

# GuideMaker

v1.0 (29.9.2014)

Oct 2, 2014

<a href="#">Overview</a>	Copyright (C) 2005 Thomas Breeden, Aglet Software
<a href="#">Installation</a>	
<a href="#">Requirements</a>	Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice appear in supporting documentation.
<a href="#">Startup</a>	
<a href="#">Settings</a>	
<a href="#">Tools</a>	
<a href="#">NodesWindow</a>	
<a href="#">EditingWindow</a>	
<a href="#">FormattedDisplay</a>	Thomas Breeden makes no representations about the suitability of this software for any purpose.
<a href="#">AGTips</a>	
<a href="#">Known</a>	
<a href="#">History</a>	It is provided "as is" without express or implied warranty.
<a href="#">Author</a>	
<a href="#">DictionaryLicense</a>	This product includes a wordlist developed by Geoff Kuenning and other unpaid contributors.

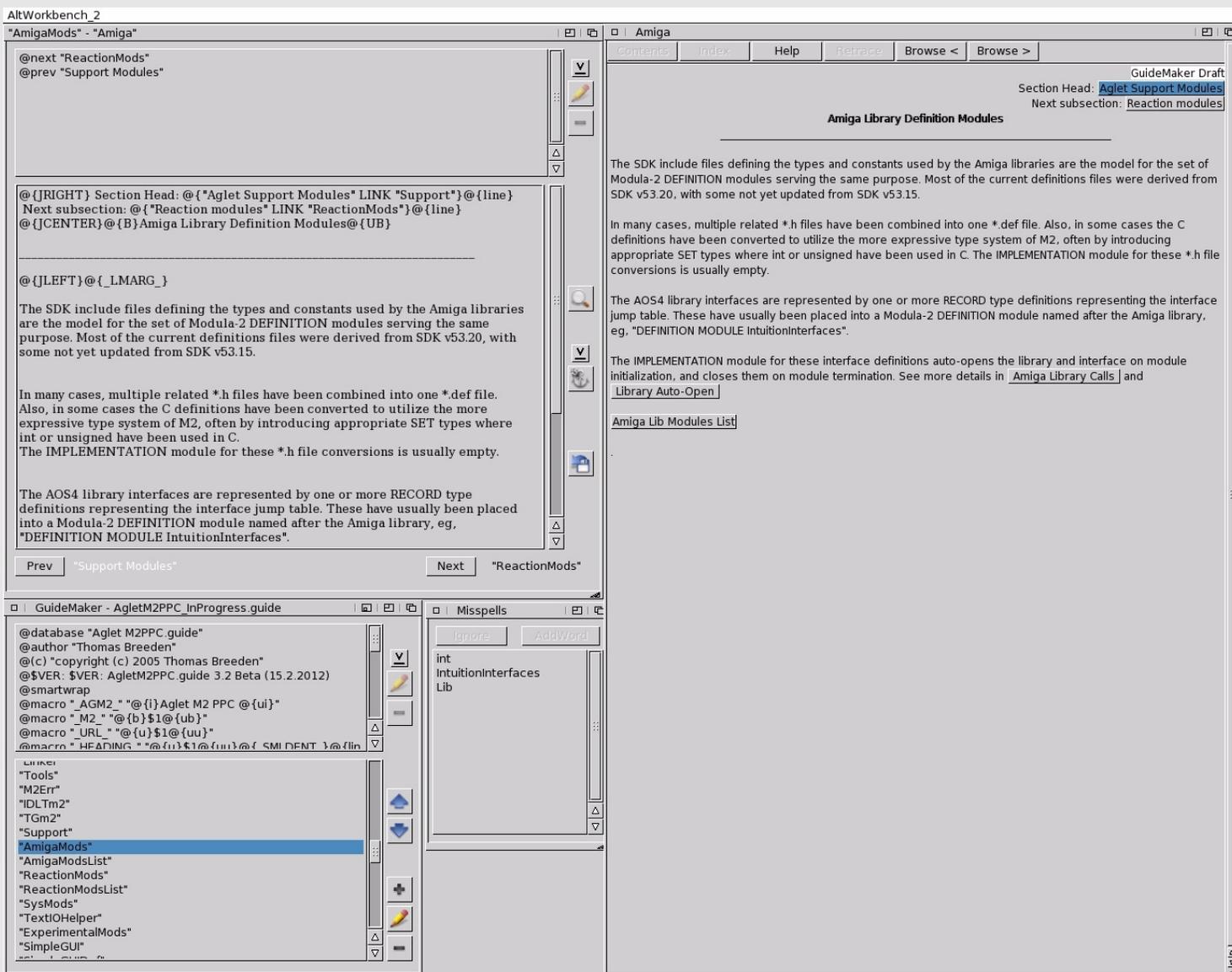
## Overview

WhatsNew v1.0

**GuideMaker** is an **AmigaGuide(R)** file creation utility with support for also providing the documentation in other formats (via RTF file export).

