Subject: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Thu, 24 Jul 2014 07:00:39 GMT
View Forum Message <> Reply to Message

Mini IDL Evaluator (MIDLE) evaluates simple IDL statements and most expressions without EXECUTE, i.e. virtual machine safe. It can be an alternative to EXECUTE in many cases.

GitHub repo:
https://github.com/ywangd/midle

It is currently at version 0.1.0 and can be also be downloaded at
https://github.com/ywangd/midle/archive/v0.1.0.zip

MIDLE implements its own parser and evaluates simple IDL statements and expressions without resorting to the power of `EXECUTE`. It even adds additional language features such as syntax for HASH and LIST literals, higher level array concatenation, bettering support for chaining function/method calls and subscripts.

MIDLE is however not without limitations. Some limitations are due to the limit of IDL language itself, notably output arguments and object property access (object method calls are OK). Others are deliberately set by design to meet the scope of MIDLE, notably program control constructs. Please refer to the GitHub page for details.

MIDLE requires IDL 8.0 or up (8.3 is recommended).

Here are a few examples using MIDLE (full documentation can be found at the GitHub repo page).

```
 ------------------------------------------------------------ --------
; Mandatory classic example
print, midle('"Hello, World!"')
midle, 'print, "Hello, World!"'

; Array of strings
midle, ['print, "STAR"', 'print, "WARS"']
; Or write them in one line
midle, 'print, "STAR" & print, "WARS"'

; Evaluate the content of given file
midle, 'filename', /file
; Passing variables
env = {num: 50}
print, midle('indgen(2,3,4, start=num)', env)

; Procedure call
midle, 'plot, indgen(50, start=100), /ynozero'
```

```
; Expressions
print, midle('-2.2 - 2 mod ((42. + 22) ^ 2 > 3 - 4.2) * 2.2 / 2.4')
print, midle('x eq 42 ? indgen(5, start=x) : indgen(5)', {x: 42})

; Assignment
print, midle('x = 42', env)
print, env.x  ; output 42
print, midle('h = Hash()', env)
print, midle('h["a"] = indgen(3,4,5)', env)
print, midle('h["a", 0, 1, 2] = 420', env)
print, (env.a)[0,1,2]  ; output 420
; List literal
print, midle('("a", "list", "literal")')

; Hash literal
print, midle('h{"x": 42, "y": 22, "description": "This is a hash literal"}')

; Higher level array concatenation:
env = {a: indgen(6,5,4,3,2), b: indgen(6,5,4,3,2, start=720)}
help, midle('[ [[[[a]]]], [[[[b]]]] ]', env)  ; concatenate on the 5th dimension

; Better support for chained function/method calls and subscripts
print, midle('list(indgen(3,4,5,6)[*,0:3:2,4,*][2,*,0,0:5:2], /extract).count()')
```

---------- --------

Comments and suggestions are welcome.

I'd like to thank Mike Galloy for his wonderful mgunit and idldoc, which I used extensively for developing MIDLE.

Cheers,
Yang

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by Craig Markwardt on Thu, 24 Jul 2014 15:02:57 GMT
View Forum Message <> Reply to Message

Very cool!

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Fri, 25 Jul 2014 01:33:11 GMT
View Forum Message <> Reply to Message

On Friday, July 25, 2014 1:02:57 AM UTC+10, Craig Markwardt wrote:

> Very cool!

Thanks Craig. Your work of PROTRANS is way cooler. Pity it has to be stopped. Otherwise it could possibly allow us to really break the limits of IDL language. Maybe a revamp of the language interface to IDL runtime!! Something like JVM and many languages it hosts.

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Mon, 28 Jul 2014 04:14:28 GMT
View Forum Message <> Reply to Message

On Thursday, July 24, 2014 5:00:39 PM UTC+10, SonicKenking wrote:
> Mini IDL Evaluator (MIDLE) evaluates simple IDL statements and most expressions without EXECUTE, i.e. virtual machine safe. It can be an alternative to EXECUTE in many cases.
>
>
>
> GitHub repo:
>
> https://github.com/ywangd/midle
>
>
>
> It is currently at version 0.1.0 and can be also be downloaded at https://github.com/ywangd/midle/archive/v0.1.0.zip
>
>
>
>
>
> MIDLE implements its own parser and evaluates simple IDL statements and expressions without resorting to the power of `EXECUTE`. It even adds additional language features such as syntax for HASH and LIST literals, higher level array concatenation, bettering support for chaining function/method calls and subscripts.
>
>
>
> MIDLE is however not without limitations. Some limitations are due to the limit of IDL language itself, notably output arguments and object property access (object method calls are OK). Others are deliberately set by design to meet the scope of MIDLE, notably program control constructs. Please refer to the GitHub page for details.
>
>
>
> MIDLE requires IDL 8.0 or up (8.3 is recommended).
>
>
>

