

# Introducing Dyalog version 18.2

Morten Kromberg



# Overview

- Release Dates & Patches
- Basic Licence
- Launching from Text Files:
  - Shell Scripts
  - Text Source
  - LOAD Parameter
- Explorer Integration
- Docker Support
- RIDE v4.4
- Developer Productivity:
  - JSON tables/datasets
  - □ATX metadata one-stop-shop
  - Generating random numbers
  - New user commands



# Release Dates & Windows Patches

- 18.2 was released to DSS Users on March 10<sup>th</sup>
  - DSS = Dyalog Support Service
- Every one else by Weds March 23<sup>rd</sup>
  
- Updates to be delivered as Microsoft Patch files (MSP) under Windows
  - Allows patching ALL components of Dyalog installation including APL code & docn
  - (was already the case on all other platforms)



# Basic Licence

Intended for

- ◆ non-commercial use
- ◆ education
- ◆ personal projects
- ◆ experiments
- ◆ sharing your experience
- ◆ proof of concepts / trials
- ◆ participating in programming competitions for cash prizes
- ◆ fun



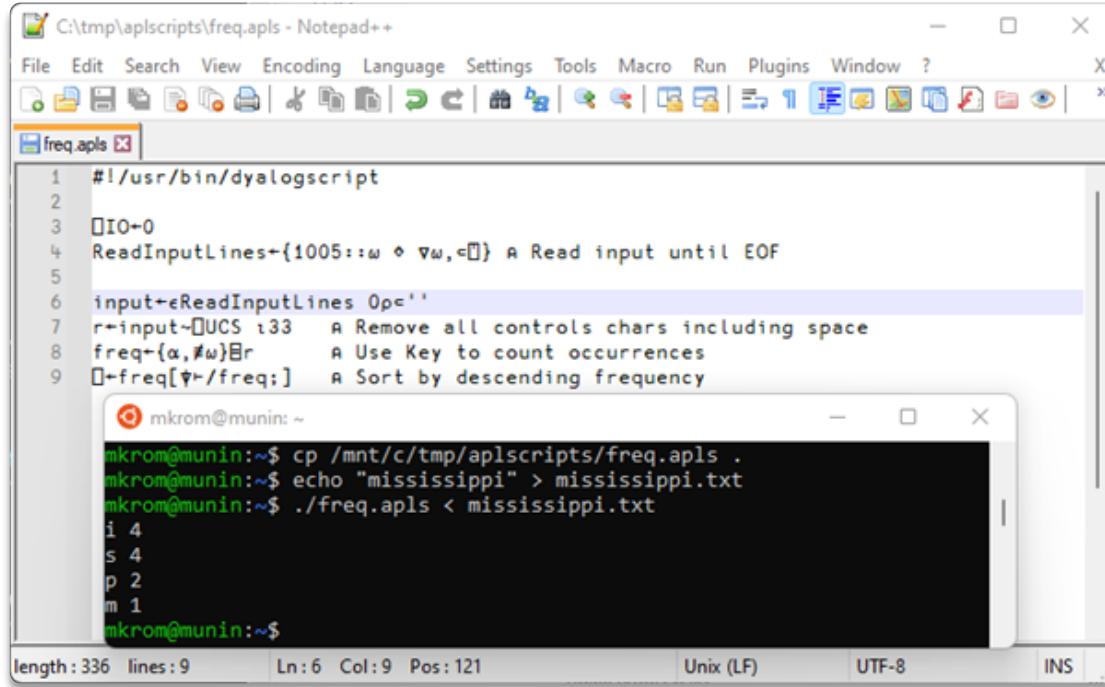
# Basic Licence

- ◆ Replaces *non-commercial licence*
- ◆ Allows distribution of Dyalog along with your work, under the default royalty licence
  - ◆ Fee is 2% of gross APL-based revenue
  - ◆ No fees due if revenue < GBP 5,000 in a calendar year
  - ◆ Multiple alternative commercial licence schemes are available
- ◆ For GBP 150 per year, you can subscribe to the Dyalog Support Service (DSS)



