

## Глава 1

# Восстановление поврежденных файлов

Просмотр файлов в программе Hexplorer

Понятие формата файла

Восстановление файлов собственными средствами  
прикладных программ

Программы для восстановления файлов

Полезные ссылки

В этой главе речь пойдет о восстановлении так называемых «битых» файлов. Файл может быть поврежден при его некорректном сохранении, особенно если вы пытались открыть, а затем сохранить файл, созданный в более новой версии прикладной программы. Такие файлы можно извлечь с поврежденного носителя. И хотя их удастся скопировать, при попытке открыть «спасенный» файл вы увидите сообщение об ошибке. После восстановления данных со случайно отформатированного диска среди найденных файлов наверняка окажется немало «битых». Наконец, подобные файлы найдутся в результате анализа данных, сохраненных операционной системой при проверке и автоматическом исправлении ошибок жесткого диска, — такие файлы имеют расширение CHK.

Считается, что файл — именованная конечная и непрерывная последовательность байтов, несущая некую информацию. Для наших целей (восстановления данных) этого определения достаточно. В тех файлах, которые приходится восстанавливать на практике, такая последовательность обычно кодирует вполне осмысленную и завершенную информацию, например текст документа или растр (множество точек) изображения.

Напомню, что 1 байт = 8 бит, то есть каждый байт может принимать значения от 0 до  $2^8 = 256$  в десятичной (dec) или  $2^8 = FF$  в шестнадцатеричной (hex) форме записи. Существуют программы, которые позволяют просматривать и изменять содержимое любого файла как простой последовательности байтов. Логично называть их байтовыми, или двоичными (бинарными), редакторами. Поскольку такие приложения обычно показывают значения байтов в шестнадцатеричном виде, за ними закрепилось название «HEX-редакторы».

Если открыть файл в любом HEX-редакторе, мы увидим шестнадцатеричное представление содержимого этого файла. О применении HEX-редакторов для восстановления содержимого файла будет сказано в конце главы, пока же просмотр в редакторе поможет понять структуру файла и то, в чем заключается ее повреждение.

## Просмотр файлов в программе Hexplorer

Программа ICY Hexplorer — простой, но достаточно функциональный бесплатный редактор двоичных данных. Ее дистрибутив ее доступен на сайте разработчиков <http://hexplorer.sourceforge.net>.

Рабочая область окна разделена по вертикали на две части фиксированной ширины (в зависимости от размера окна правее рабочей области может отображаться пустое поле). После открытия файла в левой части рабочей области отображается содержимое файла по байтам в виде их шестнадцатеричных значений (рис. 1.1). В правой части окна те же самые байты интерпретируются как ASCII-значения текстовых символов.

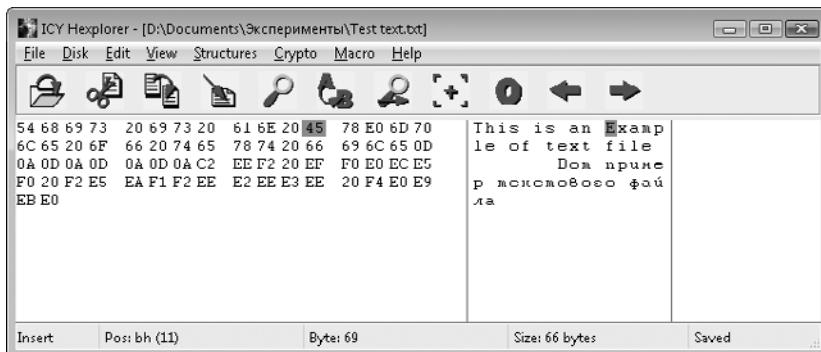


Рис. 1.1. Окно программы Hexplorer при открытом текстовом файле

Выделенный фрагмент одновременно подсвечивается в обеих половинах окна. Легко выяснить, что означает каждый байт файла, если считать, что им закодирован какой-либо текстовый символ. При рассмотрении текстовых файлов такая интерпретация совершенно справедлива. Команды меню *View* (Вид) и *Structures* (Структуры) позволяют интерпретировать содержимое файла иначе, например как точечный рисунок, и просматривать его в дочернем окне.

