

# RIDE: a new way to interact with the interpreter

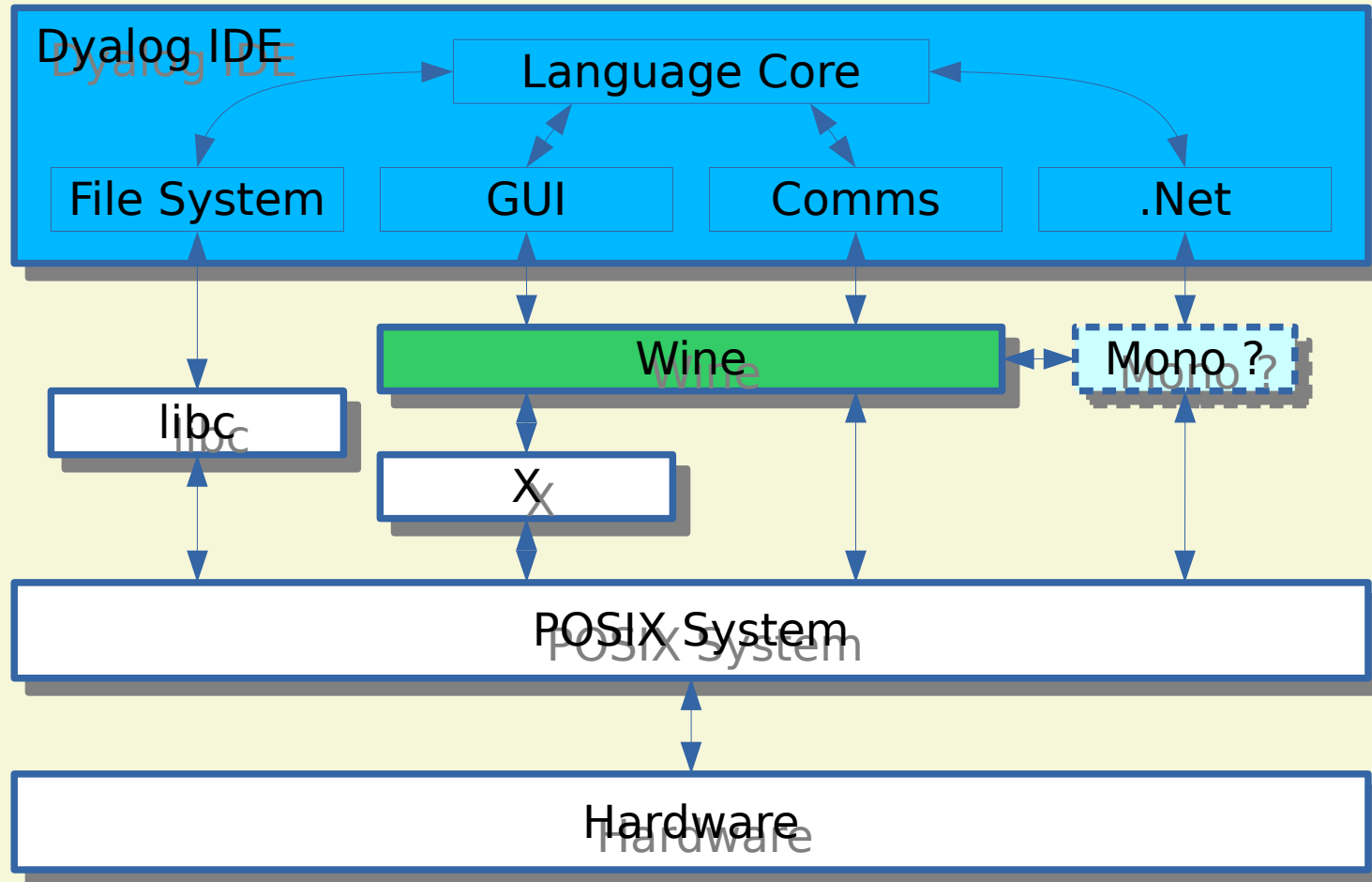
Nick Nickolov



DYALOG

Sicily 2015

## "Dyalog on Wine" Nicolas Delcros, Dyalog'06



**CLEAR WS - Dyalog APL/W-64 - [f]**

Tools View

WS [Icons] Object [Icons] Tool [Icons] Edit [Icons] Session

Language Bar [Icons]

```

Mon Aug 24 10:29:35 2015
clear ws
Dyalog APL/W-64 Version 14.1.24909
Serial No : 000000
Unicode Edition
Mon Aug 24 11:52:32 2015
clear ws
    f
    
```

**Debugger**

[Icons]

[0]	f
[1]	1

Function: nick... Pos: 1/2,0

Ready... Ins

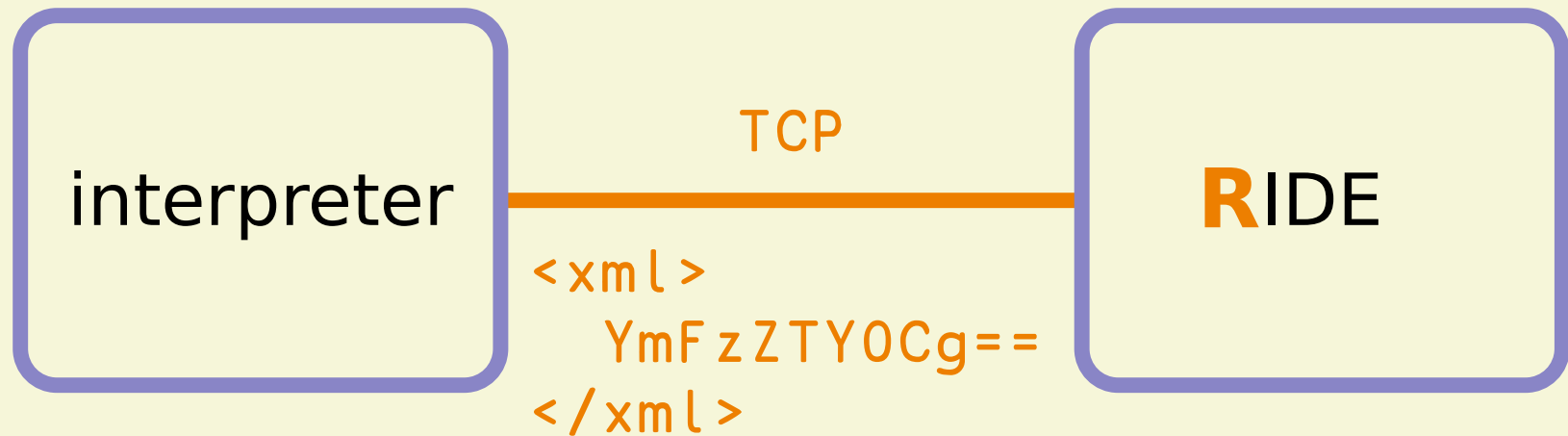
CurObj: f (Function)    &:1    DDQ:0    OTRAP    **SI:1**    IO:1    ML:1

```

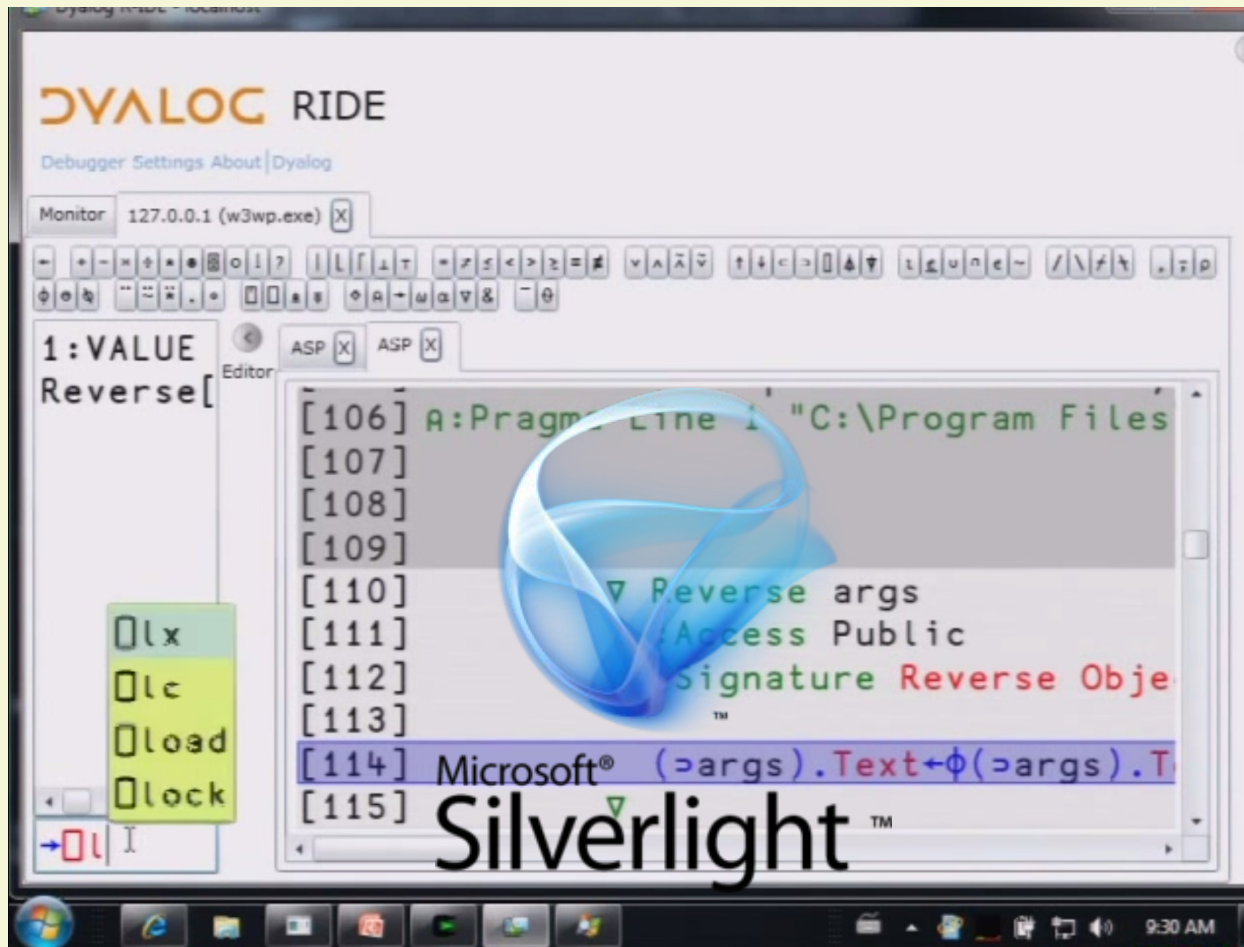
Dyalog APL/S-64 Version 15. |tree
Unicode Edition           |tree
DEBUG Build              |  }{
Mon Aug 24 19:04:19 2015 |    exec←αα
clear ws                 |
    f                     |
    )load dfns           |
/home/n/dyalog/trunk/apl/ |
d Thu Jul  2 13:57:38 2015 |
An assortment of D Functio
                            |
tree #                       |
↑~10↑↓attrib □nl 3 4      |  A What's new?
⌘notes find 'Word'       |  A Apropos "Word".
□ed'notes.contents'     |  A Workspace overview.
                            |
tree #                       |  A Workspace map.