# Launching from Text Files:

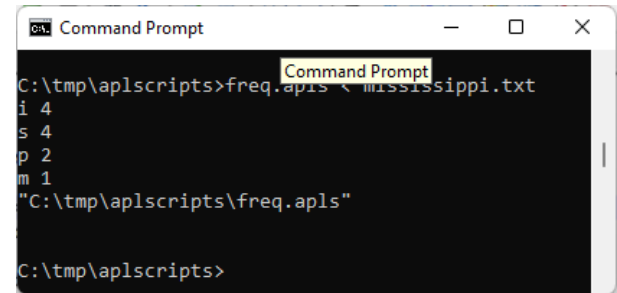
## Shell Scripts



```
C:\tmp\aplscripts\freq.apls - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
freq.apls
1 #!/usr/bin/dyalogsript
2
3 []IO-0
4 ReadInputLines+{1005::w ♦ ∇w, c[]} A Read input until EOF
5
6 input+eReadInputLines Op<'
7 r+input~[]UCS i33 A Remove all controls chars including space
8 freq+{α, #w}[]r A Use Key to count occurrences
9 []+freq[∇~/freq:] A Sort by descending frequency

mkrom@mumin: ~
mkrom@mumin:~$ cp /mnt/c/tmp/aplscripts/freq.apls .
mkrom@mumin:~$ echo "mississippi" > mississippi.txt
mkrom@mumin:~$ ./freq.apls < mississippi.txt
i 4
s 4
p 2
m 1
mkrom@mumin:~$
length: 336 lines: 9 Ln: 6 Col: 9 Pos: 121 Unix (LF) UTF-8 INS
```

- /usr/bin/dyalogsript is the script interpreter
- Under Windows, .apls files are associated with dyalogsript



```
Command Prompt
C:\tmp\aplscripts>freq.apls < mississippi.txt
i 4
s 4
p 2
m 1
"C:\tmp\aplscripts\freq.apls"

C:\tmp\aplscripts>
```



# Launching from Text Files:

## Text Source – Link 3.0

- ◆ Completely new documentation  
<https://dyalog.github.io/link/3.0/>
- ◆ Many small features and bug fixes  
<https://dyalog.github.io/link/3.0/ChangeHistory/>  
<https://github.com/Dyalog/link/issues?q=is%3Aissue+is%3Aclosed>

Most significant new features:

- ◆ )SAVE workspaces with active links and resynchronise on Load
- ◆ Support for system variable settings in source files
- ◆ Cleaner, documented API



Link User Guide

Overview

Introduction

Technical Details and Limitations

Workspaces

History of source files as text in Dyalog

Install and Upgrade

Installation

Upgrading to Link 3.0

Change History

Working with Link

Basic Usage

Setting Up Your Environment

Converting an Existing Workspace to use Link

Migrating from SALT to Link

API & Command Reference

API Overview

Link.Add

Link.Break

Link.CaseCode

Link.Create

# Introduction

*Link* enables users of Dyalog to store their APL source code in text files. This is the documentation for Link Version 3.0, which will be released in the autumn of 2021 and included with the next release of Dyalog APL. If you have an earlier version of APL or Link, you might want to read one or more of the following pages before continuing:

- **Link version 2.0** If you are actually looking for documentation of the version which was distributed with Dyalog APL versions 17.1 and 18.0.
- **Migrating to Link 3.0 from Link 2.0:** Dyalog recommends migrating to version 3.0 at your earliest convenience.
- **Migrating to Link 3.0 from SALT:** If you have APL source in text files managed by SALT that you want to migrate to Link.
- **Installation instructions:** If you want to download and install Link from the GitHub repository rather than use the version installed with APL, for example if you want to use Link 3.0 with Dyalog version 18.0.
- **The historical perspective:** Link is a step on a journey which begins more than a decade ago with the introduction of SALT for managing source code in text files, as an alternative to binary workspaces and files, and will hopefully end with the interpreter handling everything itself.

## Audience

Table of contents

Audience

What is Link?

Link is NOT...

