## Архитектура Windows. Состав и функции основных компонентов. (Стр 20)

### Основные системные каталоги

Каталог Program Files содержит папки с программами, установленными как в процессе инсталляции Windows, так и пользователями компьютера. Для каждой программы отведена отдельная папка. Если после удаления программы с компьютера, ее папка остается, то рекомендуется вручную удалить ее из каталога Program Files. Иначе при частой инсталляции и удалении программ в данной папке может накопиться изрядное количество мусора.

Папка ProgramData является скрытой, служит хранилищем общих данных, например, настроек различных приложений. К информации, сохраненной в ней, могут обращаться все пользователи, независимо от наличия или отсутствия у них прав администратора. Данную папку удобно использовать, например, чтобы хранить в ней шаблоны для программ по редактированию фотографий, видео, общие документы и так далее.

Recovery - Папка, в которой хранится образ, необходимый для входа в среду восстановления.

Папка System Volume Information по умолчанию является скрытой. Программа восстановления системы Windows создает в ней копии реестра. Т.е. здесь хранятся каталоги точек восстановления системы, отсортированные по дате создания. Windows не позволяет просматривать содержимое каталога System Volume Information обычным способом. Размер данной папки зависит от объема дискового пространства, резервируемого системой под точки восстановления.

Каталог Windows – самая важная папка, размещающаяся на активном разделе жесткого диска. В ней хранятся все исполняемые файлы операционной системы, драйвера, модули и пр.

Windows.old - Папка со старой операционной системой. Возникает на жестком диске, если установка Windows была осуществлена без форматирования. Для подавляющего большинства пользователей она не нужна, а поэтому может быть удалена.

Users - Папка для хранения профилей пользователей. По умолчанию размещается в корне диска с операционной системой. Содержит ряд стандартных папок, а также по одной папке на каждую учетную запись. В них находятся «пользовательские» подпапки, включая рабочий стол (Desktop), документы (Documents), картинки (Pictures), избранное (Favorites) и пр. Другие подпапки используются для хранения служебной информации различных приложений, браузеров и пр., они скрытые, и изменять их содержимое вручную не рекомендуется.

$Recycle.Bin - Данная папка является «Корзиной» Windows. Именно в нее перемещается вся удаляемая информация. Папка скрытая, однако, удалять можно как ее саму, так и ее содержимое. По сути, это будет равносильно очистке «Корзины» или ее части. На каждом разделе жесткого диска есть своя папка $Recycle.Bin, в которой хранятся файлы, удаленные именно с него.

C:\Windows\System32 – каталог содержит данные операционной системы, драйверов, данные реестра, утилиты для работы с системой.

C:\Windows\Temp – каталог, содержащий временные файлы приложений.

C:\Users\ <Имя пользователя> \AppData – папка содержит данные, относящиеся к учетной записи. Обычно включает в себя параметры конфигурации, которые относятся к вашему экземпляру установленной программы.

### Основные системные файлы

Файлы реестра Windows:

* **C:\Documents and Settings\Имя Пользователя\ntuser.dat** - является кустом реестра Windows HKEY\_USERS/SID
* **C:\Windows \system32\config\SAM** - содержит раздел реестра HKLM\SAM
* **C:\Windows\system32\config\SECURITY** - содержит раздел реестра HKLM\SECURITY
* **C:\Windows\system32\config\software** - содержит раздел реестра HKLM\Software
* **C:\Windows\system32\config\system** - содержит раздел реестра HKLM\System

Другие системные файлы:

* **C:\Windows\system32\config\AppEvent.Evt** - является журналом событий приложений, используемым реестром операционной системой Windows.
* **C:\Windows\system32\drivers\etc\hosts** - предназначен для сопоставления IP-адресов и имен компьютеров.
* **C:\Windows\system32\mydocs.dll** - отвечает за пользовательский интерфейс папки "Мои документы"
* **C:\Windows\system32\config\SecEvent.Evt** - является журналом событий безопасности, используемым реестром Windows
* **C:\Windows\WindowsUpdate.log** - Файл, содержащий историю автоматического обновления.
* **C:\Windows\ ntbtlog.txt** – Файл протоколирования загрузки ОС.

