External Workspaces

Morten Kromberg
Dyalog Ltd.

Stig Nielsen
SimCorp A/S

DYALOG

Sicily 2015

External Workspaces

A workspace format which allows efficient loading of functions, variables (etc) ***the first time they are used***

(rather than loading everything up front, most of which is not used in any particular user session)

How to Use Them…

- External Workspaces are generated from an active workspace using (8667⌶)

```
          )load myws
   [names] 8667⌶ 0 'myws.dxw'
```

- External Workspaces are attached (memory mapped) using (8666⌶)
  - More than one can be “attached”
- All names in the attached workspaces immediately appear in the active ws
- Used [parts of] definitions are loaded on use

Benefits

- Faster start-up
  - No need to load any code [at all!] “up front”
  - `⎕LX←'8667⌶''myws.dxw'' ⋄ Run'`
- More free space:
  - Unused code is never loaded
  - Even used code is not “in” the active workspace, but in a memory-mapped file.
  - Memory-mapped files can be shared by processes (first process to use a function loads it into memory)
- Faster runtime:
  - Workspace is smaller and simpler to manage
  - In modern machines, memory use = time

Limitations

- External Workspaces cannot be modified while any process is using them
- Cannot be shared across architectures (32/64, Classic/Unicode, Little/Big-Endian)
  - [64-bit Unicode Only in v14.1]
- Only 8 files can be mapped simultaneously in v14.1
  - Each has a fixed, unique virtual memory address
- No GUI objects or AP functions, ⎕NA or ⎕SM