

# Power Headers



*CS2/CS3/CS4*  
**InDesign Plug-in**

version 2.0

last revision: November 11, 2009

# *Power Headers*

## **Contents**

**4** “Why a Plug-in?”

**9** Examples of Use

**11** Installation

Enabling and Disabling Plug-ins.

I installed Power Headers, but I can't find its menu!

**14** Purchasing Licenses and Activation

**18** Basic Use

Conceptual Introduction:

**18** Setup

Preparing Your Document

Creating Your Variable Definitions

Basic Setup

Additional Settings:

Alternate Styles

Resetter Styles (CS3 or later)

GREP Processors (CS3 or later)

Inserting Header Variables

## **38 Updating Headers**

- Automatic Updates
- Header Frame Overrides
- Stacking Order and Groups
- Global GREP Processors
- Book Files
- Loading Header Variables

## **42 Scripting**

- Event Codes
- Sample Script
- Script Labels

## **47 Cookbook**

## **47 Addendum**

- What is “APID ToolAssistant”?

## **49 Acknowledgements**

## **50 License Agreement**

# *Power Headers*

## **“Why a Plug-in?”**

The header variables included in InDesign since the release of CS3 are very good. They work very well in the context for which they were intended. However, for many complex layout situations, they fall short. *Power Headers* addresses those shortcomings. The following is an overview of the improved functionality available with *Power Headers*.

1. InDesign variables are limited to basing your headers on either paragraph **or** character styles. In *Power Headers* there is an option to base it on paragraph style, character style, or *both* (a character style within a specific paragraph style). See [page 23](#) for more information.

2. In the “use” drop-down, InDesign has two options: “First” and “Last”. *Power Headers* has 8 options: **1.** First Word **2.** First Instance **3.** Last Word **4.** Last Instance **5.** First And Last: This option will put both the first and last instances into the header if there are more than one instance of the same variable text. This enables uses such as “Chapter 1” or “Chapter 2-3” depending on the content of the page. (There is a field for “between text” for such situations.) **6.** Current Instance: This option enables the input of the last found instance of source text unless the source text is the first line of text on the page. When an instance is found on the first line, *or if an instance is anchored to the first line*, that instance will be appear in the header. Otherwise, the *previous* instance will appear in the header. **7.** Current and Last Instance: This option combines the functionality of “Current Instance” with “First and Last”. **8.** All Instances: This option will populate the header with all instances of the selected style. See [page 24](#) for more information.

3. Conditional Text: *Power Headers* has an option to define conditional text. Conditional text replaces the text in the “Text Before” or “Text After”

fields of the configuration dialog. If the correct option is selected, the conditional text will appear only on pages which do not have source text, and if “Current Instance” is selected, the conditional text will appear on a page on which the *last instance* was used. See [page 28](#) for more information.

4. InDesign does not allow for choosing the frame order for the source text. *Power Headers* has an option to choose the frame order. The order can be either right-to-left, or left-to-right. These options are especially important in the ME version when using Right-to-Left text. The frame order can also be defined to either go down first and then across or vice versa, resulting in four frame order options: 1. “Top, Bottom, Left, Right”. 2. “Top, Bottom, Right, Left”. 3. “Left, Right, Top, Bottom”. 4. “Right, Left, Top, Bottom”. This is defined at the variable level, so you can have different frame orders in the same document. See [page 25](#) for more information.

5. InDesign headers are page reflective, meaning they always reflect the content based on which page they appear. However they cannot reflect the content on the whole spread or a single column. *Power Headers* allows you to select a “Source Range”. A source range can be defined to be a “Page”, “Spread”, or “Column”. This enables creating “spread reflective” and “column reflective” headers. See [page 23](#) for more information.

6. InDesign offers no control over which pages display header text. This requires creating a separate master page for pages which shouldn’t have the header text. *Power Headers* solves this problem by providing three inclusion options:

a) “Only Include on Pages With Source Style” will only insert a header on a page which actually has text for the header.

# *Power Headers*

b) “Include Last Instance on Pages Without Source Text” will insert a header, if there’s text of a story which has text for the header.

c) “Include Last Instance on Pages Without Story Text” will create headers from the last previous occurrence in the document. See [page 31](#) for more information.

7. *Power Headers* has an option to ignore stories that have fewer than a specified number of characters. See [page 32](#) for more information.

8. InDesign variables are document-specific, and need to be created or loaded in each document separately. *Power Headers* can create header variables across a whole book file.

9. *Power Headers* can be updated across a whole book file. See [page 41](#) for more information.

10. Alternate Styles: InDesign headers can only be based on one style. If you want the header to be taken from style 1, *or* style 2, you are stuck! *Power Headers* can define variables based on an unlimited number of styles! If you have variations on your header styles, and want the headers to reflect any of those styles, you can define “Alternate Styles” for your header definitions. All the defined alternates will be treated as if they are of the same style. See [page 32](#) for more information.

11. Smart Title Case: The standard title case of InDesign is very “dumb”. All words are capitalized no matter what they are. In the real world, this is fairly useless. *Power Headers* can create a Smart Title Case which is rule-based. The underlying word list is user-definable. See [page 26](#) for more information.

12. Localized Formatting: InDesign does not provide a way to transfer local formatting to the header. The styling must be done using character styling. It is impossible to create a header with only one of several words bolded or italicized. *Power Headers* gives an option to copy local (overridden) formatting to the header. It can even convert italics to the header base weight. See [page 29](#) for more information.

13. Text Composition: InDesign variables (including headers) are treated as a single character. Because of this, headers cannot break across lines. This makes multiple-line headers impossible. *Power Headers* keep the headers as live text, and creates multiple-line header with ease.

14. Punctuation: InDesign gives no control over removing punctuation. It is either enabled, or disabled. *Power Headers* gives a number of options. See [page 27](#) for more information.

15. Ignore Blank Paragraphs: *Power Headers* offers control over whether blank paragraphs are transferred to the headers or not. While you would usually not want to create headers from blank paragraphs, this option can be used to create blank headers if needed. See [page 30](#) for more information.

16. Paragraph Ranges: *Power Headers* offers the option of transferring an entire paragraph range to the header. This can be used for some very interesting applications. For example, a side bar can contain page reflective information. See [page 31](#) for more information.

17. Bullets and Paragraph Numbering: *Power Headers* offers the option of transferring bullets or paragraph numbering to the headers. You can use this option in conjunction with the option to transfer local formatting or GREP Processors ([feature #20 on page 8](#)) to style the bullets or paragraph numbers differently than the rest of the header text. See [page 29](#) for more information.

# *Power Headers*

18. Original Text Removal: *Power Headers* offers the option for the header to *not* contain the found text. This is useful in cases where you need only the paragraph numbers or custom text. It can also be used to remove specific headers on specific pages. See [page 30](#) for more information.

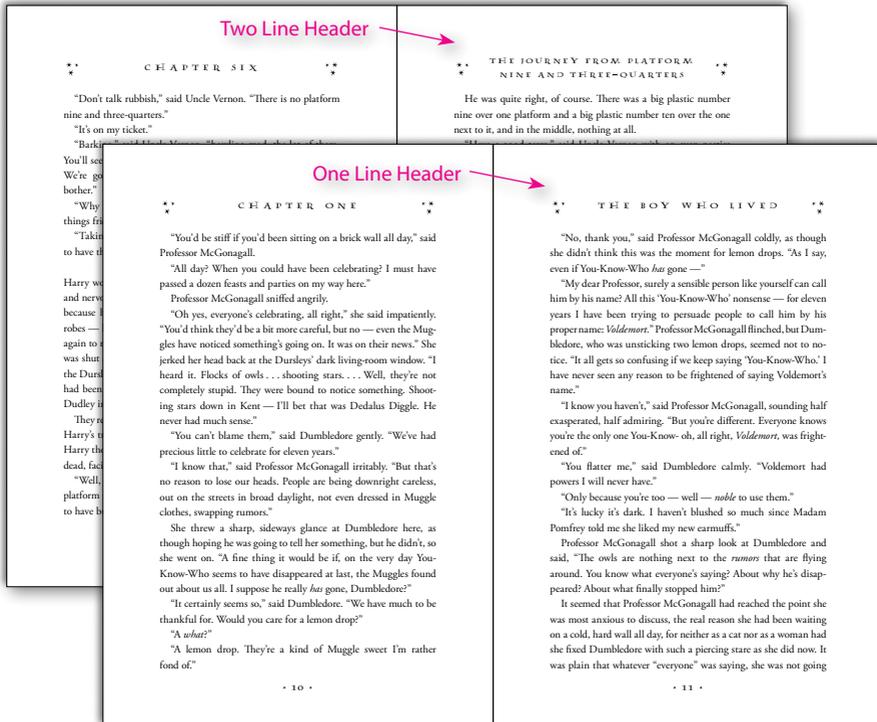
19. Resetter Styles: *Power Headers* offers the option to limit the instances of the source text which appears in the headers. It does so by removing potential header text when it finds a “Resetter Style”. You can define multiple Resetter Styles, and each one can have unique settings. See [page 33](#) for more information.

