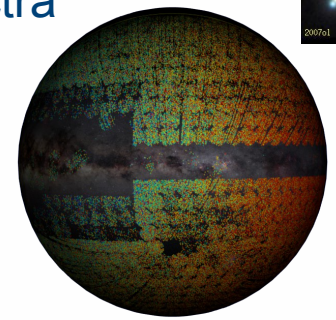
Database and VO developments at AIP, Potsdam

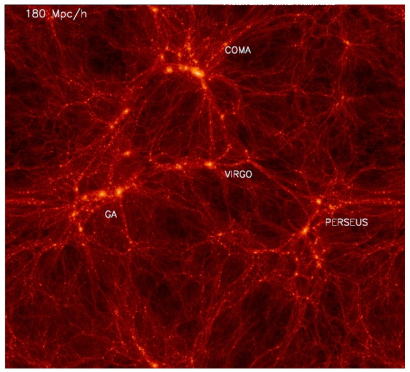
Kristin Riebe

E-Science group @
Leibniz-Institute for Astrophysics Potsdam

Leibniz-Institut für
Astrophysik Potsdam

Example data types at AIP



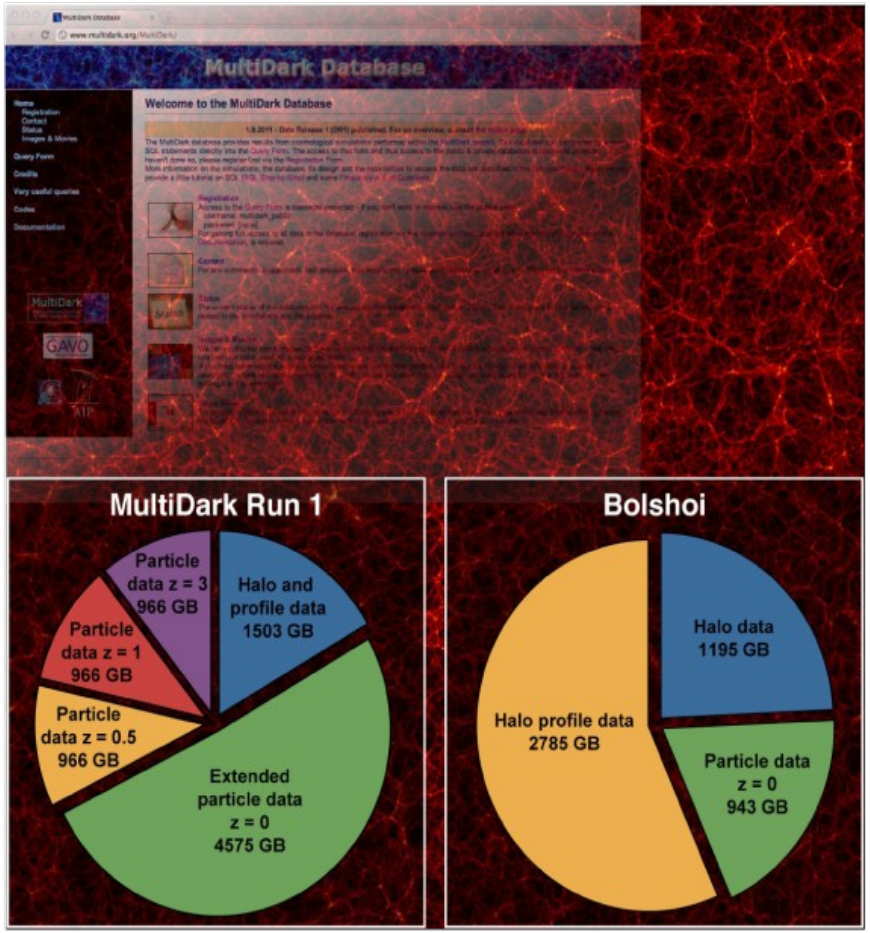


- Observations:
  - RAVE
    - Radial velocity measurements + spectra
  - SDSS
    - Mirror of DR7, catalog server
  - „minor data sets“:
    - Plate archive (historical plates)
    - CALIFA (spectra of galaxies)
    - Cepheids (collection of data for time series), ...
- Simulation data:
  - Magnetohydrodynamics
  - Cosmological simulations: particle data, dark matter halo catalogues, halo merger history, ...

