Dust & Vapor

- Laser measurements can be weakened by interacting with dust and vapor particles, which scatter the laser beam and the signal returning from the target
- Using last-pulse measurements can reduce or eliminate this interference
- Systems that are expected to work in such conditions regularly can be optimized for these environments

Traditional Photogrammetry vs. LiDAR

LiDAR	Photogrammetric
-Day or night data acquisition	-Day time collection only
-Direct acquisition of 3D collection	-Complicated and sometimes unreliable procedures
-Point cloud difficult to derive semantic information; however, intensity values can be used to produce a visually rich image like product (example of an intensity image)	-Rich in semantic information
-Active sensor	-Passive sensor

Lidar vs. Similar Technologies

LiDAR

-Use optical signals (Near Infrared, visible). Wavelength≈1μm.

-Focused beam and high frequency permit high spatial resolution (< 1m horizontal).

-Limited to clear atmospheric conditions, daytime or nighttime coverage.

Radar

-Use Microwave signals. Wavelength≈1cm(Approx. 100,000 longer than Near IR).

-Beam width limits spatial resolution.

-Can operate in presence of clouds. Daytime or nighttime coverage.

Data Ordering Details

Formats

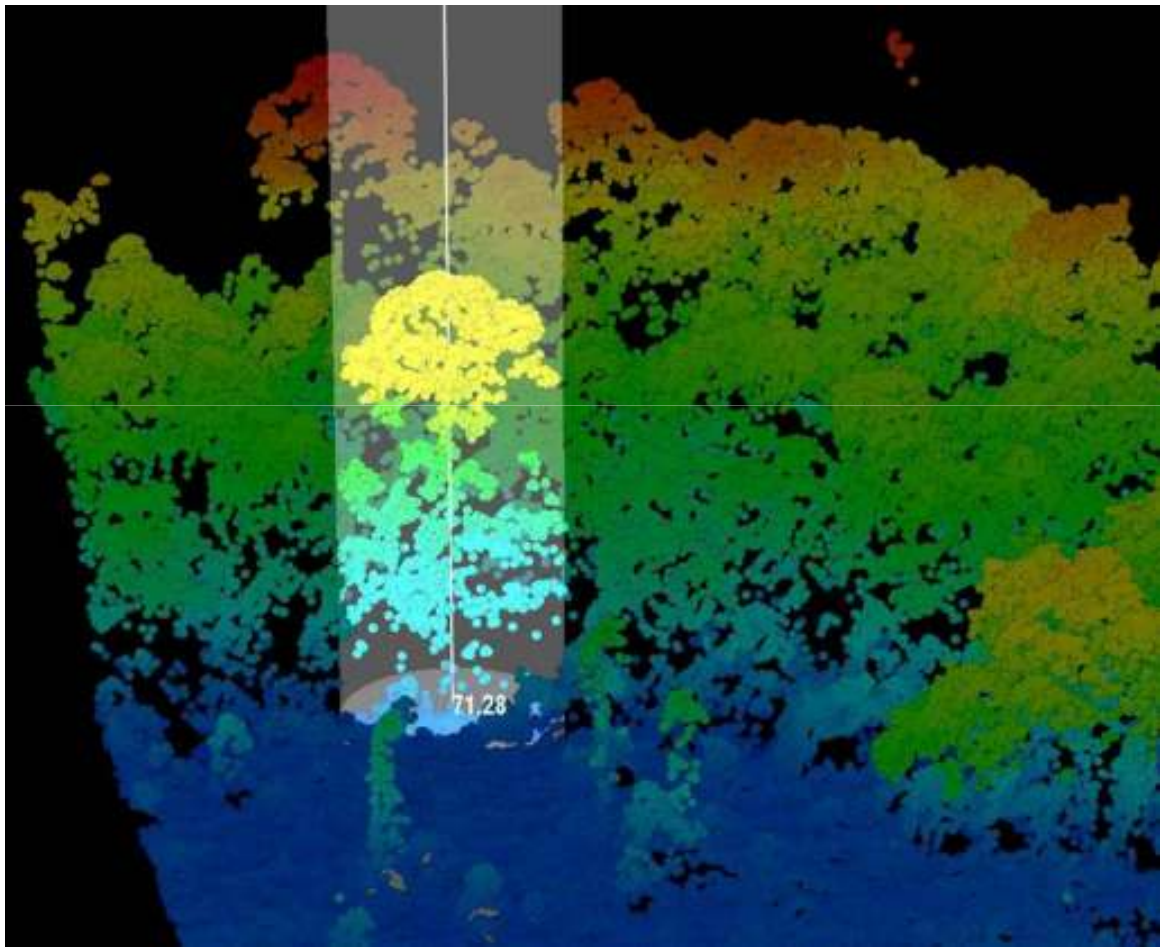
Common LiDAR Data Exchange Format - .LAS Industry Initiative (ASPRS). The LAS file format is a public file format for the interchange of LiDAR data between vendors and customers. This binary file format is an alternative to proprietary systems or a generic ASCII file interchange system used by many companies.