It combines the Reaction Text Editor gadget and other Reaction GUI elements, DataTypes for AG preview display, and control software that understands and can manipulate AG format files. The package is the beginnings of a visual development environment for AmigaGuide.

**The information** is presented in four windows on the GuideMaker screen:



1. The "Node List Window" has the nodes and global commands and gadgets for manipulating them.

2. The "Node Edit Window" contains the contents of the currently selected node and its local commands.
3. The "AmigaGuide Window" shows the formatted display.
4. The "Misspell Window", if spell checking is used, showing possible spelling errors.

### **Why?**

I needed help in writing an AmigaGuide document. AG in spite of its flaws, is still the best way to add slick and quick context-sensitive help to your Amiga program.

The newest release adds spell checking, providing an opportunity for maintainers of existing .guide files to easily cleanup some of the silly misspellings that tend to plague these files.

The other reason I put **GuideMaker** together now was prove that the native OS4 Modula-2 compiler, **Aglet M2PPC**, <http://home.ntelos.net/~tbreeden/>, I have released could be reasonably used for a non-trivial project.

## What's New in Release 1.0

WhatsNew09

### User Interface Improvements

- > Editor font is now settable via Menu, CLI Switches, or ToolTypes.
- > AG command parameter requester better customized for each AG Command, with a little parameter validation.
- > AG command parameter requester remembers its position and size when popping up.
- > Doubleclick on AG commands list item means "edit the parameters".
- > Using AISS images for buttons if possible.
- > More gadget "Help Hints" are supplied.

### More Control over Program modes

- > PubScreen, HelpHints, and EditorFont settings are saved to the startup initialization environment variable.

### Preview Window

- > The preview window now has a vertical scrollbar in it.
- > The program menustrip is available when the preview window is selected.

### Spell Checking

- > The locations of the spell check dictionary and of the user word file, and the environment variables used for them are displayed in the "About" window.
- > A progress bar is displayed when SpellCheck is first selected (and has to load its dictionary into memory).

### Docs

- > Added a few SYSTEM links in the GuideMaker.guide (this document) file using *URLOpen* to access an Internet page via a single click in AmigaGuide. Seems to work ok with Odyssey.
- > Added some "Tips" for using AmigaGuide commands and attributes.

### Bug Fixes

- > The program no longer terminates if you attempt to edit a global or local command list command that takes no parameters. (The edit button is disabled for these).
- > The AG command `@{SetTabs ...}` can be entered into the editor from its chooser list.

## What's New in Release 0.9

### **Spell Checking**

A fairly large dictionary word list file for English (American) spelling optionally can be loaded to notify the user of potential misspellings. He/she can also easily add additional words to a user word list as well.

### **Iconify Support**

The main window now has an iconify gadget.

### **Option to Make Tab Chars Explicit**

Added option to have the program automatically replace any tab characters encountered in the guide file with the "@{tab}" attribute.

### **More Control over Program modes**

ENV variable, ToolTypes, or CLI startup template are all available now to initialize the program operation modes.

### **Nice Icon for GuideMaker**

Now there is a real OS4.1 icon for the program, courtesy of Frank Ruthe. see his icon package at OS4Depot under graphics/icon/cha05e90\_icons\_2010-2.lha.

### **User Interface Improvements**

- > The startup file name requester is a littler clearer for the user.
- > Synchronized the selected node in the GuideMaker main window with that shown in the editing window.
- > Added a Help menu item.
- > Added a couple of RAMiga keystroke menu commands (for "Save" and "Quit").

### **Bug Fixes**

- > The PgmVersion is now correctly detected by the Amiga Version command.
- > If an icon already exists for the .guide file, no longer replacing the existing tool with SYS:Multiview.
- > If the file save fails, the program now gives a "try again?" requester, which gives the user a chance to close any other programs (eg, Multiview) which may have write locked the file.
- > The temp files created in T: are deleted on program exit.

### **AmiUpdate**

Updates (if any) will be available via AmiUpdate. See web page at <http://www.amiupdate.net> .

## AmigaGuide

**AmigaGuide(R)** may still have a place in the OS4+ Amiga environment. Even now, it seems to me that it offers a lot more than a text file displayed with NotePad, and is quicker than starting up a web browser to look at HTML or starting up AmigaPDF for a PDF document. It is good for interactive help within a program, and not too bad for longer documentation.

It does have many **idiosyncrasies** and quirky problems. Sometimes it is difficult to predict exactly what the AG formatter will do, so is almost imperative that you continually check it's appearance as you write it. **GuideMaker** will help in this.

The OS4 Autodocs on the AG DataType and AG Library give bare bones information, but the best source for a thorough course in how AmigaGuide works in practice is to be found in **Frank Bunton's** "How\_To\_Write\_Amigaguide" document. Look for it under "Docs/Help" at Aminet:

<http://os4.aminet.net/package/docs/help/aguidehowwrite> .

