
Subject: Re: Saving IDL History Across Sessions...

Posted by [David Fanning](#) on Wed, 04 May 2005 03:32:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

Robi writes:

> Does anybody have any clever kludge for saving your IDL history across
> sessions, such that the next time you start IDL it can recall commands
> that had been entered prior to exiting the previous IDL session? The
> possibility of this looks bleak since it looks like you can't really
> futz with the recall buffer in IDL... once you log on it starts fresh.
> But just in case somebody had ever considered this and found it
> impossible, well, then I could just go to bed and never worry about it
> ever again.

Good night! :-)

Cheers,

David

P.S. I assume you are NOT using the fab IDLWAVE mode, which
can do *everything*.

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: Saving IDL History Across Sessions...

Posted by [Mark Hadfield](#) on Wed, 04 May 2005 04:33:46 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> Robi writes:
>> Does anybody have any clever kludge for saving your IDL history across
>> sessions, such that the next time you start IDL it can recall commands
>> that had been entered prior to exiting the previous IDL session? The
>> possibility of this looks bleak since it looks like you can't really
>> futz with the recall buffer in IDL... once you log on it starts fresh.
>> But just in case somebody had ever considered this and found it
>> impossible, well, then I could just go to bed and never worry about it
>> ever again.
>
>
> Good night! :-)

Or you could wait for some future version of IDL that does this for you.
Nudge nudge wink wink.

--

Mark Hadfield "Ka puwaha te tai nei, Hoesa tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)

Subject: Re: Saving IDL History Across Sessions...
Posted by [R.Bauer](#) on Wed, 04 May 2005 05:07:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Robi wrote:

> Hi All,
>
> Does anybody have any clever kludge for saving your IDL history across
> sessions, such that the next time you start IDL it can recall commands
> that had been entered prior to exiting the previous IDL session? The
> possibility of this looks bleak since it looks like you can't really
> futz with the recall buffer in IDL... once you log on it starts fresh.
> But just in case somebody had ever considered this and found it
> impossible, well, then I could just go to bed and never worry about it
> ever again.
>
> Best -Tim.

look at journal

and then write the next day a function or procedure from your inputs.
Give them good names and describe them by using comments so you get know
them over years.

cheers
Reimar

--

Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
<http://www.fz-juelich.de/icg/icg-i/>

=====

a IDL library at Forschungszentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html

Subject: Re: Saving IDL History Across Sessions...

Posted by [savoie](#) on Wed, 04 May 2005 15:01:42 GMT

[View Forum Message](#) <> [Reply to Message](#)

Mark Hadfield <m.hadfield@niwa.co.nz> writes:

> David Fanning wrote:

>> Robi writes:

>>> Does anybody have any clever kludge for saving your IDL history across
>>> sessions,

<snip/>

>> Good night! :-)

>

> Or you could wait for some future version of IDL that does this for

> you. Nudge nudge wink wink.

Someone has to say it. IDLWAVE does this. <http://idlwave.org>, but of course, you have to be using emacs.

Matt

--

Matthew Savoie - Scientific Programmer

National Snow and Ice Data Center

(303) 735-0785 <http://nsidc.org>

Subject: Re: Saving IDL History Across Sessions...

Posted by [Tdogg](#) on Wed, 04 May 2005 18:49:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi guys,

I was asked by some folks that are putting together a package for use by users of a radio telescope whether it was possible since a lot of the time you'd like to have access (at the command line, not in a journal) to what you did last night while on the telescope.

> Someone has to say it. IDLWAVE does this. <http://idlwave.org>, but of

> course, you have to be using emacs.

So this won't apply here because we're not expecting IDL experts, and, well, IDLWAVE is too much of a hurdle to ask newbies to jump when they're trying to learn to use a 50 story telescope AND use IDL to reduce their data.

> look at journal

> and then write the next day a function or procedure from your inputs.

> Give them good names and describe them by using comments so you get know
> them over years.

Journal was my first thought... but that doesn't work, since there's no way to stick the contents of the journal back into the recall buffer unless you run everything in IDL first, which is just not the way to go.

> Or you could wait for some future version of IDL that does this for you.
> Nudge nudge wink wink.

Um, yeah, that's what I thought. What's the best way to nudge, nudge RSI that this is a good idea?

Thanks for the help fellas. Best -Tim.

Subject: Re: Saving IDL History Across Sessions...
Posted by [R.Bauer](#) on Wed, 04 May 2005 22:50:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Robi wrote:

> Hi guys,
>
> I was asked by some folks that are putting together a package for use
> by users of a radio telescope whether it was possible since a lot of
> the time you'd like to have access (at the command line, not in a
> journal) to what you did last night while on the telescope.
>
>> Someone has to say it. IDLWAVE does this. <http://idlwave.org>, but
> of
>> course, you have to be using emacs.
>
> So this won't apply here because we're not expecting IDL experts, and,
> well, IDLWAVE is too much of a hurdle to ask newbies to jump when
> they're trying to learn to use a 50 story telescope AND use IDL to
> reduce their data.