Example: MultiDark Database

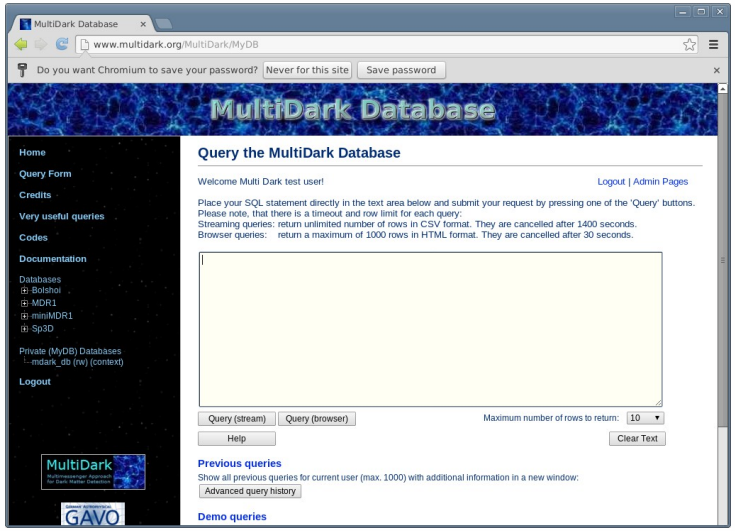
- Collaboration with Spanish MultiDark project
- cosmological simulations in a database
- 3 simulations uploaded (20 TB, 2.5 10^{11} rows)

- > 150 registered users
  > 1 million queries in 3 years
  > 4 TB downloaded



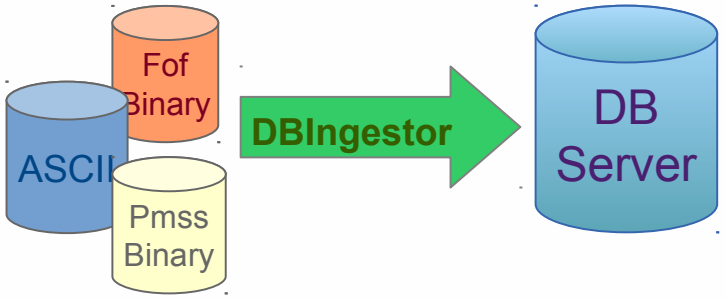
Example workflow: MultiDark Database

- Extract:
  - Cosmologists produce data, copy them to a server at AIP
- Transform:
  - We check data and reading routines, data curation, convert format
- Load:
  - Ingest data into database
- Check and test:
  - Check the data for completeness, consistency
  - Create Peano-Hilbert keys, indexes *(Spatial3D, T. Budavari, G. Lemson)*
- Publish:
  - Using simpledb (Gerard Lemson, Millennium DB)
  - Write/update documentation; update admin tables of the database
  - Inform users



Upload: DBIngestor

- Uploading different formats required tailor-made solutions
  - slow, if conversion to ASCII needed, data curation on DB
- Solution: DBIngestor library
  - Adrian Partl, https://github.com/adrpar/DBIngestor
  - adjustable to any database server
  - easy to write own file readers (AsciiIngest, FofIngest, PmssIngest)
  - apply converters during ingestion
    - e.g. unit conversion,
      type conversion (int/real),
      adding identifiers, grid indexes
  - apply asserters (not nan, inf, null etc.)
  - => transform and upload in one go
  - => easier to preserve the workflow for later reference

