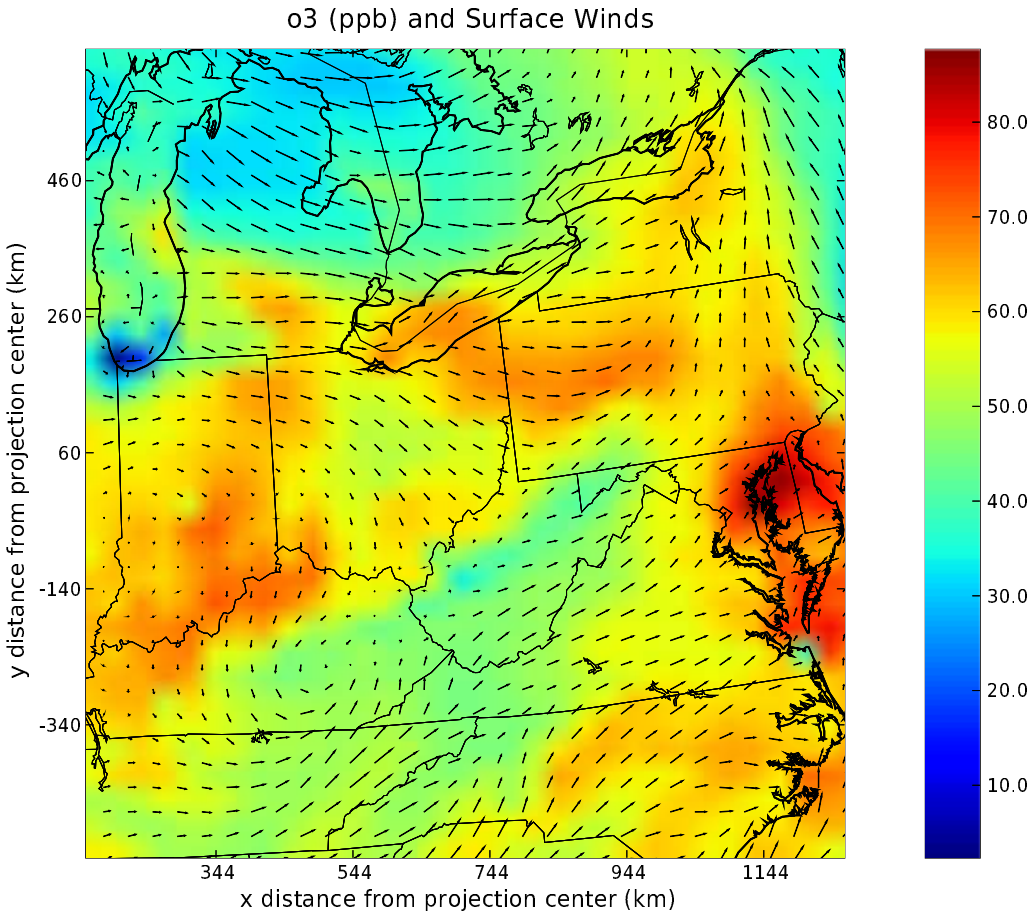
ioapiTools:
A python analysis environment for CMAQ, SMOKE, and MCIP

Alexis Zubrow
CISES, University of Chicago

Although the research described herein has been funded wholly or in part by the United States Environmental Protection Agency through STAR Cooperative Agreement #R-82940201-0 to the University of Chicago, it has not been subjected to the Agency's required peer and policy review and therefore does not necessarily reflect the views of the Agency, and no official endorsement should be inferred.

the Need

- IOAPI format
- compiled code
- analysis seperated from plotting



Functionality

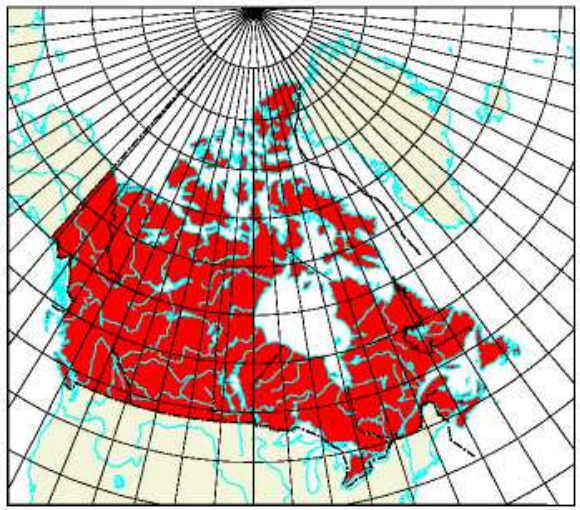
- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- Plotting
- Interpreted Language

