The Alpine Environmental Data Analysis Center

Virtual Alpine Observatory

alpendac.eu

Scheduling System for Remote Control of Instruments used for Atmospheric Observation

Martin Schumann[6], Johannes Munke[1], Patrick Hannawald[3], Stephan Hachinger[1], Inga Beck[5], Alexander Götz[1], Oleg Goussev[3], Jana Handschuh[2,3], Helmut Heller[1], Roland Mair[4], Till Rehm[5], Bianca Wittmann[4], Sabine Wüst[3], Michael Bittner[2,3], Jan Schmidt[6], Dieter Kranzlmüller[1,6]

[1] Leibniz Supercomputing Centre (LRZ), Garching b. M., Germany
[2] University of Augsburg, Institute of Physics, Augsburg, Germany
[3] German Aerospace Center (DLR), Earth Observation Center, Weßling, Germany
[4] bifa Umweltinstitut GmbH, Augsburg, Germany
[5] Environmental Research Station Schneefernerhaus (UFS), Zugspitze, Germany
[6] Ludwig Maximilian University (LMU), Munich, Germany

Overview of Work

- We describe the design and implementation of an advanced framework for the scheduling and execution of multi-user instrument systems, used by scientists in the AlpEnDAC collaboration.
- Our framework is implemented using an optimization based scheduler (Google's OR-Tools) to ensure maximum instrument use and to minimize idle times.
- We utilize state of the art deployment and execution systems, including Docker and Kubernetes, to maximize uptime and stability.
- Thorough documentation makes the system flexible, simple to use, and supports easy adaptation for various types of instruments.

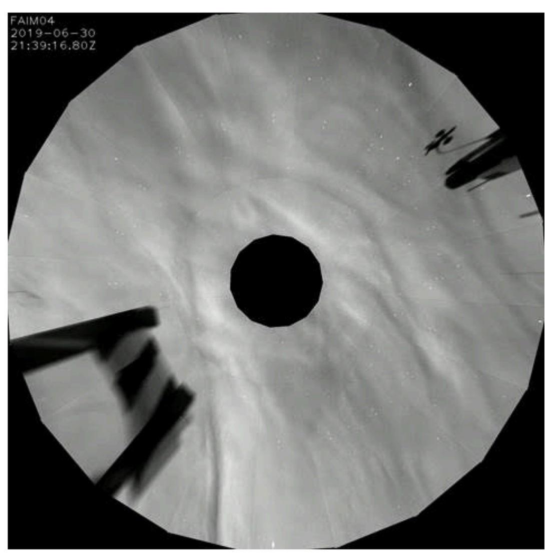
Introduction

- AlpEnDAC [Alp] is a facility for Research Data Management and on-demand simulation, which "focuses on monitoring, understanding and forecasting of environmental changes in the Alpine region".
- Run by the DLR, the University of Augsburg, the LRZ, the Environment Research Station Schneefernerhaus (UFS) and the bifa Umweltinstitut in the scope of a project collaboration. The collaboration has been funded by the Bavarian State Ministry of the Environment and Consumer Protection.
- Scientists of the "Virtual Alpine Observatory" collaboration usually use their own instruments
- One AlpEnDAC Goal: provide programmable interfaces to instruments, as the scientists could **strongly profit from mutual access to instruments and data**
- Currently missing:
  - Centralized reference point for API descriptions and access
  - Cross institution access to instruments
  - Scheduling system

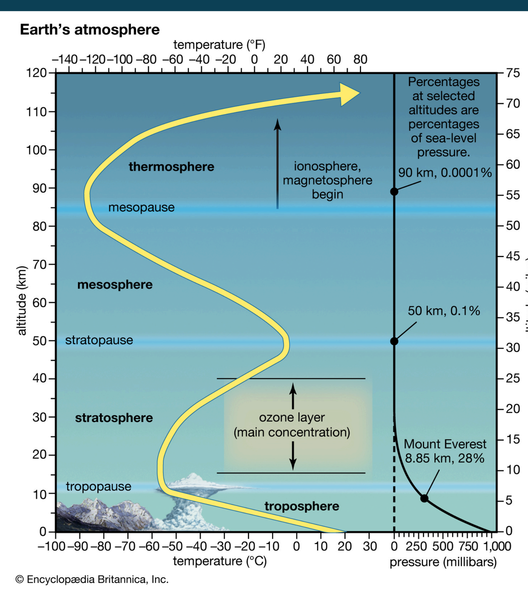
➔ **Our Solution: Operating-on-Demand (OOD) System which allows for scheduling and instrument access**

