

# **HDD REPAIR TOOL DATA RECOVERY EDITION (HRT DRE)**

**Утилита для восстановления данных**

Детализированное описание утилиты HRT DRE  
[www.bvg-group.ru](http://www.bvg-group.ru)

## СОДЕРЖАНИЕ

СОДЕРЖАНИЕ .....	2
ВВЕДЕНИЕ .....	3
ОСНОВНОЕ ПРАВИЛО .....	3
ОСНОВНЫЕ ПРИНЦИПЫ ВОССТАНОВЛЕНИЯ ИНФОРМАЦИИ .....	4
Карта.....	6
Карта на основе головок.....	7
РАБОТА С УТИЛИТОЙ .....	8
Создание нового проекта .....	8
Открытие существующего проекта.....	12
Выбор порта .....	14
Главное окно программы .....	15
Окно карты .....	17
Вкладка Properties .....	20
Вкладка Manual Mark .....	23
Вкладка ReMark .....	24
Вкладка Options.....	26
Вкладка Error Processing.....	28
Вкладка Non-standard Disk Size .....	30
РАБОТА С ДЕРЕВОМ ФАЙЛОВ .....	31
Контекстное меню .....	35
Вкладка File View .....	37
МАРКИРОВКА КАРТЫ НА ОСНОВЕ КАРТЫ ГОЛОВОК .....	39

## ВВЕДЕНИЕ

Утилита HRT DRE предназначена для восстановления данных с неисправных накопителей. Все свои возможности она развивает в паре со специализированными утилитами комплекса HRT, но возможна работа с ней и без внешних утилит, просто часть возможностей при этом не будет реализована.

## ОСНОВНОЕ ПРАВИЛО

Первое правило при восстановлении информации с неисправного накопителя: **НАКОПИТЕЛЬ В ЛЮБОЙ МОМЕНТ МОЖЕТ ОКОНЧАТЕЛЬНО ВЫЙТИ ИЗ СТРОЯ, ПОЭТОМУ ЧИСЛО ОБРАЩЕНИЙ К НЕИСПРАВНОМУ НАКОПИТЕЛЮ ДОЛЖНО БЫТЬ МИНИМАЛЬНЫМ.** Именно из этого правила выводится большинство концепций, заложенных в утилиту.

## ОСНОВНЫЕ ПРИНЦИПЫ ВОССТАНОВЛЕНИЯ ИНФОРМАЦИИ

Исходя из того, что число обращений к неисправному накопителю должно быть минимальным, нельзя производить глобальный анализ файловой структуры на нём самом. Первоначально необходимо скопировать данные на исправный накопитель. В качестве приёмника информации может выступать как физический накопитель равной или большей ёмкости, так и файл-образ. Однако, следует помнить, что файловая система FAT не позволяет полноценно работать с файлами более 2Гб (а более 4Гб – не только полноценно, но и неполноценно), поэтому файлы-образы лучше хранить на разделе с файловой системой NTFS.

Файл-образ в дальнейшем может быть скопирован на другой накопитель, либо проанализирован внешними программами, рассчитанными на работу с файлами-образами. Кроме того, файл-образ можно подключить в систему, как виртуальный диск и различные внешне утилиты восстановления данных (например, R-Studio, да и проводник ОС Windows) не заметят, что физически работают с файлом. Они будут уверены, что работа ведётся с физическим накопителем.

Посекторное копирование некоторых накопителей может занять достаточно много времени (сутки и даже недели). Чтобы ускорить процесс, необходимо применять различные ухищрения. Самое простое из них – если встретилось некоторое небольшое количество BAD-секторов подряд, предположить, что идёт пачка BAD-блоков и сделать пропуск большого количества секторов. При этом процесс ускоряется очень и очень значительно, а потери обычно – минимальны.

С другой стороны, иногда известны пути к тем или иным файлам, которые следует восстановить. ОС при обращении к этим файлам, «зависает», так как кроме записей, относящихся к нужным веткам дерева файлов, она анализирует сотни и тысячи ненужных веток. А ведь достаточно пометить к копированию на приёмник только те участки, на которых расположены интересующие нас файлы, и не копировать всё остальное. Такой приём также позволяет уменьшить время, требуемое на создание копии неисправного накопителя.

А как быть с BAD-блоками? Что делать с их содержимым? Такие блоки можно заполнять той или иной сигнатурой (типичные варианты – бесконечная строка BAD!BAD!BAD!, удобная для анализа или нейтральная константа 00000000, прелесть которой заключается в том, что если какое-то ПО берёт из такого сектора указатель, оно не уйдёт в неведомые дали). Это хорошо для тех накопителей, которые вместо ошибочных секторов возвращают нечто странное. С другой стороны, некоторые накопители (например, IBM) возвращают вполне осмысленные данные. Для таких накопителей лучше игнорировать флаг ошибки и записать данные из буфера на приёмник. Вполне возможно, что впоследствии из этих данных удастся сложить пусть повреждённый, но приемлемый файл. И, наконец, можно попытаться вычитать данные статистическим методом, то есть, считать их несколько раз и положить на каждое место то значение, которое лежало на этом месте наиболее частое количество раз. Этот способ хорош, если накопитель отдаёт данные неустойчиво

Итак, из всего вышесказанного, можно сделать выводы:

1. Данные следует копировать на исправный носитель и дальше работать уже с образом данных.
2. Образ может располагаться как на физическом накопителе, так и в файле (но файл должен быть расположен на NTFS-томе).
3. Если встретилось много BAD-блоков подряд, лучше пропустить большой участок и попытаться счастья за ним.
4. Копировать лучше только заведомо нужные участки.
5. Даже из BAD-блоков можно попытаться вычитать данные.

## **Карта**

Для реализации всех этих принципов, в утилиту введено понятие карты. Каждому сектору неисправного диска соответствует один элемент карты, который отмечает состояние сектора. Изначально, все сектора имеют метку «С данным сектором не следует выполнять никаких операций». Поэтому, если просто выбрать накопитель и начать процесс его копирования, то вскоре Вы получите радостное сообщение, что процесс завершён, но результатов не будет никаких. И это понятно. Чтобы минимизировать число обращений к источнику, карта пуста. Вы должны самостоятельно внести в неё пометки, которые заставят утилиту копировать те или иные сектора. В простейшем случае – пометьте всю карту и процесс пойдёт. Более сложные методы пометки карты будут описаны ниже.

По мере прохождения процесса, сектора из состояния «Должен быть скопирован» будут переходить либо в состояние «Скопирован», либо в то или иное проблемное состояние. Через равные промежутки времени, карта будет автоматически сохраняться. То есть, если система «зависнет», либо отключать электроэнергию или случится иная непредвиденная ситуация, пропадут только изменения состояния, произошедшие с момента предыдущего автосохранения. Все остальные сведения будут запомнены, и процесс начнётся не с самого начала.

В целом, процесс можно прерывать и начинать многократно, ведь он базируется на карте, а карта – отражает текущее состояние. Надо прервать копирование для срочного ремонта – прекрасно, прервали, затем – загрузили и начали процесс с точки останова. Выделили пофайлово недостаточный участок и необходимо докопировать другие – никаких проблем, отмечайте их и копируйте. Скопировали всё, но есть подозрения, что некоторые пропущенные (после множественных BAD-блоков) участки содержат полезную информацию – тоже всё просто, перемаркировали «Пропущенное» в «Следует скопировать» и провели процесс. Всё, что уже скопировано – не будет затронуто. То есть, не стоит бояться, что какой-либо дополнительный процесс займёт очень много времени за счёт повторных обращений. Нет, никаких повторов не будет. Время будет затрачено только для новой операции.

При отметке файлов, карта также будет самостоятельно «зеленеть». Это связано с тем, что для файлового разбора необходимо считывать ту или иную служебную информацию. А раз она всё равно считывается, то производится её автоматическая запись на приёмник. Если в дальнейшем она понадобится (при втором проходе, при иных обстоятельствах), все «зелёные» сектора будут считаны не с источника, а с приёмника.

## **Карта на основе головок**

Типичная операция при восстановлении данных с неисправных дисков – перестановка головок. При этом, возникают различные трудности. Допустим, Вам удалось найти полностью совместимые головки. Но за счёт того, что сервометки записываются уже после сборки гермоблока, даже если Вы подберете угол наклона оси головок, всё равно часто при переходе с диска на диск, накопитель теряет сервометки. В лучшем случае, это сопровождается нежелательными задержками, в худшем – стуками БМГ.

А может быть и хуже. Например, у накопителя включены головки 1 и 2, а у Вас есть совместимый БМГ, содержащий головки 0 и 1, а также БМГ, содержащий головки 2 и 3, но нет исправного БМГ с головками 1 и 2.

В обоих случаях, на помощь придёт раздельное копирование по головкам. Необходимо сначала скопировать все данные, относящиеся к головке 0, затем – все данные, относящиеся к головке 1 и т.д. (а в приведённом примере с двумя БМГ – сначала данные с головки 1, затем – сменить БМГ и скопировать данные с головки 2).

Копирование данных по физическим координатам (PCHS) затруднено, так как Вам необходимо иметь приёмник, на 100% идентичный источнику. Даже это почти нереально, ведь в современных накопителях могут быть десятки тысяч дефектов, скрытых в таблице P-LIST. Можно попытаться считывать данные по физике, а записывать их – уже по логике, то есть, замещать утилитой функции транслятора. Но в современных накопителях имеется великое множество подходов к построению трансляторов и придётся писать не одну, а десять или более утилит-копировщиков.

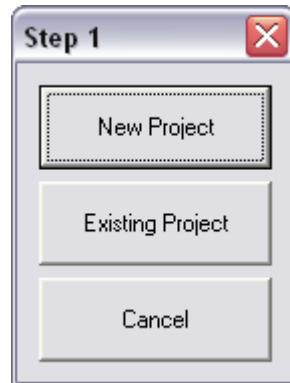
Наиболее удобный вариант – копировать данные в логических координатах, но метить в карте под копирование только сектора, принадлежащие той или иной головке. Так как трансляция – процесс длительный и не стоит проводить его столько раз, сколько у накопителя головок, в карте введено несколько резервных состояний. При построении карты по головкам, только одну помечаем на копирование, в остальные – переводим в различные резервные состояния. Когда завершится проход копирования текущей головки, можно будет просто перемаркировать очередное резервное состояние для копирования и запустить очередной проход на исполнения.

Так как преобразования к физическим координатам зависят от накопителя, для них необходимо использовать специализированные утилиты комплекса HRT.

## РАБОТА С УТИЛИТОЙ

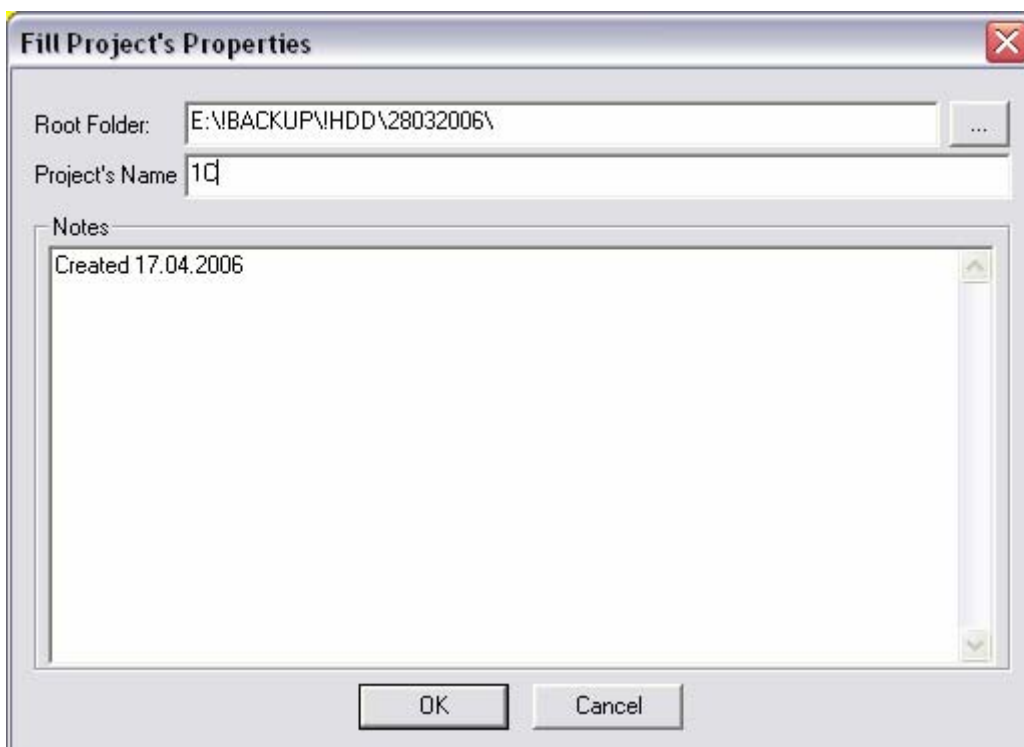
При работе с утилитой, используется понятие «проект». Это каталог, в котором сохраняются все необходимые файлы, а также – служебные пометки, которые могут добавляться в процессе работы.

При запуске утилиты, Вам будет предложено создать новый проект или выбрать ранее созданный:



### ***Создание нового проекта***

Если выбрать создание нового проекта, то Вам будет предложено заполнить его свойства:



Строка Root Folder определяет каталог, в который будут помещаться все проекты. По умолчанию, она соответствует каталогу, в котором расположен исполняемый файл. Однако, при необходимости, каталог по умолчанию может быть изменён путём правки строки Root Folder секции [DRE] файла HDD.INI.