Functionality

- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- Plotting
- Interpreted Language

Functionality

- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- Plotting
- Interpreted Language



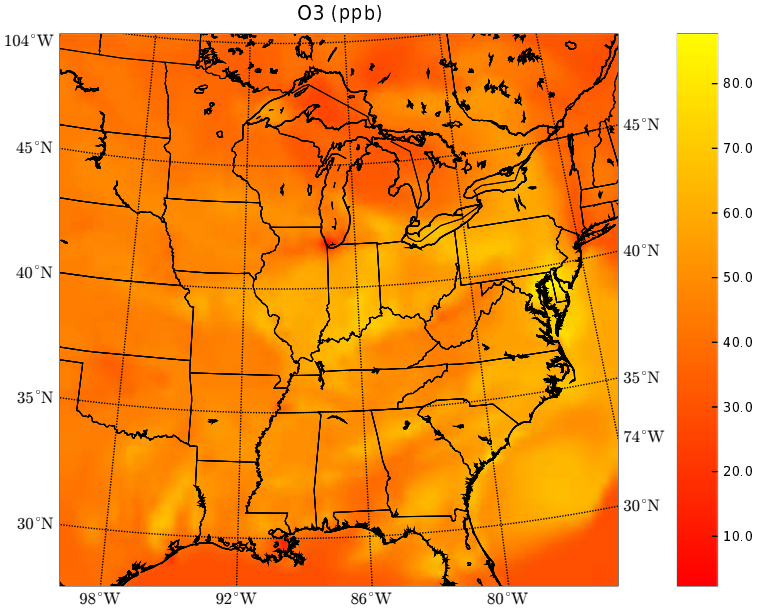
reference: atlas.gc.ca

Functionality

- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- Plotting
- Interpreted Language

Functionality

- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- **Plotting**
- Interpreted Language

