Table of Contents

1 Introduction

MW6 MaxiCode Win32 DLL can create device independent 2D MaxiCode images for your application, you can save the MaxiCode as either BMP or WMF image file or copy MaxiCode WMF image to the clipboard.

MaxiCode is a fixed-sized 2D symbology created by the United Parcel Service, it can store about 93 characters of information and is primarily used for freight sortation and tracking.

2 Installation

2.1 Trial Version

1. UnZip MW6MaxiCodeWin32.ZIP, run the setup.exe to install MaxiCode Win32 DLL.

2. The trial version MaxiCode Win32 DLL displays the "MW6 Demo" on the top of MaxiCode barcode.

2.2 Full Version

1. Uninstall the trial version MaxiCode Win32 DLL if applicable.

2. UnZip full version MaxiCode Win32 DLL .zip file and run the setup.exe to install the full version MaxiCode Win32 DLL.

3 How to Distribute It

If you want to redistribute MaxiCode Win32 DLL as part of your application, on the target machine, simply put **MaxiCodeWin32.dll** into the windows 32-bit system folder (e.g. "c:\windows\system32" or "c:\winnt\system32") for 32-bit Windows OS, or the SysWow64 folder (e.g. "c:\windows\SysWow64) for 64-bit Windows OS.

4 Office 2007

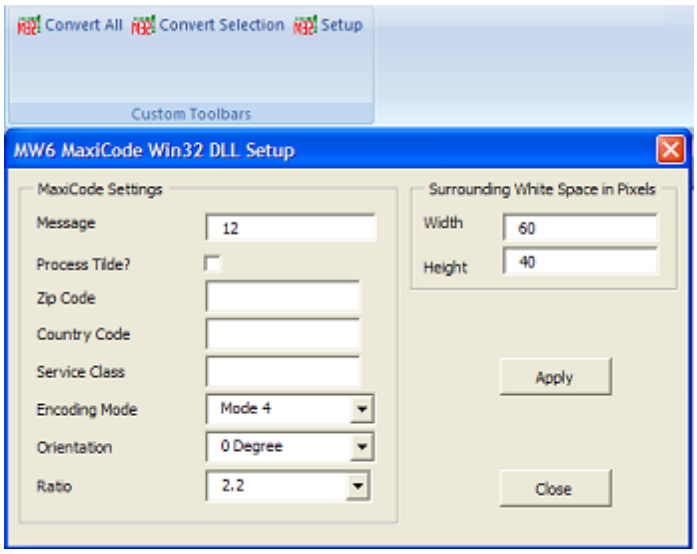
4.1 Word

4.1.1 Install Template File

1. Locate the Office Word STARTUP folder, which usually is "C:\Documents and Settings\<user name> \Application Data\Microsoft\Word\STARTUP".

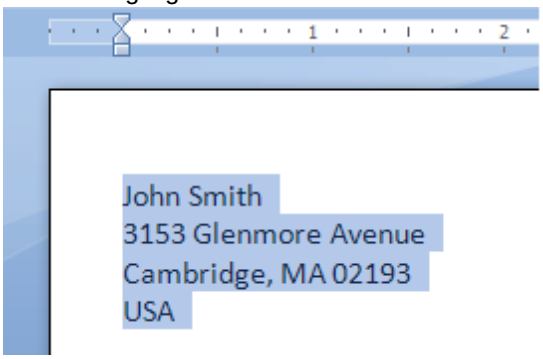
2. Copy MW6_MaxiCode_Win32.dotm to this folder.

3. Click on "**Add-Ins**", then click on "**Setup**", change the configurations for MaxiCode format, click on "**Apply**" button to allow the changes to take effect.