## Базовые свойства файловой структуры NTFS

### Операции с томом, каталогами и файлами

#### Создание hardlink, symlink, junction point (Стр 35)

**Жёсткие ссылки**

Жёсткая ссылка (HardLink) - это просто ещё одна запись в папке для данного файла.

Когда создаётся жёсткая ссылка, сам файл не копируется физически, а только появляется под ещё одним именем или в ещё одном месте, а его старые имя и местонахождение остаются нетронутыми. С этого момента жёсткая ссылка неотличима от первоначальной записи в папке. Единственное отличие - то, что для жёсткой ссылка не создаётся короткое имя файла, поэтому из ДОС-программ она не видна.

Когда меняется размер или дата файла, все соответствующие записи в папках обновляются автоматически. При удалении файла он не удаляется физически до тех пор, пока все жёсткие ссылки, указывающие на него, не будут удалены. Порядок их удаления значения не имеет. При удалении жёсткой ссылки в корзину количество ссылок у файла сохраняется.

Создавать жёсткие ссылки можно только на том же диске, на котором расположен исходный файл.

**Точки повторной обработки**

Эта технология позволяет отображать любые локальные папки на любые другие локальные папки. Например, если папка D:\JunctionPoint указывает на C:\Windows\system32 в качестве своей цели, то программа, обращающаяся к D:\ JunctionPoint\ drivers, будет на самом деле обращаться к на C:\Windows\system32\drivers. В отличие от жёстких ссылок, связи папок не обязательно должны располагаться на том же диске.

**Символические ссылки**

Символические ссылки (Symlink) на NTFS поддерживаются, начиная с Windows Vista. Они представляют собой улучшенный вариант связей каталогов - символические ссылки могут указывать как на папки, так и на файлы, как на локальные, так и на сетевые, при этом поддерживаются относительные пути.

**Mklink**

Для создания ссылок можно использовать программу mklink.exe в командной строке cmd. Формат командной строки MKLINK:

MKLINK [[/D] | [/H] | [/J]] Ссылка Назначение

* /D - Создание Symbolic Links на каталог. По умолчанию создается символическая ссылка на файл.
* /H - Создание Hard Links для файла.
* /J - Создание Junction Points для каталога.
* Ссылка - Имя новой символической ссылки.
* Назначение - Путь (относительный или абсолютный), на который ссылается создаваемая ссылка.

Создадим Жесткую ссылку (Рис 1), Точку повторной обработки (Рис 2), и Символическую ссылку (Рис 3).



Рис 1. Создание жесткой ссылки



Рис 2. Создание соединения – точки повторной обработки



Рис 3. Создание символической ссылки

Отобразим содержимое текущего каталога командой dir (Рис 4).

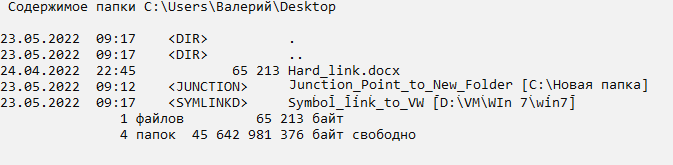
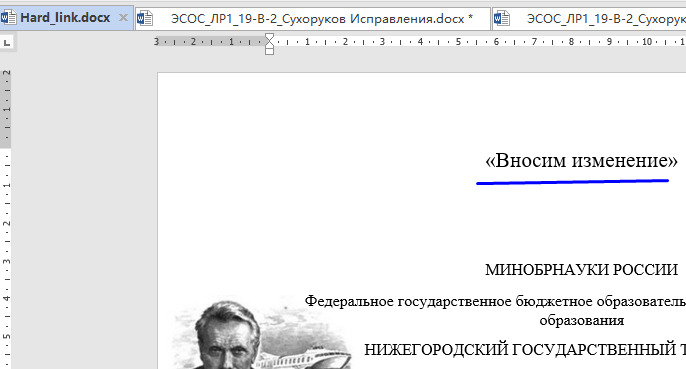


Рис 4. Содержимое каталога C:\Users\Валерий\Desktop

Проведем «эксперименты» с созданными объектами.

