



# DiversityNaviKey & RDC Infrastructures

Bernhard Seeger  
Professor für Praktische Informatik  
Philipps-Universität Marburg



## Where are we coming from?

- Biodiversity and Computer Science
  - NFDI4Biodiversity
  - DFG infrastructure project GFBio
  - BMBF infrastructure project de.NBI
- Obvious Dependencies to other Disciplines
  - Medicine
  - Climate change
    - extinction of species
  - Economical impacts
    - manual pollination



## Why one NFDI?

- **Research results are not only relevant to one discipline!**
  - Strengthening inter- and transdisciplinary research
  - Multidisciplinary research
  - Social relevance
  
- **Goals**
  - Sharing of research data and services (not only within NFDI)
  - Provisioning of collaborative tools

**⇒ Need for a common and open infrastructure**

# Service Requirements

- FAIR principles
  - unique identifiers
  - metadata description
  - common access
  - ubiquitous versioning
  
- This applies to all research objects!
  - data
  - services
  - ...

Mons, Barend, et al. "Cloudy, increasingly FAIR; revisiting the FAIR Data guiding principles for the European Open Science Cloud." *Information Services & Use* 37.1 (2017): 49-56.

# Requirements on a Common Compute Infrastructure

- **Robust and failure-tolerant**
  - always available
  - short service response regardless of the data volume
  - standardized components
- **Low cost**
  - management and maintenance
  - elastic sharing of resources
- **Trust**
  - user management
  - access rights
  - logging

## Summary of our Objectives

- Improve FAIRness of (Biodiversity) Research Objects  
⇒ Development of a common infrastructure RDC  
(Research Data Commons)
- Technical Backbone of the RDC  
⇒ Multi-cloud infrastructure

## Use-Case: Dragonflies

- an example for participation of citizen science in NFDI4Biodiversity
- status of dragonflies in Europe
  - 135 species, 15% are threatened
- current situation in Germany
  - monitoring by citizen scientists organized in GdO e.V.
  - no commonly accepted metadata
  - no national data portal
  - difficult to reuse monitoring data in recent studies



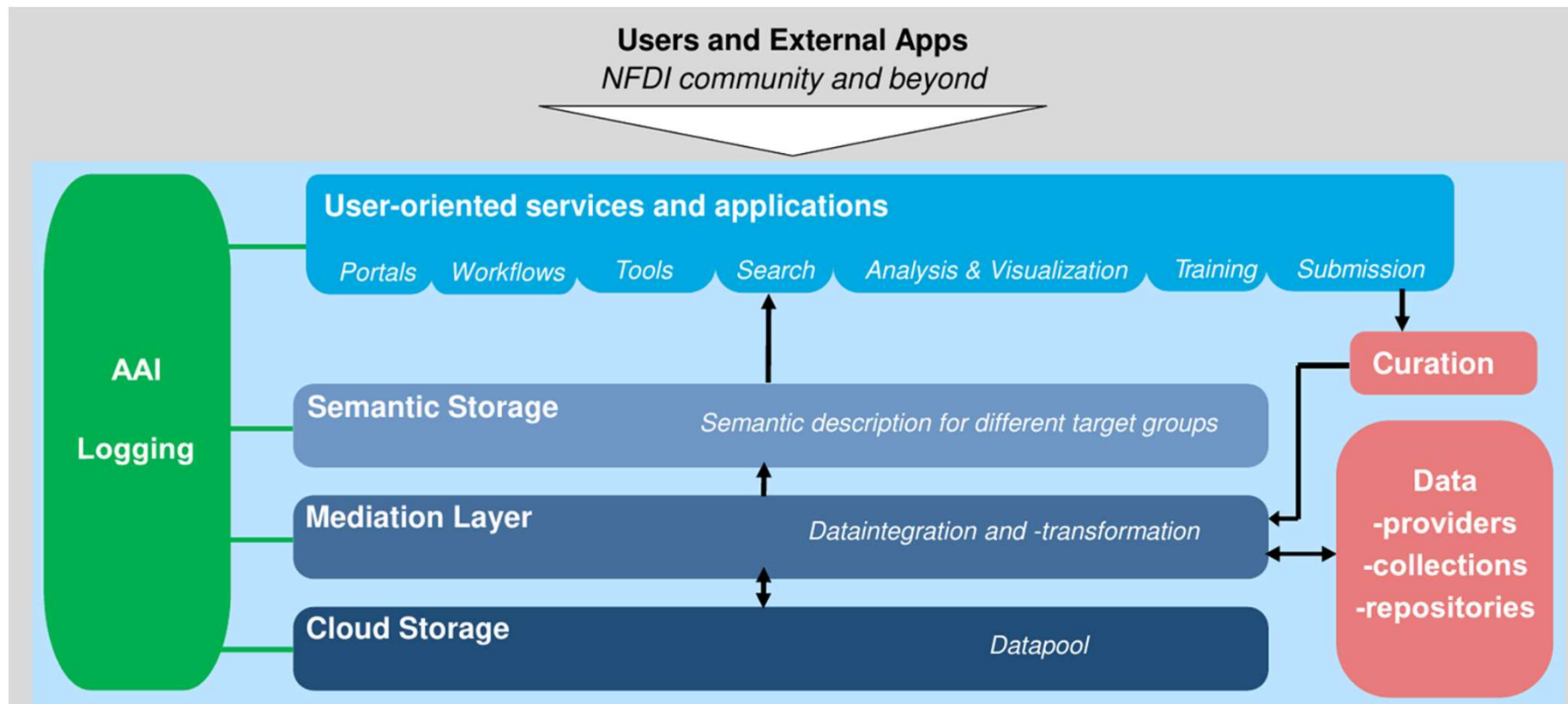
André Karwath - Eigenes Werk, CC BY-SA 2.5, <https://commons.wikimedia.org/w/index.php?curid=226067>