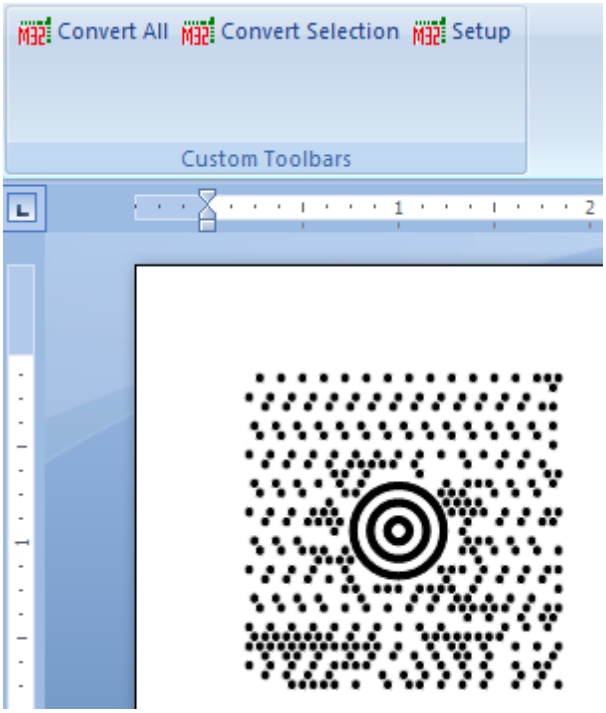


4.1.2 Create Single Barcode

1. Enter a few strings line by line and highlight them.

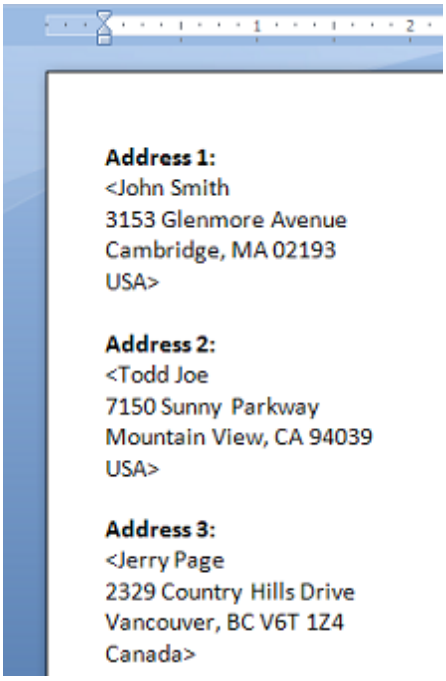


2. Click on "**Add-Ins**", then click on "**Convert Selection**" to create a MaxiCode barcode.

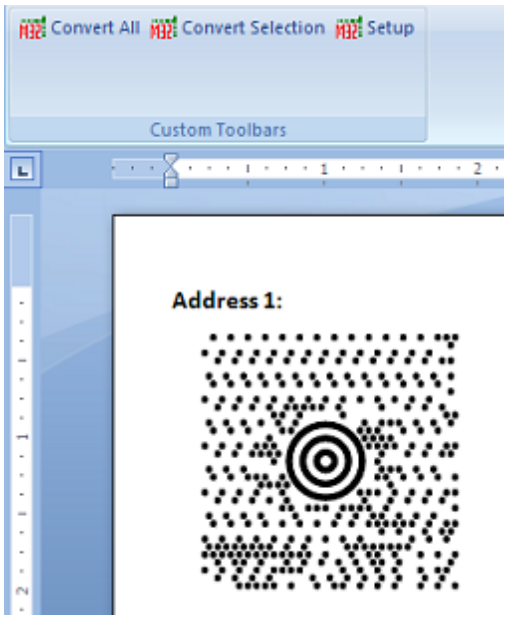


4.1.3 Create Multiple Barcodes

1. Enter a few string sections, surround those sections which will be converted to the barcodes with the "<" and ">" characters.

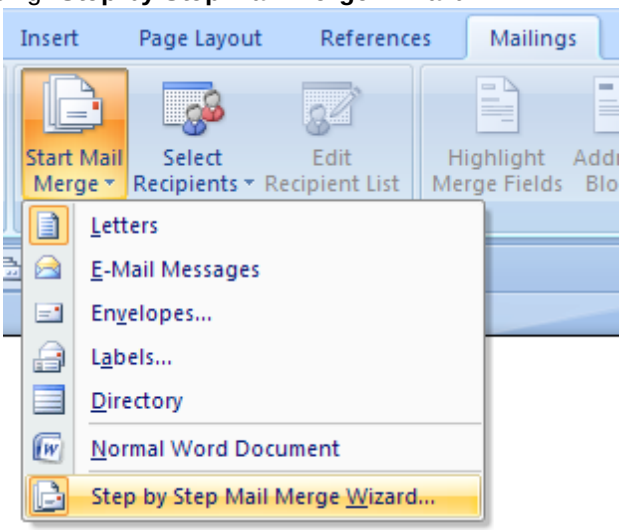


2. Click on "**Add-Ins**", then click on "**Convert All**" to create MaxiCode barcodes for the string sections surrounded with the "<" and ">" characters.

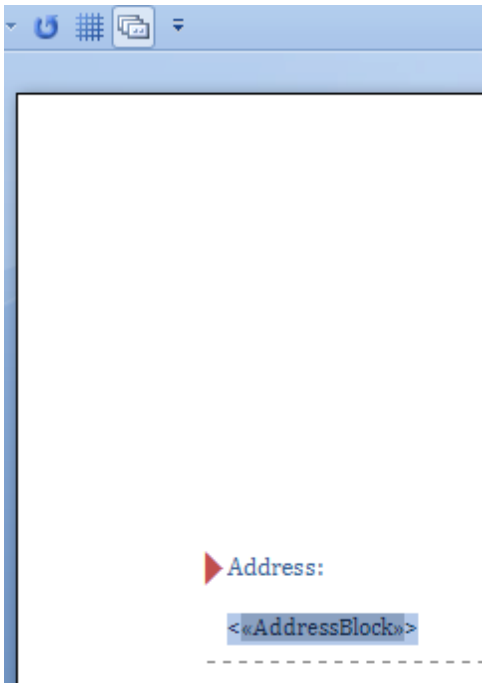


4.1.4 Mail Merge

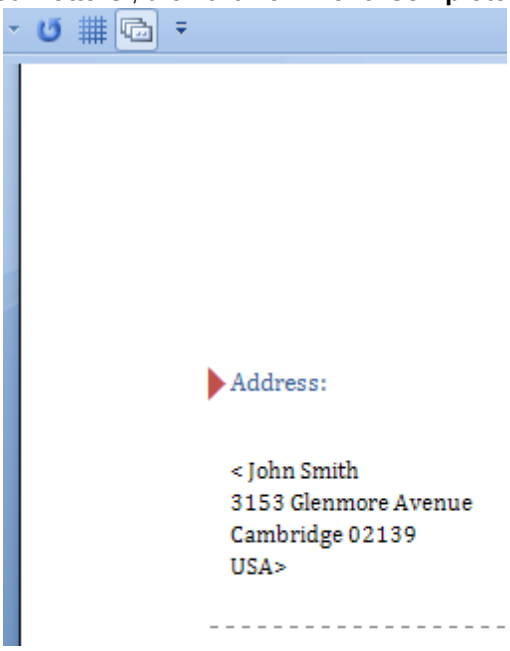
1. Create a mail merge using "**Step by Step Mail Merge Wizard**".