* Внесем изменение в файл Hard\_link.docx (Рис 5) и проверим, что исходный файл изменился (Рис 6).

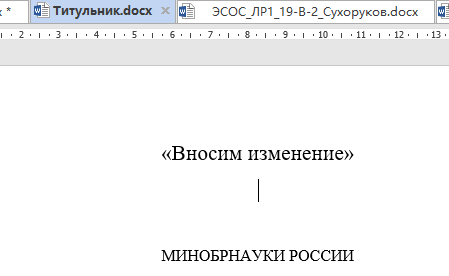
Рис 5. Внесение изменений в файл Hard\_Link.docx

Рис 6. Отображение изменений в исходном файле

**Вывод:** Изменение данных в жесткой ссылке изменяет исходный файл.

* Переместим файл C:\Study\ Титульник.docx командой move на другой диск (Рис 7). Внесем новые изменения (Рис 8), и посмотрим изменится ли исходный файл (Рис 9).



Рис 7. Перемещение файла командой move

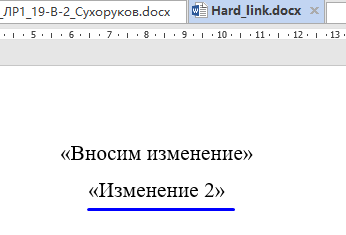


Рис 8. Изменение файла Hard\_Link.docx

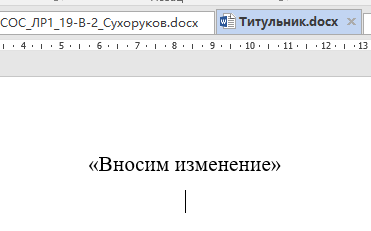


Рис 9. Содержимое файла после второго изменения.

**Вывод:** При перемещении файла на другой диск объект жесткой ссылки не может получить к нему (исходному файлу) доступ.

* Проверим функциональность Точек повторной обработки. Откроем созданный объект Junction Point и перейдем в каталог «Тестовая папка» (Рис 10).



Рис 10.

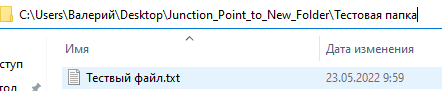
Адресная строка сформировалась так, как будто мы находимся в подпапке каталога Junction\_Point\_to\_New\_Folder. Создадим в этой директории текстовый файл (Рис 11) и посмотрим, отобразится ли он в исходной папке (Рис 12).

Рис 11.

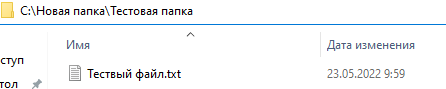


Рис 12.

В исходном каталоге появился созданный файл

**Вывод:** Изменения внутри объекта Junction Point отображаются в исходном каталоге.

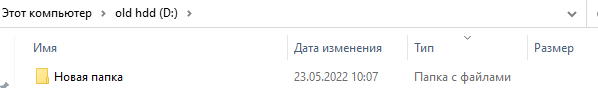
* Переместим исходный каталог на другой диск (Рис 13) и обратимся к точке повторной обработки. (Рис 14)

Рис 13. Перемещение каталога

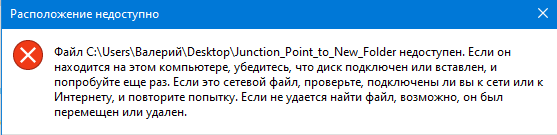


Рис 14.

Точка повторной обработки продолжает ссылаться на каталог C:\Новая папка, что можно увидеть командой dir (Рис 15). Получить доступ к каталогу по новому пути не получается.



Рис 15.

**Вывод:** При перемещении каталога подступ к нему с помощью Junction Points утрачивается.

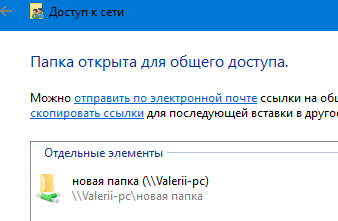
* Предоставим общий доступ к папке «Новая папка» (Рис 16) и попробуем создать для неё Junction Point (Рис 17) и Symlink (Рис 18).

