Scientific Data Management in Grid Environments using DataFinder

5th Black Forest Grid Workshop (April 23th, 2009, Freiburg)

Anastasia Eifer

German Aerospace Center (DLR), Simulation and Software Technology

http://www.dlr.de/sc

The DLR
German Aerospace Research Center
Space Agency of the Federal Republic of Germany

Sites and employees

5,800 employees working in 29 research institutes and facilities

- at 13 sites.

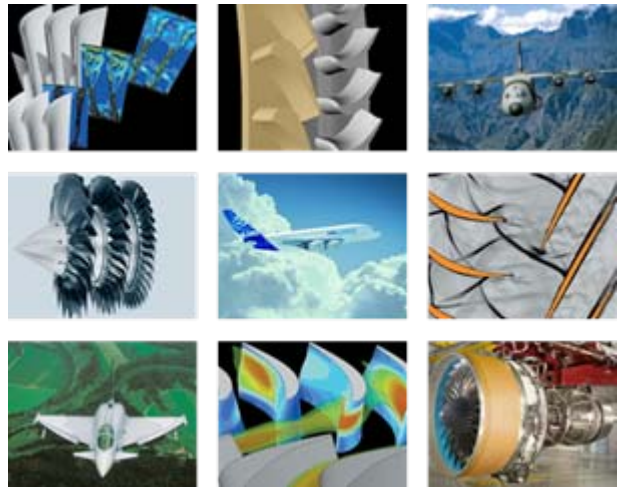
Offices in Brussels, Paris and Washington.

AeroGrid Project

AERO GRID

AeroGrid
Project Data

Grid-based cooperation between industry, research centres, and universities in aerospace engineering



Runtime: April 1, 2007 – March 30, 2010

Website: `http://www.aero-grid.de`

Project Partner

German Aerospace Center (DLR)

- Institute for Propulsion Technology
- Simulation and Software Technology (*Coord.*)

MTU Aero Engines GmbH

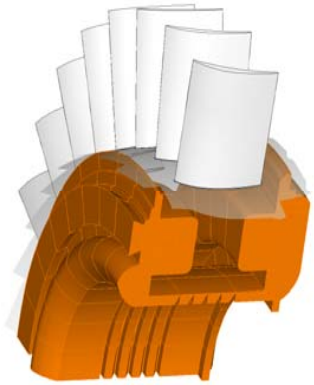
T-Systems Solutions for Research GmbH

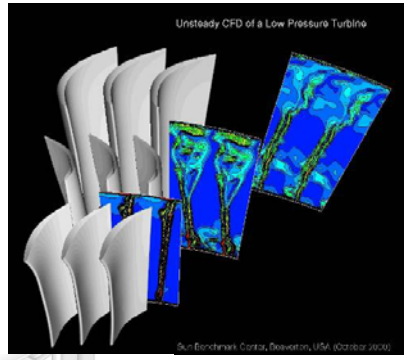
University of the Armed Forces, Munich

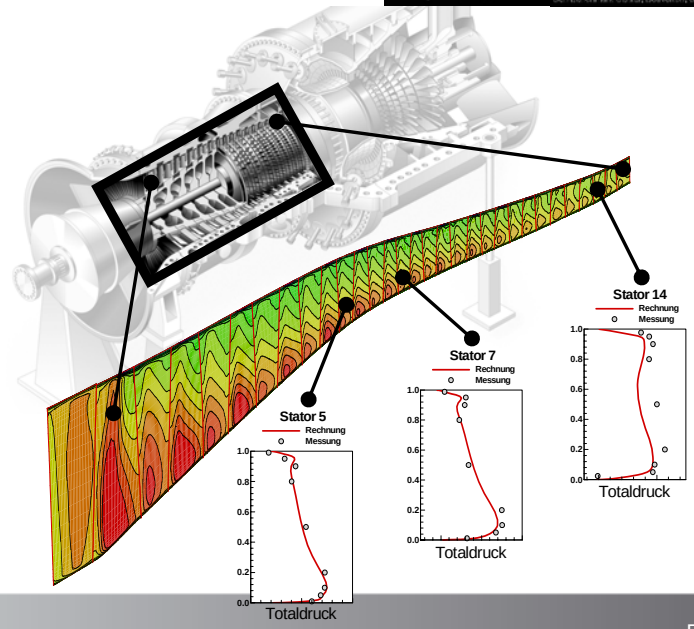
- Institute for Jet Propulsion

Background: Turbo Machinery
Simulation Tasks

- **Simulation of turbine component**
  - Design (*variants*)
  - Optimization
  - Aero elasticity
  - Aero acoustics
  - Cooling
  - Complex geometries
  - Multistage components
- **Use of the CFD-Code TRACE**
  (Institute of Propulsion Technology)



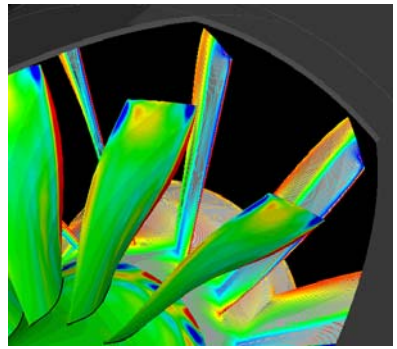




AeroGrid
Use Cases and Project Goals

Usage scenarios

- Use of computing resources via the AeroGrid
- Collaboration in designing engine components
- Co-operative further development of TRACE code

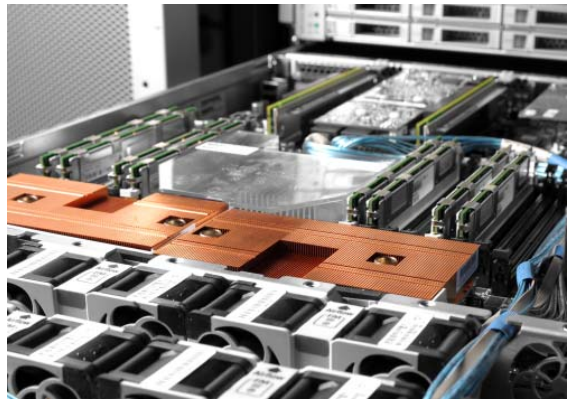


Project goals

- Allow cooperation in research and development projects
- Use of up-to-date program versions, data, and compute resources across all locations
- Detailed documentation of history of a computational process that leads to a certain result ("**Provenance**")

