Version

X7

ENGRAVEPRO, LLC.

EngravePro for CorelDRAW®

User Manual

ENGRAVEPRO FOR CORELDRAW®

User Manual

Email: sales@engravepro.com

Last Revised: May 12, 2014

Table of Contents

Software License Agreement

PLEASE READ THIS SOFTWARE LICENSE AGREEMENT CAREFULLY BEFORE DOWNLOADING, INSTALLING OR USING ENGRAVEPRO, LLC. SOFTWARE. BY DOWNLOADING OR INSTALLING THE SOFTWARE, YOU ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, THEN DO NOT DOWNLOAD, INSTALL OR USE THE SOFTWARE.
The following terms govern your use of the Software except to the extent a particular program (a) is the subject of a separate written agreement with EngravePro, LLC. or (b) includes a separate "click-on" license agreement as part of the installation process.

License. Subject to the terms and conditions of and except as otherwise provided in this Agreement, EngravePro, LLC. ("EPLLC") and its suppliers grant to Customer ("Customer") a nonexclusive and nontransferable license to use the specific EPLLC program modules, feature set(s) or feature(s) for which Customer has paid the required license fees (the "Software"), in object code form only. In addition, the foregoing license shall also be subject to each of the following limitations: Customer's use of the Software shall be limited to use on a single hardware chassis, on a single central processing unit, as applicable, or use on such greater number of chassis or central processing units as Customer may have paid EPLLC the required license fee; and NOTE: For evaluation or beta copies for which EPLLC does not charge a license fee, the above requirement to pay a license fee does not apply.

General Limitations. Except as otherwise expressly provided under this Agreement, Customer shall have no right, and Customer specifically agrees not to: (i) transfer, assign or sublicense its license rights to any other person, and any such attempted transfer, assignment or sublicense shall be void; (ii) make error corrections to or otherwise modify or adapt the Software or create derivative works based upon the Software, or to permit third parties to do the same; or (iii) decompile, decrypt, reverse engineer, disassemble or otherwise reduce the Software to human-readable form to gain access to trade secrets or confidential information in the Software.
Upgrades and Additional Copies. For purposes of this Agreement, "Software" shall include (and the terms and conditions of this Agreement shall apply to) any upgrades, updates, bug fixes or modified versions (collectively, "Upgrades") or backup copies of the Software licensed or provided to Customer by EPLLC or an authorized distributor for which Customer has paid the applicable license fees. NOTWITHSTANDING ANY OTHER PROVISION OF THIS AGREEMENT: (1) CUSTOMER HAS NO LICENSE OR RIGHT TO USE ANY SUCH ADDITIONAL COPIES OR UPGRADES UNLESS CUSTOMER, AT THE TIME OF ACQUIRING SUCH COPY OR UPGRADE, ALREADY HOLDS A VALID LICENSE TO THE ORIGINAL SOFTWARE AND HAS PAID THE APPLICABLE FEE FOR THE UPGRADE; (2) USE OF UPGRADES IS LIMITED TO EPLLC EQUIPMENT FOR WHICH CUSTOMER IS THE ORIGINAL END USER PURCHASER OR LESSEE OR WHO OTHERWISE HOLDS A VALID LICENSE TO USE THE SOFTWARE WHICH IS BEING UPGRADED; AND (3) USE OF ADDITIONAL COPIES IS LIMITED TO BACKUP PURPOSES ONLY.

Proprietary Notices. Customer agrees to maintain and reproduce all copyright and other proprietary notices on all copies, in any form, of the Software in the same form and manner that such copyright and other proprietary notices are included on the Software. Except as expressly authorized in this Agreement, Customer shall not make any copies or duplicates or any Software without the prior written permission of EPLLC. Customer may make such backup copies of the Software as may be necessary for Customer's lawful use, provided Customer affixes to such copies all copyright, confidentiality, and proprietary notices that appear on the original.

Protection of Information. Customer agrees that aspects of the Software and associated documentation, including the specific design and structure of individual programs, constitute trade secrets and/or copyrighted material of EPLLC. Customer shall not disclose, provide, or otherwise make available such trade secrets or copyrighted material in any form to any third party without the prior written consent of EPLLC. Customer shall implement reasonable security measures to protect such trade secrets and copyrighted material. Title to Software and documentation shall remain solely with EPLLC.

Limited Warranty. If Customer obtained the Software directly from EPLLC, then EPLLC warrants that during the Warranty Period (as defined below): (i) the media on which the Software is furnished will be free of defects in materials and workmanship under normal use; and (ii) the Software will substantially conform to its published specifications. The "Warranty Period" means a period beginning on the date of Customer's receipt of the Software and ending on the later of (a) ninety (90) days from the date of initial shipment of the Software by EPLLC, or (b) the end of the minimum period required by the law of the applicable jurisdiction. Except as expressly granted in this Agreement, the Software is provided AS IS. EPLLC does not warrant that the Software is error free or that Customer will be able to operate the Software without problems or interruptions. This warranty does not apply if the Software (a) is licensed for beta, evaluation, testing or demonstration purposes for which EPLLC does not receive a license fee, (b) has been altered, except by EPLLC, (c) has not been installed, operated, repaired, or maintained in accordance with instructions supplied by EPLLC, (d) has been subjected to abnormal physical or electrical stress, misuse, negligence, or accident, or (e) is used in ultra-hazardous activities. If Customer obtained the Software from a EPLLC reseller, the terms of any warranty shall be as provided by such distributor, and EPLLC provides Customer no warranty with respect to such Software.

Disclaimer of Warranties. EXCEPT AS SPECIFIED IN THIS WARRANTY, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS, AND WARRANTIES INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OR CONDITION OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, SATISFACTORY QUALITY OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE, ARE HEREBY EXCLUDED TO THE EXTENT ALLOWED BY APPLICABLE LAW. TO THE EXTENT AN IMPLIED WARRANTY CANNOT BE EXCLUDED, SUCH WARRANTY IS LIMITED IN DURATION TO THE WARRANTY PERIOD. BECAUSE SOME STATES OR JURISDICTIONS DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM JURISDICTION TO JURISDICTION.

Disclaimer of Liabilities. IN NO EVENT WILL EPLLC OR ITS SUPPLIERS BE LIABLE FOR ANY LOST REVENUE, PROFIT, OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR PUNITIVE DAMAGES HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE EVEN IF EPLLC OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event shall EPLLC's or its suppliers' liability to Customer, whether in contract, tort (including negligence), or otherwise, exceed the price paid by Customer. The foregoing limitations shall apply even if the above-stated warranty fails of its essential purpose. BECAUSE SOME STATES OR JURISDICTIONS DO NOT ALLOW LIMITATION OR EXCLUSION OF CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.

Term and Termination. This Agreement is effective until terminated. Customer may terminate this Agreement at any time by destroying all copies of Software including any documentation. Customer's license rights under this Agreement will terminate immediately without notice from EPLLC if Customer fails to comply with any provision of this Agreement. Upon termination, Customer must destroy all copies of Software in its possession or control.

Export. Software, including technical data, may be subject to U.S. export control laws, including the U.S. Export Administration Act and its associated regulations, and may be subject to export or import regulations in other countries. Customer agrees to comply strictly with all such regulations and acknowledges that it has the responsibility to obtain licenses to export, re-export, or import Software.

Restricted Rights. EPLLC's commercial software and commercial computer software documentation is provided to United States Government agencies in accordance with the terms of this Agreement, and per subparagraph "(c)" of the "Commercial Computer Software - Restricted Rights" clause at FAR 52.227-19 (June 1987). For DOD agencies, the restrictions set forth in the "Technical Data-Commercial Items" clause at DFARS 252.227-7015 (Nov 1995) shall also apply.

General. This Agreement shall be governed by and construed in accordance with the laws of the State of Florida, United States of America, as if performed wholly within the state and without giving effect to the principles of conflict of law. If any portion hereof is found to be void or unenforceable, the remaining provisions of this Agreement shall remain in full force and effect. EPLLC hereby specifically disclaims the UN Convention on Contracts for the International Sale of Goods. Except as expressly provided herein, this Agreement constitutes the entire agreement between the parties with respect to the license of the Software and supersedes any conflicting or additional terms contained in the purchase order.

Acknowledgement

With the release of EngravePro X7 you will find that EP now supports QR Codes. We would like to acknowledge the fantastic work done by QRCode.Net and their developers for providing us with the code to implement QR Codes in our software. To respect their licensing requirement we present their licensing agreement pertaining to the QR Code software below:

The MIT License (MIT)

Introduction

EngravePro is designed to enhance CorelDRAW® for the modern day Laser Engraver by offering tools otherwise afforded to Rotary Engravers through custom software applications. It is our hope EngravePro meets the needs of Laser Engravers worldwide by providing these valuable tools and by enhancing an already World Class Software Application… CorelDRAW®

EngravePro was built on a foundation of ideas from many users. EngravePro, LLC values its customers and their ideas. If you have an idea on how EngravePro can better your experience with CorelDRAW please feel free to let us know. Email us at ideas@engravepro.com

What's New!

Version 16

- **Completed the transition from Visual Basic for Applications to .Net**
- **Added QR Code Support**
- **Added the ability to save preferences to a backup file**

Version 15

- **Aligned EngravePro version number with CorelDRAW for clarity**
- **Fixed various bugs**
- **Redesigned user interface**
- **Converted to .NET**

Version 5.7

- **Bar codes** – EngravePro now does 1D ad 2D barcodes.
- **Date and time token added to VariableText.**
- **Invert Selection** – Invert your selection of currently selected items in CorelDRW.
- **Multiple changes for Windows Vista**.
- **Preferences dialog added to VariableText**.
- **Help files updated**.
- **Size selected objects to width or height of currently selected objects**.

Version 5.0

- **Circle Copy** – Copy selected objects into a circular pattern.
- **Tile Objects** – Tiles selected objects into a grid like pattern according to your spacing specifications.
- **Copy Plate** – Copy's the selected object and places it next to its parent. Like tiling only you press the Page Down key on your keyboard. Keep pressing the key and it continues to copy, moving to the next line and even the next page if necessary.
- **Make Grid** – Create a grid based upon user input. Now you can set the cell size, number of rows and columns to create a professional grid. Grid is also optimized for vector cutting.
- **Convert to Square** – Convert selected object(s) into a square.

- **Convert to line** – Convert selected nodes into a straight line. Useful for cleaning up scanned images. Also removes unnecessary nodes.
- **Reset Object** – Resets an object back to its original state after you have made sizing changes.
- **Show Direction** – Shows you graphically the direction your engraver will take when cutting out an object.
- **Make Entry Point** – Let's you change the starting point on an object. Used for cutting and plotting.
- **Cut Marks** – Create guidelines on your page for trimming paper printouts.

Version 4.1

- **Move Rulers to Home** – Automatically reset your rulers to the zero in the upper left corner of the page.
- **Move Objects to Home** – Move selected or all objects to the upper left corner of your page.
- **Flip Page Horizontally and Vertically** – Flips the page in either direction. Useful for mirroring an entire page for reverse engraving jobs.
- **Rotate Page Left and Right** – Rotates your page to the left or right, similar to change from portrait to landscape but moves all your objects on the page as well.
- **Size Selected Objects** – Sizes all selected objects to the last selected object.
- **Convert to Circle** – Converts the selected object to a circle in the size of the selected object.
- **Convert to Rectangle** – Converts the selected object to a rectangle in the size of the selected object.
- **Count Characters** – Counts characters in your document minus the spaces. Displays count based upon selected text, text on the current page and text in the entire document.
- **Set to Master** – Sets the current layout to the layout size of your engraver. For example you could create a 1 x 3 name tag by setting the page size to 1 x 3 then when you are finished laying it out choose this option and it will set your page size to the actual engraver table size.
- **Variable Text** – Added the option to import fixed width and character delimited text files. You now have the option to create your plates top to bottom as well as left to right. You can also now group plates automatically during the import. Added the last five files list to open commonly used files more quickly.

- **Copy** - You now have the option to create your plates top to bottom as well as left to right.
- **Tool Bar –** Added a toolbar to CorelDraw.

Prerequisites

Important! Please read and ensure you meet the following pre-requisites prior to continuing. Doing so will ensure a successful installation.

First you must have **CorelDRAW® version X7** installed. Visual Basic for Applications (VBA) should have been installed when you installed CorelDRAW®. In the event Visual Basic for Applications (VBA) is not installed you must re-run CorelDRAW setup.

1. If you currently have a prior version of CorelDRAW installed you must uninstall it. Failing to do so will cause EngravePro not to work properly. Upon uninstalling, reboot your PC to ensure all files have been removed.
2. Install CorelDRAW, if not already installed. If your operating system is Win XP, Vista, Windows 7 or Windows 8, you must install CorelDRAW as a user with Administrator privileges.
3. Start CorelDRAW to initialize program defaults if you have just installed CorelDRAW for the first time. Opening CorelDRAW® will allow the EngravePro setup to continue.
4. Install any CorelDRAW service packs, if any. From the download page of EngravePro it will tell you what specific version and release number you need installed. Running the EngravePro setup will also alert you if you do not have the proper version and release.
5. Close CorelDRAW.
6. Reboot your PC.

EngravePro Installation

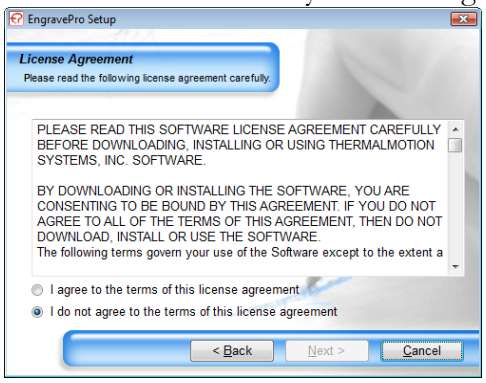
Assuming you have already installed Visual Basic for Applications, you are now ready to begin installation. What is important here is that you have CorelDRAW X7 AND Visual Basic for Applications installed. If you do not have at least those two items installed, the EngravePro setup may fail.

Please note, if you are using Win XP, Vista, Windows 7 or Windows 8 it will be necessary for you to be logged in as the administrator or as a user with administrator privileges for the machine you are installing EngravePro on. Run the setup by right clicking the setup file and clicking "Run as Administrator".

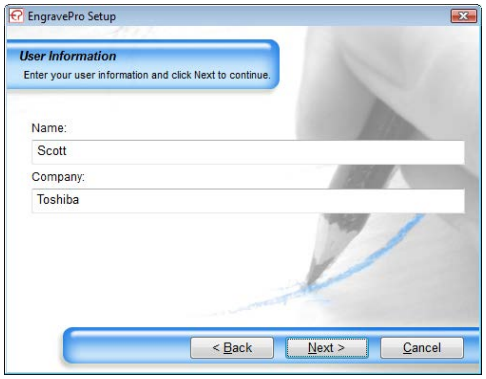
Let's begin the installation…

1. If you have not already done so, download the appropriate version of EngravePro from http://www.engravepro.com
2. Upon completing the download double click the setup file to begin installing EngravePro.
3. Upon setup starting you will see a welcome screen. Read the information and click the Next button.

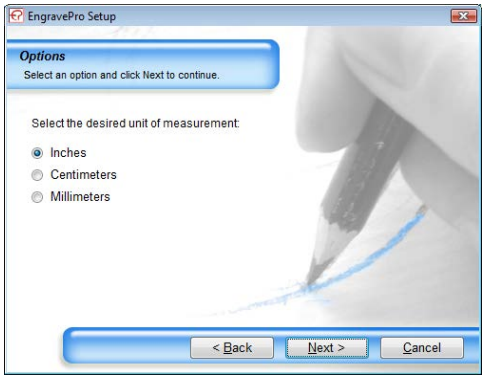
4. Read carefully the license agreement now being displayed. If you accept the license agreement click the "I agree to the terms of this license agreement" radio button then click the next button. If you do not agree click the cancel button and end the installation.



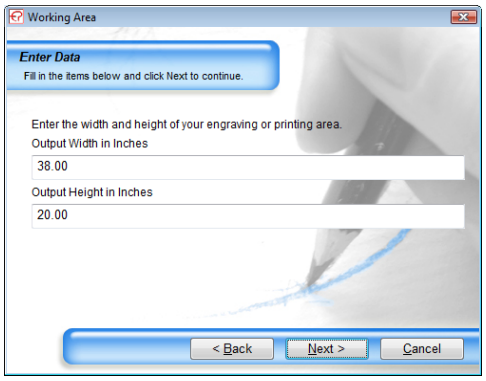
5. Setup will display a user information screen. Enter your information and click the next button.



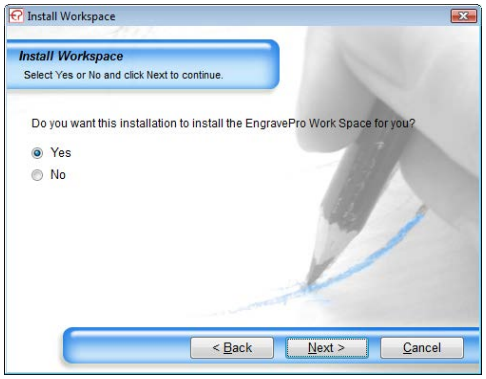
6. Next Setup will ask you the unit of measurement you want to use. Make your selection and then click the Next button.



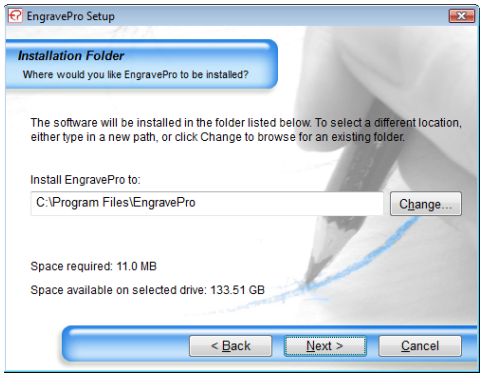
7. Next enter the width and height of the engraving or print area of your device. Press the Next button when you are done entering the requested values.



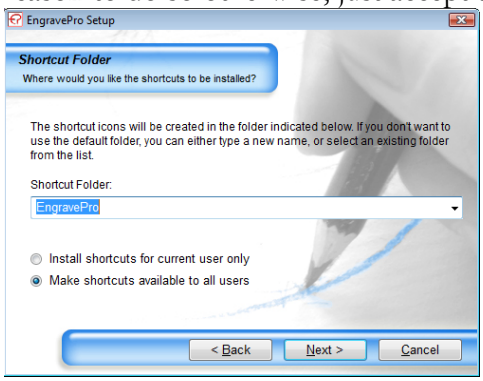
8. Setup will now ask you if you want to install the EngravePro workspace. This will change the default workspace to an EngravePro workspace. You can always import the workspace later if you want.



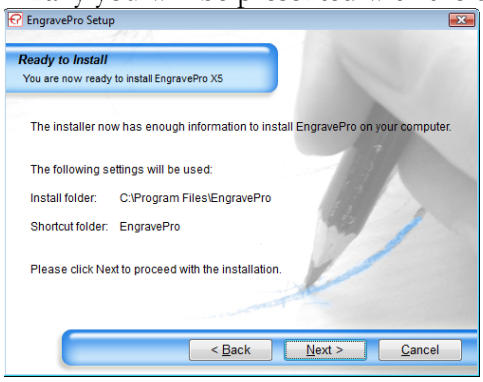
9. Setup will now display a default folder location in which to install EngravePro. Unless you have a specific reason not to, you should accept the defaults by clicking the Next button.



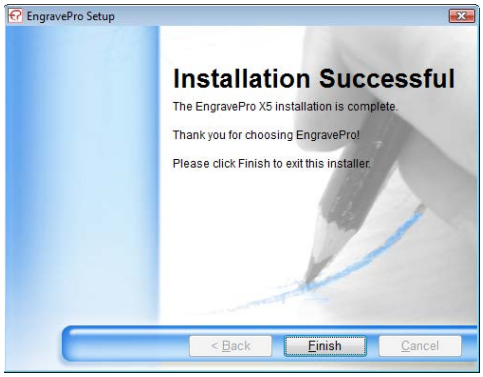
10. Next choose the shortcut menu you want to install EngravePro into. Unless you have a reason to do so otherwise, just accept the default choice shown. Click the next button.



11. Finally you will be presented with the dialog below, when ready click the next button.



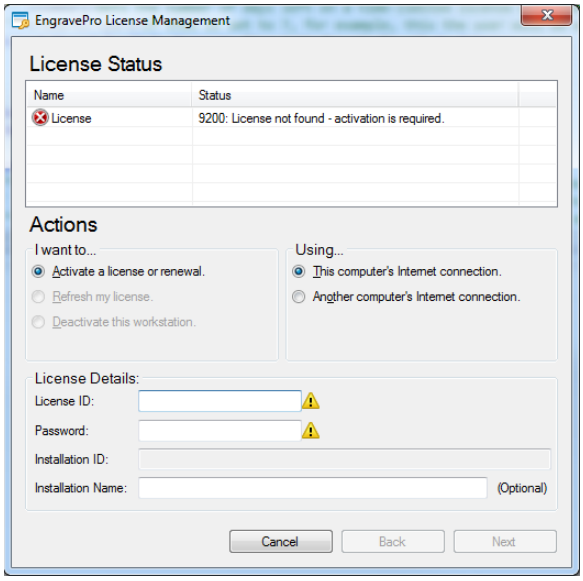
12. Once setup has completed you are ready to use EngravePro. Click the Finish button.



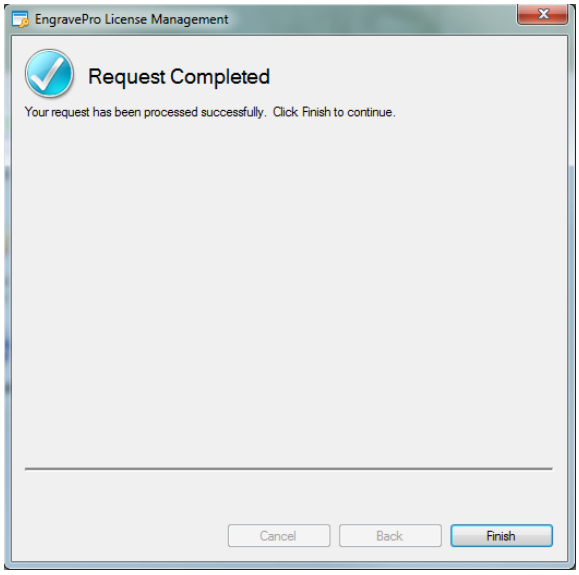
How to Register EngravePro

You have two options to register EngravePro; online or by email. Before registering you should have already purchased the software via our online store. With your receipt you would have received a License ID and password. You will need both the ID and the password to properly register and activate your software.

To start the registration or purchase EngravePro, click the Start Menu then right-click "Register EngravePro" from the EngravePro menu. Select "Run as Administrator". You will see a dialog box similar to the one below.



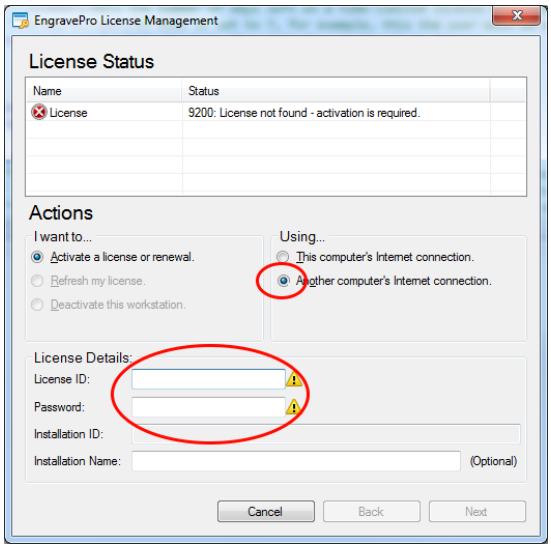
Enter the License ID and Password that was emailed to you and then click the Next button. In a few moments you will see a dialog like below indicating a successful registration.



Activating EngravePro Manually

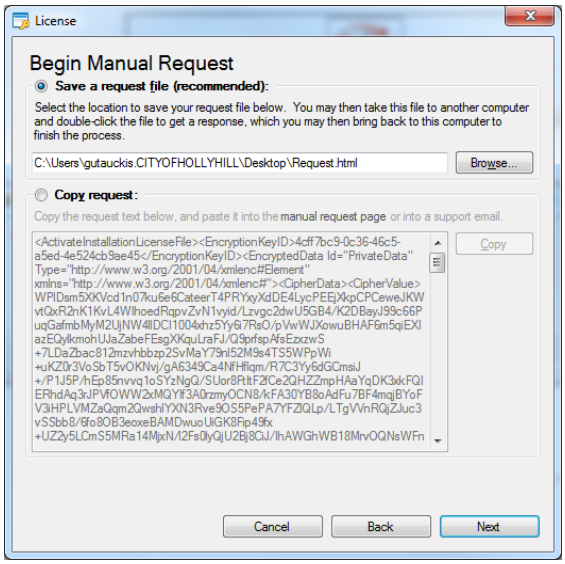
Your computer may not have internet access thus requiring you to activate manually.

To start the registration or purchase EngravePro, click the Start Menu then right-click "Register EngravePro" from the EngravePro menu. Select "Run as Administrator". You will see a dialog box similar to the one below.



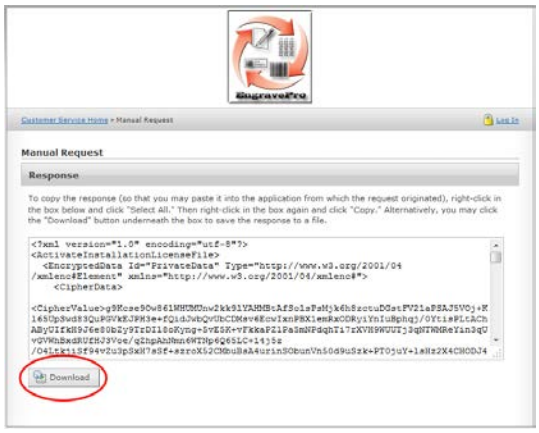
Enter the License ID and Password that was emailed to you. Select "Another computer's internet connection", then click the next button.

The following dialog box appears.