> Here are a few examples using MIDLE (full documentation can be found at the GitHub repo page).
>
>
>
> ---------- --------
>
> ; Mandatory classic example
>
> print, midle("'Hello, World!'")
>
> midle, 'print, "Hello, World!"'
>
>
>
> ; Array of strings
>
> midle, ['print, "STAR"', 'print, "WARS"']
>
> ; Or write them in one line
>
> midle, 'print, "STAR" & print, "WARS"'
>
>
>
> ; Evaluate the content of given file
>
> midle, 'filename', /file
>
> ; Passing variables
>
> env = {num: 50}
>
> print, midle('indgen(2,3,4, start=num)', env)
>
>
>
> ; Procedure call
>
> midle, 'plot, indgen(50, start=100), /ynozero'
>
>
>
> ; Expressions
>
> print, midle('-2.2 - 2 mod ((42. + 22) ^ 2 > 3 - 4.2) * 2.2 / 2.4')
>
> print, midle('x eq 42 ? indgen(5, start=x) : indgen(5)', {x: 42})

```
>
>
>
>  ; Assignment
>
>  print, midle('x = 42', env)
>
>  print, env.x  ; output 42
>
>  print, midle('h = Hash()', env)
>
>  print, midle('h["a"] = indgen(3,4,5)', env)
>
>  print, midle('h["a", 0, 1, 2] = 420', env)
>
>  print, (env.a)[0,1,2]  ; output 420
>
>  ; List literal
>
>  print, midle('("a", "list", "literal")')
>
>
>
>  ; Hash literal
>
>  print, midle('h{"x": 42, "y": 22, "description": "This is a hash literal"}')
>
>
>
>  ; Higher level array concatenation:
>
>  env = {a: indgen(6,5,4,3,2), b: indgen(6,5,4,3,2, start=720)}
>
>  help, midle('[ [[[[a]]]], [[[[b]]]] ]', env)  ; concatenate on the 5th dimension
>
>
>
>  ; Better support for chained function/method calls and subscripts
>
>  print, midle('list(indgen(3,4,5,6)[*,0:3:2,4,*][2,*,0,0:5:2], /extract).count()')
>
>
>
>   ------------------------------------------------------------ --------
>
>
>
>  Comments and suggestions are welcome.
```

>
>
>
> I'd like to thank Mike Galloy for his wonderful mgunit and idldoc, which I used extensively for developing MIDLE.
>
>
>
> Cheers,
>
> Yang

MIDLE is now at v0.2.0 and can be downloaded at
https://github.com/ywangd/midle/archive/v0.2.0.zip

Version 0.2.0 2014-07-28

New Feature: Subscripts and dot notations can now also be chained for the left-hand-side variable of assignments. This allows direct assignment to an item of a list where the list itself is inside an array.
Improve: Error handling. MIDLE now always provides helpful information if the error is due to the input string of code. It also always return the error message through the error output keyword.
Bug Fix: Implicit integer and unsigned integer are now auto-promoted to their corresponding LONG and LONG64 types when necessary.

Version 0.1.1 2014-07-25

Improve: Array subscripting optimized.
Improve: Error handling for type conversion during array concatenation
Improve: Documentations
Bug Fix: Assignment can now be done to slice of a list

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by markb77 on Mon, 28 Jul 2014 15:45:01 GMT
View Forum Message <> Reply to Message

Cool! I'm not sure that I understand the implications of this.. Using this tool would I be able to create objects over the IDL_IDLbridge without using "execute"?

If so, what are the implications of this for parallel processing in IDL using the VM?

thanks
Mark

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Tue, 29 Jul 2014 00:49:03 GMT
View Forum Message <> Reply to Message

On Tuesday, July 29, 2014 1:45:01 AM UTC+10, superchromix wrote:
> Cool! I'm not sure that I understand the implications of this.. Using this tool would I be able to create objects over the IDL_IDLbridge without using "execute"?
>
>
>
> If so, what are the implications of this for parallel processing in IDL using the VM?
>
>
>
> thanks
>
> Mark

Unfortunately no ...

As I understand it, IDLbridge's EXECUTE method has the EXECUTE function built into the core. There is no way to substitute it out for other means. Even if the EXECUTE method can be replaced, the child process still requires to compile whatever routines it is given (even they are already compiled in the parent process). This is again impossible in VM ...

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by markb77 on Tue, 29 Jul 2014 08:50:25 GMT
View Forum Message <> Reply to Message

too bad.. the absence of VM-compatible multithreading is one of the big limitations of the IDL languange at the moment..

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by markb77 on Tue, 29 Jul 2014 20:31:10 GMT
View Forum Message <> Reply to Message

I'm still trying to understand how and why to use this tool.

