Catalysts

A CLOUDFLIGHT COMPANY

| Michael Aspetsberger

The GRASP Cloud
Applications and Demonstration

Catalysts GmbH: **Michael Aspetsberger**, Stefan Amberger, Lukas Bindreiter, Petrut Cobarzan, Andreas Hangler, Christoph Holter, Verena Lanzinger, Daniel Marth, Moritz Wanzenböck

Laboratoire d'Optique Atmosphérique: Oleg Dubovik, Tatyana Lapyonok, Benjamin Torres

GRASP SAS: David Fuertes, Pavel Litvinov, Anton Lopatin, Cheng Chen

Kindai University: Itaru Sano, Sonoyo Mukai

I will not talk about meteorology…

"**Cloud computing is** a model for enabling ubiquitous, convenient, on-demand network **access to** […] **computing resources** (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned […]."

National Institute of Standards and Technology

Why does it make sense for GRASP?
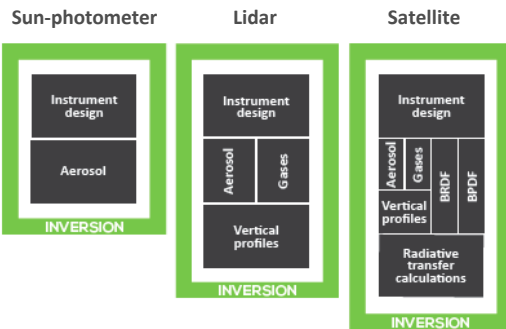
Many Instruments
POLDER, OLCI, SGLI,
photometer, lidar, etc.

Many Resources
Satellite Missions cover Terra- to Petabyte
Full On-Line Modelling

GRASP retrieval setups

Many Options
Inversion Configuration,
Forward Model Setup,
Data Preparation, etc.

Many Applications
Satellite, Ground-Based, In-Situ
Simulations, Verification, etc.

The Classic Way of Using GRASP

Standalone Operation

1. Download GRASP-Open Code
2. Install Compilers and Libraries
3. Compile GRASP

The necessary hassle

4. Prepare Input Data (e.g. SDATA)
5. Prepare Configuration

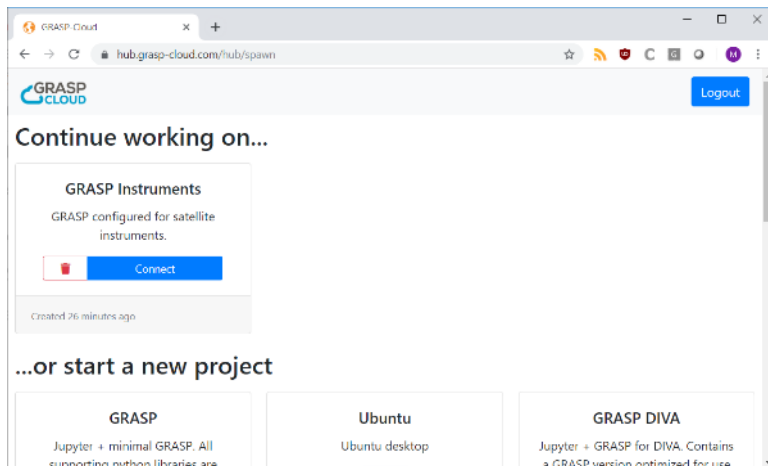**Use-case dependent**

6. Run GRASP
7. Load and Interpret Output

This is where the focus should be.

Finding a better way gave birth to the idea of a GRASP Cloud.