AeroGrid Cluster

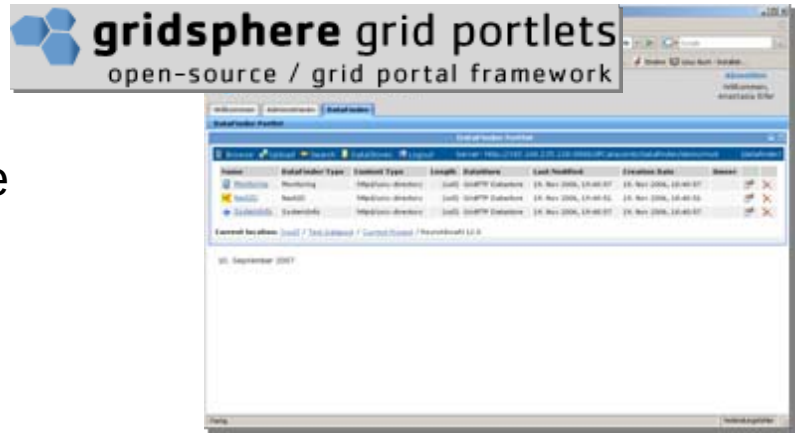
- 45 Compute nodes with 2 quadcore processors (total 360 cores)
  - Compute node: dual Intel(R) XEON quad core
  - Processortype: Intel(R) Xeon 5440 2.8 GHz
  - Main memory: 45 x 16 GB (aggregate 720 GB)
  - Network: InfiniBand + Gigabit Ethernet
  - Operating system: SLES 10
  - Operating mode: batch (TORQUE)
  - Grid middleware: UNICORE 6
- Filespace: GPFS (total 2 TByte)



AeroGrid User Interfaces

Portal

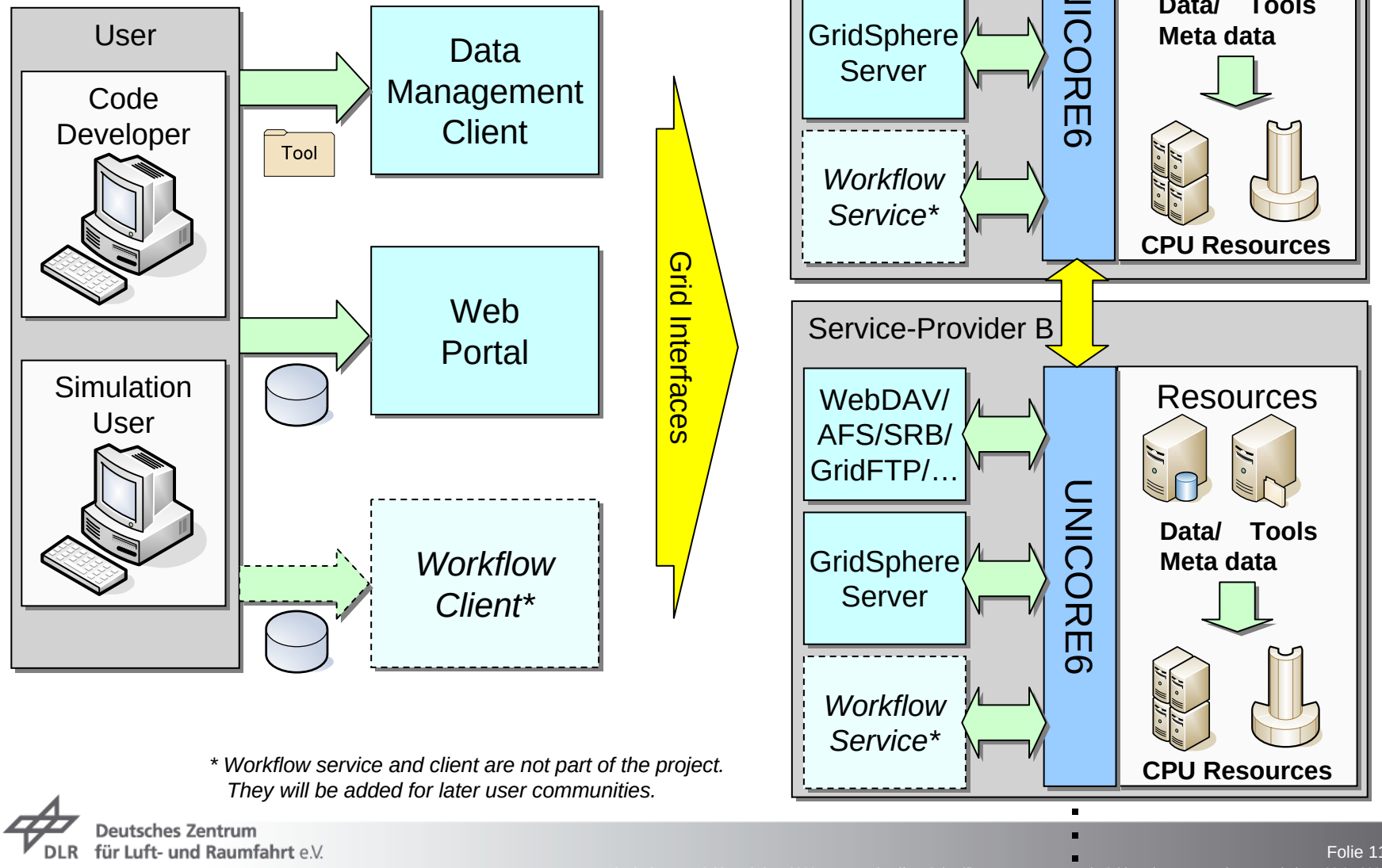
- Web-based access
- Development based on GridSphere



Client applications

- Automation of recurring tasks
- Integration in existing working environments

Cooperation in AeroGrid



** Workflow service and client are not part of the project. They will be added for later user communities.*

DataFinder

Data Management Client Software
DataFinder

Introduction
Data Management Problem

Typical organizational situations

- No central data management policy
- Every employee organizes his/her data individually
- Researchers spend about 30% of their time searching for data
- Problem with data left behind by temporary staff

Increase of data size and regulations

- Rapidly growing volume of simulation and experimental data
- Legal requirements for long-term availability of data (up to 50 years!)

