и работать гораздо легче. Вы следите за различными участками своей работы, перемещая взгляд, а не вводя команды. Так намного удобнее.

Время пика сосредоточенности

Некоторые люди находят, что им легче достичь сосредоточенности в определенное время суток. При создании обстановки, способствующей сосредоточенности, важным этапом является выяснение времени суток, когда следует сосредотачиваться, то есть того часа, когда поддержка сосредоточенности требует от вас меньше всего усилий. Когда я планирую деятельность на время пика сосредоточенности, получается так, словно я планирую включение «большого мозга». Представьте себя в разное время суток. Когда у вас голова работает лучше? С раннего утра? В начале рабочего дня? После обеда? Под конец рабочего дня? Вечером? Ночью? Люди технического склада редко относят себя к «жаво-

ронкам», но это качество имеет слабое отношение к способности человека концентрировать свое внимание после того, как он проснулся.

Время пика физической активности может не совпадать со временем максимальной умственной активности. Если у вас тот же психологический тип, что у меня, то вы чувствуете себя сонным после обеда и «клюете носом» перед монитором, будучи не в состоянии сосредоточиться на работе. Чтобы этот период не пропадал впустую, посвятите его физической работе, такой как установка нового устройства в системный блок или прокладка кабеля.

Предположим, вы нашли для себя период пика сосредоточенности. Как использовать его наиболее рационально? Реорганизуйте свой рабочий день так, чтобы работать над проектами в это пиковое время. Если в эти часы у вас происходят регулярные совещания, сдвиньте совещания на другое время. Не тратьте пиковое время на электронную почту и телефонные звонки. Безусловно, это важные дела, но они не требуют включения вашего «большого мозга». (О планировании рабочего дня мы поговорим в главе 5.)

Правило первого часа

Правило первого часа гласит, что первый час рабочего дня - обычно самый тихий час в офисе. Я не «жаворонок», но заставив себя прийти на работу с утра, за первый час я успеваю сделать гораздо больше, чем за весь остальной день, потому что меня не отвлекают.

Как вы проводите первый рабочий час? Держу пари, вы читаете электронную почту и слушаете сообщения на автоответчике. Вместо того чтобы тратить весь первый час на эти дела, попробуйте ограничиться письмами на самые важные темы (или, скажем, письмами от шефа) и закройте почтовую программу. А теперь займитесь текущим проектом. Прерываний у вас будет немного, а почта никуда не денется и через час. Кроме того, если вы пришли на работу действительно рано, то читать ваши ответы некому. Так зачем спешить?

Если у вас установлена система мониторинга сети (а по идее так оно и должно быть), можно просмотреть ее основные индикаторы, убедиться, что все в порядке, не интересуясь более подробной информацией о состоянии системы. Я, например, пользуюсь программой с открытым кодом Nagios (<http://www.nagios.org>) для мониторинга служб, за которые несу ответственность, то есть почтовых серверов, маршрутизаторов, веб-серверов и т. д. Придя утром на работу, я смотрю на страницу суммарной информации, убеждаюсь, что все индикаторы светятся зеленым и я могу потратить первый час на проекты, не беспокоясь, что какой-то компонент «вылетел», а я не знаю об этом. Поначалу я использовал Nagios в достаточно скромной конфигурации, следя лишь за состоянием маршрутизатора и за тем, отвечает ли SMTP-порт почтовому серверу. Со временем я нарастил конфигурацию, поскольку каждый сбой подсказывал мне, за чем еще я должен следить. Более

подробно программа Nagios описана в книге «Essential System Administration» (Основы системного администрирования), O'Reilly.



Если правило первого часа вам подходит, превратите его в правило первых двух часов, приходя на работу на час раньше.

Тайм-менеджмент в парке аттракционов

Давайте применим стратегию первого часа к парку аттракционов. Допустим, катание на одном аттракционе длится две минуты и одну минуту занимает переход к другому аттракциону. То есть три минуты на аттракцион. Если не стоять в очередях, за час можно успеть покататься на 20 аттракционах. Обычно в парке имеется 60 аттракционов, на которых можно провести три головокружительных часа. Это время между завтраком и обедом!

Однако в парках всегда полно народу, и если вам придется стоять в очередях по 27 минут, то за час вы успеете покататься только на двух аттракционах ($27+2+1=30$ минут на аттракцион). При таком темпе катание на всех аттракционах потребует три дня, по десять часов в день.

У кого найдется столько времени на аттракционы?

Если бы можно было получить парк в свое полное распоряжение, без очередей! Оказывается, это совсем нетрудно. Многие парки открываются на час раньше объявленного времени. Если вы появиться в парке в это время, он окажется практически в вашем распоряжении. Например, время открытия Диснейленда варьируется в течение года. Позвонив по телефону, вы можете узнать, во сколько парк откроется завтра. Какое бы время вам ни назвали, придите на час раньше, и вы обнаружите, что парк открыт. Это правда!

За первый час вы покатаетесь на двадцати аттракционах, потому что парк будет фактически пуст. По мере прихода посетителей очереди будут расти, и за следующие два часа вы скорее всего совершите меньше поездок. Когда очереди станут совсем длинными, перекусите. Другие посетители будут впустую тратить время на стояние в очередях. Ближе к полудню очереди станут короче, поскольку посетители (но не вы!) достаточно неосмотрительны, чтобы решить пообедать именно в полдень. Вскоре вы прокатитесь на всех аттракционах и допоздна сможете снова и снова кататься на тех, которые вам особенно понравились.

Тем временем все остальные либо потратят втрое больше времени, чем вы, либо опробуют лишь одну треть аттракционов.

В некоторых парках продаются экспресс-билеты, дающие право пройти на аттракцион без очереди. Теперь, зная всю эту арифметику, вы можете принять решение, стоят ли эти билеты таких денег.

Прерывания

Прерывания неизбежны. Это естественная часть нашей работы, и от нас зависит, насколько хорошо мы с ними справляемся.

Работать под управлением прерываний означает выполнять задания по мере их поступления, а не в соответствии с какой-то системой приоритетов, установленной в интересах дела. Конечно, интересы дела часто требуют именно обработки поступающих прерываний. Однако уверяю вас, что по мере вашего профессионального роста вы будете отвлекаться на прерывания все меньше и меньше. Рассмотрим организационную структуру магазина. Продавец, стоящий за прилавком, работает под управлением прерываний: к нему подходят покупатели, и он отпускает товар, отвечает на вопросы и т. д. Зато работа менеджера магазина подчиняется определенному распорядку: он открывает магазин, заказывает товары, дает распоряжения продавцам и т. д. Да, его деятельность тоже подвержена прерываниям (вопросы от персонала, непосредственные ситуации), но они составляют лишь небольшую часть работы.

Работая под управлением прерываний, мы фактически позволяем им распоряжаться нашим временем. Мы передаем управление нашим рабочим процессом в чужие руки. Естественно, я за то, чтобы вы внимательно относились к клиентам, но ваши приоритеты известны только вам. Управляя своим рабочим процессом, вы можете разумно сгруппировать задачи, чтобы сэкономить время. Например, можно выделить задачи, которые решаются в одной части здания, чтобы сократить время ходьбы по коридорам и с этажа на этаж. В главе 8 показано, что реагирование на запросы клиентов в порядке поступления может оказаться неоптимальным решением, и предложен ряд стратегий по расстановке приоритетов, позволяющих сэкономить ваше время.

Конечно, самый быстрый способ разобраться с прерыванием - крикнуть клиенту «Пошел вон!» и захлопнуть дверь. Но этот метод я рекомендую лишь тем, кто хочет лишиться рабочего места. Я встречал системных администраторов, советующих коллегам быть с клиентами поглубже, чтобы отпугнуть их. Думаю, не стоит следовать этим рекомендациям.

Перенаправление прерываний

Начнем с попытки устранить самое неприятное из возможных прерываний: к вам обращаются с проблемой, которую должен решить кто-то другой. Может быть, поступим так:

- Том, проблема с веб-сервером.
- Отлично! Сообщи мне о результатах, когда поговоришь с теми, кто отвечает за работу веб-серверов.

Нет, это было бы слишком грубо. Самое приятное в работе системного администратора то, что каждый считает вас всезнающим и всемогущим. К сожалению, большинство из нас всемогущи лишь в какой-то узкой области. Хотя не относящиеся к вашим прямым обязанностям вопросы и раздражают, нельзя сердиться на тех, кто их задает. Вы когда-нибудь обращались не по адресу *преднамеренно*? Сомневаюсь. Итак, когда кто-то задает вам вопрос, явно выходящий за рамки ваших обязанностей, поставьте себя на место этого человека. Он просто не знает, куда идти. Скорее всего, это комплимент: вы самый умный из тех, к кому он может обратиться за помощью (либо самые умные ушли обедать). В большинстве фирм далеко не очевидно, к кому лучше всего обращаться при возникновении той или иной проблемы.

Не выяснив, к кому клиент должен обратиться за помощью, вы не должны огорчаться, что он пришел не по адресу. Я довожу такую информацию до клиентов по-разному - с помощью веб-страниц, указателей на стенах, подписей в электронных сообщениях и т. д. Когда я работал в Bell Labs, у нас по дороге к комнате системных администраторов висели плакаты: «Стоп! Вы отправили электронное сообщение с просьбой о помощи?» В другой фирме первое, что я сделал, - установил внутренний веб-сайт со списком специалистов в той или иной области, к которым клиенты должны обращаться в зависимости от конкретной ситуации. В веб-браузерах эта страница была сделана стартовой, и вскоре все хорошо знали ее содержимое.

Как донести до пользователей правильную процедуру обращения за помощью? Оторвитесь от книги и оглядите свою рабочую комнату. Отойдите на 50 футов от рабочего места, развернитесь и направьтесь к нему, представляя, что вы - типичный клиент. Что вы видите на своем пути? Ведет ли он прямо к вашему рабочему месту или к другому

У вас все в порядке?

Клиенты часто достают меня вопросом «А вы знаете, что случилась неприятность?» При наличии системы мониторинга, такой как Nagios, позволяющей клиентам проводить самостоятельную проверку, количество подобных прерываний уменьшается. Однако если ваша система достаточно стабильна, у клиентов не вырабатывается привычка проверять страницу состояния системы в первую очередь. Но вы, по крайней мере, можете установить ссылку на нее на начальной странице вашей локальной сети.

Когда кто-то обнаруживает сбой в подсистеме, за которой программа Nagios не следила, я горячо благодарю его, вплоть до отправки ему электронного сообщения о том, что теперь Nagios ведет мониторинг этой подсистемы и что мы признательны ему за обнаружение сбоя, поскольку это позволило нам сделать систему мониторинга еще более эффективной.

сотруднику? Что можно сделать, чтобы клиент обратился к кому-то, кроме вас? Если у вас есть формальная многоуровневая организация системного администрирования, направляются ли клиенты к соответствующим специалистам? Как можно улучшить эту систему? Ослабить поток прерываний может большой указатель или доска объявлений, где написано, кто за что отвечает. Неплохо развесить указатели, как в аэропорту, только вместо «Регистрация», «Багаж» и «Кафе» написать «Электронная почта», «Интернет» и «Принтеры», чтобы клиенты знали, куда идти за помощью.

Можно ли обучить клиентов обращаться за помощью к соответствующему специалисту? Вполне. Первый шаг - информировать их, как они должны поступать. Затем надо сделать так, чтобы клиенты, выполняющие инструкцию, обслуживались значительно лучше. Наказание за несоблюдение правил редко бывает эффективным. Спросите любого дрессировщика, и он подтвердит, что поощрение действует лучше, чем наказание (в долгосрочной перспективе). Люди, не соблюдающие инструкцию, являются индикатором того, что она сформулирована недостаточно четко, неочевидна или неэффективна.

Увы, люди все равно будут подходить к вам, когда вы пытаетесь сосредоточиться над проектом.

Как прогнать клиента, не нахамив ему

Предположим, кто-то прерывает вашу работу. Как избавиться от него пожевлее? Секрет в том, чтобы принять его запрос со всем уважением.

Как говорилось в предыдущей главе, иногда наша работа заключается именно в обработке прерываний - в том, чтобы оградить от прерываний других системных администраторов, которым надо сосредоточиться над проектами. Однако и у нас бывает время работы над проектом, когда нам надо сохранять сосредоточенность. Как поступать, если нас отвлекают в это время?

Прежде всего, важно понять, чего от нас ожидают клиенты. В принципе клиент будет доволен, если почувствует, что его не игнорируют. Для этого не нужно тут же бросаться решать его проблему. Клиент хочет, чтобы его выслушали и заверили, что запрос будет выполнен.

Когда клиент входит в мою комнату и просит сделать что-то, что я планирую отложить на более позднее время, я вербально и визуально даю ему понять, что его запрос принят. Сперва я говорю: «Я понял Вашу проблему. Сейчас я запишу, чтобы не забыть». Далее я при нем записываю его запрос, произношу его вслух. Обычно я формулирую так: «[Сделать то-то и то-то] для [такого-то] к [такой-то дате]». Теперь я поворачиваюсь к клиенту и спрашиваю: «Я Вас правильно понял?» Если он отвечает «Да», вопрос закрыт, и клиент уходит сам, если я только не произнесу чего-нибудь такого, что разрушит ситуацию. Я экспериментально установил, что лучший вариант - сказать «Спасибо» и кивнуть.

Любой другой ответ приведет к возобновлению диалога. Когда вопрос закрыт, клиенту трудно требовать, чтобы я тут же начал действовать. Если он все-таки требует этого, значит, я недооценил срочность его проблемы, и нам следует обсудить сроки ее решения. Однако теперь уже я веду беседу и, следовательно, владею инициативой в переговорах.

Автоматизированные системы регистрации запросов тоже должны проявлять уважение к клиентам. Когда клиент отправляет электронное сообщение такой системе, она должна автоматически возвращать идентификационный номер запроса. Если клиент обращается к веб-системе, она должна немедленно вывести статус запроса, чтобы клиент был уверен, что запрос сохранен в базе данных. Людям неприятно ощущать, что они отправляют запрос в черную дыру. Идеальным вариантом был бы персональный ответ, но это нереально. Вполне достаточно автоматического подтверждения, что запрос принят. Отсутствие отклика системы держит клиента в «подвешенном» состоянии. К тому же это невежливо. Отсутствие реакции - одна из причин, по которым я не люблю отправлять большие отчеты некоторым производителям. Сейчас стало модным встраивать в приложения автоматическую отправку сообщения о произошедшем сбое. У Netscape есть FullCircle, у Microsoft - свой агент обратной связи, в Apple Mac OS X тоже есть нечто подобное. Эти производители оставляют меня в состоянии неудовлетворенности, поскольку не подтверждают получение моего сообщения. У меня нет никакого способа убедиться, что это не мистификация, призванная убедить пользователей, что производитель о них заботится, в то время как все запросы фактически уничтожаются. Я не ожидаю, что раздастся звонок от менеджера по продажам: «Помните ваше сообщение о сбое на прошлой неделе? Спасибо! Мы устранили ошибку и присвоили вам почетное звание Лучшего клиента месяца!» И все-таки было бы неплохо получить электронное уведомление о получении моего сообщения. (Должен заметить, что когда Том Рейнголд (Tom Reingold) служил в Bell Labs, он не только звонил каждому, кто прислал очередной тысячный запрос, но и приглашал этого клиента на ланч, пользуясь возможностью узнать его мнение о том, как можно улучшить обслуживание. Вот так!)

Конечно, все клиенты хотят, чтобы их запросы были удовлетворены, но каждый знает, что не всегда удастся получить желаемое. Однако если люди не чувствуют уважительного отношения, они не будут довольны. В худшем случае они решат, что вы их игнорируете; в лучшем - сочтут, что вы заняты чем-то другим, когда на самом деле вы решаете их проблему.

Разве клиенты не хотят всего и сразу? Нет. Я полагаю, что в глубине души клиенты понимают, что это не всегда возможно. Прося вас заказать новый компьютер, они отдают себе отчет, что нет смысла стоять в вашем кабинете и ждать, пока он прибывает, даже при условии поставки на следующий день. Их вполне удовлетворит, если вы примете заказ и сообщите дату выполнения заказа.

Клиентам важнее видеть деятельность, чем получить ее результаты

Однажды ко мне в комнату ворвался клиент.

«Сервер XYZ полетел!» - вскричал он в панике.

«Я им занимаюсь!» - ответил я.

Я повернулся к монитору и продолжал время от времени вводить команды. С точки зрения клиента все выглядело так, словно я вернулся к своей работе, полностью игнорируя его беспокойство.

Это было в те времена, когда еще только появились удаленные консоли и дистанционные коммутаторы «клавиатура/видеоадаптер/мышь» (KVM-устройства). Я делал все, чтобы решить проблему, но клиент не видел моей реакции на его слова.

Он был очень разочарован. По его представлениям, чтобы «починить сервер», я должен был вскочить с места, побежать по коридору, набрать секретный код на двери, ведущей в машинный зал, и начать копаться в сервере. Поскольку мое поведение не соответствовало его ожиданиям, он высказал свое недовольство, не стесняясь в выражениях. Он решил, что я собираюсь сидеть и ждать, пока сервер починится сам собой. Я сумел разубедить его, только показав, что происходит на мониторе.

Теперь, если происходит нечто подобное, я исхожу из того, что клиент ничего не знает о консольных серверах и дистанционных коммутаторах «клавиатура/видеоадаптер/мышь». Сперва я проверяю, что сервер действительно отказал и поясняю клиенту, какой тест я выполняю. Я говорю: «Запустим ping! Действительно, сервер не отвечает». Затем вместо того, чтобы броситься в машинный зал, я говорю: «Смотрите. Я могу обратиться к консоли удаленным образом, словно бы я находился в машинном зале!» Я поворачиваю монитор так, чтобы клиент видел, что я делаю. Я устраиваю небольшое «показательное выступление», а затем приступаю к непосредственному решению проблемы.

Вскоре клиенту становится скучно, и он уходит довольный, что я работаю над его вопросом.

Моя демонстрация немного тормозит работу, но все равно занимает меньше времени, чем если бы я отправился в машинный зал. А клиент удовлетворен, потому что получает визуальное подтверждение того, что его запрос принят.

Скучающий, но довольный клиент гораздо лучше клиента беспокойного и нетерпеливо ждущего.

И наоборот, клиенты будут крайне недовольны, если у них возникнет *впечатление*, что вы их игнорируете. Совершенно не имеет значения, игнорируете ли вы их на самом деле. Если вы начнете решать их проблему, но клиенты этого не будут знать, они предположат, что вы ничего не делаете. Обидно, но это правда. Надеюсь, я убедил вас, как важно принимать запросы клиентов. Не менее важно для вас организовать свое время в соответствии с вашими приоритетами. Как разрешить этот конфликт? С помощью метода «Делегируйте, регистрируйте или действуйте».

Читайте следующий раздел.

Делегируйте, регистрируйте или действуйте

Когда клиент прерывает вашу работу над проектом, у вас есть три варианта:

- **Делегировать запрос.** Если кто-то другой может решить проблему, передайте запрос ему.
- **Зарегистрировать запрос.** Если справиться с проблемой можете только вы, но она несрочная, зарегистрируйте запрос. Сделайте это убедительно для клиента; простого обещания запомнить недостаточно.
- **Действовать.** Если проблема действительно неотложная, например отказ какой-то службы, отложите текущую работу и займитесь решением проблемы.

Я каждый раз обдумываю, как поступить, - делегировать, регистрировать или действовать. Это помогает мне сосредоточиться на выполнении запроса клиента, который, увы, прервал мою работу. Подробности этого процесса описаны в следующих разделах.

Делегирование

Если вы установили взаимную защиту от прерываний, описанную в начале главы 1, вы можете отослать клиента к своему партнеру. Вовсе не обязательно говорить: «Я сейчас занят проектом, так что оправляйтесь к другому сисадмину». Можно поступить гораздо вежливее.

Поскольку людям требуется визуальное позитивное подтверждение, что их слышат и воспринимают всерьез, я думаю, что лучшей тактикой в данном случае будет звонок вашему партнеру по взаимной защите и делегирование запроса на глазах у клиента.

Люди не хотят заново объяснять свою проблему каждому, к кому их делегируют, поэтому я всегда излагаю суть вопроса своему коллеге. Я часто могу сформулировать ее в технических терминах гораздо эффективнее, чем клиент.

Итак, вот общая линия поведения. Я громко говорю: «Я попрошу Мэри заняться этой проблемой». Затем я звоню Мэри: «Привет! Тут у ме-

ня Джо. Ему нужно то-то и то-то». Я поворачиваюсь к клиенту: «Мэри ждет Вас и готова Вам помочь». Джо получает подтверждение, что его запрос принят, а Мэри готова решить проблему.



Будучи технически грамотными, мы нередко забываем, каково нетехнарю сформулировать запрос. Это очень трудно, может быть, даже страшно. Объясняя Мэри, чего хочет Джо, я облегчаю ему жизнь.

Иногда запрос оказывается достаточно сложным, и я боюсь, что в разговоре с Мэри что-то упущу. Тем не менее я должен помочь ей подготовиться. Например: «Привет! Тут у меня Джо. У него довольно сложная проблема с веб-сервером. Я его сейчас пришлю к тебе».

Конечно, бывают ситуации, когда вы спешите и просто не можете позвонить партнеру. Я думаю, неприемлемо отвечать клиенту: «А с Мэри Вы разговаривали?» Гораздо лучше: «Сейчас за это отвечает Мэри. Поговорите с ней, пожалуйста». Звучит официально и выглядит как инструкция. Следуя официальным процедурам, люди испытывают определенный комфорт.

Если ваш партнер по взаимной защите от прерываний отвечает, что не сумеет справиться с той проблемой, которую вы ему делегируете, у вас есть несколько вариантов. Во-первых, можно воспользоваться случаем, чтобы научить его самостоятельно решать такие проблемы впредь. Это окажется кстати в будущем. Во-вторых, вы можете договориться с клиентом решить его проблему позже. Если вам удастся, зарегистрируйте запрос.

Регистрация

Если вопрос может подождать, зарегистрируйте его, чтобы решить в более удобное время. Сделайте это так, чтобы он потом не потерялся. Все должно произойти на глазах клиента, чтобы он получил визуальное подтверждение, что его не игнорируют.

Если вы придерживаетесь системы Цикл, описанной в главе 5, занесите запрос в список дел. Это приемлемо для небольших задач, решаемых за короткое время.

Большие задачи я предпочитаю регистрировать с помощью специализированной программы. Я обнаружил, что приложение с открытым кодом RT от компании Best Practical (<http://www.BestPractical.com>) лучше любой коммерческой системы. (Издательство O'Reilly не так давно выпустило книгу «RT Essentials» (Основы RT), где подробно описаны конфигурирование и работа с RT.) При отправке электронного сообщения в адрес RT этот запрос автоматически регистрируется и отслеживается. Если у вас нет RT, отправьте запрос клиента самому себе по электронной почте. А пока клиентов нет, отправьте себе напоминание установить RT.

Чтобы продемонстрировать клиенту, что его запрос принят, я громко сообщаю «Сейчас я внесу Ваш запрос в список дел» или «Сейчас я сделаю запись в RT». Затем, вводя текст, я читаю его: "Джилл нужно установить принтер. Он у нее в комнате в коробке. Это надо сделать в четверг до 9 часов вечера».



Всегда записывайте время, к которому должна быть выполнена работа. Если вы напишете просто «четверг», клиент может подумать, что речь идет об утре четверга, в то время как вы имеете в виду вечер.

Введя запрос (текст которого клиент слышал), я спрашиваю: «Я ничего не упустил?» Это помогает избежать недопонимания. Кроме того, клиент получает удовлетворение от мысли, что он управляет процессом; в какой-то мере, так оно и есть.

Щелкнув по нужной кнопке, я говорю «Готово!» и возвращаюсь к своим делам. Регистрация запроса в RT, органайзере или списке дел свидетельствует о вашем профессионализме и заставляет вашего клиента доверять вам. Запись на случайных обрывках бумаги или самоклеящихся листочках производит обратный эффект.

Ни в коем случае не полагайтесь только на свою память. Я уже говорил о необходимости загружать переднюю часть мозга более важными вещами. Не пытайтесь запомнить запрос клиента. Вы умны (и отлично выглядите), но не следует поручать мозгу то, что лучше доверить бума-

Засада в коридоре

Когда кто-то останавливает меня в коридоре и обращается с просьбой, я заношу ее в список дел. Однако если у меня нет при себе органайзера, я скорее откажусь принять запрос, чем понадеюсь на свою память. Я человек честный и прямой. Я говорю примерно следующее: «Я согласен, что надо сделать то, что вы предлагаете, но сейчас я занят другой проблемой, а органайзера у меня с собой нет. Я боюсь забыть о нашем разговоре. Сделайте одолжение, отправьте мне электронное сообщение со словами «установите веб-приложение», и я все вспомню». Сообщая клиенту конкретную фразу, которую он должен мне послать, я освобождаю его от необходимости искать формулировку. Кроме того, я освобождаю свой мозг от необходимости помнить о запросе.

Если поблизости находится компьютер, с которого можно отправить электронное сообщение, я прошу человека за компьютером отправить мне сообщение, даже если он не участвовал в разговоре!

Кстати, вспомнил. Никуда не выходите без органайзера (бумажного или электронного)! Всегда носите его с собой.

ге. Запишите запрос и забудьте о нем. Освободите свой мозг. В результате нас ожидает...

Действие

Третий вариант - выполнить запрос немедленно. Вы потеряли сосредоточенность над текущей работой, но по крайней мере сделали две попытки отклонить запрос. Если его решение потребует две минуты, то дешевле сделать все сразу, чем записывать вопрос, а потом возвращаться к нему.

Конечно, если проблема неотложная или возникла аварийная ситуация, то у вас нет выбора. В конце концов, аварийная ситуация может повлиять и на вас лично, так что лучше действовать, не мешкая.

Я настоятельно рекомендую, чтобы в вашей фирме было выработано собственное определение аварийной ситуации. Вновь принятые системные администраторы получают четкую инструкцию, а если это определение будет опубликовано на внутрифирменном веб-сайте, оно послужит ориентиром для клиентов. Например, в одной группе администраторов сети, где я работал, было принято определение, что аварийной ситуацией считается сбой в сети, затрагивающий интересы более десяти пользователей. В других организациях определение аварийной ситуации основано либо на угрозе срыва сроков, либо на опасности снижения качества обслуживания.

Прежде чем вы начнете разбираться с проблемой клиента, запишите, на какой стадии вы прерываете работу, или хотя бы сохраните все файлы. Вам будет легче вернуться к проекту. Кроме того, вам будет легче сосредоточиться на новой задаче, потому что ваш мозг не будет отвлекаться, стараясь запомнить точку, в которой вы прервали свою работу.

Резюме

- Важно поддерживать сосредоточенность. Сосредоточенность достигается с помощью устранения отвлекающих факторов и эффективной обработки прерываний.
- Прерывания, в сущности, являются попыткой других людей управлять вашим временем. Прерывания - исконный враг сосредоточенности и, следовательно, тайм-менеджмента.
- Прерывания наносят вред, поскольку они задерживают выполнение текущей работы, а возвращение к ней чревато ошибками с вашей стороны. Исправление этих ошибок может занять больше времени, чем сама работа.
- Устранение отвлекающих факторов позволяет вам поддерживать сосредоточенность. Уберите все лишнее с рабочего стола и с экрана компьютера. Уберите отвлекающие вас предметы из рабочей ком-

наты. Отключите интернет-пейджеры, уведомления о новых электронных письмах и т. п.

- У каждого свой период пика умственной и физической активности. Выясните, в какое время наступает ваш пик, и соответственно планируйте рабочий день.
- Первый час рабочего дня может оказаться самым продуктивным, поскольку у вас будет минимум прерываний. Придя на работу чуть раньше коллег, вы сможете увеличить этот отрезок. Не тратьте это время на обслуживание системы; воспользуйтесь им для работы над проектами.

Некоторые общие советы

К сожалению, в этой книге невозможно дать конкретные рекомендации о том, как вы должны работать. Я даже не знаю, какая операционная система у вас установлена. Тем не менее я могу посоветовать следующее:

- **Семь раз отмерьте, один раз отрежьте.** Прежде чем совершить необратимые действия, оцените все последствия.
- **Создайте резервную копию до того, как измените файл.** Резервная копия избавит вас от многих проблем. Однако создавать ее следует до внесения изменений!
- **Если ничто не помогает, почитайте документацию.** Если вам не удастся найти решение, обратитесь к источникам, о которых вы часто забываете.
- **Во время отладки вносите только одно изменение за раз.** Внося только одно изменение, вы видите, что повлияло на систему. Так вам удастся избежать путаницы в процессе отладки.
- **Всегда тестируйте свою работу.** Некоторые люди, кажется, никогда не ошибаются. Однако я обнаружил, что они тщательно тестируют то, что сделали; просто мы этого не видим.
- **Работа не закончена, пока ее не протестирует клиент.** Вам может казаться, что вы сделали все, как надо, но пока клиент не проверил вашу работу, вы не сможете узнать, решили ли вы его проблему.
- **Самые странные проблемы часто происходят от неверной конфигурации DNS.** DNS влияет на работу многих подсистем, часто самым непонятным образом. Нередко проблема с DNS маскируется под другую проблему. Это справедливо как в отношении клиентских компьютеров, которым недоступны их DNS-серверы, так и в отношении клиентов, которые пытаются достичь адресата, пользуясь ошибочной информацией, полученной от DNS.

- Правило «Делегируйте, регистрируйте или действуйте» позволяет вам управлять своим временем. Придерживайтесь его, когда вашу работу прерывают. *Делегирование* задачи означает поручение ее другому сотруднику. *Регистрация* позволяет вам зафиксировать вопрос, но решить его в более удобное время. *Действие* неизбежно лишь в самом крайнем случае, например при отказе системы.
- Регистрируя запрос, вы получаете возможность планировать свое время, не завися от прерываний. Мы обсудим эту тему в главе 8.
- Принимая запрос клиента, вы должны всем своим видом продемонстрировать это. Убедитесь, что клиент видит, как вы регистрируете запрос, и получите его подтверждение.
- Клиент предпочитает знать наверняка, что его запрос принят (даже если он будет выполнен позже), а не гадать, дошел ли он до вас.
- Система регистрации клиентских запросов, такая как RT, позволяет вам хранить их в единой базе данных, чтобы другие системные администраторы могли их обрабатывать, а клиенты - проверять текущий статус своих запросов.
- Никогда не полагайтесь на память, если клиент обращается к вам с запросом. Записывайте запрос на бумаге или в электронном виде. Мозг пригодится вам для более важной работы.

Главное - начать

Начав какое-то дело, вы поймете, что все не так сложно, как ожидалось. Мы часто не беремся за работу, ища оправдание в том, что у нас нет на нее времени. Однако, приступив к действиям, видим, что она выполняется довольно быстро.

Одна знакомая обещала мне читать рукопись этой книги и делиться впечатлениями, но ее отзыв о первой главе пришел на несколько недель позже, чем я ожидал. Она откладывала это дело, потому что убедила себя, что ей потребуется непрерывный двухчасовой отрезок времени, чтобы выполнить работу как следует. Оказалось, что в первой главе меньше десяти страниц, и знакомая прочитала ее всю за полчаса.

Если бы она просто начала читать, а не размышляла, когда лучше начать, то справилась бы гораздо быстрее.

3

Рутинные процедуры

Слово «рутинный» имеет отрицательный оттенок. Анонсы новых программных продуктов пестрят обещаниями избавить пользователя от рутинной работы. «Рутинный» означает «скучный», не так ли?

Не так! Как системный администратор я приветствую скуку. Я мечтаю, чтобы в течение недели все происходило по расписанию, проекты завершались в срок, приложения устанавливались без проблем, а документация содержала ответы на мои вопросы. «Дайте мне хотя бы один скучный день!» - кричу я, когда главный сервер выходит из строя или клиент обращается со срочной, но неразрешимой проблемой.

Я бы все отдал за скучный месяц!

Улучшить ситуацию позволяют технические средства. Мы можем сделать свою жизнь скучнее (в хорошем смысле!) с помощью долгосрочного планирования и подходящей инфраструктуры, обеспечивающей плавный ход событий. Например: автоматизация установки программного обеспечения новых компьютеров, при которой все они устанавливаются идентично; планирование обновлений, чтобы не было компьютеров с морально устаревшими компонентами; поддержание инфраструктуры безопасности, чтобы она стала повсеместной и менее обременительной. На эти темы уже написано много книг. Я предпочитаю «The Practice of System and Network Administration» (Практическое системное администрирование и администрирование сетей), Addison Wesley.

Я не собираюсь сделать системное администрирование скучным на 100%. Не думаю, что это возможно в принципе. Пока появляются новые программные пакеты, которые необходимо испытать, и новые платформы, которые необходимо исследовать, у системных администраторов будет достаточно развлечений.

Кроме того, всегда будет оставаться определенная степень беспорядка. Системный администратор имеет дело с реальным миром, а реальный мир полон хаоса.

Однако я все-таки хочу исключить тот беспорядок, который не позволяет мне получать удовольствие от работы. Вот что следует накрепко запомнить о процедурах:

Процедуры дают нам способ подумать один раз, а сделать много раз.

Программисты давно это поняли. Они используют библиотечные коды и не создают каждую новую программу с чистого листа. Зачем каждый раз заново писать функцию вывода? Конечно, функция printf языка C не самый эффективный способ вывода отформатированных данных, но представьте, как было бы глупо (и неэффективно), если бы способ вывода данных изобретался для каждой программы отдельно.

Процедуры - очень мощный инструмент. Они позволяют нам меньше думать, сохраняя умственную энергию для более важных задач. Здесь ситуация аналогична той, когда мы записываем намеченные встречи и необходимые дела вместо того, чтобы помнить о них.



Примеры процедур

Точно так же мы можем поступать и в тайм-менеджменте, разрабатывая процедуры везде, где только возможно. Вот несколько примеров.

Процедура 1: заправка по воскресеньям

Я заправляю машину каждое воскресенье. Я разработал эту процедуру, и она хорошо служит мне.

Все началось в тот день, когда я осознал, что часто опаздываю на работу по понедельникам и что опаздываю вдвойне, если по дороге обнаруживаю, что мне необходимо заправиться. Я пробовал выезжать пораньше по понедельникам, но эта затея провалилась. В конце концов я понял, что разумнее всего будет заправляться по воскресеньям, чтобы в понедельник утром у меня было на одно дело меньше. Сработало.

Раньше я все время откладывал заправку. Это вносило в мою жизнь некий дополнительный хаос, поскольку из-за необходимости заправлять машину я опаздывал на некоторые встречи.

Я не просто изо дня в день откладывал заправку, я постоянно беспокоился! «Заправиться ли сейчас? Пожалуй, да». «Господи, я же опаздываю; может быть, заправлюсь завтра. Я уверен, что не забуду выехать из дома на час раньше». «Я же собирался заправиться вчера вечером, но так устал, что совсем забыл. Черт возьми!» Огромное количество умственной энергии уходило на пустяки.

Теперь у меня первая половина недели (а то и вся неделя, если я мало езжу) упорядочена, по крайней мере в этом отношении.

Это удачная и простая процедура, и она прекрасно работает.

Та часть моего мозга, которая активно обдумывает дела, теперь не должна была думать о заправке, и вскоре привычка стала автоматической. Выезжая из дому в воскресенье, я обязательно заливаю полный бак.

Достоинство хорошей процедуры в том, что после достаточной практики вы выполняете ее, не задумываясь. Если вы не думаете о заправке, ваша умственная энергия высвобождается для других целей. Со временем вы, возможно, забудете о том, почему выработали процедуру. Это нормально, даже хорошо. Вы ведь не задумываетесь о том, как дышите; это автоматическая функция мозга. Представьте, что каждые несколько секунд вам пришлось бы бросать все дела и вспоминать о необходимости дыхания, принимать решение, что пора бы подышать, затем сосредотачиваться, напрягать мышцы и делать вдох и выдох!

Однажды во время очередной воскресной заправки я проговорился своему пассажиру, что всегда заправляюсь по воскресеньям. Он спросил почему, но я не смог вспомнить. Я просто знал, что делаю это уже довольно давно, заведя такую процедуру, чтобы решить какую-то проблему, и что эта проблема не возникает вот уже лет десять.

Ого! Какая автономия! Мне потребовалась минута или даже две, чтобы вспомнить первоначальную причину. Как было бы здорово, если бы другие дела, вызывающие наше беспокойство, стали автоматическими функциями!

Процедура 2: никуда без органайзера

Теоретически органайзер должен быть у меня под рукой, когда он мне понадобится. Я знаю, что он нужен мне на работе. Это очевидно. Но иногда он нужен дома. Оставить ли его на работе в предположении, что он не потребуется мне сегодня вечером? Оставить его в машине или забрать домой? «Бог с ним, оставлю его здесь, он мне вечером не нужен».

Затем оказывается, что он мне все-таки нужен, но, поскольку мне лень идти за ним в машину, я соглашаюсь поужинать с друзьями в четверг, потому что мне кажется, что вечер четверга у меня свободен. Потом я либо пропускаю этот ужин (так как не занес его в список встреч), либо оказывается, что у меня на это время уже назначено другое дело, и мне приходится менять планы, доставляя массу неудобств себе и другим.

Безусловно, менять планы гораздо тяжелее, чем сходить в машину за органайзером. Однако я не иду за ним, потому что в этот момент мне кажется, что легче все же попытаться запомнить договоренность. Я выбираю наилегчайший вариант из имеющихся непосредственно сейчас, а не в теоретическом будущем, когда окажется, что я ошибся. Посмотрите на меня! Я сэкономил прогулку до машины и обратно!

Гораздо худший сценарий развивается, когда я сажусь утром в машину, чтобы поехать на работу, и обнаруживаю, что у меня нет органайзера. Я думаю: «Где же он? Здесь его нет. Я отнес его домой? Не знаю. Должно быть, я оставил его на работе».

Естественно, когда я приезжаю на работу, оказывается, что я действительно оставил органайзер дома. Теперь мне приходится весь день обходиться без него. Запланированные дела перепутываются, встречи срываются - ужас!

Чтобы помочь себе разработать эту процедуру, я нашел прекрасное закливание:

На вопрос «Брать ли с собой органайзер?» есть только один ответ: «Да».

Возникает позитивный эффект домино. Уходя с работы, я беру органайзер. Если, выйдя из дома, я замечаю, что органайзера нет в машине, я возвращаюсь за ним домой. Поскольку я всегда беру с собой органайзер, я знаю, что не оставил его на работе накануне вечером.

Вот почему я потерял органайзер только один раз за 14 лет. Каждый раз, когда я выхожу из комнаты, ухожу с работы, сажусь в машину, выхожу из машины и т.д., я обязательно держу в руке органайзер. Благодаря этому непреложному правилу привычка выработалась очень быстро и закрепилась прочно. Иногда мы кладем вещи не на место и теряем их след. Мы кладем куда-то предмет, а потом выходим из комнаты без него, потому что не имеем привычки все время носить его с собой. Если мы не носим органайзер всегда, то наш мозг не беспокоится, если у нас нет его *сейчас*. Я выработал привычку, практически тактильное пристрастие, всегда держать органайзер в руке.



В тот единственный раз, когда я потерял органайзер, я очень спешил и был вынужден нести много вещей одновременно. Должен заметить, что компания по прокату автомобилей прислала мне органайзер на другой день следующим самолетным рейсом. Мне сильно повезло.

Подумайте, каких нужных вещей сейчас при вас нет. Почему вы не носите их с собой постоянно?

Просмотрите список вещей, которые проще взять с собой, чем тратить умственную энергию на размышление о том, брать или не брать:

- Органайзер (бумажный или электронный)

- Сотовый телефон и/или пейджер
- Ручка
- Бумажник, кошелек и т. п.
- Пластиковые карточки
- Ключи (металлические или электронные)
- Пластиковые карточки с медицинской информацией, в том числе о страховке
- Ноутбук (для некоторых)

Может, кого-то и насмешит ваша экипировка. Но большинство понимает, как важно всегда иметь при себе авторучку.

Процедура 3: регулярные встречи с шефом

Мне нужно время от времени общаться с шефом. Я ценю свою независимость, но все имеет свои пределы. Соглашение о регулярных встречах позволило мне сэкономить массу времени. Если учесть все время, когда я отправлял шефу электронные письма с просьбой принять меня или разговаривал с секретаршей, то получится, что я тратил по полчаса на организацию 15-минутной встречи. Это просто глупо.

Итак, мы договорились встречаться или общаться по телефону каждый вторник в 10 часов утра независимо от того, есть ли в этом экстренная необходимость. Теперь на организацию 15-минутной встречи не уходит ни минуты.

Возможно, вы решите перенять этот опыт, особенно если вам кажется, что вы маловато общаетесь со своим шефом. Ежедневная пятиминутка в 9 часов утра может оказаться полезнее для вас и меньше досадит ему, чем ваши попытки отвлечь его в течение дня.

Парадоксально, но это помогает и в том случае, когда шеф считает, что вы требуете от него слишком много внимания. Если цель половины ваших попыток встретиться с шефом - просто договориться о беседе, то, возможно, вам лучше назначить время для регулярных совещаний. Это придаст им форму.

Процедура 4: обход подчиненных

Когда я работал в Bell Labs, у меня в подчинении было 15 других системных администраторов. Мне не хотелось командовать ими: все они были умны, трудолюбивы и независимы. Я предоставил их самим себе. Однако вскоре я узнал, что они чувствуют себя брошенными. Нужно было уделять им больше времени.

Однако организация мини-совещаний с пятнадцатью сотрудниками заняла бы больше времени, чем сами совещания, а польза от них в беспорядочной среде системного администрирования была бы минимальной. Поэтому каждый понедельник и четверг в 9 часов утра я делал обход.

Я проходил по маршруту, включающему рабочие комнаты всех системных администраторов. Так получилось, что они были сгруппированы в трех разных частях здания, поэтому у меня получалось три мини-совещания. Я входил, здоровался, и у моих подчиненных появлялась возможность обсудить насущные вопросы.

Такой обход занимал у меня полдня, но он действительно позволял решать проблемы в реальном времени, устранять «узкие места» и проявлять внимание к подчиненным.

Если у вас возникли эти вопросы, ответ на них - «Да»

С годами я решил, что на следующие вопросы всегда нужно отвечать «Да». Если они возникают вновь, теперь я не трачу умственную энергию на поиски ответа.

- Сохранить ли прямо сейчас файл, над которым я работаю?
- Взять ли с собой органайзер (или оставить его здесь)?
- Занести ли это дело в свой список дел?
- Просмотреть ли календарь, прежде чем я соглашусь на эту встречу?
- Занести ли эту договоренность в календарь?
- Просмотреть ли мои планы на вечер, прежде чем я соглашусь задержаться на работе?
- Проверить ли список утренних встреч, прежде чем я сыграю в *Half Life* еще разок до начала рабочего дня?
- Выполнить ли Цикл сегодня (или расслабиться)?
- Заправить ли машину прямо сейчас (или тянуть с заправкой до последнего момента)?
- Выполнить ли эту небольшую работу прямо сейчас (или отложить ее в надежде, что про нее забудут или она никогда не превратится в неотложную)?

Ответом на все эти вопросы будет «Да». Этот список появился в результате десяти лет работы, в течение которых у меня были проблемы (маленькие и не очень), вызванные тем, что каждый раз я размышлял, взвешивал все «за» и «против» и принимал обдуманное, но неверное решение. Я старался вести себя разумно. Прошло много времени, пока я понял: хватит размышлять! Ответ всегда «Да». Не надо ничего взвешивать, не надо тратить умственную энергию на принятие решений, не надо убеждать себя, что на этот раз ситуация в корне иная! Если вы спрашиваете себя, ответом будет «Да»!

В большинстве случаев на принятие решения уходит больше времени, чем на само дело. 10 секунд нужно на то, чтобы открыть органайзер и просмотреть календарь, и почти столько же, чтобы поразмышлять о надежности своей памяти и о ненужности проверки.

Многие из этих вопросов эквивалентны вопросу «Чьей памяти следует доверять- моей или органайзера?» Мы уже знаем, что наша память ненадежна, иначе зачем мы покупали органайзер? Так пользуйтесь им!

Я почти 10 лет разрабатывал правила на каждый из этих случаев, и по удивительному совпадению ответ на все вопросы один и тот же. Избавьте себя от болезненного опыта и поверьте мне: ответ - всегда «Да».

Еженедельные совещания у руководства проводились по вторникам. Обход в понедельник обычно снимал массу вопросов, которые в противном случае обсуждались бы на еженедельном совещании. Это позволило сократить совещания по вторникам. Короткие совещания - это здорово!

Меня удивило, как хорошо работает мой метод. Меня также удивило, что все его заметили. Однажды, подходя к комнате системных администраторов, я нечаянно подслушал, как кто-то произнес: «А вот и Том наносит нам визит», после чего раздался короткий смешок.

Они смеялись надо мной. Думаете, я что-то изменил? Стал совершать обход по разным дням, чтобы быть менее предсказуемым? Нет. Для этого я слишком невозмутим.

Со временем я заметил, что подчиненные планируют рабочий день в соответствии с моими визитами. Порой, когда я заходил, у них на доске уже был список вопросов, необходимых для обсуждения.

Из этой истории можно извлечь два урока:

- Разработайте процедуру, которая решает ваши проблемы.
- Выполняйте эту процедуру предсказуемо, чтобы другие могли планировать свои дела в соответствии с нею.

Процедура 5: обход клиентов

Если вы как системный администратор обслуживаете сотрудников, работающих в том же здании, вы можете повысить степень их удовлетворенности с помощью ежедневного обхода. Заходите к клиентам, разговаривайте с ними, отвечайте на вопросы, устраняйте мелкие проблемы, регистрируйте более сложные, чтобы заняться ими позже, и т. д. Как минимум это улучшит ваше взаимопонимание, что уже само по себе ценно.

Некоторое время я работал с системным администратором, у которого были очень застенчивые клиенты. Они были прекрасными специалистами, но плохо разбирались в компьютерах. Обычно они не сообщали о своих проблемах отчасти из-за своей застенчивости, отчасти из-за того, что предыдущий системный администратор всегда находился в дурном настроении. В результате они прибегали к каким-то малоэффективным обходным решениям. Когда это выяснилось, мой коллега быстро все исправил, облегчив им жизнь.

Когда я узнал, что он совершает ежедневный обход клиентов для решения проблем, я был потрясен. Это шло вразрез с нашей политикой фиксирования всех проблем с помощью системы регистрации запросов! Такое поведение бросало вызов нашим попыткам заставить клиентов отправлять электронные сообщения с просьбой о помощи. Разве это правильно?

Вскоре я понял, что это очень правильно. Люди (особенно те, кто не разбирается в компьютерах) обычно не сообщают о том, что причиняет им неудобство в работе, думая, что так и должно быть. Регулярный обход клиентов позволил моему коллеге устранить эти неудобства и повысить продуктивность работы. Кроме того, это помогло наладить хорошие взаимоотношения между системным администратором и клиентами. В результате системного администратора стали приглашать на обсуждение будущих проектов, что позволило избежать многих проблем.

Не применяйте этот метод, если вы не умеете говорить людям «нет». Одна из причин его эффективности заключалась в том, что мой коллега следовал принципу, являющемуся разновидностью принципа «Делегируйте, регистрируйте или действуйте», описанного в главе 2. Я назову эту разновидность так: «Исправляйте, регистрируйте или сочувствуйте».

- **Исправление.** Если проблема решалась быстрее, чем за две минуты, мой коллега исправлял все тут же.
- **Регистрация.** Если проблема была серьезней, он помогал клиенту составить электронное сообщение и отправить его системе регистрации запросов. Эта группа клиентов не привыкла отправлять запросы и побаивалась делать это. Под руководством системного администратора процедура не казалась им такой страшной.
- **Сочувствие.** Иногда клиенты обращались с невыполнимыми просьбами или сообщали об известной проблеме, решение которой было уже запланировано на будущее. Мой коллега обнаружил, что в этом случае самое лучшее - выразить сочувствие, но без снисхождения. «Да, - вздыхал он, - весьма неудобно, что система работает таким образом». Клиент соглашался и был доволен тем, что на его жалобу отреагировали. Затем мой коллега говорил: «Не думаю, что здесь можно что-то исправить, но, вероятно, кто-нибудь найдет решение». Клиент выигрывал от того, что получал подтверждение неустранимости проблемы, а не оставался в неведении. Мой коллега

выигрывал от того, что нерешаемые вопросы не попадали в систему регистрации, а он лучше понимал положение дел. Он регистрировал такие запросы в своем органайзере. Если решение все-таки находилось, он возвращался к клиенту и выглядел чудотворцем.

Важный момент заключался в том, что этот системный администратор не пытался решить каждую проблему на месте. Иногда обход клиентов был просто самым эффективным способом сбора запросов, которые выполнялись позже. В других случаях мой коллега устанавливал с клиентами хорошие отношения, позволявшие лучше понять их нужды в долгосрочной перспективе. А иногда это был просто способ выразить сочувствие людям, столкнувшимся с неразрешимой проблемой.

Я подозреваю, что когда мой коллега приступил к своим обходам, он был ошеломлен огромным количеством вопросов, с которыми к нему обращались. Как я уже сказал, не используйте этот метод, если вы не умеете говорить людям «нет». Он требует самодисциплины, иначе вы весь день проговорите с первым клиентом. Однако со временем первоначальный наплыв нерешенных вопросов рассосется, и регулярный обход станет сродни процедуре обслуживания техники.

Процедура 6: предварительная компиляция действий системного администратора

В предисловии я рассказал историю о замене магнитных лент для резервного копирования. Это была нелегкая задача, поскольку имелось восемь ленточных серверов, и на любом из них в любой день могла потребоваться новая лента (а могла и не потребоваться). Каждый день я принимался высчитывать, на каких лентах накопилось достаточно информации, чтобы нужно было установить пустую ленту (иными словами, на каких лентах не осталось места для резервных копий, которые будут созданы сегодня вечером). Затем я брал новые ленты и обходил восемь серверов, разбросанных по всему зданию.

В конце концов я осознал, что могу избежать мороки с расчетами, если буду менять ленты на «больших» серверах каждый день, а на «маленьких» один или два раза в неделю. Это привело к огромной экономии не только моего времени, но и ресурсов моего мозга.

Это был тот случай, когда надо было «не думать, а действовать». Конечно, лента при этом расходовалась не самым экономным образом, но мое время дороже ленты.

В продолжение этой истории замечу, что я обычно заменял ленты в конце рабочего дня. Если я был поглощен проектом (а почти всегда так оно и было), я вспоминал о замене лент достаточно поздно. Я и так засиживался допоздна, а из-за необходимости менять ленты уходил с работы еще позже. Когда я направлялся после работы домой или на одно из многочисленных общественных мероприятий, в которых неизменно участвовал, я злился на «эти чертовы ленты, из-за которых опять опаздываю».

Требовалось разработать более удачное расписание, и я придумал такое заклинание:

Ежедневные дела надо делать в начале дня.

Впоследствии, когда каждое мое утро стало начинаться с планирования согласно системе Цикл, наивысший приоритет я назначал делу «Замена лент».

В результате у меня стало одной головной болью меньше (в том смысле, что мне больше не нужно было весь день размышлять, какую ленту заменить), и я смог сосредоточиться на работе и меньше беспокоиться. Я приходил домой в хорошем настроении и не очень поздно. В начале каждого рабочего дня я испытывал ощущение, что сделал что-то важное, да так оно и было.

Процедура 7: в случае сбоя связывайтесь с руководством

Однажды в фирме, где я работал, произошел сбой электропитания. Ситуация усугублялась отсутствием связи между системными администраторами, с одной стороны, и руководством и клиентами, с другой. Руководство считало, что его следовало известить о проблеме раньше, а системные администраторы считали, что им не дают спокойно разобраться в проблеме. Я уверен, что с вами ничего подобного не случилось. Или все-таки случалось?..

После этого события мы разработали процедуру на будущее. В конце концов, не последний же это сбой.

Процедура была проста: в течение часа сообщить о сбое конкретному менеджеру (шефу старшего системного администратора), даже если это случилось поздно ночью. Системные администраторы далее должны каждые полчаса докладывать этому менеджеру о состоянии дел до тех пор, пока проблема не будет устранена. Он со своей стороны должен известить более высокое руководство и клиентов (если сбой не привел к нарушению связи с ними), чтобы системные администраторы могли сосредоточиться на проблеме.

Это была простая процедура, и она хорошо работала. Жаль, ее не существовало во время той первой аварии.

Если ваша компания работает на виду у всех (приветствую тружеников Amazon, Google и Yahoo!), подобная процедура должна охватывать и отдел по связям с общественностью. Важно разработать эту процедуру до первого серьезного сбоя, даже если дискуссия обещает быть напряженной. Иногда сбои столь грандиозны, что привлекают внимание прессы. Нетрудно представить себе, какая шумиха может подняться. Несколько лет назад достаточно было сказать «Интернет» или «компьютерная безопасность», чтобы сбежались представители всех средств массовой информации. (Теперь они пресытились, и сообщения типа «Дыра в системе безопасности Microsoft принесла убытки миллиону фирм» больше не считаются новостями.) Тем не менее, если ваш биз-

нес достаточно широк, важно заранее совместно с отделом по связям с общественностью разработать стратегию общения с прессой. Вы должны знать, к кому направить журналистов, если они станут вам звонить. Если такого плана нет, самое лучшее - отвечать «Без комментариев» и класть трубку еще до того, как вам захочется что-нибудь добавить. Поговорить с журналистом весьма заманчиво, но многие системные администраторы считают, что во время аварии следует решать технические проблемы. Пусть прессой занимаются те, кому положено.

Процедура 8: автоматическая проверка при выполнении некоторых действий

Чтобы не запереть ключи в машине, я выработал такую привычку: закрывая дверцу правой рукой, я сжимаю левый кулак, чтобы почувствовать ключи. Я захопываю дверцу, только если держу в левой руке ключи. Аналогичный ритуал я выполняю, выходя из дома.