Termaat, Tim, et al. "Distribution trends of European dragonflies under climate change." *Diversity and Distributions* 25.6 (2019): 936-950.

Sahlén, Göran, et al. *European red list of dragonflies*. Publications Office of the European Union, 2010.

# The Research Data Commons (RDC)



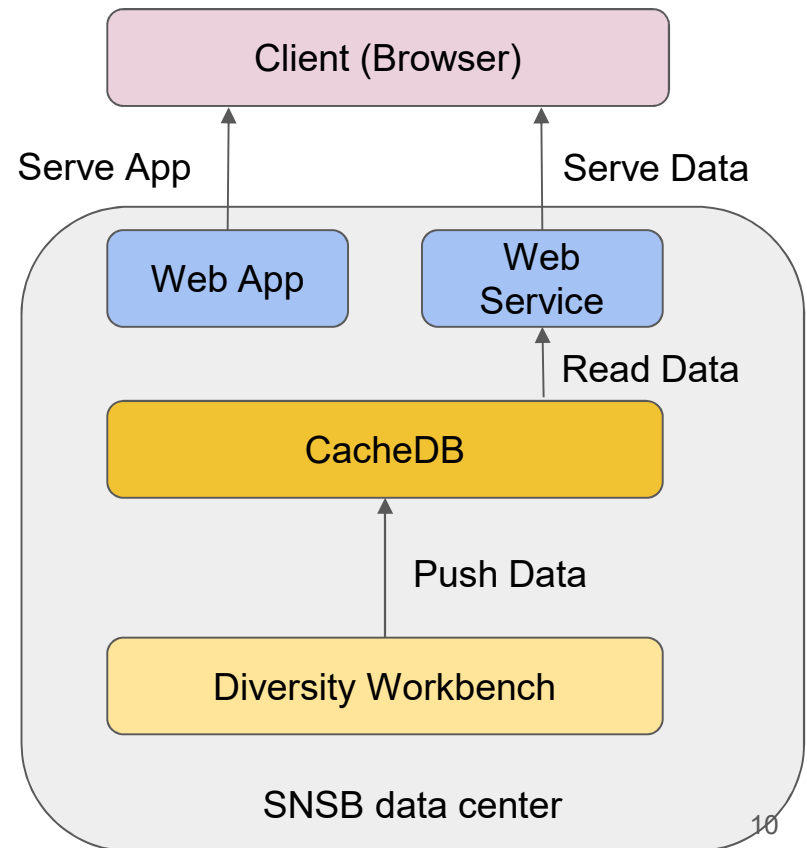


# The Cloud Storage Layer

- Common low-level API to store and organize data objects and metadata
  - HTTP calls (REST interface) to store and retrieve data
  - Parallel access on large numbers of objects
  - Grouping of objects for better organization
  - Versioning of objects to support reproducibility
- Integration of the common AAI solution (OIDC based)
- All components are designed to be scalable and support billions of objects
  - Database
  - Storage backend
  - Backend
- Optional JSON based metadata search component