20. GREP Processors: *Power Headers* offers the option of applying post processing to individual header variables. The processing can change the formatting and can change the content as well. This can be used to create headers with abbreviated text or even entirely different text than what appears in the document! See [page 35](#) for more information.

21. Global GREP Processors: *Power Headers* offers the option of creating Global GREP Processors, which are applied to the entire header frame, as opposed to only the individual header variables. Global GREP Processors not only make it easier to set up GREP Processors for an entire document; they also enable the creation of compound headers, where different variables can interact in very interesting ways. Please refer to [page 41](#) for more information.

## Examples of Use

To illustrate some of the unique capabilities of *Power Headers* we are including some screen shots of actual production files which were created using *Power Headers*. The illustrations are courtesy of Brad Walrod.



Shown above is a screen shot from the 10th-anniversary edition of Harry Potter and the Sorcerer's Stone. Some chapters feature two line headers. *Power Headers* makes it possible to create these automatically, with only one header definition for all the chapters.

# Power Headers

The illustration below shows the combined capabilities of the Smart Title Case and the ability to automatically copy local formatting to the header text. Note especially, the different capitalization on the two uses of the word “the” in the Smart Title Case at the bottom of the page.



**FIGURE 10.5** ■ *Homo rudolfensis*  
This fossil hominid is believed by some researchers to be a separate species from *Homo habilis*, owing to its larger size.

general body plan and overall morphology (bigger brains, smaller faces), here the two species are discussed as *Homo habilis*.

As Leakey and his associates recognized, *H. habilis* differs in its anatomy from the robust australopithecines dating to about the same time in East Africa and South Africa. *Australopithecus boisei* had an enormous chewing complex—its back teeth, jaws, and face were very large—but it had a small brain. In sharp contrast, *H. habilis* had a smaller chewing complex and a larger brain. Combined, the reduced chewing complex and increased brain size gave *H. habilis*'s skull a more rounded, or globular, appearance. Most anthropologists agree that these attributes indicate that *H. habilis* began the lineage leading to modern humans.

Still unconfirmed is the identity of *H. habilis*'s immediate ancestor. The anthropologist Tim White's morphological comparisons between *H. habilis* and the earlier australopithecines suggests that the ancestor was *Australopithecus garhi*, because its face, jaws, and teeth are most similar to *H. habilis*'s. White suggests that the evolutionary transition took place sometime around 3.0–2.5 mya.



Italicized Words

All Caps Source Text

**HOMO HABILIS: THE FIRST SPECIES  
OF THE GENUS HOMO**

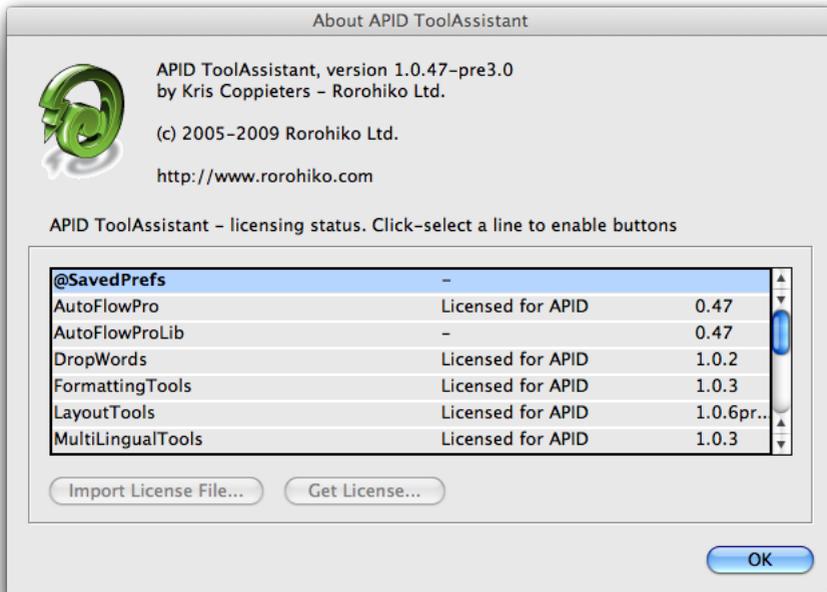
For many years, *Homo habilis* was known from just skulls and teeth. Anthropologists had no idea what the rest of the skeleton looked like. Excavations at Olduvai Gorge in the 1980s by Donald Johanson and his associates led to the discovery of a very fragmentary, but important, skeleton of *H. habilis*, known as “OH 62.”

*Homo Habilis: The First Species of the Genus Homo* 307

Italics Formatting Retained at  
Destination Font Weight

Text Converted to Smart Title Case

# Installation

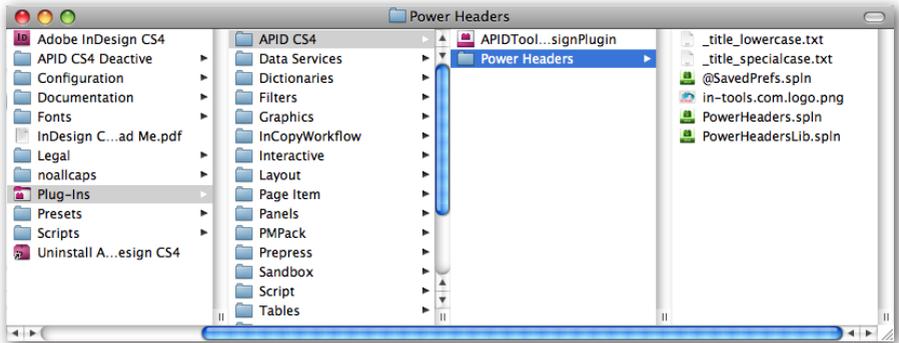


*Power Headers* was developed using a product called “APID ToolAssistant” created by Rorohiko. This necessary central controller does a lot of the low-level processing. *If you are interested in the technology behind the development, you can read the addendum on [page 47](#).*

There are four different files which must be installed for *Power Headers* to function properly: **1.** APIDToolAssistantCSx.xxx (the exact name depends on your version) **2.** @SavedPrefs.spln **3.** PowerHeaders.spln **4.** PowerHeadersLib.spln. Additionally, for the customization of the Smart Title Case you should place the files “\_title\_lowercase.txt” and “\_title\_specialcase.txt” in your plug-ins folder as well.

# Power Headers

All of these files need to be copied to InDesign's plug-ins folder. To ensure that everything is properly installed, we suggest that the *Power Headers* folder included in the download be copied in its entirety to your plug-ins folder. The APIDToolAssistant file must also be installed in the plug-ins folder. It does not need to be in the same folder as *Power Headers*. You can organize your folders as you desire.



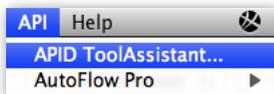
Shown above is a screen shot of how your folder structure should look after installation. We have created a folder named APID inside the Plug-ins folder to help keep things organized.

*Please note:* APID ToolAssistant replaces an older version of the plug-in. In the past, there were two versions of the API plug-in distributed by Rorohiko. Active Page Items Developer (APID), was the version developed to work with externally developed plug-ins. The free Active Page Items Runtime (APIR), was for plug-ins developed internally at Rorohiko. If you have either of these two plug-ins installed, you must remove them and install the APID ToolAssistant instead. The older APIR or APID plug-ins will not work. You must have the most current version of the APID ToolAssistant installed (version 1.0.47 or higher).

Shown above on [page 11](#) is the “About” window of API. This window can be brought up in two ways: 1. Select: InDesign → About Plug-Ins → Rorohiko Ltd. → APID ToolAssistant...



2. Select: API → APID ToolAssistant...



In-Tools plug-ins do not show up separately in the “About Plug-Ins” menu. Information on installed In-Tools plug-ins are only displayed in the “About” window of API. If the plug-ins are properly installed you should see the displayed window (or the equivalent Windows version).