Interacting with GRASP in GRASP CLOUD

Load GRASP

```
[1]:  import grasp.code
      import grasp.code.plot as gplot
      %pylab inline

      Info: Loaded /usr/lib/libpython_grasp.so
      Populating the interactive namespace from numpy and matplotlib
```
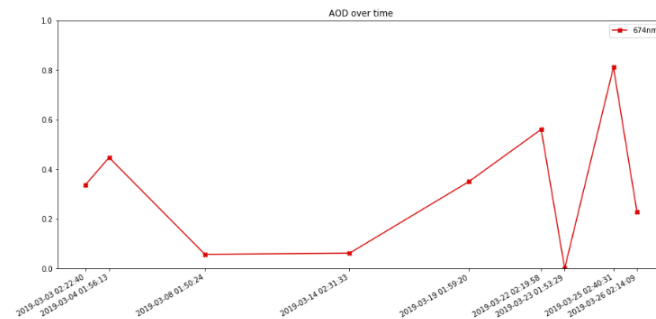
Setup and Run for SGLI

```
[2]:  data = sgli.read([380, 443, 490, 565, 674, 868, 1050, 1630, 2210])
```

```
[3]:  results = grasp.code.run(data, './yml/Process_SGLI_with_All_Intensity.yml', print_screen=False)
```
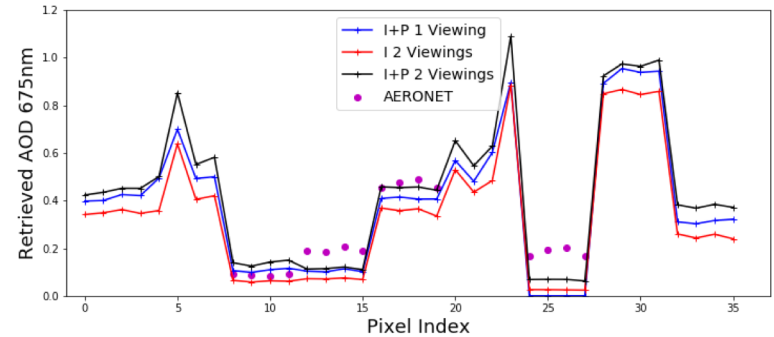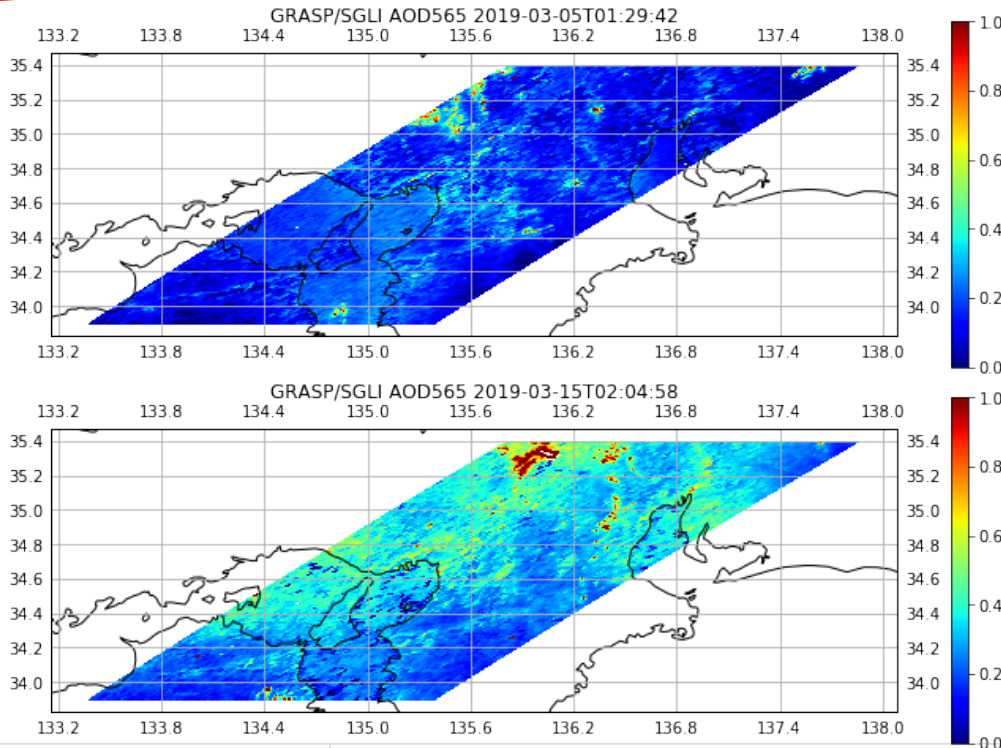
Analyze Results

```
[4]:  ts = gplot.TimeSeries("AOD over time")
      ts.plot(results[:, :, :].aerosol_opt.extt[:].sel(wl=674),xy_aggregator="mean")
      plt.ylim(0, 1.0)
```

```
[4]:  (0, 1.0)
```