- *Know & understand the flight acquisition parameters*
- *Always get the raw data (it can be reprocessed later with newer techniques/algorithms)*
- *Get an intensity image*

Projections

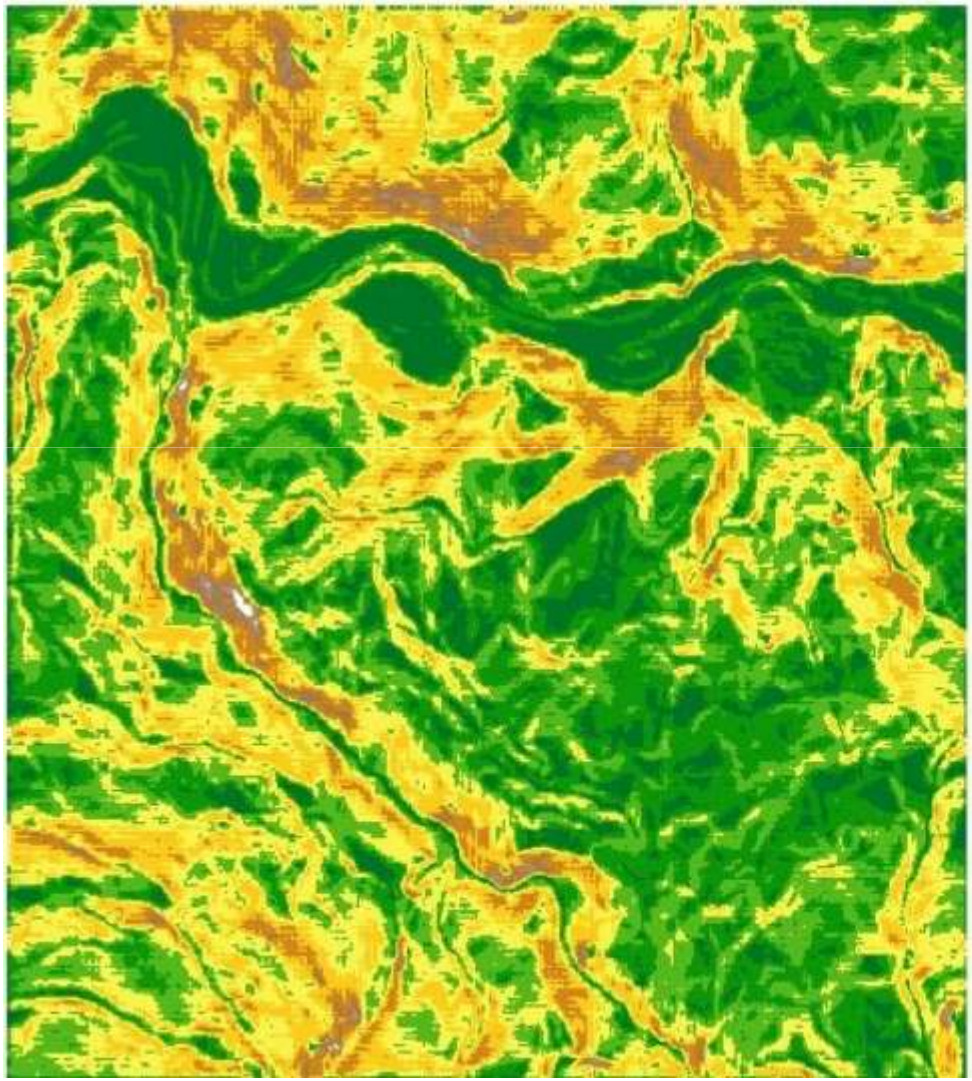
LiDAR data can be delivered in many different projections and datums. The national standard for vertical datum is the North American Vertical Datum (NAVD 88), and the national standard for horizontal datum is the North American Datum of 1983 (NAD 83).

