

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
Height Property	4
Level Property	5
Mask Property	5
ModuleSize Property	6
Orientation Property	6
PNGImage Property	7
Version Property	7
Width Property	8
2 Methods	8
GetActualRC Method	8
GetActualSize Method	9
3 Data Capacity Tables	9
Level L	9
Level M	10
Level Q	11
Level H	12
Part IV License	13
Index	0

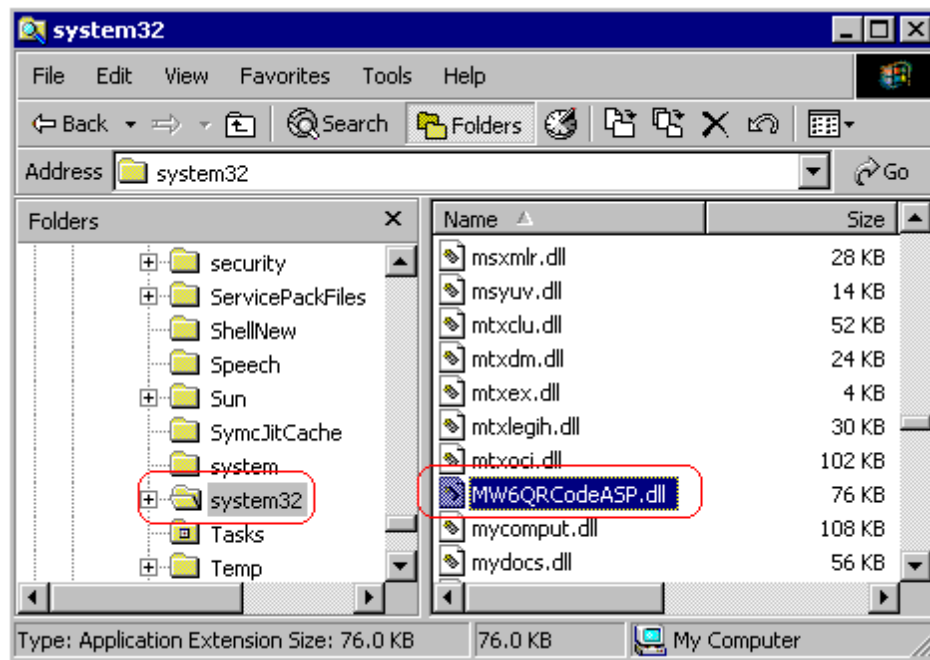
1 Introduction

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

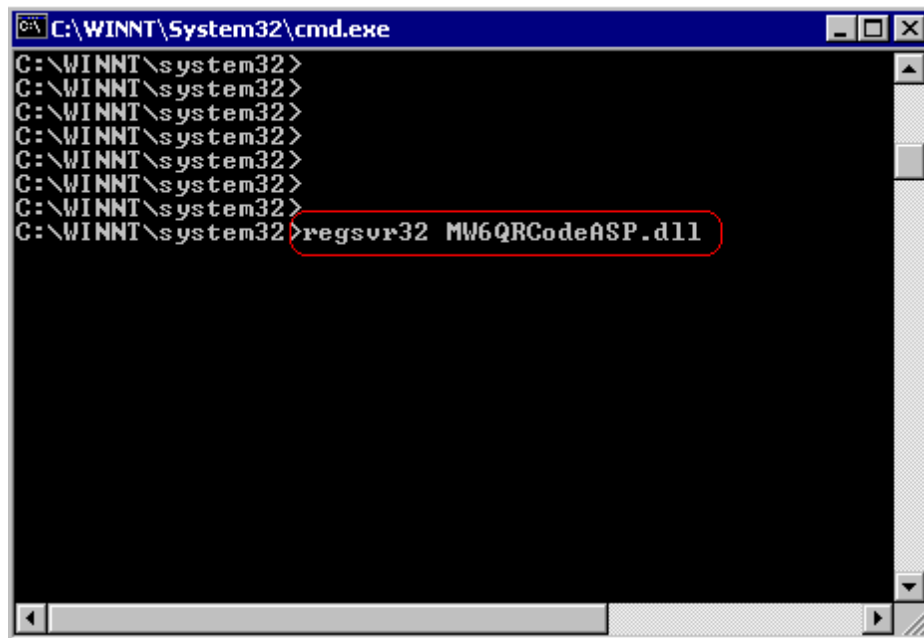
QRCode is designed to pack a lot of information in a very small space, our QRCode ASP component supports Model 2, it is capable of encoding up to 2953 bytes, 4296 alphanumeric characters, or 7089 numeric digits.