Please note the API version number near the top of the window. Your version number must say 1.0.47 or higher. Additionally it must say “APID ToolAssistant”. If it says “Active Page Item Developer” or “Active Page Item Runtime”, you have the wrong version installed. As mentioned above, both of these versions have been replaced with APID ToolAssistant.

When in doubt: search your plug-in folder for a file whose name contains “ActivePageItem”, and replace it with the new one.

In the “About APID ToolAssistant” window, there is a list of all installed plug-ins and their status. The first column lists the plug-ins. The second column shows the license status. If the plug-in is licensed, “Licensed For APID” will be displayed. If the plug-in is in demo mode, the word “Demo” will be displayed along with the number of days remaining until the plug-in expires.

# *Power Headers*

The third column shows the plug-in's version number. When trying to determine that you have the most current version of a plug-in, please refer to the version number listed in this column.

## **Enabling and Disabling Plug-ins.**

To enable or disable any individual plug-in, simply move the file into or out of the folder which contains the APID ToolAssistant plug-in. One simple solution would be to create two folders within your plug-in folder. Name one "API", and the other "API disabled". You can then easily move files back and forth.

The APID ToolAssistant is like any other plug-in, and InDesign must be restarted when it is installed or removed. However, the rest of the plug-ins do not require a restart of InDesign when they are installed or removed. Newly installed plug-ins are available in documents when they are opened. If there are open documents when you install any of the \*.spln files, those documents should be closed and reopened.

## **I installed Power Headers, but I can't find its menu!**

In-Tools plug-ins are unique, in that menu items will not appear until a document is opened. The plug-ins will appear in the plug-in list in the "About APID ToolAssistant" window, but menu items only appear after a document is opened or created.

## **Purchasing Licenses and Activation**

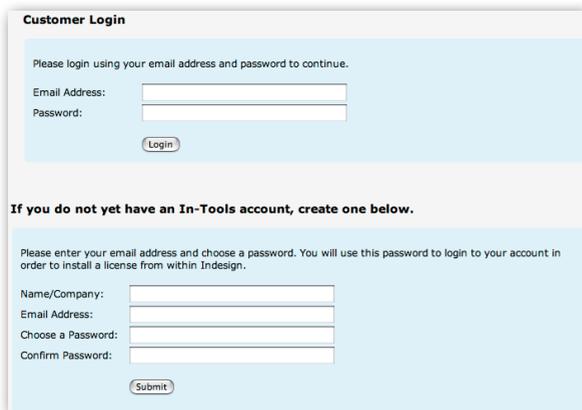
There are two ways to purchase In-Tools plug-ins: They can be purchased from the [In-Tools web site](#) prior to activation, or they can be purchased directly from within InDesign. If the plug-ins are purchased from the web site,

## Purchasing Licenses and Activation

you will establish a login and password for activation at the time of purchase. If you purchase the plug-ins from within InDesign, the payment and activation both occur simultaneously. If you plan to purchase a plug-in package, or would like to purchase a number of plug-ins in one transaction, you should purchase them from the [In-Tools web site](#) before you initiate the activation process.

No matter how you purchase the plug-ins, the process from within InDesign is the same:

1. Select the plug-in you would like to activate.
2. Click “Get License”. This will take you to the log-in page of the In-Tools web site (shown on the following page).



The screenshot shows a web form titled "Customer Login". It is divided into two main sections. The top section is for existing users, with a light blue background and the text "Please login using your email address and password to continue." It contains two input fields: "Email Address:" and "Password:", followed by a "Login" button. The bottom section is for new users, with a white background and the text "If you do not yet have an In-Tools account, create one below." It contains the text "Please enter your email address and choose a password. You will use this password to login to your account in order to install a license from within InDesign." followed by four input fields: "Name/Company:", "Email Address:", "Choose a Password:", and "Confirm Password:", followed by a "Submit" button.

If you already have an In-Tools account, fill in your login and password. Otherwise create an account now. If you already paid for your plug-ins, make sure you fill in the login information that you provided at the time of purchase. You *must* provide a valid e-mail address or you might not receive your license file.

# *Power Headers*

3. If you have not yet paid for the plug-in you will be taken to a PayPal payment page. Upon successful completion of the payment, you will be returned to the log-in page. To ensure that you are credited with your license, please don't close your browser window until PayPal returns you to our web site

4. After successfully completing the log-in procedure, verify that you are on your login page and you should see a link to download your license file.

If you do not download the file initially, you can do so at any time. On your account page there are two tables. The top table lists licenses which were already activated. The bottom table lists licenses which were paid for but not yet activated. To the right of each license on the top table is a link which says "Show License". After clicking on it, it will change to "Download License". You can then simply download the license file and save it where you like on your hard drive.

*Please note:* If you are using Safari, you will need to right-click (or control-click) on the download link to save it. Otherwise, Safari will open a window with gibberish.

The license file is specific to the installation from within which you have activated the license. It will not work on a different installation, even with the same serial number. If you need to use In-Tools plug-ins with more than one installation of InDesign, you will need to purchase an additional license for each installation.

5. Once you have your license file, go back to the "About" window in InDesign, select the plug-in you are activating and click on "Import License File". Locate the file on your hard drive and click "Open". The status next to your plug-in will change to "Licensed".

## Purchasing Licenses and Activation

*Please note:* Under certain circumstances while trying out our plug-ins, you might see the APID ToolAssistant listed as unlicensed. There is no need to purchase a license for the APID ToolAssistant if you plan on purchasing any In-Tools plug-ins. The licensing of APID ToolAssistant is handled completely transparently when you purchase an In-Tools plug-in. An In-Tools license will properly license APID ToolAssistant as well. The end result will be a licensed copy of APID ToolAssistant, but you will not see (or need) any license file for it.

For further questions and issues regarding licensing and activation, please refer to the [FAQ web page](#); or for specific problems, please contact [technical support](#).

# Power Headers

## Basic Use

### Conceptual Introduction:

*Power Headers* works on a principle of “building blocks”. Each variable definition can be one element in a string of header text. Header text can be composed of any number of header variables, as well as static header text.

For example: Your header text can be composed of [Author] [Book Title] [Chapter] [Chapter Number] [Chapter Name] [Page number]. These can all be in the same text frame, and you can mix and match InDesign standard variables, *Power Headers* variables and static text all in the same text frame.

## Setup

### Preparing Your Document

Before you create your header variables, you should follow these steps:

1. Create a text frame on your document master page. Your header text will be placed in this frame.

2. Select the frame and select Layout → Power Headers → Label Header Frame..., or right click:



3.  Type in (or select) a label for you header frame. The name of the label is unimportant, but it should be something which you will be able to identify later.

Once you label your header frame, a little label box will appear on the bottom right corner of the frame.  You can turn off the visibility of this label by selecting: View → In-Tools → Hide Header Frame Labels.

4. Create a character style with the correct styling (if any) for your variable header text. It is advisable to name it something which includes “header” or something similar.

Now you are ready to create your header variable.

#### Points to keep in mind when setting up your headers:

- ▶ **Don't Confuse the concepts of *Header Frame Labels* and *Header Variables*. Simply labeling a header frame *will not* define a header. Labeling a header frame will only allow the selected frame to contain the header which you will define in the next step.**
- ▶ If you have more than one header frame on the page (each with a different label), and don't have any special styling for the header text, (which can't be set on the paragraph level), one character style can be used for all the headers **provided that there is only one variable per frame!**
- ▶ You ***do not*** need to define more than one header label ***even if*** you are defining more than one header variable. The only reason why you might want to create more than one header label, would be to prevent header frames from being overridden from master pages unnecessarily, or to use the same character style in multiple variables. In fact, to avoid confusion, we recommend *not* defining more than one *Header Label* unless absolutely necessary.

# Power Headers

## Creating Your Variable Definitions

There are many options when defining header variables, but the defaults should be correct for most uses. Assuming the defaults are correct, there are only a few things which need to be selected. The functionality of *Power Headers* is the same in CS2, CS3 and CS4, but the user interface is enhanced for increased usability in the CS3 and later versions. Below are screen shots of the user interface for both CS2 and CS3/ CS4.

