

DYALOG

DYALOG
Eastbourne 2026

EWC in 2026



Aarush Bhat

Dyalog Ltd.

Who am I?



Aarush Bhat

With Dyalog since Jan’23. I work on EWC and Testing

What is ⎕WC

- Dyalog APL's great GUI system
- But:
  - Windows only
  - No web support

The family

⎕WC create

⎕WS set

⎕WG get

⎕NQ raise an event

⎕DQ handle events

⎕EX expunge

What is ⍞WC

- Work-a-like for Dyalog APL's great GUI system
- But:
  - Cross-platform!
  - Web support!
  - All the great things of ⎕WC and MORE!

The family

⍞WC create

⍞WS set

⍞WG get

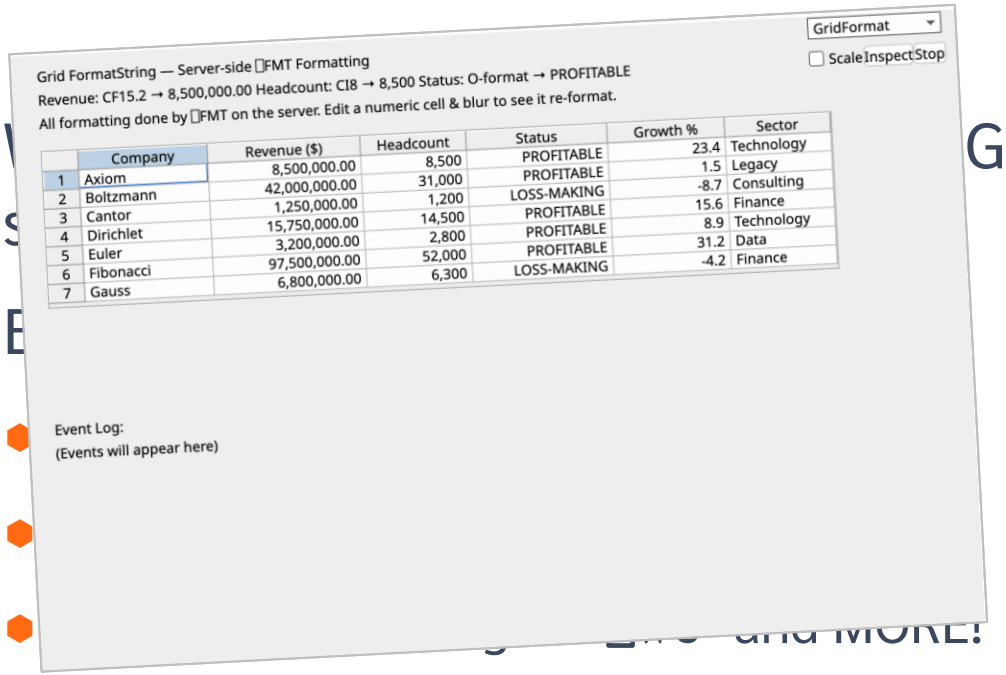
⍞NQ raise an event

⍞DQ handle events

⍞EX expunge

What is eWC

- W … GUI
- E …



The family

- **eWC** create
- **eWS** set
- **eWG** get
- **eNQ** raise an event
- **eDQ** handle events
- **eEX** expunge

What is eWC

- W
- E



GUI

The family

eWC create

eWS set

eWG get

eNQ raise an event

eDQ handle events

eEX expunge

What is eWC



The family

- **eWC** create
- **eWS** set
- **eWG** get
- **eNQ** raise an event
- **eDQ** handle events
- **eEX** expunge