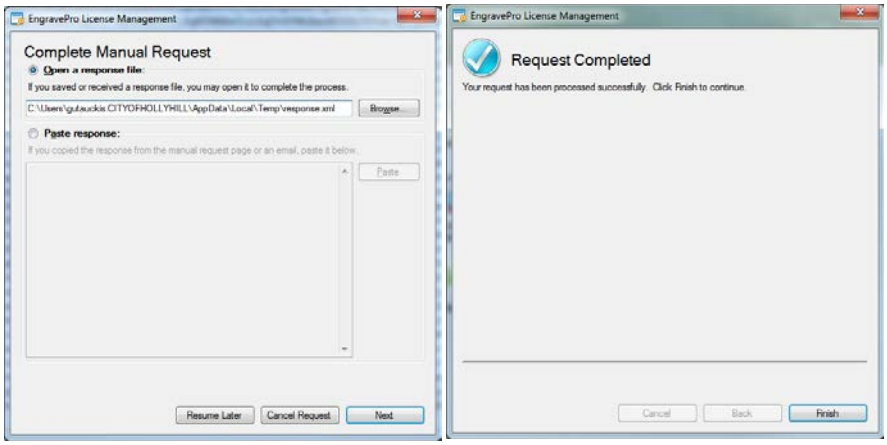
Save the request file somewhere you can upload it to the internet.

Take the file to a computer with internet access. Double click the file. Your browser will open up to a page where you can download the registration file.



Download the file and take it back to the PC you are trying to register.

Using the Browse button locate the response.xml file. Click the Next button.



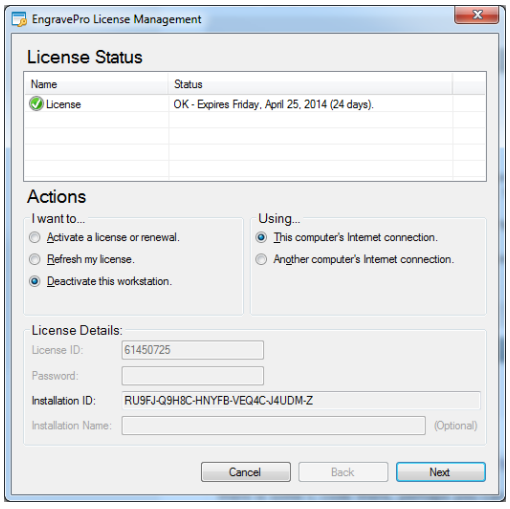
Upon a successful registration you will get a confirmation dialog.

Deactivating EngravePro

If you are changing PC's or any hardware within the current Pc it's a good idea to deactivate EngravePro. This will free up an activation opportunity and allow you to reinstall it either on the current PC or on another PC.

You may deactivate EngravePro from the PC it is currently installed on or you can deactivate it manually from another machine.
Select "Deactivate EngravePro from the Start Menu/EngravePro menu. A dialog box will appear.

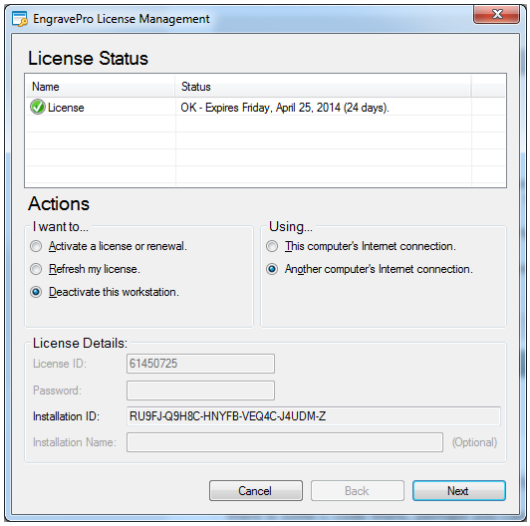
Select "Deactivate this workstation". Click the Next button.

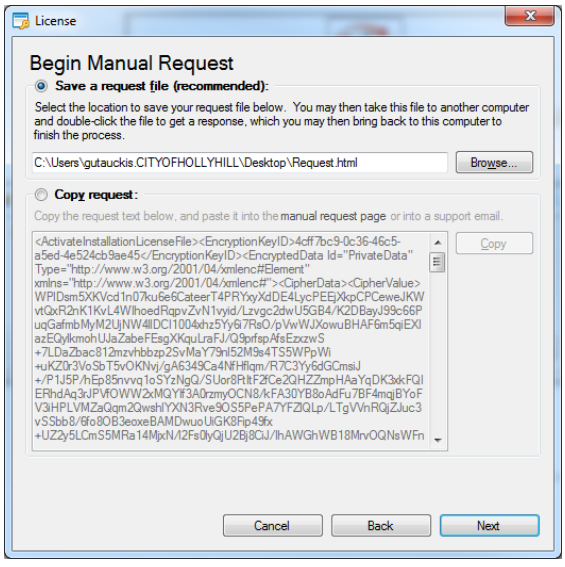


Upon a successful deactivation you will receive a confirmation message.

Deactivating EngravePro Manually

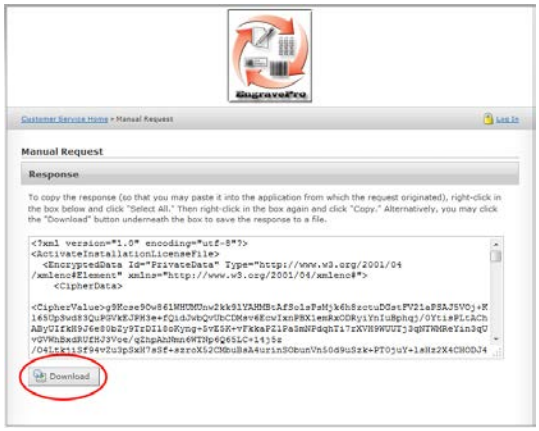
Select "Deactivate EngravePro from the Start Menu/EngravePro menu. A dialog box will appear.





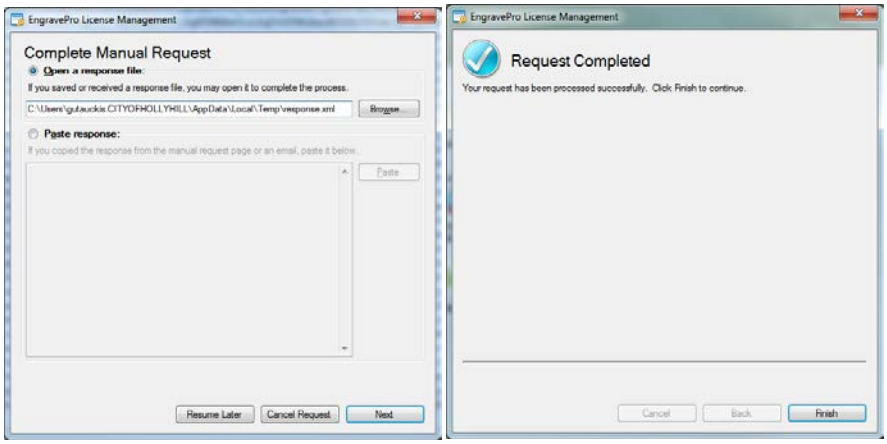
Save the request file somewhere you can upload it to the internet.

Take the file to a computer with internet access. Double click the file. Your browser will open up to a page where you can download the deactivation file.



Download the file and take it back to the PC you are trying to deactivate.

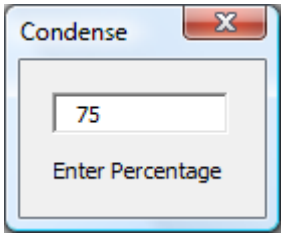
Using the Browse button locate the response.xml file. Click the Next button.



Upon a successful deactivation you will receive a confirmation message.

Auto Condense

Auto Condense will allow you to condense your text using percentages. Enter a percentage and press enter. With Auto Condense you can set all of your lines of text to the same condensing value. Note that you must enter a value greater than zero



Below are some examples of what can be done using Auto Condense. Using different values, you can achieve different effects.

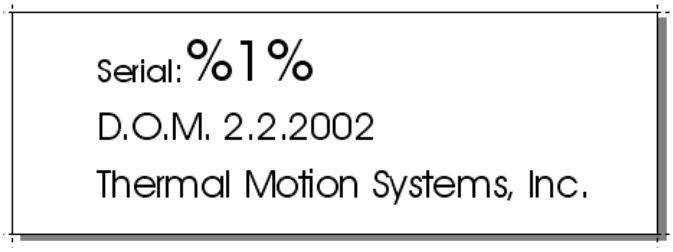
EngravePro @ 100%

EngravePro @ 80%

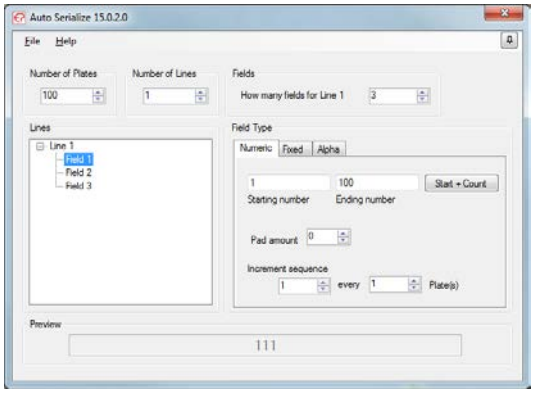
EngravePro @ 120%

Auto Serialize

Serial enables you to create sequential numbers for importing into Variable Text and finally into CorelDRAW. With Auto Serialize you can create complicated numbers based on Numbers, Letters and/or static text. Before you start you want to lay out your plate in CorelDRAW. Pictured below is a sample. You can open this sample by clicking **Sample 4** in the EngravePro folder located on your Start Menu.



Notice we only have one Variable Text line as denoted by the **%1%**.



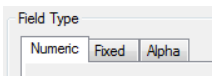
After setting up your plate and including the Variable Text line, click the EngravePro menu in CorelDRAW then click "Auto Serialize". A dialog box similar to the one on the left will appear. For this example set the Number of Plates to 100, the Number of Lines to 1 and the number of fields to 3.

Let's explain the different areas of the dialog and their functions.

This area gives you a graphical view of your job. For each line you specified in the previous dialog box you will have a separate line entry here. If you click on any of the fields, the field's values will be reflected in the corresponding fields to the right of the **Lines** area.

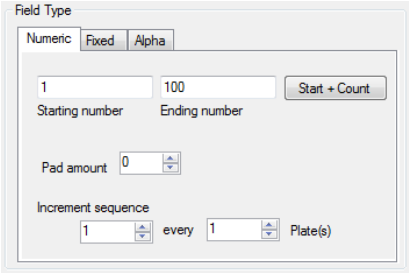
For each line you may have many fields.

Use the  to increase or decrease the number of fields for each of your lines. You may also type in a value. Notice how when you make a field selection in the **Lines** area, the **How many fields** changes to show you what line you are working on.



The **Field Type** tabs let you set what type the field will be as well as present you the options for that field type.

Your choices are:



Numeric – This field will contain only numbers. When you make this selection you will see the following:

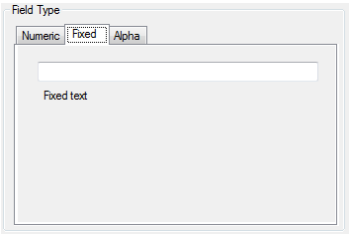
Starting Number is the number you want your first plate to start with.

Ending Number is the last number of your sequence. Usually this is the number of your last plate, however, if you want a particular number sequence to repeat itself then you would set this number to a value less then the number of plates. For example if you have 100 plates and you want this field to start over every 10 plates then you set this value to 10.

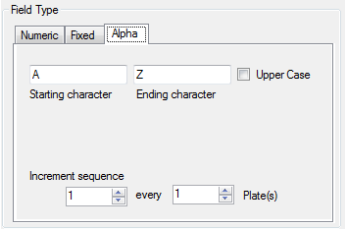
Start + Count automatically sets the ending number to the starting number plus the number of plates you specified in the first dialog. So for example if you set your number of plates to34 and you check the "Start + Count" checkbox, the ending number will automatically change to 34.

Pad Amount is the number of spaces you want to **pad** your value with zeros. For example if your ending number is 100 and you want all of your plates to have three digits you would enter a pad amount of 3. Now all plates, regardless of their number would have three digits in this field. So 1 would be .001, 2 would be .002 and so on.

Increment Sequence tells Auto Serialize how to increment your number value for this particular field. Usually this value is 1 every 1 plate. You can, for example, increment 1 for every two plates. Then for each two plates this field would have the same value.



Fixed – The field may contain any characters and will remain the same on all plates in this field position. Enter your desired text in the box. This field will be the same on all plates.



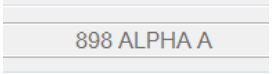
Alpha – Characters only. Alpha works the same as the Numeric field type except Alpha uses letters of the alphabet.

Starting Character is the character between "A" and "Z" you want to start with.

Ending Character is the last character you want to use in your sequence.

Upper Case will make your letters appear in capitals.

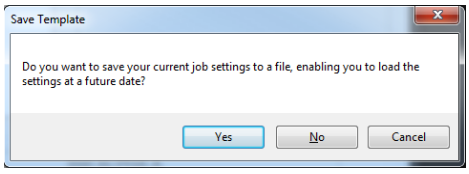
Increment Sequence tells Auto Serialize how to increment your alpha value for this particular field. Usually this value is 1 every 1 plate. You can for example, increment 1 for every two plates. Then for each two plates this field would have the same value.



Preview Line If you look towards the bottom of the dialog box you will notice a preview area. As you make changes to your line, the status field is updated to give you an accurate representation of what your line of text will look like.

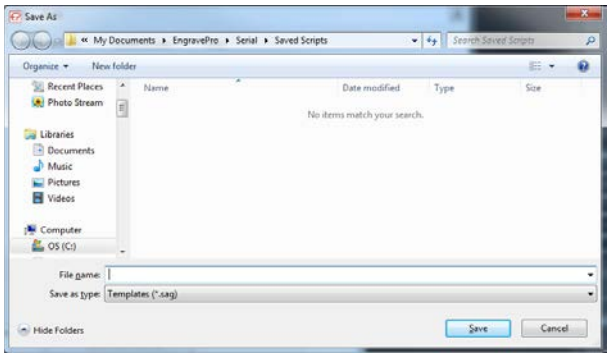
If you decide you do not want to run the job, click the **File Menu**, and then click **Exit**. Auto Serialize will close.

Once you have completed filling in your values and making changes click the **File Menu,** and then click the **Create Output File** menu item to continue with your job.



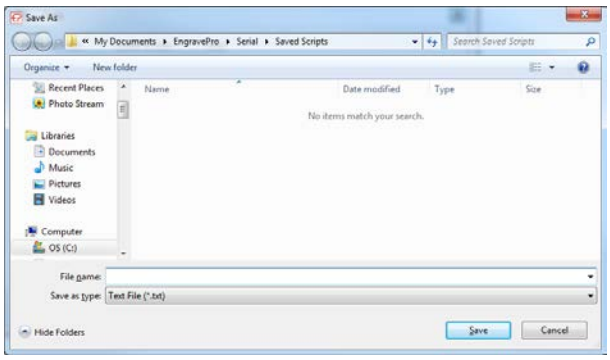
When you click the "Create Job" button, a message will appear like the one shown to the above. If you intend on running the same job again or would just like to save it, click the **Yes** button. A dialog box will appear asking you where to save the file and what to name it. The next time you run Auto Serialize you can click the **Open Script** button to load your job.

If you choose to save this job as a template then you will see the following dialog box.



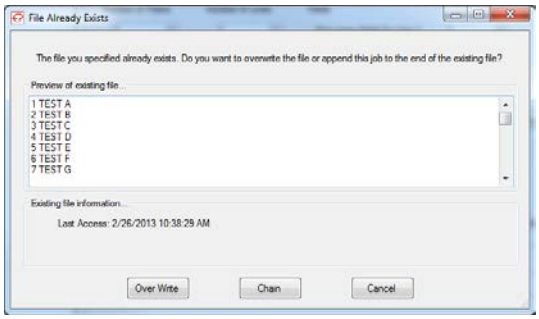
Enter the name of an existing file or the name of a new file to save your template to. Click Save once you have made your selection.

When you click the create job button, you will be presented with a File Save dialog box similar to the one shown below.



Either type in a new filename or choose an existing file name to save to. If you plan on "**Chaining**" multiple jobs together and this is the first link in your chain, then enter a new filename. If this is a subsequent job you want to add to an existing chain then choose that file. Once you have entered your selection click the Save button. See the section titled, "Chaining" for a full explanation of what chaining is and how to utilize it.

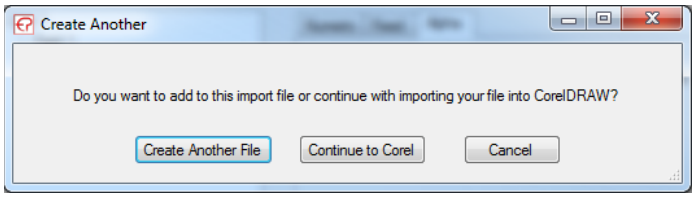
If you chose a file that already existed you will get the following dialog box.



The dialog will display the contents of the file you selected and display the date and time the file was last accessed. This should help you determine if this is the file you really want to overwrite

or Chain to. Choosing Overwrite will obviously overwrite the file you already selected, thereby erasing the contents of that file with the new job. If you select Chain, it will add the current job onto the end of the job you previously selected.

Once you have made your choice you will have the option of returning to Auto Serialize to create another job or continuing on with CorelDRAW.

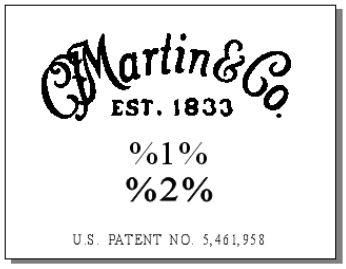


Chaining

Chaining is used to connect multiple Auto Serialize jobs together. For example, let's say you have 22 Guitars of one model and 26 Guitars of another model you want to serialize. You have a layout in CorelDRAW and you want to laser all 48 of the Guitars in one pass. Normally you would have to laser the first 22 then move your material to the upper left corner of your laser and then laser the remaining 26.

With Chaining you can create one text file you need for both of the layouts and laser both layouts together. For our example the first 22 Guitars have a Model Number of MN32D and the serial number will start with 004500. The second set of Guitars has a model number of DF75L and a starting serial number of 00232. To complete this job you would perform the following steps.

1. Start Auto Serialize from CorelDRAW where you have a job laid out like the one below:

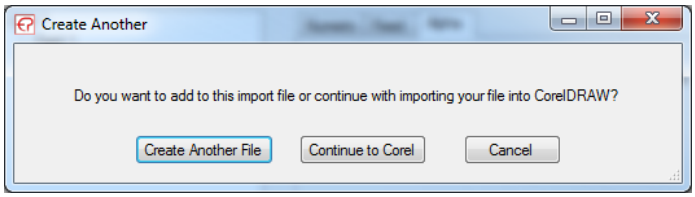


2. When Auto Serialize starts enter 22 for the number of plates and 2 for the number of lines.

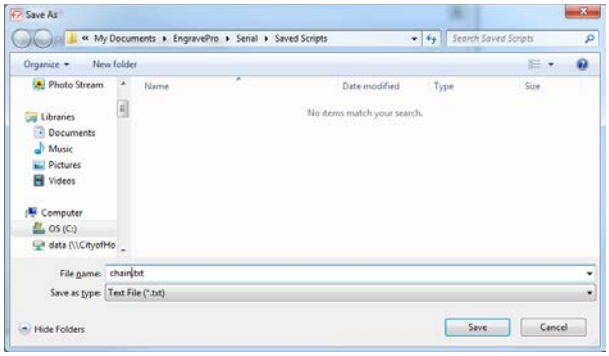
3. For Line 1: Field 1, choose the **Fixed Tab** for the field type and enter MN32D for the Fixed String.

4. Line 2: Field 1, choose the **Numeric Tab** for the field type, enter 4500 for the starting number, click the **Start + Count** Button and enter a 6 for the pad amount.

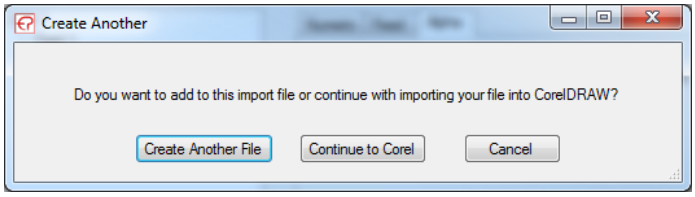
5. Now click the **File Menu,** and then click the **Create Output File** menu item to continue with your job.
6. A dialog prompts you as to whether or not you want to continue with CorelDRAW or create another file.



7. Click the Create another File button.
8. When the Job Save dialog comes up enter a new filename to save the text file to. For this example, enter chain.txt, then click the **Save** button.



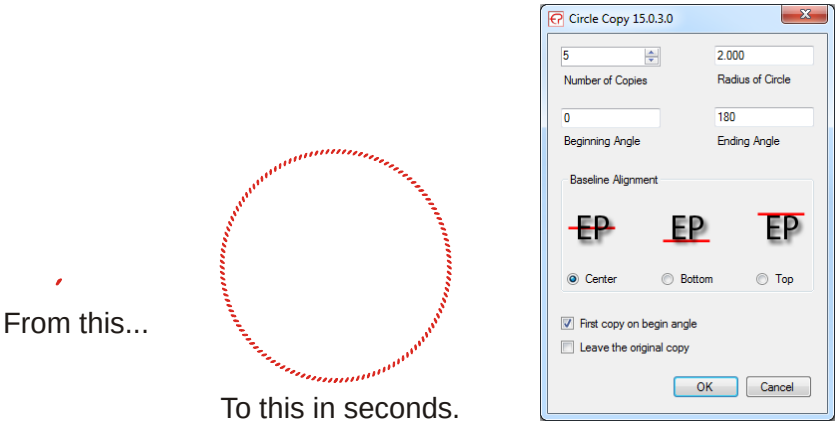
9. You are then asked if you want to save your current job settings. You may choose Yes, No or Cancel. For this example choose No.
10. Once again you are prompted for the number of plates and lines. Enter 26 for the number of plates. A 2 should already appear in the number of lines box.
11. Again for Line 1 Field 1 choose the **Fixed Tab** for the field type and enter the new model number of the guitar. In this case the new model number is DF75L.
12. Line 2: Field 1 choose the **Numeric Tab** for the field type, enter 232 for the starting number, click the **Start + Count** button twice then enter a 5 for the pad amount.
13. Now click the **File Menu,** and then click the **Create Output File** menu item to continue with your job.
14. A dialog prompts you as to whether or not you want to continue with CorelDRAW or create another file.



15. Click the **Continue to Corel** button.
16. You are then asked if you want to save your current job settings. You may choose Yes, No or Cancel. For this example choose No.
17. When you do, **Variable Text** will start where you can set your layout parameters, such as plate size, border and offsets. Continue with Variable Text. When complete you will have a CorelDRAW layout with multiple jobs combined in it. Below are a couple of the plates created with the above instructions.

Circle Copy

Circle Copy is designed to enable the engraver to easily create copies of objects into a circular fashion.



Number of Copies - Enter the number of copies you want to create from the selected original.

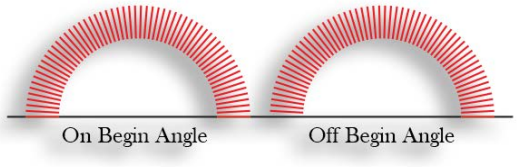
Radius of the Circle – Enter the radius of the circle you want your object to be copied onto.

Beginning Angle – Enter the beginning angle to start making copies.

Ending Angle – Enter the ending angle you want to stop making copies to.

Baseline Alignment – Allows you to specify how your objects will be positioned on the baseline of the circumference o f the circle.

First Copy on Begin Angle – Specifies you want the first copy to start on the begin angle and the last copy to be on the end angle.



Leave the Original Copy – If you check this box, Circle Copy will not delete the originally selected item you were copying from. Useful if you want to continue making more **Circle Copies** from the same original object.

Command Line

Allows you to enter commands to create object you would otherwise create using your mouse. Although it may be easier to draw a box, circle, etc with a mouse, positioning and sizing can be cumbersome. Command Line allows you to accurately draw the following objects:

Box – Circle – Line - Horizontal Guideline - Vertical Guideline

After typing in the command, press the enter key.

To Draw a Box

B x1(top), y1(left), x2(right), y2(bottom)
Where B (or b) is the Box command, x1, y1, x2, y2 are the coordinates given in your current units of the box where you want it drawn.
Example: B 0, 0, 10, 2
Draws a Box starting at 0, 0 and ending at 10, 2
Example: B „ 10, 2
Draws a Box starting at 0,0 and ending at 10,2 This is the same as the first example only this time omitting the zeros. If a blank value is left anywhere in the command it is replaced with a zero.

To Draw a Line

L x1(top), y1(left), x2(right), y2(bottom)
Where L (or l) is the Line command, x1,y1,x2,y2 are the coordinates given in your current units of the line where you want it drawn.
Example: L 0, 0, 10, 2
Draws a Line starting at 0,0 and ending at 10,2
Example: L „ 10, 2
Draws a Line starting at 0,0 and ending at 10,2
This is the same as the first example only this time omitting the zeros. If a blank value is left anywhere in the command it is replaced with a zero.

To Draw a Circle

C x(X Center), y(Y Center), radius
Where C (or c) is the circle command, x, y, radius are the coordinates given in your current units of the circle where you want it drawn.
Example: C 0, 0, 1
Draws a Circle centered at 0,0 with a radius of 1
Example: C „ 1

Draws the same circle

This is the same as the first example only this time omitting the zeros. If a blank value is left anywhere in the command it is replaced with a zero.

To Draw a Vertical Guideline

GV x(X Center)
Where GV (or gh) is the vertical guideline command, x is the coordinates given in your current units of the guideline where you want it drawn.