## Понятие формата файла

Существует множество типов и форматов файлов. В общем виде формат — это сведения о том, как программа должна интерпретировать содержимое файла при его открытии. Иногда формат путают с расширением имени файла, но это совершенно разные понятия. Операционная система может использовать расширение имени файла, чтобы быстро определить, какой программой должен открываться данный файл. Как правило, расширение должно соответствовать формату, но это вовсе не обязательно. Сведения о формате находятся в самом файле — они помещаются в начало последовательности байтов.

Текстовый файл — самый простой формат. Каждый байт текстового файла является кодом определенного символа ASCII (буквы, цифры или знака препинания), а также символов пробела, начала и конца строки. Кроме значений текстовых символов, в текстовом файле действительно больше ничего нет! Структура любых других типов файлов гораздо сложнее.

В качестве примера откроем в HEX-редакторе файл рисунка в формате BMP (рис. 1.2). В самом начале файла дана последовательность 42 4D — в текстовой интерпретации это буквы BM. Затем идет довольно длинная последовательность байтов, в которой преобладают нулевые значения, а уже за ней начинается массив самых разных значений, который продолжается до самого конца файла.

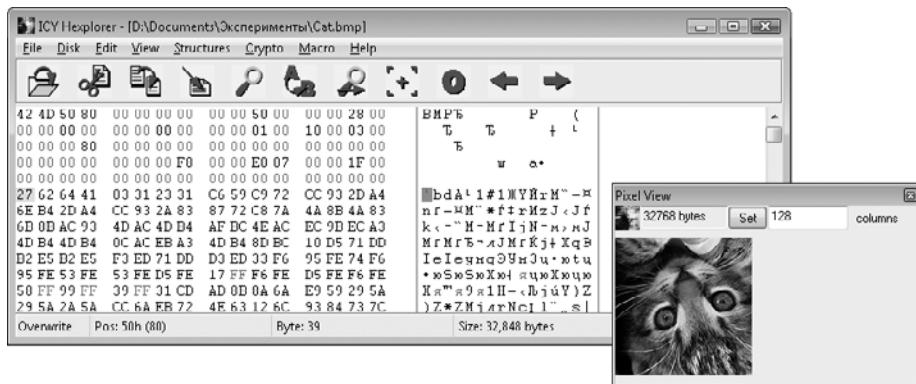


Рис. 1.2. Окно программы Hexplorer при открытом файле рисунка в формате BMP

Можно предположить, что начало файла — какая-то служебная информация, причем самые первые ее байты указывают на формат файла. Лишь затем следует то содержимое, которое программа, работающая с подобными файлами, должна интерпретировать как рисунок. Вы можете проверить данное предположение, обратившись, например, к статье в «Википедии» (<http://ru.wikipedia.org/wiki/BMP>).

### ПРИМЕЧАНИЕ



«Свободная энциклопедия» (<http://ru.wikipedia.org>) содержит огромное количество сведений и о форматах файлов, и о многом другом, что в книгу вместить просто невозможно. Еще одна полезная ссылка — <http://open-file.ru>. На этом сайте описаны почти все существующие форматы файлов.

В статье говорится, что BMP-файл действительно состоит из нескольких частей, точнее, из четырех:

- ❑ заголовок файла (**BITMAPFILEHEADER**). Первое поле заголовка (первые два байта) является сигнатурой — указанием на тип файла;
- ❑ заголовок изображения (**BITMAPINFOHEADER**) — необязательный компонент, он может отсутствовать;
- ❑ палитра, которая также может отсутствовать;
- ❑ само изображение — каждый байт описывает одну точку рисунка.