```





## "Taking APL for a Ride" John Daintree, Dyalog'10



## Windows: WPF RIDE in C#

The screenshot shows the Dyalog RIDE interface with a C# application running. The application window displays the Windows logo and the text "Windows Presentation Foundation". The RIDE interface includes a menu bar with "Debug" selected, a toolbar, and a code editor. A debug menu is open, listing options such as "Run item under cursor in tracer", "Interrupt", "Strong interrupt", "Restart all threads", "Continue executing all threads", "Step over", "Step into", "Go back", "Go forward", "Continue trace", "Continue execution", "Quit function", and "Clear breakpoints, monitors etc.". The code editor shows a loop of numbers 1 through 7, and the console output shows the numbers 1 through 7 followed by the letter 'f'.



Windows: WPF RIDE in C#  
running on .NET

WPF UI

model classes  
& communication

Linux: GTK RIDE in C#  
running on Mono

~~WPF~~GTK UI

model classes  
& communication

## "RIDE 1.0" Jonathan Manktelow, Dyalog '14

The screenshot displays the Dyalog RIDE IDE interface. The main window title is "Dyalog RIDE: ./demo.dws [192.168.137.111:4502]". Below the title bar is a menu bar with "Connection", "Fix", "Edit", "Debug", "View", "Settings", and "Help". A toolbar with various icons is visible below the menu bar. The main workspace is divided into two panes:

- [Session] Pane:** Shows the execution of a test. It displays the expression `2+2` resulting in `4`. Below this, it shows a test failure:
 

```

      a
      Test expected: lcase 'Hello, world!' => hello,
      actual:       lcase 'Hello, world!' => HELLO,
      Fail
      a
      Test expected: lcase 'Hello, world!' => hello,
      lcase[1]
      
```
- [Tracer] Pane:** Shows the source code of the `lcase` function. The code is:
 

```

      [0] lcase←{
      [1]   uc←'abcdefghijklmnopqrstuvwxyzaöå
      [2]   lc←'ABCDEFGHIJKLMNopqrstuvwxyzAAÖÅ
      [3]   (pw)p(lc..w)[(uc..w)iw]
      [4] }
      
```

 The line `lc←'ABCDEFGHIJKLMNopqrstuvwxyzAAÖÅ` is highlighted in yellow.

# "Technical Road Map" Morten Kromberg, Dyalog '14

DIALOG

## RIDE: One Final Twist



- First, there was Silverlight ...
- Switch to WPF for Windows Desktop
- ... and GTK for Linux, Mac OS, "etc"
- There will also be no "GTK" RIDE for UNIX and Mac OS
- A cross-platform HTML5/Javascript RIDE will follow the WPF version.

[#dyalog14](#)

# "RIDE 1.0"

## Jonathan & Nick, Dyalog '14

The screenshot shows a web browser window titled 'CLEAR WS' with the URL 'https://127.0.0.1:8443'. The main content area displays APL code:

```
clear ws
 1 2 3+4 5 6
5 7 9
 1 2 3+4 5 6
5 7 9
5 7 9
5 7 9
  f
2
3
f[3]
)reset
  f
2
3
4
```

Below the code is a debugger interface with a toolbar and a list of line numbers (1-5). A red horizontal line is drawn across the interface, highlighting line 4. The list of line numbers is:

```
f|
2
3
4
45
```

The bottom of the browser window shows the taskbar with the following items: 'nick@macavity: ...', 'CLEAR WS - Chro...', 'Mozilla Firefox (P...', and 'nick@macavity: ...'. The system clock in the bottom right corner shows '16:59'.