Нельзя сказать, что я часто оставлял ключи за запертой дверью, но случалось это каждый раз в самое неподходящее время, и на решение проблемы уходило несколько часов.

Полезная настройка команды ping

Очень удобно, если *ping* издает звуковой сигнал после каждой успешной попытки. Вы можете находиться в любом месте комнаты и отсоединять кабели, не подбегая то и дело к экрану для проверки *ping*.

В Linux звуковой сигнал включается с помощью опции `-a` команды *ping*.

В ОС Solaris и других UNIX-подобных системах, где отсутствует опция `-a`, можно прибегнуть к следующему трюку. Вывод программы *ping* содержит двоеточие только в тех строчках, которые сообщают об удачной попытке. Направьте вывод на вход команды `tr`, транслирующей каждое двоеточие в символ `Ctrl-G` (то есть звуковой сигнал).

```
$ ping -s 64.32.179.56 | tr : ^G
```

(Чтобы «пингование» было непрерывным, в Solaris¹ нужно задать переключатель `-s`. В других системах это не требуется.)

Чтобы последовательность `Ctrl-G` отобразилась в командной строке, вам, возможно, придется предварить ее последовательностью `Ctrl-V`. То есть вы должны напечатать:

```
$ ping -s 64.32.179.56 | TR : CTRL-V CTRL-G
```

¹ И в Windows тоже. — *Примеч. науч. ред.*

Какое отношение это имеет к системному администрированию? Есть множество автоматических проверок, которые мы можем сделать частью своей работы:

- Выходя из комнаты, запирающейся на кодовый замок, я проверяю, находится ли электронный ключ у меня в кармане. (Смежное правило: я никогда ни на секунду не кладу электронный ключ на стол, на пол, куда-то еще. Он всегда у меня в кармане, а карман всегда при мне.)
- Находясь рядом с оборудованием, я всегда проверяю поток воздуха. В частности, смотрю, не перекрыты ли вентиляторы посторонними предметами.
- Всякий раз, когда в фирме появляется новый сотрудник, я знакомлюсь с ним, решаю его текущие проблемы, если таковые имеются, и объясняю, как запрашивать помощь в будущем. Если я тут же решу его проблемы, он быстрее приступит к работе, а чем раньше я научу его пользоваться системой регистрации запросов (а не звонить мне напрямую), тем лучше я смогу планировать свое время.
- Встречая незнакомого человека, я всегда улыбаюсь, представляюсь и спрашиваю его имя. Я прошу разрешения изучить его бейдж, объясняя это тем, что моя зрительная память работает лучше. Новые сотрудники думают, что это я такой приветливый. На самом деле я слежу, не проник ли в фирму злоумышленник.
- Прежде чем отсоединить сетевой кабель, я запускаю программу *ping* (с ежесекундной проверкой), которая регистрирует обрыв связи, если я отсоединю не тот кабель.
- Каждый раз, когда я добавляю новое правило для брандмауэра, я сначала запускаю демонстрацию того, что собираюсь заблокировать, чтобы убедиться, что этой блокировки еще нет. Затем я создаю правило для брандмауэра. Я снова повторяю демонстрацию, чтобы проверить, работает ли правило. (Если я не запущу демонстрацию до создания правила, я не смогу удостовериться, что оно работает, как задумано.)

Процедура 9: создание резервной копии перед редактированием файла

Собираясь редактировать файл конфигурации, я всегда создаю его резервную копию. Я никогда не размышляю о том, достаточно ли ценность он представляет. Если возник такой вопрос, на него следует ответить «Да». Я всегда создаю резервные копии по единому правилу, не тратя время на поиски оптимального решения. Моя система состоит в том, чтобы имя копии включало в себя сегодняшнюю дату. Например, файл *named.conf* копируется в файл *named.conf-20060120* (20 января 2006 года). Я пробовал использовать дату последнего изменения файла, но пришел к выводу, что лучше включать в имя файла сего-

дняшнюю дату. Это позволяет отследить, когда я редактировал файл. В UNIX я могу просмотреть файл в репозитории RCS и увидеть историю изменений, практически уходящую в бесконечность (более подробно об этом сказано в главе 13).

Всегда есть соблазн сказать себе: «Я вношу незначительное изменение, которое всегда смогу отменить вручную» или «Я эксперт; я ничего не перепутаю». Однако опыт показывает, что все-таки лучше создать резервную копию. Особенно если через три недели придется ломать голову над тем, почему одна из служб прекратила функционирование.

Процедура 10: список вещей в дорогу

Я много путешествую. Раньше я то и дело забывал что-нибудь взять в дорогу, а если и не забывал, то все равно нервничал: а вдруг все же что-то забыл? Кому нужен такой стресс?

Теперь я записываю все, что нужно взять в поездку, в правой части странички органайзера, на которой стоит дата поездки. В течение нескольких недель (или месяцев), предшествующих поездке, я записываю в органайзер все, что мне может понадобиться. Поскольку с органайзером я не расстаюсь, ни одна идея не пропадает.

Когда я пакую чемодан, я постепенно зачеркиваю пункты этого списка.

Я создаю второй список, куда включаю вещи, которые необходимо иметь под рукой в момент выхода из дома. Это билеты, бумажник, чемоданы и т. д. Такой список помогает мне уложить все в багажник. Если меня подвозит приятель, я ставлю все эти вещи у дверей, чтобы не терять время, когда он за мной заедет.

Я использую эти списки как для командировок, так и для неделовых поездок. Я не хочу отказываться от привычки только потому, что отправляюсь в путешествие ради удовольствия. На основе старых списков я составляю новые. Базовый список хранится в моем органайзере.

Как разрабатывать собственные процедуры

Вы рассмотрели несколько процедур, которые служат мне верой и правдой. Как вам разработать собственные? Вот некоторые моменты, на которые следует обратить внимание.

- **Повторяющиеся события, не включенные в расписание.** Часто бывает, что какое-то дело или совещание повторяется несколько раз в неделю (или месяц), но в план не включается. Улучшится ли ситуация, если вы включите это дело в расписание? Не уходит ли у вас больше энергии на то, чтобы договориться о встрече, чем на подготовку к ней? Если это так, то разработайте расписание. Предложите день и час (либо какую-то последовательность дней) для регулярных совещаний и добейтесь согласия остальных участников.

- **Обслуживание техники.** Работа системного администратора во многом похожа на работу садовника. Вы должны понемногу полоть сорняки каждую неделю. Нельзя сделать всю прополку за пару выходов в начале лета и потом загорать до конца сезона. Если вам приходится выполнять какую-то работу ежедневно (еженедельно, ежемесячно), превратите ее в процедуру. Если вы затеяли выбрасывать хлам, посвятите этому делу один час ежедневно. Если вы проводите ревизию пользовательской базы данных, проверяя записи уволившихся сотрудников, просматривайте по 100 учетных записей ежедневно, пока не закончите.
- **Поддержка дружеских и деловых взаимоотношений.** Человеческие взаимоотношения требуют поддержки, и здесь тоже уместна аналогия с работой садовника. (Они расцветают, если вы трудолюбивы, чахнут, если забыли о них, и погибают, если вы чересчур усердны.) Необходимо поддерживать взаимоотношения с четырьмя группами: клиенты (или представители групп клиентов), подчиненные, системные администраторы других компаний и начальство. Регулярно ли вы общаетесь с представителями этих групп? Ключ к профессиональному росту в том, чтобы поддерживать взаимоотношения круглый год, а не только подыскивая новую работу. Включите в свое расписание ежемесячный ланч с вашим руководителем или с кем-то из вашего общества.
- **Когда размышления отнимают больше времени, чем само дело.** Если вы заметили, что на размышления о каком-то деле уходит больше времени, чем заняло бы его выполнение, сделайте его. (Не следует путать размышления о деле и размышления по ходу его выполнения.)
- **Вещи, которые вы часто забываете.** Столкнувшись в очередной раз с неприятностями из-за своей забывчивости, выработайте процедуру на будущее. Каждый вечер, приходя домой, вешайте ключи на одно и то же место, и у вас появится привычка брать их оттуда, выходя из дома. Если, например, вы должны что-то взять с собой, выходя из дома, ставьте этот предмет так, чтобы он мешал открыть дверь, и вы точно его не забудете. Необходимо согласовать эти процедуры с членами вашей семьи. Нет смысла всегда класть бумажник и ключи на столик в прихожей, если ваш(а) супруг(а) все равно уберет их куда-нибудь.
- **Незначительные или низкоприоритетные дела, которые можно (но не стоит) игнорировать.** Бывают дела, которые можно иногда проигнорировать, и ничего плохого не случится. Однако если вы поступите так неоднократно, у вас начнутся проблемы. Примерами таких дел являются замена лент для копирования, заказ расходных материалов и т. д. Занесите пункт «заказать расходные» (или что там у вас) в список дел на каждый понедельник. Лучше проигнори-

ровать это напоминание в тот раз, когда нет необходимости ничего заказывать, чем забыть сделать заказ.

- **Обучение.** Некоторые жалуются, что у них нет времени на повышение профессионального уровня. Другие планируют посещение одних курсов в год, даже если не знают, что там будет. Короче, ничего не произойдет само собой, если вы не приложите усилий.
- **Быть в курсе.** Лучше запланировать один час в неделю на чтение журналов, чем пытаться «догнать поезд» раз в несколько месяцев. Выбрасывайте все непочитанные журналы раз в месяц. Если вы не смогли прочитать журнал до выхода следующего номера, вы никогда его не прочитаете. Если у вас нет отдельного кабинета, где бы вы могли запереться и спокойно почитать, найдите другое место, где клиенты вас не достанут.

Регулярно встречайтесь с представителем клиентов

Когда я работал в Bell Labs, каждый системный администратор обслуживал две-три группы клиентов (мы все оказывали техническую поддержку всем сотрудникам, но предполагалось, что за каждым системным администратором закреплен конкретный контингент пользователей). Мы должны были встречаться с представителем группы клиентов и начальником отдела, где эта группа работала.

Была нелегко «втиснуть» эти совещания в расписание начальников отделов, но те в конце концов сдавались, если получали обещание, что совещание начнется вовремя и продлится не дольше 15 минут.

Начальники отделов считали крайне необычной ситуацию, в которой они общались непосредственно с системными администраторами, а не «играли в кошки-мышки». Они часто просили продлить совещание или в течение 15 минут ставили задачи, чтобы потом представитель клиентов и системный администратор обсуждали решение этих задач друг с другом.

Некоторые начальники отделов сопротивлялись, заявляя, что делегировали все эти вопросы представителям клиентов. Однако мы обнаружили, что совещания без представителя администрации просто неэффективны. В конце концов мы добились регулярных совещаний с каждым представителем группы клиентов в присутствии начальника отдела, потому что наше умение слушать и наши последующие действия демонстрировали ценность подобных встреч.

Удаление старых процедур

Иногда приходится обновлять процедуры.

В истории с заправкой автомобиля, рассказанной ранее в этой главе, я отметил, что в какой-то момент забыл, с чего началась эта процедура, но продолжал выполнять ее. Это вызывает некоторую тревогу. Правильно ли придерживаться процедуры, если не помнишь ее происхождение?

Я считаю, что надо доверять себе. Раз процедура установлена, значит, я взвесил все «за» и «против», в том числе и этические. В конце концов, речь идет о замене магнитных лент и заправке автомобиля, а не о жизненно важных вопросах.

Я заметил, что морально устаревшие процедуры исчезают сами собой. Когда я получил повышение, и отвечать за смену лент стал другой сотрудник, надобность в процедуре отпала естественным образом.

Процедуры способны самостоятельно модифицироваться и эволюционировать. Это вам не сценарии на языке Perl, которые, если их не отредактировать, перестают работать после переноса файлов на другой сервер. Процедуру выполняете вы. Вы человек, и всегда можете адаптироваться к новой ситуации.

Конечно, я стараюсь вести себя гибко. Если кто-то критикует мою приверженность той или иной процедуре, я внимательно выслушиваю его аргументы. Иногда оказывается, что мой собеседник прав.

Резюме

- Хорошая процедура позволяет вам экономить усилия и сокращает время на принятие решений.
- Процедуры дают вам возможность «один раз подумать и много раз сделать».
- Разработайте процедуру, согласно которой вы будете всегда записывать назначенные встречи и необходимые дела в органайзер и повсюду носить его с собой.

Чем больше процедур вы разработаете, тем меньше умственной энергии вы будете расходовать на решение незначительных вопросов и тем лучше вы сможете сосредоточиться на творческой стороне вашей профессии. Просмотрите свой рабочий день на предмет возможности создания собственных процедур. Указателями на такие возможности являются:

- Повторяющиеся события, не включенные в расписание.
- Задачи технического обслуживания.
- Поддержка взаимоотношений и рабочей обстановки.
- Ситуации, в которых размышления о каком-то деле отнимают больше времени, чем оно само.

- Вещи, которые вы часто забываете.
- Одноразовые или низкоприоритетные задачи, которые нельзя все время игнорировать.
- Повышение профессионального уровня.
- Информированность о новинках.

Запланируйте время на чтение

Вы не успеваете прочитать все компьютерные журналы, на которые подписались? Запланируйте один час на чтение каждую неделю. Найдите укромное место и читайте, сколько сможете. Выбрасывайте то, что не удалось прочитать; интересуйтесь только самой свежей информацией.

4

Система Цикл

В 1997 году я получил награду за свою общественную деятельность. В дополнение к работе системного администратора и активному участию в общественной жизни я много времени уделял четырем некоммерческим организациям, в одной из которых я был президентом, а в другой - основателем. Меня тогда спросили, как я везде успеваю. Я улыбнулся, поблагодарил за комплимент и удержался от того, чтобы сказать: «Я системный администратор! Я борюсь с хаосом, чтобы жить!»

В действительности я еще задолго до того события выяснил, как справляться с потоком запросов и делать все дела, ничего не забывая. Легко выглядеть уверенным в себе, владея тем, что в спорте называется завершающим броском.

Ваши клиенты ценят вашу способность доводить дело до конца выше, чем любые другие ваши качества. Ничто так не подрывает вашу репутацию, как невыполненное обещание. Секрет владения завершающим броском состоит в том, чтобы регистрировать все запросы и сопровождать каждый вплоть до полного завершения. Моя система заключается в методе, который я называю *Цикл*, поскольку он повторяется ежедневно, и выходная информация одного дня является входной для следующего. Помните, на школьных уроках биологии вы рисовали круговую диаграмму: из икринки появляется головастик, у него вырастают лапки, он превращается в лягушонка, а затем во взрослую лягушку, которая откладывает икринки, и все начинается сначала? Моя система - примерно то же самое, разве что каждый цикл длится 24 часа, и мне не приходится жить в пруду.

Для Цикла требуются три инструмента: список дел, скомбинированный с расписанием на текущий день, календарь и список долгосрочных жизненных целей. Храните все эти инструменты в одном месте. Сам процесс не зависит от того, используете ли вы карманный компьютер или бумажный органайзер, купленный в канцелярском магазине.

Важно держать эти три инструмента в одном месте, потому что:

- Все три базы данных взаимосвязаны. У вас должна быть возможность легко переключаться с одной на другую.
- Всегда легче отыскать один предмет, чем три.
- Вам необходимо всегда держать эти базы данных при себе, а с одним пакетом это проще, чем с тремя отдельными предметами.

В этой главе Цикл описан в общих чертах, а главы 5, 6 и 7 посвящены его составляющим: спискам дел и расписанию, календарю и жизненным целям. Возможно, это будут самые важные главы из того, что вы прочитали за всю свою бытность системным администратором.



Не доверяйте своему мозгу

Системные администраторы, вообще говоря, умны. Вы - умный. Я - умный. Мы все умные. Мы достигли нашего положения, работая головой, а не мускулами. Конечно, крепкие мускулы в нашей работе не мешают, но по сути это умственный труд. У среднего человека емкость кратковременной памяти составляет семь единиц (плюс/минус две). А как насчет среднего читателя *этой* книги? Держу пари, у вас она равна восьми или девяти, а у кого-нибудь и десяти (плюс/минус три).

Взглянув на свой список дел, я вижу 20 пунктов. Черт. Это намного больше десяти.

Я никоим образом не могу надеяться, что мой мозг удержит 20 пунктов в памяти. Мне нужна внешняя память. Вам тоже.

Надеюсь, вы не очень обиделись, когда я сказал: «Не доверяйте своему мозгу».

Я своему не доверяю. Поэтому я всегда записываю все просьбы. У меня даже есть такое заклинание:

Всегда записывать все просьбы.

Мой мозг обижается на такое недоверие. Когда кто-то обращается ко мне с просьбой, мой мозг кричит: «Я запомню! Том, убери свой орга-

найзер! Доверься мне на этот раз!» Однако, чтобы не послушаться его, мне достаточно вспомнить, как я, столкнувшись с клиентом, чью просьбу не выполнил, пытался оправдаться фразой «Я забыл».

В главе 2 я описал методику «Делегируйте, регистрируйте или действуйте». Когда мы делегируем задание, у нас нет необходимости его записывать, хотя иногда полезно сделать запись, чтобы проследить за тем, как оно будет выполнено. (Став, так сказать, сторожем своему брату.)

Кроме того, если мы собираемся *выполнить* задание тут же, записывать его тоже не нужно. Когда меня просят «Передайте соль, пожалуйста», я не записываю эту просьбу в список дел, чтобы тут же вычеркнуть ее. Это было бы глупо. Однако если ко мне обращаются с просьбой, а я отвечаю: «Конечно, вот только закончу с этим делом», я записываю просьбу. Не следует путать «когда я закончу» и «прямо сейчас». Я, например, испытываю огромное искушение не записывать просьбу в органайзер, потому что надеюсь не забыть о ней, когда справлюсь с текущей работой.

Бедные наши мозги! Они так страдают из-за предположения, что именно в мозгу находится наше эго. Иногда оно переходит все границы и перехваливает своего приятеля - мозг. Когда вы думаете «Мне не нужно это записывать» или «На этот раз можно сделать исключение; я ничего не забуду», знайте, это ваш мозг - эго размером с Монтану - обещает вам невозможное, подобно сотруднику отдела продаж Microsoft, старающемуся отработать свой оклад.

Я долго думал, что мозг - самая важная часть моего организма.
Потом я понял, кто внушает мне эту мысль.

— Эмо Филипс (Emo Philips)

Если это поможет вашему мозгу справиться с обидой, вспомните, что освобождая его от запоминания списка дел, вы сберегаете энергию мозга для более важных задач. В главе 1 я упомянул исторический анекдот об Альберте Эйнштейне, который пытался сохранить, насколько это возможно, максимум умственной энергии для физики, избавляя свой мозг от всяческих посторонних размышлений, к примеру о том, что сегодня надеть. Рассказывают также, что Эйнштейн не запоминал адреса и номера телефонов, даже собственные. Наиболее важные он записывал на листочке, который хранил в бумажнике, чтобы не загромождать свою драгоценную память. Когда кто-нибудь спрашивал его номер телефона, Эйнштейн отвечал, что номер есть в телефонном справочнике, и учтиво предлагал поискать его там. Берите пример с Эйнштейна; берегите умственную энергию для системного администрирования.

Если у меня нет при себе органайзера, а кто-то обращается ко мне за помощью (обычно это случается, когда я иду в туалет), я решительно перекалдываю ответственность за регистрацию просьбы на клиента. Например, я говорю: «Я спешу на собрание, и мне очень не хотелось бы забыть о Вашей просьбе. Обещайте, что вы отправите электронное

сообщение на адрес *help* [которое автоматически создаст запись в системе регистрации запросов] и опишете в нем свою проблему». Я считаю, что ответственность за запоминание просьбы должна быть возложена на мой органайзер или на клиента. Но только не на мой мозг.

Я не доверяю мозгу запоминание подробностей. Зато бумаге я доверяю. Если на листе бумаги имеется список дел из десяти пунктов, я не беспокоюсь, что один из них пропадет. Исчезающие чернила существуют только в фильмах про шпионов, и не было случая, чтобы мой блокнот сгрызла собака.

Я также доверяю портативным компьютерам (PDA). Конечно, я допускаю, что карманный компьютер может сломаться или данные в нем каким-то образом потеряются, но он синхронизирован с файл-сервером, где хранятся резервные копии. Учитывая, сколько раз мой мозг забывал информацию, можно сказать, что карманные компьютеры почти так же надежны, как бумага.

Идеальная PDA-среда

Когда карманные компьютеры PDA (Personal Digital Assistant - персональный цифровой помощник) еще только появились и моделей было мало, я работал в среде, стандартизированной для одной конкретной модели. Команда системных администраторов настраивала PDA так, что он синхронизировался с домашним каталогом пользователя на файл-сервере. Таким образом, для этих пользователей регулярно выполнялось резервное копирование.

На случай поломки имелся запасной PDA. Чтобы он тут же включился в работу, достаточно было синхронизировать его с данными пользователя. Поскольку PDA у всех были одинаковые, пользователь получал резервный, пока мы устраняли неисправность.

Это было роскошно. Сегодня разнообразие моделей карманных компьютеров достаточно велико, и предоставить такой сервис затруднительно, но с некоторыми оговорками все же возможно.

Почему другие системы не работают

Прежде чем я расскажу о системе Цикл, мне хотелось бы остановиться на других методах, которые широко используются системными администраторами, но результатов не дают. Это Система Беспорядочных Записей и Бесконечный Роковой Список.

Система Беспорядочных Записей заключается в том, что ее приверженец делает записи на случайных обрывках бумаги или ведет несколько списков дел и разбрасывает их, где попало. Меня особенно умиляет вид монитора, обклеенного желтыми прямоугольными листками. Что

на этих самоклеющихся бумажках? Списки дел? Напоминания о встречах? Телефонные номера? Никто не знает. Каковы их приоритеты? А вдруг какая-нибудь отклеится? Полный хаос.

Когда начальство на совещании ставит перед вами задачи, вы начинаете новый список. Теперь у вас их, как минимум, два. Потом вы один список теряете, например, он отправляется в корзину вместе с другими ненужными бумагами. После этого вы пропускаете встречи и нарушаете сроки. Ситуация не из приятных.

Другая крайность - Бесконечный Роковой Список. Обычно человек приходит к выводу, что много списков или разбросанные бумажки с записями не позволяют рационально вести дела. Он покупает блокнот и объявляет, что это его единственный список дел. Больше не будет пуганицы, не так ли? Он усердно носит блокнот с собой, записывает в него все новые дела, а завершенные дела вычеркивает. Поначалу все идет прекрасно, но затем процесс начинает давать сбои. Трудно расставлять приоритеты. Старые пункты забываются, потому что глаз к ним привывает и реагирует только на новые.

Самый серьезный недостаток этой системы, из-за которого я и называю ее Роковым Списком, состоит в том, что она порождает депрессию. Список никогда не кончается. Вы работаете и работаете, а он не становится короче! Вы вычеркиваете выполненные дела, а в конце списка появляются новые. Количество страниц начинает понемногу уменьшаться по мере того, как вы вычеркиваете пункты в середине списка, но где-то в начале стоит маленький пунктик, который, похоже, никогда не будет вычеркнут. Вы перелистываете страницы с вычеркнутыми пунктами в поисках того невычеркнутого. Вы испытываете стресс, поскольку боитесь пропустить неслучайное дело, затерявшееся среди сделанных.

Хуже всего то, что такой метод убивает ваше самоуважение. Вы не испытываете удовлетворения от того, что выполнили все дела в списке, потому что это невозможно в принципе. Это Роковой Список.

Ньюмен: Я - американский почтальон.

Джордж: Вы один из этих парней, которые постоянно сходят с ума, хватают ружье и палят в окружающих?

Ньюмен: Бывает.

Джордж: Но почему?

Ньюмен: Потому что поток писем никогда не прекращается. Они приходят, и приходят, и приходят, и нет просвета. Это бесчеловечно. Каждый день гора писем становится все выше! И ты должен их разносить, но чем больше ты их разносишь, тем больше их приходит. А потом ломается сканер штрих-кода, и это в день рассылки Publisher's Clearing House!

— «Сейнфельд», 418-я серия «Старик»

Если Система Беспорядочных Записей слишком хаотична, а Бесконечный Роковой Список порождает депрессию, то система Цикл, как го-

ворится, «в самый раз». В ней используется инструмент (бумажный органайзер или портативный компьютер), который можно носить с собой, что позволяет держать все компоненты вместе. Цикл дает вам ощущение завершенности дел в конце каждого рабочего дня, когда вы видите, что обработали весь список.

Успешные системы

Я уже сказал, как важно доводить любое дело до конца (делать «завершающий бросок») и что не следует полагаться на свой мозг, а также описал свойства провальных подходов к ведению дел. Теперь я объясню, что делает систему успешной.

Хорошая система характеризуется следующими качествами:

- **Портативность.** Вы можете взять ее с собой куда угодно.
- **Надежность.** Она запоминает все, что вам нужно, избавляя вас от этой нагрузки.
- **Управляемые единицы информации.** Не миллион мелких записей, но и не единственный Роковой Список.

Элементы, составляющие хорошую систему:

- **Календарь.** Место, куда вы записываете текущие встречи, совещания, торжественные мероприятия и т. д.
- **Список жизненных целей.** Несколько страниц для ваших долгосрочных планов и других заметок.
- **Ежедневник.** На каждый день мы имеем:
 - **Список дел.** Список дел только на этот день с расставленными приоритетами.
 - **Расписание.** Почасовое расписание на этот день.

Главный элемент системы - страничка, посвященная сегодняшнему дню. Она должна быть достаточного размера, чтобы на ней помещались список дел на этот день и расписание. Органайзеры с такими страничками производят фирмы FranklinCovey и Filofax (рис. 4.1). Альтернативой может быть PDA-компьютер (наладонник). Органайзер необходимо всегда иметь при себе, чтобы, если кто-то обратится к нам с какой-то просьбой, сразу записать ее в органайзер, а не на клочок бумаги, который потеряется раньше, чем вы перепишете его содержимое в более надежное место.

Цикл

Цикл является развитием системы, которую я успешно применял в течение десяти лет. Этот метод прост в применении и в то же время включает в себя все компоненты, необходимые системному администратору.

- **Расписание на сегодня.** Каждый день распланирован с точностью до часа.
- **Календарь встреч.** Эта часть предназначена для записи всех наших встреч, совещаний, общественных обязанностей и т. п. Событие, запланированное далее, чем на текущий месяц, записывается в этот календарь, а потом переносится на страничку с соответствующей датой.
- **Заметки.** Мы будем использовать органайзер и для хранения других заметок и списков. Например, в главе 7 описано, как составлять краткосрочные и долгосрочные планы.

Система Цикл работает следующим образом: каждый день вы тратите 10 минут на планирование дня. Процесс планирования весьма прост.

- **Составить расписание на сегодня.** В расписании на текущий день я резервирую время на все ранее намеченные встречи. Они уже записаны в моем календаре. (Подробности я привожу в главе 6.) Я подсчитываю, сколько времени осталось. Это мои рабочие часы.
- **Составить список дел на сегодня.** В этот список дел я заношу все, что должно быть сделано сегодня. Элементы списка берутся из телефонных звонков, совещаний, моего календаря, записей системы регистрации клиентских запросов и вчерашнего списка.
- **Расставить приоритеты и откорректировать расписание.** По каждому пункту я оцениваю время, необходимое для его выполнения. Затем складываю эти оценки. Если суммарное время превышает время, отведенное сегодня на работу, я переношу некоторые дела в завтрашний список. О критериях отбора дел для переноса на завтра мы поговорим позже.
- **Выполнить план.** Весь день я выполняю дела, перечисленные в списке, и участвую в совещаниях. Я все время сосредоточен на работе. Если какой-то пункт выполнен, я отмечаю его крестиком.
- **Закончить день.** В конце дня я переношу незаконченные дела в завтрашний список. Перенесенные пункты я отмечаю черточкой.
- **Покинуть офис.** Теперь я могу уходить из офиса. Я счастлив от того, что обработал каждый пункт: он либо выполнен, либо перенесен на следующий день. Ничто не забыто.
- **Повторить.** На следующий день Цикл повторяется. Список дел на этот день уже частично состоит из дел, перенесенных с предыдущих дней.

Имея каждый день новый список, мы поддерживаем в себе ощущение завершенности, когда в конце дня видим, что каждый пункт обработан. Если нам удастся выполнить все дела до конца рабочего дня, мы можем наградить себя, поработав над «приятным» проектом или уйдя домой пораньше, если у нас гибкий график. Если работы столько, что за сегодня все не переделать, настроение все же остается хорошим, потому что переполнение под нашим контролем.

Мы можем строить долгосрочные и среднесрочные планы, а не беспомощно барахтаться, пытаясь удержаться на плаву. Мы можем разбить задачу на отдельные части и запланировать выполнение каждой подзадачи на какой-то конкретный день. Мы можем достичь отдаленной цели, составив расписание на следующий месяц (и даже год) и записав этапы на разных страницах.

А еще у нас есть календарь, помогающий отследить встречи. Пользуйтесь одним календарем для деловых и личных встреч, так как управиться с одним календарем легче, чем с двумя. Благодаря комбинированному календарю вы не пропустите развлечение только потому, что забыли посмотреть в другой календарь и решили задержаться на работе.

Такой подход кажется вам слишком механическим? Слишком негибким? Вы скоро убедитесь, насколько он гибок. Весь процесс планирования отнимает не больше десяти минут ежедневно, зато избавляет вас от многих часов стресса. Разве спланировать один день нереально? Что делать, если новые дела добавляются в список в течение дня? Я обещаю ответить на этот вопрос в главе 5. Прежде чем учиться ходить, вам нужно научиться ползать.

Резюме

- Необходимо уметь регистрировать все клиентские запросы и доводить их выполнение до конца (или отклонять). Клиенты (те, кого вы обслуживаете) и менеджеры (те, кто решает, когда повышать вам зарплату) ценят это умение, потому что хотят, чтобы их запросы и проекты выполнялись, а не «спускались на тормозах».
- Хороший «завершающий бросок» - ключ к повышению по службе.
- Не дайте запросу умереть в зародыше - регистрируйте их все без исключения. Если клиент встретит вас в коридоре и попросит о чем-то, не полагайтесь на свою память. Если вы не можете тут же записать просьбу, предложите клиенту отправить ее вам по электронной почте или в систему регистрации клиентских запросов. Так вы переложите ответственность за регистрацию просьбы на клиента, не рискуя забыть о ней.
- Ничто так не оскорбляет, не злит и не разочаровывает клиента, как обещание, данное системным администратором и потом забытое.
- Чем больше дел вы должны сделать, тем труднее вам организовать свою работу. Начиная с какого-то момента вы будете тратить больше времени на организацию работы, чем на ее выполнение.
- Чтобы не забывать запросы, регистрируйте их надежным способом. Человеческий мозг не так надежен, как бумага или электронные устройства. Записывайте запросы в тот момент, когда получаете их. Всегда записывайте все запросы. Берегите свой мозг для более важных задач.

- Системы со списками дел не работают по ряду причин. Беспорядочно разбросанные записи теряются. Единый список дел превращается в угнетающий Бесконечный Роковой Список. Обе системы отрицательно влияют на самооценку.
- В системе Цикл используются календарь встреч и важных дат, список долгосрочных целей, список дел на сегодня (и на каждый день) и расписание на сегодня, позволяющее планировать работу.
- Каждый день начинается с десятиминутного планирования. Изучите календарь, чтобы понять, сколько времени у вас уйдет на встречи и совещания. Остальное время вы можете посвятить делам из сегодняшнего списка. Вы определяете, достаточно ли у вас времени для выполнения всех дел из списка, и, если потребуется, переносите низкоприоритетные дела на другие дни или ведете переговоры с клиентами о переносе сроков.

Начните прямо сейчас!

Когда вы начнете придерживаться системы Цикл, она покажется вам неудобной. Однако со временем вы почувствуете себя более комфортно. Вы приспособите ее под свой стиль, когда лучше в ней разберетесь. Психологи утверждают, что для формирования привычки требуется 21 день. В течение 21 дня упорно делайте одно и то же, чтобы ваш мозг стал относиться к этому делу как к привычке, не требующей особых усилий. Мое правило "список дел на каждый день" не станет вашей привычкой, пока вы не начнете его соблюдать. Запланируйте к выполнению вашу новую привычку на сегодня. Оставшиеся 20 дней вам будет гораздо проще.

5

Система Цикл: списки дел и расписания

Теперь, раздражив вас кратким описанием системы Цикл в главе 4, в течение трех глав подряд я буду раскрывать ее элементы, от каждодневных до долгосрочных. Эта глава посвящена управлению списком дел, то есть сегодняшнему дню. В следующей главе обсуждаются календари, позволяющие организовать время на ближайшие дни и месяцы. Наконец, в главе 7 мы поговорим о постановке долгосрочных жизненных целей. Поскольку Цикл является повторяющимся процессом, я промолчу о моментах, обсуждать которые бессмысленно, пока вы не прочли следующие главы. Не исключено, что вы захотите прочесть эти три главы несколько раз подряд.

У всех системных администраторов есть одна общая черта - рабочего времени не хватает для выполнения всей работы. К счастью, большая часть этой главы посвящена собственно управлению потоком дел. Начав с одного типичного дня, мы рассмотрим последующие, чтобы понять, как работает система.



Типичный день

Рассмотрим работу Цикла на примере одного дня.

Входя утром в офис, вы должны сразу сосредоточиться и запустить этот процесс. В противном случае вас захватят прерывания и отвлекающие факторы, которых полно вокруг: на автоответчике мигает индикатор новых сообщений, в коридоре подходят сотрудники, зовет к себе автомат с кофе, к тому же вам интересно, какой комикс появился сегодня на сайте User Friendly. Вы решаете проверить электронную почту и... через несколько часов осознаете, что половина дня пролетела впустую.

Итак, **ОСТАНОВИТЕСЬ**. Не проверяйте электронную почту и не читайте новости на сайтах. Вместо этого закройте дверь (если вам повезло с отдельным кабинетом) и выполните все шаги Цикла.

Вначале составьте план

«А можно я сперва проверю электронную почту?»

Нет. Планирование займет 10 минут. Почта может подождать.

«А вдруг произошел сбой, и мне сообщили о нем по электронной почте?»

Незначительные сбои могут подождать 10 минут. О серьезных проблемах вы обычно узнаете не по электронной почте, а, например, по запаху дыма или шуму за дверью.

Вот компромисс: выведите на экран «приборную панель» приложения, которым вы пользуетесь для мониторинга сети. Если вы видите, что все службы работают нормально, значит, проверять почту не нужно. (А разве программа мониторинга уже не известила бы вас о сбое, если он произошел?)

Друзья говорили мне, что у них достаточно силы воли, чтобы открыть почтовое приложение, проверить, нет ли важных сообщений, и затем закрыть его. У меня нет такой силы воли. Если я начну проверять почту на предмет важных сообщений, дело, как всегда, закончится тем, что я прочитаю всю почту, затем начну реагировать на запросы и вдруг вспомню, что так и не распланировал рабочий день. Поверьте мне, аварии и сбои могут подождать десять минут.

Шаг 1: составьте расписание на сегодня

Начните день с составления расписания. Посмотрите в календаре, какие совещания и встречи назначены на текущий день, и впишите их в соответствующие графы на страничке органайзера. Остальное время

можно уделить пунктам из списка дел. Несложные арифметические подсчеты покажут, сколько времени у вас в распоряжении.

Предположим, листок вашего календаря выглядит, как показано на рис. 5.1.

19	20	21
	<i>10-11 Собрание</i> <i>15-16 Джон</i>	
26	27	28

Рис. 5.1. Записи в календаре

У вас запланированы совещания на 10:00 и 15:00, каждое по часу. Вы отмечаете эти интервалы на страничке с расписанием. Вам нужен перерыв на обед, и его вы планируете с 12 до 13. Затем вы подсчитываете, сколько времени остается на дела из списка. Сейчас 8:30, и вы хотите уйти в 17:30, то есть через 9 часов. Три часа уже заняты, на другие задачи остается шесть часов. На рис. 5.2 показано, как выглядит ваше расписание в данный момент.

Шаг 2: составьте список дел на сегодня

Теперь вы составляете список сегодняшних дел и подсчитываете, сколько у вас работы (в часах). Как правило, на этот день уже запланированы какие-то дела. Вы добавляете то, что пришло вам в голову (например, гениальную мысль, осенившую вас, пока вы шли от машины до офиса) и, если сегодня первый день месяца, сверяетесь со списком долгосрочных целей (подробнее об этом сказано в главе 7).

Поскольку вы работаете с системой Цикл первый день, ваш список дел изначально пуст. Однако вы представляете себе, что должны сделать. Запишите эти дела вкратце; необязательно писать законченные предложения. После этого список может выглядеть примерно так, как показано на рис. 5.3.

Как видите, краткой записи достаточно, чтобы вы поняли задачу. Можете записать какие-то детали, например номера телефонов, логины и т. д., если вам необходимо, но в целом старайтесь быть лаконичным.

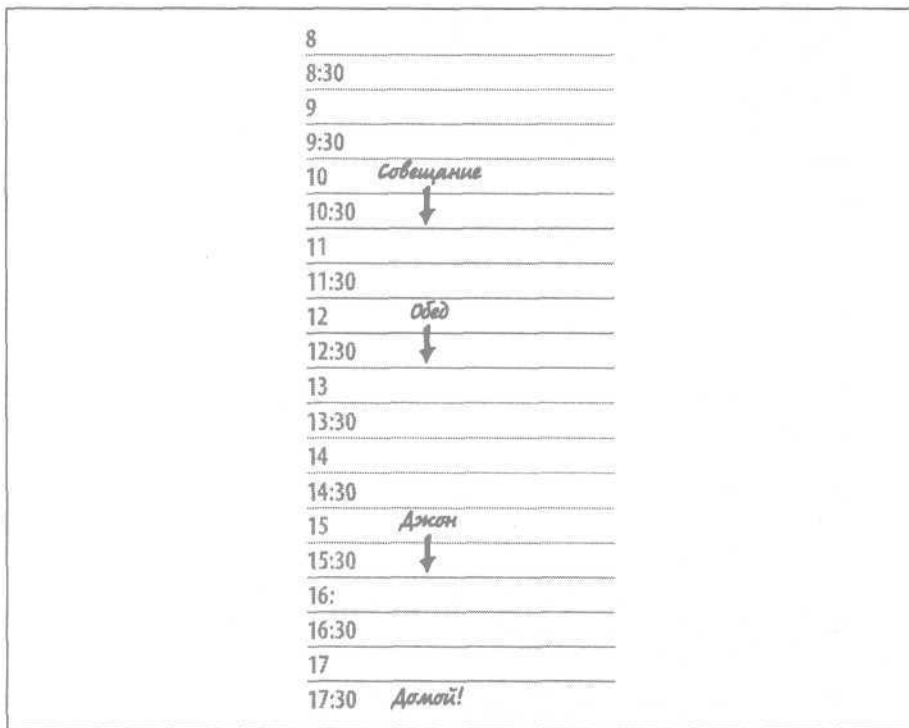


Рис. 5.2. День с размеченными интервалами

Если у вас есть свежие сообщения на автоответчике, сейчас самое время прослушать их и, возможно, что-то записать. Я имею привычку записывать на бумаге все сообщения автоответчика, даже если они не

Краткая запись	Что это означает
<i>Создать для Боба Смита, влчмй, 555-4321</i>	Создать учетную запись для нового пользователя Боба Смита (тел. 555-4321). Здесь следует указать подробности, чтобы не забыть.
<i>Модернизировать GCC</i>	Установить новую версию пакета GCC.
<i>Ошибки в Netscan</i>	Сообщить разработчикам еще об одной ошибке, найденной в программе Netscan.
<i>Демо-версия IDS</i>	Позвонить торговому представителю насчет демо-версии новой системы IDS.
<i>Компьютер для Боба</i>	Установить новый компьютер для Боба.
<i>Библиотека линк</i>	Установить новую библиотеку для новой системы создания резервных копий.
<i>Исследование программы мониторинга</i>	Исследовать новую программу мониторинга, чтобы заменить ею программу, используемую сейчас.

Рис. 5.3. Понедельник. Список дел в вашем органайзере

Сделано?	Приоритет	Время	Дело
		15	Создать для Боба Смитта, <i>блтиш</i> , 555-4321
		7 ч.	Модернизировать ГСС
		30	Ошибка в Netscape
		15	Демо-версия ГДС
		3 ч.	Компьютер для Боба
		7 ч.	Библиотека ленин
		1 ч.	Исследование программы мониторинга

Рис. 5.4. Понедельник. Добавлена оценка времени

требуют каких-то действий с моей стороны. В таком случае я могу тут же пометить этот пункт как выполненный. Это придаст мне ощущение завершенности моей работы.

Итак, сколько работы у вас на сегодня? Запишите рядом с каждым пунктом списка дел примерное время, за которое сможете его выполнить (рис. 5.4).

В этом примере вы можете прикинуть, сколько времени уйдет на все дела. Сложите часы (помечены как «ч»), прибавьте к ним остальные значения (минуты), и вы получите девять часов.

Шаг 3: расставьте приоритеты и откорректируйте план

Затем установите приоритетность дел, основываясь на какой-нибудь простой системе. Я пытался использовать 100-балльную шкалу, но она оказалась слишком сложной. Один друг заметил как-то, что в жизни есть только три уровня приоритетности:

- Срок исполнения сегодня, и дело необходимо сделать прямо сейчас.
- Срок исполнения наступит скоро.
- Все остальное.

Для простоты обозначим эти приоритеты А, В и С соответственно. На рис. 5.5 показано, как я расставил приоритеты в нашем примере.

Как справиться с переизбытком дел

Итак, для выполнения всего списка вам потребуется девять часов, а в вашем распоряжении только шесть. Как справиться с переизбытком дел?

Задержаться на работе - неверное решение. Ваша личная жизнь достаточно ценна. Вашему работодателю вовсе не нужно, чтобы из-за нехватки свободного времени вы стали раздражительны. Правильно пи-

Сделано?	Приоритет	Время	Дело
	A	15	Создать для Боба Смига, вайфай, 555-4321
	B	7 ч.	Модернизировать ГСС
	A	30	Ошибка в MySQL
	A	15	Демо-версия IDS
	A	3 ч.	Компьютер для Боба
	A	7 ч.	Библиотека лент
	C	1 ч.	Исследование программы логинирования

Рис. 5.5. Понедельник. Добавлены приоритеты

таясь, высыпаясь, занимаясь спортом и ведя активную жизнь вне работы, вы лучше работаете.

Простейший способ решить проблему - перенести на следующий день дела с приоритетом С и некоторые из дел с приоритетом В. Одно из преимуществ ежедневного списка дел заключается в том, что мы можем переносить пункты. Вот несколько полезных идей:

- **Перенести на следующий день пункты с самым низким приоритетом.** Для меня это самое обычное решение. Я потому и расставляю приоритеты, что невозможно сделать все и сразу. Следовательно, придется переносить на завтра несколько задач с приоритетами С и В.
- **Выделить сегодняшнюю подзадачу.** Выделите в задаче решаемую подзадачу, а остальное перенесите на завтра. Например, установка новой системы резервного копирования включает в себя много шагов. Сегодня вы можете ее распаковать и убедиться, что все ее части на месте, а кабели достаточно длинны. Завтра вы сможете найти добровольного помощника и установить тяжелое оборудование. На следующий день вы сконфигурируете драйверы. Вы каждый день будете понемногу продвигаться и выполните задание в срок. Разбив его на этапы, запишите каждый этап в отдельный список дел на день. Это подходящий метод для задач, выполнение которых может неожиданно застопориться. Например, первый этап следует выполнить немедленно. Если вы обнаружите нехватку каких-либо частей, у вас будет время добиться их поставки. Об отсутствии кабеля лучше узнать сейчас, чем в последний день.
- **Сократить задачу (уменьшить время на ее выполнение).** Иногда удается найти способ сокращения времени, отпущенного на выполнение задачи. Например, устанавливая компьютер для Боба, вы обнаружили, что он не прочь поболтать и что работа идет вдвое быстрее, если его нет поблизости. Или, наоборот, Боб технически достаточно

грамотен, чтобы самостоятельно установить на компьютере дополнительное программное обеспечение. Возможно, он даже не прочь сделать все сам, если вы разрешите. Если ему потребуются пакеты, он попросит у вас диски, или вы можете сказать ему, что именно не установлено, пообещав зайти позже и все доустановить. Боб сможет немедленно приступить к работе за компьютером, а вы - перейти к другим делам. И, конечно, чтобы не забыть, включите в завтрашний список дел пункт «Закончить установку ПО для Боба».

- **Изменить оценку трудоемкости.** Оценивая время, необходимое для выполнения задачи, нужно всегда преувеличивать ее трудоемкость. Так безопаснее. Однако если вы переусердствовали в этом, то обнаружите, что задачу вполне можно выполнить и за меньшее время.
- **Делегировать.** Иногда удается перепоручить работу другому сотруднику. Начинающие сисадмины с нетерпением ждут сложных заданий, чтобы приобрести новые навыки. Конечно, у вас не всегда есть полномочия делегировать работу, поэтому читайте следующий совет.
- **Попросить шефа помочь вам с расстановкой приоритетов.** Составив список дел и определив приоритеты и примерное время, можно обратиться к шефу за помощью в расстановке приоритетов. Если вы никогда этого не делали, то, возможно, сочтете этот совет чересчур фантастичным, но в действительности менеджерам часто кажется, что они слабо влияют на работу подчиненных, и вопрос «Правильно ли я расставил приоритеты?» заставляет их приободриться. (Конечно, задавая этот вопрос каждый день, вы заработаете жирный минус в графе «Умение работать самостоятельно» на ежегодной аттестации.) В моей практике был случай, когда мой шеф очень четко расставил приоритеты в моей работе, и это помогло мне лучше выполнять свои обязанности в целом. В другой раз шеф обнаружил в моем списке дела, переданные другому отделу, и вычеркнул их (со словами: «Джо Шмоу должен научиться делать это самостоятельно. Я поговорю с его начальником».) Иногда с меня снимали целые категории обязанностей («Передайте Джо, что мы больше этим не занимаемся, а если у него возникнут проблемы, пусть выходит на меня!»), а иногда шеф перепоручал мои задачи другим. Я вижу, что многие системные администраторы и не догадываются о такой возможности, а ведь она при разумном подходе является мощнейшим инструментом тайм-менеджмента.
- **Отложить встречу.** Это, конечно, может быть плохим решением. Перекраивание расписания - сплошной кошмар, многих оно раздражает и может сорвать сроки проекта. Как вариант вы можете не явиться на встречу или послать своего представителя. Если намечается двухчасовое собрание, а по поводу нового сервера от вас требуется только сказать, что он поставляется в синем корпусе, а не в красном, то отправьте кого-нибудь вместо себя. (А когда начнут обсуждать серьезные вопросы, ваш представитель сможет вызвать вас на собра-

ние по телефону.) Отложить встречу лучше, чем не явиться на нее. Я заметил: когда я договариваюсь о переносе встречи лично или по телефону (поскольку электронная почта недостаточно интерактивна), собеседник часто готов сократить встречу (и обсуждать лишь самые важные вопросы). Например, выясняется, что его интересовал лишь цвет корпуса сервера. Так вот, сервер будет синий.

- Задержаться на работе. Я привожу этот вариант исключительно для полноты картины. Это наихудшее решение. Большинство людей способно продуктивно работать лишь четыре-пять часов в день. Все остальное - бег на месте. Поэтому такие книги, как «Extreme Programming» (Экстремальное программирование), O'Reilly, или «PeopleWare» (Человеческий фактор), Dorset House, рекомендуют исключить сверхурочную работу. Конечно, системному администратору иногда приходится работать допоздна. Как сказано в разделе «Делегируйте, регистрируйте или действуйте» главы 2, если случилась серьезная авария, клиенты надеются, что на ликвидацию последствий брошены все силы, и вы работаете, пока проблема не будет решена.

Вернемся к нашему примеру со списком дел.

Попробуем перенести низкоприоритетные дела на следующий день. У нас есть дело с приоритетом С — «Исследование программы мониторинга». Его и перенесем.

Если вы пользуетесь портативным компьютером, перенести запись в другой список не составляет труда. Если у вас бумажный органайзер, пометьте пункт списка черточкой, чтобы обозначить, что он перенесен, и впишите его в завтрашний список.

Вы уменьшили объем работы на час. Вам остается разобраться еще с двумя часами.

Сделано?	Приоритет	Время	Дело
	A	15	Создать для Боба Смига, встий, 555-4321
—	B	7 ч.	Модернизировать ГСС
	A	30	Ошибка в Netscape
	A	15	Демо-версия IDS
	A	3 ч.	Компьютер для Боба
	A	7 ч.	Библиотека лент
—	C	1 ч.	Исследование программы мониторинга

Рис. 5.6. Полностью загруженный понедельник. Часть дел перенесена на вторник

К счастью, имеется дело с приоритетом В (модернизация ГСС), которое тоже можно перенести. Повторите ту же операцию (переместите запись в портативном компьютере или поставьте черточку и перепишите на другую страницу в органайзере). Теперь список выглядит, как показано на рис. 5.6, и соответствует количеству часов, имеющих-ся в вашем распоряжении.

Работа над долгосрочными проектами

Как вы работаете над длительными проектами? Как определить сегодняшние дела, если список дел растянут на полгода?

Важно разбить большой проект на отдельные этапы. У очень больших проектов обычно есть менеджеры, которые сделают это за вас. В своем проекте вы должны все делать самостоятельно. Найдите время, чтобы разбить проект над подзадачи и оценить трудоемкость каждой. Сделайте запись о каждом этапе на страничке того дня, в который вы собираетесь приступить к данному этапу, или сделайте пометку в календаре, если этот день достаточно далек. Подобный подход к планированию сильно впечатляет менеджеров.

В моем ежедневном списке дел я записываю название проекта и текущий этап. Так я напоминаю себе об отдаленной цели, пока работаю над повседневными задачами. Например: «Реорганизация сети: карта текущей сети».

Шаг 4: выполняйте план

Весь день придерживайтесь плана как можно точнее. Вначале занимайтесь делами с приоритетом А, затем делами с приоритетом В, а если останется время, то делами с приоритетом С.

Полезно иметь какой-нибудь таймер или другой механизм напоминания о намеченных совещаниях и встречах, чтобы не приходилось постоянно отрываться от работы и смотреть на часы.

Закончив одно дело, приступайте к следующему. Не теряйте времени.

Многие жалуются, что тратят массу времени на то, чтобы решить, за какое дело взяться следующим. Простейшее решение заключается в том, чтобы сделать все дела с приоритетом А в том порядке, в каком они идут в списке, затем также поступить с делами с приоритетом В и, наконец, с приоритетом С. Мы улучшим эту систему в главе 8, а пока вы не ошибетесь, если просто будете выполнять дела по порядку. Как правило, первыми в списке указаны дела, перенесенные с предыдущих дней. Иными словами, дела, которые вы отложили, «всплывают» вверх по списку, и их надо делать в первую очередь.

Старайтесь не упускать момент и переходить к следующему делу, как только закончили предыдущее. Поступайте так даже в том случае, когда на работу ушло меньше времени, чем вы предполагали; возможно, так вам удастся предотвратить нехватку времени, если на другое дело его потребуется больше, чем запланировано.

Время от времени делайте паузы для физической разминки. Если вы справились с заданием досрочно, пройдите энергичным шагом по коридорам. Работа за компьютером ограничивает вашу подвижность. Прихватите с собой папку для бумаг, чтобы произвести впечатление, будто вы идете по важному делу. Никто ни о чем не догадается.

Итак, закончив дела с приоритетом А, приступайте к делам с приоритетом В. Разобравшись с ними, побалуйте себя самым интересным делом с приоритетом С.

Шаг 5: закончите рабочий день

Редко удастся справиться со всеми делами из списка, тем не менее все его пункты без исключения следует обработать. Пункт списка *обработан*, если вы уделили ему достаточно внимания.

Последние полчаса рабочего дня посвятите анализу невыполненных пунктов. Если среди них есть дела с приоритетом А, не пускайте ситуацию на самотек. Позвоните сотруднику, который ждет решения своего вопроса, и выработайте совместный план действий. Если же вы сами установили сроки (как обычно и бывает), перенесите пункты списка на следующий день.

Допустим, сегодня вы успели распаковать систему резервного копирования, убедившись в наличии всех нужных кабелей, и даже отрепетировали ее монтаж, чтобы проверить, достаточно ли кабели длинны. Однако вы не нашли никого, кто согласился бы помочь вам смонтировать оборудование. Значит, обработка этого пункта списка состоит в том, чтобы вычеркнуть выполненные подзадачи проекта, а остальные перенести на следующий день.

На портативном компьютере вы просто копируете записи в список на вторник. На страничке бумажного органайзера вы можете рядом с пунктом «Библиотека лент» списка написать «Распаковано и проверено», отметить этот пункт черточкой, а в список на вторник записать «Монтаж системы».

Все несделанные дела с приоритетами В и С тоже надо перенести на другой день. Теперь ваш список выглядит, как показано на рис. 5.7.

Получилось, что на пятичасовую работу ушло шесть часов.

Шаг 6: можно идти домой

На рис. 5.7 вы видите список, в первом столбце которого по каждому пункту есть отметка. Вы обработали все пункты списка. Не все дела за-

Сделано?	Приоритет	Время	Дело
X	A	15	Создание для Боба Смитта, bsmith, 555-4321
—	B	7 ч.	Модернизировать ГСС
X	A	30	Отписка в Миллен
X	A	15	Дано-версия ГСС
X	A	3 ч.	Копирование для Боба
—	A	7 ч.	Библиотека лент распакована и проверена
—	C	1 ч.	Исследование программы моделирования

Рис. 5.7. Список дел на понедельник в конце рабочего дня

кончены, но все обработаны. Обработка пункта подразумевает, что ему уделено достаточно внимания. Для низкоприоритетных пунктов это иногда означает простой перенос на завтра. Важно то, что вы ничего не забыли.

