

# Porting Existing Software to the DANSE Distributed Architecture

Mike McKerns

Materials Science

California Institute of Technology

# Introduction

The underpinnings of the DANSE software architecture are written in Python -- a platform independent, interactive, object-oriented programming language.

Some languages, such as C or C++, can easily be fully integrated ("bound") into Python, however other languages, such as Java, are currently unable to be bound by Python.

Does this mean that all existing software written in a "non-bindable" language must be excluded from the DANSE infrastructure? ... The answer is no!

Python allows for a limited, but very functional integration.

# Minimal Integration into DANSE

Python can be used as the medium to implement the desired existing software from the Cobra interface.

If the existing software can be called from a command prompt, python can be used in a simple manner.

```
#!/usr/bin/env python
import os
os.system("Isaw_exec.sh")
```

This implementation is limited however, because data may not be easily passed to the desired software.

# Thinking Small

By "peeling off" the GUI, and using a command prompt to call modules directly, flexibility of the code increases.

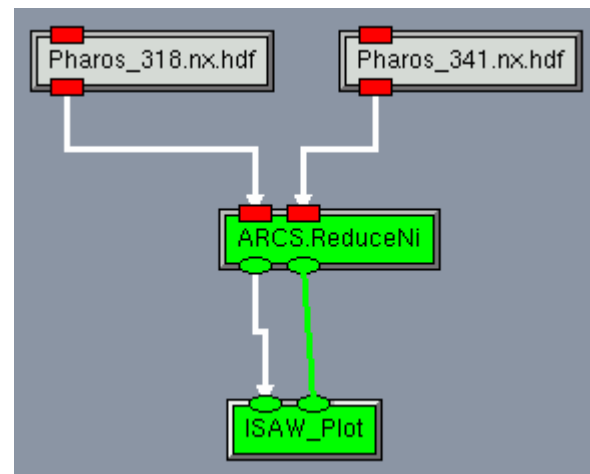
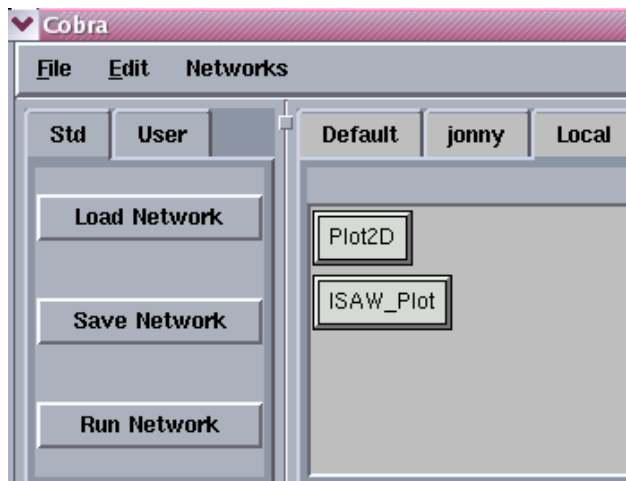
Providing a 'main' function in a module allows for an isolated call to the module of interest.

```
public static void main( String args[] ) {  
    << declarations >>  
    op = new ARCSloadASCII(filename, head, 0, 1, 2, 0);  
    obj = op.getResult();  
    System.out.println("Operator returned: " + obj );  
    if ( obj instanceof DataSet ) {  
        vm1 = new ViewManager( (DataSet)obj, iviewer );  
    }  
}
```

# Modularity is Functional

If the module requires a filename or a data stream, this can be passed to the module at the command line.

Using the existing software in a modular fashion allows the user to utilize the strengths of the particular software, while allowing the DANSE software to overcome any of that particular software's inherent limitations.



# It can't be that easy, can it?

If the existing software is designed in a modular fashion, and can be made to accommodate command line arguments, the new code required is minimal.

## Examples:

Calling ISAW (from Python)

Calling ISAW (from Python) to manipulate data

Calling ISAW (from Python) to view a data file or a data stream

Using Python in this fashion, however, is a "one-way street", where passing data back to Cobra may possibly only be done by first writing to a file.

# Summary

The DANSE software architecture utilizes the platform independence, interactivity, object-oriented nature of Python. Further, Python is very easy to learn and use.

Python can bind code modules together to provide a extremely functional package.

The DANSE project seeks to unite the neutron community, and the ability to integrate existing software into the DANSE structure strengthens these bonds.