Applications



Forestry Management

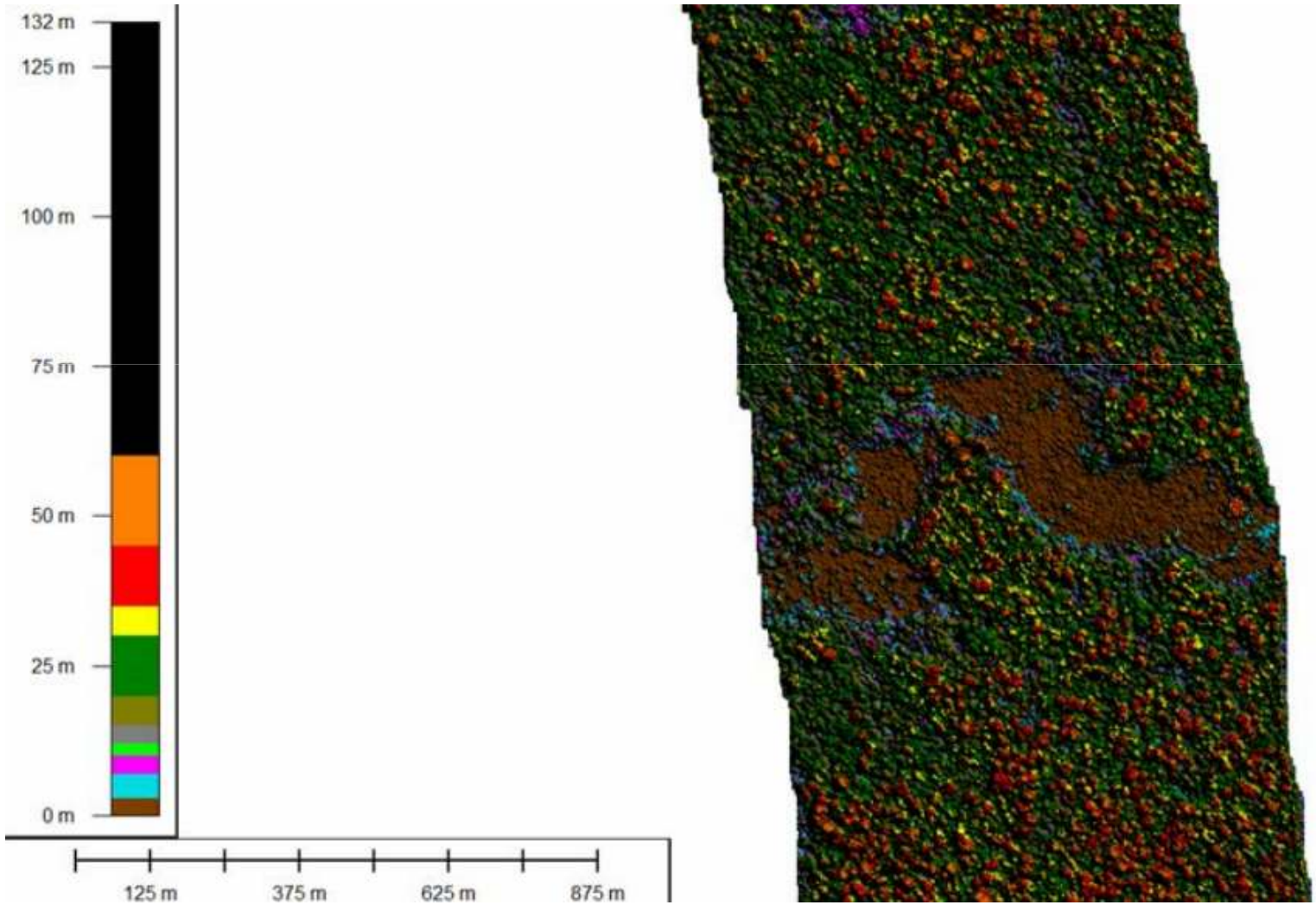
Slope Map



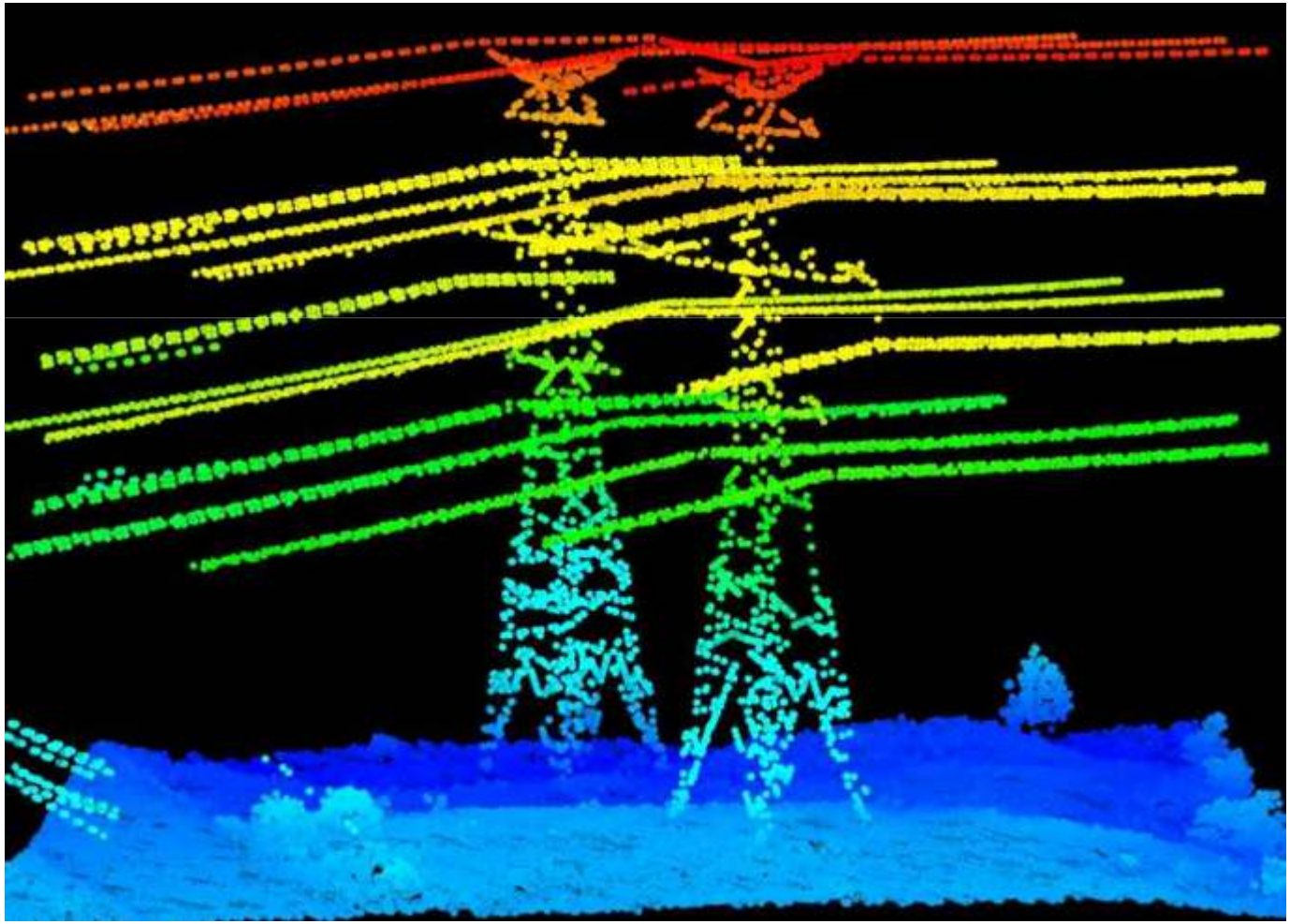
Slope of Surface from River1.shp

- 0.046 - 7.852
- 7.852 - 15.658
- 15.658 - 23.465
- 23.465 - 31.271
- 31.271 - 39.077
- 39.077 - 46.883
- 46.883 - 54.69
- 54.69 - 62.496
- 62.496 - 70.302
- No Data

Canopy Height Map



Energy Transmission Lines



Python

Why Python?

- High-level language, can do a lot with relatively little code