Вы можете взглянуть на список и с удовлетворением отметить для себя, что все его пункты обработаны. Поздравьте себя. Улыбнитесь. Одевайтесь и идите домой в хорошем настроении. Вы это заслужили.

Было время, когда я каждый день уходил с работы в ужасном настроении. Мне казалось, что я все работаю и работаю, но ничего не успеваю. Теперь я использую систему Цикл. Я просматриваю список дел, вижу, что обработал каждый пункт, и у меня возникает ощущение завершенности. Я покидаю офис с улыбкой на лице.

Шаг 7: все сначала

Предположим, сегодня вторник. Вы повторяете Цикл с сегодняшним списком дел.

Сегодня я научу вас работать в чуть более сложной ситуации. В частности, я покажу, как Цикл работает с системой регистрации запросов, с автоответчиком и в условиях постоянных прерываний.

Составьте расписание на сегодня

Вы должны начинать каждый день с проверки календаря на предмет встреч и заносить их в расписание на день. Предположим, на сегодня у вас ничего не назначено, и ваша страничка с расписанием остается пустой, если не считать запись о часовом перерыве на обед. Остается восемь часов на работу - нормальный рабочий день.

Пока все неплохо!

Составьте список дел на сегодня (вторник)

Вчера вы получили четыре новых задачи. Назовем их Задача 1, Задача 2, Задача 3 и Задача 4. Это низкоприоритетные дела, которые были поручены вам на вчерашнем совещании. Они не так важны, как то, чем вы занимались в понедельник, и вы записали их в список сегодняшних дел (на вторник), полагая, что это реалистичный вариант. К тому же в списке есть дела, перенесенные из вчерашнего списка.

Когда вы пришли утром на работу, на автоответчике было три голосовых сообщения. Вы их прослушали и занесли в список, хотя они и не требовали каких-то действий с вашей стороны.

Первое сообщение информировало всех сотрудников компании, что восточный вход в здание фирмы закрыт. Вы бесконечно благодарны руководству компании, которое потратило свое драгоценное время на запись этого сообщения, иначе вы бы ни за что не догадались, что означают объявления «Восточный вход закрыт», развешанные по всем коридорам. Поскольку это сообщение не требует от вас никакой реакции, смело вычеркивайте его.

Следующее сообщение пришло от вашего поставщика из Cisco. Вы записываете его телефон, поскольку собираетесь перезвонить ему (однако, прежде чем вычеркнуть пункт, вы обязательно проверите, совпадает ли этот телефон с тем, что хранится в вашей базе данных, описанной в главе 12). Третье сообщение от другого поставщика. Ему вы перезванивать не собираетесь, так что вы просто пишете в списке «спам» и ставите крестик в графе «Сделано». Теперь ваш список на вторник выглядит, как показано на рис. 5.8.

День еще не начался, а у вас уже два пункта вычеркнуто! Отлично!

Вы можете удивиться: зачем заносить в список пункты, чтобы тут же их вычеркнуть? Таким способом вы ведете журнал телефонных звонков, которые, как говорится, «к делу на пришьешь». Это одна из причин, по которым я предпочитаю бумажный органайзер электронному. Меньше мороки с записью-вычеркиванием.

Я уже несколько раз отмечал, что системному администратору хорошо бы иметь какую-нибудь систему регистрации клиентских запросов. Как включить ее в свой Цикл? Ежедневно выделяйте некоторый отрезок времени специально для обслуживания клиентских запросов. Когда-то я работал в фирме, где должен был треть дня обрабатывать такие запросы. Каждый день я добавлял в список дел двухчасовую работу под названием «Клиентские запросы». Это не означает, что я разбирался с запросами только в течение первых двух часов рабочего дня. Я резервировал время на тот случай, если появится неотложный запрос.

Реагирование на многочисленные прерывания обычно требует часа ежедневно. Эти прерывания составляют важную часть моей работы, поэтому я резервирую время и для них.

Сделано?	Приоритет	Время	Дело
		1 ч.	Исследование программы мочинорича
		7 ч.	Модернизировать ГСС
		1 ч.	Смонтировать библиотеку лини
		1 ч.	Задача 1
		1 ч.	Задача 2
		1 ч.	Задача 3
		1 ч.	Задача 4
X			Мог. Вашицкий вид заявки
		15	М/о: Cisco, Држа, +1 800-555-111
X			М/о.

Рис. 5.8. Вторник. Список дел после прослушивания автоответчика



Если у меня установлена взаимная защита от прерываний, в расписании на время действия защиты я пишу аббревиатуру «ВЗП». Любая работа над проектом, выполняемая в это время, как подарок.

Теперь можно подсчитать, сколько часов потребуется, чтобы переделать все дела. Ни много ни мало 11 с четвертью! При 8-часовом рабочем дне самое время расставить приоритеты. Все пункты, которые нужно выполнить сегодня, автоматически получают приоритет А. Система резервного копирования, которую вы еще не установили, имела приоритет А вчера, значит, получит его и сегодня.

Расставьте приоритеты и откорректируйте план

Какие приоритеты присвоить пунктам «Клиентские запросы» и «Прерывания»? С одной стороны, их необходимо выполнять каждый день, значит, они должны иметь наивысший приоритет. Однако пункт «Прерывания» - это нечто вроде буфера, который нужен на случай появления реальных прерываний. Вы имеете определенную свободу действий и можете пометить этот пункт литерой В.

В списке присутствует несколько дел с приоритетом В и очень много с приоритетом С (что вполне нормально). Сам список выглядит так, как показано на рис. 5.9.

Вчера общее время дел с приоритетом А и В превысило количество ваших рабочих часов, и вам пришлось перенести кое-что на сегодня. Сегодня дела с этими приоритетами потребуют 6 часов 15 минут. Переизбыток дел не наблюдается, и переносить что-то на завтра не нужно.

Сделано?	Приоритет	Время	Дело
	C	1 ч.	Исследование программы мониторинга
	B	7 ч.	Модернизировать ССС
	A	1 ч.	Смонтировать библиотеку ленин
	C	1 ч.	Задача 1
	C	1 ч.	Задача 2
	C	1 ч.	Задача 3
	C	1 ч.	Задача 4
X			А/с: Воспитный фонд-запрос
	A	15	А/с: Cisco, Диска, +1 800-555-111
X			А/с: стиж
	A	7 ч.	Клиентские запросы
	B	1 ч.	Прерывания

Рис. 5.9. Вторник. Список дел после расстановки приоритетов и оценки времени

Я заметил, что если у меня не очень много дел с приоритетом А и я закончу их в первой половине дня, то остаток дня протекает менее напряженно. Я выполняю дела с приоритетом В плюс столько дел с приоритетом С, сколько успею. В конце дня я переношу на завтра все незавершенные дела без каких-либо угрызений совести. Такой подход не создает дополнительных стрессов и позволяет гораздо лучше справляться с прерываниями. Воспользуемся им сегодня.

Не нужно сдвигать какие-либо дела на среду прямо сейчас. Если понадобится, вы сделаете это в конце дня.

Выполняйте план

Теперь выполняйте дела с приоритетом А, пока не сделаете все. Работа с запросами клиентов может привести к возникновению новых дел. Например, если запрос не может быть выполнен за один присест, имеет смысл занести его в список дел. Предположим, запрос №43001 из системы регистрации запросов требует исправления ошибки в пакетном задании, выполняющемся ночью, и проверки того, что исправление выполнено правильно. Вы можете внести коррективы, а в списке завтрашних дел сделать запись о необходимости проверки (рис. 5.10).

Если для реагирования на другой запрос понадобится заказать и установить программный продукт, вы можете оформить заказ сегодня, а затем сделать запись о предполагаемой доставке на страничке соответствующего дня.

Сделано?	Приоритет	Время	Дело
		15	43001: проверишь накам

Рис. 5.10. Добавление запроса из системы регистрации в список дел на среду

Я пользуюсь органайзером для отслеживания запросов, над которыми активно работаю. Однако список запросов, адресованных мне в системе регистрации, гораздо длиннее, но я не включаю их все в личный список дел на сегодня. Я записываю только те запросы, которые обрабатываю в настоящее время и которые нужно обработать к койкретной дате.

Интеграция портативного компьютера и системы регистрации запросов

Я не видел ни одной системы регистрации запросов, которая бы интегрировалась с портативным компьютером. Я уверен, что такие системы есть, просто мне они не попадались.

Если бы у меня была такая система, я бы присваивал приоритет А или В запросам, над которыми активно работаю, а всем прочим - приоритет Z. Можно пофантазировать, что по мере приближения критического срока исполнения система автоматически повышала бы приоритет отложенного запроса до уровня А. Важной особенностью такой системы была бы возможность включения запросов в список дел, но с тем чтобы в моем списке дел могли быть не только запросы. Включив в список дел пункт «Заняться стиркой», я не рассчитываю, что система занесет его в корпоративную базу данных.

Затем выполните все дела с приоритетом В. Поскольку пункт «Прерывания» буферный, вы не обязаны бездельничать, хотя никто и не прерывает вас.

Закончите рабочий день и покиньте офис

В конце рабочего дня уделите несколько минут обработке оставшихся пунктов. Невыполненные задания переносятся на следующий день, и вы покидаете офис с улыбкой, потому что обработали все задания.

Полезные советы

Эта система обладает достаточной гибкостью, и, столкнувшись с новыми обстоятельствами, вы легко адаптируете ее к ним. В этом разделе перечислены некоторые полезные приемы.

Большие проекты

Работая над крупным проектом, разбивайте его на отдельные этапы и распределяйте задачи по спискам дел на разные дни. Например, впишите название этапа в список дел на каждый понедельник в течение месяца, отведенного на этот этап.

Что делать, если вы закончили работу рано?

Если вдруг случится чудо, и вы сделаете все намеченное на сегодня, как вы поступите? Я думаю, вам следует поощрить себя. Вот некоторые идеи:

- Создайте небольшой задел на завтра.
- Займитесь каким-нибудь интересным проектом, до которого не доходили руки.
- Почитайте журнал из тех, что накопились за последнее время.
- Просмотрите систему регистрации запросов и удалите из нее устаревшие записи.
- Приберитесь в комнате, на рабочем месте, в почтовом ящике или во всей лаборатории.
- Сходите к шефу и попросите дополнительную работу. (Шутка!)
- Посидите за рабочим столом минут пятнадцать, ничего не делая. Проблемы найдут вас сами.
- Если у вас гибкий рабочий график, отправляйтесь домой. Вы это заслужили!

Задания, полученные в течение дня

Предположим, вы идеально распланировали рабочий день. Вы с точностью до минуты подсчитали время, необходимое для каждого задания, и уверены, что все будет сделано к концу рабочего дня.

Разумеется, думать так - все равно, что напрашиваться на проблемы. Именно в такие дни где-то после обеда к вам заходит шеф и излагает «блестящую» идею, для реализации которой потребуется выполнить множество дел, и ваш идеальный план проваливается.

Вот почему я рекомендую оценивать трудоемкость задач лишь приблизительно.

Итак, что делать, если в течение дня появляются новые дела? Один час в день на прерывания у вас уже запланирован. Если же возникнет серьезная проблема (на решение которой потребуется, скажем, три часа), то вы должны пересмотреть свои планы.

Подсчитайте, сколько рабочего времени у вас осталось, и посмотрите, помещаются ли в этот отрезок дела с приоритетами А и В. Если нет, перенесите их на следующий день вышеописанным способом. Как правило, приходится переносить все дела с приоритетом В и С. Если же

времени не хватает даже для дел с приоритетом А, вам придется поговорить с теми, кто ожидает их выполнения. Возможно, это как раз ваш шеф. Скорее всего, он проявит понимание и поможет вам расставить приоритеты по-новому. Если же это кто-то другой, то он заслуживает, как минимум, электронного письма с объяснениями и обещанием выполнить его запрос завтра.

Личные дела

Для своих личных дел я пользуюсь той же системой списков. Мой органайзер содержит записи и о стирке, и о магазинах. Так я практикуюсь в использовании системы, что благотворно сказывается на моей работе.

Если бы я применял разные системы для работы и личной жизни, мне пришлось бы носить с собой два органайзера. Думаю, одного вполне достаточно!

Пользуясь бумажным органайзером, я могу компоновать записи на страничке. То есть личные дела я заносу в нижнюю часть списка дел каждого дня. В результате на страничке он визуальнo распадается на два списка. Если вы предпочитаете портативный компьютер, то можете прибегнуть к какому-нибудь техническому трюку, например разделить задачи по категориям.

Уходя с работы, я всегда просматриваю список личных дел и проверяю, не нужно ли что-то сделать по дороге домой.

Настройка бумажного органайзера для системы Цикл

Если вы пользуетесь бумажным органайзером (попросту, блокнотом), то всегда сможете найти в канцелярском магазине странички, размеченные, как надо. Вам понадобятся:

- 12 страниц с календарем на месяц.
- Странички на каждый день месяца. В продаже имеются страницы с напечатанной датой от 1 января до 31 декабря. Обычно они размечены так, что одна сторона отведена под расписание, а другая - под список дел. Вам остается лишь раз в месяц добавлять в органайзер 30 страничек.
- Страницы для дополнительных заметок в конце блокнота. На них вы сможете записывать долгосрочные цели и вообще все, что сочтете нужным. (Это необязательные страницы.)
- Переплет с механизмом для закрепления страниц.

Если у вас не особо масштабные проекты, то вы сможете обойтись и органайзером небольшого формата. В органайзерах побольше внизу каждой страницы есть несколько линеек для заметок. Тем не менее, длинный список дел может у вас не поместиться даже на такой странице.

Настройка портативного компьютера для системы Цикл

Если вы пользуетесь портативным компьютером (PDA), у вас богатый выбор вариантов. Как правило, такие компьютеры поставляются с программным обеспечением, позволяющим вести календарь встреч и списки дел, а также делать заметки. Однако многие программы третьих фирм предоставят вам гораздо больше возможностей.

Пакет DateBook (DB) V (<http://www.pimlicosoftware.com>) завоевал мои симпатии уже довольно давно, потому что он позволяет без труда реализовать концепцию «список дел на каждый день». Для меня это критерий того, что должен уметь портативный компьютер. Без DateBook V он остается просто игрушкой. Этот программный пакет позволяет очень быстро вводить информацию с помощью шаблонов, эффективно работать с цветом для выделения важных дел, включать предварительное оповещение о событии. Последняя функция для меня особенно важна. До появления DateBook V мне приходилось устанавливать напоминание о чем-нибудь дне рождения и отдельное напоминание за неделю до этого дня, чтобы успеть купить подарок. Пользуясь пакетом DateBook V, я просто включаю предварительное оповещение об указанном событии. Пакет существует только в версии для PalmOS. Одним из его достоинств является поддержка структур данных PalmOS, так что все ваши записи синхронизируются корректно.

Life Balance (LB) (<http://www.llamagraphics.com>) не просто замечательный программный пакет, это целая философия. Пакет Life Balance поддерживает все составляющие Цикла (списки дел, календари и т. д.), а также реализует концепцию сбалансированности ваших жизненных целей. Предположим, вы решили распределить свое время между тремя проектами и личными делами. Если вы уделяете мало времени одной из этих категорий, приоритет дел из ее списка постепенно повышается. В конечном счете это позволит сбалансировать дела. Концепция, безусловно, замечательная, и многие пользователи не нарадуются этим пакетом. Он доступен для Mac OS X и Windows и синхронизируется с версией PalmOS.

Как-то я вел семинар по тайм-менеджменту для системных администраторов, и один из слушателей сказал мне, что, заканчивая какое-то дело, он просто мечтает, чтобы кто-нибудь «загрузил» его новым. Он признался, что слишком долго размышляет о том, чем заняться дальше. И тогда другой слушатель рассказал нам о пакете Life Balance. Вы программируете, как хотели бы сбалансировать свою жизнь, а пакет заботится о том, чтобы ваши дела соответствовали вашим целям.

Еще одним достоинством Life Balance является то, что вам не надо сортировать дела по категориям (например, Работа или Дом), - просто пометьте каждый пункт местом (или местами), где можете заняться данным делом. Когда вы в следующий раз окажетесь в гастрономе (или

в кабинете шефа), просто щелкните по названию этого места, - и Life Balance выведет список дел, которые должны быть сделаны здесь. Очень удобно!

Чем пользуюсь я сам? Я перепробовал множество платформ. И все же должен признаться, что всегда возвращался к той, с которой когда-то начал, - к блокноту-органайзеру в кожаном переплете с разноцветными страничками на каждый день. Думаю, что, даже если бы моей первой платформой был бутерброд с арахисовым маслом и бананом, я всегда бы возвращался к нему, поскольку, выработав привычку, стремлюсь ее придерживаться.

Если подводят поставщики, берите дело в свои руки

Умение доводить дело до конца подразумевает не просто отслеживание рабочих вопросов. Главное - получить результат. Один мой сослуживец никак не мог понять, почему его начальник недоволен задержкой проекта, ведь это поставщик не звонил в течение двух недель. Не коллега же виноват, что поставщик куда-то запропастился, правда?

Мир несовершенен. Вы не можете полагаться на то, что поставщик перезвонит вам. Вы должны взять на себя ответственность за своевременное завершение проекта. Вот несколько советов:

- **Звонить поставщику раз в день, пока не застанете его.** Звоните каждый день. Не ждите, пока он проявит инициативу.
- **Звоните поставщику рано утром.** Если вы «поймаете» нужного сотрудника в начале рабочего дня, он, возможно, займется вашим вопросом. Если вы позвоните в конце дня, о вашей просьбе к утру забудут. (Они не читали эту книгу.)
- **Регистрируйте в органайзере все свои звонки.** Записи о звонках могут оказаться полезными, если дела действительно плохи.
- **Всегда оставляйте сообщение на автоответчике.** Вам нужны доказательства того, что вы позвонили. Позвонить и не оставить сообщение - все равно, что не звонить вовсе. Не надо выдумывать что-то оригинальное. Каждый раз просто говорите: «Говорит [ваше имя]. Пожалуйста, перезвоните мне по телефону [ваш номер телефона]. Мне необходимо [что вам нужно] для проекта [название проекта]. Спасибо». Если вы не укажете причину звонка, это будут прятки по телефону. Если вы сообщите, зачем звонили, поставщик будет работать по вашему делу, даже если не дозвонится до вас. Впрочем, будьте лаконичны и назовите свой телефонный номер в начале сообщения, а не в конце.

С учетом вышесказанного ваш заказ не считается принятым, пока поставщик не сообщит номер заказа и/или дату поставки.

Меня много раз подводили поставщики (и отделы закупок), срывая сроки поставки: «Ах, я не могу выполнить ваш заказ, потому что вы не прислали еще одну форму». Почему мне сразу не сказали? Разве эти люди не получают процент с продаж? Или они нашли такой загадочный бизнес, в котором доход обеспечивает непоставка товара?

Из многолетнего опыта я сделал вывод, что не следует задавать вопрос вроде: «Вам еще что-нибудь от меня нужно?» Я наивно полагал, что торговый представитель с сильной мотивацией воспользуется этой возможностью, чтобы поскорее заключить сделку и получить свой процент. Нет. Реальная проблема заключается в том, что причина торможения процесса не всегда имеет отношение ко мне. Возможно, требуется какое-то подтверждение из банка или конструкторы прибора должны получить чью-то подпись. С технической точки зрения здесь должен действовать отдел закупок, а не я. Люди не желают чувствовать себя обязанными вам, поэтому они вежливо говорят вам в ответ «Нет». На самом деле, если я знаю, что процесс затормозился не по моей вине, я могу, по меньшей мере, подтолкнуть его, то есть звонить в отдел закупок и поторапливать их.

Поэтому, когда договоренность, казалось бы, уже достигнута, я задаю волшебный вопрос: «Не могли бы вы сообщить мне, когда придет товар?» И тогда в голове торгового представителя что-то шелкает, и он либо дает конкретный ответ, либо сообщает о препонах: этот товар не будет поставляться до следующей весны, или он не передал в финансовый отдел мою заявку, хотя я и заполнил все формы. (Оба примера взяты из реальной жизни.)

Узнав дату поставки, я задаю следующий вопрос: «Не могли бы вы сообщить мне номер заказа?» Это единственный способ проверить, не сел ли мой заказ на очередную мель. Если проект важный, я звоню ежедневно, пока не получу номер заказа. Я всегда звоню утром и, если попадаю на автоответчик, каждый раз оставляю вежливое сообщение.

Во всех остальных аспектах я сторонник высоких технологий и постоянно модернизирую системы, которыми пользуюсь. Я регулярно перехожу на более производительные компьютеры и более современные операционные системы. Я пользовался электронной почтой задолго до того, как она стала повседневым явлением. Я создал одну из самых больших WiFi-сетей задолго до того, как эта технология стала называться WiFi, и купил Tivo до того, как большинство моих друзей узнало, что это такое. Тем не менее, когда дело касается тайм-менеджмента,

я предпочитаю бумагу. Я люблю писать большими буквами, сокращать, рисовать стрелочки и кружочки. Если кто-то начинает забрасывать меня информацией, я предпочитаю сразу записывать ее, не дожидаясь, пока микропроцессор проснется. С 1991 по 2004 год я пользовался одним и тем же органайзером - большим (21,6x28 см), в кожаном переплете, а в январе 2005 года выбрал органайзер поменьше (14x21,6 см, но по-прежнему в кожаном переплете!).

Просто я - человек привычки.

Дело в том, что пользоваться надо тем, что вам подходит. Попробуйте все системы (особенно программные пакеты - они обычно представлены бесплатной версией с ограниченным сроком использования). Попросите у кого-нибудь из друзей или коллег портативный компьютер - ненадолго, хотя бы на несколько минут в обеденный перерыв. Зайдите в канцелярский магазин и посмотрите, какие бывают ежедневники и органайзеры.

Резюме

- В течение дня выполняйте намеченный план. Отмечайте выполненные пункты крестиком, а перенесенные - черточкой.
- Ближе к концу рабочего дня обработайте все невыполненные пункты, чтобы задержки в выполнении запросов не были неприятным сюрпризом для клиентов.
- В конце дня все пункты оказываются обработанными в том смысле, что они либо завершены, либо каким-то образом перенесены на последующие дни. Цель этой процедуры в том, чтобы покинуть офис не с ощущением непосильной ноши, а с сознанием того, что все задачи под контролем. Вы можете уйти домой с улыбкой, зная, что справились с сегодняшними делами. Завтра будет новая работа.
- Если вы переделали все дела до конца рабочего дня, наградите себя. Вы можете заняться своим личным проектом, почитать техническую литературу, навести порядок в помещении, создать задел на следующий день и т. д. Если у вас гибкий график, можно уйти домой пораньше.
- Если в течение дня появляются новые дела, их следует запланировать на завтра, либо придется перетряхнуть сегодняшнее расписание, чтобы выполнить их сегодня. Если по ходу рабочего дня постоянно возникают дела, которые надо сделать «прямо сейчас», можно уже при планировании отвести определенный отрезок времени на обработку этих прерываний.
- Управлять личными делами можно по этой же системе. Применяя единую систему для работы и личной жизни, вы, во-первых, практикуетесь в системе Цикл, во-вторых, носите с собой только один инструмент тайм-менеджмента, наконец, не забываете о личных делах, от чего ваша жизнь делается светлее.

- Важно не то, какой системы вы придерживаетесь, а то, что это система. Иными словами, система Цикл хороша для меня, но если вас устраивает другая, применяйте ее. Правда, если вы читаете эту книгу, вероятно, у вас нет собственной системы. Начните применять систему Цикл, а потом адаптируйте ее под свои привычки и стиль жизни.
- Систему Цикл можно реализовать с помощью портативного компьютера или бумажного органайзера. У каждого инструмента есть свои плюсы и минусы. Попробуйте оба и выберите тот, что вам больше подходит.

6

Система Цикл: работа с календарем

В главе 5 мы подробно рассмотрели, как планировать свой день, пользуясь списком дел как составной частью системы Цикл, представленной в главе 4. Сейчас мы увидим, как эффективная работа с календарем позволяет выполнять рутинные процедуры и превращает цепь дней в связную и согласованную последовательность.

В каждом органайзере имеется календарь, иногда озаглавленный Календарь встреч (datebook). Это не только хранилище информации на каждый конкретный день (встречи, сроки, дни рождения, этапы проекта), но и универсальное средство долгосрочного планирования (карьерный рост, протяженные проекты, планирование жизни вне работы).

Календари позволяют нам рассмотреть картину в целом. В начале карьеры системного администратора я все время беспокоился, что не вижу общую картину своей работы. Мне казалось, что несмотря на упорный труд я стою на месте. Все изменилось, когда я стал мыслить в терминах календаря. Разумеется, ежедневный список дел помогает мне понять, что я должен сделать сегодня, но благодаря календарю я могу оценить картину в целом.



Как пользоваться календарем

В системе Цикл календарь органайзера используется для трех целей. Во-первых, для резервирования времени под совещания и встречи, назначенные не на сегодня. Во-вторых, для напоминания о важных моментах в будущем, таких как дни рождения и годовщины каких-то событий. В-третьих, если у вас бумажный органайзер, в календаре вы ведете список дел, намеченных на отдаленное будущее. (Например, если вы ведете ежедневные списки дел только на текущий месяц, то в календарь записываете дела на следующий месяц.) Вот более подробное описание структуры календаря:

- **Встречи и совещания.** Каждый раз, договариваясь о встрече или узнав о совещании, записывайте это в календарь. Пользуйтесь календарем для резервирования времени на будущие дела. Это также поможет вам избежать конфликтов.
- **Важные моменты.** Записывайте дни рождения, годовщины и прочие важные даты, например корпоративные праздники или дни, когда ваши коллеги уходят в отпуск.
- **Списки дел на будущее.** Наконец, если вы пользуетесь бумажным органайзером, вы можете записывать в календаре дела, намеченные на отдаленное будущее. Например, в ваш органайзер, скорее всего, помещается лишь месячный запас страничек с ежедневными списками дел. Если вы должны выполнить профилактический осмотр цветного принтера через два месяца после его установки, можно отметить этот день в календаре и перенести запись в ежедневный список дел, когда эта дата наступит.

Принцип прост: всегда все записывайте и каждый день заглядывайте в календарь.

Во время десятиминутного планирования своего дня посмотрите в календаре, что у вас намечено на сегодня. Составьте расписание с учетом сегодняшних встреч. Дела, перечисленные в календаре, занесите в сегодняшний список. Важные даты и записи о сроках окончания работ можно записать в список дел как дополнительные пункты.

Календарь в моем органайзере расчерчен на довольно крупные прямоугольники. Мне это нравится. Я использую разные области прямоугольника для разных целей. В верхней части я пишу дни рождения и годовщины. Чуть ниже я отмечаю отпуска сотрудников и вообще все, что может продолжаться несколько дней. В самой нижней части записаны дела, запланированные на вечер. Поскольку я обычно не планирую больше одного дела на вечер, мне достаточно одной последней строчки. Середину прямоугольника я заполняю текущими делами. В центре - обед. Выше - утренние совещания, а ниже - встречи, назначенные на вторую половину дня. Пример показан на рис. 6.1.

Самое трудное - выработать привычку все записывать. В следующих разделах я даю советы, как этого добиться.

19	20	21
	10-11 Совещание	12 Обед
26	27	28
	15-16 Дзюжон	

Рис. 6.1. Пример записи в моем календаре

Никогда не пропускайте встречи

Случается ли, что вы пропускаете встречу или совещание? Часто ли вы извиняетесь перед коллегами за то, что не явились на встречу? Конечно, у вас была уважительная причина: вы заработались и забыли о встрече. К сожалению, «Я забыл» - плохое оправдание.

Ничто так не вредит вашей репутации, как неявка на встречу с клиентом. Первый шаг к созданию имиджа надежного системного администратора - всегда приходиться, если вы пообещали. Пропускать развлечения и какие-то полезные для вас мероприятия тоже нехорошо.

Своевременная явка демонстрирует вашу ответственность, коллеги видят в вас надежного человека. Вы проявляете уважение к тем, кто попросил вас о встрече, а они отплатят вам, явившись на встречу, когда попросите вы.

Секрет того, чтобы никогда не пропускать встречу, прост: всегда пользуйтесь календарем. Записывайте все встречи; никогда не договаривайтесь о встрече, пока не сверитесь с календарем, чтобы не было конфликтов.

И, самое главное, не полагайтесь на память. Органайзер - подходящий инструмент для хранения дат, а ваш мозг - нет. Эту фразу я уже повторил несколько раз, а ведь это только шестая глава. И я еще не раз повторю: берегите свой мозг для обдумывания важных дел. Храните информацию в органайзере. Не доверяйте своему мозгу.

Хочу сделать маленькое признание. Было время, когда я регулярно пропускал встречи. Хуже того, я назначал несколько встреч на одно время. Я мог согласиться на какую-то дату, не справившись с календарем. Это было самонадеянно, но я думал: «Так, 4 июня... Дата знакомая,

но не припомню, чтобы на нее что-то было назначено, поэтому, скорее всего, я буду свободен». Конечно, дата казалась мне знакомой, потому что на нее уже что-то было запланировано!

Получалось крайне неловко. Звоня кому-то о переносе на другое время, тратишь свое время и создаешь другому проблемы. Время, потраченное на перенос встреч, уже не вернуть. К счастью, это в прошлом, потому что теперь я прилежно записываю все в календарь.

И еще одно признание. Причина, по которой я не мог заглянуть в календарь, заключалась в том, что я стеснялся сказать: «Подождите секунду, я достану календарь». Мне почему-то казалось, что человеку трудно подождать. Разумеется, это не так, перекроить расписание куда труднее. Я понимал, что это глупо, но ничего не мог с собой поделать. (Некоторые знакомые признавались мне в своей боязни показаться слишком чванливыми и самодовольными: «Посмотрите-ка! Я так всем нужен, что приходится вести календарь встреч. Сейчас я вас впишу».)

Наконец, я решил, что пора сломать эту плохую привычку. Помню, как боялся, впервые прося собеседницу подождать, пока я сбегаю за календарем. Я еле выдавил из себя: «ОК, подходит. Не могли бы Вы подождать, пока я посмотрю, свободен ли у меня этот день?» Я замолчал и ждал ответа так, словно спросил что-то несусветное, к примеру не одолжит ли она мне денег или как называется столица штата Висконсин. Я весь вспотел от страха. А она ответила: «Да, разумеется». Через полминуты я вернулся с календарем в руках. Она нисколько не была раздосадована ожиданием. Она понимала, зачем мне нужен календарь. Более того, она достала свой календарь. Моя вечно спешащая натура ничуть не пострадала от такой задержки. И конец света не наступил.

Всегда звоните, если вам приходится опаздывать или пропускать встречу

Лучше позвонить перед началом встречи, чем оставлять человека в неведении. Даже если вам неловко предупреждать о своем опоздании, в век сотовых телефонов не позвонить непростительно. Имеет смысл повесить в офисе на видных местах внутренние телефоны сотрудников, чтобы с каждым из них можно было быстро связаться.

Короткое сообщение «Я звоню, потому что вынужден опоздать» гораздо лучше сбивчивых оправданий, когда вы заявитесь с опозданием.

И никогда не лгите. Сказать правду лучше, потому что тогда вам не придется запоминать, кому и что вы наврали. Это будет дополнительной нагрузкой на вашу память, а мы решили использовать мозг только для важной работы.

Надеюсь, что мой личный опыт поможет вам справиться с вашими страхами. Если я смог уцелеть, сможете и вы.

Итак, начинайте пользоваться календарем прямо сейчас. Запишите какую-нибудь встречу. (Если у вас ничего не запланировано, назначьте встречу самому себе на завтра в обеденный перерыв.)

Обычно труднее всего сделать что-то в первый раз. Возможно, вы никогда не пользовались этой секцией своего органайзера или, скажем, вам придется сбегать в магазин за страничками с календарем.

Я подожду (даже если вы побежите в магазин).

Ну что, вернулись? Вы занесли в календарь свою первую встречу?

Ничего страшного, не правда ли?

Один календарь для деловой и личной жизни

Баланс имеет значение. Работа, семья, друзья, общественная деятельность, личные проекты, сон - все это, безусловно, важные вещи. Я твердо убежден, что для них нужен единый календарь.

Чтобы работа не подавила вашу личную жизнь, ведите единый календарь (или объедините календари в портативном компьютере). Если вас просят задержаться после работы, вы сможете заглянуть в календарь и проверить, свободен ли у вас вечер.

Бывало, уходя с работы, я вдруг вспоминал, что приглашен на вечеринку или запланировал какое-то другое развлечение. И я мчался туда, как безумный, чтобы успеть до шапочного разбора. Оказавшись на месте, я уже не мог развлекаться, потому что был слишком зол на себя за забывчивость. Теперь все намеченное на вечер входит в мой ежедневный план. Успеть в гости вовремя для меня так же важно, как закончить проект в срок. Я напряженно работаю весь день, но вечер - мой.

Большинство программ для портативных компьютеров позволяет вести несколько календарей, а затем сводить их в одном окне. Имеет смысл настроить программу синхронизации на портативном компьютере так, чтобы ваш личный календарь не синхронизировался с календарем на рабочем компьютере (а рабочий календарь не синхронизировался с календарем на вашем домашнем компьютере). Скорее всего, вы не захотите, чтобы ваши планы поразвлекаться хранились в памяти рабочего компьютера.

Повторяющиеся задачи

Все на свете повторяется. Это касается и совещаний, и замены масла в автомобиле, и ураганов, и приготовления пищи. Многие процедуры, описанные в предыдущей главе, стали повторяющимися событиями. Одно из преимуществ портативного компьютера перед бумажным органайзером состоит в том, что достаточно однажды запрограммировать

Автоматическое напоминание о событиях для владельца бумажного органайзера

В отличие от портативного компьютера, бумажный органайзер нельзя запрограммировать так, чтобы он издавал звуковой сигнал, когда придет время отправляться на совещание. Что можно сделать в таком случае?

Запрограммируйте другую систему, чтобы она предупреждала вас о встречах, назначенных на сегодня. Сделайте это частью утренней процедуры. Пусть календарь в вашем органайзере служит основным источником информации, а какая-нибудь программируемая система оповещает вас о сегодняшних событиях.

По поводу систем напоминания:

- У пользователей Windows и Мака имеется огромный выбор таких программ. Поищите в Гугле строку «программа-будильник», и вы найдете множество программ от бесплатных до сравнительно недорогих.
- Приложения iCal (для Мака), Outlook (для Windows), а также программы с открытым кодом Evolution и KOrganizer просигнализируют вам о приближающемся совещании (что очень полезно, если вы постоянно работаете за компьютером, на котором установлены названные программы).
- Для напоминания о ближайшей встрече установите будильник на наручных часах или сотовом телефоне.
- Пользователям UNIX-подобных систем, Mac OS X, BSD, Debian Linux и HP Tru64 Unix доступна команда `leave`, предупреждающая о том, что пора уходить на совещание.
- Запрограммируйте сервер так, чтобы он в нужный момент отправлял вам сообщение на сотовый телефон или пейджер. В системе UNIX/Linux можно воспользоваться службой `at` и отправлять сообщение на сотовый телефон в нужное время:

```
$ at 11:50
>echo Meet Bob for lunch | mail 19085552323@teleflip.com
> ^D
```

Помните, чтобы успеть добраться до места встречи, всегда устанавливайте время с запасом, будь то двухминутная поездка на лифте на другой этаж или двухчасовая поездка на автомобиле в другой город.

повторяющееся событие, и оно будет отслеживаться. Вот примеры событий, которые имеет смысл занести в календарь:

- Еженедельные совещания.
- Регулярные встречи.
- Ближайшие конференции.
- Планы на отпуск.
- Сроки проектов.
- Вечеринки. (Я записываю их, получив письменное приглашение с просьбой ответить. Я записываю, даже если не собираюсь туда идти, чтобы нечаянно не отправить отказ дважды.)
- Расписание футбольных матчей вашего сына.
- Дата выхода квартального отчета вашей фирмы.
- Все известные вам дни рождения. (В том числе дни рождения знаменитостей! Приятно узнать, что 20 апреля — день рождения легендарного музыканта Тито Пуэнте.)
- День разговоров на пиратском языке (19 сентября), День полотенца (25 мая) и День системного администратора (28 июля).

Запишите это в свой органайзер (компьютер) прямо сейчас. Выработайте привычку заносить в календарь все новые даты, как только услышите о них.

Мы записываем то, что считаем ценной информацией

Записывая информацию в календарь, вы тем самым демонстрируете ее ценность для вас. Если вы договариваетесь с клиентом о встрече и тут же в его присутствии делаете запись о ней, вы показываете, что эта встреча важна для вас. Это касается и деловых, и личных отношений. Представьте, вы договорились с девушкой о свидании, и она записывает время и место в органайзере. Теперь можно не сомневаться, что она придет, не так ли?

Преподаватель на курсах тайм-менеджмента рассказал мне, как он заметил, что его дочурка поняла: если договоренность записана, о ней никто не забудет. Когда он пообещал сводить ее в следующее воскресенье в зоопарк, она схватила большой зеленый фломастер и написала в его органайзере огромными буквами «ЗООПАРК!». Это подействовало.

Запись повторяющихся задач в органайзере

Если вы пользуетесь бумажным органайзером, у вас есть несколько способов, позволяющих не забыть о повторяющихся событиях. Органайзер, в принципе, может стать системой напоминания. Я храню в органайзере списки еженедельных, ежемесячных и ежегодных событий.

Каждый понедельник я просматриваю еженедельный список и заполняю календарь на неделю. Первого числа каждого месяца я просматриваю ежемесячный список и заполняю календарь на месяц. В первый день нового года я заполняю календарь на наступивший год.

Не слишком вдаваясь в подробности, вот что конкретно я делаю:

В качестве закладки в органайзере я использую прозрачный пластиковый конвертик с листком бумаги внутри. На листке у меня записаны все еженедельные встречи. В понедельник я заносу их в графы расписания на каждый день недели. Если неделя у меня не слишком загружена, я просто просматриваю эту закладку каждый день.

Ежемесячные события я отслеживаю, когда вставляю в органайзер странички на следующий месяц. Это может произойти в любой день месяца. Впрочем, ежемесячные совещания лучше сразу отметить на год вперед. Одна общественная организация, с которой я сотрудничаю, каждый декабрь публикует расписание собраний на год вперед. Получив это расписание, я тут же отмечаю все собрания в календаре.

Ежегодные события - это, как правило, дни рождения. Их список я храню в компьютере. (Раньше я врал. Для простоты изложения.) Купив блок страничек для органайзера на следующий год, с помощью этого списка я размечаю ежегодные события в календаре. У меня традиция: в первый день наступившего года заносить в календарь все ежегодные события.

Портативный компьютер тоже годится для хранения дат отдаленных событий, таких как конференции. Блок страничек, который я покупаю для органайзера, включает листок для записи событий на несколько лет вперед. Ежегодно, меняя страницы, я просматриваю эти листочки и переписываю их содержимое в календарь на этот год. Я предвижу совпадающие даты свадеб и выпускных вечеров на два-три года вперед.

Узнайте свой личный ритм

Все в природе подчиняется определенным ритмам. Соглашаясь прийти на встречу или собрание, хорошо бы учитывать свой личный ритм.

В течение дня я могу работать с очень высокой производительностью два часа.

Один час - это первый час до того, как все сотрудники собрались в офисе. Я не "жаворонок", но я заметил, что если заставлю себя притащиться в офис за час до начала рабочего дня, то сделаю за первый час больше, чем за все оставшееся рабочее время, потому что никто не будет меня отвлекать. Очень важно не растратить этот час на занятия вроде разбора электронной почты. Используйте его для работы над проектом, требующим полной сосредоточенности. (Примеры приведены в главе 2.)

Второй час - это час вашего пика активности. В это время вы можете сконцентрироваться на работе лучше, чем в остальную часть дня. Я на-

зываю его «часом напряженной умственной работы». У каждого этот пик наступает в разное время суток. Для некоторых это день, для других - поздний вечер или раннее утро. Это чисто биологический феномен (поищите в Гугле «биоритмы»). Планируя работу астронавтов на орбите, NASA учитывает их биоритмы. Многие учитывают свои биоритмы, составляя расписание на день, чтобы заняться самой трудной работой, когда у них естественный прилив энергии и способность к концентрации.

Попробуйте на следующей неделе выяснить, в какое время у вас наступает пик умственной активности. Например, настройте компьютер так, чтобы он ежечасно издавал звуковой сигнал. Услышав гудок, оцените свой уровень активности и способность к концентрации по десятибалльной шкале и запишите эти данные. Выясните, в какое время оба показателя достигают высоких значений.

Узнав свой биоритм, попробуйте изменить свое расписание в соответствии с ним.

Занимайтесь умственной работой на пике своей умственной деятельности. Отводите это время под самые важные проекты (требующие максимальной сосредоточенности). Старайтесь назначать совещания на другое время, если только совещание не потребует от вас серьезной умственной работы. (А такие совещания бывают редко.)

Время пика вашей активности с возрастом может измениться. Когда я был моложе, мой максимум приходился на два часа ночи. Сейчас это два часа дня.

Узнайте ритм своей фирмы

У бизнеса тоже есть ритмы. Определив годовые ритмы фирмы, в которой работаете, вы сможете согласовать с ними свои планы. В противном случае вам придется плыть против течения. Календарь как средство долгосрочного планирования прекрасно подходит для этой цели.

У каждой фирмы в течение года бывают как напряженные, так и относительно спокойные периоды. Вы можете планировать задачи системного администрирования в соответствии с этими периодами.

Я работал в фирме, выпускавшей три версии программного продукта в течение года. Каждые 120 дней появлялась новая версия. Как правило, первый месяц уходил на маркетинговые исследования и планирование функциональных возможностей. Следующие 60 дней уходили на разработку, две недели - на тестирование качества и еще две на тиражирование ПО и документации. Сто двадцатый день был днем выпуска новой версии. Для системного администратора это было счастьем, поскольку система работала, как часы, трижды в течение года.

В течение первого месяца цикла большинство сотрудников сидело на совещаниях и сетевая активность была низка. Работала только электрон-

ная почта, а по остальным функциям можно было проводить профилактику или модернизацию. В последние дни каждого цикла специальная группа сотрудников определяла, какая версия ОС и компилятора будет использована для следующего выпуска. В течение первого месяца текущего цикла группа системных администраторов устанавливала эти инструменты. На этапе разработки допускались запланированные простои оборудования, и системные администраторы работали в обычном режиме. Однако ближе к концу 120-дневного цикла запланированные простои запрещались. Это была очень напряженная часть цикла, в течение которой новый код поступал в группу тестирования качества практически ежедневно. В этот период системным администраторам было нечего делать. Часть команды оставалась на случай непредвиденных обстоятельств, а остальные уходили в отпуск. После тестирования, на этапе тиражирования, стабильная работа требовалась только от тех компонентов системы, которые участвовали в тиражировании. Остальные сотрудники отдыхали. Затем цикл повторялся.

Поскольку работа системных администраторов планировалась в соответствии с деловым циклом фирмы, все протекало гладко.

Другим типичным примером цикла в бизнесе является декабрьский всплеск. Например, не секрет, что половину объема продаж многие фирмы реализуют в пред рождественский период, часто работая в убыток остальную часть года. В течение предпраздничной лихорадки компьютерная сеть, обслуживающая бизнес, должна работать абсолютно стабильно. Час простоя может обернуться потерей миллионов. Поэтому в такое время у системных администраторов мало запланированной работы и много незапланированной, в основном по устранению сбоев и обслуживанию перегруженных серверов. Разработчиков к продукту уже не допускают, чтобы у них не было соблазна внести «полезные» нововведения, подвергая риску весь проект. Самое напряженное время для всех наступает за несколько месяцев до этого периода, когда все усилия направлены на то, чтобы новая система заработала. Самое спокойное время - первая неделя года, когда сотрудники восстанавливают силы.

Учебные заведения подчиняются вполне очевидному ритму. Их деятельность сосредоточена вокруг таких основных событий, как вступительные экзамены, зачисление новых студентов, бюджетные ассигнования, сессии, выпуск студентов, летние каникулы.

В кадровом вопросе тоже есть циклы. Например, если речь идет о найме молодых специалистов, то процесс должен начаться в феврале с размещения объявлений в колледжах в расчете на заполнение вакансий в мае. Если же фирма планирует нанять более опытных работников, то, скорее всего, у таких людей есть дети, и они предпочтут менять место работы не в течение учебного года, а летом. В конце года на работу с кадрами могут повлиять и другие факторы. Редко когда мне удавалось добиться согласия начальства на прием нового сотрудника в декаб-

ре. Иногда из-за того, что начальство в это время в отпуске, но чаще дело было в том, что прием на работу человека в конце года портит отчетность. Учитывайте вышесказанное, планируя нанимать сотрудников.

Мне нравится работать в фирмах с четко выраженной цикличностью деятельности. Это существенно облегчает планирование. Если же я работаю в фирме, действующей без явных циклов, я стараюсь найти циклы неофициальные или, если возможно, подчинить работу фирмы определенному циклу. Однажды я работал в компании, производившей ПО без определенного расписания. Я стал бороться за цикличность выпусков программного продукта и добился своего. От этого выиграли все. Отдел маркетинга и финансовый отдел стали планировать работу в соответствии с циклом, и практически была снята проблема несвоевременного ухода программистов в отпуск.

У фирмы, в которой вы работаете, наверняка есть свой цикл. Он может быть коротким и укладываться в месяц, а может быть растянут на весь год. Если вы работаете в NASA, ваш цикл может длиться несколько лет в соответствии с продолжительностью космических проектов. Если вы работаете в политической организации, ее цикл зависит от периодичности выборов кампаний.

Не поленитесь изучить ритм своей фирмы. Возможно, вам понадобится спросить шефа, как он представляет себе бизнес-цикл. Прояснив этот вопрос, поищите ответы и на другие:

- Каков бизнес-цикл вашей фирмы?
- Как вам лучше планировать свои проекты?
- В какое время вам лучше всего уходить в отпуск?
- Может ли группа системных администраторов лучше планировать свою работу над проектами?
- Можно ли осуществлять системное администрирование в соответствии с напряженными и спокойными периодами бизнес-цикла?
- Если деятельность фирмы не имеет периодичности, можно ли придать ей циклический характер? А что если установить циклическое расписание работы системных администраторов в расчете на то, что другие подстроятся к этому циклу?

Резюме

- Ведение календаря важно для вас и вашей карьеры. Пунктуальность ассоциируется с ответственностью и надежностью. Люди, не являющиеся на встречи и забывающие о совещаниях, не получают повышения по службе.
- Без хорошо организованного календаря вы рискуете пропустить важные события как по работе, так и в личной жизни. Необходимо соблюдать баланс между работой, семьей, общественной деятельно-

стью, личными проектами, сном и т. д. Ваш календарь поможет вам в этом.

- В органайзере должна быть секция для записи встреч (совещаний и прочих событий). Записывайте дату и время встречи, как только договоритесь о ней. Не соглашайтесь на встречу, пока не сверитесь с календарем.
- Ваш календарь является составной частью системы Цикл. Вы записываете в него встречи, совещания, важные события и прочую информацию. Планируя свой день, начните с календаря, чтобы составить расписание и дополнить список дел на сегодня.
- В бумажном органайзере можно записывать события в графы календаря по определенной системе. Я сначала вписываю дни рождения и годовщины, затем отпуска и многодневные мероприятия. В середине графы календаря я пишу мини-расписание на этот день: утренние встречи, обеденный перерыв, встречи во второй половине дня. Нижнюю часть графы я отвожу под планы на вечер.
- Договариваясь о встрече, учитывайте свой ритм активности. Если есть выбор, планируйте умственную работу на время пика своей умственной активности.
- Составляя совместные планы, всегда сверяйтесь с календарем, прежде чем согласитесь на встречу. Не смущайтесь тем, что вашему собеседнику придется подождать, пока вы достанете и откроете органайзер.
- Пользуйтесь автоматическими средствами напоминания о встречах. Установите подачу сигнала на портативном компьютере. Если у вас бумажный органайзер, используйте альтернативную технологию (например, будильник).
- В бумажный органайзер можно записывать повторяющиеся события на основе списков еженедельных, ежемесячных и ежегодных событий. В понедельник запишите в календарь все еженедельные встречи. В первый день месяца запишите ежемесячные встречи. А в первый день нового года - ежегодные события.
- У большинства фирм есть ежегодный ритм деятельности. Например, самое напряженное время у торгующих предприятий приходится на декабрь. Выяснив этот ритм, вы сможете планировать свои проекты в соответствии с ним. В противном случае вы будете грести против течения. Если у вашей фирмы нет четко выраженного ритма, определите ритм для себя.

7

Система Цикл: долгосрочные цели

К шестидесяти годам я хочу стать пенсионером, имеющим достаточно средств к существованию.

В течение ближайших трех лет я хочу получить повышение и стать руководителем группы.

В следующем месяце я хочу получше изучить ядро Linux.

Я хочу, чтобы через 24 часа у меня все было постирано и поглажено.

Иногда мне хочется назначить свидание порнозвезде.

Вы можете добиться всего, чего захотите, если серьезно настроитесь на это. Большинство людей игнорирует логический процесс постановки целей, планирования шагов для достижения этих целей и воплощения этих шагов в жизнь. Вместо этого они просто ждут, что все произойдет само собой.

Овладев тайм-менеджментом в совершенстве, я осознал, что, несмотря на мой высокий профессиональный уровень, в моей жизни за прошедший год ничего не произошло. Я не переехал в дом, о котором мечтал, вычислительная техника, которую я обслуживаю на работе, осталась прежней, и я никак не приблизился к свиданию с порнозвездой. Колеса моей жизни буксовали.

Действительность такова, что вы достигнете большего, если установите определенные цели. Исследования выявили, что люди, добившиеся успеха в жизни, ставили цели и стремились к ним. Другие исследования показали: большинство неудачников думает, что, если ничего не предпринимать, им все равно представятся какие-то возможности. Иными словами, неудачники надеются на удачу. Но под лежащий камень вода не течет. От друзей я слышал, что в шахматах все зависит от везения: чем больше играешь, тем больше везет. Так и в жизни.

Однако меня заинтересовало утверждение, что постановка цели без приложения усилий по ее достижению все-таки лучше, чем полное от-

существование целей. Если задуматься, это утверждение не лишено смысла. Если у вас нет определенных целей, то вы не сможете распознать те многочисленные счастливые возможности, которые открываются перед вами случайно. Предположим, вашему шефу потребовался добровольный помощник в составлении бюджета на следующий год. Скорее всего, это означает массу работы без особого вознаграждения. Я прекрасно пойму вас, если вы не захотите в этом участвовать. Однако если одна из поставленных вами целей заключается в том, чтобы стать руководителем группы, вы воспримете эту ситуацию как возможность поучаствовать в планировании деятельности группы. Если ваша цель — получить должность менеджера, то вы расцените просьбу шефа как возможность научиться составлять бюджет и лучше подготовиться к этой должности. И наоборот, если ваша цель — остаться инженером, пресекая попытки руководства продвинуть вас в менеджеры, вы увидите здесь другую возможность — не прилагать к этому руку. (Я видел слишком много хороших технических специалистов, ставших менеджерами поневоле.)

Приемы, обсуждавшиеся в этой книге, отлично годятся для выполнения повседневной работы. А как насчет долгосрочных целей, для достижения которых требуются годы?

Для начала задайте себе следующие вопросы:

- Каким я хочу видеть мое рабочее окружение (вычислительную технику) через два года?
- Какую карьеру я хочу сделать за пять лет?
- Какое общественное и финансовое положение я хочу занимать через десять лет?
- Какую жизнь я хочу вести, выйдя на пенсию?

Технические приемы в этом случае очень просты. Вам следует сформулировать цели, расставить приоритеты и разработать шаги для достижения этих целей. Затем вы заносите эти шаги в список дел и расставляете их в календаре.

Возможно, вы решите, что я обрушил на вас слишком много информации, к которой вы еще не готовы. Вы барахтаетесь в бескрайнем океане



проблем тайм-менеджмента. Хорошо. Бегло просмотрите эту главу и перечитайте ее, почувствовав, что овладели основами тайм-менеджмента.

Секретный прием

Скажу вам по секрету: нужно записывать свои цели. Пока они находятся у вас в голове, они не так конкретны, как вам кажется. Они слишком туманны. Невозможно оценить их, рассказать о них кому-нибудь. С ними невозможно работать.

Процесс изложения на бумаге заставляет вас конкретизировать их. К тому же в готовом списке целей гораздо удобнее расставлять приоритеты.

Цели, записанные на бумаге, можно показать другому человеку. Если есть кто-то, чье мнение дорого для вас, вы можете обсудить эти цели вместе. Иногда мы забываем, что наши близкие не умеют читать мысли. Рассказывая о своих целях, мы получаем поддержку и, так сказать, тестируем реальность этих целей. Мы с большей вероятностью достигнем цели, если поделимся с кем-то. Это подтолкнет нас к действиям.

Впрочем, записать неопределенные цели довольно легко. Вы можете конкретизировать свою цель, ответив на следующие вопросы:

- Чего я хочу добиться?
- Когда я хочу добиться этого?

Все забывают про второй вопрос. Может статься, вы никогда не начнете действовать, не установив определенный срок. В начале главы я привел список целей, и у каждой был указан срок.

Очень важно, чтобы цели были измеримыми. Большинство целей в начале главы удовлетворяет этому условию. «Достаточно средств к существованию» можно выразить более конкретно: моя пенсия должна составлять 70% моего теперешнего дохода. Изучить ядро Linux «лучше» - цель, плохо поддающаяся измерению. Я могу исправить этот недостаток, добавив какой-нибудь ориентир, например разработку простого драйвера устройства.