Situation similar at many organizations

- All ~30 DLR institutes
- Other research labs and agencies
- Industry

DataFinder
Short Overview

DataFinder

- Efficient management of scientific and technical data
- Focus on huge data sets

Development of the DataFinder by DLR

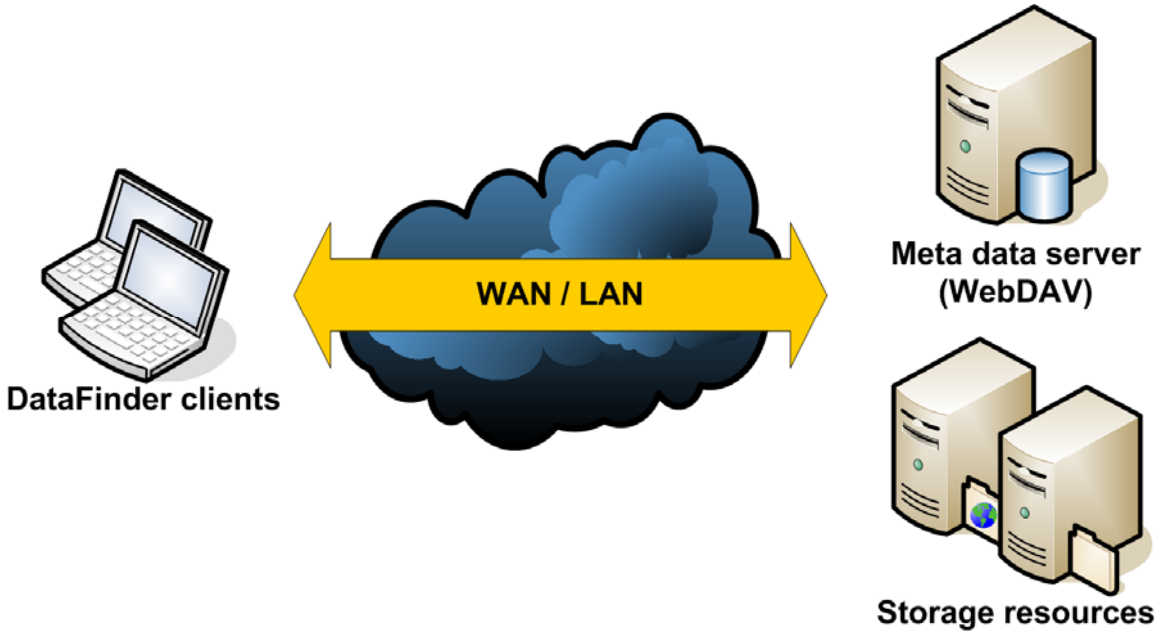
- Available as Open-Source-Software

Primary functionality

- Structuring of data through assignment of meta information and self-defined data models
- Complex search mechanism to find data
- Flexible usage of heterogeneous storage resources
- Integration in the working environment

DataFinder Overview
Basic Concept

- **Client-Server solution**
- Based on **open and stable standards**, such as XML and WebDAV
- Extensive use of standard software components (open source / commercial), **limited own development** at client side



DataFinder Overview
Client and Server

Client

- User client
- Administrator client
- Implementation: Python with Qt

Server

- **WebDAV server** for meta data and data structure
- **Data Store** concept
  - Abstracts access to managed data
  - Flexible usage of heterogeneous storage resources
- Implementation: Various **existing server solutions** (third-party)

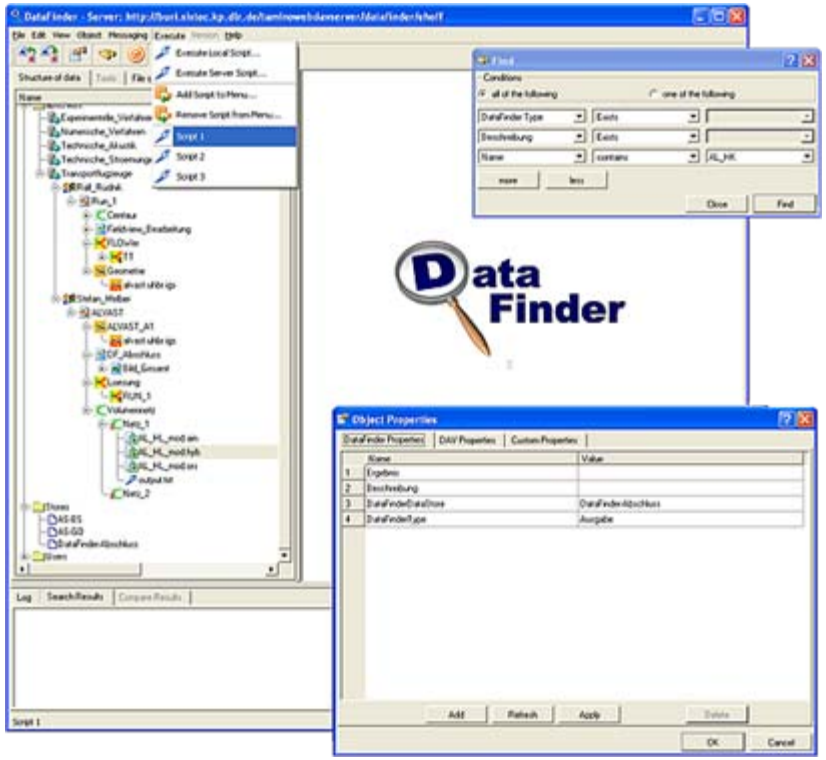
WebDAV
Web-based Distributed Authoring & Versioning

- Extension of HTTP
- Allows to manage files on remote servers collaboratively
- WebDAV supports
  - Resources (“files”)
  - Collections (“directories”)
  - Properties (“meta data”, in XML format)
  - Locking
- WebDAV extensions
  - Versioning (DeltaV)
  - Access control (ACP)
  - Search (DASL)