➔ **Aim of Work: Create modern framework and API for submitting jobs and data to AlpEnDAC**

Reference Implementation for a FAIM System

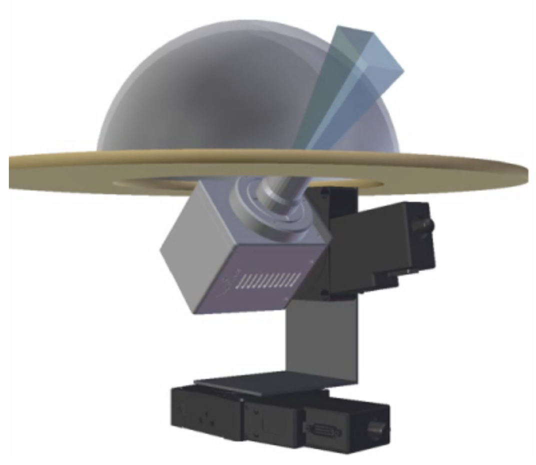


A full "allsky" airglow composite image taken using the scanning FAIM system

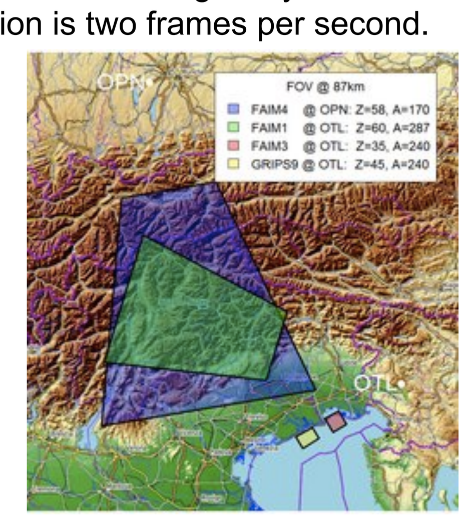


"Layers of Earth's atmosphere". [ENC] Imaging takes place in the upper mesosphere/lower thermosphere (alt.: ~86km).

- The FAIM system (operated by DLR Oberpfaffenhofen) is used to observe "OH nightglow emission" in the short-wave infrared located at an altitude of ~82-90km in order to obtain information about atmospheric dynamics above specific areas in the Alps. It is also possible to combine observations with other instruments at specific locations.
- The camera is equipped with a pan-tilt-mechanism allowing it to scan the night sky horizon to horizon (default) or point to a region of interest on demand. Its temporal resolution is two frames per second.