"Welcome"  
Gitte Christensen, Dyalog'14

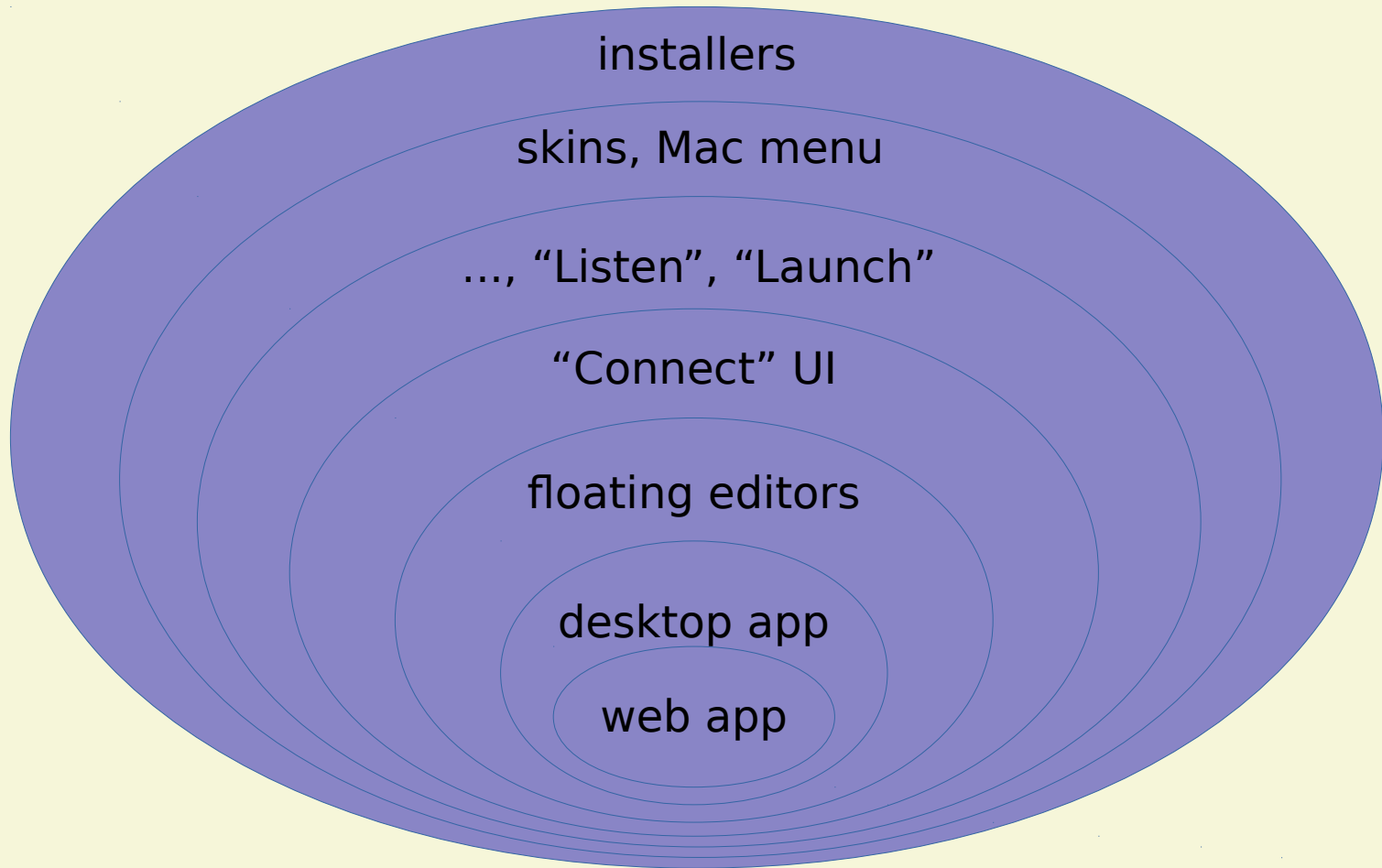
- RIDE to make sure you can develop on one platform, deploy on another and still debug you code comfortably
- - and sometimes we don't even get to deploy before we have to start all over again
- - but we are happy to do so



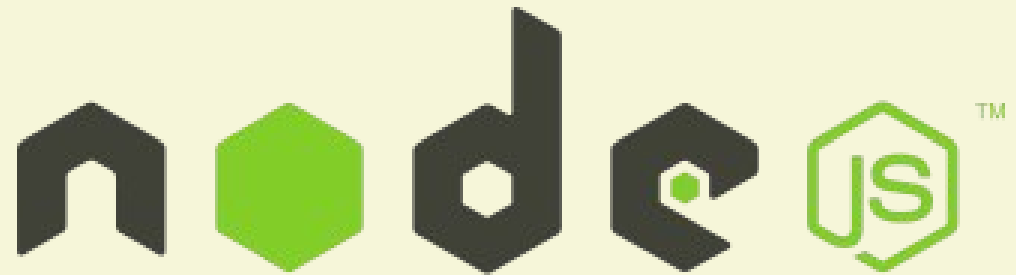
"Sisyfos" by Petr Kavan



that was 1 year ago, then it grew...



standing on the shoulders of giants



\*not to scale

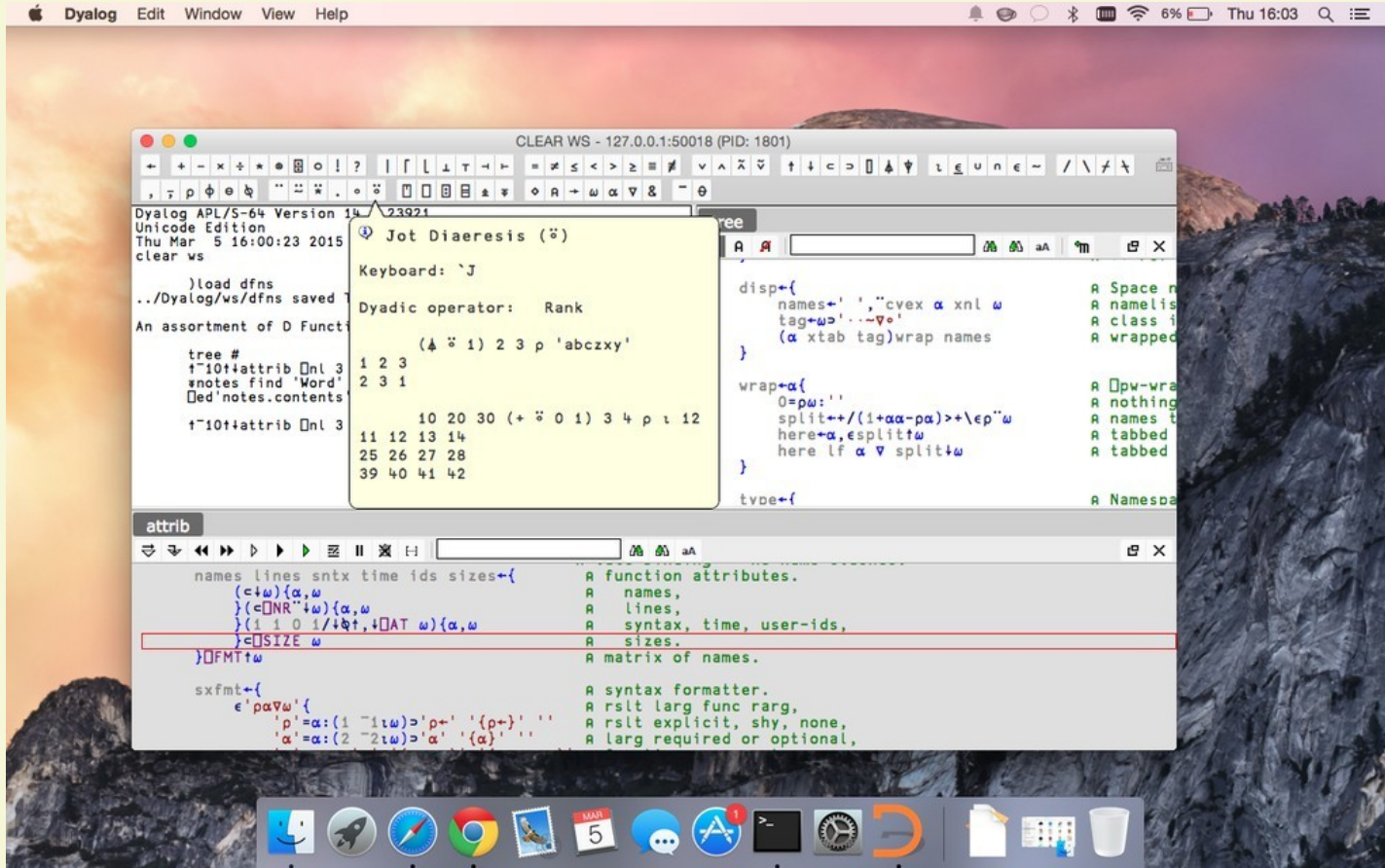


The screenshot shows the Windows Task Manager window with the 'Details' tab selected. The window title is 'Task Manager'. The menu bar includes 'File', 'Options', and 'View'. The tabs at the top are 'Processes', 'Performance', 'App history', 'Startup', 'Users', 'Details', and 'Services'. The 'Details' tab is active, displaying a list of running processes with columns for Name, PID, Status, User name, CPU, Memory, and Description. At the bottom left, there is a 'Fewer details' button with an upward arrow icon. At the bottom right, there is an 'End task' button.