### CS2

Define Header Variable

Variable Name: Header 1

\*Variable\* Style: Header

Header Frame Label: Header Frame

Source Char. style: Any  Copy Local Formatting

Source Para. style: Section Header  Italics at Base Weight

Source Range: Page

Type: First Instance

Frame Order: Top,Bottom,Left,Right

Change Case: No Change

Punctuation: Remove Trailing White Space

Text Before:

Text After:

Text Between/Conditional:

Conditional Text:  After  
 Before

Options:

Only Include on Pages with source style

Include Last Instance on Pages without source style

Include Last Instance on Pages without story text

Include Conditional Text:  When No Source is on Page  
 When Next Source is on Page

More Options: Ignore Stories Less Than 100 Characters

## CS3/ CS4

Define Header Variable

Variable Definition

Variable Name:

Variable Style:

Header Frame Label:

Source Styles

Source Char. Style:

Source Para. Style:   Don't Transfer Text

Settings:

Source Range:

Type:

Frame Order:

Change Case:

Punctuation:

Text Before:

Text After:

Conditional Text:

Before  After

Copy Local Formatting  Italics at Base Weight  Copy Paragraph Numbering

Ignore Blank Paragraphs  Limit Instances to One Paragraph

Options:

Only Include on Pages with source style

Include Last Instance on Pages without source style

Include Last Instance on Pages without story text

Include Conditional Text

When No Source is on Page

When Next Source is on Page

More Options:

Ignore Stories Less Than  Characters

As you can see, the layout of the dialog is somewhat different between the two versions, but most of the options are basically the same. There are some options which are specific to CS3 and later. These will be discussed later in the manual.

# Power Headers

## Basic Setup

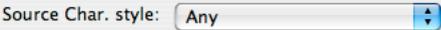
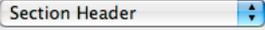
There are four basic things which must be chosen while setting up the variable.

1.  Variable Name: Header 1 Name your variable: Type into this field the name of your variable. What you name it is unimportant. It should be whatever name is easiest for you to identify.
2.  Variable Style: Header Pick your variable style: In the drop down list shown, select the character style which you set up in step #4 in the previous section, “Preparing Your Document”.
3.  Header Frame Label: Header ... Select your header frame label: The drop-down list contains all labeled frames on your document master pages. Select the label you used in step #3 in the previous section.

*Note: If you labeled a frame on a regular page (i.e. not a master page), that label will not show up in the list. You **can** use frames on regular pages, but you must label a frame on a master page somewhere with the same label. The master page does not need to be used.*

- ▣▣▣▣➔ Header frame labels are not required. The headers can be set up without them. It is however, advisable to use them. There are four reasons for using the labels: **1:** The updating of headers is quicker when labels are used. **2:** If labels are not used, it is necessary to override all master frames for the headers to properly update. This makes a mess of your pages. **3:** The use of labels allows for semi-automatic removal of overrides on header frames. **4:** The labels enable the use of just one character style for any number of header frames.

## Setup / Additional Settings:

4.  Pick your source styles: The  source styles are the applied styles in the text which will be picked up to be copied to your header (i.e. the text which will “fill” your variable text). You must select a style. It can be either a paragraph style, a character style, or both. If you select both a paragraph style and a character style, then text will be applied to your header, only if you have the selected character style *within* the selected paragraph style. If the character style is applied to text in a different paragraph, the text will be ignored. *Please note: Characters are only found if the character style is applied directly. Styles applied through nested styles will not be found.*

Congratulations! You have succeeded in setting up a basic header. If you have properly followed all the steps, and inserted a header variable in some text ([page 37](#)), when you select Layout --> Power Headers --> Update Document Headers, the headers will be updated to reflect the page text.

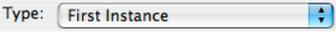
### Additional Settings:

While the default settings for headers might fit your needs, it is very probable that they won't. The options for updating the headers are very powerful and very easy to set up, but it does require some understanding of what each option does. Below is a list of all the selectable options, and instructions on how to use them.

1.  Source Range: This option allows definition of “Page Headers”, “Spread Headers”, or “Column Headers”. *Page Headers* will find instances on the same page only, *Spread Headers* will find instances on the whole spread, and *Column Headers* will find instances in the same column as the header. The columns are defined by column guides; any part of a text frame which falls inside the vertical bounds

# Power Headers

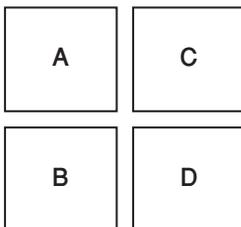
defined by the column guides will be defined as belonging to that column. If a *text* frame spans more than one column, all *header* frames in those columns will reflect that text. If a *header* frame spans more than one column, the header text will reflect the *last* column of that span (i.e. the first or last instance of the *last column*).

2.  Type: This drop down list has seven options:
- 1: First instance: This option finds the first instance of your selected style on the page.
  - 2: Last Instance: This option finds the last instance of your selected style on the page.
  - 3: First word: This option finds the first word on the page with your selected style applied.
  - 4: Last word: This option finds the last word on the page with your selected style applied.
  - 5: First and Last Instance: This option inserts both the first and the last instance of the selected style on the page into the header placeholder. When selecting this option you should select the text which goes between the two instances in the “Between” field ([see setting #7 on page 28](#)). If the headers are selected to run also on pages without source text ([see setting #16 on page 31](#)), the last instance will be used on following pages.
  - 6: Current Instance: The text inserted in the header using this option depends on where the source content is located. If there is no source text found, the header will show the last found instance. If there is source text found *but it is not in the first line of text on the page*, the last found instance will also be used. If there is source text in the first line on the page, or the text is anchored to the first line, the found source text will be used.
  - 7: Current and Last Instance: This option combines the “current Instance” with the “last instance”. The “text between” option in “Current and Last Instance” is similar to the “text between” option in “First and Last Instance”.
  - 8: All Instances: This will fill the header with all instances on the page.

*Please Note:* Transferring formatting will not work on *All Instances* if punctuation removal is selected. If you need to transfer formatting and remove punctuation on all instances, you should remove the punctuation using GREP Processors instead. Please see [page 35](#) for an explanation of GREP Processors.

☛ The first and last instances can span more than one paragraph. This offers the ability to create multi-line / multi-paragraph headers. Using the appropriate “Remove Punctuation” option (see [setting #5 on page 27](#)), multiple paragraphs can be compressed into one, or be left as separate paragraphs. However, care must be taken when applying styles to ensure that more text than intended is not copied to the header.

3.  Frame Order: This option sets the search order for the text. The search order determines what is considered the “first instance/word” and what is considered the “last instance/word”. Since this setting is set on the individual variable level, you can have more than one such setting per document. This is especially useful when laying out multi-lingual books in which some of the text is right-to-left, and some is left-to-right. There are four frame order options: **1:** “Top, Bottom, Left, Right”. **2:** “Top, Bottom, Right, Left”. **3:** “Left, Right, Top, Bottom”. **4:** “Right, Left, Top, Bottom”.



The diagram on the left illustrates four text frames on a page. If option 1 is selected, the search order will be “A, B, C, D”. If option 2 is selected, the search order will be “C, D, A, B”. If option 3 is selected, the search

# Power Headers

order will be “A, C, B, D”. If option 4 is selected, the search order will be “C, A, D, B”.

4.  Change Case: These settings give you the option to change the case of the words which are inserted into your header. The options are fairly straight-forward. The choices are: “No Change”, “UPPER CASE”, “lower case”, and “Title Case”. Except for “Title Case”, they are self-explanatory.

“Title Case” works intelligently. This means that only words which are “supposed to” be capitalized in a title are capitalized. The rules which the title case uses are as follows:

The first and last words are always capitalized, as well as any word following a period (full stop) or colon. Any compound word separated by a “/”, “.”, dash or m-dash will have the letter immediately following the separator capitalized. Any other word will be capitalized unless it appears in the `_title_lowercase.txt` file. Any word appearing in the file will be lowercase only if it is not a “starting” word. For example: In the title, “The Last of the Mohicans”, the first word “the” will be capitalized, while the second time “the” appears will not. This is because the first word is the beginning of the sentence. Similarly, in the title, “Lion: The King of the Jungle”, the first word “the” will be capitalized because it follows a colon. The lowercase word file can be edited to include any words you like. All words should be separated by a single space.

Any word appearing in the `_title_specialcase.txt` file will always be capitalized exactly as it appears in the file, regardless of position. This is useful for words like “InDesign”, “USA”, “UK”, etc., which have capitals in the middle of the word. This file can also be used to force lowercase on specific words

and override the default rules. So, let's assume you would like the word "a" to always be lowercase, even if it is the first word in your header text. You would simply put the word "a" in your special case file, and it will always be lowercase. When a word appears in both files, the special case file supercedes the lowercase one.