2 Installation and Testing

1. The trial version QRCode ASP component appends "MW6 Demo" to the string encoded with the QRCode 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 MW6QRCodeASP.dll to the current folder.



3. From the current folder, run "regsvr32 MW6QRCodeASP.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\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>
C:\WINNT\system32>regsvr32 MW6QRCodeASP.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 QRCode 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 QRCode barcode.

Property Data Type

OLE_COLOR

Remarks

The default value is white color.

3.1.2 BarColor Property

Sets the color of the QRCode 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 QRCode ASP component.

Property Data Type

BSTR

Remarks

The default value is "12".

3.1.5 Height Property

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

Property Data Type

short

Remarks

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

See Also

GetActualSize() Method

3.1.6 Level Property

Sets the level of error correction allowing recovery.

Property Data Type

short

Remarks

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

Value	Comment
0	Level L
1	Level M
2	Level Q
3	Level H

3.1.7 Mask Property

Sets the mask pattern for improving the readability.

Property Data Type

short

Remarks

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

Value	Comment
0	Auto
1	Mask 0
2	Mask 1
3	Mask 2
4	Mask 3
5	Mask 4
6	Mask 5
7	Mask 6

8	Mask 7
---	--------

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 QRCode 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 QRCode 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 QRCode barcode.

Property Data Type

VARIANT

Remarks

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

3.1.11 Version Property

Sets the version of the QRCode barcode.

Property Data Type

short

Remarks

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

Value	Description
0	Auto
1	21 X 21
2	25 X 25
3	29 X 29
4	33 X 33
5	37 X 37
6	41 X 41
7	45 X 45
8	49 X 49
9	53 X 53
10	57 X 57
11	61 X 61
12	65 X 65
13	69 X 69
14	73 X 73
15	77 X 77
16	81 X 81
17	85 X 85
18	89 X 89
19	93 X 93
20	97 X 97
21	101 X 101
22	105 X 105
23	109 X 109
24	113 X 113
25	117 X 117
26	121 X 121

27	125 X 125
28	129 X 129
29	133 X 133
30	137 X 137
31	141 X 141
32	145 X 145
33	149 X 149
34	153 X 153
35	157 X 157
36	161 X 161
37	165 X 165
38	169 X 169
39	173 X 173
40	177 X 177

If you set *Version* to 0 (Auto version), our QRCode ASP component will automatically choose an appropriate version with enough data capacity to encode the string.

If you set *Version* to other values and the data capacity of the selected version is not big enough to encode the string, our QRCode ASP component will also automatically choose an appropriate version with bigger data capacity to encode the string.

3.1.12 Width Property

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

Property Data Type

short

Remarks

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

See Also

`GetActualSize()` Method

3.2 Methods

3.2.1 GetActualRC Method

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

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

Parameters

ActualRows

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

ActualCols

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

Remarks

If you set *Version* to 0 (Auto version), QRCode 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 *Version* to other values and the data capacity of the selected format is not big enough to encode the string, QRCode 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 *Version* property.

3.2.2 GetActualSize Method

Gets the actual size of the QRCode 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 QRCode barcode (in pixels).