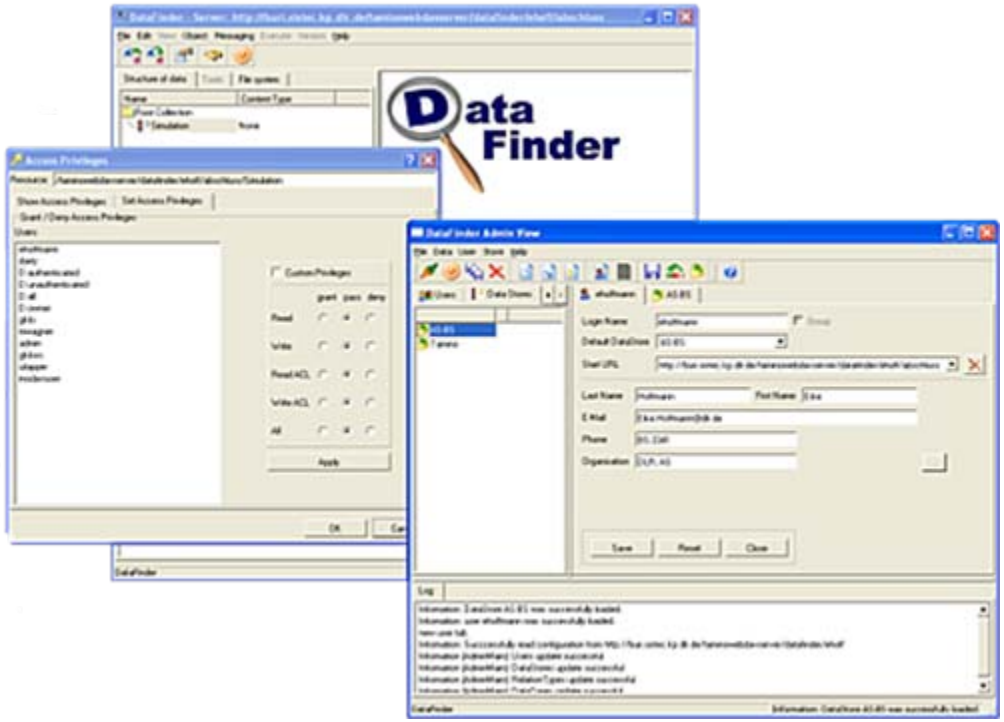
DataFinder Client

Graphical User Interfaces

User Client



Administrator Client



Implementation in Python with Qt/PyQt

DataFinder Server
Supported WebDAV servers

Commercial Server Solution

- Tamino XML database (Software AG)

Open Source Server Solutions

- Apache HTTP Web server and module mod_dav
  - Default storage: file system (mod_dav_fs)
  - Module Catacomb (mod_dav_repos) + Relational database (**http://catacomb.tigris.org**)

Configuration and Customization

Preparing DataFinder for certain “use cases”

Requirements Analysis

- Analyze data, working environment, and users workflows

Configuration

- Define and configure data model
- Configure distributed storage resources (Data Stores)

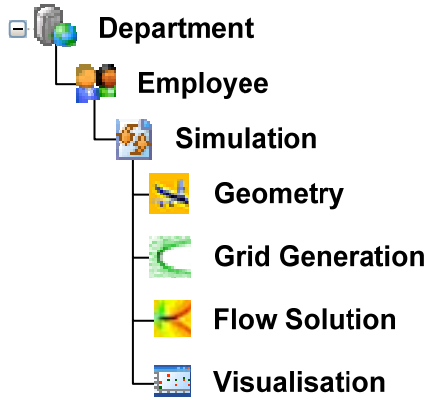
Customization

- Write functional extensions with Python scripts

DataFinder Configuration
Data Model and Data Stores

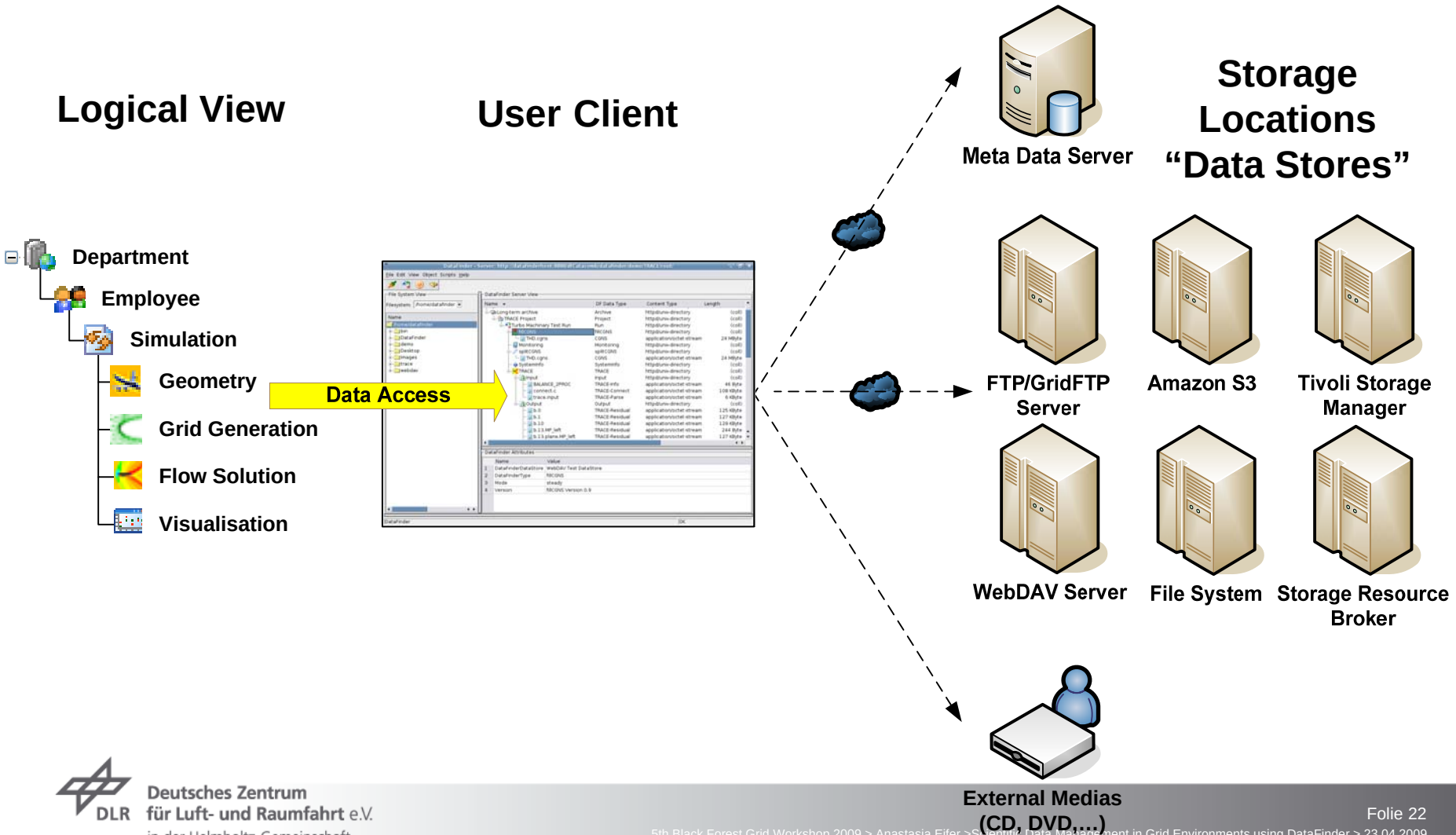
Logical view to data

- Definition of data structuring and meta data (“data model”)
- Separated storage of data structure / meta data and actual data files
- Flexible use of (distributed) storage resources
  - File system, WebDAV, FTP, GridFTP
  - Amazon S3 (Simple Storage Service)
  - Tivoli Storage Manager (TSM)
  - Storage Resource Broker (SRB)
- Complex search mechanism to find data