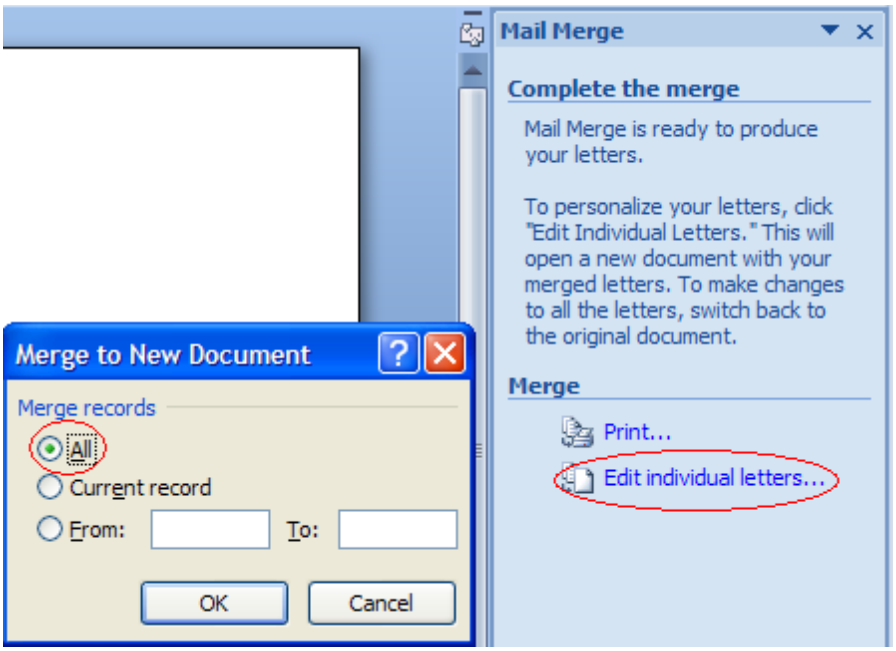
2. Surround the section which will be converted to MaxiCode barcode with the "<" and ">" characters and highlight it.



3. Click on "**Next: Preview your letters**", then click on "**Next: Complete the merge**".



4. Click on "**Edit individual letters**", this opens "**Merge to New Document**" dialog, click on "**All**" and then click on "**OK**" button.



5. Click on "**Add-Ins**", then click on "**Convert All**" to create MaxiCode barcodes.

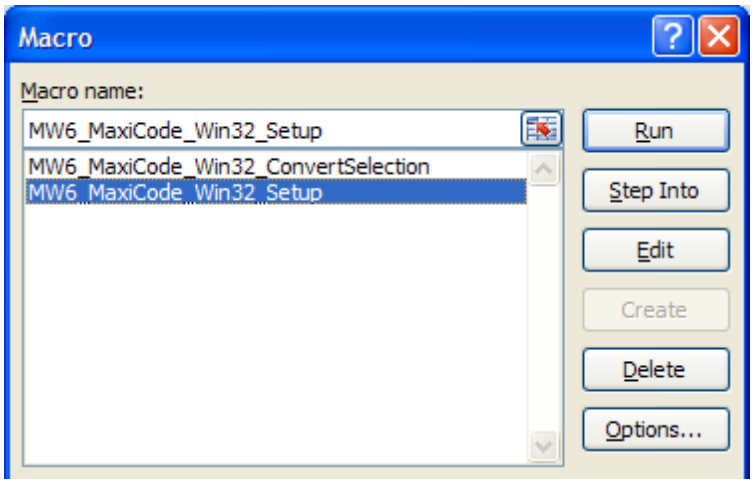
4.2 Excel

4.2.1 Change Settings

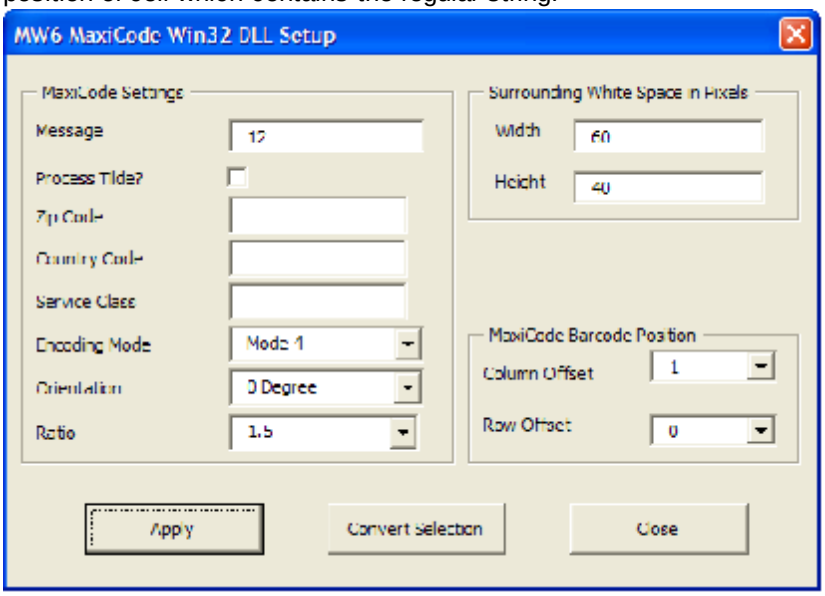
1. In Excel, open MW6_MaxiCode_Win32.xlsm.

2. If you see "**Security Warning, Macros have been disabled**", click on "**Options**" to open "**Microsoft Office Security Options**" dialog, toggle on "**Enable this content**" check box.

3. Click on "**Developer**" > "**Macros**", select "**MW6_MaxiCode_Win32_Setup**".

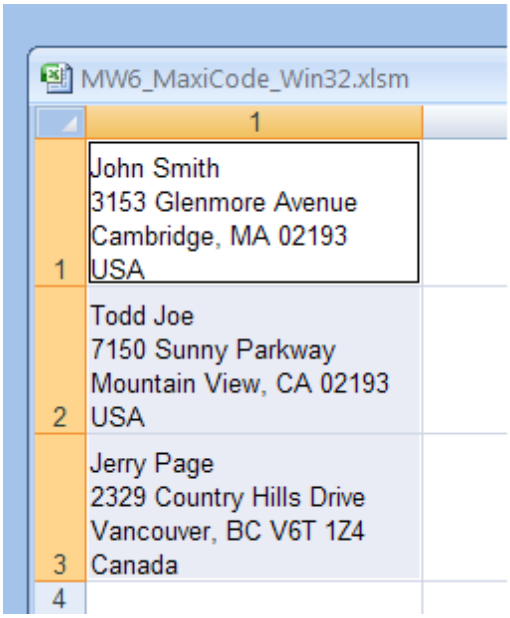


4. Choose a few appropriate values for MaxiCode configurations, click on "**Apply**" button to allow the changes to take effect, "**Column Offset**" and "**Row Offset**" are used to specify the barcode position relative to the position of cell which contains the regular string.

