



GE Smallworld Emacs 25.3~29.3 (64-bit) Extensions

That You too might find useful

This work is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License or any later version. This work is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See version 3 of the GNU General Public License for more details.

Table of Contents

Creating complete directory structure.....	5
Visual Bookmarks.....	5
Create & Save Frames Layout.....	6
New Use of Scratch Buffer	7
Java Development Environment for Emacs (JDEE)	7
MiniMap MOde	8
Two simply useful new features:	8
Intelligent AutoComplete	9
Custom Toolbar and Tab-bar	9
Save/Restore editing Sessions	9
Dynamic Abbreviation.....	10
Dynamic Abbreviation with Expansion of Multiple choices	10
Predictive Abbreviation.....	10
Automatic insertion of the corresponding {{ and “	11
Highlight text matching PATTERN	12
Open Recent Files	12
Snippets for Magik Coding	13
Dedicated mode.....	13
Class Browser F3-j / F3-b	14
TEMP.magik by F4-m / F12-m	14
ECB – Emacs Code Browser	14
Integration with an External Editor.....	15
Magik Smeller	15
Print in colour or black&white.....	16
Outline mode for Magik files.....	16
Show/hide Line numbers	16
Visual Bookmarks.....	17
Current Method name	18
[Index] menu	18

Additional highlighting during the incremental search.....	18
Speedbar (F11).....	19
Magik code folding.....	20
Magik Language Reference	21
Save/Restore Emacs Sessions.....	22
List of key-bindings	23
Key bindings for Magik mode	27
Key bindings for other modes	28

“Emacs is not an editor. Emacs is a way of thinking about the world, and as such is a way of thinking about editors. When you ask what Emacs does, you are asking a question with no answer, because Emacs doesn't do, it is done to. Emacs just is ... I hope this makes things clearer.” -

Scott Dorsey



Why is it called – Emacs?

The name 'Emacs' was originally chosen as an abbreviation of Editor **MAC**roS and was created by Free Software Foundation.

This document includes highlights of some new and existing features and fine points in this customised version of Emacs 29.1

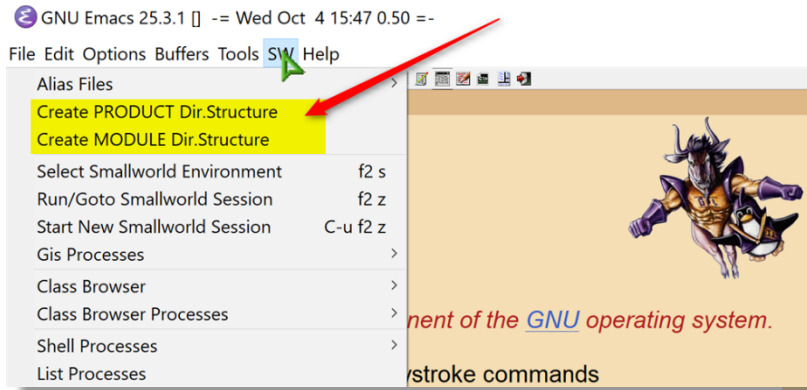
This customised version based on the GNU Emacs 29.3 (x86_64-w64-mingw32) of 2020-08-21 and Smallworld additions to the Emacs release with GE Smallworld 4.3.

Some screenshots used from Emacs 25.3.1

All comments and suggestions please forward to <igor@hydepark-consulting.com> . You can download this Emacs from HydePark-Consulting.com/Emacs and view explanations of added features and Emacs useful tips on MagikEmacs.com

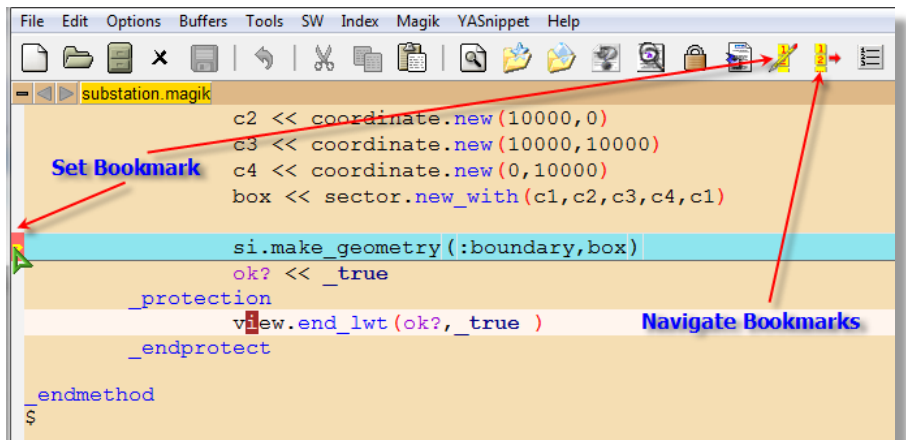
Creating complete directory structure

Creating complete directory structure with few default files (*product.def, module.def and load_list.txt*) for a PRODUCT or MODULE using Emacs menu options. See examples for TEST_PRODUCT & test_module directories:



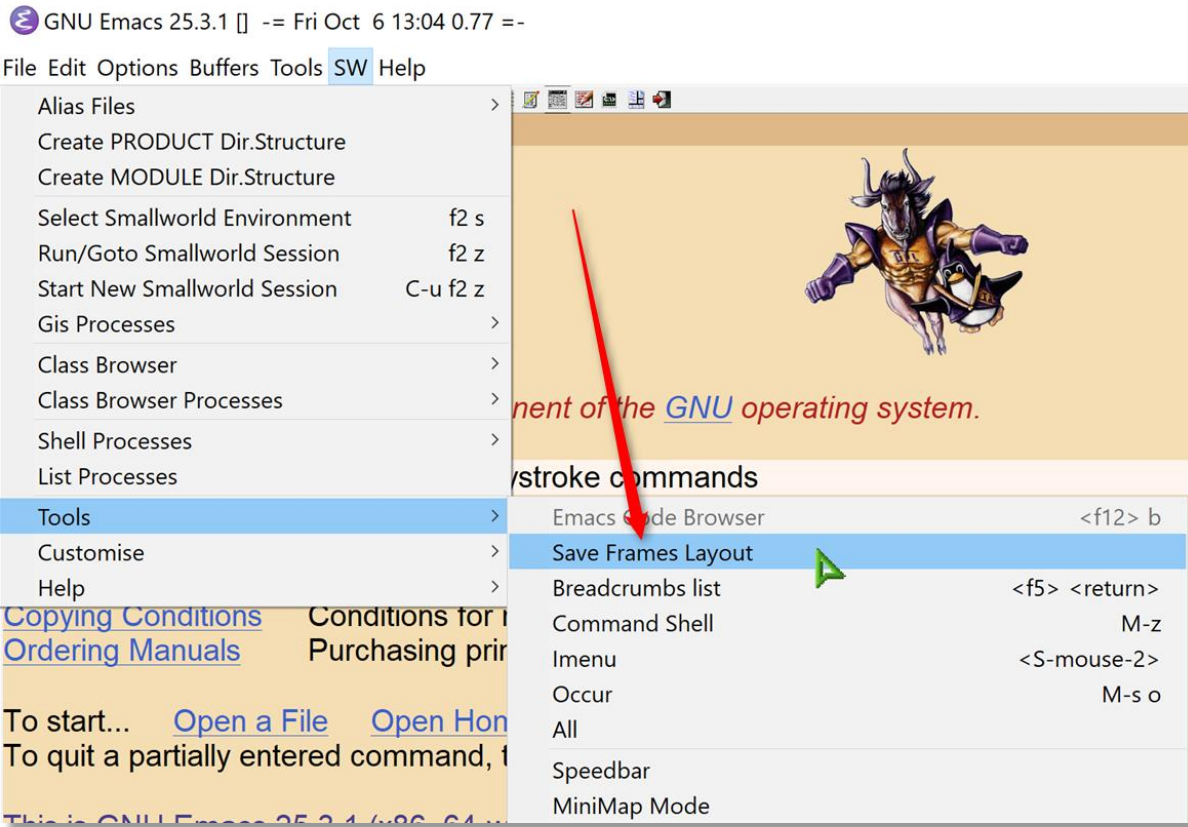
Visual Bookmarks

- **Visual Bookmarks** provides easily visible bookmarks, which can be set/cleared via **CTRL-F5** and Toolbar icon OR simply by *clicking on a LEFT fringe*. Navigation between bookmarks F5-n/F5-p or using the Toolbar icon. List bookmarks - use Alt-x 'bm-show' (current buffer) and Alt-x 'bm-show-all' (all open buffers).



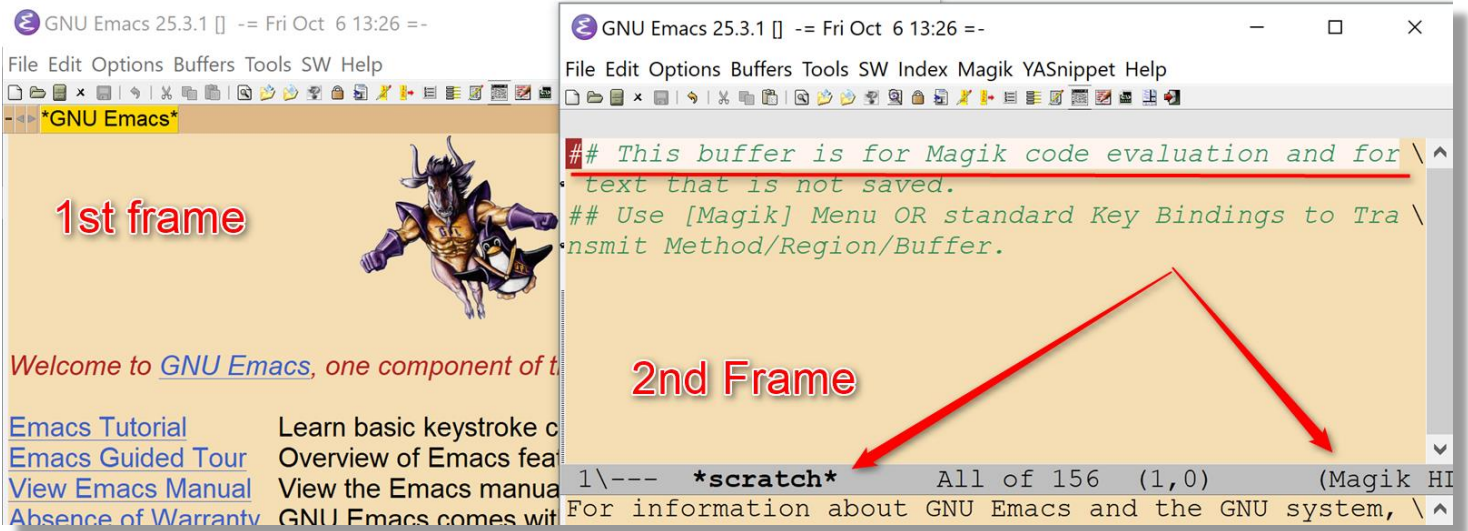
Create & Save Frames Layout

- **Save Frames Layout** – allows to save the current Emacs Frames layout (number of Frames and positions on a screen) and restore saved Frames configuration on Emacs startup.