Could MIDLE be used as... an IDL console implemented within an IDL program?

Otherwise, could you please explain what was your motivation for creating this project?

thanks!
Mark

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by rdahni on Wed, 30 Jul 2014 06:22:05 GMT
View Forum Message <> Reply to Message

On Wednesday, July 30, 2014 6:31:10 AM UTC+10, superchromix wrote:
> I'm still trying to understand how and why to use this tool.

I've just successfully used MIDLE to evaluate IDL expressions stored in configuration files for an interactive GUI-based application, thus avoiding the use of EXECUTE, enabling the application to use the freely available IDL Virtual Machine (i.e. no IDL license required).

For example, see Radar Visualiser at http://www.metvis.com.au/radarvis/

Robert

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Wed, 30 Jul 2014 06:35:47 GMT
View Forum Message <> Reply to Message

On Wednesday, July 30, 2014 6:31:10 AM UTC+10, superchromix wrote:
> I'm still trying to understand how and why to use this tool.
>
>
>
> Could MIDLE be used as... an IDL console implemented within an IDL program?
>
>
>
> Otherwise, could you please explain what was your motivation for creating this project?
>
>
>
> thanks!
>
> Mark

Hi Mark,

The original motivation of MIDLE came from the desire to use a script containing a list of IDL assignments as a configuration file of an application that runs in IDL VM. As EXECUTE cannot be used in VM, MIDLE is developed to parse the configuration file.

It is like if you use XML for configuration file, you need a XML parser. Similarly, if you use IDL statements for configuration file, you need an IDL parser and this is where MIDLE comes in.

Though it was not the intention of MIDLE, I guess it is possible to use it as a mini console (a pare down version of IDL) in a GUI application for some interactive data analysis. It could be useful as

it exposes some underlying IDL powers to end-users who do not own IDL licenses.

Cheers,
Yang

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by SonicKenking on Wed, 30 Jul 2014 11:36:01 GMT
View Forum Message <> Reply to Message

On Wednesday, July 30, 2014 4:35:47 PM UTC+10, SonicKenking wrote:
> On Wednesday, July 30, 2014 6:31:10 AM UTC+10, superchromix wrote:
>
>> I'm still trying to understand how and why to use this tool.
>
>>
>
>>
>
>>
>
>> Could MIDLE be used as... an IDL console implemented within an IDL program?
>
>>
>
>>
>
>>
>
>> Otherwise, could you please explain what was your motivation for creating this project?
>
>>
>
>>
>
>>
>
>> thanks!
>
>>
>
>> Mark
>
>
>
> Hi Mark,
>
>

>
> The original motivation of MIDLE came from the desire to use a script containing a list of IDL assignments as a configuration file of an application that runs in IDL VM. As EXECUTE cannot be used in VM, MIDLE is developed to parse the configuration file.
>
>
>
> It is like if you use XML for configuration file, you need a XML parser. Similarly, if you use IDL statements for configuration file, you need an IDL parser and this is where MIDLE comes in.
>
>
>
> Though it was not the intention of MIDLE, I guess it is possible to use it as a mini console (a pare down version of IDL) in a GUI application for some interactive data analysis. It could be useful as it exposes some underlying IDL powers to end-users who do not own IDL licenses.
>
>
>
> Cheers,
>
> Yang

Another potential usage is to provide a scripting language interpreter for a GUI application. Something like how you can script Photoshop with VBScript. In MIDLE's case, the scripting language happens to share the same syntax as the language with which the application itself is written in.

MIDLE as it is now lacks support for program control constructs, e.g. IF/ELSE, FOR, etc, which is pretty much mandatory for a proper scripting language. But it is possible to add them. In fact, this potential now really makes me want to add more features to MIDLE.

Subject: Re: [ANN] MIDLE - Almost an Alternative to EXECUTE
Posted by markb77 on Wed, 30 Jul 2014 13:34:31 GMT
View Forum Message <> Reply to Message

On Wednesday, July 30, 2014 1:36:01 PM UTC+2, SonicKenking wrote:

> Another potential usage is to provide a scripting language interpreter for a GUI application. Something like how you can script Photoshop with VBScript. In MIDLE's case, the scripting language happens to share the same syntax as the language with which the application itself is written in.
>
>
>
> MIDLE as it is now lacks support for program control constructs, e.g. IF/ELSE, FOR, etc, which is pretty much mandatory for a proper scripting language. But it is possible to add them. In fact, this potential now really makes me want to add more features to MIDLE.

I'm not sure that I fully understand this, but it seems very useful. I am also writing a GUI application which would benefit from the ability to be scripted...

ah - I see. If one had their IDL code already built into a SAV file... using this tool one would still be able to add features to the code by writing a "plugin" module (in IDL) and then using MIDLE to execute it. This would also be VM-safe. cool!