What is an idl expert? We all do learn new things every day. You should think on explanation how to write functions to the users instead of always do all in the idl commandline. It is much easier to use a function as to write all the code into the command line everytime it is needed or recall it.

cheers

Reimar

--

Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
<http://www.fz-juelich.de/icg/icg-i/>

=====

a IDL library at Forschungszentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html

Subject: Re: Saving IDL History Across Sessions...
Posted by [Tdogg](#) on Thu, 05 May 2005 04:05:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Reimar Bauer wrote:

- > What is an idl expert? We all do learn new things every day. You should
- > think on explanation how to write functions to the users instead of always
- > do all in the idl commandline. It is much easier to use a function as to
- > write all the code into the command line everytime it is needed or recall

Hi Reimar,

For someone to show up at an observatory who might have never seen IDL and yet wants to look at a spectrum coming from the telescope in a timely fashion, I think it would be safe to say (as a longtime user of IDLWAVE) that the user is not looking for an hour-long tutorial on IDLWAVE before actually running the IDL code. Worse, who died and made emacs the required editor for an observer to use? I'm afraid I'll just have to disagree with your assessment of the relative ease of packing commands into functions or procedures, compiling them and calling them rather than calling a module that does one specific task at the command line when you are INTERACTING with your DATA in this LANGUAGE.

But, point is, you definitely can't fill the recall buffer when IDL is started up, so unless you're using IDLWAVE you're outta luck. Just wanted to be sure there were no clever ways around it. Thanks for the responses.

Best -Tim.

Subject: Re: Saving IDL History Across Sessions...
Posted by [Ken Mankoff](#) on Thu, 05 May 2005 04:43:02 GMT

On Thu, 4 May 2005, Robi wrote:

> who died and made emacs the required editor for an observer to use?
vi? Just kidding {dons flame-retardant suit}.

What platform are you on? What environment do you use normally?
xterm? idlde? Terminal.app? I'm going to assume xterm on *nix.

Who said your users need to be aware that they are in emacs? How is
the IDLWAVE shell any different from an xterm + IDL session?

(Answer: it is much better!) What about this in an xterm?:

```
emacs -nw -f idlwave-shell
```

The only problem with this is if a routine crashes, they will find
themselves looking at source code (the line the error occurred on)
rather than a normal IDL error message. You could probably turn this
feature off (ask JD), or just put a big sticky-note on the monitor
telling them to hit the three necessary keys to return to the shell.
And some would say that is a feature not a problem.

> But, point is, you definitely can't fill the recall buffer when
> IDL is started up, so unless you're using IDLWAVE you're outta
> luck. Just wanted to be sure there were no clever ways around it.
> Thanks for the responses.

I can think of some other solutions, much uglier hacks than the
above, but i'll list them just for brainstorming sake:

Write a wrapper (AppleScript, windows Macro, something else?) that
takes the journal and simulates a user re-typing each command.

This could work, or might not work (if a file you try to read has
been removed), or could be slow (if some commands are
computationally long) or could be disastrous (if you overwrite
files). But it can be done.

If commands are slow, and you don't want to actually execute them,
have your wrapper precede each command by a bogus character (line
number of journal, letter 'x', whatever (ex: 42_print, 'foo')). Now
they will all fail (and therefore run very quickly) to run a
previous command, I have to press up-arrow to get to it, and then
delete the 42_. Better than your current situation maybe?

If you want to run the actual command, so that your data/vars are
valid, this would be an improvement on IDLWAVE. But now you run into
issues with non-existent files you want to load, or writing files
you don't want to overwrite.

To stop yourself from overwriting files,

```
cd, 'safe_dir'  
reload_cmds ; external macro/AppleScript inputter  
cd, '..'
```

and that will stop files from getting destroyed. Unless, of course, someone wrote something out with an absolute path.

You could re-mount the disk in read-only mode, reload commands, and then re-mount in read-write mode. Ack! Ugly.

You could tell people not to quit IDL. Now you don't need to reload the commands!

You could build a wrapper to IDL that pipes everything through the Unix tee command (or RPC?) to a 2nd IDL session. Now even if they quit there is a 'backup' IDL that is identical. I'm not sure how you could restore the backup.

In short, I think there are multiple solutions out there. Some elegant, some ugly. Some are easier for you to implement (IDLWAVE) than others (re-mounting disks) and others are easier for your users (IDLWAVE) than others (not quitting).

-k.