# Use Case DiversityNaviKey - Current State

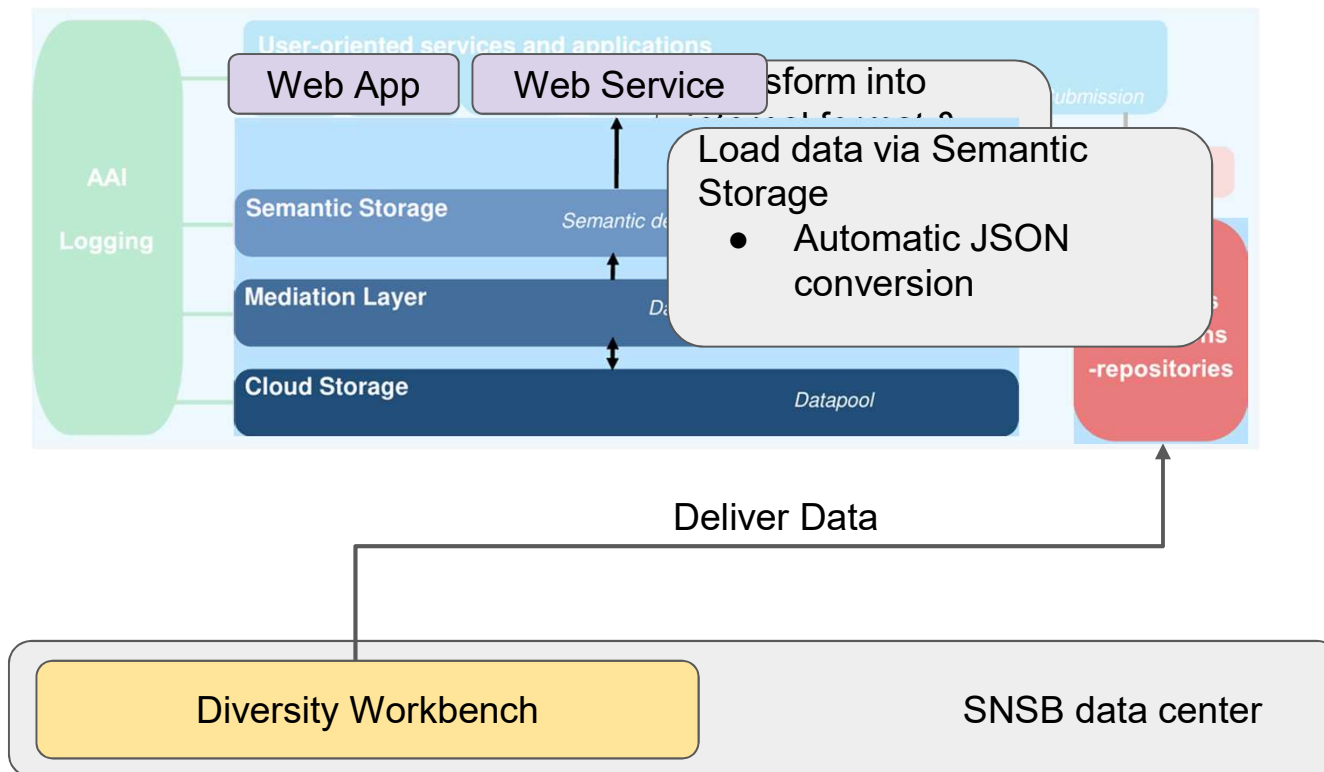
- Progressive WebApp backed by two Endpoints
  - Web App - Deliver HTML/CSS/JS
  - Web Service - Deliver (mostly) static data
- Data service by CacheDB (PostgreSQL)
  - Flat data format
  - No filtering
- CacheDB regularly updated by Diversity Workbench
- Hosted at SNSB data center



# Use Case DiversityNaviKey - NFDI Integration

- Substitution of CacheDB
  - Replacement with flat files stored in CORE-Storage of RDC
    - Optimized internal format
  - Periodically updates from Diversity Workbench
    - Data delivered to NFDI's ETL / ELT component
- Web App and Web Service
  - Implementation as container in the service layer of RDC
  - Web Service accesses data via Semantic Storage
    - Automatically converts internal format into JSON understood by the client app

# Use Case DiversityNaviKey - NFDI Integration



## Summary & Discussion

- First approach to integrating DiversityNaviKey into RDC

### Advantages

- Versioning, AAI & Scalability
- Dockerizing software via Kubernetes