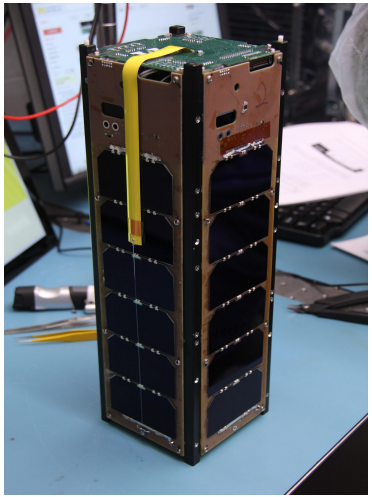
Applying Modern Software System Design to Small Satellite Development and Operations

Michael Wilson
Michigan eXploration Laboratory - University of Michigan

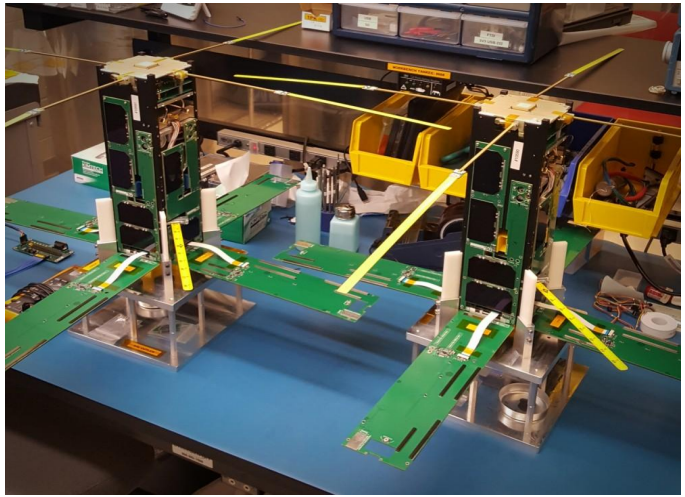
34th Annual Small Satellite Conference

Michigan eXploration Laboratory

Fall 2018: Our previous operations system was lost due to backup failures, but we were in the midst of developing and operating 3 flight spacecraft and 3 engineering development units.



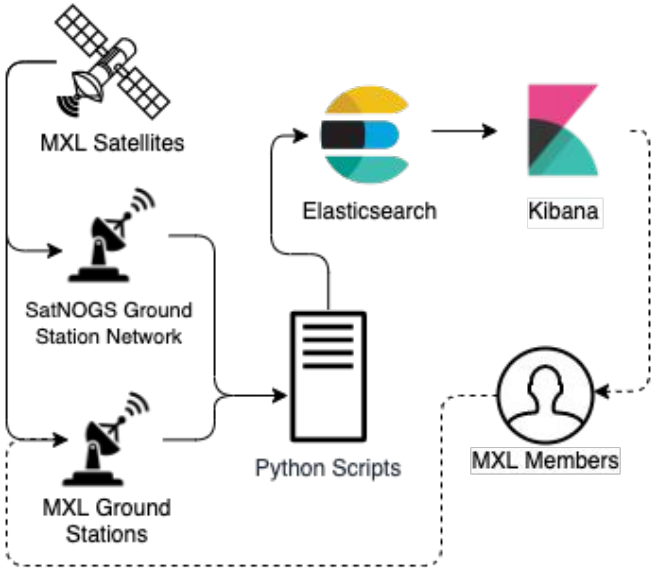
GRIFEX

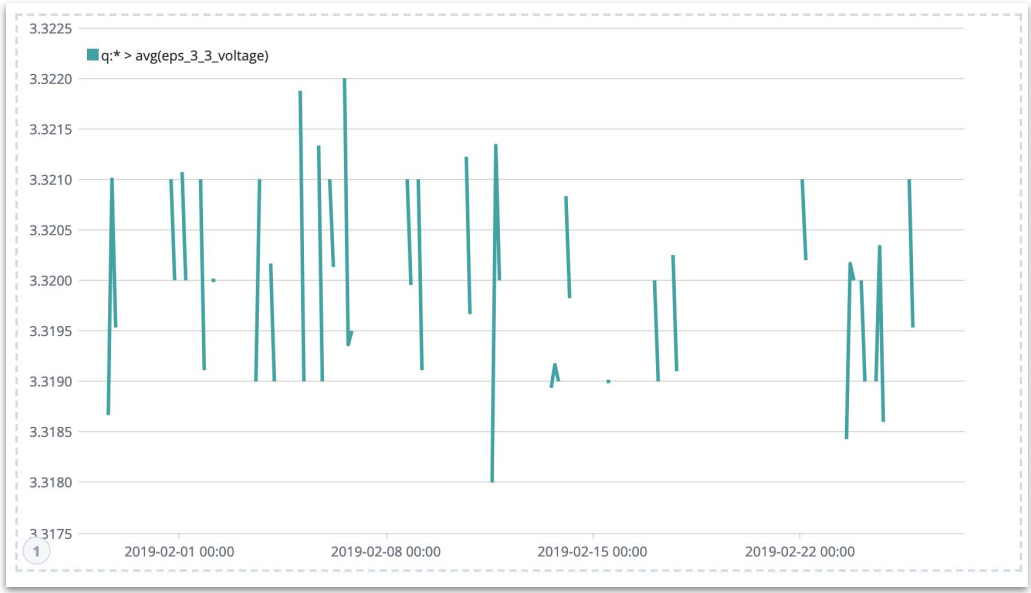


E-TBEx

MC 9

Initial work - Python scripts create simple graphs





Python scripts pulled data from the SatNOGS API and our ground station logs and dumped packets into Elasticsearch, where they could be viewed using a Kibana dashboard.

System Goals

An effective and lasting solution should enable the following:

- Reliable, thorough testing
- Rapid, stable, feature development
- Knowledge transfer between students
- Task automation
- Operational visibility

These goals are achieved using *modern software system design*.