Link fundamentals

Functions vs. User Commands

    User commands

    API functions

Further reading

Frequently Asked Questions



# Launching from Text Files:

## LOAD Parameter

Supported in 18.0:

- Function file
- Namespace file
- Class file

Windows:

```
dyalog.exe LOAD=/myapp/foo.aplf
```

Linux:

```
LOAD=/myapp/foo.aplf dyalog
```



# Launching from Text Files:

## LOAD Parameter

### Supported in 18.0:

- Function file
- Namespace file
- Class file

### Windows:

```
dyalog.exe LOAD=/myapp
```

### Linux:

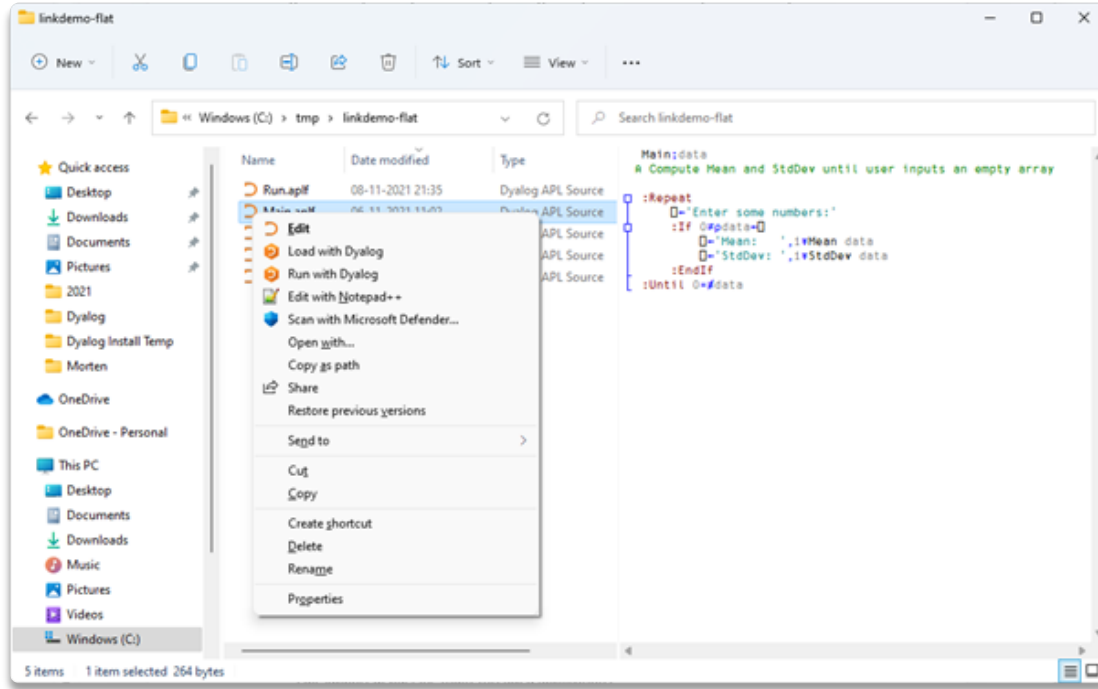
```
LOAD=/myapp dyalog
```

### New in 18.2:

- Namespace directory
- Calls function Run unless `-x` set



# Windows Explorer Integration



Context menu:

- Edit
- Load with Dyalog
- Run with Dyalog

]FileAssociations  
user command has been  
enhanced



```
]fileAssociations status
```

```
— Status for ALL users —
```

```
Installed instances are:
```

```
18.2U64 18.2C32 18.0U64 17.1U64 17.1C32 16.0U64
```

```
Current associations:
```

Directories	18.2C32	(menu items shown)
Source Preview	18.2U64	(enabled)
Workspace Preview	18.2U64	(enabled)
dcfg	18.2C32	default action is Edit
dws	18.2C32	default action is Run
dyalogscript	18.2C32	default action is Run
dyapp	18.2C32	default action is Run
source	18.2C32	default action is Edit

```
]fileAssociations 182U64
```

```
Selected instance: 18.2 Unicode-64
```

```
Backup written to C:/Users/mkrom/AppData/Local/Temp\efa-b  
0 deleted, 33 set. User = all.
```

## Edit File Associations

## Current associations for ALL users:

Scripts	(.apls)	18.2U64
Sources	(.apl?, .dyalog)	18.2U64
SALT apps	(.dyapp)	18.2U64
Workspaces	(.dws)	18.2U64
Directories		18.2U64
Preview		18.2U64