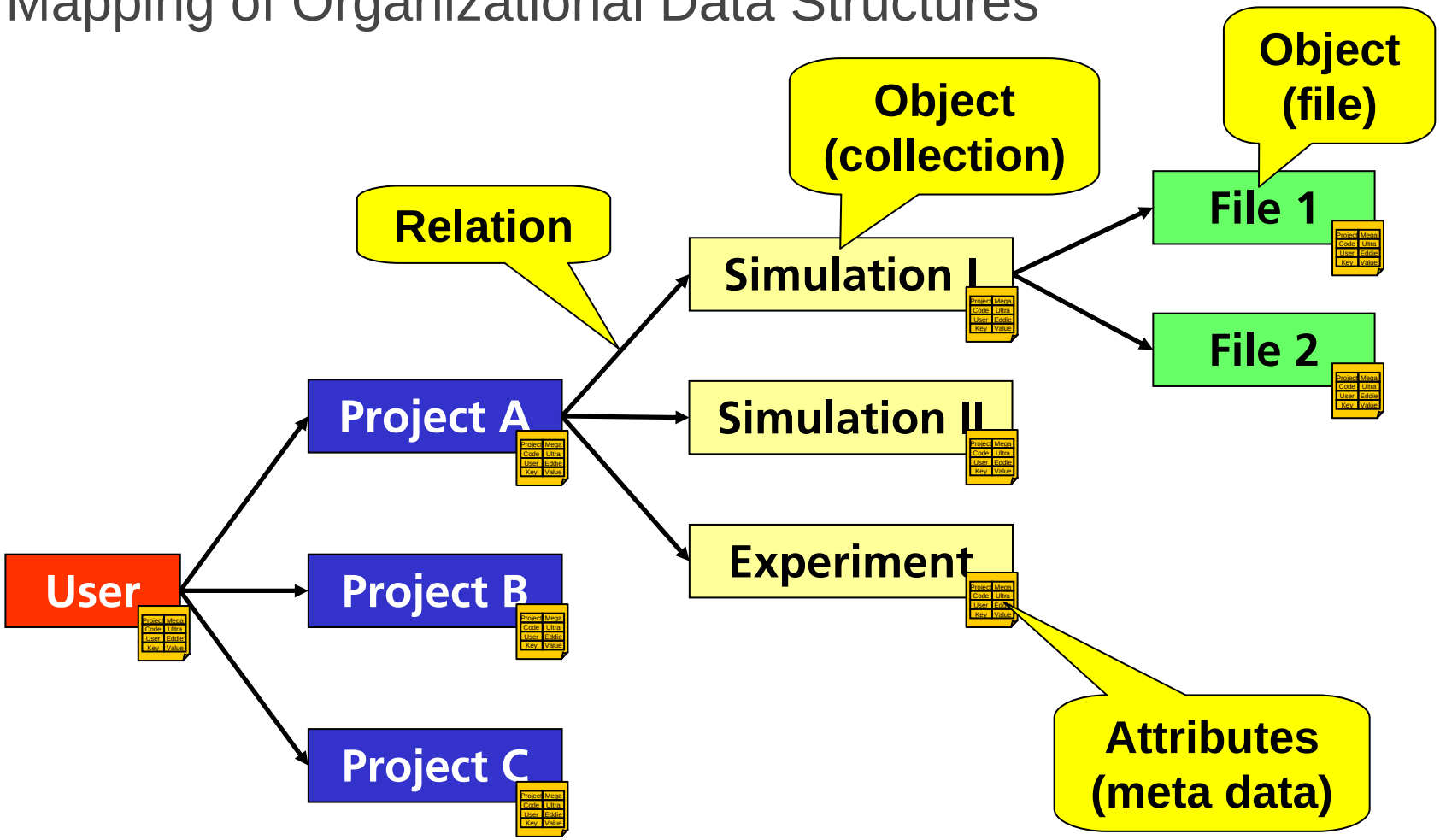
DataFinder

Mass Data Storage using “Data Stores“



Data Structure

Mapping of Organizational Data Structures



Customization
Python-Scripting for Extension and Automation

Integration of DataFinder with environment

- User, infrastructure, software, …

Extension of DataFinder by Python scripts

- Actions for resources (i.e., files, directories)
- User interface extensions

Typical automations and customizations

- Data migration and data import
- Start of external application (with downloaded data files)
- Extraction of meta data from result files
- Automation of recurring tasks (“workflows”)