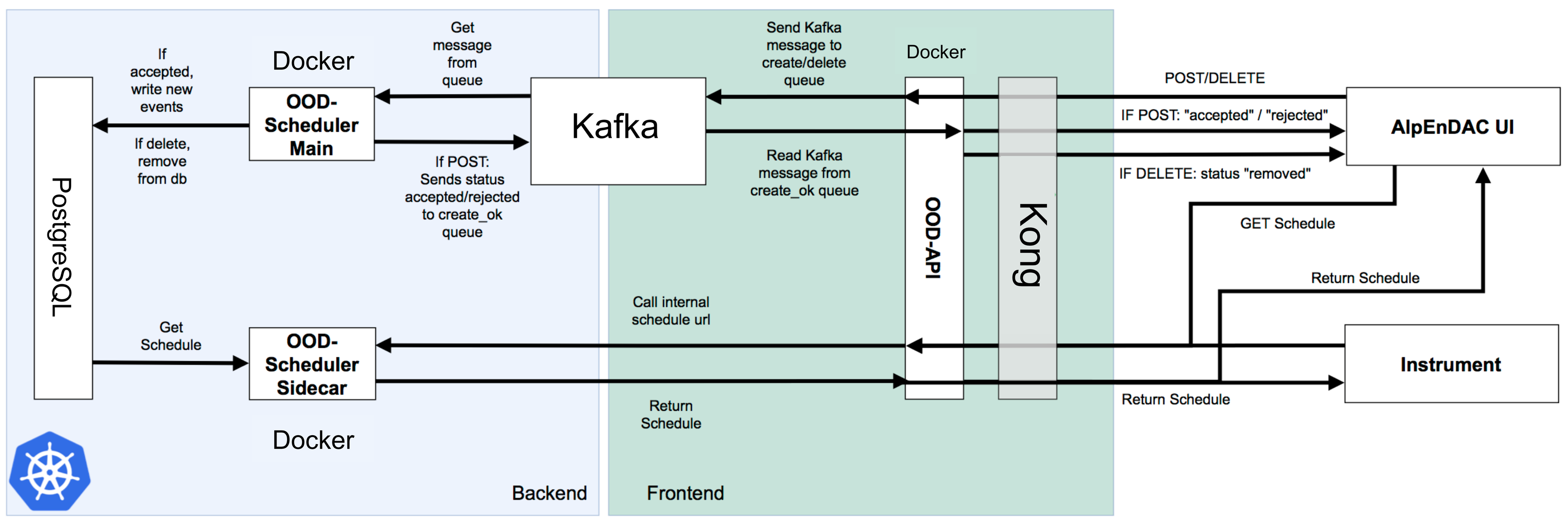


An illustration of the scanning FAIM system. Azimuth and zenith angles are adjusted by 2 rotation stages.



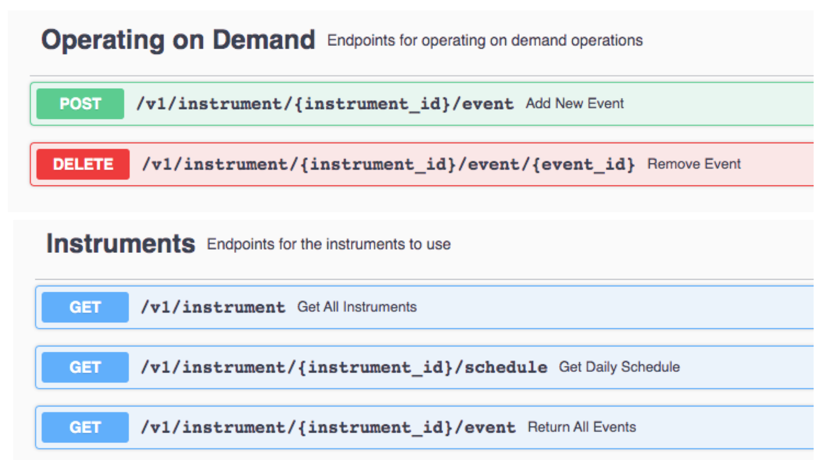
Location of various FAIM cameras in the Alps.
"OPN" = Oberpfaffenhoffen in Germany
"OTL" = Otlica in Slovenia.

Architecture



REST API

- HTTPS API endpoints (using OpenAPI) for request submission and instrument communication
- Endpoints to add/delete events
- Users can get a schedule for the current night and a list of all submitted events
- Instrument gets schedule from the schedule endpoint and executes these commands
- Framework allows for expanding to other instruments



Scheduling Software

The system uses a scheduler, which receives a list of possible jobs that should be executed and finds the **optimal subset of jobs** and their **temporal ordering** that can be actually be executed. Scheduling conflicts occur when multiple users want to use an instrument at the same time, measuring different things, e.g. for FAIM, certain portions of the sky.

A scheduler is needed, because:

- scheduling is important to ensure **max fairness and utilization** of instruments,
- the system must be able to schedule instruments remotely,
- and the framework and scheduling system being able to **adapt** and be used for a variety of different instrument needs makes it **useful for many applications**.