Рис 16. Предоставление доступа к папке по сети.



Рис 17. Попытка создания точки повторной обработки на сетевую папку



Рис 18. Создание символической ссылки на сетевую папку

При создании точки повторной обработки произошла ошибка, символическая ссылка создалась успешно. Выполним команду dir (Рис 19) и просмотрим адрес символической ссылки.

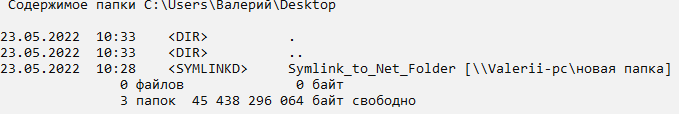


Рис 19.

У символической ссылки адрес сетевого ресурса.

**Вывод:** Символические ссылки можно создавать на сетевые ресурсы, точки повторной обработки – только на локальные.

#### Использование команды fsutil (Стр 36)

Fsutil является служебной программой командной строки, которая используется для выполнения связанных задач файловых систем FAT и NTFS, таких как управление точками повторной обработки, управление разреженными файлами, отключение тома или расширение тома.

* Получение информации о файловой системе в целом – fsutil fsinfo. Запросим название всех подключённых дисков (Рис 20), и тип дисков D: и E: (Рис 21).



Рис 20. Название подключенных логических дисков

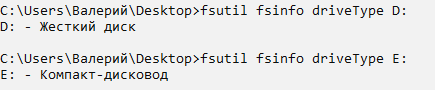


Рис 21. Типы дисков D и E

* Получение информации о томах – fsutil volume. Отобразим информацию о свободном месте на диске С (Рис 22).

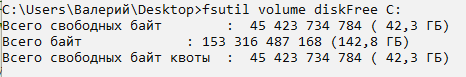


Рис 22. Количество свободного места на диске С:

Команда отображает Объем диска, свободную часть и объем диска, который может быть выделен квотой. Дисковая квота - способ ограничения размера дискового пространства на томе для использования каждым пользователем. Дисковые квоты позволяют заранее отследить и ограничить использование дискового пространства.

* Получение информации о файлах– fsutil file. Отобразим ID файла ЭСОС\_ЛР1\_19-В-2\_Сухоруков.docx (Рис 23).



Рис 23. Получение ID файла.

Результатом выполнения является уникальный номер файла в файловой системе.

### Примеры использования chkdsk, diskpart, vssadmin (Стр 38)

* **Chkdsk** проверяет файловую систему и метаданные файловой системы тома на наличие логических и физических ошибок.

Выполним поиск ошибок на диске I и исправим их – chkdsk I: /R /F. Параметр “/F” указывает CHKDSK выполнить исправление найденных ошибок; “/R” указывает программе обнаружить повреждённые секторы на диске и восстановить информацию, которую возможно прочитать.

Диск проверяется в 4 этапа:

* На первом этапе происходит «Проверка базовой структуры файловой системы» - файлы с записями в MFT (главная файловая таблица) сравниваются и проверяются.
* На втором этапе происходит «Проверка связей имен файлов» - проверяются директории и структура. Они проверяются на соответствие размеру файла и временной метки информации.
* На третьем этапе происходит «Проверка дескрипторов безопасности» - файлы и папки проверяются на наличие дескрипторов безопасности.
* На четвертом этапе происходит «Поиск поврежденных кластеров в данных пользовательских файлов» - Секторы кластера, включая данные, проверяются на возможность использования.
* На пятом этапе происходит «Поиск поврежденных и свободных кластеров» - секторы кластера проверяются на отсутствие данных.

В результате выполнения выводится информация об обнаружении ошибок, их исправлении и информации о диске (Рис 24).