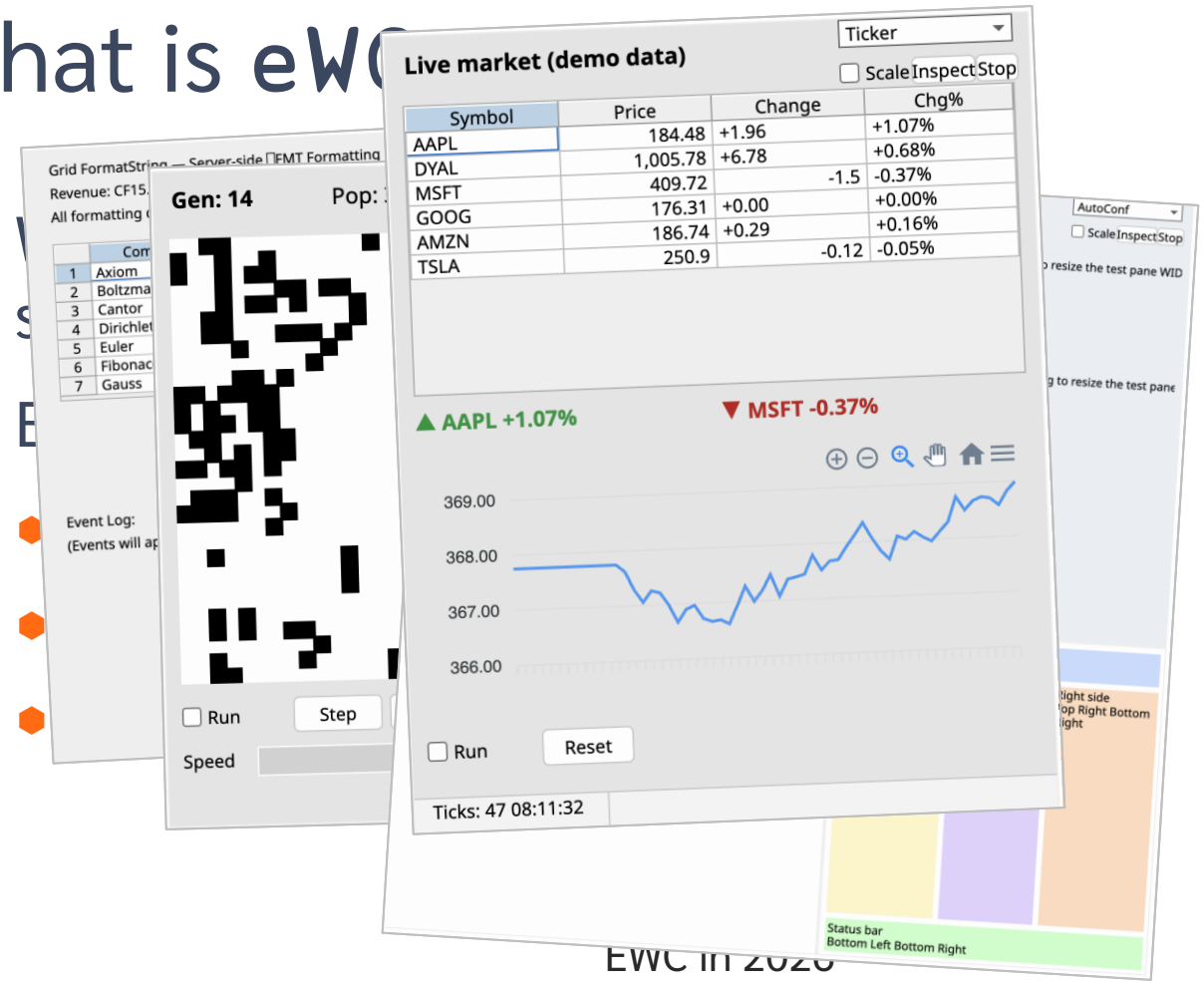
What is eWC



The family

- **eWC** create
- **eWS** set
- **eWG** get
- **eNQ** raise an event
- **eDQ** handle events
- **eEX** expunge

What is eWC



The family

- **eWC** create
- **eWS** set
- **eWG** get
- **eNQ** raise an event
- **eDQ** handle events
- **eEX** expunge


Ticker
EdgeStyle (rich) and Border (0/1) are INDEPENDENT properties
Each row = one component. Last two columns set BOTH properties.
EdgeStyle
Scale
Inspect
Stop
Ridge
Groove
Recess
Plinth
Shadow
None
Border=0
Border=1
B=1+Ridge
B=1+None
Default
B=1+Def
Edit
Label
List
Group
SubForm
EdgeStyle wins on appearance: B=1+Ridge always shows a Ridge border, regardless of Border value.
EdgeStyle="None" is treated as no EdgeStyle, so Border alone decides: B=1+None draws a plain frame, B=0+None draws nothing.
B=0 + no EdgeStyle: no border drawn (default state for Edit/Label/SubForm).
EdgeStyle="Default" (and any unrecognised value) is treated as no 3-D edge: Border alone decides, so B=1+Default draws a plain frame.
Status bar
Bottom Left Bottom Right

