



Extensible SSEL Beacon Parser for HAM Operators

Andrew Seel
Summer 2013 REU Project
August 2, 2013

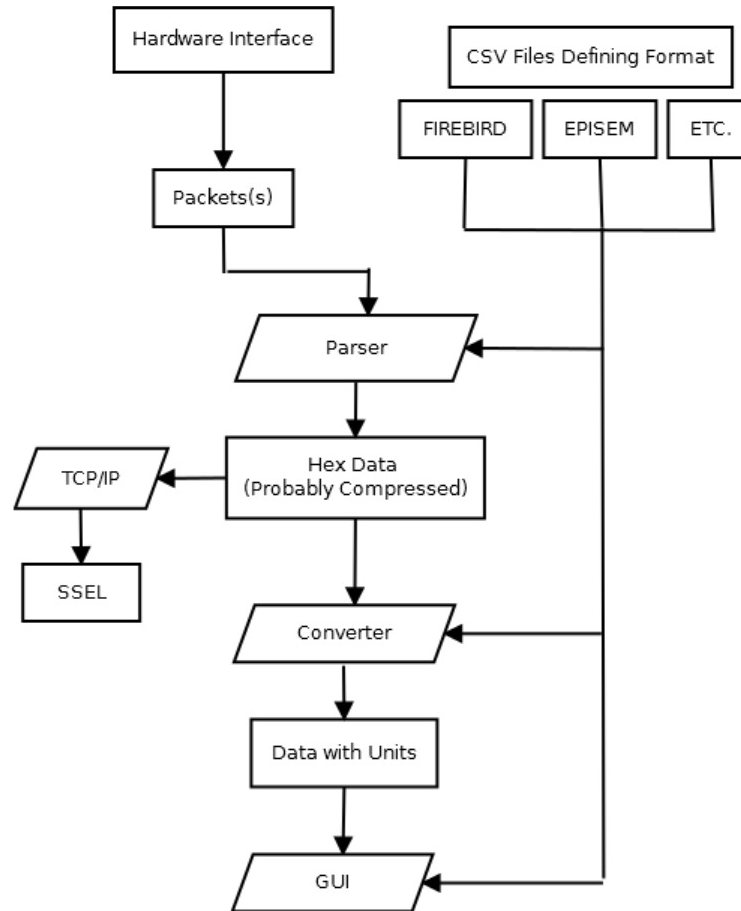


Requirements

- Monitor serial stream
- Identify, collect, and validate SSEL packets
- Send validated SSEL packets to SSEL
- Select mission of interest (e.g. FIREBIRD, EPISIM)
- Display and/or plot variables of user interest
- Distributable as a single executable (Not Open Source)



Original Flowchart





Programming Language

- Python 2.7
- Tkinter
 - Cross-platform GUI toolkit
- Matplotlib and Numpy
 - Graphing
- PySerial
 - Serial Communications
- PyInstaller
 - 'Compiles' python files into a single standalone file



