Dynamic Cloud-Based Data Collection System

George F. Lawton III
Lawton Software, LLC

The Problem

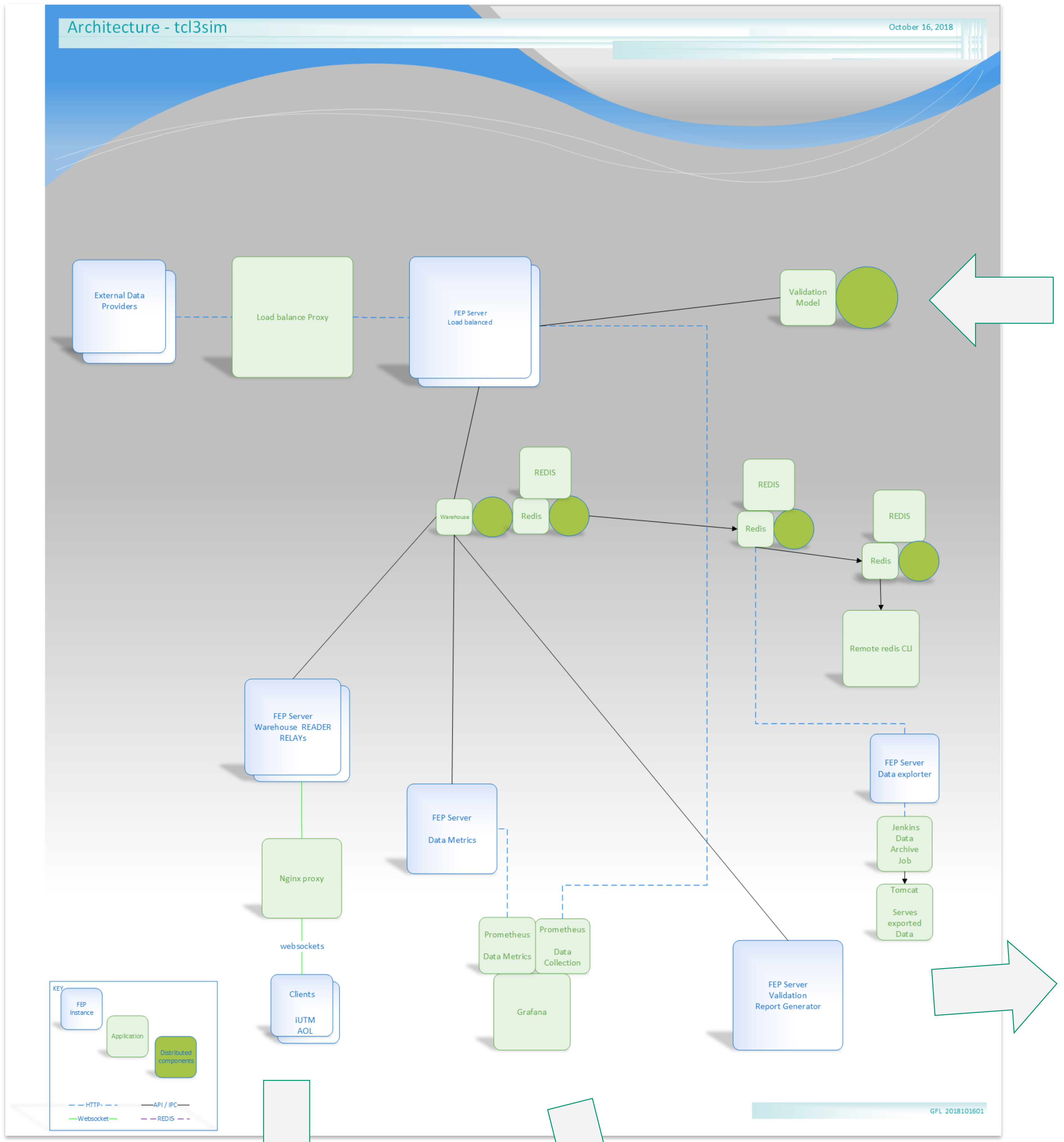
Airspace research conducted at NASA Ames Research Center required a data collection system that could collect data from multiple data providers distributed across the nation and interconnected via the internet. The nature of the research caused rapidly changing data schemas, which can be a challenge with fixed schema-based data collection and visual display systems. Breaking schema changes caused faults with legacy systems.