EXAMPLE: I always use 2(Two) Emacs Frames – 1st Frame split horizontally (Ctrl-x-2) with *gis* buffer (locked/dedicated mode) and Class Browser, 2nd Frame using for coding. So, I use this feature to save the Frames Layout and restore on the startup. Use **<Alt-f>** to switch between frames.

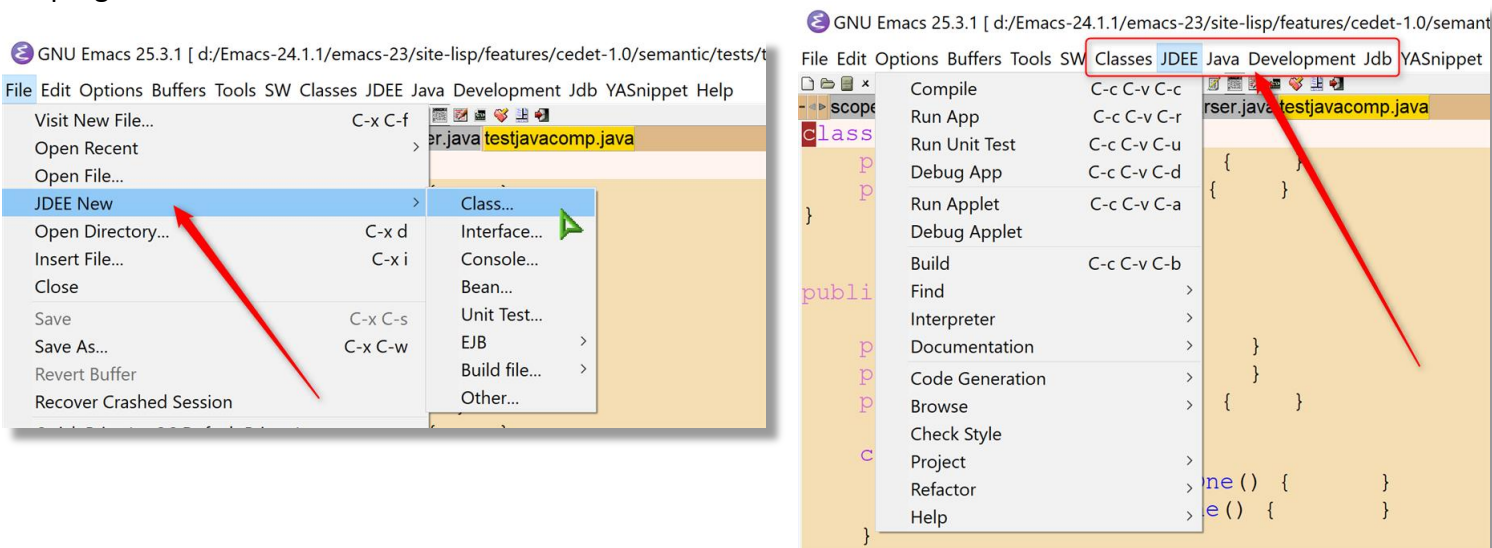
New Use of Scratch Buffer



NOTE: another NEW feature - now the ***scratch*** buffer can be used for Magik code evaluation (see above)

Java Development Environment for Emacs (JDEE)

- The **Java Development Environment for Emacs (JDEE)** is an add on software package for Emacs that assists in Java programming. JDEE provides ability editing, compiling, running, debugging, and browsing large Java programs.

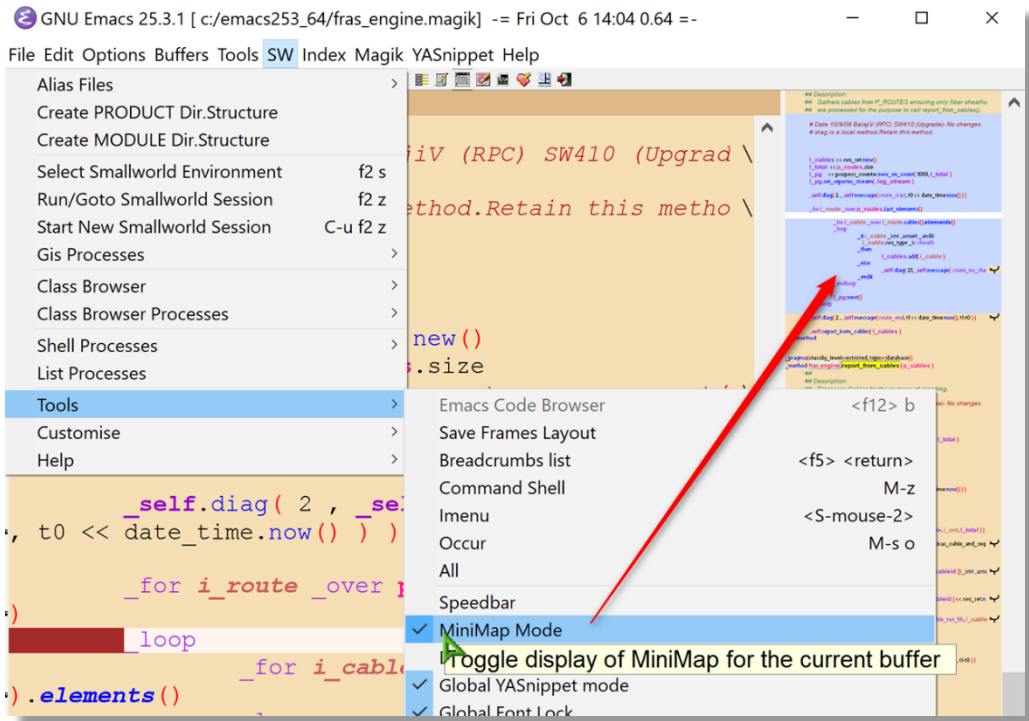


Before you can use the JDEE you must tell it where the Java compiler, debugger, and virtual machine are located on your system. Refer to < "...\emacs293\share\emacs\29.3\site-lisp\features\jdee-2.4.2\Emacs JDE User's Guide.html"> on how to select and register JDK. Or simply edit these values in the "...\emacs-custom":

```
'(jdee-jdk (quote ("11.0")))  
'(jdee-jdk-registry (quote (("11.0" . "C:\\Java\\jdk-11"))))
```

MiniMap MMode

- **MINIMAP mode** - smaller display of the current buffer on the left side. It highlights the currently shown region and updates its position automatically. You can navigate in the minibar by dragging the active region with the mouse, which will scroll the corresponding edit buffer.



Two simply useful new features:

- Use **Ctrl-Shft-d** to **duplicate** current line of code
- Use **Ctrl-Alt-Up/Down** arrow to **move current line** or selected **block of text** Up and Down

Watch MagikEmacs.com for more explanations on custom features of Emacs IDE for GE Smallworld.

Intelligent AutoComplete (thank you to Andrew Maguire, GE), which understands classes, methods(functions) and even `<_self>` - offering only methods defined on “self” class :

```

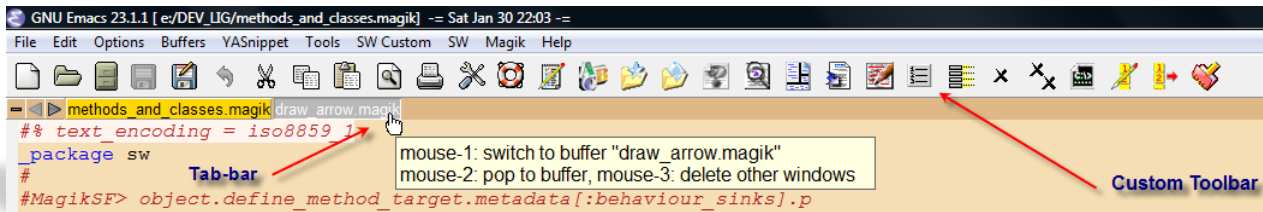
_pragma (classify_level=restricted)
_method substitution.get_spatial_context_names ()
##
## Returns extra spatial context names to add to the
## application defined spatial contexts.
##
>> { :internals }

_endmethod
$
_pragma (classify_level=debug)
_method substitution.draw_internals (window, my_geometry, rwo_style, draw_flag?)
##
## A draw method which renders the substation's internal world
## within the bounds of its geographic location geometry.
##
_dynamic !current_rendering_context!

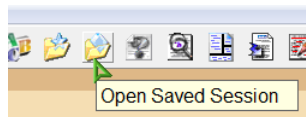
_self.get_spatial_context_names ()
geom get_spatial_context
ace_get_spatial_context() f get_spatial_context_names()
an_s get_spatial_context_names | RESTRICTED!
get_spatial_context_names() f Returns extra spatial context names to add to the
_if get_spatial_context_element_for_world() f application defined spatial contexts.
( gsf << rc.geometry_set_factory ) _isnt _unset
_then
display_scale_name << :auto
an_sts << gsf.get_style_system_for( ace_name, display_scale_name )
_endif

```

Custom Toolbar and Tab-bar (use `<Ctrl-Tab>` to navigate between tabs).



Save/Restore editing Sessions, which allows you with one click of a button to **load multiple files** from a previous closed editing session. Even a cursor positions in the open files will be preserved from the previous session.



