Graph Database Software For A Cosmology Telescope

Dhananjhay Bansal(Supervised by Adam D. Hincks)

David A. Dunlap Department of Astronomy & Astrophysics, University of Toronto, 50 St. George St., Toronto ON M5S 3H4, Canada

[1]

UNIVERSITY OF ALBERTA [6]

UNIVERSITY OF TORONTO [2]

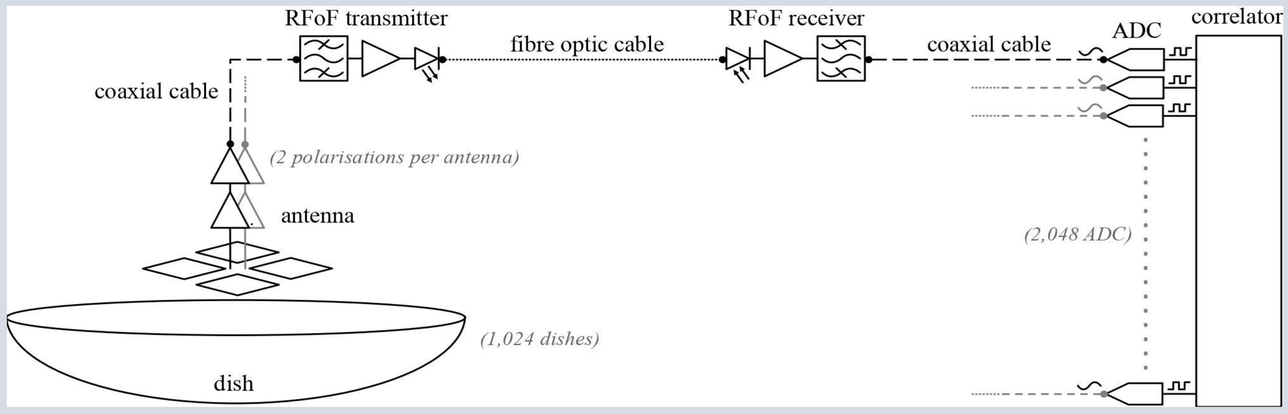
HIRAX

- The Hydrogen Intensity and Real-Time Analysis eXperiment (HIRAX) is a 21 cm intensity mapping experiment.
- 21 cm is the emission line of neutral hydrogen and it used to trace the gas in the large-scale structure.
- The full experiment will have over 2,000 signal chains.

Artist's rendering of HIRAX array. [3]

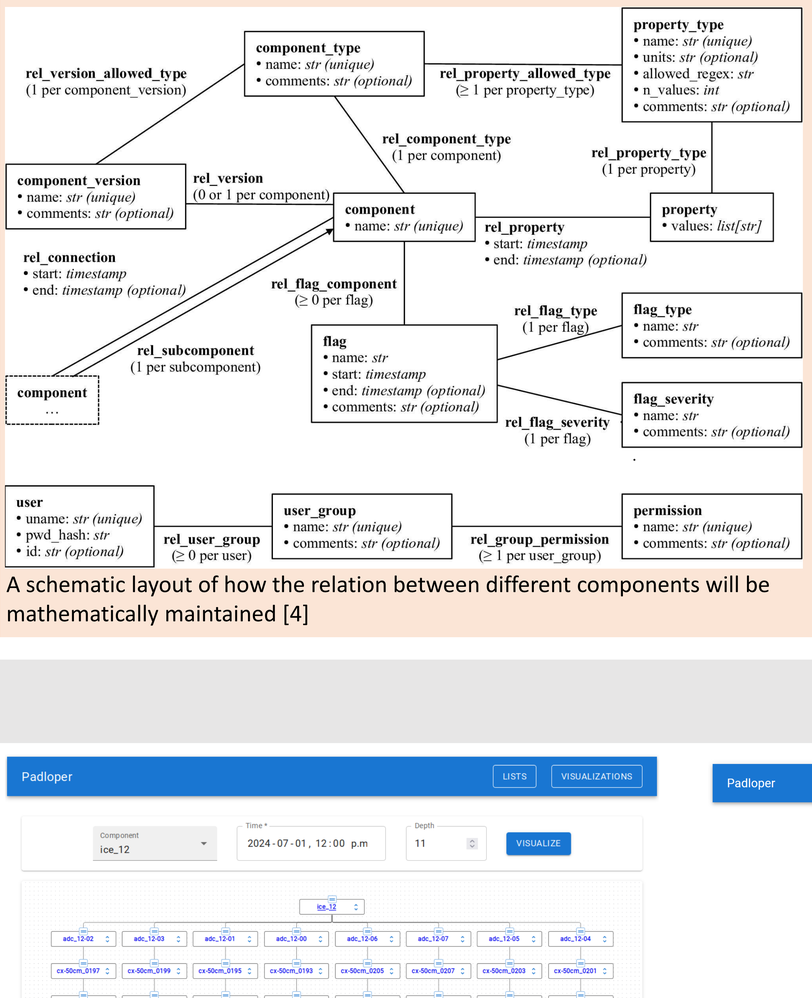
What Is A Signal Chain ?