Modern Software System Design

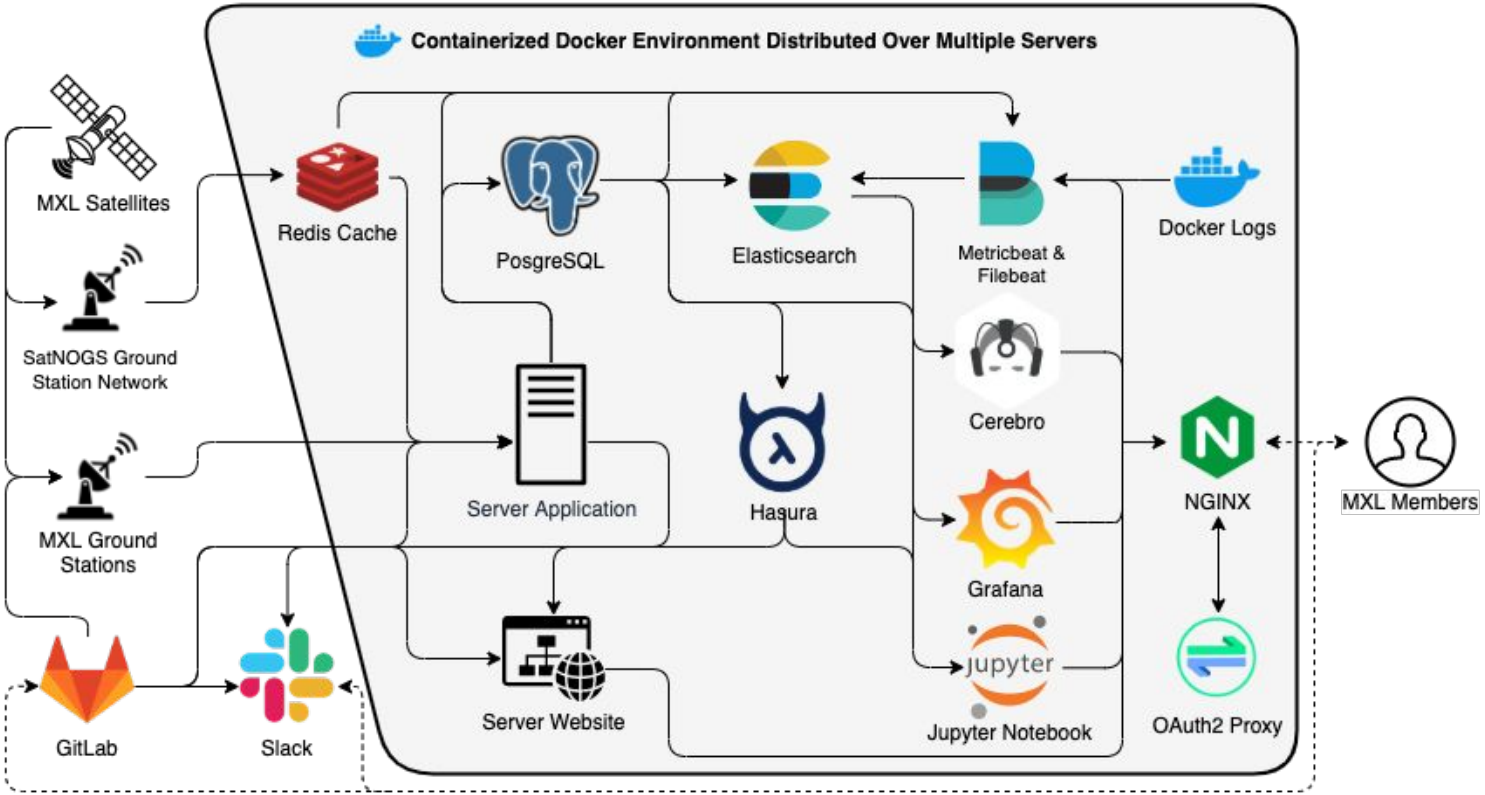
3 Core Principles

- Small - applications are broken down into granular, easily tested, components
- Networked - applications can communicate in a distributed manner and are therefore more resilient in the face of failures and easier to manage at the service level
- Developer Oriented - applications follow simple architectures to which engineers can easily contribute and deliver changes

The application of these principles toward a distributed satellite operations and development system resulted in the creation of the MXL Integrated Data Analysis System (MIDAS).

https://www.nginx.com/blog/principles-of-modern-application-development/

MIDAS Architecture



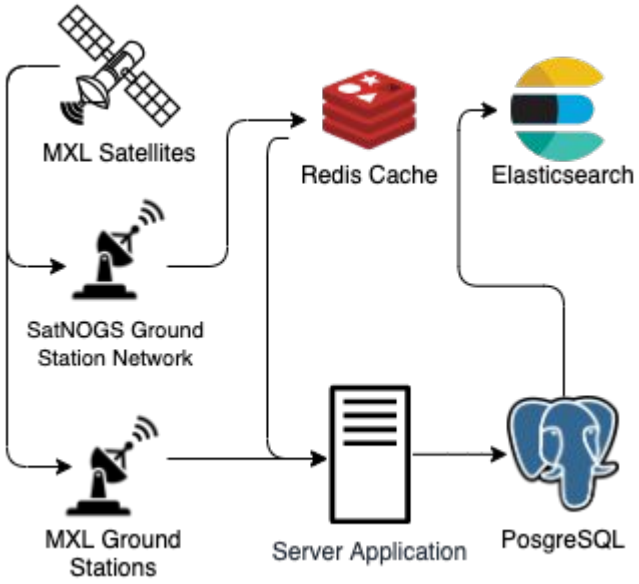
Areas of Improvement

- Telemetry aggregation and analysis
- Ground systems and satellite communications
- Systems monitoring and development
- Team communications and collaboration