A Solution

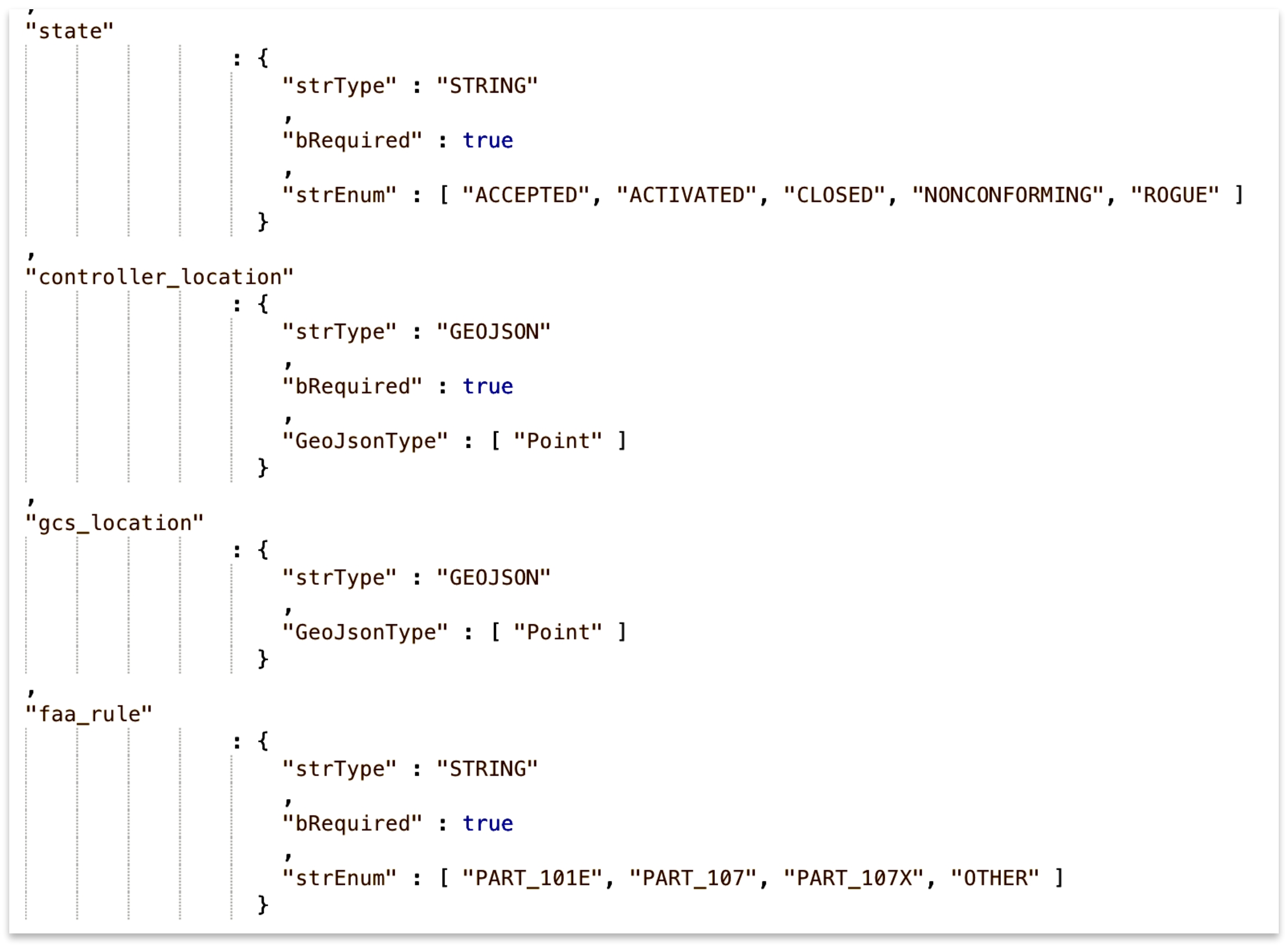
A cloud-based, schema-less, and dynamic validation data collection system solved the problem. This system was conceptualized and built around dynamic validation models.

data collection We integrated comprehensive system health metrics with alerting into the system, and built the system to provide a comprehensive solution for data collection and archiving of rapidly evolving incoming data. The system collects incoming Java Script Object Notation (JSON) data via a HTTP RESTful interface.

All incoming data was subjected to a validation engine. This engine validated all fields within the data packet against a dynamic validation model. Models could be changed in real-time to provide dynamic validation. The validation results were collected and used to generate periodic reports and fed into the metrics subsystem.