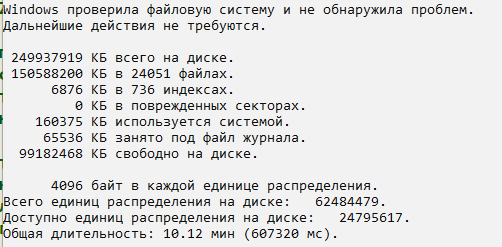
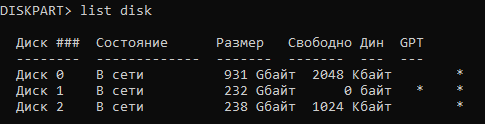


Рис 24. Результат проверки системы

* **DiskPart** помогает управлять дисками компьютера (дисками, разделами, томами или виртуальными жесткими дисками).

Отобразим список подключенных дисков – **list disk** (Рис 29). Результатом выполнения команды является пронумерованный список дисков.



Выберем диск 0 (**select disk 1)** и получим информацию о нем (**detail disk)** (Рис 30).

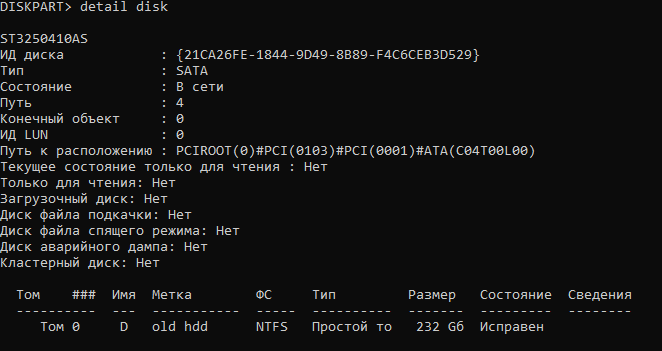


Рис 30. Информация о диске

В описании диска находится его ID, тип, состояние, список томов, наличие на диске файла подкачки и другая информация.

Назначим тому с номером 0 имя “S”. Для этого выберем его (**select volume 0**) и используем команду **assign letter=S** (Рис 31).

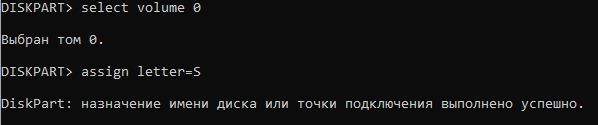


Рис 31. Изменение имени диска

* Утилита **Vssadmin** предназначена для администрирования в командной строке службы теневого копирования томов.

Выведем список существующих теневых копий томов - vssadmin List Shadows (Рис 32).

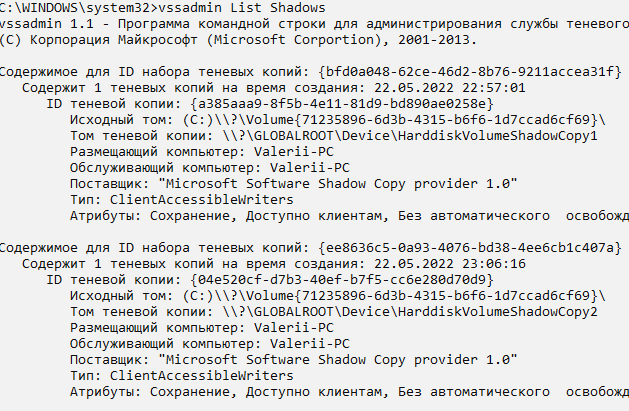


Рис 32. Список существующих теневых копий

На текущий момент в системе создано две теневых копии. Удалим более старую копию диска С: командой vssadmin Delete Shadows /For=C: /Oldest (Рис 33).

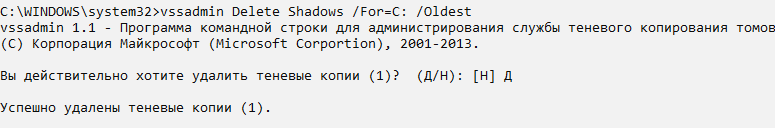


Рис 33. Удаление теневой копии

Утилита запросила подтверждения. После соглашения произошло успешное удаление.