Строка Project's Name определяет уникальное имя проекта. На её основе будет сформировано имя каталога, в который будет помещаться вся информация, относящаяся к проекту.

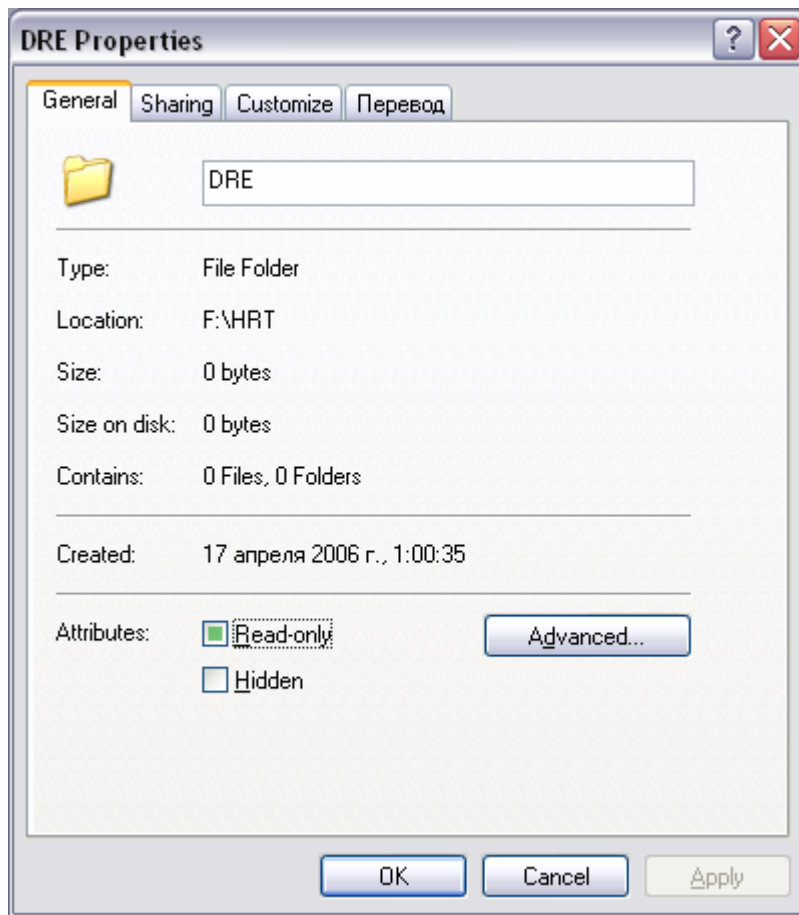
В строке Notes Вы можете делать любые заметки, касающиеся ремонта. Кто принёс, что необходимо восстановить, какие проблемы были в процессе работы и т.п. Заметки могут быть изменены при каждом последующем открытии проекта. По умолчанию, туда подставляется дата создания проекта, так как она необходима в большинстве случаев.

После нажатия на кнопку ОК, Вам будет предложено выбрать тип приёмника.

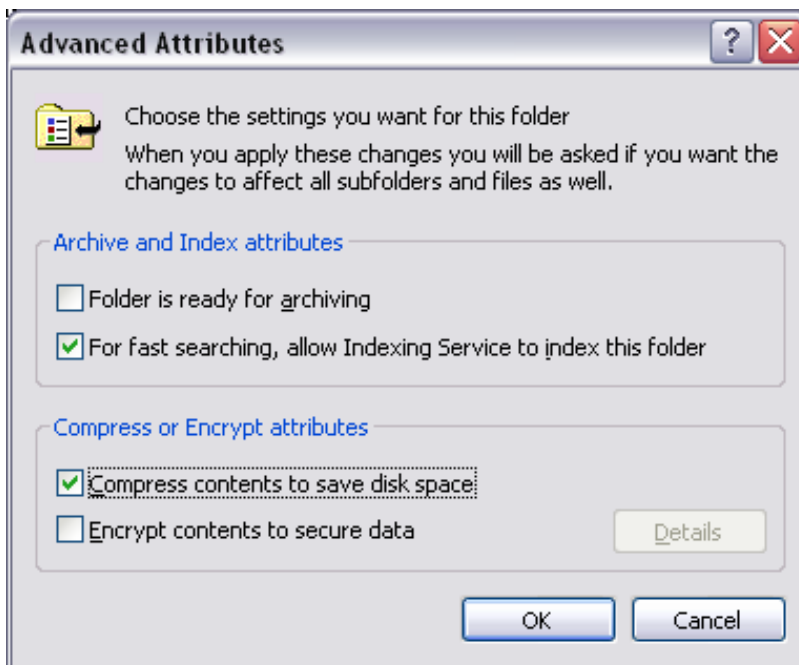


Приёмник типа файл будет располагаться в каталоге проекта и иметь имя 0.BIN. **БУДЕТ НЕ ЛИШНИМ ПОВТОРИТЬ, ЧТО ПРОЕКТЫ ЖЕЛАТЕЛЬНО РАЗМЕЩАТЬ НА ТОМАХ NTFS, ПРИ ЭТОМ, СЛЕДУЕТ СЛЕДИТЬ, ЧТОБЫ НА ТОМЕ БЫЛО ДОСТАТОЧНО СВОБОДНОГО МЕСТА.** Также следует напомнить, что NTFS предоставляет возможность сжимать содержимое каталогов. Временные затраты при этом невелики, так что рекомендуется каталогу проектов присвоить атрибут «Сжатый».

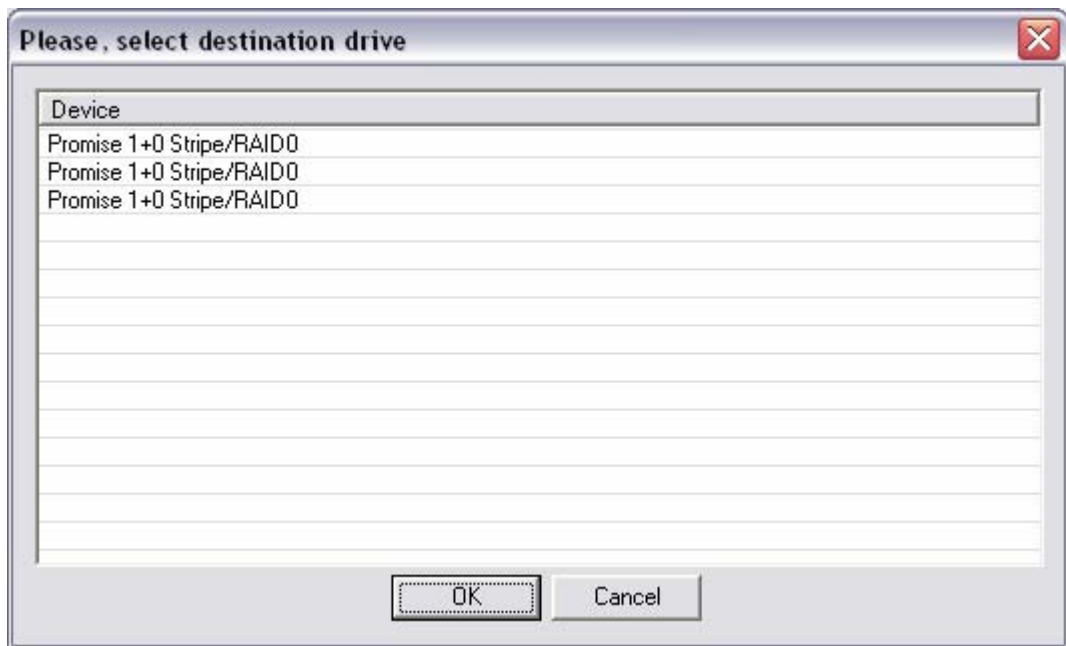
Чтобы установить атрибут файла «Сжатый» следует выбрать каталог проекта (или корневой каталог всех проектов, если предполагается тотальное сжатие) средствами проводника Windows и вызвать его свойства:



Здесь нажмите кнопку «Другие» и в появившемся окне установите флажок «Сжимать содержимое для экономии места на диске».



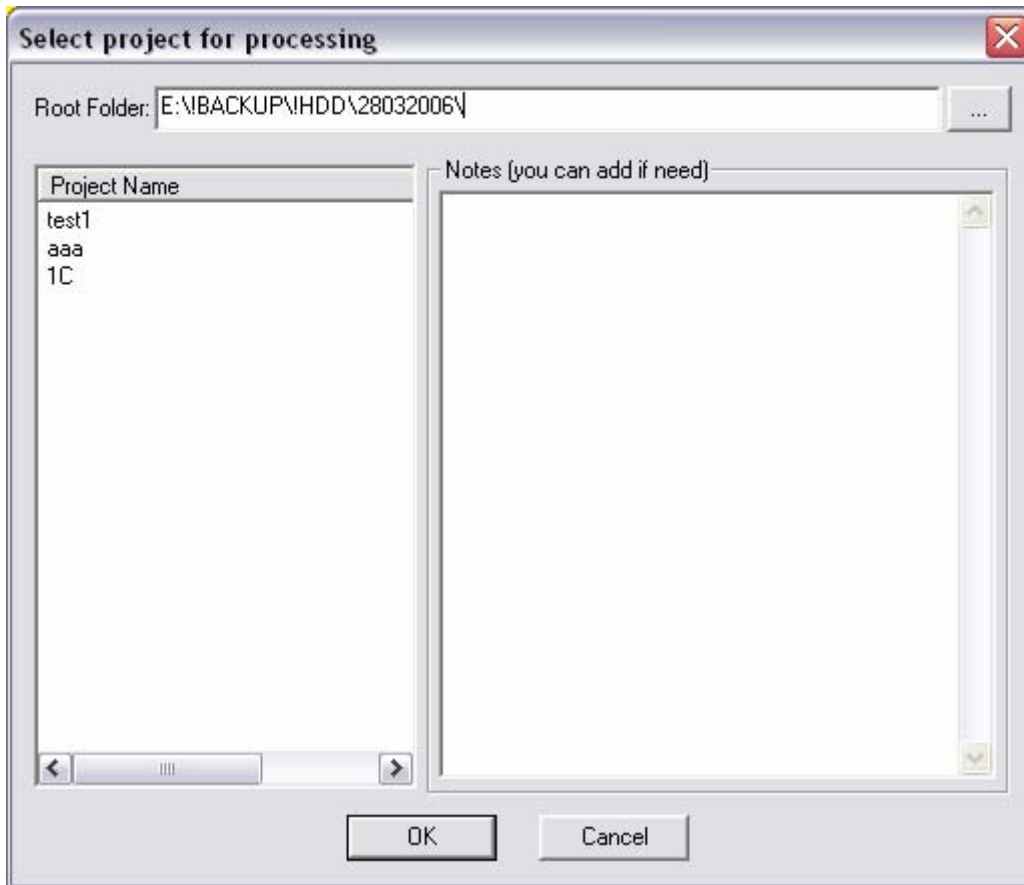
В случае с приёмником-накопителем, Вам будет дополнительно предложено выбрать его.



**БУДЬТЕ ВНИМАТЕЛЬНЫ И НЕ ВЫБИРАЙТЕ СИСТЕМНЫЕ ДИСКИ. В ПРОТИВНОМ СЛУЧАЕ, РАНО ИЛИ ПОЗДНО, СИСТЕМА ПЕРЕСТАНЕТ РАБОТАТЬ.** Это – ещё одна причина, по которой не рекомендуется выбирать в качестве приёмника физические накопители, а пользоваться файлами-образами.

## Открытие существующего проекта

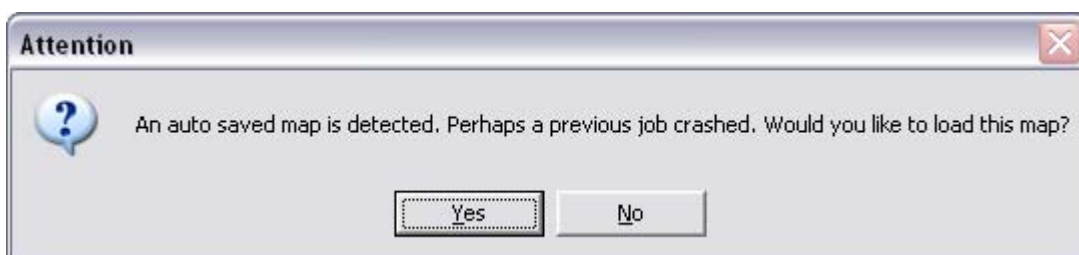
При открытии существующего проекта, Вам будет выведен список всех проектов, расположенных в текущем корневом каталоге



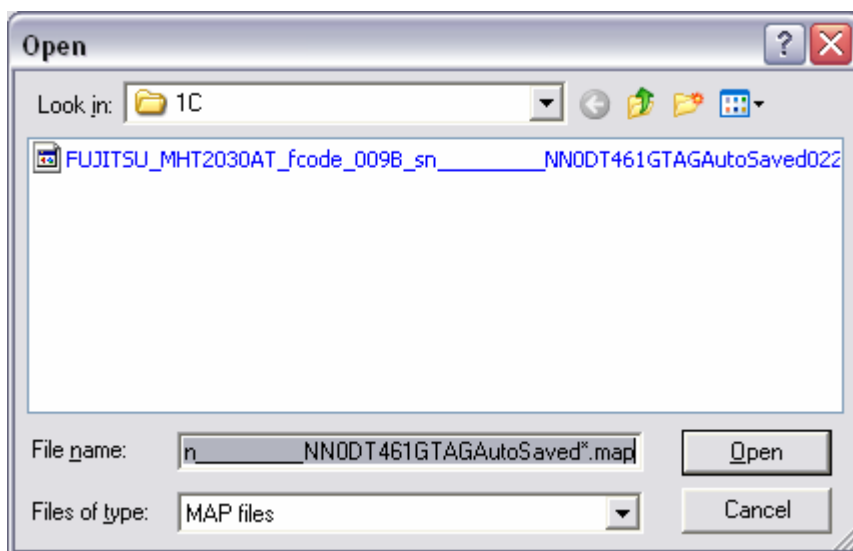
Сменить корневой каталог можно, нажав кнопку «...»