Example: GV 1
Draws a vertical guideline 1 unit from the left of your origin
Example: GV -1
Draws a vertical guideline 1 unit from the right of your origin

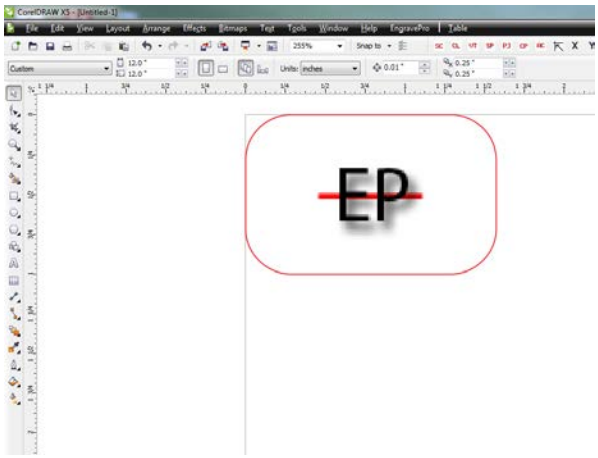
To Draw a Horizontal Guideline

GH y(Y Center)
Where GH (or gh) is the Horizontal guideline command, x is the coordinates given in your current units of the guideline where you want it drawn.
Example: GH 1
Draws a Horizontal guideline 1 unit up from your origin.
Example: GH -1
Draws a Horizontal guideline 1 unit down from your origin.

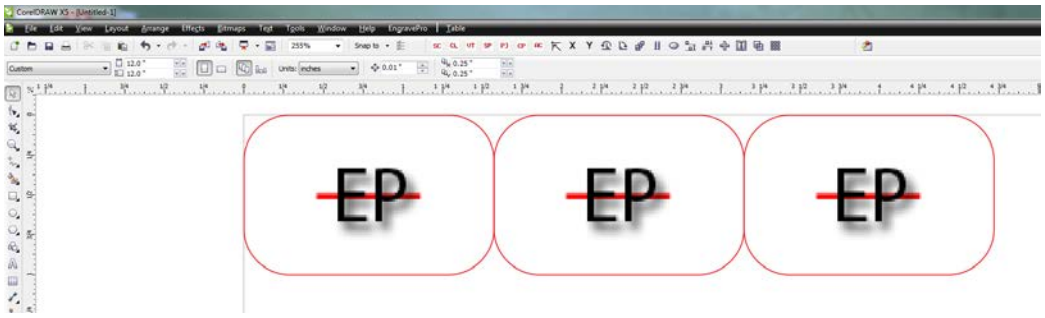
Copy Plate

Using the **PageDown** button on your keyboard, you can automatically create a copy of the selected object with one key stroke.

Start with your first "plate" in the upper right corner of your page like below. Make sure it is selected.



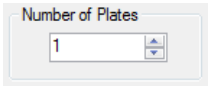
Now press the PageDown key on your keyboard. EngravePro will make as many copies as times you press the key. If EngravePro runs out of room on the line it automatically wraps to the next line. If EngravePro runs out of room on the page, it creates a new page and adds the plate to it.



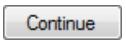
Copy Pro

CopyPro is designed to enable the engraver to easily position and copy plates for the purpose of matrix engraving. CopyPro allows the engrave to set margins and offsets for plate positioning as well as creating borders, both fixed and custom.

Let's describe the user interface first. Below you will find a table with each of the commands or value boxes named and to the right a short explanation of what it does.



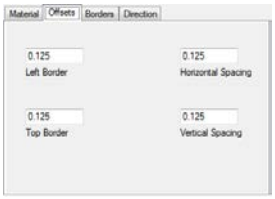
Tell CopyPro how many copies you want to make of your plate. A setting of one here will move the plate into the upper left corner of your matrix plate and create the appropriate border if selected, but will not duplicate any plates.



Click the Continue button to copy your plate based on your entered settings.

Use this button to set the size of your engraving table.

The **Material Tab** is where you set the height and width of your engraving material.

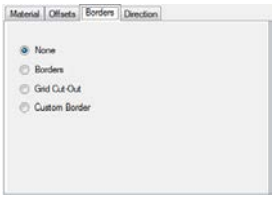


The **Offsets Tab** is where you set your left and top margin as well as you in-between space.

Left Border - The amount of space to put between the left edge of your Matrix Plate and the left edge of your first column of plates.

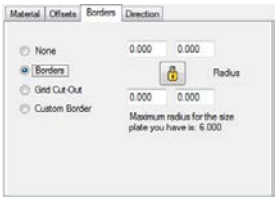
Top Border - The amount of space to put between the top edge of your Matrix Plate and the top edge of your first column of plates.

Horizontal Spacing - Amount of space to put between plates left to right.
Vertical Spacing - Amount of space to put between plates top to bottom.

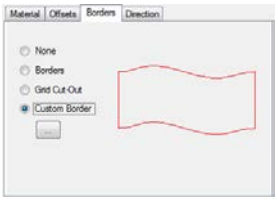


The Borders tab will let you define what type of borders or grid you want. You have a choice of None, Grid cutout, Borders or Custom. None and Grid cutout have no special settings other than to select them.

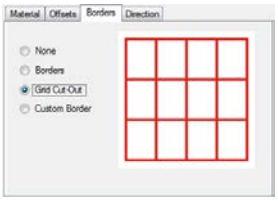
The other two are further defined below.



When you select **Borders**, you have the option of putting a square/rectangular border around your plate. Or, if you prefer you can set each corner to a radius. If you click the Lock/Unlock button and you are in Lock Mode, making a change to one corner will be reflected in all corners.

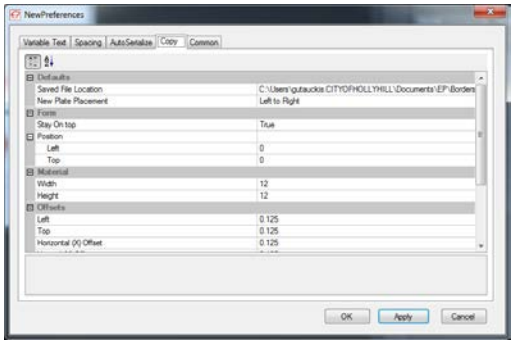


A **Custom Border** is one you create in CorelDRAW or any other drawing package capable of outputting Enhanced Windows Metafiles (EMF). The preview window to the right of the Choose button shows you what your border looks like. To set your border click the Choose button and select your desired border.



Grid Cut-Out will simply create a grid around your plates. This grid is optimized for cutter and laser paths to help reduce the amount of travel and time your machine takes to complete the job.

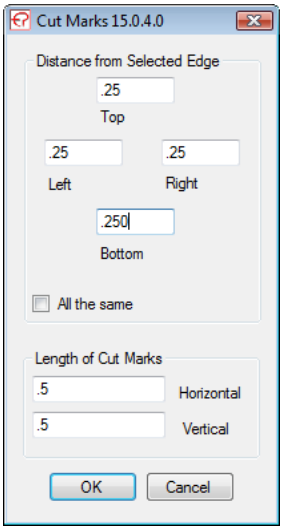
The **Direction Tab** lets you specify which direction you want CopyPro to create your plates.

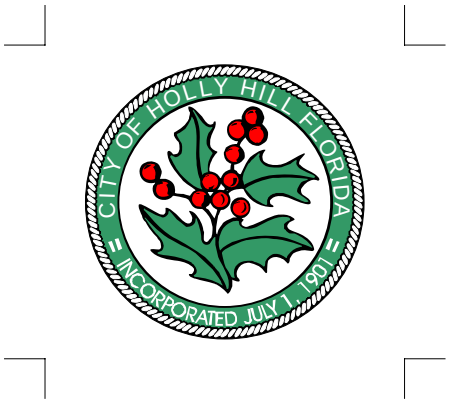


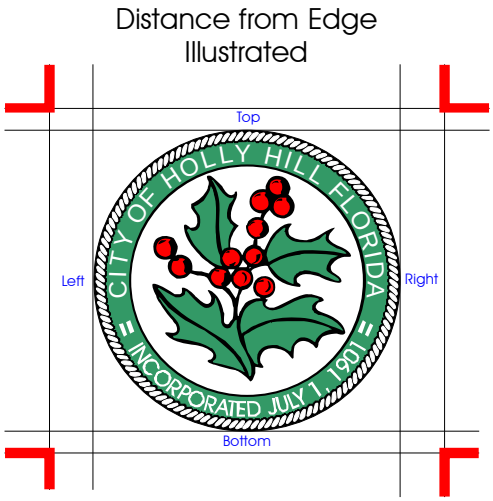
This is the Preferences dialog. You can set the size of your engraving table here by selecting the **Common Tab**.

Cut Marks

Cut marks are visual guides to help cut out objects from paper. To use cut Marks, select the object you want to frame with cut marks and then select Cut Marks from the EngravePro menu. For example using the settings in this dialog box illustration we get the following results:





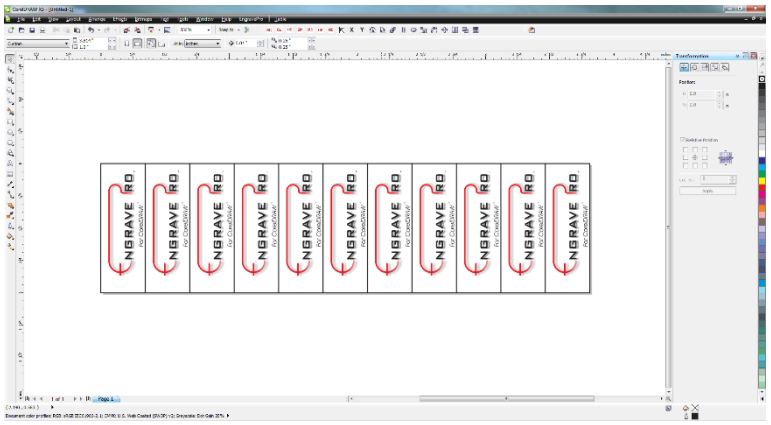


Length of Cut Mark is simply the length of the mark itself. You can specify both the horizontal and the vertical length.

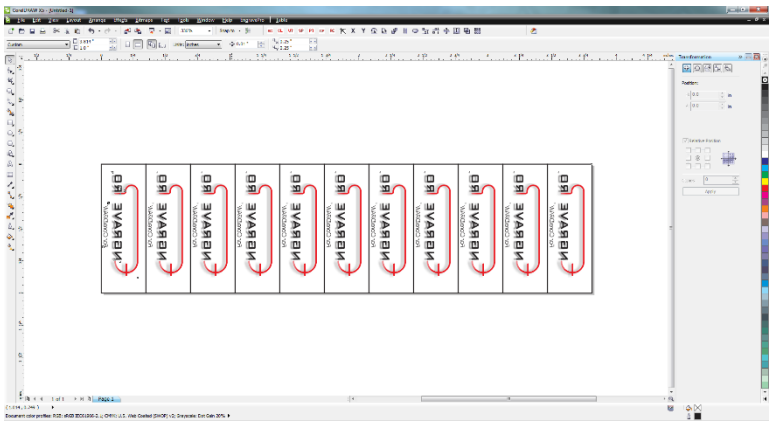
Flip Page

Mirror your page including all the objects on the page. Flip Page is for mirroring an entire page for reverse engraving jobs. You can flip either horizontally (left to right) or vertically (top to bottom).

Before



After



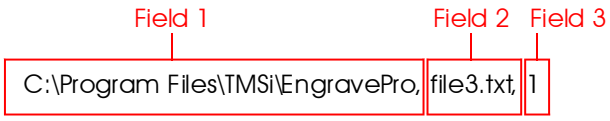
Group List File

If you want to combine text from two or more files then you need to create a Group List File (GLF). A GLF can be created in Notepad or any other text editor.

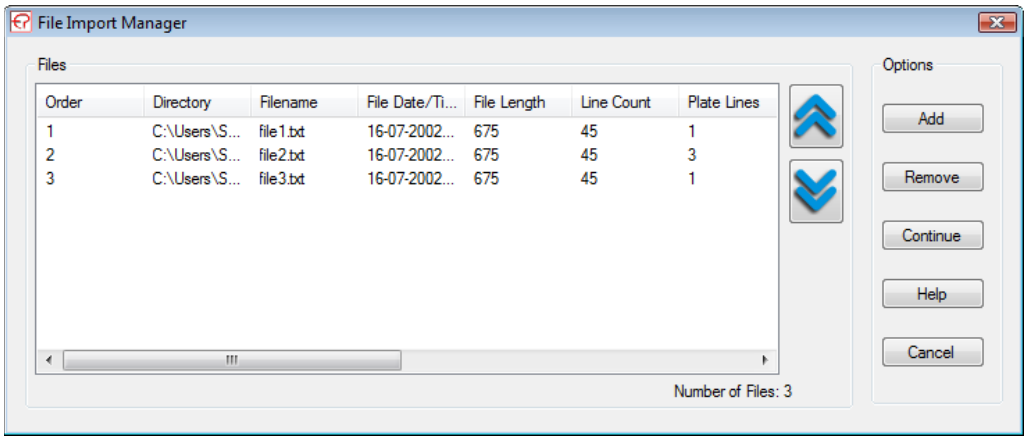
The following is an example of what a GLF may contain. You can open this sample in the folder "My Documents\EP\Sample Files". The file is called "GROUP.LST" Please be aware your file may differ in content than the one you see below, but will still work the same.

C:\Program Files\TMSi\EngravePro, file1.txt, 1
C:\Program Files\TMSi\EngravePro, file2.txt, 3
C:\Program Files\TMSi\EngravePro, file3.txt, 1

There are three fields of information for each line entered. The first field is the directory where the file is stored. The second field is the file name and the third field is how many lines from the specified file will go on each plate. Each field must be separated by a comma.



If you started Variable Text and you chose a GLF then the following dialog box will appear when you click the Create Output Job menu item from the File Menu.



Let's look at what each button of the File Import Manager does:

The **Add** button adds a new file to your list. Clicking the button will open an Open File dialog box allowing you to choose new files to add.

The **Remove** button removes the currently selected item from your list.

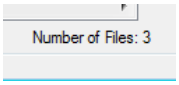
The **Continue** button proceeds with your Variable Text job.

The **Cancel** button cancels the Group List File import and returns you to the main screen.

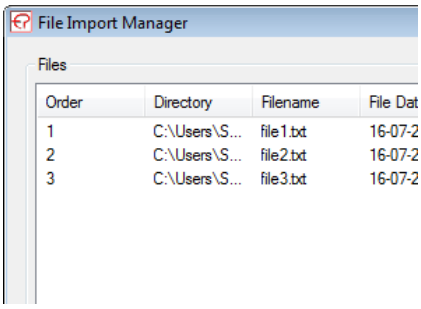
The **Help** button displays this help manual.

The **Up** and **Down** buttons will move your file up or down in the list, thus changing the Order in which they are imported.

Now let's examine the interface.



At the lower right corner of the file list grid you will see display of how many files you have listed. This is useful when there are too many to easily count.



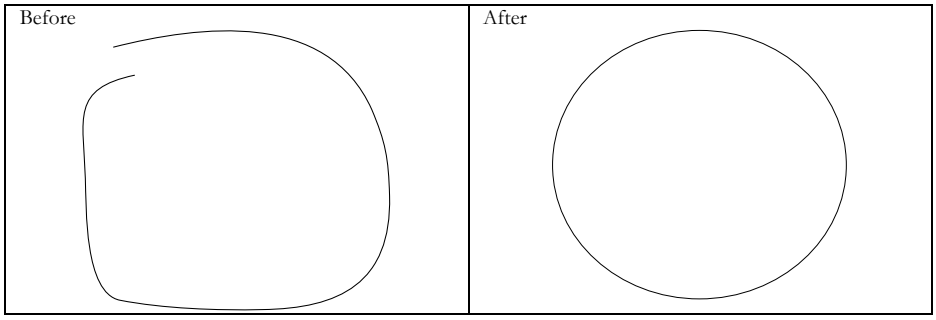
This area is where your file information is. The columns are labeled as follows:

Order	Determines in which order the files are imported.
Directory	Where the specified file is stored on you PC.
Filename	The name of the text file you want to import.
Date/Time	The date and time of when the file was created.
File Length	The size of the file in bytes. One byte equals one character.
Line Count	Number of lines in the file.
Plate Lines	How many lines from this file you want to import for each plate.

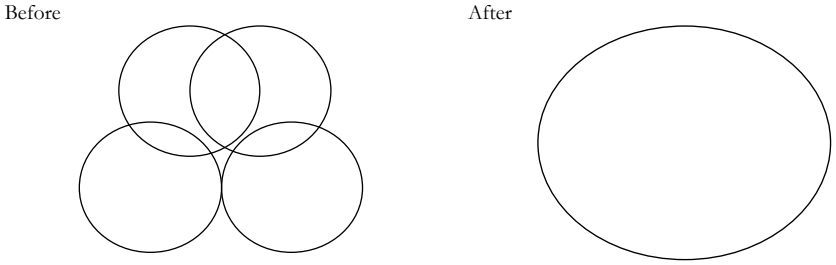
The only value you are allowed to change is the Plate Lines cell. To change the value, either double click Plate Lines cell with your mouse or press the Enter key when the line is selected. After entering a new value press the Enter key again to save your changes.

Make Circle

Converts the selected object or objects to a circle in the size of the selected object. If you have multiple objects selected it creates a circle that is the total size of the selection box of the selected objects



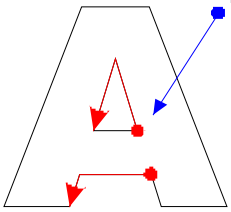
You can also select multiple objects and convert the group to a single circle.



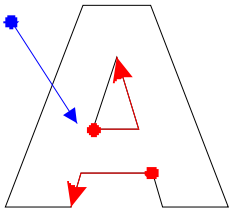
Make Entry Point

Make Entry Point allows you to change where your engraver will start cutting a curve object. Look at the following illustration.

This is my original entry point
for the inside of this letter A



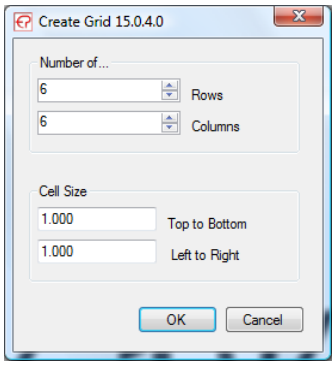
By selecting a different node
and using Make Entry Point
I have changed the starting point
this object.

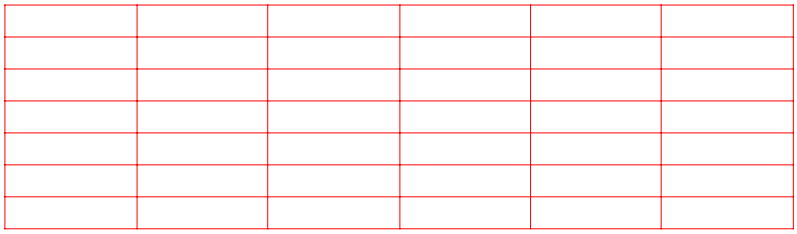


It is especially import to remember that you can only change the entry point of a "curve" object. This means you will have to "convert to curves" any text, circle, square objects you created using Corel tools. To convert to curves, select the object you want to modify, click the Arrange menu, then click convert to curves.

Make Grid

Make grid allows you to automatically create a grid pattern of the size you specify. You set the number of rows and columns and how big the cell is.





Results

Now you can create grid cells the size you want and the lines are already optimized for vector output.

Rows – Enter the number of rows you want.

Columns – Enter the number of columns you want.

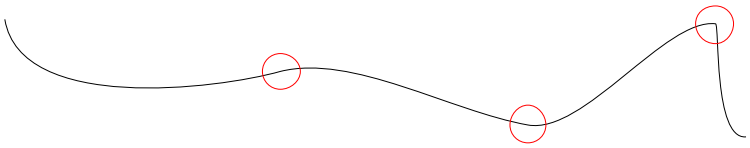
Cell Size

Top to Bottom – Enter the vertical height of the cell.

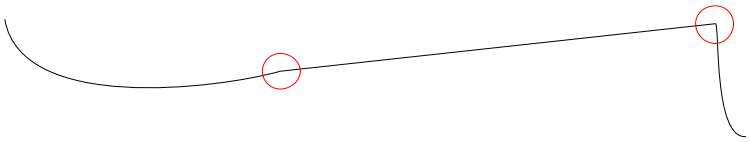
Left to Right – Enter the horizontal width of the cell.

Make Line

Select at least three nodes with the pick tool on a curve and select Make Line from the EngravePro menu. Make Line will automatically straighten the line for you. For example:



Before with the three nodes highlighted

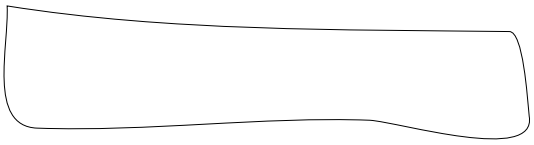


After with the two remaining nodes highlighted

Make Line not only straightens the line, but removes no longer needed nodes.

Make Square

Make Square will make a box out of the select object(s) by calculating the outermost limits of the selection.



Before Make Square

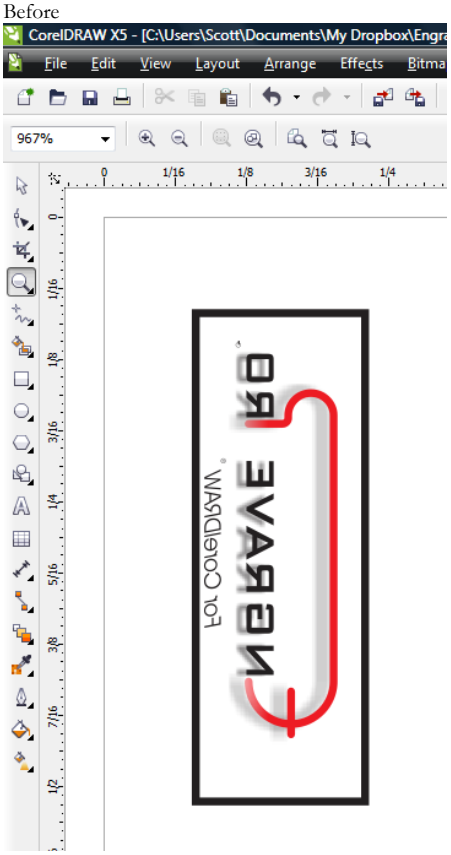
After Make Square

To use make Square, simply select the object or objects you want to make a square from then click Make Square form the EngravePro menu.

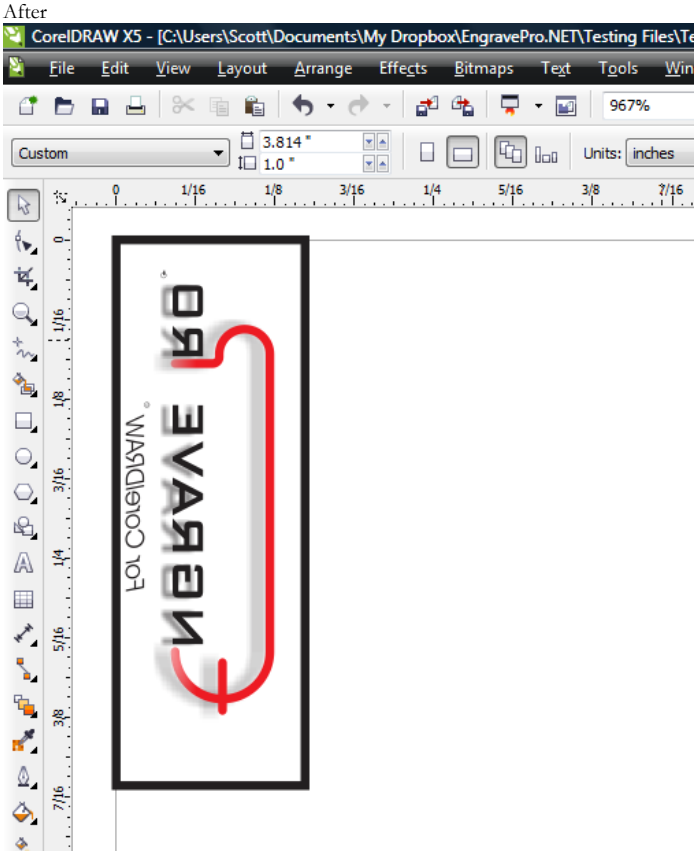
Move Objects Home

Move selected or all objects to the upper left corner of your page. Most times you want to save material by engraving objects as close to the upper left corner of the page as possible. While you can certainly move the objects with your mouse, it is easier to simply select the object and then click Move Objects Home.

Before



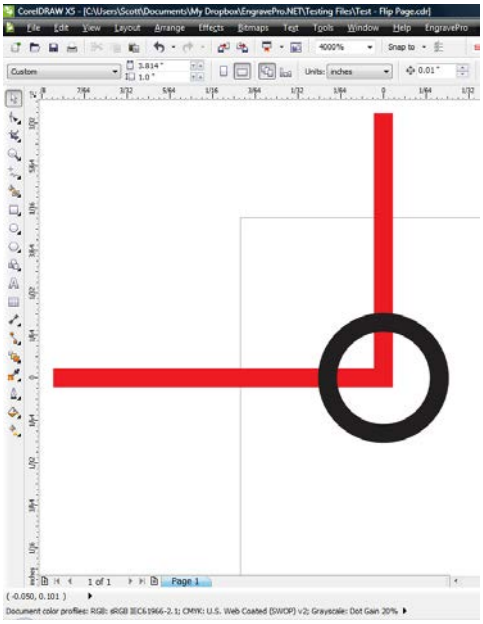
After



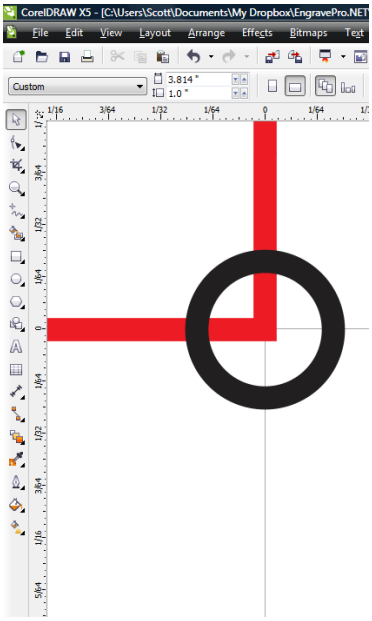
Move Rulers to Home

Move Rulers to Home – Automatically reset your rulers to the zero in the upper left corner of the page.

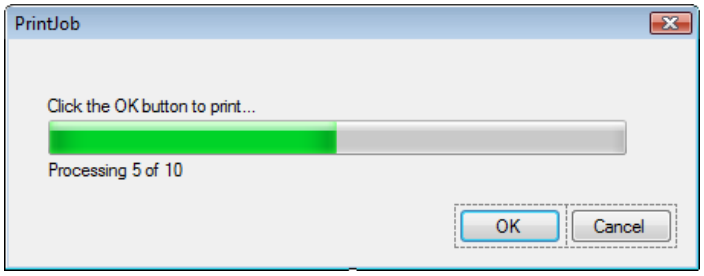
Before



After



Print Job



Print Job will take a Corel Layout containing multiple pages and print each page to the laser one page at a time.

