Reading/Writing a MS Word file in PHP

Asked 16 years, 2 months ago Modified 5 years, 5 months ago Viewed 141k times

Part of PHP Collective

33

Is it possible to read and write Word (2003 and 2007) files in PHP without using a COM object? I know that I can:

```
$file = fopen('c:\file.doc', 'w+');
fwrite($file, $text);
fclose();
```

but Word will read it as an HTML file not a native .doc file.

PHP php ms-word read-write

Share Improve this question Follow

asked Oct 9, 2008 at 18:09

UnkwnTech

90.7k ● 65 ● 191 ● 234

I find it HIGHLY unlikely that you could achieve this without using COM. – Peter Bailey Oct 9, 2008 at 18:32

16 Answers

Sorted by: Highest score (default)

29

Reading binary Word documents would involve creating a parser according to the published file format specifications for the DOC format. I think this is no real feasible solution.

You could use the Microsoft Office XML formats for reading and writing Word files - this is compatible with the 2003 and 2007 version of Word. For reading you have to ensure that the Word documents are saved in the correct format (it's called Word 2003 XML-Document in Word 2007). For writing you just have to follow the openly available XML schema. I've never used this format for writing out Office documents from PHP, but I'm using it for reading in an Excel worksheet (naturally saved as XML-Spreadsheet 2003) and displaying its data on a web page. As the files are plainly XML data it's no problem to navigate within and figure out how to extract the data you need.

The other option - a Word 2007 only option (if the OpenXML file formats are not installed in your Word 2003) - would be to ressort to OpenXML. As databyss pointed out here the DOCX file format is just a ZIP archive with XML files included. There are a lot of resources on MSDN regarding the OpenXML file format, so you should be able to figure out how to read the data you want. Writing will be much more complicated I think - it just depends on how much time you'll invest.

Perhaps you can have a look at PHPExcel which is a library able to write to Excel 2007 files and read from Excel 2007 files using the OpenXML standard. You could get an idea of the work involved when trying to read and write OpenXML Word documents.

Share

Improve this answer

Follow

edited May 23, 2017 at 12:25

Community Bot

1 • 1

answered Nov 5, 2008 at 13:04

Stefan Gehrig

83.5k • 24 • 161 • 192

1 It seems the ppl at PHPExcel have made PHPWord to create word documents. – Basic Jul 22, 2012 at 16:45

18

this works with vs < office 2007 and its pure PHP, no COM crap, still trying to figure 2007

```php
<?php



/**************************************************************
This approach uses detection of NUL (chr(00)) and end line (chr(13))
to decide where the text is:
- divide the file contents up by chr(13)
- reject any slices containing a NUL
- stitch the rest together again
- clean up with a regular expression
**************************************************************/

function parseWord($userDoc)
{
    $fileHandle = fopen($userDoc, "r");
    $line = @fread($fileHandle, filesize($userDoc));
    $lines = explode(chr(0x0D),$line);
    $outtext = "";
    foreach($lines as $thisline)
      {
        $pos = strpos($thisline, chr(0x00));
        if (($pos !== FALSE)||(strlen($thisline)==0))
          {
          } else {
            $outtext .= $thisline." ";
          }
      }
     $outtext = preg_replace("/[^a-zA-Z0-9\s\,\.\-\n\r\t@\/\_\
```

```
(\)]/","",$outtext);
    return $outtext;
}

$userDoc = "cv.doc";

$text = parseWord($userDoc);
echo $text;


?>
```

Share

Improve this answer

Follow

edited Jan 24, 2009 at 3:59

UnkwnTech

90.7k ● 65 ● 191 ● 234

answered Nov 5, 2008 at 12:35

Mac

3 Do not use this if you want to preserve Umlaute. – Jan Beck May 4, 2012 at 15:41

I find some special characters that cannot be parsed in this function. – Roger Ng Jul 23, 2013 at 11:24

8

You can use Antiword, it is a free MS Word reader for Linux and most popular OS.

```
$document_file = 'c:\file.doc';
$text_from_doc = shell_exec('/usr/local/bin/antiword '.$document_file);
```

Share Improve this answer Follow

answered May 23, 2009 at 0:57

Mantichora

385 ● 4 ● 8

8 The problem with this type of solution is that it assumes that one is able to install software on the server. – UnkwnTech May 24, 2009 at 7:42

2 Bit of a long time, but correct me if i'm wrong. `C:\file.doc` is a windows directory and `/usr/local/bin` is a Linux/Unix directory? – Daryl Gill Apr 4, 2013 at 0:54

