



Mainz, April 30th 2024



Migrating to Dyalog

Morten Kromberg, CTO

A New Wave of Migrants

- ◆ Dyalog APL was created by Dyadic Systems Ltd in 1981
 - ◆ To replace mainframe APL systems
- ◆ Almost all users of Dyalog APL are migrants
 - ◆ from SHARP APL, IBM APL2, APL+Win, APLX, or DEC APLSF, or ...
- ◆ Waves of migrants
 - ◆ Death of mainframes and minicomputers (1980's)
 - ◆ Superior support for Windows GUI (1990's)
 - ◆ Now, "the cloud" (& a few more mainframes being shut down)



Why migrate to Dyalog?

- ◆ Dyalog re-invests 90-95% of revenues in
 - ◆ Enhancing APL core technology
 - ◆ Creating tools for APL developers
 - ◆ Marketing APL outside the current APL community
- ◆ Combined revenues of products and services based on Dyalog APL exceeds \$1Bn per year – and is growing



Selected Features added 2006-2024

- Web Server and Web Service Frameworks
- Run APL as a Windows Service
- Public Docker Containers
- Remote IDE for debugging service processes
- Health Monitor for monitoring multiple processes
- Parallel and Asynchronous Execution
- New Data Types:
 - 128-bit Decimal Floating Point
 - Complex Numbers
- Functional Programming (dfns)
- New primitives: Key, Stencil, Where, ...
- Significant steps towards an APL compiler
- Many speed-ups of interpreter algorithms
- Object Orientation
- Microsoft.Net Integration
- HTMLRenderer object embeds Chromium Web Browser engine
- 64-bit: *NO* workspace or component file size limits
- Unicode Support, APL Source in Text Files
- Secure TCP Sockets w/ IPv6 Support
- Encryption Toolkit
- Regular Expressions (PCRE) built-in to APL
- XML and JSON parsers for fast conversion to (and from) APL structures

Most features identical across all platforms

Migrating to Dyalog



APL Germany e.U.

Spring '24

Why Migrate to Dyalog?

Developer tools are free, cross-platform and mostly open source:

Name	Description
SQAPL	ODBC Interface (also ADO and ADO.NET)
Jarvis	HTTP/JSON and REST service framework
HttpCommand	HTTP client
SAWS	SOAP service framework
Conga	TCP and UDP layer
SharpPlot	Business and Technical graphics
<input type="checkbox"/> XML, <input type="checkbox"/> JSON, <input type="checkbox"/> CSV	Built in to interpreter

Name	Description
RConnect	Interface to R
MiServer / DUI	Web Application Framework
Docker Containers	Published examples
Link	Interface to source code management
APLProcess	And isolates

Why Migrate to Dyalog?

Emerging Tools

Name	Description
Cider	Project Management
Tatin	Package Manager
NuGet	Interface to .NET Packages
Selenium	Automated GUI testing
Jupyter	Jupyter notebooks containing APL
eWC	JavaScript emulation of Win32 GUI
Arrow & Parquet	Data Science data formats

Separately Licensed Tools

Name	Description
DFS	Dyalog File Server ("SHAREFILE")
Static Analysis	Static Analysis of APL Code (code linting and vulnerability detection) Planned for 2025

Why Migrate to Dyalog?

Dyalog is 100% Cross Platform

- ◆ Born under UNIX (Solaris, AIX, ...)
 - ◆ Ported to DOS, Windows, Linux (ARM, Intel), MacOS (Intel, Mx)
- ◆ Single source for all platforms
 - ◆ Workspaces and component files compatible across all platforms
- ◆ All tools are tested on all platforms
 - ◆ Exceptions where O/S does not provide a feature
 - ◆ .NET not under AIX, many Windows-only features like DDE, COM/OLE

Remote IDE (RIDE)

- Connect to and debug Dyalog APL running on any platform
- From Windows, Linux or MacOS
- Or indeed a browser running anywhere...
 - Interpreter can serve up RIDE as a web app

RIDE running in a browser

The screenshot shows a web browser window with the address bar displaying `/opt/mdyalog/19.0/64/unicode` and `wsl:8888`. The browser's address bar also shows "Not secure". The browser's menu bar includes "Edit", "View", "Window", "Action", "Threads", and "Help". The browser's toolbar shows various navigation and utility icons. The main content area displays a terminal window with the following text:

```
)load dfns
/opt/mdyalog/19.0/64/unicode/ws/dfns.dws A saved Mon Jan 29 11:5

An assortment of D Functions and Operators.

tree #           A Workspace map.
↑~10↑↑attrib [n] 3 4 A What's new?
↵notes find 'Word' A Apropos "Word".
[ed]notes.contents' A Workspace overview.

)sh uname -a
Linux munin 5.15.146.1-microsoft-standard-WSL2 #1 SMP Thu Jan 11

)ed cal
```

On the right side of the terminal window, there is a vertical list of function names and their descriptions, such as "A Calend", "A includ", "A centre", "A relati", "A relati", "A each m", "A month", "A centre", "A calend", "A month", "A day-of", "A 1st qu", "A 2nd", "A 3rd", "A 4th", "A month", "A year a", "A epoch", and "A 31 mon".

At the bottom of the terminal window, there is a status bar showing system information: `&: 1 [DQ: 0 [TRAP [SI: 0 [IO: 1 [ML: 1 Pos: 0/25,1`

Why Migrate to Dyalog?

Dyalog is "Cloud Ready"

- ARM and Intel Linux versions
- Public Docker containers
- Remote IDE
- Text-based source supports "Continuous Integration"
 - Build & deploy containers on commit or push
- User community starting to gain significant experience
- Working on tools to port Windows GUI to HTML/JS

Why Migrate to Dyalog?

- ◆ Dyalog APL is fast!
- ◆ Core algorithms regularly updated to take advantage of new hardware and new theory
- ◆ Research into a compiler continues

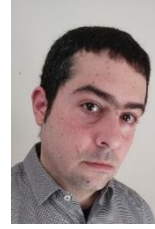
Why Migrate to Dyalog?