Возможно, устанавливая цели, вы захотите с кем-то посоветоваться. Это может быть член вашей семьи, священник, шеф, сосед, близкий друг или кто-то иной. У каждого читателя этой книги свой список людей, с которыми он захочет посоветоваться. Кто в списке у вас?

Собираясь записать свои жизненные цели, подумайте о своей системе ценностей. В чем, по-вашему, заключается ваше предназначение? Хотите ли вы помогать людям или считаете, что каждый должен решать свои проблемы сам? Хотите ли вы быть богатым или счастливым (а, может, и тем и другим одновременно)? Цените ли вы независимость или сотрудничество? Что для вас важнее — общественное или личное?

У вас могут быть разные системы ценностей на работе и дома. Дома вы можете быть заботливым родителем, который в равной мере любит и поддерживает всех своих детей. А на работе все может зависеть от конкуренции или распределения ролей.

Постановка целей

Книга «How to Get Control of Your Time and Your Life» (Как управлять своим временем и своей жизнью), Signet, - классика тайм-менеджмента. В ней доказывается необходимость списков краткосрочных, среднесрочных и долгосрочных целей и рекомендуется разбивать их на категории А, В и С по степени важности (категория А имеет наивысший приоритет).

Давайте так и поступим.

Возьмите большой лист бумаги и расчертите его на шесть частей, как показано на рис. 7.1 (тем, у кого много целей или размашистый почерк, возможно, потребуется несколько листов).

Вы собираетесь заполнить все прямоугольники списками соответствующих целей. Можете расставить другие временные ориентиры, если считаете, что ваши цели должны быть сгруппированы иначе.

Теперь составьте списки целей. Чтобы вам было легче начать, приведу некоторые советы:

1 месяц

Обычно сюда заносятся небольшие проекты, над которыми вы сейчас размышляете. Завершение текущих проектов, замена части оборудования и т. д.

1 год

Это более крупные проекты. Как правило, они связаны с реорганизацией, как технической («заменить текущую службу каталогов единой системой регистрации»), так и структурной («разбить груп-

	Профессиональные	Личные
1 месяц		
1 год		
5 лет		

Рис. 7.1. Лист для постановки целей

пу системного администрирования на несколько команд быстрого реагирования на запросы клиентов»).

5 лет

Сюда записывайте долгосрочные проекты, связанные с переменами в карьере («сдать экзамен МВА и перейти на руководящую должность») или в личной жизни («жениться»).

Не беспокойтесь о порядке, в котором вы записываете цели, и о том, достаточно ли хороши формулировки, чтобы их кому-нибудь показать. Просто перечислите их. Я подожду.

Честное слово, я подожду. Не читайте дальше, пока не заполните все графы. Не в уме, а на бумаге.

Вы все еще ничего не написали, а? Вы подумали, что вернетесь к этой главе и заполните таблицу позже. У всех упражнений в этой книге есть одна общая черта: от них нет пользы, пока вы их не выполнили. Итак, достаньте лист бумаги и приступайте!

Я подожду...

Честное слово...

Ну что, готово? Хорошо.

Теперь просмотрите список, чтобы убедиться, что каждая цель измерима. Может ли посторонний человек оценить ситуацию и определить, что цель достигнута? Сможете ли вы представить цифры и осязаемые результаты в качестве доказательства достижения цели? Перечитайте список и проверьте. Я подожду еще.

Теперь у каждой цели проставьте приоритет - А, В или С. Приоритет А означает, что вы безоговорочно должны добиться данной цели. Приоритет В присваивайте самым важным из оставшихся целей. Приоритет С ставьте рядом с целями, которых «неплохо бы» достичь. Это, скорее, хорошие идеи, чем важные дела. Схема полностью аналогична приведенной в главе 4.

Расставьте приоритеты прямо сейчас. Я не смогу убедить вас в важности этих упражнений, пока вы сами это не поймете.

Все оказалось не так просто, как вы ожидали, не правда ли? Вам захотелось поставить «А» у каждого пункта? Лично мне хотелось. Расстановка приоритетов может потребовать столько же времени, если не больше, чем составление самого списка.



Возможно, вы захотите расписать свои «жизненные цели», например, где вы хотите жить, выйдя на пенсию, и какое финансовое положение хотите при этом иметь. Проводя аналогию с тем, как сложные проценты вклада увеличивают сумму на вашем счете, скажу: чем раньше вы приступите к финансовому планированию, тем лучше.

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

- Каждый вечер изучать книгу в течение часа, пока не прочитаю до конца.
- Написать нетривиальную программу с использованием полученных знаний.
- Я хочу, чтобы через 24 часа у меня все было постирано и поглажено.
 - Купить стиральный порошок.
 - Постирать белье в стиральной машине.
 - Высушить белье.
 - Погладить белье и убрать его в шкаф.
- Иногда мне хочется назначить свидание порнозвезде.
 - Бывать в тех местах, где больше всего шансов познакомиться с порнозвездой.
 - Выяснить, где находятся такие места.

(Вы, вероятно, заметили, что некоторые шаги расположены в странном порядке. Как я уже говорил, иногда приходится начинать с конца.)

Не следует относиться к этим записям как к высеченным на камне. Нередко мы обнаруживаем самые неожиданные промежуточные цели на пути к основной. Дороти так хотела встретиться с Волшебником страны Оз, чтобы он помог ей вернуться домой. Однако, встретив его, она узнала, что прежде чем она сможет вернуться, ей предстоит еще многое сделать. Так и в жизни.

Занесение шагов в расписание

Теперь, поняв, чего вы хотите достичь и каковы шаги на пути к вашей цели, можно занести их в календарь наравне с другими делами.

Благодаря рабочему графику или семейным обязанностям, в какое-то время года или месяца вы загружены меньше обычного. Выберите наиболее удобное для вас время. (Если в какой-то день вам не удалось добиться намеченного, не расстраивайтесь. Система Цикл позволяет вам переносить дела на следующий день.)

Посмотрите на список шагов и подумайте, в каком порядке их лучше всего выполнять. Занесите один-два пункта из каждого списка в список дел на самый подходящий день. Возьмем в качестве примера изучение Linux. Предположим, сегодня понедельник. Я записываю пункт «Попросить рекомендации» в сегодняшний список дел. К среде я, скорее всего, получу какие-то сведения, поэтому на страничке, относящейся к среде, я записываю «Купить книгу, которую мне порекомендовали». Я не настолько расточителен, чтобы переплачивать Амазону за доставку книги в течение суток, поэтому пункт «Час чтения книги по Linux» я занесу в список дел на понедельник. То же самое я запишу и на других страничках со вторника по пятницу. Если я пользуюсь портатив-

ным компьютером, я вызываю функцию копирования пункта списка дел. Я не знаю, сколько времени у меня уйдет на чтение книги, но могу наметить последний пункт списка (написание программы с использованием новых знаний) на какой-то день через неделю. Если я не занесу эти пункты в список дел, они вообще никогда не будут выполнены.

Насчет стирки в течение 24 часов я пошутил; это, скорее, обычное дело, чем жизненная цель. Тем не менее, это хороший пример того, как следует связывать цели со списком дел. Первые три шага я записываю в месте, отведенном для дел, намеченных на вечер. Последний пункт («Погладить белье и убрать его шкаф») я заносу в список дел на следующий день.

Если у вас много целей, описанный процесс может показаться устрашающим. В таком случае вам следует либо распределить цели по более продолжительному интервалу времени, либо понизить некоторые приоритеты.

Имея портативный компьютер, легко намечать дела на отдаленное будущее. Однако я заметил, что лучше этого не делать. Эта запись может потеряться, либо я могу к тому времени забыть, к чему она относится. Для каждой цели я планирую только следующий шаг, максимум два или три. Завершив очередной шаг, я лучше представляю, сколько времени потребуется на следующие.

Бумажный органайзер вмещает ограниченное количество страничек. Как правило, вы будете вставлять только блок для следующего месяца. Поэтому у вас не получится планировать отдаленное будущее. Вы можете записывать шаги в календаре, как встречи и совещания. Пометьте себе, что следующий шаг должен начаться через три месяца. Например, напишите «Узнать, какая бывает страховка».

Достоинство этого метода в том, что вы никуда не спешите, но при этом неуклонно продвигаетесь к своей цели.

Прямо сейчас: запишите хотя бы один шаг из каждого списка в портативный компьютер или бумажный органайзер.

Регулярный просмотр списка целей

Вы взяли хороший старт. Однако вам нужен способ, который обеспечит системный подход. В первый день каждого месяца уделите некоторое время планированию своих целей. Закройте дверь (или найдите тихое место) и проделайте следующее:

- **Просмотр целей.** Просмотрите и обновите ваш список целей. Вычеркните достигнутые цели. Если с момента последнего просмотра у вас появились какие-то новые цели, решите, действительно ли стоит их добиваться. Если это так, присвойте им приоритеты. Возможно, после появления новых целей вам захочется изменить приоритеты прежних.

- **Просмотр шагов.** Просмотрите и обновите список следующих шагов. По мере выполнения шагов, как и раньше, заносите новые в список дел.

Со временем вы научитесь распределять записи о шагах по календарю. Я стараюсь заносить их в список дел на понедельник. Тогда при планировании недели я смогу передвинуть их на другой день недели, если понадобится.

Как не забыть об этом? Установите в органайзере повторяющееся событие «Просмотр целей и шагов». Пусть оно повторяется первого числа каждого месяца. Теперь у вас есть постоянный механизм напоминания об этом процессе.

Если вы пользуетесь бумажным органайзером, вставьте в него страничку с повторяющимися событиями и просматривайте ее в начале каждого месяца. Всякий раз, вставляя в органайзер блок страниц на следующий месяц, я просматриваю листок с повторяющимися событиями и с его помощью отмечаю в своем календаре шаги, ведущие к различным целям.

Резюме

- Для достижения долгосрочных целей вам нужно сформулировать их и действовать по соответствующим направлениям.
- Не записав свои цели, вы так и будете топтаться на месте в надежде, что вам повезет.
- Цели должны быть измеряемыми: результат должен быть осязаемым или выражаться в цифрах так, чтобы другой человек мог его проверить.
- У целей должны быть указаны сроки: если вы знаете, когда нужно достичь цели, это помогает вам придерживаться определенного ритма.
- Начните с перечисления целей, которых вы хотите достичь на работе и в личной жизни через месяц, через год и через пять лет. Расставьте приоритеты. Составьте список шагов, необходимых для достижения ваших целей. Занесите следующий шаг для каждой цели в список дел на будущее. Раз в месяц просматривайте цели и шаги, меняйте приоритеты, если надо, и заносите следующие шаги в список дел.
- Выполняйте следующие шаги, как любые другие дела из списка. Со временем каждая цель будет достигнута или будет под контролем.
- Регулярно просматривайте список целей. Добавляйте новые и удаляйте достигнутые. Обрабатывайте шаги соответствующим образом.

8

Расстановка приоритетов

В этой главе описывается процесс расстановки приоритетов «снизу вверх». Вначале я уточню то, чего поверхностно коснулся в главе 5, - расстановка приоритетов в сегодняшнем списке дел. Затем я собираюсь обсудить приоритеты более крупных проектов. Наконец, мы поговорим о том, как управлять шефом.



Расстановка приоритетов в списке дел

Итак, вы сидите за рабочим столом, и перед вами список сегодняшних дел. Десятки пунктов. Как вы определите, с чего начать?

Этот раздел посвящен расстановке приоритетов в этом списке. В разных ситуациях приходится использовать разные подходы. В предыдущих главах мы придерживались очень простой схемы: если дело нужно выполнить сегодня, оно имеет приоритет А; если его нужно закончить скоро (но не сегодня), оно получает приоритет В; все остальные дела имеют приоритет С.

«А что делать, если все дела в списке имеют приоритет А?»

Прочитать эту главу.

Выполнение дел в порядке их следования в списке

Я часто слышу от системных администраторов, что у них уйма времени уходит на размышления о том, что делать дальше. Я и сам замечал, что могу потратить пять и больше минут только на то, чтобы читать список дел и думать, какой пункт выполнить следующим. Если суммировать все это время, получится не так уж мало.

Если вы теряете время на раздумья о том, что делать дальше, остановитесь. Упростите процедуру принятия решений до предела: просто начните с первого пункта списка и двигайтесь сверху вниз. За то время, что вы сидите и размышляете, вы смогли бы выполнить пару простых дел. Кроме того, в результате переноса невыполненных пунктов на следующий день более старые пункты всплывают к началу списка. Если вы выполните их в первую очередь, это будет хорошим началом дня.



В детстве моей домашней обязанностью было выносить мусор вечером в понедельник и четверг. Я это ненавидел. Я жаловался на судьбу, откладывал до последнего и всячески пытался избежать этой обязанности. (Думаю, все дети так же относятся к домашним обязанностям.) Хоть это и был большой, трехэтажный викторианский дом, мне хватило бы десяти минут на то, чтобы опорожнить все мусорные корзины. Но вот что занятно! Прежде чем взяться за дело, я изобретал тысячу способов задержаться, теряя при этом полчаса. Есть много ситуаций, когда собственно на дело уходит гораздо меньше времени и усилий, чем на попытки его избежать.

Выполнять дела в том порядке, в каком они идут в вашем списке, гораздо лучше, чем откладывать их на потом. Приведу лозунг фирмы Nike: «Просто сделай это».

Если ваш список настолько короткий, что вы можете переделать все дела за один день, такой подход тем более оправдан. Если неважно, в какое время дня вы выполнили ту или иную задачу, не все ли равно, в каком порядке вы их выполняли?

Здесь уместна аналогия с загруженностью компьютерной сети. Когда нагрузка невелика, службы, связанные с телефонией, не испытывают проблем. Но когда нагрузка в сети возрастает, эти службы работают лучше, если принята одна из более сложных схем расстановки приоритетов или система оценки качества обслуживания (quality of service, QoS). При малой нагрузке годится любая схема. При большой требуется что-то более структурированное. Если список дел невелик, можно расставлять приоритеты как угодно. Если мы завалены делами, нам потребуется схема посложнее.

Развивая аналогию, хочу спросить вас: знаете ли вы, что система оценки качества обслуживания часто сводится не к тому, что какие-то пакеты обслуживаются лучше? Фактически она сводится к тому, что некоторые пакеты обслуживаются хуже! Если разобраться, внутри системы оценки качества обслуживания происходят довольно любопытные вещи. При небольшой загрузке сеть работает так, словно никакой системы оценки качества нет. Пакеты приходят и уходят. Но когда нагрузка возрастает, отсутствие системы оценки качества приводит к отбрасыванию любого пакета, поступившего последним. Иными словами, в буфере нет места для новых пакетов, и они игнорируются. Совсем другое дело, если работает система оценки качества обслуживания. Когда буфер переполнен, последний пакет не отбрасывается; вместо этого из буфера выкидывается пакет с более низким приоритетом. То есть, платя интернет-провайдеру за лучшее обслуживание при определенном трафике, вы платите за то, чтобы вас не выкинули при повышенной нагрузке. Вы буквально даете провайдеру взятку, чтобы, когда нагрузка в сети возрастает, он выкидывал пакет другого клиента!

То же самое происходит при расстановке приоритетов ваших заданий. В вашем распоряжении ограниченные ресурсы и ограниченный запас времени. Если мы завалены работой, то начинаем недовольно ворчать по поводу любого нового запроса. В действительности же нам следует просмотреть список текущих дел и определить, нет ли там пунктов, выполнение которых можно отсрочить. (К сожалению, взятки нам не светят!)

Расстановка приоритетов с учетом ожиданий клиентов

Вот маленький секрет, который я узнал от Ральфа Лоуры (Ralph Loura), когда он был моим начальником в Bell Labs. Если у вас имеется список дел, то при любом порядке их выполнения будет затрачено (примерно) одно и то же время. Однако если вы будете выполнять их в том порядке, какого ждут от вас клиенты, они будут считать, что вы работаете быстро. При том же объеме работы для вас больше понимания у клиентов. Круто, а?!

Чего же ждут от вас клиенты? Разумеется, каждый из них хочет, чтобы его запрос был выполнен немедленно, тем не менее, отдавая себе отчет в том, что на все требуется время. Представление может быть далеким от реальности часто по причине непонимания компьютерных технологий, но все же мы можем разбить ожидания пользователей на несколько категорий:

- **Некоторые запросы должны выполняться быстро.** В качестве примеров можно привести восстановление пароля, выделение IP-адреса и удаление защищенного файла. У таких запросов есть одна общая особенность. Как правило, это малозначительные дела, которые задерживают выполнение чего-то серьезного (со стороны пользователя). Представьте себе раздражение пользователя, который не

- может приступить к работе, пока не восстановлен пароль, а вы выполняете его просьбу лишь через несколько часов.
- **Срочные задачи, включающие период ожидания («Поспешите и подождите»), должны начинаться как можно раньше.** Задачи, выполнение которых предшествует другим задачам, должны, по мнению пользователей, начинаться безотлагательно. Например, заказ какой-нибудь детали включает в себя собственно размещение заказа и долгий период ожидания поставки. Только после этого вышедшую из строя деталь можно заменить на новую. Если ожидание длится две недели, то пользователь рассчитывает, что вы быстро сделаете заказ, и две недели не растянутся на три.
 - **Некоторые запросы выполняются долго.** Сюда относятся установка нового компьютера, создание службы «с нуля» и вообще все, что связано с поставкой товаров. Даже если продавец обещает доставку в течение суток, люди понимают, что их запрос нельзя выполнить немедленно.
 - **При аварийной ситуации вся остальная работа прекращается.** Последняя категория - аварийные ситуации. Пользователи не только считают, что в таких случаях приостанавливается любая другая работа, но и верят в то, что все системные администраторы заняты ликвидацией аварии. Как правило, пользователи не знают, что в группе системных администраторов есть разделение труда.

Теперь, лучше представляя ожидания клиентов, вы можете воспользоваться этим знанием. Как? Предположим, ваш список дел выглядит так, как показано на рис. 8.1.

Выполнив эти дела в том порядке, в каком они перечислены, вы, скорее всего, будете весьма довольны своей производительностью. Вы сделали все в тот день, когда требовалось, и у вас ушло шесть с половиной часов (плюс час на обед). Очень неплохо для вас.

Задача	Описание	Ожидаемое время работы	Фактическое время работы	Время окончания
T1	Восстановить пароль	1 минута	10 минут	9:10
T2	Создать новую учетную запись	На следующий день	20 минут	9:30
T3	Установить новый сервер	На следующий день	4 часа (+ 1 час на обед)	14:30
T4	Добавить новую CGI-область в веб-сервер	1 час	30 минут	15:00
T5	Заказать программный пакет	1 час	1 час	16:00
T6	Исправить незначительную ошибку в NetNews	10 минут	25 минут	16:25
T7	Выделить IP-адрес	2 минуты	5 минут	16:30

Рис. 8.1. Задачи, не упорядоченные в соответствии с ожиданиями пользователей

Однако ваша деятельность разошлась с представлениями клиентов о том, сколько времени должно уйти на решение их проблем. Сотрудник, сделавший запрос T7, был вынужден целый день ждать того, на что, по его мнению, требуется пара минут. Если бы этим клиентом был я, то вспылал бы. Из-за отсутствия IP-адреса на целый день задержалась установка нового лабораторного оборудования.

(На самом деле, скорее всего, раздраженный нетерпеливый клиент не стал бы ждать целый день. Он «пинговал» бы IP-адреса до тех пор, пока не нашел свободный на тот момент и не воспользовался бы им. Мог случиться какой-нибудь конфликт, и вся система перестала бы работать, и вы потеряли бы на этом целый день. Впрочем, я отвлекся...).

Давайте изменим порядок выполнения дел с учетом представления клиентов о том, сколько времени требует та или иная задача. Задачи, о которых известно, что они решаются быстро, мы переставим в начало и выполним с утра. Остальные можно выполнить позже. Из этого правила мы сделаем одно исключение, в чем вы вскоре убедитесь. Список с новым порядком задач представлен на рис. 8.2.

Вы начинаете свой день с выполнения задач T1 и T7, потому что клиенты ожидают их скорого выполнения, и еще потому, что эти задачи, скорее всего, тормозят крупные проекты. Вы, конечно, уложите в предполагаемые сроки выполнения этих задач.

Следующая задача (T5) относится к категории «Поспеши и подожди». Независимо от того, насколько быстро вы оформите заказ, он будет выполнен только через день-два. Однако если вы не будете тянуть с заказом, то избежите претензий со стороны клиента.

Следующая задача (T4) выполняется за 30 минут. Можете проверить.

Задача T2 не требует много времени, но ожидание ее выполнения выражается в сроках, а не в часах и минутах. Если завтра к работе при-

Задача	Описание	Ожидаемое время работы	Фактическое время работы	Время окончания
T1	Восстановить пароль	1 минута	10 минут	9:10
T7	Выделить IP-адрес	2 минуты	5 минут	9:15
T5	Заказать программный пакет	1 час	1 час	10:15
T4	Добавить новую CGI-область в веб-сервер	1 час	30 минут	10:45
T2	Создать новую учетную запись	На следующий день	20 минут	11:05
T3	Установить новый сервер	На следующий день	4 часа (+ 1 час на обед)	16:05
T6	Исправить незначительную ошибку в NetNews	10 минут	25 минут	16:30

Рис. 8.2. Задачи, упорядоченные в соответствии с ожиданиями пользователей

ступает новый сотрудник, ожидается, что к его приходу учетная запись будет создана независимо от того, создадите вы ее за минуту или за день, утром или вечером. Поскольку для задачи установлен крайний срок, важно, чтобы она просто была выполнена.

Если бы возникла аварийная ситуация (например, из-за того, что два хоста оказались сконфигурированы с одним IP-адресом) и вся работа остановилась, то наше расписание было бы нарушено. Тем не менее я все равно постарался бы создать учетную запись до появления нового сотрудника. Другие дела пришлось бы отложить на день, и это не вызвало бы непонимания клиентов в условиях аварийной ситуации. Но ради создания учетной записи я готов остаться на работе допоздна, чтобы не нарушить срок.

Установку нового сервера (Т3) я отношу к категории «черных дыр». По идее, требуется несколько минут на монтаж оборудования, около часа на установку операционной системы и чуть больше часа на ее конфигурирование. По крайней мере, так считают производители. Мы-то, системные администраторы, знаем, что это нереально. В первый раз монтируя незнакомое устройство, вы не один час потратите на то, чтобы разобраться со своеобразной системой монтажа, изобретенной производителем. Процесс установки операционной системы на сервере, как правило, включает в себя ее многократную установку, настройку и последующее удаление, пока не будет достигнута идеальная конфигурация. (Этот ящик будет стоять тут несколько лет, так что имеет смысл потратить время на то, чтобы довести все до ума.) Конфигурирование тоже никогда не укладывается в предполагаемые рамки. Следовательно, браться за задачи типа «черная дыра» следует только после успешного завершения задач, время выполнения которых прогнозируется точнее.

Присваивая приоритет задаче Т6, мы немного отклонились от правил. Учитывая ожидаемое время выполнения этой задачи, можно было ожидать, что я запланирую ее на более раннее время - где-нибудь до задачи Т3. Однако на всех сайтах, с которыми я работал, сетевые новости Usenet имеют низкий приоритет, и работа с ними считается очень простой, чем-то вроде подарка для сотрудника. (Я никогда не встречал интернет-провайдера, у которого ситуация была бы иной.) Таким образом, решение низкоприоритетного вопроса переносится в конец списка. Общее правило гласит: если все согласны с низким приоритетом какой-то задачи (или он установлен руководством), то ее следует поместить в конец списка. Задумайтесь: если кто-то пожалуется, что одна из других задач не выполнена, как вы будете смотреть в глаза шефу, объясняя, что отложили клиентский запрос, занимаясь устранением небольшой ошибки в Usenet? Нет, только не это!

Просто? Да. Вам потребуется какая-то практика, но клиенты быстро заметят разницу.

Делегируйте, регистрируйте или действуйте

Когда я излагаю эту систему, основное возражение, выдвигаемое моими слушателями, заключается в том, что их список дел меняется. В начале рабочего дня у них нет фиксированного списка дел, которые надо сделать. Новые дела добавляются в список на протяжении всего дня.

В таком случае следует применять технологию «Делегируйте, регистрируйте или действуйте», описанную в главе 2, где речь шла о прерываниях. Решая, как поступить, вы можете руководствоваться ожиданиями клиентов.

Запрос на восстановление пароля должен быть выполнен быстро, поскольку он задерживает другую работу. Следовательно, лучше сделать это самому, чем делегировать запрос другому сотруднику. И уж, конечно, не стоит регистрировать этот запрос, откладывая его выполнение и расстраивая планы клиента.

Взаимная защита от прерываний

Описанные приемы годятся не только для расстановки приоритетов в вашем личном списке дел, но и для более масштабного планирования. Воспользуйтесь ими для организации работы всего отдела компьютерной поддержки!

Помните технику взаимной защиты от прерываний, описанную в главе 1? По сути, она позволяет вам действовать с учетом ожиданий клиентов. Ваш коллега перехватывает все прерывания в течение половины дня, чтобы вы могли поработать над проектами, а во второй половине дня вы меняетесь ролями. Фактически этот подход обеспечивает постоянное наличие системного администратора, готового снять вопросы, решение которых, по мнению клиентов, не займет много времени.

Большинство служб технической поддержки имеют двухуровневую структуру. Сотрудники первого уровня отвечают на телефонные звонки и переадресуют их на второй уровень, только если не в состоянии решить вопрос самостоятельно. Это не что иное, как вариант взаимной защиты от прерываний для всей команды, который обеспечивает время отклика, соответствующее ожиданиям пользователей!

Расстановка приоритетов, основанная на ожиданиях клиентов, и применение взаимной защиты от прерываний копируют работу службы технической поддержки, что свидетельствует в пользу предложенного мной подхода. А, может, как раз наоборот: двухуровневая структура службы технической поддержки хороша именно потому, что учитывает ожидания пользователей. В любом случае это круто, не так ли?

Приоритеты проектов

В предыдущих разделах были описаны способы расстановки приоритетов отдельных задач. Теперь я приведу некоторые полезные приемы присваивания приоритетов проектам.

Достижение наибольшего эффекта

Предположим, вы со своими коллегами-сисадминами по итогам мозгового штурма запланировали на следующий год двадцать замечательных проектов. Однако выделенного бюджета и имеющегося в вашем распоряжении персонала недостаточно для выполнения всех проектов. Какие выбрать?

Обычно я считаю, что получается лучше, когда выбираешь проекты, сулящие «максимальный эффект».

Заманчиво начать с самых легких проектов. Вы знаете, что делать; по поводу этих проектов нет разногласий, так что вы, по крайней мере, уверены, что они будут завершены.

Также заманчиво взяться за проекты, интересные лично вам, или за политически безопасные проекты, или за проекты, являющиеся логическим продолжением прежних проектов.

Устояв перед этими соблазнами, определите проекты, которые будут иметь максимальный положительный эффект в вашей организации. По сути, я утверждаю, что лучше осуществить один крупный проект со значительным положительным эффектом, чем несколько поверхностных проектов. Я убеждался в этом неоднократно. Когда вся команда работает над одним проектом, дело идет лучше, чем если бы каждый вел отдельный проект. Это происходит, потому что вместе мы работаем лучше.

Можно взглянуть на ситуацию с другой точки зрения. Все проекты можно разбить на четыре категории, как показано на рис. 8.3.

Очевидно, что начинать следует с проектов категории А. Простые проекты, имеющие значительный эффект, встречаются редко, и если у вас волшебным образом появляется такая работа, ее надо выполнять без раздумий. (Предупреждение: будьте бдительны, принадлежность проекта к категории А может быть кажущейся.)

Столь же очевидно, что нужно избегать проектов категории D. За трудоемкий проект, который ничего, в сущности, не изменит, не стоит и браться.

	Простые (требуют мало усилий)	Сложные (требуют много усилий)
Значительный положительный эффект	A	B
Поверхностный эффект	C	D

Рис. 8.3. Эффект проекта и затрачиваемые усилия

Однако большинство проектов принадлежит к категории В или С, а человек по натуре склонен к более легким проектам С. Вы можете заполнить весь год простыми проектами, отчитаться об их выполнении и выглядеть героем. В то же время в наиболее успешных компаниях применяется практика поощрения сотрудников, берущихся за проекты категории В - трудные, но необходимые.

Переформулируем вопрос в терминах возврата на инвестиции (return on investment, ROI), чтобы подтвердить разумность описанного подхода. Предположим, вы собираетесь в будущем году потратить определенную сумму. Вложите ли вы деньги в большое количество несущественных проектов? Нет, вы постараетесь найти более эффективный проект и инвестируете средства в него.

Важно убедиться, что выбранный вами проект скоординирован с задачами вашей фирмы. Это важно как для компании, так и для вас. В результате вас будут выше ценить.

Запросы, поступающие от начальства

Если шеф просит вас что-либо сделать, и это простая задача (а не крупный проект), сделайте это немедленно. Например, если шеф попросил вас выяснить, сколько (примерно) имеется компьютеров со старой версией Windows, сообщите ему это через несколько минут.

Здесь полезно представлять картину в целом. Как правило, начальство дает подобные задания, занимаясь долгосрочным планированием или бюджетом, и вы можете задержать шефа на целый день, если не дадите быстрый ответ. Не исключено, что он оценивает материальные и трудовые затраты для перевода всех компьютеров на последнюю версию Windows. Целый проект повиснет в ожидании вашего ответа.

Почему это важно? Во-первых, ваш шеф решает, когда повышать вам зарплату. Стоит ли продолжать?

Пожалуй, да. В распоряжении вашего шефа имеется фиксированная сумма, которую он может распределять на повышения. Если он больше выделит Монике, то Ларри достанется меньше. Вам нужно, чтобы, просматривая список сотрудников и встретив там вашу фамилию, он подумал: «Он так быстро сообщил мне, сколько у нас компьютеров со старой версией Windows. Он вообще всегда быстро делает то, что я прошу». Не хотите же вы, в самом деле, чтобы он подумал: «Помнится, работа над бюджетом весь день стояла, пока я ждал ту статистику». Или того хуже: «Сорвав сроки, я упал в глазах шефа, а все потому, что такой-то никогда не торопится предоставить мне нужную информацию. Не видать такому-то повышения в этом году».

Как управлять шефом

Многие считают такое управление улицей с односторонним движением. Я думаю иначе. Управление - это взаимоотношения, и их развитие

отчасти зависит от вас. Без хороших отношений с шефом трудно чего-то добиться или сделать карьеру. И наоборот, хорошие отношения позволят вам достичь большего в профессиональном отношении, получить удовлетворение от работы и ускорить продвижение по службе.

Поискав в Сети фразу «как управлять своим боссом», вы получите множество ссылок на прекрасные статьи и книги. Значит, многие понимают, что нужно улучшать взаимоотношения со своим руководством. Запланируйте время на чтение хотя бы некоторых из этих статей.

Я считаю, что три ключевых момента в управлении шефом - получать от него помощь в продвижении по службе, правильно делегировать наверх задачи, а также понимать его цели и участвовать в их достижении.

Доведите до сведения своего шефа ваши карьерные цели

Проинформируйте шефа, кем вы хотите стать через два года, или через пять, или через десять лет. Ваш начальник не экстрасенс. Скорее всего, если вы просто хорошо работаете, он может думать, что вас все устраивает. Ему проще считать, что все подчиненные довольны положением дел. Но вы-то предпочли бы двигаться по служебной лестнице, не так ли? Вы хотите занять ведущую позицию в группе или перейти на руководящую должность, хотя, возможно, вы работаете системным администратором, просто чтобы заработать на учебу и стать художником, актером или писателем.

Важно, чтобы шефу были известны ваши цели. Вот и расскажите ему о своих целях и мечтах. Не стесняйтесь, но и не будьте, как заезженная пластинка. Раз в год в беседе с глазу на глаз сообщайте ему о том, какой вы видите свою дальнейшую карьеру. Помнится, зайдя в кабинет к Лесу Ллойд (Les Lloyd), я сказал: «Лес, может, я пока и новичок, но в будущем хочу стать одним из менеджеров вашего компьютерного центра». Минуту подумав, он рассказал, чего я должен достичь в профессиональном отношении, чтобы он смог считать меня кандидатом на эту должность. В то лето я работал не покладая рук и вскоре получил повышение. Несколько похожих историй у меня было и в других фирмах, где я работал после университета.

Делегируйте наверх задачи, только если требуются полномочия вашего шефа

Делегирование наверх означает перекладывание какой-то задачи на вашего шефа. Секрет в том, чтобы знать, когда это можно делать, а когда нет. Если вы попытаетесь делегировать шефу задачу в неподходящий момент, это будет выглядеть так, словно вы уваливаете от работы.

Вот подходящий случай: вам не удастся убедить клиентку в том, что ее отдел должен платить за сервер. К примеру, она считает, что платить должен ваш отдел компьютерной поддержки, либо вообще не понимает, зачем нужен сервер. Здесь имеет смысл обратиться за помощью к шефу,

Решайте проблемы на соответствующем уровне

Не обсуждайте технические вопросы с вице-президентом фирмы и не решайте политические проблемы с технологами.

Руководители, как правило, хотят иметь дело с руководителями того же уровня. Вице-президенты общаются с вице-президентами. Директора достигают наилучшего взаимопонимания с другими директорами. Если вы хотите пересечь эти горизонталы, свяжитесь с сотрудником вашего уровня, подчиненным тому, с кем вам нужно переговорить. А еще лучше - обратитесь к своему шефу, который гораздо лучше ориентируется в структуре организации.

потому что здесь он может применить свои полномочия. Он может решать деловые вопросы, тогда как ваши полномочия относятся к техническим вопросам. А еще благодаря своим полномочиям ваш шеф понимает политическую ситуацию в фирме. Возможно, он знает, что эту клиентку не переубедить, либо каким-то образом решит вопрос на более высоком уровне.

Чтобы не оставалось неясных моментов, рассмотрим ситуацию, в которой делегирование наверх неуместно. Пусть сервер уже куплен и ждет, когда его установят. Стоит ли просить шефа, чтобы он установил сервер? Вообще говоря, нет. Здесь не требуются его полномочия. У вашего шефа может быть техническое образование и достаточно опыта, чтобы установить сервер быстрее, чем это сделаете вы, но если уж он поручил это дело вам, попытка переложить его на шефа будет выглядеть как уклонение от ответственности.

С другой стороны, если вы не знаете, как установить этот сервер, допустимо обратиться к шефу за помощью. В этом случае вы просите его не выполнить работу, а научить вас. Здесь *действительно* требуются его полномочия. Основная функция руководителя состоит в распределении ресурсов. Он решит, заняться ли ему вашим обучением лично или поручить это другому сотруднику. Попросив научить вас устанавливать сервер, вы поступаете правильно, потому что фактически просите шефа выделить ресурсы на ваше обучение.

Понимайте цели своего шефа и помогайте ему в их достижении

Если вы хотите, чтобы шеф помогал вам, вы должны помогать ему. «Почему это? Это ведь его работа - решать вопросы вроде управления карьерой, так?» Конечно, в техническом смысле это так, но медом вы приманите больше мух, чем уксусом. Ваш руководитель станет энергичнее продвигать вас к успеху, если увидит, что вы помогаете его продвижению.

Уточняю: в данном случае успех означает соответствие целям шефа. Я говорил, что ваш шеф не экстрасенс, и не стоит рассчитывать на то, что он догадается о ваших карьерных планах. И вы не пытайтесь разгадать его планы с помощью телепатии. Чтобы сориентироваться, прислушайтесь к более опытным коллегам, возможно, им известны мотивы вашего шефа. Но лучше всего поговорить с самим шефом.

Несколько раз начальники удивляли меня, когда я наедине спрашивал, каковы их цели и чем я могу помочь в достижении этих целей. Обычно я формулировал вопрос следующим образом: «По каким критериям Ваш шеф оценивает Вашу работу? Зная эти критерии и ориентируясь на них, я смогу эффективнее организовать работу команды».

В одном случае шеф перечислил конкретные технические проекты, которые ему нужно завершить в этом году. Оказалось, что он «продал» их вышестоящему руководству, и оно ожидает их завершения. Я вскоре понял, что значительная часть моей работы не имеет никакого отношения к этим проектам, и изменил приоритеты своей деятельности, чтобы способствовать успеху шефа. Он заметил это и не остался в долгу.

В другой раз шеф сообщил мне критерии, определяющие размер его премии в конце года. Сначала я подумал: «Что за меркантильность!», а потом: «Кто я такой, чтобы судить?» И расставил приоритеты в своей работе так, чтобы эти критерии были удовлетворены. В сущности, моя цель состояла в том, чтобы шеф получил максимальную премию. Это позволило бы мне с большей вероятностью получить то, что я хочу, будь то продвижение по службе, повышение оклада или новый супер-пупер-компьютер на рабочем месте. Разве это неэтично? Конечно, нет (при условии, что я не совершаю неэтичных поступков, двигаясь к цели).

Тут может возникнуть любопытная проблема. Что, если критерии оценки деятельности моего шефа не соответствуют интересам фирмы, как я их понимаю? Например, я считаю, что фирме выгоднее повысить качество продукции, а работу шефа оценивают по росту объема выпуска. Придется доверять руководителям шефа, которые эти критерии устанавливали. Как вариант, можно стремиться к обоим целям. Трудно? Если вы настолько умны, что видите выгоду фирмы лучше, чем начальники вашего начальника, то у вас не должно быть проблем с поиском пути, ведущего к обоим целям сразу.

Я не считаю циничным подход, при котором вы даете руководству именно то, что оно просит. Однако порой критерии оценивания вашего шефа непреднамеренно провоцируют его на плохую работу. Как-то раз мне довелось посетить некий отдел технической поддержки, начальник которого получал премию за снижение среднего времени проявления реакции на запрос клиента. (Вы уже поняли, к чему это приводило?) Сотрудники отдела хватали трубку после первого гудка (или быстро считывали запрос, поступивший в базу данных) и просили клиента подождать. Обслуживание не улучшалось, но работа коллектива удовлетворяла критериям. На следующий год руководство стало изме-

рять производительность по среднему времени решения проблемы. Как вы могли догадаться, запросы закрывались довольно быстро независимо от того, были ли они удовлетворены в действительности. Типичный разговор с клиентом был таким: «Я закрою этот запрос; если мое предложение не поможет Вам решить проблему, перезвоните». И вновь клиенты оставались недовольными.

Если руководители управляют делами на основе одних только числовых показателей, но некомпетентны в создании показателей, стимулирующих желаемое поведение подчиненных, то таким руководителям следует либо научиться разрабатывать подходящие показатели, либо вовсе отказаться от их применения.



Моему другу довелось работать в фирме-провайдере, где работа отдела продаж оценивалась по количеству заказанных Т1-подключений, а отмененные впоследствии заказы не учитывались. Начальник отдела призывал всех искать клиентов, которые бы заключали контракт и отменяли его, когда дело дошло до установки оборудования. Неэтично? На первый взгляд да. Однако прошло больше года, а руководство не изменило свой критерий. Значит, оно не считало, что есть какие-то проблемы. Начальник и сотрудники отдела радовались, получая кучу наград за соответствие целям своего руководства. Разорился ли этот интернет-провайдер в конечном счете? Естественно. Интернет - самое подходящее место для разорения фирм с некомпетентным руководством.

Мой друг, спросив своего шефа о его целях, получил совершенно неожиданный ответ. Его шеф достиг предпенсионного возраста и хотел провести свой последний год тихо и без сюрпризов. Фактически речь шла о том, чтобы в течение следующего года работать поменьше, поскольку работа подразумевает появление новых проектов, а новые проекты - это новые риски. Шеф просто хотел дотянуть до пенсии. Мой друг понял, что у него есть три варианта: (1) отдохнуть год, (2) интенсивно работать, чтобы стать кандидатом на место шефа и (3) обновить свое резюме и начать поиски новой работы. Он выбрал первый вариант. Это был самый приятный его год в той фирме. Он провел его, посещая разнообразные курсы повышения квалификации и семинары. Обучение пошло ему впрок, и он получил место шефа, ушедшего на пенсию. Полагаю, в конце концов все сложилось удачно.

Если шеф увидит, что вы помогаете ему добиться успеха, перед вами откроется множество дверей. Шеф приложит больше усилий, помогая вам продвинуться по службе, вы все чаще будете первым получать предложения по участию во «вкусных» проектах и небольшие, но важные награды в виде нового оборудования. Естественно, это не мешает вам получать и более весомые повышения и премии. Важнее всего то, что благодарный шеф не забудет о вас, идя на повышение.

С этой точки зрения окончательным ориентиром в расстановке ваших приоритетов должен быть успех вашего шефа.

Действие является выражением приоритетов.

- Махатма Ганди

Резюме

- Если у вас много работы, расстановка приоритетов приобретает огромное значение. Правильные приоритеты еще более важны, если у вас физически не хватает времени на все дела. Если вы не очень загружены, годится любая схема расстановки приоритетов.
- Выполнение дел в порядке их следования в списке - хороший подход, когда дел немного. Поскольку более старые дела обычно «всплывают» к началу списка, вы их непременно выполните. Если вы не знаете, какое дело выполнять следующим, приступайте к следующему по порядку. Лучше выполнить первое дело в списке, чем терять время на размышления о том, какое из дел делать первым.
- Расстановка приоритетов в соответствии с ожиданиями клиентов подразумевает, что начинать следует с тех дел, которые, по мнению пользователей, занимают мало времени. Клиенты считают, что быстро должны выполняться те незначительные запросы, из-за которых задерживаются крупные проекты. Расставляя приоритеты по этой схеме, вы потратите столько же рабочего времени, сколько и при любой другой, а ваши клиенты будут довольны.
- Назначая одному из проектов наивысший приоритет, основывайте свой выбор на эффекте. Проект с максимальным эффектом, требующий значительных усилий для его завершения, полезнее для вас, чем более простой проект с малым эффектом.
- Запросам, исходящим от вашего шефа, следует назначить особый приоритет. Они нередко вызваны причинами, о которых вы и не подозреваете. Не допускайте, чтобы из-за вас задержался его более важный проект!
- Чтобы управлять своим шефом, вы должны делать три вещи: доводить до его сведения свои карьерные цели; делегировать наверх только те задачи, которые требуют применения его полномочий; понимать его цели и способствовать их достижению. Делая это, вы лучше согласуетесь со своим шефом, а он станет более гибко реагировать на ваши запросы, зная, что вы действуете в его интересах.

9

Как справиться со стрессом

У вас стресс? Ну, разумеется! Ведь вы - системный администратор!

Я не врач и не специалист по стрессу, но поделюсь с вами тем, что узнал за эти годы. Я подвержен стрессу - спросите любого, с кем я работал. Но, думается, сейчас я справляюсь с ним лучше, чем раньше.

В этой главе я расскажу о некоторых распространенных источниках стресса и о том, как с ними обращаться. Я дам несколько советов, как проводить свободное время, и расскажу о том, как я научился расслабляться. Эти советы я повторяю снова и снова как на лекциях, так и дружески беседуя с сослуживцами и коллегами-системными администраторами. По большей части это мои собственные выводы, а не результаты научных исследований. Не мудрствуя лукаво и не вдаваясь в излишние подробности, я дам вам некоторые полезные рекомендации и подскажу, где можно поискать дополнительную информацию.

Стресс - это напряжение, которое испытывают наше тело и психика при смене обстановки. Стресс бывает положительный и отрицательный. Положительный стресс связан с ожиданием чего-то хорошего и приятным волнением, стимулирует наше творчество, помогает нам победить в спортивном состязании или на выборах, позволяет достичь того, что раньше казалось невозможным. Отрицательный стресс разрушителен. Он приводит к сердечным заболеваниям, депрессии и ранней седине. Чтобы оставаться психически здоровым, важно уметь управлять стрессом.

Удивительно, но одна и та же ситуация может вызвать положительный стресс у одного человека и отрицательный у другого. Все зависит от нашего отношения к ситуации. Например, люди по-разному реагируют на комплименты начальства. Одним это приятно. Другие испытывают противоположные чувства. Они могут подумать: «Ох, теперь от меня всегда будут ждать каких-то достижений! Я не могу жить под

давлением!» Или: «Он похвалил меня при всех. Теперь все меня ненавидят!» При этом возникает страх, что другие сотрудники могут оказаться ревнивыми и мстительными.

Стресс вызывают не события вокруг нас, а наша реакция на них. В том, что касается эффективного управления стрессом, мне очень помогли две книги «Feeling Good: The New Mood Therapy» (Как чувствовать себя хорошо: Новая терапия настроения), Collins, и «The Feeling Good Handbook» (Руководство по хорошему самочувствию), Plume. Эти книги многим помогли изменить свою жизнь.

Мы должны серьезно относиться к своему отдыху, чтобы извлечь из него максимум пользы. Мы должны сознательно планировать время на отдых и научиться не откладывать его. Это вопрос психической гигиены. Вы можете время от времени не чистить зубы, но в конечном счете это нанесет вред вашему здоровью. Аналогичным образом вы можете время от времени забывать о психической гигиене, но в долгосрочной перспективе это обернется проблемами. Чем больше вы практикуетесь в разумной организации отдыха, тем лучше у вас это получается.

Все курильщики, которых я знаю, не выглядят особо утомленными на работе. Думаю, никотин тут ни при чем. Все дело в том, что каждые два часа они выходят на улицу (у нас не разрешается курить в помещении фирмы). Это прерывает их рабочий день. Когда накопился стресс, они выходят на улицу. Я не призываю к курению, но настоятельно рекомендую вам каждые два часа выходить на улицу, чтобы размяться и посмотреть вокруг.



Перегруженность и конфликтующие указания

Работая системным администратором, я заметил, что для меня самыми сильными источниками стресса являются ощущение перегруженности работой и противоречивые указания начальства.

Почувствовав перегрузку, я вспоминаю о приемах ведения списка дел, описанных в главе 5. Я запираю дверь рабочей комнаты (или прячусь в конференц-зале), чтобы мне никто не мешал, и применяю систему

Цикл. Вскоре я обнаруживаю, что ощущение перегруженности возникло из-за того, что я пропустил какие-то шаги. Я прохожу через все законы и планирую остаток дня, перенося отдельные пункты списка на следующий день, и т. д. В результате я чувствую себя лучше, поскольку все дела в моем списке так или иначе обработаны.

Когда я *действительно* перегружен, этот метод не помогает, и мне приходится переходить на уровень выше, то есть обращаться к шефу. Хороший шеф всегда поможет вам расставить приоритеты, если вы перегружены работой. Признание в перегруженности - не проявление слабости. Просьба о помощи - признак силы. Чтобы попросить помощи, придется набраться мужества, а еще больше его нужно, чтобы последовать полученному совету.

Если шеф находится вне пределов досягаемости, вам поможет кто угодно. Расскажите обо всем кому-нибудь - это позволит вам справиться со стрессом. Может, вы и не получите никакого совета, зато, по крайней мере, выговоритесь. Это уже половина дела. Если вас выслушали, ваше настроение улучшится. Правда, обычно какой-то совет все же дают или помогают как-то по-другому забыть о неприятностях. Мне от этого всегда становится легче.

Сам процесс изложения проблемы собеседнику помогает нам решить эту проблему. Вы замечали, что решение проблемы нередко приходит на ум, когда вы пытаетесь рассказать о ней кому-то другому? В жизни полно ситуаций, когда вы говорите собеседнику: «Тут возникла одна проблема. Если я...» и вдруг ясно видите решение и понимаете, что можно не продолжать фразу. Это случается сплошь и рядом.

В книге «The Practice of Programming» (Практическое программирование), Addison Wesley, приводится история об одном сотруднике Bell Labs, который славился умением помогать коллегам в решении сложных технических проблем. К сожалению, он не каждый раз мог оторваться от работы, чтобы выслушать того, кто обратился к нему за помощью, поэтому держал в кабинете игрушечного медвежонка. Если этот человек был очень занят, он просто говорил пришедшему: «Расскажите все мишке». Этот прием срабатывал превосходно. Вскоре герой истории заметил, что люди, заходя в его кабинет, направляются напрямиком к медвежонку.

Понять, что у вас стресс и что вам нужно взять паузу и прибегнуть к описанным приемам, не так-то просто. Испытывая сильный стресс, я не могу осознать, что пора прекратить работу и заняться снятием стресса. Как-то я договорился с одним коллегой обозначить неким кодовым словом фразу: «Ты сейчас в таком сильном стрессе, что даже сам этого не замечаешь». Это кодовое слово можно было произносить при посторонних без тени смущения. Он говорил это слово мне, а я ему. Очень помогало.

Получение противоречивых указаний начальства - еще один распространенный источник стресса у системных администраторов. Нередко

у системного администратора два руководителя, которые по-разному расставляют приоритеты. Вы пытаетесь выполнить все указания, и это удается, пока вы не перегружены работой.

В идеальном случае можно было бы свести обоих шефов и позволить им разобраться друг с другом. К сожалению, это не всегда возможно. Если вам такое удастся, постарайтесь убедить своих начальников записать для вас систему приоритетов, чтобы вы могли обратиться к этому документу при очередном конфликте задач. А если вы вынуждены заглядывать в этот «Договор о мирном сосуществовании» слишком часто, возможно, потребуется какой-то организационный сдвиг, в корне пресекающий эту проблему. Вы также можете договориться с тем шефом из двух, который вам больше нравится, что будете выполнять исключительно его указания.

Невозможность разрешения подобного конфликта - одна из самых распространенных причин ухода системных администраторов с такого места работы. В сущности, это действительно серьезно. Быть может,

Сон как средство от стресса

Правильный режим сна снимает многие проблемы. Потребность в сне у каждого человека своя. Если вы спите сколько нужно, это позволяет вам лучше справиться со стрессом.

В особенно тяжелую неделю лишний час сна заметно помогает мне противостоять стрессу. Я лучше себя чувствую, более спокоен и легче общаюсь с другими.

Проблема в том, что найти лишний час для сна не так-то просто. Как правило, мы не можем вставать на час позже, значит, единственный вариант - лечь на час раньше. Это нелегко! Так много хороших телепередач, книг, форумов и сайтов в Интернете, компьютерных игр и т. д.

Я не могу лечь спать пораньше без посторонней помощи. Я прошу своих близких помочь мне (то есть заставить меня сделать это). Если вы живете один, пусть вам позвонит друг и велит вам лечь. Или поставьте будильник, чтобы он напомнил вам, что пора спать.

Я не могу лечь в кровать на час раньше просто так. Это целая процедура. Чтобы подготовиться ко сну, я должен оставить все дела за полчаса до сна. Мне довольно трудно ничего не делать, но обычно я достигаю такого состояния минут за пятнадцать. Я включаю мысленный таймер с обратным отсчетом. При T-120 я прекращаю есть и пить. При T-30 я умываюсь. При T-15 я начинаю бездельничать. При T-0 я выключаю свет и ложусь в кровать. При T+5 я... хррр...

вы лучше всего справитесь со стрессом, обновив свое резюме и раскрыв газету на разделе объявлений о вакансиях.

Отпуск

Позвольте мне открыть вам маленький секрет, касающийся отпуска.

Работодатель предоставляет вам отпуск не из благотворительности и не из хорошего отношения лично к вам.

Он делает это, потому что с вами трудно работать, когда вы в состоянии стресса. А если не смягчать формулировку, то, откладывая отпуск, вы превращаетесь в головную боль для своих коллег, и с вами никто не хочет иметь дело. Вы раздражительны, непонятливы и упрямы. **ТАК ВОЗЬМИТЕ ОТПУСК, ЧЕРТ ПОБЕРИ!**

Простите за грубость, это для вашей же пользы.

Удачный отпуск настолько освобождает от работы ваш мозг, что он может отдохнуть. На то, чтобы забыть о работе в достаточной степени и полностью перейти в режим отдыха, может уйти несколько дней. Только после этого ваш организм начнет самовосстанавливаться. Мне нужно как минимум три дня для перехода в режим отдыха и затем не меньше шести дней, чтобы почувствовать себя свежим. Добавьте пару дней на дорогу и день на то, чтобы втянуться в повседневную жизнь, и вы получите 12-14 дней, необходимых для полноценного отпуска. В любом случае вам потребуется не меньше.

Рассмотрим некоторые распространенные ошибки, допускаемые системными администраторами в отношении отпуска:

- **Использование незапланированного отпуска для выполнения мелких дел по дому и т. д.** Это не отпуск. Это использование времени отпуска, но оно не имеет отношения к отдыху. Делайте домашние дела вместо того, чтобы сидеть за компьютером, или придите на работу пораньше и отпроситесь с работы на час-другой для личных дел.
- **Присоединение отгула к выходным.** Это немного напоминает отпуск, но здесь отсутствует период перехода в режим отдыха. К тому же когда я пытался так поступить, у меня всегда оставались невыполненные дела, которые я планировал на выходные. Это порождает еще больший стресс. Серия подобных удлинненных уикэндов тоже не имеет эффекта.
- **Проверка электронной почты каждые несколько часов во время отпуска.** Беря в отпуск ноутбук и проверяя электронную почту, вы не сможете отдохнуть. Каждый раз, проверяя почту, вы переводите мозг в рабочий режим, и вам потребуется три дня для возвращения в режим отдыха. Большинство отелей предоставляет за небольшую плату доступ в Интернет. Я предпочел бы отель, где мне за небольшую плату гарантируют полную невозможность доступа в Интернет.

Не проверять электронную почту трудно. Очень трудно. Отправляясь в отпуск, я непременно отключаю себе доступ к виртуальной частной сети, иначе мне не удержаться от проверки электронной почты. Это как болезнь.

В первые несколько дней отпуска я обычно продолжаю думать о работе. Я заметил за собой, что если не удастся избавиться от этих мыслей, мне нужно записать их в расчете вернуться к этим записям, выйдя из отпуска. В противном случае я буду стараться не забыть их и, следовательно, не смогу выкинуть их из головы.