Also see the tutorial on the **AmigaOS 4 Wiki**:

[http://wiki.amigaos.net/wiki/AmigaGuide\\_101](http://wiki.amigaos.net/wiki/AmigaGuide_101)

and the **AmigaGuide Autodoc**:

[http://wiki.amigaos.net/amiga/autodocs/amigaguide\\_dtc.txt](http://wiki.amigaos.net/amiga/autodocs/amigaguide_dtc.txt).

**GuideMaker** will successfully load most pre-existing AmigaGuide files (ie, those not created by **GuideMaker** itself).

and see a few [AGTips](#) here.

## Installation

Unpack the archive into the directory of your choice.

You will get these files in a subdirectory named "GM":

- the GuideMaker executable.
  - GuideMaker.guide and GuideMaker.pdf.
  - the Dictionary.wordlist file.
  - a few JPG files to which GuideMaker.guide has links.
  - the dictionary word list license statement.
  - a readme file and history file.
  - the GuideMaker.release version file.
- a small program, *GetTopScreen* , is included to support the separate display of image files on a public screen. (ie, I have not discovered how an AG file can pass on the name of the public screen on which it is running.)

The latest release of this document does not use *GetTopScreen* , instead displaying inline via an AG in-line alternative in @{"picture" LINK file.jpg/MAIN}.

## Requirements

1. AOS4 (July 2007 update or later). AOS 4.1 upd 6 is recommended for the most stable *Multiview* .
2. AmigaGuide (v40), distributed with AmigaOS 3.1 and AOS4+. The latest AOS4.1 distributed AmigaGuide (version 53.2) is recommended.
3. "T:" assigned to a temporaries directory.
4. A Reaction Text Editor gadget *later than v52.1* is highly recommended for stability. (first publicly released with AOS 4.1)

## Startup

**GuideMaker** is can be started from the **CLI** or the **WB**.

**Template:** "FILE, PUBSCREEN/K, EDITFONT/K, EDITFONTSIZE/N, MAKEICONS/S, NOMAKEICONS/S, AUTOPREVIEW/S, AUTOPREVIEWOFF/S, USETABATTR/S, USETABATTROFF/S, HELPHINTS/S, HELPHINTSOFF/S, SPELLCHECK/S, SPELLCHECKOFF/S"

Any of the template settings can also be used as a ToolType for project icons with APPDIR:GuideMaker as their tool when starting from the WB.

If no **FILE** is given, it will open up a file requester and ask for one. If the name given does not exist, it will offer to create a new AG file.

"**PUBSCREEN** <screenname>" will cause the program to startup on one of your predefined Prefs public screens (GuideMaker does not create screens). If no **PUBSCREEN** is given it will open on the default public screen.

See the next page for the settings: **MAKEICONS**, **AUTOPREVIEW**, **USETABATTR**, **SPELLCHECK**, **HELPHINTS**, **EDITORFONT**

## Settings

**GuideMaker** is has some mode settings available for toggling in the program menu strip:

- |                   |   |
|-------------------|---|
| "Make Icons"      | - If selected, an icon will be saved along with the AmigaGuide file. Its default tool will be <i>MultiView</i> .                      |
| "Auto Preview"    | - If selected, the preview window will automatically update each time the node editing window goes to a new node.                     |
| "Use @ {tab}"     | - If selected, the program will automatically replace any Tab characters encountered with the AG attribute command "@ {tab}" .        |
| "Spell Check"     | - If selected, Each new node loaded into the Edit Window will be spellchecked and potential errors displayed in the Misspells window. |
| "Help Hints"      | - Turns gadget help bubbles on or off.  |
| "Editor Font ..." | - Allows selection of the font to use in the editor window.   |

---

These settings will be retained in the global session environment whenever you change any one of them via the menu. They can be saved permanently across boots (ie, in ENVARC:) via the menu item:

"Save Settings (Permanent)"

The environment variable name is *Guidemaker/GMSwitches* .

The corresponding CLI switches (or ToolTypes) will turn on the their respective setting, overriding any settings set to "off" by default or by the environment variables.

## Tools

### Export As RTF

This will write out the current AG file as an RTF formatted file.

Formatting is fairly well converted, and intra-document LINKs to AG nodes are preserved as live references to "bookmarks" in RTF. External links (as those to images) are not handled however.

The RTF background is white, while AmigaGuide's is grey. This program will take that into consideration when doing a translate of "@{bg background}", but not "@{bpen 0}".

One of the main benefit of the RTF file is the ability to import it into *OpenOffice* or *LibreOffice* on Linux, Mac, or Windows, and use that program to create a PDF file.

*FinalWriter* on Amiga will usually import the RTF File successfully, but does not handle LINKs nor colored text.