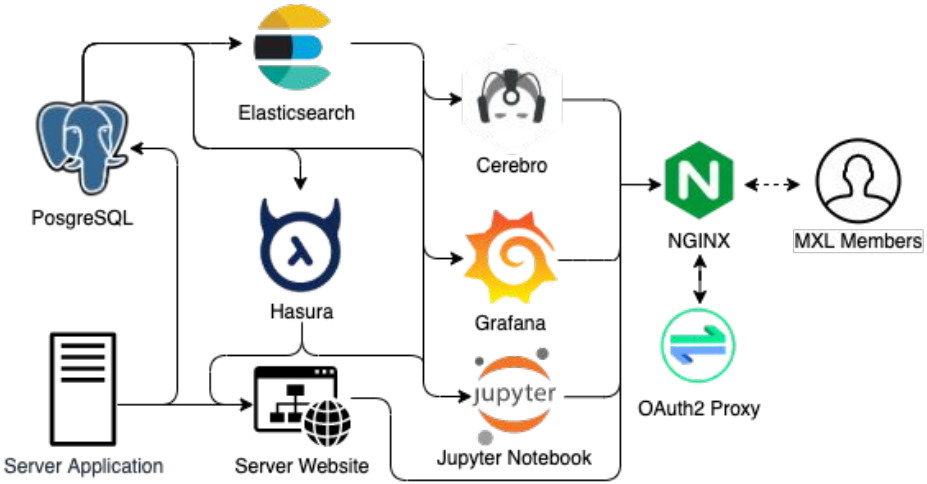
Telemetry aggregation and analysis - Architecture

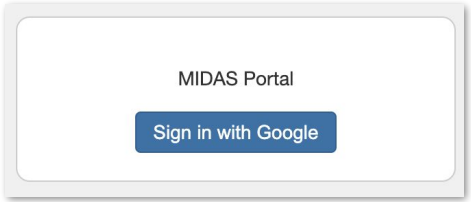
The server aggregates telemetry data from the SatNOGS API and our own ground stations.

MIDAS *extracts* data from these sources, *transforms* it, and *loads* it into the PostgreSQL and Elasticsearch data stores (ETL Pipeline).



Telemetry aggregation and analysis - Architecture

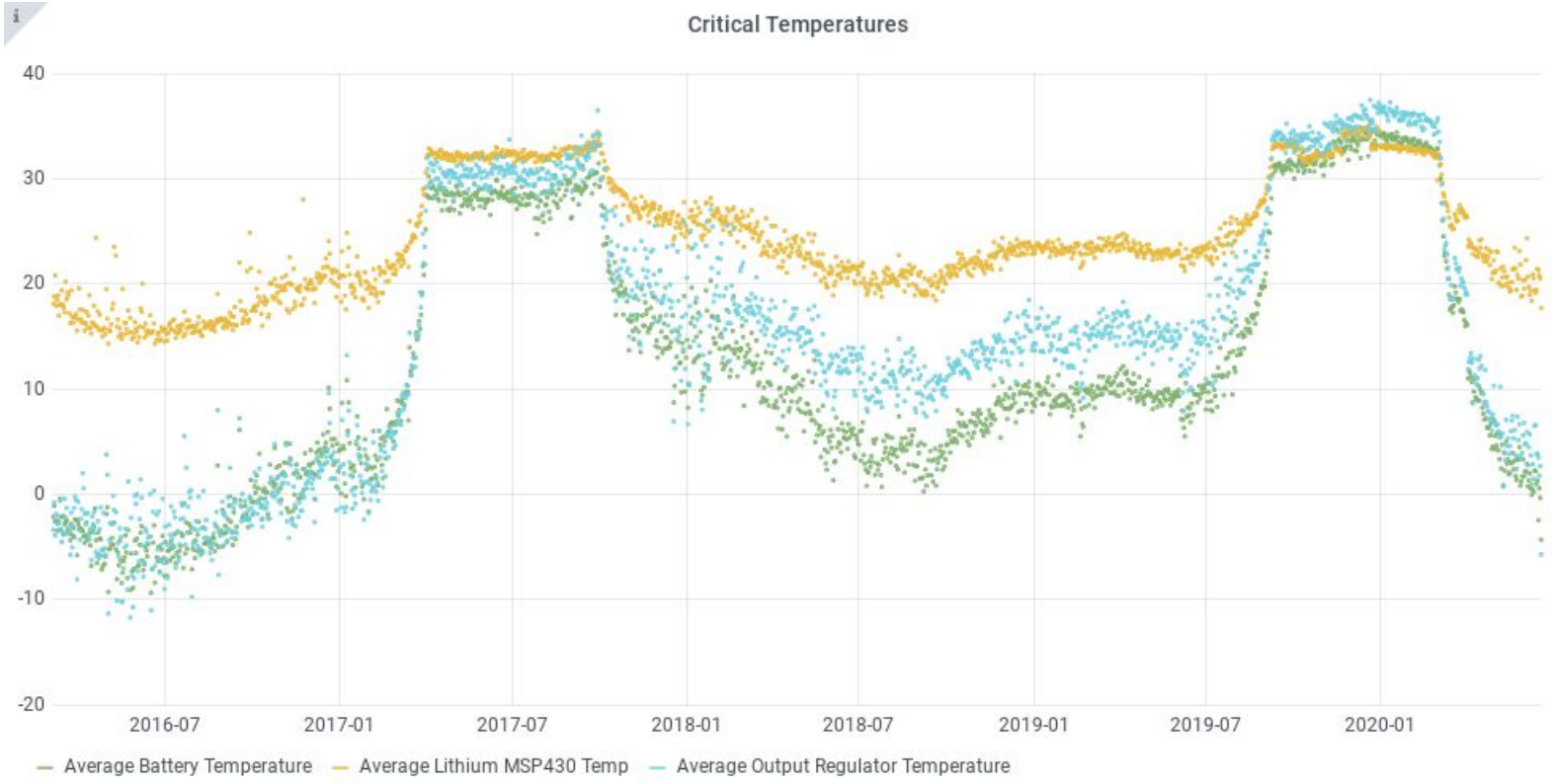




MIDAS services including Grafana, Hasura, Jupyter Notebook, Cerebro, and our own custom dashboards query and analyze data from PostgreSQL and Elasticsearch.