Simplified Python Interface to …

- … process arbitrary data, tweak configuration, and analyze result data,

GRASP/SGLI Experiments

GRASP/SGLI AOD565 2019-03-05T01:29:42

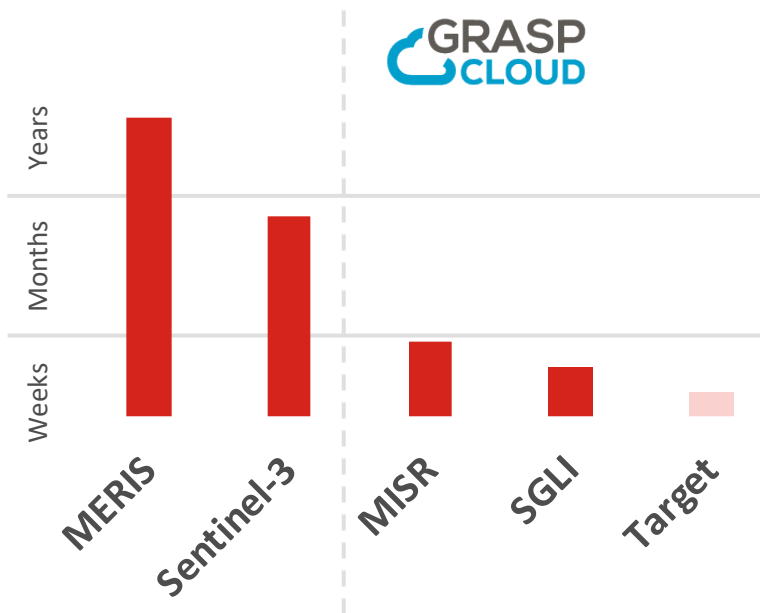GRASP/SGLI AOD565 2019-03-15T02:04:58

Explore different setups quickly

- Single vs. Multi Angle

- Polarization vs Intensity-Only

The effort of testing new instruments

Why is the effort decreasing?

- GRASP Cloud Standard Library for pre-processing

- Same Environment for Dev and Processing

- Newer Products are easier to use
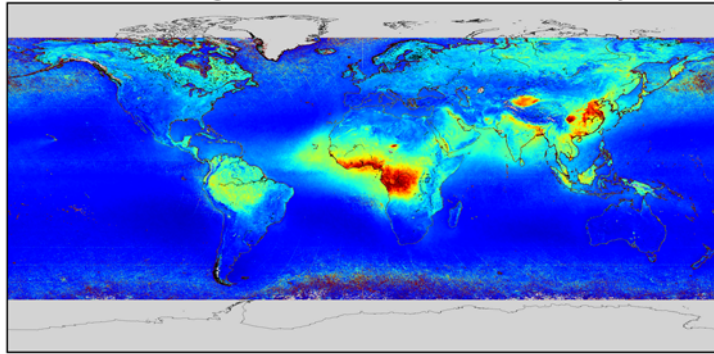
But Primarily: Less Round Trips!

- Data prep, retrieval and validation in one place

Recent Full Mission Re-Processing
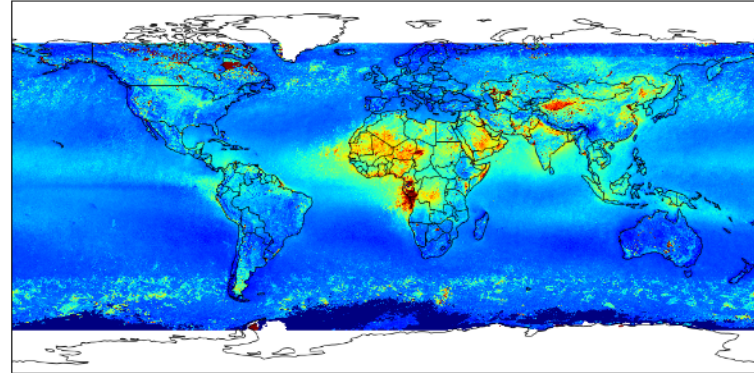
Processing Time: Months → Weeks

AOD 565 averaged for entire POLDER archive (~9 years)

PARASOL/POLDER (internal)

GRASP/OLCI-A AOD560 2016-04-25 to 2019-09-18

Sentinel-3A/OLCI (on behalf of EUM)