При выборе тех или иных записей в списке проекта, также будет отображаться комментарий, связанный с данным проектом. Вы можете не только просмотреть комментарий, но и отредактировать его. При нажатии на кнопку ОК, изменения будут сохранены, а проект – открыт.

С каждым проектом неразрывно связана карта. Карта может быть сохранена автоматически или вручную. Если при открытии проекта в его каталоге обнаружен хотя бы один файл с автоматически сохранённой картой, утилита выдаст запрос:

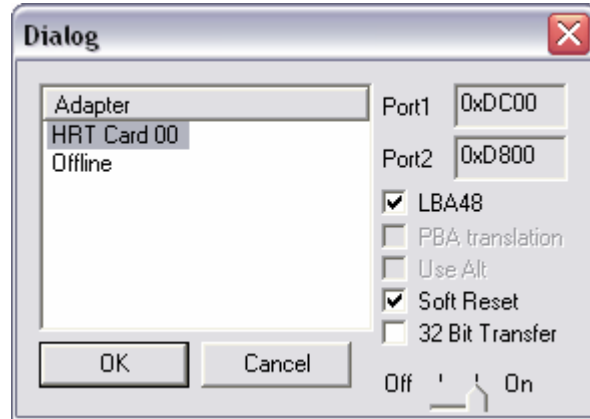


Крайне рекомендуется ответить на него положительно и выбрать автосохранённую (или иную) карту. Если отказаться от загрузки карты, то вся история предыдущей работы не будет восстановлена и процесс копирования начнётся с начала. Выбор файла с картой осуществляется штатным диалогом Windows. Стоит особо обратить внимание, что синий цвет шрифта в диалоге означает, что все файлы, расположенные в нём – сжатые средствами ОС Windows. Карта сама по себе занимает достаточно много места, но прекрасно архивируется, поэтому ещё раз обращаем Ваше внимание, что в ФС NTFS можно включить режим сжатия.



## Выбор порта

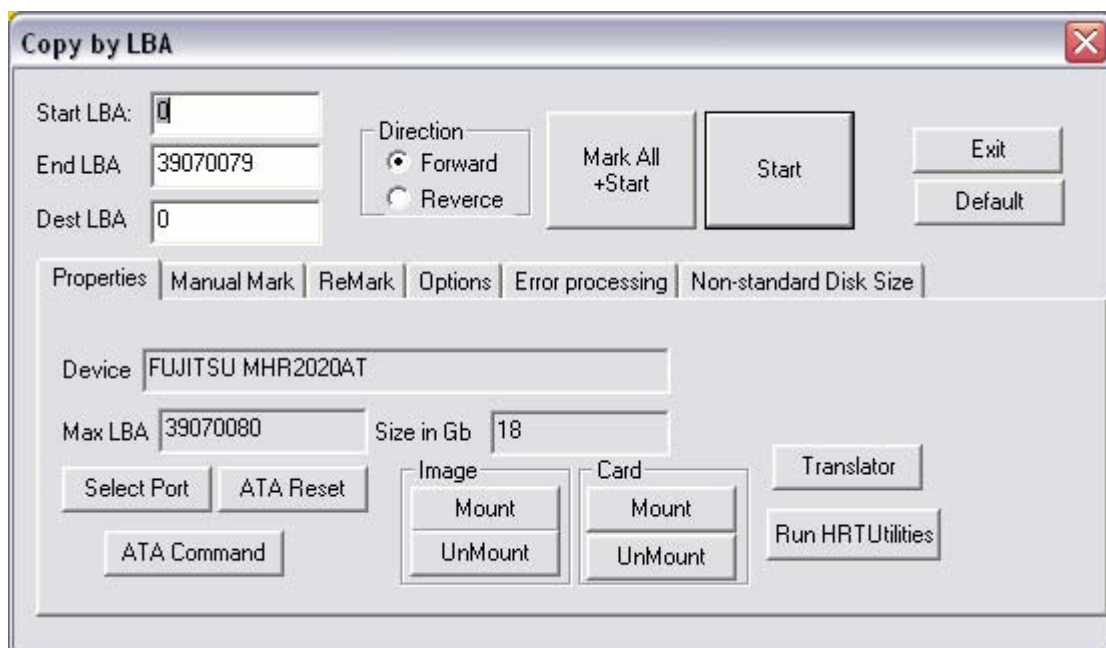
Диалог выбора порта – стандартный для комплекса HRT. Однако, стоит особенно остановиться на флажке 32 Bit Transfer.



При копировании данных, крайне важна скорость доступа к накопителю. Чем выше скорость, тем быстрее данные будут скопированы. При современных объемах, при копировании в режиме PIO речь идёт не о часах, а о десятках часов. Чтобы увеличить скорость вдвое, в адаптер HRT версии L23 и новее, был введён 32 разрядный режим доступа к накопителю. Однако он работает не на всех материнских платах (например, точно не работает на платах с чипсетом VIA, по некоторым сведениям, зависает на некоторых материнских платах с чипсетом Intel, но на платах фирмы ASUS с чипсетом Intel такого эффекта не обнаружено). Поэтому, крайне желательно, чтобы данный флажок при работе с утилитой был установлен. Но если у Вас чипсет VIA или при копировании происходят спонтанные «зависания» системы, то снимите данный флажок.

## Главное окно программы

Главное окно программы содержит общую часть, расположенную сверху, а также переключаемые вкладки, расположенные ниже. В общей части сосредоточены параметры копирования.



Начальный и конечный LBA в комментариях не нуждаются. Они задают координаты начала и конца выполнения операции. Параметр Dest LBA обычно равен начальному LBA. Его назначение следующее: Допустим, Вы работаете с накопителем большой ёмкости. С него следует восстановить достаточно небольшой раздел. У Вас в наличии имеется только накопитель-приёмник малой ёмкости, на который указанный раздел поместится, а весь накопитель - нет. Как известно, ОС Windows прекрасно справится с накопителем, у которого нет таблицы разделов, а с LBA0 расположен сам раздел (режим «большой дискеты»). Параметр Dest LBA позволяет создать такую «большую дискету». Достаточно выбрать параметр Start LBA, равный началу копируемого раздела, а Dest LBA равным нулю. И данные будут смещены в начало накопителя-приёмника (или файла-образа). Как всегда, напомним, что при использовании сжатых каталогов NTFS данная проблема не стоит. Она стоит при копировании на накопитель-приёмник или в обычный каталог NTFS. Узнать параметры разделов можно через механизм Bark By FS, описанный ниже.

Группа кнопок Direction задаёт направление копирования. Оно может вестись или вперёд или назад. Обычно накопители копируются вперёд. Если же в начале имеется много секторов, замедляющих скорость копирования («хрюкающие» сектора), то можно запустить копирование назад. Как ни странно, иногда при копировании назад, удаётся вычитать даже те сектора, которые выдавали ошибку при прямом копировании. То есть, смена направления может быть произведена, если у Вас возникли затруднения при копировании вперёд. Причём достаточно прервать копирование, сменить направление и снова запустить процесс. Так как копирование базируется на карте, к накопителю-источнику не будет произведено ни одного лишнего обращения. Как только процесс дойдёт до той точки, где прервалось копирование в прямом направлении, он завершится.

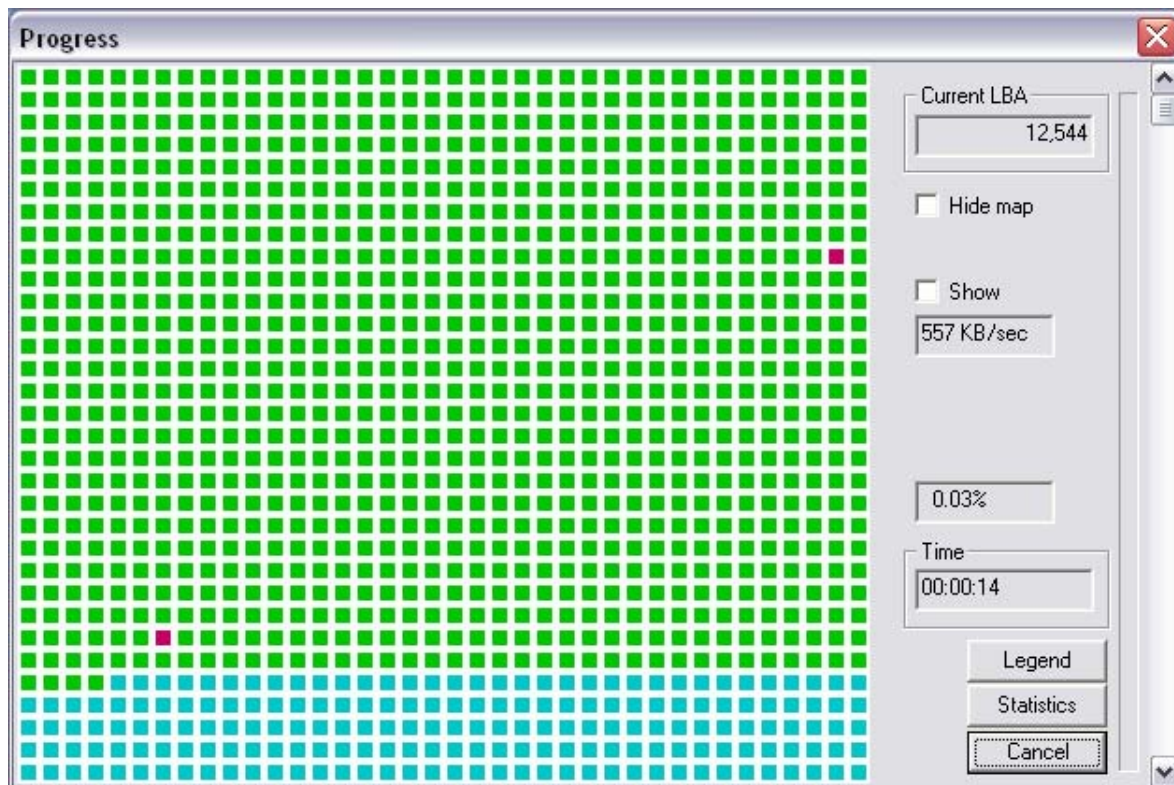
Кнопка Mark All+Start запускает самый простой процесс копирования – без предварительной подготовки. Она просто отмечает ВСЕ сектора накопителя-источника, как подлежащие копированию и начинает процесс. **В простейшем случае, из всех кнопок нужна только эта. А все настройки при этом можно оставить по умолчанию.**

Кнопка же Start начнёт процесс для заранее подготовленной карты. Если карта не подготовлена (проект только что создан), процесс завершится довольно быстро, к источнику не будет произведено ни одного обращения, а на приёмник не будет записано ни одного байта данных. Карту следует подготовить одним из следующих методов (каждый из которых будет рассмотрен ниже):

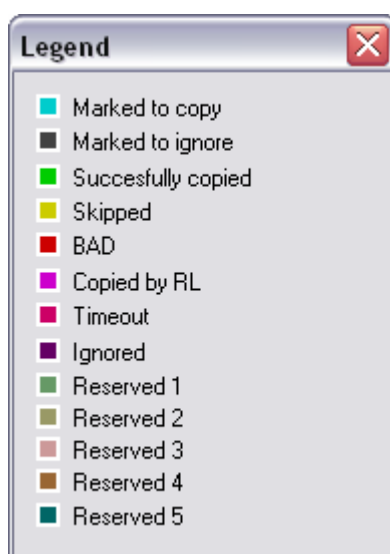
- Прямым редактированием.
- Редактированием на основе Файловой Системы.
- Поголовочным построением при помощи специализированной утилиты комплекса HRT.
- Редактированием карты по результатам предыдущего прохода копирования.