Я слышал, как некоторые системные администраторы хвастались тем, что не берут отпуск: «Эта фирма без меня не выживет! Я горжусь тем, что у меня не было отпуска несколько лет». Для меня подобное заявление - повод для беспокойства. Как менеджер, я опасаясь, что у системного администратора разовьется комплекс мученика. Человек с комплексом мученика полагает, что все у него в долгу, поскольку он платит такую высокую цену за то, чтобы в фирме все было в порядке. С таким человеком становится невозможно работать. Я заметил, что сотрудник в этой ситуации в конце концов начинает чувствовать себя заваленным работой, загнанным в угол и не способен найти выход. Человек с подобными настроениями, как правило, увольняется, причем неожиданно. В результате я теряю технически одаренного специалиста, которого нечем заменить.



Я меньше чувствую себя загнанным в угол, если могу уйти с работы легко и без мук совести. Этому весьма способствует правильное ведение документации. В главе 12 рассказывается, как облегчить документирование процессов.

С деловой точки зрения у большого отпуска есть еще одно достоинство: он позволяет определить, где заканчивается ваша зона действия и/или насколько удачно ведется ваша документация. Хороший системный администратор исходит из предположения, что завтра его может сбить грузовик, а фирма должна остаться на плаву и без него. Большой отпуск позволяет проверить эту теорию без вреда для здоровья.

Вот несколько моих советов по поводу ухода в отпуск:

- За две недели до отпуска выясните, какие ваши функции должны быть переданы другому сотруднику, и потратьте достаточно времени на его обучение.
- За неделю до отпуска убедитесь, что коллега, который будет вас замещать, способен выполнять ваши обязанности, не задавая вопросов.
- За день до отпуска не делайте ничего как системный администратор. Не вносите изменения, которые трудно или невозможно исправить. Если искушение велико, отвлекитесь на что-нибудь. Например, составляйте документацию.

- Вернувшись из отпуска, проверьте, какие сбои произошли в ваше отсутствие. Как правило, оказывается, что какая-то часть системы откажала и фирме пришлось обходиться без нее до вашего возвращения. Почему произошел отказ? Что нужно было задокументировать?

Возможно, вы возразите: «Но разве я смогу научить кого-нибудь *всему*, что знаю?» А вы и не должны это делать. Ему достаточно знать столько, сколько требуется, чтобы фирма продержалась без вас неделю или две. Например, пусть в ваши обязанности входит генерирование некоторой отчетной документации каждую неделю. Сотрудник, который вас будет замещать, не обязан уметь создавать новые шаблоны отчетов. Ему достаточно уметь генерировать отчеты по имеющимся шаблонам. Если понадобится новая форма отчета, она может подождать до вашего возвращения. В самом крайнем случае ваши коллеги распечатают несколько отчетов и с помощью ножниц и клея сконструируют бумажку необходимого формата. (В действительности человек, требующий отчет, отнесется к ситуации с пониманием, потому что сам периодически уходит в отпуск.)

Рассмотрим еще пример. Вы отвечаете за создание резервных копий. Сотрудник, остающийся за вас, должен уметь выполнять ежедневное копирование и восстанавливать удаленный файл или каталог. Ему не нужно знать, как добавить в план копирования новый том или как реорганизовать план. Маловероятно, чтобы без вас в системе появился новый диск, а если кого-то из руководства осенит блестящая мысль о новом плане копирования, реализацию этого плана вполне можно будет отложить на неделю. Конечно, если произойдет катастрофический сбой и RAID или другая система восстановления будет не в состоянии спасти данные, вам придется прервать отпуск, но так ли часто это случается? (А если подобная авария чревата ежедневными миллионными убытками, то почему в этой фирме восстанавливать утраченные данные умеет только один сотрудник? Но я отвлекся...)

Когда другой сотрудник уже научился замещать вас, вы должны скоординировать свои отпуска, чтобы они не пришлись на одно время. Это нормальная рабочая практика. В большинстве известных мне фирм директор согласует свой отпуск с главным бухгалтером, а главный инженер - со своим непосредственным руководителем. Мы играем не менее важную роль и должны поступать так же.

Йога, медитация и массаж

В юности у меня были своеобразные заблуждения по поводу йоги, медитации и массажа. Я считал их странными и не заслуживающими внимания. Как я ошибался! Это прекрасные методы снятия телесного и умственного напряжения. Они являются важной частью хорошей программы релаксации и борьбы со стрессом, позволяющей поддерживать высокий уровень психического здоровья.

Прежде чем продолжать, я должен побороть смущение и рассказать о своих заблуждениях относительно йоги, медитации и массажа:

- **Йога.** Мне казалось, что йога - это экзотический комплекс упражнений для развития гибкости. Я с детства был не в ладах с гимнастикой и считал, что никогда не достигну гибкости, поэтому я игнорировал йогу. В самом деле, зачем моему телу быть гибким?
- **Медитация.** Я полагал, что медитация является религиозным ритуалом. Она не имела отношения к религии, которую я исповедовал, поэтому я закрывал на нее глаза (или правильнее будет сказать, я закрывал свой третий глаз?)
- **Массаж.** У меня были противоречивые и неверные представления о массаже. Во-первых, я считал, что это прихоть чудаковатых богачей. Я не был богат, и массаж меня не интересовал. Во-вторых, еще в детстве я видел Питера Селлера (Peter Sellers) в роли массажиста в «Мяппет-шоу». Он скручивал тело Кермита самым невероятным образом. Я не видел смысла кому бы то ни было подвергать себя такой процедуре.

Забавно, как долго мы находимся под влиянием детских впечатлений.

Когда мне было около двадцати пяти, я записался в группу релаксации. Мы занимались раз в неделю в течение восьми недель. Я бы не пошел в эту группу, если бы заранее знал, что в ней меня будут обучать йоге, медитации и массажу. Повторяю, я считал эти занятия странными и не интересовался ими. Как я ошибался!

В основе занятий лежала идея о том, что следует испробовать различные методы релаксации. Не предполагалось, что все они понравятся каждому из нас. Однако осознав, что какой-то метод вам подходит, вы могли изучить его глубже. Что-то вроде любимого блюда для каждого постоянного посетителя рестораника.

Итак, вот что я узнал:

- **Йога.** Йога - это древний способ достижения телесного расслабления. Принимая различные позы и оставаясь в них, вы расслабляете части тела. Мышцы расслабляются. Когда это происходит, расслабляется психика. Кто бы мог подумать?
- **Медитация.** Медитация - это целенаправленное успокоение психики. Если вам удастся перекрыть все посторонние раздражители и мысли, ваш мозг расслабляется, а вслед за ним и тело. Чем выше ваше мастерство, тем более удивительные результаты вы получаете. Боже мой, как я ошибался прежде!
- **Массаж.** Массаж - это физическое манипулирование мышцами с целью их расслабления. Я испытываю боли в спине, и для меня было неожиданным узнать, что это не спина болит, а мышцы спины так напряжены, что болят. Они не расслабляются из-за боли, а из-за их напряжения я чувствую боль. В результате процесс поддерживает сам себя. Массаж заставляет мышцы расслабиться и пере-

водит их в спокойное состояние. Я заметил, что ежемесячный массаж предотвращает излишнее напряжение мышц спины (как ежемесячная перезагрузка серверов Windows позволяет им долго сохранять работоспособность).

Я привожу здесь эти сведения потому, что вы, вполне возможно, пренебрегаете йогой, медитацией и массажем по незнанию. Я предлагаю вам изучить эти методики. Только испробовав их на себе, вы сможете понять, подходят ли они вам. Скорее всего, не все они вам подойдут. Но будет достаточно, если вам поможет хотя бы одна.

В США достаточно центров, предлагающих подобную подготовку. Обычно они размещают рекламу в каталогах, посвященных здоровому образу жизни, которые вы можете найти в ближайшем магазине, торгующем натуральными продуктами. Центры, подобные Институту Омега в Райнбеке, штат Нью-Йорк (недалеко от Вудстока) (<http://www.eomega.com>), предлагают вводные занятия по выходным, что может послужить прекрасным началом. Ваше местное отделение YMCA (Young Men's Christian Association, Христианский союз молодых людей), институт культурных программ или институт образования взрослых тоже могут предлагать подобные программы. Пospрашивайте окружающих.

Резюме

- Системное администрирование- работа, сопряженная со стрессом. Следовательно, умение справиться со стрессом должно входить в число ваших профессиональных навыков.
- Нередко самыми сильными источниками стресса являются ощущение перегруженности работой или конфликтующие указания от нескольких начальников, по-разному расставляющих приоритеты.
- Справиться с ощущением перегруженности хорошо помогает технология тайм-менеджмента, особенно система Цикл.
- Проблему стресса, вызванного конфликтующими приоритетами, следует пресекать в корне. Для этого необходимо провести переговоры с источниками конфликта или свести заинтересованные стороны для диалога.
- Релаксацией можно управлять, и этому надо учиться. Вам следует выделить в своем графике время для целенаправленной релаксации с помощью йоги, медитации, массажа или аналогичных методов.
- Вы должны научиться приемам релаксации, чтобы достичь положительных результатов. Вы должны запланировать сознательную релаксацию, так же как планируете замену масла в машине.
- Отпуск - это не подарок вам от работодателя. Это часть вашего режима релаксации и программы снятия стресса. Вы уходите в отпуск, чтобы поддержать свое психическое здоровье. Отказаться от

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

очередного отпуска - все равно, что отказаться от очередной чистки зубов. Вам может казаться, что на этот раз в отпуск можно не уходить, но в конечном счете это ударит по вашему здоровью. Если вы растратите время отпуска на повседневные домашние дела, такой отпуск не поможет вам справиться со стрессом.

- Я не люблю каркать, но хороший сисадмин допускает, что если его завтра собьет грузовик, фирма сможет продолжить работать и без него. Большой отпуск - прекрасный способ без ущерба для здоровья протестировать план замещения системного администратора и качество документации. Возможно, вам удастся найти и устранить пробелы в плане замещения, когда вы вернетесь из одно- или двухнедельного отпуска.
- Устройте себе перерыв. Глубоко вздохните. Никто не говорил, покидая этот мир: «Как жаль, что я провел на работе так мало времени». Расслабьтесь, все это только нули и единицы.

10

Управление электронной почтой

Наша манера обрабатывать электронную почту может много рассказать о том, как мы управляем своим временем. Многие системные администраторы идут на поводу у электронной почты, а не управляют ею. В этой главе обсуждается, как следует и как не следует обращаться с электронной почтой. Я предлагаю улучшенную технологию обработки электронной почты, описываю, как разобраться со старыми письмами, которые у вас, возможно, накопились, и касаюсь других вопросов, относящихся к электронной почте.



Обработка вашей электронной почты

Программа чтения электронной почты не очень хороший инструмент тайм-менеджмента. В этом быстро убедятся те, кто попытается одновременно работать с папкой входящих писем и списком дел. Пару дней все работает прекрасно, но затем вы вдруг получаете много писем, и ваш метод с треском проваливается. Письма перемешиваются со списком дел, и нет никакой возможности расставить приоритеты и справиться с происходящим.

Моя первая рекомендация: ваш почтовый ящик должен быть пуст.

Чтобы ящик был пустым, вам понадобится разработать инструкцию о том, как поступать с каждым пришедшим письмом. В любом случае последним в инструкции должен быть пункт «удалить письмо», иначе ваш почтовый ящик станет заполняться. Если вы не удалите новое письмо вскоре после его получения, вам придется постоянно перебирать старые письма, чтобы выяснить, как с ними поступить. Иными словами, вы дважды (или больше) прочтете каждое письмо, прежде чем обработаете его. Это не очень эффективно.

Когда в главе 2 мы обсуждали прерывания, я описал систему «делегировать, регистрировать или действовать». При обработке электронной почты у нас больше возможностей:

- Отфильтровать
- Удалить, не читая
- Прочитать и...
 - Удалить
 - Сохранить
 - Ответить и удалить
 - Переадресовать (переслать) и удалить
- Тут же выполнить запрос и удалить письмо

Я понимаю, что такому опытному пользователю электронной почты, как вы, эти пункты кажутся очевидными и не требуют комментариев, но прошу меня простить. Возможно, вы знаете, как надо обрабатывать электронную почту, но поступаете ли вы так, как надо? В следующих разделах я раскрываю пункты этого списка и даю советы, основанные на личном опыте.

Фильтрация

Почтовые фильтры являются важной составной частью процесса обработки электронной почты. Если письма автоматически сортируются по содержанию, теме или отправителю, я могу разработать процедуры.

Основная масса электронных писем приходит ко мне от рассылок, на которые я подписался. Я создаю отдельную папку для каждой рассылки и устанавливаю фильтры для размещения таких писем в соответствующих папках.

Я группирую эти папки в две папки более высокого уровня. Первая папка содержит папки (рассылки), которые я читаю каждый день. Для меня это как ежедневная газета. Я стремлюсь к тому, чтобы в этой группе было не очень много сообщений и я мог бы прочитать все, накопившееся за день, за 15 минут.

Во второй папке высокого уровня находятся папки для менее важных рассылок. Если мне не удастся прочитать их к пятнице, я удаляю все

содержимое этой группы папок, не читая. Таким образом, я избегаю накопления устаревших сообщений. Я удаляю их в полном убеждении, что если бы их содержимое было действительно важно, я бы так или иначе с ним ознакомился.

Кроме того, у меня есть неофициальная группа рассылок. Сообщения приходят туда так редко, что нет смысла устанавливать для них специальный фильтр. Они попадают прямо в папку *Входящие*. Примером такой рассылки являются анонсы новых версий почтовой программы sendmail. Эти анонсы приходят настолько редко, что вполне могут сразу попадать в папку входящих писем. Создавать для них фильтр было бы излишней тратой сил, а иметь папку, которая почти всегда пуста, нет смысла.

У меня есть еще одно правило, касающееся списков рассылки. Раз в месяц я просматриваю рассылки, на которые подписался, и отменяю подписку на одну из них. Это процедура (см. главу 6), которую я выполняю в первый рабочий день каждого месяца. Иногда она бывает очень простой, например, если я подписался на рассылку, оказавшуюся не особо полезной. Иногда мне нелегко принять решение, но я все равно выполняю эту процедуру. В противном случае я, в конце концов, буду подписан на все рассылки мира. Некоторые люди придерживаются подобной процедуры для своего гардероба: покупая новую одежду, они выбрасывают ровно столько же старой. Вот заклинание, которое должно вам помочь:

Если непонятно, есть ли польза от рассылки, значит, никакой пользы нет.

Удаление без чтения

Есть категория писем, которые я могу удалить, не читая. Как правило, это объявления администрации здания, в котором расположена фирма, спам и иной «хлам», не имеющий отношения к моей жизни.

Работая в Bell Labs, я часто получал обычной почтой сообщения о том, что какой-то вход в здание будет закрыт. Эти объявления дублировались электронными сообщениями, причем неоднократно. Безусловно, подъехав к перекрытому входу, я бы и сам заметил груды строительных материалов и запрещающие знаки. В конце концов я сделал вывод, что если тема письма не относится к факторам, влияющим на работу компьютеров (электропитание, вентиляция и т. д.), я могу удалить это письмо, не читая.

Чтение и...

Письмо, которое мы прочитали, должно быть как-то обработано. Моя цель - прикоснуться к письму только один раз. Под *прикоснуться* я подразумеваю «отреагировать на письмо и отправить туда, где оно в конечном счете должно находиться». Если у меня нет времени прочитать

письмо, я оставляю его неп прочитанным до более удобного момента. Я заметил, что, если я не дочитаю какое-то письмо до конца в надежде закончить чтение потом, мне впоследствии приходится перечитывать все письмо целиком. То есть я прочитаю, по крайней мере, часть письма дважды, а это нерационально. Итак, я выработал для себя правило: если я начал читать письмо, я должен дочитать его и обработать одним из методов, описанных в этой главе.

Фильтрация электронной почты на сервере

Я предпочитаю, чтобы моя электронная почта фильтровалась на сервере. Обычно при этом необходимо пользоваться протоколом IMAP и нельзя пользоваться протоколом POP.¹ Хотя большинство современных почтовых клиентов способно сортировать принятые сообщения, фильтрация почты на сервере имеет значительные преимущества.

Во-первых, я использую разные почтовые программы на разных компьютерах. Я не могу поддерживать все фильтры на всех компьютерах в синхронизированном состоянии. Зато с этим прекрасно справится IMAP.

Во-вторых, фильтры, установленные мной на сервере, срабатывают в момент прибытия письма, а не тогда, когда я запускаю почтовый клиент. Иными словами, фильтры работают и в мое отсутствие, и когда почтовый клиент не запущен. Это означает, что я могу создать фильтр, который будет отправлять копию письма на пейджер или сотовый телефон или вызывать команду для обработки этого письма.

Я работал в одной фирме, где секретари отправляли электронные сообщения всем сотрудникам, если после очередной презентации оставалось угощение. Я часто пропускал такие объявления, потому что находился в машинном зале. Это продолжалось, пока я не установил фильтр на сервере. Любое письмо, в теме которого содержалось слово «ланч» или «еда», копировалось на мой пейджер (сотовых телефонов тогда еще не было). Случалось, я успевал к угощению первым.

¹ Имеются в виду протоколы приема почты IMAP4 и POP3 (в текущих реализациях), выбор одного из которых осуществляется в настройках практически любого современного почтового клиента. К настоящему времени, тем не менее, пользователи чаще всего выбирают как раз POP3, а некоторые (не очень многие) администраторы почтовых серверов вообще не предоставляют доступ к почтовому серверу по протоколу IMAP4. О достоинствах IMAP и настройках почтовых клиентов можно почитать здесь: http://www7.pochta.ru/help.php?caption_id=15&lng=ru&skin=pochta_ru. - Примеч. науч.ред.

Теперь я получаю на сотовый телефон копии сообщений, в которых так или иначе упоминается еда или содержится слово «срочно», а также сообщений от шефа, от шефа моего шефа, от моих близких и от некоторых других важных персон. Это не только позволяет мне сосредоточиться на работе (мне не требуется постоянно проверять почту), но и гарантирует, что я не пропущу действительно важное сообщение.

Если ваш сервер предоставляет вам доступ к командной строке UNIX/Linux, вполне вероятно, что для фильтрации на сервере вы сможете использовать утилиту *procmail* (<http://www.procmail.org>). Я так люблю эту утилиту, что часто говорю коллегам: «Если вы не используете *procmail*, то работаете слишком много».

Некоторые почтовые серверы, работающие по протоколу IMAP4, реализуют фильтрацию на основе Sieve. Это открытый стандарт серверной фильтрации, ставший популярным благодаря серверу Cyrus IMAP для UNIX/Linux. В таком случае можно с помощью любого клиента обновлять фильтры на любом сервере, удовлетворяющем требованиям IETF RFC 3028 (<http://www.ietf.org/rfc/rfc3028.txt>). Адрес домашней страницы Sieve: <http://www.cyrusoft.com/sieve/>.

Удаление

Все мы получаем письма, которые можно удалить сразу по прочтении. Это письма, не требующие от нас никаких действий. В эту категорию часто попадают отправленные мне копии писем, адресованных другим людям, а также письма, подтверждающие получение сообщения от меня.

Я получаю огромное количество писем, автоматически сгенерированных различными системами. Request Tracker от Best Practical отправляет мне копии сообщений о любых изменениях в запросах в определенных категориях очередей. Это позволяет мне следить за тем, что происходит с запросами. Если от меня не требуется оперативное вмешательство, я читаю и удаляю такие сообщения.

Сохранение

Я стараюсь хранить как можно меньше почтовых сообщений. Я знаю многих, кто хранит все полученные сообщения. Они заводят пятьсот папок и по несколько минут размышляют, в какую из них лучше всего поместить то или иное письмо. Мой девиз: «Сомневаешься - выбрасывай!» Если через несколько дней я пойму, что мне нужна эта информация, я смогу отыскать ее в папке *Удаленные*. Если информация понадобится мне значительно позже, я обращусь к первоисточнику или найду крохобора, который расплывает время, сохраняя каждое полученное письмо.

Некоторые настраивают фильтр так, что копия каждого входящего письма (кроме рассылок) сохраняется в архивной папке. Это позволяет им безболезненно удалять все сообщения. Если впоследствии окажется, что удалено нечто важное, в архиве найдется копия. Я полагаю, что по мере удешевления места на жестких дисках популярность этой меры будет расти. Когда-нибудь в почтовых программах появятся специальные возможности, упрощающие подобный подход.



У архивирования всей электронной почты есть юридические аспекты. Выясните правила хранения почты, принятые в вашей фирме.

Почта, которую я сохраняю, распределяется по двум папкам: Сохраненные и Финансы. Все финансовые документы я сохраняю во второй папке, а остальные письма - в первой. Раньше у меня была тысяча папок на все случаи жизни. Оказалось, что прокрутка их списка отнимает больше времени, чем заслуживают письма в них. Теперь, если мне нужно старое письмо, оно находится либо в папке Сохраненные, либо в папке Финансы.

Ответ и удаление

Если письмо требует ответа, на него следует отвечать немедленно, чтобы не заставлять человека ждать.

Проблема в том, что иногда ответ на письмо требует много времени, которого у меня в данный момент нет. В таком случае, чтобы письмо не потерялось, я переносу его туда, где веду список текущих дел. После этого я могу удалить письмо из папки Входящие.

Например, я обычно отвечаю отправителю: «Я занес Ваше письмо в список дел и отправлю Вам более полный ответ до [дата]». Затем я пересылаю письмо в нашу систему регистрации клиентских запросов.

Система наподобие RT от Best Practical (<http://bestpractical.com>) позволяет вам сделать это за один шаг. Просто перешлите все письмо (включая вложенные файлы) отправителю, а копию отправьте по адресу системы, которая регистрирует новый клиентский запрос. В начало

Папка «Перья»

Кроме папок Сохраненные и Финансы, у меня есть папка Перья.

Если кто-то меня похвалил, то это очередное «перо мне на шляпу». Поэтому все благодарственные или хвалебные письма я сохраняю в этой папке. Если мой день не удался, я перечитываю эти письма, чтобы поднять себе настроение.

А еще эта папка помогает мне писать годовой отчет.

письма добавьте примерно такой текст: «Здравствуйте! Я получил Ваше письмо. Я отвечу Вам до [дата]».

И порядок.

Иногда удобнее записать запрос в органайзер и отправить человеку письмо с обещанием ответить позже.

Как бы то ни было, письмо обработано, и больше его не нужно хранить. (Если у вас нет системы регистрации запросов, я настоятельно рекомендую вам присвоить наивысший приоритет задаче по ее установке. Некоторые очень хорошие системы, включая вышеупомянутую RT, распространяются бесплатно.)

Раньше я полагал, что отвечать на каждое электронное письмо приходится из вежливости. Я сказал «вежливость»? Я думал, это моя обязанность! Теперь я отвечаю на очень небольшой процент писем. Если кто-то присылает мне анекдот, я не отвечаю: «Спасибо, это смешно». Тем более, я не отвечаю: «Черт возьми, я выхожу в Интернет с 1987 года и читал эту шутку миллион раз». Я просто удаляю письмо и перехожу к следующему.

Конечно, если на письмо необходимо ответить, это совсем другое дело. В таком случае я отправляю его обратно с лаконичным ответом. Включая оригинальное письмо в ответное, я избавляю себя от необходимости объяснять контекст. Жизнь слишком коротка, чтобы писать длинные письма.

Переадресация (пересылка) и удаление

Бывает, что запрос, содержащийся в письме, следует переадресовать кому-то другому. Переадресуя письмо, я всегда посылаю отправителю копию, чтобы он знал, кому я переадресовал его запрос. Иногда я заносю запрос в список дел в органайзере, чтобы отследить его выполнение к определенному дню. Это позволяет мне остаться в теме и убедиться, что запрос не затерялся.

В некоторых случаях нет необходимости отслеживать реакцию на переадресованные письма - например, если я переадресую письмо шефу или коллегам, чтобы они были в курсе дел. Я также не отвечаю на письма, распространяющие в Интернете «свежие» шутки, например историю о семилетнем английском мальчике по имени Крэйг Шерголд, который хотел попасть в Книгу рекордов Гиннеса, собрав самую большую коллекцию открыток. Это просто городской фольклор.

Выполнение запроса и удаление

Важные запросы или запросы, требующие мало времени, следует выполнять немедленно. Как правило, это запросы от шефа или несложные просьбы клиентов, которые быстрее выполнить, чем занести в систему регистрации или органайзер. Если какое-то дело требует меньше двух минут, легче сразу сделать его, чем записать и сделать позже.

Приступайте, не медля

В описанной системе самое трудное - начать. Если у вас 2000 входящих писем, моя система покажется вам чем-то из области фантазии.

Что я рекомендую? Забыть о действительно старых письмах и двигаться дальше.

В некоторых почтовых клиентах есть специальные возможности архивирования. Впрочем, нет ничего проще, чем самому создать папку с именем СтарыеПисьма-2005-11-19 (или какое у вас число на календаре?) и перенести в нее все письма, полученные на данный момент.

Теперь ваша папка Входящие пуста, а если вам понадобится что-то из архива, то он всегда под рукой.

Если же вы целый год не трогали ту папку, перепишите ее на CD-ROM, удалите, в общем, сделайте, что хотите, но уберите ее из почтового ящика. Если вы не открывали папку в течение года, маловероятно, что вы ее когда-нибудь откроете. Ваш почтовый клиент будет работать быстрее, потому что ему больше не нужно будет поддерживать такой огромный указатель.

Я мечтаю, чтобы кто-нибудь написал программу, которая ищет все электронные письма, лежащие в ящике больше года, и переносит их на CD-ROM. А еще я мечтаю о Java-апплете, который позволил бы мне выполнять поиск среди этих сохраненных писем и восстанавливать указанные письма в почтовом клиенте. К сожалению, таких программ нет. Я даже придумал, как назвать такую технологию: «Маринованная электронная почта» (как консервы). Если вы напишете подобную программу, можете взять для нее это название - дарю.

Вам кажется, что невозможно вот так сразу выбросить две тысячи писем?

Позвольте спросить вас, а когда они будут обработаны?

В следующем месяце?

И как долго вы уже говорите «в следующем месяце»?

До того как мной была разработана методика, описанная ранее в этой главе, я без особого успеха пытался применять две другие:

- **Обработка 100 случайно выбранных писем в день.** Я думал, что если я каждый день буду обрабатывать и удалять по сто старых писем, то за месяц я очищу свой почтовый ящик. Однако писем в нем становилось все больше! Я не смог поддерживать этот процесс в течение целого месяца. Кроме того, обработка ста писем может потребовать около часа. То есть для завершения проекта должно потребоваться 20-30 часов - больше, чем половина недели. Я могу провести это время с большей пользой.
- **Обработка писем по отправителю.** Другой метод, который я опробовал, заключался в обработке писем по отправителю. Я пытался

разобраться со всеми письмами от конкретного человека. В результате я породил поток писем типа «Вам это все еще надо?», «Как насчет этого?» и «Я наконец-то прочитал этот анекдот, спасибо, очень смешно». В конце концов я разобрался с этим отправителем навсегда... или до следующего раза, когда я соберусь терять время на электронной почте.

Метод обработки писем по отправителю хотя и провалился, имел определенное преимущество перед методом обработки ста писем день. Он включал в себя расстановку приоритетов. Я мог выбирать самых важных для меня людей, а не просматривать письма случайных отправителей.

И все же давайте посмотрим правде в глаза. Если у вас в почтовом ящике 1000 (или около того) писем, полагаю, вы должны признать, что на самом деле не обработаете их никогда.

К сожалению, это так.

По себе знаю, что признать это очень трудно. Но однажды я взглянул на самые старые письма из этой огромной кучи и осознал, что некоторым из них больше пяти лет - это письма из другой эпохи.

Если вы ответите на подобное письмо, человек подумает, что вы сошли с ума или что ваш ответ долго блуждал по Интернету. Некоторые отпускают шутки насчет путешествия во времени.

Что могло произойти в худшем случае? Если бы письмо было действительно важным, вы получили бы еще одно на эту тему или у вас возникли бы какие-нибудь проблемы. Непомерные почтовые ящики заполнены письмами, которые, в сущности, мертвы.

Итак, если вы никогда не обработаете эти письма, что мешает вам перенести их в архив и забыть о них? Без всех этих сообщений, съедающих память и другие ресурсы, ваш почтовый клиент будет быстрее работать. И запуститься он будет быстрее.

Резюме

- Большинство системных администраторов получает больше электронных писем, чем они могут обработать. Если вы не управляете своей электронной почтой, она начинает управлять вами. Возьмите контроль над электронной почтой в свои руки, и это будет огромным шагом на пути к полному контролю над вашим временем.
- Папка *Входящие* - неудачное место для ведения списка дел.
- В идеале папка *Входящие* должна быть пустой. Поэтому все ваши действия с очередным поступившим письмом должны завершаться либо удалением этого письма, либо сохранением его в другой папке. Для достижения этого идеала я предлагаю вам метод, подразумевающий обработку каждого письма с помощью одной из следую-

щих операций: фильтрация, удаление без чтения, чтение и выполнение какого-то действия, выполнение запроса и удаление письма.

- **Фильтрация.** Пользуйтесь программной фильтрацией для предварительной обработки электронной почты и автоматизации многих задач.
- **Удаление без чтения.** Некоторые типы писем можно спокойно удалять, не читая.
- **Чтение и выполнение действия.** Если письмо требуется прочитать, переслать, зарегистрировать (в органайзере или системе регистрации запросов) или сохранить в определенной папке, выполните требуемое действие и удалите письмо из папки *Входящие*. Не позволяйте ему там залеживаться.
- **Выполнение запроса и удаление письма.** Если содержащийся в письме запрос требует нескольких минут, выполните его тут же и удалите письмо.
- Если вы задумали разгрести кучу писем, скопившихся на данный момент, я рекомендую вам сохранить ее где-нибудь и забыть о ней. Если бы какое-то из этих писем содержало действительно важную информацию, она дошла бы до вас иным путем. Электронная почта эфемерна. Чем старше письмо, тем меньшую ценность оно представляет.

11

Исключение непроизводительных затрат времени

Эта глава поможет вам распознать причины непроизводительных затрат времени. В ней также предлагаются способы исключения этих затрат.

Позвольте, я немного расскажу о себе. Я обожаю читать новости Usenet (NetNews). Я могу читать доски объявлений часами. До появления Всемирной паутины Usenet занимал большую часть времени, проведенного мной в Интернете. Если бы не было Usenet, я бы учился на одних пятерки. Чертов Usenet!

В свою защиту скажу, что я был весьма прилежным читателем новостей. Я регулярно обновлял версии NewsReader в постоянной надежде получить ту, что позволила бы мне читать больше статей за меньшее время. Я даже проводил контрольные замеры.

Вы не поверите, с какой скоростью я читал статьи. Серьезно - другие фанаты новостей смотрели на меня с завистью и спрашивали, как это мне удается.



В один прекрасный день я оптимизировал этот процесс самым радикальным образом. Я решил вовсе прекратить чтение групп новостей. Выигрыш составил два часа ежедневно.

Самым эффективным способом оптимизации любого процесса является отказ от процесса. Лучше исключить процесс, чем автоматизировать его. (А если вы обязаны его автоматизировать, читайте главу 13).

Проблема в том, чтобы распознать, что стоит, а что не стоит исключать.

Что такое непроизводительные затраты времени?

Я определяю непроизводительные затраты времени как деятельность с низким соотношением полезного результата к затраченному времени.

Всякая деятельность имеет какой-то полезный результат. Если вы пять часов подряд играли в компьютерную игру, то полезным результатом будет полученное вами удовольствие. Однако большую ценность для вас могут представлять другие полезные результаты. Например, если вы те же пять часов будете заниматься ремонтом в квартире, чтобы повысить качество своей жизни, полезный результат окажется более долговременным, чем уничтожение миллиона врагов, состоящих из разноцветных пикселей.

Причины затрат времени на работе могут быть самыми разными: телефонные разговоры с чересчур болтливыми собеседниками; неэффективно организованные процессы; ожидание друзей-коллег, с которыми вы ходите обедать, чтобы наконец-то отправиться в столовую.

Как избежать соблазнительных затрат времени

Против некоторых видов непроизводительных затрат времени просто невозможно устоять. Здесь все сводится к правилу «знай меру». Короткая беседа у автомата с газировкой прервет монотонный рабочий день, позволяя вам вернуться на место с ощущением передышки. А вот о многочасовой беседе на тему, не связанную с работой, этого не скажешь.

В том-то и проблема, что меру трудно соблюсти. Оскар Уайльд сказал: «Я могу устоять против чего угодно, кроме искушения». Трудно сказать себе «Я всего минутку поиграю в компьютерную игру» или «Я только просмотрю темы электронных писем и прочитаю лишь самые важные». Вскоре вы уже удаляете спам, отвечаете на запросы, а потом вы смотрите на часы, и оказывается, что прошло несколько часов!

Как же быть?

Я могу избежать соблазна, если установлю для себя правила и заклинания (см. главу 3) и затем найду способ претворять их в жизнь.

Было бы неплохо, если бы наш мозг каждые пять минут задавал себе вопрос: «А какая польза от того, чем я сейчас занимаюсь?» Это помогло бы нам распознать, что мы втянулись в непроизводительную трату времени, и прервать это занятие. К сожалению, мы устроены иначе.

Я обнаружил, что лучше установить для себя некоторые правила. Например: «Когда зазвонит будильник, я перестану играть в эту игру». У меня дома есть старый кухонный таймер с громким сигналом. Чтобы остановить его дребезжание, требуются обе руки (одна - чтобы придерживать таймер, другая - чтобы вернуть стрелку на ноль). Я не могу просто хлопнуть по кнопке и вернуться к игре. (А еще я в восторге от этого парадокса: окружить себя высокими технологиями, но пользоваться допотопным таймером!)

На работе я выглядел бы глупо, постоянно выключая механический таймер, да и коллег беспокоил бы его звон. Поэтому я пользуюсь другими сигнализаторами, например iCal.

Правило: устанавливайте таймер прежде, чем отвлекетесь «только на минутку».

Вообще, я заметил, что работаю продуктивнее, когда дверь в мой кабинет закрыта (меньше прерываний и шума), тем не менее порой присутствие в моем кабинете коллеги облегчает борьбу с искушениями.

Ничто так не облегчает борьбу с искушением, как правильное воспитание, устойчивая система ценностей и... свидетели.

- Франклин Джонс

Совместная работа с коллегой над одним проектом помогает оставаться сосредоточенным. Во-первых, если меня отрывают от работы, у меня есть отговорка: «Извините, сейчас я работаю с таким-то. Не могли бы вы заглянуть попозже?» Однако более важная причина заключается в том, что я даже и не думаю об искушениях. Например, я не могу при коллеге проверить другой мой почтовый ящик, предназначенный для личных писем.

Распространенные причины непроизводительных затрат времени

Вот некоторые распространенные причины непроизводительных затрат рабочего времени и советы по борьбе с ними.

- **Ненужные пункты в списке дел.** У каждого в списке дел есть какое-то количество мусора. Покажите свой список дел шефу и посмотрите, не упадет ли он в обморок. Возможно, он вычеркнет некоторые пункты; возможно, он сам снимет некоторые вопросы. Не исключено, что в вашем списке есть ежедневно повторяющиеся пункты, о которых шеф и не подозревает. Однажды, когда я показал свой список шефу, он заметил, что в моем расписании ежедневно отводится 15 минут на проверку и устранение одной и той же проблемы. Вскоре он даже принял решение заменить неисправную систему!
- **Слишком много подписок на рассылки.** На сколько рассылок вы подписаны - на 10, 20, 100? Я ежемесячно отказываюсь от одной

рассылки. Это повторяющийся пункт в моем списке дел, назначенный на первую число каждого месяца. Я заметил за собой, что на протяжении месяца я подписываюсь на рассылки, которые кажутся мне интересными. Если бы я регулярно не отказывался от подписок, то в конце концов был бы подписан на все существующие рассылки. (Советы по организации работы с электронной почтой приведены в главе 10.)

- **Электронные доски объявлений, группы новостей и т. д.** В большинстве таких систем отношение сигнал/шум столь невелико, что они редко имеют какую-то ценность для работы.
- **Чаты.** Я согласен с тем, что многие применяют IRC и интернет-пейджеры для более эффективной работы. Однако применение этих систем в нерабочих целях, по-видимому, приносит мало пользы, особенно в рабочее время.
- **Коллеги, проходящие мимо.** Я заметил, что коллеги, идущие по своим делам, с большей вероятностью остановятся около моего стола, чтобы поболтать, если я сижу лицом к двери. Переседайте так, чтобы люди, проходящие по коридору, не могли поймать ваш взгляд.
- **Торговые агенты и вербовщики.** Раньше мне стоило огромного труда отвязаться от позвонившего торгового агента или вербовщика. Теперь я знаю, что секрет в том, чтобы сказать «Спасибо, меня это не интересует» и положить трубку. Только один раз за десять лет я таким образом отказался от разговора, от которого не следовало отказываться. Тот человек перезвонил мне еще раз.
- **Работа вручную.** Есть множество языков программирования, созданных специально для того, чтобы системным администраторам было легче автоматизировать свою работу. Самый распространенный из них - Perl. Набирают популярность Python и Ruby. Поищите книгу по какому-нибудь из этих языков в ближайшем магазине технической книги. Огромную помощь в автоматизации вашей работы окажет вам серия Cookbook (Книга рецептов) издательства O'Reilly. Вопросы автоматизации более подробно освещаются в главе 13.

Общение с коллегами

Вопреки распространенному мнению технари - очень общительные люди. Разговоры возле автомата с газировкой имеют определенное положительное влияние на рабочий процесс, но всему есть предел.

Однажды у меня был коллега, который любил разговаривать о политике и мог провести полдня, информируя о текущих событиях окружающих (которым он не давал и слова вставить). Пользы от участия в таких разговорах не было никакой, тем не менее они собирали немало людей. Я почти всегда был с ним не согласен и часто втягивался в дискуссию, потому что не хотел, чтобы его точка зрения оставалась непоколебимой. Думаю, что и другие сотрудники, не согласные с ним, испытывали желание бросить работу и вступить в дебаты.

Увы, все эти разговоры были пустой тратой времени. Я даже не могу сказать, что они имели хоть какой-то положительный результат, потому что взгляды этого человека раздражали и отвлекали от работы. Я не мог положить этому конец, потому что не был его начальником, а никто из начальства, похоже, не считал происходящее большой проблемой. Но, по меньшей мере, я не должен был позволять себе втянуться в спор. В конце концов я научился распознавать ситуацию и либо переводить разговор на темы, связанные с работой, либо тихо уходить на рабочее место.

Подумайте о времени, потраченном вами на разговоры с коллегами о фильме «Властелин колец»; о новой книжке комиксов, которую собираются экранизировать; о том, что лучше - Emacs или vi (либо, к примеру, Windows или Linux), или о чем вы еще там болтали. Имели эти споры хоть какую-то ценность? Конечно, если вы не особо загружены работой, от разговоров с коллегами есть определенная польза. А если эти беседы длятся больше часа?

Думаю, многие даже не осознают, как много времени отнимает у них эта привычка. Задайтесь целью прохронометрировать следующий марафонский разговор о том, чем «Вавилон 5» лучше «Звездного пути». Вы будете удивлены, как трудно определить момент, когда вы включились в дискуссию. Вы будете не менее удивлены, осознав, сколько времени она продолжалась.

Будьте бдительны в отношении подобных разговоров. Научитесь улавливать грань, после которой они из краткого обмена впечатлениями превращаются в непроизводительную трату времени, и возвращайтесь к работе.

Непроизводительные совещания

Нередко оказывается, что мы участвуем в нескольких совещаниях подряд. В этом нет ничего плохого, если они делу на пользу, но вам следует научиться отличать совещания, на которых время тратится впустую.

Есть множество видов совещаний, но я предлагаю разбить их на два общих типа: статусные и рабочие. На *статусных* совещаниях люди просто отчитываются о том, как продвигается работа над их частью проекта. На *рабочих* решаются производственные вопросы.

Если на статусном совещании вы попытаетесь решить конкретную проблему, это будет непроизводительной тратой времени для каждого из участников. Если вопрос не может быть решен быстро, выберите момент, когда смогут встретиться все заинтересованные стороны, и не отнимайте время у остальных. Я заметил, что самым удачным моментом для этого является время сразу после окончания текущего совещания. Люди, не связанные с проблемой, могут покинуть зал, а те, кто должен ее решить, уже присутствуют, и нет необходимости согласовывать их рабочие графики.

Рассылая объявления о совещании, я всегда указываю, будет ли оно рабочим или статусным. Тогда люди знают, чего им следует ожидать, и находятся в соответствующем настроении. Кроме того, ведущему совещание будет легче пресечь посторонние разговоры, если они возникнут.

Как участник, я отдаю себе отчет, обернется ли совещание непроизводительной тратой времени. Если я приду вовремя, это существенно повысит эффективность совещания. Подготовившись (прочитав материалы по теме совещания), я не буду тормозить ход встречи. Если же материалы должен представить я, то рассылка копий участникам за день до совещания позволит лучше подготовиться им (а это означает,

Советы ведущим

- Всегда рассылайте по электронной почте напоминания о совещании за 24 часа до его начала.
- Укажите, будет ли совещание статусным или рабочим.
- Укажите точную дату совещания (например, «пятница, 2 декабря»), а не просто «завтра».
- Укажите время совещания. Если участники находятся в разных часовых поясах, укажите время и часовой пояс - не следует предполагать, что участникам известен ваш часовой пояс. А еще лучше укажите время для всех часовых поясов.
- Укажите URL документов, которые понадобятся участникам совещания. Даже если это третье совещание на одну и ту же тему, продолжайте указывать URL. (Вкладывайте сам текст документа в электронное письмо, только если у вас нет иного выбора или если у некоторых из участников нет доступа к вашей Wiki-системе.)
- Укажите повестку дня - простой список вопросов, которые предполагается обсудить.
- Появляйтесь за пять минут до начала совещания или заранее, если требуется установить аудио/видео оборудование.
- Если вы хотите, чтобы участники собрались вовремя, не указывайте время начала 14:00. Укажите 13:54. Люди придут, чтобы выяснить, почему совещание назначено на такое странное время.
- Всегда начинайте вовремя. Если некоторые сотрудники постоянно опаздывают, начните совещание в указанное время и в первую очередь обсудите самые важные вопросы. Не повторяйте обсужденное для опоздавших; скажите им, что вся информация приведена в протоколе совещания. Люди перестанут опаздывать. (Они возненавидят вас, но придти станут вовремя.)

что я должен быть готов еще на день раньше.) Я всегда отправляю напоминание о совещании за 24 часа до его начала. Это сокращает число опоздавших и тех, кто вовсе не явился, понизив тем самым эффективность совещания.

Я научился во время совещаний воздерживаться от разговоров с соседями. Я понял, что мои остроумные комментарии, несмотря на их ценность для окружающих, уводят совещание в сторону. Итак, я приучил себя держать рот на замке. По мере распространения технологии WiFi оказалось, что мои замечания можно отправить на интернет-пейджер одному или двум друзьям. Это удовлетворяет мою потребность быть услышанным и не отвлекает других участников совещания. Если какое-то совещание является абсолютно пустой тратой времени, но не посетить его невозможно, я беру с собой ноутбук и работаю, одним ухом прислушиваясь к выступающим. Впрочем, чтобы соблюсти приличия, приходится садиться на самое незаметное место и заниматься этим, лишь когда на совещании очень много народу.

Естественно, работа не единственное место, где мы впустую тратим время.

Магазин видеопродукции: что выбрать

Приходилось ли вам бродить по магазину видеопродукции, пытаться вспомнить, что вы хотели взять напрокат? Случалось ли, что ваши друзья или коллеги упоминали хороший фильм, но именно в магазине его название вылетало у вас из головы? Такое бывает с каждым из нас.

Есть решение.

Ведите в своем органайзере список фильмов, о которых вы слышали похвальные отзывы. Если у вас портативный компьютер, создайте запись с названием «Видео». Если вы пользуетесь бумажным органайзером, выделите для этого списка особую страничку. Всякий раз, когда кто-то упомянет фильм, который вам захотелось посмотреть, заносите его в список.

Теперь, когда вы придете в магазин, вы откроете органайзер и возьмете первый фильм в списке. У вас будет больше времени на просмотр фильмов, и вы будете меньше времени тратить на попытки вспомнить их названия.

В качестве альтернативы могу предложить службу Netflix (<http://netflix.com>), которая высылает DVD напрокат по почте. На указанном сайте вы можете создать список фильмов, которые вам хочется посмотреть. Когда вы возвращаете DVD, фирма немедленно высылает первый свободный DVD из этого списка. Если ваш коллега похвалит какой-нибудь фильм, вы заходите на сайт Netflix и добавляете этот фильм в свой список. Вы можете даже заносить в список фильмы, еще не выпущенные на DVD. Они постепенно поднимутся в начало списка, и вы получите их, как только они появятся. Это идеальный вариант для случая «подождите, пока фильм появится на видео».

На этом сайте вы можете реорганизовать пункты списка. Есть утилиты для Windows и Mac, позволяющие манипулировать списком с большим удобством, чем непосредственно на сайте. Например, очень популярна утилита Netflix Freak для Mac OS X.

Телевидение: смотреть меньше плохих передач

Поясню: я люблю смотреть телевизор. Я не из тех снобов, которые гордо заявляют, что не смотрят телепередачи и, более того, не имеют телевизора. Я считаю, что телевидение - замечательная штука. За последние 15 лет оно из средства для абсолютно пустого времяпрепровождения превратилось в средство пустого времяпрепровождения с вкраплением жемчужин. Проблема в том, чтобы обнаружить жемчужины и избежать всего остального.

Проблема телевидения с точки зрения тайм-менеджмента: оно предполагает, что мы будем подстраивать свою жизнь под трансляцию передач, а не исходит из предположения, что мы хотим посмотреть какую-то конкретную передачу.

В результате, включая телевизор, мы смотрим не то, что хотели бы, а лучшую передачу из числа транслируемых в данный момент. Крайне неэффективно.

Но вот появились DVR, цифровые видеоманитофоны. Они позволяют вам записывать любимые телепередачи и смотреть их в удобное для вас время. Есть много моделей, но системному администратору милее прочих Tivo, где внутри используется Linux.

Я купил себе Tivo, чтобы *меньше* смотреть телевизор, то есть смотреть только лучшие передачи. Иными словами, если у меня появится время смотреть телевизор, я хочу видеть жемчужины, а не удовлетворяться лучшей передачей из числа предложенных на данное время.

Через несколько месяцев после покупки я заметил, что смотрю телевизор все больше и больше. Нужно было как-то побороть эту тенденцию.

И тогда я изобрел «Три правила Тома для просмотра Tivo». Они помогают мне тратить меньше времени на просмотр телепередач.

Правило 1

Просмотрев передачу до конца, следует удалить ее. Отговорки типа «Я посмотрю это еще раз» не принимаются. У вас нет времени посмотреть все, что записано, не говоря уж о том, чтобы пересматривать.

Правило 2

Добавляя новый пункт в список автоматически записываемых передач (функция Season Passes), следует удалить из него передачу равной продолжительности и периодичности выхода в эфир. В качестве альтернативы предлагаю ежемесячно удалять из этого списка как минимум одну часовую телепередачу (или две получасовых и т. д.).

Правило 3

Если у какой-то записи истек срок хранения, не продлевайте его; пусть запись будет удалена автоматически. Не позволяйте себе архивировать ее на ленту в надежде когда-нибудь просмотреть эту телепередачу. (См. Правило 1 насчет того, сколько у вас свободного времени.) Научитесь спокойно относиться к автоматическому удалению старых записей. Что касается меня, единственным исключением из этого правила являются три передачи, ради которых я собственноручно и купил Tivo: «The Daily Show» (Ежедневное Шоу), «The West Wing» (Западное крыло) и «24».

Это мои личные правила. Я разработал их, чтобы Tivo помогал мне сократить время, которое я трачу на телевизор. Ваша система ценностей может отличаться от моей.

Дела домашние

Домашние дела способны отнять массу времени. Если у вас нет стиральной машины, то тратить два часа в неделю на стирку в прачечной-автомате довольно расточительно, особенно если вы много работаете. Я предпочитаю отдавать белье в прачечную. Недалеко от моего дома есть два места, где по 85 центов за фунт мне выстирают и выглядят белье. Вместо того чтобы убивать полдня каждую неделю, я сдаю грязное белье в прачечную по дороге на работу и забираю чистое на следующий день по дороге домой.

В неделю у меня уходит на это 20 долларов. Хотя 80 долларов в месяц - это немалые деньги, я готов их потратить, чтобы освободить себе время на общение с друзьями, общественную деятельность или, скажем, на написание этой книги. Затраты того стоят.

Другие домашние дела тоже требуют уйму времени. Вызов уборщика один-два раза в неделю сэкономит вам время и позволит поддерживать жилище в приличном состоянии. Как правило, такая услуга включает в себя обработку пылесосом всех ковров и пола, мытье пола в кухне, удаление пыли со всех поверхностей и мытье санузла сверху донизу.

Чистая квартира имеет множество достоинств. Вы можете пригласить гостей. Люди с большей охотой придут в квартиру, в которой царит порядок. Время, которое вы сэкономите, обратившись в фирму по уборке квартир, можно будет потратить на устройство вечеринки с друзьями. Несмотря на то что общаться через Интернет проще, живое общение гораздо ценнее и создает более прочные дружеские связи. Кроме того, вызов уборщика, скорее всего, обойдется вам дешевле, чем встреча с друзьями в ресторане.

Дополнительное (и неожиданное) достоинство такого подхода - вы будете вынуждены навести порядок в квартире накануне прихода уборщика. Лично мне такая процедура предварительной уборки, проводимая два раза в месяц, позволяет поддерживать порядок в доме.

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

Услуга по уборке квартиры оказывается особенно выгодной, когда ее стоимость делится на несколько человек. Если вы, например, снимаете квартиру с друзьями, то будете избавлены от споров, чья очередь убирать. Я не могу представить себе квартиру, населенную четырьмя системными администраторами-холостяками, которая не выглядела бы так, словно по ней пронесся ураган и в которой не пахло бы, как в уборной. Короче говоря, так вы решите множество проблем.

Установка программ и монтаж аппаратуры

Раз речь зашла о том, чтобы нанимать других людей, а не выполнять работу самому, замечу, что имеет смысл вызывать специалиста по установке программ или аппаратуры и в фирме, где вы работаете, если, конечно, бюджет это позволяет. Это особенно эффективно в том случае, если эта работа выполняется однократно.

В качестве примера рассмотрим установку крупной системы резервного копирования. Процесс делится на два этапа. Во-первых, производятся монтаж аппаратуры и установка программного обеспечения. Этот этап заканчивается, когда у нас есть один сервер, для которого должным образом создаются резервные копии файлов. Второй этап включает в себя периодическое добавление/конфигурирование/удаление систем, являющихся предметом резервного копирования.

Изучение первого этапа требует огромных усилий, а отдача от него невелика. Вы потратите несколько дней (или недель) на то, чтобы смонтировать оборудование, разобраться в плохой документации и одолеть программные проблемы. Вы больше никогда не воспользуетесь приобретенными знаниями, потому что, раз эта система установлена, вторую устанавливать не потребуется. Работа, на которую вы потратили несколько недель, могла бы быть выполнена специалистом за день-два, потому что он занимается этим регулярно. Это его профессия. Он знает все подводные камни и умеет их обходить.

Второй этап процесса установки имеет гораздо большую отдачу. Если вы научитесь добавлять новый сервер, конфигурировать сервер для копирования нового диска и удалять сервер или диск, то такое умение представляет значительную ценность. Вы будете регулярно пользоваться полученными знаниями.

Другим примером является автоматизированная установка операционной системы. Настройка рабочей станции на автоматическую загрузку операционной системы и необходимых приложений может представлять определенную трудность, но имеет огромную отдачу, особенно если вы часто переустанавливаете системы или закупаете много новых компьютеров. В качестве примера систем автоматической установки могу назвать Microsoft RIS, Solaris JumpStart, Red Hat KickStart и FreeBSD NetBoot. Гораздо эффективнее заплатить специалисту, чтобы он настроил систему и обучил вас ее обслуживать (устанавливать но-

вые программы, к примеру), чем в одиночку продираться через первоначальную установку.

Подобное консультирование может оказаться дорогим, а, значит, его следует предусмотреть при составлении бюджета. Даже если плата за установку составит 20, 50 или 100 процентов от стоимости купленного оборудования и программного обеспечения, услуги специалиста по первоначальной установке окупятся. Особенно если вы при этом сможете заниматься другими проектами.

Если вы последуете моему совету, обязательно присутствуйте при работе специалиста и просите его объяснить, что происходит. Вы сможете перенять его опыт и лучше понять работу системы, что пригодится вам в случае непредвиденных затруднений. Вы потратите какое-то время, но его уйдет гораздо меньше, чем при попытке установить систему самостоятельно.

Прочее

Есть множество других дел, которые отнимают у нас массу времени и которые можно организовать гораздо лучше. Я надеюсь, что приведенный выше список непроизводительных затрат времени на работе и дома встряхнет вашу память и поможет вам найти в своей жизни дела, которые можно либо лучше организовать, либо исключить вовсе. Конечно, то, что для одного является непроизводительной тратой времени, для другого имеет огромное значение. Мы все очень разные.

Стратегия и тактика

С точки зрения системного администратора непроизводительной тратой времени является любое дело, которое можно было бы исключить, если бы нашлось время на создание инфраструктуры, позволяющей от этого дела отказаться. Иными словами, лучшим средством тайм-менеджмента для системного администратора является хорошая программно-аппаратная инфраструктура.

Стратегические задачи связаны с долговременным планированием, например с разработкой политики безопасности, получением поддержки от руководства и проведением этой политики в жизнь. Тактические задачи имеют отношение к конкретным процессам, таким как форматирование жесткого диска или установка нового компьютера.

Проблема в том, что мы настолько поглощены тактическими задачами, что у нас не остается времени на решение стратегических. Мы так заняты вытиранием луж на полу, что нам некогда починить текущий кран.