5. Punctuation:  Punctuation Removal: *Power Headers* offers a number of options for punctuation removal: **1:** Don't Remove. **2:** Remove Trailing White Space. **3:** Remove All Extra Space. **4:** Remove Trailing Punctuation. **5:** Remove All Punctuation.

"Don't Remove" does exactly that. It will remove nothing. This includes any trailing returns you might have. If you select a paragraph style as your source style, and select "Don't Remove", you will get a return after the header text. This might be useful if you want a header which spans a number of lines, but if not, your trailing static text in your header will likely "disappear" when it gets overset by the extra return.

The default setting is "Remove Trailing White Space". This is the most common need in headers. It will remove any spaces, tabs, returns, soft returns, or what-have-you, which follow the found text.

"Remove All Extra Space" will convert all of the above mentioned characters to regular spaces. It will also convert double spaces to single spaces. So, if you have two paragraphs which will both go into the header, and you need them to become one, "Remove All Extra Space" will do that for you.

"Remove Trailing Punctuation" does everything "Remove Extra Space" does, but it will also remove trailing punctuation. This includes ",?";!"

# Power Headers

“Remove All Punctuation” will remove all extra spaces and all punctuation, regardless of where it is situated.

If you need more specific character removal, make sure to read about GREP Processors ([page 35](#)). GREP Processors allows for very precise processing.

6.  Text Before /  Text After: These two fields are for text which should be inserted either before or after (or both) your variable text. The reason you might want to use this option rather than simply typing the text into your header frame, is that text will not appear on a page which doesn't receive variable header text. (See [setting #16 on page 31](#) about inclusion options.)

7.  Text Between: Text Between: This option is to be used in conjunction with the “First and Last”, “Current and Last”, and “All Instances” options described in [setting #2](#). This is the static text to be inserted between the two, provided that there is two or more instances to be inserted into the header. If there is less than two instances found, this text will be ignored.

8.  Conditional Text: Conditional Text: This option is available *if any type besides the ones which support text between* is selected. It will replace the “Text Before” or “Text After” in cases where the source text does not appear on the page. In the case that the “Current Instance” is selected, it will replace the “Text Before” or “Text After” on pages which have a found instance, but is not on the first line (which causes the last instance to appear). Both of these options are optional. (See [setting #17 on page 32](#))

9.  Before  After This option determines whether *conditional text* ([setting #8 above](#)) replaces the “Text Before” or the “Text After”.

10.  Copy Local Formatting Copy Local Formatting: This option should be checked if you want local formatting to be copied to your header. The definition of “local formatting” depends on what kind of styles you selected as the source styles. If you selected a strictly paragraph-based header (i.e. you selected “Any” as the character style), local formatting will be defined as any characters styled differently than the paragraph style. This can be through character styles, or as local overrides. If however you selected a character style as your source style, local formatting will be defined only as local overrides. *Please note: Copy Local Formatting cannot be used together with Remove All Punctuation. In the CS3 version it will not be selectable, and in the CS2 version, you will get a warning.*

11.  Italics at Base Weight Italics at Base Weight: This option is to be used in conjunction with “Copy Local Formatting”. If this option is checked, and your local formatting contains italics, the italics will be applied at whatever weight you set your headers. This option assumes your font uses standard naming conventions. It only works if “italics” is named as such. It will not work for “oblique” or “tilted” or whatever other terms might be used. Some fonts have a number prefix for different fonts in a family. Such named fonts will not work either, because the names will not match. It also assumes that the base weight is called “Regular”. If it’s called “Roman” or some other name, and the header is styled at that weight, it will not work either. It *will* however work with fonts which have different optical sizes.

12.  Copy Paragraph Numbering Copy Paragraph Numbering: If this option is selected, any bullets and numbers applied to the source text will be copied to the header. This includes any prefixes or suffixes. With this option selected, the default behavior is to copy the text along with the paragraph num-

# Power Headers

bers. If you need just the numbers, you can select “Don’t Transfer Text” (setting #13 below).

13.  Don't Transfer Text Don't Transfer Text: The main function of this option is to complement the “Copy Paragraph Numbering” option. With this option selected, the found text will *not* be transferred to the header. Only Text Before, Text After, Text Between/Conditional Text, and paragraph numbers and bullets will be transferred. If you need *only* the paragraph numbers to appear in the headers, select this option.

▣▣▣▣➔ There are some useful tricks made possible by the *Don't Transfer Text* option:

◆ Smart use of conditional text can be used to insert custom text using any conditions which can be defined in the conditions options.

◆ Variables are applied in order. This enables some very interesting applications: Let's say you want a header on any page which (for example) *doesn't* start a chapter, you can do this easily. It requires defining two header variables—both of which should be defined with the style which starts your chapters. Set the first variable to create headers on all pages. In the second variable, set it to create headers only on pages with source text, but select *Don't Transfer Text*. The second variable will effectively erase the text created by the first variable!

14.  Ignore Blank Paragraphs Ignore Blank Paragraphs: When there is a paragraph with no text, you probably do not want it to be used for the source text. This option will cause any paragraphs which have no printing characters to be ignored and not be considered a found instance.

This option is only accessible in CS3 or above. In CS2, the behavior will always be to ignore blank paragraphs.

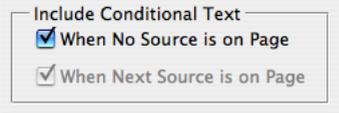
15.  Limit Instances to One Paragraph Limit Instances to One Paragraph: When the source text contains a span of paragraphs which all match the source style, you have the option of choosing how much text will get transferred to the header. If this option is selected, only the first or last paragraph (whichever is defined in [setting #2](#)) in the span will appear in the header. If the option is *not* selected, the entire span will be copied to the header. This can result in a header with multiple paragraphs. The paragraphs can be shortened to a single paragraph using the punctuation removal options ([setting #5 above](#)) or GREP processors ([page 35](#)).

This option is only accessible in CS3 or above. In CS2, the behavior will always be to limit instances to one paragraph.

*Please note:* “Limit Instances to One Paragraph” can effect whether *Power Headers* considers there to be more than one instance on a page. If the option is selected, the first paragraph will be considered the first instance, and the last paragraph the last instance. If it’s not selected, and there’s one span of many paragraphs found, *there will be only one instance*—of the whole span.

16.  Only Include on Pages with source style  
 Include Last Instance on Pages without source style  
 Include Last Instance on Pages without story text Inclusion Options: There are three inclusion options. The first option will only create header text if there is text on the page to transfer. The second option will create header text from the last instance, provided that there is a story on the page which has the style selected in the variable. The third option will create header text from the last instance on every subsequent page. The “Last Instance” can be either a word or a found instance, depending on the header type.

# Power Headers

17.  Conditional Text Options: In this section, there are two check boxes. These check boxes define when the *conditional text* is used (see [setting #8 on page 28](#)). If the first check box is selected (When No Source is on Page), the conditional text is used on pages which have no source text at all. The second check-box is for use in conjunction with the “Current Instance” option. When “When Next Source is on Page” is selected, and the “Current Instance” is selected, the conditional text will be used when the previous instance is used and a source is found on the page (i.e. the found source text is not in the first line of text).
18.  Limiting Story Size: This option ignores stories smaller than the specified size. The purpose of the limit is to speed up the updating of the headers. Setting this properly is extremely important! If the number is set too high, your text can be missed. If the number is set too low, and you have a lot of short stories (for example – side heads), updating headers can take considerably longer!

## Alternate Styles



This option is designed to deal with situations where there is more than one style used in similar fashions. This enables any number of styles to be used to define the header variable. The correct instance of any one of these styles will produce header text.

The setup of the alternate styles is somewhat different between CS2 and CS3 or later. In CS2, the alternates are defined using a menu item: Layout → Power Headers → Define Alternate Style. The menu item needs to be selected again for each alternate defined. In CS3 the alternates are defined

from within the variable definition dialog. Using this Button:  in CS3 or CS4 will bring up a dialog which allows the creation and deletion of alternates. In CS2, CS3, and CS4, choosing to create a new alternate will bring up a dialog which looks like this:



There are two drop-down lists in the dialog. The options in the drop-downs are the same as the style definitions in the main dialog. Any combination of character styles and paragraphs styles can be defined as alternates. These settings can be set irrespective of the style settings in the main dialog. The only limitation is that there must be at least one style selected.