*Cinnamon Writer* 0.75 and 0.751 does not import the RTF File successfully.

### Spell Check

If you turn on spell checking GuideMaker will load a dictionary wordlist file into memory (and also a user wordlist, if found). With "Spell Check On" each node is checked at the point it is loaded into the Edit Window, and potential misspellings are displayed in the Misspell Window.

Selecting a word in the Misspell Window will also select it in the Edit window.

The AddWord button will add the selected word to the User Wordlist.

Once loaded (takes about 5 secs on Sam 440 or AmigaOne), the memory hash table uses a good deal of storage (10 to 20 MB), but spell checking is very fast.

Spellchecking Environment Variables are:

DICTIONARY_WORDLIST	dictionary name to use, a plain text file one word per line - default is "PROGDIR:Dictionary.wordlist"
WORDLIST	Auxiliary text words dictionary - default is "PROGDIR:User.wordList"

The supplied "dictionary.wordlist" file is derived from the American medium list supplied with the *ISpell* (ver 3.1.20) package, and as such it is covered by the same copyright and license as *ISpell*.  
*ISpell* itself is not used, nor is any code from it.

## Nodes Window

This is the Nodes Window:



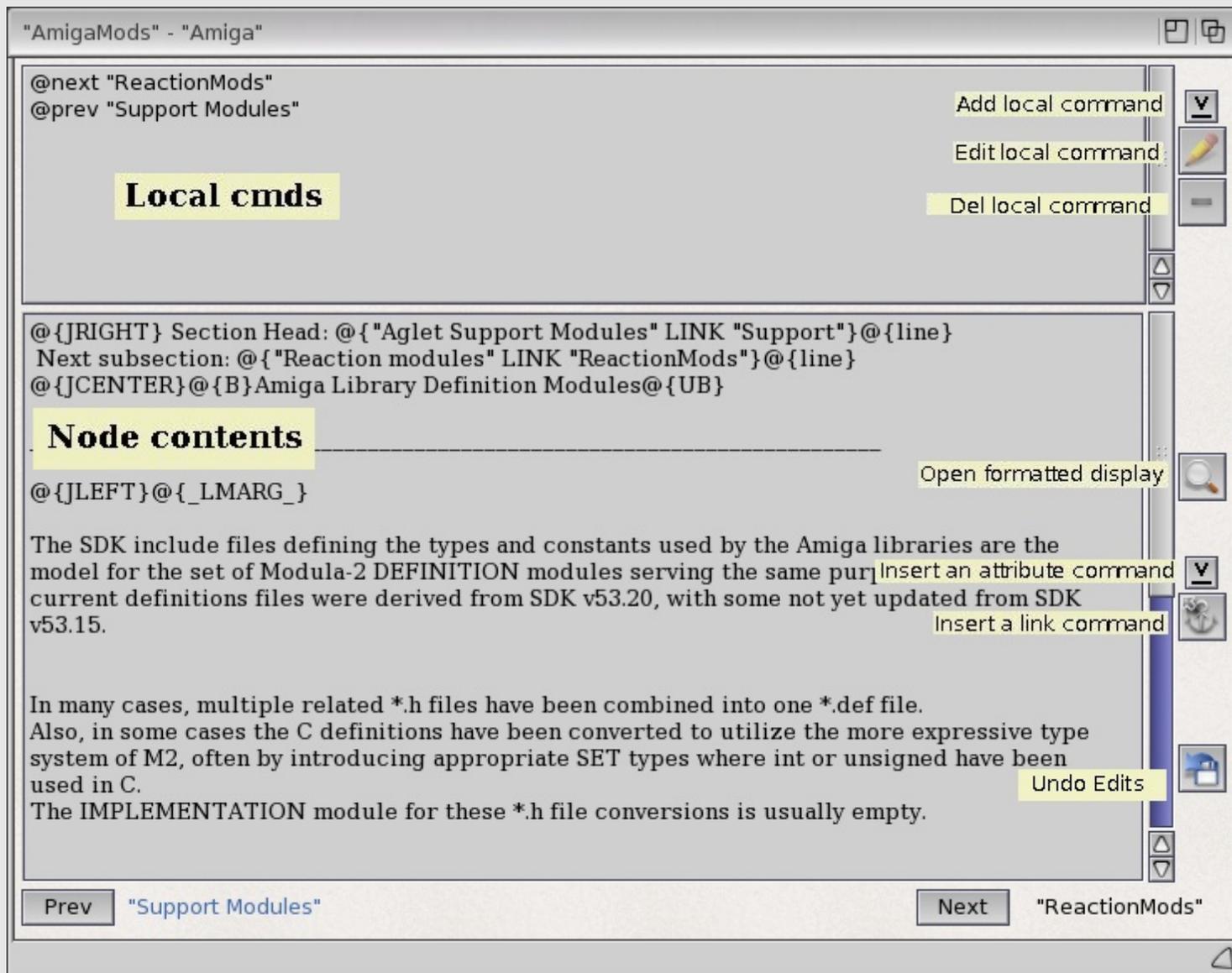
The **top** part of the Nodes Window is for inserting the "global" AG commands that will apply to every node in the file (unless overridden by a node-local command within a node). The top three gadgets allow adding a new global command, modifying an existing one, and deleting a selected one.

