



Abstract

We describe a simple implementation of a RESTful web service for astronomical datasets. The service is implemented in **web.py**, a minimalist Python web framework that uses the model view controller (MVC) architectural pattern. The services use clean and simple URIs to access methods within Python classes. We show an example implementation that interfaces with a molecular line catalog.

What is web.py?

web.py is a minimalist framework for developing web apps in Python. It comes complete with a mini web server for testing, and separates the Python app code from the view templates.

<http://webpy.org/>



MySQL or SQLite

Model

Python Class

View

XML templates

Web Server

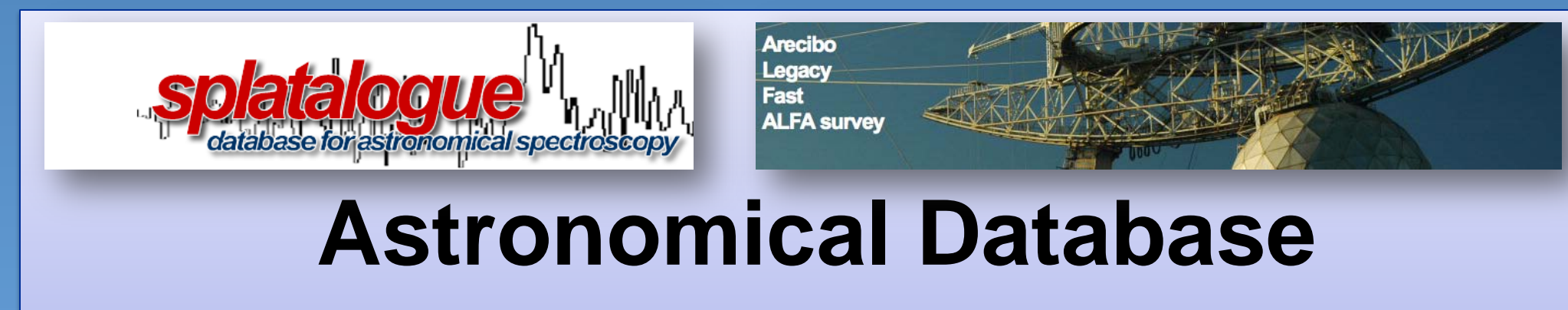
Controller

Class Methods

How about an example?

Sure! A MySQL database for Splatalogue is used as the data source. However, the framework lends itself very well to development with many astronomical databases. A few simple Python classes with methods for REST calls can be used. View templates with minimal code serve as the interface to the web server.

The web.py framework easily allows a REST style (Representational State Transfer) service to be written. We hope to experiment with other astronomical datasets and metadata, as this framework allows for rapid prototyping.



Astronomical Database

API Search by:

Wavelength Frequency Chemical Name Object Identifier Position Redshift

RESTful GET:

HTML

VOTable XML

Formatted Text

Does a user need to know Python to retrieve a result?

Not at all! URIs for web.py application are similar to what you would see for a Django app and are cleaner than typical PHP or Java services.

For example:

<http://slap.net/freq/vot/155.4/155.5>

[http://slap.net/name/html/Carbon Monoxide](http://slap.net/name/html/Carbon%20Monoxide)

[http://slap.net/objectid/csv/NGC 4254](http://slap.net/objectid/csv/NGC%204254)

Can documentation and a tutorial for new users be created?

Yes! Sphinx is a Python application for creating elegant documentation with inline examples and tutorials – advantageous for astronomers who might be new to a particular computing or web service concept.



Learning more...

I'm happy to show simple demos of the code to meeting participants.

Acknowledgements

Thanks to Tony Remijan for providing a Splatalogue copy and Joe Masters for an introduction to the Sphinx Documentation system.