## Окно карты

Текущее состояние карты всегда можно посмотреть в параллельном окне карты, внешний вид которого представлен ниже:

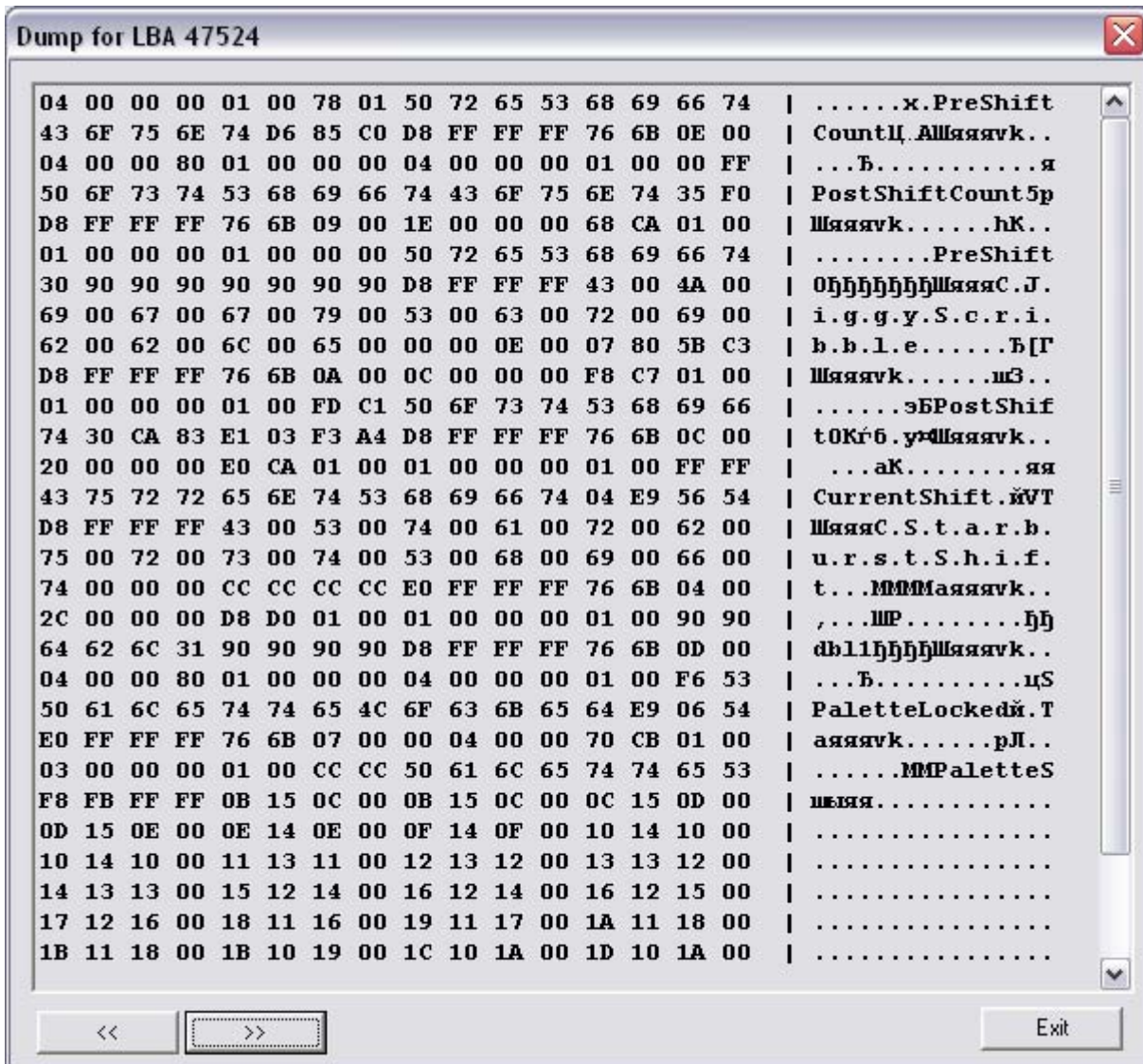


Один квадратик в карте соответствует одному сектору копируемого накопителя. Цвет квадратика определяет состояние сектора. Соответствие цветов состояниям можно узнать, нажав на кнопку Legend



Поле Current LBA показывает номер текущего обрабатываемого сектора. Во время копирования это текущий копируемый сектор. Если же щёлкнуть по любому из квадратиков «мышью», то в поле появится значение LBA, соответствующего квадратику.

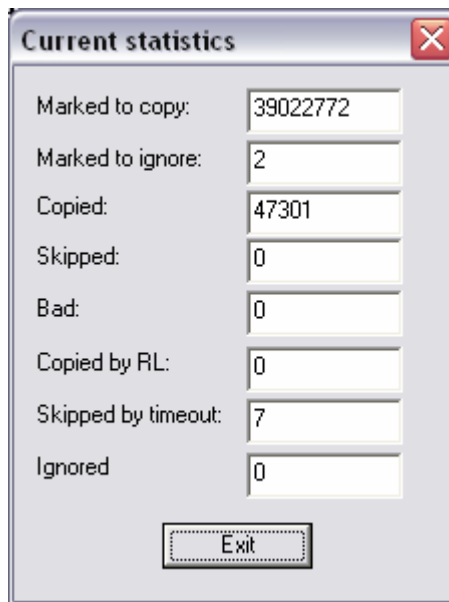
Двойной щелчок по квадратику приведёт к отображению его дампа



Флажок Hide Map позволяет ускорить процесс копирования примерно на 30% за счёт того, что система не будет тратить время на перерисовку карты. Рекомендуется снимать его только в случаях, когда процесс копирования подтормаживает, чтобы рассмотреть характер проблемы, а в остальных случаях – устанавливать.

Поле Time автоматически обновляется в процессе копирования, отображая время, затраченное на сам процесс.

Кнопка Statistics позволяет посмотреть текущую статистику отметки секторов в карте



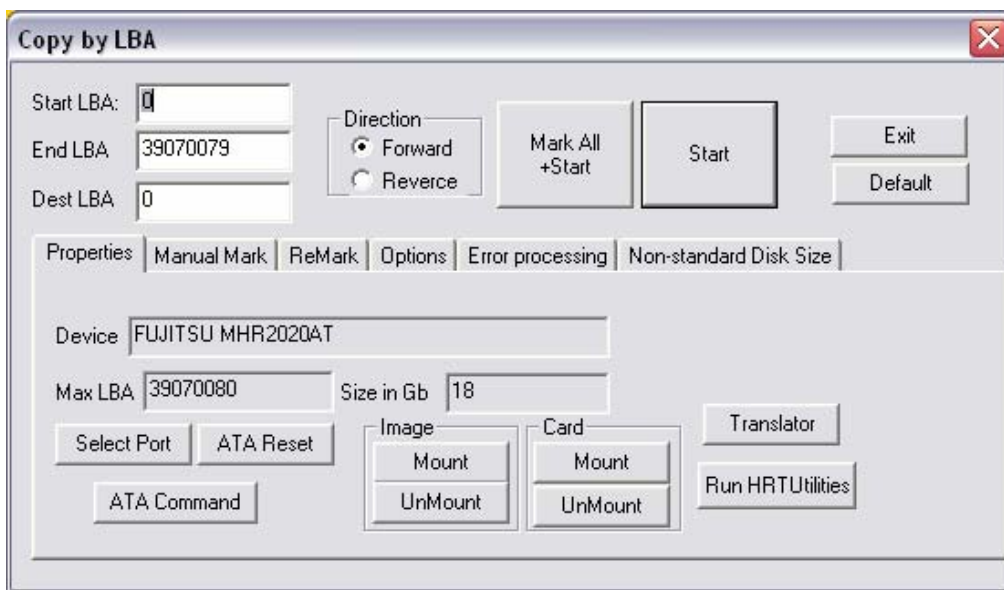
The image shows a dialog box titled "Current statistics" with a close button (X) in the top right corner. It contains several rows of labels and text boxes with numerical values:

Marked to copy:	39022772
Marked to ignore:	2
Copied:	47301
Skipped:	0
Bad:	0
Copied by RL:	0
Skipped by timeout:	7
Ignored	0

At the bottom center of the dialog box is a button labeled "Exit".

Кнопка Cancel работает, когда идёт процесс копирования, прерывая его. Это необходимо для временной приостановки процесса, например, если необходимо подправить параметры, ответственные за обработку ошибок, построить карту по головкам, изменить направление копирования или, в конце концов, использовать адаптер HRT для более срочного заказа. Впоследствии процесс будет продолжен согласно карте, то есть, с точки, где было произведено его прерывание.

## Вкладка Properties



Данная вкладка отображает свойства накопителя, а также - кнопки вспомогательных операций. Свойства накопителя необходимы для контроля качества его определения. Если свойства не заполнены или заполнены некорректно, накопитель требует предварительного ремонта при помощи специализированных утилит комплекса HRT. Иногда удаётся обойтись без специальных действий при неверном определении длины за счёт вкладки Non Standard Disk Size, описанной ниже, но это скорее исключение, чем правило, так как обычно IDE накопители сами отсекают попытки обращения за пределы рабочей области.

Кнопка Select Port позволяет выбрать адаптер HRT. Это необходимо в том случае, если Вы входили в утилиту через режим Offline.

Кнопка ATA Reset позволяет послать накопителю сигнал сброса.

Кнопка Mount Card позволяет временно подключить адаптер HRT в роли системного адаптера. При этом, накопитель, подключённый к адаптеру, будет считаться системой обычным накопителем. Это бывает полезно для оперативно оживлённых накопителей, у которых ни в коем случае нельзя отключать питание, с которых необходимо считать малое число файлов. В этом случае, достаточно оживить накопитель средствами комплекса HRT, после чего смонтировать адаптер в системе и быстро скопировать необходимые файлы.

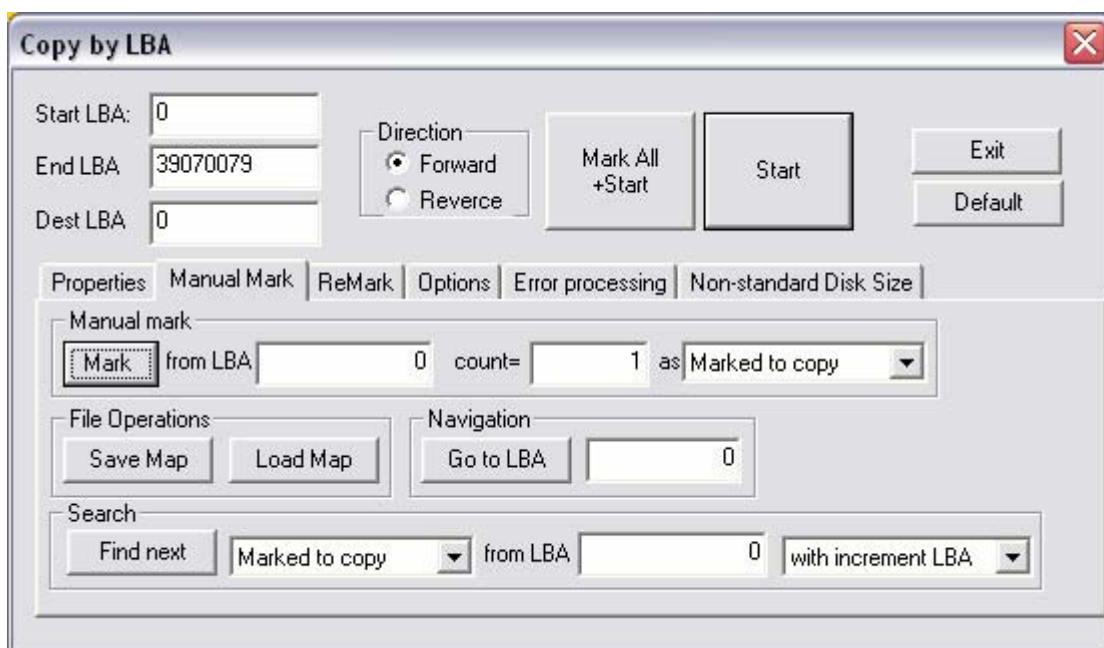
Кнопка UnMount Card – напротив, отключает монтирование адаптера в систему. Не следует постоянно держать адаптер смонтированным, так как во-первых, в это время не рекомендуется обращаться к нему из любых программ комплекса HRT (во избежание конфликтов), а во-вторых, ОС WINDOWS очень часто зависает, когда к ней подключён слабо читающийся накопитель. В нашем случае, накопитель именно неисправен и ОС имеет все права «зависнуть» без предупреждения.

Кнопки Mount Image и UnMount Image позволяют смонтировать или размонтировать файл-образ. Работа с файлом-образом гораздо более устойчива, в виду того, что на нём отсутствуют BAD-блоки. Но всё равно, ОС WINDOWS может некорректно работать даже и при логических проблемах в файловой системе.





### Вкладка Manual Mark



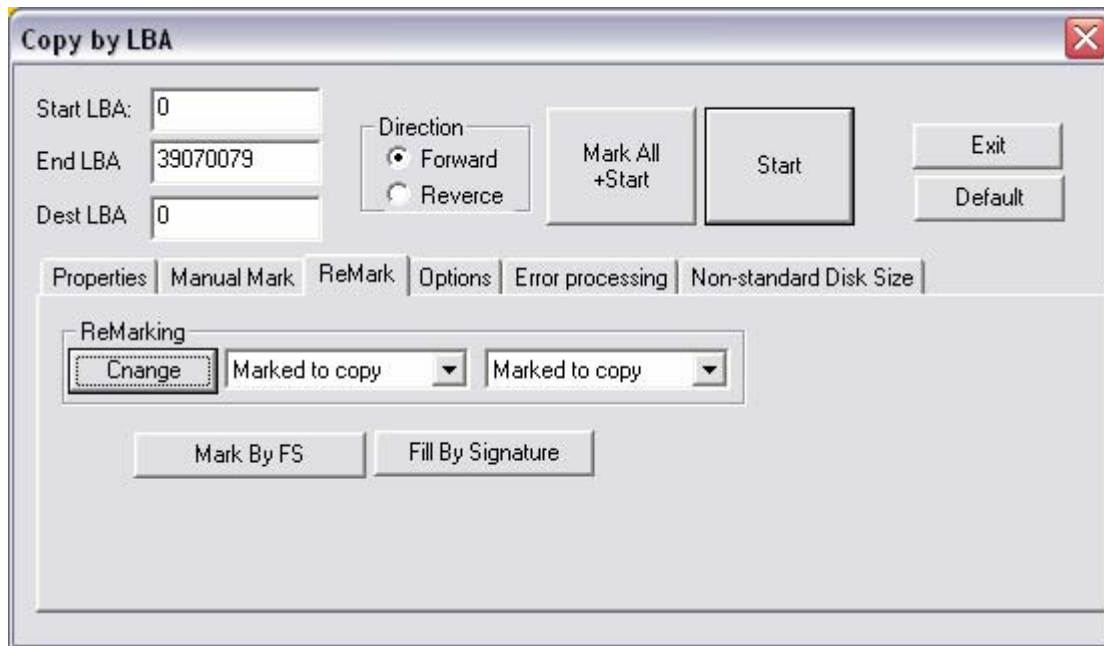
Данная вкладка предназначена для ручной маркировки карты.