The rest of the Nodes Window consists of a sequential list of the **current nodes** in the file. Similar gadgets are here for adding new nodes, modifying a node's title, and deleting the selected node. The two additional gadgets are for moving nodes up or down in the list.

**Click** on a node name and it will appear in the [EditingWindow](#) .

## Node Editing Window

This is the Editing Window:



The **top** part of the Node Editing Window is similar to that of the Nodes Window, except that it is for node-local commands that apply to this node only.

The rest of the Node Editing Window is a **Reaction TextEditor** gadget containing the node contents. You can type in text and use the basic editor keyboard commands (RAMiga-c, RAMiga-v, RAMiga-u, etc) and mouse click commands.

You can also use the **drop-down gadget** at the right to select from the list of attribute commands. After you enter the command's parameters, it will be inserted into the text at the cursor position.

Choosing "@{b}" ON when there is a **selected block** in the Text Editor will also automatically insert a "@{ub}" at the end of the block. Ditto for "@{i}" and "@{u}".

Buttons are available in this window for

- > (re)displaying this node in the formatted preview window
- > inserting one of the LinkOp embedded commands: LINK, RX, RSX, and SYSTEM.
- > removing all edits you have made since this node's contents were brought into the Node Editing Window

The **Next button** at the bottom will load the appropriate following node into the Node Editing Window - either the one specified by the "@{next nodename}" node-local command, if any, or simply the sequentially next node if there is no such node-local command (and similarly for the **Prev button**).

Note that the displayed name of the next or prev node appears in **color** if a node by that name does not exist in the Guide file.

## Formatted Display

**GuideMaker** produces the Formatted Display by writing out files (GuideMakerTmp<pid>.guide) to the "T:" directory, consisting of only one node, the one currently residing in the Node Editing Window, and using the AmigaGuide Datatype in order to display it.

**Note:** The link buttons in this window are not active.

If "Auto Preview" is on, the formatted display will be updated each time you switch to a different node.

## AmigaGuide Tips

The AmigaWiki docs mentioned above in [AmigaGuide](#) do a fair job of describing the AG "Commands", but not so much for the AG "Attributes", which format the text. Frank Bunton's tutorial does better, but I can add some information and clarification.

The following are based on testing with AmigaGuide datatype and library 53.2 (with "@smartwrap" set on globally), but also apply to AmigaGuide 40.

### @{par} and @{line}

With AG 40, these two attributes were added, along with some line-oriented and paragraph-oriented attributes in order to give more flexibility to the text layout. "@{par}" can be thought of as "end-this-paragraph-and-begin-the-next-one".

Normally, line breaks in the paragraph being processed are treated as if they were spaces unless two occur consecutively resulting in an empty line. The empty line triggers a new paragraph; it is *almost* equivalent to entering a @{par}.

The "@{line}" attribute can be used to insert a hard line break without starting a new paragraph.

Though it is not documented explicitly, a number of the v40 attributes are applied only "per-line" rather than "per-character", like text style and color attributes. This makes their exact placement in the text with respect to paragraphs and lines important.

### Position of Attribute Commands in the text

The current value of a line-oriented attribute when the AG starts the processing of the line determines how that attribute will be handled for that line. Using them can be confusing because it seems that some attributes are processed at the beginning of a line (e.g., right margin) and some are processed at the end of a line (eg. justification).

[ParIndent](#)   [LeftMargin](#)   [Justify](#)   [CodeBody](#)

## AmigaGuide Tips, cont'd

**Paragraph Indentation**

Paragraph first line indentation can be set via the "@{pari n}" attribute.

This attribute takes effect only on the first line of a paragraph. As noted in the Autodocs, a negative value produces a nice hanging indent for paragraphs.

There is a **glitch** with "@{pari n}": it does not seem to recognize the first line of a new paragraph unless the "@{par}" is explicitly used.

eg, The following node text:

---

```
@{lindent 24}@{pari -12}@{par}This is a hanging indent@{line}which hangs off to the
left.@{line}
of the left margin, and only the first line "hangs".
```

```
@{line}To use it, you can't rely on the "blank line produces a new paragraph"
behaviour@{line}as stated in the docs.
```

```
@{par}You must issue an explicit "par" instruction before the line you wish to begin the next
paragraph, and then smart wraparound and the
@{line}"line" attribute will continue the paragraph as desired.
```

---

Generates this display:

```
    This is a hanging indent
      which hangs off to the left.
```

```
    of the left margin, and only the first line "hangs".
```

```
    To use it, you can't rely on the "blank line produces a new paragraph"
    behaviour
    as stated in the docs.
```

```
    You must issue the explicit "par" instruction before the line you wish to
    begin the next paragraph, and then smart wraparound and the
```

```
    "line" attribute will continue the paragraph as desired.
```

## AmigaGuide Tips , cont'd

**Left Margin**

You can use the "`@{lindent}`" attribute to set the left margin of the text line either between or within paragraphs.