DataFinder Grid Integration

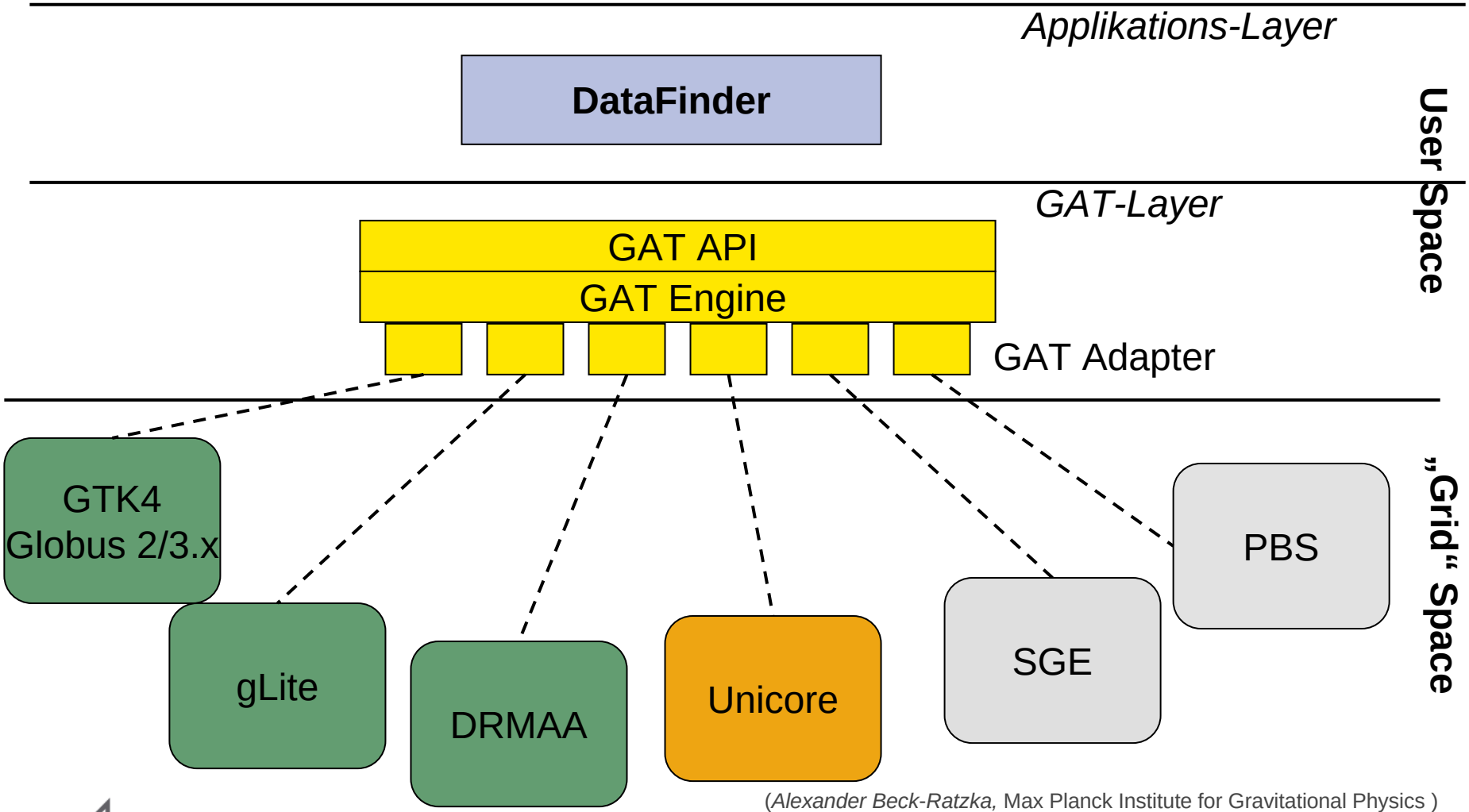
GAT (Grid Application Toolkit)

- provides a simple API to several grid applications
- developed during the Gridlab project
  - mainly developed at the Albert-Einstein-Institute / Max-Plank-Institute for Gravitational Physics and
  - at the Center for Computation and Technology at the Louisiana State University

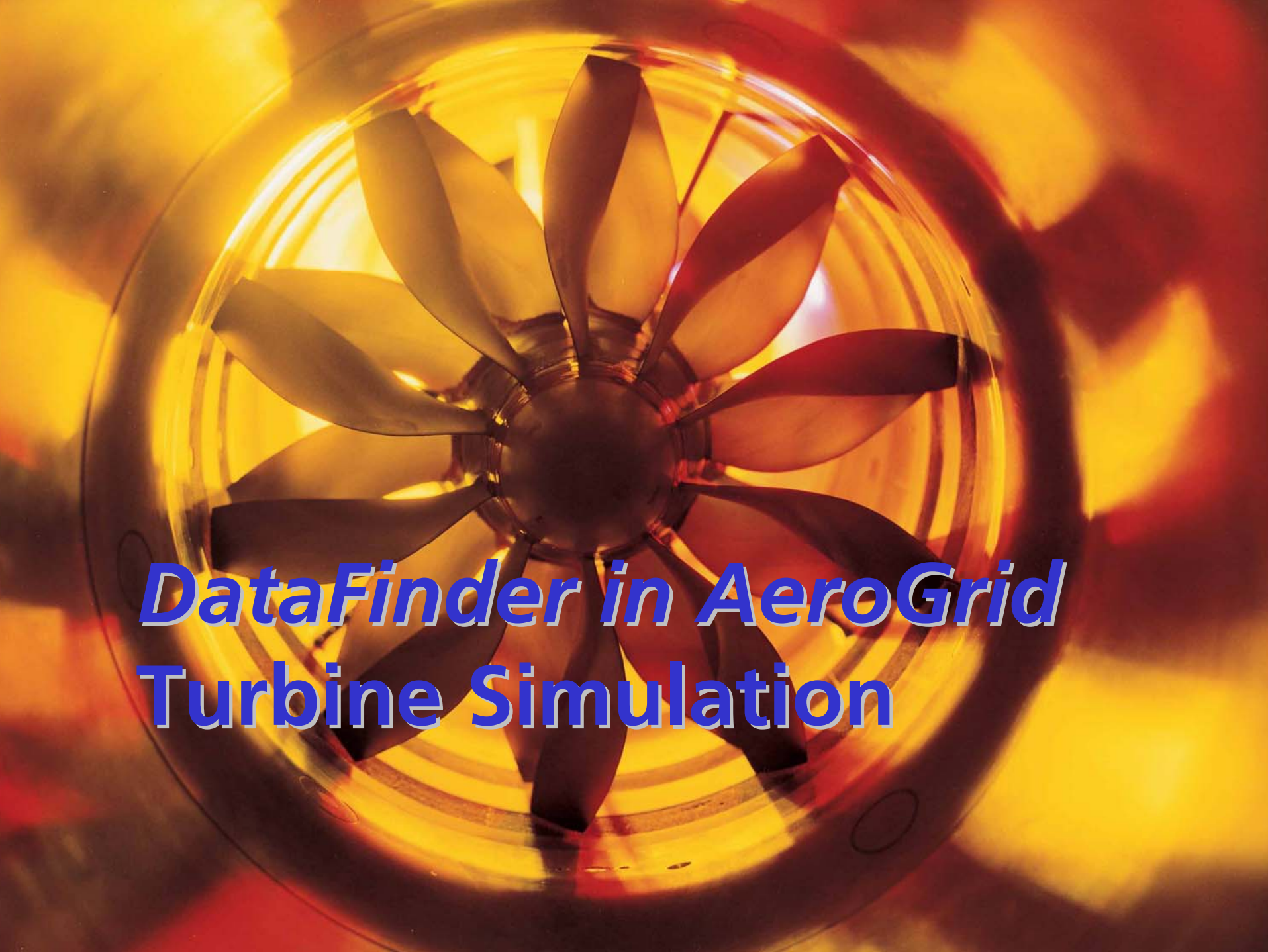
SAGA working group (Simple API for Grid Applications)