Здесь необходимо сделать важный вывод. Искажение или утрата данных, относящихся к первым трем частям, может привести к тому, что файл перестанет открываться предназначенной для этого программой, хотя вся значимая информация в нем цела. В этом случае появится сообщение о том, что файл имеет неправильный формат или поврежден. Утрата же или искажение данных самого изображения никак не повлияет на открытие файла, но в открывшемся рисунке выпадут или будут искажены отдельные фрагменты.

В качестве эксперимента можно сделать несколько копий файла, намеренно испортить каждую из них в разных местах с помощью HEX-редактора, а затем сохранить. Открывая эти файлы стандартным средством просмотра **Windows** или программой **Paint**, вы сможете убедиться в справедливости предположения: некоторые файлы не открываются, некоторые открываются, но с искажением содержимого.

Подобным образом устроен любой файл, хотя спецификация частей файла зависит от его типа. Описания почти всех форматов опубликованы, их можно найти в Интернете. Некоторые из детальных описаний занимают не один десяток страниц. Заметим, что вразумительное описание форматов файлов **Microsoft Office** найти труднее всего. Но все же существуют общие положения, к которым можно отнести следующие.

- В начале файла всегда находится его заголовок (Header). Сам заголовок тоже обладает внутренней структурой.
  - Заголовок начинается сигнатурой, или «магическим числом». Это фрагмент кода, который однозначно дает понять, что это за файл или какой программой он создан. Например, EXE-файл всегда начинается с символов MZ, архив RAR содержит в своем коде сигнатуру RAR.
  - Другие поля заголовка могут содержать информацию о размере файла, указывать, на каком байте заканчиваются заголовки и начинаются собственно данные, и т. д.
- В файле могут присутствовать специфические для данного формата заголовки, например для файлов изображений они указывают алгоритм сжатия, глубину цвета, размеры изображения в точках и т. д. Для звуковых файлов такими параметрами будут частота и разрядность оцифровки, количество каналов, алгоритм сжатия и пр.
- В некоторых файлах могут присутствовать вспомогательные компоненты: палитра, кодовая таблица, внедренные данные шрифтов, макросов и т. д. В отдельных форматах, наподобие документов **Microsoft Office** или **CorelDRAW**, такие данные могут занимать больше места, чем все остальное. Достаточно посмотреть, каков размер «пустого» документа **Microsoft Word 2003!**
- Собственно уникальные данные документа. Иногда в сложных документах эти данные организованы в виде цепочек, или порций (Chunks). В таком случае где-то в заголовках дается ссылка на начало и размер каждой порции.
- Указатель конца файла или контрольная сумма (Check sum) подтверждают целостность файла. Они присутствуют далеко не во всех форматах.

Очевидно, любой поврежденный файл можно восстановить или хотя бы извлечь из него полезную информацию, открыв и отредактировав такой файл HEX-редактором. Если испорчена информация о формате, ее (теоретически) можно воссоздать,

а затем открыть файл штатным приложением. Если повреждена область самих данных, из файла можно извлечь только то, что осталось.

Восстановление или извлечение полезной информации из поврежденных файлов вручную — трудоемкая задача. По сути, она сродни программированию. Проблема даже не в том, чтобы разобраться в структуре формата, а в том, что для поиска и создания новой записи нужных байтов потребуется масса рутинных и утомительных пересчетов, проб и проверок. Поэтому на практике сначала целесообразно прибегнуть к двум другим способам, если есть такая возможность.

## Восстановление файлов собственными средствами прикладных программ

В некоторых прикладных программах изначально заложена возможность тестирования и восстановления поврежденных документов. Эти возможности довольно ограничены, но иногда и их оказывается достаточно. С другой стороны, кому, как не разработчикам программ, известны все тонкости используемых форматов, а также алгоритмы их создания и восстановления.



### ВНИМАНИЕ

Перед любыми попытками восстановления «битого» файла необходимо сделать несколько его копий. Не стоит экспериментировать с единственным исходным экземпляром!

## Восстановление документов средствами Microsoft Word

В программе Microsoft Word предусмотрена функция открытия документов с восстановлением. Кроме того, в полный состав пакета Microsoft Office входит конвертер восстановления текста. По умолчанию он не устанавливается.