Subject: Re: Saving IDL History Across Sessions...
Posted by [R.Bauer](#) on Thu, 05 May 2005 07:23:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Robi wrote:

```
> Reimar Bauer wrote:  
>> What is an idl expert? We all do learn new things every day. You  
> should  
>> think on explanation how to write functions to the users instead of  
> always  
>> do all in the idl commandline. It is much easier to use a function as  
> to  
>> write all the code into the command line everytime it is needed or  
> recall  
>  
> Hi Reimar,  
>  
> For someone to show up at an observatory who might have never seen IDL  
> and yet wants to look at a spectrum coming from the telescope in a
```

> timely fashion, I think it would be safe to say (as a longtime user of
> IDLWAVE) that the user is not looking for an hour-long tutorial on
> IDLWAVE before actually running the IDL code. Worse, who died and made
> emacs the required editor for an observer to use? I'm afraid I'll just
> have to disagree with your assessment of the relative ease of packing
> commands into functions or procedures, compiling them and calling them
> rather than calling a module that does one specific task at the command
> line when you are INTERACTING with your DATA in this LANGUAGE.

Hi Tim,

we were caught 10 years ago in the same situation. A few amount of us starts to write routines and sometimes later we have build up dependend on this routines the first internal lessons for all the idl users of our department. This was mostly about using self written routines for common problems each user has. We did five lessons about 90 minutes in a computer room showing people that it is possible to extend the idl language by user written routines. It was an interactive course with 19 colleagues are willing to learn on 12 computers. Later on this was repeated always showing new stuff or more elegant ways to solve the common problems. This was the time were the library of ICG was born too. Sometime later we recognized that's only idl is necessary and user written code of course to do anything we want with the data. The library grows up and standards were defined e.g. the icg-data-structure, how to make time series analyses and so on

Now and always since 1994 it was possible for each of my colleagues who is interested in idl to get an idl excercise lesson on ForschungsZentrum Jülich over three days. The lesson is for beginners with no programming acknowledgment. There were a lot of people willing to go three days to this excercise too.

The interaction with data in this language on the IDL> Prompt is always done to tryout something or to test different things but then mostly an available routine gets an extension or a new routine is build.

I agree to you that it will take time to build functions and procedure from things you long time have done interactive. But it takes almost longer if you still waiting to do it. Probably you could start with main programs. After running one you have access to all variables too. Or with procedures were you define all parameters you want to return in the parameter list. If you start it does not matter if the code does not look pretty excellent. It should only do the thing you want it to do.

While Ken gives advertices for the recall buffer I just want to show here only that we were in a similiar situation in the past and how much effort you and your colleagues have if all could speak the same foreigner language. ;-)

cheers

Reimar

--

Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
<http://www.fz-juelich.de/icg/icg-i/>

=====

a IDL library at Forschungszentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html

Subject: Re: Saving IDL History Across Sessions...
Posted by [JD Smith](#) on Thu, 05 May 2005 18:03:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thu, 05 May 2005 00:43:02 -0400, Ken Mankoff wrote:

>
> On Thu, 4 May 2005, Robi wrote:
>> who died and made emacs the required editor for an observer to use?
> vi? Just kidding {dons flame-retardant suit}.
>
> What platform are you on? What environment do you use normally?
> xterm? idlde? Terminal.app? I'm going to assume xterm on *nix.
>
> Who said your users need to be aware that they are in emacs? How is
> the IDLWAVE shell any different from an xterm + IDL session?
> (Answer: it is much better!) What about this in an xterm?:
> emacs -nw -f idlwave-shell

Make this a button on your toolbar... it's what I do. In fact, make a
button which runs:

```
emacs --eval "(idlwave-shell 'quick)" --eval '(idlwave-shell-send-command "observe")'
```

Where "observe" is your main IDL routine, and you'll be up and running
in no time. You've acknowledged that your users don't need to know
all of IDL to be productive. Similarly, you don't need to know
everything about IDLWAVE to use it. In fact I suspect the majority of
IDLWAVE users don't even know they're using IDLWAVE. If you can type
commands at a prompt, you can use IDLWAVE (not use it well, but use it
at least as well as an XTerm).

If you want to make your saved command history very deep, try:

```
(setq comint-input-ring-size 1024)
```

in the default .emacs for the users.

- > The only problem with this is if a routine crashes, they will find
- > themselves looking at source code (the line the error occurred on)
- > rather than a normal IDL error message. You could probably turn this
- > feature off (ask JD), or just put a big sticky-note on the monitor
- > telling them to hit the three necessary keys to return to the shell.
- > And some would say that is a feature not a problem.

As they would in IDLDE or an XTerm in the event of a crash. A sticky which tells them to type "retall" is all you need (true in the IDLDE, IDLWAVE, or an XTerm). Even better, bind "retall" to a key like F4:

```
(add-hook 'idlwave-shell-mode-hook
  (lambda ()
    (local-set-key [f4] 'idlwave-shell-retall)))
```

in the default .emacs, and attach a label to the keyboard key. You could make as many of these as you want, customized for various observatory-specific functions to run. Doesn't get much easier for a user than that.

JD