User requests are authenticated using University of Michigan Google accounts and proxied to their appropriate services, e.g. https://grafana.mxl-ops.engin.umich.edu/

Telemetry aggregation and analysis - Grafana



Telemetry aggregation and analysis - GraphQL

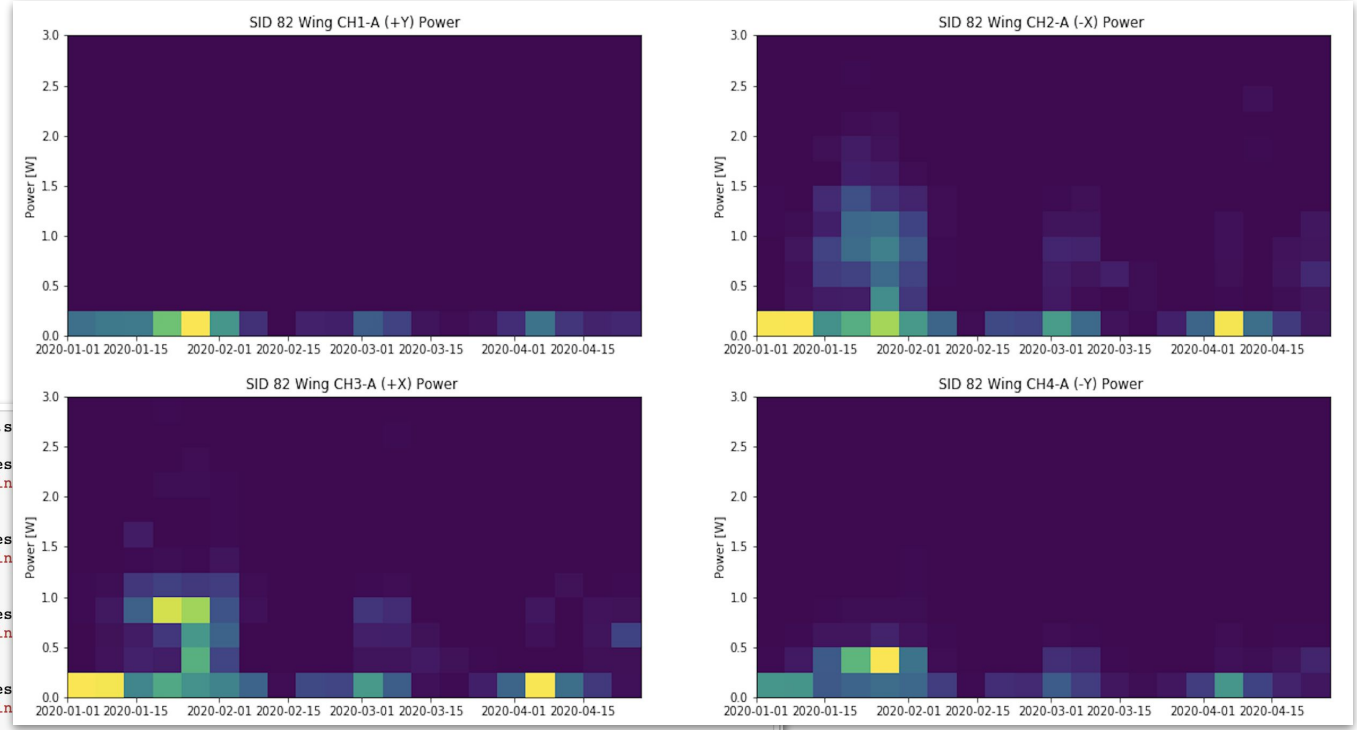
```
query MyQuery {
  mxl_pass(
    where: {
      satnogs_id: {_is_null: false},
      packets: {packet: {flag: {_eq: "1"}}}
    },
    order_by: {aos: desc},
    limit: 5) {
    aos
    ground_station { name }
    craft { name }
    packets_aggregate {
      aggregate {
        count
      }
    }
  }
}
```

MIDAS's Hasura service provides users with a GraphQL endpoint that enables intuitive queries on relational and graph data.


```
[{
   "aos":"2020-06-16T17:05:52+00:00",
   "ground_station":{
      "name":"CA,USA-Fixed UHF"
   },
   "craft":{
      "name":"GRIFEX"
   },
   "packets_aggregate":{
      "aggregate":{
         "count":99
      }
   }
}]
```

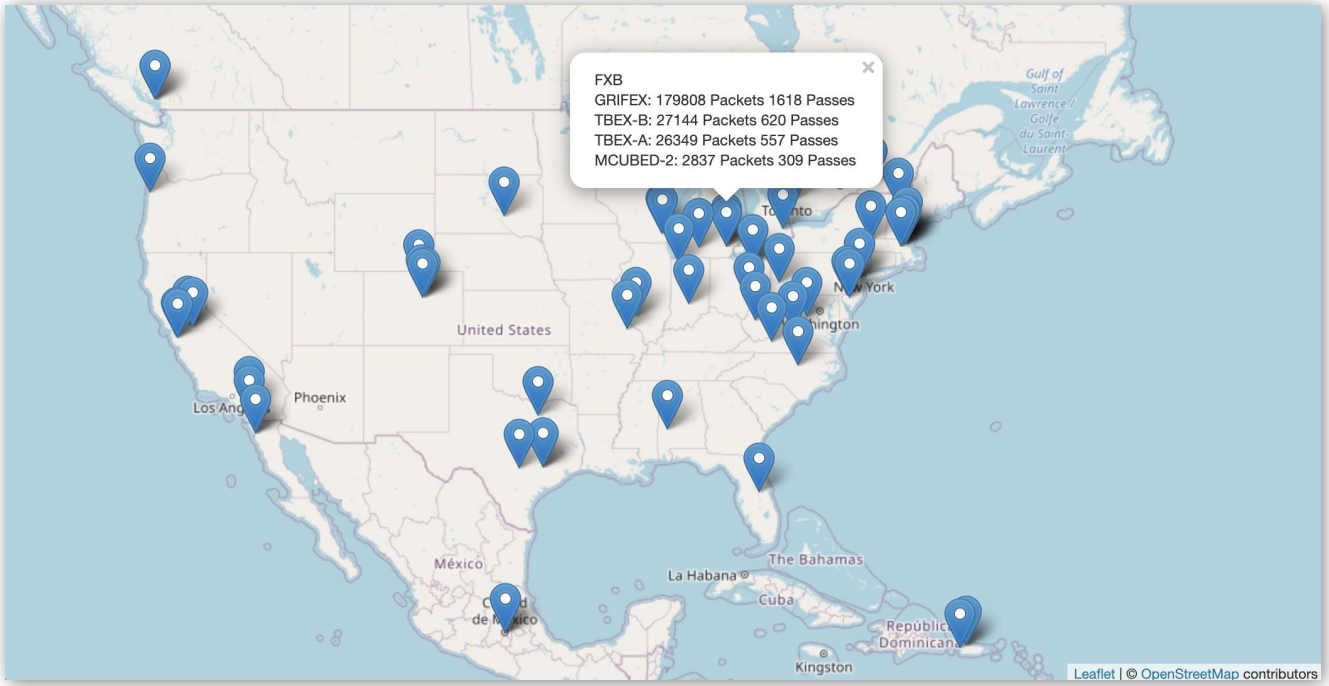
Telemetry aggregation and analysis - Jupyter