One of the caveats is that this will only work with the default printer. So if the printer you want to output to is not the default printer you must set it to be the default prior to using PrintJob.

Once you have the desired printer set as your default, select PrintJob from the EngravePro Menu.

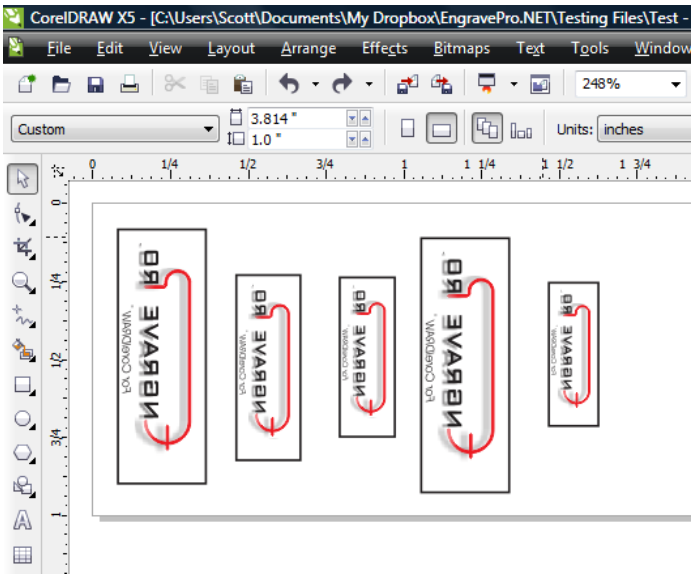
Your default printer properties dialog will appear. Adjust your printer settings and click the OK button.

The dialog box above will appear. Once your printer is ready to receive the files, click the OK button.

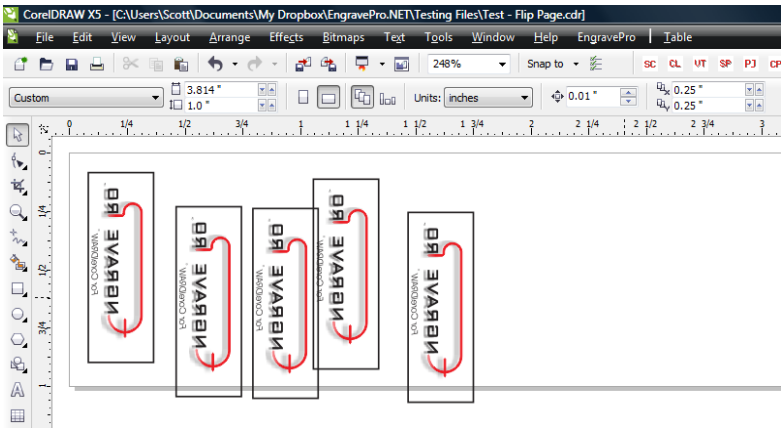
Size Selected Objects

Sizes all selected objects to the last selected object.

Before



After



Size Selected Objects Width

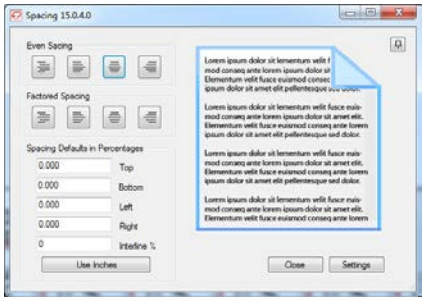
Sizes all selected objects to the last selected objects width.

Size Selected Objects Height

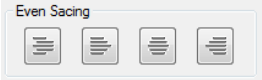
Sizes all selected objects to the last selected objects height.

SpacePlate

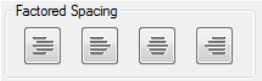
Below is the SpacePlate toolbox. You can even space selected objects or factor space objects in CorelDRAW.



Values entered into SpacePlate are either a percentage of plate width and/or height or actual units (inches, centimeters, etc). Interline Spacing is a percentage of the line height itself. Clicking the button at the bottom of the toolbox will change the type of value between Inches and Percentages. When clicking the button it will automatically convert your values based on the current size of your plate. For example, if the button is labeled Percentages, you have 10 percent in the left and right margins and your plate is 3 inches left to right then clicking the button to Inches will change your value of 10 to .3. Conversely, if you click the button again back to Percentages it will automatically calculate .3 to be 10 percent of your current plate width.



E-Spacing is Even Spacing of Selected/Unselected Lines. You may either have your text selected or not. If your text is selected the spacing will occur between the top and the bottom lines that are selected. If you select your lines of text by clicking the first line then Shift + Clicking the other lines, all lines will be adjusted based upon the last line you selected. If your text is not selected all lines on the plate will be spaced and the text width will be adjusted according to the left and right margins you set.



F-Spacing is Factored Spacing of Unselected Lines. Factored spacing takes into account the percentage entered into the percentage boxes. Numbers entered are the desired percentage of plate space you want to use.

Top and Bottom values - Enter an amount you want to add to the top and/or bottom of the plate for line spacing. These values are only considered when you are factor spacing.

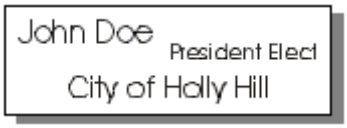
Left and Right values - Left and right margins can be set and will be used when you choose left or right flush even or factored spacing. This way you can set your left margin to .250 and click the Left Flush button thus aligning all of your text .250 from the left side of your plate.

Interline % value - The Interline spacing percentage is used to space lines according to their height and is only used during a Factored Spacing command. Interline spacing is always in percentages of the line regardless of the Inches/Percentages button state.

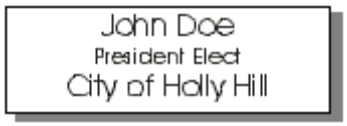
Note: Line selection is not supported for Factored Spacing.

What each button does...

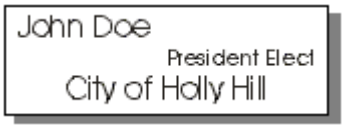
Taking the example plate to the right:



Even Spacing

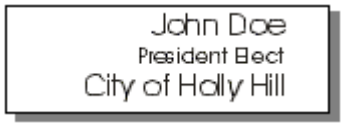


Lines are evenly spaced top to bottom and centered.



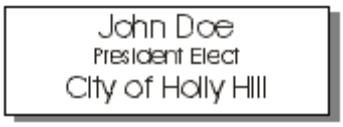
Lines are evenly spaced top to bottom and left as they were laid out on the plate.

Lines are evenly spaced top to bottom and left justified.

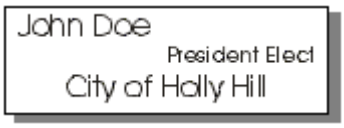


Lines are evenly spaced top to bottom and right justified.

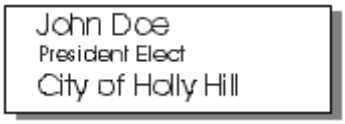
Factored Spacing



Lines are factor spaced top to bottom and centered, based upon the percentage entered into the Interline spacing box.

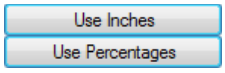


Lines are factor spaced top to bottom and left as they were laid out on the plate, based upon the percentage entered into the Interline spacing box.



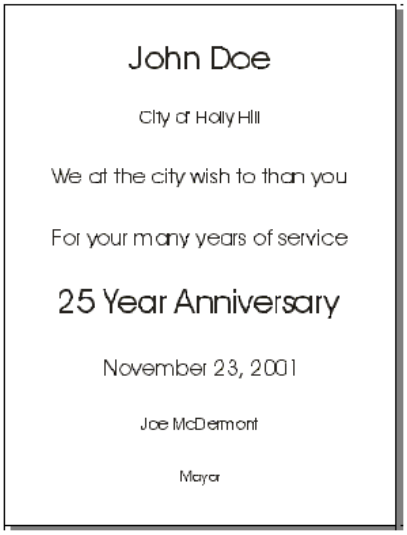
Lines are factor spaced top to bottom and left justified, based upon the percentage entered into the Interline spacing box.

Lines are factor spaced top to bottom and right justified, based upon the percentage entered into the Interline spacing box.



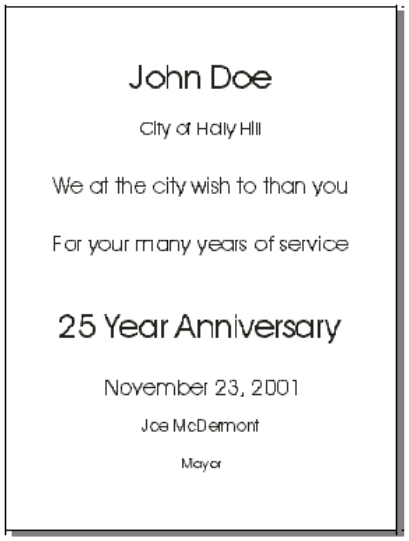
When the button is labeled Inches, values entered into the Top, Left, Right and Bottom are considered to be in actual inches. If the button is labeled Percentages then the values you enter are considered to be in percentages of the plate. Left and Right are based upon a percentage of the plate width. Top and Bottom are a percentage of the plate height.

Some more examples of Spacing



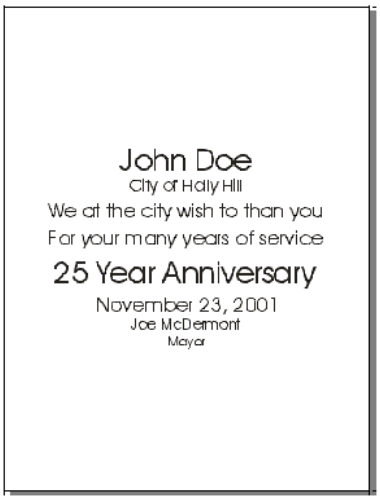
To the left, an even spaced plaque.

Factored Spacing with a 10 percent top and bottom margin with 300 percent interline spacing



Factored Spacing with a 20 percent top and bottom margin with 300 percent interline spacing.

Factored Spacing with a 40 percent top and bottom margin with 150 percent interline spacing.



Show Direction

Lets you see the entry point and the direction your laser or engraving machine will take when engraving the selected object.

Select the object you want to show the direction for and click Show Direction from the EngravePro menu.

The red circle indicates the starting point for the engraver and the arrow indicates the direction of the movement.

Reset Object

Reset Objects resets the selected object to its original created condition. So for example if we have some text on our page that we modified by adding some kerning and stretching it with the selection tool, reset object will remove all of these modifications.

Before User Editing

EngravePro

After User Editing

EngravePro

After Using Reset Object

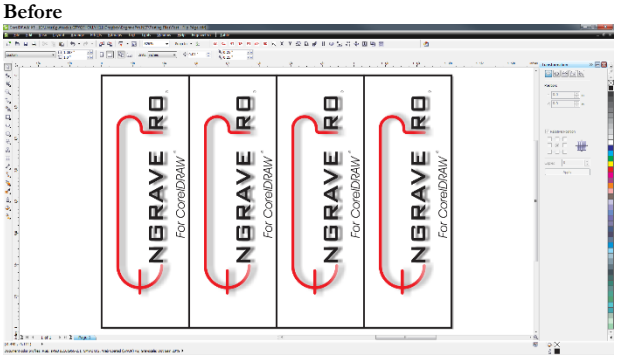
EngravePro

This feature also works with other object types such as rectangles, lines and circles.

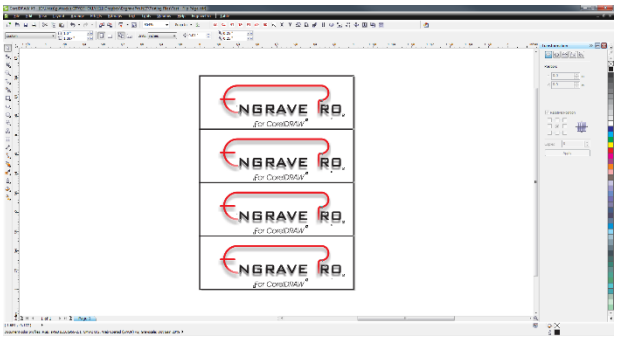
Rotate Page

Rotates your page to the left or right, similar to change from portrait to landscape but moves all your objects on the page as well.

Before



After

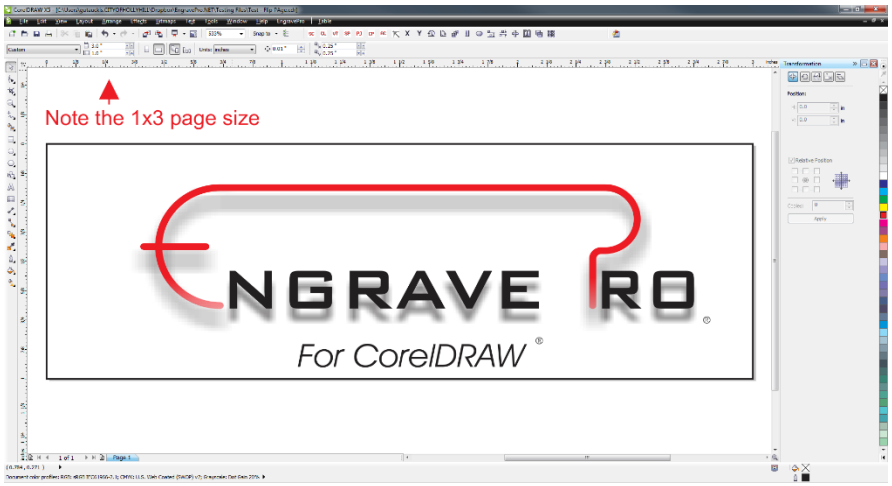


Set to Master

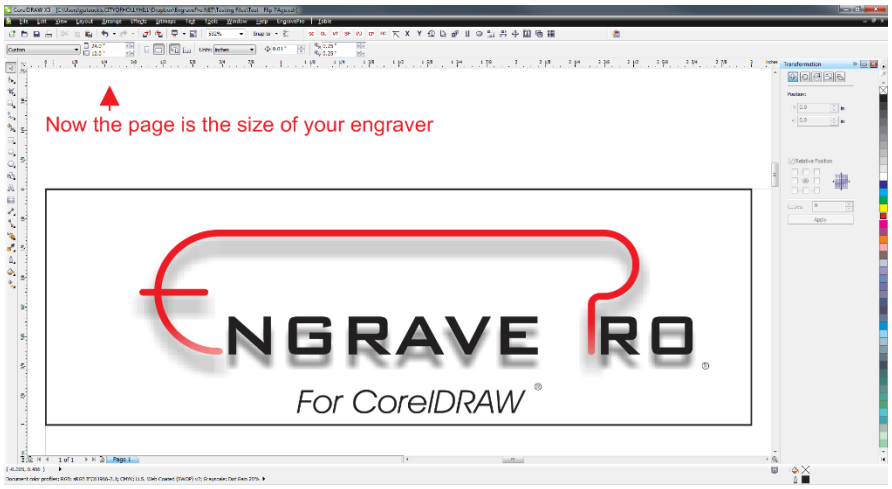
Set the current layout to the layout size of your engraver. The maximum layout size for your engraver is set during installation of EngravePro. If you need to change this size, click CopyPro. In the dialog box click the Settings button and enter your maximum layout size which is normally the maximum engraving area for your laser or engraver.

To illustrate you could create a 1 x 3 name tag by setting the paper size in CorelDRAW to 1 x 3 then when you are finished laying out your graphic choose Set to Master and it will set your page size to the actual engraver table size and move your graphic to the upper left corner of the page.

Before

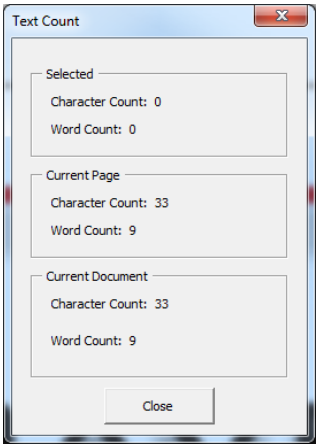


After

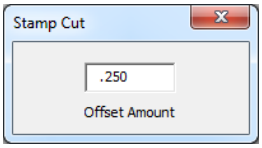


Text Count

Text Count counts the number of characters in your document. Displays count based upon selected text, text on the current page and text in the entire document. Text Count does not count the number of spaces in your layout.



StampCut



StampCut does pretty much what contouring does only one better. If you are doing rubber stamps and want your laser to cut the stamp out, use StampCut to automatically draw the cut out line.

For example take the following stamp:

John Doe
123 Anywhere Road
Anywhere, USA 12345

If we use StampCut to create an outline we end up with the following in CorelDRAW:

This will produce a cut border at the set distance, around your stamp. Of course the effect is not only great for stamps but for signs as well.

Variable Text

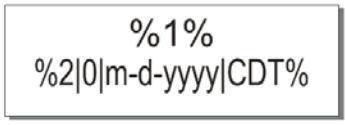
Variable Text is used to import text and/or pictures from files into a pre-existing CorelDRAW layout. You can either import in a batch mode where you will engrave one plate at a time or you can import into a grid layout where Variable Text will create the master sheet and at your request draw a grid around the layout or even put borders around each plate.

Tokens

VariableText works by using "Tokens" or placeholders. Currently VariableText allows for 4 types of tokens:

CDT is a token to be used when you want to automatically insert the current date into your plate.

> **Example:** %1|0|m-d-yyyy|CDT% produces 10-28-2007
>
> Each field is separated by the pipe symbol. This symbol is usually located above your backslash key. It's the symbol that looks like two little bars on top of each other.
>
> **%1**|0|m-d-yyyy|CDT% field one is the number is the sequence on your plate. So for example you could have:



The first line is read from the file you selected and the second line the date is put in.

%1|**0**|m-d-yyyy|CDT%, the second field you can set to a zero or one. Zero (0) specifies that the date be what it is down to the millisecond VariableText processes that line. If you specify a one (1) the date and time is determined at the beginning of the import and is the same for each plate imported.

%1|0|**m-d-yyyy**|CDT%, this field can be any of the formatting codes as specified in the table below.

The following table identifies characters you can use to create user-defined date/time formats:

Character	Description
(:)	Time separator. In some locales, other characters may be used to represent the time separator. The time separator separates hours, minutes, and seconds when time values are formatted. The actual character used as the time separator in formatted output is determined by your system settings.
(/)	Date separator. In some locales, other characters may be used to represent the date separator. The date separator separates the day, month, and year when date values are formatted. The actual character used as the date separator in formatted output is determined by your system settings.
c	Display the date as ddddd and display the time as ttttt, in that order. Display only date information if there is no fractional part to the date serial number; display only time information if there is no integer portion.
d	Display the day as a number without a leading zero (1 – 31).
dd	Display the day as a number with a leading zero (01 – 31).
ddd	Display the day as an abbreviation (Sun – Sat).
dddd	Display the day as a full name (Sunday – Saturday).
ddddd	Display the date as a complete date (including day, month, and year), formatted according to your system's short date format setting. The default short date format is m/d/yy.
dddddd	Display a date serial number as a complete date (including day, month, and year) formatted according to the long date setting recognized by your system. The default long date format is mmmm dd, yyyy.
w	Display the day of the week as a number (1 for Sunday through 7 for Saturday).
ww	Display the week of the year as a number (1 – 54).
m	Display the month as a number without a leading zero (1 – 12). If m immediately follows h or hh, the minute rather than the month is displayed.
mm	Display the month as a number with a leading zero (01 – 12). If m immediately follows h or hh, the minute rather than the month is displayed.
mmm	Display the month as an abbreviation (Jan – Dec).
mmmm	Display the month as a full month name (January – December).
q	Display the quarter of the year as a number (1 – 4).

y	Display the day of the year as a number (1 – 366).
yy	Display the year as a 2-digit number (00 – 99).
yyyy	Display the year as a 4-digit number (100 – 9999).
h	Display the hour as a number without leading zeros (0 – 23).
Hh	Display the hour as a number with leading zeros (00 – 23).
N	Display the minute as a number without leading zeros (0 – 59).
Nn	Display the minute as a number with leading zeros (00 – 59).
S	Display the second as a number without leading zeros (0 – 59).
Ss	Display the second as a number with leading zeros (00 – 59).
t t t t t	Display a time as a complete time (including hour, minute, and second), formatted using the time separator defined by the time format recognized by your system. A leading zero is displayed if the leading zero option is selected and the time is before 10:00 A.M. or P.M. The default time format is h:mm:ss.
AM/PM	Use the 12-hour clock and display an uppercase AM with any hour before noon; display an uppercase PM with any hour between noon and 11:59 P.M.
am/pm	Use the 12-hour clock and display a lowercase AM with any hour before noon; display a lowercase PM with any hour between noon and 11:59 P.M.
A/P	Use the 12-hour clock and display an uppercase A with any hour before noon; display an uppercase P with any hour between noon and 11:59 P.M.
a/p	Use the 12-hour clock and display a lowercase A with any hour before noon; display a lowercase P with any hour between noon and 11:59 P.M.
AMPM	Use the 12-hour clock and display the AM string literal as defined by your system with any hour before noon; display the PM string literal as defined by your system with any hour between noon and 11:59 P.M. AMPM can be either uppercase or lowercase, but the case of the string displayed matches the string as defined by your system settings. The default format is AM/PM.

PIC is used to specify that an image file is to be imported. It expects the line of text read from your file to be fully qualified path and filename to a valid image file. Example: %1PIC%

See **Importing Graphics** for more information.

BAR is used to specify barcodes to be generated.

The token is formatted as such: %1|*bar type*|BAR%

Bar type can be any of the following codes used to specify which bar code you want produced. Substitute the barcode name for *bar type* to have VariableText generate a barcode for you.

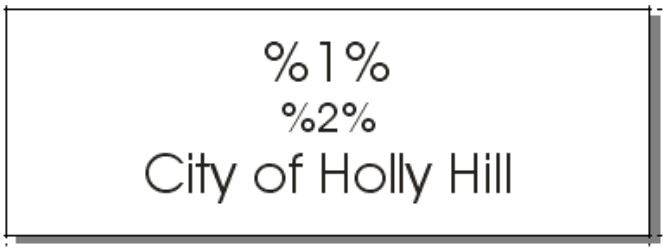
Bar Code Name	Sample Bar Code
code39 code39_full_ascii code39_hibc	

codabar	
code93 code128	
ucc_ean_128	
interleaved_2of5	
postnet	
upc_a upc_e	
ean_jan_8 ean_jan_13	
bookland	
msi_plessey	
pdf417	
aztec	
datamatrix	

maxicode	
rss14 rss14stacked rss14stackedomni rsslimited	

#, the last token is the default one used to import lines of text from your input file. Tokens are set as %1%, %2%, %3%, etc. and correspond to each line of text in your file.

Before you start to use the Variable Text feature you must first setup a plate in CorelDRAW. If you look in your Start Menu and click on EngravePro you will see 3 sample jobs. Click "Sample 1". A CorelDRAW job will open and reveal the following layout.



Notice the %1% and %2%. This formatting is what tells Variable Text where to place your text from your import file. Each line from your text file corresponds to the particular formatting number. In the above job Variable Text will import two lines of text from the file you specified and then create a new plate. For each line of text you want imported from your file into one plate, you must have one formatting line, numbered sequentially. Group lines of text will not work. If you group formatting lines together Variable Text will ignore the line.

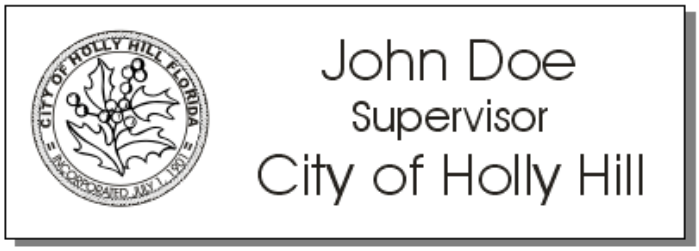
Importing Graphics

If you want to use Variable Text to automatically import graphic files you need to add a paragraph text line like what is shown below. The paragraph box should be sized according to the desired size of your imported graphic and should contain the normal placeholder as in the example below.

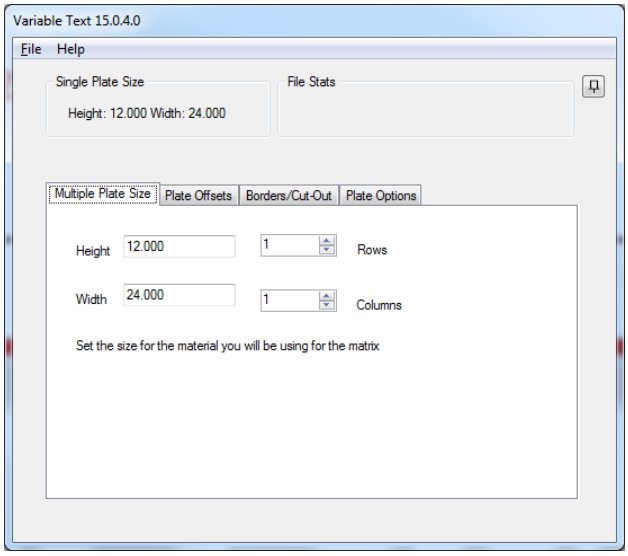
In your text file the line of text that corresponds to you image will contain the full path and filename of the image to be imported. So for example, If I wanted a name tag that had "John Doe", "Supervisor" with the city logo my text will would contain the following three lines:

John Doe
Supervisor
c:\Clipart\citylogo.eps

This produces the following name tag:



You should note that the Editor available when you click "Create a File" is capable of creating text files from directories. This will automatically create a text file of all graphic files in a specified folder for you.
Going back to the original example you should at this point select each line of text and set its justification in CorelDRAW. Doing this ensures proper placement of the line. By default CorelDRAW has its justification set to none. When you are satisfied with your plate layout, click the EngravePro menu and select Variable Text. The following dialog appears:



From the File Menu you have three methods of input.

Methods of Input

Open - Opens a pre-existing text file or alternatively a Group List File. A Group List File is a text file containing the names of individual texts file you want to import into the same plate or job.