Name	PID	Status	User name	CPU	Memory	Description
EndpointService...	1888	Running	SYSTEM	00	180,268 K	Endpoint Host Service
OUTLOOK.EXE	6688	Running	...	00	170,568 K	Microsoft Outlook
Spotify.exe	7604	Running	...	00	168,672 K	Spotify
ride20.exe	124...	Running	...	00	137,372 K	ride20.exe
chrome.exe	6800	Running	...	00	116,540 K	Google Chrome
svchost.exe	912	Running	SYSTEM	00	114,696 K	Host Process for Windows Services
chrome.exe	3200	Running	...	00	101,468 K	Google Chrome
SourceTree.exe	191...	Running	...	00	94,980 K	SourceTree
Skype.exe	6544	Running	...	01	90,792 K	Skype
Dropbox.exe	9068	Running	...	00	82,856 K	Dropbox
SearchIndexer.e...	4140	Running	SYSTEM	00	65,820 K	Microsoft Windows Search Indexer
chrome.exe	163...	Running	...	00	56,512 K	Google Chrome



"Exciting times ahead for  
Dyalog APL on Mac OS X!"  
@DyalogAPL, March 2015



## "RIDE 2.0" on Windows

The screenshot shows the Dyalog APL IDE interface. The left pane displays a file explorer with a tree structure:

- ~ packS packT packU packX pack2
- ~ queens queensX quotes quzzle
- ~ refs refws remlink remnode re
- ~ select sieve soup span splay
- ~ sudoku sudokuX sudoku\_bfs suf
- ~ truth\_tables tview type ucase
- ~ vwise wGraphs wcost while wms
- ~ xtimes xws
- scripts
  - ~ Binding Blank\_removal Cholesk
  - ~ \_dyalogX \_fk \_joy abc acc edi
  - ~ assign attrib avl baby bags b
  - ~ colsum compidn cond cxdraw da
  - ~ displayr displays dist dlb dm
  - ~ factors fibonacci file filefi
  - ~ gcd getfile hex hexdump hexf
  - ~ key ksphere kt lcase le lex l
  - ~ merge merge2 mmind mns morse
  - ~ packD packH packN packQ packR
  - ~ popnode pow pred putfile quee
  - ~ refws remlink remnode rep rep
  - ~ select sieve span splay squad
  - ~ ticks time timestamp tnest to
  - ~ up utf8 utf8get utf8getput ut
  - ~ wsdiff wsmerge wspan wsreq xh

The right pane shows a code editor with the following APL code:

```

avl ← {
  [0] IO ML ← 0 1
  [1]
  [2] get ← {
  [3]   α ← 0
  [4]   (k v) ← subs α
  [5]   k ← ω : v
  [6]   dir ← - / ⋆ ω k
  [7]   sub sib ← dir wise subs
  [8]   sub ∇ ω
  [9] }
  [10]
  [11] put ← {
  [12]   α ← 0 : (ω 0 (0 0)) 1
  [13]   (kv obal subs) (key val)
  [14]   key ⇒ kv : (ω obal subs) 0
  [15]   dir obv ← 1 - 1 × - / ⋆ key (⇒)
  [16]   sub sib ← dir wise subs
  [17]   nsub inc ← sub ∇ ω
  [18]   new ← obv proj kv obal (n)
  [19]   inc ← 0 : new 0
  [20]   (dir balance new) (obal)
  [21] }
  [22]
  [23]

```

Characters

Commands

Colours

Characters

Commands

Colours



APL 385

θ θ ϕ ϕ ϕ ϕ ρ ρ ρ

Ο Ο ο ο ο ι ι ι ι ι ι ι ι ι ι

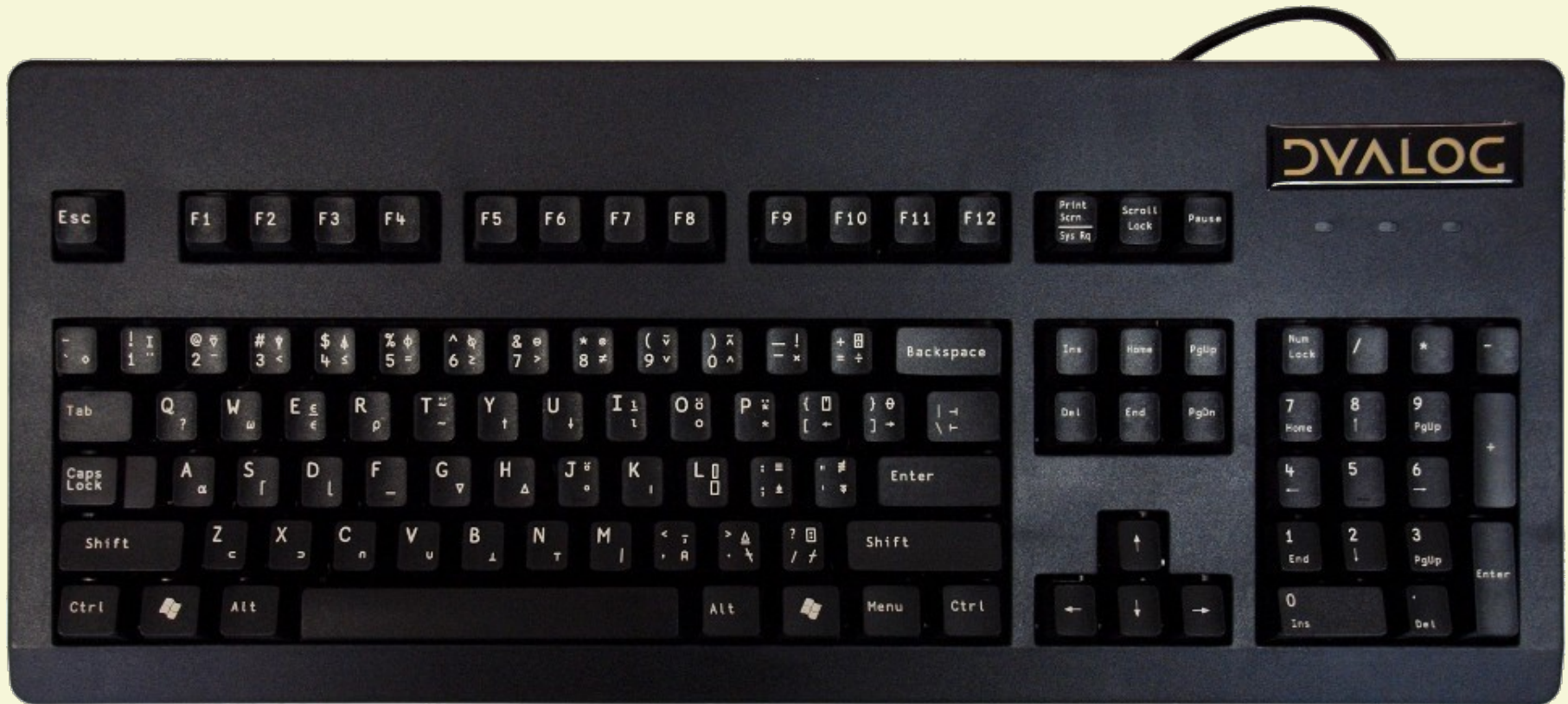
α α α α ω ω ω ω × × × ×

## IBM 2741 Terminal (1965)

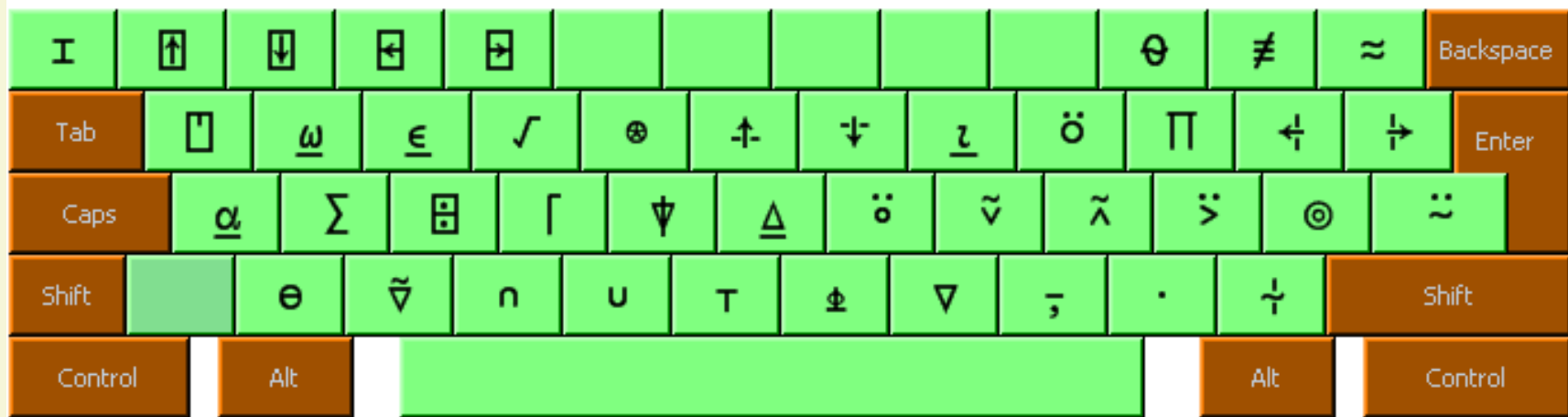




## hardware



## "An APL Keyboard for the Future", Adrian Smith, 2006



**Text Services and Input Languages**

General | Language Bar | Advanced Key Settings

Default input language  
Select one of the installed input languages to use as the default for all input fields.