Dyalog APL is carefully designed to last. For example:

- ◆ Dyalog APL is tightly integrated with .NET
 - ◆ ... and still supports the old .NET Framework
 - ◆ However, Dyalog APL does not and WILL NOT depend on .NET
 - ◆ It also runs under IBM AIX, where .NET does not exist

- ◆ Dyalog **will** remain very portable and independent of "temporary" frameworks

The Real Reason to Pick Dyalog APL



HOW to Migrate to Dyalog APL

1. From IBM / Logon APL2
2. From APL+Win or MicroAPL APLX

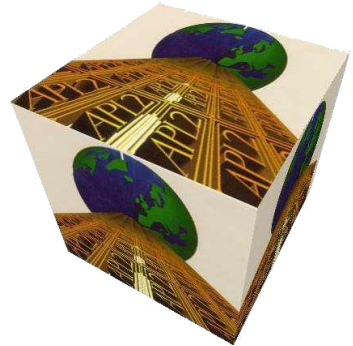
From APL2

Relatively straightforward

- A few language differences
- User Interfaces and file I/O are usually handled by simple cover-functions and possible to emulate automatically
- Linux or Windows apps may be making external calls which will require "tweaking"
- Considering implementing "format by example" but so far it has not been necessary

```
'555-5555' ⍋ nums
```

- Easy to model in APL if necessary



Recent / Active APL2 Migrations

- ◆ Insurance company
 - ◆ No UI, manipulates text and Excel files
 - ◆ Handled by European Consulting Partner
- ◆ Sandvik (Sweden) – in progress: Mainframe APL2 direct to Docker Containers and HTML/svg
 - ◆ Handled by Tiamatica in Malmö (Gilgamesh Athoraya)
- ◆ BIG Jewellers: Windows
 - ◆ Handled by customer "with a little help"
- ◆ Two more under discussion
 - ◆ (Germany, Canada)

Migrated APL2 Mainframe UI

```
Locate Sort
CAPP/COR TEST ----- Routine definition - variables ----- 23-11-10 13:00

Routine.....: X802WM Saved: 23-10-05 12:04 by: STC
Description...: TEST AV SOAP GETLANGS WEBSERVICE
Open for enhanced dialog: Y Yes/No
Prompt variable that contains the information "grade": _____
Var. Cha
Name Num Length Type Send Explanation Line 20 of 99
ART C L X ARTICLE
BART C L ARTICLE
BB C L A DUMMY
CA C L CHARACTER DUMMY
CA1 C L DUMMY
CA2 C L X DUMMY
CA3 C L X DUMMY
CA4 C L A DUMMY
CA5 C L DUMMY
CA6 C L DUMMY
CB C L CHARACTER DUMMY
CC C L CHARACTER DUMMY
CD C L X CHARACTER DUMMY
CE C L A CHARACTER DUMMY
CF C L CHARACTER DUMMY
CG C L CHARACTER DUMMY
CH C L A CHARACTER DUMMY
CHA C L CHARACTER DUMMY
CHA1 C L --
CHA2 C L DUMMY

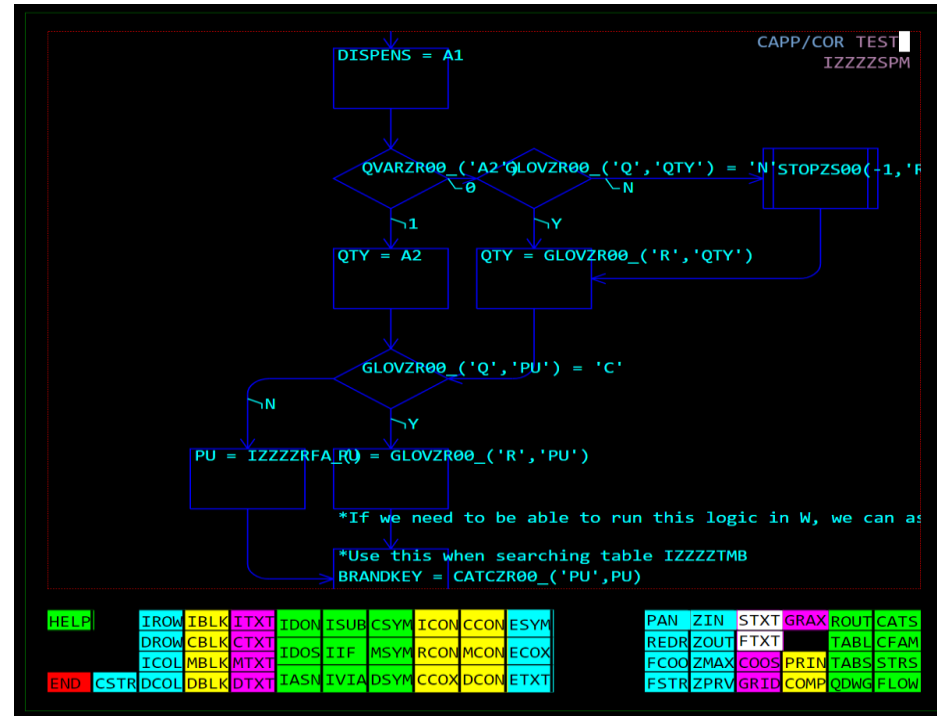
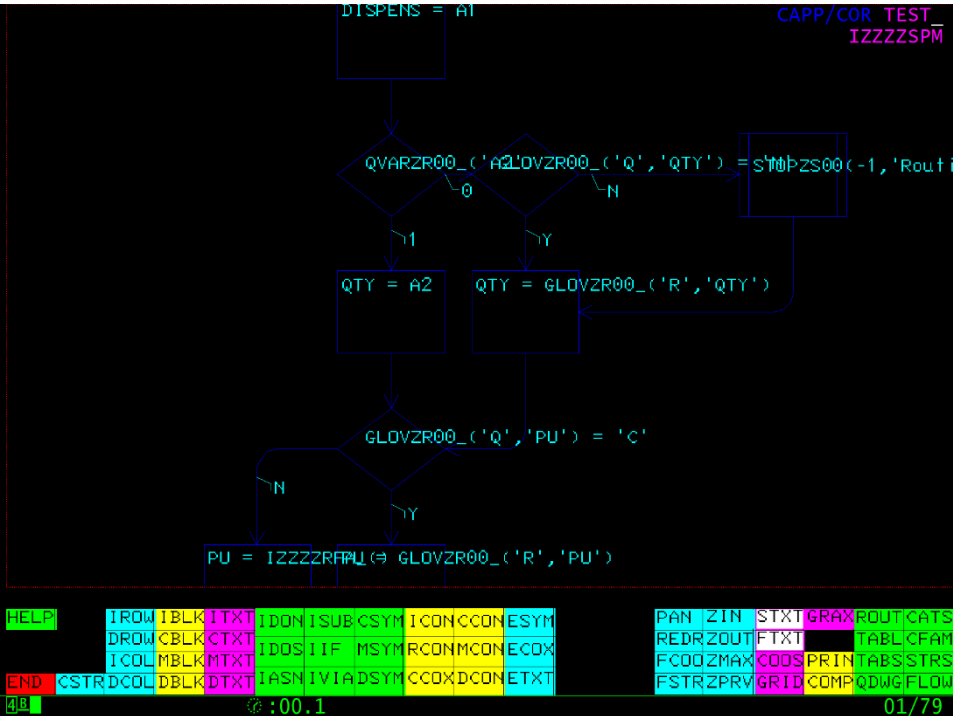
F1=Help F3=End F6=Prompt F7=Up F8=Down
```

```
Locate Sort
CAPP/COR TEST ----- Routine definition - variables ----- 23-11-10 13:04

Routine.....: X802WM Saved: 23-10-05 12:04 by: STC
Description...: TEST AV SOAP GETLANGS WEBSERVICE
Open for enhanced dialog: Y Yes/No
Prompt variable that contains the information "grade": _____
Var. Cha
Name Num Length Type Send Explanation Line 20 of 99
ART C L X ARTICLE
BART C L ARTICLE
BB C L A DUMMY
CA C L CHARACTER DUMMY
CA1 C L DUMMY
CA2 C L X DUMMY
CA3 C L X DUMMY
CA4 C L A DUMMY
CA5 C L DUMMY
CA6 C L DUMMY
CB C L CHARACTER DUMMY
CC C L CHARACTER DUMMY
CD C L X CHARACTER DUMMY
CE C L A CHARACTER DUMMY
CF C L CHARACTER DUMMY
CG C L CHARACTER DUMMY
CH C L A CHARACTER DUMMY
CHA C L CHARACTER DUMMY
CHA1 C L --
CHA2 C L DUMMY

F1=Help F3=End F6=Prompt F7=Up F8=Down
```