The current value of this attribute at the time the line begins is used. To make sure this works as you expect, when you want to set the left margin, it is a good idea to put the "`@{lindent}`" just before an explicit "`@{line}`" command.

eg, The following node text:

---

```
@{lindent 12}@{line}The left margin for this line is indented 12. If it line wraps then the wrapping will obey the left margin as well, as can be seen here.
```

To be sure you are in control:

```
@{lindent 24}@{line}Use the line attribute to force a new line and precede it with the line indent attribute. Then the new left margin takes effect at that point and continues across more line wraps .
```

---

Generates this display:

```
The left margin for this line is indented 12. If it line wraps then the wrapping will obey the left margin as well, as can be seen here.
```

To be sure you are in control:

```
Use the line attribute to force a new line and precede it with the line indent attribute. Then the new left margin takes effect at that point and continues across more line wraps .
```

## AmigaGuide Tips , cont'd

**Line Justification**

You can set each line to be left, center, or right justified with the corresponding justification attributes.

Normally you can use the appropriate "@{j...}" attribute before or after a "@{line}" or "@{par}" and, as noted above, and make sure you don't issue a different one until after the end of the last line you want processed with that justification.

This gives you a block of lines, each justified individually, as so:.

This paragraph is centered  
line by line  
one line at a time

However, deliberately or inadvertantly, you can change justification in the middle of a paragraph.

eg, The following node text:

---

@{jleft}@{par}

This is a very long paragraph started with the left justify attribute ON and demonstrates that it is left justified to the left margin on both wrapped and@{line}explicitly entered lines. But if a new centering attribute is set before@{line}the end of the last line, it will effect it.

@{jcenter}@{par}

---

Generates this display:

This is a very long paragraph started with the left justify attribute ON and demonstrates that it is left justified to the left margin on both wrapped and explicitly entered lines. But if a new centering attribute is set before  
the end of the last line, it will effect it.

## AmigaGuide Tips , cont'd

**Code and Body**

These are paired attributes you can use to temporarily turn off any line wrapping in the middle of a node's text.

"@{Code}" begins a section of unprocessed text output and "@{Body}"  
e.g.

```
@{code}

(*-----*)
PROCEDURE CreateAISSImgButton(VAR g:Gadg; AISSname:ARRAY OF CHAR):BOOLEAN;
(*-----*)

VAR basefn :DynStr;
    sqSiz  :DimensionSpec;
    iobj   :ImgObj;

BEGIN

DynStr0.dStrInC(AISSname, basefn);

iobj := GetAISSImage(basefn, AISSstyleSet{NormalAISS, SelectedAISS, GhostedAISS},
TRUE(*FindStdFirst*));
(*Assert(iobj # NilImage, "CreateAISSImgButton FAIL");*)
IF iobj = NilImage THEN
    RETURN FALSE;
END;

GetImageSize(iobj, sqSiz.minsize.x, sqSiz.minsize.y);

g := NewButton("Img", ModifierSet{ShowImg}+FixedSize);

INC(sqSiz.minsize.y, 5);
sqSiz.minsize.x := sqSiz.minsize.y;      (* square the button off *)
sqSiz.maxsize := sqSiz.minsize;
AsgGadgDimensions(g, BothDimensions, sqSiz);

AsgGadgImg(g, iobj);

DynStr0.dStrDispose(basefn);  (* ok? *)

RETURN TRUE;

END CreateAISSImgButton;

@{Body}
```

And after "@{body}" line wrapping begins again. Notice that other AG attributes, such as the left margin, are still in force.

Also, you may need to switch the node's font to a monospaced one in order to get the correct indentation that you created when you wrote the code. AmigaGuide has no facilities for using more than one font in a node.

## Known Bugs and Deficiencies

### Missing Features

- > The optional line number@ parameter of a LINK is not in the input requester.
- > Some AG commands are not yet included: e.g., @{beep}
- > There is no localization. Lo siento. Tal vez puedo añadir una de Español si haya interés.
- > Etc.

### File Being Edited also Open in Multiview

Note that if you have your guide file opened in *MultiView* while you are editing it in **GuideMaker** then a GuideMaker "File Save" will fail. The program will notify you and give you a chance to close *MultiView* and try save again so as not to lose your edits.