What we will cover

The basics

- What ▯WC & EWC is
- Who builds it, and how it works end to end
- How are EWC and ▯WC related

This year

- The Boring Release: a year of unbreaking EWC
- What landed, how we work, and what LLMs changed
- Porting an app and plugins
- Two demos, then now and next

EWC: Dyalog'24

Why EWC?



1. Protect existing investments

- Since 1990, ⎕WC has made it possible for domain experts to be "full stack" developers
- ⎕WC is perhaps the most important reason why Dyalog has been successful
- ⎕WC is is restricted to the Microsoft Windows desktop, which is fading

1 Everywhere WC DYALOG

EWC: Dyalog'24

Why EWC?

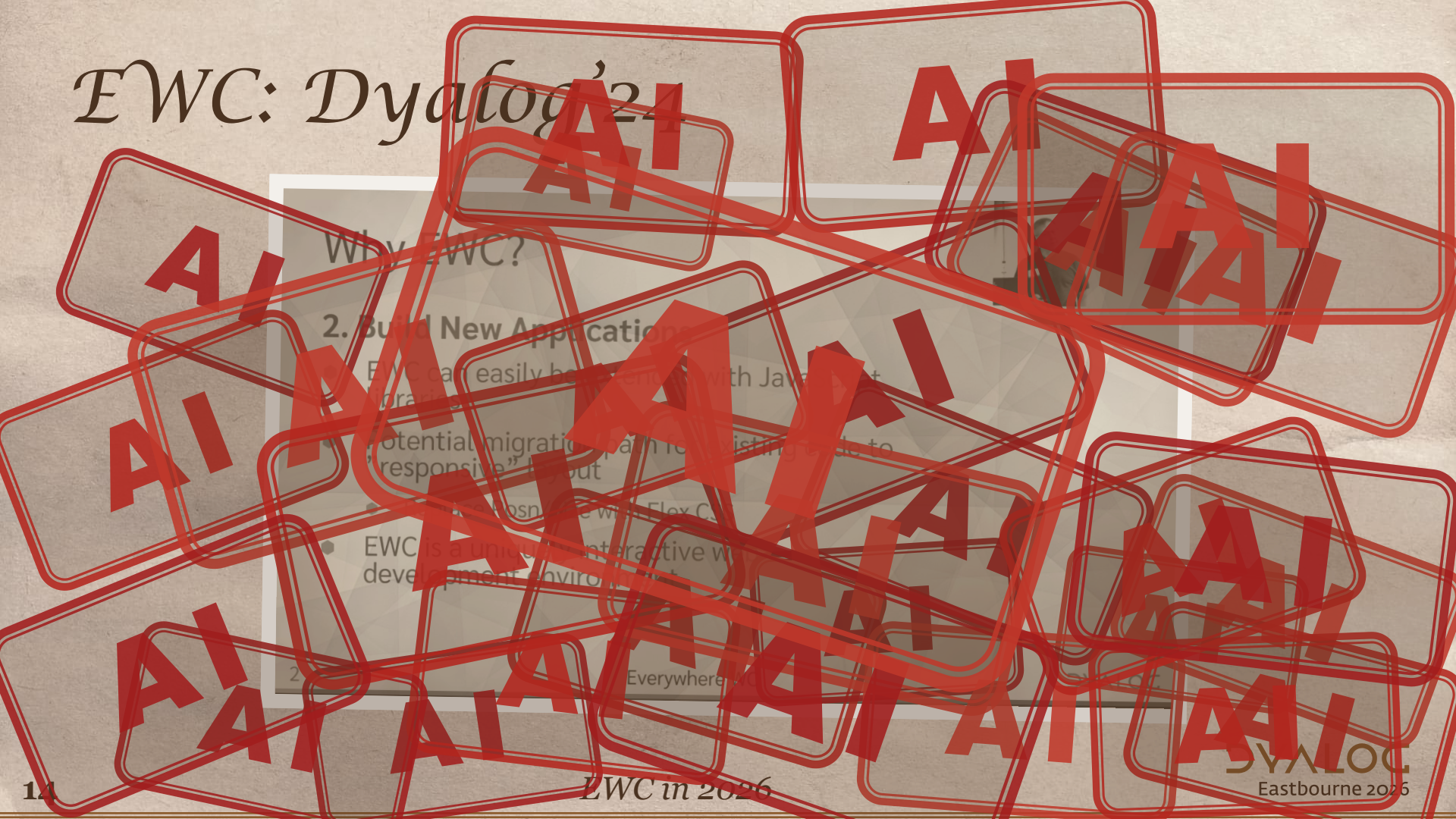


2. Build New Applications

- EWC can easily be extended with JavaScript libraries
- Potential migration path for existing code to "responsive" layout
  - Replace Posn/Size with Flex CSS
- EWC is a uniquely interactive web development environment

2 Everywhere WC DYALOG

EWC: Dyalog'24



Who builds EWC



Morten Kromberg

CTO of Dyalog. Original developer of EWC.



Neil Kirsopp

Dyalog's resident JavaScript expert. Has been working on EWC since Oct'24.



Aarush Bhat

With Dyalog since Jan'23. Joined team EWC this year. Develops EWC and steers where to take it.

A small team, working in the open on dyalog/ewc.

How EWC works



Your APL app always talks APL!

A “drop-in” ⎕WC replacement

- `EWC` creates `eWC` `eWS` `eWG…` in your namespace
- Same object names
- Same properties
- Same events
- Exactly same syntax `⎕WC→eWC`

```
'e',1↓'⎕WC'
eWC
☺
```