Current Progress

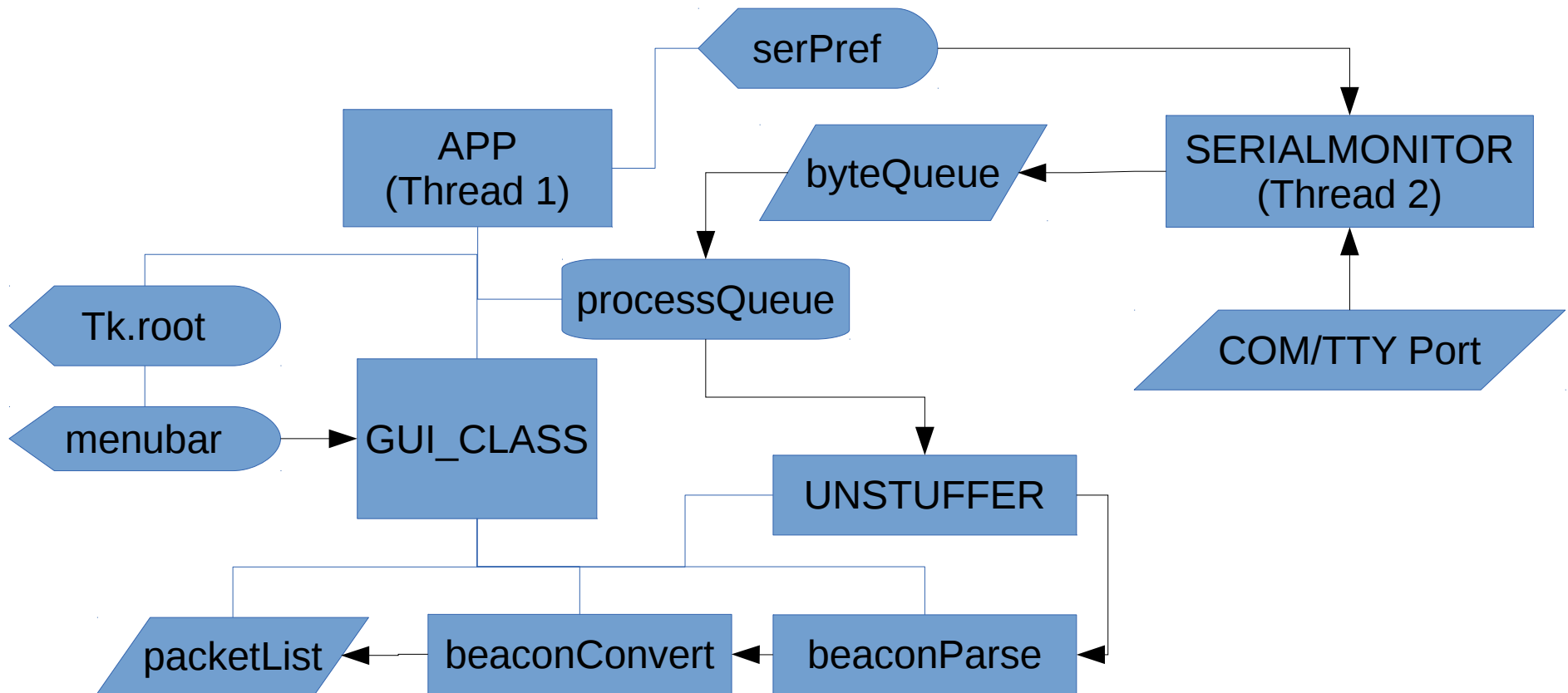
- Barebones GUI
- Serial Port selection
 - Attempts to autodetect ports on the system
 - Allows for user to type in port name if correct port isn't detected
- Detects FIREBIRD packets
 - Unstuffs, Checks CRC, Checks spacecraft ID fields
- Parses FIREBIRD packets into Python Dictionaries
 - {"BEACON_SYSTEM_TIME_YEAR": 2013, "BEACON_SYSTEM_TIME_MONTH": 8, etc}
- Graphs user selected variable from a series of packets
 - Updates when a new packet is received or variable selection changes