- Padloper will keep track of connections, locations and states of each individual component—dual-polarization antennas, amplifiers, filters, cables, etc.
- Each new configurational modification will come with a timestamp so that data analysis can be performed efficiently.
- Since each signal chain can be different contingent on the type/version of components, the software will provide visualization of any signal chain at any given point of time.



Representation of a signal chain comprising of some of the main components in the experiment [4]

The Database



A schematic layout of how the relation between different components will be mathematically maintained [4]

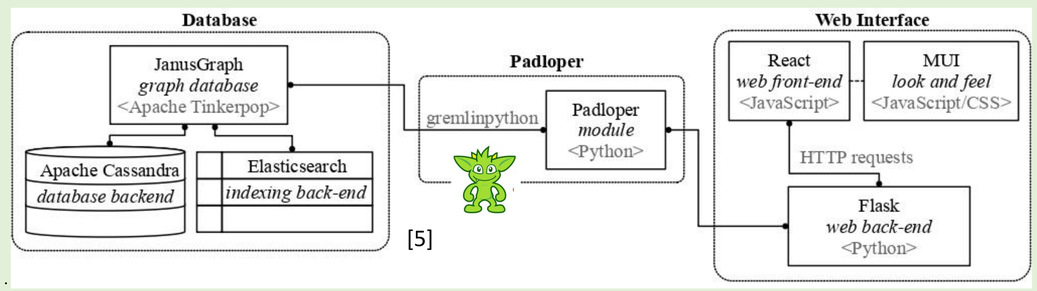
- The basic layout of how the relation between data is established in the database is represented by this mathematical graph.
- The vertices are represented by rectangles that box the vertices' categories(displayed in bold) along with the other attributes and edges are the connecting lines between vertices.
- As the experiment goes on, there will be adjustments in a component's properties and connections, therefore, the edges will consist of starting and ending timestamps.

Simple tasks such as finding the ongoing properties and connections of a component, or the associated flags, can be executed using graph traversal for which well developed algorithms exist

```
g.V().has("category", "component_type").has("name", "antenna")
g.E().hasLabel("rel_connection").has("start_time",P.lte(t)).has("end_time",P.gt(t))
```

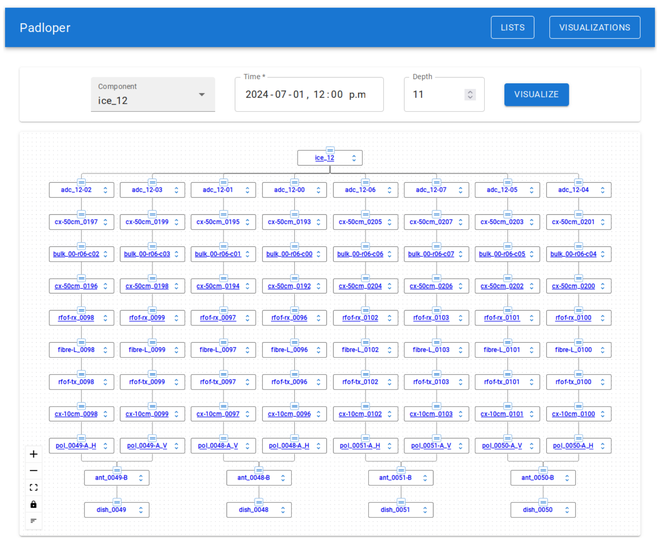
- The first line is a typical example of a gremlin query: It returns the relevant vertices in the database have the category of a 'component_type' and simultaneously have the name antenna.
- The second one returns the edges that have the label 'rel_connection' and were active at the given time t.