To perform extended analysis, users load data into Jupyter Notebook, where they can manipulate it using Python.



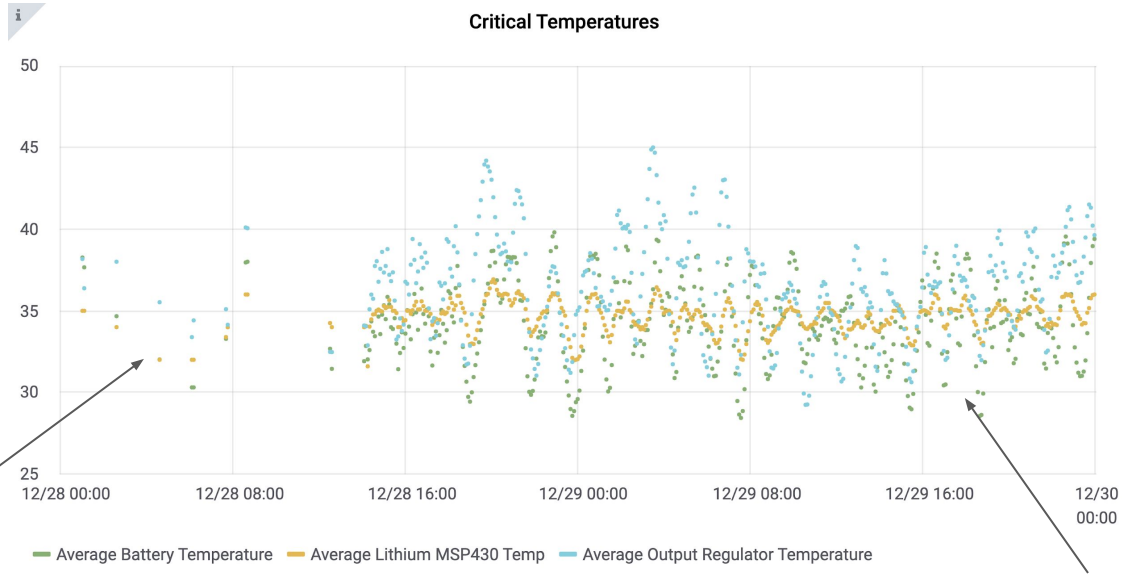

```
In [6]:
fig, [[ax1,ax2],[ax3,ax4]] = plt.s

ax1.pcolor(xedges1_datetime,yedges
ax1.set_title('SID ' + sid + ' Win
ax1.set_ylabel('Power [W]')

ax2.pcolor(xedges2_datetime,yedges
ax2.set_title('SID ' + sid + ' Win
ax2.set_ylabel('Power [W]')

ax3.pcolor(xedges3_datetime,yedges
ax3.set_title('SID ' + sid + ' Win
ax3.set_ylabel('Power [W]')

ax4.pcolor(xedges4_datetime,yedges
ax4.set_title('SID ' + sid + ' Win
ax4.set_ylabel('Power [W]')

plt.savefig(sid+ ch + ' Wing Panel Power 2D Histogram_'+ startString + '_' + endString + '.png')
```

Ground systems and satellite communications



MIDAS integration between MXL ground stations and SatNOGS gives us uniform data across 100s of locations around the globe.

