

# Copy Fields Contents Pro

© 2005 The New Hampton Group

## Installation and Usage

**Copy Fields Contents Pro** provides an easy way to duplicate the contents of any set of ACT! fields into another set of fields on and between the ACT! current contact, current group and/or current company record(s). **Copy Fields Contents Pro** works with Contact fields, Group fields and Company fields. Any number of field pairs can be copied within and between these three records, with a single execution of **Copy Fields Contents**.

**Copy Fields Contents Pro** can be used either as a trigger or as a custom command. It will operate successfully only if ACT! is running, there is an ACT! database open, and the user is in edit mode (normal operation mode, ie, not layout designer or lookup mode) in either the Contact Detail View, the Group Detail View or the Company Detail View (ie, not Contact List View or Task List View, etc.)

**Copy Fields Contents Pro** uses a “command” or “argument” string which specifies, in field pairs, the source and destination field names. Any number of pairs can be specified in the command, so that multiple fields can be copied at one time. And, since the command value can be set to different values in different trigger definitions and/or custom command definitions, you can copy different sets of fields with a single copy of **Copy Fields Contents Pro**.

## Installation

1. Shut down all running programs
2. Run **Install\_CopyFieldsContentsPro.exe**
3. Follow the instructions on the screen. Restart your computer if prompted to do so.

This will create (if necessary) the \TNHG subfolder within the folder that Windows “knows” as the “Program Files” folder – usually c:\Program Files. It will create the \TNHG\CopyFieldsContentsPro subfolder, and install **CopyFieldsContentsPro.exe**, and other supporting files, into that folder. The installation process will also copy into the \ACT\Act for Win 7\Plugins subfolder and into the \ACT\Act for Windows\Plugins subfolder (within that same “Program Files” folder) the supporting files necessary to operate within ACT!2005 and ACT!2006.

## Usage

The command to execute **Copy Fields ContentsPro** consists of the entire path-and-filename of the Copy Fields Contents Pro program, followed by the “command string”. Please note that if there are any “blanks” in either the path-and-filename, or in the command-string, then it must be enclosed in “quotes” – Windows treats the blank as a “terminator”, and the quotes are necessary to tell Windows that the string continues past the blank.

## The Program Name

If the path-and-filename contains any blanks (such as `c:\program files\tnhg...`), then that entire path-and-filename must be enclosed in quotes, such as:

```
"C:\Program Files\TNHG\COPYFIELDSCONTENTSPro\COPYFIELDSCONTENTSPro.exe"
```

## The Command String

The “command string” consists of entries that start with a “question mark”, followed by a single letter, and “equal sign” and then more information. For example:

```
?V=T
```

```
?L=<C:User1>=<G:Description>
```

```
?L=<A:Address 1> = <C:Billing Address>
```

Each “Question mark-Letter-Equals” string is used to tell Copy Fields Contents Pro a specific detail of its operation:

**?V=** indicates whether or not you want to view the Copy Fields Contents Pro window while it is performing its copy operation.

- ?V=T indicates that you DO want to see the window, and that you will need to click the [OK] button on that window when the copying is complete.
- ?V=F indicates that you do NOT want to see the window. This is the default value, and if you don’t want to see the window, you can leave out this entire ?V= “argument”

**?L=** indicates a list of fields to be copied, in the form of <destination field> = <source field>. You can have as many pairs of fields as you need, each pair separated by a comma. So, for example, if you want to copy User 2 into User 4 and User 5 into User 6, you would use

?L=<C:User 4>=<C:User 2>, <C:User 6>=<C:User 5>

Some points to remember when constructing your list:

