Gramps Andy's Fireside Chat v4.1

DYALOG

Elsinore 2017

Andy Shiers

4.1 ?

- There's a difference between rm and mv !

- (There's also a difference between

```
rm  -r  andys*
rm  -r  andys  *
```

)

☹

Agenda

- AOB
- Triggers
- 16.0
  - Changes
  - Late changes
  - Very late changes
  - Exceedingly late changes
- 17.0
- Miscellaneous items
- APLMON

AOB

- DECF
  - will remain on AIX because the hardware supports them (7 times faster than software emulated)
  - Dyalog will convert between DECF and BID on the fly
- HTMLRenderer
  - In 16.0 for Windows
  - In 16.0 for macOS
  - Linux - you've seen as much as I have !
  - Pi - to follow
  - AIX - no support for Electron

AOB

- GitHub repositories
  - Installation images have most recent version when made
    - Can always update yourself from GitHub
  - https://github.com/dyalog
  - Includes RIDE.
    - Report issues here or to support

Triggers

- A trigger is a function that is called when the variable or field it is associated with is altered
- Often used with databinding
- Change of value causes trigger function to be called
- Are tricksy things !
  - If in doubt, ⎕FX ⎕VR Trigger (not ⎕OR)
  - Gotcha: do NOT work if edited with the APL line editor !
  - [post-Dyalog '17: this restriction will be removed]
  - (∇foo∇ does not use the line editor)

Triggers

```
∇trigger arg
:Implements Trigger myvar
arg.Name,' has value ',⍕⍎arg.Name
∇
```

- Trigger functions must be in the same namespace as the variable that they are related to
- As of 16.0 have Triggers and Global Triggers
- Order of firing is not guaranteed

Triggers

- Standard Trigger
  - `:Implements Trigger varname`
  - acts on one variable in same namespace
  - argument to trigger function has
    - Name, (OldValue) and NewValue
  - relationship is broken when the variable is deleted
    - or shadowed
    - recreate the relationship by refixing the function

Triggers

- Global Triggers (16.0)
  - `:Implements Trigger *`
  - acts on all global variables in same namespace
    - not semi-globals
  - argument to trigger function only has Name
  - can be useful to spot when global variables are inadvertently created

Triggers

- Demo

16.0 Changes

- CONTINUEs no longer dropped
  - except for)CONTINUE
  - can be re-enabled using `2704⌶`

- CONTINUEs no longer automatically)LOADed
  - just another workspace
  - no way of re-enabling this feature

16.0 Changes

- System commands take parameters
  - `)save -force`
    - If it can be saved as that file, it will be, no questions asked
    - No more

```
)save myws
)wsid andys
)save myws
Not saved, this ws is andys.dws
```

    - Doesn't matter what the existing file is, it will be overwritten if permissions allow

16.0 Changes

- System commands take parameters
  - `)si {n} {-tid=tn}`
    - -tid: show for thread tn only

16.0 Changes

-)si n
  - show n lines only

```
      )si
#.koo[1]*
#.joo[1]
#.hoo[1]
#.goo[1]
#.foo[1]
      )si 2
#.koo[1]*
#.joo[1]
      )si ¯2
#.goo[1]
#.foo[1]
```

16.0 Late additions

- Autocomplete no longer ignores invalid input (bug fix)

16.0 Late changes

- ⎕nget now generates a FILE ACCESS ERROR when given a directory to read on all platforms:

```
      ⎕nget'.'
FILE ACCESS ERROR: Unable to open file
      ⎕NGET'.'
     ^
```

16.0 Very late changes

- A few fixes have been made in 17.0 which are most likely to be back-ported to 16.0:
    - ⎕fix (and Editor) will support ∇ in dfn in scripted objects
    - Editor and Tracer now remember if stop bits are set/unset
    - Docked Tracer windows get smaller each time you hit Continue
    - Added Select All to Editor pop-up menu

    - Syntax colouring improvements
    - Editor opens existing object, rather than a new one
    - Less moving around of text when flipping between Editor and Tracer

16.0 Exceedingly late changes

Just joking .. What I meant was

17.0

- 32-bit component files will be read-only AND must be opened with ReadOnly variant
  - **<u>This requires a code change !</u>**

```
      'u:/tmp/32bit'⎕ftie 1
FILE ACCESS ERROR: This file must be converted to a large-span
file or tied using the ReadOnly variant option
      'u:/tmp/32bit'⎕FTIE 1
     ∧
      '32bit'(⎕Ftie ⍠'ReadOnly' 1)111
      '64bit'⎕fcopy 111     ⍝ create 64-bit copy
```

[post-Dyalog '17: the requirement to use the ReadOnly variant is being reviewed and is likely to be removed]

17.0

- Likely to continue to tighten up on the language over time
  - Check that your code is "good" and that you're not relying on the interpreter being too lenient
    - Compiler is more strict; your code may need to change
  - What should we do with things like `⎕FUNTIE ¯1` ?
    - `⎕FUNTIE ¯1` is currently a no-op - should be a `DOMAIN ERROR`; unlikely to change this

The Editor (and Friends)

- Search and Find use PCRE for when regular expressions enabled
- Added new buttons
  - Match Whole Word, RegEx Search, Clear Search String

Changes to locating Shared Libraries/DLLs

- Stripped out complexity, and guessing at where the libraries might be
- If it's non-standard, you must tell the interpreter where to look

Changes to locating Shared Libraries/DLLs

- Windows:
  1. interpreter calls LoadLibrary()
- Non-Windows:
  1. interpreter looks in $DYALOG/lib,
  2. call dlopen()
- Note:

  If your library calls a second library which is not present, *all* operating systems report an error against the first library !

Conga and Shared Libraries/DLLs

- Default path

```
DRC.Init ''       ⍝ Compatibility with Conga 2
Conga.Init ''     ⍝ Conga 3
```

- Override

```
conga_lib←path_to_conga_library
DRC.Init conga_lib
conga_lib Conga.Init''
```

APLMON

- Captures usage of primitives and system functions
- Help determine the order in which Dyalog attempts to improve performance
- See where the time is spent in your application

```
    )xload ws
    2⎕nq'.' 'aplmon' 'myws_mon.csv'
    ⍎⎕lx
    2⎕nq'.' 'aplmon' ''
```

APLMON

- Demo

The end

- andys@dyalog.com