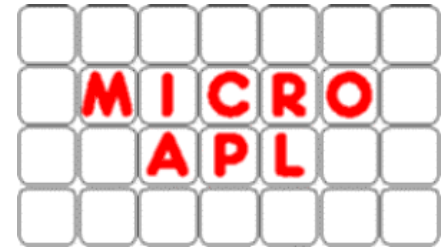
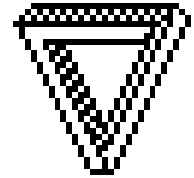
Migrated APL2 Mainframe UI



From APL+Win or MicroAPL APLX

Same language differences as APL2, plus:

- ◆ Many system functions & control structures not found in Dyalog APL
- ◆ Double quotes ("Don't do this!")
- ◆ Advanced Graphical User Interfaces
- ◆ Calls to external libraries



APLX Migrations

- MicroAPL stopped developing APLX in 2016
 - Dyalog hosts a download of the last free version
- Dyalog developed migration tools in 2016

Recent / Active APL+Win Migrations

- ◆ Two European Insurance companies
 - ◆ One with GUI, completely rewritten in Dyalog APL, the other a pure service converted to Jarvis in Linux containers
 - ◆ Handled by a European consulting partner
- ◆ METSIM® - in progress
 - ◆ Dyalog engaged to perform this migration
 - ◆ Will be used to develop tools to automate migration, including the Graphical User Interface
- ◆ Met one more potential migrant last week

Differences which are "easy"

/ is sometimes a function in Dyalog APL

```
1 0 1 / '' 'ABC' 'DEF' 'GHI'
```

Translate /'' to ◦/''

```
'AC' 'DF' 'GI' vs  
'ABC' 'GHI'
```

←Y

Translate ← to {}

Not supported in Dyalog

Other "Easy" Differences

□ XLIB

System function not in Dyalog

```
R←ΔXLIB X
X,←' '*↓~≠X
:If 0∈ρR←↑⇒□NINFO□1←X
  :If v/'?*'∈X
    R←0 0ρ' '
  :Else
    'XFHOST ERROR FindFirstFile 1 0 3 The system cannot find the path specified.'
    □SIGNAL 22
  :EndIf
:Else
  R←R[ΔR;]
:EndIf
```

Tricky Differences

`A B[I]`

`f.g` when `f` or `g`
are not scalar functions

Fortunately, these are very rare in practice

`:LeaveIf`

`A (B[I])` or
`(A B)[I]` ?

Detect and rewrite

Enhance Interpreter

The Hard Parts

- ◆ Component Files
- ◆ User Interfaces (especially Graphical)
- ◆ Other I/O (e.g. SQL Databases)
- ◆ External Library Calls

Component Files

- ◆ We will develop an APL+Win COM server to read/write APL+Win component files directly from Dyalog APL
- ◆ Avoid the need for "big bang" data migrations
 - ◆ Component files can be migrated over time

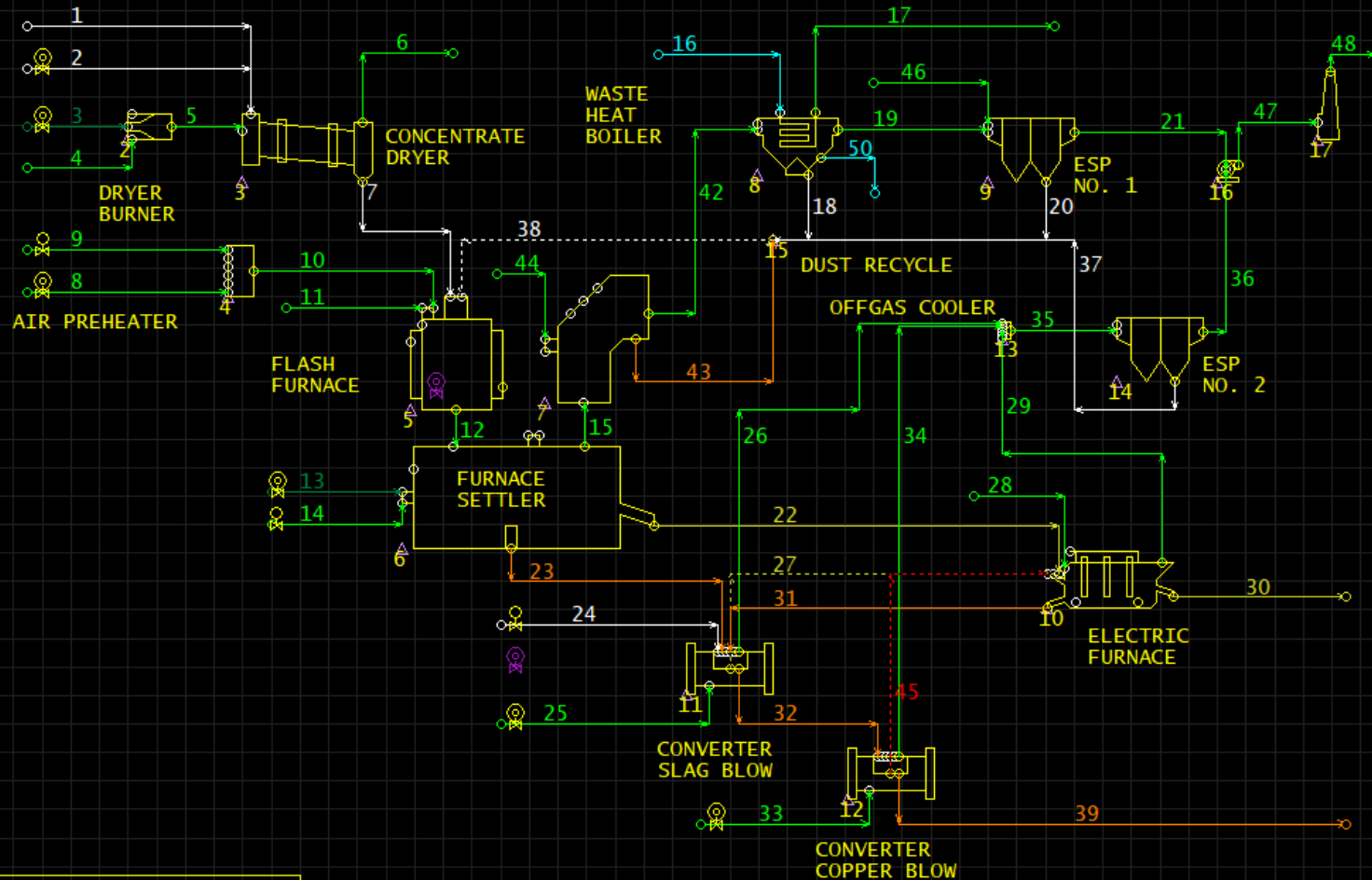
The Elephant in the Room: OWI ?