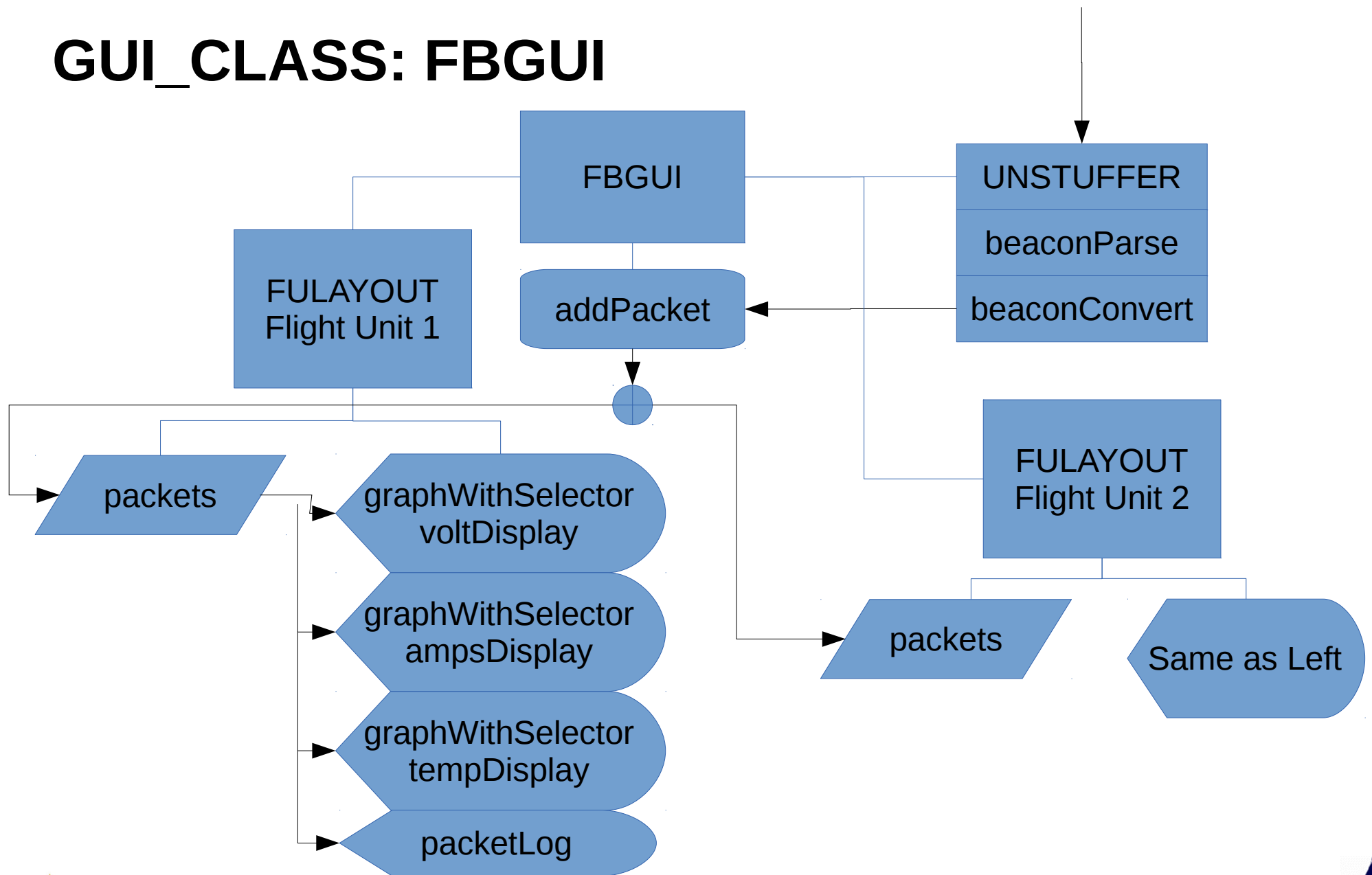
Current Classes

- APP
 - FBGUI
 - FULAYOUT
 - graphWithSelector
 - UNSTUFFER
 - UN_ENUMS
 - beaconParse
 - beaconConvert
- SERIALMONITOR

