Subject: reading wav files

Posted by inettle1 on Tue, 16 Jan 2001 01:38:48 GMT

View Forum Message <> Reply to Message

Does anyone have a routine/set of routines to read \*.wav files in IDL? And if so, do you feel like sharing it??:)

Thanks for your time! Jeff Nettles (jnettle1@utk.edu)

Subject: Re: Reading Wav files

Posted by David Fanning on Wed, 21 Aug 2002 16:59:09 GMT

View Forum Message <> Reply to Message

Mike Alport (alport@nu.ac.za) writes:

- > This seems a trivial problem: the IDL 5.5 documentation gives the following
- > syntax for reading sound .wav files:
- > Result = READ\_WAV (Filename [, Rate])
- > So, when I try it, I get:
- > IDL> Result = READ\_WAV( "ramsay0150.wav" )
- > % Variable is undefined: READ\_WAV.

>

> Have they switched the calling procedure name on us?

No, it is definitely a function. Try this:

IDL> .compile read\_wav
IDL> Result = READ\_WAV( "ramsay0150.wav" )

Cheers,

David

--

David W. Fanning, Ph.D.

Fanning Software Consulting, Inc.

Phone: 970-221-0438, E-mail: david@dfanning.com

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

Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Reading Wav files

Posted by JD Smith on Wed, 21 Aug 2002 20:24:41 GMT

View Forum Message <> Reply to Message

On Wed, 21 Aug 2002 09:08:35 -0700, Mike Alport wrote:

- > This seems a trivial problem: the IDL 5.5 documentation gives the
- > following syntax for reading sound .wav files: Result = READ\_WAV (
- > Filename [, Rate] ) So, when I try it, I get: IDL> Result = READ WAV(
- > "ramsay0150.wav" ) % Variable is undefined: READ\_WAV.

>

> Have they switched the calling procedure name on us? Thanks, Mike Alport

This is probably related to the ()/[] degeneracy in IDL array indexing. I can replicate your error:

```
IDL> result = read_wav('/usr/share/licq/sounds/fun/Auth.wav')
% Compiled module: READ_WAV.
IDL> read_wav=2
IDL> result = read_wav('/usr/share/licq/sounds/fun/Auth.wav')
IDL> help,result
RESULT BYTE = Array[5400]
```

No problem thus far.. but what if I start IDL again with:

```
IDL> read_wav=1
IDL> r=temporary(read_wav)
IDL> help
% At $MAIN$
I INT = 167
R INT = 1
READ WAV UNDEFINED = <Undefined>
```

Uh, oh, an "undefined variable" READ\_WAV exists. You can imagine what will happen now:

```
IDL> result = read_wav('/usr/share/licq/sounds/fun/Auth.wav')
% Variable is undefined: READ_WAV.
% Execution halted at: $MAIN$
```

IDL thinks you're trying to index an undefined variable! The problem is, variables sometimes exists as "undefined" zombies, even after you think they're gone. In this state, that can still "shadow" or overlap with function calls. A few options for solving the problem:

- 1. Find out where the undefined variable READ\_WAV is getting created, and eliminate that.
- Use "compile\_opt STRICTARR" at the beginning of your procedure, or at the main level:

IDL> compile opt STRICTARR

IDL> result = read\_wav('/usr/share/licq/sounds/fun/Auth.wav')
% Compiled module: READ WAV.

- 3. Use "forward\_function READ\_WAV" to declare it a function.
- 4. Clamor for RSI to make STRICTARR the default (though duck as Craig throws an eraser at your head.)

You could also just put "compile\_opt STRICTARR" in your startup file, and I think it then applies to all routines and interactive commands (anybody know for sure?).

Good luck,

JD

Subject: Re: Reading Wav files
Posted by Mike Alport on Thu, 22 Aug 2002 08:21:47 GMT

Posted by Mike Alport on Thu, 22 Aug 2002 06.21.47

View Forum Message <> Reply to Message

Thanks guys - both your suggestions have worked. When I was first playing around with read\_wav, I must have inavertently made it a variable. An explicit ".compile read\_wav" then seems overide that earlier definition. I have now also learnt about STRICTARR.

Thanks again,

Mike

PS - now I just need to play .wav files from IDL - I see that Matt Feinstein asked the same question a few days ago wrt IDL\_Tools.

"JD Smith" <jdsmith@as.arizona.edu> wrote in message news:pan.2002.08.21.20.24.41.192394.2937@as.arizona.edu...

- > On Wed, 21 Aug 2002 09:08:35 -0700, Mike Alport wrote:
- >> This seems a trivial problem: the IDL 5.5 documentation gives the
- >> following syntax for reading sound .wav files: Result = READ\_WAV (
  >> Filename [, Rate] ) So, when I try it, I get: IDL> Result = READ\_WAV(
- >> "ramsay0150.wav") % Variable is undefined: READ\_WAV.

>>

>

- >> Have they switched the calling procedure name on us? Thanks, Mike Alport
- > This is probably related to the ()/[] degeneracy in IDL array indexing.
- > I can replicate your error:

>

- > IDL> result = read\_wav('/usr/share/licq/sounds/fun/Auth.wav')
- > % Compiled module: READ\_WAV.

```
> IDL> read wav=2
> IDL> result = read wav('/usr/share/licg/sounds/fun/Auth.wav')
> IDL> help,result
> RESULT
                 BYTE
                          = Array[5400]
>
  No problem thus far.. but what if I start IDL again with:
>
> IDL> read_wav=1
> IDL> r=temporary(read wav)
> IDL> help
> % At $MAIN$
> l
            INT
                        167
> R
             INT
                    =
                           1
> READ_WAV
                   UNDEFINED = <Undefined>
>
> Uh, oh, an "undefined variable" READ_WAV exists. You can imagine what
> will happen now:
>
> IDL> result = read wav('/usr/share/licg/sounds/fun/Auth.wav')
> % Variable is undefined: READ WAV.
> % Execution halted at: $MAIN$
>
> IDL thinks you're trying to index an undefined variable! The problem
> is, variables sometimes exists as "undefined" zombies, even after you
> think they're gone. In this state, that can still "shadow" or overlap
> with function calls. A few options for solving the problem:
>
  1. Find out where the undefined variable READ_WAV is getting created,
    and eliminate that.
>
>
  2. Use "compile_opt STRICTARR" at the beginning of your procedure, or
    at the main level:
>
>
    IDL> compile_opt STRICTARR
>
    IDL> result = read_wav('/usr/share/licq/sounds/fun/Auth.wav')
>
    % Compiled module: READ WAV.
>
>
  3. Use "forward function READ WAV" to declare it a function.
>
  4. Clamor for RSI to make STRICTARR the default (though duck as Craig
>
    throws an eraser at your head.)
>
  You could also just put "compile_opt STRICTARR" in your startup file,
> and I think it then applies to all routines and interactive commands
  (anybody know for sure?).
>
> Good luck,
>
```

Subject: Re: reading wav files

Posted by wem on Thu, 03 Nov 2005 10:51:10 GMT

View Forum Message <> Reply to Message

## rlayberry@hotmail.com wrote:

- > I want to be able to read .wav audio files in IDL, such that after
- > stripping the header I can look at the actual values. I can't find a
- > website with the actual binary format of wav files explained. I
- > remember doing something like this years ago. I read wav files and
- > then displayed them and processed them etc (I was comparing wav files
- > before and after being sent VOIP) just cant remember how I did it. I
- > was also using PV-WZVE before which may have had a wav file reader?

>

> thanks

>

> russ

>

You want something like this?: http://www.borg.com/~jglatt/tech/wave.htm