

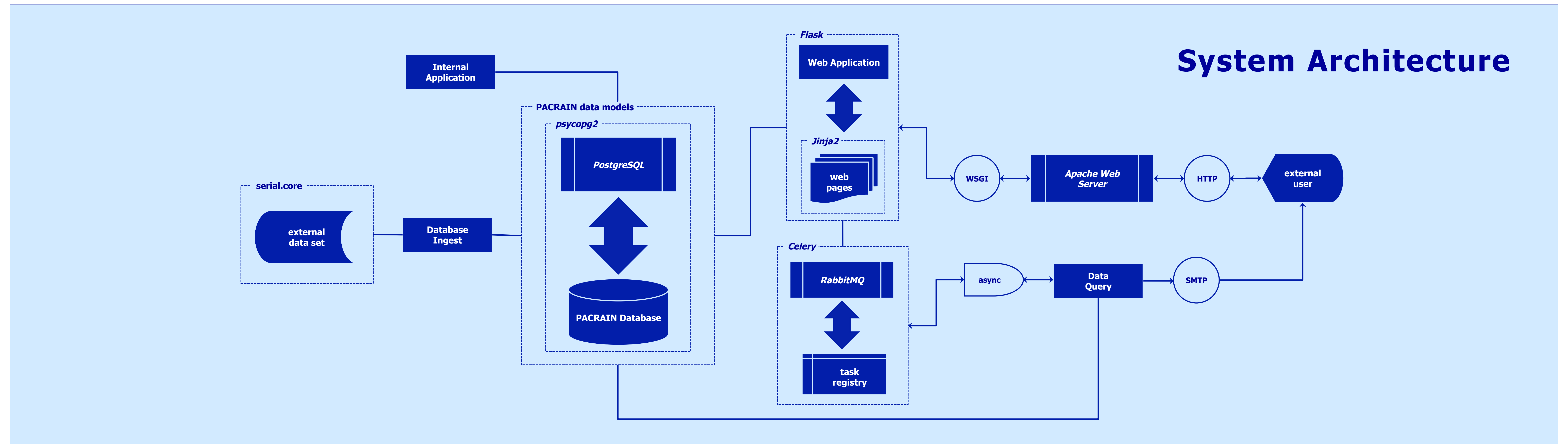
# USING PYTHON AS AN INTEGRATED SOFTWARE PLATFORM FOR THE PACRAIN PROGRAM

Michael D. Klatt<sup>1</sup>, J. S. Greene<sup>2</sup>, M. L. Morrissey<sup>3</sup>

University of Oklahoma · <sup>1</sup>College of Atmospheric & Geographic Sciences, <sup>2</sup>Department of Geography and Environmental Sustainability, <sup>3</sup>School of Meteorology