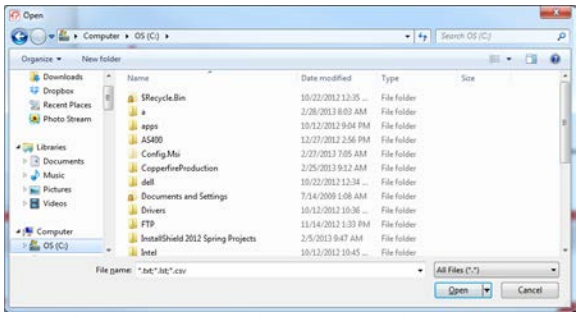
New - Opens up a text editing window similar to NotePad. From there you can type your lines of text for your current job. You may also save or open a text file from within the Editor.

From Clipboard - Pastes text from the clipboard into your job. If you have text on the clipboard this button will appear enabled.

Methods of Input Detailed

Open

If you want to input a pre-existing file, click this button.
A File Open dialog box will open similar to the one shown below.

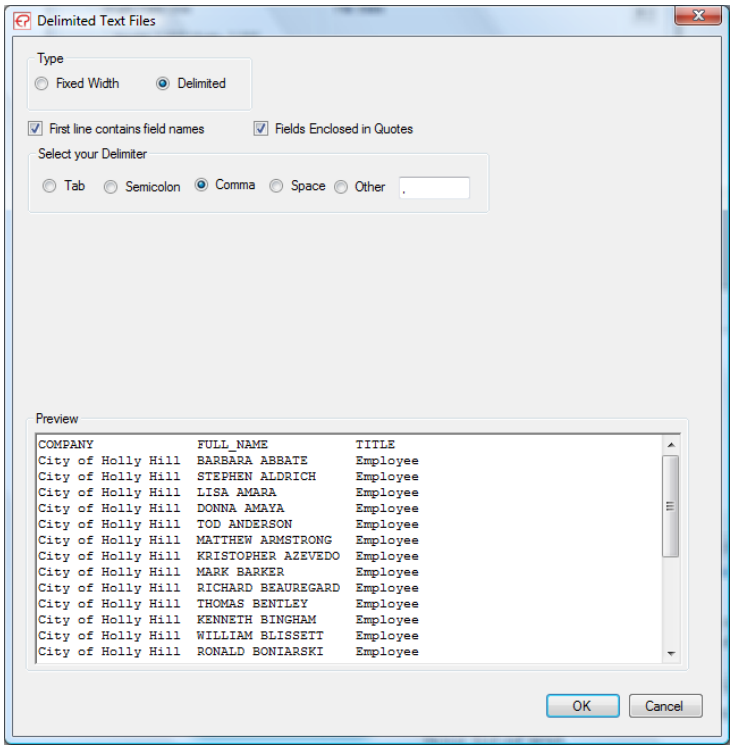


Note: If you want to work with a Group List File see the instructions at the end of this section.

When opening text files you have a few choices as to the kind of text file you open.

Standard Text File – Contains a line by line format for importing. Each line is separated from the next by a carriage return. Variable text associates each line with one placeholder in your job layout.

Delimited Text File – Text file contains lines of text with each line containing multiple fields separated by a specified character. Usually the character is a comma or a semi-colon. For example**: Jansen & Son's Insulation;745 Commerce Drive;Venice;FL;34292**
The above line of text contains 5 fields and uses the semi-colon to separate them. It would be expected that in your job layout you would set up a plate that had 5 placeholders labeled %1% through %5%.



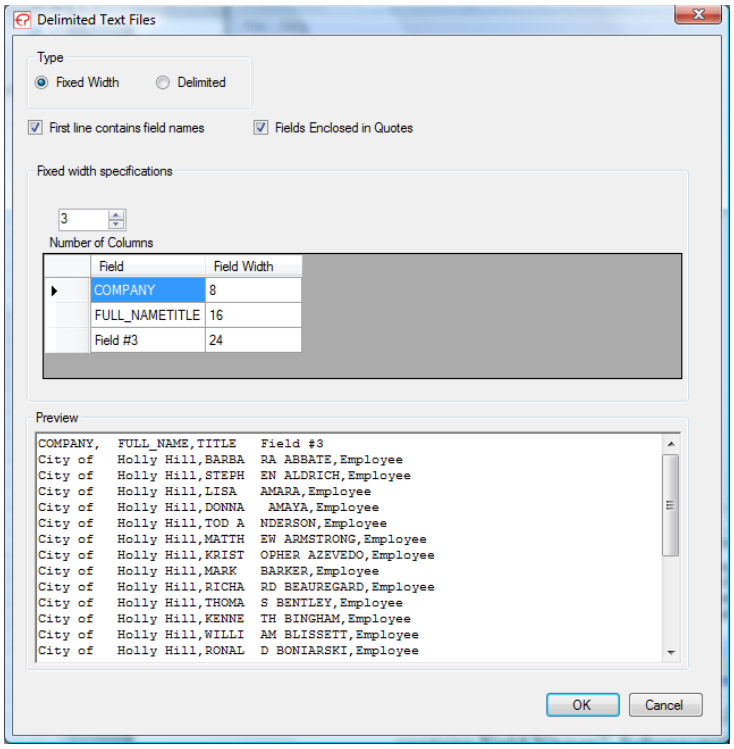
If the text file you are working with has field names set check the box marked, "First Line contains Field Names". Subsequently Variable Text will not import the first line of the text file.

If you are using different character delimiters than what is already listed you can click the "Other" option thereby allowing you to enter the character of your choice.

Fixed Width Text File – Text file contains lines of text with each line containing fields of a fixed known width.

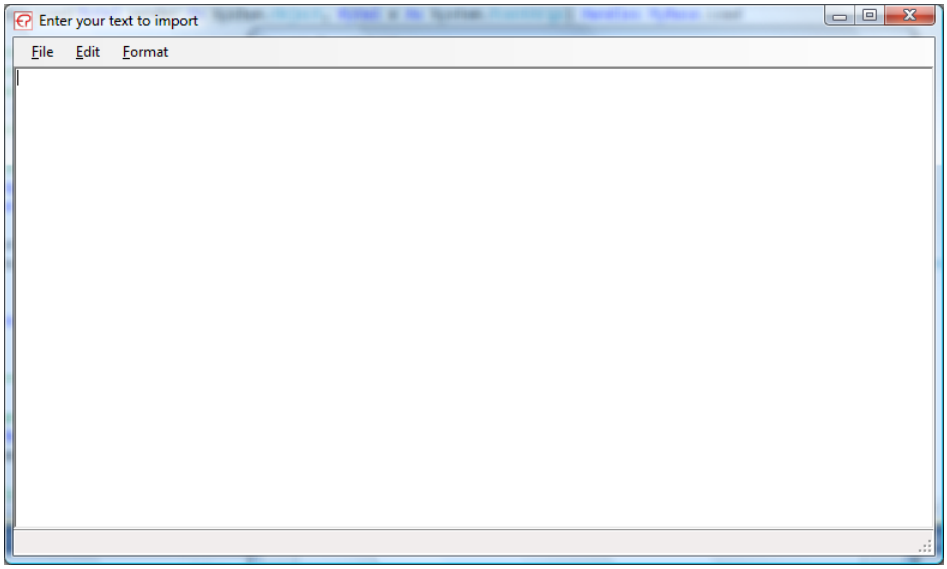
For example: **Jansen & Son's Insulation 745 Commerce Drive Venice FL 34292**

The above line of text contains 5 fields and each field is a set number of characters long. The first field is 26 characters long; the second field is 19 characters long, and so on.



Set the number of columns, and then set the column width for each column you specified.
Note: To work with either fixed width or delimited text files the name of your text file must have CSV for the file extension. For example, **companies.csv** would tell Variable Text you want to work with delimited or fixed width text files while an extension of TXT would not.

If you want to create a file while in the Variable Text program click **New** from the **File Menu**. A text editor will appear for you to enter your list of text.



You can simply enter text in the window much like you would if you were using Notepad.

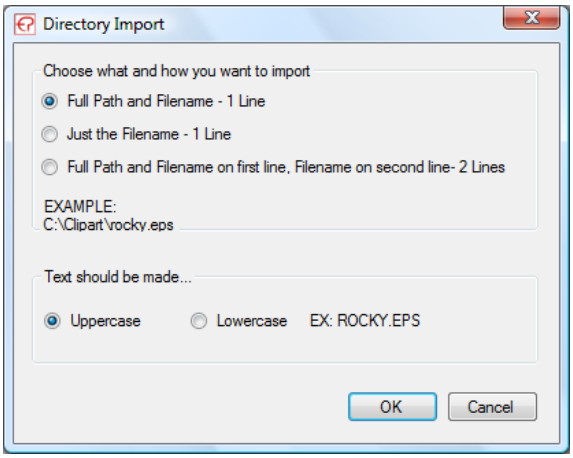
Menu Choice Functions Explained

File Menu

Continue – Proceeds to the next step in your Variable Text job.
Open – Opens a pre-existing text file
Save – Saves your work to a file
Import

Folder Listing

Allows you to create a text file of a chosen folder on your PC. This is beneficial when you are importing pictures into Variable Text. When you choose this menu option the following dialog box will appear:



The options are as follows:

Full Path and Filename

Create a single line of text in your file containing the full path and filename.

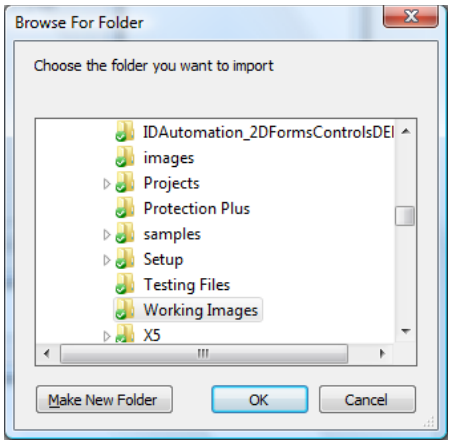
Just the Filename

Create a single line of text in your file containing only the filename

Full Path and Filename on first line, Filename on second line

Creates two lines of text in your file containing the full path and filename on the first line and the filename on the second line. This is useful when you want to create a thumbnail sheet in CorelDRAW and you want to label each thumbnail with the filename.

After choosing your options, click the OK button to continue. The following Dialog Box will appear:



Navigate to the folder containing the files you want and click the OK button. Your text editor will now contain the information you selected.

Cancel – Closes the editor.

Edit Menu

Undo – Will undo your last edit.

Cut – Cuts the selected text to the clipboard and removes it from your text window.

Copy – Copies the selected text to the clipboard.

Paste – Pastes text on the clipboard into your text window.

Select All – Selects all text in the window.

Change Case

Upper – Changes all selected text to upper case.

Example: City Manager changes to CITY MANAGER.

Lower - Changes all selected text to lower case.

Example: City Manager changes to city manager.

First Letter - Changes all selected text so that the first letter of each word is capitalized.

Example: city manager changes to City Manager.

First Word - Changes the first letter of the first word to uppercase.

Example: City Manager changes to City manager.

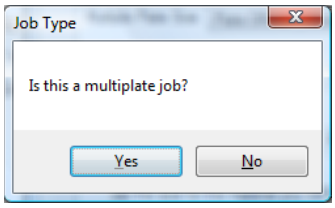
Invert – Inverts the case of all selected text. So if the character is capitalized this function will make it lower case. If the letter is lower case then it will capitalize it.

Example: City Manager changes to cITY mANAGER.

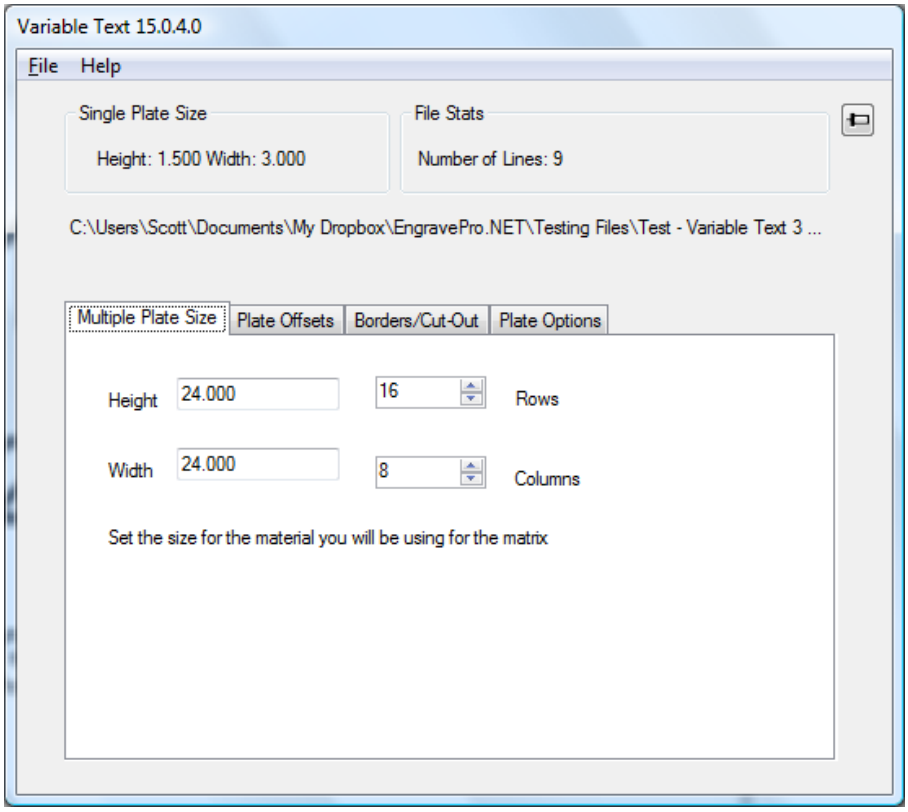
Reverse – Reverse the selected text.

Example: City Manager changes to reganaM ytiC.

Simply uses the text on the clipboard to fill in your plates. If there is no text on the clipboard this button will be grayed out and unavailable. The clipboard is useful if you want to copy lines of text from another application.



Regardless of what method of input you use you will see the following dialogs. Choose your method of import and complete any steps if necessary. When you do, the message on the left will appear. If you intend to do each plate one at a time, then answer the message No. If you are doing a grid of plates then answer, Yes. Answering Yes will display the Plate Size dialog box.

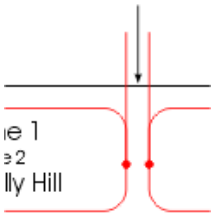


The **Single Plate Size** values cannot be changed here. Those values were set in Corel Draw when you set the page size of your layout.

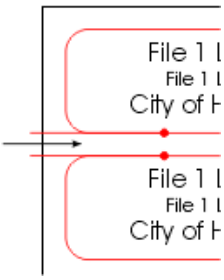
Plate Options allows you to group each individual plate, convert all text to curves and specify which direction to create your plates in.

Multiple Plate Size is the size of your material not your single plate size. This value you can change and should be larger than your Single Plate Size values. You may also specify the number of Rows and Columns you want to create. For example you may always want to set your Multiple Plate Size to the size of your laser table and use the Rows and Columns to specify how many plates you want vertically and horizontally.

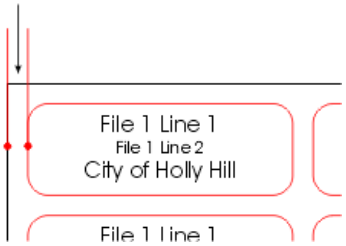
Plate Offsets



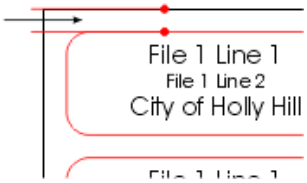
Horizontal Spacing - The amount of distance you want added between your plates in the X (Left to Right) direction. This value is only used when you choose Borders or None. The value has no effect for Grid Cut-Out.



Vertical Spacing - The amount of distance you want added between your plates in the Y direction. This value is only used when you choose Borders or None. The value has no effect for Grid Cut-Out.

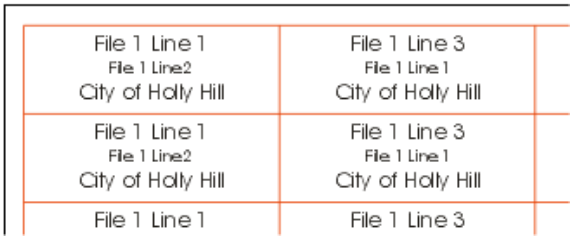


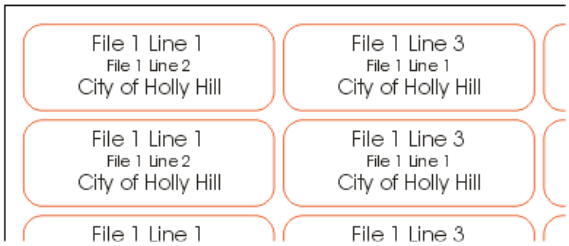
Left Border - The amount of space you want to add between the left side of your master plate to the left side of your first single plate.



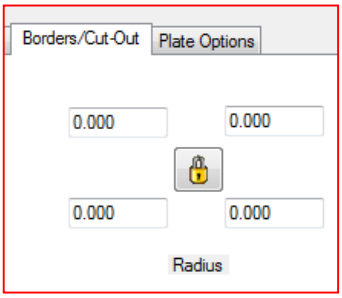
Top Border - The amount of space you want to add between the top of your master plate to the top of your first single plate.

Grid Cut-Out will add a grid to the job for the purpose of cutting out your plates. When this is selected Horizontal/Vertical Spacing has no effect. Left/Top border will be added if there is a value greater than 0 in either of the text fields.





Borders will allow you to automatically place borders around your plates at the Single Plate Size and optionally allow you to round the corners. The radius is a value in inches. You may enter a radius equal to one half the length or height, whichever is smaller.

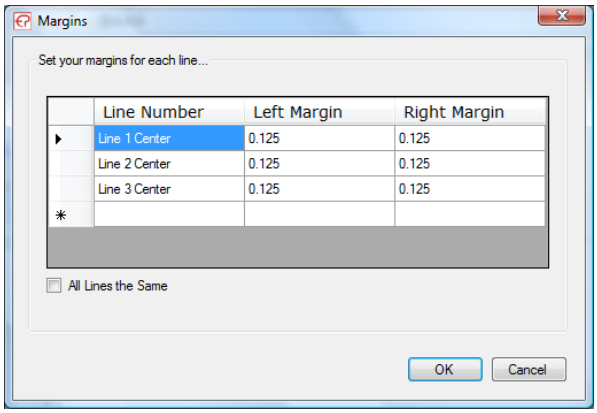


When the button is in the locked mode you may enter a value in any of the four corners of the radius and it will automatically change the other three corners for you.

Lets you enter a radius in each corner independent of the other three corners.

Convert Text To Curves – After Variable Text has imported the line of text it converts the text to curves when this check box is checked. It has been shown that some fonts do not engrave well on lasers when printed to the laser as a font. Converting the text to curves improves engraving quality.

Once you have entered the desired values click the Okay button. The next message you will see is one asking if you want to specify margins. Clicking Yes will result in the following dialog box being displayed.



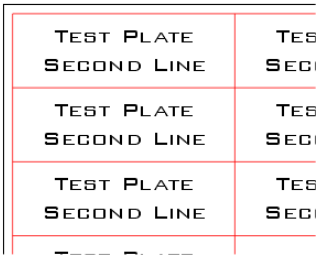
You will notice that for each line of Variable Text you specified in CorelDRAW there is an entry in the dialog box. You may specify each lines margin accordingly. If all lines are to have the same margin, enter the margins for the first line then click the All Lines the Same checkbox. Keep in mind that when you set margins, all your lines will be condensed to fit the specified width and they will be centered on the line.

Navigating around the fields in this dialog box is easy. Click on the first value to be entered, press the Enter key. You will notice your selection box will move to the next cell. Press Enter again and

you will be in edit mode. Enter your values and repeat the process. Upon completing all the margins, click the Okay button.

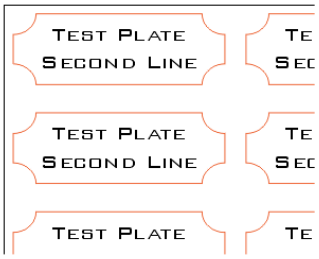
As Variable Text process your file a Status Bar will appear to let you know the progress of the import.

Examples of Borders and Grid Cutout



Grid Cutout with a .125 top and left border.

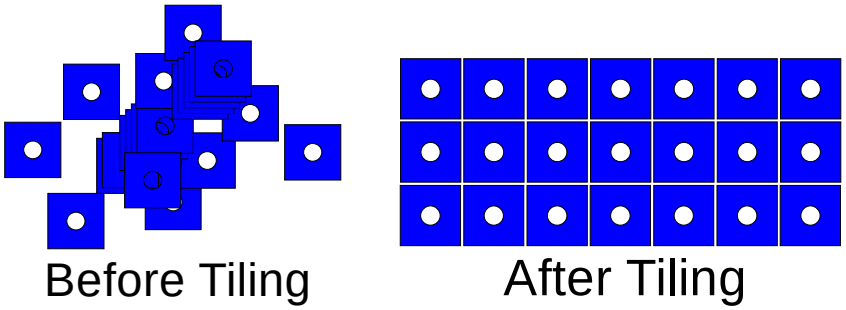
Borders are selected. They have a .125 top and left border and a .3 horizontal Space and a .4 Vertical Space.



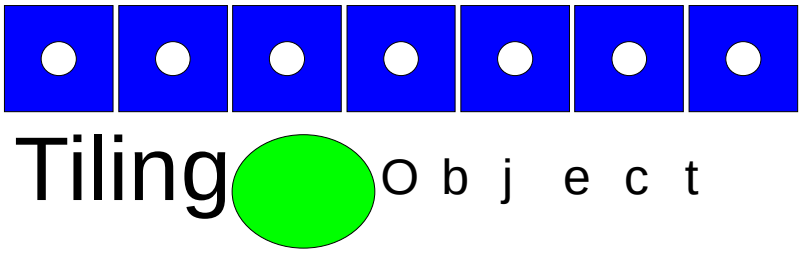
Custom Borders are selected. They have a .125 top and left border and a .3 horizontal Space and a .4 Vertical Space.

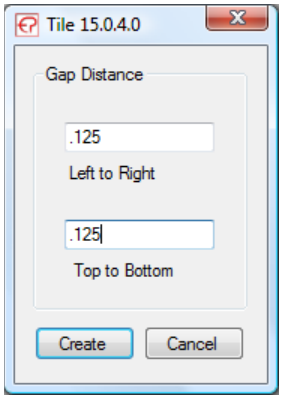
Tile Objects

Tile Objects automatically tiles the selected objects based upon the distance you specify. Take the following layout for example.



Tiling works with any object not just objects of the same type. So for example you can mix text in with drawing objects or clipart.





To tile simply create your objects on the page, select only the objects you want to tile then click Tile Objects in the EngravePro menu. Enter the gap distance between tiles and click ok. Once you are satisfied with the results click the close button.

Creating Custom Borders

When creating custom borders a few important points should be considered.
Draw your border at the proper aspect ratio. In other words, if your plate is going to be 1 inch tall and 12 inches long, create your border at 1 inch tall and 12 long or a 1:12 ratio. If you make it 3 inches tall and 12 inches long, when you use it for your border it will distort, thus producing undesired results.

If you are intending on using the border as a cutout, make sure you set the line width to .001 and its color to red before saving the file.

By default **CopyPro** looks in the "\Program Files\TMSi\EngravePro\Borders ", Directory for borders. However, you can load your borders from anywhere on your PC.

When saving your Border from CorelDRAW, select your border and export it with the "Selected Only" option in the file save dialog box.

You must save your file as an Enhanced Windows Metafile (EMF).

If your border is an odd shape, it may be better to put it in your CorelDRAW layout instead of importing it through Copy Pro. The reason is simply Copy Pro does not reposition your lines to fit within a particular border. Although CopyPro does just fine with borders that are rectangular in shape, odd shaped borders present unique layout issues. For example consider the two plates below:

Figure 1 with a rectangular border and imported works fine.

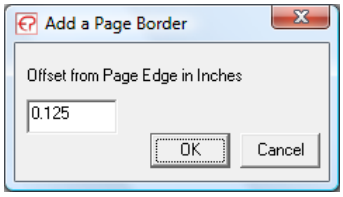
Figure 2 however does not work, especially because the aspect ratio of the border does not match the plate. Here our example is a 1 by 3 inch plate. The border was designed for a 1.5 by 3 plate. The result is a border shortened thus running into your text. The solution here would be to put your border into your layout while in CorelDRAW and reposition your lines of text.

Invert Selection

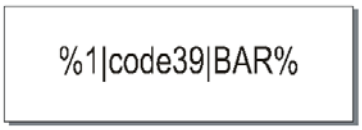
Inverts your current selection such that all items that are currently selected will be unselected and all items that were initially unselected become selected.

Set Border

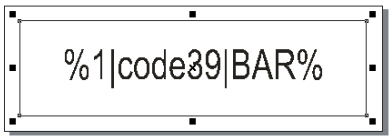
Creates a border on your page. The value entered is the amount of distance between the edge of the plate and the border itself.



Before

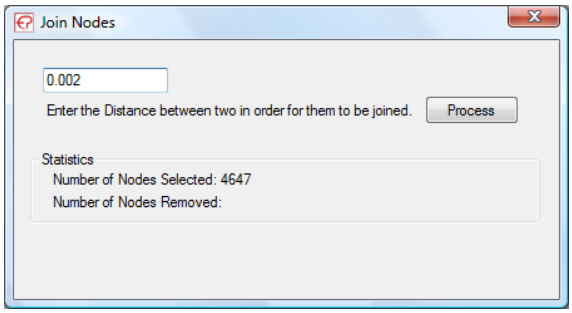


After



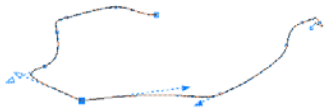
Join Nodes

Joins two or nodes together. The nodes must be from the same object and be an end node. In the case of having two or more lines, select both lines then combine them prior to joining nodes. This feature is very useful especially with imported CAD drawing where there are many overlapping points.



Before

After



Barcodes

Beginning with version 5.7 EngravePro now creates barcodes with its VariableText module. While this User Document attempts to explain the various barcodes available to you, ultimately it is up to you to understand each bar code and its appropriate use. Keep in mind some barcodes expect your text to be numeric in content and if the text is inappropriate for the selected barcode an **"!!!ERROR!!!"** will be displayed instead of the bar code.

Using the barcode is quite simple. Lay out your CorelDRAW job as you would normally do except using the bar code token.