Если вы установите ДНСП-сервер, вам не придется тратить время на распределение IP-адресов. Вы не будете по несколько дней подряд устранять последствия атаки на систему, если у вас есть современная и мощная программа борьбы с вирусами и спамом, сеть со средствами

самозащиты и политика безопасности, одобренная высшим руководством фирмы. Вы не будете проводить полдня за устранением непонятных проблем с Windows, вызванных мелкими ошибками в настройках, если у вас разработана процедура автоматической установки операционных систем и каждый новый компьютер вводится в эксплуатацию соответствующим образом. Вам не придется ночью напролет восстанавливать данные с резервных лент, если ваша серверная инфраструктура включает в себя надлежащую вентиляцию, источники бесперебойного питания и избыточную дисковую подсистему (RAID). (Впрочем, RAID не панацея в случае серьезной аварии.)

Ключ к решению проблемы в выделении времени на стратегические проекты. Занесите их в календарь и распределите время на отдельные этапы в вашем списке дел. Я придерживаюсь того правила, что у меня всегда должен быть один текущий стратегический проект. Я бы хотел иметь 50, но если я буду заниматься ими всеми, то ни один не закончу. Лучше реализовать один хороший проект, чем начать 50 и ни один не довести до конца. Советы в разделе «Достижение наибольшего эффекта» главы 8 помогут вам правильно выбрать этот единый проект. Согласуйте с заинтересованными сторонами выбор самого эффективного проекта и работайте над ним всей командой, пока он не будет воплощен в жизнь.

Если вам нужна хорошая книга по этой теме, рекомендую «The Practice of System and Network Administration» (Практика системного и сетевого администрирования). В ней больше 700 страниц, и она достаточно полно раскрывает тему. Впрочем, мое мнение может быть несколько предвзятым.

Резюме

- Непроизводительные затраты времени - это деятельность с низким соотношением полезного результата и затраченного времени. Вместо того чтобы оптимизировать подобную деятельность, попробуйте вовсе отказаться от нее.
- Некоторые дела могут растягиваться, заполнив собой все ваше время. Вы можете прервать их, если установите для себя какой-то временной лимит. Выработайте ограничительные правила, например: «Начиная [такое-то дело], я установлю таймер, который подаст звуковой сигнал через 10 минут».
- Ничто так не помогает побороть искушение, как присутствие свидетеля. Работая в одном кабинете с коллегой, вы менее склонны отвлекаться от работы на посторонние дела.
- В современной жизни присутствует множество факторов, приводящих к непроизводительным затратам времени. Это запланированные необязательные дела, подписка на рассылки по электронной почте, интернет-чаты, посторонние дискуссии в рабочее время, на-

зойливые торговые агенты и вербовщики, процессы, выполняемые вручную, но подлежащие автоматизации, и т. д. Распознав подобный фактор, избавьтесь от него.

- Дома вы тоже можете бороться с такими факторами с помощью цифрового видеомэгнитофона для просмотра только важных телепередач, списка фильмов, которые хотите взять напрокат, и службы Netflix, чтобы меньше времени проводить в магазинах видеопродукции; пользуйтесь услугами прачечных и фирм бытового обслуживания, чтобы освободить больше времени для удовольствий.
- Изучение процесса установки системы, устанавливаемой только один раз, малополезно. К примеру, в случае централизованной установки системы резервного копирования выделение из бюджета средств на услуги специалиста обернется значительным выигрышем.
- С точки зрения системного администратора непроизводительной затратой времени является любое дело, которое можно было бы исключить, если бы нашлось время на создание инфраструктуры, позволяющей от этого дела отказаться. Иными словами, лучшим средством тайм-менеджмента для системного администратора является хорошая программно-аппаратная инфраструктура. Мысля стратегически, вы сможете исключить тактические задачи в долгосрочной перспективе. Иными словами, вы сможете перестать подтирать лужи на полу, починив кран.

12

Документация

В этой главе рассказывается, как хороший архив документов помогает системному администратору лучше работать и эффективнее управлять своим временем.

Но вначале поговорим о том, почему мы не любим, боимся и вообще избегаем составлять документацию.

Мы с настороженностью относимся к любой просьбе задокументировать то, чем мы занимаемся, потому что она звучит как признак грядущего увольнения. Если мы будем документировать свою деятельность, то нас смогут заменить кем-то другим. Иногда просьба все документировать исходит извне группы системного администрирования, обычно от того, кто излишне беспокоится по поводу стандарта ISO 9001 и не понимает, что процессы документирования ведут в тупик, а не в обход.

Очень трудно приступить к составлению документа. Слово «документация» ассоциируется у нас с тысячестраничной книгой, в которой описано то, чем мы занимаемся, как именно мы это делаем и вообще как все работает. Ну и с чего прикажете начать, если вдруг придется это описывать?

Системные администраторы, как правило, перфекционисты. Мы не в состоянии задокументировать *все*. Зачем начинать проект, если он никогда не будет закончен? Написание документации требует столько времени, что она нередко устаревает уже в процессе написания. Зачем составлять документ, который будет бесполезен, когда мы его закончим?

Кроме того, у наших дверей всегда толпятся клиенты с неотложными просьбами, которые, как правило, важнее документирования. При написании документации нужно долго работать, не отвлекаясь. У какого системного администратора есть такая возможность?

Наконец, профи ненавидят печатную документацию. Зачем дразнить гусей?

В этой главе предлагается настолько необычный подход, что я не хочу называть его документированием. Вместо этого мы будем создавать хранилище информации, полезное, доступное и легко обновляемое. Самое главное, оно поможет нам в деле тайм-менеджмента.



Документация, которая нужна вам

Что нужно системному администратору вместо громоздкой и устрашающей документации? Вам нужно хранилище информации, полезное с точки зрения тайм-менеджмента. Возможно, у вашего шефа есть свои резоны требовать, чтобы вы вели документацию. Я советую вам исходить из собственных соображений - эгоистических. Стройте хранилища информации, которые будут служить вам и вашим целям тайм-менеджмента, а не каким-то посторонним целям вашего шефа или отдела контроля качества. Собственно, системному администратору требуются два хранилища информации:

- **Хранилище, используемое клиентами.** Это документы, доступ к которым, по вашему мнению, должны иметь все пользователи вашей сети. Например, сюда входит описание общей политики системного администрирования и процедур, которые необходимо соблюдать.
- **Хранилище для внутреннего пользования ИТ-персоналом.** Информация, нужная лично вам для работы, например контактная информация производителей, полученные вами инструкции по задачам и т. д.

Первое хранилище экономит ваше время, стимулируя клиентов на более самостоятельное поведение. Они будут меньше вас беспокоить. Зачем клиенту звонить вам по какому-то вопросу, если ответ можно прочитать? Клиенты будут обращаться к вам, только если им что-то непонятно. Многие клиенты предпочитают разобраться в проблеме самостоятельно, потому что стесняются задавать глупые вопросы.

Второе хранилище будет полезным в той степени, в какой вы его таковым сделаете. В частности, вы будете заносить в него все процессы, процедуры и справочные материалы, которые нужны именно вам. Это

хранилище освобождает вас от необходимости помнить все. Оно позволяет вашему мозгу сохранять сосредоточенность, а это уже ценно.

Я предлагаю создать два хранилища, потому что первое должно быть легко доступно всем клиентам, а второе может содержать информацию, доступ к которой должен быть ограничен по соображениям безопасности.

В этих двух хранилищах вы должны собрать следующие материалы:

- Правила обращения за помощью (возможно, в виде простого дерева решений).
- Общие направления политики системного администрирования (со ссылками на соответствующие страницы документации отдела кадров и/или юридического отдела).
- Список производителей и сервисных организаций и их контактная информация.
- Список процедур, которые вам часто приходится выполнять, или процедур, которые должен уметь выполнять замещающий вас сотрудник.
- Простая схема сети, которую сможет использовать для справки новый (или временный) сотрудник вашей группы.

Вы должны будете разместить эту информацию соответственно в открытой и закрытой частях веб-сайта. Чтобы вам было легче начать, я приведу образец для каждого хранилища. Чтобы вам было легче сопровождать свои хранилища, я рекомендую вам воспользоваться технологией Wiki. Если она вам незнакома, подробные сведения вы можете почерпнуть в разделе «Технология Wiki» этой главы. Пока просто запомните, что Wiki – это тип веб-сайта, который очень легко обновлять.

Объявив это хранилище *живым документом*, вы избавитесь от боязни никогда не завершить его построение. В отличие от обычного документа, который приходится печатать заново всякий раз, когда в него внесены очередные изменения, живой документ просто поддерживается в локальной сети. Вы обновляете его по мере необходимости. Применительно к такому хранилищу термин «построено» не означает, что его создание завершено и теперь можно его распечатать. Он означает, что хранилище появилось на свет и готово развиваться.

Хранилище, используемое клиентами

Первый веб-сайт находится в свободном доступе и содержит документацию, необходимую вашим клиентам.

Если ваше хранилище просматривает клиент, главная страница должна быть максимально простой. Приведу образец. Создайте начальную страницу со следующими заголовками и разделами:

Как получить помощь

Перечислите возможные способы в виде маркированного списка.

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

Как запросить новые возможности

Перечислите возможности, для которых может понадобиться предпринять некоторые действия, и приведите инструкцию или ссылку, с которой можно начать. В качестве примеров могу назвать запросы на получение доступа к виртуальной частной сети (VPN) и на выделение дополнительного места на сайте.

Политика системного администрирования

Маркированный список ссылок на документы о политике, написанные вами, а также ссылки на соответствующие страницы документации отдела кадров или юридического отдела.

Свод всех установленных вами политик

Со ссылками на соответствующие страницы документации отдела кадров или юридического отдела.

Этого образца достаточно для любой небольшой группы системного администрирования, у которой пока еще нет подобного сайта. Если вы работаете в такой крупной информационно-технологической организации, что мой образец покажется вам смешным, то у вас, скорее всего, уже есть соответствующая домашняя страница или веб-сайт, и этот образец вам попросту не нужен. Тем не менее, я был удивлен обилием ИТ-организаций, на веб-сайтах которых отсутствует как минимум один из вышеперечисленных элементов. Я также считаю, что подразделения больших организаций сильно выиграли бы, обладая персональными хранилищами информации.

Политика системного администрирования - это правила, по которым живут пользователи ваших компьютеров и сетей. Сюда входят правила безопасности, соглашения об уровнях сервисной поддержки, правила пользования компьютерами и сетями, этикет, правила привилегированного доступа к информации и т. д. В разделе «Политика системного администрирования» приведите ссылки на существующие документы в формате HTML, MS Word или PDF. Если таких документов пока нет, не помещайте этот раздел на главную страницу. Однако внесите в свой список дел пункт о разработке подобных документов. Если у вас нет собственных идей на этот счет, прочитайте главу 7 (о безопасности) и главу 9 (об этике) книги «The Practice of System and Network Administration» (Практика системного и сетевого администрирования). Я рекомендую вам начать с правил пользования компьютерами и сетями. Если юристы или кадровики вашей фирмы разработали такие правила, укажите ссылки на соответствующие документы. Если приведенный образец показал, что у вас отсутствует политика системного администрирования, то это не так уж плохо.

Этот образец является всего лишь стартовой площадкой. Со временем вы сами поймете, что следует добавить, а что изменить.

Если у вас есть время и ресурсы, то следующий шаг должен заключаться в улучшении дизайна этой страницы, чтобы клиентам захоте-

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

лось сделать ее стартовой. Тогда у них появится стимул часто заходить на ваш сайт и искать на нем необходимую информацию, например правила, установленные для пользователей. Добавьте несколько полезных функций, например окно поиска Google, биржевые сводки или новости вашей компании. Задавайте эту страницу в качестве стартовой на каждом новом компьютере.

Внутренняя ИТ-документация

Второе хранилище содержит документы для внутреннего пользования, то есть документы, необходимые вам и членам вашей группы. Эти документы будут содержать важную информацию, и их следует как-то защитить от посторонних, например с помощью пароля. Как правило, это хранилище является защищенной областью единого хранилища документов.

Если у вас еще нет подобного хранилища, приведу образец:

- **Контактная информация производителей и условия сервисной поддержки.** Ссылка на список производителей и на договоры по обслуживанию техники.
- **Процедуры системного администрирования.** Список выполняемых вами процедур, которые должен уметь выполнять замещающий вас сотрудник. Примеры, в том числе инструкции по созданию учетных записей для новых пользователей и по удалению записей уволившихся сотрудников.
- **Схемы сети.** Ссылка на простую схему сети, которую сможет использовать для справки новый (или временный) сотрудник вашей группы. Это может быть ссылка на страницу со схемами.

Рассмотрим пункты этого списка более подробно.

Контактная информация производителей и условия сервисной поддержки

В разделе «Контакты производителей» поместите ссылку на каждого производителя, с которым имеете дело. Каждая ссылка должна вести на страницу, содержащую номер телефона торгового представителя, номер телефона службы технической поддержки и данные, которые вам будет необходимо сообщить, если вы столкнетесь с проблемой. Например, для одного из производителей я храню на этой странице номер его телефона, пункты голосового меню его справочной системы и ответы на вопросы, которые мне наверняка зададут: номер телефона, который служба поддержки использует как идентификатор моей учетной записи, номер моего договора о сервисной поддержке и т. д. Если с каким-то производителем был заключен отдельный договор на обслуживание каждой единицы оборудования, я храню на странице все номера договоров в виде таблицы. В этой таблице также содержится ссылка на процедуру восстановления пароля для каждого устройства и ссылка на локальную кэшированную копию этой процедуры.

Возможно, вы захотите воспользоваться соответствующей серверной функцией для создания единой страницы, которая содержала бы все остальные. Можно время от времени распечатывать эту суперстраницу и держать ее в компьютерном зале на случай сбоев. А если вы действительно круты, то напишете сценарий, который будет распечатывать эту страницу первого числа каждого месяца, если за прошедший месяц в ней появились какие-то изменения.

Обычно, связываясь с производителем, я держу под рукой эту страницу, хотя вся контактная информация есть в моей адресной книге. Так я могу убедиться, что страница в центральном хранилище содержит самую свежую информацию. Обнаружив, что информация устарела, я тут же обновляю ее.

Процедуры системного администрирования

Невозможно написать отдельные процедуры для всего, чем вы занимаетесь, да это и не требуется. Тем не менее, я советую вам задокументировать сложные процедуры, которые вы выполняете относительно редко, а также те процедуры, которые вы ненавидите.

Пример сложной процедуры - отсоединение зеркала RAID, а затем его присоединение/синхронизация. Вы можете «разбить зеркало» (то есть отделить основной диск от его зеркала) перед переходом на новую версию операционной системы. Если модернизация ОС не состоялась, вы смонтируете ту часть зеркала, которая не была изменена. В случае успеха вы присоедините и синхронизируете зеркало. При этом применяются довольно сложные команды. Поэтому, затевая нечто подобное в следующий раз, создайте веб-страницу и запишите все использованные вами команды, сопроводив их пояснениями. В будущем вы воспользуетесь этой страницей как справочником, и вся процедура займет у вас меньше времени.

Если есть несколько способов выполнения некоторой процедуры, но для вашей системы годится только один, задокументируйте его (и поясните, почему следует применять именно его). Часто бывает, что документ с инструкциями, найденный в Интернете или пришедший вместе с программным продуктом, предлагает много путей, ведущих к одной цели, но вы на опыте убедились, что вам подходит лишь один из них. Сохраните весь документ и снабдите его комментариями, например: «Применить вариант 3», или «Не делать этого», или «Эта комбинация прошла на сервере В, но не годится на остальных». Выделите комментарии цветом. Проявите уважение к авторским правам создателей документа!

Я часто создаю документы, по сути просто технологические карты (инструкции). Это не так страшно, как написать огромный документ, описывающий все в мельчайших подробностях. Я плохо запоминаю детали, поэтому пошел по пути технологических карт. Поскольку информация в хранилище легко обновляется, другие сотрудники со временем дополняют документ. Часто это вырастает в полноценную документацию.

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

Другой тип процедур, подлежащий документированию, - процедуры, которые вы не любите выполнять. Конечно, неплохо было бы задокументировать все свои действия, но где взять время? Просто задокументируйте процедуры, которые вы не любите выполнять, и у вас будет материал, позволяющий обучить этим процедурам другого сотрудника. Лично я терпеть не могу создавать новые учетные записи. Хотя я максимально автоматизировал этот процесс, для меня это все еще головная боль. Есть вещи, которые невозможно автоматизировать, например такие пункты моей технологической карты, как «Навестить сотрудника в его первый рабочий день и поинтересоваться, есть ли у него вопросы», или «Повторить визит через неделю». Итак, я задокументировал команду, создающую учетную запись, действия по проверке корректности новой учетной записи, а также другие действия, которые необходимо выполнить, когда появляется новый сотрудник. Это не «Война и мир» и даже не документ, разбитый на абзацы. Это просто маркированный список с комментариями. Но теперь, когда процесс задокументирован, я надеюсь переложить его на кого-нибудь другого. В главе 2 я уже говорил о делегировании. Хорошее хранилище документов значительно упрощает делегирование заданий.

Черт возьми, моя стратегия в том, чтобы заполучить побольше подчиненных. Я документирую все ненавистные мне процедуры, которые я поручил бы помощнику, если бы он у меня имелся. Как только фирма снова станет набирать персонал, я предложу руководству список обязанностей моего возможного помощника: создание новых учетных записей, замена лент для резервного копирования, обычные проблемы принтера и т. д. Как здорово, что все эти дела хорошо задокументированы и их можно передать другому!

Персонал набирают нечасто, но это нестрашно. Мне не нужен штатный сотрудник. Как только возьмут кого-нибудь для обслуживания системы в отделе разработки ПО, тут я и подсуну свою веб-страницу с процедурами и задачами, которые можно на него повесить. Вот такой я мерзкий тип!

Схемы сети

И наконец, хранилище должно содержать схему сети. Если таковая имеется, разместите ссылку на нее. Если нет, нарисуйте сперва простую картину вашей локальной сети с главными серверами. Изобразите что-то вроде облака, обозначающего все настольные компьютеры и ноутбуки. В одной фирме, где я работал, мне приходилось периодически рисовать на доске схему сети, чтобы показать местоположение своего компьютера. (Схема состояла из четырех точек, обозначающих наши сайты, пяти соединительных линий глобальной сети и стрелки к облачку, обозначающему интернет-соединение.) Добавление этой простой схемы в информационное хранилище было хорошей отправной точкой. Вам хватит десяти минут, чтобы нарисовать приблизительный вариант схемы и разместить его на веб-странице.

Самые пылкие сисадмины, вероятно, будут настаивать на схеме, нарисованной в программе Visio, с реалистичными пиктограммами серверов, расположение которых должно быть показано с точностью до миллиметра, но это как мышеловка. Вам не случалось, засев за рисование какой-то схемы, вдруг обнаружить, что весь день ушел только на то, чтобы довести ее до совершенства? В этой мышеловке нет сыра! Потратьте 10 минут, а не 10 часов. Я предпочитаю инструменты, не предназначенные для детализации, что вынуждает меня ограничиваться лишь общей схемой, не увязая в деталях. Как правило, я создаю схемы в PowerPoint и заносу в хранилище оригинал и его PDF-копию.

Если вы вправду не в состоянии противиться желанию создать идеальную схему, начертите ее на доске, сфотографируйте цифровой мыльницей и поместите в хранилище фото. Это быстро и действительно хорошо работает. (Если кто-то заявит, что схему надо перерисовать в профессиональном графическом пакете, обеспечьте ему доступ на запись в хранилище и скажите: «С нетерпением жду Ваших результатов».)

Кроме того, вам следует задокументировать важные информационные потоки внутри вашей фирмы: процесс приема и отправки электронной почты, расположение серверов службы каталогов и т. д.

Технология Wiki

Чтобы создать веб-сайт (хранилище информации) с легко обновляемыми страницами, воспользуйтесь технологией Wiki. Строго говоря, *Wiki* - это концепция, а не конкретный программный пакет. Есть много программных пакетов, предлагающих пользователю функциональность Wiki. Есть и оригинальное приложение Wiki (в переводе с гавайского - «быстрый»), но также есть TWiki, KwikiKwiki, PHPWiki и т. д. Просто замечательно, что многие разработчики создают программные системы, которыми вы можете пользоваться.



Я не применял технологию Wiki из-за глупого названия. Я думал: «Ни за что не буду пользоваться системой с таким дурацким названием, даже если она превращает свинец в золото». Я даже не пытался выяснить, что это такое. Три года спустя, начав работать с Wiki-приложением, которое установил другой сотрудник, я обнаружил, что моя производительность заметно повысилась. Как же я пожалел, что так долго игнорировал Wiki!

Так что же такое Wiki? Это веб-сайт, на котором любой посетитель может редактировать любую страницу, а установка ссылок на страницы чрезвычайно проста.

Не безумие ли это? В смысле, если любой посетитель может редактировать любую страницу, как быть с вандализмом? Кто-то может зайти на него, все удалить, занести в систему неверную информацию и т. д. Это будет катастрофа!

Не беспокойтесь, есть функциональные возможности, полностью исключающие подобные проблемы. Давайте сначала рассмотрим положительные стороны:

- **Легко добавлять новые страницы.** Любой посетитель может добавлять новые страницы. Если младший системный администратор заключит контракт с новым производителем, он легко добавит страницу, касающуюся этого производителя, разместит на ней первичную контактную информацию и т. д.
- **Wiki-сайты централизованны и доступны.** Любой пользователь может получить доступ к ним с помощью веб-браузера (если все его элементы управления на месте). На клиентской стороне не нужно никакого специального программного обеспечения.
- **Каждый может внести свой вклад.** Любой посетитель сайта может отредактировать любую страницу, если заметит опечатку или владеет дополнительной информацией. Документ может появиться на свет в виде небольшого маркированного списка, потом кто-нибудь добавит в него новые пункты, а затем кто-то еще превратит документ в полноценное описание процедуры.
- **Wiki-страницы отражают текущую ситуацию.** Когда добавляются новые страницы, снимается одна из самых больших проблем, связанных с документированием, состоящая в том, что документы нередко устаревают до момента опубликования. Wiki- это набор страниц, которые может немедленно обновить тот, кто обнаружил устаревшую информацию.

Проблема с использованием хранилищ документов обычно заключается в наличии препятствий к использованию. Пользователи вынуждены запрашивать учетную запись, разрешение на доступ и т. д.

Wiki-нотация и ссылки на страницы

Wiki ослабляет все эти препятствия. Вам не нужно проходить специальный курс обучения этой технологии, знание HTML тоже не требуется. Вам не нужна учетная запись для чтения документов. Если у вас нет учетной записи, а вы хотите отредактировать документ, то тут же получите возможность создать ее. Учетные записи создаются с таким уровнем доступа по умолчанию, что пользователи могут выполнять большинство базовых операций. Самое приятное, что, хотя пользователи могут писать на чистом HTML, есть так называемая Wiki-нотация, позволяющая им вводить простой текст, который Wiki затем отформатирует. Например, Wiki понимает, что слова, окруженные звездочками, символами подчеркивания или иными, должны быть как-то отформатированы. Если вы введете **такой текст**, то на экране появится **такой текст**. А если --такой текст--, то на экране появится *такой текст*. Если вы создадите список, каждая строчка которого начинается со звездочки, Wiki преобразует эти строчки в маркированный HTML-список с тегами ``. Большинство людей быстро осваивает эту кодировку, поскольку

уже пользуется ею в электронной почте, а если кто не пользуется, то справочной информации по форматированию в Сети предостаточно.

Создание ссылок на Wiki тоже не составляет труда. Если вставить во вводимый текст URL-адрес, Wiki превратит его в ссылку. Установка ссылок на другие Wiki-страницы проходит еще забавнее. У Wiki-страниц есть имена, имеющие специальный формат WikiWord. Программистам, работающим на Perl, он известен как CamelCase или StudlyCaps. Имя представляет собой слово, содержащее буквы как в верхнем, так и в нижнем регистре. Например, страницу можно назвать СписокПолезныхВещей. Каждый раз, когда вы используете в тексте это имя, Wiki превращает его в ссылку на страницу, даже если страницы с таким именем нет. В последнем случае шелчок по ссылке позволит пользователю создать страницу с этим именем. Иными словами, чтобы создать новую страницу, создайте ссылку, щелкните по ней и приступайте к редактированию.

Размещать документы на Wiki-сайте тоже очень просто. Документ присоединяется к странице. Таким образом, любая страница может стать контейнером для документов в формате PDF, Word и др. Однажды мне поручили организовать хранение документов Microsoft Word для сотрудников, не разбирающихся в технике. Я создал Wiki-страницу, назвал ее ТекущийПроект и показал им, как присоединять к ней документы. Wiki автоматически выводит таблицу присоединенных файлов. Может, кто и не осилит Wiki-нотацию, но уж присоединить документ к странице все в состоянии. Возникает разделение труда: специалисты создают Wiki-страницы и структуру хранилища, а те, кто далек от техники, присоединяют страницы. По мере привыкания к технологии Wiki эти пользователи смогут освоить и более сложные задачи.

Предотвращение Wiki-вандализма

Для Wiki-страниц есть комплекс социальных и технических средств, гарантирующий, что вандалы и злоумышленники не разрушат хранилище информации.

Во-первых, есть простой общественный контроль: каждое изменение регистрируется в журнале с указанием автора изменений. Удивительно эффективный метод! Полагаю, факт регистрации автора на 90% удерживает пользователей от внесения безответственных исправлений. Это тем более справедливо в корпоративных условиях.

Есть и технические средства борьбы с вандализмом. Все Wiki-страницы хранятся в системах, подобных RCS, CVS, Subversion или Microsoft SourceSafe. Это обеспечивает практически бесконечные возможности отката. Вы без труда отмените исправления или сравните две версии страницы для контроля изменений. Когда последствия вандализма легко устранимы, сам акт нередко теряет привлекательность в глазах хулигана. Если бы аэрозольная краска смывалась дождем, не было бы элемента азарта в том, чтобы написать на эстакаде «Фрэнсин любит Харви».

Вернуться в каталог учебников

<http://учебники.информ2000.пф/учебники.shtml>

В большинстве Wiki-систем есть контроль доступа. Для каждой страницы или группы страниц можно установить, кто может их читать, редактировать или переименовывать. По умолчанию каждый посетитель может отредактировать любую страницу, в чем и заключается «принцип Wiki». Однако вы скорее предпочтете, чтобы главная страница, меню и некоторые другие страницы были доступны для редактирования ограниченному кругу пользователей.

Пуристы Wiki заявляют, что подобный контроль доступа не нужен, потому что в том-то и прелесть Wiki-культуры, что как вандалу несложно испортить страницу, так и кому-то еще несложно тут же восстановить ее. Это правда, но я лучше сплю ночью, зная, что страницу с номером телефона моей службы технической поддержки вправе отредактировать только один человек - я сам. В Wiki-культуре фраза «Wiki с ориентацией на бизнес» равносильна фразе «Wiki с контролем доступа».

Завершающим ударом против вандализма в Wiki является уведомление по электронной почте. Большинство Wiki-систем могут отправлять уведомления по электронной почте при любом изменении страницы. Обычно такое уведомление содержит информацию о том, что было изменено (HTML-отчет о различиях), и вы можете быстро определить, было ли изменение благотворным или вредоносным. Некоторые Wiki-системы по умолчанию извещают веб-мастера о любом изменении. Пожалуй, это превышение необходимой самообороны.



Хотя стремление задокументировать абсолютно все можно понять, ни в коем случае не указывайте пароли на веб-странице. Не стоит рисковать, даже если она защищена паролем и находится на безопасном сервере. Например, однажды я обнаружил сайт, который его владельцы считали безопасным, потому что пароли хранились на странице, защищенной паролем и доступной только через SSL-соединение. Однако пользователи с учетными записями shell-account могли войти в систему и просматривать файлы напрямую. Поскольку это был основной сервер подразделения, такие учетные записи были у всех.

Система Wiki, с которой я больше всего работал, называется TWiki (<http://www.twiki.org>). Ее основным достоинством является управление доступом на добавление. Есть и другие системы - от чрезвычайно простых (одна из них написана на языке awk) до полнофункциональных. Некоторые крупные системы включают в себя технологию Wiki как одну из функциональных возможностей, другие целиком построены на концепции Wiki, например изумительная всеобъемлющая энциклопедия с открытым доступом - Wikipedia (<http://www.wikipedia.org>).

Резюме

- Хранилище документов - прекрасный инструмент тайм-менеджмента.

- Хранилище, ориентированное на клиентов, предоставляет им необходимую информацию, и они меньше беспокоят вас.
- Хранилище для внутренней ИТ-информации позволяет вам организовать справочный материал для экономии времени впоследствии. Особенно полезными могут оказаться технологические карты - короткие заметки о том, как следует выполнять сложные процедуры, - вашим коллегам не придется изобретать велосипед.
- Хорошо задокументированную процедуру легче делегировать другому сотруднику. Это позволит вам вычеркнуть пункт из своего списка дел, поручив эту процедуру кому-то другому.
- Технология Wiki облегчает доступ к хранилищу и не требует знания HTML. Позволив всем пользователям редактировать (почти) все страницы, вы добьетесь того, что страницы будут содержать самую свежую информацию.
- Создавать хранилище документов для группы системного администрирования не так страшно, как может показаться. Выбирая, что именно документировать, вы управляете масштабом хранилища. Приведенные в этой главе образцы помогут вам в начинании. Технология Wiki позволяет документам динамически расти и развиваться.
- Часто бывает трудно составить новый документ. Технология Wiki облегчает задачу разработки нового документа, создавая для вас все ссылки. Первоначальный документ может быть простой технологической картой, которая впоследствии будет дополняться. Вам совсем не обязательно с самого начала создавать безупречный документ. Напишите то, что важно в данный момент, и позвольте документу развиваться.

Личное хранилище информации

Некоторые сведения мне хочется иметь под рукой, где бы я ни находился, но памяти портативного компьютера недостаточно для их хранения. Я не стремлюсь к тому, чтобы эта информация была доступна всегда и сразу, но желательно иметь к ней хоть какой-то доступ. Размещение в Сети делает ее доступной почти из любого места, особенно по мере распространения технологии WiFi. Установить каталог, защищенный паролем, достаточно просто.

Часть информации я держу в хранилище Subversion. Подобно CVS и Microsoft SourceSafe, технология Subversion позволяет мне читать и обновлять информацию в хранилище из любой точки сети. Технология Subversion часто применяется для хранения исходного кода и отслеживания изменений. Теоретически, где бы я ни находился, я могу либо загрузить последнюю версию хранилища, либо обратиться через SSH к компьютеру, у которого есть связь с хранилищем. В личном хранилище я держу очень большую адресную книгу и некоторую другую информацию.

13

Автоматизация

Автоматизация наших задач - отдельное удовольствие. В какой еще профессии можно запрограммировать машины, чтобы они делали за нас нашу работу? Ах, если бы это было так просто! Автоматизация задачи требует времени, но выигрыш может быть огромным.

Я не собираюсь здесь давать уроки Perl, Python, Ruby, UNIX *shell*, VBasic или Kix32. В этой главе я расскажу зачем, что и как следует автоматизировать. Кроме того, я приведу фрагменты кода, которые помогают мне в работе уже много лет.

Достоинство автоматизации очевидно. Она сокращает нам объем работы, потому что автоматизированная задача требует от нас меньше вни-

Достаточно ли автоматизирована ваша работа?

Адам Москович (Adam Moskowitz), известный сисадмин, сказал мне, что для него «лакмусовой бумажкой» уровня автоматизации является возможность поручить работу менее квалифицированному сотруднику. Например, в одной фирме он автоматизировал задачу ежедневного резервного копирования диска до такой степени, что замену ленты могла выполнить секретарша. Каждый день система отправляла ей и Адаму электронное сообщение с отчетом о статусе резервного копирования, проведенного предыдущей ночью. Как правило, в сообщении содержались указания, какую ленту следует заменить. Если происходил какой-то сбой, секретарша знала, что ничего не следует предпринимать, пока Адам не устранит проблему сам. Со временем он усовершенствовал процедуру так, что система автоматически справлялась со все большим количеством аварийных ситуаций. В конце концов он добился, что система могла работать без его вмешательства месяцами.

мания и времени или, благодаря «хрону» (cron) UNIX или планировщику Windows, выполняется автоматически, без нашего участия. Неожиданным положительным эффектом автоматизации является простота делегирования автоматизированной задачи. Любая задача, которую вы переложили на кого-то другого, уже является маленькой победой.

В этой главе терминами «сценарий» и «программа» я буду обозначать разные понятия. *Сценарий* - это короткая программа, возможно всего из нескольких строк. Типичным сценарием является BAT-файл, несколько строк на языке Perl или небольшой *shell-файл* UNIX. *Программой* я буду называть более длинные программы, разработка которых требует обдумывания и планирования. Как правило, формальный процесс создания программы включает сбор требований, разработку и тестирование. Программы обычно пишутся на компилируемых языках, таких как C++. Интерпретируемые языки, вроде Perl, тоже подходят для создания больших программ, но используются реже. Программисты, пишущие на языке Perl, называют свой код сценарием, если он невелик, и программой, если код имеет значительный объем.



Что автоматизировать?

Трудно найти время для автоматизирования процессов, поэтому приходится выбирать. Мы не можем автоматизировать всю нашу работу. Задачи, встающие перед системными администраторами, распадаются на четыре общие категории:

- **Простые задачи, выполняемые один раз.** К первой категории относится большая часть вашей повседневной работы. Если задача проста и вы выполняете ее один раз, нет смысла ее автоматизировать. На автоматизацию уйдет больше времени, чем на саму задачу.
- **Сложные задачи, выполняемые один раз.** Вторая категория включает задачи, достаточно сложные, чтобы выполнить их с первой попытки, поэтому, оформив последнюю (работающую) последовательность команд в виде сценария, вы получите инструкцию для ее следующего выполнения. Если вам понадобилось сделать что-то

один раз, когда-нибудь придется делать это еще. Задачи этой категории состоят из длинных последовательностей команд, причем каждую команду лучше тестировать отдельно, постепенно строя цепочку из работающих команд. После этого вы сможете использовать эту последовательность с уверенностью, что она работает.

- **Простые задачи, выполняемые часто.** Третья категория - очевидный случай, здесь отдача от автоматизации будет максимальной. Время, потраченное вами на автоматизацию процедуры, вскоре окупится, потому что вы будете выполнять задачу снова и снова. Всегда автоматизируйте скучные и повторяющиеся дела.
- **Сложные задачи, выполняемые часто.** Четвертая категория - ловушка для системных администраторов, которые часто взваливают на себя больше, чем могут нести. Эта категория требует, чтобы вы убедили руководство в необходимости выделить ресурсы (время и деньги) на решение проблемы. Результатом может быть покупка коммерческого программного продукта, интеграция бесплатных инструментов и/или инструментов с открытым кодом в вашу систему или разработка собственного решения.

Теперь, специально для читателей, мыслящих зрительными образами, приведу эти категории в виде таблицы (рис. 13.1).

	Простые задачи	Сложные задачи
Выполняются однократно	Выполните вручную	Автоматизируйте
Выполняются часто	Автоматизируйте	Купите или разработайте приложение

Рис. 13.1. Категории задач системного администрирования

Многие удивляются, узнав, что я автоматизирую простые, но часто выполняемые задачи. Если задача проста, зачем ее автоматизировать? Я автоматизирую многие процессы из второй и третьей категории - от крупных задач до небольших последовательностей команд - по одной причине. Автоматизация процесса придает ему повторяемость и масштабируемость, гарантируя выполнение без ошибок:

- **Повторяемость означает, что я могу многократно выполнять этот процесс.** Например, устанавливая новые компьютеры, я хочу, чтобы все они запускались с одной конфигурацией программного обеспечения и с одинаковыми настройками. В противном случае обслуживание этих компьютеров превратится в кошмар. Если я автоматизирую процесс установки, он станет повторяемым, и все компьютеры будут иметь одинаковую конфигурацию. Если у меня какой-то процесс работает, я хочу, чтобы он каждый раз работал одинаково.

- **Автоматизация избавляет меня от необходимости запоминать редко выполняемые сложные процессы.** Иногда очень много времени уходит на выяснение опций командной строки, необходимых для выполнения поставленной задачи. Я превращаю в сценарий даже одиночную команду, чтобы через несколько месяцев мне не пришлось изобретать велосипед. Это, так сказать, долгосрочная повторяемость. Например, в Mac OS X я могу «прожечь» ISO-образ диска на CD-ROM с помощью команды `hduutil`. Однако вместо того чтобы читать руководство всякий раз, когда мне нужно вспомнить, какие опции подходят лучше всего, я инкапсулировал эту команду в сценарий. Теперь я всегда смогу использовать текст сценария для справки, даже не запуская его.
- **Масштабируемость.** Это качество означает, что я смогу выполнить процесс независимо от того, как разрастется моя сеть. Автоматизировав процесс один раз, я смогу запускать сценарий на всех компьютерах, распространяя свое умение на все узлы сети. Например, очень легко изменить настройку конкретного SSH-сервера. Несколько секунд работы в текстовом редакторе, и файл `sshd_config` изменен. Однако если автоматизировать этот процесс, то я смогу запустить его на сотнях компьютеров, возможно, оставив его выполняться ночью. Мне не нужно будет присутствовать при его выполнении и волноваться о том, на скольких компьютерах он работает — на 10 или 10 000.
- **Автоматизация помогает избежать опечаток.** Многие команды трудно ввести с клавиатуры без ошибок. Например, ту короткую последовательность команд, которую я постоянно использую. В нескольких строчках мне приходится трижды вводить имя пользователя и дважды - его учетный номер. Все это несложно набрать на клавиатуре, но довольно просто допустить опечатку. Превратив эту задачу в сценарий, я исключаю опечатки. Даже если надо ввести лишь несколько строк, имеет смысл создать сценарий.

Процесс автоматизации

Чтобы что-то автоматизировать, сначала следует выполнить это вручную. Затем нужно написать код для каждого шага. После этого вы должны собрать эти фрагменты кода воедино, тестируя каждое добавление. В завершение необходимо протестировать всю систему.

Шаг 1: выполнение процесса вручную

Первый шаг при автоматизации процесса состоит в обязательном выполнении его вручную. ЗадOCUMENTИРУЙТЕ каждый шаг и убедитесь, что вы знаете, как закодировать его. Потом соберите все фрагменты кода.

Один мой юный помощник постоянно подходил ко мне с просьбой помочь ему автоматизировать тот или иной процесс: «Я бьюсь над этой проблемой уже несколько часов! Я в тупике!»

«О'кей,» - отвечаю я, - «Покажи, как ты это делаешь вручную».

«Я не знаю. Мне не удалось это выяснить».

«Вот в этом твоя главная проблема, балбес. Понял?»

Как было сказано в главе 12, одним из плюсов документирования является то, что запись шагов процесса позволяет вам впоследствии автоматизировать его. Я не шучу. Если у меня нет времени автоматизировать какую-то задачу, я привожу ее поэтапное описание на своем Wiki-сайте и поручаю кому-нибудь ее выполнить. Тем самым я достигаю сразу две цели. Во-первых, я расширяю документацию, описывающую работу нашей системы. Во-вторых, я делаю первый шаг на пути автоматизации этой задачи!



Документируйте все шаги процесса, а затем автоматизируйте их. Если вы не в состоянии записать шаги, вы никогда не сможете автоматизировать их.

Запись процесса уже вынуждает вас идентифицировать все его этапы. Вместо того чтобы держать их в голове, вы покажете документ другим, и они смогут проверить его на практике.

Если у вас нет Wiki-сайта, воспользуйтесь ручкой и бумагой или текстовым файлом. Выполняйте процесс вручную и записывайте, что вы делаете. Каждую введенную вами команду следует включить в документ.

Шаг 2: кодирование каждого шага

Преобразуйте каждый шаг в команду или короткую программу. Протестируйте каждый шаг отдельно. Иными словами, напишите несколько маленьких сценариев, каждый из которых позволит убедиться, что код для соответствующего шага верен.

Если на каком-то шаге вы используете графический пользовательский интерфейс, необходимо найти его эквивалент в виде последовательности команд. В некоторых операционных системах это просто. Например, в SAM (System Administration Manager, диспетчер системного администрирования) из HP-UX есть кнопка, позволяющая отобразить команду операции, которую диспетчер выполнит следующей. В Mac OS X есть Automator и AppleScript, позволяющие автоматизировать операции, выполняемые с помощью графического интерфейса. Для Windows есть множество аналогичных утилит. Однако инструменты, автоматизирующие щелчки по кнопкам, могут оказаться менее эффективны, чем непосредственная установка ключей реестра или записей LDAP.

Рекомендуемая литература для администраторов Microsoft Windows:

- «Windows Server Cookbook» (Сервер Windows. Сборник рецептов), O'Reilly. Прочитав ее от корки до корки, вы узнаете массу полезных вещей. Вас удивит, как много операций, которые вы привыкли выполнять с помощью графического интерфейса, может быть записано

в виде сценариев путем несложного редактирования реестра. Эта книга откроет вам глаза на многое. Примеры приводятся на нескольких языках, как правило на VBasic и Perl.

- «Perl for System Administration», O'Reilly.¹ Эта книга особенно пригодится тем, кто работает и в UNIX, и в Windows. Как следует из названия, основное внимание в ней уделено языку Perl, и люди с опытом работы в Enterprise или UNIX с легкостью осилят ее. Кроме того, она будет полезна тем, кто много работает с ActiveDirectory и/или LDAP.
- «Win32 Perl Scripting: The Administrator's Handbook» (Создание сценариев на Perl для Win32. Справочник системного администратора), Sams. Тоже весьма полезная книга, особенно если вы только начинаете писать сценарии.

Рекомендуемые книги для администраторов UNIX/Linux:

- «Perl for System Administration» (см. выше).
- «UNIX System Administration Handbook»,² Prentice Hall PTR. Эта книга не только учит вас основам системного администрирования в UNIX, но и содержит много полезных ссылок и инструментов. В большинстве примеров приводится командная строка, что облегчит написание сценариев.
- «Essential System Administration» (Основы системного администрирования), O'Reilly. Еще одна великолепная книга с примерами, в которых используется командная строка.
- «Advanced Bash-Scripting Guide» (Расширенное руководство по написанию скриптов bash). Посетите сайт <http://www.tldp.org/guides.html>.

Шаг 3: сборка шагов воедино

Если код каждого шага работает, вы можете собрать все фрагменты в один сценарий.

При сборке кода все-таки лучше добавлять по одному шагу за раз. Тестируйте код после каждого добавления. Такой подход называется итеративной разработкой и является оптимальным при автоматизации. Тестируя сценарий после каждого добавления, вы в большей степени уверены, что вся конструкция работает, как задумано.

Например, когда на работу приходит новый сотрудник, нужно создать для него запись в каталоге LDAP, выделить ему место на внутреннем веб-сервере и протестировать его учетную запись, чтобы убедиться в ее

¹ Д. Бланк-Эдельман «Perl для системного администрирования», Символ-Плюс, 2001.

² Эви Немет, Гарт Снайдер и др. «UNIX: руководство системного администратора», BHV - Киев, 1998.

корректности. Каждое из этих действий может быть автоматизировано само по себе. Убедитесь, что команды, выполняемые на каждом шаге, работают. Затем соберите первую группу команд в сценарий и протестируйте его. Убедитесь, что последовательность команд работает и выводится необходимая вам отладочная информация. Запустите сценарий и убедитесь в корректности записей LDAP. Если все работает, добавьте следующую группу команд и протестируйте, что получилось. Убедитесь, что записи LDAP по-прежнему корректны и что пространство на внутреннем веб-сервере выделено. Добавьте еще одну группу команд и снова все проверьте.

Шаг 4: тестирование всего процесса

Наконец, мы должны протестировать процесс в целом. Если мы тестировали его после добавления каждого шага, тут будет совсем немного работы.

Вообще-то программисты не любят тестировать. Им хочется, чтобы программа правильно работала с первого запуска. Если выполнять тестирование на каждом шаге, оно не выглядит особо трудоемким; в результате на последней стадии приходится делать не так уж много.

Простые задачи, выполняемые часто

Приведу несколько примеров часто выполняемых простых задач. Обращаю внимание системных администраторов Windows, что эти примеры ориентированы на UNIX/Linux. Тем не менее, общие принципы применимы ко всем операционным системам.

Сокращения для команд

Большинство систем с командной строкой имеет ту или иную возможность создания псевдонимов. Это позволяет вам создавать новые команды на основе существующих. В каждой операционной системе принят свой синтаксис. В системе UNIX имеется множество языков оболочки (командной строки), самыми популярными из которых являются `bash` и `csh`. Они сильно различаются, и вы заметите, что, в первую очередь, в `bash` приходится употреблять знак равенства. Я приведу примеры для обеих оболочек.



Примеры на `bash` будут работать в любой оболочке, смоделированной по оригинальной Bourne Shell, созданной Стивом Борном (Steve Bourne) (`/bin/sh`), например в Korn Shell (`/bin/ksh`) и Z Shell (`/bin/zsh`). Аналогичным образом примеры на `csh` будут работать в любой оболочке, имеющей корни `csh`, включая оболочку TeXex C shell (`/bin/tcsh`).

Как попасть в нужный каталог

Мне нередко приходится выдавать команду `cd` для перехода в каталог с очень длинным путем. Вот пример использования псевдонима:

bash:

```
alias book='cd ~/tal/projects/books/time/chapters'
```

csh:

```
alias book 'cd ~/tal/projects/books/time/chapters'
```

Теперь я могу набирать на клавиатуре `book` каждый раз, когда мне нужно перейти в каталог, где находятся файлы для текущей книги. Если я возьмусь написать другую книгу, я обновлю этот псевдоним. (Я вожу «book» в течение последних шести лет!)

Это не только позволяет сэкономить на вводе. Вы избавлены от необходимости запоминать путь к каталогу. Освободить свою память всегда полезно.

Чтобы сделать псевдоним постоянным, вы должны добавить указанную строку в файл `.profile`, `.bashrc` (`bash`) или `.cshrc` (`csh`). Эти файлы считываются только во время входа в систему, так что либо выйдите из системы и войдите обратно, либо введите команду `source`, чтобы файлы были прочитаны снова.

bash:

```
./profile
```

csh:

```
source ~/.cshrc
```

(Примечание: в `bash` эта команда обозначается точкой.)

Псевдоним может обозначать целую последовательность команд. Отделяйте их друг от друга точкой с запятой. Вот пример, где выполняется переход в некоторый каталог и устанавливается переменная окружения в зависимости от того, работаем мы в системе А или В:

bash:

```
alias inva='cd ~/tal/projects/inventory/groupa ; export INVSTYLE=A'  
alias invb='cd ~/tal/projects/inventory/groupb ; export INVSTYLE=B'
```

csh:

```
alias inva 'cd ~/tal/projects/inventory/groupa ; setenv INVSTYLE A'  
alias invb 'cd ~/tal/projects/inventory/groupb ; setenv INVSTYLE B'
```

Вместо точки с запятой можно поставить двойной амперсанд для обозначения такого условия: «Выполнить следующую команду только в том случае, если первая завершилась удачно». Этот прием полезен, если нужно избежать выполнения команды не в том каталоге. Напри-

мер, вы хотите перейти в некоторый каталог и поставить там метку времени в журнале. Однако если команда `cd` не будет выполнена (сервер недоступен), не следует ставить метку времени в журнале текущего каталога.

bash:

```
alias rank='cd /home/rank/data && date >>.log'
```

csh:

```
alias rank 'cd /home/rank/data && date >>.log'
```



Не пытайтесь превратить одну операционную систему в другую. Псевдонимы сами по себе хороши, но ими нельзя злоупотреблять. Я часто видел людей, создающих десятки псевдонимов для того, чтобы симулировать DOS в UNIX. Думаю, это плохая идея. Так вы никогда не научитесь работать в UNIX, а, оказавшись за чужим компьютером, где нет ваших псевдонимов, попадете впросак.

Сокращения для имен компьютеров

Если вам приходится снова и снова вводить имена каких-то компьютеров, вы можете сэкономить немного времени, создав псевдонимы. Например, если вы часто обращаетесь к серверу *ramanujan.company.com*, то можете создать псевдоним (запись DNS CNAME) *ram.company.com*. Это имя чуть проще набирать на клавиатуре.

Проблема, связанная с таким подходом, заключается в том, что он может превратить сопровождение в кошмар. Если пользователи станут употреблять оба имени, вам придется сопровождать два имени. Вопрос: как бы создать псевдоним, известный только вам, не беспокоя других пользователей?

Как правило, часто обращаясь к какому-то серверу, я почти во всех случаях использую SSH (Secure SHell). Это безопасная (криптографически защищенная) альтернатива *telnet* и *rsh*. С ее помощью вы также можете копировать файлы (вводя *scp* вместо *gcp*), и многие программы, например *rsync*, работают с SSH. Система Unix SSH (OpenSSH и ее сестры) позволяет вам установить псевдонимы, известные всем пользователям UNIX, либо псевдонимы, доступные только вам.

Чтобы ограничиться только своими SSH-сеансами, добавьте псевдонимы в файл *~/.ssh/config*. Если же вы хотите предоставить псевдонимы в распоряжение всех пользователей, добавьте псевдонимы в файл */etc/ssh_config* или */etc/ssh/ssh_config* в зависимости от конфигурации вашей системы. В следующем примере я создаю псевдоним *es*, чтобы избавить себя от необходимости вводить *www.everythingsysadmin.com*:

```
Host es
  HostName www.everythingsysadmin.com
```

Теперь я могу не только вводить `ssh es` вместо `ssh www.everythingsysadmin.com`, но и использовать этот псевдоним в командах `scp`, `sftp`, `rsync` и других. Более того, сценарии и программы, которые я не могу изменить, будут автоматически воспринимать новые настройки. Вот несколько примеров:

```
$ ssh es
$ scp file.txt es:/tmp/
$ rsync ex:/home/project/alpha ~/project/alpha
```

Я использую `ssh es` так часто, что создал псевдоним на уровне оболочки:

bash:

```
alias es='ssh es'
```

csh:

```
alias es 'ssh es'
```

В результате теперь я могу вводить в командной строке `es`, чтобы войти в систему, или с помощью всех тех же двух букв обозначить хост в командах `scp` или `rsync`. Ведь круто?

Возникает соблазн создать двухбуквенные псевдонимы для всех серверов на свете. Однако вскоре вы обнаружите, что вспоминаете обозначения дольше, чем вводите имена с клавиатуры. Я ограничил себя несколькими серверами, к которым обращаюсь через SSH.

На странице `ssh_config(5)` справочной системы *man* перечислено много других опций конфигурации. Например, иногда я обращаюсь к серверу, для которого требуется весьма специфическая комбинация опций в командной строке. (Это доморощенная версия SSH-сервера, которая не только не обладает всеми функциональными возможностями, но просто отказывается работать, получая строку, которую не понимает.) Команда, которую я должен ввести, выглядит так:

```
$ ssh -x -o RSAAuthentication=yes -o PasswordAuthentication=yes -o
ChallengeResponseAuthentication=no -1 peter.example.net
```

Я мог бы установить псевдоним, но вместо этого изменил конфигурацию SSH, и все системы, использующие SSH, работают, как надо. Если сценарий, который я не могу изменить, использует SSH для доступа к тому серверу, он воспримет эти настройки.

Соответствующие строки в моем файле `~/.ssh/config` выглядят так:

```
Host peter.example.net
    ForwardX11 no
    RSAAuthentication yes
    PasswordAuthentication yes
    ChallengeResponseAuthentication no
    Compression no
    Protocol 1
```

У SSH-клиентов для Windows обычно есть графический интерфейс, позволяющий сохранить настройки профиля, которые следует использовать для конкретного сервера (или нескольких серверов).

Чем больше вы знаете про SSH, тем больше вы можете сделать. Есть много хороших книг и электронных справочников с подробным описанием SSH, например «SSH, The Secure Shell: The Definitive Guide» (SSH. Безопасная оболочка: полное руководство), O'Reilly. Если в SSH и есть то, что должен знать каждый системный администратор (но, возможно, не знает), то это следующее: как настроить открытые/закрытые (public/private) ключи, чтобы, не снижая уровень безопасности, исключить необходимость ввода пароля при обращении с одного конкретного компьютера к другому через SSH.

Make-файл для каждого хоста

Этот раздел относится исключительно к системам UNIX/Linux. Тот, кто работает в Windows, может его пропустить.

В системах UNIX/Linux важная информация часто хранится в обычных текстовых файлах, которые можно отредактировать вручную. В некоторых случаях после редактирования файла необходимо выполнить специальную команду, чтобы сообщить системе, что информация изменилась.

Например, после редактирования файла */etc/aliases* (к которому обращаются sendmail, Postfix и различные пакеты пересылки почты), вы должны выполнить команду *newaliases*. Ее легко запомнить, не так ли?

А какую команду надо выполнить после редактирования файла *transports* программы Postfix? Возможно, *newtransports*? Нет, это было бы слишком просто. Вы должны выполнить *postmap transports*. А еще есть команда *m4*, которую нужно выполнить после редактирования *m4*-файлов и т. д., и т. п.

Как обратиться к нужному серверу с помощью SSH

Предположим, у вас три сервера: *server1.example.com*, *server2.example.com* и *server3.example.com*. На них расположено много веб-сайтов, и вам трудно запомнить, какой сайт на каком сервере. Где находится сайт *www.everythingsysadmin.com* - на первом или третьем сервере? Вы думаете, что на третьем, но не исключено, что его перенесли на второй из-за нехватки места на диске. Зачем вообще все это запоминать? Нет никакой необходимости редактировать файл конфигурации. Просто укажите имя хоста веб-сайта в команде SSH! Например, введите *ssh www.everythingsysadmin.com*, и вы окажетесь на нужном сервере. Это довольно очевидно, но вы не поверите, как часто люди забывают о такой возможности!