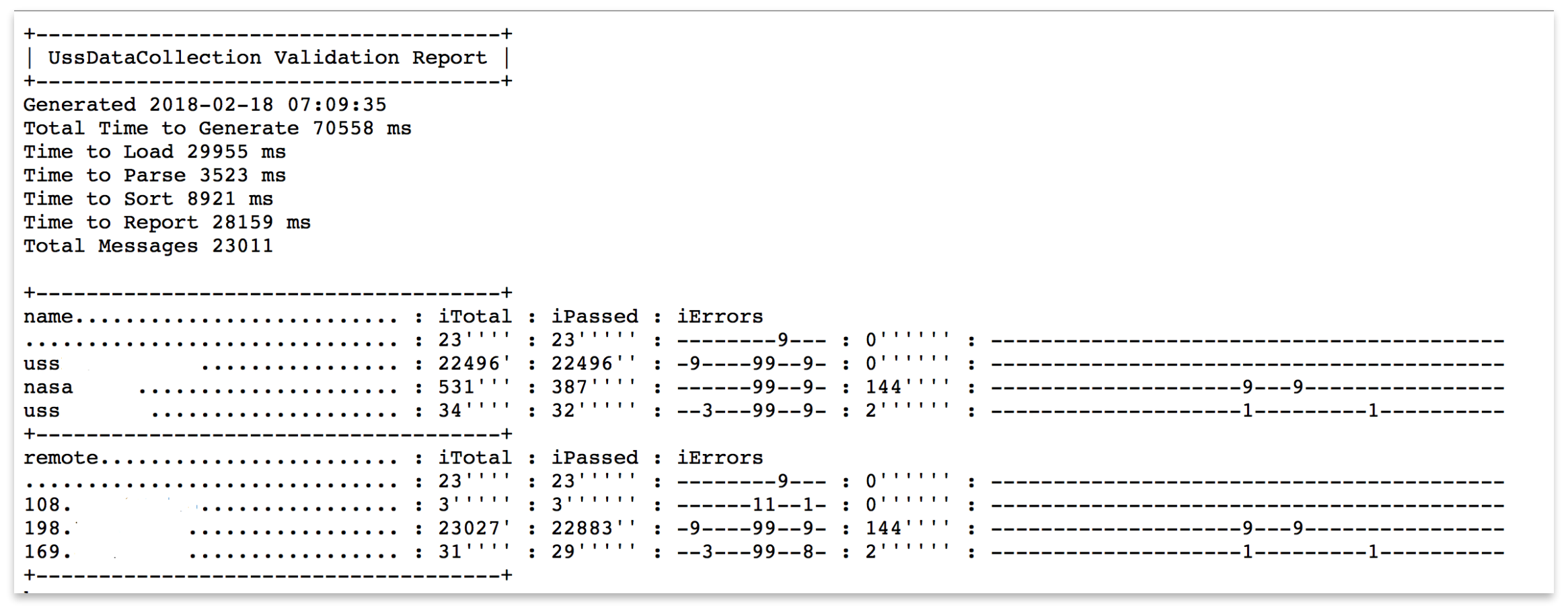


System Architecture



```
,
"state"
                    : {
                        "strType" : "STRING"
                        ,
                        "bRequired" : true
                        ,
                        "strEnum" : [ "ACCEPTED", "ACTIVATED", "CLOSED", "NONCONFORMING", "ROGUE" ]
                    }
,
"controller_location"
                    : {
                        "strType" : "GEOJSON"
                        ,
                        "bRequired" : true
                        ,
                        "GeoJsonType" : [ "Point" ]
                    }
,
"gcs_location"
                    : {
                        "strType" : "GEOJSON"
                        ,
                        "GeoJsonType" : [ "Point" ]
                    }
,
"faa_rule"
                    : {
                        "strType" : "STRING"
                        ,
                        "bRequired" : true
                        ,
                        "strEnum" : [ "PART_101E", "PART_107", "PART_107X", "OTHER" ]
                    }
```

