Integrating EPICS and LabVIEW on Windows using DCOM

Freddie Akeroyd
ISIS Computing Group

Science & Technology Facilities Council
ISIS

Basic Idea

- Can we access our existing LabVIEW drivers from EPICS without modifying the VI and/or changing LabVIEW version?
- Already have experience of accessing LabVIEW front panels via DCOM, so try creating an IOC to do this
- Use EPICS ASYN driver framework to simplify writing

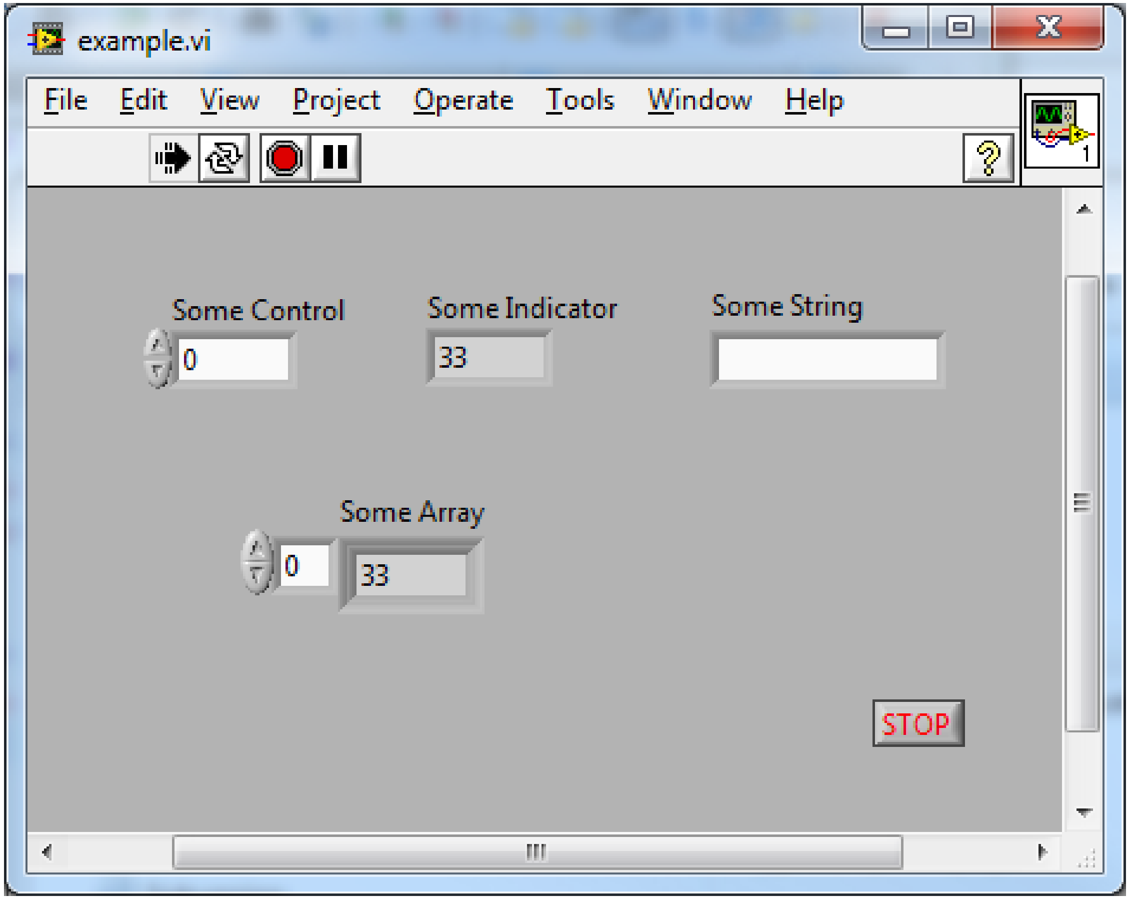
Note: not the only way

- National Instruments have added EPICS support to recent versions of LabVIEW
  - Channel access client is a free download
  - Server requires the DSC module
  - Only available in 32bit version of LabVIEW
- We are looking at using this for new VIs
  - Though can programatically convert old VIs too
  - Talk to Kathryn Baker (ISIS computing group) if you want to know more

Implementation

- LabVIEW DCOM interface added as ASYN driver
- ASYN “port name” and driver “parameters” mapped to LabVIEW front panel variables
  - Via separate XML configuration file
- Mapping loaded at IOC startup
  - Can also set other options at this point e.g. automatically start VIs if not running

Example VI


example.vi
File
Edit
View
Project
Operate
Tools
Window
Help
1
Some Control
0
Some Indicator
33
Some String
Some Array
0
33
STOP