У кого найдется время заучить все команды, которые нужно выполнить после редактирования файлов? Такие подробности должен помнить сам компьютер.

На помощь приходит команда *make*! Вы можете считать ее программным инструментом, чем-то вроде компилятора. Она позволяет вам указать, какую команду нужно запускать для обновления одного файла, если в другой были внесены изменения.



Команда *make* является одним из самых мощных инструментов системного администрирования, которые когда-либо были изобретены. Я слышал, что программисты тоже считают ее полезной!

У команды *make* больше функциональных возможностей, чем было мужей у Элизабет Тейлор, поэтому я лишь вкратце представлю ее вам. (Прочитав только первые две главы практически любой книги, посвященной команде *make*, вы узнаете 99% того, что необходимо для большинства задач системного администрирования, и в 10 раз больше, чем знают коллеги.)

Краткое описание команды *make*

Команда *make* читает файл конфигурации, весьма уместно именуемый *Makefile*. В этом файле находятся инструкции, которые сообщают команде *make*, что она должна делать.

Каждая инструкция имеет такой формат:

```
whole: partA partB partC
    команда, создающая whole
```

Инструкция начинается с имени файла, который должен быть создан. Затем стоит двоеточие, после которого указаны файлы, необходимые для построения главного.¹ В этом примере создается файл *whole*, и устанавливается соотношение между *partA*, *partB* и *partC*. Если один из них будет отредактирован, мы должны будем выполнить команду, создающую *whole*.

Приведу вполне реальный пример:

```
aliases.db: aliases
    newaliases
@echo Done updating aliases
```

¹ Традиционная особенность синтаксиса команды *make* состоит в том, что имена файлов, следующие за двоеточием, и отступы командных строк должны отделяться одним или несколькими символами табуляции, но ни в коем случае не пробелами; на распечатке они неразличимы, на это многие «попадались»; см. далее у автора. — *Примеч. науч. ред.*

Этот код означает, что, если файл *aliases* будет изменен, команда *newaliases* регенерирует файл *aliases.db*. В случае успеха будет выведено сообщение «Done updating aliases» (Выполнено обновление *aliases*).

Обратите внимание на то, что вторая и третья строчки кода написаны с отступом. Отступ обязательно достигается табуляцией, а не последовательностью пробелов. Почему? Наверное, создатель команды *make* хотел наказать меня за каждую попытку скопировать участок кода в системе, преобразующей табуляцию в пробелы. Впрочем, я не принимаю это на свой счет.

Обновление файла не происходит автоматически. Вы должны только запустить команду *make*:

```
Server1# make aliases.db
newaliases
Done updating aliases
Server1#
```

Вот так! Команда *make* прочитала свой файл конфигурации, выяснила, что файл *aliases* новее, чем *aliases.db* (сравнив их метки времени), и определила, что запуск команды *newaliases* приведет к обновлению файла *aliasesAb*. Попробуем запустить *make* еще раз:

```
Server1# make aliases.db
Server1#
```

Сообщение об обновлении не выводится. Почему? Потому что теперь согласно меткам времени ничего делать не нужно, ведь файл *aliases.db* новее файла *aliases*. Команда *make* ленива и выполняет минимум работы, необходимый для получения результата. Она принимает решения на основе меток времени в файлах.

Вот еще один пример кода в файле *Makefile*:

```
file1.output: file1.input
command1 <file.input >file.output

file2.output: file2.input
command2 file2.input >$$
```

В первом случае команда, которую следует выполнить, использует стандартные потоки ввода/вывода *stdin* и *stdout* (перенаправление обозначено символами *<* и *>*), чтобы прочитать *fileXinput* и произвести запись в *file, output*. Второй фрагмент аналогичен первому, но команда берет имя входного из командной строки и перенаправляет вывод... куда? Конструкция *\$\$* означает «файл, создаваемый этой инструкцией», в нашем случае это *file2.output*. Почему не придумали мнемоническое обозначение, вроде *\$te* или *\$this*? Неизвестно. Вам совсем не обязательно использовать обозначение *\$\$*, но с ним вы будете выглядеть умнее своих коллег.

Команда *make*, запущенная без параметров, выполняет первую инструкцию из файла *Makefile*. По традиции первая инструкция называется *all*, и она выполняет все инструкции, которые должны быть выпол-

нены по умолчанию. Таким образом, запуск *make* выполняет все важные инструкции. Возможно, это будут не все инструкции, а лишь те, которые вы хотите выполнить по умолчанию. Эта инструкция может выглядеть, например, так:

```
all: aliases.db access.db
```

Команда *make* без параметров проверяет, не обновлялись ли файлы *aliases.db* и *access.db*. Поскольку в инструкции *all* не указано никакой команды, файл с именем *all* создан не будет. Тогда *make* будет считать, что файл *all* устарел («не существует» эквивалентно «устарел»). Вскоре вы поймете важность такой интерпретации.

Не будем забывать, что команда *make* ленива. Если файл *access.db* устарел, а второй файл - нет, то она выполнит только действия по обновлению *access.db*. Если для обновления файла потребуются какие-то дополнительные действия, а для них что-то еще, то команда *make* рационально выполнит только необходимый минимум работы.

Кроме инструкции *all*, я обычно пишу еще пару-тройку полезных команд:

```
reload:
    postfix reload

stop:
    postfix stop

start:
    postfix start
```

Обсудим, что они означают. Если я введу *make reload*, команда *make* заметит отсутствие файла *reload* и запустит команду *postfix reload* в надежде, что та создаст такой файл. Ага! Я ее перехитрил! Указанная команда перегружает конфигурацию Postfix. Она не создает никакого файла *reload*! Когда я выполню *make reload* в следующий раз, команда *make* поступит точно так же. Другими словами, если вы хотите, чтобы некоторое действие выполнялось всегда, сделайте так, чтобы инструкция не создавала файл, который надеется создать команда *make*.

Имея код, приведенный выше, я могу перезагрузить, остановить и запустить postfix, введя команду *make reload*, *make stop* и *make start* соответственно. Если есть другие процессы, которые требуется остановить (например, IMAP сервер, веб-клиент электронной почты и т. д.), я включу необходимые команды в инструкции. Мне не нужно запоминать эти команды.

Сейчас я должен признать, что раньше чуть-чуть передернул. Я сказал, что каждая инструкция начинается с имени файла, который нужно создать, затем идет двоеточие, а затем - список файлов, образующих главный. Так вот, команде *make* неизвестно, состоит ли в действительности создаваемый файл из перечисленных. У нее нет способа

проверить это. Элементы, указанные после двоеточия, являются всего лишь параметрами, которые не должны быть устаревшими.

Приведу простой пример реального файла *Makefile*, который запускает Postfix и содержит инструкции обновления индекса для файлов *aliases* и *access*. В начале файла вы заметите константы (*NEWALIASES*, *PDIR* и т. д.), используемые далее в файле. Обратный слэш (\) в конце строки кода служит для переноса длинных строк:

```
NEWALIASES=/usr/sbin/newaliases
PDIR=/etc/postfix
POSTMAP=/usr/local/postfix/sbin/postmap

# Команды

all: $(PDIR)/aliases.pag $(PDIR)/aliases.dir \
      $(PDIR)/access.dir $(PDIR)/access.pag reload

reload:
    postfix reload

stop:
    postfix stop

start:
    postfix start

#
# Когда aliases изменится, сгенерировать файлы .pag and .dir
#
$(PDIR)/aliases.pag $(PDIR)/aliases.dir: $(PDIR)/aliases
    $(NEWALIASES)

#
# Когда access изменится, сгенерировать файлы .pag and .dir
#
$(PDIR)/access.dir $(PDIR)/access.pag: $(PDIR)/access
    $(POSTMAP) $(PDIR)/access
```

Теперь я могу отредактировать файл *aliases* или *access* и ввести команду *make*. Мне не нужно помнить, что команды обновления индексов сильно различаются. Я не должен помнить о необходимости перезагружать конфигурацию Postfix, потому что соответствующая команда включена в инструкцию *all*. Конструкция *reload* в конце *all* будет каждый раз запускать эту инструкцию.

С помощью команды *make* можно также поддерживать свежие версии файлов на разных серверах. Предположим, что на обоих наших почтовых серверах файлы *aliases* должны быть одинаковыми. Мы решаем отредактировать файл на одном сервере и скопировать его на сервер *server2*. Инструкция может выглядеть, например, так:

```
push.aliases.done: $(PDIR)/aliases
    scp $(PDIR)/aliases server2:$(PDIR)/aliases
    touch $@
```


Мы копируем файл на `server2` командой `scp`, затем применяем команду `touch` к файлу `push.aliases.done`. Поскольку этот файл создается после успешной операции копирования, мы можем построить инструкции так, что копирование будет выполняться только в случае необходимости. Мы также можем принудительно скопировать файл, если просто удалим `push.aliases.done` и введем команду `make`. Традиционно вводится инструкция `clean`, которая удаляет все файлы `*.done` и прочие файлы, сгенерированные автоматически.

В файлах, имена которых оканчиваются на `.done`, нет ничего особенного. Это обычные файлы с меткой времени или флагом для имени.

Рассмотрим развернутый пример. Имеются два файла, подлежащие индексации после редактирования: `aliases` и `access`. Если хотя бы один из них проиндексирован заново, выдается команда перезагрузки `Postfix`. Кроме того, оба файла копируются на `server2`, если они были изменены. Наконец, команда `cd /etc && make` выполняется на `server2` тогда и только тогда, когда на него был скопирован хотя бы один файл.

Будьте внимательны, создавая инструкции. Правильно указывайте параметры и применяйте команду `touch` к файлам `*.done`, если потребуется. Команда `make` выполнит лишь минимум работы, необходимый для обновления системы.

```
#
# Makefile для server1
#
NEWALISES=/usr/sbin/newaliases
PDIR=/etc/postfix
POSTMAP=/usr/local/postfix/sbin/postmap

#
# "Команды" высокого уровня
#
all: aliases.done access.done reload_if_needed.done push

push: push.done

reload:
    postfix reload

stop:
    postfix stop

start:
    postfix start

reload_if_needed.done: aliases.done access.done
    postfix reload
    touch reload_if_needed.done

clean:
    rm -f \
        $(PDIR)/aliases.pag $(PDIR)/aliases.dir \
        $(PDIR)/access.dir $(PDIR)/access.pag \
```

```

push.aliases.done push.access.done \
reload_if_needed.done

#
# Инструкции для конкретных файлов,
# которым требуется индексация/регенерация
#
# Если aliases изменится, сгенерировать файлы .pag and .dir
aliases.done: $(PDIR)/aliases.pag $(PDIR)/aliases.dir
$(PDIR)/aliases.pag $(PDIR)/aliases.dir: $(PDIR)/aliases
$(NEWALIASES)

# Если access изменится, сгенерировать файлы .pag and .dir
access.done: $(PDIR)/access.dir $(PDIR)/access.pag
$(PDIR)/access.dir $(PDIR)/access.pag: $(PDIR)/access
$(POSTMAP) $(PDIR)/access

#
# Копирование
#

push.done: push.aliases.done push.access.done
ssh server2 "cd /etc && make"
touch $@

push.aliases.done: aliases.done
scp $(PDIR)/aliases server2:$(PDIR)/aliases
touch $@

push.access.done: access.done
scp $(PDIR)/access server2:$(PDIR)/access
touch $@

```

Этот *Makefile* является для вас хорошей стартовой площадкой. Он довольно сложен, потому что нам нужна гарантия того, что Postfix перезагрузится, лишь когда это абсолютно необходимо.

Такой *Makefile* избавляет вас от необходимости помнить множество команд, в том числе те, которые необходимы для обновления конкретных файлов. Вы больше не боитесь забыть какую-то команду. Многие сложные процедуры теперь сводятся к двум шагам:

1. Отредактировать нужный файл.
2. Ввести команду `make`.

Команда *make* является универсальным инструментом для соединения нескольких автоматизированных процессов. Однажды я должен был объединить несколько процессов и процедур для трех больших сетей. В каждой сети была своя система сопровождения псевдонимов, хостов и прочей административной информации. Разобравшись в процедурах для каждой сети, я построил *Makefile* для главных серверов этих сетей. Имена инструкций верхнего уровня были одинаковыми

для всех трех сетей, но команды, которые они выполняли, для каждой сети были свои.

В мои стратегические планы входило создание нового главного сервера, который в конечном счете заменил бы все серверы, доставшиеся мне, так сказать, в наследство. Первоначально *Makefile* нового главного сервера просто вызывал команду *make* на трех главных серверах с помощью команды *rsh* (это было задолго до появления *ssh*). Затем я поочередно перенес инструкции на новый сервер. Вначале я решил, что новый главный сервер должен быть единственным источником информации для файла *aliases*. Я слил файлы *aliases* всех трех сетей и разместил результат на главном сервере. Протестировав его, я создал инструкции, которые копируют этот объединенный файл на прежние главные серверы так, словно это был их собственный файл. Я поступил аналогичным образом с каждым файлом и каждой базой данных.

Поскольку каждое изменение было незначительным и конкретным, я мог выполнять тестирование итеративно. Произведя буквально несколько сотен изменений, я добился того, что все серверы «пели в унисон». В этот момент мне не составило труда исключить старые главные серверы и поставить новый над всеми клиентами.



Любой файл, автоматически копируемый на другие серверы, должен обязательно содержать в начале комментарий, информирующий других системных администраторов, откуда файл пришел и где его следует редактировать.

Вот комментарий, который я пишу:

```
# ЭТОТ ФАЙЛ СОПРОВОЖДАЕТСЯ НА СЕРВЕРЕ:  
# server1.example.com  
# Редактируйте его при помощи команды: xed file.txt  
# Если вы отредактируете его на любом другом компьютере,  
# он будет перезагружен. БУДЬТЕ ВНИМАТЕЛЬНЫ!
```

Поскольку в комментарии упоминается *xed*, я должен пояснить, что это такое. Есть несколько программ с именем *xed*, но эту конкретную программу можно найти по адресу <http://www.nightcoder.com/code/xed>. Она вызывает редактор, которым вы обычно пользуетесь (*EDITOR* можно установить в *vi*, *rico*, *emacs* и т. д.), после того как заблокирует файл. Это обязательное условие для любого сайта, на котором один компьютер используют несколько системных администраторов. Если вы отслеживаете изменения в файле с помощью *RCS*, эта система регистрирует все попытки отредактировать файл. Вы получаете практически бесконечный откат и журнал, где записано, кто и что изменил. Если вы заметите, что последний месяц система ведет себя как-то странно, то проверьте, кто редактировал файл месяц назад. Будьте снисходительны: все мы допускаем ошибки.

Сложные задачи, выполняемые однократно

Автоматизация работы предполагает, что, выполняя нечто сложное, мы записываем последовательность действий. В этом случае повторить действия впоследствии много проще. Это как если бы мы строили для себя некоторую подстраховку.

Инкапсуляция сложной команды

Бывает, что на создание команды, которая делает именно то, что нужно, уходит несколько часов. Например, есть программа, создающая ISO-образы диска для последующей записи на CD-ROM. На странице ее описания перечислены сотни параметров, но команда, позволяющая создать образ, читаемый в Windows, UNIX и Mac OS X, достаточно проста:

```
$ mkisofs -D -l -J -r -L -f -P "Имя автора" -V "метка диска" -copyright \  
copyright.txt -o disk.iso /directory/of/files
```

Конечно, все можно проделать с помощью графического интерфейса, но вы не получите никакого удовольствия (а также возможности создать сценарий).

Эта команда позволяет вам делать то, что недоступно в большинстве графических интерфейсов, например сообщать об авторских правах, указывать имя автора и т. д.

Она хорошо встраивается в JJAT-файл (в DOS) или в сценарий оболочки UNIX/Linux.

Вот пример сценария `makeimage1`, в котором используется эта команда:

```
#!/bin/bash  
  
mkisofs -D -l -J -r -L -f -P "Limoncelli" -V `date -u +%m%d` $*
```

Конструкция `'date -u +%m%d'` создает метку тома, содержащую текущую дату.

Я долго не мог создавать хорошие сценарии, потому что не знал, как обрабатывать параметры командной строки. Расскажу, как следует копировать их в сценарий.

Конструкция `$*` в сценарии означает «любые элементы командной строки». Если вы введете:

```
$ makeimage1 cdrom/
```

то последовательность `$*` будет заменена на `cdrom/`.

Поскольку `$*` работает и с несколькими аргументами, вы можете ввести:

```
$ makeimage1 cdrom/ dir1/ dir2/
```

В этом случае вместо `$*` будут подставлены все три аргумента. Команда `mkisofs` сольет (`merge`) все три каталога на CD-ROM. Если вы хотите

указать конкретные элементы командной строки, пишете \$1, \$2 и т. д. В нашем примере \$1 будет соответствовать `cdrom/`, а \$2 - `dir1/`.

Еще одной причиной, не позволявшей мне писать хорошие сценарии, было мое неумение обрабатывать флаги командной строки, например `scriptname -q file1.txt`. Если сценарий, который требовалось написать, был настолько сложен, что требовал учета флагов командной строки, я использовал другой язык или вообще отказывался от создания сценария. Оказывается, в *bash* есть функция `getopt`, которая выполняет синтаксический разбор. Однако руководство по *bash* написано не совсем понятно. В нем сказано, как работает `getopt`, но не объясняется, как пользоваться этой функцией. Наконец, я нашел подходящий пример и с тех пор периодически копирую его. Неважно, как он работает. Чтобы воспользоваться им, вам совсем необязательно понимать, что он делает и почему. А выглядит он так:

```
args='getopt ab: $*'
if [ $? != 0 ]
then
    echo "Usage: command [-a] [-b file.txt] file1 file2 ..."
    exit -1
fi
set -- $args
for i
do
    case "$i"
    in
        -a)
            FLAGA=1
            shift
            ;;
        -b)
            ITEMB="$2" ; shift
            shift
            ;;
        --)
            shift; break
    ;;
    esac
done
```

Предполагается, что есть некая команда с флагами `-a` и `-b`. Второй флаг особый, потому что за ним должен следовать аргумент, например `-b file.txt`. Из первой строчки понятно, что за командой `getopt` следуют буквы, которые могут быть флагами. После флага, которому требуется дополнительный аргумент, стоит двоеточие. Далее мы видим оператор `case` (выбор) для каждого возможного аргумента. Код оператора либо устанавливает флаг, либо устанавливает флаг и сохраняет аргумент.

А что такое \$2? И какой смысл в --)? И что делает set? И как поживает Наоми? Все это вы сможете узнать потом. А сейчас воспользуйтесь образцом, и он будет работать.

(Ну ладно. Если вы вправду хотите в этом разобраться, прочитайте «Advanced Bash-Scripting Guide» (Расширенное руководство по написанию скриптов bash) по адресу <http://www.tldp.org/LDP/abs/html>.)

Теперь я приведу более развернутый пример, иллюстрирующий некоторые дополнительные возможности. Во-первых, в нем определяется функция usage, предназначенная для вывода справочной информации. Интересная особенность этой функции - команда echo, растянутая на несколько строк. Неплохо, не так ли? bash не возражает. Во-вторых, в этом сценарии проверяется наличие минимального количества (MINITEMS) элементов командной строки после обработки аргументов. Наконец, сценарий демонстрирует обработку флагов, переопределяющих умолчания.

Пожалуйста, пользуйтесь этим кодом, если вам понадобится преобразовать простой сценарий в сценарий, обрабатывающий параметры и флаги:

```
#!/bin/bash

MINITEMS=1

function usage
{
    echo "
Usage: $0 [-d] [-a author] [-c file.txt] [-h] dir1 [dir1 ...]

    -d  debug, don't actual run command
    -a  author name of the author
    -c  copyright override default copyright file
    -h  this help message
...
    exit 1
}

# Задание умолчаний:
DEBUG=false
DEBUGCMD=
AUTHOR=
COPYRIGHT=copyright.txt

# Обработка аргументов командной строки
# с возможным переопределением умолчаний

args=`getopt da:c:h $*`
if [ $? != 0 ]
then
    usage
fi
set -- $args
```

```
for i
do
  case "$i"
  in
    -h)
      usage
      shift
      ;;
    -a)
      AUTHOR="$2"; shift
      shift
      ;;
    -c)
      COPYRIGHT="$2"; shift
      shift
      ;;
    -d)
      DEBUG=true
      shift
      ;;
    --)
      shift; break;;
  esac
done

if $DEBUG ; then
  echo DEBUG MODE ENABLED.
  DEBUGCMD=echo
fi

# Проверка наличия минимального количества элементов
# командной строки
if $DEBUG ; then echo ITEM COUNT = $# ; fi
if [ $# -lt "$MINITEMS" ]; then
  usage
fi

# Если первый аргумент особый, запомнить его:
# THEITEM="$1" ; shift
# Клонировать эту строчку для каждого элемента,
# который хотите сохранить.
# Не забудьте при этом откорректировать значение
# переменной MINITEMS.

# Если вы хотите обработать остальные элементы,
# делайте это здесь:
#for i in $* ; do
#  echo Looky! Looky! I got $i
#done
```

```
if [ ! -z "$COPYRIGHT" ];  
then  
    if $DEBUG ; then echo Setting copyright to: $COPYRIGHT ; fi  
    CRFLAG="-copyright $COPYRIGHT"  
fi  
  
LABEL='date -u +%Y%m%d'  
  
$DEBUGCMD mkisofs -D -l -J -r -L -f -P "$AUTHOR" -V $LABEL $CRFLAG $*
```

Построение длинной командной строки

Самый лучший способ научиться сцеплять команды UNIX/Linux в один длинный канал - заглядывать через плечо того, кто этим занимается. Сейчас я попробую научить вас делать это, создав на ваших глазах небольшую утилиту.



Книга «Think UNIX» (Думайте в духе UNIX), Que, - отличный учебник по объединению инструментов UNIX/Linux в длинные команды.

Самой мощной технологией, представленной в UNIX/Linux, является возможность объединить несколько команд аналогично тому, как наращивают садовый шланг для полива. Если у вас есть программа, которая преобразует входной текст в верхний регистр, и программа, сортирующая строки в файле, то вы можете сцепить их друг с другом. В результате вы получите команду для преобразования строк в верхний регистр и их вывода в заданном порядке. Все, что от вас требуется, - это поставить символ «» между командами. Выходная информация одной команды поступит на вход следующей:

```
$ cat файл | toupper | sort
```

Тем, кто не знаком с UNIX/Linux, сообщу, что `cat` - это команда, выводящая файл. Программу `toupper` я написал для преобразования текста в верхний регистр, а `sort` - программа, сортирующая строки текста. Все они прекрасно стыкуются.

Теперь напишем более сложную утилиту. Это будет программа, определяющая, какой компьютер в вашей локальной сети наиболее вероятно заражен червем. Программа будет представлять собой один длинный канал.

Звучит неожиданно? Все, что будет делать наша программа, - это искать подозрительные на заражение компьютеры. Иными словами, она выведет список хостов, нуждающихся в более внимательном осмотре. Тем не менее, ваши коллеги будут удивлены, уверяя вас.

Программа не заменит вам хорошее антивирусное обеспечение, но я выбрал этот пример как хорошую иллюстрацию некоторых рудиментарных приемов программирования оболочки. Кроме того, вы узнаете кое-

Вернуться в каталог учебников

<http://учебники.информ2000.пф/учебники.shtml>

что новое о работе сетей. Когда мы закончим, в ваших руках будет простой инструмент, которым вы сможете пользоваться для диагностики этой конкретной проблемы в своей сети. С помощью этой программы мне удалось убедить руководство в необходимости купить настоящее антивирусное программное обеспечение.

Каков признак того, что компьютер инфицирован червем? Надо проверить, какие компьютеры чаще других отправляют ARP-пакеты.

Черви/вирусы/шпионские программы нередко пытаются соединиться со случайно выбранным компьютером локальной сети. Когда компьютер впервые пытается связаться с локальным IP-адресом, он отправляет ARP-пакет для выяснения Ethernet (MAC)-адреса. Нормальные (неинфицированные) компьютеры обычно общаются лишь с несколькими компьютерами: с серверами, которые им нужны, и с их локальным маршрутизатором. Если обнаружится, что какой-то компьютер отправляет значительно больше ARP-пакетов, чем остальные, скорее всего, это можно считать признаком его заражения.

Построим простой канал в оболочке, который будет собирать следующие 100 ARP-пакетов в сети и определять, какой хост сгенерировал больше пакетов, чем другие хосты того же уровня. Что-то вроде конкурса «Кто пошлет больше пакетов». В прошлую такую проверку я обнаружил, что червями заражены два из пятидесяти компьютеров сети.

Команды, приведенные ниже, должны работать в любой системе UNIX/Linux или другой UNIX-подобной системе. Вам понадобится команда *tcpdump* и доступ с правами *root*. Команда *which tcpdump* сообщит вам, установлена ли команда *tcpdump* в вашей системе. Проверка пакетов в сети имеет этические аспекты. Выполняйте ее, только если у вас есть разрешение.

Вот команда, получившаяся у меня в итоге (прошу прощения за испорченный сюрприз):

```
$ sudo tcpdump -l -n arp | grep 'arp who-has' | \  
head -100 | awk '{ print $NF }' | sort | uniq -c | sort -n
```

Она такая длинная, что не помещается в одной строке этой книги, поэтому я разбил ее на две строки, а символом переноса служит обратный слэш. Вам не нужно набирать обратный слэш при вводе команды.¹ Не следует и нажимать клавишу Enter в этом месте.

Вывод команды выглядит так:

```
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on en0, link-type EN10MB (Ethernet), capture size 96 bytes
```

¹ Если используемая вами терминальная программа допускает ввод строки такой длины (что обычно выполняется). Но если эту команду предполагается, в свою очередь, использовать в тексте скрипта, то при переносе необходимо набирать обратный слэш. - *Примеч. науч. ред.*

```
1 192.168.1.104
2 192.168.1.231
5 192.168.1.251
7 192.168.1.11
7 192.168.1.148
7 192.168.1.230
8 192.168.1.254
11 192.168.1.56
21 192.168.1.91
30 192.168.1.111
101 packets captured
3079 packets received by filter
0 packets dropped by kernel
```

Не обращайте внимание на две первые и три последние строки. В остальных строках указано количество пакетов и IP-адрес. Этот эксперимент показал, что хост 192.168.1.111 отправил 30 ARP-пакетов, а хост 192.168.104 - только один пакет. Большинство машин редко отправляли пакеты за исследуемый период, но два хоста отправили в 4-6 раз больше пакетов, чем остальные. Это свидетельствовало о проблеме. Быстрое сканирование антивирусной программой - и оба хоста стали как новенькие.

А теперь расскажу, как я строил эту командную строку. Я начал с команды:

```
$ sudo tcpdump -l -n arp
```

Здесь `sudo` означает, что следующая команда должна быть выполнена с правами пользователя `root`. Весьма вероятно, что она запросит пароль. Если в вашей среде нет `sudo`, вы можете воспользоваться аналогичной командой или выполнить эту последовательность от имени пользователя `root`.¹ Будьте внимательны. Человеку свойственно ошибаться, но настоящий храбрец не боится работать от имени `root`.

Команда `tcpdump` прослушивает Ethernet. Флаг `-l` нужен, если мы собираемся направить вывод другой программе. Дело в том, что, в отличие от других команд, `tcpdump` выполняет буферизацию выходных данных для ускорения работы. Однако при направлении вывода в канал нам это не нужно. Флаг `-p` отключает поиск в DNS для каждого найденного IP-адреса. Параметр `arp` означает, что команда `tcpdump` должна отслеживать только ARP-пакеты.

(Если вас беспокоит этическая сторона вопроса, то должен сообщить вам хорошую новость. Если вы отфильтруете все, кроме ARP-пакетов, то на выходе будет очень мало частной информации.)

Выполните эту команду на своем компьютере. Вообще, если вы будете проверять на практике все, о чем здесь прочитаете, вы многому научи-

¹ Во многих UNIX-подобных системах в этом может помочь команда `su`. — Примеч. науч. ред.

тесь. Эта команда не удаляет никакие данные. Но должен вас предупредить, что отслеживание пакетов может быть незаконным. Делайте это, только если у вас есть соответствующее разрешение.

Когда я запускаю эту команду, вывод выглядит так:

```
$ sudo tcpdump -n -l arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on en0, link-type EN10MB (Ethernet), capture size 96 bytes
19:10:48.212755 arp who-has 192.168.1.110 (85:70:48:a0:00:10) tell 192.168.
 1.10
19:10:48.743185 arp who-has 192.168.1.96 tell 192.168.1.92
19:10:48.743189 arp reply 192.168.1.2 is-at 00:0e:e7:7a:b2:24 19:10:48.
 743198 arp who-has 192.168.1.96 tell 192.168.1.111
^C
```

Чтобы прекратить вывод, я нажимаю клавиши **Ctrl-C**. Иначе он будет продолжаться вечно.

Если вы получили сообщение об ошибке доступа, скорее всего, вы запустили команду не как суперпользователь. Команда *tcpdump* может быть выполнена только пользователем *root*. Вы ведь не хотите, чтобы вашу сеть прослушивал кто попало, правда?

После заголовка идут строки вида «arp who-has X tell Y». Здесь Y - это хост, который задал вопрос. Вопрос имел примерно такое содержание: «Уважаемый хост с IP-адресом X, не сообщите ли вы свой Ethernet (MAC)-адрес?» Вопрос отправляется широкоэвещательным пакетом, поэтому мы видим все ARP-запросы в нашей локальной сети. Однако мы видим не так уж много ответов, поскольку они отправляются в виде однонаправленных пакетов, а мы находимся на концентраторе. В данном случае виден только один ответ, потому что мы находимся на одном концентраторе с тем компьютером (а, может, это компьютер, выполняющий команду, - этого я вам не скажу). Тем не менее, все хорошо; ведь нам нужны только те, кто задает вопросы.

Итак, у вас есть источник информации. Преобразуем ее так, чтобы ею можно было воспользоваться.

Во-первых, изолируем те строки вывода, которые нас интересуют. В данном случае это строки, содержащие «arp who-has»:

```
$ sudo tcpdump -l -n arp | egrep 'arp who-has'
```

Мы можем запустить эту команду и убедиться, что она ведет себя, как ожидалось. Единственная проблема в том, что команда работает безостановочно, пока мы не нажмем **Ctrl-C**. Мы хотим получить достаточное количество строк и затем обработать их. Ограничимся первой сотней строк:

```
$ sudo tcpdump -l -n arp | grep 'arp who-has' | head -100
```

Снова запустим команду и убедимся, что все хорошо. Во время тестирования этой команды у меня не хватило терпения, и я изменил 100 на 10.

Вернуться в каталог учебников

<http://учебники.информ2000.рф/учебники.shtml>

Это дало мне уверенность, что команда работает правильно, и в окончательном варианте я поставил 100. Вы, конечно, заметите в выводе кучу заголовков. Они направляются в поток `stderr` (прямо на экран) и не передаются в команду `grep`.

Итак, у нас есть сто строк с интересующими нас данными. Настало время статистических подсчетов. Какие хосты генерируют больше всего ARP-пакетов? Нам нужно выделить IP-адреса всех хостов, сгенерировавших ARP-пакет, и как-то подсчитать их количество. Начнем с выделения IP-адресов. Это шестое поле каждой строки, и мы можем воспользоваться следующей командой:

```
awk '{ print $6 }'
```

Этот маленький фрагмент кода *awk* является прекрасной идиомой для выделения конкретного поля из каждой строки текста.

Должен сознаться, что мне было лень подсчитывать, в каком поле находятся данные, которые я хотел выделить. Было похоже, что это пятое поле, и я сначала пробовал указать \$5. Не сработало, тогда я указал \$6. Конечно! Мне следовало помнить, что в *awk* поля нумеруются с единицы, а не с нуля. Преимущество тестирования командной строки на каждом шаге в том и заключается, что мы достаточно рано обнаруживаем такие мелкие ошибки. Представьте, что было бы, если бы я написал всю командную строку сразу, а затем стал бы искать эту ошибку!

Я ленив и нетерпелив. Мне не хотелось каждый раз ждать, пока будут собраны все сто ARP-пакетов. Поэтому я сохранил их один раз и затем использовал эти результаты.

Я сохранил результаты во временном файле:

```
$ sudo tcpdump -l -n arp | grep 'arp who-has' | head -100 >/tmp/x
```

Затем я применил свой код *awk* к этому временному файлу:

```
$ cat /tmp/x | awk '{ print $5 }'  
tell  
tell  
tell  
tell  
...
```

Оказывается, мне нужно не пятое поле. Попробуем шестое:

```
$ cat /tmp/x | awk '{ print $6 }'  
192.168.1.110  
192.168.1.10  
192.168.1.92  
...
```

Да, так лучше.

Как бы то ни было, впоследствии я понял, что можно ублажать свою лень другим способом. Конструкция `$NF` обозначает последнее поле и позволяет мне вообще ничего не подсчитывать:

```
$ cat /tmp/x | awk '{ print $NF }'  
192.168.1.110  
192.168.1.10  
192.168.1.92  
...
```

А почему не `$LF`? Ну, это было бы слишком просто. А если серьезно, `NF` означает «number of fields» (количество полей). То есть `$NF` обозначает `NF`-е поле слева. Но все это неважно. Просто запомните, что, если вам нужно последнее поле в строке, вы можете добавить `$NF` в `awk`:

```
$ sudo tcpdump -l -n arp | egrep 'arp who-has' \  
| head -100 | awk '{ print $NF }'
```

Итак, на выходе мы получаем список IP-адресов. Проверьте это. (Именно так! Проверьте. Я подожду.)

Теперь мы хотим подсчитать, сколько раз каждый IP-адрес появляется в нашем списке. Есть одна идиома, которую я всегда применяю именно для этой цели:

```
sort | uniq -c
```

Она сортирует данные, а затем выполняет команду `uniq`, которая исключает повторения из отсортированного списка. (То есть технически она устраняет повторения соседних строк, а сортировка списка гарантирует, что одинаковые строки будут соседствовать.) Флаг `-c` включает подсчет повторений и вставляет это число в начало каждой строки. Результат выглядит так:

```
...  
11 192.168.1.111  
7 192.168.1.230  
30 192.168.1.254  
8 192.168.1.56  
21 192.168.1.91  
...
```

Мы почти у цели! Мы знаем, сколько ARP-пакетов отправил каждый хост. Последнее, что мы должны сделать, - это отсортировать список, чтобы выявить самые «общительные» хосты. Для этого добавляем в конец канала конструкцию `| sort -n`, которая выполнит сортировку по номеру:

```
$ sudo tcpdump -l -n arp | egrep 'arp who-has' | head -100 \  
| awk '{ print $NF }' | sort | uniq -c | sort -n
```

Выполнив эту команду, мы увидим отсортированный список. Если сеть не очень загружена, выполнение команды займет какое-то время.

В локальной сети из 50 компьютеров, когда было мало пользователей, мне понадобилось около часа. Но это было уже после обнаружения компьютеров со шпионскими программами и их очистки. Перед этим на сбор ста ARP-пакетов я потратил всего несколько минут.

В вашей домашней сети с одним или двумя компьютерами на выполнение команды может уйти несколько дней. Хосты кэшируют информацию, собранную ими с помощью ARP-запросов. Поэтому, поработав некоторое время, компьютер крайне редко отправляет ARP-пакеты,¹ если единственный компьютер, с которым он общается (в локальной сети), - это ваш маршрутизатор.

Зато в сети из ста хостов подозреваемые обнаруживаются достаточно быстро.

Теперь в вашем распоряжении имеется простой инструмент обнаружения атаки червей. Он не заменит тысячдолларовую систему распознавания атак и хорошую антивирусную программу, но, безусловно, поможет вам локализовать проблему, если она возникнет. Главное - он бесплатный, а вы узнали кое-что новое о программировании оболочки.

Если вам захотелось отточить свое мастерство программиста, вот несколько небольших проектов:

- Команда *tcpdump* выводит некоторую информацию в поток `stderr`. Есть ли способ предотвратить вывод этих сообщений? Если нет, то как получить менее «засоренный» вывод?
- Преобразуйте описанную команду в сценарий. Поместите его в свой каталог *bin*, чтобы воспользоваться им в будущем.
- Расширьте этот сценарий так, чтобы можно было определять, какой NIC сканировать, или добавьте другие возможности, которые сочтете нужными.
- Можно запрограммировать команду *tcpdump* так, что она будет собирать только ARP-пакеты типа «who-has», и вы сможете обойтись без команды *grep*. Для этого изучите *tcpdump* поглубже.
- Команда *tcpdump* может выполнять те же действия, что и `head -100`. Для этого изучите *tcpdump* поглубже. Получилось ли то же самое, что при `head -100`? Лучше или хуже стало?
- *awk* - это полноценный язык программирования. Воспользуйтесь им, чтобы исключить аргументы `grep` и `head`. Как вы думаете, почему я предпочел выполнять три процесса, вместо того чтобы просто возложить все на *awk*?

¹ Хосты кэшируют ARP-информацию, но на весьма непродолжительное время - порядка нескольких минут - в зависимости от используемой ОС, т. е. отправка ARP-запросов будет редкой, но не столь уж крайне редкой. - Примеч. науч. ред.

Microsoft Excel как альтернативный графический интерфейс

Разработка графического интерфейса приложения требует 90% всех усилий. Вот способ создания графического интерфейса для ленивых: храните данные в Microsoft Excel, но напишите макрос, который будет загружать эти данные на сервер для последующей обработки.

Однажды я таким образом написал целое приложение. У нас был веб-сайт, на котором были представлены различные события. Я устал обновлять веб-страницу вручную, но понимал, что для самостоятельной поддержки веб-сайта у секретарши не хватает технических навыков.

Тогда я задумался над графическим интерфейсом, позволяющим вносить обновления даже неподготовленному пользователю. У меня были громадные планы: большая база данных MySQL и движок на PHP, который позволил бы пользователям входить в систему, обновлять информацию, пополнять список событий и т. д. Система должна была автоматически генерировать веб-страницы. Все было прекрасно на бумаге, и я не сомневаюсь, что если бы в моем распоряжении было сто лет на написание кода, результат был бы замечательным.

Потом я понял, что на самом деле все изменения будет вносить только один человек. Тогда я создал для секретарши электронную таблицу Excel со всей необходимой информацией и написал макрос, который сохранял эту таблицу дважды: один раз на сервере в виде текстового файла с символом табуляции в качестве разделителя, а второй раз - в виде XLS-файла. Процесс на сервере должен был анализировать файл с табуляцией и генерировать веб-страницу автоматически.

Пример таблицы показан на рис. 13.2.

Создание кнопки выполняется в несколько шагов.

Во-первых, для запоминания нужных действий используйте функцию записи макроса:

1. Запишите макрос: последовательно выберите команды меню Tools (Сервис) → Macro (Макрос) → Record New Macro (Начать запись).
2. Назовите макрос Save (Сохранить).
3. Сохраните таблицу как файл с символом табуляции в качестве разделителя на сетевом файловом сервере.
4. Сохраните файл в формате MS Excel Workbook (Книга Microsoft Excel) (.xls) в своем каталоге.

Важно, чтобы последняя операция сохранения файла использовала самый мощный формат MS Excel Workbook (Книга Microsoft Excel), потому что таким образом будет установлен формат сохранения по умолчанию. Если затем кто-то сохранит файл с помощью команды File (Файл) → Save (Сохранить), то будет использован именно этот формат.

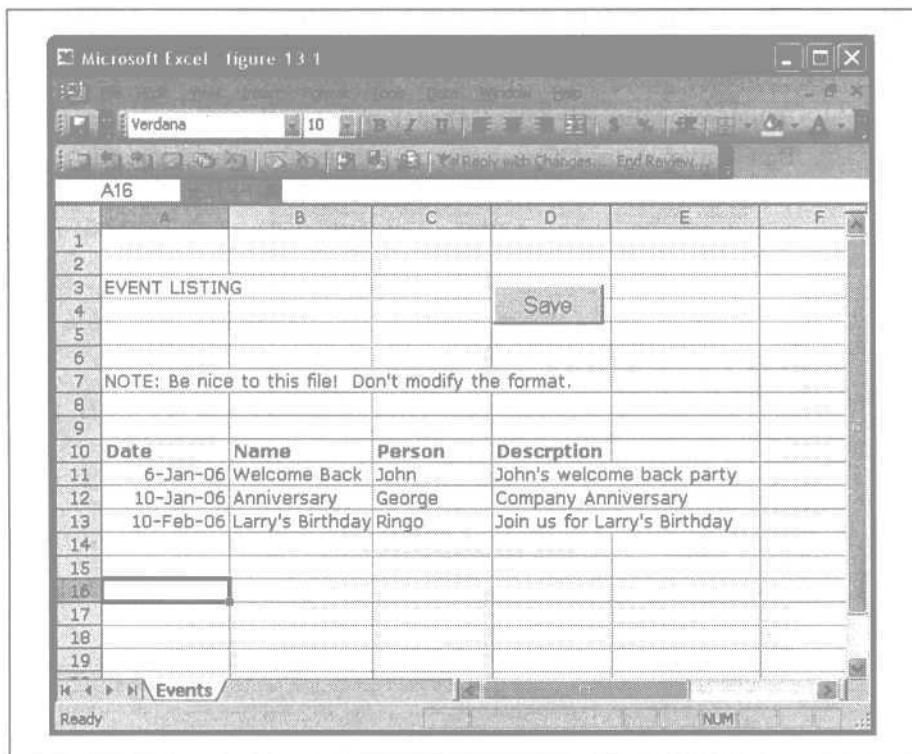


Рис. 13.2. Электронная таблица со списком событий

5. Щелкните по кнопке Stop (Остановить запись) на панели инструментов, появляющейся на экране в режиме записи макроса.

Теперь создайте кнопку и закрепите за ней макрос:

1. Выведите панель инструментов Forms (Формы): выберите команду View (Вид) → Toolbars (Панель инструментов) → Forms (Формы).
2. Щелкните по элементу Button (Кнопка) (выглядит как обычный прямоугольник).
3. Нарисуйте кнопку в том месте, где она должна находиться в таблице.
4. В ответ на вопрос выберите макрос, который только что создали.
5. Если вам впоследствии понадобится отредактировать кнопку, щелкните по ней при нажатой клавише Ctrl.

Протестируйте макрос, щелкнув по кнопке. Отлично! Все работает! Проверьте дату и время сохранения файлов, чтобы убедиться, что таблица была сохранена дважды. (Возможно, Excel дважды спросит вас, нужно ли замещать существующий файл. Ответьте «Да».)

Если вы хотите немного подправить макрос, это просто. Первое, что я сделал, - отредактировал макрос как раз в том месте, где выполняется сохранение файла:

1. Выберите команду Tools (Сервис) → Macro (Макрос) → Macros (Макросы).
2. Выберите макрос в списке открывшегося окна и щелкните по кнопке Edit (Изменить). Откроется редактор Visual Basic.
3. Закончив редактирование, сохраните файл и закройте редактор Visual Basic.



В макросах Microsoft для переноса длинных строк используется символ подчеркивания (_).

Окончательный макрос выглядит так:

```
Sub Save()  
  
    Macro recorded 5/22/2005 by Thomas Limoncelli  
  
    ActiveWorkbook.SaveAs Filename:= _  
        "Y:\calendar\EventList.txt", FileFormat:= _  
        xlText, CreateBackup:=False  
    ActiveWorkbook.SaveAs Filename:= _  
        "Y:\calendar\EventList.xls", FileFormat:= _  
        xlNormal, Password:="", WriteResPassword:="", _  
        ReadOnlyRecommended:=False _  
        , CreateBackup:=False  
  
End Sub
```

Теперь, когда у меня есть файл с символами табуляции в качестве разделителей, хранящийся на файловом сервере, мне нетрудно написать сценарий, который откроет этот файл, извлечет из него полезную информацию и на основании этой информации сгенерирует веб-страницу.

С тех пор я часто пользуюсь этим приемом в ситуациях, когда мне не хочется писать пользовательский интерфейс, а пользователь умеет работать с MS Excel.

Предоставление другим пользователям прав суперпользователя

Меня нередко просят предоставить обычным пользователям право выполнять привилегированные операции, разрешенные только администратору. Это может быть опасно, и здесь требуется большая осторожность.

В UNIX/Linux есть утилита *sudo*, позволяющая системному администратору предоставить пользователю возможность выполнить какую-нибудь команду под видом другого пользователя. Это очень строгая

утилита, и она требует, чтобы системный администратор четко указал, какой пользователь (пользователи) какую команду (команды) будет выполнять от имени какого пользователя.

Например, вы можете сконфигурировать утилиту так, что конкретный пользователь сможет выполнить некоторую команду от имени *root*. Не сомневайтесь, что *sudo* позволит только этому человеку выполнять от имени *root* только эту команду, но важно, чтобы эта утилита проверяла параметры и не запрещала привилегированным пользователям выполнять разрешенные им операции.



Очень рискованно создавать систему, в которой «обычным» пользователям разрешено выполнять «привилегированные» операции. В истории компьютерной безопасности полно случаев, когда добросовестные программисты случайно создавали в системе безопасности «дыры», позволяющие любому пользователю выполнять любую команду от имени пользователя *root* или *administrator*.

Если вы испытываете сомнения, прочитайте книгу по безопасности или поищите ответы в FAQ.

Предположим, что UNIX-команду *mount* для обращения к CD-ROM можно выполнять только пользователю *root*. Недопустимо конфигурировать *sudo* так, чтобы рядовой пользователь мог выполнить команду *mount* с любыми параметрами от имени *root*. Так он сможет вызвать сбой в работе системы или нарушить ее безопасность. Гораздо лучше, если вы разрешите пользователю выполнять от имени *root* некую новую команду (скажем, *mounted*). Эта команда убедится, что пользователь указал именно те дисководы, к которым ему разрешено обращаться (с учетом умолчаний), и выполнит необходимые действия. Вероятно, вы предоставите этому пользователю и команду *unmounted*.

Автоматизируя операции, выполняемые другими пользователями, я предпочитаю строить три уровня:

- **Уровень 1.** Программа, выполняющая основную задачу.
- **Уровень 2.** Программа, которую пользователь запустит при помощи *sudo*. Она примет входные данные, проверит их, убедится, что он не пытается выполнить подозрительные действия, и вызовет первую программу.
- **Уровень 3.** Дружественный пользователю интерфейс доступа к предыдущим слоям, например веб-интерфейс или программа с меню.

Приведу пример. В одной фирме, где я работал, имелась процедура публикации новой версии веб-сайта фирмы в Интернете. В процедуре были задействованы три веб-сервера (на самом деле это были виртуальные серверы на двух компьютерах, но эти подробности несущественны).

www-draft.example.com

Здесь разрабатывалась новая версия нашего веб-сайта.

www-qa.example.com

Сюда копировался новый вариант сайта для тестирования качества. Сразу после создания копии для этих файлов устанавливался режим «только чтение». Если отдел контроля качества принимал сайт, мы должны были иметь гарантию, что опубликованным в Интернете окажется именно представленный вариант.

www.example.com

Это был собственно сайт, который видела публика.

Веб-дизайнеры просили системных администраторов скопировать новую версию на *www-qa.example.com*. Отдел контроля качества, одобрив сайт, сообщал системным администраторам, что его можно опубликовать.

Выполнение каждой из этих двух операций было автоматизировано командами:

readyforqa

Копировала эскиз сайта на сайт отдела контроля качества.

golive

Копировала сайт отдела контроля качества на сайт в Интернете.

Отделу маркетинга требовался способ внесения экстренных изменений, когда сотрудники отдела контроля качества были вне досягаемости. Мы создали еще одну команду:

emergency-draft-to-live

Она копировала эскиз сайта на сайт в Интернете после того, как несколько раз переспрашивала «Are you sure?» (Вы уверены?).

Эти три сценария образовали Уровень 2, который я упомянул выше. Уровнем 1 был сценарий, который фактически копировал данные с одного сайта на другой, попутно создавая резервную копию и устанавливая запрет на запись в файлы (а также меняя владельцев файлов). Уровень 1 был доступен только из учетной записи *root*, потому что он менял владельцев файлов и обращался к серверам по защищенным каналам.

Команда *sudo* была запрограммирована так, как показано в табл. 13.1.

Таблица 13.1. Таблица разрешений на обновление веб-сайта

	Веб-разработчики	Контроль качества	Маркетинг
Readyforqa	X		X
Golive		X	
Emergency-draft-to-live			X

Мы приложили определенные усилия, чтобы заставить руководство подписаться под этой схемой, т. е. поставить свои реальные подписи

для гарантии того, что они понимают схему, с которой согласны на словах. Процесс получения подписей, как правило, шел очень туго. Он длился неделями. Представление информации в виде схемы облегчило руководству принятие решений. Менеджеры могли изучать ее и вносить изменения, сколько захотят. Преобразование окончательного варианта в файл конфигурации *sudo* было делом техники.

В отношении Уровня 3 мы приняли решение о необходимости предоставить пользователям удобный способ обращения к этим командам. В качестве варианта мы рассматривали веб-интерфейс, но в данном случае пользователи удовлетворились программой, которая выводила меню с перечнем опций и запускала нужную команду.

Меню работало без всякой привилегий (т. е. не под *sudo*), но вызывало программы Уровня 2 с помощью *sudo*, если это требовалось.

Резюме

- Автоматизация - великая вещь. Она экономит ваше время. Кроме того, она позволяет вам поручить работу другому менее квалифицированному сотруднику.
- Есть четыре типа задач, с которыми сталкиваются системные администраторы:
 - Простые задачи, выполняемые однократно
 - Сложные задачи, выполняемые однократно
 - Простые задачи, выполняемые часто
 - Сложные задачи, выполняемые часто
- Сложные однократные задачи и простые задачи, выполняемые часто, являются самыми подходящими кандидатами на автоматизацию. Несмотря на возможный соблазн автоматизировать сложные задачи, выполняемые часто, для их выполнения лучше использовать готовые программные пакеты (коммерческие или бесплатные).
- Чтобы автоматизировать процесс, сначала убедитесь, что вы в состоянии выполнить его вручную. Затем задокументируйте все шаги и автоматизируйте каждый шаг в отдельности. После этого соберите шаги воедино.
- Используя псевдонимы, вы облегчите себе ввод команд. Это справедливо и в отношении систем с командной строкой, и для приложений, например SSH. Установите псевдоним по возможности максимально близко к приложению. Например, если задать псевдоним в файле конфигурации SSH, то все системы, работающие с SSH, будут использовать этот псевдоним.
- Команда *make* систем UNIX/Linux - исключительно мощный инструмент. Она предназначена не только для программистов. Вы можете применять ее для автоматизации задач системного администриро-

- вания. В системах UNIX/Linux, особенно на серверах, создавайте файл *Makefile* в каталоге */etc* для автоматизации повседневных задач, таких как индексация псевдонимов, копирование данных и т. д.
- Языки оболочек *bash* и */bin/sh* гораздо сложнее и мощнее, чем вы думаете. Примеры, приведенные в этой главе, демонстрируют, как следует обрабатывать опции командной строки и даже как написать небольшой детектор вредоносных программ!
 - Если вы пишете длинную командную строку, тестируйте ее по частям.
 - Если вы пишете код для пользователей, возрастает важность интерфейса. Существуют технические приемы, позволяющие создавать хорошие пользовательские интерфейсы. Вы можете избавить себя от хлопот, переложив ввод данных на программу вроде MS Excel. Вы также можете создать программу с меню или веб-интерфейс, чтобы позволить обычным пользователям получать доступ к привилегированным системам.
 - Создавая код, позволяющий пользователям выполнять привилегированные операции, будьте предельно внимательны. В качестве базы используйте имеющиеся хорошо зарекомендовавшие себя средства безопасности, такие как *sudo*. С помощью таблицы разрешений объясните руководству, какой пользователь какие привилегии будет иметь. Разрешать или запрещать - дело руководства, а ваша задача заключается в том, чтобы помочь ему разобраться в вопросе. Прежде чем ввести систему в эксплуатацию, заручитесь согласием руководства.
 - Развиваясь как системный администратор, вы будете автоматизировать все больше и больше задач. Я советую вам изучить язык программирования, подходящий для целей системного администрирования, например Perl, Python, Ruby или Shell. Также обращайте внимание на технику работы в конкретных операционных системах. Эти вопросы освещаются в книгах серии «Cookbook» (Сборник рецептов) издательства O'Reilly, упомянутых ранее в этой книге.

Эпилог

Поздравляю! Вы прошли весь путь до конца. Что теперь?

Во-первых, я рекомендую вам перечитать эту книгу. Повторенье мать ученья. Если вы перечитаете (или хотя бы просмотрите) книгу, пока она еще свежа в вашей памяти, положительный эффект усилится.

Во-вторых, мы лучше учимся на практике. Чем больше вы применяете на практике приемы, изложенные в этой книге, тем лучше вы ими овладеваете. В какой-то момент вы обнаружите, что усвоили эти приемы достаточно хорошо, чтобы творчески развивать их в зависимости от вашего стиля жизни или конкретной ситуации. Один мой читатель заметил, что ему стало легче вести списки ежедневных дел, когда он в своем органайзере поменял местами странички списка дел и расписания. Кто бы мог подумать! Хорошо все, что приносит хорошие результаты! Я лишь советую опробовать сперва мой подход, чтобы получить представление о системе в целом.

В-третьих, будьте готовы к пробуксовке. Иногда вы будете возвращаться к своим старым привычкам. В этом нет ничего страшного, если вы отдадите себе отчет в происходящем и при первой возможности вернетесь к изложенной системе. Не исключено, что вам будет легче это сделать, перечитав соответствующую главу.

В-четвертых, вы, вероятно, захотите прочитать традиционные книги по тайм-менеджменту, не ориентированные именно на системных администраторов. Моя книга ориентирована на специфику профессии системного администратора и не уделяет внимание многим общим вопросам, хорошо освещенным в других работах. Рекомендую прочитать книгу Дэвида Аллена (David Allen) «Getting Things Done: The Art of Stress-Free Productivity»¹, <http://www.davidco.com>.