Ground systems and satellite communications

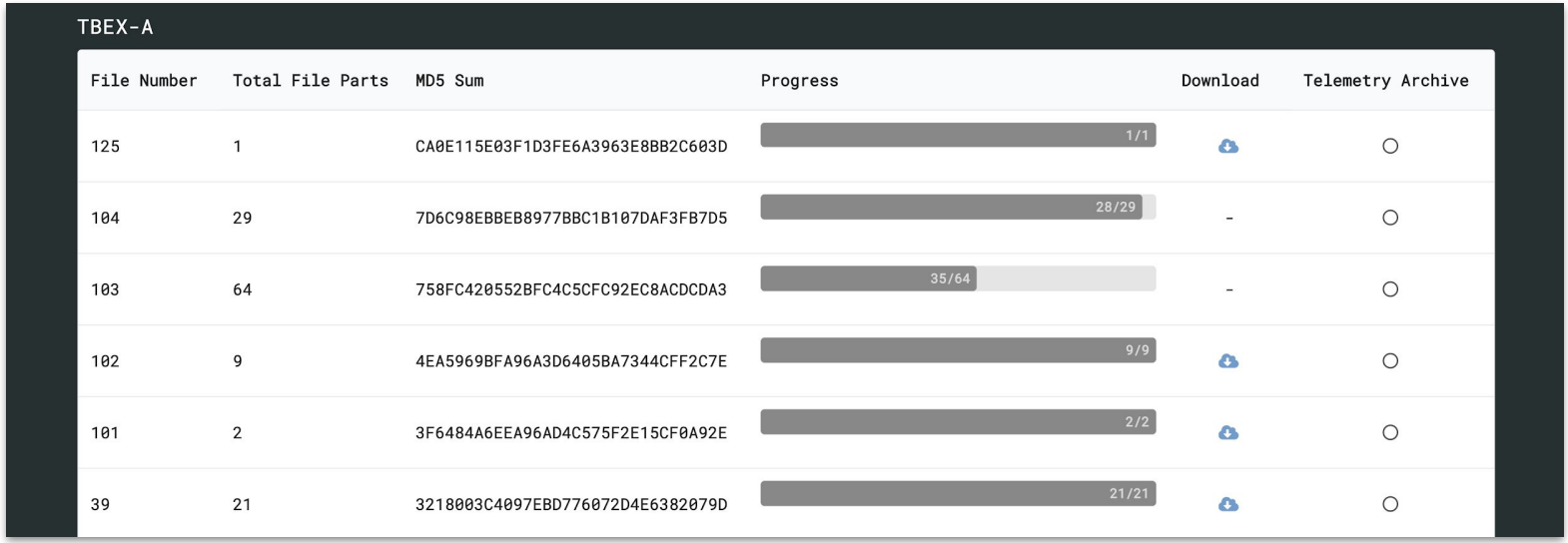


Several Dozen Beacons Every Day

10 Second Resolution Beacon Telemetry

Downlinked telemetry archives provide a **125x** increase in data resolution.

Ground systems and satellite communications

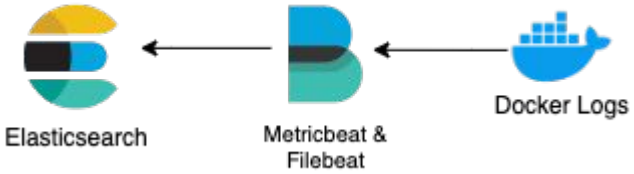


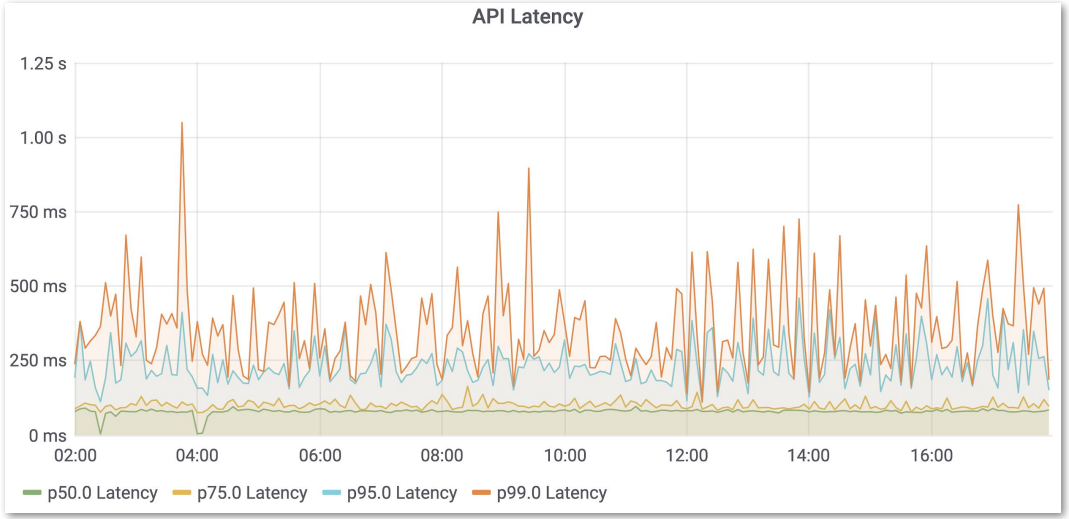
Custom data displays show file download progress and an MD5 sum for checking data integrity.

Systems monitoring and development - Grafana

Logs and metrics from all MIDAS Docker containers are pushed into Elasticsearch, enabling high operational visibility.

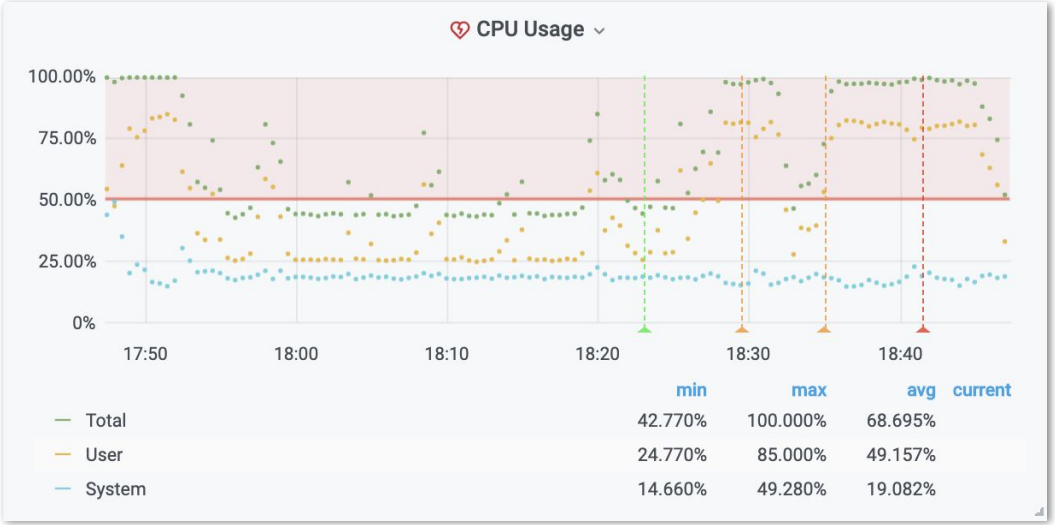
Just as we have telemetry and operations for our spacecraft, we now have them for our ground systems as well.