- Supposedly easier to learn than its main competitor, Perl.
- Fairly popular among high-level languages
- Robust support for object-oriented programming
- Support for integration with other languages

For More Information?

http://python.org/

- documentation, tutorials, beginners guide, core distribution, ...

Books include:

- *Learning Python* by Mark Lutz
- *Python Essential Reference* by David Beazley
- *Python Cookbook*, ed. by Martelli, Ravenscroft and Ascher
- (online at http://code.activestate.com/recipes/langs/python/)
- http://wiki.python.org/moin/PythonBooks

4 Major Versions of Python

- "Python" or "CPython" is written in C/C++
  - Version 2.7 came out in mid-2000
  - Version 3.1.2 came out in early 2010

- "Jython" is written in Java for the JVM
- "IronPython" is written in C# for the .Net environment

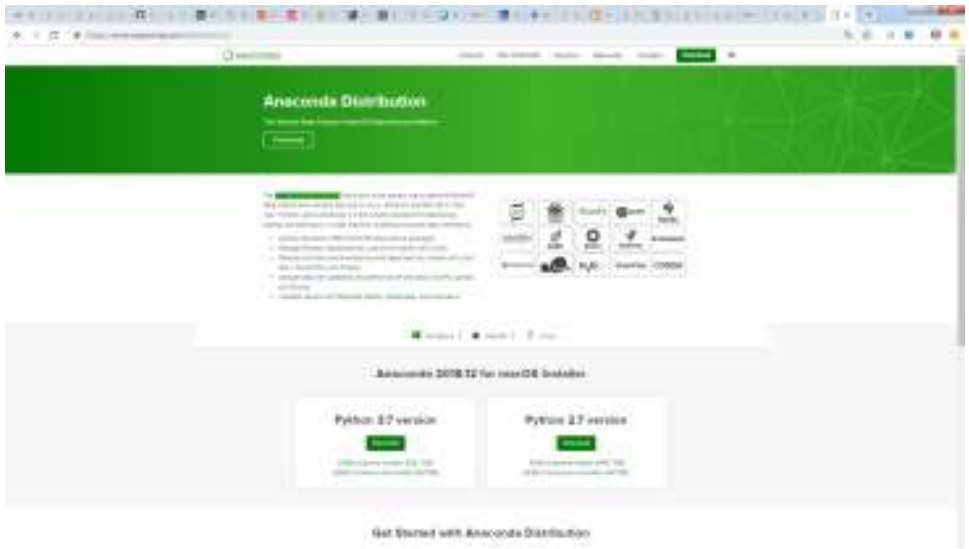
Development Environments

what IDE to use?