1 @UnkwnTech: as long as the program doesn't require elevated permission, most programs can be installed in any directory that you do have permission to write to. You can then use the full path to refer to the program, or add the install directory to your PATH variable. – Lie Ryan Jan 5, 2014 at 4:14

@LieRyan you missed the point, if your running this in a shared hosting environment you most often can't install any software regardless of the directory. – UnkwnTech Jan 7, 2014 at 0:23

@UnkwnTech: by installing, I meant simply copying it to any directory you have write permission on and setting the execute bit. This works in any shared hosting environment that gives you ssh access or at least the ability to execute scripts (i.e. the only environment this wouldn't work is on static file only hosting, but then you won't be talking about PHP anyway). If

you only have ftp access and no ssh, it's still possible, though you may need to write a few PHP script to set the execute bit. – Lie Ryan Jan 7, 2014 at 7:03

6

I don't know about reading native Word documents in PHP, but if you want to write a Word document in PHP, WordprocessingML (aka WordML) might be a good solution. All you have to do is create an XML document in the correct format. I believe Word 2003 and 2007 both support WordML.

Share Improve this answer Follow

answered Oct 10, 2008 at 0:23

Joe Lencioni
10.5k ● 18 ● 57 ● 68

6

Just updating the code

```php
<?php

/**************************************************************
This approach uses detection of NUL (chr(00)) and end line (chr(13))
to decide where the text is:
- divide the file contents up by chr(13)
- reject any slices containing a NUL
- stitch the rest together again
- clean up with a regular expression
**************************************************************/

function parseWord($userDoc)
{
    $fileHandle = fopen($userDoc, "r");
    $word_text = @fread($fileHandle, filesize($userDoc));
    $line = "";
    $tam = filesize($userDoc);
    $nulos = 0;
    $caracteres = 0;
    for($i=1536; $i<$tam; $i++)
    {
        $line .= $word_text[$i];

        if( $word_text[$i] == 0)
        {
            $nulos++;
        }
        else
        {
            $nulos=0;
            $caracteres++;
        }

        if( $nulos>1996)
        {
            break;
        }
    }
```

```
    //echo $caracteres;

    $lines = explode(chr(0x0D),$line);
    //$outtext = "<pre>";

    $outtext = "";
    foreach($lines as $thisline)
    {
        $tam = strlen($thisline);
        if( !$tam )
        {
            continue;
        }

        $new_line = "";
        for($i=0; $i<$tam; $i++)
        {
            $onechar = $thisline[$i];
            if( $onechar > chr(240) )
            {
                continue;
            }

            if( $onechar >= chr(0x20) )
            {
                $caracteres++;
                $new_line .= $onechar;
            }

            if( $onechar == chr(0x14) )
            {
                $new_line .= "</a>";
            }

            if( $onechar == chr(0x07) )
            {
                $new_line .= "\t";
                if( isset($thisline[$i+1]) )
                {
                    if( $thisline[$i+1] == chr(0x07) )
                    {
                        $new_line .= "\n";
                    }
                }
            }
        }
        //troca por hiperlink
        $new_line = str_replace("HYPERLINK" ,"<a href=",$new_line);
        $new_line = str_replace("\o" ,">",$new_line);
        $new_line .= "\n";

        //link de imagens
        $new_line = str_replace("INCLUDEPICTURE" ,"<br><img src=",$new_line);
        $new_line = str_replace("\*" ,"><br>",$new_line);
        $new_line = str_replace("MERGEFORMATINET" ,"",$new_line);


        $outtext .= nl2br($new_line);
    }

 return $outtext;
}
```

```
$userDoc = "custo.doc";
$userDoc = "Cultura.doc";
$text = parseWord($userDoc);

echo $text;


?>
```

Share

Improve this answer

Follow

edited Apr 4, 2011 at 12:56

Bill the Lizard

405k ● 211 ● 572 ● 889

answered Apr 4, 2011 at 2:43

WIlson

61 ● 1 ● 1

Although interesting, this failed to find the start of a Word97 document, and cut the document off. I found it's in the 1536 and 1996 numbers, which should be determined by parsing, not arbitrary hardcoding. As well, the special chars like smart quotes, ellipses, em-dash, and special single quotes all were stripped, and I saw a lot of ampersands throughout the output. So, this is an interesting start, but needs a lot of refinement. – Volomike Aug 11, 2011 at 16:51

You may also want to reference this tutorial on how to convert special MS Word characters: toao.net/48-replacing-smart-quotes-and-em-dashes-in-mysql – Volomike Aug 11, 2011 at 16:52

1 the function produces some strange chars: "Œ'ÛJA†ïßaÈ}7Û"ÒÙÞH¡w"ë„™ìwì¤Ú¾½..." – Yoong Kim Jul 16, 2012 at 10:04

@Volomike change `$nulus` to a higher number to avoid the break. – Peyman Aug 3, 2017 at 15:19

5

Most probably you won't be able to read Word documents without COM.

