A ReSTful Interface for CubeSats

Shaun Houlihan
Pumpkin, Inc.
CubeSat Software Interfaces

- Many mission unique protocols
- Low level manipulation
- Protocol libraries
- Platform lock
- Interface documents
- Etc...
Software Interfaces Ideally..

• Universal
• Language independent
• Platform independent
• High Level
• Able to leverage open source code
• Simple to document

• Where would we find something like that?
ReST - “The software architectural style of the Web”¹

- **Representational State Transfer**
- **ReSTful system constraints:**
  1. Client-Server
  2. Stateless
  3. Cache
  4. Uniform Interface
  5. Layered
  6. Code on Demand

- **Protocol + Data Format not explicitly stated**
  - But **HTTP + JSON** are web standard
What does using ReST look like?

• Browser demo

```json
{
  "eci_dx_kms": 2.4453357689384707,
  "eci_dy_kms": -5.236508654841261,
  "eci_dz_kms": -5.032283151039424,
  "eci_x_km": -6129.230374071741,
  "eci_y_km": -69.38074892405864,
  "eci_z_km": -2911.6373186760843,
  "timestamp": "2016-04-21T22:16:54.104683"
}
```
import httplib, json

conn = httplib.HTTPConnection("192.168.1.101")
conn.request("GET", "/datasets/orbital_state")
response = conn.getresponse()  # --- Requesting

if response.status == 200:
    data = json.loads(response.read())
    print data['eci_x_km']  # --- Error Checking
                            # --- Parsing
                            # --- Printing
Payload Example - JavaScript

```
var request = new XMLHttpRequest();
request.open("GET", "http://192.168.1.101/datasets/orbital_state", true);
request.send(null);       // Requesting
if (request.status == 200) {
    data = JSON.parse(conn.responseText);  // Parsing
    document.write(dataeci_x_km);          // Printing
}
```

ReST/HTTP Extensibility

- Augment with ground & web systems
- Third-party ‘Apps’ & payloads
ReST Implementation - Hardware

• SUPERNova Bus
  ▪ 64W Solar Power
  ▪ Ethernet Switch (in development)
  ▪ Up to 4U Payload Volume

• BeagleboneBlack Flight Computer
  ▪ ~2W
  ▪ 1GHz ARM CPU
  ▪ Linux
ReST Implementation - Software

• Augments high-TRL bus flight software
  ▪ Low-level interface still available

• “API First”
  ▪ Describe interface with using “Swagger.io” standard & tools
    ◆ http://www.swagger.io
  ▪ Auto-generate documentation from spec
  ▪ Auto-generate HTTP server stubs from spec
  ▪ Fill in the callbacks
Swagger Document Generation
Payload Demo...

```python
#!/usr/bin/python
import httpplib, json, time, math
connection = httpplib.HTTPConnection("192.168.2.101", port=9090, timeout=2)
while True:
    try:
        connection.connect()
        connection.request("GET", "/supernova/orbital_state")
        response = connection.getresponse()
        if response.status == 200:
            state = json.loads(response.read())
            position = (state["eci_x_km"], state["eci_y_km"], state["eci_z_km"])
            distance = math.sqrt(sum([math.pow(x, 2) for x in position]))
            earth_rad = 6371
            print "Position at " + state["timestamp"] + "\n" + position
            print "Hello from " + str(distance - earth_rad) + " km!\n"
    except:
        print "No Connection"
    time.sleep(4)
```
Conclusions + Future

• **ReST fits Pumpkin approach to Nanosats**
  - Open, Standardized, Accessible, Scalable, Modular, Customizable
  - “Fly Your Laptop”

• **Development at Pumpkin:**
  - HTTP payload interface to SUPERNOVA
  - SUPERNOVA Bus Simulator
Q&A Session

Thank you for attending this Pumpkin presentation at the 2016 CubeSat Developer’s Workshop!
Appendix

• **Further Reading**
  - http://whatisrest.com
  - http://coap.technology
  - http://cbor.io
  - http://swagger.io

• **Speaker information**
  - Shaun Houlihan is an Engineer at Pumpkin involved in developing SUPERNOVA hardware and software. Before joining Pumpkin he worked in the aerospace and consumer electronics industries. Contact Shaun at shaun@pumpkininc.com.

• **CubeSat Kit information**
  - More information on Pumpkin’s CubeSat Kit can be found at http://www.cubesatkit.com/. Patented and Patents pending.

• **Copyright**
  © 2000-2016 Pumpkin, Inc. All rights reserved. Pumpkin and the Pumpkin logo, Salvo and the Salvo logo, The RTOS that runs in tiny places, CubeSat Kit and the CubeSat Kit logo, CubeSat Kit Bus, nanoLab Kit and the nanoLab Kit logo, MISC and SUPERNOVA are all trademarks of Pumpkin, Inc. Don’t leave Earth without it is a service mark of Pumpkin, Inc. All other trademarks and logos are the property of their respective owners. No endorsements of or by third parties listed are implied. All specifications subject to change without notice. Unless stated otherwise, all photographs, images and illustrations are the property of Pumpkin, Inc. and may not be used without permission.
Notice

This presentation is available online at:

First presented at the CubeSat Developer's Workshop in San Luis Obispo, California on Friday, April 22, 2016