### Picture Links in GuideMaker.guide not Appearing

The referenced JPG files should reside in the same directory as the .guide file.

### Program Unexpected Exceptions

Exceptions caused by program errors will be trapped and the location of the error displayed. **GuideMaker** will attempt to abort "gracefully", giving you a chance to save the file before exit.

However, it is always good practice while working to do a "File | Save" from the program menu often rather than wait until all editing is done. Currently **GuideMaker** does not do any timed backups.

### Multi-item Menu Selection

Not supported.

## Version History

[History Older](#)**v1.0** (29.9.2014)

- **Doc:** Using the *URLOpen* program with the SYSTEM AG command to open web pages. Tested with Odyssey for OS4.
- **Doc:** Added a section, "AmigaGuideTips", to clarify `@{par}`, `@{line}`, `@{pari}`, `@{lindent}`, and justification attributes.
- **Doc:** Now just using xxx.jpg/MAIN, which seems to work better than previously, to show pictures inline instead of invoking the GetTopScreen program.
- **Bug:** Requester for `@{pari}` turned a negative indent into a very large cardinal. Changed the "n" parameter to INTEGER.
- **Add:** Using AISS for buttons if possible. Ok for 32 and 16 bit screens, but not 8 bit (LUT) screens. If no AISS or a 8 bit screen, fall back to a simple image.
- **Bug:** Entering a non-existing node name to Next/Prev and then correcting the name to an existing node name and then updating it to a non-existing node name terminated the program with an Assert FAIL. Fixed.
- **Bug:** Program Version appeared to be blank.  
The problem was that the `@$VER:` string of the AmigaGuide command came first in the file. Put an "IMPORT GuideMakerVersion;" at the top of GuideMaker.mod.
- **Add:** Now there is a Font Requester popup button in the NewCmd requester for `@font`.
- **Chg:** Three different pop-up requesters now rather than trying to handle all command and attribute input with one.
- **Add:** Now double-click on existing Cmd in Global or Local command list will automatically invoke an edit requester on the command command just as if it were a click on the edit button.
- **Add:** included PubScreen, if any, in saved settings
- **Add:** The font of the editor window can now be set.  
EDITFONT and EDITFONTSIZE work the same in environment setting and CLI/WBTool. The CLI/WBTool override anything set in environment setting var, "GuideMaker/GMSwitches"

- **Chg:** Now disabling input to other windows when a popup window opens for info. i.e., a modal popup
- **Bug:** Preview's "GuideMaker Draft" display line came out as a white band on 16 and 8 bit screens. Changed to TextPen.
- **Add:** Within an editing session, popup requesters (and the SpellCheck window) come up at the same window position they were left at when last closed.
- **Add:** Displaying the locations of the spellcheck dictionary and user word file, and displaying the locations of these in the "About" popup (as well as the names of the associated environment variables).
- **Add:** Now disallowing an empty target in the LINK popup requester.
- **Add:** Customized HelpHints for the input fields of the the popup command and attribute requesters.
- **Add:** For some global commands where it does not make sense to have more than one instance, such as \$VER, this is enforced.
- **Add:** The preview window title is now correct if a @Title local command is used (which overrides any title added on node creation).
- **Bug:** You would get a program EXCEPTION and termination on attempting to Edit a Global or Local Command which has no parameters. Now the Edit Button is disabled if one of those commands is selected.
- **Bug:** Rarely, you could get a program EXCEPTION and termination on attempting to make a LINK to a target node which does not exist.
- **Bug:** The popup requester for the SetTabs attribute no longer terminates with an EXCEPTION. The tab values are accepted as a string: numbers each separated by a space.
- **Add:** A progress bar displays when the the spell check word lists are first loaded into memory.
- **Chg:** The AG global command "@\$VER:" GM now parses out the optional comment part that can appear after the specified parts and uses a separate string field in the popup for it.
- **Bug:** The MAIN node now cannot be deleted or moved up or down.
- **Add:** The preview window now has a vertical scrollbar in it.
- **Add:** HelpHints can now be turned off/on.

## Version History, Cont'd

### v0.9 (14.10.2011)

- Nice custom icon for the program, courtesy of Frank Ruthe. see his icon package at OS4Depot [http://os4depot.net/index.php?function=showfile&file=graphics/icon/cha05e90\\_icons\\_2010-3.lha](http://os4depot.net/index.php?function=showfile&file=graphics/icon/cha05e90_icons_2010-3.lha)
- Synchronized the selected node in the GuideMaker main window with that shown in the editing window. ie, if Next or Prev moves the Edit window node, then the main windows selected node will be updated.
- Added a Help menu item. For now it simply spawns "Multiview PROGDIR:GuideMaker.guide".
- The PgmVersion was not being displayed by the Amiga Version command.
- The initial program startup was made a little more user friendly w/r/t the guide file name.
- The temp files created for the AmigaGuide live display are now named with the program's pid, so that possibly two instances of GuideMaker could run at once. Also, GuideMakerShow termination code attempts to delete these files.
- WriteItOut() now gives a "try again?" requester if it fails, so that the user always has a chance to close Multiview open on the same guide file - the most usual cause of write fail here.
- If an icon with a Tool already exists, no longer replacing the tool with SYS:Multiview.
- Spell checking support added. Using a plain text word file derived from the iSpell american.med and american.aff files with a few words added for Amiga. An additional, user wordlist file is also read in if found.