- treats GAT as a prototype Implementation
- the goal of SAGA is to provide a new standardized API for grid applications

DataFinder Grid Integration: JavaGAT

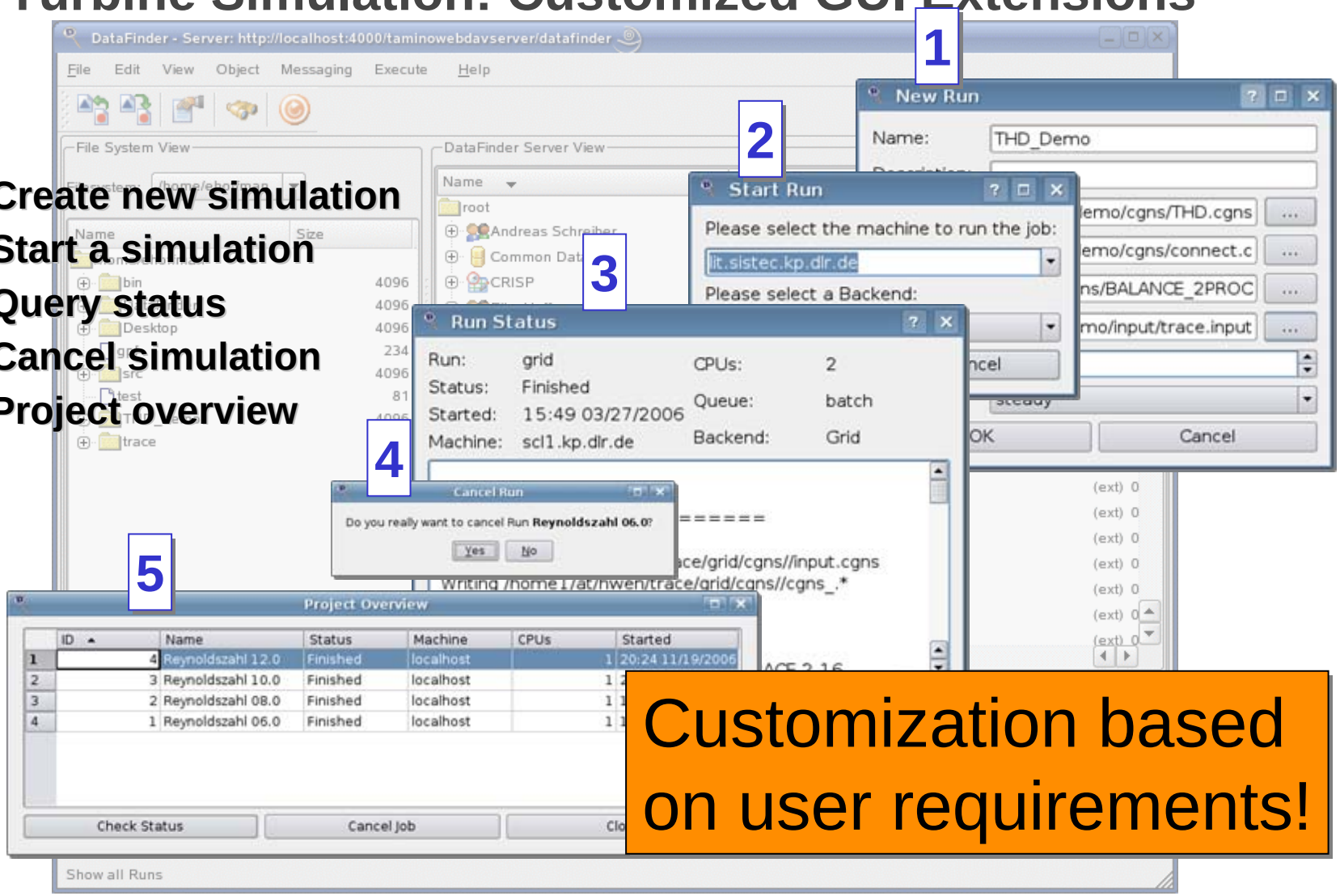


(*Alexander Beck-Ratzka,* Max Planck Institute for Gravitational Physics)


DataFinder in AeroGrid
Turbine Simulation

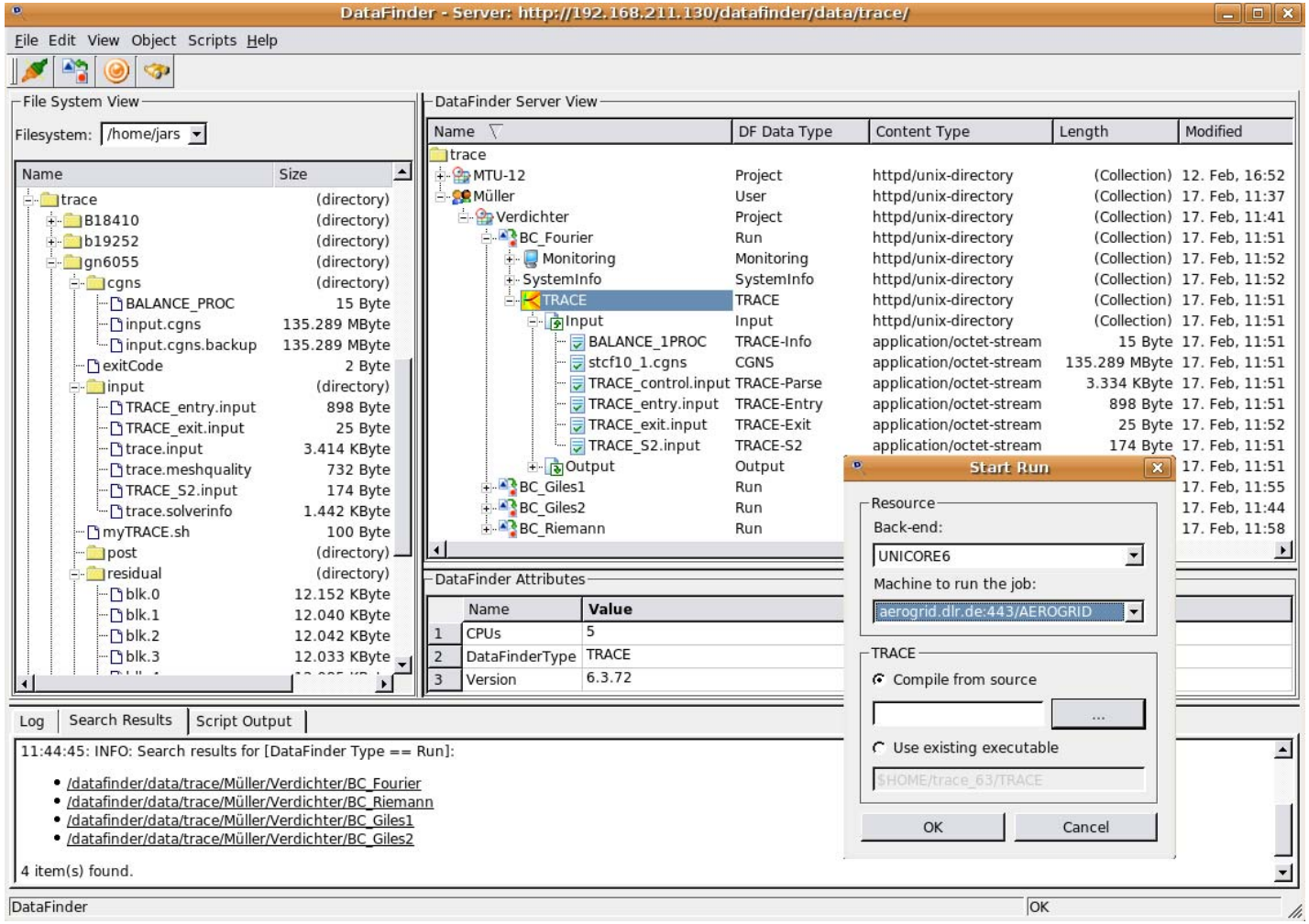
Turbine Simulation: Customized GUI Extensions