Dyalog is building an emulator to support the METSIM[®] migration

- ◆ Our goal is that no significant changes to application code will be required
- ◆ METSIM[®] screen shots follow
 - ◆ Many thanks to Alex Holtzapple, CEO of MSI

FLASH SMELTER EXAMPLE

Stream Number



Stream 322

Output Level: 0 Design Factor: 0 Maximum Flow: 0
 Box Number: 0 Variables 1 2 3

322 SI LI
 IR Slurry Label GC OK Cancel

	MI/DY		Wt. Frac.	gpl	MI/DY		Wt. Frac.	gpl	MI/DY
SOLIDS	2819.6251	aH2O	0.8245512	977.40919	17055.895	H	1	0.0922685	109.37361
SLD-ORG	0	aH2SO4	0.0000101	0.012	0.2094012	C	6	0	0
AQUEOUS	20685.064	aH2CO3	0	0	0	N	7	0	0
ORGANIC	0	aNaSO4	0.0119156	14.124588	246.47556	O	8	0.8235821	976.26044
MOLTEN	0	aCoSO4	0.0004037	0.4786549	8.3525794	Na	11	0.0000077	0.0091656
MAITE	0	aCo2(SO4)	0	0	0	Mg	12	0.0309770	36.719704
SLAG	0	aFeSO4	0.0000014	0.00167	0.0291416	Al	13	0.0002659	0.3152151
GAS	0	aFe2SO43	0.0002499	0.2962831	5.1701727	Si	14	0	0
TOTAL	23504.689	aAl2SO43	0.0016860	1.9985948	34.875693	S	16	0.0457426	54.222563
% SOLID	0.1199601	aCa(OH)2	0	0	0	Cl	17	0	0
Contrl C	0	aCaSO4	0.0015753	1.8674523	32.587241	Ca	20	0.0004637	0.5497767
Temp C	25	aCr2SO43	0.0000057	0.0067845	0.1183903	Sc	21	0.0006129	0.7265744
Temp F	77	aCuSO4	0.0004087	0.4845496	8.4554426	Cr	24	0.0000015	0.0017990
Pres kPa	101.325	aMgSO4	0.1533737	181.80664	3172.5452	Mn	25	0.0011638	1.3795873
Pres kPag	0	aMnSO4	0.0031988	3.7918587	66.168338	Fe	26	0.0000703	0.0833716
Pres psia	14.695949	aNaCl	0	0	0	Co	27	0.0001535	0.1819971
Pres psig	0	aNa2CO3	0	0	0	Ni	28	0.0045199	5.3579245
Time	1	aNa2SO4	0.0000238	0.0283146	0.4940931	Cu	29	0.0001627	0.1929071
Gal/min	3408.8166	aNaOH	0	0	0	Zn	30	0.0000072	0.0086240
L/sec	215.06279	aSc2(SO4)	0.0025775	3.0553814	53.316732				
L/min	12903.767	aZnSO4	0.0000179	0.0212951	0.3716020				
M3/hr	774.22605	aSO4-	0	0	0				
NM3/hr	772.28958	aNH3	0	0	0				

Practicing

- ◆ APL+WIN comes with a handful of GUI demonstration applications

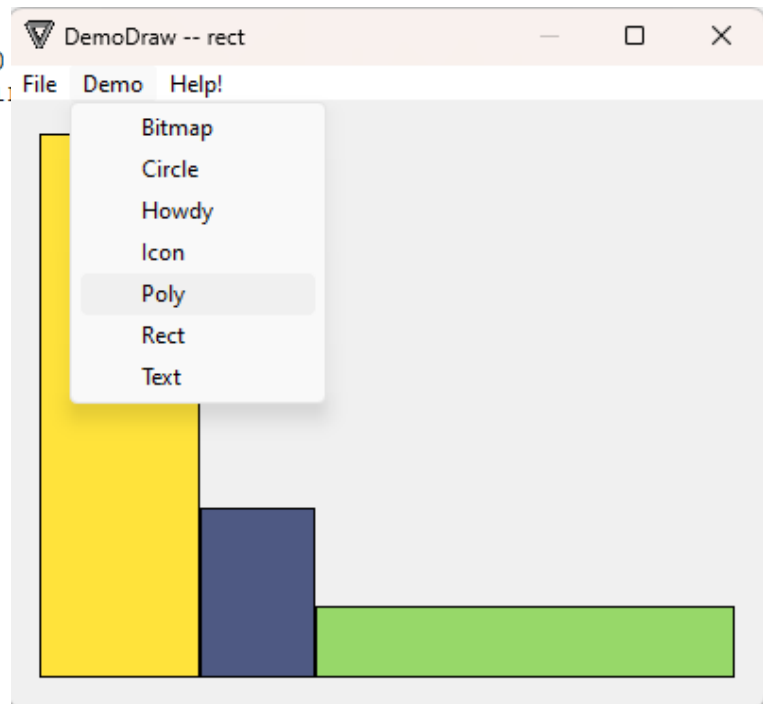
My copy of APL+Win is a little dated...

```
)load Examples\DEMODRAW  
C:\APLWIN11\EXAMPLES\DEMODRAW SAVED 12. september 2010  
This DEMODRAW workspace is distributed with the APL+Win.  
It contains a demonstration of the Draw method.  
Run:
```

```
DemoDraw
```

```
Copyright 2005-2006, APLNow LLC.
```

```
DemoDraw
```