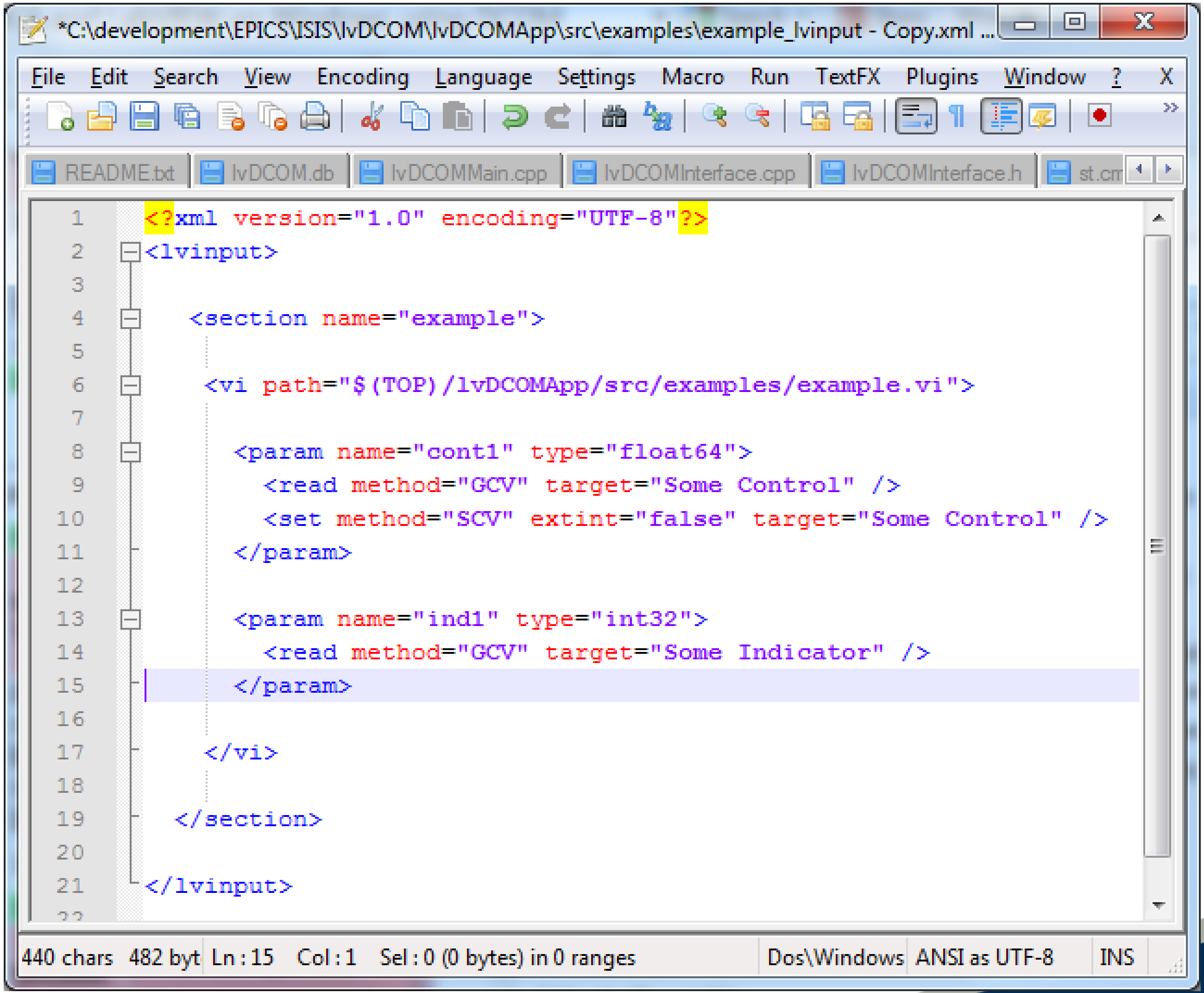
EPICS record example

```
record(ai, "$(P)IND1") {
   field(DTYP, "asynInt32")
   field(INP,  "@asyn(ex1,0,0)ind1")
   field(PREC, "3")
   field(SCAN, ".1 second")
}
```

In IOC startup st.cmd

```
lvDCOMConfigure("ex1", "example", "$(TOP)/lvDCOMApp/src/examples/example_lvinput.xml", "", 6)
```

XML Config File


*C:\development\EPICS\ISIS\lvDCOM\lvDCOMApp\src\examples\example_lvinput - Copy.xml ...
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ? X
README.txt
lvDCOM.db
lvDCOMMain.cpp
lvDCOMInterface.cpp
lvDCOMInterface.h
st.cm

```xml
<?xml version="1.0" encoding="UTF-8"?>
<lvinput>

   <section name="example">

    <vi path="$(TOP)/lvDCOMApp/src/examples/example.vi">

      <param name="cont1" type="float64">
        <read method="GCV" target="Some Control" />
        <set method="SCV" extint="false" target="Some Control" />
      </param>

      <param name="ind1" type="int32">
        <read method="GCV" target="Some Indicator" />
      </param>

    </vi>

  </section>

</lvinput>
```

440 chars 482 byt Ln : 15 Col : 1 Sel : 0 (0 bytes) in 0 ranges Dos\Windows ANSI as UTF-8 INS

Features

- Can communicate with either LabVIEW VIs or compiled LabVIEW applications
- Can automatically launch, start or stop VIs
- Access to full IOC functionality and other extensions e.g. autosave
- Clean interface between EPICS and LabVIEW
  - Low risk of interfering with existing operation

Features (cont.)

- Need to poll LabVIEW to notice value changes
  - LabVIEW events not directly visible over DCOM
- You have both an IOC and LabVIEW VI to maintain
  - The VI may be a third party VI though
  - The IOC should only need configuring
- Uses ATL for DCOM, so requires full version of Visual Studio to compile IOC
  - I can supply a statically linked executable

Possible Future Extensions

- Allow “I/O interrupt” record scanning
  - ASYN driver triggers record processing rather than periodic record scanning
  - Driver still needs to poll LabVIEW, but not tied to standard scan rates or mechanisms
- Allow LabVIEW to provide a timestamp rather than using the EPICS scan timestamp

Summary

- Provides a simple way to rapidly expose LabVIEW variables to the EPICS environment
- Still being developed – suggestions for additional features/improvements welcomed
- Happy to share code with community
  - Will be posting code on web
  - Or drop me an email (freddie.akeroyd@stfc.ac.uk)