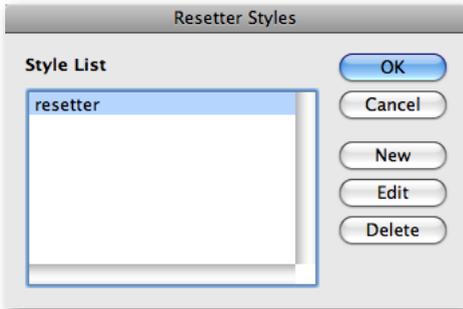
### Resetter Styles (CS3 or later)

This option is only accessible in CS3 or above. Unfortunately, in CS2, there is no way to reset the header, based on styles.

Resetter Styles offer a tremendous amount of control over which text will appear in the header and when. When a Resetter Style is encountered, some or all of the found style instances will be removed from the potential text that the header may contain. Resetter Styles can be used in a lot of advanced uses, but probably the most common use is in hierarchical headers such as the ones you see in this manual, where there is main header text followed by sub-header text—but only if applicable. When a new section starts, the sub-header text from the previous section must be removed, as well as any instances on the current page prior to the start of the current section.

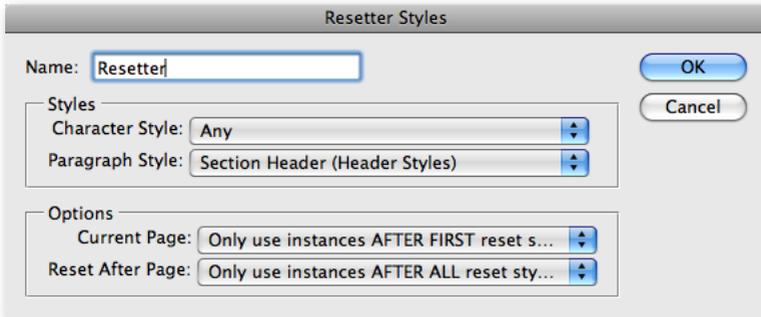
Resetter Styles are defined within a specific variable by clicking on the “Resetter Styles...” button.

# Power Headers



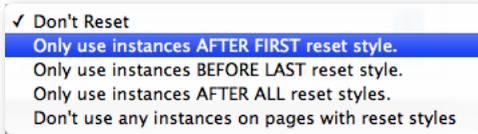
You can define an unlimited number of Resetter Styles, and each Resetter Style can have different settings. To create a new Resetter Style, click “New”. To edit an existing one, select the desired Resetter Style and click “Edit” or double click the selection. To remove a Resetter Style, select the style you wish to remove and click “Delete”.

As with the other source style options of *Power Headers*, Resetter Styles can be selected as either paragraph styles, character styles, or both.



There are two basic functions for Resetter Styles: Limiting the range on a current page, and removing header text which was picked up on a previous page.

For the current page, there are five options:



**1:** “Don’t Reset” will do nothing. **2:** “Only use Instances AFTER FIRST reset style” will cause *Power Headers* to discount any instances on the current page before a Resetter Style. **3:** “Only use Instances BEFORE LAST reset style” will cause *Power Headers* to discount any instances after the last Resetter Style on the page. **4:** “Only use Instances AFTER ALL reset style” will cause *Power Headers* to discount all instances each time a Resetter Style is encountered for the current page. **5:** “Don’t use any instances on pages with reset styles” will cause *Power Headers* to discount all instances if a Resetter Style is encountered anywhere on the current page.

For following pages there are three options:



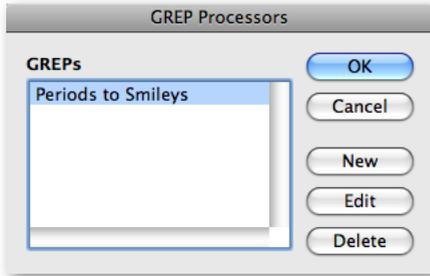
**1:** “Don’t Reset” will do nothing. **2:** “Only use Instances AFTER ALL reset style” will cause *Power Headers* to discount all instances each time a Resetter Style is encountered on a previous page. **3:** “Reset style erases all instances on the same or previous page” will cause *Power Headers* to blank out the headers after any page that Resetter Style is seen.

## GREP Processors (CS3 or later) GREP Processors...

GREP processors offer a tremendous amount of flexibility in controlling the content and formatting of your headers. They can be used to style part of your header with a specific character style. They can be used to add or

# Power Headers

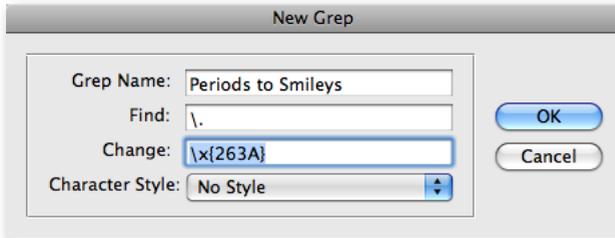
remove parts of your header according to your specific needs. Clicking the GREP Processors button will bring up the following dialog:



You can add as many GREP Processors to your variables as you like. Click on the New button to create a new GREP Processor, the Edit button to change an existing one, and the delete button to remove the selected one.

GREP Processors utilizes features that were not available in InDesign prior to CS3, and therefore is only available in CS3 or above.

Clicking the “New” or the “Edit” button will bring up the dialog below. In the dialog, there are two required settings: the GREP Name, and the Find contents. If the Change contents and Character style are both left blank, the GREP Processor will remove the specified GREP query from the header. If the Change field is filled, in, the found text will be replaced with the change text, and if a character style is selected, the style will be applied. Any GREP which InDesign accepts in the Find/Change dialog, is acceptable for GREP Processors as well.



*Please Note:* GREP Processors are applied in the order that they are defined. There are times that the order of the GREPs can effect the end results, so take care as to which order they are defined.

If you need GREP Processors to be applied to all variables, you can create Global GREP Processors (page 41).

## Inserting Header Variables



Once a header variable is defined, the variable can be inserted into a text frame by right-clicking in a header frame which is labeled with the correct label. If the variable is defined without a header label assigned, the variable can be inserted in any text frame. The variable will replace the selected text. Creating the variable will automatically apply the correct style and wrap the variable in xml tags. If the tags are visible in the view preferences, there will be a visual indication of where the header variable starts and stops.

If the text frame is not labeled with the correct header label, no header variable will be inserted.

# Power Headers

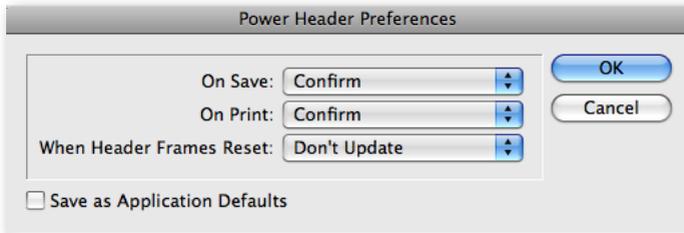
## Updating Headers

*Power Headers* does not work dynamically. This means that updating headers is required after the initial setup, and then again after each re-flow of text which could affect the header content. It is however, very easy to update them. Simply select “Layout → Power Headers → Update Document Headers”. This will update all of your defined header variables.

## Automatic Updates

There are certain times when the headers can be automatically updated to help prevent mistakes. There are three main triggers for this: 1) When a document is saved. 2) When a document is printed. 3) When Header Frames become detached from a master page, or become doubled up.

There is a preference dialog for controlling the behavior in each of these three situations:



For each of these three situations, you can decide to either update without warning, update after a confirm, or not update.

If the “Save as Application Defaults” check-box is selected, the updating preferences will be saved as the default for the application. The application defaults will be used unless specific preferences are set on a specific document.

*Please Note:* Due to performance issues in CS2, the automatic updating when the document is saved or printed is disabled in CS2. The document can only be automatically updated when header frames become invalid.

### Header Frame Overrides

When *Power Headers* updates the header text, any header frames which reside on master pages are automatically overridden. This means that the frames which started off being on the master page and were not editable, become located on the individual pages and are editable like any other text frame. This is analogous to command/control-shift-clicking on a master frame.

This can become an issue when re-flowing pages, and especially when applying master pages. When applying a new master page to an existing one, any overridden master page items get completely detached from the master page, and if the item exists on both masters, you wind up with double items. This is particularly an issue with header frames. When the duplicate frames appear on the page, both copies would get updated by *Power Headers*. *Power Headers* solves this problem by automatically removing the original master frame when frames become doubled up. Additionally, when the master header frame is edited, the overridden master frame on the local pages will be automatically updated on the next update of *Power Headers*.