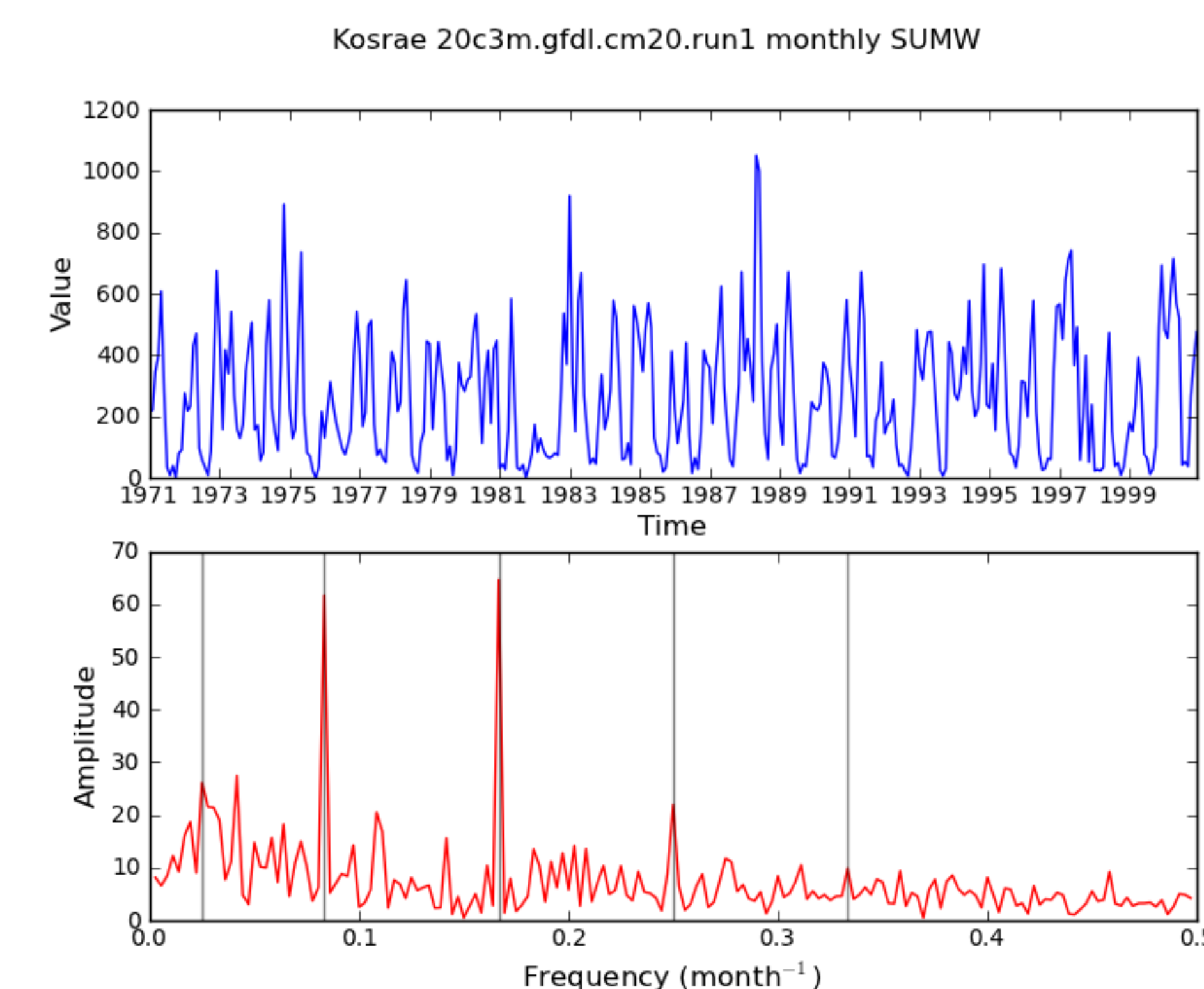
## Pacific Rainfall Program (PACRAIN)

- Data collection, research, and observation networks for the tropical Pacific
- Schools of the Pacific Rainfall Climate Experiment
- Online rainfall database
  - 2.75 million rainfall records
  - ~1000 sites
  - data begins in 1874
- [pacrain.evac.ou.edu](http://pacrain.evac.ou.edu)