Writing was covered in this topic

Share

Improve this answer

Follow

edited May 23, 2017 at 12:10

Community Bot

1 ● 1

answered Oct 10, 2008 at 2:17

Sergey Kornilov

1,792 ● 2 ● 13 ● 22

3

2007 might be a bit complicated as well.

The .docx format is a zip file that contains a few folders with other files in them for formatting and other stuff.

Rename a .docx file to .zip and you'll see what I mean.

So if you can work within zip files in PHP, you should be on the right path.

Share

Improve this answer

Follow

answered Oct 10, 2008 at 15:24

community wiki
databyss

2

www.phplivedocx.org is a SOAP based service that means that you always need to be online for testing the Files also does not have enough examples for its use . Strangely I found only after 2 days of downloading (requires additionaly zend framework too) that its a SOAP based program(cursed me !!!)...I think without COM its just not possible on a Linux server and the only idea is to change the doc file in another usable file which PHP can parse...

Share Improve this answer Follow

answered Sep 13, 2009 at 17:45

jahajee

2

Source gotten from

> Use following class directly to read word document

```
class DocxConversion{
    private $filename;

    public function __construct($filePath) {
        $this->filename = $filePath;
    }

    private function read_doc() {
        $fileHandle = fopen($this->filename, "r");
        $line = @fread($fileHandle, filesize($this->filename));
        $lines = explode(chr(0x0D),$line);
        $outtext = "";
        foreach($lines as $thisline)
          {
            $pos = strpos($thisline, chr(0x00));
            if (($pos !== FALSE)||(strlen($thisline)==0))
              {
              } else {
                $outtext .= $thisline." ";
              }
          }
         $outtext = preg_replace("/[^a-zA-Z0-9\s\,\.\-\n\r\t@\/\_\
(\)]/","",$outtext);
        return $outtext;
    }

    private function read_docx(){

        $striped_content = '';
        $content = '';
```

```
        $zip = zip_open($this->filename);

        if (!$zip || is_numeric($zip)) return false;

        while ($zip_entry = zip_read($zip)) {

            if (zip_entry_open($zip, $zip_entry) == FALSE) continue;

            if (zip_entry_name($zip_entry) != "word/document.xml") continue;

            $content .= zip_entry_read($zip_entry,
zip_entry_filesize($zip_entry));

            zip_entry_close($zip_entry);
        }// end while

        zip_close($zip);

        $content = str_replace('</w:r></w:p></w:tc><w:tc>', " ", $content);
        $content = str_replace('</w:r></w:p>', "\r\n", $content);
        $striped_content = strip_tags($content);

        return $striped_content;
    }

 /************************excel sheet***********************************/

function xlsx_to_text($input_file){
    $xml_filename = "xl/sharedStrings.xml"; //content file name
    $zip_handle = new ZipArchive;
    $output_text = "";
    if(true === $zip_handle->open($input_file)){
        if(($xml_index = $zip_handle->locateName($xml_filename)) !== false){
            $xml_datas = $zip_handle->getFromIndex($xml_index);
            $xml_handle = DOMDocument::loadXML($xml_datas, LIBXML_NOENT |
LIBXML_XINCLUDE | LIBXML_NOERROR | LIBXML_NOWARNING);
            $output_text = strip_tags($xml_handle->saveXML());
        }else{
            $output_text .="";
        }
        $zip_handle->close();
    }else{
    $output_text .="";
    }
    return $output_text;
}

/************************power point files*****************************/
function pptx_to_text($input_file){
    $zip_handle = new ZipArchive;
    $output_text = "";
    if(true === $zip_handle->open($input_file)){
        $slide_number = 1; //loop through slide files
        while(($xml_index = $zip_handle-
>locateName("ppt/slides/slide".$slide_number.".xml")) !== false){
            $xml_datas = $zip_handle->getFromIndex($xml_index);
            $xml_handle = DOMDocument::loadXML($xml_datas, LIBXML_NOENT |
LIBXML_XINCLUDE | LIBXML_NOERROR | LIBXML_NOWARNING);
            $output_text .= strip_tags($xml_handle->saveXML());
            $slide_number++;
        }
        if($slide_number == 1){
```

```
            $output_text .="";
        }
        $zip_handle->close();
    }else{
    $output_text .="";
    }
    return $output_text;
}


    public function convertToText() {

        if(isset($this->filename) && !file_exists($this->filename)) {
            return "File Not exists";
        }

        $fileArray = pathinfo($this->filename);
        $file_ext  = $fileArray['extension'];
        if($file_ext == "doc" || $file_ext == "docx" || $file_ext == "xlsx" ||
$file_ext == "pptx")
        {
            if($file_ext == "doc") {
                return $this->read_doc();
            } elseif($file_ext == "docx") {
                return $this->read_docx();
            } elseif($file_ext == "xlsx") {
                return $this->xlsx_to_text();
            }elseif($file_ext == "pptx") {
                return $this->pptx_to_text();
            }
        } else {
            return "Invalid File Type";
        }
    }

}

$docObj = new DocxConversion("test.docx"); //replace your document name with
correct extension doc or docx
echo $docText= $docObj->convertToText();
```

