

DIALOG

Elsinore 2023



Brian Becker

APL Tools Architect, Dyalog LTD



Elsinore 2023

APL Tools Update

Brian Becker

APL Tools Architect, Dyalog LTD



An APL Tool...

- is a software component that the user can:
 - embed in their APL application
 - use to develop their APL application
- is generally written in APL, but not always
- is not an application in and of itself



APL Tools at Dyalog

- ◆ 60+ workspaces delivered with Windows Dyalog
 - ◆ Core utilities, tutorials, games, demonstrations
- ◆ Built-in
 - ◆ included as user commands or]LOAD'able
- ◆ ~20 GitHub Repositories
- ◆ 2 Tatin APL Packages (so far)



Web Enabling Tools

- Jarvis – JSON and REST web service framework
- HttpCommand – send HTTP commands and process server response
- DUI/MiServer – HTML web server and content creation
- SAWS – SOAP/XML web service framework



Data Tools

- ◆ loaddata.dws – read and write Excel, csv, XML
- ◆ newleaf.dws – format and produce PostScript documents
- ◆ sharpplot.dws – produce high quality SVG or PDF graphs
- ◆ Math – eigen, domino, Fourier



Interface Tools

- ◆ conga.dws – TCP/IP toolkit
- ◆ sqapl.dws – SQL ODBC
- ◆ aplservice.dws – run as a Windows service
- ◆ APLProcess – spawn an interpreter in another process
- ◆ isolate.dws – parallel execution in multiple APL processes
- ◆ pynapl – Python/APL interface
- ◆ DUI – cross platform UI



Development/Deployment Aids

- Link – map workspace to folder structure
- DBuildTest – framework to build and test applications
- Cider – project management
- Tatin – package management



Tool Availability

- ◆ To the extent possible, we try to support:
 - ◆ The latest 3 releases of Dyalog APL, plus future releases in the pipeline. Today that means 17.1, 18.0, 18.2, 19.0 and 20.0
 - ◆ All applicable platforms: Windows, Mac, Linux, Pi and AIX
 - ◆ Classic and Unicode
 - ◆ 32 and 64 bit
 - ◆ Containerized, headless, scripted

Releases ◦., Platforms ◦., Classic/Unicode ◦., 32/64



2024 Goals

- Work with customers to identify tools that will help them succeed
- Make it easier to find what tools are available
 - Website, blogs, forums
- Make tools easier to learn
 - Documentation, webcasts, samples and tutorials
- More comprehensive testing
 - broader test suites
 - CITA – Continuous Integration Testing in APL



Specific 2024 Goals

- ⬢ HttpCommand – add features per customer needs, webcasts
- ⬢ Jarvis – "industrialize", finish documentation, webcasts
- ⬢ WebSocketServer – come up with a better name, publish version 1
- ⬢ APLProcess – document and publish release
- ⬢ Packages – publish more Dyalog packages on Tatin (you're not retiring that easily Morten 😊)



HttpCommand

- ◆ Designed with APLers in mind to issue an HTTP request and return the response from the host in a manner useful to the user.
- ◆ Used for:
 - ◆ Interacting with web services
 - ◆ Downloading files from the web
 - ◆ Scraping web pages
 - ◆ Any HTTP interaction
- ◆ Demo now -]demo /dyalog23/httpcommand



HttpCommand

```
url←'https://api.github.com/orgs/Dyalog/repos'  
params←('per_page' 5) ('sort' 'pushed') ('direction' 'desc')
```

```
⊖ r←HttpCommand.GetJSON 'get' url params  
[rc: 0 | msg: | HTTP Status: 200 "OK" | #Data: 59]
```

```
↑r.Data.(name pushed_at)
```

HttpCommand	2023-10-14T20:13:50Z
Jarvis	2023-10-14T14:47:09Z
CITA	2023-10-13T12:44:56Z
link	2023-10-12T15:12:02Z
dyalog-jupyter-kernel	2023-10-11T15:53:12Z



Jarvis

- ◆ A few years ago, a customer wanted to make the APL functions their group had written available to other (non-APL) parts of the company.
- ◆ As often happens, Morten had an idea on how to address their need... (and asked me to make it happen)
- ◆ Over a weekend, I wrote the JSONServer, and it was well received.
- ◆ Some time later another customer wanted the capability to deliver a RESTful web service with JSONServer.
- ◆ JSONServer → Json And Rest serVice → JARVIS



Jarvis

- Jarvis is a framework that makes it easy to expose your APL functionality as a web service.
- How easy? Let's see...
- Demo time -]demo /dyalog23/jarvis



Suppose...

- ◆ You have a web application with a HTML/CSS/JavaScript client.
- ◆ If you use standard HTTP requests, the only way to get updated information from the server is to ask for it.
- ◆ Wouldn't it be nice if the server could "push" updated information in real time without the client having to ask for it.
- ◆ WebSockets can accomplish precisely that (and more)



Sneak Peek at WebSocketServer

- ◆ WebSocketServer is, well, a WebSocket server.
- ◆ What's a WebSocket? I'm glad you asked...
- ◆ WebSocketServer is designed to facilitate different modes of interaction
 - ◆ Publish/Subscribe (PubSub)
 - ◆ Asynchronous Remote Procedure Control (ARPC)
 - ◆ any others we think will be useful and implementable
- ◆ It *may* become integrated with Jarvis
- ◆ Demo time -]demo /dyalog23/wss



Tools, Tools, and More Tools

- There are several other sources of tools in the APL community
 - Adám Brudzewsky – <https://github.com/abrudz>
 - Kai Jeager – <https://github.com/aplteam>
 - Tatin – <https://tatin.dev>
 - Carlisle Group – <https://github.com/the-carlisle-group>



Summary

- ◆ Our customers are our partners – we're in this together
 - ◆ If you have a need, tell us. Jarvis exists today because a customer communicated a need to Morten...
 - ◆ Many features in our tools are in direct response to feedback from our customers.
 - ◆ Report bugs, suggest features, contribute code