There is a menu item: “Remove Overrides on Header Frames...” under the *Power Headers* menu. This will return the (chosen) header frame on selected pages to the master page (and will revert any variable header text to the default master text). This option was included in the original release of *Power Headers* because it lacked the automatic removal of master frame overrides. *Power Headers* as of version 2.0 will automatically remove overrides when

# *Power Headers*

necessary, so this menu item should not be needed under normal circumstances.

## **Stacking Order and Groups**

When overriding master page objects in InDesign, the stacking order can be affected. This is because master page items are always below regular page items. When master page items are overridden, they become regular items, which sometimes causes changes in the stacking order. The exact behavior was changed between CS2 and CS3. In CS2, all items when overridden are moved to the top of the stacking order, while in CS3 they are kept on the bottom of the stacking order. When overriding master header frames, *Power Headers* always overrides them in the order in which the header variables are defined.

If the stacking order of header frames is important, it can be handled in one of a few ways: **1.** Place the header frames on different layers. Objects on different layers keep their stacking order when overridden. **2.** Group the frames. If a header frame is grouped with other objects, the entire group will be overridden together. When removing overrides on the header frames, the entire group will be returned to the master page as well. **3.** Make sure to define the header variables in the order which will keep them stacked properly: For CS2 this would mean defining the bottom-most one first. In CS3, the top-most one should be defined first.

*Please note:* Overridden header frames should not be grouped or ungrouped on local pages. If they are not kept in their previous “grouped state”, removing overrides may not work correctly.

### Global GREP Processors

Besides the variable-specific GREP Processors, it is possible to define Global GREP Processors. The difference between the two, is that the variable-specific ones are only applied to the text of the specific variable, and at the time the variable is created. Global GREP Processors will be applied to the entire header frame. They are applied after all variables are updated.

One example of a very powerful application of Global GREP Processors is “compound headers”, for example, you can create a header which can contain [Chapter 1, Verse 1 – Chapter 2, Verse 7] or [Chapter 1, Verse 1 – Verse 11], depending on the content. To define such a header, you would create two header variables as First and Last (or Current and Last) Instance. For the Text Between of the Chapter Header, you’d use some uniquely identifiable text, and create a GREP query to remove the text between and reorder the position of the last instance of the chapter header to between the two instances of the verse header.

### Book Files

Headers can also be run on all documents in a book file. To run the headers on a book file, both the book file and any single document must be open.

Running headers on book files is a two step process. First the header variables must be synchronized across all the book files. The defined master document is used for the definitions. To synchronize the variables, select Layout → Power Headers → Synchronize Book Variables. Once the book documents all have their header variables defined, all that needs to be done is select Layout → Power Headers → Update Book Headers. This process updates all the book files, but doesn’t save them. Rather, it leaves them all

# *Power Headers*

open so you have the option of reviewing the changes before saving and closing the files.

## Loading Header Variables

Once you define variables in one document, those variables can be loaded in another document as well. To do so, select: Layout → Power Headers → Load Header Variables... You will be presented with a dialog to pick a document. Once you select a document, the variable definitions in the source document will be loaded.

*Please Note:* If you have more than one style in your document with the same name, the variables might not be loaded correctly, so please try to keep your style names unique.

## Scripting

Most of *Power Headers*' functionality can be accessed via scripting. Scripting access to *Power Headers* is provided by sending custom events to the plug-in. To do this you must first get a reference to the *Power Headers* plug-in. This is done using the `loadedScriptedPlugins` document method. The `loadedScriptedPlugins` method returns an Array. Therefore, to create a variable which references *Power Headers* you'd write:

### **ExtendScript Code:**

```
var powerHeaders = myDoc.loadedScriptedPlugins("powerheaders")[0]
```

### **AppleScript Code:**

```
tell myDoc
  set powerHeaders to item 1 of (loaded scripted plugins label "powerheaders")
end tell
```

**Visual Basic Code:**

```
Set powerHeaders = myDoc.LoadedScriptedPlugins("powerheaders")[1]
```

Once you have a valid reference to *Power Headers*, you simply tell *Power Headers* to handle a custom event:

**ExtendScript Code:**

```
powerHeaders.handleScriptEvent (eventCode:String,[eventData:String],[delay:  
integer]);
```

**AppleScript Code:**

```
tell powerHeaders  
  handle script event event code (String) [event data (String)] [delay (Integer)]  
end tell
```

**Visual Basic Code:**

```
powerHeaders.HandleScriptEvent (eventCode:String,[eventData:String],[delay:  
integer])
```

In the above code, the event code is the custom event which you tell *Power Headers* to handle. The event data is the data sent to the event handler, although not all events require event data. The delay in milliseconds is when the event is sent. This argument is totally optional. If left out, the event will be sent immediately. If it's 0, it will be sent when InDesign becomes idle. Otherwise it will be sent after the specified delay.

# Power Headers

## Event Codes

Below is a list of acceptable event codes along with the description of what they do and the required event data (if any):

*Please Note:* The event codes and event data are case sensitive.

*Please Note:* The loadedScriptedPlugin property is a child of a *document*. The event will be handled in the document containing the loadedScriptedPlugin specified.

Event Code	Description	Event Data
externalDisableHeaders	Temporary disables Power Headers for the parent document of the specified plugin. This is useful for making a script run quicker.	N/A
Example in ExtendScript: powerHeaders.handleScriptEvent ("externalDisableHeaders");		
externalEnableHeaders	Re-enables Power Headers functionality after it has been disabled.	N/A
Example: powerHeaders.handleScriptEvent ("externalEnableHeaders");		
externalRunHeaders	Causes Power Headers to update all header variables in the document	N/A
Example: powerHeaders.handleScriptEvent ("externalRunHeaders");		
externalLoadHeaders	Loads Power Header variables from a different document	String of the fullName of the source document

Event Code	Description	Event Data
Example: (given a reference to a File object of the source file) <code>myFileName = String(myFile.fullName)</code> <code>powerHeaders.handleScriptEvent ("externalLoadHeaders",myFileName);</code>		
externalLabelFrame	Gives a Header Frame a label	Comma separated string of two items: the text frame id and the Header Label
Example: (given a text frame with an id of 291) <code>powerHeaders.handleScriptEvent ("externalLabelFrame", "291,Header Frame");</code>		

# *Power Headers*

## Sample Script

Below is a sample Javascript which will update all *Power Headers* variables in the current document:

```
var doc = app.documents[0];
var powerHeaders = doc.loadedScriptedPlugins ("powerheaders")[0];
if(powerHeaders){// if Power Headers is not loaded, the item will be undefined
    powerHeaders.handleScriptEvent ("externalRunHeaders");
} else {
    alert("Power Headers is not loaded!");
}
```

Here's the same script in AppleScript:

```
tell application "Adobe InDesign CS4"
    set theDocument to active document
    tell theDocument
        set powerHeadersPluginList to loaded scripted plugins label "powerheaders"
        if length of items of powerHeadersPluginList is greater than 0 then
            set powerHeaders to item 1 of powerHeadersPluginList
            tell powerHeaders
                handle script event code "externalRunHeaders"
            end tell
        end if
    end tell
end tell
```

### Script Labels

APID, (and in turn *Power Headers*) makes extensive use of script labels for tracking objects. If you must use script labels on text frames, ideally you should use private labels (used with the insert label and extract label methods). If you must use the public labels you should *append* the existing labels instead of simply replacing the existing ones. Otherwise you stand the risk of trampling *Power Headers* settings specific to text frames.

In other words, always use:

```
myTextFrame.insertLabel("myCustomLabel")
```

or if absolutely necessary:

```
myTextFrame.label = myStoryOrPageItem.label+"myCustomLabel"
```

instead of:

```
myTextFrame.label = "myCustomLabel"
```

## Cookbook

For some examples of how to use *Power Headers*, [please refer to the \*Power Header\* tutorial web page here.](#)

## Addendum

### What is “APID ToolAssistant”?

Classically, there are only two ways to automate InDesign. One way is by scripting. Scripts are a series of commands written in one of the three supported scripting languages — AppleScript for the Macintosh, Visual Basic for Windows, or ExtendScript for both platforms. ExtendScript is Adobe’s

# *Power Headers*

version of Javascript. For scripts to run, they must be placed in the application script folder and explicitly run. For automation to be truly automatic, it requires “event processing”. This means that certain events trigger specific processes to take place. There is limited event-triggered scripting available in CS3.

To achieve true integration, plug-ins must be programmed using C++. Developing C++ plug-ins is a very involved, and lengthy process. APID ToolAssistant is a plug-in which allows very fine grained event processing using ExtendScript. It allows for creation of advanced plug-ins in a fraction of the time required when programming using C++. It also aids in the creation of hybrid plug-ins which mix ExtendScript and C++ for maximum efficiency.