Now export the APL+Win code to APL Transfer Format

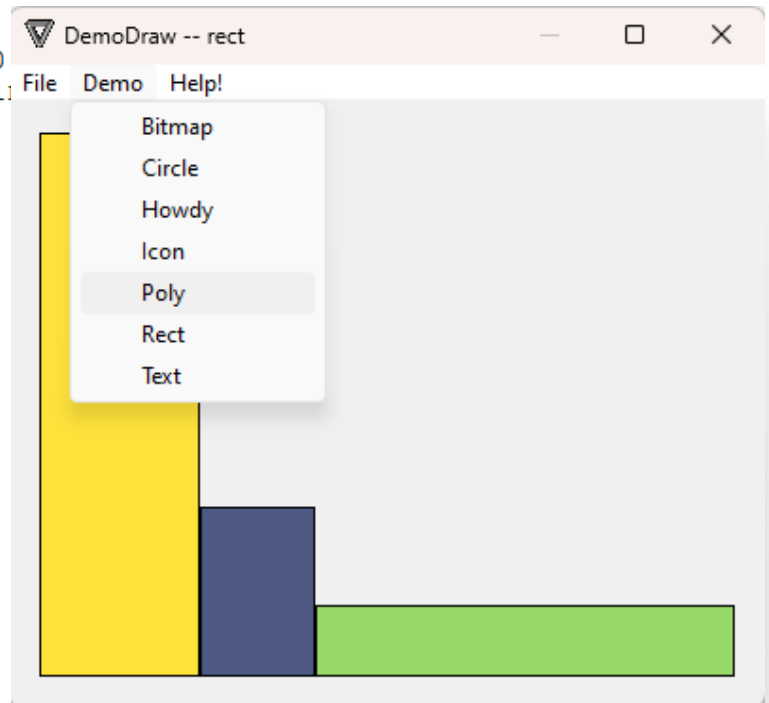
```
)load Examples\DEMODRAW  
C:\APLWIN11\EXAMPLES\DEMODRAW SAVED 12. september 2010  
This DEMODRAW workspace is distributed with the APL+Win.  
It contains a demonstration of the Draw method.  
Run:
```

```
DemoDraw
```

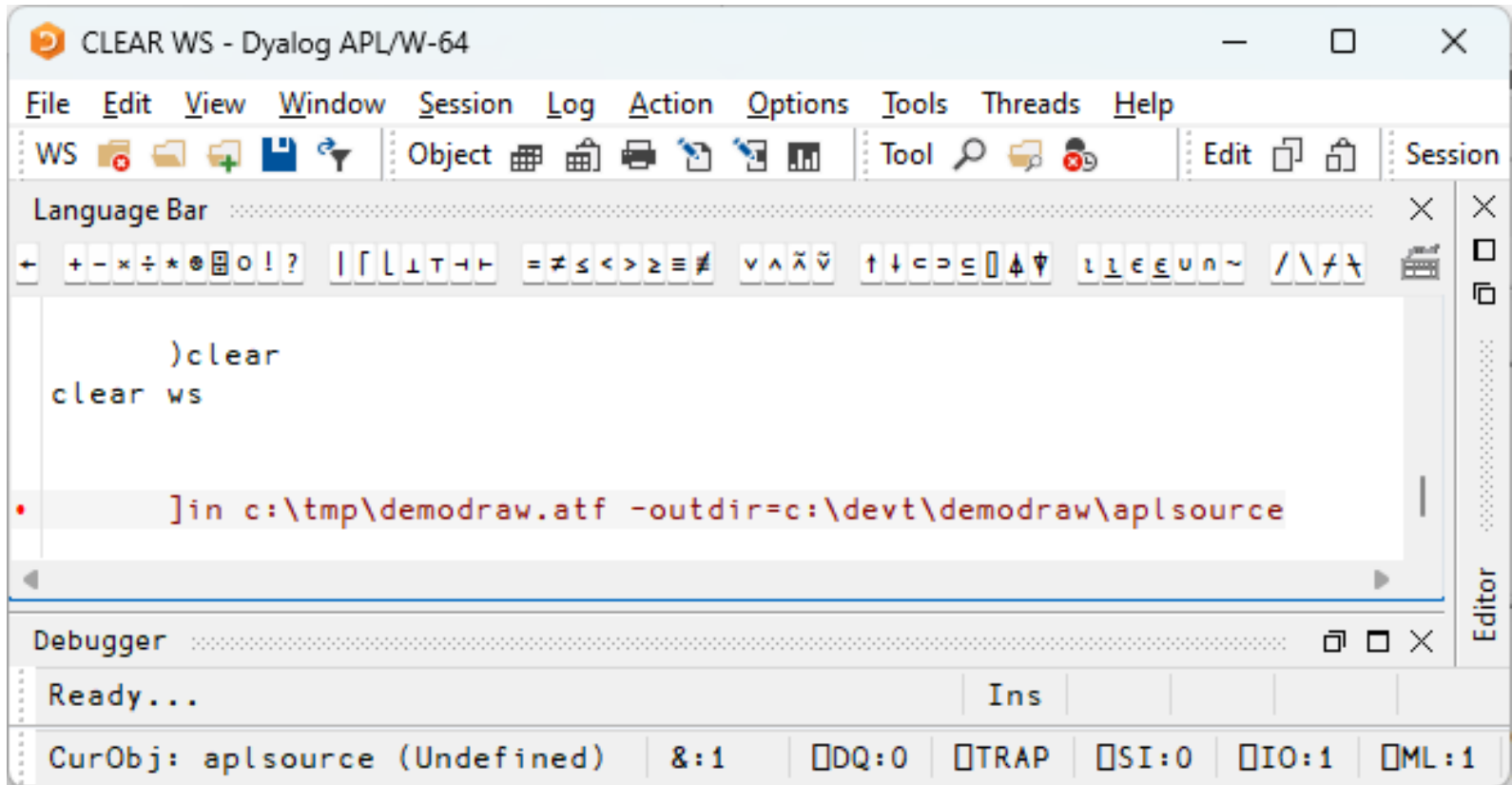
```
Copyright 2005-2006, APLNow LLC.
```

```
DemoDraw
```

```
]out c:\tmp\demodraw.atf|
```