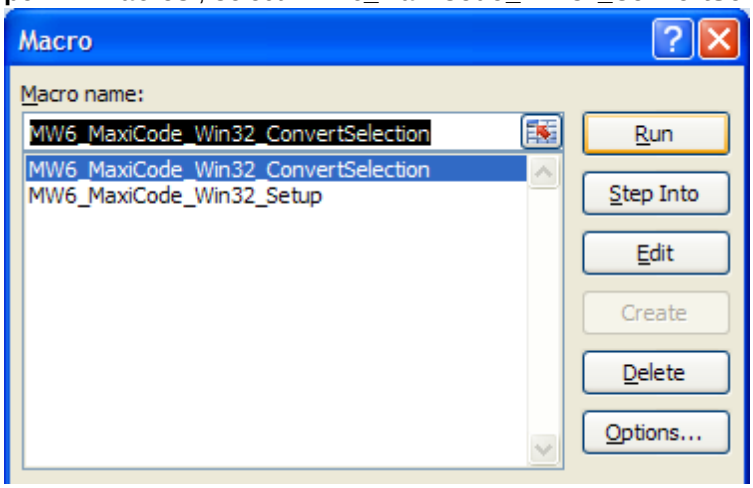


4.2.2 Create Multiple Barcodes

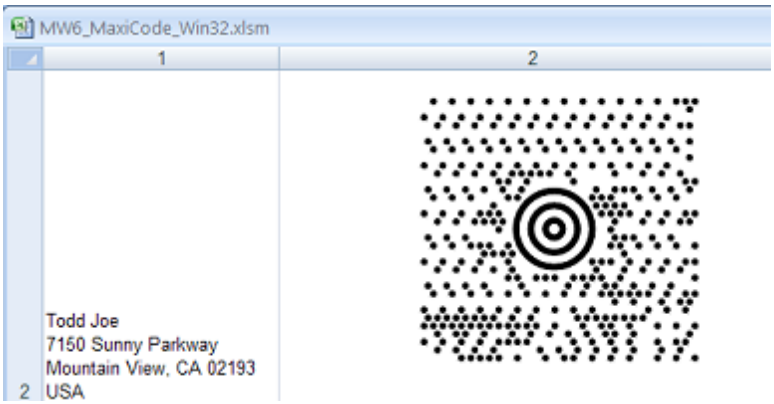
1. Select a few cells.



2. Click on "**Developer**" > "**Macros**", select "**MW6_MaxiCode_Win32_ConvertSelection**".



3. Click on "**Run**" to create the barcodes for the selected cells.



5 Office 2000 & 2003

5.1 Word Demo

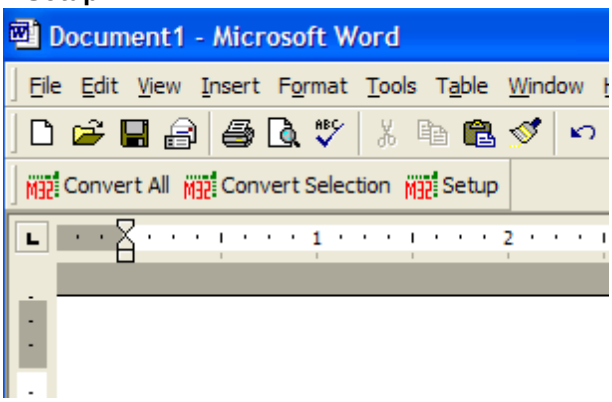
5.1.1 Install Template File

1. Locate the Word Startup folder, the Startup folder can be found in the following locations:

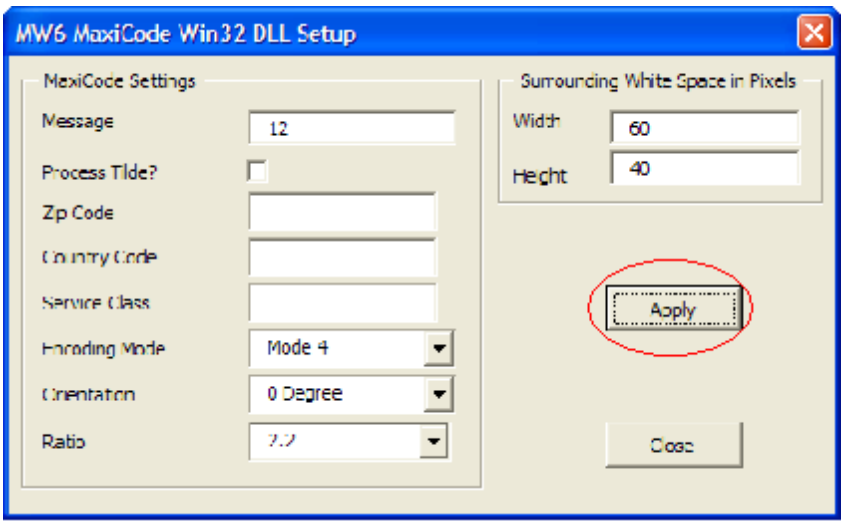
OS	Location
Windows 2000/XP	"C:\Documents and Settings\<user name>\Application Data\Microsoft\Word\Startup"
Windows NT4	"C:\Winnt\Profiles\<user name>\Application Data\Microsoft\Word\Startup"
Windows 95, 98, ME	Office XP: "C:\Program Files\Microsoft Office\Office10\Startup" Office 2000/97: "C:\Program Files\Microsoft Office\Office\Startup"

2. Copy MW6_MaxiCode_Win32.dot, which usually is in the folder "c:\Program Files\MW6 Win32 DLL \MaxiCode", to the Word Startup folder.