## Select instance to associate:

- 18.2 Unicode-64
- 18.2 Classic-32
- 18.0 Unicode-64
- 17.1 Unicode-64
- 17.1 Classic-32
- 16.0 Unicode-64

- Show Windows Explorer context menu items for directories
- Create a backup of the relevant registry settings

Set associations for: All users


Apply

Cancel

# Docker Support

```
mkrom@munin:~$ cat /mnt/c/tmp/helloworld/helloworld.aplf
helloworld
@-'Hello World!'
```

```
mkrom@munin:~$ docker run -v /mnt/c/tmp/helloworld:/app dyalog/dyalog
```



```
https://www.dyalog.com

found aplf file Launching with LOAD=/app/helloworld.aplf
Dyalog APL/S-64 Version 18.2.45298
Serial number: UNREGISTERED - not for commercial use
+-----+
| Dyalog is free for non-commercial use but is not free software.
| A basic licence can be used for experiments and proof of
| concept until the point in time that it is of value.
| For further information visit
| https://www.dyalog.com/prices-and-licences.htm
+-----+
Fri Mar  4 14:29:33 2022
Loaded: #.helloworld from "/app/helloworld.aplf"
Hello World!
```

- Launch containers on source files
- Public containers under Basic Licence:

If you publish containers derived from the public Dyalog containers you **MUST** include our LICENSE file in a prominent location

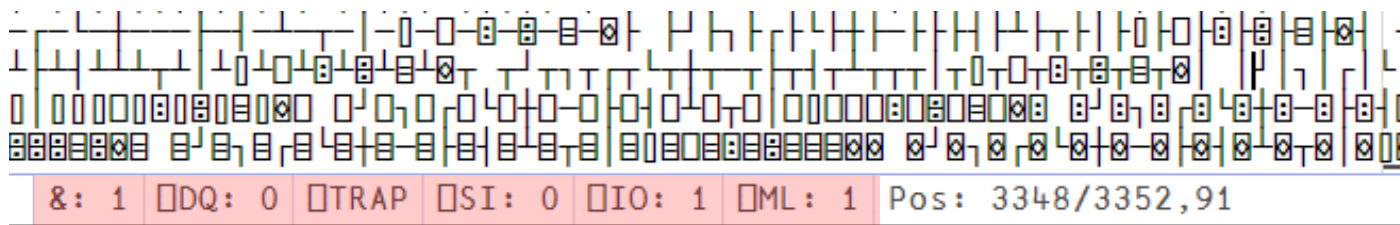
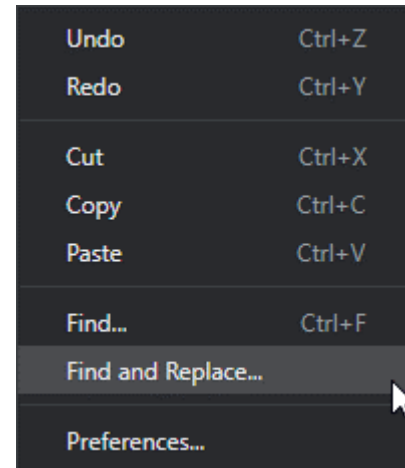
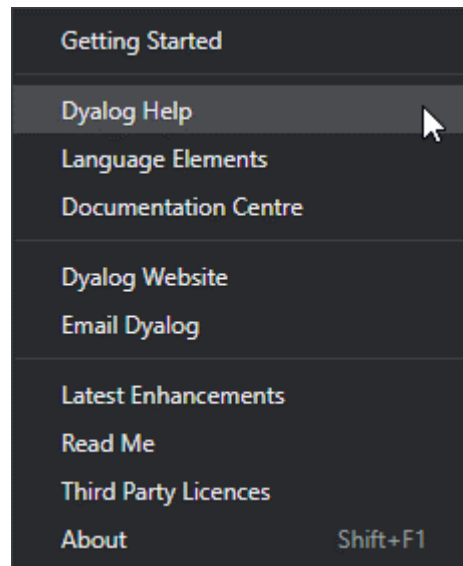
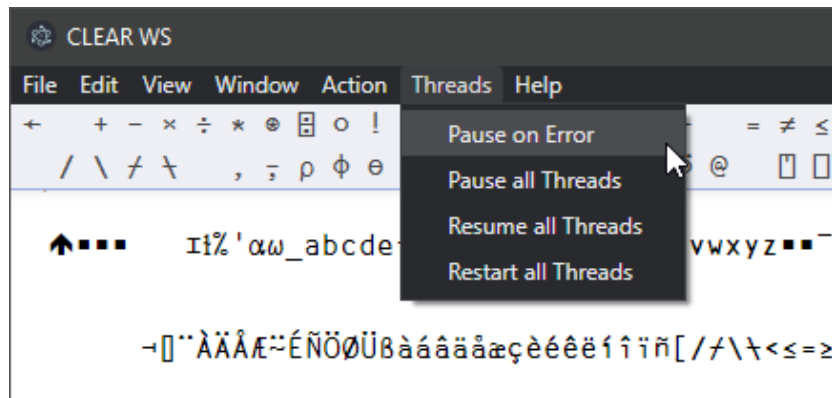