Группа элементов Manual Mark позволяет отметить какой-либо участок для исполнения. Например, это может быть удобно, чтобы отметить для копирования один раздел, координаты расположения которого Вам известны.

Группа File Operations позволяет сохранить текущую карту (кнопка Save Map) или загрузить её (кнопка Load Map). Это может пригодиться для создания резервной копии карты перед началом ручных операций с ней.

Кнопка Search позволяет найти ближайший элемент карты, имеющий заданное значение. Она удобна для поиска BAD-блоков, либо областей, пропущенных при копировании или иных сходных целей.

### Вкладка ReMark



Данная вкладка предназначена для автоматизированной или полуавтоматизированной перемаркировки карты.

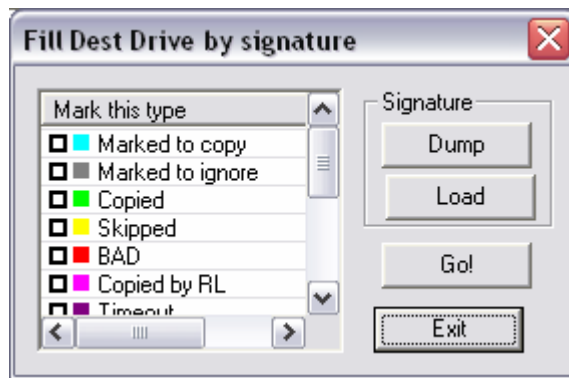
Группа элементов Change позволяет изменить все элементы со значением из левого поля, присвоив им значения из правого поля. Границы начала и конца обработки берутся из полей Start LBA и End LBA главной части окна.

Данная операция может пригодиться для следующих целей:

- Смена полей ReservedX на Marked to copy с целью начала копирования очередной головки после завершения работы с предыдущей.
- Смена полей Marked To Copy на Marked To Ignore или ReservedX при раздельном копировании по головкам, если видно, что текущая головка даёт массовые Bad блоки.
- Смена полей Skipped на Marked to Copy, если после копирования хочется вычитать все пропущенные сектора.
- Для прочих сходных целей.

Кнопка Mark By FS предназначена для перемаркировки карты на основе файловой системы. Данный механизм будет рассмотрен ниже.

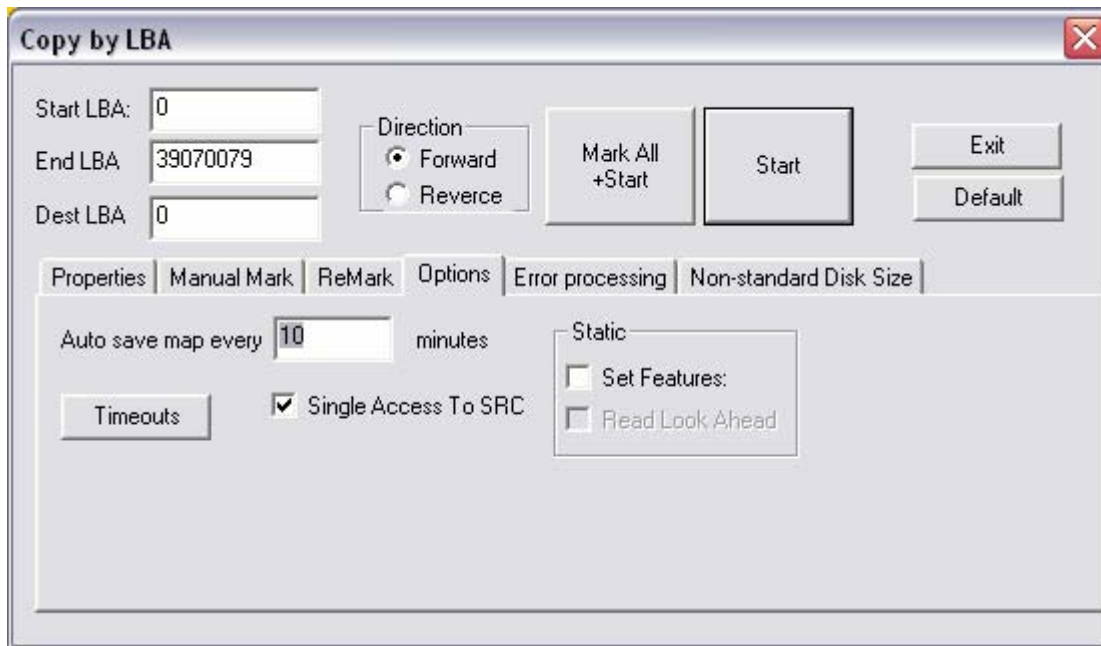
Кнопка Fill By signature производит заполнение секторов приёмника той или иной сигнатурой после завершения процесса копирования. Это может быть полезно для заполнения пропущенных секторов или иных целей. При нажатии на эту кнопку, появляется дополнительное окно настройки:



Установите флажки около типов секторов, подлежащих заполнению (допускается выбрать не одно, а несколько значений, например, все поля ReservedX или иные).

Сигнатуру можно задать вручную (кнопка Dump) или загрузить из файла (кнопка Load). После этого, нажмите кнопку Go! и начнётся процесс заполнения. Обращений к источнику при этом не будет.

## Вкладка Options

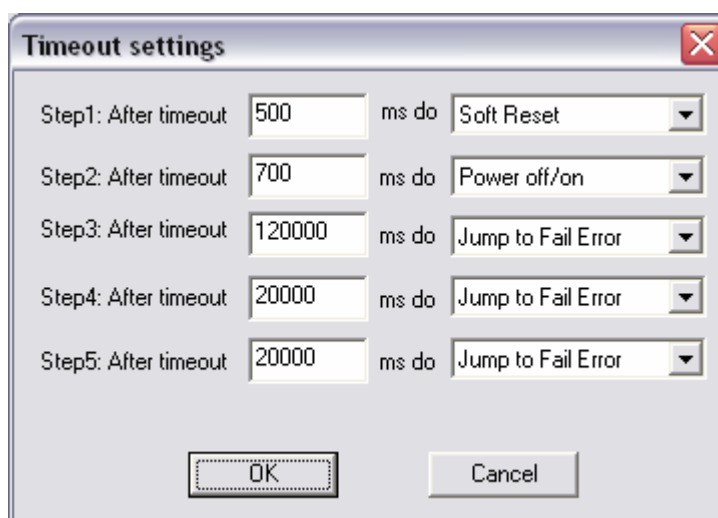


На данной вкладке сосредоточены настройки утилиты.

Поле Auto Save Every XXX minutes – задаёт период, с которым будет производиться сохранение карты. Чем меньше этот период, тем меньше потеряется при «зависании» системы. С другой стороны, время сохранения карты для накопителя, ёмкостью 300 гигабайт, достаточно значительно. И чем чаще карта сохраняется, тем больше накладные расходы по времени. Поэтому оптимальное значение – 1 раз в 10 минут.

Кнопка Timeouts позволяет настроить реакцию утилиты на таймауты, связанными с «задумчивостью» или «полным зависанием» накопителя. При этом, накопитель слишком долго не сбрасывает флаг BSY. Иногда «слишком долго» - это единицы секунд, иногда – минуты, а иногда – бесконечность. Чтобы вывести накопитель из «зависшего» состояния чаще всего требуется подать на него сигнал Reset. Этот метод является достаточно быстрым, но не всегда дающим желаемый результат. Некоторые накопители даже после подачи сигнала Reset не выходят в готовность. В этом случае, можно выключить питание и включить его снова. После применения данного метода процессор накопителя точно выйдет из «зависания», но накопитель был выведен в готовность при помощи «плясок с бубном» в комплексе HRT, то все эти «пляски» придётся повторять заново. А зависаний во время копирования могут быть сотни или даже тысячи, на то он и неисправный накопитель, чтобы работать нештатно.

Исходя из этого, необходимо тщательно продумать механизм реакции комплекса на возникающие таймауты. Ниже приведён внешний вид диалога настройки таймаутов:



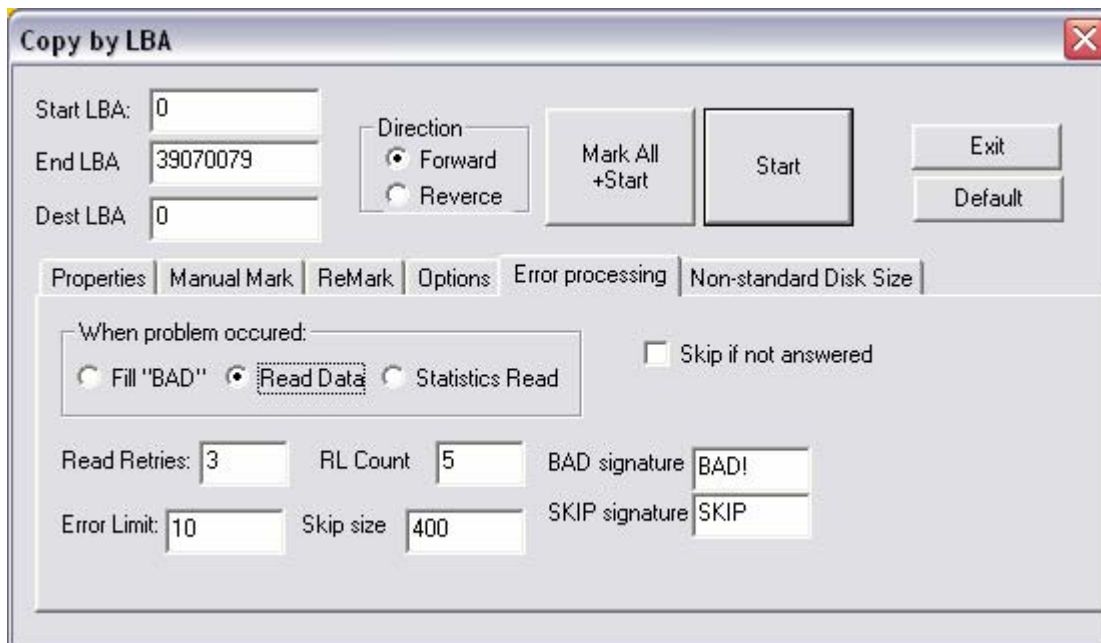
Процесс вывода накопителя из таймаута ведётся по шагам. Если во время копирования возник таймаут, указанный на шаге 1, то будет выполнено действие, расположенное в строке Шаг 1. Если при выполнении этого действия снова возник таймаут (например, накопитель не отреагировал на программный сброс), причём величина его достигла, указанной в строке Шаг 2, то выполняется действие, соответствующее данной строке. И так далее. Число строк взято с запасом. Можно, например, постепенно наращивать таймаут ожидания Reset или применять более ухищрённые приёмы. Если ничего не помогло, то процесс копирования будет прерван (но опять же, за счёт использования карты, Вы сможете принять меры и процесс пойдёт именно с точки останова, просто следует помнить, что недостаточно запустить процесс и оставить систему без присмотра на пару дней, возможно, уже через 2-3 часа потребуются Ваше вмешательство).

Флажок Single Access to SRC включает чрезвычайно важную функцию – **ОДНОКРАТНОГО ДОСТУПА К ИСТОЧНИКУ**. Если установлен этот флажок, то гарантируется, что к каждому сектору источника утилита обратится **НЕ БОЛЕЕ ОДНОГО РАЗА**. При этом, сектор будет автоматически скопирован на приёмник и уже дальнейшая работа будет вестись именно с копией данных. Этот флажок может быть сброшен только в одном случае – если необходимо рассмотреть файловую систему накопителя, а под рукой нет приёмника, достаточного объёма. Но и в этом случае, на помощь приходит файл-образ, размещённый в сжатом каталоге NTFS.

Наконец, группа флажков Read Look Ahead позволяет управлять упреждающим чтением накопителя. Если оно включено, процесс копирования идёт быстрее. Но если накопитель сильно неисправен, включение упреждающего чтения может приводить к сбоям в его работе.

Если флажок Set Features сброшен, не подаётся никаких команд управления упреждающим чтением накопителю. Если установлен – упреждающее чтение или принудительно включается или принудительно выключается. Что именно – зависит от состояния флажка Read Look Ahead.

## Вкладка Error Processing



На данной вкладке сосредоточены элементы настройки для обработки BAD блоков.

Поле ввода Read Retries задаёт число попыток чтения перед тем, как сектор считается сбойным. Иногда удаётся вычитать данные со второго, третьего и т.д. раза, так как накопитель подстраивает свои текущие параметры под текущую дорожку. Однако, не стоит увлекаться чрезмерным увеличением данного параметра, ведь обработка сбойного сектора – это и время, затраченное на «хрюканье» и полёт головки над сбойным участком (на котором может располагаться царапина), от чего головка может выйти из строя, не докопировав поверхность до конца.

