
Table of Contents

Foreword	0
Part I Introduction	2
Part II Installation and Testing	2
Part III Reference Guide	3
1 Properties	3
BackColor Property	3
BarColor Property	3
BorderStyle Property	4
Data Property	4
HandleTilde Property	4
Height Property	5
Mode Property	5
ModuleSize Property	6
Orientation Property	6
PNGImage Property	6
PreferredFormat Property	7
Width Property	8
2 Methods	8
GetActualRC Method	8
GetActualSize Method	9
SetStructuredAppend Method	9
Part IV License	9
Index	0

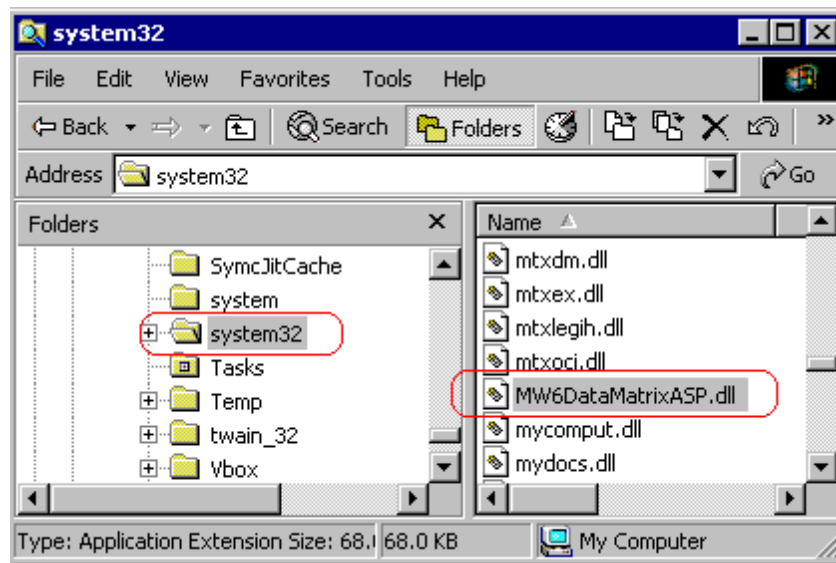
1 Introduction

MW6 DataMatrix ASP component is an ATL COM product which can add professional quality 2D DataMatrix PNG format images to your web pages hosted on the IIS server.

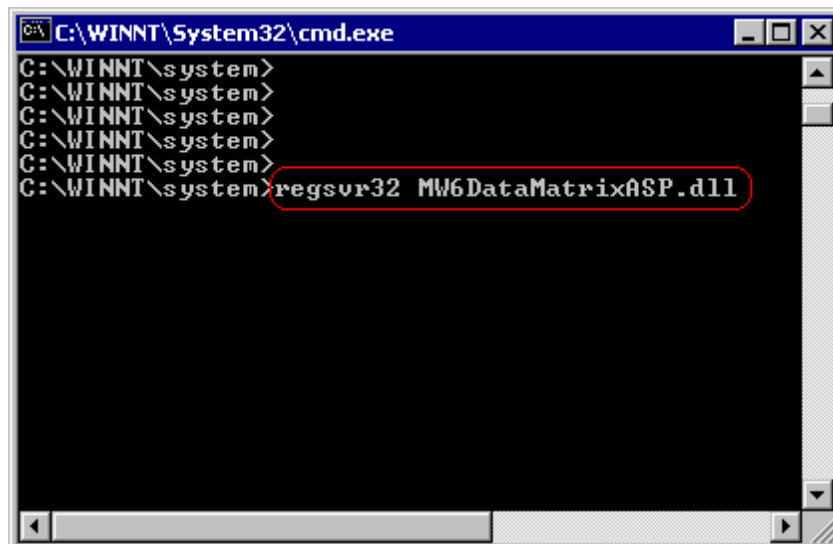
DataMatrix is designed to pack a lot of information in a very small space, our DataMatrix ASP control supports the ECC-200 version, it is capable of encoding 1556 bytes, 2335 alphanumeric characters, or 3116 numeric digits.

2 Installation and Testing

1. The trial version DataMatrix ASP component appends "MW6 Demo" to the string encoded with the DataMatrix format.
2. For 32-bit version Windows OS, go to the 32-bit system folder (e.g. "c:\winnt\system32" or "c:\windows\system32") of the IIS server. For 64-bit version Windows OS, go to the SysWOW64 folder (e.g. "c:\windows\SysWOW64") of the IIS server. Copy "MW6DataMatrixASP.dll" to the current folder.



3. From the current folder, run "regsvr32 MW6DataMatrixASP.dll" to register it. For Windows Vista or above, you need to use an elevated Command Prompt to run `regsvr32.exe` command, click "Start" > "All Programs" > "Accessories", right-click "Command Prompt", and then click "Run" as administrator.



```
C:\WINNT\System32\cmd.exe
C:\WINNT\system>
C:\WINNT\system>
C:\WINNT\system>
C:\WINNT\system>
C:\WINNT\system>
C:\WINNT\system> regsvr32 MW6DataMatrixASP.dll
```

4. Copy Demo.html, Createlmng.asp and Show.asp to a folder of the IIS server where Active Server Pages are enabled, for example, you can create one folder "C:\inetpub\wwwroot\MyFolder" and copy those 3 files to this folder.
5. Enter the URL of Demo.html to your browser for verifying whether DataMatrix ASP component is working or not, for example, you can enter <http://localhost/MyFolder/Demo.html> for testing it on the IIS server itself.

