Overview

- File portability between platforms
  – Windows, and UNIX®, Linux, OS X
- Why now?
  – Code repositories
  – Raspberry Pi and OS X
- Part A: Dyalog files
- Part B: User files

History

- Dyalog originally on UNIX systems
- Different hardware platforms but workspace portability from the outset
- File extensions not significant to OS
- Windows: another hardware platform with many differences…
- … including filename extensions

14.0 Demo

Cross-platform, up to V14.0

- Workspaces are portable between platforms

- Windows filename extensions are dealt with largely transparently

- Great for moving to a new platform

- File name differences not so well suited to code sharing – now a "must have"

- 14.1 started to address that

14.1 demo

Cross platform, in V14.1

- Started to deal with inconsistencies in file naming

- Workspace naming only

- Configurable using WSEXT

15.0 and beyond

- Extend to all Dyalog file types?
- Extensions by default on all platforms?

File system access

- Native files
- ⎕SH 'ls *.txt'
- ⎕CMD 'mkdir \tmp\APLtemp'
- ⎕NA

15.0 demo

Portable file functions – 15.0

- APL functions to access host file system
  - Parse and "normalise" filenames
  - Delete files
  - Create and delete directories
  - Obtain file and directory properties
  - Selectively list directory contents

Portable file functions – 15.0

- Still being designed
- Portable between platforms
  - Consistent syntax
  - Consistent naming (slashes, globbing)
  - Consistent result format

Summary

- Cross-platform file extensions
  - .DWS vs no file extension, etc.
- Portable file functions
  - `⎕FILEPARTS, ⎕MKDIR, ⎕DEL, ⎕DIR`