3. Open up Word, click on "**Setup**".

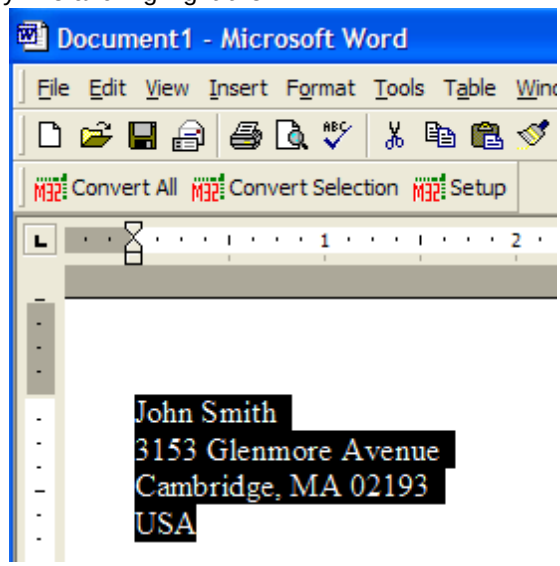


4. Choose a few appropriate values for MaxiCode configurations, click on "**Apply**" button to allow the changes to take effect.

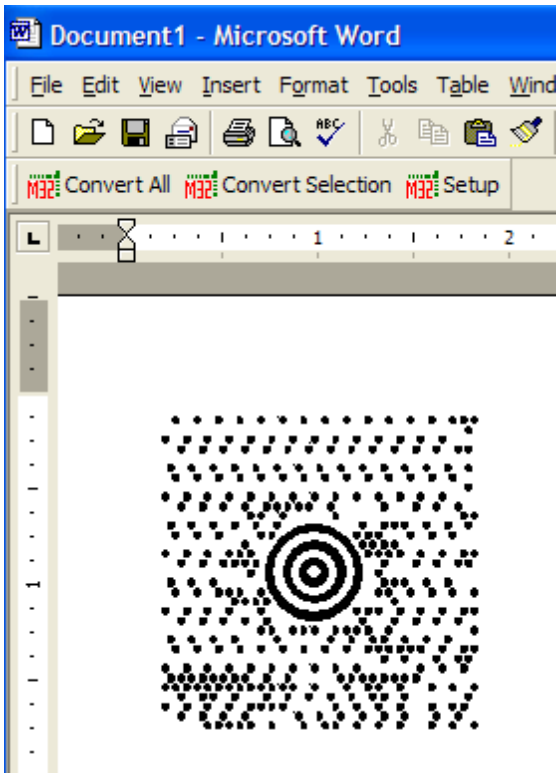


5.1.2 Create Single Barcode

1. Enter a few strings line by line and highlight them.

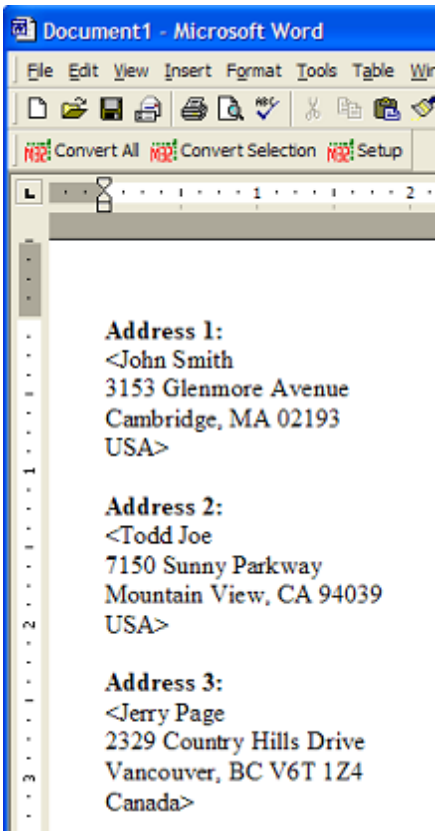


2. Click on "**Convert Selection**" to create a MaxiCode barcode.

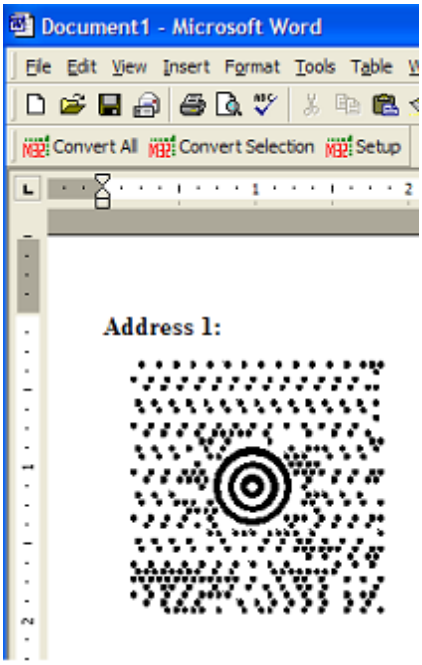


5.1.3 Create Multiple Barcodes

1. Enter a few paragraphs, surround those paragraphs which will be converted to MaxiCode barcodes with the "<" and ">" characters.