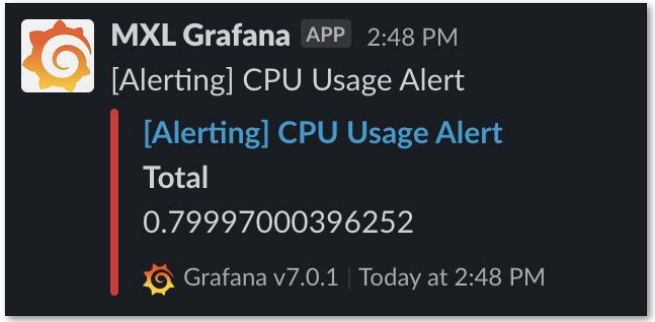




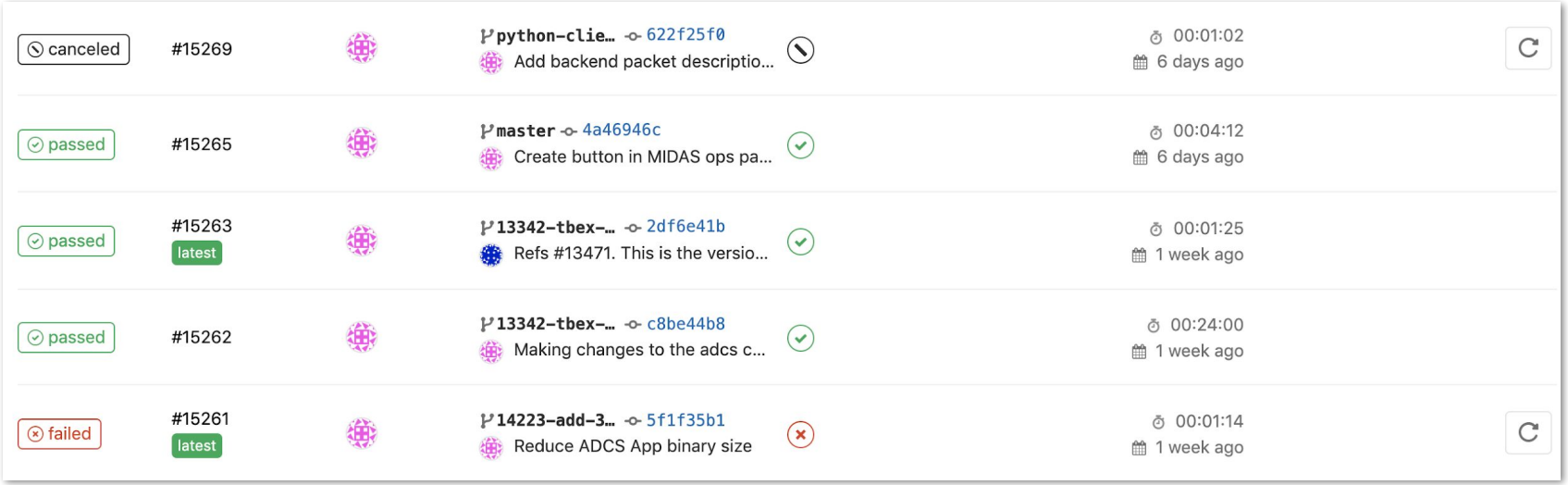
Systems monitoring and development - Grafana

Monitoring metrics with such tools as Grafana allows the creation of alert systems that notify our team via Slack in the event of system errors, whether on the ground or in orbit.



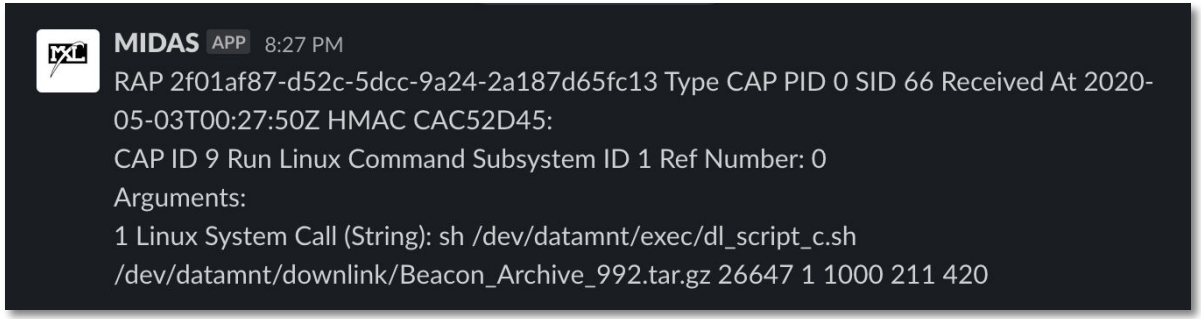


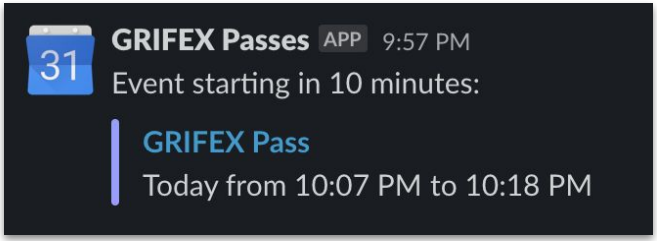
Systems monitoring and development - GitLab



