

# Re: Byte Array to String

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2007-11/msg02224.html>

---

- *From:* "AG" <[NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx](mailto:NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Fri, 23 Nov 2007 08:19:58 -0500
- 

Thanks Steven.

Encoding.Default, which is 1252 does work.as I reported in my response to Henk's post.

I knew there had to be a simple solution.

Thanks to all responders.

—

AG

Email: [discussATadhdataDOTcom](mailto:discussATadhdataDOTcom)

"Steven Cheng[MSFT]" <[stcheng@xxxxxxxxxxxxxxxxxxxx](mailto:stcheng@xxxxxxxxxxxxxxxxxxxx)> wrote in message <news:udZHkTZLIHA.5204@xxxxxxxxxxxxxxxxxxxx>

Thanks for your reply,

Yes, for text file, if we doesn't get the correct encoding/charset, the retrieved text will mismatch the original characters.

For your scenario, I think VBA may use the default system locale to encoding the characters. You can also try

"Encoding.Default" as the parameter in the SreamReader's constructor.

"Encoding.Default" means the current system ANSI codepage. If this still not work, I think the VBA is producing the file like a binary format one(doesn't use a consistent encoding for the entire file) and thus, using binary read mode to decode it individually should be reasonable.

Anyway, if you have any further questions on this, welcome to post here.

Sincerely,

Steven Cheng

Microsoft MSDN Online Support Lead

This posting is provided "AS IS" with no warranties, and confers no rights.

Re: Byte Array to String

---

Reply-To: "AG" <NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx>  
From: "AG" <NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx>  
References: <eMdm3uLLIHA.4948@xxxxxxxxxxxxxxxxxxxx>

<wUDfEONLIHA.7800@xxxxxxxxxxxxxxxxxxxx>

Subject: Re: Byte Array to String  
Date: Thu, 22 Nov 2007 09:25:49 -0500

Thanks for the reply Steven.

I ended up reading as byte and converting myself because text reading mode (streamreader) produced the wrong characters for the extended ASCII characters.

Perhaps a bit more of an explanation.

The file is created by an Access application using VBA, as a method of exporting some database data. Since the data may contain all the usual record and field separators like crlf, commas, tabs, quotes, etc., the extended ASCII chars are used as record and field separators.

It is created using the Open for append method and data added via the

Print

method, as follows. This method can not be changed, as it is in use in too many locations.

```
Dim strRecord as string  
strRecord = "field1data" & Chr(128) & "field2data" & Chr(128) &
```

```
"field3data"
```

```
& Chr(129)  
Open <thefile> For Append As #1  
Print #1, strRecord  
Close #1
```

As you can see, there is no BOM.

```
The file is easily opened and read in VBA using Open For Binary:  
Dim strFileData as String  
Open <thefile> For Binary As #1
```

Re: Byte Array to String

```
strFileData = space(FileLen(<thefile>)  
Get #1, , strFileData  
Close #1
```

This all works fine in VBA. Now, I would like to read the file using .NET framework.

While my method of using Chr() on each byte works, it would seem that

there

should be a similar simple method in .NET to get the file contents without looping through each byte.

According to the help file, Chr uses the Encoding class to return the appropriate character, so isn't there a method in the Encoding class that would perform the operation on the entire stream?

--

AG

Email: discussATadhdataDOTcom

"Steven Cheng[MSFT]" <stcheng@xxxxxxxxxxxxxxxxxxxxxx> wrote in message

[news:wUDfEONLIHA.7800@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:wUDfEONLIHA.7800@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx)

Hi AG,

If the file contains character that exceed the ASCII char code scope (and those chars are stored correctly), that means the file's content is not stored as ASCII encoding (single byte charset).

Generally speaking, if you're reading a text file (which means its content are character text rather than unreadable binary content), you should use text reading mode to read them (rather than read them as byte and convert them your self).

And to read file as text mode, you need to know what is the encoding/charset of the text file's content. this info is needed when you try reading the file in Text Mode. For example, you can use the "StreamReader" class in .net to read file in text mode as below:

Re: Byte Array to String

```
=====
StreamReader sr = new StreamReader("inputfile.txt",
Encoding.UTF8);
string content = sr.ReadToEnd();

sr.Close();
=====
```

or you can also let the StreamReader to determine the encoding automatically (through file's BOM). But BOM(Byte Order mark) is not existent in text file:

```
=====
StreamReader sr1 = new StreamReader("inputfile.txt", true);

string content1 = sr1.ReadToEnd();

sr1.Close();
=====
```

for your case, I think the file's encoding is likely not UTF8, and if you use UTF8 to decode the byte, you'll probably get wrong character.

Sincerely,

Steven Cheng

Microsoft MSDN Online Support Lead

This posting is provided "AS IS" with no warranties, and confers no rights.

```
-----
Reply-To: "AG"
<NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx>
From: "AG"
<NOSPAMa-giam@xxxxxxxxxxxxxxxxxxxx>
Subject: Byte Array to String
Date: Wed, 21 Nov 2007 22:56:55 -0500
```

I have a file that contains ASCII and Extended ASCII characters.

## Re: Byte Array to String

I need to get the file contents into a string, but the Extended ASCII characters (dec 128 and 129) are being changed to dec 63.

I have tried several methods, but here is the one I thought would have worked.

```
Dim strReturn As String
Dim arBytes() As Byte
arBytes =
System.IO.File.ReadAllBytes(<myfile>)
strReturn =
System.Text.Encoding.UTF8.GetString(arBytes)
```

When I examine strReturn, I find that the chars that should be chr(128)

and

chr(129) are all chr(63).

The only thing I could get to work is

```
Dim strReturn As String = String.Empty
Dim arBytes() As Byte
Dim sB As New StringBuilder
Dim byT As Byte

arBytes =
System.IO.File.ReadAllBytes(strPathFile)
For Each byT In arBytes
sB.Append(Chr(byT))
Next
strReturn = sB.ToString
```

Can anyone offer an explanation, and/or a better method?

--

AG  
Email: discussATadhddataDOTcom

Re: Byte Array to String