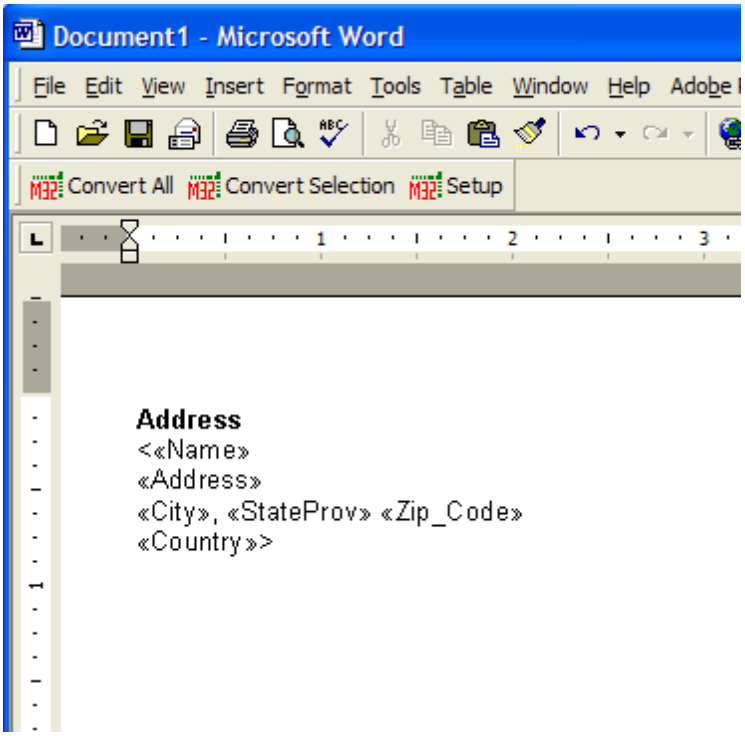


2. Click on "**Convert All**" to create MaxiCode barcodes for the paragraphs surrounded with the "<" and ">" characters.

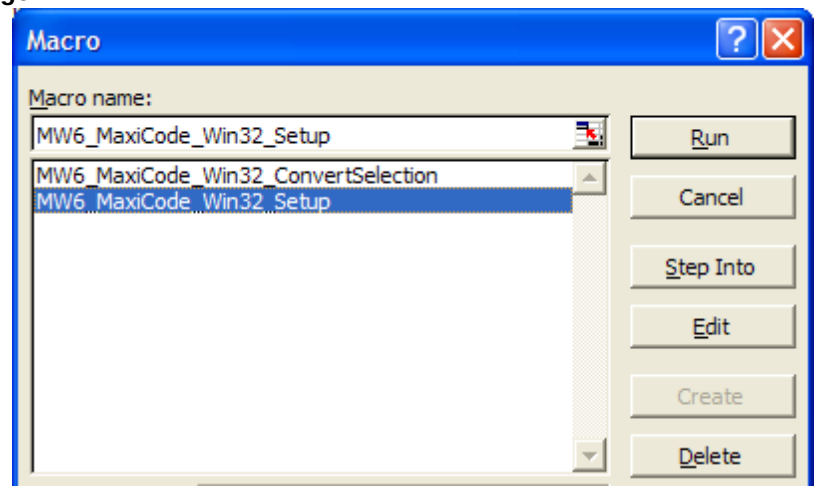


5.1.4 Mail Merge

1. In Mail Merge, surround the paragraphs which will be converted to MaxiCode barcodes with the "<" and ">" characters.



2. Click on "**Merge ...**"



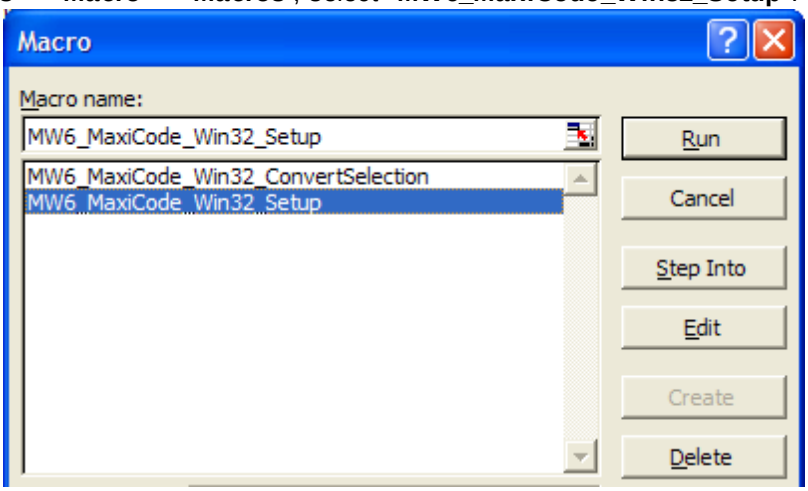
3. Click on "**Convert All**" to create MaxiCode barcodes for the paragraphs surrounded with the "<" and ">" characters.