1. Create new simulation
2. Start a simulation
3. Query status
4. Cancel simulation
5. Project overview



Customization based on user requirements!

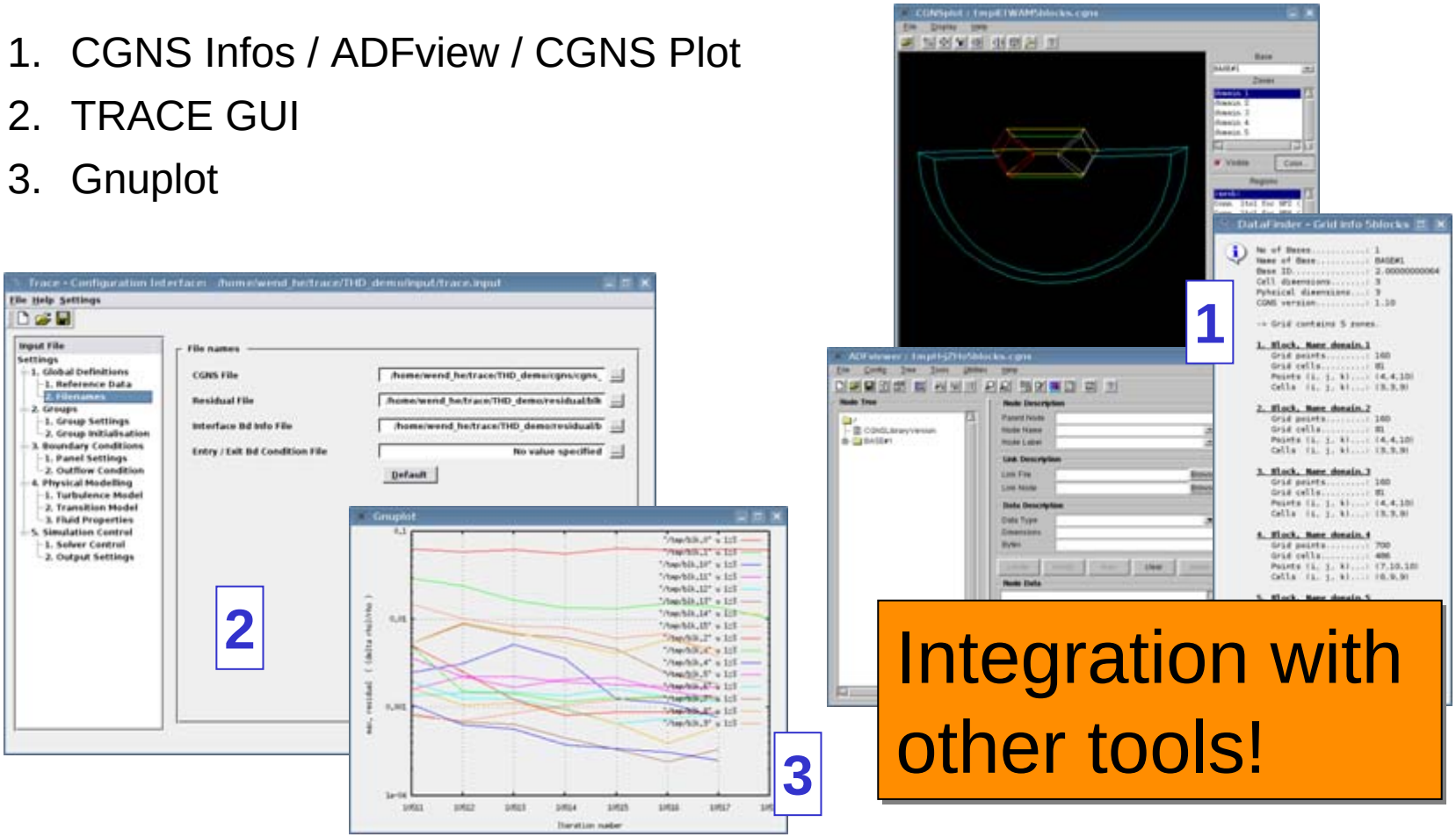
Turbine Simulation: Graphical User Interface



Turbine Simulation

Starting External Applications

1. CGNS Infos / ADFview / CGNS Plot
2. TRACE GUI
3. Gnuplot



Availability

- DataFinder core available as Open Source
  - BSD License
  - **http://sourceforge.net/projects/datafinder**
- Extended versions / extensions are proprietary

Links

DataFinder Web site

➜ **http://www.dlr.de/datafinder**

DataFinder Open Source

➜ **http://sourceforge.net/projects/datafinder**

Python WebDAV library

➜ **http://sourceforge.net/projects/pythonwebdavlib**

Catacomb

➜ **http://catacomb.tigris.org**

AeroGrid Project

➜ **http://www.aero-grid.de**

Thank you for your attention!

- **Contact:**

Anastasia Eifer

DLR Simulation and Software Technology, Cologne

Email: Anastasia.Eifer@dlr.de