Core App Flowchart: Current Implementation

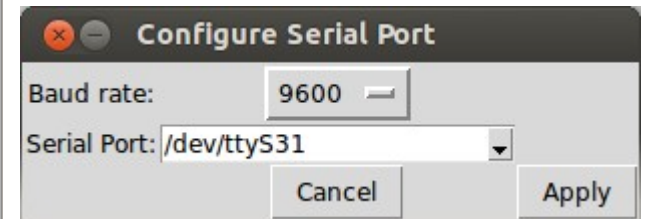
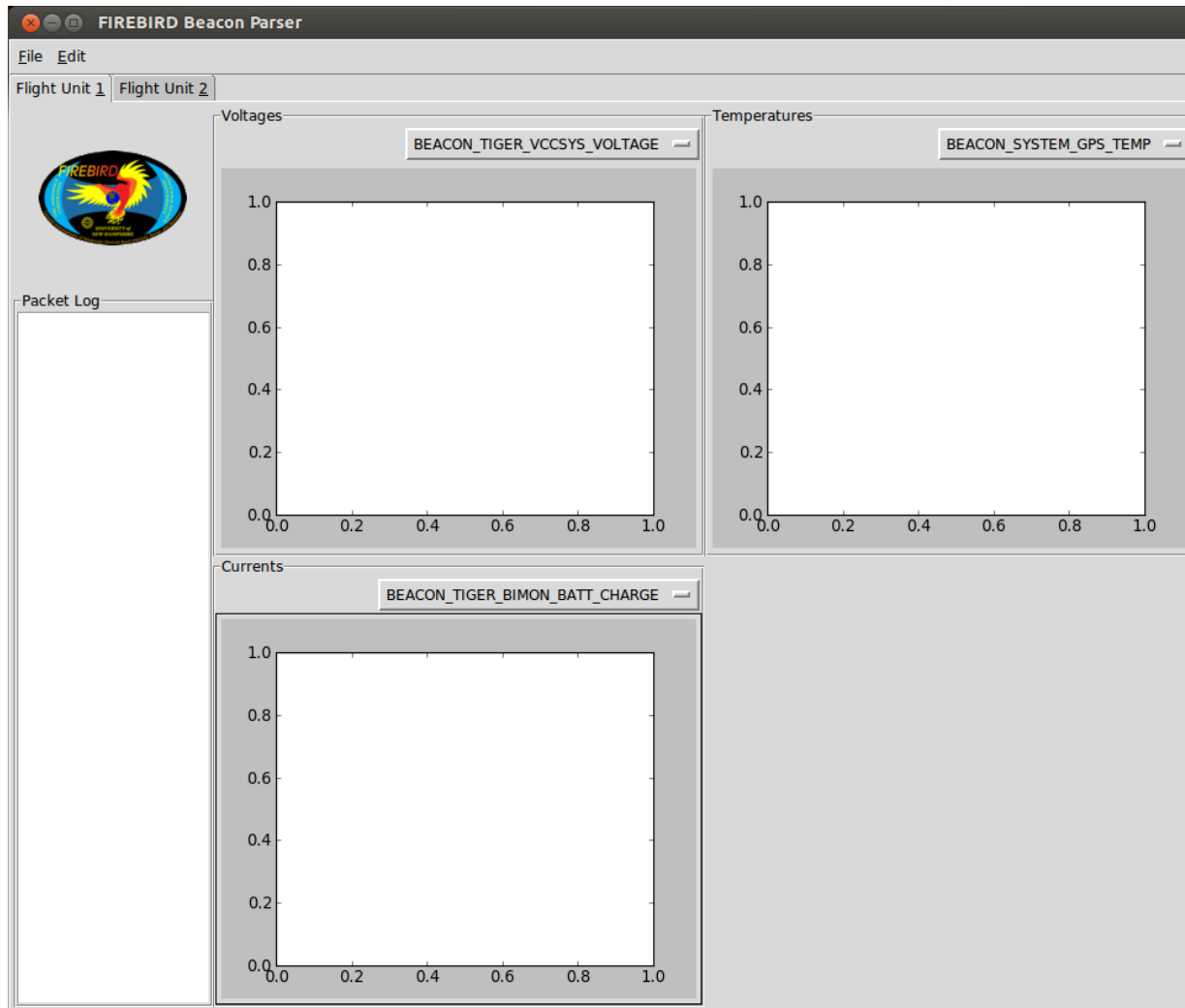