The Software Stack

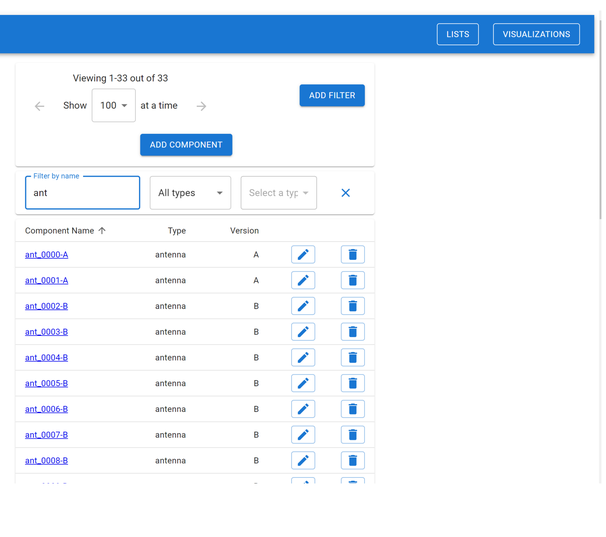


Representation of the general layout of the Padloper software [4]

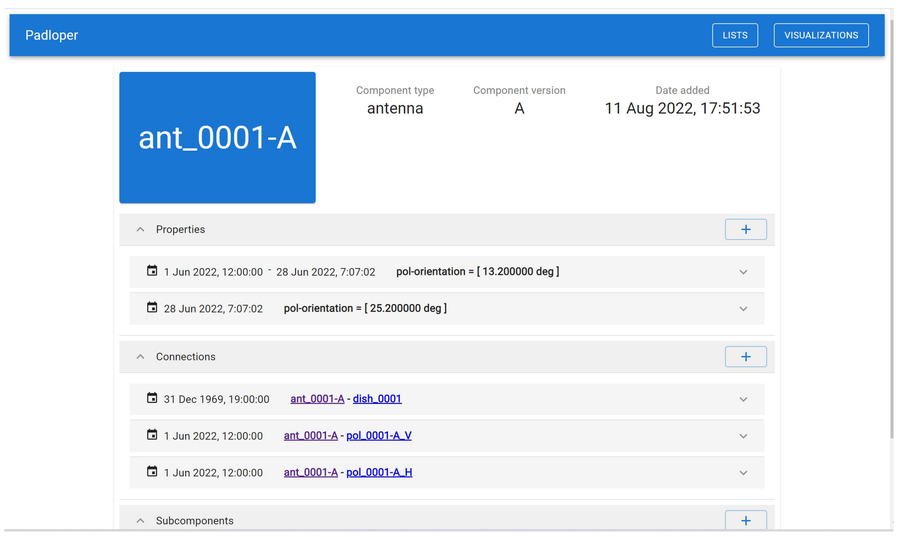
Release The Padloper !



Representation of a set of 8 signal chains at a given point in time. [4]



Filtering antennas from the list of all the components. [4]



A specific component page displaying the component's properties, connections, subcomponent connections and associated flags. [4]

Upcoming Improvements

- Implementation of the authentication system is not fully complete. Only the framework in the backend has been laid out but no logic has been set up yet.
- Optimizing JanusGraph indices seems to be an important development right now. Since the software would be handling tens of thousands of components and their connections, indexing is crucial to the software's longevity.

References

[1] [https://www.nationalgeographic.org/projects/photo-ark/animal/homopus-areolatus/]
[2] Credit for the photo: [https://studyarchitecture.com/school/university-of-toronto/]
[3] Credit: Cynthia Chiang
[4] Hincks, Zavyalov & Bansal, Proc. SPIE. 12189-11 (2022, in prep)
[5] [https://tinkerpop.apache.org/docs/current/reference/]
[6] Credit for the photo: Ruben Gonzalez

Acknowledgement

- Anatoly Zavyalov