3 Reference Guide

3.1 Properties

3.1.1 BackColor Property

Sets the background color of the DataMatrix barcode.

Property Data Type

OLE_COLOR

Remarks

The default value is white color.

3.1.2 BarColor Property

Sets the color of the DataMatrix barcode and text.

Property Data Type

OLE_COLOR

Remarks

The default value is black color.

3.1.3 BorderStyle Property

Sets the style of the border rectangle.

Property Data Type

short

Remarks

The default value is 0, this property can be one of the following values:

Value	Description
0	No Border
1	Dash Border
2	Solid Border

3.1.4 Data Property

Sets the message to encode with DataMatrix ASP component.

Property Data Type

BSTR

Remarks

The default value is "12".

3.1.5 HandleTilde Property

Sets a boolean flag indicating whether to process the tilde character "~" or not.

Property Data Type

VARIANT_BOOL

Remarks

If this property is set to TRUE, non-printable characters can be passed to DataMatrix ASP component by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

"~1" is used to indicate FNC1. For example, "~10107612345678900~117100503" can be used to generate GS1 DataMatrix "(01)0107612345678900(17)100503", and "~110AC34563G3" can be used to generate GS1 DataMatrix "(10)AC34563G3".

"~5" is used to indicate Macro 5. For example, "~5ABCDEF[GS]123456" can be used to generate DataMatrix "[>[RS]05[GS]ABCDEF[GS]123456[RS]][EOT]".

"~6" is used to indicate Macro 6. For example, "~6ABCDEF[GS]123456" can be used to generate DataMatrix "[>[RS]06[GS]ABCDEF[GS]123456[RS]][EOT]".

[RS] is the record separator with ASCII value 30, [GS] is the group separator with ASCII value 29, and [EOT] is the end of transmission with ASCII value 4.

3.1.6 Height Property

Sets the height, in pixels, of the image which contains the DataMatrix barcode.

Property Data Type

short

Remarks

First call GetActualSize() method to obtain the actual size of the DataMatrix barcode, then use this property and Width property to set image size by adding surrounding white space around the DataMatrix barcode.

See Also

GetActualSize() Method

3.1.7 Mode Property

Sets the encoding mode of the DataMatrix barcode.

Property Data Type

short

Remarks

This parameter can be one of the following values:

Value	Description
0	ASCII mode for mainly encoding ASCII characters (0-127)
1	C40 mode for mainly encoding numeric and upper case characters
2	Text mode for mainly encoding numeric and lower case characters
3	Base256 mode for mainly encoding bytes of data

3.1.8 ModuleSize Property

Sets the size (width/height) of the square-shaped module.

Property Data Type

float

Remarks

The default value is 0.07, internally our DataMatrix ASP control converts the module size from centimeters to pixels based on the device resolution, round up or round down float pixel value to the nearest integer.

The centimeter to pixel conversion formula is :

$$size_in_pixels = size_in_centimeters * device_resolution / 2.54$$

For example, if you render barcode on computer screen and the screen resolution is 96dpi.

(1) Set ModuleSize property to 0.04, $size_in_pixels = 0.04 * 96 / 2.54 = 1.5118$, round up 1.5118 to 2, so actual module size is 2 pixels.

(2) Set ModuleSize property to 0.06, $size_in_pixels = 0.06 * 96 / 2.54 = 2.2677$, round down 2.2677 to 2, so actual module size is 2 pixels.

(3) Set ModuleSize property to 0.07, $size_in_pixels = 0.07 * 96 / 2.54 = 2.6456$, round up 2.6456 to 3, so actual module size is 3 pixels.

Different ModuleSize property values might end up with same module size in pixels due to performing rounding operations.

3.1.9 Orientation Property

Sets the orientation of the DataMatrix barcode.

Property Data Type

short

Remarks

The default value is 0, this property can be one of the following values:

Value	Description
0	0 degree
1	90 degrees
2	180 degrees
3	270 degrees

3.1.10 PNGImage Property

Gets PNG format image data stream of the DataMatrix barcode.

Property Data Type

VARIANT

Remarks

Use ASP Response.BinaryWrite() method and this property to display a DataMatrix barcode image on a web page.

3.1.11 PreferredFormat Property

Sets the format of the DataMatrix barcode.

Property Data Type

short

Remarks

The default value is 0, this property can be one of the following values.