Convert the ATF file to Dyalog Source files



The screenshot shows the CLEAR WS - Dyalog APL/W-64 application window. The menu bar includes File, Edit, View, Window, Session, Log, Action, Options, Tools, Threads, and Help. The toolbar contains icons for file operations and editing. The Language Bar is visible above the editor. The editor displays the following code:

```
)clear  
clear ws  
  
]in c:\tmp\demodraw.atf -outdir=c:\devt\demodraw\aplsource
```

The status bar at the bottom shows "Ready..." and "CurObj: aplsource (Undefined)".

c:\tmp\demodraw.dws - Dyalog APL/W-64

File Edit View Window Session Log Action Options Tools Threads Help

WS Object Tool Edit Session APL385 Unicode 16

Language Bar

```

]in c:\tmp\demodraw.atf -outdir=c:\devt\demodraw\aplsource
Linked: □SE.input.c.inout.AtfIn ↔ C:\Devt\AtfIn\APLSource\AtfIn
PROCESSING "c:\tmp\demodraw.atf" (9266 bytes):
  Variable: c:\devt\demodraw\aplsource/□PP.apla
  Variable: c:\devt\demodraw\aplsource/□IO.apla
  Variable: c:\devt\demodraw\aplsource/□CT.apla
  Variable: c:\devt\demodraw\aplsource/□RL.apla
  Variable: □PR *** FAILED ***
  Variable: □LX *** FAILED ***
  Function: c:\devt\demodraw\aplsource/DemoDraw.aplf
  Variable: c:\devt\demodraw\aplsource/Describe.apla
  Function: c:\devt\demodraw\aplsource/IsForm.aplf
  Function: c:\devt\demodraw\aplsource/Wfree.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawDemoClearH_Click.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawDemoClear_Click.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawDemoPrint_Click.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawDemoReplay_Click.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawHelp_Click.aplf
  Function: c:\devt\demodraw\aplsource/fmDraw_Make.aplf
  Function: c:\devt\demodraw\aplsource/fmDraw_MouseDown.aplf
  Function: c:\devt\demodraw\aplsource/fmDraw_Paint.aplf
  Function: c:\devt\demodraw\aplsource/fmDraw_Show.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawΔ.aplf
  Function: c:\devt\demodraw\aplsource/fmDrawΔDemo.aplf
SUCCESS:
  □SAVE 'c:\tmp\demodraw' A to save as Dyalog Workspace
  †QuadThings A to see the systems variables modified
  †PROBLEMS A to see unfixed objects
  |

```

Debugger Ready...

CurObj: tmp (Undefined) 8:1 □DQ:0 □TRAP □SI:0 □IO:1 □ML:1



Name	Date modified	Type	Size
□CT.apla	06-04-2024 13:35	APLA File	1 KB
□IO.apla	06-04-2024 13:35	APLA File	1 KB
□PP.apla	06-04-2024 13:35	APLA File	1 KB
□RL.apla	06-04-2024 13:35	APLA File	7 KB
📄 DemoDraw.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 Describe.apla	06-04-2024 13:35	APLA File	1 KB
📄 fmDraw_Make.aplf	06-04-2024 13:35	Dyalog APL Source	2 KB
📄 fmDraw_MouseDown.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDraw_Paint.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDraw_Show.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawΔ.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawΔDemo.aplf	06-04-2024 13:35	Dyalog APL Source	2 KB
📄 fmDrawDemoClear_Click.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawDemoClearH_Click.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawDemoPrint_Click.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawDemoReplay_Click.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 fmDrawHelp_Click.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 IsForm.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB
📄 Wfree.aplf	06-04-2024 13:35	Dyalog APL Source	1 KB

```
DemoDraw;R;WinDir;fmDrawΔdemo;fmDrawΔfonts;fmDrawΔhist;fm
[ a▽DemoDraw -- Run the Draw demo
  a Build the form it doesn't already exist
  :if ~IsForm 'fmDraw'
    fmDraw_Make
  :end

  a Wait on the form
  R←'fmDraw' □wi 'Wait'
```

NB this is original APL+Win Source

For METSIM®, we plan to update the APL+Win environment to run off text files too.

Next, map code from APL+Win to Dyalog

```
]todyalog aplsource c:\devt\demodraw\dyalog a2k  
Using c:\devt\demodraw\aplsource\atfmap.txt  
20 files processed
```

```

:catch%:else
:catchall%:else
:endtry%:endtrap
:returnif%→0/↵
:try *%:trap 0
:try%:trap 0
;□ALX%;ΔQALX
;□ELX%;ΔQELX
;□SA%;ΔQSA
;□WSELF%;ΔWSELF
□ALX%ΔQALX
□ALX←%#.A2K.ΔSetALX
□AV%#.A2K.ΔAV
□CHDIR%#.A2K.ΔCHDIR
□CHDIR%#.A2K.ΔCHDIR
□CN*%□N
□CRLF%(□UCS 13 10)
□CURSOR%#.A2K.ΔCURSOR
□DEF%□FX
□DR%#.A2K.ΔDR
□ELX%ΔQELX
□ENLIST%{□ml←1◊εω}

```

```

□FSTIE%#.A2K.ΔFSTIE
□FTIE%#.A2K.ΔFTIE
□HTOPIC%#.A2K.ΔHTOPIC
□IDLIST%#.A2K.ΔIDLIST
□IDLOC%#.A2K.ΔIDLOC
□INT%#.A2K.ΔINT
□KEYLOG%#.A2K.ΔKEYLOG
□KEYW%#.A2K.ΔKEYW
□LIB%#.A2K.ΔLIB
□LIBD%#.A2K.ΔLIBD
□LIBS%#.A2K.ΔLIBS
□LOG%#.A2K.ΔLOG
□MF%□MONITOR
□MIX%#.A2K.ΔMIX
□NA%#.A2K.ΔNA
□PEEK%#.A2K.ΔPEEK
□PENCLOSE%ε
□PFKEYS%#.A2K.ΔPFKEYS
□POKE%#.A2K.ΔPOKE
□POKES%#.A2K.ΔPOKES
□REPL%/
□SA%ΔQSA

```

```

□TCBEL%(□UCS 7)
□TCBS%(□UCS 8)
□TCESC%(□UCS 27)
□TCFF%(□UCS 12)
□TCHT%(□UCS 9)
□TCLF%(□UCS 10)
□TCNL%(□UCS 13)
□TCNUL%(□UCS 0)
□TYPE%#.A2K.ΔTYPE
□UCMD%#.A2K.ΔUCMD
□UCS%#.A2K.ΔUCS
□USERID%□AN
□VI%#.A2K.ΔVI
□WCALL%#.A2K.ΔWCALL
□WGIVE%#.A2K.ΔWGIVE
□WI%#.A2K.ΔWI
□WIN%#.A2K.ΔWIN
□WINDOW%#.A2K.ΔWINDOW
□WKEYS%#.A2K.ΔWKEYS
□WSELF%ΔWSELF
□WSSIZE%(2000ι0)
□XFDUP%#.A2K.ΔXFDUP