ActualHeight

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

3.3 Data Capacity Tables

3.3.1 Level L

Version	Capacity (in digits)	Capacity (in alphanumeric characters)	Capacity (in bytes)
1	41	25	17
2	77	47	32
3	127	77	53
4	187	114	78
5	255	154	106
6	322	195	134
7	370	224	154
8	461	279	192
9	552	335	230
10	652	395	271
11	772	468	321
12	883	535	367
13	1022	619	425

14	1101	667	458
15	1250	758	520
16	1408	854	586
17	1548	938	644
18	1725	1046	718
19	1903	1153	792
20	2061	1249	858
21	2232	1352	929
22	2409	1460	1003
23	2620	1588	1091
24	2812	1704	1171
25	3057	1853	1273
26	3283	1990	1367
27	3517	2132	1465
28	3669	2223	1528
29	3909	2369	1628
30	4158	2520	1732
31	4417	2677	1840
32	4686	2840	1952
33	4965	3009	2068
34	5253	3183	2188
35	5529	3351	2303
36	5836	3537	2431
37	6153	3729	2563
38	6479	3927	2699
39	6743	4087	2809
40	7089	4296	2953

3.3.2 Level M

Version	Capacity (in digits)	Capacity (in alphanumeric characters)	Capacity (in bytes)
1	34	20	14
2	63	38	26
3	101	61	42
4	149	90	62
5	202	122	84
6	255	154	106
7	293	178	122
8	365	221	152
9	432	262	180
10	513	311	213
11	604	366	251
12	691	419	287
13	796	483	331
14	871	528	362
15	991	600	412
16	1082	656	450
17	1212	734	504
18	1346	816	560
19	1500	909	624
20	1600	970	666

21	1708	1035	711
22	1872	1134	779
23	2059	1248	857
24	2188	1326	911
25	2395	1451	997
26	2544	1542	1059
27	2701	1637	1125
28	2857	1732	1190
29	3035	1839	1264
30	3289	1994	1370
31	3486	2113	1452
32	3693	2238	1538
33	3909	2369	1628
34	4134	2506	1722
35	4343	2632	1809
36	4588	2780	1911
37	4775	2894	1989
38	5039	3054	2099
39	5313	3220	2213
40	5596	3391	2331

3.3.3 Level Q

Version	Capacity (in digits)	Capacity (in alphanumeric characters)	Capacity (in bytes)
1	27	16	11
2	48	29	20
3	77	47	32
4	111	67	46
5	144	87	60
6	178	108	74
7	207	125	86
8	259	157	108
9	312	189	130
10	364	221	151
11	427	259	177
12	489	296	203
13	580	352	241
14	621	376	258
15	703	426	292
16	775	470	322
17	876	531	364
18	948	574	394
19	1063	644	442
20	1159	702	482
21	1224	742	509
22	1358	823	565
23	1468	890	611
24	1588	963	661
25	1718	1041	715
26	1804	1094	751
27	1933	1172	805

28	2085	1263	868
29	2181	1322	908
30	2358	1429	982
31	2473	1499	1030
32	2670	1618	1112
33	2805	1700	1168
34	2949	1787	1228
35	3081	1867	1283
36	3244	1966	1351
37	3417	2071	1423
38	3599	2181	1499
39	3791	2298	1597
40	3993	2420	1663

3.3.4 Level H

Version	Capacity (in digits)	Capacity (in alphanumeric characters)	Capacity (in bytes)
1	17	10	7
2	34	20	14
3	58	35	24
4	82	50	34
5	106	64	44
6	139	84	58
7	154	93	64
8	202	122	84
9	235	143	98
10	288	174	119
11	331	200	137
12	374	227	155
13	427	259	177
14	468	283	194
15	530	321	220
16	602	365	250
17	674	408	280
18	746	452	310
19	813	493	338
20	919	557	382
21	969	587	403
22	1056	640	439
23	1108	672	461
24	1228	744	511
25	1286	779	535
26	1425	864	593
27	1501	910	625
28	1581	958	658
29	1677	1016	698
30	1782	1080	742
31	1897	1150	790
32	2022	1226	842
33	2157	1307	898
34	2301	1394	958

35	2361	1431	983
36	2524	1530	1051
37	2625	1591	1093
38	2735	1658	1139
39	2927	1774	1219
40	3057	1852	1273

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.