English (United Kingdom) - Dyalog APL IME (en-GB)

Installed services  
Select the services that you want for each input language shown in the list. Use the Add and Remove buttons to modify this list.

- DA Danish (Denmark)
  - Keyboard
    - Dyalog APL IME (da-DK)
    - Danish
- EN English (United Kingdom)
  - Keyboard
    - US
    - United Kingdom
    - Dyalog APL IME (en-GB)**
- BG Bulgarian (Bulgaria)
  - Keyboard
    - Bulgarian (Typewriter)

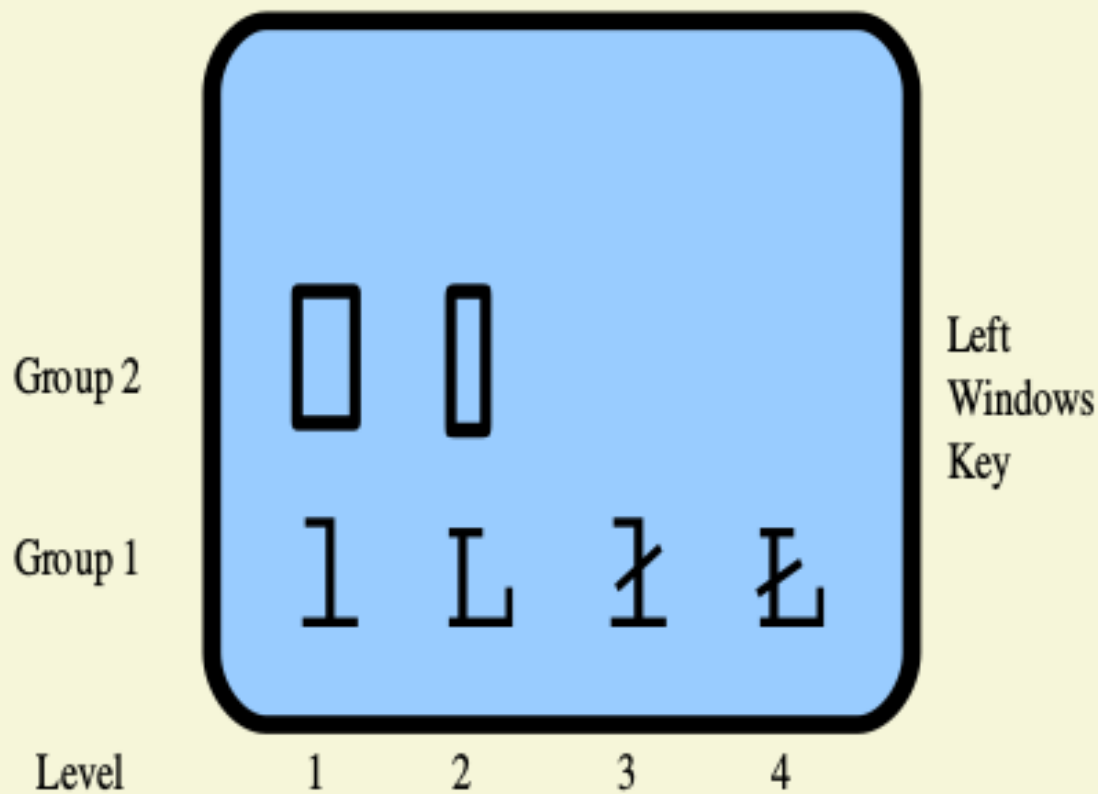
Buttons: Add..., Remove, Properties..., Move Up, Move Down