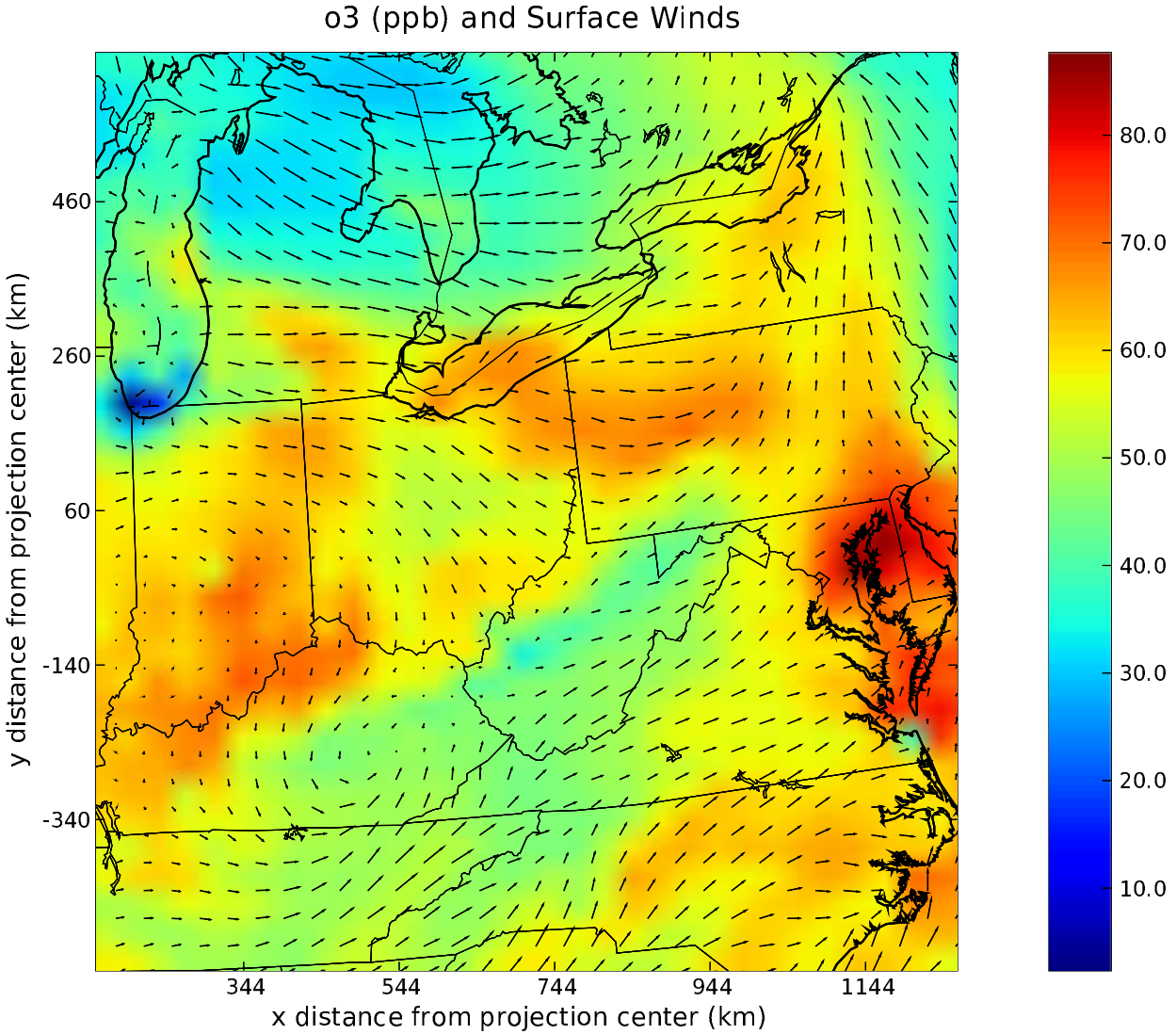


Functionality

- Reading/Writing IOAPI files
- Export to CF netCDF
- Subsetting & Coordinate Transforms
- Access to Metadata
- Plotting
- Interpreted Language

why python?

- Interpreted Language
- Clarity
- Glue
- Extendible


o3 (ppb) and Surface Winds
460
260
60
-140
-340
y distance from projection center (km)
344
544
744
944
1144
x distance from projection center (km)
80.0
70.0
60.0
50.0
40.0
30.0
20.0
10.0