Что делать со всем этим «новым» свободным временем?

Приемы, описанные в этой книге, подарят вам несколько свободных часов (если не дней) в неделю. Экономя чуть больше часа каждый день, вы выполните тот же объем работы за 4-дневную рабочую неделю.

¹ Дэвид Аллен «Как привести дела в порядок: искусство продуктивности без стресса», Вильямс, 2007.

Что делать с этим свободным временем?

Пожалуйста, не растрачивайте его впустую. Умоляю вас. Едва начав претворять приемы тайм-менеджмента в свою жизнь, я стал отдавать высвободившееся время моей тогдашней страсти - чтению новостей Usenet. Думаю, современный эквивалент этого занятия - чтение RSS-каналов с новостями, блогов, веб-сайтов и т. д. Все это пустая трата времени. Пожалуйста, не расходуйте свое вновь найденное время, умоная непроизводительные дела в своей жизни.

У меня есть идея получше.

Отведите это вновь найденное время на борьбу с несправедливостью.

Самый распространенный вид несправедливости, ежедневно наблюдаемый мной, - это то, как корпорации крадут наши жизни у наших семей. Однажды утром мы просыпаемся и обнаруживаем, что наши дети выросли, не зная нас, или что наши любимые ушли, потому что нас почти никогда не было рядом. И мы еще удивляемся: «Как быстро промчалось время!»

Раньше существовал так называемый «неявный общественный договор». Мы работаем на фирму 40 часов в неделю, а она платит нам, чтобы у нас были средства к существованию, и что-то отчисляет в пенсионный фонд нам на старость. Это была честная сделка. Но теперь корпорации отбирают у нас все больше времени без какой бы то ни было компенсации. Профессионал работает 60-70 часов в неделю, а потом подпадает под массовое сокращение штатов из-за ошибочных решений бестолковых топ-менеджеров, зарплата которых в 100, если не в 1000, раз превышает его зарплату. В 1990-е годы я работал в AT&T/Lucent, и нам постоянно напоминали, что могут уволить нас в любой момент независимо от того, насколько хорошо мы справляемся с работой. Нам говорили, что надо радоваться переходу от гарантированных пенсий к принципу «каждый сам за себя» в соответствии с новым пенсионным планом, принятым фирмой. И при этом в последние годы моей работы в этой компании руководство было поражено и встревожено снижением лояльности сотрудников. Лояльность - улица с двусторонним движением.

Хотите совершить нечто радикальное, революционное? Воспользуйтесь приемами, изложенными в этой книге, чтобы восстановить 40-часовую рабочую неделю и повысить качество личной жизни:

- **Отработав 40-часовую неделю, отправляйтесь домой.** Будьте настолько же лояльны к своему работодателю, насколько он лоялен к вам. Идите домой, отработав ежедневные 8 или еженедельные 40 часов.
- **Уделяйте больше времени любимому человеку.** Каждый день дарите ей/ему поцелуй продолжительностью не меньше 12 секунд (не 12 быстрых поцелуев; считайте до 12 во время одного долгого поцелуя. Почувствуйте разницу!) Говорите, как высоко вы цените ее/его.

Включайте в свое расписание романтические свидания. Если в вашем карманном компьютере есть генератор случайных чисел, ежедневно планируйте звонок любимому человеку на случайное время (отведите на разговор хотя бы 15 минут, чтобы не показаться торопливым).

- **Проводите больше времени с детьми.** Если у вас есть дети, занимайтесь с ними чем-то еще, кроме совместного просмотра телепередач. Не знаете, чем заняться? Попробуйте спросить у них. Все еще не знаете? Обратитесь туда, где профи отыскивают ответы на все свои вопросы: поищите в Google строку «Куда пойти с детьми в [ваш город]». Нет своих детей? Станьте любящей тетушкой или добрым дядюшкой, о которых, возможно, сами мечтали в детстве.
- **Звоните родителям и другим людям, сыгравшим важную роль в вашей жизни.** Включите в расписание регулярный звонок родителям и выделяйте на разговор не меньше часа. Они будут вам очень благодарны, а вы оцените важность такого общения, когда их не станет.

В нашем мире есть и множество других форм несправедливости. Мои родители воспитали меня в убеждении, что люди не должны голодать, что расизм - зло, потому что необходимо ко всем людям относиться одинаково, и что мир на Земле - это Божья воля. Поэтому я борюсь с нищетой, расизмом и милитаризмом, где только могу.

Решите для себя, какой вид несправедливости беспокоит вас больше всего, и используйте для борьбы с ним свои научно-технические знания:

- **Помогите какой-нибудь общественной организации, борющейся с несправедливостью, организовав ее веб-сайт.** Найдите веб-сайт такой организации и помогайте его поддерживать. А еще лучше, установите систему управления контентом или ПО для ведения блогов, чтобы волонтеры этой организации могли поддерживать сайт без вашей помощи. Обеспечьте сайт системой, позволяющей вносить пожертвования «одним щелчком», например <http://www.Just-Giving.com>.
- **Помогите общественной организации в обслуживании компьютеров.** Раз в неделю заходите в офис общественной организации, чтобы проверить состояние компьютеров и ответить на технические вопросы. Установите там автоматически обновляемые антивирусные программы. Проверьте работу системы резервного копирования данных. Если организации потребуются новые приложения, окажите помощь в их поиске и установке.
- **Станьте членом общественного совета школы, которую закончили.** В детстве я был очень недоволен тем, что в моей школе упор делался на спортивные занятия, а не на преподавание точных наук. Места в школьном совете часто можно получить без всякого конкурса, но иногда соискателям приходится участвовать в избирательных компаниях с бюджетом в несколько сотен долларов. Получив место

в школьном совете, вы сможете напрямую влиять на бюджет и приоритеты развития школы. Субсидируйте программы борьбы с хулиганством, шахматные клубы, общества борьбы за права сексуальных меньшинств, преподавание музыки и живописи.

- **Примите участие в работе государственного муниципального или общественного учреждения.** Ситуация с принятием плохо продуманных законов не изменится, пока в работе государственных учреждений не примет участие достаточное количество профессионалов.

Борьба с несправедливостью подобна разбрасыванию желудей на своем пути. Вернувшись, вы можете обнаружить выросшее деревце, в другой раз ничего не найдете. Впрочем, в большинстве случаев вы так никогда и не узнаете, насколько вы изменили мир и скольким людям вы помогли. Вы просто должны верить, что ваши усилия потрачены не зря.

Мир вам.

— Том Лимончелли

Алфавитный указатель

A

Allen, David, 219

D

DateBook V, программа для портативного компьютера, 98

DVR, цифровые видеоманитоны, 165

E

Excel как альтернативный графический интерфейс, 211, 214

L

Life Balance, программа для портативного компьютера, 98

Linux Makefiles, 193

Loura, Ralph, 126

M

make, команда, 194

Makefiles (UNIX/Linux), 193

P

PDA (Personal Digital Assistant), портативный компьютер, 26

DateBook V, программа для портативного компьютера, 98

Life Balance, программа для портативного компьютера, 98

Цикл и, 98

Q

QoS (quality of service), 125

R

ROI, возврат на инвестиции, 132

RT, запросы и, 49, 153

S

sudo, утилита, 214

T

Tivo, 165

U

UNIX Makefile, 193

Usenet, непроизводительные затраты времени, 161

W

Wiki

введение, 178

нотация, 179

предотвращение вандализма, 180

ссылки на страницы, 179

A

автоматизация

Excel как альтернативный графический интерфейс, 211, 214

Makefile (UNIX/Linux), 193

sudo, утилита, 214

введение, 183

инкапсуляция, 201

категории, 184

кодирование, 187

командная строка, 205, 211

проблема выбора, 184, 186

программа, 184

ручной этап, 186
сокращения для
имен компьютеров, 191
команд, 189
сценарий, 184
тестирование, 189

Б

базы данных
бумажный органайзер, 26
портативный компьютер, 26
Цикл, 72
Бесконечный Роковой Список, 75
беспорядок и процедуры, 54
Бланк-Эдельман, Дэвид, 11
большие проекты, 96
бумажный органайзер, 26
Цикл, 97

В

важные события, 104
вербовщики, непроизводительные
затраты времени, 161
взаимная защита от прерываний, 23
делегирование, 48
приоритеты, 130
видеомагазин, непроизводительные
затраты времени, 164
возврат на инвестиции (ROI), 132
временное замещение и отпуск, 143
время пика сосредоточенности, 40
время работы над проектом, 24
встречи, 104
опоздание, 106
пропущенные, 105
процедуры, 58
встречи (с.м. календарь, Цикл), 103

Д

дела, порядок следования в списке, 124,
126
делегирование, 48
наверх, 133
приоритеты, 130
деловой календарь, 107
доведение дела до конца, 71
документация
введение, 171

для внутреннего пользования, 172,
175, 178
контактная информация
производителей, 175
процедуры, 176
схемы сети, 177
условия сервисной поддержки,
175
для клиентов, 173, 175
приоритеты, 172, 178
технология Wiki, 178, 181
хранилище, 172
долгосрочные проекты, 89

З

задачи, повторяющиеся, 107, 110
заклинания, 28
запись
приоритеты, 130
в блокноте, 36
запросы
делегирование, 48
регистрация, 49
Цикл, 72, 74
засада в коридоре, 50

И

имена компьютеров
сокращения, 191
инкапсуляция и автоматизация, 201
исправляй, регистрируй или
сочувствуй, 61
источники информации
бумажный органайзер, 26
портативный компьютер, 26
Цикл, 72
итеративная разработка, 188

К

календарь, Цикл
встречи, 104
опоздание, 106
пропущенные, 105
единый календарь, 107
личные ритмы, 110, 111
обзор, 104
повторяющиеся задачи, 107, 110
ритмы фирмы, 111, 113
совещания, 104

пропущенные, 105
список дел на будущее, 104
кодирование и автоматизация, 187
командная строка, автоматизация, 205,
211
команды
сокращения, 189
инкапсуляция и автоматизация, 201
контактная информация производи-
телей, документирование, 175
корректировка, Цикл, 93
пример, 85, 89

Л

личная жизнь, 30
личные дела, 97
личный календарь, 107
Ллойд, Лес, 133
Лоура, Ральф, 126

М

массаж и стресс, 145
медитация и стресс, 145
многозадачность
сосредоточенность, 38
мозг и Цикл, 72, 74
монтаж аппаратуры, приглашение
специалиста, 167

Н

наладонник, 76
начальство
просьбы, приоритеты, 132
сообщение карьерных целей, 133
ненужные пункты в списке дел, 160
непроизводительные затраты времени
Usenet, 161
вербовщики, 161
видеомагазин, 164
коллеги, 161
монтаж аппаратуры, 167
общение с коллегами, 161
определение, 159
работа вручную, 161
рассылки, 160
совещания, 162
список дел, 160
способы борьбы, 159
стирка, 166

стратегические задачи, 168
тактические задачи, 168
телевидение, 165
типичные, 160
торговые агенты, 161
уборка в доме, 166
установка программ, 167
чаты, 161
электронные доски объявлений, 161
нотация и технология Wiki, 179

О

обещания (см. доведение дела до конца),
71
обработка прерываний, 45
обстановка, сосредоточенность, 37
общение с коллегами,
непроизводительные затраты
времени, 161
окна
многозадачность, 39
рациональная организация, 40
органайзеры, 26
процедуры, 56
Цикл, 76
отпуск и стресс, 142, 144

П

перегруженность и стресс, 139, 142
поведение в присутствии клиента, 47
политические проблемы, 134
портативный компьютер, 26
DateBook V, 98
Life Balance, 98
системы напоминания, 108
правило первого часа, 41
прерывания
взаимная защита от прерываний, 23
делегирование, 48
действие, 51
обзор, 43
обработка, 45
перенаправление, 43, 45
правило первого часа, 41
регистрация, 49
сосредоточенность, 34, 53
привычки, 28
принципы тайм-менеджмента
единая база данных, 26
заклинания, 28

- личная жизнь, 30
- одна база данных, 27
- привычки, 28
- процедуры, 28
- сосредоточенность, 30
- экономия ресурса мозга, 27
- приоритеты
 - взаимная защита от прерываний, 130
 - делегирование, 130
 - документация, 172, 178
 - ожидания клиентов, 130
 - проекты, 131
 - просьбы начальства, 132
 - регистрация, 130
 - список дел, 124, 130
 - учет ожиданий клиентов, 126
 - Цикл, 78
 - пример, 85, 89
- программы и автоматизация, 184
- проекты
 - большие, 96
 - приоритеты, 131
- процедура
 - автоматической проверки, 64
 - замены магнитных лент, 62
 - заправки автомобиля, 55
 - обхода клиентов, 60
 - обхода подчиненных, 58
 - редактирования файла, 65
 - сборов в поездку, 66
- процедуры, 28
 - (с.м. системы), 76
 - беспорядок, 54
 - введение, 54
 - документация для внутреннего пользования, 176
 - обновление, 69
 - примеры, 55, 66
 - советы по разработке, 66, 68
 - удаление старых процедур, 69
- Р**
 - работа вручную, непроизводительные затраты времени, 161
 - рабочее место, сосредоточенность, 37
 - расписание
 - встречи, 104
 - опоздание, 106
 - пропущенные, 105
 - процедуры, 58
 - дела, порядок следования в списке, 124, 126
 - долгосрочные проекты, 89
 - задачи повторяющиеся, 107, 110
 - переизбыток дел, 85
 - совещания, 104
 - непроизводительные, 162
 - пропущенные, 105
 - советы организаторам, 164
 - составление, 91
 - цели, 121
 - Цикл, 78
 - коррекция, 78
 - пример, 82, 83
 - регистрация
 - важная информация, 109
 - запросы клиентов, 49
 - засада в коридоре, 50
 - ресурс мозга, экономия, 27
 - Рейнголд, Том, 46
 - ритмы
 - личные, 110, 111
 - фирмы, 111, 113
- С**
 - связь с общественностью, процедуры, 63
 - Система Беспорядочных Записей, 74
 - система оценки качества обслуживания, 125
 - системное администрирование как стиль жизни, 12
 - системный администратор 1-го и 2-го уровня, 24
 - системы
 - (с.м. процедуры), 76
 - успешные, 76
 - системы напоминания, бумажный органайзер, 108
 - совещания, 104
 - непроизводительные, 162
 - пропущенные, 105
 - советы организаторам, 163, 164
 - эффективные, 164
 - сокращения
 - для имен компьютеров, 191
 - для команд, 189
 - сон, 36
 - стресс, 141
 - многозадачность, 39
 - сообщение на автоответчике, 36

- сосредоточенность, 30
 время пика, 40
 многозадачность, 38
 мозг, 35
 обстановка, 37
 правило первого часа, 41
 прерывания, 34, 53
- список дел
 Бесконечный Роковой Список, 75
 ненужные пункты, 160
 непроизводительные затраты времени, 160
 порядок следования, 124, 126
 приоритеты, 124, 130
 составление, 92
 управление, 81
 Цикл, 78
 личные дела, 97
 новые задания, 96
 планы на будущее, 104
 повторяющиеся задачи, 107, 110
 пример, 83, 85
- ссылки на страницы и технология Wiki, 179
- стиль жизни системного администратора, 80, 102
- стратегические задачи, непроизводительные затраты времени, 168
- стратегия в парке аттракционов, 42
- стресс
 источники стресса, 138
 конфликтующие указания, 139, 142
 отпуск, 142, 144
 перегруженность, 139, 142
 приемы снятия, 144, 146
 сон, 141
- схемы сети, 177
- сценарии и автоматизация, 184
- Т**
- тактические задачи, непроизводительные затраты времени, 168
- тайм-менеджмент
 принципы для системных администраторов, 26, 31
 трудности, 25
- телевидение, непроизводительные затраты времени, 165
- тестирование и автоматизация, 189
- типичные непроизводительные затраты времени, 160
- торговые агенты, непроизводительные затраты времени, 161
- трудности тайм-менеджмента, 25
- У**
- уборка в доме, непроизводительные затраты времени, 166
- управление шефом, 132, 137
- условия сервисной поддержки, документация для внутреннего пользования, 175
- установка программ, непроизводительные затраты времени, 167
- Ф**
- фильтры, обработка электронной почты, 149
 серверная, 151
- Х**
- Хоган, Кристина, 9
- хранилища
 документация для внутреннего пользования, 172
 клиентов, 172, 173, 175
- Ц**
- цели
 карьерные, 133
- Цикл
 введение, 115
 запись, 117
 планирование последующих шагов, 120
 постановка, 118
 просмотр списка, 122
 расписание, 121
- Цикл
 большие проекты, 96
 бумажный органайзер, 97
 введение, 71
 выполнение плана, пример, 89, 90
 досрочное выполнение, 96
 завершение рабочего дня, 90

Цикл

- запросы, 72, 74
- календарь, 103, 107
 - личные ритмы, 110, 111
 - обзор, 104
 - повторяющиеся задачи, 107, 110
 - ритмы фирмы, 111, 113
- корректировка, 93
 - пример, 85, 89
- мозг, 72, 74
- обзор, 78
- окончание рабочего дня, пример, 90
- органайзер, 76
- повторение, пример, 91
- приоритеты, 78, 93
 - пример, 85, 89
- расписание, 78
 - коррекция, 78
 - пример, 82, 83
 - составление, 91
- список дел, 78
 - личные дела, 97
 - новые задания, 96
 - планы на будущее, 104
 - повторяющиеся задачи, 107, 110
 - пример, 83
 - составление, 92
- цели
 - введение, 115
 - запись, 117
 - планирование последующих шагов, 120
 - постановка, 118
 - просмотр списка, 122
 - расписание, 121
- цифровые видеомэгафоны, 165

Ч

- чаты, непроизводительные затраты времени, 161

Э

- экономия ресурса мозга, 27
- электронная почта
 - обработка, 148, 149
 - выполнение запроса и удаление сообщения, 154
 - ответ и удаление, 153
 - папка Перья, 153
 - сохранение сообщений, 152
 - удаление сообщений, 150
 - фильтры, 149
 - серверные, 151
 - чтение, 150
- рассылки, непроизводительные затраты времени, 160
- электронные доски объявлений, непроизводительные затраты времени, 161

www.natahaus.ru

Электронная версия данной книги создана исключительно для ознакомления! Реализация данной электронной книги в любых интернет-магазинах, и на CD (DVD) дисках с целью получения прибыли, незаконна и запрещена! По вопросам приобретения печатной или электронной версии данной книги обращайтесь непосредственно к законным издателям, их представителям, либо в соответствующие организации торговли!

www.natahaus.ru

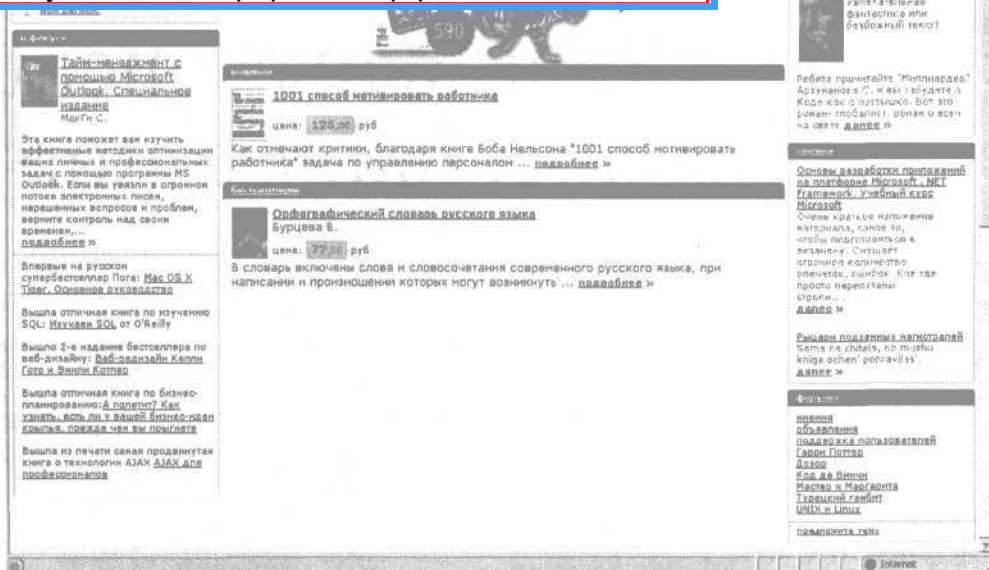
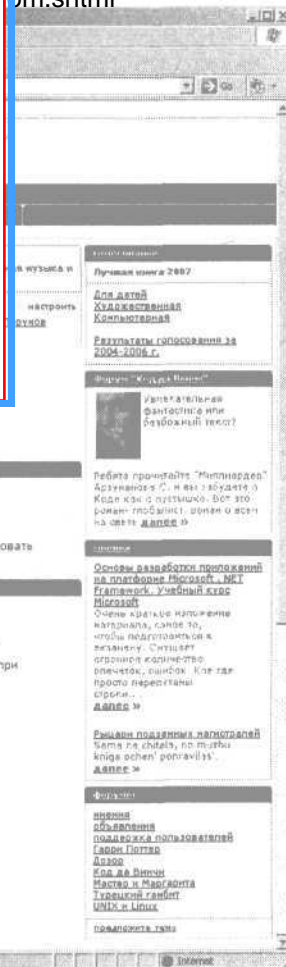
НАПИСАНИЕ на ЗАКАЗ:

1. Дипломы, курсовые, чертежи...
 2. Диссертации и научные работы.
 3. Школьные задания.
- Онлайн-консультации.

ЛЮБАЯ тематика,
в том числе ТЕХНИКА.

Приглашаем авторов.

УЧЕБНИКИ, ДИПЛОМЫ, ДИССЕРТАЦИИ:
полные тексты в электронной библиотеке
www.учебники.информ2000.рф.



- старейший интернет-магазин России
- 11 лет на рынке
- скидки постоянным покупателям
- безопасность покупок

Интернет-магазин Books.Ru основан в 1996 году. Мы торгуем книгами по всему миру, доставляя их в самые удаленные города и страны.

Самые новые и интересные книги, диски и видеокассеты вы всегда сможете найти у нас первыми. Интересный ассор-

тимент, оперативная доставка и удобные способы оплаты делают покупки в нашем магазине приятным времяпрепровождением.

Постоянство и надежность нашей работы гарантируют безопасность покупок. Ждем вас в нашем магазине!

Books.Ru – ваш ближайший книжный магазин

тел/факс Москва (495) 945-8100
Санкт-Петербург (812) 324-5353



Фредерик БРУКС

Мифический человеко-месяц Как создаются программные системы

304 стр., книга в продаже

Не о многих книгах по управлению созданием программных систем можно сказать, что они оказывают на читателей такое же влияние и так же не устаревают, как «Мифический человеко-месяц».

Это своего рода библия для разработчиков программного обеспечения во всем мире, вышедшая еще в 1975 году. Тогда же был издан и ее перевод на русский язык, и уже давно эта книга стала библиографической редкостью. В США полагают, что без ее прочтения не может состояться ни один крупный руководитель программного проекта.

Каждому, кто руководит сложными проектами, Брукс предлагает оригинальную смесь реальных примеров проектирования и нестандартных идей. Очерки, составляющие книгу, взяты из его собственного опыта работы руководителем проекта по созданию семейства компьютеров IBM System/360 и операционной системы OS/360.

Спустя 20 лет после первой публикации книги Брукс произвел ревизию всех своих первоначальных идей и добавил новые – как для читателей, уже знакомых с его работой, так и для тех, кто открывает ее для себя впервые.



Том ДЕМАРКО и Тимоти ЛИСТЕР

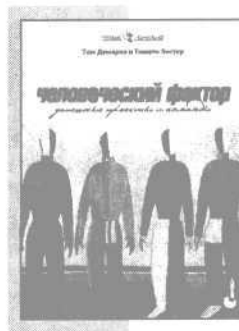
Человеческий фактор: успешные проекты и команды, 2-е издание

256 стр., книга в продаже

Первое издание, ставшее всемирным бестселлером об управлении проектами разработки ПО, содержало революционные по тем временам (1987 г.) идеи, которые выдержали проверку временем. Во втором издании авторы скорректировали свои выводы и добавили несколько новых глав.

Ценность этой книги в том, что в ней описываются принципы, за каждым из которых стоит реальная история. Все главы содержат наблюдения и новаторские подходы, которые заставят читателей и руководителей увидеть важные вопросы в новом, более разумном ракурсе.

С юмором и мудростью, обретенными за годы руководства и консультирования, Демарко и Листер демонстрируют, что самые сложные проблемы разработки ПО имеют человеческую, а не техническую природу. Они не дают простых ответов, но дают правильные, подтвержденные научными исследованиями. Издание предназначено в первую очередь руководителям проектов, но будет полезно и рядовым программистам.





Санжей МИШРА, Алан БЬЮЛИ

Секреты Oracle SQL

368 стр., книга в продаже

Большинство книг по SQL не выходит за рамки обсуждения синтаксиса и азов применения. Данное издание – одно из исключений, где авторы, используя Oracle8i и Oracle9i, на большом количестве примеров показывают, как творчески применять мощные и гибкие средства SQL для быстрого создания эффективных и удобных для сопровождения запросов в среде Oracle. Если вы Java-программист, администратор базы данных или программист на PL/SQL, эта книга для вас. Вы сможете повысить производительность своего труда и приобрести уверенность в правильности написания ваших SQL-запросов.

В книге описываются наиболее важные и полезные свойства Oracle SQL и пути их применения для решения конкретных задач. Вы найдете множество оригинальных приемов, которые можно использовать для улучшения ваших собственных приложений. Вы научитесь: применять ANSI-совместимый синтаксис объединения; работать с новыми типами для времени и даты; максимально использовать такие конструкции SQL, как подзапросы, слияния, группы и объединения; обрабатывать иерархические данные; применять инструкции DECODE и CASE для реализации условной логики в SQL-запросах; строить запросы, работающие с разделами, объектами и коллекциями.



Рик ГРИНВАЛЬД, Дэвид КРЕЙНС

Oracle. Справочник

976 стр., книга в продаже

Oracle, ведущая СУБД масштаба предприятия, – это сложная система, предлагающая несметное множество продуктов, языков и инструментов. Следующие один за другим обновления, релизы и выпуски делают все более сложной задачу пользователя, стремящегося справиться с могучим потоком часто меняющейся информации об этой СУБД и ее возможностях. Задача справочника – объединить наиболее важную информацию по архитектуре, синтаксису и пользовательским интерфейсам Oracle. Синтез формы и содержания этой книги от O'Reilly дает лаконичный и очень доступный настольный справочник, содержащий важнейшие команды, конструкции языка, параметры и форматы файлов Oracle.

«Oracle. Справочник» – это кладезь информации, необходимой администраторам БД, разработчикам на PL/SQL и Java, системным и сетевым администраторам, а также специалистам по безопасности, имеющим дело с БД Oracle. Эта книга не раз поможет программистам, использующим как Oracle9i версии 2, так и более ранние продукты, при написании кода, работающего с данными Oracle.



Кэри МИЛЛСАП, Джефф ХОЛЬТ

Oracle. Оптимизация производительности

464 стр., книга в продаже

Оптимизация производительности БД Oracle считается очень сложной задачей, подвластной лишь черной магии. Успехи настройки нередко случайны и достигаются скорее за счет интуиции, комбинируемой с методом проб и ошибок. Известные исследователи Oracle, Миллсап и Хольт, в практическом руководстве «Oracle. Оптимизация производительности» подробно описывают надежный, воспроизводимый и четкий метод выявления проблем производительности системы, позволяющий с уверенностью сказать, в чем причина любой из них.

Ключом к методу Миллсапа и Хольта является тот факт, что программное обеспечение БД Oracle оснащено инструментами, способными предоставить информацию о том, на что тратится время при обработке запросов. Метод включает три этапа: выбор пользовательской операции, оптимизация которой наиболее важна с точки зрения бизнеса; сбор корректно выбранных данных расширенной трассировки SQL, относящихся к данной операции, и выявление по этим данным места и причин перерасхода времени; поиск наиболее эффективного способа повышения производительности (уменьшения времени отклика) данной операции. Издание предназначено администраторам и разработчикам БД Oracle.



Поль ДЮБУА

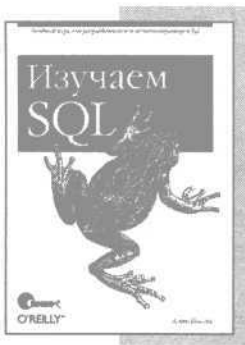
MySQL. Сборник рецептов

1056 стр., книга в продаже

«MySQL. Сборник рецептов» – это всеобъемлющий сборник задач, ежедневно возникающих у программистов, их решений и практических примеров. Сборник будет полезен всем пользователям MySQL независимо от уровня их подготовки. Каждой задаче, обсуждаемой в книге, соответствует проработанное решение или рецепт с небольшим фрагментом кода, который можно вставлять прямо в приложение. Работа каждого фрагмента подробно поясняется, что позволяет разобраться, как и почему все это работает, и применить готовые приемы к схожим ситуациям. Материал книги пригодится и опытным разработчикам MySQL – им не придется писать весь код с нуля. На веб-сайте книги находятся все сценарии, написанные для API таких языков, как Perl, Python, Java и PHP.

В книге обсуждаются: использование сценариев в сочетании с MySQL для чтения запросов из файла; формирование запросов, решающих часто встречающиеся задачи; создание сценариев MySQL для Web; взаимодействие с сервером; изменение структуры таблиц за счет добавления, удаления или изменения столбцов; импорт и экспорт данных; выявление, подсчет и удаление дубликатов, а также предотвращение их появления; вычисление различных статистических характеристик.





Алан БЬЮЛИ

Изучаем SQL

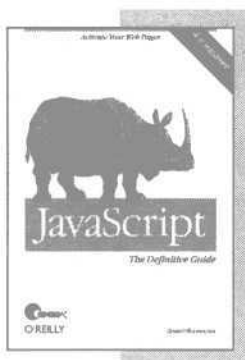
312 стр., книга в продаже

Книга Алана Бьюли, эксперта по языку SQL, – прекрасный учебник для тех, кто еще не знает, но хочет освоить этот язык. Книга не только позволит приобрести начальные знания, но и расскажет о наиболее популярных мощных средствах языка SQL, используемых опытными программистами.

Многие книги по SQL грешат скучным изложением основ. Здесь же автор в стиле живого рассказа обсуждает SQL-выражения и блоки, различные типы условий, показывает, как посредством соединения таблиц создавать запросы к нескольким таблицам, рассматривает наборы данных и как они могут взаимодействовать в запросах, демонстрирует встроены и агрегатные функции, показывает, как и где используются подзапросы. Подробно описаны различные типы соединений таблиц, применение условной логики, работа с транзакциями, индексы и ограничения.

Поскольку лучший способ изучения SQL – это практика, автор создает учебную БД MySQL и приводит множество вариантов реальных запросов, охватывающих весь теоретический материал. Примеры кода можно использовать в своих программах и документации.

Книга предназначена разработчикам приложений БД, администраторам БД и тем, кто создает отчеты.



Дэвид ФЛЭНАГАН

JavaScript. Подробное руководство, 5-е издание

1040 стр., IV кв. 2007 г.

Пятое издание книги полностью обновлено. Рассматриваются взаимодействие с протоколом HTTP и применение технологии Ajax, обработка XML-документов, создание графики на стороне клиента с помощью тега <canvas>, пространства имен в JavaScript, необходимые для разработки сложных программ, классы, замыкания, Flash и встраивание сценариев JavaScript в Java-приложения.

Часть I знакомит с основами JavaScript. В части II описывается среда разработки сценариев, предоставляемая веб-браузерами. Многочисленные примеры демонстрируют как генерировать оглавление HTML-документа, отображать анимированные изображения DHTML, автоматизировать проверку правильности заполнения форм, создавать всплывающие подсказки с использованием Ajax, как применять XPath и XSLT для обработки XML-документов, загруженных с помощью Ajax. Часть III – обширный справочник по основам JavaScript (классы, объекты, конструкторы, методы, функции, свойства и константы, определенные в JavaScript 1.5 и ECMAScript 3). Часть IV – справочник по клиентскому JavaScript. Здесь описываются API веб-браузеров, стандарт DOM API Level 2 и недавно появившиеся стандарты: объект XMLHttpRequest и тег <canvas>.



Дэвид СКЛЯР, Адам ТРАХТЕНБЕРГ

RНР. Сборник рецептов

672 стр., книга в продаже

Широкий набор возможностей RНР, удобный синтаксис и поддержка различных ОС и веб-серверов делают его идеальным языком для быстрой веб-разработки. Книга содержит уникальную обширную коллекцию наилучших приемов решения повседневных задач программирования на RНР. Для каждой задачи дается решение, или «рецепт», – короткий, концентрированный фрагмент кода, который вы сможете вставить в свое приложение. При этом авторы объясняют, как и почему работает программа, что позволяет научиться адаптировать приемы решения к схожим задачам.

Представлено более 250 рецептов – от самых простых, таких как посылка запроса в базу данных и получение доступа к URL, до полноценных программ, демонстрирующих более трудные задачи, например вывод HTML-таблиц и создание диаграмм. Рассмотрена работа со строками, числами, датами и временем, а также с массивами, файлами и каталогами. Обсуждаются переменные, функции, классы и объекты, регулярные выражения, шифрование и безопасность, интернет-службы, графика, интернационализация и локализация, PEAR, RНР в командной строке и RНР-GTK, формы, XML и доступ к базам данных.

Книга будет полезна всем, кто программирует на RНР, независимо от уровня их подготовки.



Ларри УОЛЛ, Том КРИСТИАНСЕН, Джон ОРВАНТ

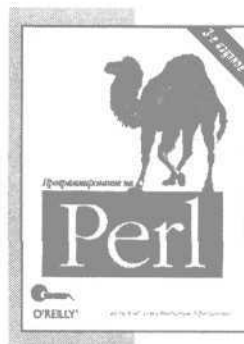
Программирование на Perl, 3-е издание

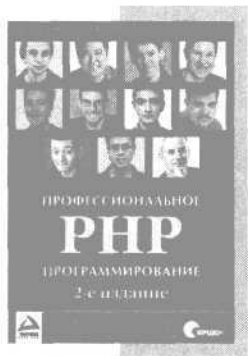
1152 стр., книга в продаже

Ларри Уолл – создатель Perl и автор этой книги, написанной очень хорошо и интересно. Perl – мощный язык программирования, который с момента его появления в 1988 году стал весьма популярен и вырос, и это обстоятельство потребовало нового – уже третьего – издания книги. Теперь в нем более тысячи страниц, содержащих как введение в язык Perl для новичков в программировании, так и отличный справочник по языку.

Помимо прочего Ларри еще и лингвист. Возможно поэтому Perl стал необычайно гибким языком, где одного и того же можно достичь многими путями, как это прекрасно демонстрирует автор. Предыдущие издания книги широко известны в компьютерных кругах и стали библией программистов на Perl.

Что нового в этом издании? Практически все. Третье издание не просто расширено в соответствии с релизом Perl 5.6, но и полностью реорганизовано и усилено множеством примеров. Большая часть разделов существенно переработана, например разделы, посвященные объектно-ориентированному программированию и регулярным выражениям. Кроме того, добавлено множество новых глав, рассматривающих работу с профилями, Unicode, организацию поточной обработки, компилирование и внутреннюю организацию Perl.





Луис АРГЕРИХ и др.

Профессиональное PHP программирование, 2-е издание

1048 стр., книга в продаже

PHP – язык сценариев для создания динамических веб-страниц, принадлежащий к категории продуктов open source. Предоставляет простое и универсальное, переносимое между платформами решение для электронной коммерции, сложных веб-приложений, включая управляемые базами данных.

Подробно рассмотрено создание самых современных веб-приложений, хорошо масштабируемых, оптимальным образом использующих базы данных и соединяющихся с внутренней сетью на основе многозвенной архитектуры. Приводятся примеры кода, в том числе клиенты FTP и электронной почты, сложные структуры данных, управление сеансами и создание безопасных программ. Обсуждаются поддержка LDAP, связь PHP с XML, базами данных на примере MySQL и PostgreSQL, безопасность, оптимизация и интернационализация приложений, библиотеки расширений PHP.

Эта книга адресована PHP-программистам, желающим повысить свое мастерство. Предполагается, что читатель знаком с программированием и базами данных, но книга будет полезна каждому, кто способен с помощью PHP писать и сопровождать небольшие веб-приложения, интересуется программированием в сетях.



Николас ЗАКАС, Джереми МАК-ПИК, Джо ФОСЕТ

AJAX для профессионалов

488 стр., книга в продаже

AJAX объединяет такие проверенные временем технологии, как CSS, XML и JavaScript, и предоставляет разработчикам возможность создавать более сложные и более динамичные пользовательские интерфейсы, а также отменяет господство стандарта «щелкни и жди», начавшееся еще со времен появления Всемирной паутины.

В книге рассматриваются различные способы выполнения запросов и поясняется, когда и при каких обстоятельствах должна использоваться та или иная методика. Из книги вы узнаете о различных методиках и шаблонах проектирования AJAX, применяемых для организации взаимодействия между клиентом и сервером в ваших веб-приложениях, на вашем сайте. Материал последующих глав основывается на предыдущих, поэтому к концу книги вы получите практические знания, необходимые для реализации ваших собственных решений на основе AJAX.

Книга предназначена разработчикам, желающим повысить эргономические качества своих сайтов и веб-приложений. Знакомство с JavaScript, HTML и CSS обязательно, как и некоторый опыт работы с технологиями разработки серверных сценариев, таких как PHP и .NET.



Энди ГУТМАНС, Стиг БАККЕН, Дерик ПЕТАНС

PHP5. Профессиональное программирование

704 стр., книга в продаже

Каким бы сложным ни был ваш программный проект, вы сможете сделать его более мощным и совершенным, прочитав подробное руководство, написанное ведущими разработчиками PHP 5.

Авторы делятся своими уникальными знаниями и предлагают множество реалистичных примеров. Вы познакомитесь с особенностями новой объектной модели PHP 5, важнейшими шаблонами проектирования, улучшенной поддержкой XML и веб-сервисов. Только здесь вы найдете высокоэффективные решения для разработки собственных веб-приложений, расширений, пакетов и сценариев командной строки. Рассматриваются: синтаксис и расширенные объектно-ориентированные возможности PHP – от свойств и методов до полиморфизма, интерфейсов и рефлексии; обработка ошибок; написание мощных веб-приложений (обработка ввода пользователя, работа с cookies, сессии); библиотека PEAR; интеграция с MySQL, SQLite и другими СУБД.

Те, кто перешел на PHP недавно, смогут быстро адаптироваться к этому языку, а опытные PHP-программисты, знакомые с его простотой и удобством в работе, откроют для себя всю его необычайную мощь.



Кристиан ДАРИ, Богдан БРИНЗАРЕ, Филип ЧЕРЧЕЗ-ТОЗА,
Михай БУСИКА

AJAX и PHP

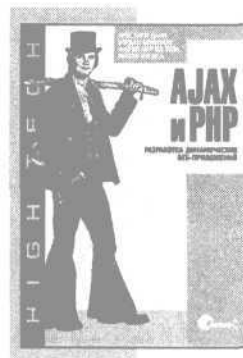
Разработка динамических веб-приложений

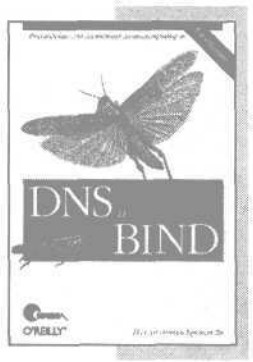
336 стр., книга в продаже

Эта книга – самый удобный и полезный ресурс, который поможет вам войти в захватывающий мир AJAX. Вы научитесь писать более эффективные веб-приложения на PHP благодаря использованию всего спектра возможностей технологий AJAX. Применение AJAX в связке с PHP и MySQL описывается на многочисленных примерах, иллюстрирующих распространенные типовые задачи и лучшие приемы их решения, которые читатель сможет использовать в собственных проектах.

Рассмотрены следующие темы: верификация заполнения форм на стороне сервера; чат-приложение, основанное на технологии AJAX; реализация подсказок и функции автодополнения; построение диаграмм в реальном времени средствами SVG; настраиваемые и редактируемые таблицы на основе баз данных; реализация RSS-агрегатора; построение сортируемых списков с поддержкой механизма drag-and-drop.

Издание предназначено тем, кто владеет базовыми знаниями PHP, XML, JavaScript и MySQL и хочет узнать все тонкости функционирования AJAX и взаимодействия составляющих этой технологии.





Пол АЛЬБИТЦ, Крикет ЛИ
DNS и BIND, 4-е издание

696 стр., книга в продаже

Данное издание включает информацию о версии 9 пакета BIND, в которой реализованы новые и очень важные механизмы, а также о версии 8, входящей в состав большинства действующих коммерческих разработок. Пакеты BIND 8 и 9 значительно повышают безопасность служб DNS.

Эта книга будет полезна как администраторам, ежедневно работающим с DNS, так и пользователям, желающим более подробно изучить принципы работы Интернета. В ней рассмотрены: функциональность, принципы работы, причины использования DNS; поиск собственного места в пространстве имен Интернета; установка и настройка серверов имен; применение MX-записей для маршрутизации почты; настройка узлов на работу с DNS; разделение доменов на поддомены; обеспечение безопасности DNS; новые возможности BIND 9; расширения системы безопасности DNS (DNSSEC) и подписи транзакций (TSIG); распределение нагрузки между серверами имен; динамические обновления, асинхронные уведомления об изменениях зон, пошаговая передача зон; устранение неполадок (применение *nslookup* и *dig*, чтение отладочного вывода); DNS-программирование с применением библиотеки поискового анализатора и Perl-модуля Net::DNS.



Крэйг ХАНТ
**TCP/IP. Сетевое администрирование,
3-е издание**

816 стр., книга в продаже

Третье издание «TCP/IP. Сетевое администрирование» – это полноценное руководство по настройке и сопровождению сети TCP/IP, которое предназначается как системным администраторам, так и пользователям домашних компьютеров с доступом к сети Интернет. Повествование начинается с основ: зачем нужны протоколы, как они работают, как адресация и маршрутизация позволяют передавать данные по сети и как настроить сетевое соединение.

Помимо базовой настройки книга рассказывает о современных протоколах маршрутизации (RIPv2, OSPF и BGP) и пакете *gated*, который реализует работу с ними. Кроме того, книга является руководством по настройке многих важных сетевых служб, в том числе DNS, Apache, sendmail, Samba, PPP и DHCP. Две главы посвящены безопасности и разрешению проблем. Справочные приложения подробно описывают синтаксис таких программ, как *gated*, *pppd*, *named*, *dhcpcd* и *sendmail*.

Издание включает новую главу, посвященную важной задаче настройки веб-сервера Apache, и раздел о настройке Samba для организации совместного доступа к файлам и принтерам в гетерогенной сети UNIX/Windows. Книга охватывает реализации TCP/IP для систем Linux, Solaris, BSD и System V.



Майкл ЛУКАС

FreeBSD. Подробное руководство

616 стр., книга в продаже

FreeBSD – мощная, гибкая и бесплатная операционная система семейства UNIX – является предпочтительной серверной платформой для многих предприятий. Однако настраивать ее сложнее, чем UNIX или Linux, а эффективно применять – труднее.

Исчерпывающее руководство по FreeBSD написано одним из активных участников проекта FreeBSD. В нем охвачены установка системы, работа в сети, вопросы безопасности, сетевые сервисы, производительность системы, тонкая настройка ядра, файловые системы, SMP, проведение обновлений, устранение неполадок и многое другое.

Вы узнаете, как проводить резервное копирование данных в масштабе всей системы и на уровне отдельных файлов, как конфигурировать ядро FreeBSD, как обеспечить безопасность системы, как настраивать системное ПО, управлять сетевыми сервисами и файловыми системами, применять DNS и настраивать почтовые, веб- и FTP-сервисы на стороне сервера и клиента, как восстанавливать работоспособность системы в редких аварийных ситуациях, проводить отладку и создавать отчеты о неисправностях, как контролировать работу системы с помощью инструментов измерения производительности.



Маттиас Калле ДАЛХАЙМЕР, Мэтт УЭЛШ

Запускаем Linux, 5-е издание

1040 стр., III кв. 2007 г.

Классический труд, вышедший уже 5-м изданием, значительно расширен с целью отразить зрелость операционной системы и изобилие существующего ПО. Такие горячие темы, как воспроизведение звука и видео, ПО для рабочих групп и фильтрация спама, соседствуют рядом с основами настройки и администрирования, которые всегда делали книгу популярной. Издание охватывает не только основные способы обмена информацией (электронная почта, навигация во Сети и обмен мгновенными сообщениями), но и описывает тонкости настройки сети, включая соединение по коммутируемым линиям, ADSL и кабельные модемы.

Перечень новых тем включает шифрование электронной почты и шифрующие файловые системы, передовые методы работы с командными оболочками и приложения удаленной регистрации. Классические темы, касающиеся загрузки, управления пакетами, пересборки ядра и настройки X Window, также обновлены.

Предвидя возможные трудности, авторы предлагают надежные решения и дают четкие и ясные инструкции, которые обеспечат вам успешную работу в Linux. Материал излагается просто и ясно, но представлен достаточно полно, чтобы служить руководством для начинающих и в то же время обеспечить новой информацией опытных пользователей, стремящихся побольше узнать о Linux.





Брайан КЕРНИГАН и Роб ПАЙК

UNIX. Программное окружение

416 стр., книга в продаже

Авторы рассказывают не только, как работает UNIX, но и почему он работает именно так. Они пишут в предисловии, что стремились показать философию программирования под UNIX, базирующуюся на отношениях между программами, и потому большую часть книги посвятили обсуждению индивидуальных инструментов в контексте их совместного использования, в том числе и для построения новых программ. Чтобы правильно работать с UNIX, необходимо понять, не только как применять его программы, но и как они взаимодействуют с окружением.

Вход в систему, работа с файлами и каталогами, основы окружения (переменные, маски), фильтры, циклы, сигналы, аргументы, стандартный ввод-вывод, введение в системные вызовы и в программирование с использованием lex, yacc и make, подготовка документации с помощью troff, tbl и eqn – все это рассмотрено в данной книге, адресованной тем, кто хочет научиться программировать с помощью всех инструментов, поставляемых с UNIX.

Может показаться, что перечисленного слишком много для одной книги, но это действительно очень хорошее введение в программное окружение UNIX, не утратившее своей актуальности со времен своего первого издания в 1984 году.



Филипп ТОРЧИНСКИЙ

UNIX. Практическое пособие администратора, 2-е издание

400 стр., книга в продаже

Книга рассчитана на тех, кто планирует освоить системное администрирование какого-либо UNIX, например FreeBSD, Linux, Solaris или любого другого. Она также будет полезна тем, кто уже работает системным администратором UNIX – они смогут применять ее как справочник.

Автор рассказывает обо всех системах UNIX, приводятся примеры из FreeBSD, Linux и Solaris. Представлена самая свежая информация по UNIX, описываются важные, но пока недостаточно обсуждаемые технологии, например PAM. Основное внимание уделяется тем задачам, которые принято решать под UNIX в России.

Предполагается, что читатель владеет компьютером и представляет себе задачи системного администрирования. Для чтения не требуется опыт работы с UNIX – в книге описаны как основы взаимодействия с системой, так и детали системного администрирования. Легче всего будет читать книгу тем, кто уже администрировал какие-нибудь системы, например Windows NT, Novell Netware, VAX VMS.



Жан-Поль Д. МУСУМЕСИ, Майкл ЛУКИДЕС

Настройка производительности UNIX-систем, 2-е издание

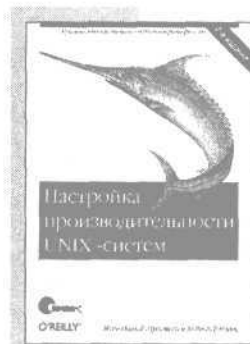
408 стр., книга в продаже

В книге дается ответ на два важных вопроса: как добиться максимального эффекта без покупки дополнительного оборудования и когда новые средства (память, быстрые диски, процессоры и сетевые интерфейсы) все же стоит приобрести? Чтобы повысить быстродействие, нужно хорошо представлять работу компьютеров и сетей и понимать, как программы соперничают за системные ресурсы.

В полностью обновленном 2-м издании внимание сосредоточено на системах Solaris и Linux, хотя обсуждаемые принципы применимы к любым системам.

В книге рассмотрены следующие темы:

- Определение параметров и управление рабочим процессом
- Методы измерения производительности
- Архитектура и производительность процессора
- Оптимизация использования памяти
- Оптимизация работы дисков и дисковых массивов
- Обнаружение перегруженных или неработоспособных участков сети
- Оптимизация программного кода



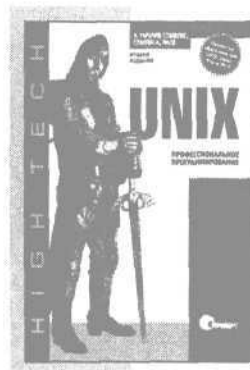
Ричард СТИВЕНС, Стивен РАГО

UNIX. Профессиональное программирование, 2-е издание

1040 стр., книга в продаже

Издание представляет собой подробнейшее справочное руководство для любого профессионального программиста, работающего с UNIX. Стивену Раго удалось обновить и дополнить текст фундаментального классического труда Стивенса, сохранив при этом стиль оригинала. Содержание всех тем, примеров и прикладных программ обновлено в соответствии с последними версиями наиболее популярных реализаций UNIX.

Среди важных дополнений главы, посвященные потокам и разработке многопоточных программ, использованию интерфейса сокетов для организации межпроцессного взаимодействия (IPC), а также широкий охват интерфейсов, добавленных в последней версии POSIX.1. Аспекты прикладного программного интерфейса разъясняются на простых и понятных примерах, протестированных на 4-х платформах: FreeBSD 5.2.1, Linux 2.4.22, Solaris 9 и Mac OS X 10.3. Описывается множество ловушек, о которых следует помнить при написании программ для различных реализаций UNIX, и показывается, как их избежать, опираясь на стандарты POSIX.1 и Single UNIX Specification.



Основано в 1995 году

О нас

Наша специализация – книги компьютерной тематики. Наши издания – плод сотрудничества известных зарубежных и отечественных авторов, высококлассных переводчиков и компетентных научных редакторов. Среди наших деловых партнеров издательства: O'Reilly, Pearson Education, NewRiders, Addison Wesley и другие.

O'REILLY*



New
Riders



Где купить

Наши книги вы можете купить во всех крупных книжных магазинах России, Украины, Белоруссии и других стран СНГ. Однако по минимальным ценам и оптом они продаются:

Санкт-Петербург:

главный офис издательства –

В.О. 16 линия, д. 7 (м. Василеостровская),
тел. (812) 324-5353

Москва:

московский филиал издательства –

ул. Беговая, д. 13 (м. Динамо),
тел. (495) 945-8100

Заказ книг через Интернет

в розницу: <http://www.symbol.ru>

<http://www.books.ru>

оптом: <http://opt.books.ru>

по обычной почте

199034, С. Петербург, 16 линия, д. 7.

Издательство «Символ-Плюс».

Бесплатный каталог изданий высылается по запросу.

Приглашаем к сотрудничеству



Мы приглашаем к сотрудничеству умных и талантливых авторов, переводчиков и редакторов. За более подробной информацией обращайтесь, пожалуйста, на сайт издательства: www.symbol.ru.

Также на нашем сайте вы можете высказать свое мнение и замечания о наших книгах. Ждем ваших писем!

Тайм-менеджмент для системных администраторов



Время – огромная ценность, особенно если вы работаете системным администратором. Никакая другая работа не заставляет в такой степени разрываться на части. Пользователи то и дело отвлекают вас просьбами, мешая работать над долгосрочными проектами, которые вам поручил начальник, а ваши «подопечные» компьютеры создают проблемы в самый неподходящий момент. И даже овладев всеми тонкостями профессии, вы задерживаетесь по вечерам и работаете по выходным только для того, чтобы соответствовать предъявляемым требованиям. Давление на вас велико, и оно постоянно растет.

Основное внимание в книге «Тайм-менеджмент для системных администраторов» уделяется приемам, которые помогут вам не только вести повседневные дела, но и справляться с неизбежными критическими ситуациями. Книга насыщена примерами, взятыми автором из его долгой карьеры, на протяжении которой он занимался обслуживанием рабочих станций и серверов, а также разработкой программного обеспечения и систем безопасности. Это значит, что вы получите практические советы бывалого человека, а не академические банальности, бесполезные для системного администратора в его борьбе с хаосом.

Вы научитесь:

- Управлять отвлекающими факторами
- Исключать непроизводительные затраты времени
- Вести эффективный календарь
- Разрабатывать процедуры для регулярно совершаемых действий
- Сосредотачиваться на текущей задаче
- Расставлять приоритеты в соответствии с ожиданиями пользователей
- Документировать и автоматизировать рабочие процессы для ускорения их выполнения

Томас А. Лимончелли обладает почти 20-летним опытом системного администрирования и с 2003 года участвует в работе конференций по тайм-менеджменту. Он работал как в небольших, так и в крупных фирмах, включая Bell Lab и AT&T, выступает с докладами на конференциях и с лекциями по всему миру. Его книга «The Practice of System and Network Administration» (Практика системного и сетевого администрирования) считается лучшим справочным пособием по системному администрированию.

КАТЕГОРИЯ: СИСТЕМНОЕ АДМИНИСТРИРОВАНИЕ

УРОВЕНЬ ПОДГОТОВКИ ЧИТАТЕЛЕЙ: СРЕДНИЙ

Издательство «Символ-Плюс»
(812) 324-5353, (495) 945-8100

ISBN-13: 978-5-93286-090-8
ISBN-10: 5-93286-090-1



9 785932 860908



<http://учебники.информ2000.ru/учебники.shtml>


www.symbol.ru