Поле Error Limit определяет количество BAD-блоков подряд, после которых будет выполнен пропуск копирования. Длина Skip Size, в свою очередь, определяет длину пропуска. Этот механизм позволяет обойти массовые BAD-блоки, не тратя время на их скрупулёзное вычитывание. Авторам часто приходилось слышать: «А если на накопителе кругом BAD-ы, то вообще не будет считано ничего». Это утверждение в корне неверно по следующим причинам:

- Статистика говорит, что если BAD-блоки одиночные, они не кучкуются в стабильные группы по 10 секторов (именно такое количество взято по умолчанию)
- Даже если будет пропущено всё – никто не мешает перемаркировать карту (перевести сектора из состояния Skipped в Marked to copy) и провести ещё один проход копирования, теперь уже – только для них
- Пропуск идёт не строго на Skip Size секторов. Сначала выполняется прыжок на Skip Size, затем – начинается уменьшение номера сектора до тех пор, пока не встретится ошибка. Если шло 11 BAD-блоков подряд, то первые 10 уйдут на обнаружение плохой цепочки, а одиннадцатый (и только он!) уйдёт в пропуск. Правда, если идут 11 BAD-блоков, затем – 100 хороших и далее – всего один BAD, то всё-таки будет пропущена цепочка из 101 блока. Но повторяем, перемаркировка карты спасёт дело в любом случае.

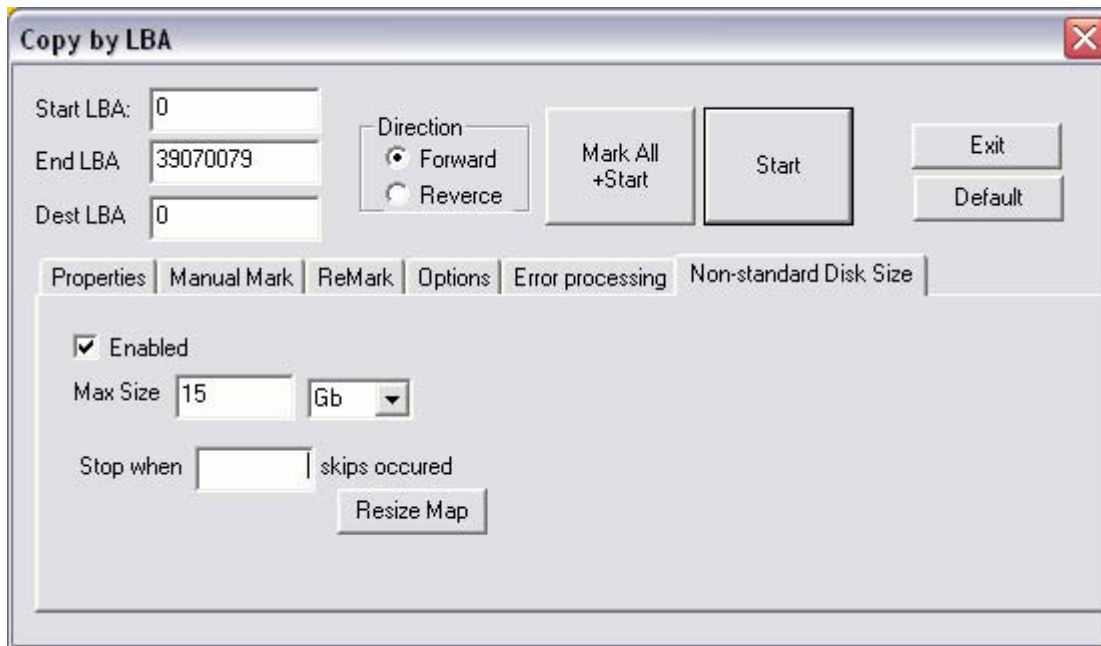
Группа When problem occurred задаёт реакцию системы на проблему. Многие накопители при ошибке всё равно отдают данные в буфере. Для них самая лучшая реакция – взять пусть частично разрушенные, но всё равно данные. Для этого следует выбрать радио кнопку Read Data (и именно она выбрана по умолчанию).

Бывает, что данные каждый раз несколько изменяются. В этом случае, можно составить статистику. Какие данные считываются чаще, те и являются наиболее достоверными. В этом случае, выбирайте радио кнопку Statistics. Поле RL Count при этом определяет число чтений, на основе которых будет составляться статистика. И, наконец, если выбрана Fill “BAD”, то никаких попыток спасения данных производиться не будет, а сектор будет заполнен четырьмя циклическими байтами (сами байты можно настраивать в поле BAD Signature, обычно это или BAD! Или пустая строка, соответствующая занулению сектора). Данный метод оставлен для обеспечения максимального быстродействия системы, однако его следует выбирать лишь в крайнем случае. Всегда после завершения работы с накопителем, можно заполнить сигнатурой любые сектора приёмника, основываясь не на сиюминутных сведениях («вот произошла ошибка, надо заполнить сектор сигнатурой, пока не забыли»), а на данных карты.

То же касается и сигнатуры SKIP. Можно метить сектора приёмника, пришедшиеся на пропущенные области, этой сигнатурой в процессе копирования (так как приёмник работает в режиме DMA, это практически не вызывает накладных расходов), а можно – заполнить всё уже после окончания копирования.

Если установлен флажок Skip if not answered, то после секторов, на которых возник таймаут, производится автоматический пропуск, не ожидая накопления нескольких BAD-блоков подряд. Его следует устанавливать на накопителях, где ошибки таймаута следуют одна за одной для того, чтобы при первом проходе не тратить время на множественные «хрюканья» головок, а отложить их анализ на второй и последующий проходы.

### Вкладка Non-standard Disk Size

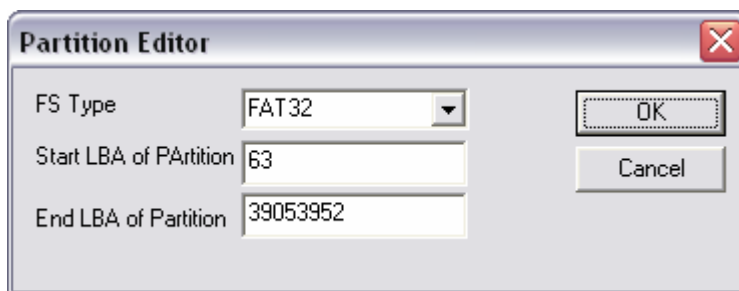


Данная вкладка позволяет обойти проблему некоторых накопителей, которые отдают неверный объём (обычно – нулевой), но при этом позволяют считывать сектора данных. В этом случае, войдите на данную вкладку, установите флажок Enabled, далее – размер в мегабайтах или гигабайтах и нажмите кнопку Resize Map. Размер лучше задавать с запасом сверху.

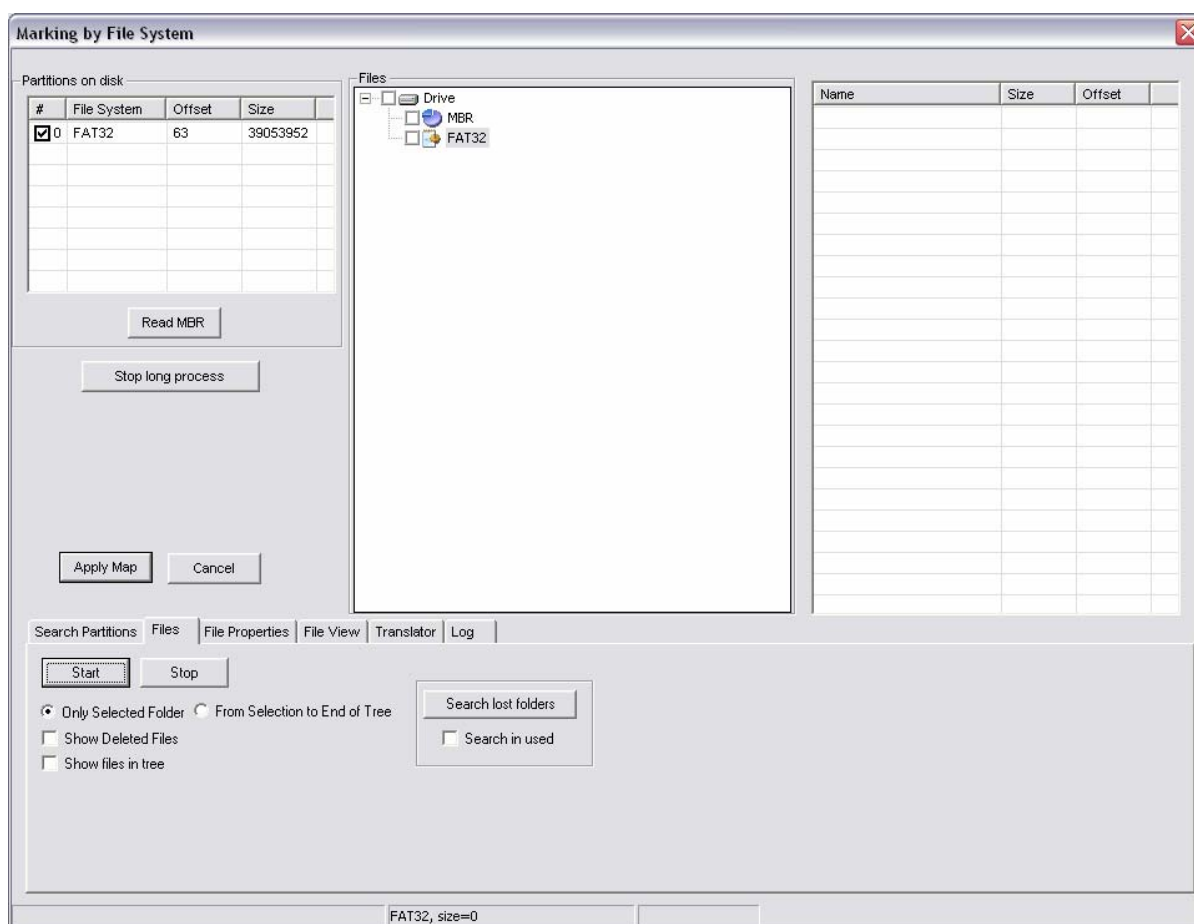
Чтобы исключить трату времени или даже стуки, которые вполне могут произойти при выходе головок за основное рабочее пространство диска, Вы можете установить предел количества пропусков подряд в поле Stop when XXX skips occurred. Таким образом, если встретится X BAD-блоков подряд, произойдёт пропуск. Если же произойдёт Y пропусков подряд, диск будет считаться за невозстанавливаемый и произойдёт останов процесса копирования. Дальнейшее решение о судьбе такого накопителя Вам следует принять самостоятельно.







Когда список разделов составлен, можно приступить к поиску файлов. Механизм для этого сосредоточен на вкладке Files. Кроме того, основное действие с файлами (сканирование ветки каталога) размещено в контекстном меню дерева каталогов.

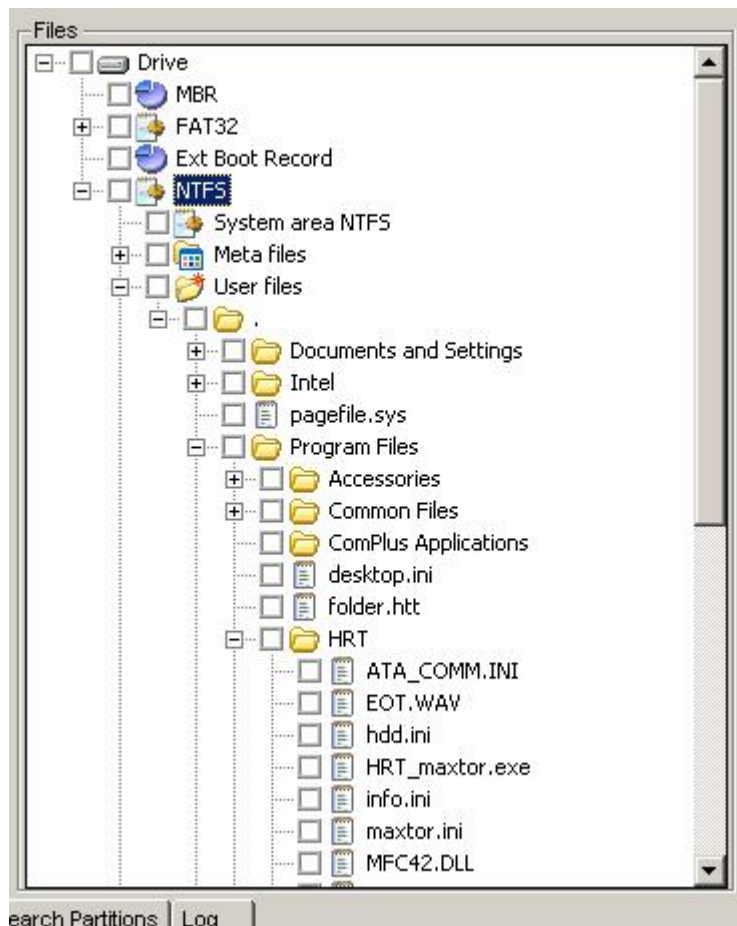


По умолчанию запрограммирован следующий случай: у Вас в руках накопитель, сканирование полной файловой системы которого занимает массу времени. ОС, пытаясь пробежать по всем ветвям дерева файлов, впадает в «задумчивость», которая длится часами. С другой стороны, Вам известны примерные пути, по которым лежат файлы, которые следует восстановить и тратить время на разбор совершенно не нужных веток – не хочется. Такой режим работы включает радио кнопка Only Selected Folder на вкладке Files.

Для построения первичного дерева файлов, выделите интересующие Вас разделы среди найденных ранее (если точнее, то снимите флажки с не интересующих) и нажмите кнопку Start на вкладке Files. Перед Вами появится первичное дерево. Теперь ставьте курсор на ветку User Files, нажимайте правую кнопку «Мыши» и снова жмите Scan. В появившемся дереве – вновь выбирайте интересующий Вас каталог и углубляйтесь в него. И так – пока не найдёте требуемые файлы. В дальнейшем – процесс можно повторить для

иных ветвей дерева. При таком способе разбора, Вы застрахованы от проблем, происходящих при обращении к каталогам, которые и не надо восстанавливать.