Замечу, что есть еще и третья функция, связанная с использованием автосохранения документов Word в процессе работы приложения. Однако она относится скорее к области резервного копирования: копии документа периодически сохраняются в той же папке, что и открытый документ, либо в папке `\Application Data\Microsoft\Word` в профиле пользователя. Если приложение Microsoft Word аварийно завершило работу до корректного сохранения документа пользователем, то при очередном его открытии появится панель Восстановление документов, на которой будут показаны существующие резервные копии файла. Скорее всего, при восстановлении действительно «битых» файлов эта функция не сработает.

Восстанавливать файлы следует средствами Microsoft Word той же версии, в которой документ был создан и сохранен, либо более новой. В Microsoft Word 2003

зайдите в окно настроек (команда меню Сервис ▶ Параметры) и на вкладке Общие установите флажок Подтверждать преобразование при открытии. В Microsoft Word 2007 этот параметр находится в окне настроек группы Дополнительно ▶ Общие. Сохраните настройки.

Дальше есть два способа. Сначала воспользуйтесь встроенной функцией восстановления документа.

1. В меню Файл выберите команду Открыть. Появится окно Открытие документа.
2. Выберите в нем файл для восстановления.
3. Нажмите на стрелку рядом с кнопкой Открыть и выберите команду Открыть и восстановить (рис. 1.3).

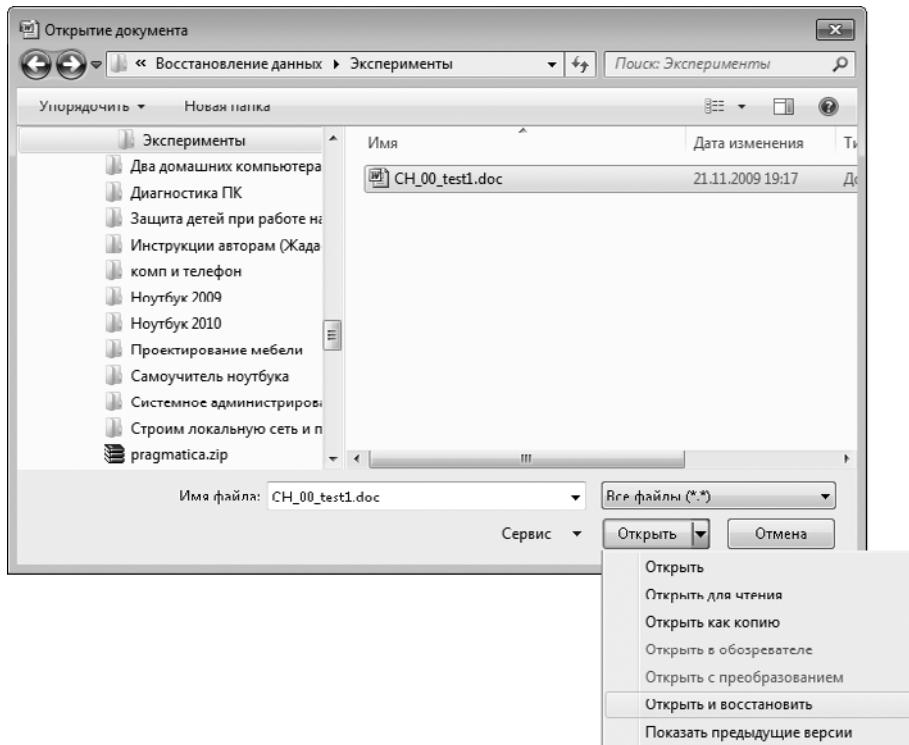


Рис. 1.3. Открытие документа с восстановлением

Возможно, документ успешно откроется. Открыв документ, сохраните его под другим именем, а лучше — скопируйте все содержимое и вставьте его в новый документ. Однако эксперименты показывают, что таким образом можно исправить только самые незначительные дефекты. Чуть большими возможностями обладает конвертер восстановления текста.