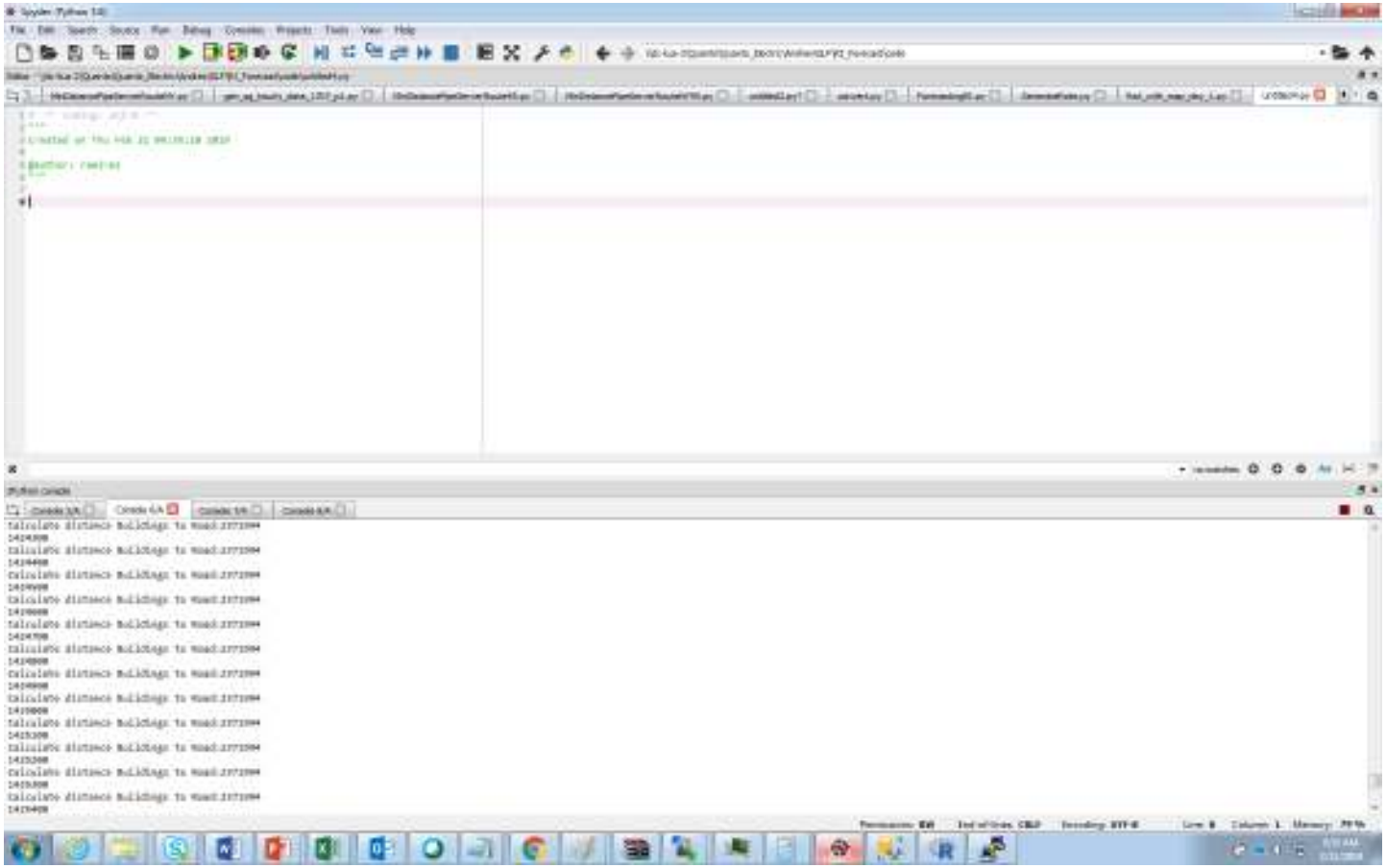
1. Spyder
2. Jupyter Notebook
3. Pycharm (made by the folks at JetBrain)
4. PyDev with Eclipse
5. Emacs
6. Vim
7. TextMate
8. Gedit
9. Idle
10. PIDA (Linux)(VIM Based)
11. NotePad++ (Windows)
12. BlueFish (Linux)

Anaconda

- Spyder as well as others IDEs and python libraries are distributed under the open-source platform: Anaconda
- https://www.anaconda.com/distribution/



Spyder



Python Interactive Shell

```
% python
Python 2.6.1 (r261:67515, Feb 11 2010, 00:51:29)
[GCC 4.2.1 (Apple Inc. build 5646)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

You can type things directly into a running Python session

```
>>> 2+3*4
14
>>> name = "Andrew"
>>> name
'Andrew'
>>> print "Hello", name
Hello Andrew
>>>
```