Если же Вам необходимо дерево без подобных мучений, установите радио кнопку From Selection to End Of Tree и поиск будет вестись от текущей точки до конца. В случае кнопки Scan – от корня каталога, в случае использования контекстного меню – от выделенного узла. Например, если узел – начало ветки, содержащей базы данных, то логично, что от этого места и до конца, будут нужны все файлы и каталоги и здесь можно перейти от штучного разбора к массовому. При этом, всё, что расположено вне этой ветки, всё равно не будет влиять.



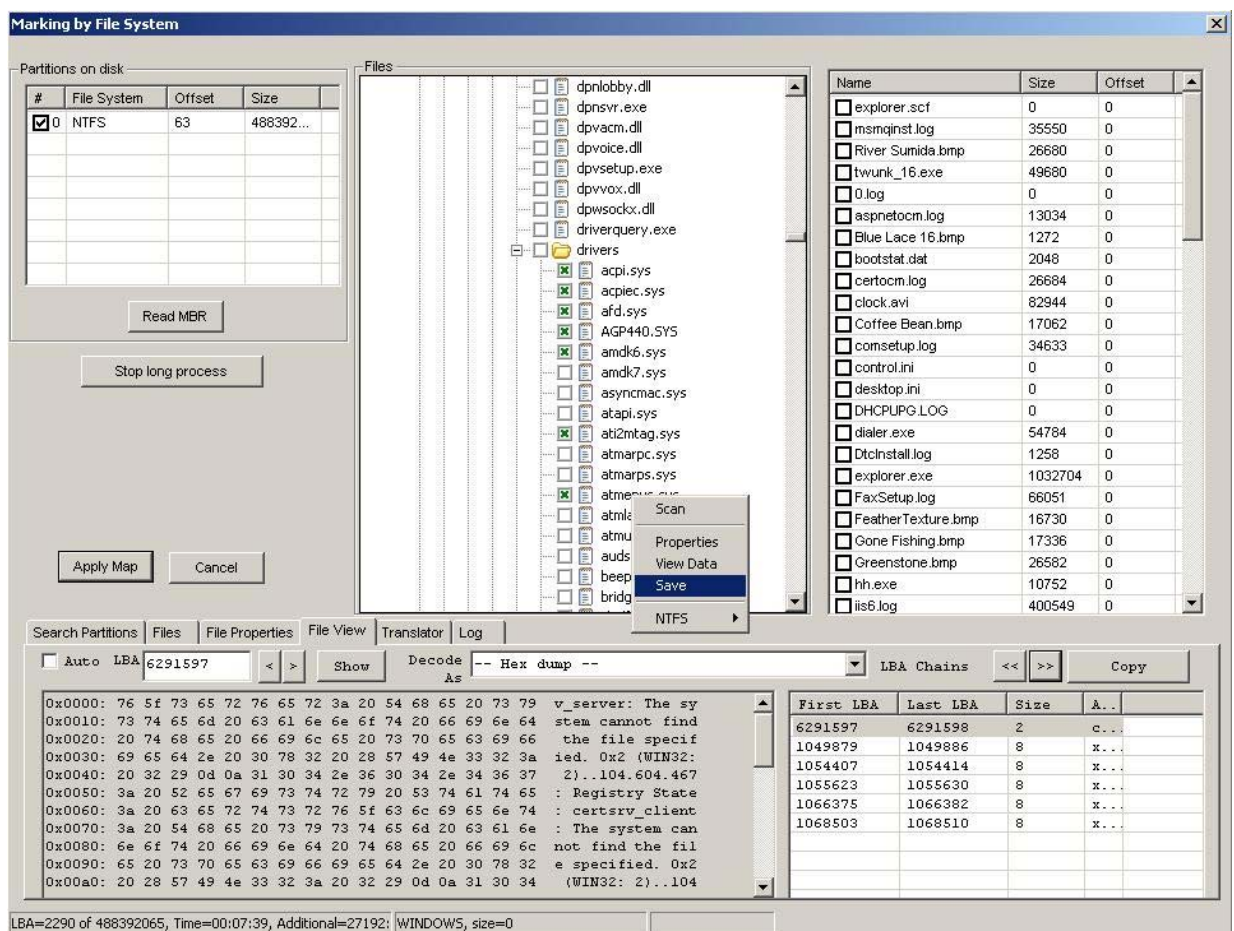
Бывают случаи, когда у накопителя не так много BAD-блоков, но зато разрушена файловая система. При этом каталоги на диске имеются, но потеряны ссылки на них из корневых узлов (например, в результате действия вирусов). Чтобы найти такие каталоги, можно воспользоваться кнопкой Search Lost Folders. При этом будет произведено сканирование всей поверхности (разумеется, относящейся к выбранным разделам) и все обнаруженные каталоги будут помещены в дерево каталогов.

Логично предположить, что в областях диска, занятых как каталогами, так и лежащими в них файлах, не будет найдено ни одного каталога. Это позволяет значительно сократить время сканирования сильно заполненного диска. Однако, на то диск и неисправный, чтобы у него всё было неправильно. Если будет обнаружена ссылка на файл, который накроет собой область, в которой на самом деле лежат не данные, а каталоги, то они будут пропущены. Чтобы избежать подобной ситуации, можно включить режим гарантированного полного сканирования поверхности, для чего следует установить флажок Search in used.

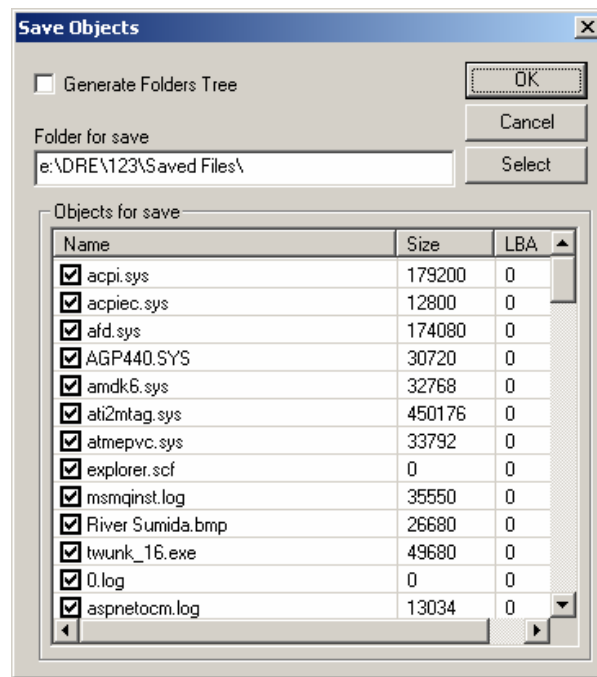
## Контекстное меню

Итак, дерево построено, что дальше? Дальше – можно попытаться проанализировать файлы на правильность. Для этого, выбрав в дереве файл, в контекстном меню выберите пункт Properties для просмотра свойств файла или пункт View Data для просмотра тела файла. Описание вкладки File View, автоматически открываемой при этом, будет описано ниже.

Пункт Save позволяет сохранить файлы, отмеченные флажками в каталог проекта, минуя операцию посекторного копирования. Такой режим может пригодиться, если требуется скопировать малое количество файлов (единицы или десятки, реже – сотни). После выбора Пункта, Вам будет предложено уточнить список сохраняемых файлов:



Отметили файлы и выбрали пункт Save



Убедились, что всё необходимое присутствует в списке

Если какие-либо файлы выбраны по ошибке, просто снимите с них выделение. Далее, установите способ сохранения – всё в один каталог или создавать дерево файлов, после чего нажимайте кнопку ОК. При этом начнётся попытка сохранения вычитывания и сохранения всех файлов.

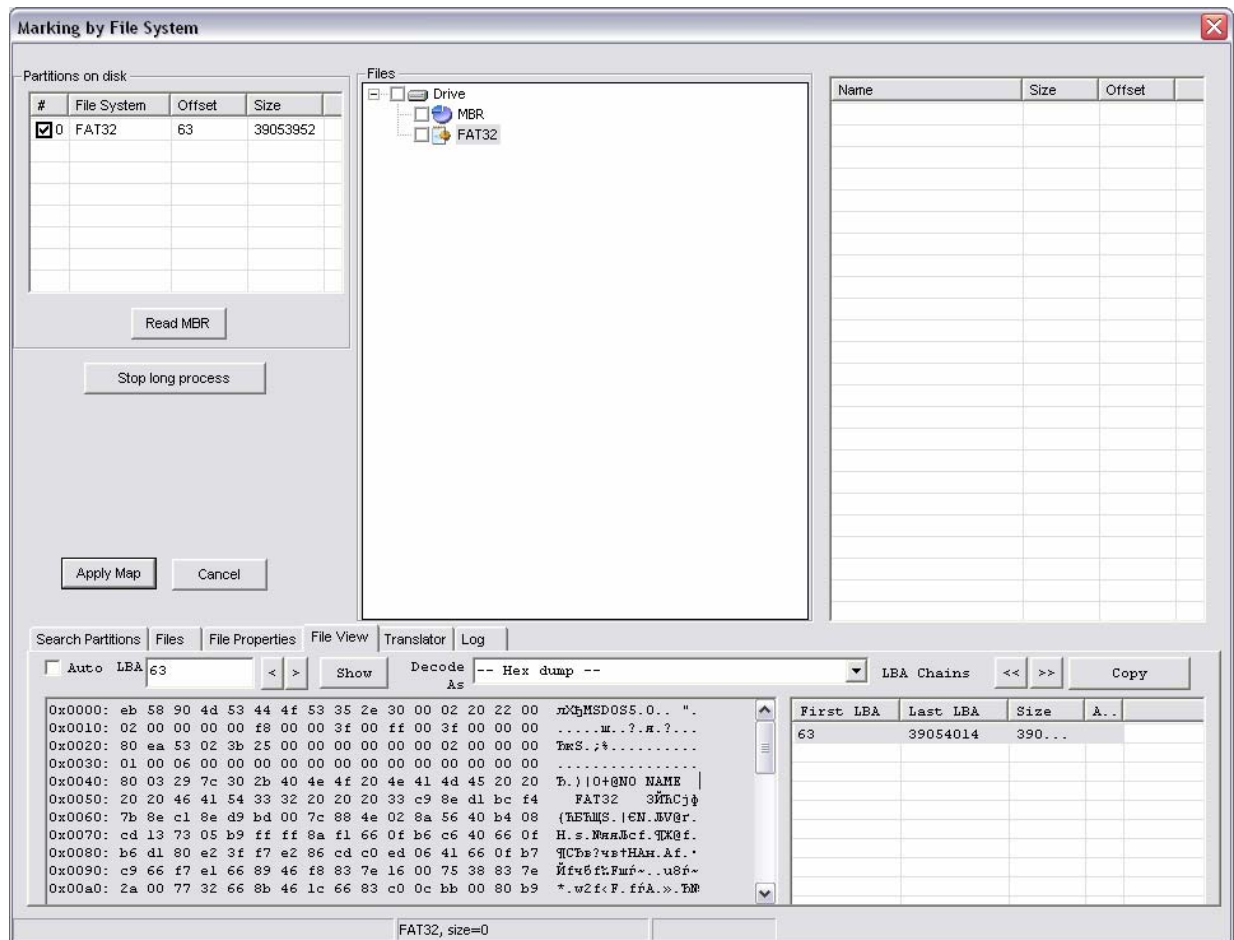
Кдинственный недостаток при таком методе – работа с накопителем, у которого разошлись копии FAT. При этом, во время построения дерева, в протокол будет помещено следующее сообщение:

Warinig!!! Fat copies are different, LBA %d and %d. Alternate LBA chain will be created

При занесении такого файла в карту, будут отмечены к выполнению сектора, относящиеся и к первому варианту и ко второму, чтобы при анализе образа при помощи узкоспециализированного ПО (той же R-STUDIO) был материал для манёвра. При сохранении же, будет сохранён только первый вариант.

Если продолжать рассказ о контекстном меню, то стоит остановиться на пункте NTFS->Mark Used. Дело в том, что файловая система NTFS содержит замечательную вещь – битовую карту занятого пространства. Если сектора с битовой картой не повреждены, то достоверную карту копирования всего пространства, содержащего файлы, можно построить на её основе. Выберите данный пункт и у Вас будет готова карта для посекторного копирования с гарантией, что в неё включено всё, что необходимо (правда, и файл pagefile.sys – тоже, а на накопителях IBM он часто является основным тормозом, так как почти весь состоит из BAD-блоков и имеет огромную длину). Достоинство данного метода отметки состоит в его скорости (не требуется читать большое количество секторов, описывающих файловую систему), надёжности (чем меньше секторов, тем меньше BAD блоков, а значит – меньше шансов наткнуться на потерянные данные) и простоте (меньше обработки – меньше алгоритмических ошибок).

## Вкладка File View

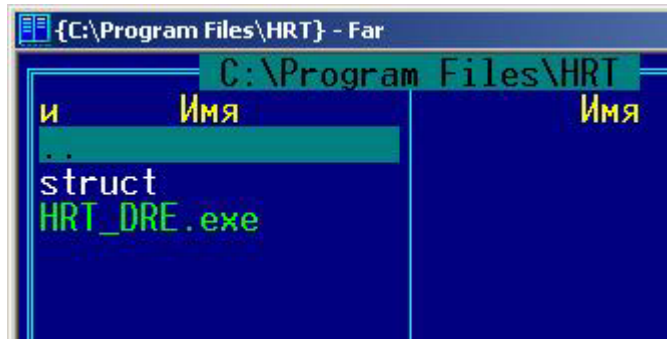


Поле LBA содержит текущий отображаемый LBA. По умолчанию, оно заполняется первым LBA, занимаемым файлом. При необходимости (например, с целью поиска данных или сдвига транслятора), Вы можете ввести любое значение в данное поле, после чего нажать кнопку Show. При этом, будет отображён сектор, LBA которого был введён. Переход на соседние сектора осуществляется при помощи кнопок "<" и ">". При этом, нажимать кнопку Show не требуется.