# RIDE v4.4

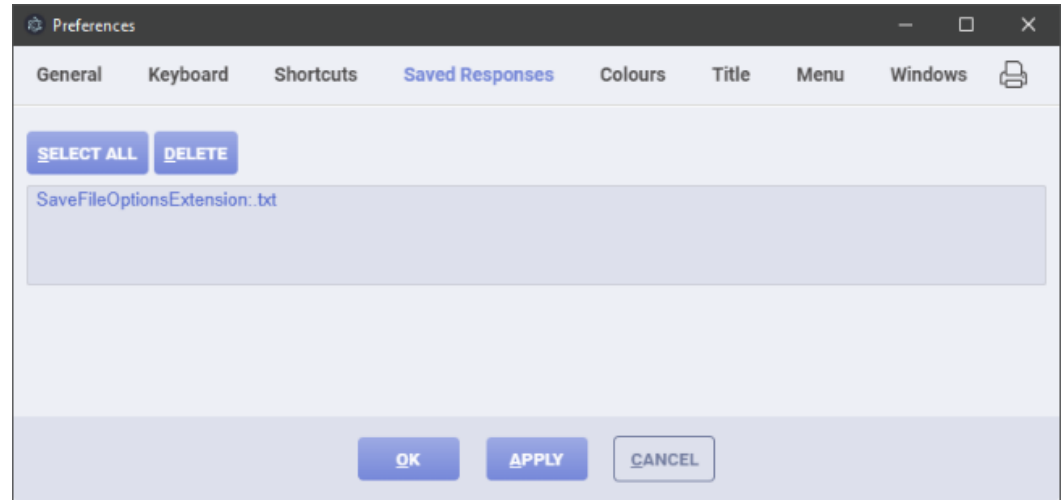
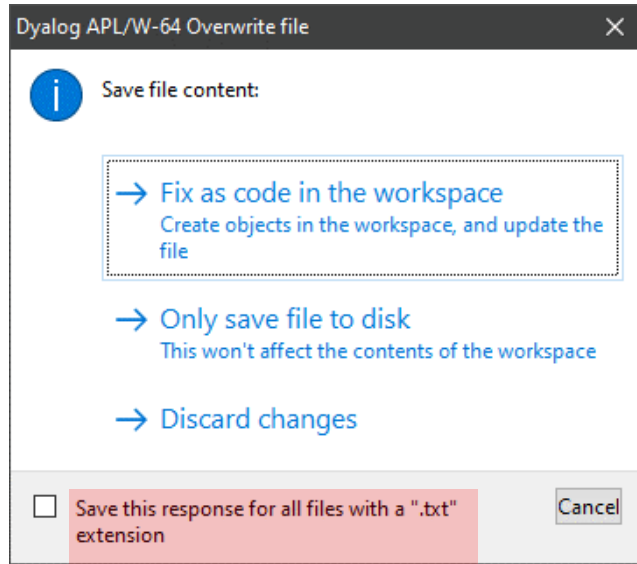
- ◆ Debug multi-threaded applications
- ◆ Language bar and help adapts to Dyalog version
- ◆ Save responses to prompts
- ◆ ... and many more tweaks and fixes, see <https://github.com/Dyalog/ride/milestone/4?closed=1>