```

Take advantage of Git and VS Code

The screenshot shows the Visual Studio Code interface with a Git commit dialog open on the left. The dialog displays the commit message "Message (Ctrl+Enter to commit on 'aplplus')", a "Commit" button, and a list of changes. The file "fmDraw_Show.aplf" is selected in the changes list. The main editor area shows a diff view for "fmDraw_Show.aplf" in the "Working Tree". The diff highlights changes in lines 4, 5, and 6. Line 4 shows a change in the font name from "ARIAL" to "ARIAL", and line 5 shows a change in the window title from "144p" to "144p(UCS 0)".

```
1 fmDraw_Show;R
2 fmDrawΔdemo+'poly'
3 fmDrawΔpage+fmDrawΔhist+a
4- fmDrawΔfonts+□wi ':Draw' ((c'?Font':'ARIAL'),','.5x18)
4+ fmDrawΔfonts+Δwi ':Draw' ((c'?Font':'ARIAL'),','.5x18)
5- R←□wcall.'GetWindowsDirectory'.(144p□tcnu1) 144
5+ R←#.A2K.ΔWCALL.'GetWindowsDirectory'.(144p(□UCS 0)) 144
6 R←(□io>R)†(1+□io)>R
7 WinDir+(-'\ '=™1†R)†R
8
```

Original
APL+Win code

```
fmDraw_Show.aplf (Working Tree) M X
aplsource > fmDraw_Show.aplf
1 fmDraw_Show;R
2 ·fmDrawΔdemo←'poly'
3 ·fmDrawΔpage←fmDrawΔhist←a
4- ·fmDrawΔfonts←□wi·':Draw'·((c'?Font'·'ARIAL')·,·".5×18)
5- ·R←□wcall·'GetWindowsDirectory'·(144p□tcnu1)·144
6 ·R←(□io>R)↑(1+□io)⊃R
7 ·WinDir←(-'\ '=^-1↑R)↓R
8
```

Converted to
Dyalog

```
1 fmDraw_Show;R
2 ·fmDrawΔdemo←'poly'
3 ·fmDrawΔpage←fmDrawΔhist←a
4+ ·fmDrawΔfonts←Δwi·':Draw'·((c'?Font'·'ARIAL')·,·".5×18)
5+ ·R←#.A2K.ΔWCALL·'GetWindowsDirectory'·(144p(□UCS·0))·144
6 ·R←(□io>R)↑(1+□io)⊃R
7 ·WinDir←(-'\ '=^-1↑R)↓R
8
```

So what is #.A2K.ΔWCALL ?



devt > Atfln > APLSource > xfr > U > A2K > Search A2K

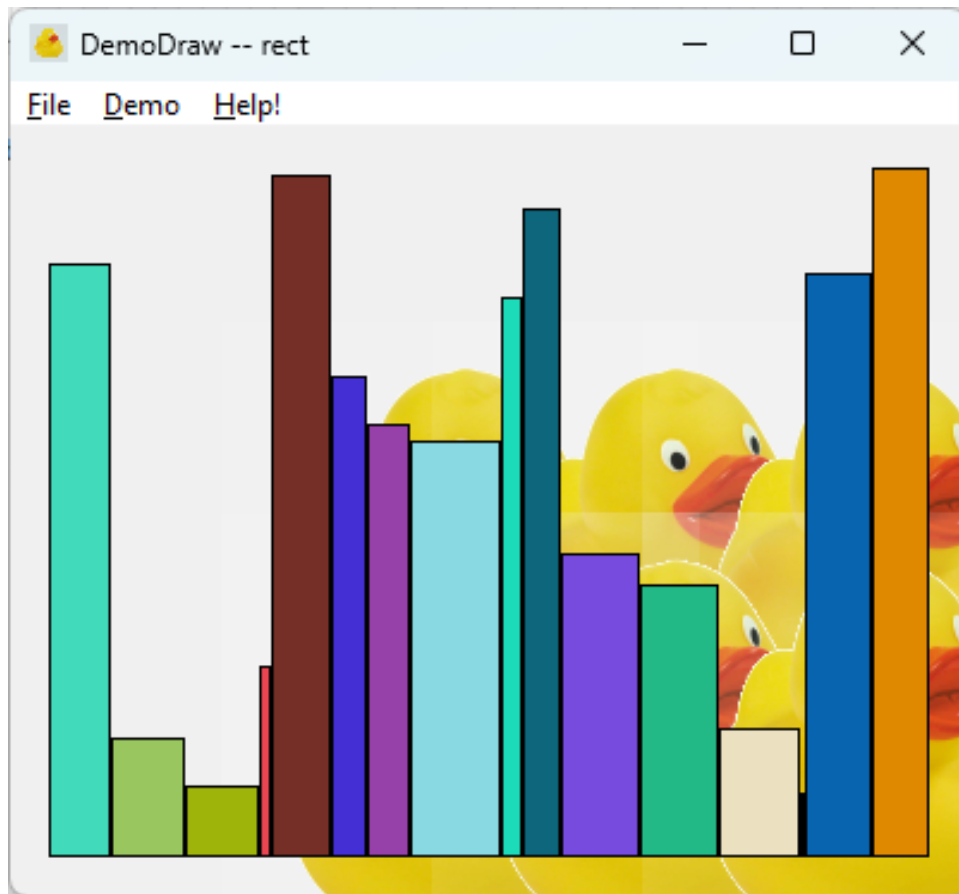
Sort View Preview

APLPΔSYS.aplc	ΔIDLOC.aplf	ΔTYPE.aplf
bigstring	ΔINT.aplf	ΔVI.aplf
ΔAV.apla	ΔLIB.aplf	ΔWCALL.aplf
ΔCHDIR.aplf	ΔLIBD.aplf	ΔWGIVE.aplf
ΔCOPY.aplf	ΔLIBS.aplf	ΔWSELF.apla
ΔDR.aplf	ΔLOG.aplf	ΔXFDUP.aplf
ΔENLIST.aplf	ΔMIX.aplf	ΔXLIB.aplf
ΔERASE.aplf	ΔNA.aplf	ΔIO.apla
ΔESC.aplf	ΔNUL.aplf	ΔML.apla
ΔFI.aplf	ΔPCOPY.aplf	GLOBALΔTRAP.apla
ΔFIRST.aplf	ΔPEEK.aplf	GLOBALTRAP.aplf
ΔFSTIE.aplf	ΔPOKE.aplf	
ΔFTIE.aplf	ΔSEG.aplf	
ΔHT.aplf	ΔSPLIT.aplf	
ΔIDLIST.aplf	ΔTCBEL.aplf	

```

r←ΔWCALL args
:Select 1>,⊆args
:Case 'W_Init'
    r←'' (⌘1⌘2 ⌘NQ '.' 'GetCommandLineArgs')
:Else
    'This WCALL not currently supported' ⌘SIGNAL 11
:EndSelect
  
```

So what is #.A2K.ΔWCALL ?