## Механизмы обмена данными между приложениями

### Связывание и встраивание объекта в документ (Стр 45)

Создадим новый документ для проверки функционирования связывания и встраивания. С помощью Excel создадим график функции, который будем встраивать и связывать. (Рис 34)

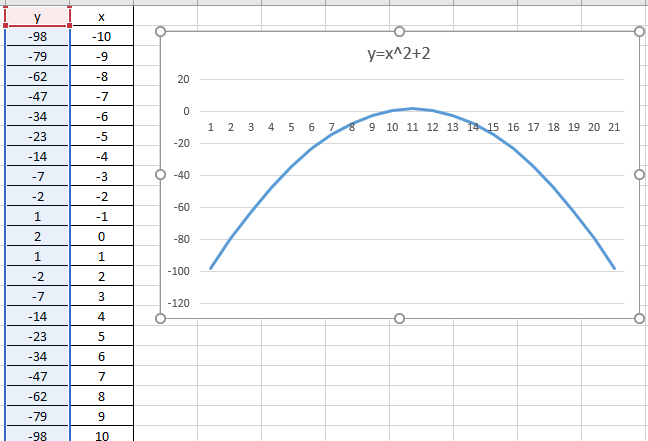


Рис 34.

В документе выберем меню «Специальная вставка» и вставим двумя способами вставим (Рис 35) и свяжем (Рис 36).

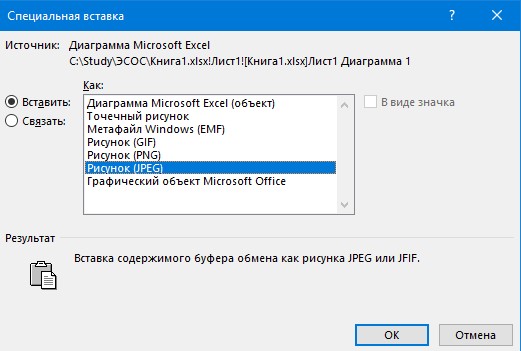


Рис 35. Вставка объекта

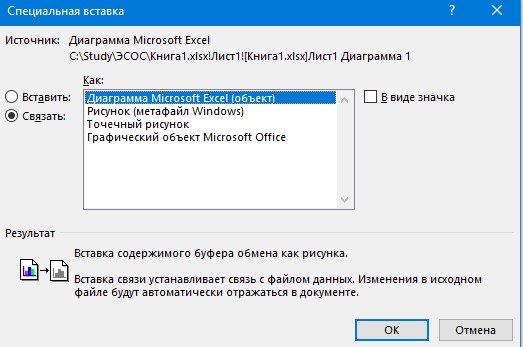


Рис 36. Связывание объекта

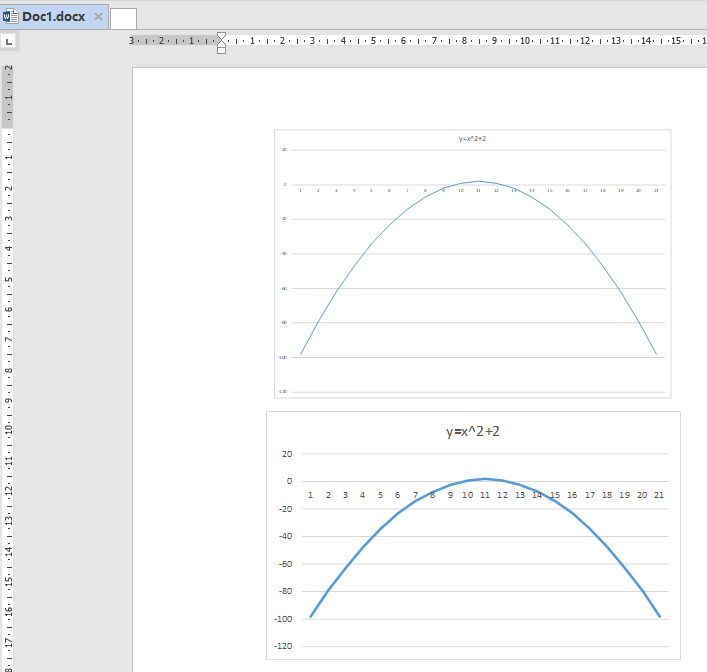


Рис 37. Документ со встроенным и связанным объектами

Изменим данные в файле Excel (Рис 38) и откроем документ заново для обновления данных связи.