%1%|code39|BAR% for example will produce a Code 39 barcode, representing the text from your input file.

There are various options for each of the many bar codes VariableText will create. To review these options you should click the "Options" button in VariableText. Many settings control different preferences such as comments, bar code height, etc. VariableText uses this setting to format your barcode and takes no format control from CorelDRAW. This is contrary to using VariableText normally where size, font, color, etc., are derived from the line of text used by the token.

Normal Code 39

The Normal CODE 39 is a variable length symbology that can encode the following 44 characters: 1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ-. *$/+%. Code 39 is the most popular symbology in the non-retail world and is used extensively in manufacturing, military, and health care applications. Each Code 39 bar code is framed by a start/stop character represented by an asterisk (*). The Asterisk is reserved for this purpose and may not be used in the body of a message. VariableText automatically adds the start and stop character to each bar code therefore you should not include them as part of your bar code message. If you select the Normal version of Code 39 and your bar code text contains lower case characters, the control will convert them to upper case. If your bar code text contains invalid characters, the control will raise an error event with the error code for an "Invalid Message" error and a bar code will not be generated.
Code 39 allows for an optional (modulo 43) check. The health care industry has adopted the use of this check character for health care applications (HIBC bar codes). To enable the Code 39 check, set the Code39 Check Digit preference to True. When this option is enabled, the control will automatically calculate and append the proper check character to all Code 39 symbols. If you set the Symbology to Code39_HIBC then the control will automatically add the check digit no matter what value the Code39 Check Digit preference is set to.

Full ASCII Code 39

The FULL ASCII version of Code 39 is a modification of the NORMAL (standard) version that can encode the complete 128 ASCII character set (including asterisks). The Full ASCII version is implemented by using the four characters: $/+%. as shift characters to change the meanings of the rest of the characters in the Normal Code 39 character set. Because the Full ASCII version uses shift characters in combination with other standard characters to represent data not in the Normal Code 39 character set, each non-standard character requires twice the width of a standard character in a printed symbol. The table below shows the character combinations used to produce the Full ASCII version of Code 39.

ASCII	CODE 39	ASCII	CODE 39	ASCII	CODE 39	ASCII	CODE 39
NUL	%U	SP	SPACE	@	%V	`	%W
SOH	$A	!	/A	A	A	a	+A
STX	$B	"	/B	B	B	b	+B
ETX	$C	#	/C	C	C	c	+C
EOT	$D	$	/D	D	D	d	+D
ENQ	$E	%	/E	E	E	e	+E
ACK	$F	&	/F	F	F	f	+F
BEL	$G	'	/G	G	G	g	+G
BS	$H	(	/H	H	H	h	+H
HT	$I	)	/I	I	I	i	+I
LF	$J	*	/J	J	J	j	+J
VT	$K	+	/K	K	K	k	+K
FF	$L	,	/L	L	L	l	+L
CR	$M	-	-	M	M	m	+M
SO	$N	.	.	N	N	n	+N
SI	$O	/	/O	O	O	o	+O
DLE	$P	0	0	P	P	p	+P
DC1	$Q	1	1	Q	Q	q	+Q
DC2	$R	2	2	R	R	r	+R
DC3	$S	3	3	S	S	s	+S
DC4	$T	4	4	T	T	t	+T
NAK	$U	5	5	U	U	u	+U
SYN	$V	6	6	V	V	v	+V
ETB	$W	7	7	W	W	w	+W
CAN	$X	8	8	X	X	x	+X
EM	$Y	9	9	Y	Y	y	+Y
SUB	$Z	:	/Z	Z	Z	z	+Z
ESC	%A	;	%F	[	%K	{	%P
FS	%B	<	%G	/	%L	:	%Q
GS	%C	=	%H	]	%M	}	%R
RS	%D	>	%I	^	%N	~	%S
US	%E	?	%J		%O	DEL	%T,%X,%Y,%Z

Note: Because all of the characters used to implement Full ASCII Code 39 are part of the Normal Code 39 character set, readers that do not support Full ASCII Code 39 will still read Full ASCII Code 39 symbols. The reader will output shifted characters as if they were normal Code 39 characters. For example the following string: PROGRAMMING=FUN will be read as: PROGRAMMING%HFUN by a reader that only supports Normal Code 39. Instead of converting the Full ASCII encoded characters %H to an equal sign, the reader blindly outputs %H.

HIBC Code 39

HIBC or "Health Industry Bar Code" is a specification for a standard method of encoding data using Code 39 in healthcare applications. The standard requires that the first character in

a HIBC bar code be a plus character (+) and that the symbol use a MOD 43 check digit. To generate HIBC bar codes, all you need to do is to set the bar code type to Code39_HIBC and VariableText will automatically add the starting plus sign to your bar code message as well as calculate and append the check digit for you.

Code 39 Symbol Concatenation

Some bar code readers support a feature of Code 39 that allows for concatenation of two or more bar codes so that you can break long messages into multiple, shorter, symbols. If the first data character of a Code 39 symbol is a space, some readers will store the remainder of the symbol in a buffer and not transmit the data.
This operation continues for all successive Code 39 symbols with a leading space, with each message appended to the previous one. When a message without a leading space is read, it is appended to the previously scanned data and the entire buffer is transmitted as one long message.

UPC-A, UPC-E, and UPC Supplemental

UPC-A is a 12 digit, numeric symbology used in retail applications. UPC-A symbols consist of 11 data digits and one check digit. The first digit is a number system digit that usually represents the type of product being identified. The following 5 digits are a manufacturer's code and the next 5 digits are used to identify a specific product. UPC numbers are assigned to manufacturers by the Uniform Code Council (UCC).

For more information, contact the UCC at 8163 Old Yankee Road, Suite J, Dayton, OH 45458
Tel: 937-435-3870, web site: http://www.uc-council.org

UPC-E is a smaller, six-digit UPC symbology for number system 0. It is often used for small retail items like candy and cigarettes. UPC-E is also called "zero suppressed" because UPC-E compresses a normal 12-digit UPC-A code into a six digit code by "suppressing" the number system digit, trailing zeros in the manufacturers code and leading zeros in the product identification part of the bar code. A seventh check digit is encoded into a parity pattern for the six main digits. Many bar code readers can be configured to automatically uncompress UPC-E symbols into their 12-digit equivalent before transferring the data to a computer.

When specifying UPC-A or UPC-E messages, you may pass a message with either 6,7,11 or 12 digits. If you pass a message with 6 or 7 digits VariableText will generate a UPC-E symbol. If you pass 7

digits the 7th digit will be removed. (The control will assume that you are entering a message complete with check digit.) If you pass a message with 11 or 12 digits VariableText will generate a UPC-A symbol. If you pass 12 digits the 12th digit will be removed. (Again the control will assume that you are entering a message with a check digit.) VariableText will automatically calculate the check digit for you and appends it to your bar code message text. This insures that you always have the correct check digit.
Both UPC-A and UPC-E allow for a supplemental two or five digit number to be appended to the main bar code symbol. The supplemental is a small additional bar code that is added onto the right side of a standard UPC symbol. This supplemental message was designed for use on publications and periodicals. To include a supplemental message, append it to the main message with a comma separating it from the main message. If you enter a supplemental message, it must consist of either two or five numeric digits.

The following table contains sample messages for the different variations of UPC symbols.

Message	Symbol Generated
123456	UPC-E
123456,12345	UPC-E with a five digit supplemental.
123456789012	UPC-A
12345678901,12	UPC-A with a two digit supplemental.

EAN-8 / EAN-13, and EAN Supplemental

EAN or European Article Numbering system (also called JAN in Japan) is a European version of UPC. It uses the same size requirements and a similar encoding scheme as UPC codes.

EAN-8 encodes 8 numeric digits consisting of two country code digits, five data digits and one check digit. EAN-13 is the European version of UPC-A. The difference between EAN-13 and UPC-A is that EAN-13 encodes a 13th digit into the parity pattern of the left six digits of a UPC-A symbol. This 13th digit, combined with the 12th digit, usually represent a country code.

EAN bar code numbers are assigned to specific products and manufacturers by an organization called ICOF located in Brussels, Belgium. Tel: 011-32-2218-7674, web site: http://www.ean.be/

When specifying EAN-8 or EAN-13 messages, you may pass a message with either 7, 8, 12 or 13 digits. If you pass a message with 7 or 8 digits VariableText will generate a EAN-8 symbol. If you pass 8 digits, the 8th digit will be removed. (VariableText will assume that you are entering a message complete with check digit.) If you pass a message with 12 or 13 digits VariableText will generate a EAN-13 symbol. If you pass 13 digits, the 13th digit will be removed. (Again the control will assume that you are entering a message with a check digit.) VariableText automatically calculates the check

digit for you and appends it to your bar code message text. This insures that you always have the correct check digit.

Like UPC, EAN-8 and EAN-13 allow for a supplemental two or five digit number to be appended to the main bar code symbol. The supplemental is simply a small additional bar code that is added onto the right side of a standard EAN symbol. To include a supplemental message, append it to the main message with a comma separating it from the main message. If you enter a supplemental message, it must consist of either two or five numeric digits.

The following table contains sample messages for the different variations of EAN symbols.

Message	Symbol Generated
1234567	EAN-8 no supplemental
1234567,12345	EAN-8 with a five digit supplemental.
123456789012	EAN-13 no supplemental
1234567890128	EAN-13 no supplemental
123456789012,12	EAN-13 with a two digit supplemental.

BookLand

EAN-13 has been adopted as the standard in the publishing industry for encoding ISBN numbers on books. An ISBN or BookLand bar code is simply an EAN-13 symbol consisting of the first 9 digits of an ISBN number preceded by the digits 978 and terminated with a standard EAN check digit. The supplemental in an ISBN bar code is usually the retail price of the book preceded by the digit 5 or it can be the number 90000 when no price is specified.
For example, if you want to encode the ISBN number 1-56276-008-4 and the price of the book is $29.95 then you could use 978156276008 as the main bar code message and 52995 for the supplemental with the Symbology property set to bcEAN_JAN_13 and VariableText would generate the correct BookLand bar code.

If you set the bar code type to BookLand, then VariableText only requires you to use the 10-digit ISBN number (with or without dashes) and optionally a price for the supplemental separated from the ISBN number with a comma. This feature saves you the trouble of having to remove the 10th digit from the ISBN number, insert the preceding 978 and further process the price supplemental.

The following table contains sample messages for the different variations of BookLand symbols. The following samples apply only when the bar code type is set to BookLand.

BookLand Message	Message in Generated BookLand Bar Code
1-56276-008-4	97815602763 no supplemental
1562760084,29.95	97815602763 with supplemental 52995
1-56276-008-4,2995	97815602763 with supplemental 52995
1-56276-008-4,395	97815602763 with supplemental 50395
1234567890,90000	9781234567897 with supplemental 90000
1234567890,12	9781234567897 with 2 digit supplemental 12

CODE 93

CODE 93 is a variable length symbology that can encode the complete 128 ASCII character set. Code 93 was developed as an enhancement to the CODE 39 symbology by providing a slightly higher character density than CODE 39. CODE 93 also incorporates two check digits as an added measure of security.
Although CODE 93 is considered more robust than CODE 39, it has never achieved the same popularity as Code 39. CODE 93 bar codes are framed by a special start/stop character. VariableText will automatically add the start and stop characters as well as the check digits to each Code 93 bar code therefore you should not attempt to include them as part of your bar code message.

CODABAR

CodaBar is a variable length symbology that allows encoding of the following 20 characters: 0123456789-$:/.+ABCD. CodaBar is commonly used in libraries, blood banks, and the air parcel business. CodaBar uses the characters A B C and D only as start and stop characters. Thus, the first and last digits of a CodaBar message must be A B C or D and the body of the message should not contain these characters. VariableText will allow any length of CodaBar message as long as it contains valid characters and starts and ends with a valid start/stop character. If you use lower case letters for A B C or D, VariableText will convert them to upper case.

INTERLEAVED 2 OF 5 (ITF)

Interleaved 2 of 5 is a high-density variable length numeric only symbology that encodes digit pairs in an interleaved manner. The odd position digits are encoded in the bars and the even position digits are encoded in the spaces. Because of this, Interleaved 2 of 5 bar codes must consist of an even number of digits. Also, because partial scans of I 2 of 5 bar codes have a slight chance of being decoded as a valid (but shorter) bar code, readers are usually configured to read a fixed (even) number of digits when reading Interleaved 2 of 5 symbols. The number of digits are usually pre-defined for a particular application and all readers used in the application are programmed to only accept ITF bar codes of the chosen length. Shorter data can be left padded with zeros to fit the proper length. VariableText will only accept numeric digits for Interleaved 2 of 5 bar codes. If an odd number of digits is entered, the control will Left-Pad one zero to the number entered.

Interleaved 2 of 5 optionally allows for a weighted modulo 10 check character for special situations where data security is important. To enable the Interleaved 2 of 5 check character, set the I2of5 Check Digit preference to True and VariableText will automatically calculate and append the proper check character to all Interleaved 2 of 5 symbols.

CODE 128

Code 128 is a variable length, high density, alphanumeric symbology. Code 128 has 106 different bar and space patterns and each pattern can have one of three different meanings depending on which of three different character sets is employed. Special start characters tell the reader which of the character sets is initially being used and three special shift codes permit changing character sets inside a symbol. One character set encodes all upper case characters and ASCII control characters, another encodes all upper and lower case characters and the third set encodes numeric digit pairs 00 through 99. This third character set effectively doubles the code density when printing numeric data. Code 128 also employs a check digit for data security. In addition to ASCII characters, Code 128 also allows encoding of four special function codes (FNC1 - FNC4). The meaning of function code FNC1 and FNC4 were originally left open for application specific purposes. Recently an agreement was made by the Automatic Identification Manufacturers Assoc. (AIM) and the European Article Numbering Assoc. (EAN) to reserve FNC1 for use in EAN applications. FNC4 remains available for use in closed system applications. FNC2 is used to instruct a bar code reader to concatenate the message in a bar code symbol with the message in the next symbol. FNC3 is used to instruct a bar code reader to

perform a reset. When FNC3 is encoded anywhere in a symbol, any data also contained in the symbol is discarded. The four function codes can be included in a message by using the ASCII characters ASCII 128 for FNC1, ASCII 129 for FNC2, ASCII 130 for FNC3 and ASCII 131 for FNC4.

Note: VariableText will automatically select the proper character sets and insert the necessary start character and shift codes so that the resulting bar code will be as short as possible. The check digit will also be calculated automatically by VariableText.

EAN/UCC 128

The EAN/UCC 128 symbology is a variation of the original Code 128 symbology designed primarily for use in product identification applications. The EAN/UCC 128 specification uses the same code set as Code 128 except that it does not allow function codes FNC2-FNC4 to be used in a symbol and FNC1 is used as part of the start code in the symbol. Some applications for EAN/UCC 128 require an additional Mod 10 check digit.
You can enable this check digit by setting the UccEan Check Digit property to True. If the optional check digit is enabled, the data that you supply in the Message property must consist of numeric characters only otherwise an error event is raised and no bar code symbol is generated.

MSI-PLESSEY

MSI-PLESSEY is a variable length, numeric only, symbology. The symbology is one of the earliest bar code symbologies ever developed and is based on a four bit binary number scheme. Each symbol is framed by a start and a stop pattern and contains a check character that is calculated from the values of each of the encoded data digits. MSI-Plessey is rarely used in anything other than grocery store shelf marking applications. Because the symbology is so rarely used, most modern bar code readers do not provide support for MSI-Plessey symbols.

POSTNET

POSTNET (Postal Numeric Encoding Technique) is a 5, 9 or 11 digit numeric only bar code symbology used by the US Postal Service to encode ZIP Code information for automatic mail sorting.

The bar code may represent a five digit ZIP Code (32 bars), a nine digit ZIP + 4 code (52 bars) or an eleven digit Delivery Point code (62 bars).

POSTNET is unlike other bar codes because data is encoded in the height of the bars instead of in the widths of the bars and spaces. Most commercially available bar code readers cannot decode POSTNET.

This symbology was chosen by the Postal Service mainly because it is extremely easy to print on almost any type of printer. POSTNET is a fixed dimension symbology meaning that the height, width and spacing of all bars must fit within tight tolerances. VariableText will only create POSTNET bar codes that follow the guidelines published by the Postal Service. The control does not allow you to set the size of POSTNET bar codes. In other words the Narrow Bar Width, Bar Width Reduction and the Bar Height preferences are ignored. POSTNET also does not support human readable text, quiet zones, bearer bars or optional check digits. VariableText will ignore non-numeric data in any POSTNET bar code message. For example, if you supply "Chicago, IL 60601-3222" for the Message property of a POSTNET bar code, the control will still create a correct bar code.

PDF417

PDF417 is a high density 2 dimensional bar code symbology that resembles a stacked set of smaller bar codes. The symbology is capable of encoding the entire (256 character) ASCII or ANSI character set. PDF stands for "Portable Data File" because it can encode as many as 2710 data characters in a single bar code.

The complete specification for PDF417 provides many encoding options including data compaction options, error checking and correction options, and variable size and aspect ratio symbols. The low level structure of a PDF417 symbol consists of an array of code words (small bar and space patterns) that are grouped together and stacked on top of each other to produce the complete printed symbol. An individual code word consists of a bar and space pattern 17 modules wide. The user may specify the module width, the module height, and the overall aspect ratio (overall height to width ratio) for the complete symbol. A complete PDF417 symbol consists of at least 3 rows of up to 30 code words and may contain up to 90 code word rows per symbol with a maximum of 928 code words per symbol.

The code words in a PDF417 symbol are generated using one of three data compaction modes currently defined in the symbology specifications. This allows more than one character to be encoded into a single data code word. Because different data compaction algorithms may be used, it is possible for different printed symbols to be created from the same input data. VariableText will automatically switch between the different data compaction modes to produce the smallest possible bar code symbol for a given bar code message.

The PDF417 symbology also allows for varying degrees of data security or error correction and detection.

Nine different error correction levels are available with each higher level adding additional overhead to the printed symbol. VariableText also provides an "automatic" error correction option that can help make the most efficient use of the error correction capabilities in PDF417.
VariableText allows complete control over all features of PDF417.

PDF417 Bar Code Dimensions

A PDF417 bar code symbol consists of multiple rows of data encoded in units called code words. Each symbol can contain from 3 to 90 rows and each row consists of a Start/Stop pattern and from 3 to 32 code words (1 to 30 code words for data and 2 for Right and Left Row Indicators). The smallest element in a PDF417 symbol is called a module. Each code word consists of a unique pattern of 4 bars and 4 spaces each with a width of up to 6 modules within a total width of 17 modules per code word. This is where the 417 comes from in the name PDF417 - 4 bars within 17 modules. Because the number of code words in a row and the total number of rows in a symbol are variable quantities, the three primary dimensions used to define the size of a PDF417 bar code are the PDF Module Width, PDF Module Height, and the PDF Aspect Ratio preferences. The PDF Module Height and PDF Module Width values define the height and width of the smallest rectangular element in the PDF417 symbol and the PDF Aspect Ratio value specifies the overall height to width ratio of the symbol.

The best choice for the PDF Module Width preference depends partly on the resolution of your bar code reading equipment and also on the resolution of the printer being used to produce the bar code. The specification for PDF417 recommends that the module width should fall in a range between 10 and 30 mils. The smallest allowable module width defined in the symbology specification is 6.56 mils. (This translates to 2 printer dots when printing to a 300 DPI laser printer.) The best way to determine the ideal module width for your application is to actually print out a sample bar code using several different values and try reading each one with your scanning equipment. Again, you should choose the value that produces bar codes with the best read rate.

The recommended value for the PDF Module Height is three times the value for the PDF Module Width however the symbol specifications allow for module heights as small as 10 mils (3 printer dots on a 300 DPI laser printer).

The value for the PDF Aspect Ratio property determines the overall shape of the PDF417 symbol and is defined as the overall height to width ratio. Higher values for the PDF Aspect Ratio (greater than 1) produce tall, thin PDF417 bar codes and small values (greater than zero and less than 1) produce short, wide bar codes. A value of 1 should produce approximately square bar codes. Because each row of data in a PDF417 symbol contains 2 additional code words of overhead (right and left row indicators) as well as start and stop patterns, tall and thin

bar codes with lots of rows (high aspect ratios) will be larger in total area and contain more overhead than short wide bar codes (low aspect ratios). On the other hand, short wide bar codes (low aspect ratios) may be difficult to read depending on the type of scanner that you use.

NOTE: For any given PDF417 bar code symbol there are only a small number of possible aspect ratios that are physically possible. Because of this, you will probably not be able to produce a bar code with the exact PDF Aspect Ratio that you specify. VariableText will automatically generate a bar code with the closest match to the PDF Aspect Ratio value that you specify, within the limits of the symbol specification.

VariableText also provides more detailed control over the size and shape of a PDF417 symbol by allowing you to specify the maximum number of rows and a maximum number of data code word columns in a symbol using the PDF Max Rows and PDF Max Cols properties. The default maximum number of data rows and columns are the maximum values allowable in the PDF417 symbol specification (i.e. 30 rows and 90 Columns). If you specify smaller values for these parameters, you are essentially defining an upper limit to the overall height or width of all generated PDF symbols. For example, suppose you need to generate a PDF symbol that absolutely must be no wider than two inches and you chose 10 mils for the PDF Module Width property. Because there are 17 modules in a code word and there are 69 modules of overhead per row, (start/stop patterns and right/left row indicators) the maximum number of code word columns allowable in a 2 inch wide symbol can be calculated using the formula:

10mils x [(17 x PDF Max Cols) + 69]<=2000 mils (2 inches)

If we solve the equation above we obtain the value of 7 for the Maximum number of code word columns (PDF Max Cols). Thus, if we set the Maximum Number of Data Columns parameter to 7, VariableText will only produce PDF417 symbols that are less than 2 inches wide.

You can also cause VariableText to produce a version of PDF417 called Truncated PDF417 that removes the Right Row Indicator code words and the stop pattern on the right hand side of the symbol thus reducing the total size of a PDF417 symbol. When creating Truncated PDF417 symbols, there are only 35 overhead modules per row instead of the normal 69 overhead modules. This option can be selected by setting the PDF Truncated Symbol preference to True. Truncated is shown below.

PDF417 Error Detection and Correction

One of PDF417's primary features is error detection and correction or data security. Each PDF417 symbol has 2 code words for error detection. (Similar to a check digit in standard bar code symbols.) The error correction capacity may be selected by the user, based on the needs of a specific application. Error correction compensates for label defects and misdecodes. There are essentially 2 types of errors that can occur in a bar code symbol, Erasures and Misdecodes. Erasures are missing or undecodable

code words and misdecodes are errors that cause the reader to interpret a particular code word incorrectly. Nine error correction levels numbered 0 through 8 are available in the symbol specification. Each higher security level allows for a higher number of erasures and misdecodes to be recovered from. (Since it requires 2 code words to recover from a misdecode, one to detect the error and one to correct for it, a given security level can support half the number of misdecodes that it can undecoded or missing code words.)

Since error correction is encoded into additional code words, each higher security level adds additional overhead to a printed symbol. Higher security levels reduce the maximum number of data characters that can be encoded and also increase the size of a printed symbol. (A PDF417 symbol can contain a maximum of 928 total code words for data and error correction combined.)

The relationship between security level, error correction capacity and the number of additional code words or overhead required for a given security level is as follows:

Error Correction Level	Maximum Limit of Allowable Erasures + (2 x Misdecodes)	Symbol Overhead (Number of Additional Code words)
0	0	2
1	2	4
2	6	8
3	14	16
4	30	32
5	62	64
6	126	128
7	254	256
8	510	512

The PDF Security Level property allows you to select a specific PDF417 error correction level from 0 to 8. You can also have the VariableText automatically select a security level based on a percentage of total symbol area to be used for error correction. If you set the PDF Security Level property to 9, you can then pass a percentage value in the PDF Pct Overhead property (from 0% to 99%). This "automatic" PDF Security Level mode is the best way to specify a security level because it guarantees that you will not waste symbol real estate with more error correction overhead than is necessary for small messages. It also insures that enough error correction will be generated for larger messages - A small message only needs a small amount of error correction capacity while a larger message needs more. If you set the value for the PDF Pct Overhead property to zero, the default percentage of 10% overhead will be used.

Aztec Code

Aztec Code is a high density 2 dimensional matrix style symbology that can encode up to 3750 characters from the entire 256 byte ASCII character set. The symbol is built on a square grid with a bulls-eye pattern at its center. Data is encoded in a series of "layers" that circle around the bulls-eye pattern. Each additional layer completely surrounds the previous layer thus causing the symbol to grow in size as more data is encoded. Aztec's primary features include: a wide range of sizes allowing

both small and large messages to be encoded, orientation independent scanning and a user selectable error checking and correction mechanism.

The smallest element in an Aztec symbol is called a "module" (i.e. a square dot). The module size and the amount of error correction are the only "dimensions" that can be specified for an Aztec symbol and both are user selectable. It is recommended that the module size should range between 15 to 30 mils in order to be readable by most of the scanners that are currently available. The module size is specified using the Matrix Module Size property.

The overall size of an Aztec symbol is dependent on the Matrix Module Size setting, the total amount of encoded data and also on the level of error correction capacity chosen by the user. The smallest Aztec symbol is 15 modules square and can encode up to 14 digits with 40% error correction. The largest symbol is 151 modules square and can encode 3000 characters or 3750 numeric digits with 25% error correction. There are four types of Aztec symbols, “Normal”, "Compact", "Full Range" and "Menu" symbols. You specify the type of Aztec symbol to generate using the Aztec Symbol Type property. Normal Aztec symbols can grow or shrink in size depending on the amount of data that you are encoding. With Normal Aztec symbols you also specify the amount of error correction overhead to include by specifying the percentage of the total symbol area to use for error correction. The Aztec Error Correction property is used for this purpose. Compact symbols have a smaller bulls-eye pattern and are limited in their overall size to having up to four "layers" of data surrounding the bulls-eye pattern. Full Range symbols have a larger bulls-eye pattern and can have up to 32 layers of data surrounding the bulls-eye. Do not confuse "Layers" with modules. A layer actually is made up of two stacked modules resembling a domino with each domino laid out around the bulls-eye pattern so that the long edge of the domino points away from the center of the symbol. If you set VariableText to generate Compact or Full Range Aztec symbols, you must also specify the total number of data layers in the symbol using the Aztec Total Layers property. If the total number of data layers is greater than the amount of layers required to encode a particular message, the remaining layers are filled with error correction data. If the amount of data that you need to encode will not fit in the number of data layers that you specify, an error event will be raised and no bar code symbol will be generated. Menu symbols are a special type of Aztec bar code that are typically used by scanner manufacturers to create bar codes that contain commands for enabling and disabling features in a bar code reader. Menu symbols can be either Compact or Full Range symbols and are only useful if you know the commands that a particular reader will recognize. VariableText does not provide support for Aztec Menu symbols.