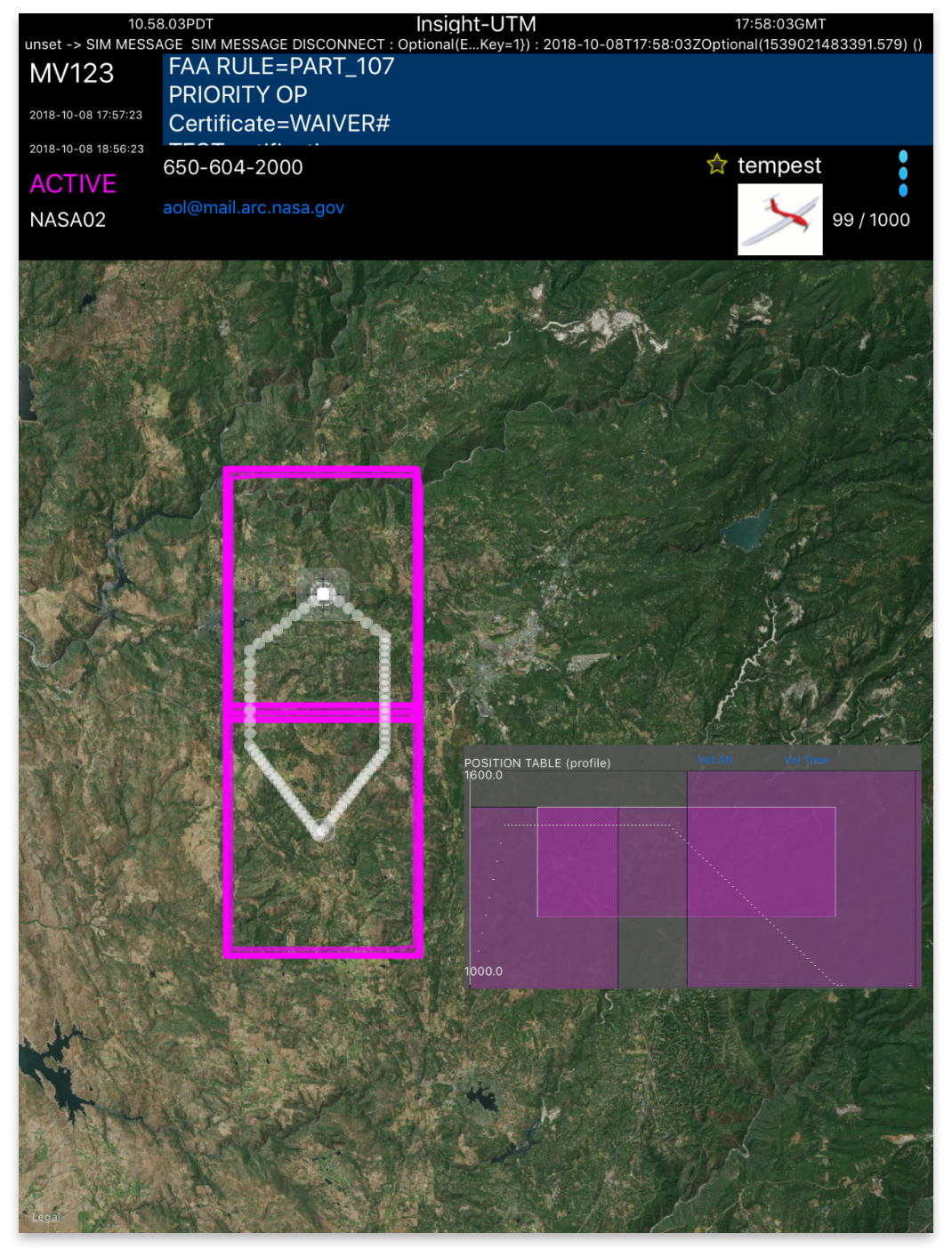
Dynamic Validation Model (real-time editable)