Example Script

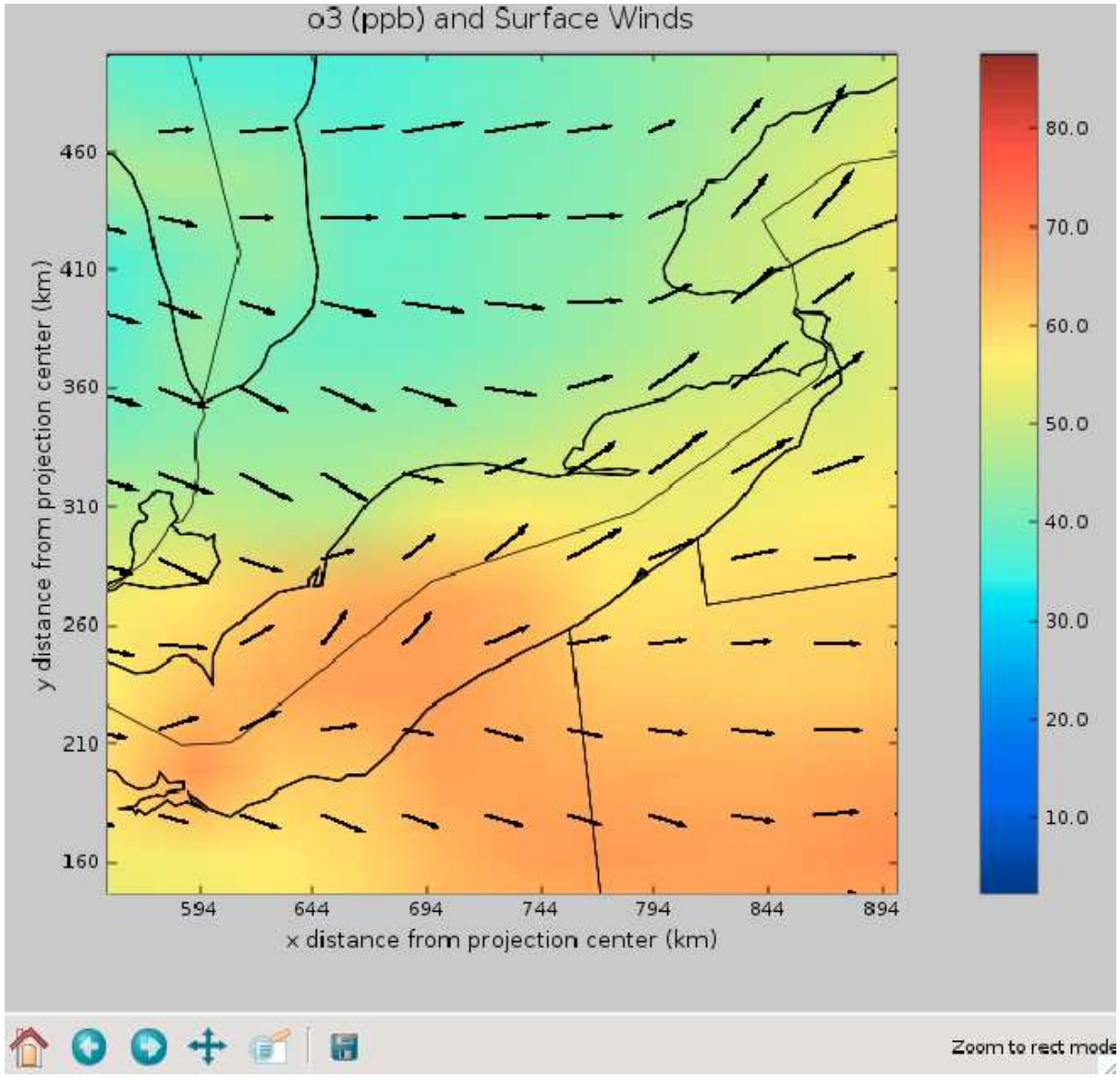
```
## load module
from ioapiTools import *

## open 2 IOAPI files
f = open("CCTM_ACONC.D1.038")
g = open("METDOT3D_D1.038")

## Get o3 from first file
o3 = f("o3")

## Get 2 components of the winds from the MCIP file
u =g("uwind")
v =g("vwind")
```

Script (cont)

```
## Convert o3 from ppm --> ppb
o3 = o3 * 1000.
o3.units = "ppb"

## Subset the variables
o3_sub = o3.IOsubset(domainLst=[(-88, 35), (-73, 45)])

u_sub = u.IOsubset(domainLst=[(-88, 35), (-73, 45)], \
                   layerLst=0)
v_sub = v.IOsubset(domainLst=[(-88, 35), (-73, 45)], \
                   layerLst=0)
```

Script (cont)

```
## Contour plot of o3
cDct = o3_sub(12).contour(title="o3 (ppb) and Surface Winds")

## Plot the wind vectors on top of the contour
ovector(u_sub(12), v_sub(12), cDct)

## Save subsetted o3 to new IOAPI file
k = open("o3_slice.ioapi","w",iofileFlag)
k.write(o3_sub)
k.close()
```


o3 (ppb) and Surface Winds
460
410
360
310
260
210
160
y distance from projection center (km)
594
644
694
744
794
844
894
x distance from projection center (km)
80.0
70.0
60.0
50.0
40.0
30.0
20.0
10.0
Zoom to rect mode

Websites

python: www.python.org

CDAT: www-pcmdi.llnl.gov/software-portal/cdat/

email contact: azubrow@uchicago.edu