Validated datasets will be made available on www.grasp-open.com/products/, NRT available on user-request.
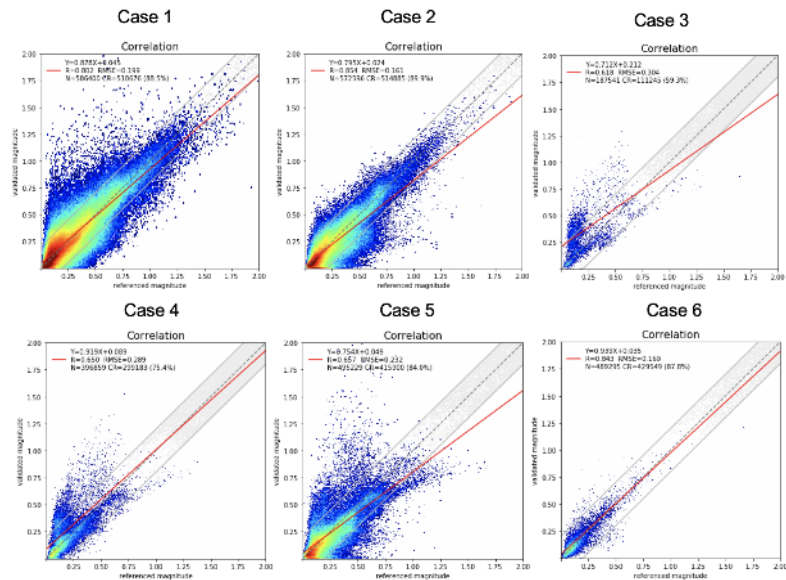
Intercomparison with GRASP CLOUD

Access to Intercomparison Datasets

- Ground-based, Satellite
- Models (planned)

Standard Routines

- Co-registration, Statistics
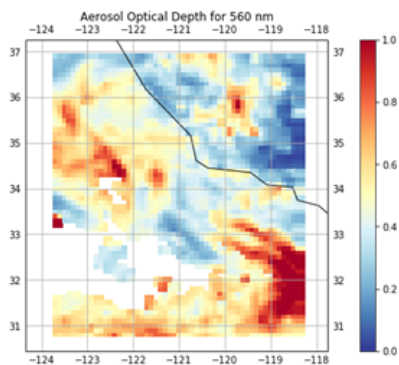
Automation & Reproducability

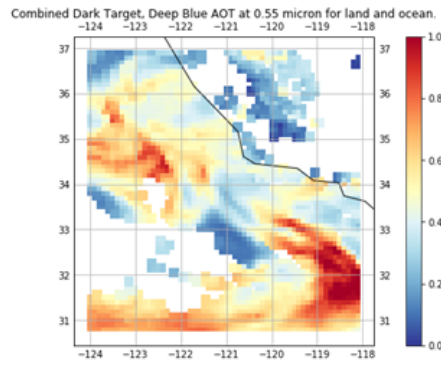Different PARASOL/POLDER Retrieval Setups vs. AERONET

The 2018 California Campfire
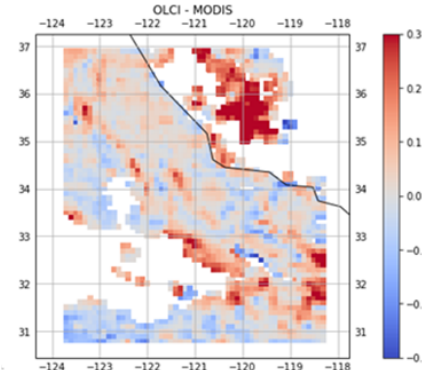
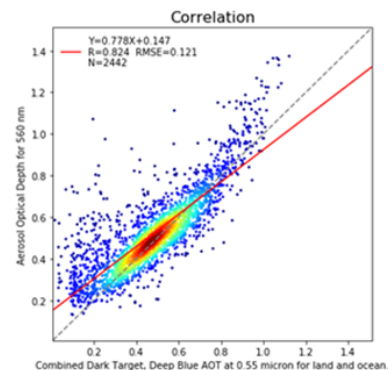## 2018-11-11 from GRASP/OLCI-A and MODIS TERRA DT/DP

1) Select AOI   2) Co-register   3) Difference   4) Statistics

Having intercomparison datasets and a consistent approach is essential!

What makes GRASP Cloud special?

1. **It makes GRASP easy to use.**

2. **It has an archive of data for inter-comparison.**

3. **It is powered by big infrastructure resources.**

"I would highlight how easy it is to run GRASP (in my case over 2 000 000 single retrievals) and how easy it is to do validation with pre-existing products."
Benjamin Torres, Univ. Lille

"GRASP is available in so many flavors. Everything that runs in GRASP Cloud can scale to near-real-time and re-processing using the same environment."
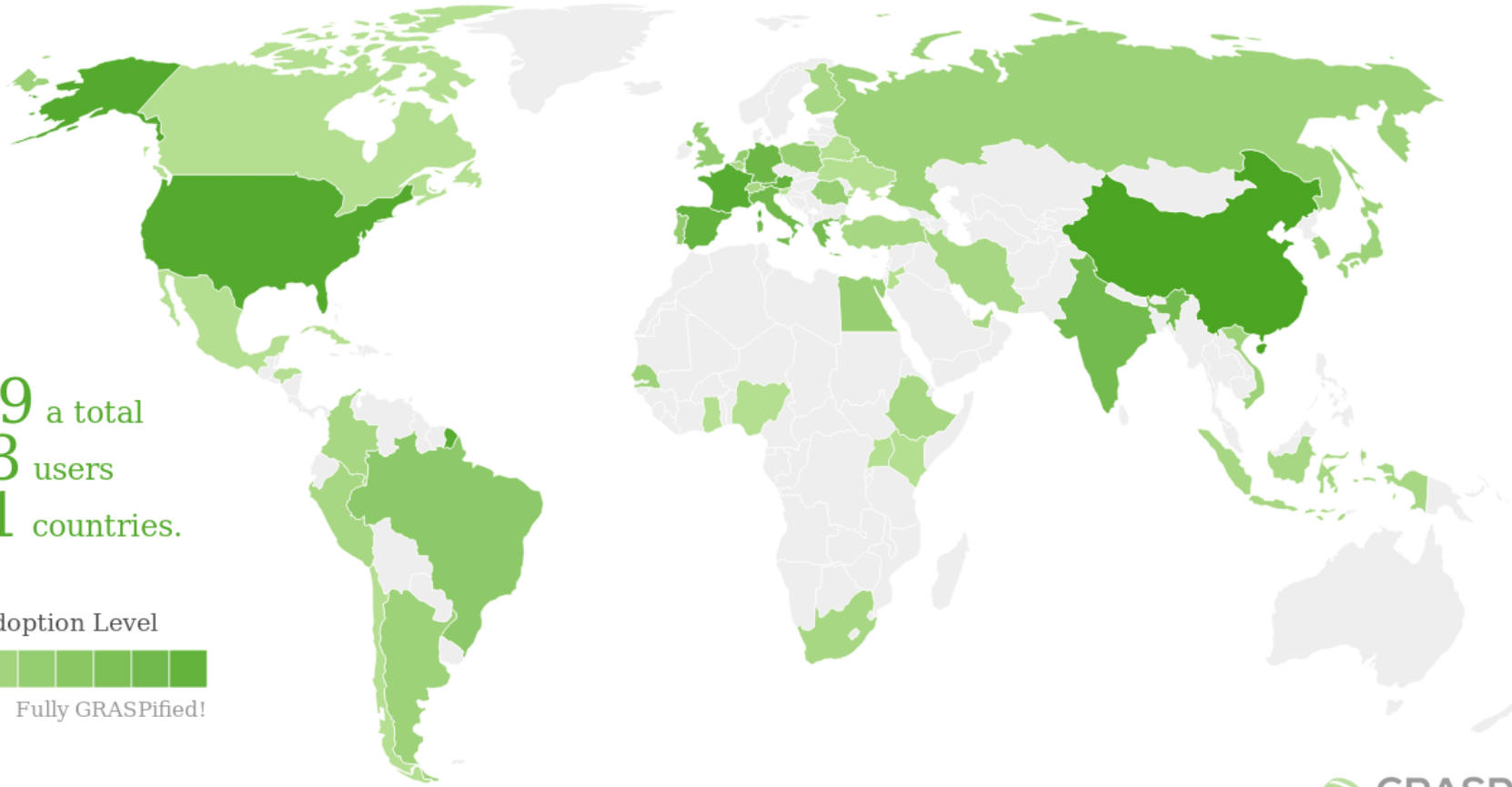Stefan Amberger, Catalysts

By 2019 a total of 403 users in 51 countries.

GRASP Adoption Level

Fully GRASPified!

GRASP

https://www.grasp-open.com/contact/