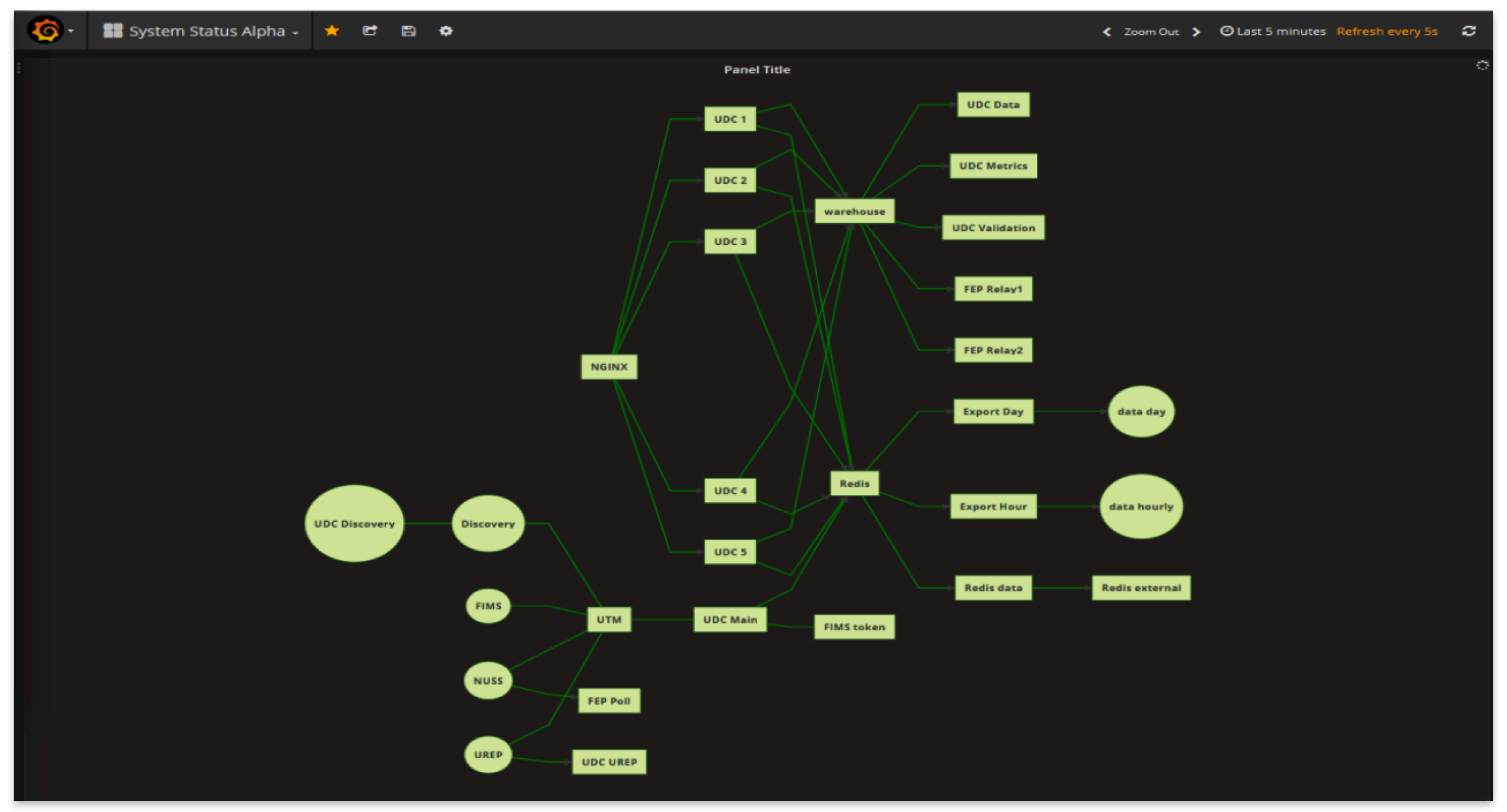
```
+--------------------------------------+
| UssDataCollection Validation Report |
+--------------------------------------+
Generated 2018-02-18 07:09:35
Total Time to Generate 70558 ms
Time to Load 29955 ms
Time to Parse 3523 ms
Time to Sort 8921 ms
Time to Report 28159 ms
Total Messages 23011

+--------------------------------------+
name.......................... : iTotal : iPassed : iErrors
.............................. : 23'''' : 23''''' : --------9--- : 0'''''' : ----------------------------------------
uss             ................ : 22496' : 22496'' : -9----99--9- : 0'''''' : ----------------------------------------
nasa      ..................... : 531''' : 387'''' : ------99--9- : 144'''' : --------------------9---9---------------
uss        .................... : 34'''' : 32''''' : --3---99--9- : 2'''''' : --------------------1---------1----------
+--------------------------------------+
remote........................ : iTotal : iPassed : iErrors
.............................. : 23'''' : 23''''' : --------9--- : 0'''''' : ----------------------------------------
108.            ................ : 3''''' : 3'''''' : ------11--1- : 0'''''' : ----------------------------------------
198.           ................. : 23027' : 22883'' : -9----99--9- : 144'''' : --------------------9---9---------------
169.           ................. : 31'''' : 29''''' : --3---99--8- : 2'''''' : --------------------1---------1----------
+--------------------------------------+
```

