Constellation Paper Status

Simon Spannagel, DESY
EDDA Project Meeting
07/07/2025

A *Constellation* Paper

Goal

- Have a reference paper for people to cite when using the software
- Document our scientific work
- Increase visibility :)

Contents

- Context of development (hackathons, …)
- Overview of framework, also technical depth
- Examples of current users

Status

- First draft mostly written, 15 pages currently
- First comments from co-author implemented
- Missing parts:
  - Section about new Telemetry Console interface
  - Examples (all of them – asked MADMAX, Prevas/SKB, DESY/H2M)
  - Conclusions

Table of Contents

Contents

1. Introduction

Sequence Diagrams

- Spent some dime writing sequence diagrams for all protocols
- Idea: describe working principle without being too technical

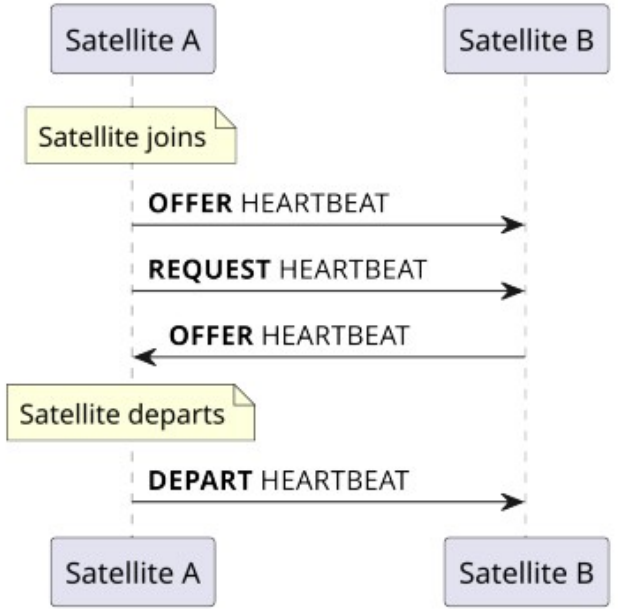

Figure 1: Sequence diagram for CHIRP showing the message flow between an already running satellite (B) and a newly started satellite (A). Satellite A offers a service and requests a service from others. Satellite B answers the request, and finally Satellite A departs again.

User Interfaces with Screenshots

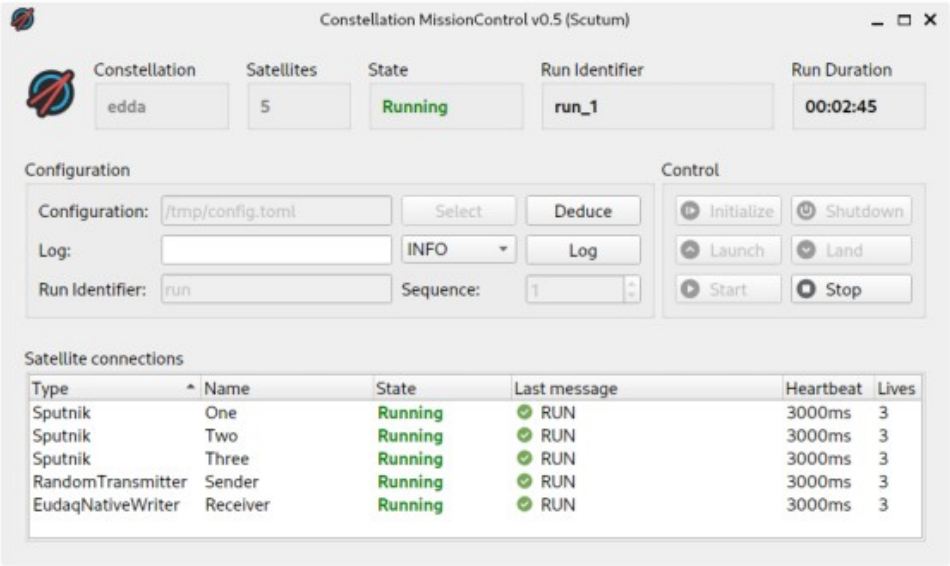

Figure 14: Main window of the MissionControl controller interface, displaying a Constellation in running state with five satellites connected.

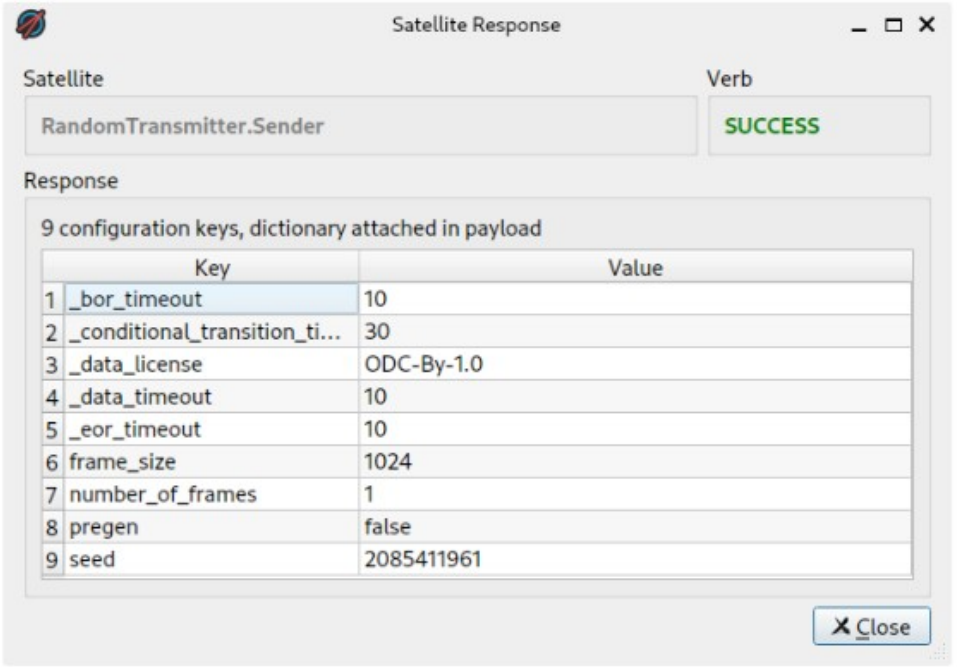

Figure 15: Satellite command response dialog of the MissionControl controller interface, displaying the command payload to the `get_config` command.

- The waiting satellite runs into the timeout for condi-

Short Python Code Example

- Part of user interface section
- Demonstrate that the framework can be controller entirely from a script

```python
import time
from constellation.core.controller import
↪  ScriptableController

# Create controller
ctrl = ScriptableController(group_name)
constellation = ctrl.constellation

for ivl in [8, 64, 128]:
    # Reconfigure to new parameter value
    cfg = {"interval": ivl}
    constellation.Sputnik.reconfigure(cfg)
    ctrl.await_state(SatelliteState.ORBIT)
    # Repeat measurement four times:
    for run in range(1, 4):
        constellation.start(f"i{ivl}_r{run}")
        ctrl.await_state(SatelliteState.RUN)
        # Run for 15 seconds
        time.sleep(15)
        constellation.stop()
        ctrl.await_state(SatelliteState.ORBIT)
```

Next steps

- Implementing comments from first readers
- Adding example sections as they come
- Distribute to the EDDA mailing list for feedback

Authorship – Suggestion:

- Everyone from EDDA may sign the paper as co-author if they wish
- We do this as opt-in:
  - Answer to the draft mail and say that you want to sign
  - Tell us what should appear in the CrEdiT statement
- Author order: main authors first, contributors alphabetically afterwards


DESY.