- Each fieldname is enclosed in “carets” < > similar to what you’ll see in mail-merge templates.
- Each fieldname includes as the first letter an indication of which record “type” that field appears on. A indicates the Company record, C indicates the Contact record, and G indicates the Group record. The record type is separated from the fieldname by a colon, ie <A:Discount> refers to the Discount field on the Company record.
- The DESTINATION field goes first, followed by the equal-sign, followed by the SOURCE field. Think “this field COMES FROM that field”.
- Field PAIRS are separated by a comma, with or without spaces.
- The list of pairs can be very long – no limit to the number of pairs.
- If you are using fields from the Contact record, a great way to develop your list of field pairs is to use the Write menu > New Email/Letter Template option. Pick your first destination field and insert it into the blank document. Type “=” (without the quotes). Select your first source field and insert it into the document. Type “,” (comma space without the quotes). Pick your second destination field and insert it into the document. Keep repeating this process until all your field pairs are entered. Now use the Edit menu > Select all option then Edit > Copy to copy your list to the clipboard. Paste it after the ?L= in your trigger definition, or paste it into a blank Notepad document and save it for use with the ?F= argument described below.

**?F=** indicates that the List is to be read from a file instead of being included in the trigger command itself. This can be very handy if your list is really long, or if it changes frequently, or if you’d rather use a text editor to establish or maintain your list. When you use the ?F= argument, the entire-path-and-filename follows the equal-sign. So, for example, you might use

?F=S:\ACT\Common\Triggers\CFCPUserFields.txt

if your list of fields is in the CFCPUserFields.txt file, in the subfolder \ACT\Common\Triggers in the shared folder S:. The contents of this file should contain ONLY the list of field pairs; it should NOT include ?L= or ?F=.

## The Batch File

The version of ACT! in commercial release at the time of this writing does not allow a trigger to include a “command string” or “argument list”, like the one that tells Copy Fields Contents Pro how to operate. Therefore, the trigger definition must be placed in a “DOS .bat file”, and that file used as the trigger definition on the ACT! field. When the .bat file “runs”, a DOS window will appear on the screen, showing the commands included in the .bat file and waiting for the commands to finish. This can be somewhat disconcerting to users, so you might want to include a command to hide this window. (A utility is provided in the \TNHG folder, in case you need one. Its name is `ConsoleTool.exe`, and it uses the `/hide` “switch” to tell DOS to hide the window. An example of its use is included below.) On the other hand, if you are copying a large number of fields, or copying fields between records, it may take ACT! a moment to complete the requested operation. In this case, the presence of the DOS window serves as a reminder to wait for the process to finish. (The sample .bat file is included in this installation illustrates how to make this reminder more effective.)

As an example, you might use Notepad (or any other text editor) to create a file that contains the following lines:

```
"C:\program files\TNHG\ConsoleTool.exe" /hide
"C:\program files\TNHG\CopyFieldsContentsPro\copyfieldscontentspro.exe" _
"?L=<C:User 6>=<G:Description>, <C:User 4>=<A:Territory> ?V=T"
```

(There are really only two lines in this example but the page isn't wide enough to show the entire second line on a line by itself. The underscore `_` is not really part of the line – it is there only to indicate that the REAL line continues on the next “text line”.)

This example uses the `ConsoleTool` to hide the DOS window (`/hide`), then runs the Copy Fields Contents Pro program (`copyfieldscontentspro.exe`), shows the window while it operates (`?V=T`), and copies the contents of the Group “Description” field into the Contact “User6” field (`<C:User 6>=<G:Description>`) and the contents of the Company “Territory” field into the Contact “User 4” (`<C:User 4>=<A:Territory>`).

If you save this file as

`CFCPUser6and4.bat`

in the `s:\act\common` folder, then your trigger would be

`s:\act\common\CFCPUser6and4.bat`

Another quick example:

If the `f:\triggers\CFCPCompanyAddress.bat` file contains

```
"C:\program files\tnhg\consoletool.exe" /hide
"C:\program files\tnhg\copyfieldscontentspro\copyfieldscontentspro.exe" _
"?L=<C:Billing City>=<A:City>, <C:Billing State>=<A:State>,<C:Billing Zip>= _
<A:Zip>, <C:Billing Address 1>=<A:Address 1>"
```

And the trigger is set to

```
F:\triggers\CFCPCCompanyAddress.bat
```

then the Contact record's Billing Address fields will be copied from the "main" address fields on the current Company record when the trigger executes, and the user will not have to look at either the DOS window or the Copy Fields Contents Pro window while the fields are copied.