## Research Applications

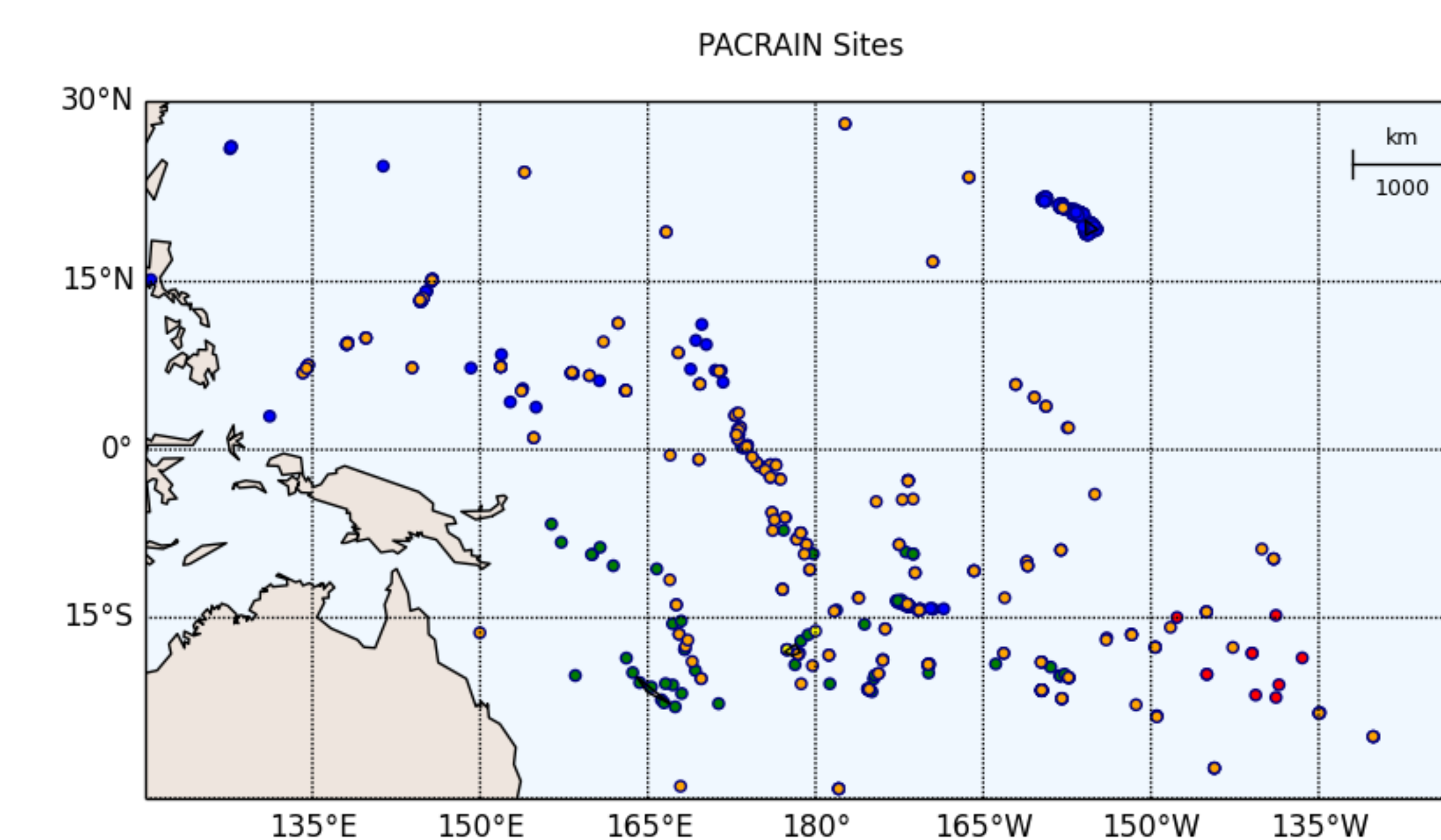
- NumPy* for high-performance data arrays
- SciPy* for scientific computing
- matplotlib* for visualizing data



20<sup>th</sup> Century simulation of monthly rainfall (top) at Kosrae, FSM produced by the GFDL CM 2.0 model for the CMIP3 archive. The Fourier analysis (bottom) shows that the time series is dominated by interannual variability (0.0833 month<sup>-1</sup>) and a wet season/dry season pattern (0.1667 month<sup>-1</sup>).

## GIS Applications

- Fiona* for reading/writing GIS data formats
- pyproj* for geospatial transformations
- Shapely* for geometric analysis
- Basemap* for creating maps



Map of sites in the PACRAIN database, color-coded by data source: NCDC (blue), NIWA (green), Meteo-France (red), Fiji Meteorological Service (yellow), R. C. Taylor's An Atlas of Pacific Islands Rainfall (orange). Many sites have data from multiple sources.

## Web Applications

- Flask* for implementing WSGI applications
- Jinja2* for generating dynamic HTML documents
- WTForms* for handling HTML form data
- Celery* for asynchronous processing

```

1 {% extends "layout.jinja.html" %}
2 {% block title %}PACRAIN - Pacific Rainfall Program{% endblock %}
3 {% block author %}Michael Klatt{% endblock %}
4 {% block copyright %}2014{% endblock %}
5 {% block modtime %}2014-03-17{% endblock %}
6
7 {% block sidebar %}
8 <div class="first">Site Search</div>
9 <form action="http://www.google.com/search" method="get">
10 <ul>
11 <li><input name="q" type="text" maxlength="255"></li>
12 <li><input name="sitesearch" type="hidden" value="pacrain.ou.edu"></li>
13 <li><input name="submit" type="submit" value="Google"></li>
14 </ul>
15 </form>
16
17 <div>Announcements</div>
18 <div>
19 <div>2014-04-01</div><div>Next scheduled database update</div>
20 <div>2014-03-03</div><div>Latest database update</div>
21 </div>
22 {% endblock sidebar %}
    
```

Sample of a Jinja2 template for generating dynamic HTML. Jinja2 is the default template engine used by the Flask web application framework. Inheritance can be used to generate multiple web pages from a common layout.

## Why Python?

- Easy to learn, yet powerful
- Encourages readability and maintainability
- "Batteries included" in standard library
- Scientific programming is well-supported
- Portable across all major platforms
- Open-source and freely available
- [www.python.org](http://www.python.org)