Dynamic Abbreviation - it is a standard function in the Emacs, but, from my experience, not everybody knows about. It is allowing you to write just a few characters of words you've written earlier to be able to expand them. Possible completions searched in the text in the current buffer, and if not found there, in all open buffers. To expand a word, just put the point right after the word and press **<Alt-/>**. As soon as you start typing – you are able to use this feature. Typing the next letter will automatically narrow your possible completions.

```
# Push it out to the owning frame
# respond to.
_self.framework.databus.make_data

_self.app
endmethod
```

```
# Push it out to the owning fra
# respond to.
_self.framework.databus.make_da

_self.application
endmethod
completed
```

Dynamic Abbreviation with Expansion of Multiple choices – press the same key second time and you will get multi-choice candidates in a tooltip.

```
# Push it out to the owning frame
# respond to.
_self.framework.databus.make_data

_self.app
endmethod
```

- (a): app_dev_examples
- (s): appear
- (d): applies

Predictive Abbreviation. It's fairly similar to Dynamic Abbreviation expansion, which works based on the contents of the current buffer (or other buffers). Predictive abbreviation expansion works based on the previously written text. Unlike Dynamic Abbreviation, the text is analysed during idle time, while Emacs is doing nothing else. **As you type you will be offered possible expansion for the word at the left from the cursor position.** Typing new characters will automatically offer different completion. If the correct completion is offered – just press **<Tab>** to complete the word. Light blue colour indicates that there are possible different candidates for the expansion. To see other candidates – press **<Tab>** once again and choice of possible completions will be shown. If the completion is shown in the dark blue colour – that's mean this is the only possible candidate.

```
# Determine which data
# should do with the da
_self.d|ata|
```

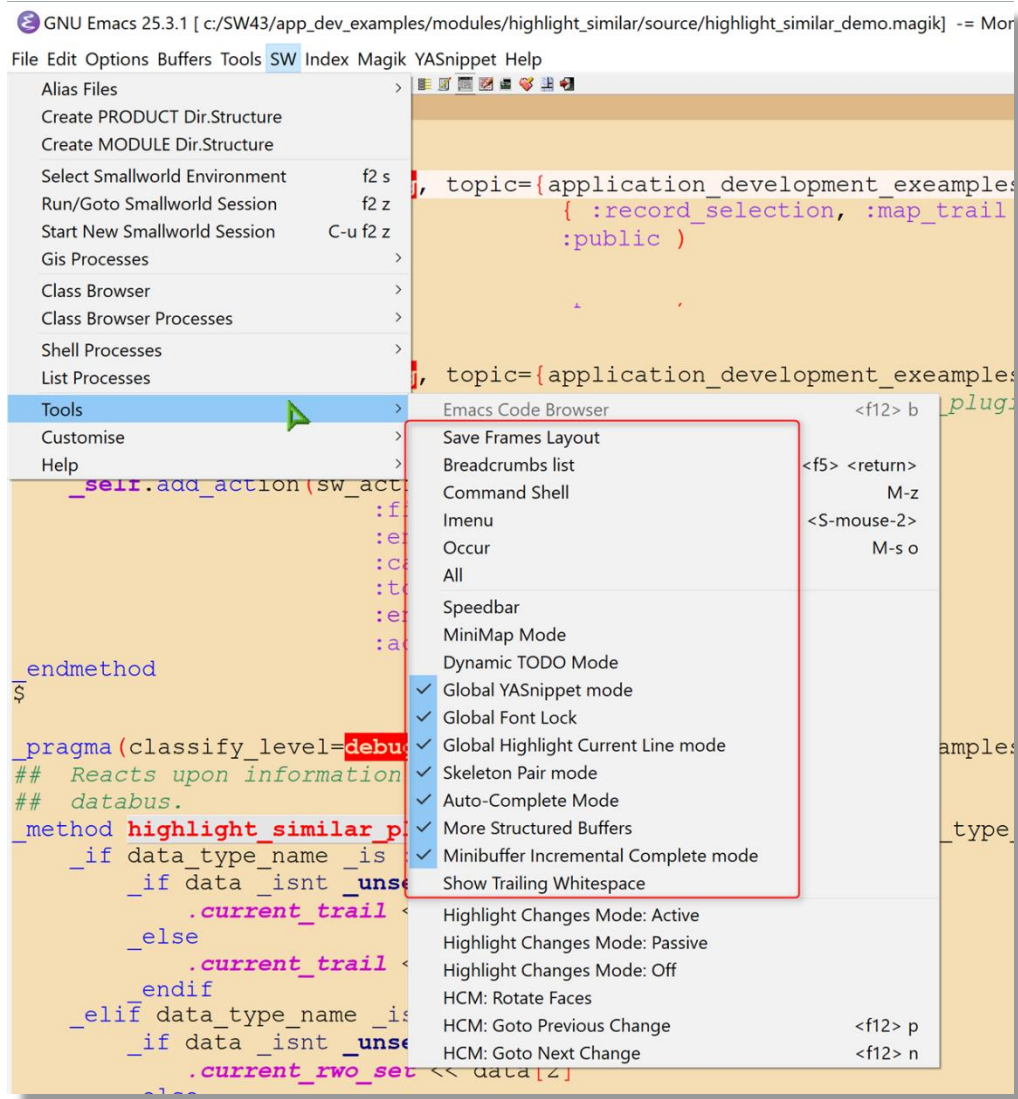
```
# Determine which data
# should do with the da
_self.data|bus|
```

```
# Determine which data types we a
# should do with the data that is
_self.databusdatabus 1
data_type_name 2
database 3
```

```
# Determine which data t
# should do with the dat
_self.data_|type_name|
```

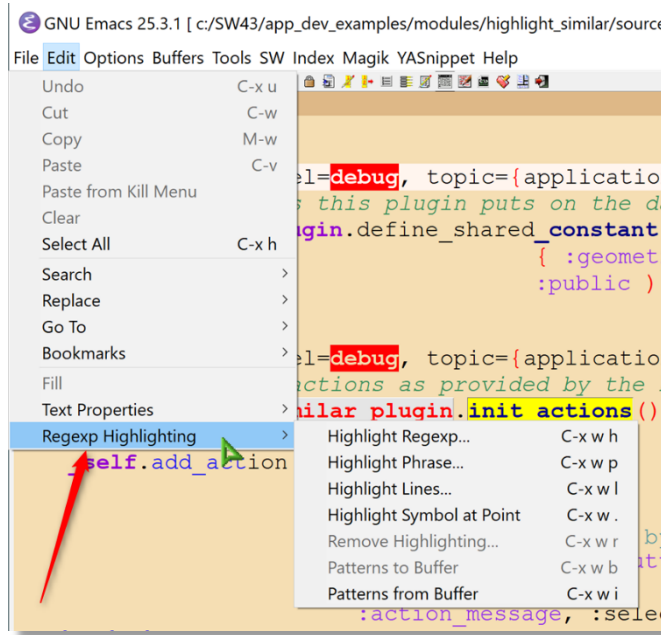
Automatic insertion of the corresponding `{` and `"`. Thus typing ``('` will normally insert ``(')` and put the cursor between them. Otherwise, when the `region` is active, it will be wrapped in the parentheses.

Use SW->Tools menu to activate/deactivate many useful options.



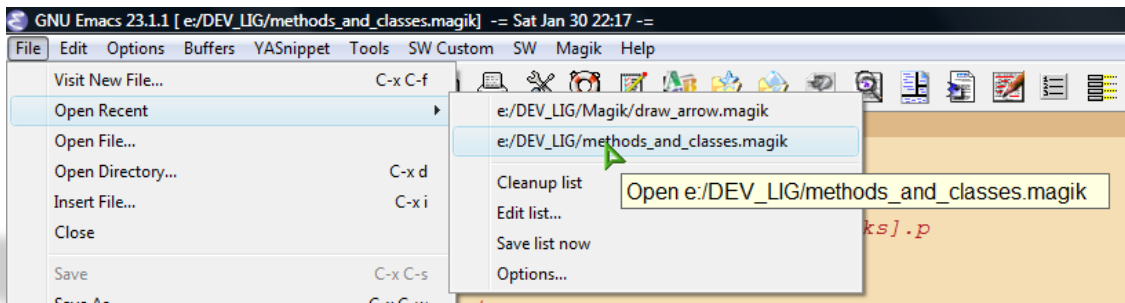
Highlight text matching PATTERN.

Examples: In a source code highlight a variable to quickly see all places it is modified or referenced. New occurrences of that variable will be highlighted as they are typed. In a *gis* or another buffer that is showing lots of output, highlight the parts of the output you're interested in. Those patterns will be highlighted as soon as they appear in the buffer. Useful if you have a lot of output during debugging. This option is located on the **Edit menu**.



Open Recent Files

- Additional menu item for **visiting files** that were operated on **recently**

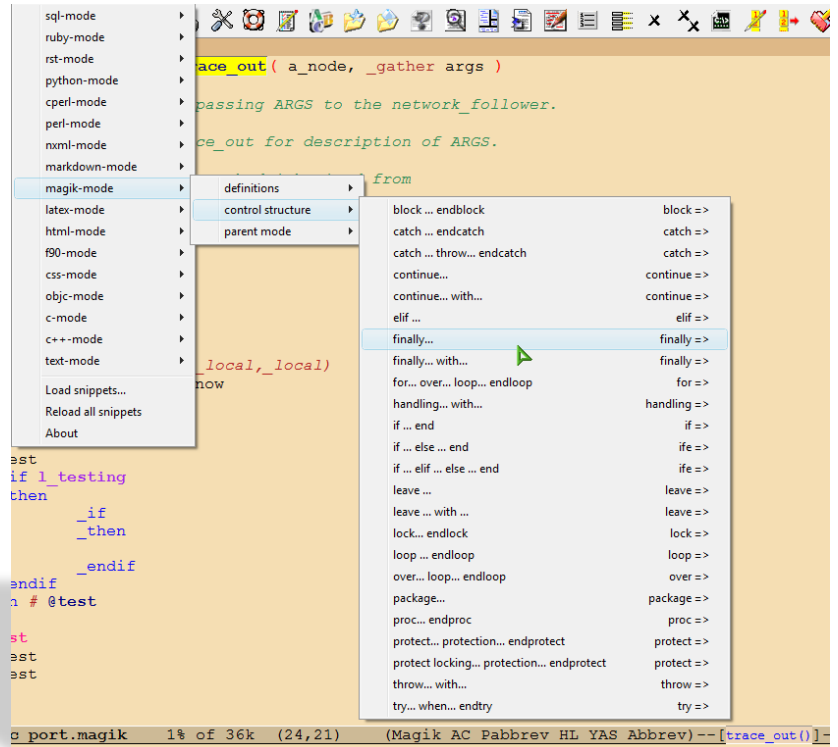


Snippets for Magik Coding


It allows you to type an abbreviation and automatically expand it into function templates (similar to electric-magik). Navigation between snippet fields – using **<Tab>**. Templates will expand automatically as you type them and they are available via menu too.

```
_then >> _true
_else >> _false
_endif
_for <lvalue,tuple>_over <iter invocation>
_loop
_endloop
```

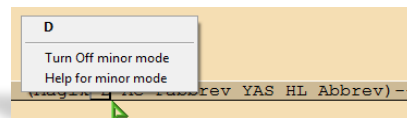
```
example_framework magik **TEMP** magik
_method example_framework, example_test_method ()
##
##
_for i_el_over _self.fast_elements ()
_loop
_if i_el_is_unset
_then
_leave
_endif
_endloop
_endmethod
$
```



Dedicated mode

- when a  buffer is "dedicated" or locked, Emacs will not select files into that buffer. Dedicated buffers will have "D" shown in the mode line and marked lock icon. I found it very useful to lock the *gis* buffer. NOTE: All minor modes are shown in the mode line and you can use the mouse to switch it OFF or get Help on the specific mode.

```
(Magik [D] AC [P]abbrev YAS HL [A]bbrev)--[trace_out()]--
```

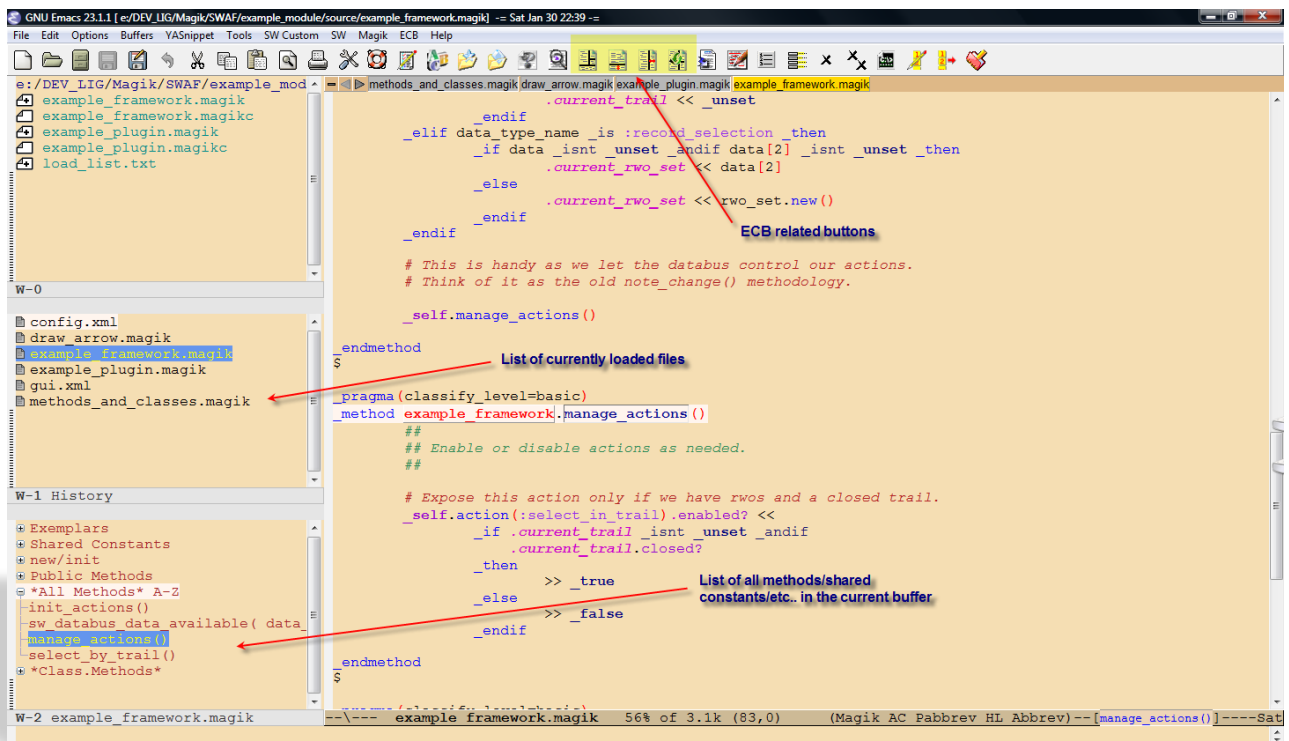


Class Browser F3-j / F3-b – allows you to go back in the File Navigation history, so you are able to return to the point in a code where you came from during your F3-j key journey.

TEMP.magik by F4-m / F12-m – copy current magik method to the Work (**Temp** .magik) buffer. Very helpful while making temporary changes in the code or just testing/debugging.

- **F2-b** – to compile **message files <*.msg>** directly from Emacs.

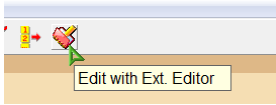
ECB – Emacs Code Browser which displays a number of informational windows that allow for easy source code navigation and overview.



Integration with an External Editor

This will allow you to load the file from the Emacs, if necessary, into an external editor of your choice to do some additional editing/searching or other file manipulations.

Use "*C:\Emacs293\NP++Magik_Language.XML*" to import Magik highlighting into Notepad++

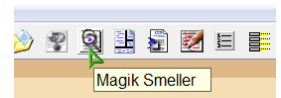


```
is\modules\highlight_similar\source\highlight_similar_demo.magik - Notepad++
ng Language Settings Tools Macro Run Plugins Window ?
highlight_similar_demo.magik
49  _pragma(classify_level=debug, topic={application_development_exeamples}, usa
50  ## Reacts upon information being available on the framework's
51  ## databus.
52  _method highlight_similar_plugin.sw_databus_data_available( data_type_name, (
53  _if data_type_name _is :map_trail _then
54  _if data _isnt _unset _then
55  .current_trail << data[2]
56  _else
57  .current_trail << _unset
58  _endif
59  _elif data_type_name _is :record_selection _then
60  _if data _isnt _unset _andif data[2] _isnt _unset _then
61  .current_rwo_set << data[2]
62  _else
63  .current_rwo_set << rwo_set.new()
64  _endif
```

Magik Smeller

which helps you to find possible problems in the code. A tool to interactively report actual or possible code syntax problems that are acceptable to the Magik compiler but do not meet development standards or may be considered inefficient.

A new buffer called ***Magik Smeller*** will appear showing a list of all possible code "smells", their line number in the Magik file and a short description. To jump back to a line of code that has been reported as a smell, place your cursor on that line in the ***Magik Smeller*** and press "F3-j".



```
*Magik Smeller*
Final Smell Score: 45
Smell per line: 0.381356

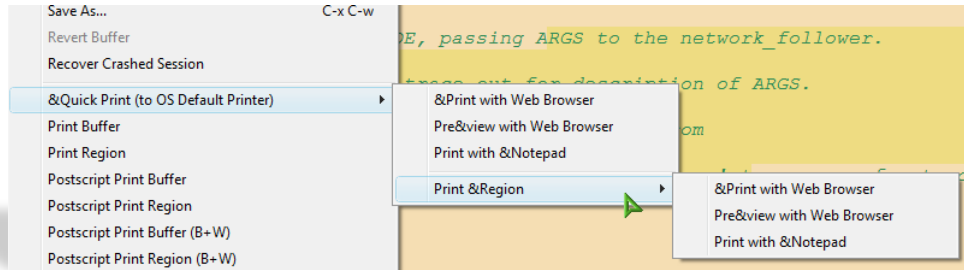
A Magik Smell is a piece of code that points to a possible coding problem.
To jump to the code in the original file that a line number is referring to, place your cursor

(5) TEXT ENCODING
=====
a Magik file MUST have a text_encoding declaration as its first line. For example:
## text_encoding = iso8859_1
Line 1:

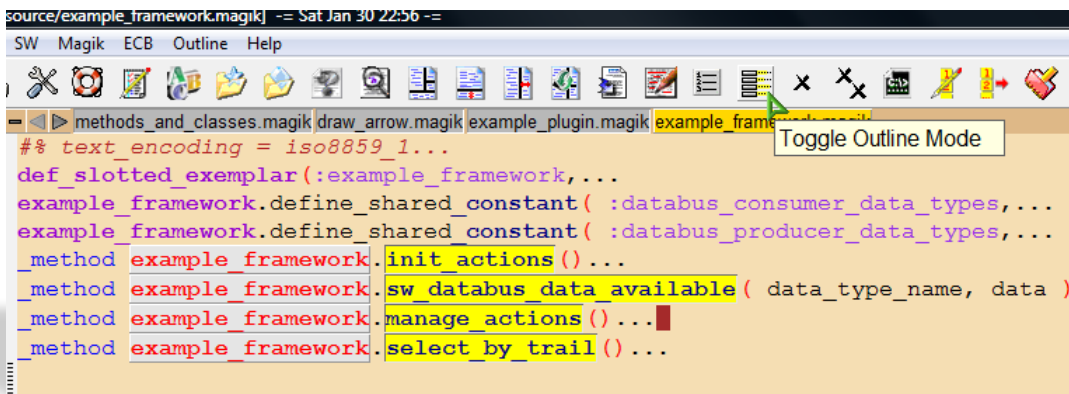
(5) PACKAGE DECLARATION
=====
a Magik file MUST have a _package declaration at the top (usually the second line). For exam
_package sw
Line 1:

-1\*- *Magik Smeller* Top of 1.7k (1,0) (Magik Smeller HL Abbrev)---Sun Feb 7 21:58-
```

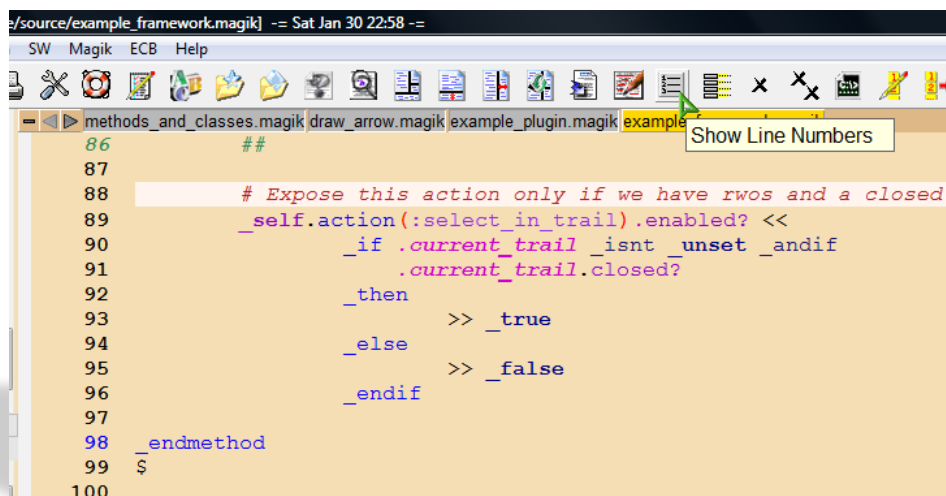

Print in colour or black&white directly from the Emacs. Preview in colour – will create an HTML page with current font locking highlights in the code.



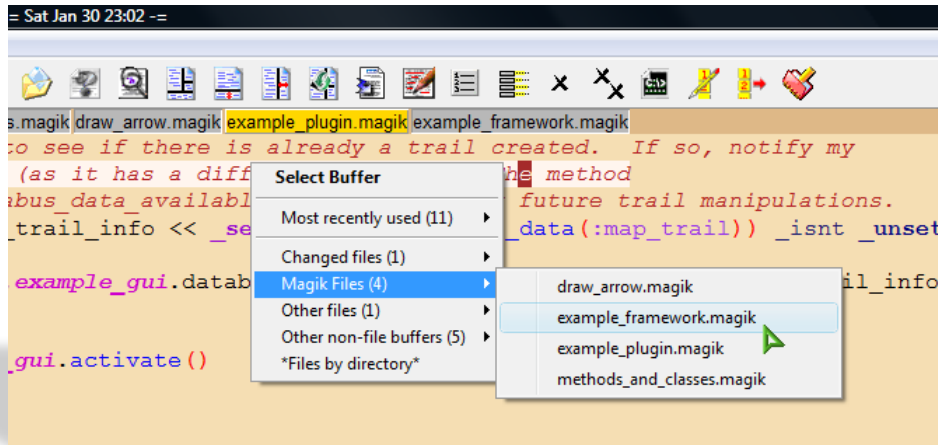
Outline mode for Magik files. Using keys **F3+Up/Down** in the Outline mode the Magik file can be collapsed to display only the method names and methods can be unfolded one by one for editing.



Show/hide Line numbers in a buffer.



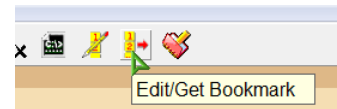
- Buffer selection- **<Ctrl>+<left mouse button>** will bring up the floating menu at the cursor position OR use **<Ctrl>+<Tab>** to navigate between tabs in the Tab-bar.



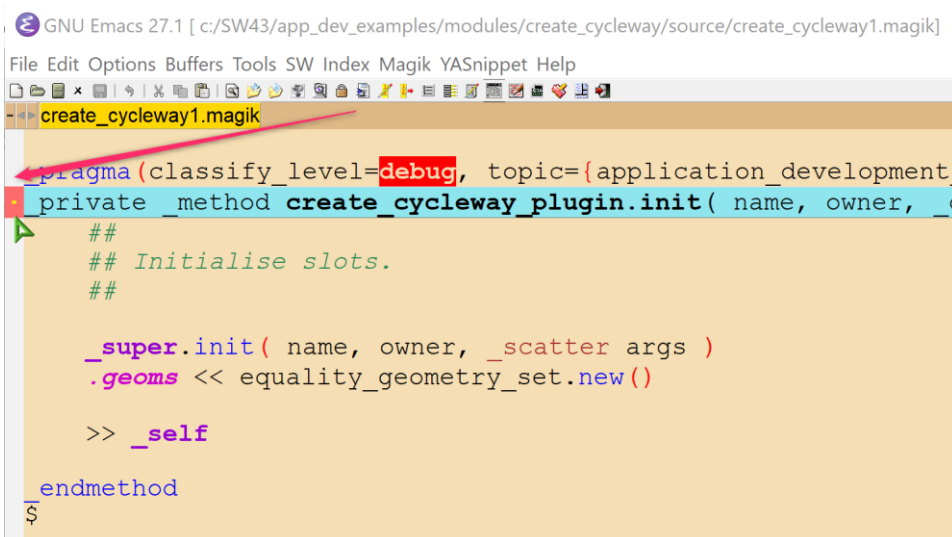
- Save all file **backups** (*.magik~) in the single configurable directory (system %TEMP% – by default).

Use **'breadcrumbs'** **<F5-Spc>** - to set series of **Breadcrumb Bookmarks** with quick cycling through them using **<F5-Up/Down/Left/Right>**.

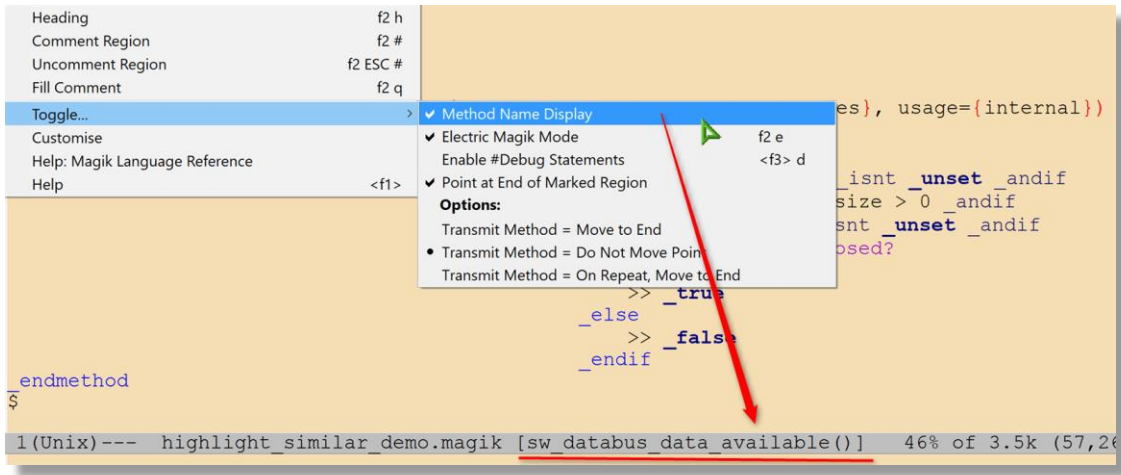
Visual Bookmarks – use toolbar buttons to Set and Navigate between bokmarks



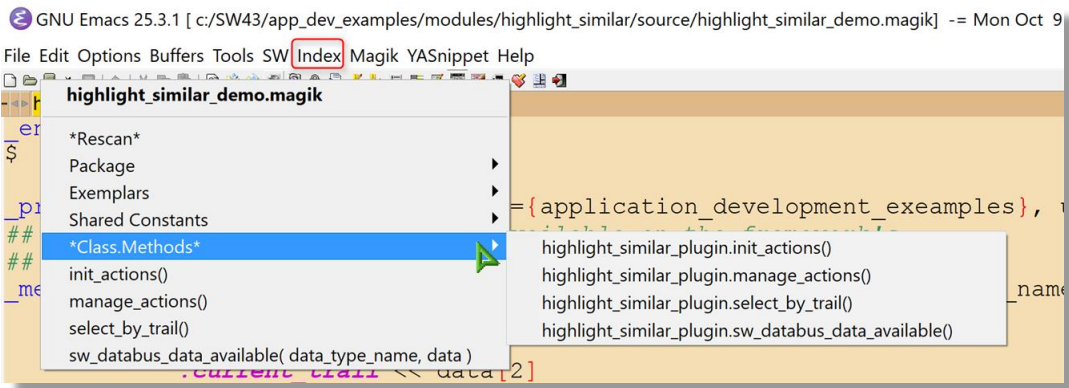
Or simply click on the left fringe to set/remove a bookmark



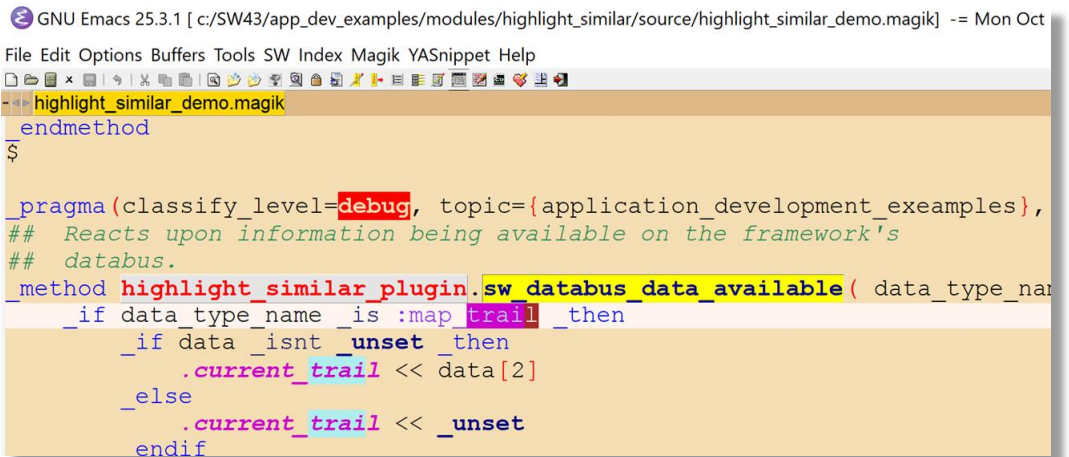
Current Method name in the mode line can be toggled from the [Magik] menu

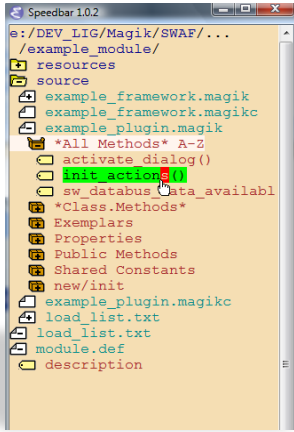


[Index] menu allowing quickly navigate between methods or use <Ctrl+RightClick>



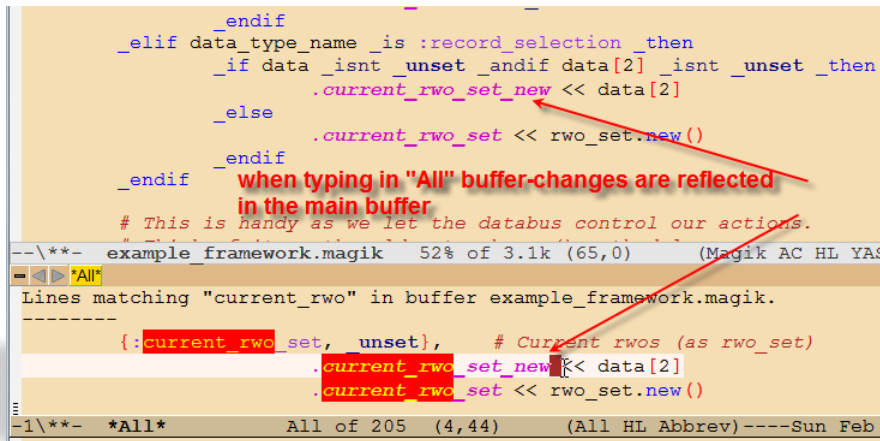
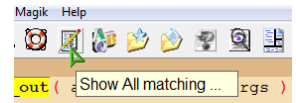
Additional highlighting during the incremental search



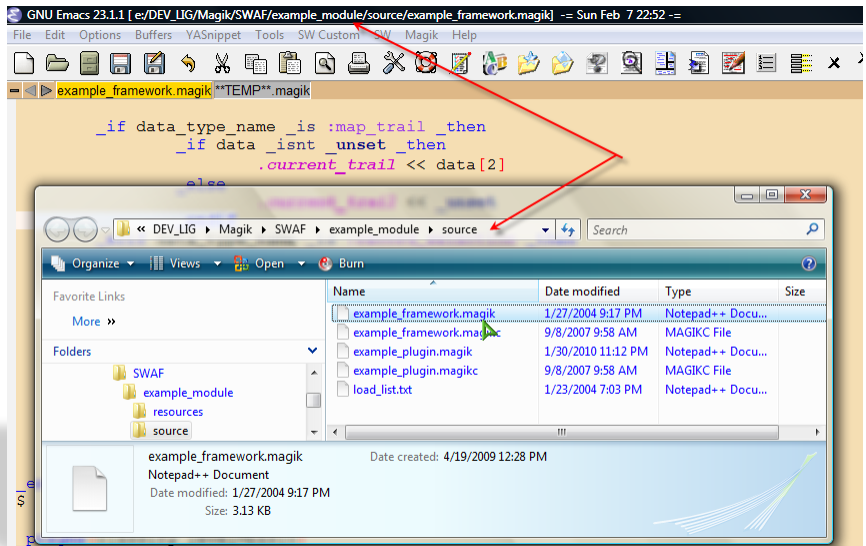


Speedbar (F11) - a special skinny frame with a specialized directory listing in it. This listing will have both directories and filtered files in it. You can then load files into your Emacs buffer, or expand the files to display all the methods/progs/constants/.... in them and jump to those tags. You can also expand multiple directories into your Speedbar frame.

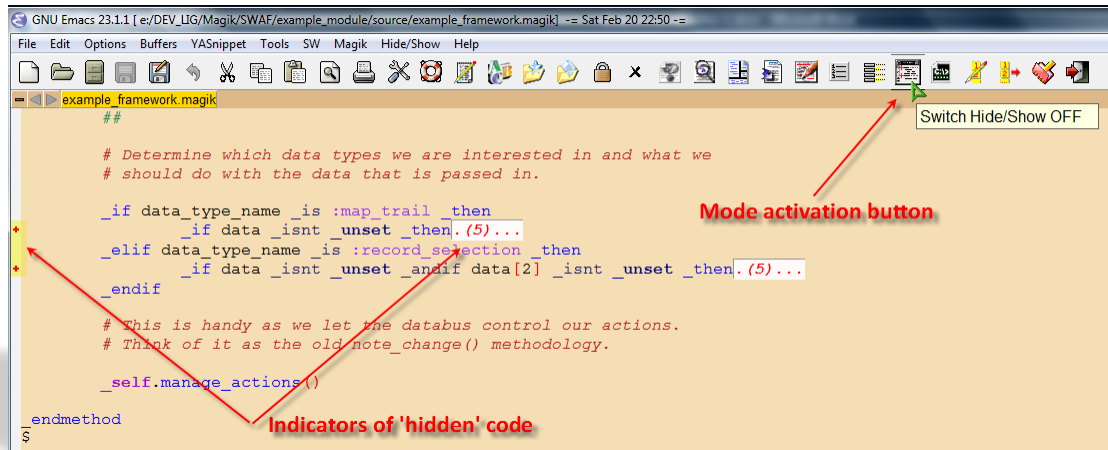
- Use **'Alt-x all <Enter>'** or the icon to **list\EDIT** all lines matching a given regexp in ***All* buffer**. **Changes made in the *All* buffer is propagated to the original buffer**. Use **'F3-j'** from *All* buffer to go to a match in the original file. **Note that changes to the '*All*' buffer are propagated back to the original buffer.**



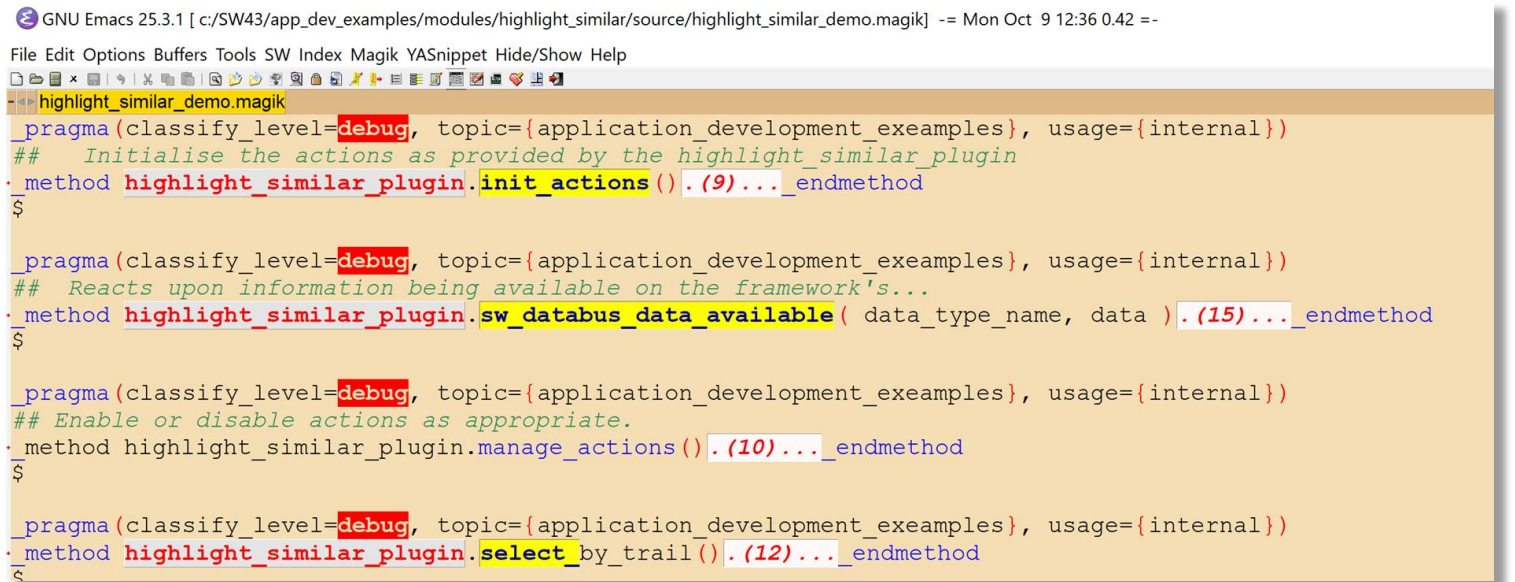
- While you have a file loaded into the Emacs buffer pressing **<F12-F12>** will open **Windows Explorer** in the directory of the current file.



Magik code folding – where large blocks of code can be distracting. With the code folding feature you can hide the code not important to your current work. Code folding cleans your workspace. Use **<Ctrl-,>** key to **hide the block of code** while your cursor is inside the desired block.

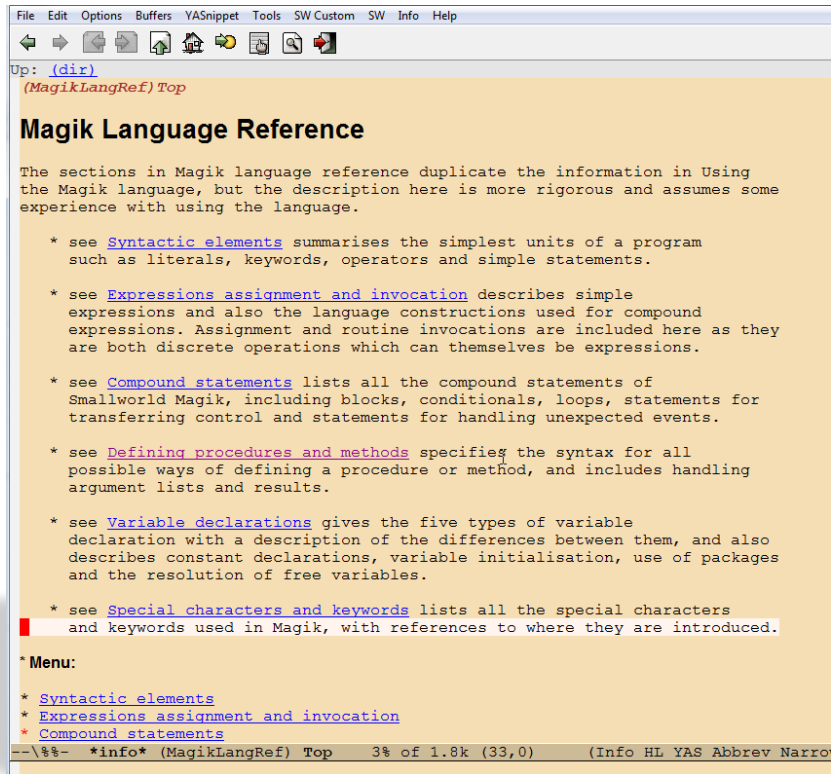


- Use <Ctrl-,> key to hide/show code on the 'method' level



Magik Language Reference

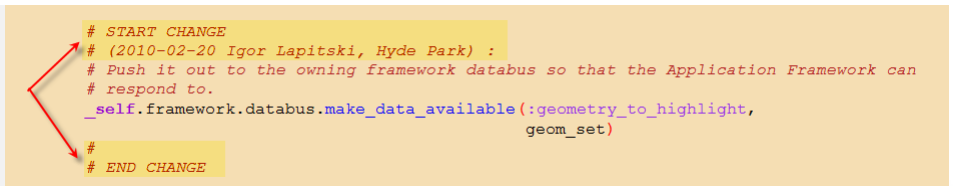
available within Emacs.



- Use `%USER_EMACS%` environment variable in `gis_aliases` to load your local Emacs settings, like
(setq programmer-tag "User Name, Company Name")
(setq project-name "Project Name")
(setq programmer-time-stamp-format "%:y-%02m-%02d"); %02H:%02M:%02S")

- `[Alt-n]` to insert-programmer-note;
`[Alt-m]` to insert-programmer-modified-note;

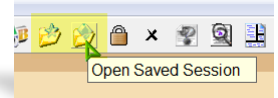
`[Alt-c]` to insert-change-note around selected block – just select the block of code and press `<Alt-c>` and note change comments will be placed around the selected code; OR just use this to create a note change.



- `[Alt-h]` to insert-method-comment;

Save/Restore Emacs Sessions

That allows you to organize your different projects into "sessions". Each Session has a name (file name). When you 'Save Session', the list of all opened files is saved into the current session and all the files you were editing will be reloaded the next time you 'Restore Session'. This allows you to keep different files open in different Emacs sessions. By using 'Save/Restore Session' you are activating 'Session auto-saving mechanism' - on Emacs exit current Session will be saved automatically.



- To make prominent marks in '*gis*' buffer or comments in 'Magik' buffer use '@' at the beginning of the line in '*gis*' buffer and '#@' in 'Magik' buffer. I found this useful when you have a lots of output in the *gis* buffer and you need to see specific line – just use this in your code `write("@Important info: ", a_variable)` and you will be able to see this line in the output very clearly.

```
# Determine which data types we are interested in
# should do with the data that is passed in.
#@ This is the comment that I want to see
#@ Review
_if data_type_name _is :map_trail _then
  _if data _isnt _unset _then
    .current_trail << data[2]
  _else
    .current_trail << _unset
```

- Use **F2-b** to send `<*.msg>` file to running GIS buffer. This will load current *.msg file into an open currently image.

- Emacs will display highlighting on whatever parenthesis matches the one before or after point. See message in Mini-buffer describing the match when the matching parenthesis is off-screen. Message contains the line number.

```
:enabled?, _false,
:action_message, :select_by_trail|()|)
endmethod
--\**-- example framework.magik 39% of 3.2k (47,72) (Magik ns+ AC Pabbrev
Matches self.add action(... [-7-])
```

- Use **Ctrl-TAB** to switch/cycling between Emacs buffers.

- When using `<Ctrl-x/Ctrl-b>` to bring up Buffer Selection Menu – use following keys: `<c>` - to change the configuration; `<s>` - to save buffers marked with '*' ; `<d>` - to delete buffer

MR Buffer	Size	Mode	File
highlight_similar_demo.magik	3549	Magik	c:/SW43/app_dev_examples/mo
% *GNU Emacs*	649	Fundamental	
scratch	156	Magik	
*% *Messages*	3010	Messages	
JDEE bsh	0	Comint	

1\%%- *buffer-selection* All of 570 (3,0) (Buffer-Selection-Menu HL yas Al
Selected configuration: all

List of key-bindings which might be useful in everyday Emacs operations with Smallworld environment:

<p> <code><f2> TAB</code> hippie-expand <code><f2> RET</code> magik-transmit-thing <code><f2> ESC</code> Prefix Command <code><f2> #</code> magik-comment-region <code><f2> X</code> dev-tools-object-inspector <code><f2> [</code> toggle-debug <code><f2> b</code> magik-transmit-buffer <code><f2> e</code> electric-magik-mode <code><f2> g</code> gis <code><f2> k</code> sw-reload-dotemacs <code><f2> m</code> magik-transmit-method <code><f2> q</code> fill-magik-public-comment <code><f2> r</code> magik-transmit-region <code><f2> s</code> gis-version-selection <code><f2> t</code> add-trace-statement above <code><f3> t</code> addtrace-statement below <code><f2> x</code> deep-print variable <code><f2> z</code> gis <code><f2> <f1></code> sw-help-keys <code><f2> <f7></code> magik-transmit-method <code><f2> <f8></code> magik-transmit-region <code><C-S> d</code> duplicate current line <code><C-M> <up/down></code> move current line </p>	<p> <code><f3> /</code> cb-and-clear <code><f3> ?</code> cb-help <code><f3> b</code> cb-jump-back <code><f3> c</code> cb-paste-class <code><f3> d</code> toggle-transmit-magik-debug <code><f3> j</code> cb-jump-to-source <code><f3> m</code> cb-paste-method <code><f3> <f3></code> cb <code><f4> c</code> magik-copy-method <code><f4> d</code> dev-tools-debug-method <code><f4> e</code> magik-ediff-methods <code><f4> m</code> magik-copy-method-to-buffer <code><f4> n</code> magik-set-work-buffer-name <code><f4> o</code> dev-tools-object-inspector <code><f4> r</code> magik-copy-region-to-buffer <code><f4> s</code> magik-add-debug-statement <code><f4> w</code> magik-compare-methods <code><f4> <f4></code> magik-symbol-complete <code><f4> b</code> bury-buffer <code><f4> g</code> goto-line <code><f5> g</code> goto-line <code><f6></code> magik-copy-method <code><f7></code> magik-transmit-method <code><f8></code> magik-transmit-region <code><f9></code> magik-mark-method <code><f11></code> speedbar-get-focus <code><f12> -</code> ecb-cycle-ecb-buffers <code><f12> =</code> ecb-maximize-window-methods <code><f12> b</code> ecb-minor-mode <code><f12> c</code> ecb-toggle-compile-window <code><f12> e</code> send-file-to-external-editor <code><f12> h</code> ecb-toggle-compile-window <code><f12> q</code> ecb-redraw-layout <code><f12> r</code> ecb-restore-window-sizes <code><f12> t</code> ecb-toggle-layout <code><f12> u</code> ecb-rebuild-methods-buffer <code><f12> w</code> ecb-toggle-ecb-windows <code><M-/></code> dabbrev-expand <code><M-d></code> close window <code><M-b></code> move cursor to another buffer <code><M-f></code> move cursor to another frame <code><M-m></code> insert-modified-note </p>
<p> BOOKMARKS / BREADCRUMPS <code><C-f5></code> toggle visual bookmark <code><f5-Spc></code> set breadcrumb bookmark <code><f5> <delete></code> bc-goto-current <code><f5> <down></code> bc-local-next <code><f5> <left></code> bc-previous <code><f5> <return></code> bc-list <code><f5> <right></code> bc-next <code><f5> <up></code> bc-local-previous <code><f5> a</code> bm-toggle-cycle-all-buffers <code><f5> d</code> bookmark-delete <code><f5> e</code> edit-bookmarks <code><f5> l</code> bookmark-load <code><f5> n</code> bm-next <code><f5> p</code> bm-previous <code><f5> s</code> bookmark-set <code><f5> w</code> bookmark-write </p>	

<f12> <f12>	w32shell-explorer-here	<M-n>	insert-programmer-note
<f12> m	magik-copy-method-to-buffer	<M-c>	insert-change-note
<f12> n	highlight-next-change	<M-h>	insert-method-comment
<f12> p	highlight-previous-change	<C-Tab>	toggle between files
<f12> e	send-file-to-external-editor		

Many of those settings are possible to customise, use the menu **SW->Customose->Customise Smallworld Emacs.**

The background colour of the custom version of the Emacs (wheat) also possible to change. All custom settings will be saved in the **<.emacs-custom>** file in your %HOME% directory. See http://www.hydepark-consulting.com/Emacs/Emacs_colours.pdf for all possible colours.

```

GNU Emacs 23.1.1 [] -- Mon Feb 15 00:07 --
File Edit Options Buffers YASnippet Tools SW Custom Help

"Customize Group: Smallworld" "Customize Group: Misc Custom"
Operate on all settings in this buffer that are not marked HIDDEN:
Set for current session Save for future sessions
Undo edits Reset to saved Erase customizations Exit

Parent groups: Emacs
Parent group documentation: Manual.

/- Smallworld group: Smallworld Development group-----\
State: visible group members are all at standard values.

Menu Sw Tools Submenu:
State: HIDDEN, invoke "Show" in the previous line to show.
List of Menu entries for SW->Tools submenu.

Module Option Save Magik: y Toggle on (non-nil)
State: STANDARD.
If t, save .magikc files when loading module.

Module Option Force Reload: y Toggle on (non-nil)
State: STANDARD.
If t, save .magikc files when loading module.

Resources : Group for Smallworld Emacs Message internationalisation.
Aliases : Customise Magik aliases files group.
Magik : Customise Magik Language group.
Class Browser : Running Magik Class Browser.
Gis : Running Smallworld GIS.
Misc Custom : LIG Custom Emacs User Setting
Product : Customise Magik product.def files group.

-1\*- *Customize Group: Smallworld* 13% of 1.8k (40,0) (Custom YAS H

```

```

GNU Emacs 23.1.1 [] -- Mon Feb 15 00:09 --
File Edit Options Buffers YASnippet Tools SW Custom Help

"Customize Group: Smallworld" "Customize Group: Misc Custom"
ECB load on Emacs start up?

Pair Mode: y Value Menu OFF
State: STANDARD.
Pairing mode on Emacs start up?

Toggle Pabbrev Mode: y Value Menu ON
State: STANDARD.
Predictive Abbreviation on Emacs start up?

Toggle Ac Auto Start: y Value Menu Require: 3
State: SAVED and set.
Non-nil means completion will be started automatically.

Initial Frame Width: y 135
State: STANDARD.
WIDTH of the Initial Frame

Initial Frame Height: y 43
State: STANDARD.
HEIGHT of the Initial Frame

Custom Background Color: y wheat
State: STANDARD.
Emacs background color

Custom Modeline Background Color: y wheat3
State: STANDARD.
Emacs modeline background color

Custom Foreground Color: y black
State: STANDARD.
Emacs foreground color

Custom Backup Directory: y C:\TEMP
State: STANDARD.

-1\*- *Customize Group: Misc Custom* 37% of 1.7k (44,45) (C

```

- There are several predefined Custom colour Themes for Emacs

GNU Emacs 25.3.1 [c:/SW43/app_dev_examples/modules/highlight_similar/source/highlight_similar_demo.magik] -- Mon Oct

File Edit Options Buffers Tools SW Index Magik YASnippet Hide/Show Help

- Highlight Active Region
- Highlight Matching Parentheses
- Line Wrapping in This Buffer >
- Default Search Options >
- Use CUA Keys (Cut/Paste with C-x/C-c/C-v)
- Use Directory Names in Buffer Names
- Save Place in Files between Sessions
- Blink Cursor
- Enter Debugger on Error
- Enter Debugger on Quit/C-g
- Multilingual Environment >
- Show/Hide >
- Set Default Font...
- Save Options
- Manage Emacs Packages
- Customize Emacs >
 - Custom Themes
 - Choose a pre-defined customization theme
 - Browse Customization Groups
 - Saved Options
 - New Options...
 - Specific Option...
 - Specific Face...
 - Specific Group...
 - All Settings Matching...
 - Options Matching...
 - Faces Matching...

```

      _endif
      topic={application_development_exeamples},
      gin.select_by_trail()
      closed by the current trail belonging
      be as those seelcted.
      new ()
      _set.all_collections().fast_elements()
      i.name)
      application.database.rwo_set().select(:col
      ered rwo set.geometr v set(.current_trail.wc
      _self.databus_make_data_
      1(Unix)--- highlight_similar
      *Custom Themes*
      Type RET or click to enable/d
      Type ? to describe the theme
      Theme files are named *-theme
      Note: Your custom settings t
      To migrate your settin
      Save Theme Settings
      Select more than one theme at a time
      Available Custom Themes:
      adwaita -- Face colors similar to the default theme of Gnome 3 (Adwaita)
      deeper-blue -- Face colors using a deep blue background.
      dichromacy -- Face colors suitable for red/green color-blind users.
      leuven -- Face colors with a light background.
      light-blue -- Face colors utilizing a light blue background.
      manoj-dark -- Very high contrast faces with a black background.
      misterioso -- Predominantly blue/cyan faces on a dark cyan background.
      tango-dark -- Face colors using the Tango palette (dark background).
      tango -- Face colors using the Tango palette (light background)
      1\%*- *Custom Themes* Top of 1.3k (1,0) [(Themes HL yas Abbrev)
  
```

Use them based on your own taste:

```
GNU Emacs 25.3.1 [ c:/SW43/app_dev_examples/modules/highlight_similar/source/highlight_similar_demo.magik ] -- Mon Oct 9 13:14 0.38 ==
File Edit Options Buffers Tools SW Index Magik YASnippet Hide/Show Help
highlight_similar_demo.magik
  _else
    .current_trail << _unset
  endif
  _elif data_type_name_is :record_selection_then
    _if data_isnt_unset_andif data[2] isnt_unset_then
      .current_rwo_set << data[2]
    else
      .current_rwo_set << rwo_set.new()
    endif
  _endif
  _self.manage_actions()
endmethod
$
#pragma(classify_level=debug, topic={application_development_exeamples}, usage={internal})
## Enable or disable actions as appropriate.
method highlight_similar_plugin.manage_actions()
  _self.action(:filter_to_trail).enabled? << _if .current_rwo_set isnt_unset_andif
    .current_rwo_set.size > 0_andif
    .current_trail isnt_unset_andif
    .current_trail.closed?
  _then
    >> _true
  _else
    >> _false
  _endif
endmethod
$
```

```
GNU Emacs 25.3.1 [ c:/SW43/app_dev_examples/modules/highlight_similar/source/highlight_similar_demo.magik ] -- Mon Oct 9 13:19 0.64 ==
File Edit Options Buffers Tools SW Index Magik YASnippet Hide/Show Help
highlight_similar_demo.magik
  _else
    .current_rwo_set << rwo_set.new()
  endif
  _endif
  _self.manage_actions()
endmethod
$
#pragma(classify_level=debug, topic={application_development_exeamples}, usage={internal})
## Enable or disable actions as appropriate.
method highlight_similar_plugin.manage_actions()
  _self.action(:filter_to_trail).enabled? << _if .current_rwo_set isnt_unset_andif
    .current_rwo_set.size > 0_andif
    .current_trail isnt_unset_andif
    .current_trail.closed?
  _then
    >> _true
  _else
    >> _false
  _endif
endmethod
$
```

Key bindings for Magik mode

Several of these commands interact with a Smallworld session. If there is a current Gis buffer, or only one Gis buffer, then the corresponding Smallworld session is used; otherwise, you are prompted for the required Gis buffer.

Commands which provide an interface to the Development Tools application are only applicable when this application is available in the relevant Smallworld session. The application is loaded and started if necessary; any errors are reported to the relevant Gis buffer.

F2, B	Transmit the buffer to a Smallworld session.
F2, R or F2, F8 or F8	Transmit the current region to a Smallworld session.
F2, M or F2, F7 or F7	Transmit the method which contains the cursor position to a Smallworld session.
F2, Enter	Transmit the top-level Magik language construction surrounding the point in the current buffer, such as a method or procedure definition, to a Smallworld session.
F2, \$	Transmit the chunk of Magik code, delimited by \$ lines, and containing the cursor position, to a Smallworld session.
F4, F4	For a running Smallworld session, finds the symbol that starts with the current word, or lists all the available completions.
F4, O or F2, Shift+X	Inspect a Magik object. Prompts for an expression which is evaluated within a Smallworld session to return the object to inspect; the word at the cursor position is available as a default.
F2, X	inspect; the word at the cursor position is available as a default.
F4, O and F2, Shift+X	display the object in an Object Inspector from the Development Tools application.
F2, X	displays the object in an Emacs Deep Print buffer.
F4, D	Send a method to a Magik Debugger from the Development Tools application in a Smallworld session.
F2, Spacebar	Insert the Magik template for the preceding text
F2, #	Add a leading # to each line in the region to convert them to comments.
F2, Esc, #	Remove a leading # from each line in the region.
F2, Q	Fill comment line. Formats the current line into separate comment lines, adding leading indentation and # or ## to match the initial line.
F2, H	Format the current line as a comment heading.
F2, T	Add a trace statement above current line. When executed, the text of the current line is output.
F3, T	Add a trace statement below current line. When executed, the text of the current line is output.
F4, S	Add a Debug statement; a toggle is available so that Debug statements can be transmitted as comments or as executable commands.
F4, R	Copy the current region to a work buffer.
F4, M	Copy the current method to a work buffer.
F4, N	Open a work buffer and give it a name.
Ctrl+M+H or F9	Mark the current method.
F4, C or F6	Copy the current method.
F4, W	Compare methods using standard Emacs facilities.
F4, E	interfaces to the Emacs command compare-windows.

Key bindings for other modes

Some key bindings are available for the other Smallworld major modes.

Aliases mode

Ctrl+C Run command definition. In the Aliases buffer, finds the command definition which contains the cursor position and uses the launcher program to run it; otherwise prompts for the required definition.

Environment mode

A buffer in Environment mode is readonly so alphanumeric keys are used as shortcuts for commands.

Enter Select the Smallworld Core product indicated by the cursor middle-clickposition; this product installation defines the environment for any new Smallworld sessions.

Spacebar Move the cursor to the next product installation in the buffer.

R Run the Gis command in the environment defined by the cursor position. Does not change the environment for any other sessions.

+ Add a new product installation; prompts for the Smallworld Core product location, and a name and version for the installation. Only available if a version file is in use.

Q Quit; closes the buffer without making any change to the environment.

Loadlist mode

F2, B Transmit buffer. Uses the load list to load Magik files into a current Smallworld session; prompts for the required Gis process if there is more than one.

F2, C Toggle to control whether compiled Magikc files are saved.

Ctrl+C, R Refresh buffer. Uses the current contents of the directory containing the load list file to update the contents of the buffer. You can change the order of entries and save a new load list file if required.

Message mode

F2, B Transmit buffer. Compiles the buffer contents and loads into a current Smallworld session; prompts for the required Gis process if there is more than one.

F2, C Compile message file. Compiles the buffer contents ready for loading, but does not load it.

F2, Up Arrow Move the cursor to the previous or next message.

F2, Down Arrow

F2, M Mark message.

Module mode

F2, B Transmit buffer. Uses the buffer contents to load all module source files and all compiled message files into a current Smallworld session; prompts for the required Gis process if there is more than one.

F2, D Reload module definition. Loads the buffer contents into a current Smallworld session; does not load module source files or resources.

F2, C Compile message files. Compiles all message files for the module ready for loading.

F2, Shift+R Remove module definition from a current Smallworld session. This is sometimes required to ensure the correct functionality is included in the session.

F2, S Toggle the option to save compiled Magikc files when loading module source files.

F2, R Toggle the option to force reloading of a module which is already loaded.

Product mode

- F2, B Transmit buffer. Uses the buffer contents to add the product to a current Smallworld session; prompts for the required Gis process if there is more than one.
- F2, R Reinitialise product. Uses the buffer contents to redefine the product in a current Smallworld session where the product is already defined.