5.2 Excel Demo

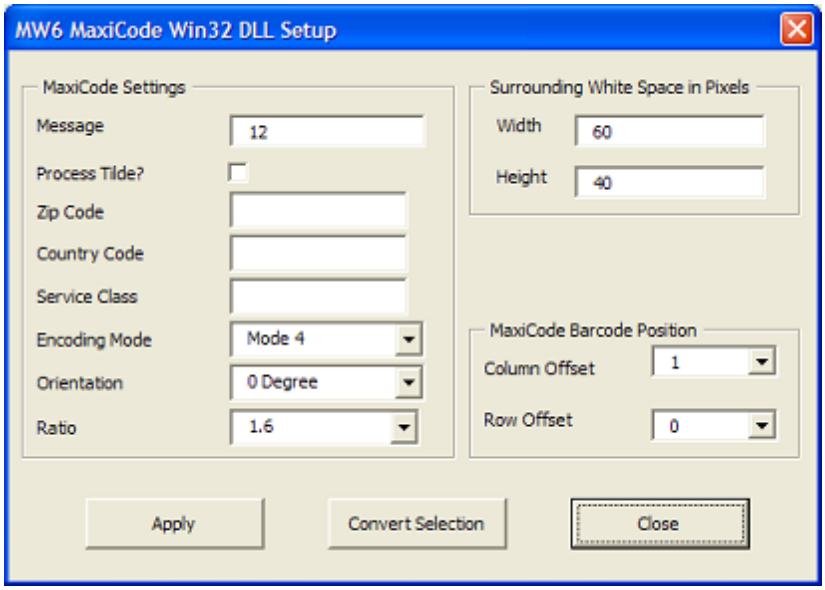
5.2.1 Change Settings

1. In Excel, open MW6_MaxiCode_Win32.XLS.

2. Click on "**Tools**" > "**Macro**" > "**Macros**", select "**MW6_MaxiCode_Win32_Setup**".



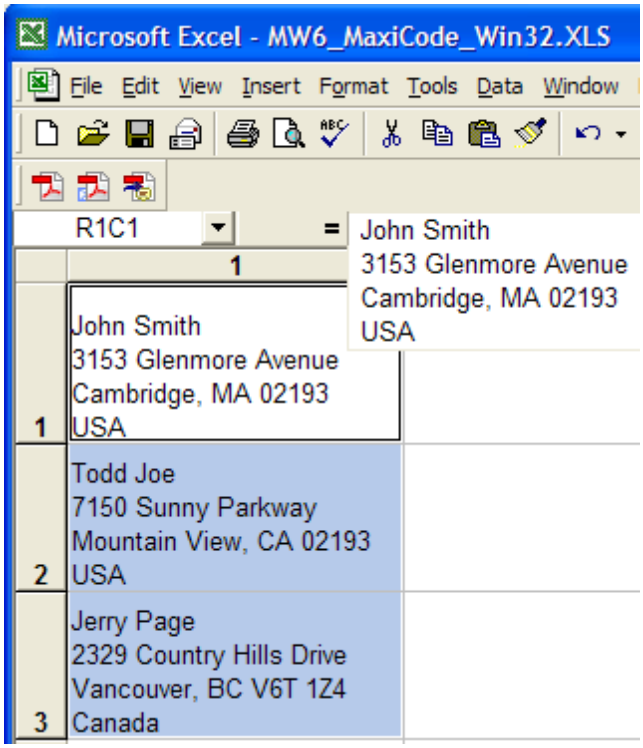
3. Click on "**Run**".



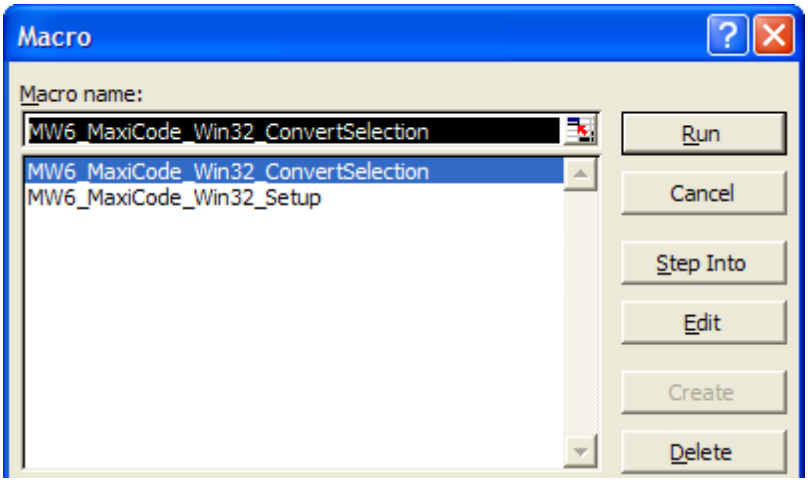
4. Choose a few appropriate values for MaxiCode configurations, click on "**Apply**" button to allow the changes to take effect, "Column Offset" and "Row Offset" are used to specify MaxiCode barcode position relative to the position of the cell which contains the regular string.

5.2.2 Create Multiple Barcodes

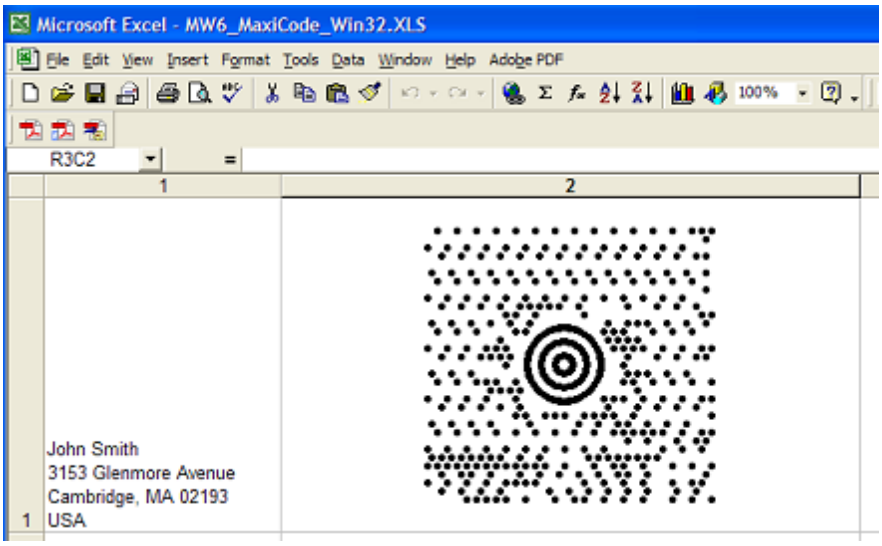
1. Select a few cells.



2. Click on "**Tools**" > "**Macro**" > "**Macros**", select "**MW6_MaxiCode_Win32_ConvertSelection**".



3. Click on "**Run**" to create MaxiCode barcodes for the selected cells.



6 Reference Guide

6.1 MCAppearance Function

Sets up the parameters of MaxiCode barcode appearance.

```
void MCAppearance(double Ratio, WORD Orientation);
```

Parameters

Ratio

Specifies the ratio value which is used to enlarge or shrink the MaxiCode barcode. The default value is 1, a valid value must be between 0.7 and 3.

Orientation

Specifies the orientation of the MaxiCode barcode, this parameter can be one of the following values:

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

6.2 MCConfigure Function

Sets up the parameters of MaxiCode barcode.

```
void MCConfigure(WORD Mode, LPCTSTR ZipCode, LPCTSTR CountryCode, LPCTSTR ServiceClass);
```

Parameters

Mode

Indicates which encoding mode is used, this parameter can be one of the following values.

Value	Description
0	Mode 2
1	Mode 3
2	Mode 4
3	Mode 5

ZipCode

Specifies the zip code of MaxiCode barcode.

CountryCode

Specifies the 3-digit country code of MaxiCode barcode.

ServiceClass

Specifies the 3-digit service class of MaxiCode barcode.

HandleTilde

Indicates whether to process the tilde character "~" or not, if this parameter is set to TRUE, non-printable characters can be passed to the library 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).

See Also

Special Format Message

6.3 MCCopyToClipboard Function

Copies the MaxiCode barcode WMF format image into the system clipboard.

```
BOOL MCCopyToClipboard();
```

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use MCGetActualSize() function to obtain the actual size of the MaxiCode barcode and use MCSetSize() function to set the image size by adding surrounding white space around the MaxiCode barcode.