Data Matrix

Data Matrix is a high density 2 dimensional matrix style bar code symbology that can encode up to 3116 characters from the entire 256 byte ASCII character set. The symbol is built on a square grid arranged with a finder pattern around the perimeter of the bar code symbol.

There are two types of Data Matrix symbols each using a different error checking and correction scheme (ECC). The different types of Data Matrix symbols are identified using the terminology "ECC" followed by a number representing the type of error correction that is used by the encoding software. ECC 000 to ECC 140 are the original type of Data Matrix symbols and are now considered obsolete. The newest version of Data Matrix is called ECC 200 and is recommended for all new Data Matrix applications. The ECC 200 version of Data Matrix uses a much more efficient algorithm for encoding data in a symbol as well as an advanced error checking and correction scheme. VariableText fully supports all variations of the Data Matrix symbology however the author of the original symbology specification (CI Matrix Co.) highly recommends that ECC 000 - ECC 140 be used only where absolutely necessary to remain compatible with older systems.

VariableText allows you to select the ECC value using the Data Matrix Error Correction property. Each of the different ECC options are outlined below:

ECC 000 - ECC 140

Data Matrix symbols designated by the terms ECC 000 to ECC 140 are the original type of Data Matrix symbol that uses a convolutional error correction scheme. There are actually five levels of error correction available for this type of Data Matrix symbol with each higher level of error correction designated as follows.

ECC 000	Provides no error correction
ECC 050	Provides error correction for damage of up to 2.8% of the printed symbol.
ECC 080	Provides error correction for damage of up to 5.5 % of the printed symbol.
ECC 100	Provides error correction for damage of up to 12.6% of the printed symbol.
ECC 140	Provides error correction for damage of up to 25% of the printed symbol.

For all five ECC levels ECC 000 - ECC 140 there is also a user selectable option for a "Data Format" which defines the type of data that may be encoded in a Data Matrix symbol. The data format is selected using the Data Matrix Format ID property. The available formats are listed below:

Format ID	Allowable Data In The Bar Code Message
1	Numeric digits 0 to 9 and the space character
2	Upper case alpha A-Z and the space character
3	Upper case alphanumeric A-Z, 0-9 and the space character
4	A-Z, 0-9, space, minus, period, comma & forward slash (/)
5	7 bit ASCII - all ASCII characters between ASCII 0 to ASCII 127
6	8 bit ASCII - all ASCII characters between ASCII 0 to ASCII 255

ECC200

ECC 200 is the latest and most advanced version of Data Matrix and is therefore strongly recommended for use in all new Data Matrix applications. ECC 200 uses a Reed-Solomon error correction algorithm that will automatically provide a varying degree of error correction for damage ranging from a minimum of roughly 20% to greater than 60% damage depending on the amount of data encoded ECC200 also uses a code set switching mechanism that is much more efficient at packing data into a symbol than any of the earlier encoding schemes (ECC 000 - ECC 140). ECC 200

is capable of encoding the entire 8 bit ASCII character set in a highly efficient manner and therefore does not provide for or require any Data Format options. In addition to encoding 8 bit ASCII data, ECC 200 also allows for a special function code called "FNC1" as well as special indicators for features like "Structured Append" and "Extended Channel Interpretation".
Structured append is a means for generating multiple bar codes containing a much larger data message than can be encoded into a single symbol. With structured append, each bar code contains a portion of a larger data message along with a number that identifies the portion of the message contained in the symbol.
When a bar code reader scans a symbol that is part of a sequence, it will not transmit the data until all symbols in a sequence have been read. It does not matter what order the bar codes are read in - the reader will correctly build the complete message.
Extended Channel Interpretation is a mechanism for creating user definable data encoding schemes. For example, suppose you wanted to replace the standard ASCII character set with a different character set (a foreign language character set for example), you could use the Extended Channel Interpretation feature in ECC 200 to indicate that the message encoded in a symbol conforms to the new interpretation.
Notes: To encode data to conform to specific industry standard it needs to be authorized by AIM International.

VariableText supports all features of the ECC 200 version of Data Matrix. Because ECC 200 supports the entire 8 bit ASCII character set and therefore cannot use ASCII character values to represent special features (like FNC1), the control provides two methods for interpreting the input data message. The control provides a Data Matrix Tilde Codes property that determines how the input data message is interpreted. If the Data Matrix Tilde Codes property is set to False then the Message property is assumed to not contain any special function codes like FNC1 or Extended Channel Interpretation codes. For most normal applications this option would normally be used.
If the Data Matrix Tilde Codes property is set to True then the tilde character (~) can be used in the input message as an indicator that the character(s) following the tilde are to be interpreted with a special meaning as outlined below. To encode a tilde (~) use the string: ~~ (i.e. two tilde characters). If no tilde characters or Nulls (ASCII 0) are present in the input message, then setting the Data Matrix Tilde Codes property to True has no effect on the resulting bar code symbol.

Tilde Control Codes

~X (a tilde character followed by any upper case alpha character) is used as a shift character for inserting control codes (characters with ASCII values 0 to 26) into a bar code message. For example, ~@ = NUL, ~A = ASCII 1,~G = BEL (ASCII 7), ~M = ASCII 13 (carriage return). If you need to insert ASCII control codes into a message, take the ASCII value for the control code (1-26) and find the corresponding letter in the alphabet and precede it with a tilde. i.e. The ASCII value for a carriage return character is ASCII 13 and the thirteenth letter of the alphabet is "M" therefore to insert a carriage return in a bar code message, you would use "~M". Note: You can also pass control codes directly to the DLL without having to use the ~ before an alpha character. For example you could use either an ASCII 13 character or the sequence ~M to represent a carriage return.

~1 is used to represent the FNC1 code and is followed by normal data.\

To encode data to conform to specific industry standards as authorized by AIM International, a FNC1 character shall appear in the first or second symbol character position (or fifth or sixth data position of the first symbol of structured append). FNC1 encoded in any other position is used as a field separator and shall be transmitted as a GS control character (ASCII value 29).

Notes: To encode data to conform to specific industry standard, it needs to be authorized by AIM International. Contact AIM International at Tel: 703-391-7621 or email: adc@aimi.org
If the FNC1 code is used in the second character position, the input data before '~1' must be, between 'A' and 'Z', or between 'a' and 'z' or 2-digits between '01' and '99'. ~2 : is used to represent Structured Append and must be followed by a three 3-digit number between 1 and 255 representing the symbol sequence as well as a file identifier of six numeric digits. The file identifier is used to uniquely identify a sequence so that only logically linked sequences are processed as part of the same sequence. The symbol sequence identifier is a number between 1 and 255 that indicates the position of the symbol within a sequence of up to 16 symbols. The sequence identifier actually contains two four bit values representing the sequence number and the total number of symbols in the sequence (i.e. m of n where m is the sequence number and n is the total number of symbols). The upper four bits of this value represent the position of the particular symbol as the binary value of (m-1) and the lower order four bits identify the total number of symbols to be concatenated as the binary value of (17-n). For example, symbol 3 in a sequence of 7 symbols with file ID: 001015 is represented by ~2042001015. The number 042 is derived as follows: 3-1=2 which equals 0010 when represented as a 4 bit binary number. 17-7=10, which equals 1010, when represented as a 4 bit binary number. After concatenating the two 4 bit binary values we end up with 00101010 which equals 42 in decimal.
~3 : Indicates that a message is to be used for reader programming purposes and is followed by normal data. This feature is only useful if you know the specific programming commands for your bar code reader. ~5 and ~6 : indicates that the data will contain an abbreviated format header and trailer followed by normal data. The ~5 or ~6 must appear as the first two characters in a message and must not be used in conjunction with structured append. Data Matrix provides a means of abbreviating an industry specific header and trailer in one symbol character. This feature exists to reduce the number of characters needed to encode data using certain structured formats. If a ~5 is used as the first two characters of a message, the header [)> + ASCII 30 + 05 + ASCII 29 will be transmitted by the reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted following the data. Likewise, if a ~6 is used as the first two characters of a message, the header [)>+ ASCII 30 + 06 + ASCII 29 will be transmitted by the reader before the data in the message and the trailer ASCII 30 + ASCII 4 will be transmitted following the data.

~7NNNNNN is used to indicate Extended Channel NNNNNN where NNNNNN is 6-digit EC value (000000 - 999999). e.g. Extended Channel 9 is represented by ~7000009 ~dNNN creates ASCII decimal value NNN for a codeword (must be 3 digits).

Please refer to the official Data Matrix symbology specification for details on the meanings of all codeword values for ECC 200. Contact AIM International at Tel: 703-391-7621 or email: adc@aimi.org

Rectangular Data Matrix Symbols

The ECC 200 version of Data Matrix supports the ability to produce rectangular bar codes. VariableText uses the Data Matrix Format ID property to indicate the overall shape of all ECC200 bar codes. Six options are available that force the bar code symbol to be both fixed in size and rectangular in shape. The overall size of the rectangular symbols is dependent on the size of the module that you have selected for the Matrix Module Size property. The table below shows the 6 different rectangular symbol options available along with size in modules and the amount of data that may be encoded for each option.

Format ID	Shape of bar code and amount of data that may be encoded
1 - 6	Square - symbols grow in size depending on amount of data being encoded.
7	Rectangular - 8 modules tall by 18 modules wide - may contain up to 10 bytes
8	Rectangular - 8 modules tall by 32 modules wide - may contain up to 20 bytes
9	Rectangular -12 modules tall by 26 modules wide - may contain up to 32 bytes
10	Rectangular -12 modules tall by 36 modules wide - may contain up to 44 bytes
11	Rectangular -16 modules tall by 36 modules wide - may contain up to 64 bytes
12	Rectangular -16 modules tall by 48 modules wide - may contain up to 98 bytes

MaxiCode

MaxiCode is a fixed size matrix style symbology made up of rows of hexagonal modules arranged around a bulls-eye finder pattern. A MaxiCode symbol has 884 hexagonal modules arranged in 33 rows with each row containing up to 30 modules. The maximum data capacity for a MaxiCode symbol is 93 Alphanumeric or 138 Numeric characters. The symbology was designed by United Parcel Service for package tracking applications. The design of the MaxiCode symbology was chosen because it is well suited to high speed, orientation independent scanning. Although the capacity of a MaxiCode symbol is not as high as other matrix style symbologies, it was primarily designed to encode address data, which rarely requires more than 80 characters. MaxiCode symbols actually encode two messages; a Primary and a Secondary message. The Primary message normally encodes a postal code, a 3 digit country code and a 3 digit class of service number. The Secondary message normally encodes an address and any other required data.

MaxiCode Modes:

MaxiCode has several "Modes" that determine how data is encoded in the symbol. The original MaxiCode specification supported modes 0 and 1, which are now obsolete and are therefore not supported by the VariableText. The mode is selected using the Maxi Code Mode property. The current specification supports the following Modes:

Mode 2	Structured Carrier Message – Numeric Postal Codes (up to 9 digits)
Mode 3	Structured Carrier Message – AlphaNumeric Postal Codes (up to 6 characters)
Mode 4	Standard Symbol – Standard Error Correction
Mode 5	Standard Symbol – Enhanced Error Correction (not supported)
Mode 6	Reader Programming Mode

Modes 2 and 3 are reserved for use as a destination sortation symbol for use by carriers in the transportation industry. In Modes 2 and 3 a postal code, a country code and a service class number must be supplied along with a secondary message usually consisting of an address.
Mode 4 is designed to encode data as a "standard bar code symbol" where the data encoded in the symbol is not restricted to a specific application. In other words, Mode 4 should be used in all general purpose bar code applications other than transportation industry applications. Mode 5 is similar to mode 4 except that a higher level of error correction is employed. Because more symbol real estate us used for error correction in Mode 5, the amount of actual data that can be encoded in a Mode 5 symbol is reduced. The VariableText does not support Mode 5.
Mode 6 is reserved for bar code reader programming purposes and it has been left up to the bar code reader manufacturers to determine how to interpret data encoded using Mode 6.

Structured Carrier Messages in Mode 2 and 3 MaxiCode Symbols:

VariableText provides input properties for each of the three required Primary message fields (Postal Code, Country Code and Service Class) as well as a property for the secondary message (the Message property). United Parcel Service has defined a special standard for formatting data in a MaxiCode symbol for use in UPS package tracking applications. The standard defines several things including the content and format of the data in the primary and secondary messages, a special sequence of "header" characters, and the specific characters to be used as delimiters or format codes and application identifiers.

For complete details on how to encode Structured Carrier Messages conforming to the UPS standards, please contact your local United Parcel Service representative or visit the UPS web site at http://www.maxicode.com.

VariableText does not attempt to format the data in any way except in a special case as outlined below. In other words, when you use VariableText to produce a Mode 2 or Mode 3 bar code, you must provide all required data for the primary message and the data in the secondary message must contain all required header, delimiter and special formatting characters.
As a convenience VariableText will automatically parse out a Postal Code, the Country Code and the Service Class from any message that is passed in as a structured carrier message starting with the character sequence: []>RS01 GS yy. In other words, if the input message (passed in the Message property) to the control starts with the string: []>RS 01 GS yyaaaaaaaaa GS bbb GS ccc GS where yy is a two digit year, aaaaaaaaa is a valid postal code, bbb is a three digit country code and ccc is a class value, VariableText will automatically set the Postal Code, Country Code and Service Class parameters for the primary message to the values embedded in the input message. These values will also be removed from the secondary message.
(Note: The RS and GS symbols used here represent ASCII characters 30 and 29 respectively.)
For example, if the following message is supplied in the Message property, the Postal Code, Country Code and Service Class will be extracted from the Message and placed in the Maxi Zip Code, Maxi

Country Code and Maxi Code Class properties automatically so that you do not need to fill in these properties in advance.

]>RS01GS98152382802 GS 840GS001GS1Z00004951GSUPSNGS06X710GS159GS1234567GS 1/1GSGSYGS634 ALPHA DRIVEGSPITTSBURGHGSPARSEOT

If the above message is supplied in the Message property with the Maxi Code Mode property set to 2 or 3, the Postal Code value "152382802" and the Country Code value "840" and the Service Class value "001" will be removed from the secondary message and automatically encoded in the primary message. The actual message that will be encoded in the secondary message will be:
]>RS01 GS981Z00004951 GS UPSN GS 06X710 GS 159 GS 1234567 GS 1/1 GS GS Y GS 634 ALPHA DRIVEGSPITTSBURGHGSPA RS EOT
Note: In Mode 2, the Postal Code parameter must be either a 5 or 9 digit number (all numeric characters) and in Mode 3, the Postal Code may contain up to 6 Alphanumeric characters.

Quick Response Code (QR Code)

This taken from Wikipedia with their permission subject to the following licensing agreement:
http://creativecommons.org/licenses/by-sa/3.0/

http://en.wikipedia.org/wiki/QR_code

QR code (abbreviated from **Quick Response Code**) is the trademark for a type of matrix barcode (or two-dimensional bar code) first designed for the automotive industry in Japan. Bar codes are optical machine-readable labels attached to items that record information related to the item. It was initially patented; however, its patent holder has chosen not to exercise those rights.[1] Recently, the QR Code system has become popular outside the automotive industry due to its fast readability and greater storage capacity compared to standard UPC barcodes. The code consists of black modules (square dots) arranged in a square grid on a white background. The information encoded may be made up of four standardized types ("modes") of data (numeric, alphanumeric, byte / binary, Kanji) or, through supported extensions, virtually any type of data.[2]

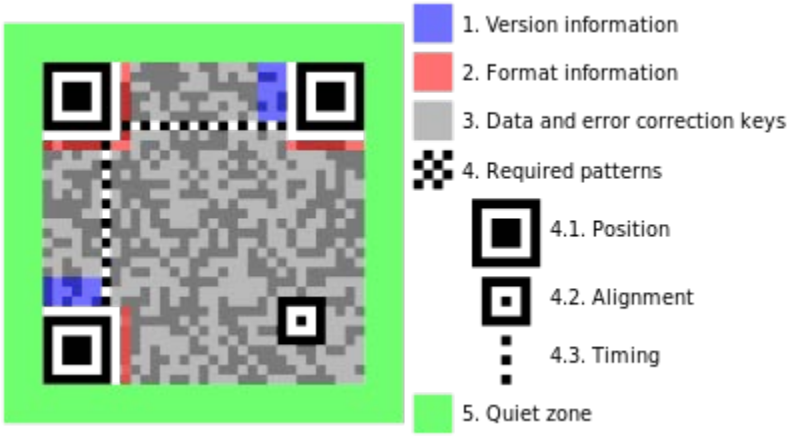
A QR code is read by an imaging device, such as a camera, and formatted algorithmically by underlying software using Reed-Solomon error correction until the image can be appropriately interpreted. Data is then extracted from patterns present in both horizontal and vertical components of the image. As a variety of industries utilize the QR code today, the applications for use can vary from product tracking, item identification, time tracking, document management and general marketing purposes.[3]

Invention

The QR code was invented in Japan by the Toyota subsidiary Denso Wave in 1994 to track vehicles during manufacture. It was designed to allow high-speed component scanning.[4] It has since become one of the most popular types of two-dimensional barcodes.[5]

Unlike the older one-dimensional barcode that was designed to be mechanically scanned by a narrow beam of light to extract data, the QR code is detected as a 2-dimensional digital image by a semiconductor image sensor and is then digitally analyzed by a programmed processor. The processor locates the three distinctive squares at the corners of the image, and uses a smaller square near the fourth corner to normalize the image for size, orientation, and angle of viewing. The small dots are then converted to binary numbers and their validity checked with an error-correcting code.

Standards



There are several standards in documents covering the physical encoding of QR codes:[6]

- October 1997 – AIM (Association for Automatic Identification and Mobility) International[7]
- January 1999 – JIS X 0510
- June 2000 – ISO/IEC 18004:2000 *Information technology – Automatic identification and data capture techniques – Bar code symbology – QR code* (now withdrawn)
  Defines QR code models 1 and 2 symbols.
- 1 September 2006 – ISO/IEC 18004:2006 *Information technology – Automatic identification and data capture techniques – QR code 2005 bar code symbology specification*
  Defines QR code 2005 symbols, an extension of QR code model 2. Does not specify how to read QR code model 1 symbols, or require this for compliance.

At the application layer, there is some variation between most of the implementations. Japan's NTT DoCoMo has established de facto standards for the encoding of URLs, contact information, and several other data types.[8] The open-source "ZXing" project maintains a list of QR code data types.[9]

Uses

Originally designed for industrial uses, QR codes have become common in consumer advertising. Smartphone users can install an app with a QR-code scanner that can read a displayed code and convert it to a URL directing the smartphone's browser to the website of a company, store, or product associated with that code providing specific information.

"In the shopping industry, knowing what causes the consumers to be motivated when approaching products by the use of QR codes, advertisers and marketers can use the behavior of scanning to get consumers to buy, causing it to have the best impact on ad and marketing design."[10] As a result, the QR code has become a focus of advertising strategy, since it provides quick and effortless access to the brand's website.[11][12] Beyond mere convenience to the consumer, the importance of this capability is that it increases the conversion rate (that is, increases the chance that contact with the advertisement will convert to a sale), by coaxing qualified prospects further down the conversion funnel without any delay or effort, bringing the viewer to the advertiser's site immediately, where a longer and more targeted sales pitch may continue.

Although initially used to track parts in vehicle manufacturing, QR codes are now (as of 2012) used over a much wider range of applications, including commercial tracking, entertainment and transport ticketing, product/loyalty marketing (examples: mobile couponing where a company's discounted and percent discount can be captured using a QR code decoder which is a mobile app, or storing a company's information such as address and related information alongside its alpha-numeric text data as can be seen in Yellow Pages directory), and in-store product labeling. It can also be used in storing personal information for use by government. An example of this is Philippines National Bureau of Investigation (NBI) where NBI clearances now come with a QR code. Many of these applications target mobile-phone users (via mobile tagging). Users may receive text, add a vCard contact to their device, open a Uniform Resource Identifier (URI), or compose an e-mail or text message after scanning QR codes. They can generate and print their own QR codes for others to scan and use by visiting one of several pay or free QR code-generating sites or apps. Google has a popular API to generate QR codes,[13] and apps for scanning QR codes can be found on nearly all smartphone devices.[14]

QR codes storing addresses and Uniform Resource Locators (URLs) may appear in magazines, on signs, on buses, on business cards, or on almost any object about which users might need information. Users with a camera phone equipped with the correct reader application can scan the image of the QR code to display text, contact information, connect to a wireless network, or open a web page in the telephone's browser. This act of linking from physical world objects is termed hardlinking or object hyperlinking. QR codes also may be linked to a location to track where a code has been scanned. Either the application that scans the QR code retrieves the geo information by using GPS and cell tower triangulation (aGPS) or the URL encoded in the QR code itself is associated with a location.[15]

QR codes have been used and printed on Chinese train tickets since 2010[16]

In June 2011, The Royal Dutch Mint (Koninklijke Nederlandse Munt) issued the world's first official coin with a QR code to celebrate the centennial of its current building and premises. The coin was able to be scanned by a smartphone and link to a special website with contents about the historical event and design of the coin.[17] This was the first time a QR code was used on currency.

Mobile operating systems

QR codes can be used in Google's Android operating system and iOS devices (iPhone/iPod/iPad), as well as by using Google Goggles, 3rd party barcode scanners, and the Nintendo 3DS. The browser supports URI redirection, which allows QR codes to send metadata to existing applications on the device. mbarcode[18] is a QR code reader for the Maemo operating system. In Apple's iOS, a QR code reader is not natively included, but more than fifty paid and free apps are available with both the ability to scan the codes and hard-link to an external URL. Google Goggles is an example of one of many applications that can scan and hard-link URLs for iOS and Android. With BlackBerry devices, the App World application can natively scan QR codes and load any recognized Web URLs on the device's Web browser. Windows Phone 7.5 is able to scan QR codes through the Bing search app.

URLs

URLs aided marketing conversion rates even in the pre-smartphone era but during those years faced several limitations: ad viewers usually had to type the URL and often did not have a web browser in front of them at the moment they viewed the ad. The chances were high that they would forget to visit the site later, not bother to type a URL, or forget what URL to type. Friendly URLs decreased these risks but did not eliminate them. Some of these disadvantages to URL conversion rates are fading away now that smartphones are putting web access and voice recognition in constant reach. Thus an advert viewer need only reach for his or her phone and speak the URL, at the moment of ad contact, rather than remember to type it into a PC later.[19]

Virtual stores

During the month of June 2011, according to one study, 14 million mobile users scanned a QR code or a barcode. Some 58% of those users scanned a QR or bar code from their homes, while 39% scanned from retail stores; 53% of the 14 million users were men between the ages of 18 and 34.[20] The use of QR codes for "virtual store" formats started in South Korea,[21] and Argentina,[22] but is currently expanding globally.[23] Big companies such as Walmart, Procter & Gamble and Woolworths have already adopted the Virtual Store concept.[24]

Code payments

QR codes can be used to store a bank account information or a credit card information, or they can be specifically designed to work with particular payment provider applications. There are several trial applications of QR code payments across the world.[25][26]

In November 2012, QR code payments were deployed on a larger scale in the Czech Republic when an open format for payment information exchange - a Short Payment Descriptor - was introduced and endorsed by the Czech Banking Association as the official local solution for QR payments.[27]

Website login

QR codes can be used to log in into websites: a QR Code is shown on the login page on a computer screen, and when a registered user scans it with a verified smartphone, they will automatically be logged in on the computer. Authentication is performed by the smartphone which contacts the server. A QR code login method called "Sesame" was trialled by Google in January 2012.[28]

Storage

The amount of data that can be stored in the QR code symbol depends on the datatype (*mode*, or input character set), version (1, …, 40, indicating the overall dimensions of the symbol), and error correction level. The maximum storage capacities occur for 40-L symbols (version 40, error correction level L):[5][29]

Maximum character storage capacity (40-L)
character refers to individual values of the input mode/datatype

Input mode	max. characters	bits/char	possible characters, default encoding
Numeric only	7,089	3⅓	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Alphanumeric	4,296	5½	0–9, A–Z (upper-case only), space, $, %, *, +, -, ., /, :
Binary/byte	2,953	8	ISO 8859-1
Kanji/kana	1,817	13	Shift JIS X 0208

Here are some sample QR code symbols:

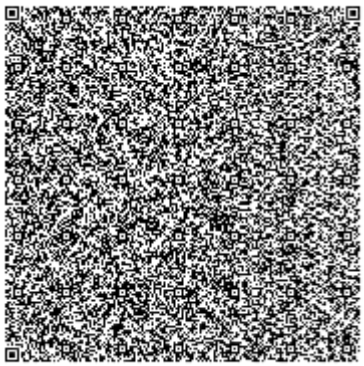
- Version 1 (21×21). Content: "Ver1"

- Version 2 (25×25). Content: "Version 2"

- Version 3 (29×29). Content: "Version 3 QR code"

- Version 4 (33×33). Content: "Version 4 QR code, up to 50 char"

- Version 10 (57×57). Content: "Version 10 QR code, up to 174 char at H level, with 57x57 modules and plenty of error correction to go around. Note that there are additional tracking boxes"



-

Version 40 (177×177). Content: "Version 40 QR code can contain up to 1852 chars. (…)" (a total of 1,264 characters of ordinary/ASCII text, taken from an early version of this Wikipedia article)

Encryption

Japanese visa with a QR code

Encrypted QR codes, which are not very common, have a few implementations. An Android app,[30] for example, manages encryption and decryption of QR codes using the DES algorithm (56 bits).[31] The Japanese immigration system uses encrypted QR codes when issuing visa in passports [32] as shown in the figure here.

Error correction

Damaged but still decodable QR code

Example of a QR code with artistic embellishment that will still scan correctly thanks to error correction.

Codewords are 8 bits long and use the Reed–Solomon error correction algorithm with four error correction levels. The higher the error correction level, the less storage capacity. The following table lists the approximate error correction capability at each of the four levels:

Level L (Low)	7% of codewords can be restored.
Level M (Medium)	15% of codewords can be restored.
Level Q (Quartile)[33]	25% of codewords can be restored.
Level H (High)	30% of codewords can be restored.