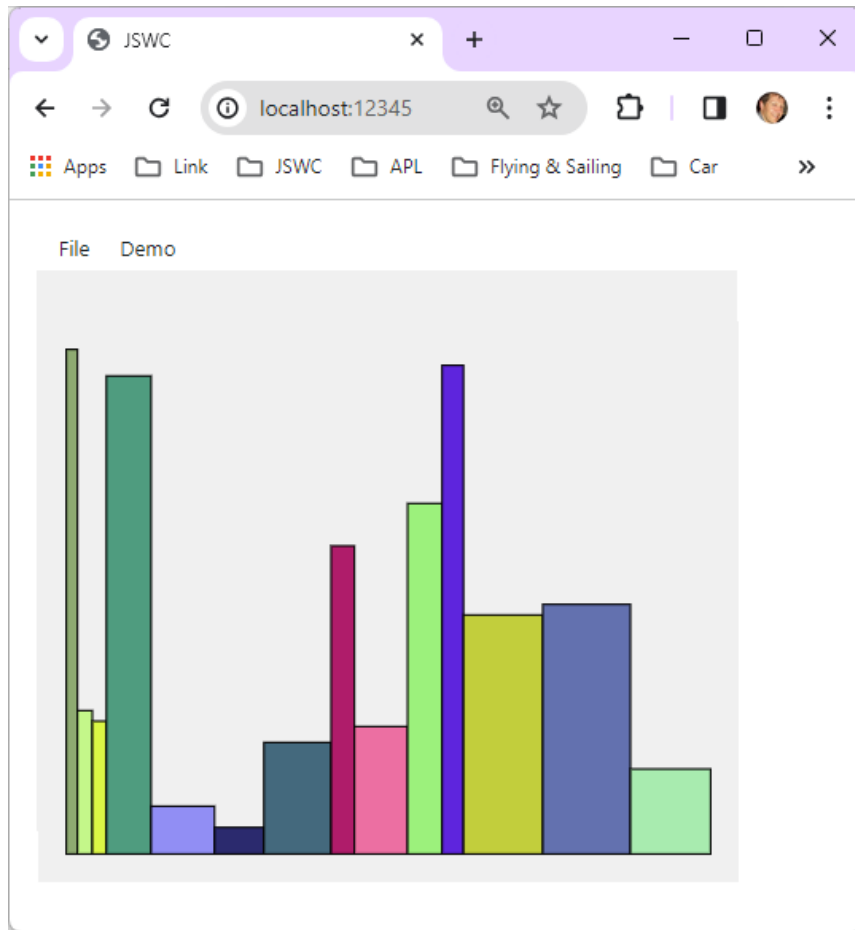


Migrating to Dyalog



APL Germany e.V.

Spring '24



Migrating to Dyalog



APL Germany e.U.

Spring '24

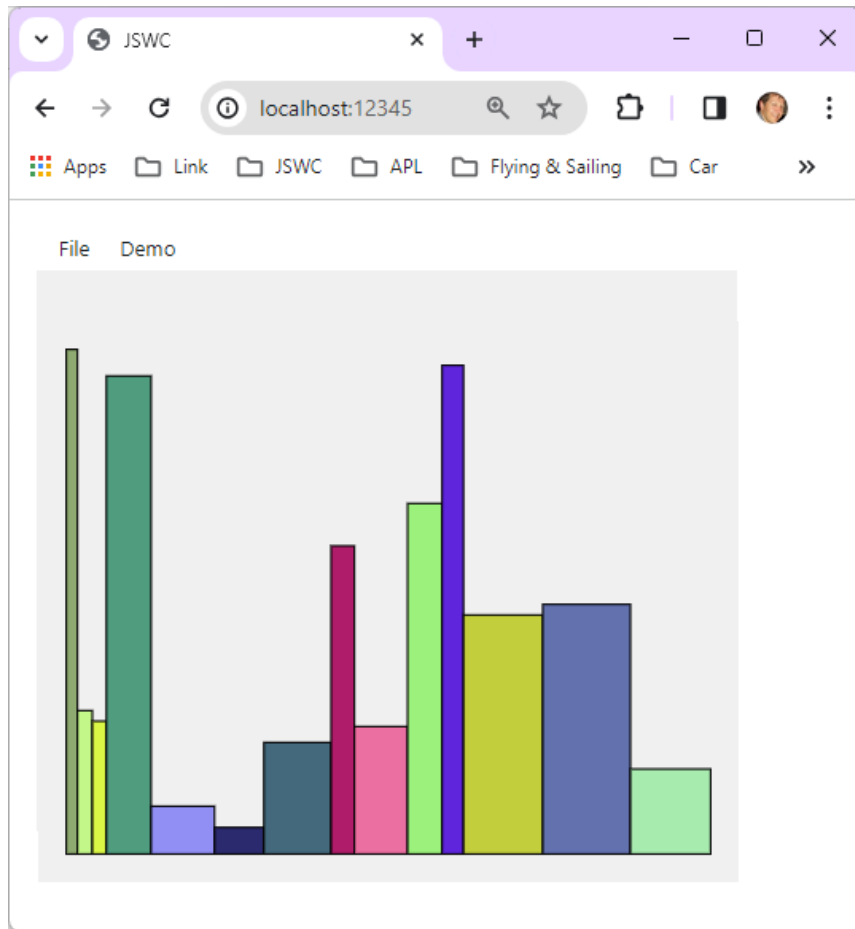
Status

- We are reworking and enhancing the tools developed for APLX migrations
- Will enumerate differences between APL+Win and Dyalog
- Will create emulation functions as required
- We **may** also decide to add new features to Dyalog v20 (which should start user testing in late 2024)
 - For example :LeaveIf

Status

- ◆ Dyalog has been contracted to port the METSIM[®] application
- ◆ Hired one new APL developer in January, aiming for 1 more
- ◆ We will have ~1.5-2 full time equivalent resources working on migration tools until further notice

- ◆ **All the resulting tools and documentation will be free and open source**



Migrating to Dyalog



APL Germany e.U.

Spring '24