If spell checking is used the word list file is completely read into memory (takes about 5 secs on Sam 440 or AmigaOne) and accessed in a hash table. Uses up 10 to 20 MB of memory, but is very fast.

*ISpell* itself is not used nor is any code from it.

-Added option to have the program automatically replace any tab characters encountered in the guide file with the @`{tab}` attribute.

-Iconify/Uniconify support added.

-ENV variable "GuideMaker/GMSwitches" added.  
If found, used with the four settings MAKEICONS, AUTOPREVIEW, USETABATTR, and SPELLCHECK to set or clear these on startup. If GetEnv Guidemaker/GMSwitchs is "MAKEICONS,SPELLCHECK" then those two will be set on upon startup and the other two will be set off.

-Command line template is now  
"FILE,PUBSCREEN/K,MAKEICONS/S,AUTOPREVIEW/S, USETABATTR/S,SPELLCHECK/S"

-Added a couple of keystroke menu commands RAmiga-s for "Save" and RAmiga-q for "Quit".

#### **v0.8**

<not released>

#### **v0.7** (5.11.2009)

-Erratic bug that sometimes caused problems when local or global header commands were deleted should be gone now.

#### **v0.6** (15.10.2009)

-Better match between the *GuideMaker* preview and *Multiview* 's AG display when there is no explicit global "@font" specification in the AG file. *GuideMaker* now also uses the Prefs System Default Font in this case.

-Now if the program is started with a PUBSCREEN <psname> switch or tooltype and no Prefs public screen with that name exists, it will open on the default public screen (Workbench).

-The distributed tool icon had an active PUBSCREEN tooltype used for testing that is now inactivated by parentheses.

#### **v0.5** (16.9.2009)

-"Make Icons" and "Auto Preview" options were added.

-TextEditor gadget now has a scroller.

-If a block is selected in the TextEditor Gadget, inserting one of the character attribute commands (b,i,u) will automatically bracket the block with the appropriate undo attribute.

-Implemented "Export as RTF" option

**v0.4** (27.4.2008)

- Again, changed the way I am doing the AmigaGuide Preview. Even using Multiview, it was unstable. Now using the AmigaGuide datatype directly into my own window.
- Even with fixed size fonts, TextEditor Gadget 52.1 (July '07) may freeze the AOS input on clicking along the left border. Probably won't happen if you have few processes running, however.
- Now loading the editor window with the node contents on a single-click in the node list window rather than requiring a double click.
- Fixed LinkOp entry so that a blank target or text is not accepted.
- Appears that Multiview does not like cycles in the prev/next navigation graph. I recommend against it.

(2.3.2008) - Rebuilt after making some fixes in the compiler.

**V0.3** (23.2.2008)

- Changed the way I was doing the AmigaGuide Preview. Could never get the use of AmigaLibrary OpenAmigaGuideAsync() / SendAmigaGuideCmd() / CloseAmigaGuideAsync() to be stable. Instead, now I am having the program startup Multiview and using its ARexx interface to open the temporary preview .guide files.
- Neatened up the Menus a bit.

**V0.2** (15.1.2008)

- first released version, PPC Native. > [Preview Unstable](#)

**v0.1** (5.12.2007)

- first PPC Native version, never released

**v0.0** (31.12.2006)

- M68K version, never released

Tom Breeden  
tmb\_free\_cat@runbox.com

Aglet Software  
PO Box 99  
Free Union, Virginia, 22940 USA

Please feel free to send bug reports, suggestions, poetry, songs, etc.

The **GuideMaker** "dictionary.wordlist" file is derived from the american medium list supplied with the *ISpell* (ver 3.1.20) package, and as such it is covered by the same copyright and license as *ISpell* .

*ISpell* can be found at <http://fmg-www.cs.ucla.edu/geoff/ispell.html>.

This word list are under the same copyright as *ISpell* itself:

----- Copyright 1993, Geoff Kuenning, Granada Hills, CA

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All modifications to the source code must be clearly marked as such. Binary redistributions based on modified source code must be clearly marked as modified versions in the documentation and/or other materials provided with the distribution.
4. All advertising materials mentioning features or use of this software must display the following acknowledgment:  
This product includes software developed by Geoff Kuenning and other unpaid contributors.
5. The name of Geoff Kuenning may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY GEOFF KUENNING AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL GEOFF KUENNING OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.