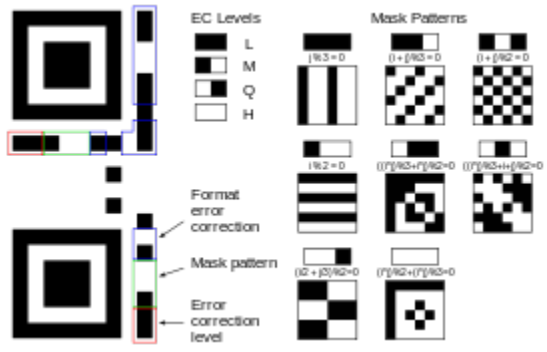
Due to the design of Reed–Solomon codes and the use of 8-bit codewords, an individual code block cannot be more than 255 codewords in length. Since the larger QR symbols contain much more data than that, it is necessary to break the message up into multiple blocks. The QR specification does not use the largest possible block size, though; instead, it defines the block sizes so that no more than 30 error-correction symbols appear in each block. This means that at most 15 errors per block can be corrected, which limits the complexity of certain steps in the decoding algorithm. The code blocks are then interleaved together, making it less likely that localized damage to a QR symbol will overwhelm the capacity of any single block.

Thanks to error correction, it is possible to create artistic QR codes that still scan correctly, but contain intentional errors to make them more readable or attractive to the human eye, as well as to incorporate colors, logos, and other features into the QR code block.[34][35]

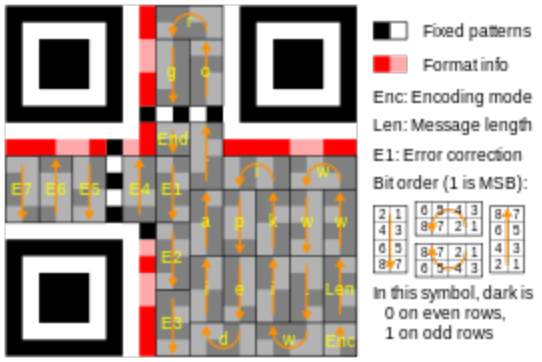
Encoding

The format information records two things: the error correction level and the mask pattern used for the symbol. Masking is used to break up patterns in the data area that might confuse a scanner, such as large blank areas or misleading features that look like the locator marks. The mask patterns are defined on a grid that is repeated as necessary to cover the whole symbol. Modules corresponding to the dark areas of the mask are inverted. The format information is protected from errors with a BCH code, and two complete copies are included in each QR symbol.[3]

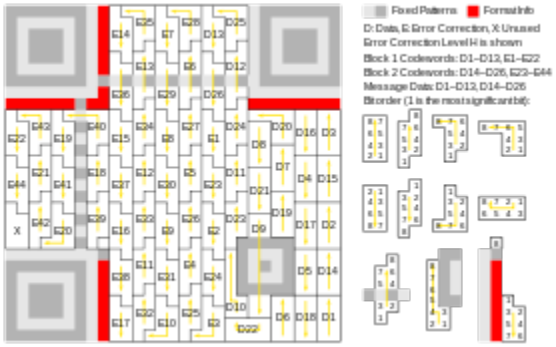
The message data is placed from right to left in a zigzag pattern, as shown below. In larger symbols, this is complicated by the presence of the alignment patterns and the use of multiple interleaved error-correction blocks.



- Meaning of format information



- Message placement within a QR symbol



- Larger symbol illustrating interleaved blocks

Four-bit indicators are used to select the encoding mode and convey other information. Encoding modes can be mixed as needed within a QR symbol.

Encoding modes

Indicator	Meaning
0001	Numeric encoding (10 bits per 3 digits)
0010	Alphanumeric encoding (11 bits per 2 characters)
0100	Byte encoding (8 bits per character)
1000	Kanji encoding (13 bits per character)
0011	Structured append (used to split a message across multiple QR symbols)
0111	Extended Channel Interpretation (select alternate character set or encoding)
0101	FNC1 in first position (see Code 128 for more information)
1001	FNC1 in second position
0000	End of message

After every indicator that selects an encoding mode is a length field that tells how many characters are encoded in that mode. The number of bits in the length field depends on the encoding and the symbol version.

Number of bits per length field

Encoding	Ver. 1–9	10–26	27–40
Numeric	10	12	14
Alphanumeric	9	11	13
Byte	8	16	16
Kanji	8	10	12

Alphanumeric encoding mode stores a message more compactly than the byte mode can, but cannot store lower-case letters and has only a limited selection of punctuation marks, which are sufficient for most web addresses. Two characters are coded in an 11-bit value by this formula:

$$V = 45 \times C_1 + C_2$$

Alphanumeric character codes

Code	Character	Code	Character	Code	Character	Code	Character	Code	Character
00	0	09	9	18	I	27	R	36	SP
01	1	10	A	19	J	28	S	37	$
02	2	11	B	20	K	29	T	38	%
03	3	12	C	21	L	30	U	39	*
04	4	13	D	22	M	31	V	40	+
05	5	14	E	23	N	32	W	41	–
06	6	15	F	24	O	33	X	42	.
07	7	16	G	25	P	34	Y	43	/
08	8	17	H	26	Q	35	Z	44	:

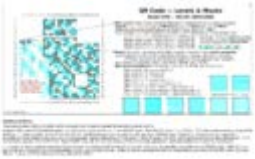
Decoding example

The following images offer more information about the QR code.

- 1 — Introduction

- 2 — Structure

- 3 — Layout & Encoding



- 4 — Levels & Masks

- 5 — Protocols

License

The use of QR codes is free of any license. The QR code is clearly defined and published as an ISO standard.

Denso Wave owns the patent rights on QR codes, but has chosen not to exercise them.[6] In the USA, the granted QR code patent is US 5726435, and in Japan JP 2938338. The European Patent Office granted patent "EPO 0672994". to Denso Wave, which was then validated into French, British, and German patents, all of which are still in force as of November 2011.

The word **QR code** itself is a registered trademark of Denso Wave Incorporated.[36] In UK, the trademark is registered as E921775, the word "QR Code", with a filing date of 03/09/1998.[37] The UK version of the trademark is based on the Kabushiki Kaisha Denso (DENSO CORPORATION) trademark, filed as Trademark 000921775, the word "QR Code", on 03/09/1998 and registered on 6/12/1999 with the European Union OHIM (Office for Harmonization in the Internal Market).[38] The U.S. Trademark for the word "QR Code" is Trademark 2435991 and was filed on 29 September 1998 with an amended registration date of 13 March 2001, assigned to Denso Corporation.[39]

Variants

Example of *Micro QR*

Micro QR code is a smaller version of the QR code standard for applications with less ability to handle large scans. There are different forms of Micro QR codes as well. The highest of these can hold 35 numeric characters.[40]

Risks

Malicious QR codes combined with a permissive reader can put a computer's contents and user's privacy at risk. This practice is known as "attagging", a portmanteau of "attack tagging".[41] They are easily created and can be affixed over legitimate QR codes.[42] On a smartphone, the reader's permissions may allow use of the camera, full Internet access, read/write contact data, GPS, read browser history, read/write local storage, and global system changes.[43][44][45]

Risks include linking to dangerous web sites with browser exploits, enabling the microphone/camera/GPS, and then streaming those feeds to a remote server, analysis of sensitive data (passwords, files, contacts, transactions),[46] and sending email/SMS/IM messages or DDOS packets as part of a botnet, corrupting privacy settings, stealing identity,[47] and even containing malicious logic themselves such as JavaScript [48] or a virus.[49][50] These actions could occur in the background while the user is only seeing the reader opening a seemingly harmless web page.[51] In Russia, a malicious QR code caused phones that scanned it to send premium texts at a fee of US$6 each.[41]

RSS-14, RSS-14 Stacked, RSS-14 Stacked Omni directional and RSS Limited

RSS (Reduced Space Symbology) is a relatively new symbology that encodes up to 14 numeric digits along with an optional two-dimensional "Composite" bar code component that can encode up to 338 additional bytes of alpha/numeric data. The linear portion of a RSS-14 bar code symbol is capable of

encoding 14 digits comprised of 13 numeric digits of user specified data and a single check digit that is automatically calculated for you. (i.e. 14 total digits). If you specify less than 13 digits for the linear portion of a RSS bar code, the number will automatically be left padded with an appropriate number of zeros so that the total number of digits (not counting the check digit) is 13.

The principle use of the RSS symbology is to identify items that cannot be marked with current linear symbols because of size restrictions. The Composite symbols provide additional supply chain data while allowing for the co-existence of symbologies already being used.

For more information on the RSS14 symbology, visit: http://www.uc-council.org
There are currently several variations of the RSS symbology including: RSS-14, RSS-14 Truncated, RSS-14 Stacked, RSS-14 Stacked Omnidirectional, RSS Limited and RSS Expanded.

RSS-14 and RSS-14 Truncated Symbols
For standard RSS-14 bar code symbols, the recommended Narrow Bar Width is 13 mils and the recommended Bar Height is 33 times the Narrow Bar Width (i.e. .43 inches for a standard symbol). The RSS-14 Truncated symbology is identical to the standard RSS-14 symbology except that the height of the bars is reduced to 14 times the Narrow Bar Width (i.e. .18 inches for a standard RSS-14 Truncated symbol). Although VariableText does not have a specific RSS-14 Truncated Symbology option, you can produce RSS-14 Truncated bar codes by selecting RSS-14 for the Symbology property and then setting the Bar Height property to 14 times the value that you choose for the Narrow Bar Width property.

To include a composite message in a RSS-14 bar code, append the composite message to the main numeric message in the Message property with a comma between the two strings. For example, if you set the Message property to: "1234567890123,Testing 123", the RSS-14 bar code that is produced will have "1234567890123" encoded in the main symbol and "Testing 123" encoded in the composite portion of the symbol.

RSS bar codes do not require a "quiet zone" at either end of the bar code symbol however if you enable the Quiet Zone property, a quiet zone ten times the Narrow Bar Width value will be included at both ends of the symbol.

RSS-14 Stacked
RSS-14 Stacked is a variation of the standard RSS-14 symbology that allows the bar code to be printed in two rows. The purpose of RSS-14 Stacked is to reduce the overall length of the RSS-14 linear bar code so it can fit better on certain packaging configurations. This form of RSS-14 can be printed very small and is not generally intended for Omni directional scanning. It is used to encode 14 digits of numerical data and just like the Standard RSS-14 symbology; it can also contain a composite component.

The top row of the linear portion of a RSS-14 Stacked bar code symbol is always 5 times the narrow bar width in height and the bottom row is always 7 times the narrow bar width in height and the two rows are separated by a special separator pattern that is 1 times the narrow bar width in height. With

RSS-14 Stacked bar codes, you cannot set the bar height independently of the Narrow Bar Width value.

RSS-14 Stacked Omni directional

The RSS-14 Stacked Omni directional symbology is almost identical to RSS-14 Stacked except that the two rows of the linear portion of a RSS-14 Stacked Omni directional bar code symbol are always 33 times the narrow bar width in height and the two rows are separated by a special separator pattern that is 3 times the narrow bar width in height. The unlike RSS-14 Stacked, the RSS-14 Stacked Omni directional symbology is designed to facilitate unidirectional scanning.

With RSS-14 Stacked Omni directional bar codes, you cannot set the bar height independently of the Narrow Bar Width value.

RSS Limited

RSS Expanded is yet another smaller variation of the RSS-14 symbology. It is printed in a single row however it is slightly narrower than a standard RSS-14 symbol. The number that you encode in a RSS Limited bar code must be less than or equal to 1999999999999. In other words the first digit in a full 13-digit number cannot be greater than one.

RSS Expanded

RSS Expanded is a variable length linear symbology capable of encoding up to 74 numeric digits or 41 alphanumeric characters in either a single or multiple rows. RSS Expanded is currently not supported by VariableText.

Sample Job

XYZ Company requested you produce 100 lasered tags for their new product line. They want each tag to be numbered sequentially and contain two additional lines of text. The first line is to have the word "Serial:" followed by the serial number, the next line is to say "Mfgr. Date: 4.15.2002" and finally the third line is to say, "XYZ Incorporated". All text should be centered. They want the tags to be 1.5 inches top to bottom and 3.5 inches left to right. Each plate is to have corners with a radius of .25 inches. The serial number consists of 3 fields. The first field will be static text of "AS400-". The second field will be a 3 digit numeric field consisting of numbers 1 thru 100. The third field will be Alpha and will start with the letter "A" and repeat as needed.

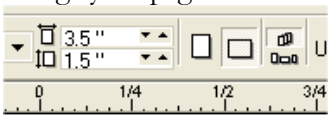
This will be step-by-step instructions on how to use the EngravePro tools. Although each tool is designed to be used separately, a simple job utilizing all tools should give you some insight as to what each tool is capable of and how to use them.

Getting Started

1. Open CorelDRAW.

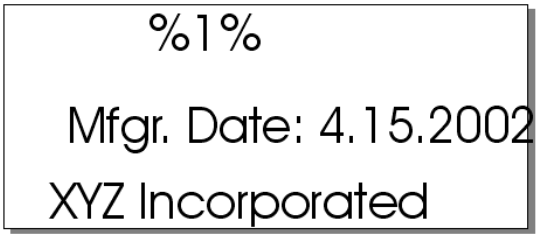
2. Create a new document.

3. Change your page size to 1.5 inches tall by 3.5 inches left to right. You can do this by entering values on your toolbar as in the photo or click the Tools menu, then click Customization. When the dialog appears click Document, Page and finally Size. There you can enter your values as well.



4. Put three lines of Artistic Text on your page by clicking the tool in your tool work bar, then clicking on your page. For the first line of text enter, "%1%" without the quotes. Now click your mouse elsewhere on your page and enter the following text, "Mfgr. Date: 4.15.2002", again without quotes. For the third line of text enter, "XYZ Incorporated".

5. You plate should look similar to the one shown.



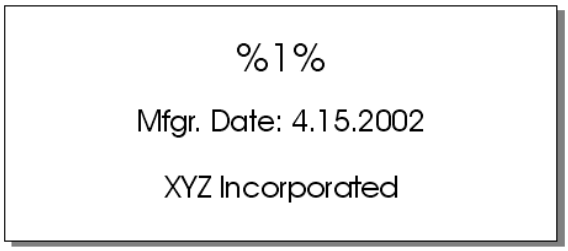
6. Now select the %1% line. Change its height to .250 or about 18 points.

7. Select the second and third line and change their letter height to .188 or about 14 points.

8. Click the EngravePro menu and click SpacePlate.

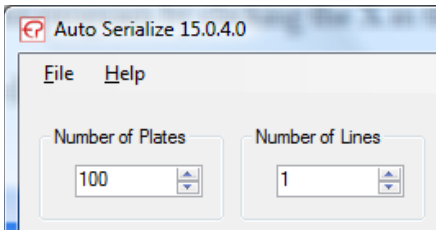
9. Making sure there are no lines selected, click the Even Space Centered Icon  .

10. Your plate should now look like this.



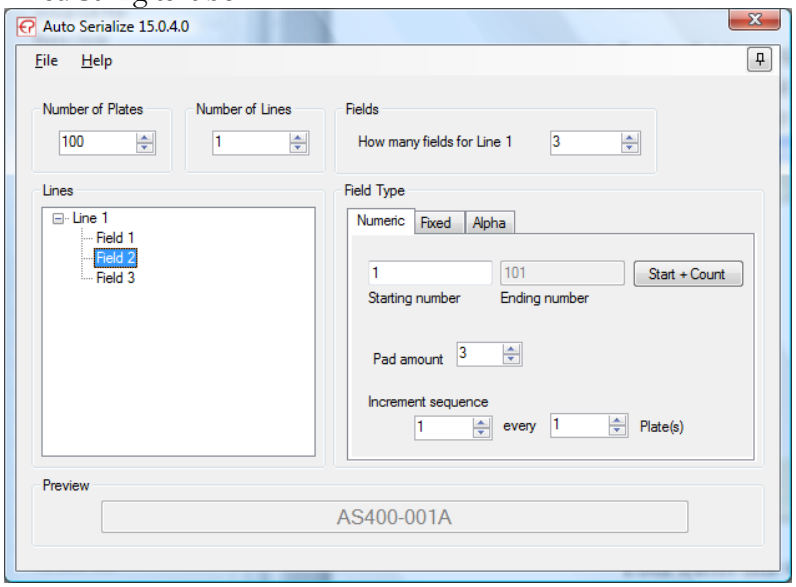
11. Exit the SpacePlate program by clicking the X in the upper right corner of the toolbox.

12. Click the EngravePro menu and select Auto Serialize. Make your settings like the ones shown below.



13. After setting your values like what is shown above, click the Next button.

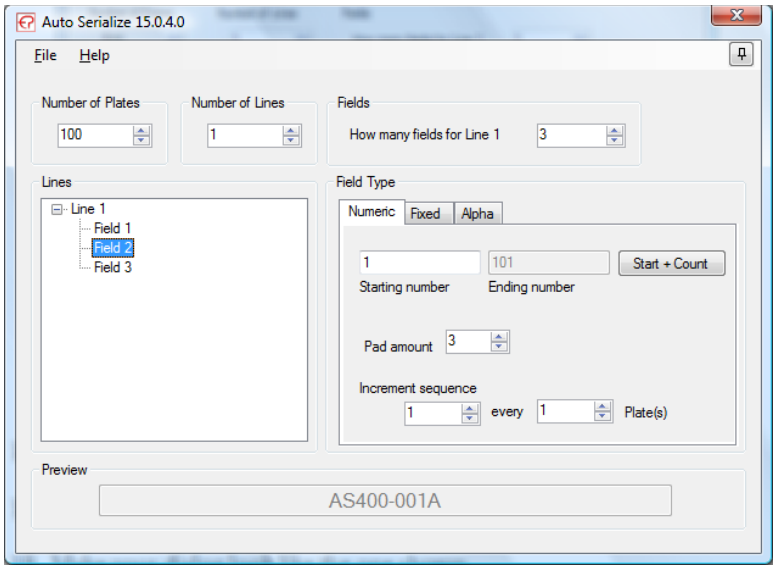
14. In the fields section enter a 3 for how many fields you want. In the left hand Lines window select Field 1 if it in not already selected. Click the Fixed option and enter "AS400-", into the Fixed String text box.



Your status line should look like the one shown.

15. Select Field 2 in the Lines area.

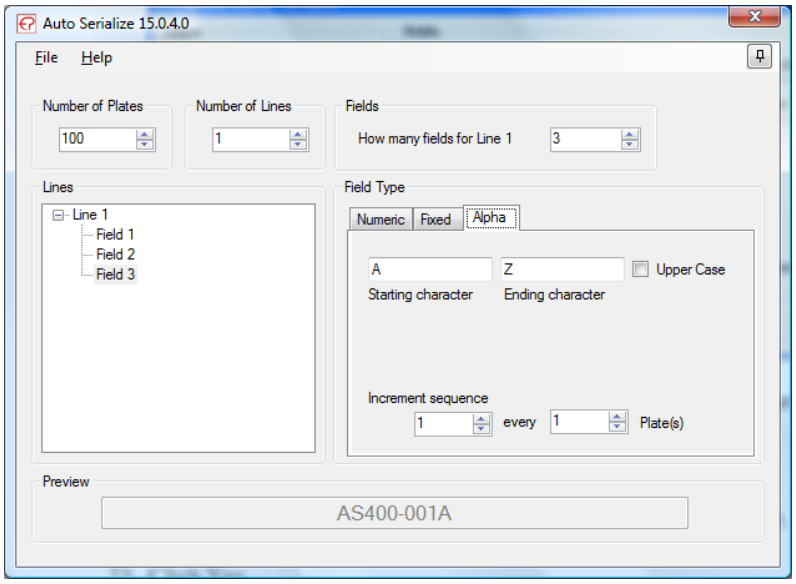
16. Select the Numeric option. Set the pad amount to 3. This will ensure there are always three digits. Your dialog should look like the one below.



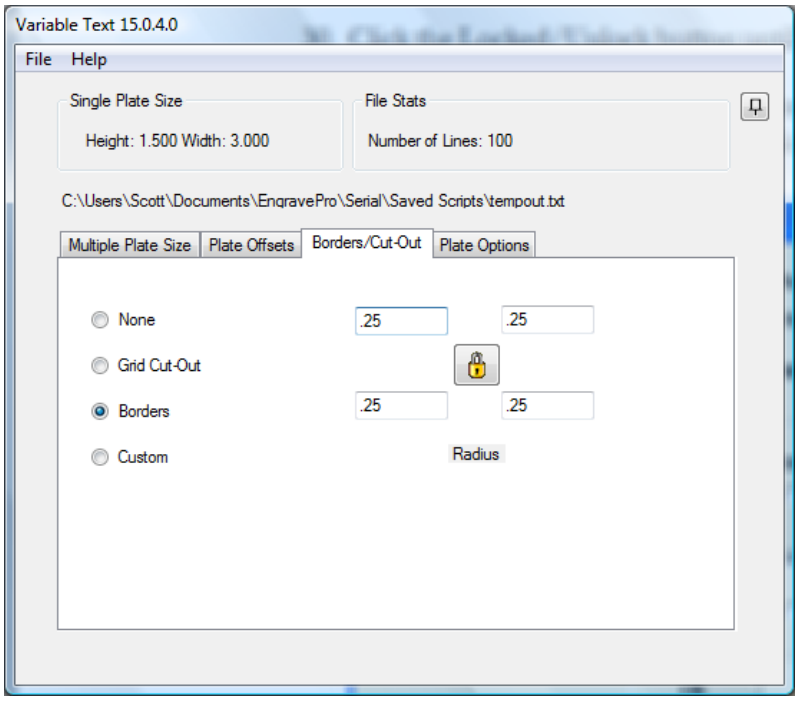
17. Now select Field 3 in the Lines area.

18. Select the Alpha option.

19. Make your dialog look like the one shown.



20. Once you have completed these steps you are ready to continue.
21. Click the Create Job button.
22. A message appears asking you if you want to save the template.
23. Click Yes.
24. Enter a name for you job in the folder you want to save it to.
25. Click the Save button. This will save your template and invoke the Variable Text application.
26. A message appears asking if this is a multiplate job. As you want to interact with the machine as little as possible click Yes. The Plate Size dialog appears.
27. Set your Multiple Plate Size to 12 inches in height by 13 inches in width.
28. For Plate Offsets enter .125 for all four of the values.
29. Select the Borders option in the Borders/Cut-Out area.
30. Click the Locked/Unlock button until it reads Locked
31. Choose any of the four boxes and enter .25 for the Radius. All four boxes will now reflect the value and your Plate Size dialog should look like the one below.



32. Click the Ok button.

33. A message appears asking if you want to set margins. Click Yes.

34. Enter a left and right margin of .250 by clicking the Left Margin, entering your value, then either press the Enter key twice or click on the Right Margin. Finish by entering the Right Margin value and pressing enter again.

35. Click the Ok button.

36. In a few seconds EngravePro will create all of your plates, serialized and ready for you to engrave.

Here is a cutaway view of what you will see in CorelDRAW.

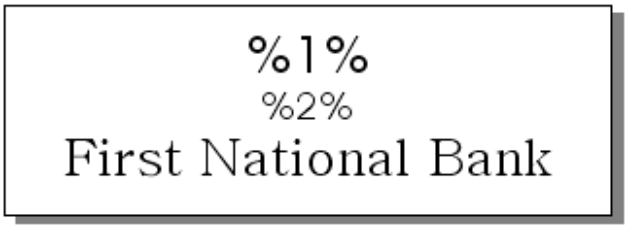
Tips

Variable Text

Always set the justification of your text (i.e. left, right, center) prior to using Variable Text or Auto Serialize.

Always save your "template layout" prior to running Variable Text so that you may use File|Revert if you are not happy with the results. This way you may easily and quickly repeat your Variable Text function after corrections or changes are made.

Open up the sample job in the EngravePro program group located under Start|Programs|EngravePro. This way you will see how to establish variable line locations. Variable lines are simply the places that you wish your different text such as names to go on each plate. We use %1% as the first line locator for the first text insertion. If the layout has more than one line of variable text we use %2% and so on. There is no limit to the number of variable lines. In a simple name badge layout you may have three lines. The first line will be where the name goes. Line two may be where the appropriate title goes. The third line will be the same on every badge. The layout or "Template" would look something like this.



In the Variable Text mode you can open up a saved txt file, type in a list of variables such as names, or names and titles, or copy a list of variables for another location such as e-mail, or any other application where the required text may currently be.

These same basic rules also apply to the Auto Serialize function.

SpacePlate

The SpacePlate screen works with selected objects in their current location or with all objects balanced on the plate with no objects selected. If you wish to spread all lines of text evenly on the plate using either Even (E) or Factored (F) space break the lines of text apart using the function Break Artistic Text under the Arrange menu.

Even Space refers to how all ungrouped objects will have the same vertical space between them. In addition the top and bottom margins will be the same.

Factor Space will place more vertical space around taller lines than shorter lines. In other words the vertical space above and below a line or object is based on factor of its height. In addition you may specify the top and bottom margins you wish to use.

In either mode grouped objects or lines of text that have not been broke apart will maintain their interline spacing. The will be positioned on the plate in relationship to the other objects in the layout.

A picture is worth a thousand words as they say. Create a layout of any type using grouped and ungrouped text or objects to see how the different functions in SpacePlate work.

Appendix

Using the Undo feature of Corel Draw

Note: This pertains to Variable Text Only.

Beginning with EngravePro for CorelDRAW X7 we have added the ability to undo a Variable Text job. The feature is disabled by default upon installation. Before enabling this feature there are a few considerations to be made.

Every time Variable Text imports text and creates a plate it causes a copy and paste operation from within CorelDRAW. There are three to be exact. Each one takes up memory and requires at least an Undo level of 3. So for example if you routinely create a 100 plate job, you will have to have an undo level of 300. For testing purposes I have set my undo level to 1500, which is enough to do pretty much any job I want to do.

The other issue is memory. Often users report sluggish performance or an outright crash after using EngravePro with a large amount of plates or doing multiple jobs. Enabling the Undo feature of EngravePro will exasperate the issue.

With that said if you want to enable the undo feature simply go to preference from Variable Text and change it to True.

To change your undo levels in CorelDraw go to the Tools Menu, Options, Work Space and then General.