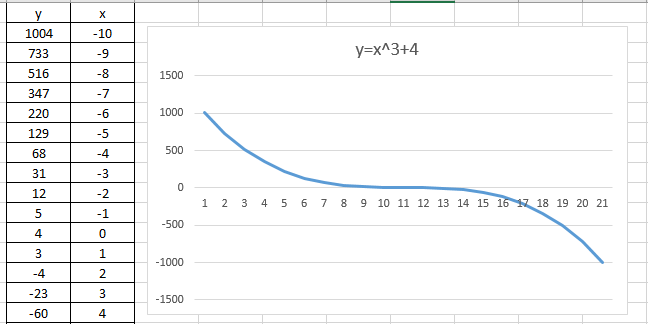


Рис 38. Обновленные данные Excel файла

После перезагрузки документа связанный элемент изменился, встроенный остался таким, какой был (Рис 39).

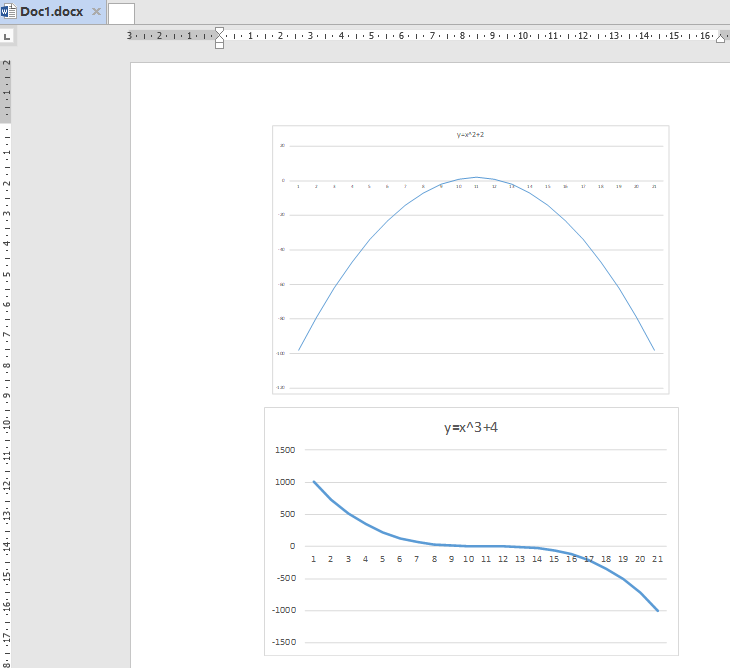


Рис 39. Измененный документ

Перенесем текстовый документ на USB-накопитель, Excel файл оставим в исходной папке. Откроем документ на другом компьютере. Связанный объект не смог построить ссылку для доступа к исходному и не отображается (Рис 40).

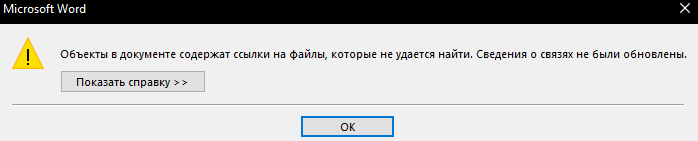


Рис 40. Невозможность отображения связанного элемента при отсутствии исходного

*Вывод по сравнению встраивания и связывания*

Встраивание файла позволяет редактировать встроенный объект внутри файла. Встроенный объект не нужно переносить вместе с основным документом, т.к. он является его частью. Связывание объекта и файла позволяет автоматически обновлять документы, которые связаны с файлом. Это удобно, если один объект связывается со множеством документов т.к. его изменение приведёт к изменению сразу всех документов. Минусом связывания является необходимость переносить его вместе с основным файлом.