Value	Description	Data Capacity		
		Numeric	Alphanumeric	Byte
0	Auto format			
1	10 X 10 format	6	3	1
2	12 X 12 format	10	6	3
3	14 X 14 format	16	10	6
4	16 X 16 format	24	16	10
5	18 X 18 format	36	25	16
6	20 X 20 format	44	31	20
7	22 X 22 format	60	43	28
8	24 X 24 format	72	52	34
9	26 X 26 format	88	64	42
10	32 X 32 format	124	91	60
11	36 X 36 format	172	127	84
12	40 X 40 format	228	169	112
13	44 X 44 format	288	214	142
14	48 X 48 format	348	259	172
15	52 X 52 format	408	304	202
16	64 X 64 format	560	418	278
17	72 X 72 format	736	550	366
18	80 X 80 format	912	682	454
19	88 X 88 format	1152	862	574
20	96 X 96 format	1392	1042	694
21	104 X 104 format	1632	1222	814
22	120 X 120 format	2100	1573	1048
23	132 X 132 format	2608	1954	1302
24	144 X 144 format	3116	2335	1556
25	8 X 18 format	10	6	3
26	8 X 32 format	20	13	8
27	12 X 26 format	32	22	14
28	12 X 36 format	44	31	20
29	16 X 36 format	64	46	30

30	16 X 48 format	98	72	47
----	----------------	----	----	----

If you set *PreferredFormat* to 0 (Auto format), our DataMatrix ASP control will automatically choose an appropriate format with enough data capacity to encode the string.

If you set *PreferredFormat* to other values and the data capacity of the selected format is not big enough to encode the string, our DataMatrix ASP control will also automatically choose an appropriate format with bigger data capacity to encode the string.

3.1.12 Width Property

Sets the width, in pixels, of the image which contains the DataMatrix barcode.

Property Data Type

short

Remarks

First call *GetActualSize()* method to obtain the actual size of the DataMatrix barcode, then use this property and *Height* property to set image size by adding surrounding white space around the DataMatrix barcode.

See Also

GetActualSize() Method

3.2 Methods

3.2.1 GetActualRC Method

Gets the actual numbers of rows and columns for the DataMatrix barcode.

```
void GetActualRC(VARIANT *ActualRows, VARIANT *ActualCols);
```

Parameters

ActualRows

A pointer to the variable that receives the final number of rows for the DataMatrix barcode.

ActualCols

A pointer to the variable that receives the final number of columns for the DataMatrix barcode.

Remarks

If you set *PreferredFormat* to 0 (Auto format), DataMatrix ASP control will automatically choose an appropriate format with enough data capacity to encode the string, use this method to retrieve the information about the final numbers of rows and columns.

If you set PreferredFormat to other values and the data capacity of the selected format is not big enough to encode the string, DataMatrix ASP control will also automatically choose an appropriate format with bigger data capacity to encode the string, so the final numbers of rows and columns might not be equal to the numbers of rows and columns specified by the PreferredFormat property.

3.2.2 GetActualSize Method

Gets the actual size of the DataMatrix barcode which is rendered onto computer screen.

```
void GetActualSize(VARIANT *ActualWidth, VARIANT *ActualHeight);
```

Parameters

ActualWidth

A pointer to the variable that receives the width of the DataMatrix barcode (in pixels).

ActualHeight

A pointer to the variable that receives the height of the DataMatrix barcode (in pixels).

3.2.3 SetStructuredAppend Method

Specifies which symbol this is in a sequence and the total number of symbols in the sequence.

```
void SetStructuredAppend(VARIANT_BOOL AllowSA, short SymbolID, short SymbolCount);
```

Parameters

AllowSA

Indicates whether the structured append is allowed in the current DataMatrix barcode, if this is FALSE, the parameters *SymbolID* and *SymbolCount* are irrelevant.

SymbolID

Specifies which symbol this is in a sequence, the parameter must be between 1 and *SymbolCount*.

SymbolCount

Specifies the total number of symbols in the sequence, the maximum value is 16, which means that up to 16 symbols can be linked together using the structured append protocol.

Remarks

Don't call this method if you don't need the structured append feature.

4 License

License agreement

This License Agreement ("LA") is the legal agreement between you and MW6 Technologies, Inc. ("MW6") for the font, and any electronic documentation ("Package"). By using, copying or installing the Package, you agree to be bound by the terms of this LA. If you don't agree to the terms in this LA,

immediately remove unused Package.

1. License

* The Single Server License allows the use of the software (up to 10,000 users) on ONE server with ONE CPU in your organization.

* The 2 Server License allows the use of the software (up to 10,000 users) on 2 servers (each server has only 1 CPU) in your organization.

* The 3 Server License allows the use of the software (up to 10,000 users) on 3 servers (each server has only 1 CPU) in your organization.

* The 4 Server License allows the use of the software (up to 10,000 users) on 4 servers (each server has only 1 CPU) in your organization.

* The 5 Server License allows the use of the software (up to 10,000 users) on 5 servers (each server has only 1 CPU) in your organization.

* The Unlimited Developer License allows the use of the software (unlimited number of users) on unlimited number of servers (each server has unlimited number of CPUs) in your organization.

2. User Disclaimer

The software is provided "as is" without warrant of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. MW6 assumes no liability for damages, direct or consequential, which may result from the use of the software. Further, MW6 assumes no liability for losses caused by misuse or abuse of the software. This responsibility rests solely with the end user.

3. Copyright

The software and any electronic documentation are the proprietary products of MW6 and are protected by copyright and other intellectual property laws.