&more



Runs anywhere
Windows, Linux and macOS



Runs in a browser
Same GUI in a window or a browser tab



Three modes
Desktop, Browser or Multi, one codebase



Native components
Windows ribbon, now on every platform



HTML inside
A Div control embeds HTML directly



Flex layout
Optional Flex layout inside a Div



Plugins
Your own components, or ask us for one

The Boring Release

A year of unbreaking EWC

Docker · Playwright · visual regression · LLMs · a lot of unglamorous fixing

What landed and how we work

More of □WC covered

- Parity with □WC is close for the everyday classes

Publishing

- Single repository, one version
- Tagged releases

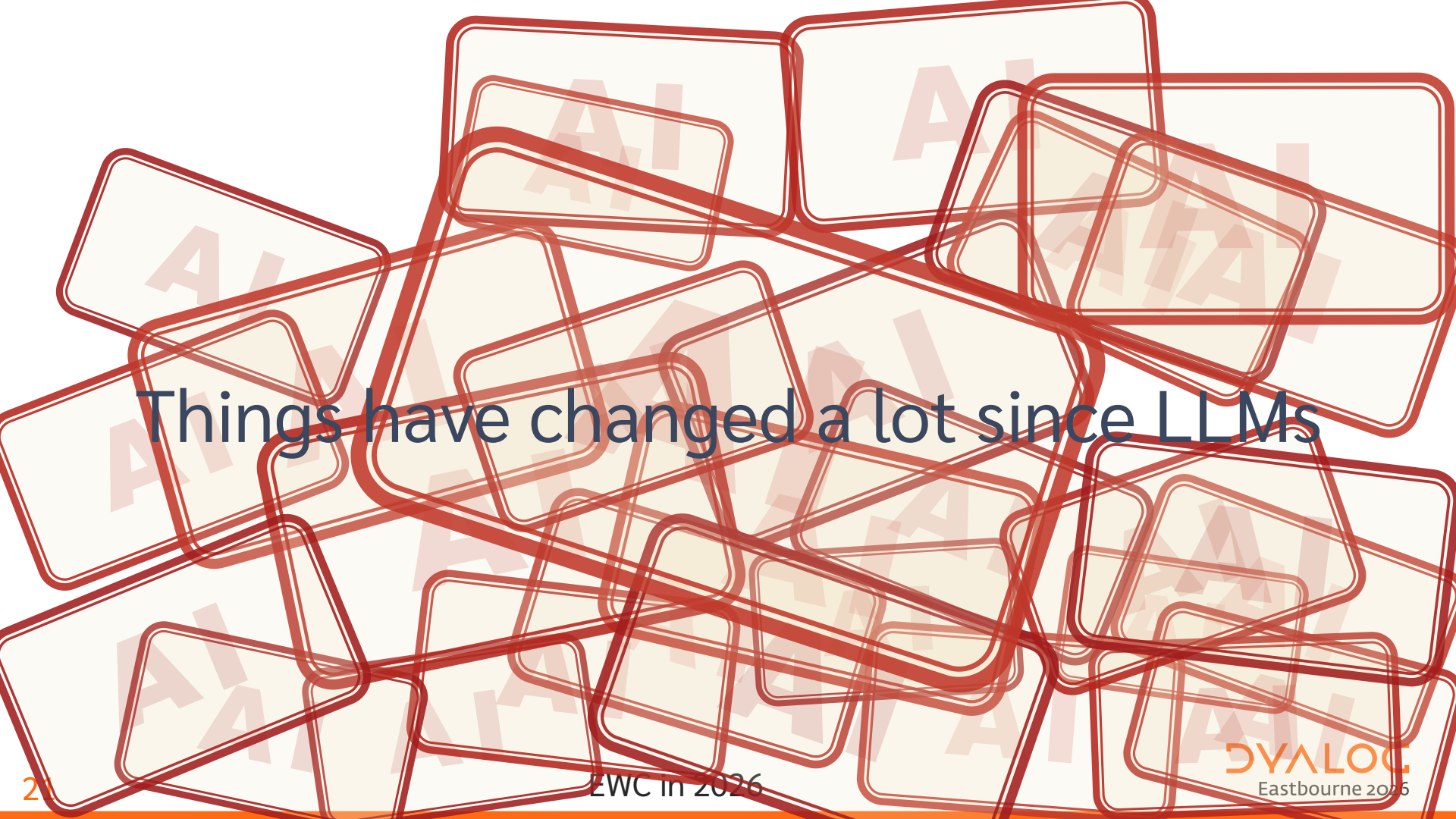
LLMs in the loop

- Help with components, reviews and tests

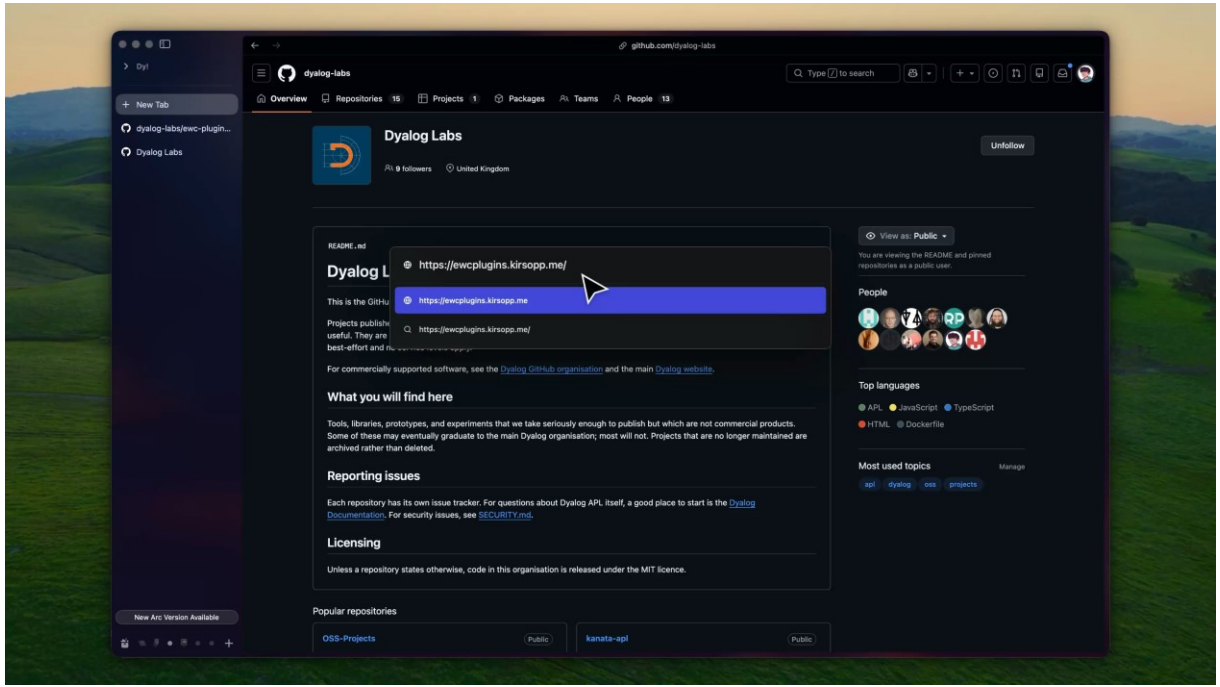
Testing

- Playwright suite, 40 tests on Grid alone
- Visual regression in CI, human-approved

Things have changed a lot since LLMs



Expand EWC as you like - Plugins!?



- Write them yourself
- Or… Just ask us for one

Porting a □WC app