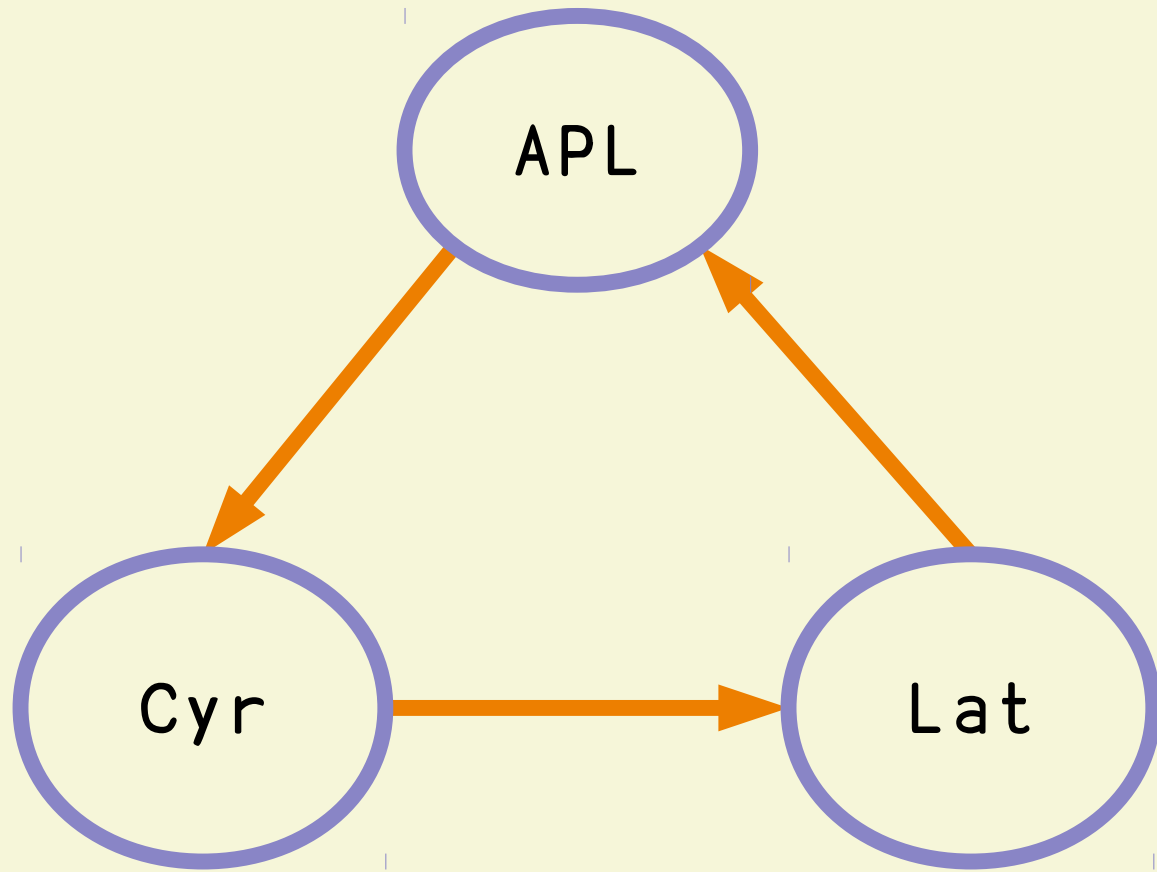
Buttons: OK, Cancel, Apply

# IME

US  
United States-International  
✓ **Dyalog APL IME (en-US)**

## "Keyboards in the New World" Geoff Streeter, 2007





mine



ctrl + i → ι

ctrl + r → ρ

ctrl + % → ϕ

• • •

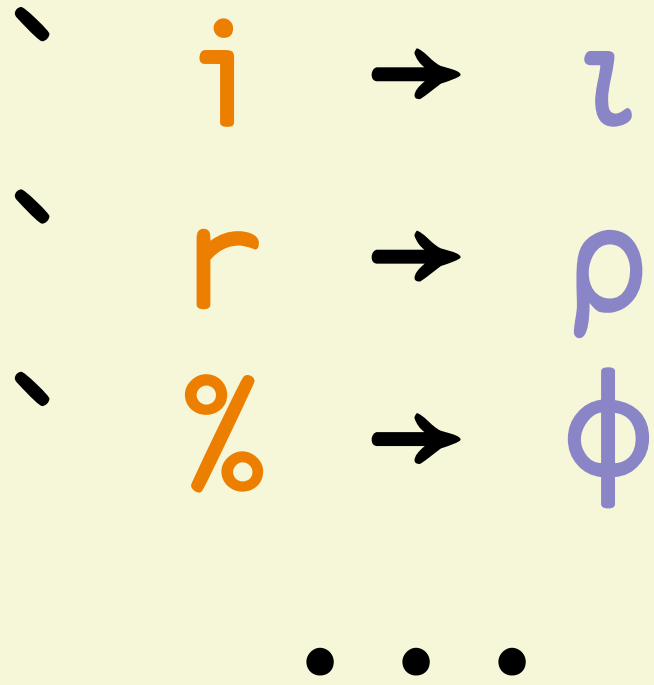
~~ctrl~~+i → ι

~~ctrl~~+r → ρ

~~ctrl~~+% → φ

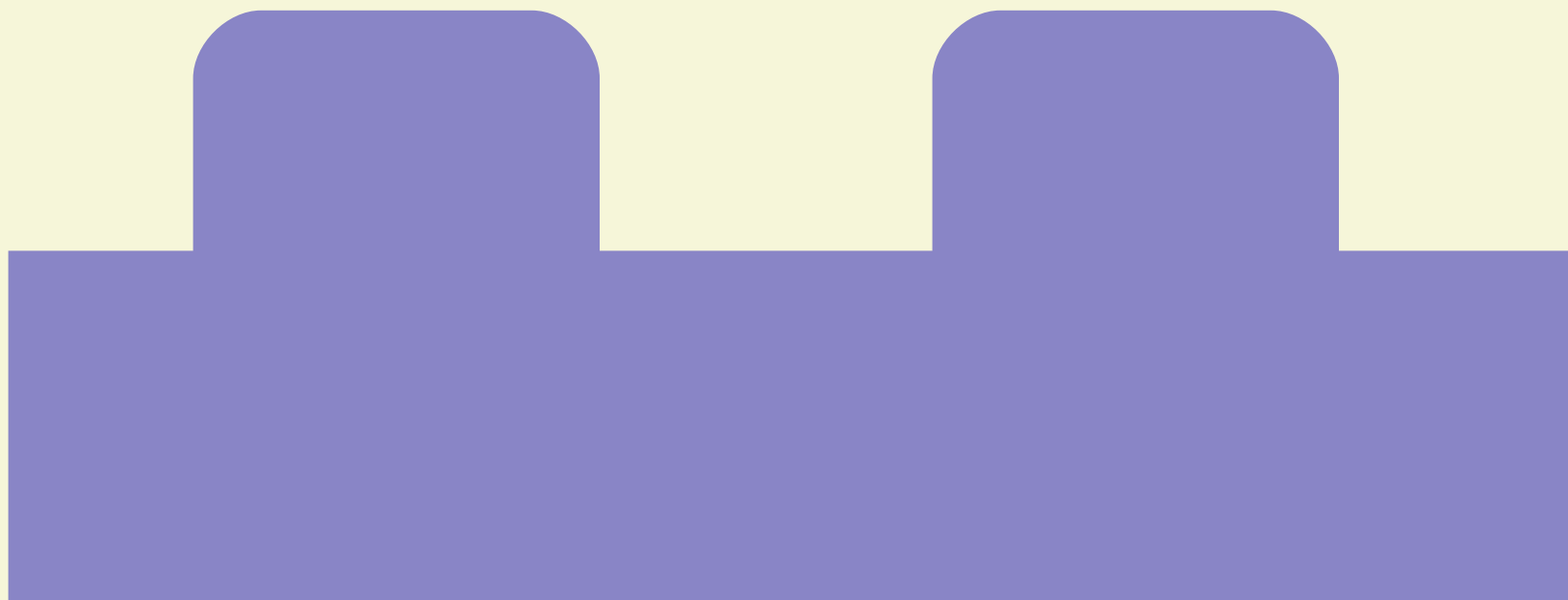
• • •





ctrl

other  
key



backquote

other  
key



The screenshot shows a web browser window with the URL `tryapl.org`. The page has a purple header with the text "Try APL" and two buttons: "APL Related News Items" and "APL Keyboard". Below the header is a navigation bar with buttons for "Hi!", "APL", "Primer", "Links", "About", and "Learn".

The main content area contains the following text:

**Welcome to TryAPL!**

This site is to help you learn about **APL** by using it. Type APL expressions in the input field on the right and hit enter to see the results. Here are a few examples (which you can click on) to get you started:

`2+2` No points for guessing the answer to this  
`4 2 3 + 8 5 7` Functions apply to arrays  
`⊖10`  
`+⊖1`  
`×⊖1`  
`avg`  
`avg`  
`thro`  
`+/1=`  
`+/(`  
`'Hel`

A virtual keyboard is overlaid on the bottom half of the page, featuring a standard QWERTY layout with additional APL-specific keys: "APL", "Previous", and "Next".