# RIDE 4.4 Highlights



# RIDE 4.4 Highlights





# Developer Productivity:

## JSON tables/datasets

```
fields ← 'Item' 'Price' 'Qty'  
items  ← 'Knife' 'Fork'  
price  ← 3 4  
qty    ← 23 45  
fields;↑items price qty
```

Item	Price	Qty
Knife	3	23
Fork	4	45

```
cols ← items price qty  
table ← c4(cols fields)  
(⎕JSON⎕'Compact'0)table
```

```
[  
  {  
    "Item": "Knife",  
    "Price": 3,  
    "Qty": 23  
  },  
  {  
    "Item": "Fork",  
    "Price": 4,  
    "Qty": 45  
  }  
]
```



# Developer Productivity:

## □ ATX metadata one-stop-shop

### ◆ Unifies:

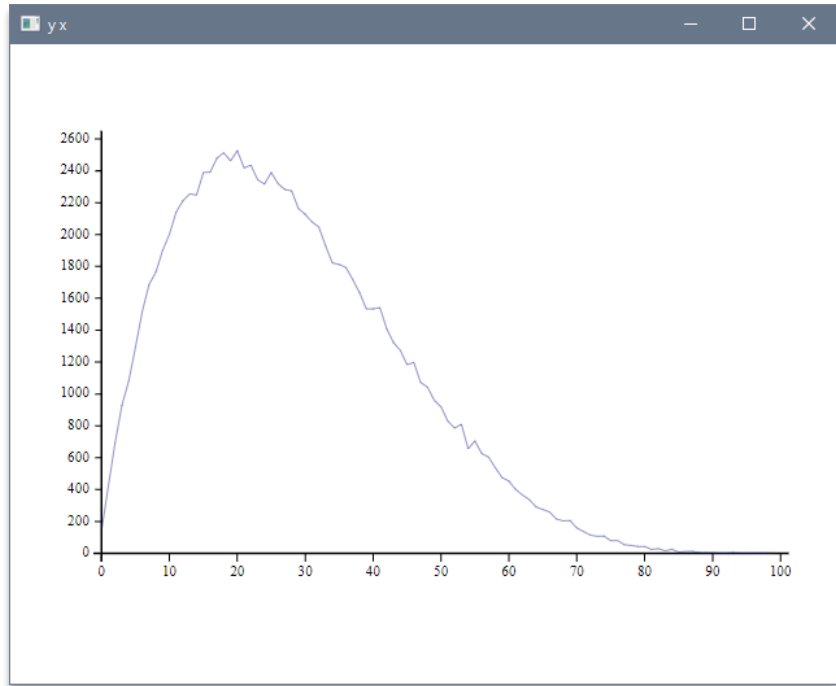
- ◆ □ SIZE Size of Object
- ◆ □ NC Name Classification
- ◆ □ NR Nested Representation
- ◆ □ SRC Source
- ◆ □ AT Attributes
- ◆ 5179I Loaded Object Info

### ◆ Additional metadata:

- ◆ Last edit UTC
- ◆ Source "as typed"
- ◆ Normalised source
- ◆ Best available source
- ◆ More to come...