- Link the `ewc` folder and call `EWC.Init` with a mode
- Swap `□WC` `□WS` `□WG` `□NQ` `□EX` `□DQ` for their `e`-prefixed workalikes
- EUREKA! It works!
- Small problems, not a big deal ☺
- Arachnid 🕷 moved across in 0 lines of change

```
⍝ three lines
]link.create #
  /path/to/ewc
EWC.Init 'Desktop'
'F1' eWC 'Form'
  'Hello' (10 10)
```

We are experimenting!

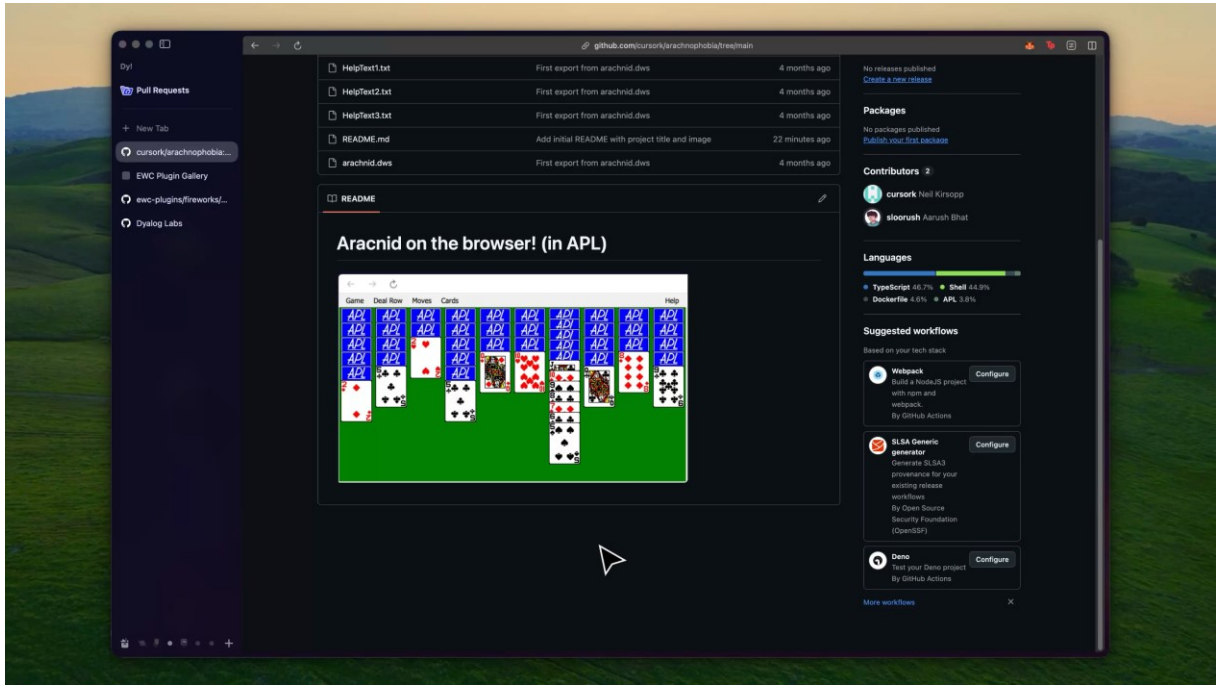
Arachnid

The famous Dyalog APL Arachnid from 1993, ported. Same code as the ⎕WC app, just on a new platform.