**Preferences** [X]

Layout | Shortcuts | Code | Colours | Title | Menu

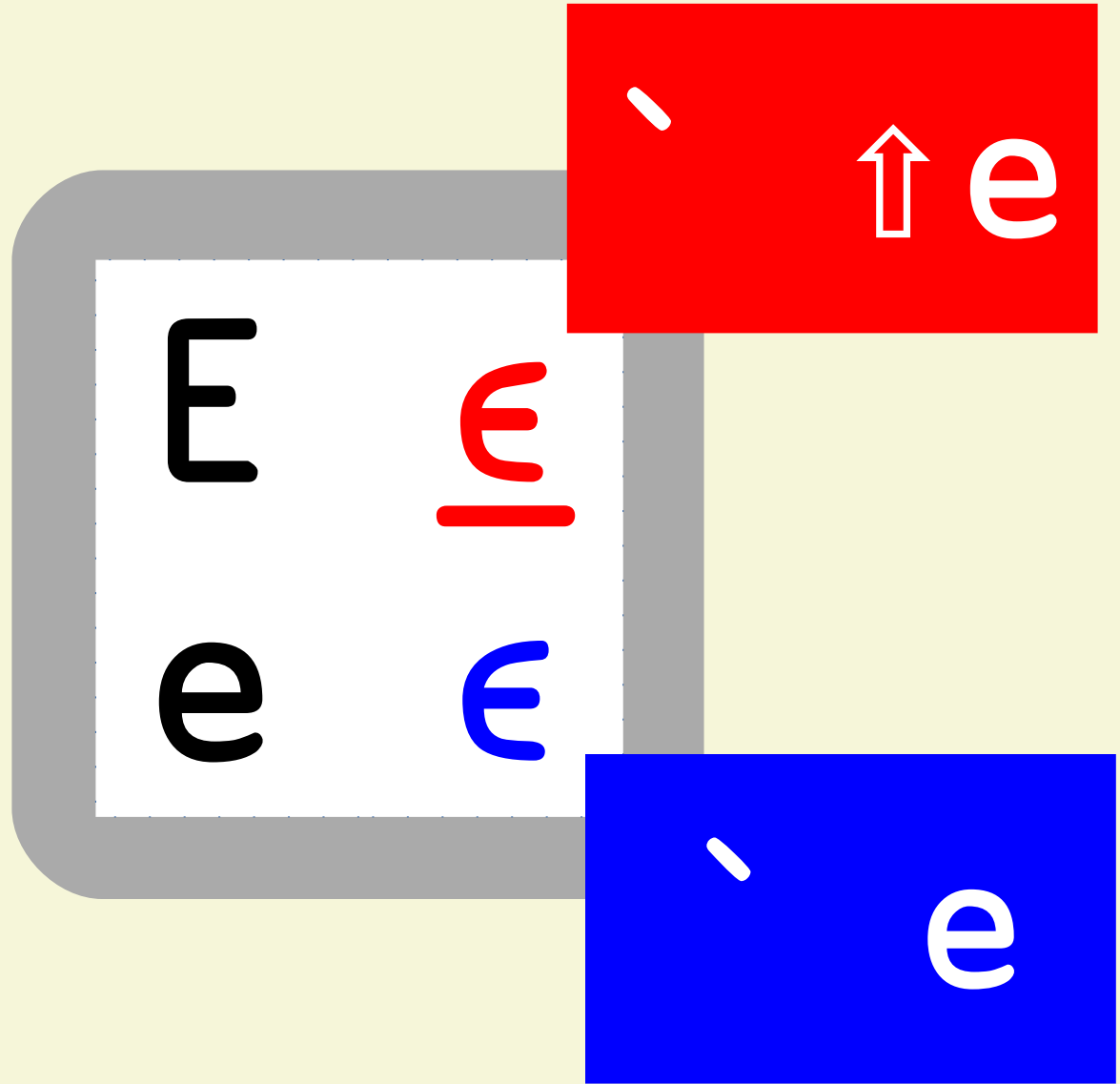
Prefix: ` ↑x ` ↑x x ` x Reset

Locale: **US** ▾

~	! I	@ ÷	# ψ	\$ Δ	% ϕ	^ ϕ	& ϑ	* ⊗	( ÷ )	~	- !	+ ⊞	←
` ◊	1 ¨	2 -	3 <	4 ≤	5 =	6 ≥	7 >	8 ≠	9 v	0 ^	- x	= ÷	
↔	Q	W	E €	R	T ÷	Y	U	I l	O ö	P *	{ □ }	θ	-
	q ?	w ω	e €	r ρ	t ~	y ↑	u ↓	i l	o o	p *	[ ← ]	→	\ r
Caps	A	S	D	F	G	H	J ö	K □	L □	: ≡	" ≠		↵
	a α	s l	d l	f _	g v	h Δ	j o	k ' l □	; ±	' ϕ			
↑	Z	X	C	V	B	N	M	< ;	> Δ	? ⊞			↑
	z c	x >	c n	v u	b l	n T	m	, A	. x	/ f			

OK Apply Cancel





## autocompletion and language bar

```

\
⌈ \! I-beam
≠ \ " tally/not match
∇ \# grade down
⋈ \$ grade up
ϕ \% reverse/rotate
⊖ \& ϕ[⊖io]
⌘ \ ' format
∨ \ ( nor
⌘ \ ) nand
⊗ \ * logarithm
⊞ \ + matrix inv/div
⌈ \ , comment
× \ - signum/times
\ \ . \[⊖io]
÷ \ / /[⊖io]
^ \ 0 and (LCM)

```

**Circle Style (ϕ)**

Keyboard: ` %

Monadic function: Reverse

```

      ϕ 8 2 5 6.6 ^2
^-2 6.6 5 2 8

```

ϕ 3 2 ρ16

```

2 1
4 3
6 5

```

Dyadic function: Rotate

```

      1 2 ^-1 ϕ 3 3ρ19
2 3 1
6 4 5
9 7 8

```

```

\
I `! I-beam
≠ `" tally/not match
ψ `# grade down
⤴ `⋄ grade up
ϕ `% reverse/rotate
⊖ `& ϕ[⊖io]
⌘ `` format
⋮ `( nor
⋮ `) nand
⊗ `* logarithm
⊞ `+ matrix inv/div
A `, comment
× `− signum/times
⋮ `.\ [⊖io]
⋮ `/ [⊖io]
^ `0 and (LCM)
\

```

$\backslash i \leftrightarrow \backslash \backslash \text{iota}$   
 $\backslash r \leftrightarrow \backslash \backslash \text{rho}$   
 $\backslash \% \leftrightarrow \backslash \backslash \text{reverse}$



...

```

÷ `= ``reciprocal
| `m ``residue
| `m ``remainder
τ `n ``representation
/ ``reduce
/ ``replicate
≠ `/ ``reducefirst
ρ `r ``reshape
ϕ `% ``reverse
⊖ `& ``reversefirst
∇ `g ``recur

```

``re

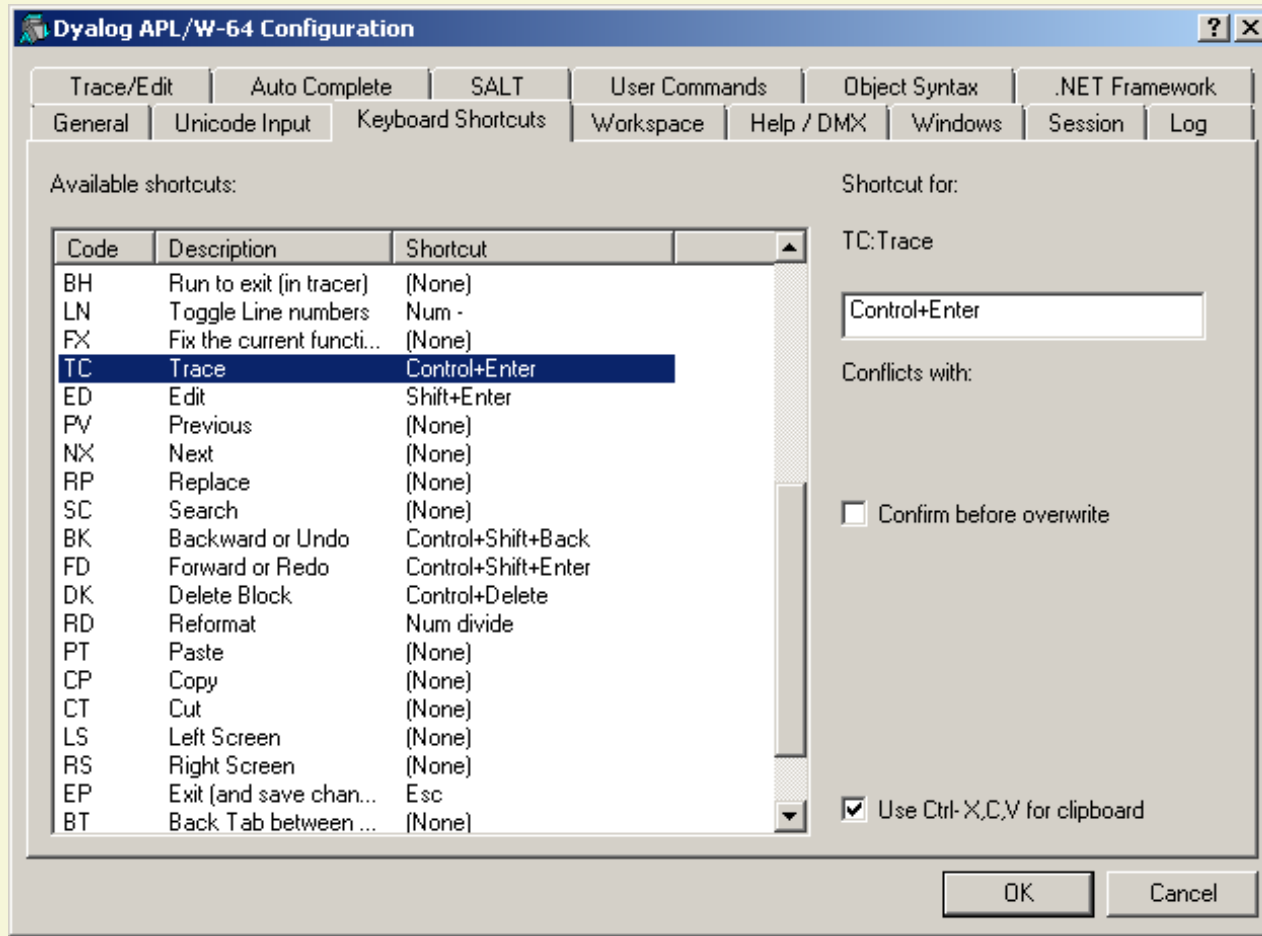


Characters

**Commands**

Colours

old



command codes

new

remove

reset to default

Preferences

Layout Shortcuts Code Colours Title Menu

Search

About	ABT	Shift-F1	x	+	↶
Align comments	AC			+	↶
Add comments	AO			+	↶
Backward or Undo	BK	Shift-Ctrl-Backspace	x	+	↶
Toggle breakpoint	BP			+	↶
Back Tab between windows	BT	Shift-Ctrl-Tab	x	+	↶
Connect	CNC			+	↶
Toggle key display mode	DMK			+	↶
Next line in demo	DMN			+	↶
Previous line in demo	DMP			+	↶
Load demo file	DMR			+	↶
Delete comments	DO			+	↶

OK Apply Cancel

commands:

show preferences

align comments

toggle fold

expand selection

...

Unicode Private Use Area

U+F800 ... U+F8FF

Characters

Commands

**Colours**

slide from "In-browser code editing"  
Marijn Haverbeke, GOTO conference, 2012

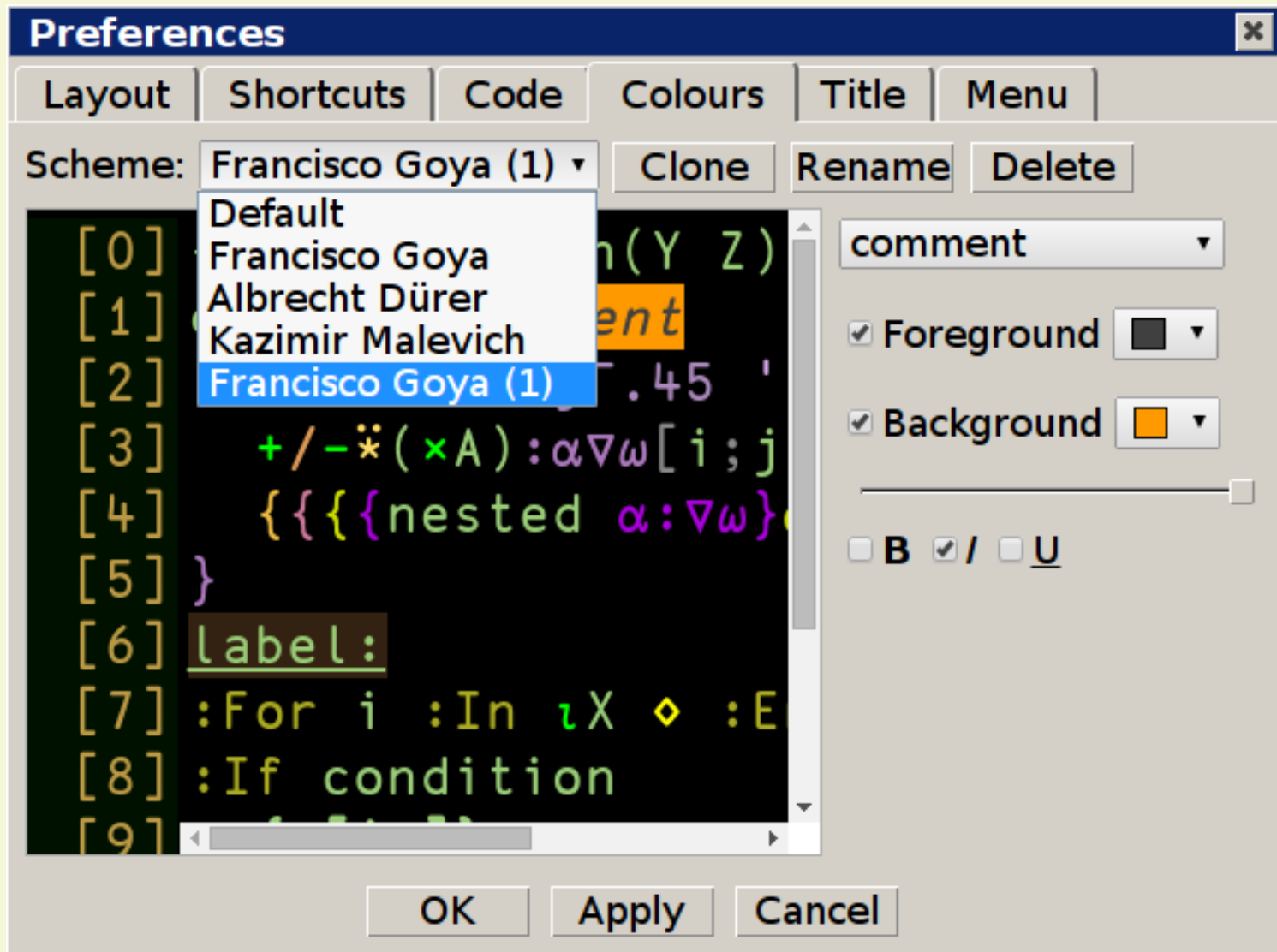
**startState**: `function()` → `state`

**token**: `function(stream, state)` → `string`



**mode state updates**





# DIALOG

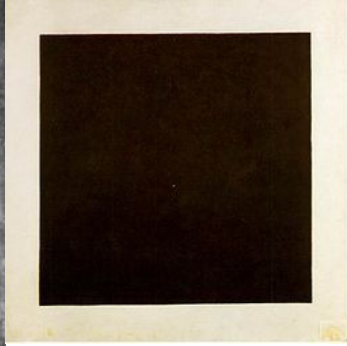
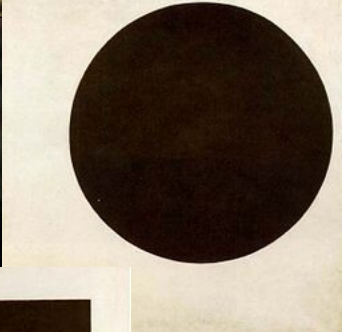
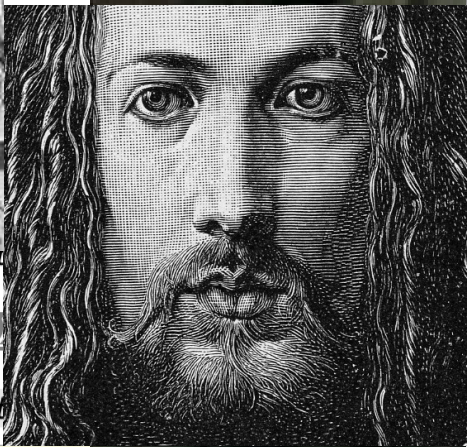
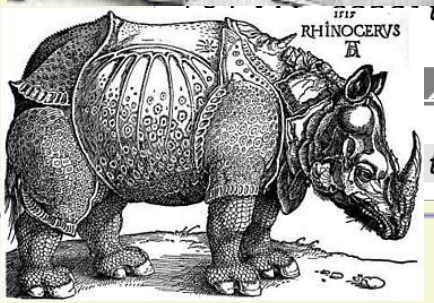
Scheme: Default

```
[0] {R}+{X}tradfn(  
[1] dfn←{ A commen  
[2] 0 -1.2e-3j-.  
[3] +/-*(xA):α∇ω  
[4] {{{nested α  
[5] }  
[6] label:  
[7] :For i :In ιX  
[8] :If condition
```



◇ glo  
] }  
ch

dfn(  
commen



```

[0] fnrepl←{
[1]     α←⊞THIS
[2] ▢     1:shy←{ω←c''}α.{⊞ML←3
[3] ▢         (↓⊞NL 3 4){
[4] ▢             (⊞NR α){
[5] |                 α≡ω:''
[6] |                 ⊞FX ω
[7] |             }(⊞NR α){
[8] |                 fm to←ω
[9] |                 cvex←{(+\fm∈ω)cω}fm,α
[10] |                 (pto)↓∈{to,(pfm)↓ω}''cvex
[11] |                 }''cω
[12] |             }''cω
[13] |         },''ω
[14] }

```

*A Function str  
A default curre  
A shy rslt: cha  
A target space  
A function sou  
A no change: ig  
A name of chang  
A function line  
A target and re  
A partitioned  
A collected wi  
A each line of  
A each function  
A find and rep*

## rainbow braces { $\alpha\omega\nabla$ :}

```

[0] fnrepl←{
[1]     α←□THIS
[2] ▢ 1:shy←{ω←c''}α.{□ML←3
[3] ▢     (↓□NL 3 4){
[4] ▢         (□NR α){
[5] |             α≡ω:''
[6] |             †□FX ω
[7] |         }(□NR α){
[8] |             fm to←ω
[9] |             cvex←{(+\fm∈ω)←ω}fm,α
[10] |             (pto)†∈{to,(pfm)†ω}''cvex
[11] |         }''←ω
[12] |     }''←ω
[13] |     },''ω
[14] }

```

*A Function str  
 A default curre  
 A shy rslt: cha  
 A target space  
 A function sou  
 A no change: ig  
 A name of chang  
 A function line  
 A target and re  
 A partitioned  
 A collected wi  
 A each line of  
 A each function  
 A find and rep*

grab: my.dyalog.com

whinge: nick@dyalog.com  
support@dyalog.com