Поле Decode As содержит метод декодирования данных. По умолчанию доступны методы:

- Hex dump –
- ASCII Text –
- UNICODE Text –

Кроме того, Вы можете задавать свои структуры для декодирования. Структуры должны располагаться в каталоге struct, расположенном на том же уровне, что и файл hrt\_dre. Каждый файл, расположенный в этом каталоге, считается структурой.



Допустимые ключевые слова в структурах:

Ключевое слово	Комментарий
BYTE	Байт данных (8 разрядов)
BYTE []	Массив восьмиразрядных байтов
WORD	Слово данных (16 разрядов)
DWORD	Двойное слово данных (32 разряда)
INT64	64-разрядное целое, со знаком
UINT64	64 разрядное целое, без знака
CHAR []	Массив ASCII-символов
#repeat	Сразу за текущей структурой следует ещё одна

Пример структуры (файл FAT16 Directory.dat):

```

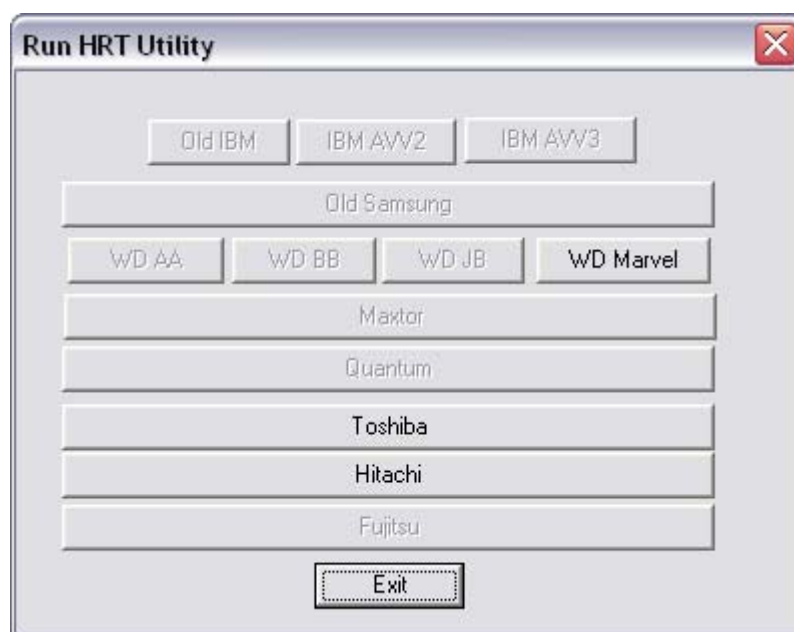
CHAR      Name[8];
CHAR      Ext[3];
BYTE      Attrib;
BYTE      Reserved[10];
WORD      CreateTime;
WORD      CreateDate;
WORD      Cluster;
DWORD     Size;
#repeat

```

Все структуры пользователя добавляются к данному списку и у Вас имеется возможность рассматривать данные в виде структур.

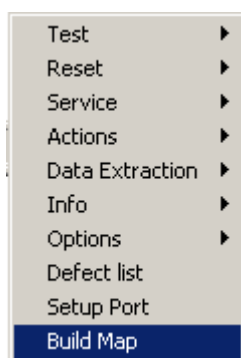
## МАРКИРОВКА КАРТЫ НА ОСНОВЕ КАРТЫ ГОЛОВОК

Как уже отмечалось выше, очень часто необходимо производить раздельное копирование данных по головкам. Для этого, в свою очередь, необходим запуск специализированных утилит комплекса HRT, так как именно они знают механизмы трансляции для тех или иных накопителей. Для запуска утилиты, нажмите Run HRT Utility на вкладке Properties главного окна.

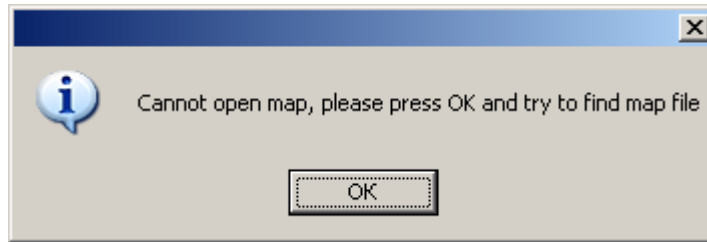


Утилита автоматически проверяет активные лицензии в карте HRT и даёт возможность выбрать только активированные утилиты комплекса.

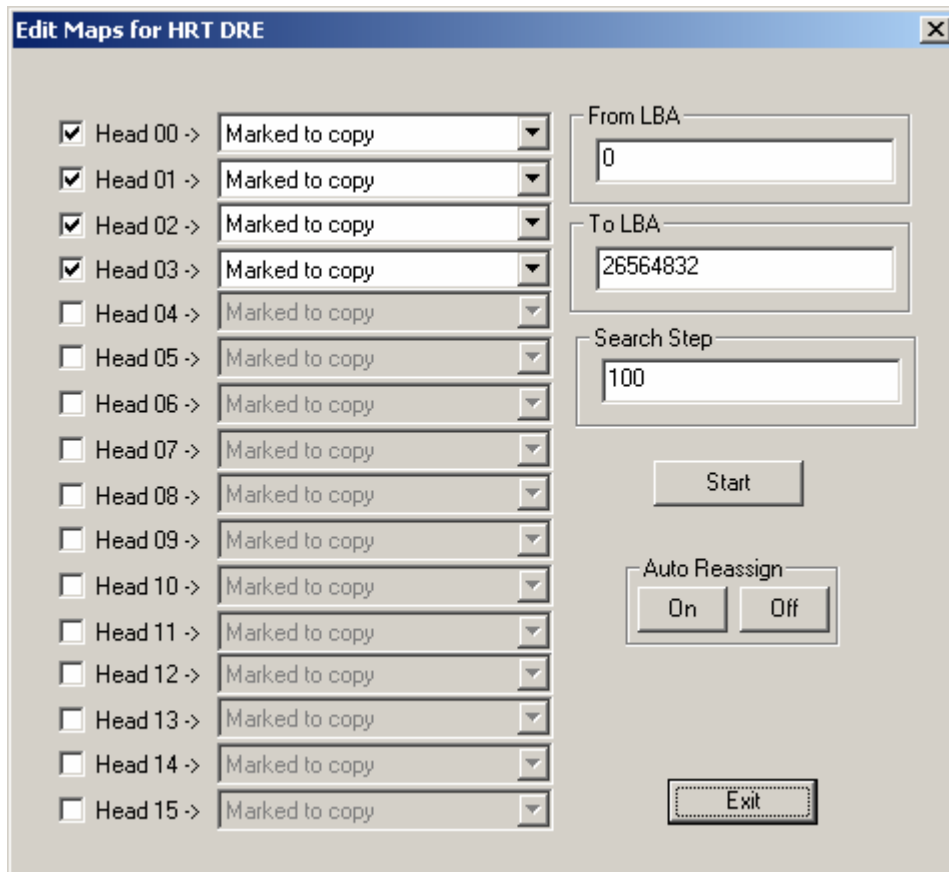
В отличие от специализированной утилиты HRT, запущенной автономно, в данном случае в контекстном меню окна накопителя любой из утилит появляется дополнительный пункт Build Map:



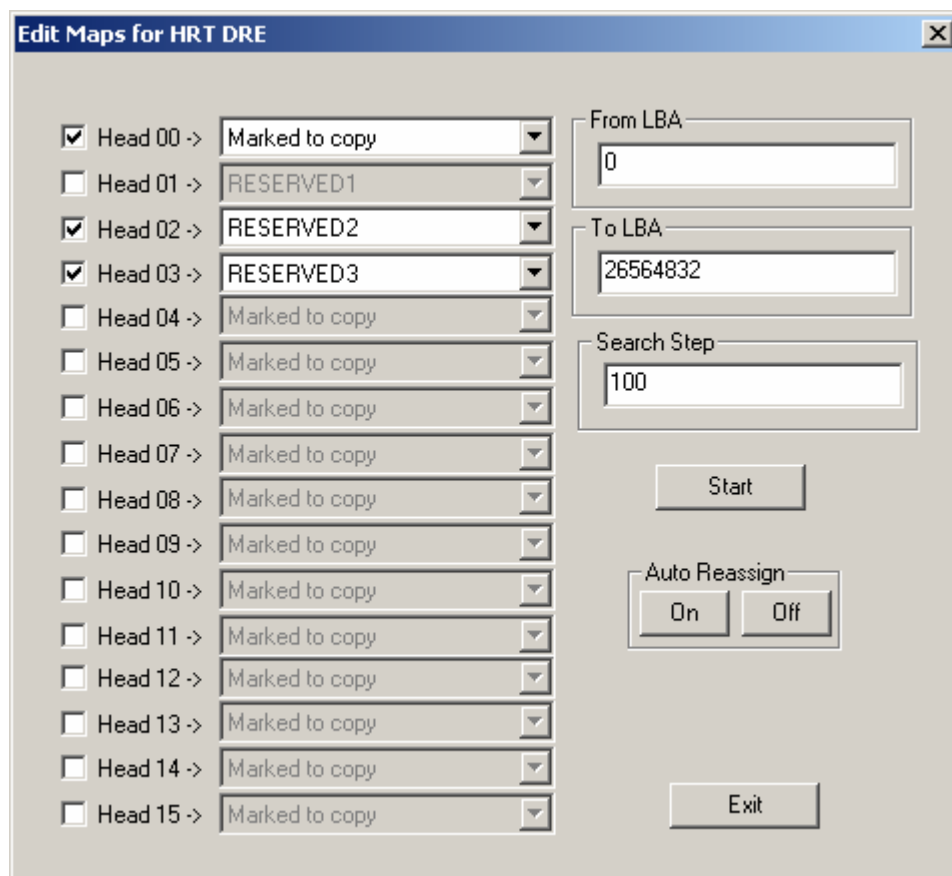
При его нажатии, автоматически загружается последняя автосохранённая карта утилиты HRT DRE. Если карта по той или иной причине отсутствует, будет выдано сообщение:



И Вы сможете выбрать любой другой файл с картой (например, сохранённый вручную). Далее, будет выдан следующий диалог:



Флажок около номера головки определяет, участвует она в перемаркировке или нет. Если флажок снят, то элементы карты, соответствующие таким головкам, сохраняют своё предыдущее состояние. Если установлен, то все сектора, пришедшиеся на эту головку, получают состояние, указанное справа от флажка. Например:



Здесь головка 0 отмечена непосредственно на копирование, головка 1 не будет изменена, головка 2 будет отмечена состоянием Reserved 1, головка 3 – состоянием Reserved 2. Напомним, что состояния Reserved введены в связи с тем, что процесс трансляции – достаточно медленный и нецелесообразно проводить его ради каждой головки. Поэтому процесс стоит проводить комплексно, за один проход. А как разделять головки? Именно для решения данной задачи и введены состояния, в которые сектор не может перейти при работе программы. Когда закончится проход копирования очередной головки, следует перемаркировать состояние Reserved X в Marked to copy и начать очередной проход.

Поля From LBA и To LBA задают границы трансляции. Они позволяют сократить время процесса, если требуется произвести отметку только части карты на большом накопителе. На поле Search Step следует остановиться подробнее.

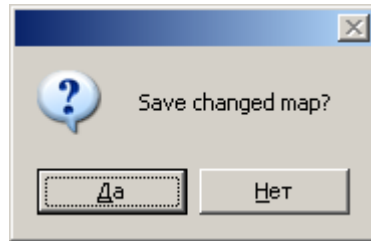
Трансляция каждого сектора в общем случае, будет вестись чрезвычайно долго. Поэтому программа транслирует сектор LBA и LBA + Step. Если и там и там номер головки совпадает, начинается анализ следующего промежутка, размером Step. Если же номер головки изменился, то на рассматриваемом диапазоне имеется точка перехода между головками. Эта точка ищется методом половинного деления.

Таким образом, чем больше параметра Search Step, тем быстрее завершится процесс трансляции. Но с другой стороны, если параметр задан слишком большой, то на рассматриваемом диапазоне может произойти не одна, а две (или даже более) смены головок, а утилита сочтёт это за одну смену и отработает некорректно.

Существуют накопители, переключающие головки по окончании каждого цилиндра (обычно это старые накопители – Quantum, Fujitsu и т.п.). Для них данный параметр должен быть не более половины минимального SPT (именно из этих

соображений выбрано значение по умолчанию 100). Большинство же современных накопителей осуществляют трансляцию «кольцами». При этом по мере роста LBA переключается номер цилиндра, а номер головки – остаётся прежним. И лишь по достижении определённого значения (например, на каждом 128-м цилиндре) происходит смена головки. Для таких накопителей можно задавать значения в районе 1000 и даже более. Характер трансляции можно выявить, пользуясь диалогом Issue Translator, имеющимся в большинстве утилит комплекса HRT.

Когда отметка закончена, закройте диалог. Вам будет задан вопрос:



Если Вы считаете, что процесс завершён успешно – согласитесь с сохранением и выберите файл, в который следует сохранить изменения (обычно это тот же файл, что и исходный, но с целью безопасности, исходный файл можно оставить, а сохраниться – под другим именем).

Если впоследствии загрузить такую карту в утилиту HRT DRE, то будет видно, что она перемаркирована в соответствии с распределением данных по головкам:

