KISTERS

Python frameworks for the integration of a real-time data hub for meteorological and hydrological forecasting – benchmarks and design decisions.

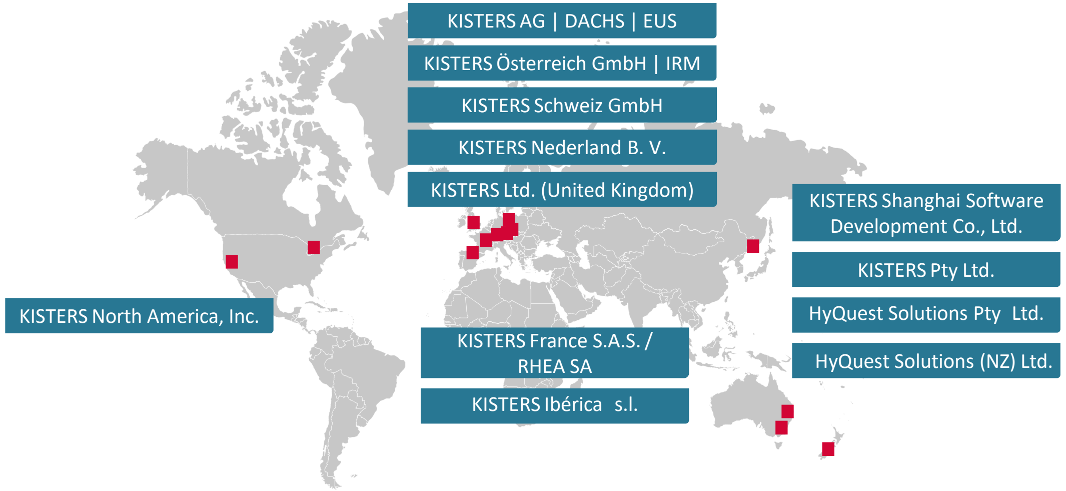
2018 Workshop on developing Python frameworks for earth system sciences

Alberto Sabater Morales, Jackie Leng

KISTERS at a glance

KISTERS

The KISTERS Group



KISTERS AG | DACHS | EUS

KISTERS Österreich GmbH | IRM

KISTERS Schweiz GmbH

KISTERS Nederland B. V.

KISTERS Ltd. (United Kingdom)

KISTERS North America, Inc.

KISTERS France S.A.S. / RHEA SA

KISTERS Ibérica s.l.

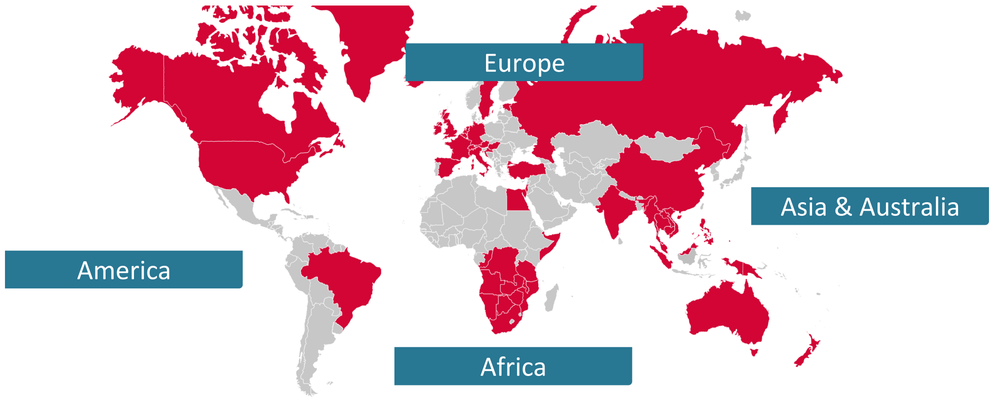
KISTERS Shanghai Software Development Co., Ltd.

KISTERS Pty Ltd.

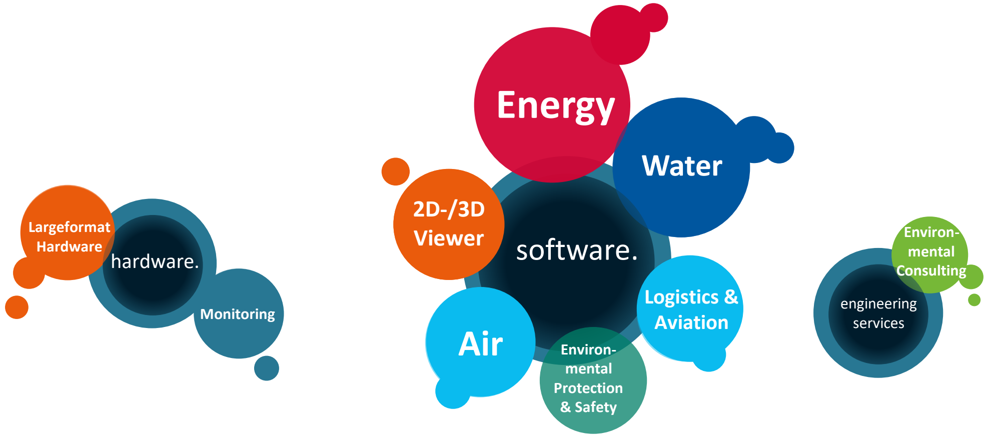
HyQuest Solutions Pty Ltd.

HyQuest Solutions (NZ) Ltd.

A strong customer base



Markets & software products



Largeformat Hardware

hardware.

Monitoring

Energy

Water

2D-/3D Viewer

software.

Logistics & Aviation

Air

Environmental Protection & Safety

Environmental Consulting

engineering services

Corporate figures

Key figures	2017
Number of permanent employees	➢ 500
Number of subsidiaries	15
Revenue in million euros	73

Motivation

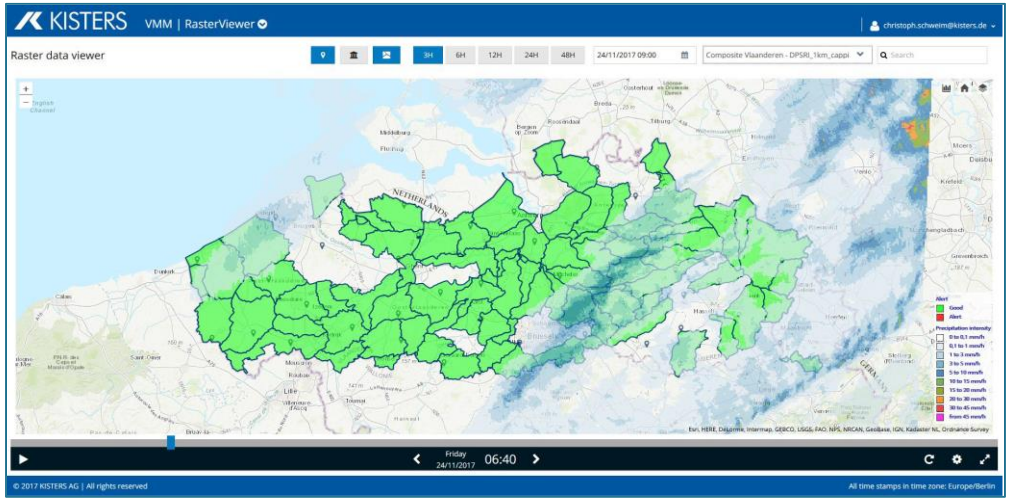
Why we work with Python? From the perspective of a Java developer

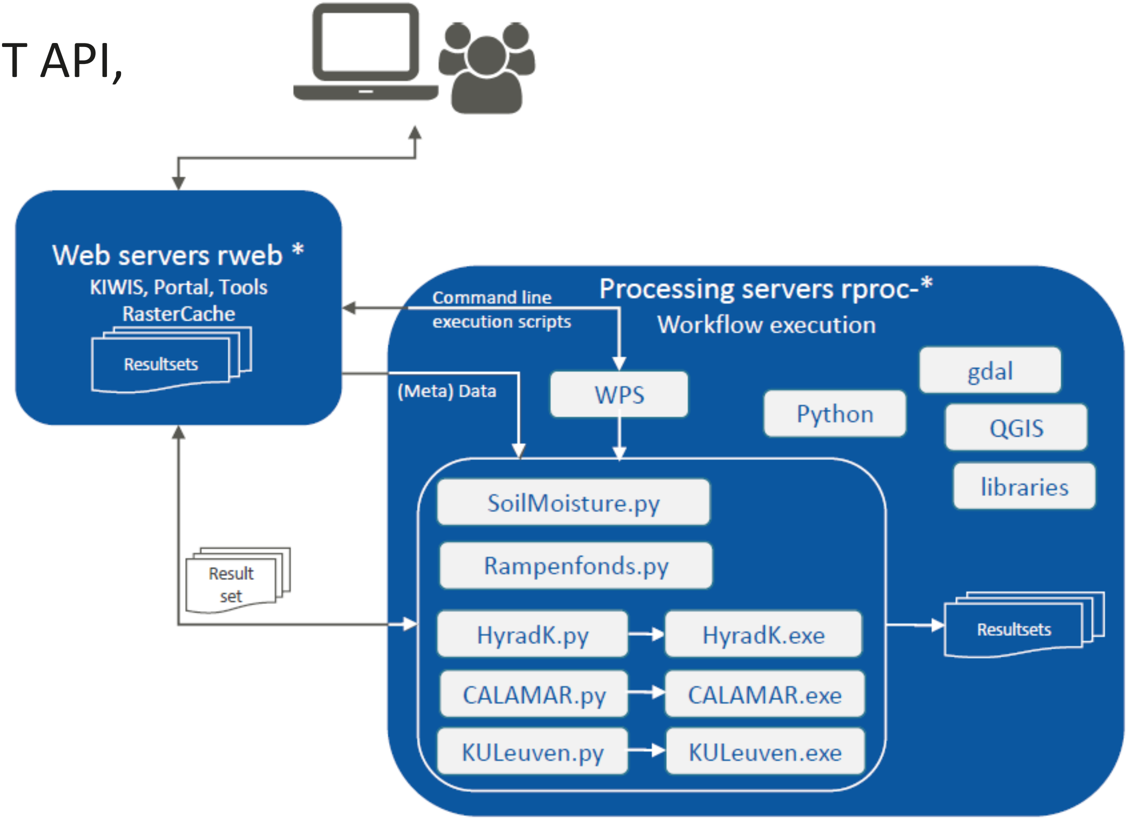
- Traditional server backend development at KISTERS relies on Java for decades
  → developers had a rather critical attitude towards Python, “Just scripting and not a complete programming language!”
- Increasing need for a complementary approach, Python became a hot candidate due to:
  - the availability of tons of libraries for scientific computing, data analytics and visualization, gridded data processing etc.
  - and the fact that it is an actual programming language (plus it is easy to learn).
- New software architecture heavily relies on microservices and well-defined REST APIs or protocols such as GRPC.
  → much easier to integrate additional programming languages.
- Since 2018, we are establishing Python as a main programming language next to Java in particular for data analytics, (custom) data validation and gridded data processing.

Use Cases and Applications

Management of gridded data at VMM, Belgium (in operation from early 2018)

- Traditional on-premise installation for a governmental agency.
- Main focus on radar data and related processing algorithms:
  - Data download and ingest scripts.
  - Integration of third-party algorithms through REST API, in particular Python-wrapped radar calibration.
- Web-based data visualization.


KISTERS
VMM | RasterViewer
christoph.schweim@kisters.de
Raster data viewer
3H
6H
12H
24H
48H
24/11/2017 09:00
Composite Vlaanderen - DPSRI_1km_cappi
Search
Friday 24/11/2017
06:40
© 2017 KISTERS AG | All rights reserved
All time stamps in time zone: Europe/Berlin

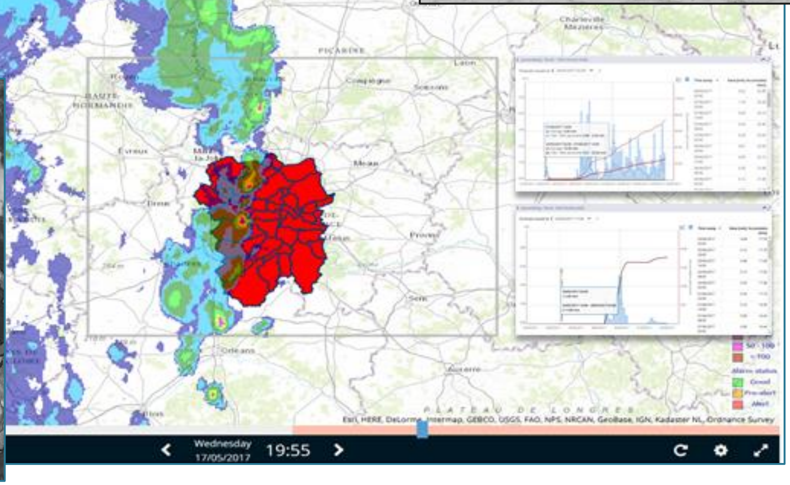


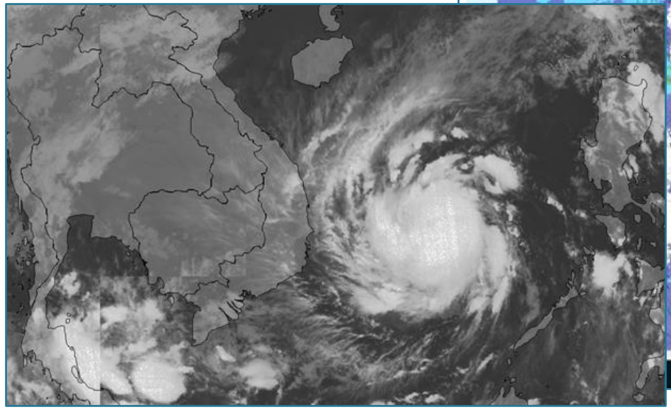
Use Cases and Applications (2)

KISTERS

Central Data Hub and Forecasting Support System for MONRE-VNMHA, Vietnam

- Worldbank-funded project to establish a central datahub and a hydrological and marine forecasting system at the national hydromet agency in Vietnam.
- Daily ingest of about 50-100 GB from gauging networks, local radar and its composites, satellite images as well as several global and regional NWP.
  → R&D to upscale the array storage component to volumes of several 100 TBs, next generation storage solution in Python is currently under development.
- Custom Python coding to:
  - identify the tracks of cyclones e.g. from NWPs.
  - analyze the skills of meteorological and hydrological forecasts.
  - analyze the compliance to business procedures, i.e. in-time publication of forecast bulletin.



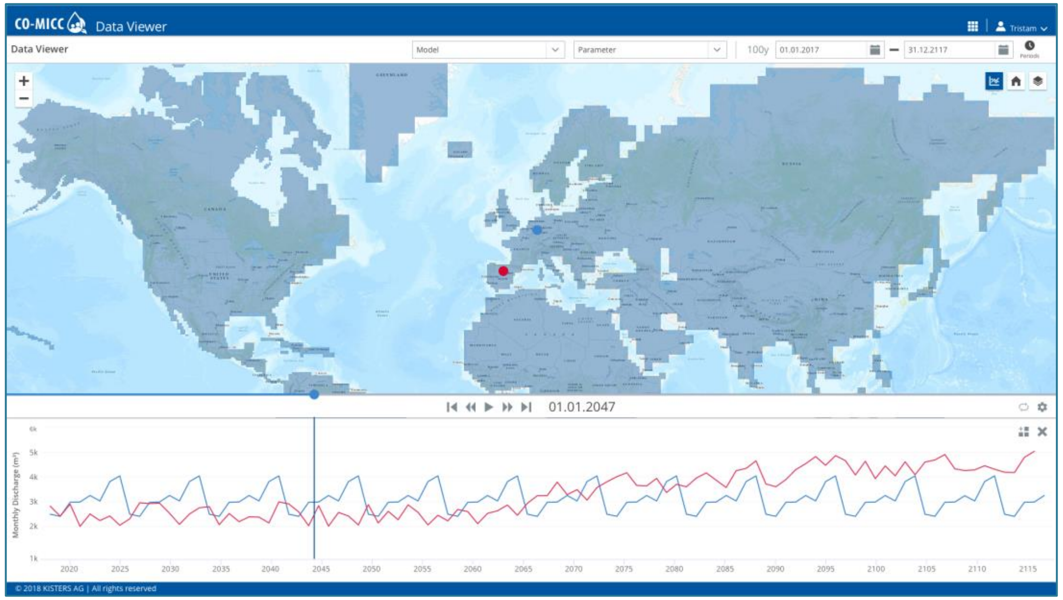


Use Cases and Applications (3)

KISTERS

ERA4CS CO-MICC (https://www.co-micc.eu/), EU research project by Frankfurt University and partners

- Title: Supporting risk assessment and adaptation at multiple spatial scales: Co-development of methods to utilize uncertain multi-model based information on freshwater-related hazards of climate change.
- KISTERS tasks:
  - Cloud-based repository for project data: approximately 30TB of meteorological forcing and hydrological model results.
  - Web portal for spatial visualization of dynamic ensembles of multi-model simulations.
  - Custom analytics for individual spots, river basins and countries.



→ tools heavily rely on Python as regards the storage backend and analytics.

Review of existing tools and design decisions

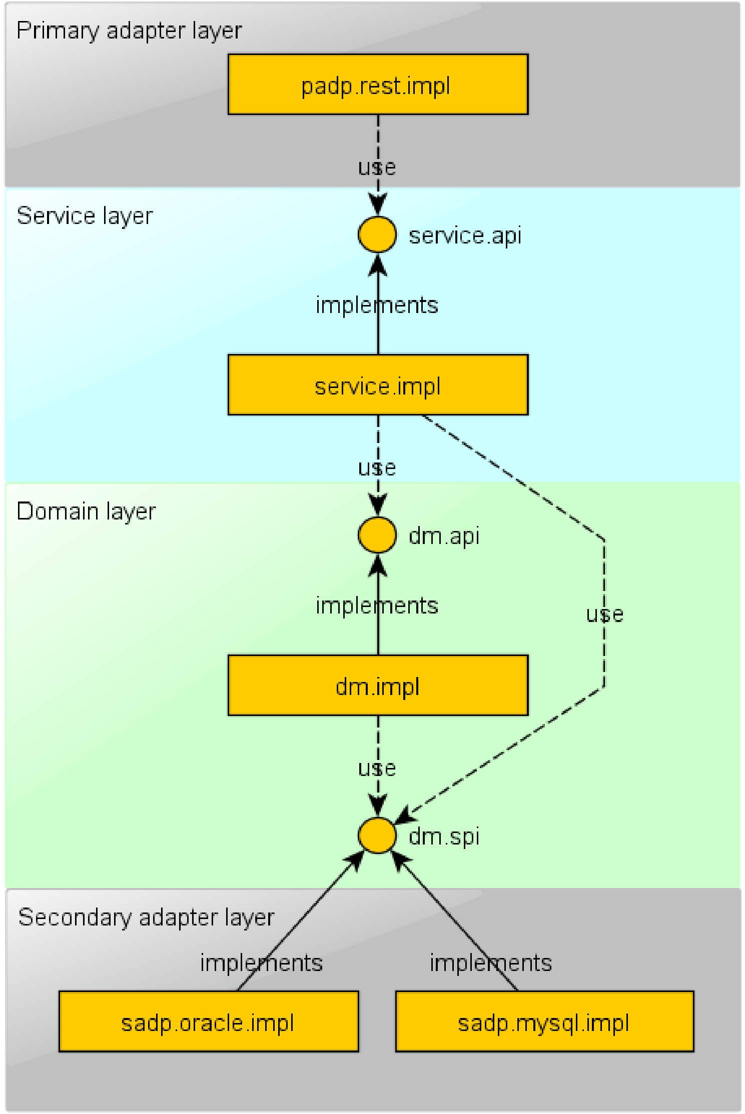
Evaluation of existing technologies and tools

- Evaluation of existing solutions for array storage such as **RASDAMAN, THREDDS, SciDB** etc.
  → either too expensive in enterprise version or only partially fulfilling our requirements.
- **NETCDF4** as the most appropriate format for import/export and the internal storage of data, external support by Franscesc Alted to evaluate several compression schemes, chunk storage options and Python libraries.
- The Python **Zarr** package is an interesting alternative to NETCDF4 for the internal storage: pure Python with transparent storage as key-values pairs.
  → interesting future concept to store data in large, distributed key-value DBs.
- **Hadoop/HBase** as BLOB storage to store chunks.
  → interesting combination with Zarr, but requires more work in particular on bulk IO for better performance.

Review of existing tools and design decisions

Port-adapter/hexagonal architecture for flexible Python integration

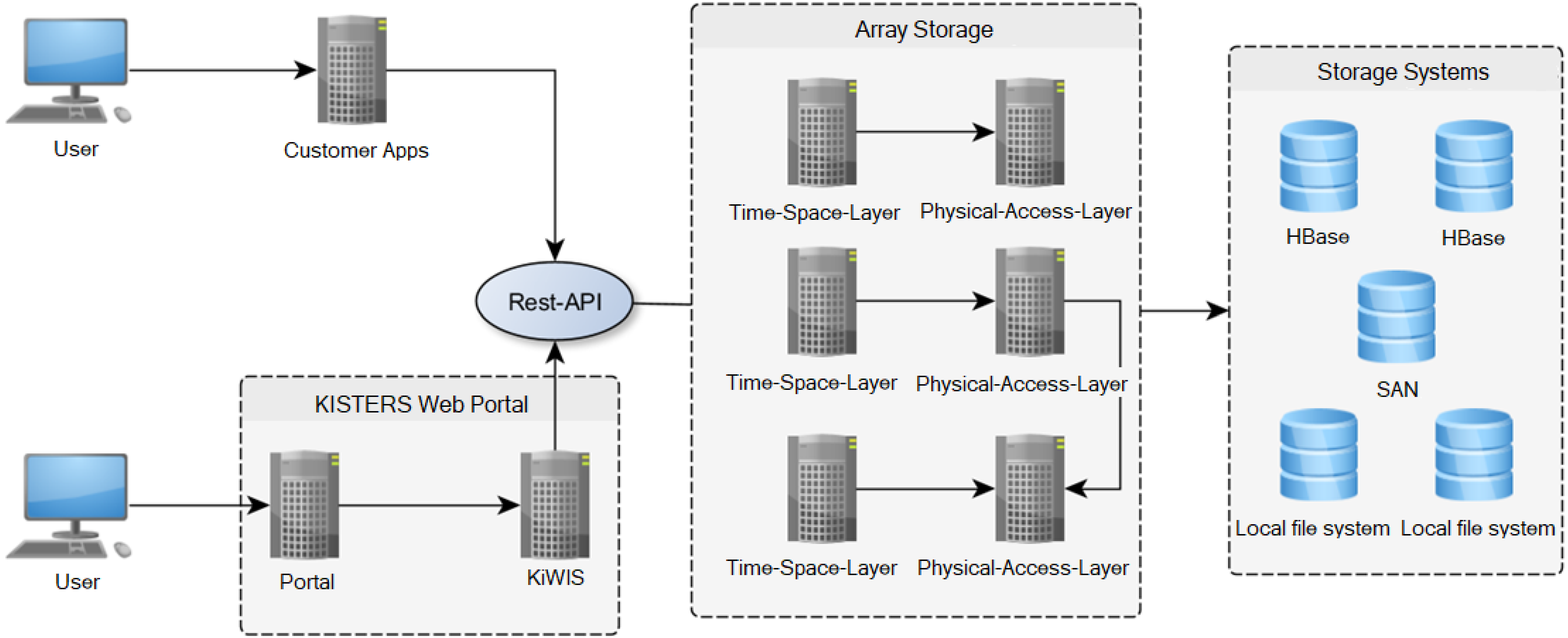
- Based on domain driven development.
- Separate business logic from implementations.
- System is easier to maintain, test, refactor, extend.
- Layers:
  - Domain layer
  - Ports
  - Adapters
  - (Applications layer)



Review of existing tools and design decisions

KISTERS

Design decision: Build a scalable microservices architecture primarily based on Python tools



Review of existing tools and design decisions

Underlying Python packages

- **NetCDF4** for IO and to handle the file system storage in NetCDF format according to the CF convention.
- **Zarr** using cache storage through an HBase adapter.
- **HappyBase** handling the connections to Hbase.
- **GDAL** providing support for raster data in GeoTiff and other formats.
- **Flask-RESTPlus** building REST APIs.
- **Celery** to coordinate asynchronous, scalable write actions.

Conclusions and Outlook

... after about a year of significant investments into Python ...

- We will use the new Array Storage component in production in a number of projects until the end of 2018.
- We expect a higher engagement of KISTERS in the open source community and some contribution to the Python packages indicated before.
- KISTERS started to publish first components as open source software:
  → https://kisterswatertime-series.readthedocs.io/en/latest/
  → coordination of our user community as regards Python IO and analytics

We actively look for collaboration partners in the academic, research and governmental domain.

KISTERS AG

Pascalstraße 8+10
D-52076 Aachen

Phone +49 2408 9385-0
Fax +49 2408 9385-555
info@kisters.de

www.kisters.de

File name:	2018-10-31_PythonEarthScience_Kisters.pptx
Creation date:	2018-09-17
Presentation date:	2018-10-31
Author:	Alberto Sabater Morales, Jackie Leng
Speaker:	Alberto Sabater Morales, Jackie Leng