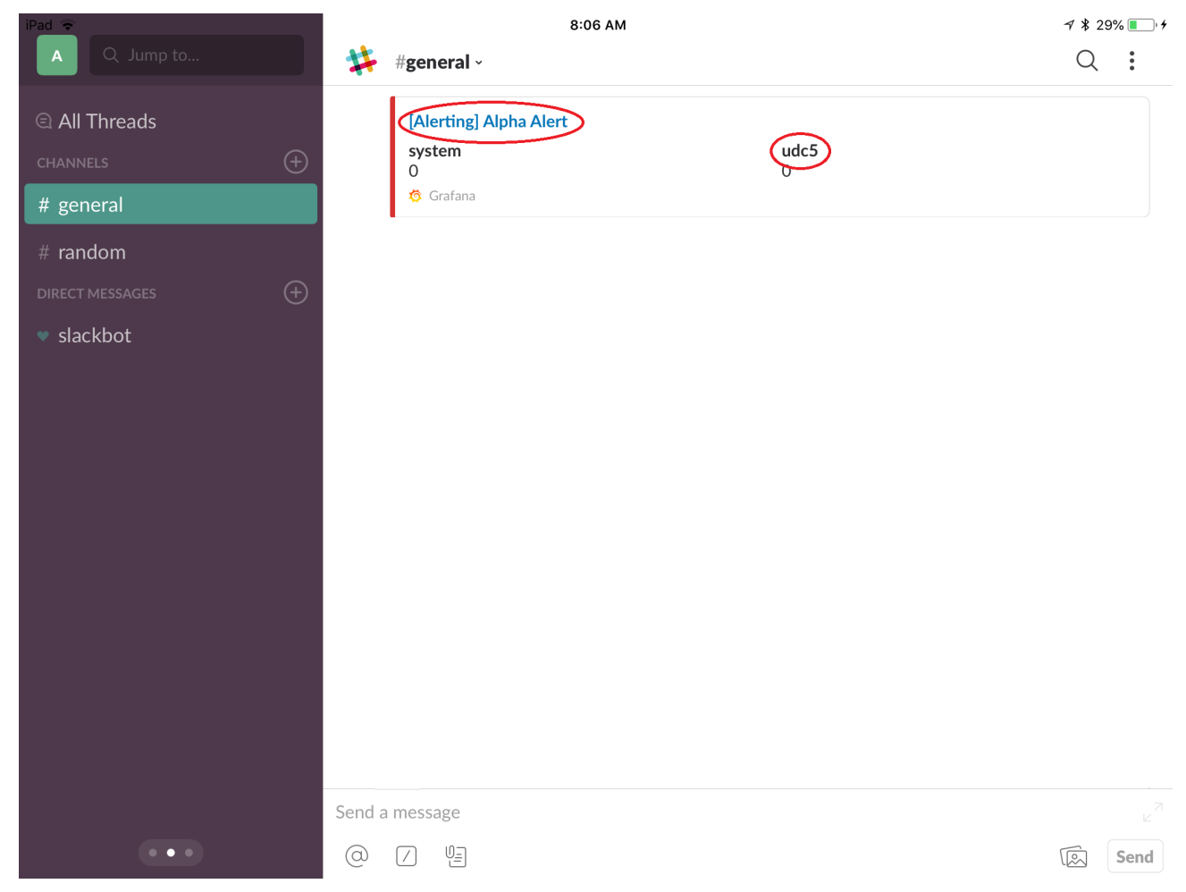
Web based reports



iOS app



Grafana System status



Slack Alerting

The metrics subsystem provided data metrics of all data validity results as well as system health status. We collected and cataloged metrics with Prometheus, which fed Grafana dashboard displays. The front-end dashboards provided both high-level system overall health status as well as detailed incoming data validation results. The Grafana alerting was linked to email and Slack notifications for real-time system alerts.

The system was used during a large-scale distributed simulation and flight tests involving 6 test sites and more than 30 vehicles across the CONUS.

All data was stored in redundant disparate data stores for data protection. A legacy proprietary data store was used as a backup system and a distributed database Cassandra was used for long-term archiving.

Time based fields in the data produced interesting outcomes. For example, we found a recurring issue with timestamps submitted in the future when they should have been in the past, triggering a validation failure. The root-cause of this error was a lack of aggressive system time synchronization amongst the data providers. The dynamic validation system provided a mechanism to adjust the time thresholds in real-time.

The data collection and validation system proved effective in mitigating incoming data issues through simple, real-time validation model adjustment.

Future enhancements to the system will entail a real-time report generator. This would address current performance limitations with the validation report generation engine.

This work was accomplished on contract with NASA Ames Research Center at Moffett Field, CA.