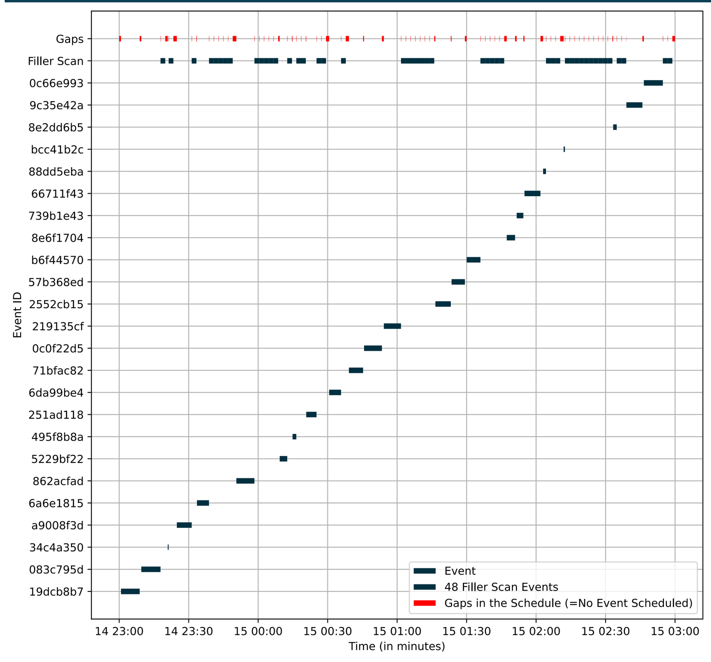
Requirements:

- Ease of implementation and reuse/adaptability for other (future) instruments
- Fast generation of the schedule (Goal: 2 seconds)
- High "quality" of schedules: scheduler should **minimize idle time**

- Possible Candidates: **Google's OR-Tools [PF]** and **NASA's EUROPA [BBD+12]**

➔ For our system, we selected **OR-Tools** as it met all the requirements

Production-Scale Schedule Example



Sequence to add a new event:

1. *User* submits request using the AlpEnDAC UI
2. The request is sent to the *frontend* (OOD-API), which adds it to a Kafka queue for the backend
3. The *backend* reads the message from the queue and tries to schedule the event.
4. If the event fits into the schedule, an updated schedule is written into the database. This update might include other events being removed from the DB, and includes new filler events being generated (to minimize idle time).
5. The *backend* sends an "accepted" (event was added) or "rejected" (it was not) message over Kafka to the *frontend*.
6. The *user* receives a response if their request was accepted or rejected.

- Input: 50 randomly generated events
- 24 events scheduled
- Filling schedule gaps: 1h 38m less idle time, out of a total of 4h

Discussion and Analysis

✓ Requirements fully met:
  - ✓ System can be reused easily and is easy to adapt
  - ✓ OR-Tools SAT Solver minimizes idle time as much as possible
  - ✓ Execution speed is <= 2s for the full roundtrip, including generating a new schedule
- Positive feedback for implementation from users of FAIM and in production use for some time
  ➔ shows viability of the platform
- Scheduler enables better utilization and fairness
- Constraint programming (OR-Tools) vs. ML-based algorithms:
  - Better insight into decisions
  - Predictability
  - Schedules can be proven optimal

References

[Alp] AlpEnDAC: *AlpEnDAC Homepage*. https://www.alpendac.eu/

[PF] Perron, Laurent ; Furnon, Vincent: OR-Tools. https://developers.google.com/optimization/. – Version 8.1, 2019-7-19

[Enc] Encyclopædia Britannica: Layers of Earth's atmosphere. https://www.britannica.com/science/mesopause/images-videos#/media/1/376810/99826 accessed on 06-02-2022

[BBD+12] Barreiro, Javier ; Boyce, Matthew ; Do, Minh ; Frank, Jeremy ; Iatauro, Michael ; Kichkaylo, Tatiana ; Morris, Paul ; Ong, James ; Remolina, Emilio ; Smith, Tristan u. a.: EUROPA: A platform for AI planning, scheduling, constraint programming, and optimization. In: 4th International Competition on Knowledge Engineering for Planning and Scheduling (ICKEPS) (2012)

Contact: rdm@lists.lrz.de (Research Data Management Team of LRZ)

lrz DLR Universität Augsburg University bifa Umweltinstitut Umwelt Forschungsstation Schneefernerhaus

financed by
Bavarian State Ministry of the Environment and Consumer Protection