GUI_CLASS: FBGUI



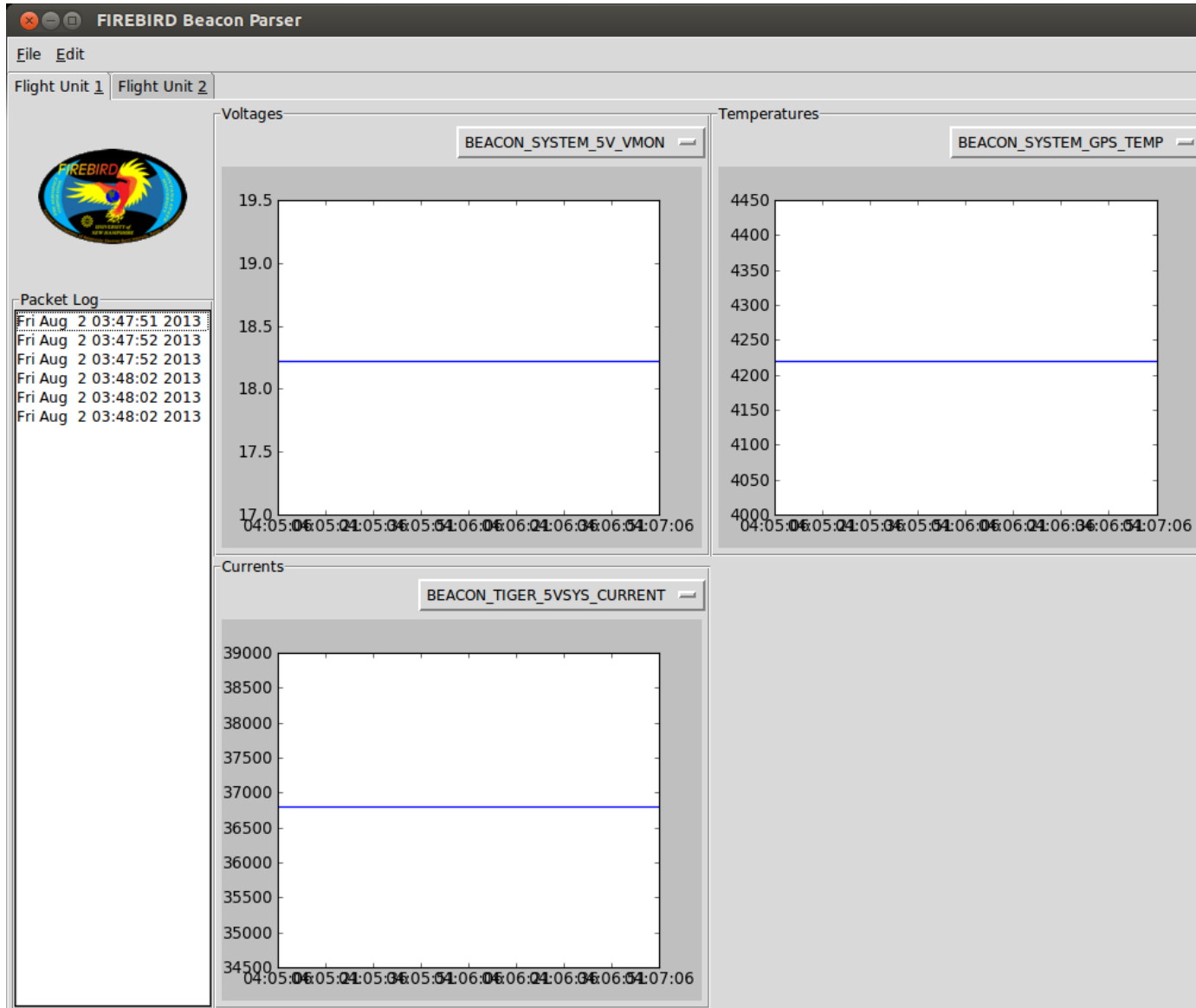


Screenshots





Screenshot: with fake data





To Do

- Verify serial data parsing with data from the Firebird engineering unit
- List the remaining variables in the bottom right quadrant
 - Use values from either first packet or packet currently highlighted in the packetLog
- Export packets to SSEL
- Write GUI_CLASSES for other missions
- General GUI cleanup