Share Improve this answer Follow

answered Jul 3, 2019 at 10:25

Mohamed Faalil

104 ● 5

1

Office 2007 .docx should be possible since it's an XML standard. Word 2003 most likely requires COM to read, even with the standards now published by MS, since those standards are huge. I haven't seen many libraries written to match them yet.

Share Improve this answer Follow

answered Oct 10, 2008 at 2:45

acrosman

12.9k ● 10 ● 41 ● 56

1

I don't know what you are going to use it for, but I needed .doc support for search indexing; What I did was use a little commandline tool called "catdoc"; This transfers the contents of the Word document to plain text so it can be indexed. If you need to keep formatting and stuff this is not your tool.

Share Improve this answer Follow

answered Oct 10, 2008 at 15:25

fijter

18k ● 2 ● 26 ● 28

1

phpLiveDocx is a Zend Framework component and can read and write DOC and DOCX files in PHP on Linux, Windows and Mac.

See the project web site at:

http://www.phplivedocx.org

Share Improve this answer Follow

answered May 14, 2009 at 7:03

Leo Bonnafe

Reference Link is dead – Zameer Khan Mar 19, 2017 at 13:00

1

One way to manipulate Word files with PHP that you may find interesting is with the help of PHPDocX. You may see how it works having a look at its online tutorial. You can insert or extract contents or even merge multiple Word files into a asingle one.

Share Improve this answer Follow

answered Sep 28, 2012 at 16:44

Eduardo

90 ● 1

0

Would the .rtf format work for your purposes? .rtf can easily be converted to and from .doc format, but it is written in plaintext (with control commands embedded). This is how I plan to integrate my application with Word documents.

Share Improve this answer Follow

answered Jan 24, 2009 at 5:09

Josh Smeaton

48.7k ● 24 ● 135 ● 165

Circumstance is irrelivent the question was weather or not it was possible, but thanks.
– UnkwnTech Jan 24, 2009 at 11:54

0

even i'm working on same kind of project [An Onlinw Word Processor]! But i've choosen c#.net and ASP.net. But through the survey i did; i got to know that

> By Using Open XML SDK and VSTO [Visual Studio Tools For Office]

we may easily work with a word file manipulate them and even convert internally to different into several formats such as .odt,.pdf,.docx etc..

> **So, goto msdn.microsoft.com and be thorough about the office development tab. Its the easiest way to do this as all functions we need to implement are already available in .net!!**

But as u want to do ur project in PHP, u can do it in Visual Studio and .net as PHP is also one of the .net Compliant Language!!

Share Improve this answer Follow

answered Sep 5, 2010 at 14:17

Noddy Cha
851 ● 1 ● 13 ● 19

0

I have the same case I guess I am going to use a cheap 50 mega windows based hosting with free domain to use it to convert my files on, for PHP server. And linking them is easy. All you need is make an ASP.NET page that recieves the doc file via post and replies it via HTTP so simple CURL would do it.

Share Improve this answer Follow

answered Oct 11, 2010 at 19:12

Omer
1

1 Seems like this is the only way to do it after all. Can you provide more details ? I mean, am I supposed to go and purchase a windows hosting and use it to run a PHP code (that uses the COM library) to create the .doc/x file? – Dewan159 Jul 3, 2012 at 16:30