Without the APID ToolAssistant, it would not have been possible to create the collection of plug-ins we offer in the same amount of development time. This increased programming efficiency means more functionality for the end user and lower prices.

Additionally, APID ToolAssistant offers the ability to attach scripts to specific objects. These attached scripts can be run automatically when triggered by specific events. Anyone who has a licensed version of the APID ToolAssistant plug-in installed can create and use these attached scripts. There are also additional scripting properties and methods which are usable by scripters. For more information see Rorohiko’s web site.

### Acknowledgements

We wish to thank the many people who have helped make In-Tools plug-ins possible.

To those who give graciously of their time and expertise on the InDesign scripting forum: thank you for introducing us to automating within InDesign. Most specifically, Dave Saunders, Olav Kvern, and Peter Kahrel.

To Kris Coppieters, thank you for your incredible support throughout the entire development process. Your help with product improvements and with coding tips and techniques has been incredible. Your support has been above and beyond what we could have reasonably expected. It has really been a pleasure working together. Thank you.

To all our beta testers, thank you for the valuable input which helped shaped our plug-ins. To Brad Walrod: a special thank you for your unrelenting comments and suggestions. To Peter Gold and Raphael Freeman: thank you for your many useful suggestions.

# *Power Headers*

## **License Agreement**

**Please read the following document carefully!**

This is a legal agreement (the Agreement) between you and Print Israel LLC (In-Tools), the developer of the In-Tools software. This Agreement pertains to your use of the In-Tools software and documentation which are provided to you by In-Tools (collectively, the Software). By copying, installing, or otherwise using the Software, you accept all the terms and provisions of the Agreement.

### **Terms of Use**

The provisions of the [In-Tools Privacy Policy](#) are incorporated herein and made a part hereof.

### **License**

This Agreement grants you a personal, non-exclusive, non-transferable, non-sublicensable right to use one copy of the Software for use on a single computer and/or workstation for each license that you purchase. You may try the Software before purchasing a license, for the specified amount of time in the materials accompanying the Software. YOU UNDERSTAND AND AGREE THAT BY USING THE TRIAL SOFTWARE, AFTER THE AMOUNT OF TIME YOU MAY NOT BE ABLE TO CONTINUE TO USE THE SOFTWARE UNLESS YOU PURCHASE A LICENSE.

In-Tools reserves all rights in the Software not expressly granted herein, including without limitation ownership and proprietary rights.

### **License Restrictions**

You may not reverse engineer, decompile or disassemble the Software. The restrictions contained herein apply equally to any updates that may be provided to you by In-Tools.

### **Intellectual Property Rights and Third Party Materials**

All service marks, logos, trade names, trade dress, and trademarks of In-Tools (collectively Marks) are the exclusive property of In-Tools and nothing in this Agreement shall grant you the license to use such Marks. All intellectual property rights in the

Software are owned exclusively by In-Tools and are protected by United States of America copyright laws and international copyright treaty provisions. Any violation of the terms set forth herein is expressly prohibited by law and may result in severe civil and criminal penalties.

### Disclaimer of Warranties

ALL SOFTWARE PROVIDED BY IN-TOOLS ARE PROVIDED AS IS. IN-TOOLS AND ITS AGENTS, PARTNERS, OR EMPLOYEES MAKE NO REPRESENTATION OR WARRANTY TO YOU OR ANY OTHER PERSON OR ENTITY, WHETHER EXPRESS, IMPLIED, OR STATUTORY, AS TO THE DESCRIPTION, QUALITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY, COMPLETENESS, OR FITNESS FOR A PARTICULAR USE OR PURPOSE AS TO THE SOFTWARE PROVIDED TO YOU, OR AS TO ANY OTHER MATTER, ALL SUCH WARRANTIES HEREBY BEING EXPRESSLY EXCLUDED AND DISCLAIMED.

NEITHER IN-TOOLS NOR ANY OF ITS AGENTS, PARTNERS, OR EMPLOYEES WARRANTS THAT THE SOFTWARE OR SERVICES ARE ERROR FREE OR WILL OPERATE WITHOUT CAUSING LOSS OR INTERRUPTION YOUR COMPUTER USAGE OR DATA. YOU ASSUME TOTAL RESPONSIBILITY AND RISK FOR YOUR USE OF THE SOFTWARE.

### Limitation of Liability

IN NO EVENT SHALL IN-TOOLS, ITS AFFILIATES, SUBSIDIARIES, PARENT COMPANIES, AGENTS, NETWORK SERVICE PROVIDERS, PARTNERS, OR EMPLOYEES BE LIABLE TO YOU OR ANY THIRD PARTY IN ANY RESPECT FOR ANY COSTS OR DAMAGES ARISING EITHER DIRECTLY OR INDIRECTLY FROM THE USE OF IN-TOOLS SOFTWARE INCLUDING WITHOUT LIMITATION ANY ACTUAL, INCIDENTAL, CONSEQUENTIAL, EXEMPLARY, PUNITIVE, RELIANCE OR SPECIAL DAMAGES, OR FOR ANY LOSS OF REVENUE, PROFITS, USE, DATA, GOODWILL OR BUSINESS OPPORTUNITIES OF ANY KIND OR NATURE WHATSOEVER, ARISING IN ANY

# *Power Headers*

MANNER FROM ANY CAUSE OF ACTION OR CLAIM RELATING TO THIS AGREEMENT, TO THE SERVICES PROVIDED BY IN-TOOLS OR TO THE USE OF THE SOFTWARE. THIS LIMITATION OF LIABILITY APPLIES EQUALLY TO ANY THIRD PARTIES WITH WHOM IN-TOOLS ENTERS INTO A BINDING AGREEMENT TO MARKET, DISPLAY AND/OR DISTRIBUTE THE SOFTWARE TO END USERS AND SUCH THIRD PARTIES SHALL NOT BE LIABLE TO YOU OR ANY OTHER THIRD PARTY FOR ANY OF THE DAMAGES SET FORTH ABOVE.

## **Export Control Laws.**

You agree to comply with all laws, rules and regulations applicable to the export of the Software. Specifically, you shall not export, re-export or transship the Software, or the direct product thereof, in violation of any United States laws and regulations which may from time to time be applicable. None of the Software or underlying information or technology may be downloaded or otherwise exported or re-exported (i) into, or to a national or resident of, Cuba, Iran, Iraq, Libya, North Korea, Sudan, Syria or any other country to which the U.S. has embargoed goods; or (ii) to anyone on the U.S. Treasury Department's list of Specially Designated Nationals or the U.S. Commerce Department's Table of Denial Order. By downloading or using the Software, you are agreeing to the aforementioned and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list.

## **Jurisdiction and Other Provisions**

This Agreement and the relationship between In-Tools and you shall be governed in accordance with the laws of the State of New York. Any dispute between you and In-Tools regarding this Agreement will be subject to the exclusive jurisdiction of the state and federal courts in the State of New York. You agree to submit to exclusive jurisdiction in the State of New York, and you expressly waive all defenses to jurisdiction.

This Agreement is the entire agreement between you and In-Tools relating to your use of the Software and supersedes any other communications or advertising with respect

to the Software. If any provision of this Agreement is held invalid, the remainder of this Agreement will continue in full force and effect. In-Tools failure to insist upon or enforce strict performance of any provision of this Agreement shall not be construed as a waiver of any provision or right. Neither the course of conduct between the parties nor trade practice shall act to modify any provision of this Agreement. In-Tools may assign its rights and duties under this Agreement to any party at any time without notice to you.

### Modifications of Agreement

In-Tools may modify this Agreement at any time without providing notice to you. Such modification will be made by posting a revised agreement on its web site and such modification shall be deemed effective immediately upon posting of the modified Agreement. Your continued access or use of the Software or Services shall be deemed your acceptance of the modified Agreement. You agree to check our web site periodically to review such modifications.

### Termination or Modifications of Software

You may terminate this Agreement at any time. In-Tools may immediately terminate the Agreement if you breach any representation, warranty, agreement or obligation contained or referred to in this Agreement. Upon termination, you must dispose of the Software and all copies or versions of the Software by destroying the Software. In-Tools reserves the right to modify the Software at any time without providing notice to you. In-Tools right to modify the Software applies to all or any aspect of the Software.

YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT AND UNDERSTAND IT, AND THAT, BY USING THE SOFTWARE, WEB SITE, YOU AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS.