

HB_Env

Bill Eaves

COLLABORATORS

	<i>TITLE :</i> HB_Env		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Bill Eaves	August 25, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	HB_Env	1
1.1	HBasic Environment for Golded 6	1
1.2	Requirements	2
1.3	History	2
1.4	Details	2
1.5	compiling	4
1.6	Future	5
1.7	Installation	5
1.8	CopyFiles	6
1.9	FileList	7
1.10	SetUP	8
1.11	ARexx Scripts	10
1.12	Glossary	10
1.13	Hints and Tips	11
1.14	GOLDED	12

Chapter 1

HB_Env

1.1 HBasic Environment for Golded 6

HiSoft BASIC environment for use with GoldEd v6.2x

Since version 6.21 the text editor Golded has included a generic parser which allows users to add their own colour highlighting of keywords.

This gave me the idea to create a GoldEd environment for HiSoft Basic to keep track of larger projects and highlight Basic keywords and System calls from user defined sub programmes and functions. See SCREENSHOT (note to view this requires you to have a Jpeg datatype installed).

In addition to the parser it seemed appropriate to use other features of Golded such as the Quickinfo.api to give information on Amiga OS keywords and also Explorer.api to give a list of other basic and header files in the same project directory.

With appropriate commenting it is also possible to use Golded's Folding feature to fold away large areas of code just leaving a reference visible that can be unfolded when desired.

Note this Golded environment is NOT a replacement for the HiSoft BASIC internal editor but adds some functionality missing in the HiSoft editor. You will still need to use the HB editor to run and compile the final code.

Requirements

Details

Installation

List of files contained in the archive

Hints and Tips

History

Future

1.2 Requirements

REQUIREMENTS TO USE THIS GOLDED ENVIRONMENT

HiSoft Basic 2 - there is not much point using this otherwise !

GoldEd v6.0 or higher, only version 6 and above supports generic parsing.

Golded C++ or C environments installed, these can be removed later if necessary.

1.3 History

HISTORY

Version 1.0 March 2000

First release. Golded Environment for HiSoft Basic2.

Features, Syntax highlighting, function scanner, quickinfo for OS functions, file ↔ explorer folding via comment statements and bracket matching.

Bug fixes - none. You can't fix bugs of a previously non existing version :)

1.4 Details

What this GoldED environment contains

Any HiSoft Basic files loaded into Golded will automatically start the HBasic filetype. Other files can be forced into using this filetype by clicking the RMB within Golded, selecting Properties and entering the word BASIC in the Filetype field.

SYNTAX PARSER

The Syntax parser will recognise the following different types of word with the HBasic file.

HS Basic Commands or Statements.	See the HSB manual for more details.
HS Basic Functions.	The functions used in HS Basic.
HS Basic compiler	Reserved words used by the HSB complier.
Other HS Basic words	that I want to distinguish from more general Statements in HSB2.
Amiga OS constants.	These are defined in the HS Basic .bc include files
Amiga OS "SUB programmes".	These are the OS functions that do not return values and have been defined a SUB in the HS Basic .bh header files
Amiga OS Functions.	These do return values and are defined as true functions in the HS Basic .bh header files.

Note that is possible to add dictionaries for third party libraries that have .bh and .bc header files plus corresponding .bmap file for use with HiSoft Basic. Two Arexx scripts are included that will read any .bh and .bc files in a directory and create dictionary files in Ram: that can be copied to Golded and renamed appropriately.

QUICKINFO

This will give information about the Amiga OS SUB or FUNCTION under the cursor. The API to do this already exists in all the C environments that Golded uses and you will need to setup Golded to use this API set up.

The database is different from the one used by the C environments as Hi Soft Basic can have variable data type IDs such as & before the opening function bracket. It can also have white space. Otherwise the Quickinfo gives identical information to those used in C environments.

SCANNER

The scanner will show a list of SUBS and Functions in the current document. Note if you project is spread over several files then functions in other files will not be shown. However the scanner window can be kept open and another document selected and

if this Gold Ed window is highlighted then clicking Update in the Scanner window ←
will show ←
the SUBS and FUNCTIONS in the currently active document.

I have not included a scanner for programme labels as there already exists an ←
Arexx label ←
scanner which will perform the same function. This scanner can be found in the ←
archive ←
"twGED_scanner.lha" by Thiels Wellpott. Follow the installation instructions in ←
this archive ←
if you need to use it. However, scanning for labels is not ideal as multi line ←
commands are ←
picked up as well due to the presence of the : symbol. eg CLS:PRINT X will be ←
shown ←
as a label. Note this is a problem with HiSoft Basic 2 as described on page 424 of ←
the manual.

EXPLORER

The Explorer window will show other basic files or header files in the same ←
directory or ←
sub-directories. This is useful for a project consisting of one main file which ←
includes other ←
basic and header files. Again the API is provided as part of the Golded C or C++ ←
environments ←
and you will need to set this up set up.

FOLDING

If is possible to fold sections of code into a single line. This is useful for ←
long files with ←
many sections of code. Both beginning and end fold markers are set as '*** but it ←
is possible to ←
change this, providing that the fold marker is a comment.

```
'*** My Huge Subprogramme
Lots of Basic lines here...
'***
```

Will be shown when folded as;

```
> My Huge Subprogramme
```

Note you must unfold all folded text if you want to use Golded's text search ←
function, it cannot ←
find text within folded areas.

What this GED environment will and will not do

1.5 compiling

What this GED environment will and will not do

This environment (currently at any rate) is merely an aid to editing text for use with HiSoft Basic 2 and should be run in conjunction with the Hi-Soft Basic editor. It is not meant to replace the HiSoft Editor.

It allows keyword highlighting and function scanning that the Hi-Soft Basic editor does not have. You do still need to use the HiSoft editor to link with the compiler which is not supported so far in my GED environment. Syntax checks and compilation should be done from the Hi-Soft basic editor.

The GED environment does not convert Basic keywords to UPPER case automatically. To do this you need to use the HSB editor.

1.6 Future

FUTURE

1. Fix any bugs or problems reported :o)
2. Look into adding the ability to run the HiSoft Basic compiler directly from Golded.
3. Maybe add a toolbar with the following functions.
 - a. A toolbar icon to access the function scanner and/or label scanner.
 - b. An icon to launch the HiSoft Basic editor containing the current file.
 - c. An icon to run the HiSoft Basic compiler directly from Golded, if implemented.
4. Use Golded's Context settings to auto indent various features such as IF blocks etc.
5. Add a proper installer.

1.7 Installation

INSTALLATION

The installation is manual, I do not know how to write installations that will directly modify the Golded Registry. However manual installation is straightforward involving only two steps, copying the files and then setting up Golded. Some basic knowledge of using a file manager such as Directory Opus or using the Shell will help.

1.8 CopyFiles

Copying the necessary Files.

All the files you need are in the Files folder of the main archive.

STEP ONE

First of all you need to create a directory called HBasic2 in Golded:add-ons. It does not need an icon but there is no harm if one is created. Copy both the api and syntax folders from this archive to Golded:add-ons/HBasic2.

Note: The API folder does NOT contain the following three files as they are part of the Golded distribution and cannot be distributed separately. You can need to copy them to the HBasic2/api folder.

If you do not have a C type environment already installed, you will need to install one, copy the three files to HBasic2/api and then uninstall the C environment if it is no longer required.

The three files needed are:

- explorer.api
- mirror.api
- quickinfo.api

Note. Experienced users may not need to copy these files but access them directly from their current path if a C type environment is already in use.

Note2. Do NOT copy explorer.config from the C++ distribution but use the one provided in this archive in the api folder.

STEP TWO

No more directories need to be created but make use of existing folders in the Golded: assign.

Copy Scanner/HBasic2 from this archive to Golded:etc/scanner

Copy the three files from Presets in this archive to Golded:etc/registry/presets

List of files contained in the archive

1.9 FileList

The following files are provided in this archive.

1. The Documents folder.

This guide and supporting images.

2. The Files Folder

API folder

```
explorer.config    - Configuration file for the explorer.api
os3.basic          - Configuration file for the quickinfo.api
```

Note the following are needed but not provided due to copyright reasons.

```
mirror.api
explorer.api
quickinfo.api
```

SYNTAX folder

```
Compiler
Functions
OS_Constants'x' (five files 1 to 5)
OS_Functions
OS_Sub
Special
Statements
```

These are the Dictionary Files, plain ascii text files that are used by the ↔
Generic Parser to
highlight different types of keywords.

SCANNER folder

```
HBasic2
```

PRESETS folder.

```
hbasic.api
hbasic.syntax
hbasic.misc
```

These files are used to set up the new HBasic filetype in Golded.

SCRIPTS folder

Contains two Arexx scripts to create dictionary files for third party HBasic2 ↔
libraries.

These are tools and not needed to use the environment and can be copied anywhere ↔
the user wishes.

1.10 SetUP

Setting Up Golded for the HBasic2 Environment.

Firstly ensure that the necessary files have been copied from the archive to the correct places in the Golded: directory and that you have access to the Golded: C type API files.

Run Golded and select Customise from the Extras menu.

Highlight Filetypes and click the PLUS icon, a new filetype called Unnamed will appear.

Rename this using a linger-click rather than a normal double mouse click, change the name to something meaningful such as HBasic2.

Click the small + sign next to HBasic2 and you will see 2 folders, Configuration and Activation.

Setting the Activation.

Highlight Activation and once more click the PLUS icon. Linger Click the text that appears for the

new entry and enter the text "(#?.bas|#?.bh|#?.bc|BASIC)" without the quotes. This will enable

Gilded to recognise HBasic files or change the filetype by typing BASIC in the Properties filetype entry.

Setting the Configuration.

1. Plug Ins

Highlight the Configuration folder and click the PLUS icon. Golded's Pool requester will appear.

Click the small + next to Plug-ins, there should be an entry hbasic.api, if not then ensure all

files have been copied correctly to Golded:etc/registry/presets. Highlight hbasic.api and click

on the OK icon. An entry Plug-ins will appear below. Double click this and there should be three

entries for the available plugins. These will read:

```
gilded:add-ons/hbasic2/api/explorer.api
gilded:add-ons/hbasic2/api/mirror.api
gilded:add-ons/hbasic2/api/quickinfo.api
```

Note. Advanced users can linger-click to change the above paths to look at existing API files in say a C++ file if they wish to save disk space.

Double click the path for explorer.api and the config entry should read CONFIG="golded:add-ons/hbasic2/api/explorer.config" do not change this.

Double click the mirror.api path and the config entry should be blank. Leave this.

Double click the quickinfo.api path and the entry should read golded:add-ons/hbasic2/api/os3.basic do not change this.

Click the SAVE to store the api@plugin settings.

If not proceeding to setting up the Syntax settings immediately then click the ↔ next SAVE button.

It is probably best to continue with the next stage.

2. Syntax

Highlight the Configuration folder and click the PLUS icon. Golded's Pool ↔ requester will appear.

Click the small + next to Syntax Highlighting, there should be an entry hbasic. ↔ syntax, if not

then ensure all files have been copied correctly to Golded:etc/registry/presets. ↔ Highlight

hbasic.syntax and click on the OK icon. An entry Syntax will appear below. Double ↔ click this

and the Syntax window will appear.

The first option is where you set the colours for the various types of keyword as ↔ per normal

practice with Golded. There should be 10 entries,

There is no sample page provided as this is not available for the Generic parser ↔ which the HBasic environment uses.

Click the Details icon and the window changes to show how the parser is setup. Now ↔ click the button marked Details.

Users with low memory or slower CPUs may wish to delete one or more of the ↔ OS_Constants dictionary

files. These have been arranged so that OS_Constants1 contains keywords for the OS ↔ most likely to

be used in an HBasic programme and OS_Constants5 has keywords unlikely to be used ↔ .

To delete a dictionary click the small plus next to OS_Constants and you will see ↔ the 5 dictionary

files, highlight the one to delete and click on the X button.

Click OK and then on the SAVE to save the Syntax settings.

3. Folding

Highlight the Configuration folder and click the PLUS icon. Golded's Pool ↔ requester will appear.

Click the small + next to Miscellaneous, there should be an entry hbasic.misc
Highlight hbasic.misc and click on the OK icon. An entry Miscellaneous will appear
below. There
is no need to make any further corrections. Click OK and then on the SAVE to save
the Misc
settings.

Click the next SAVE button and the HBasic2 environment setup is completed.

1.11 ARexx Scripts

Two arexx scripts are supplied to create Golded Dictionary files for using third
party HiSoft
Basic library calls. The first script is HB_Functions.rexx which scans for *.bh
files and creates
a file in Ram: called Functions and one called Subs. The second script is
HB_Constants.rexx
which scans for *.bc files and creates a file in ram: called Constants.

These can be copied to Golded:etc/syntax/HBasic2 from where they can be added to
Golded's
Dictionaries. It is best to rename these to specific names like My_Functions etc.

The scripts work on all the *.bh and *.bc files in a given directory so do not use
the HBasic/BH
directory or you may be in for quite a wait while all the files are processed. The
best way is
to copy the third party *.bh and *.bc files into a clean directory, ram: would be
perfect, and then
run the scripts.

The syntax to use is "rx scriptname directory" eg. rx HB_Functions.rexx ram: will
create a Subs
file and a Functions file for any *.bh files in ram:

1.12 Glossary

Linger Click.

Golded uses this rather strange way of selecting text items in order to edit them.
A double click
is normally used to enter parameters or in some cases has no function. The linger
click is a
single click followed by another single click after a short period. If you find
that you have
difficulty, either nothing happens or a double click is detected instead then
alter the workbench
Input preferences and reduce the time delay for the double click.

1.13 Hints and Tips

Hints and Tips

This environment for Golded will work without paying attention to the following but may not be at its best.

Use nice fonts

A small font is best for use with the explorer. This can be edited in the explorer .config file.

It is currently set up to use XHelvetica.font size 9 which most users will have already. If the font in the config file is not found then Golded uses the same font used for the text display. I actually use tny.font size 8 which can be found on Aminet.

Programming Style

There are a few things that need to be adhered to to make the environment highlight things properly. NOTE. These will NOT have any affect on the HiSoft Basic programme itself. It will only affect the way Golded displays the text.

1. Variables.

In HiSoft Basic you can have variables A% and A& etc which are different entities. If the qualifier is omitted then a variable A will be of the type defined in the programme such as DEF INT A-Z or explicitly as a constant in a header file. I originally did not include the & symbol in the words highlighted from the OS header files but then decided as a similar variable with a % or \$ sign would also be highlighted, decided that only variables as exactly defined in the header files would get highlighted. So make sure that when using functions, sub programmes and constants for the Amiga OS that you do include the type qualifier. This will not affect your code at all but will affect whether an OS call is marked as such in Golded. It does also encourage better programming practice by ensuring that variables always have the type qualifier and avoids confusion.

I doubt anyone would decide to define a variable GT_Underscore% which would be similar to the OS constant GT_Underscore& and then rely on chance to use GT_Underscore and pick up the one which was defined first. However even in the HiSoft Basic examples variables are used that are identical

in name to some OS constants if they were used !

2. Comments and Compiler Options.

In HiSoft Basic, you can use both REM and ' to comment out lines.

The first generic parser for Golded would only let you use one item, I chose the ' to highlight comments. Later versions of Golded will now (at my request) let you define more than one word to define a single line comment. However I chose to just restrict highlighting comments just to the ' symbol. There is a reason for limiting to just the one method.

HiSoft Basic has compiler options that are a comment followed by a keyword beginning with a \$ symbol. I wanted to highlight compiler options. If they begin with a ' symbol then they will be shown as a comment but if they start REM then the associated compiler keywords are shown as such.

So although it makes no difference to your Basic code it is best to use ' for comments and REM for defining compiler options. It does not matter if you do not stick to this but the highlighting in Golded will not be as useful as it could.

3. Comments and Folding.

Golded lets you fold subroutines etc into a single line. I did think of using something like ' FOLD for this but decided to use what most people may have as comments anyway. It is of course possible to change the folding string to suit your own needs but it MUST be in the form of a comment so it is transparent to HiSoft Basic.

The fold symbol now for both starting and ending folds is '*** as similar comments are often used at the beginning of routines for clarity anyway. Note that ' *** will not work but it is easy to do a global replace in Golded if you want to use the folding feature. Note the latter is useful for displaying comments outwith any folded structures. See Example1 and Example2

1.14 GOLDED

ABOUT GOLDED.

Golded is an extremely configurable text editor for the Amiga.

The following information which is taken from the Golded manual, the website will tell you all you need to know.

WWW SUPPORT

Visit the internet support site to find the latest news, updates and tools:

<http://members.tripod.com/golded>