# Developer Productivity: Generating random numbers

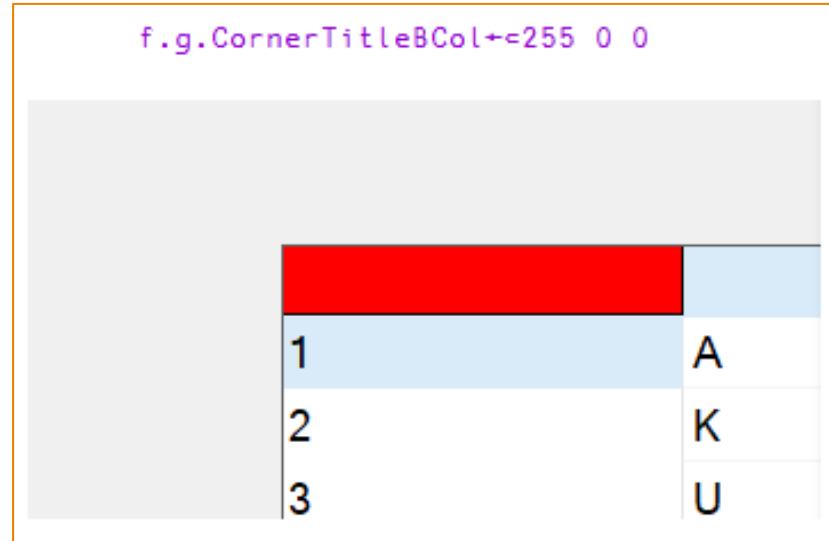


```
Dist ← 16808I
val ← 2 5 Dist 'Beta' 1e5
res ← 100
bin ← [res×val
x ← -1+ιres
y ← {≠ω}⊖x,bin
]plot y x
```



# New Win32 GUI (GWC) features!!!

- CornerTitleBCol on Grid
- FireOnce on Timer



# Developer Productivity:

## New user commands

```
]get github.com/mkromberg/linkdemo/tree/main/stats
```



# Developer Productivity:

## New user commands

```
]get github.com/mkromberg/linkdemo/tree/main/stats
```

Working on it...



# Developer Productivity:

## New user commands

```
]get https://github.com/mkromberg/linkdemo/tree/main/stats
```

Working on it.....

```
#.stats
```

```
stats.inds←113
```

```
]names 0
```

```
2: bin res val x y    3: Dist    9: stats
```

```
]names stats
```

```
stats.Mean stats.Root stats.StdDev
```



# Developer Productivity:

## New user commands

```
]aplcart general mean
```

```
X,Y,Z:any M,N:num I,J:int A,B:Bool C,D:char f,g,h:fn ax:axis s:scal v:vec m:mat
```

---

```
Ms( $\frac{1}{N} \sum_{i=1}^N x_i$ )N Generalised mean
```

```
Ms( $\frac{1}{N} \sum_{i=1}^N x_i^\alpha$ )N Generalised mean
```

---

Showing 2 of 2 matches

```
GM←( $\frac{1}{N} \sum_{i=1}^N x_i^\alpha$ )1/α
```

```
1 GM 1 2 3 4 A Average
```

2.5

```
2 GM 1 2 3 4 A Quadratic Mean
```

2.738612788

```
-1GM 1 2 3 4 A Harmonic Mean
```

1.92





# Developer Productivity:

## New user commands

```
]repr (3 4 p 12) -format=json  
[[1,2,3,4],  
 [5,6,7,8],  
 [9,10,11,12]]
```

```
]repr (3 4 p 12) -format=xml  
<array>  
  <shape>3 4</shape>  
  <list>1 2 3 4 5 6 7 8 9 10 11 12</list>  
</array>
```



# Developer Productivity:

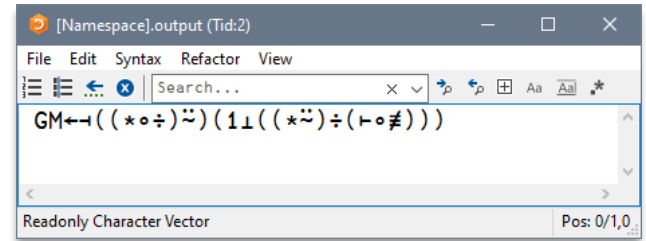
## New user commands

```
]repr (3 4ρι12) -format=apl  
(3 4ρ1+□io-~ι12)
```

```
GM←←*ο÷~1⊥*~÷†ο≠
```

```
]repr GM  
GM←←(( *ο÷)~)(1⊥(( *~)÷(†ο≠)))
```

```
]view ]repr GM
```



# Summary

- Release Date & Patches
- Basic Licence
- Launching from Text Files:
  - Shell Scripts
  - Text Source
  - LOAD Parameter
- Explorer Integration
- Docker Support
- RIDE v4.4
- Developer Productivity:
  - JSON tables/datasets
  - □ATX metadata one-stop-shop
  - Generating random numbers
  - New user commands