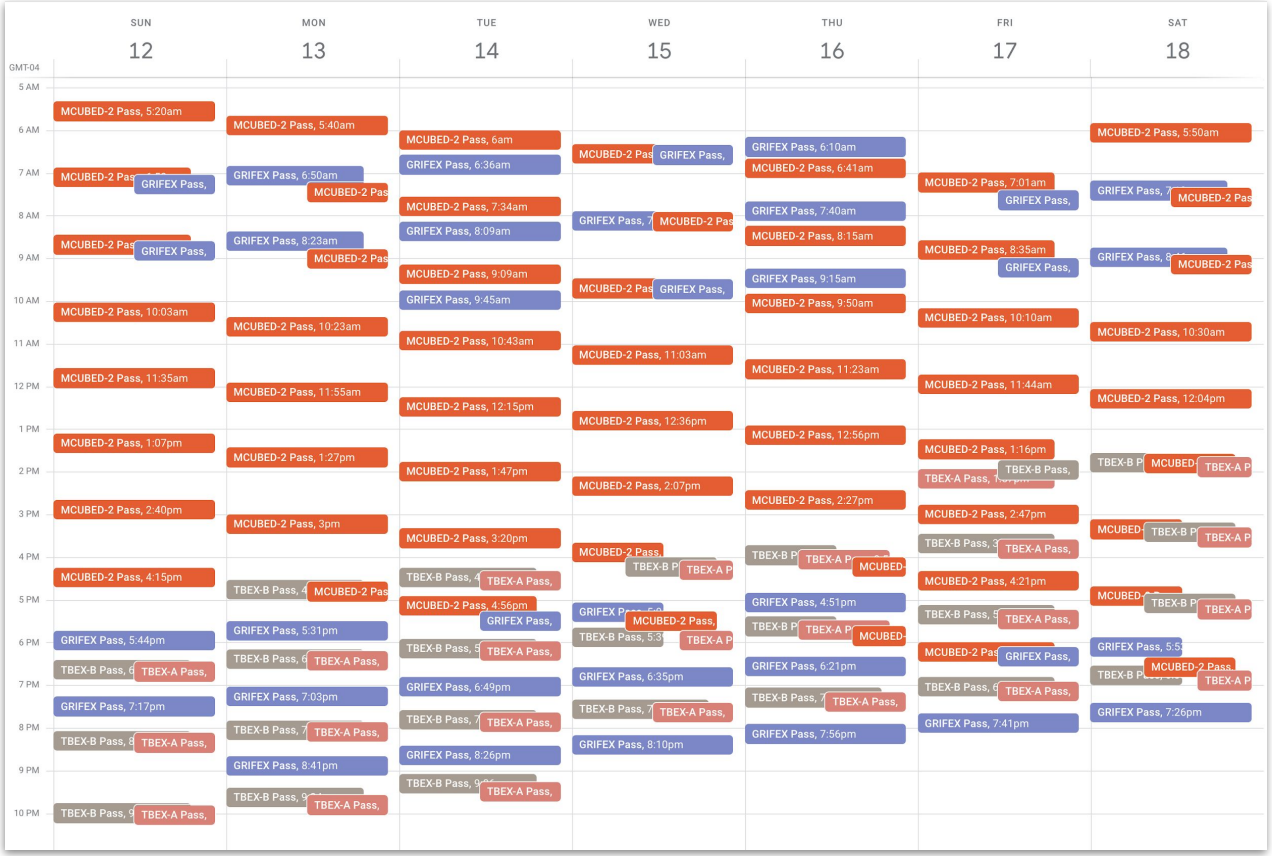
GitLab pipelines allow us to perform automated end-to-end tests, periodically run data backups, and ensure that any software changes are validated before being put into production on the ground, or in orbit.

Team communications and collaboration - Slack





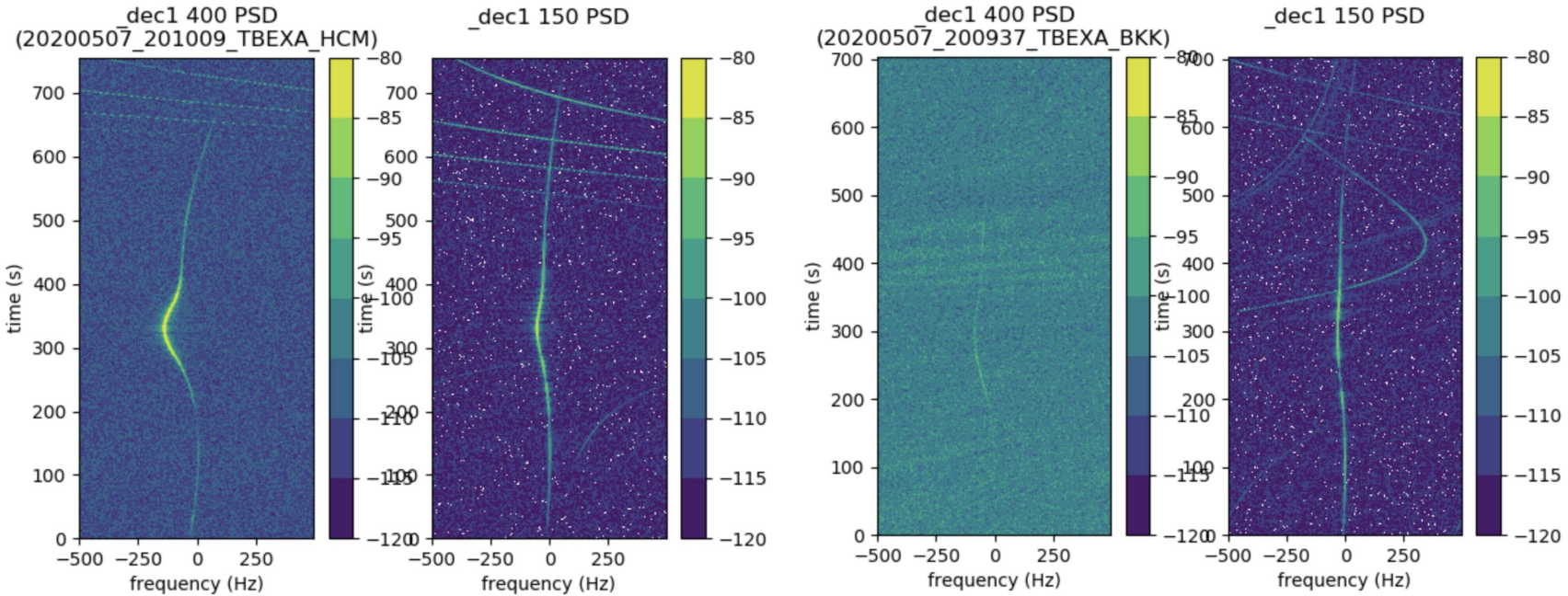
Team communications and collaboration - GCal



Conclusions

- By using modern software system design—keeping our applications small, networked, and developer oriented, MIDAS has undoubtedly transformed MXL satellite operations for the better.
- Automation, monitoring, and communication tools have enabled remote work during the COVID-19 pandemic.
- Our work clearly illustrates that modern software solutions that emphasize system autonomy and measurability have the power to dramatically improve small satellite operations and development.
- Next steps include automated command campaigns to ease the load on our operators and migrating our servers to the cloud to decrease maintenance time.

MIDAS Scheduled E-TBEx Operations In Bangkok and Ho Chi Minh



Questions?