With only ⎕→e changes

Demo time!

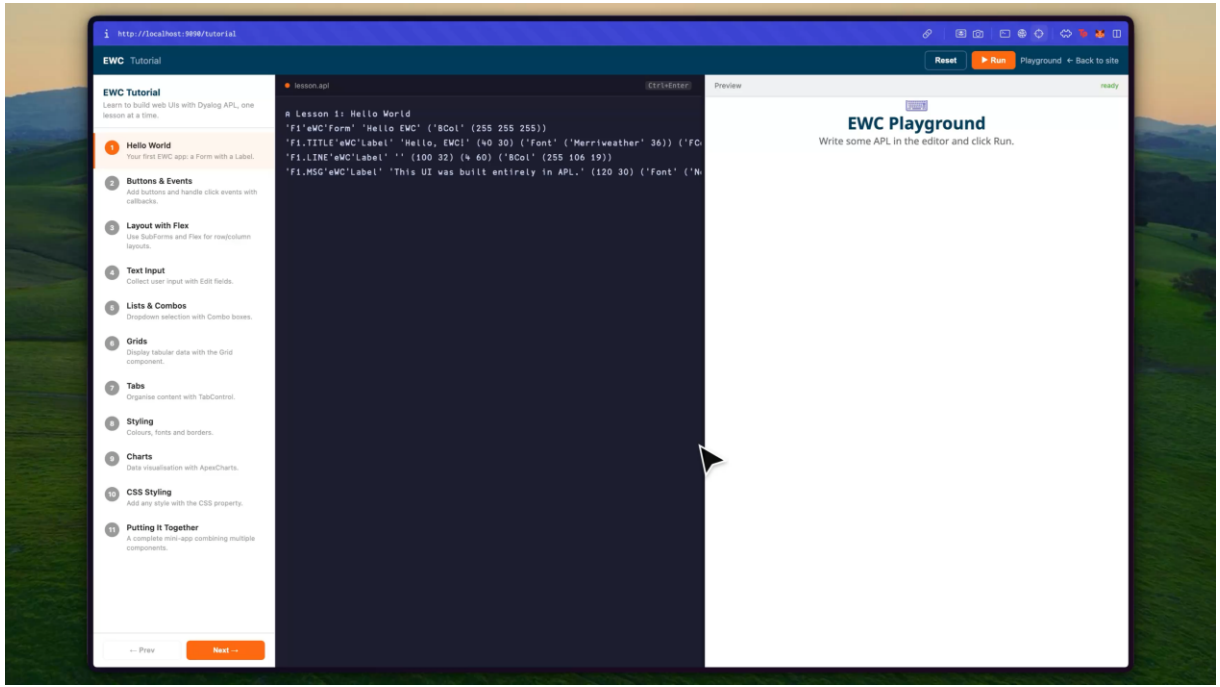
Arachnid on the browser!



□ → e

.dws → text files

An EWC tutorial in EWC



Now, and next

Where we are

- Parity with ⎕WC is close for the everyday classes
- Porting is cheap
- Releases are safe, tested, and deliberately boring
- Still early-adopter territory: not yet supported through normal Dyalog channels

What is next

- Close the remaining gaps in the object reference, class by class
- Your apps: Onboard - tell us what breaks and it goes on the list

Things to take away

- **“Drop-in”, no rewrite.** Swap `⎕WC` and friends for their e-prefixed workalikes; your app keeps talking APL.
- **Three modes.** Desktop, Browser and Multi from the same code, on Web, Windows, Linux and macOS.
- **A boring year, on purpose.** Docker, Playwright and visual regression in CI, so releases are stable.
- **Early-adopter territory.** Parity is close for everyday classes; remaining gaps close class by class and based on users.




eWC

Questions

> *The joy of porting software is discovering all the hidden assumptions you didn't know you made about everything.*
>
> **— Anonymous Developer**

Try it. Port something small. Tell us what breaks.

Aarush Bhat · Dyalog Ltd. · aarush@dyalog.com
Neil Kirsopp · Dyalog Ltd. · neil@dyalog.com

ewc@dyalog.com