See Also

MCGetActualSize() Function | MCSetSize() Function

6.4 MCGetActualSize Function

Gets the actual size of the MaxiCode barcode which is rendered onto either computer screen or other devices such as printers.

```
void MCGetActualSize(BOOL ScreenIsTarget,
                     DWORD TargetHDC,
                     DWORD *ActualWidth,
                     DWORD *ActualHeight);
```

Parameters

ScreenIsTarget

Indicates whether the MaxiCode barcode is rendered onto computer screen or not.

TargetHDC

Device context on which to render the MaxiCode barcode, if the parameter *ScreenIsTarget* is set to TRUE, set this parameter to NULL.

ActualWidth

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

ActualHeight

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

6.5 MCRender Function

Renders the MaxiCode barcode onto the device such as computer screen or printers.

```
void MCRender(DWORD hDC, WORD x, WORD y);
```

Parameters

hDC

Device context on which to render the MaxiCode barcode.

x

The x coordinate, in pixels, of the top left corner of the MaxiCode barcode .

y

The y coordinate, in pixels, of the top left corner of the MaxiCode barcode.

6.6 MCSaveAsBMP Function

Saves the MaxiCode barcode image as a BMP file.

```
BOOL MCSaveAsBMP(LPCTSTR FileName);
```

Parameters

FileName

A string that contains the name of the file to which to save BMP format MaxiCode barcode image.

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use MCGetActualSize() function to obtain the actual size of the MaxiCode barcode and use MCSetSize() function to set image size by adding surrounding white space around the MaxiCode barcode.

See Also

MCGetActualSize() Function | MCSetSize() Function

6.7 MCSaveAsWMF Function

Saves the MaxiCode barcode image as a WMF file.

```
BOOL MCSaveAsWMF(LPCTSTR FileName);
```

Parameters

FileName

A string that contains the name of the file to which to save WMF format MaxiCode barcode image.

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use MCGetActualSize() function to obtain the actual size of the MaxiCode barcode and use MCSetSize() function to set image size by adding surrounding white space around the MaxiCode barcode.

See Also

MCGetActualSize() Function | MCSetSize() Function

6.8 MCSetBackColor Function

Specifies the RGB triplet of the background color.

```
void MCSetBackColor(WORD red, WORD green, WORD blue);
```

Parameters

red

Specifies the value of red component for a RGB triplet, the valid value should be between 0 and 255.

green

Specifies the value of green component for a RGB triplet, the valid value should be between 0 and 255.

blue

Specifies the value of blue component for a RGB triplet, the valid value should be between 0 and 255.

6.9 MCSetBarColor Function

Specifies the RGB triplet of the bar color.

```
void MCSetBarColor(WORD red, WORD green, WORD blue);
```

Parameters

red

Specifies the value of red component for a RGB triplet, the valid value should be between 0 and 255.

green

Specifies the value of green component for a RGB triplet, the valid value should be between 0 and 255.

blue

Specifies the value of blue component for a RGB triplet, the valid value should be between 0 and 255.

6.10 MCSetDefault Function

Initializes the MaxiCode barcode parameters with the default values.

```
void MCSetDefault();
```

6.11 MCSetMessage Function

Specifies the message to encode using the appropriate parameters.

```
void MCSetMessage(LPCTSTR Message);
```

Parameters

Message

A string that contains the message to encode using the appropriate parameters.

6.12 MCSetSize Function

Sets the size of the image which contains the MaxiCode barcode.

```
void MCSetSize(DWORD Width, DWORD Height);
```

Parameters

Width

The width, in pixels, of the image.

Height

The height, in pixels, of the image.

Remarks

First call MCGetActualSize() function to obtain the actual size of the MaxiCode barcode, then use this function to set image size by adding surrounding white space around the MaxiCode barcode.

See Also

MCGetActualSize() Function

6.13 MCSetStructuredAppend Function

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

```
void MCSetStructuredAppend(BOOL AllowSA, WORD SymbolID, WORD SymbolCount);
```

Parameters

AllowSA

Indicates whether the structured append is allowed in the current MaxiCode 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 8, which means that up to 8 symbols can be linked together using the structured append protocol.

Remarks

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

7 Special Format Message

If the "Message" parameter is properly formatted and begins with the 7 characters "[)><RS>01<GS>", ZipCode, CountryCode and ServiceClass properties will be automatically overridden.

For example, let's set the "Data" property to the following value:

[)><RS>01<GS>9615238<GS>840<GS>001<GS>AIM, Inc<GS>634 Alpha Drive<GS>Pittsburgh<GS>PA<RS><EOT>

In this format, the identifier "[)><RS>01<GS>" is followed by a date (YY), in this example, it is "96".

The above data is encoded in a particular manner as follows:

1) The first 9 data characters [)><RS>01<GS>YY are extracted

2) The next 3 data elements separated by <GS>, representing respectively the zip code, country code and service class, are extracted and encoded in the primary message. In this example, they are 15238, 840 and 001, so Zipcode, Country and ServiceClass properties are automatically overridden.

3) The remaining string preceded with [)><RS>01<GS>YY is encoded in the secondary message. In this example, it is

[)><RS>01<GS>YYAIM, Inc<GS>634 Alpha Drive<GS>Pittsburgh<GS>PA<RS><EOT>

Remarks:

<RS>, <GS> and <EOT> indicate 3 characters with ASCII values 30, 29 and 4 respectively.

8 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 User License allows the use of the software on ONE computer by ONE person in your organization.

* The Site License allows the use of the software at exactly 1 physical site by up to 10,000 users in your organization.

* The Single Developer License allows 1 developer in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties, **each individual developer requires a separate Single Developer License as long as he or she needs access to MW6's product(s) and document(s)**.

* The 2 Developer License allows 2 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 3 Developer License allows 3 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 4 Developer License allows 4 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 5 Developer License allows 5 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The Unlimited Developer License allows unlimited number of developers in your organization the royalty-free distribution (unlimited number of users) of the software to the third parties.

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.