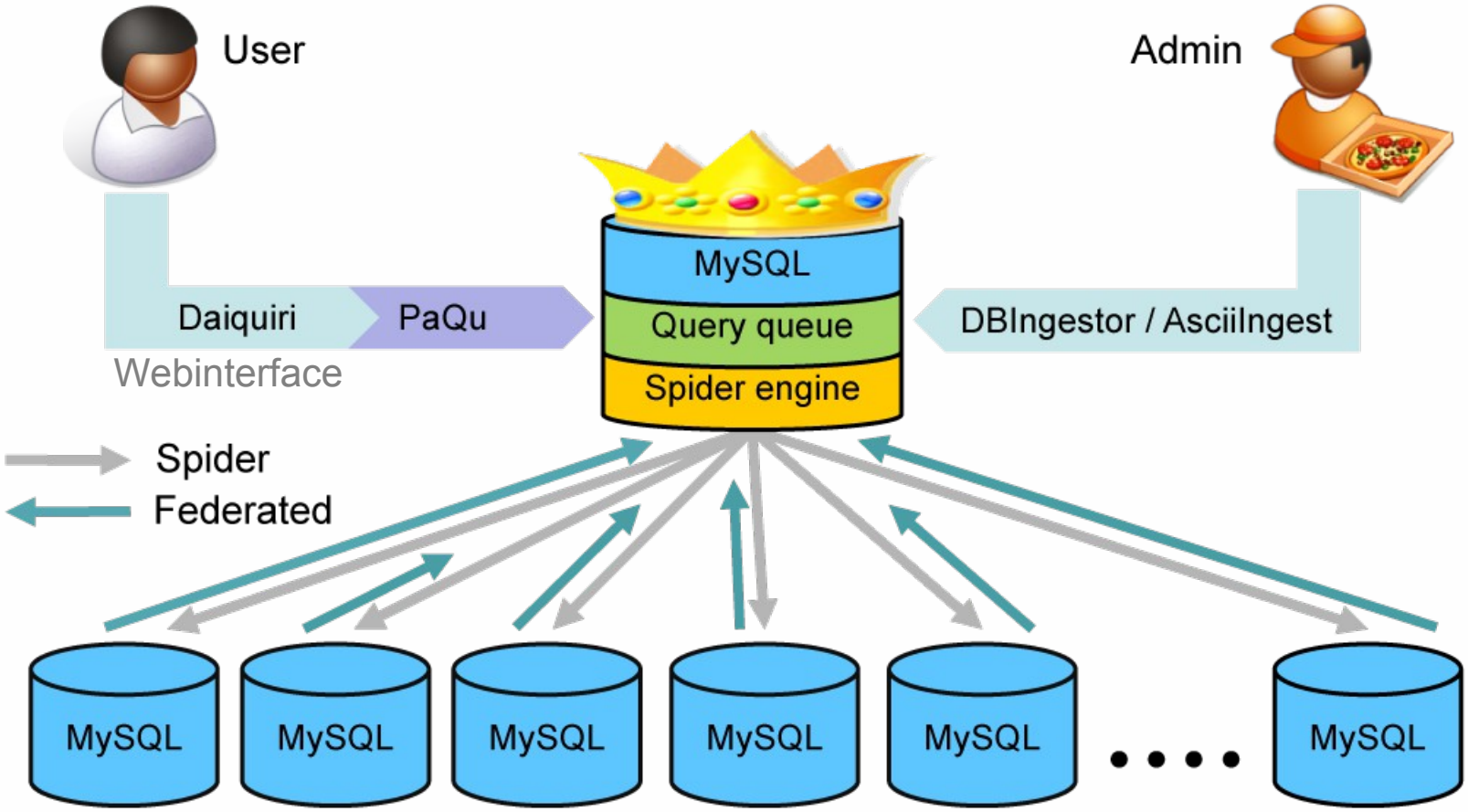


Fast access to data: MySQL cluster

- Previous database server:
  - 1 Microsoft SQL Server => expensive license, not easy to share
  - serving raw particle data for simulation snapshots is quite slow
  - Index on particle data (~ 10^{10} particles) ~ 1 week

- Solution:
  - use MyISAM engine of MySQL/MariaDB
  - => no transactions (need fast select, rarely upload)
  - => Spider engine (Kentoku Shiba) for distributed queries available
  - => data distributed over 10 nodes, queries much faster!
  - Spider engine now included with MariaDB!

MySQL cluster with Spider engine



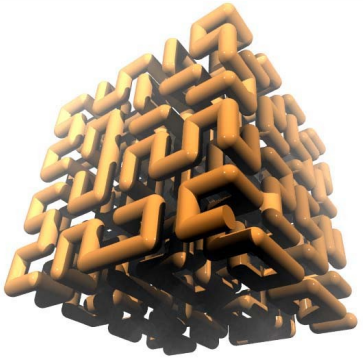
PaQu + QueryQueue

- PaQu:
  - reformulates queries, based on Shard-Query
  - e.g.: aggregate function count
    = count on each node + sum on head node

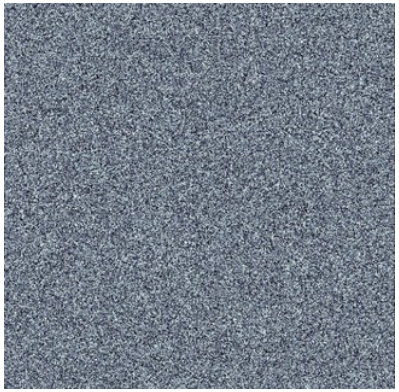
- QueryQueue:
  - allow asynchronous job submission
  - plugin for MySQL, supports priorities
  - control number of executing jobs on server
  - jobs stored in user table for later retrieval

Further MySQL plugins

- C-library libhilbert
  - For creating indexes of space-filling Peano-Hilbert curve in 20 dimensions

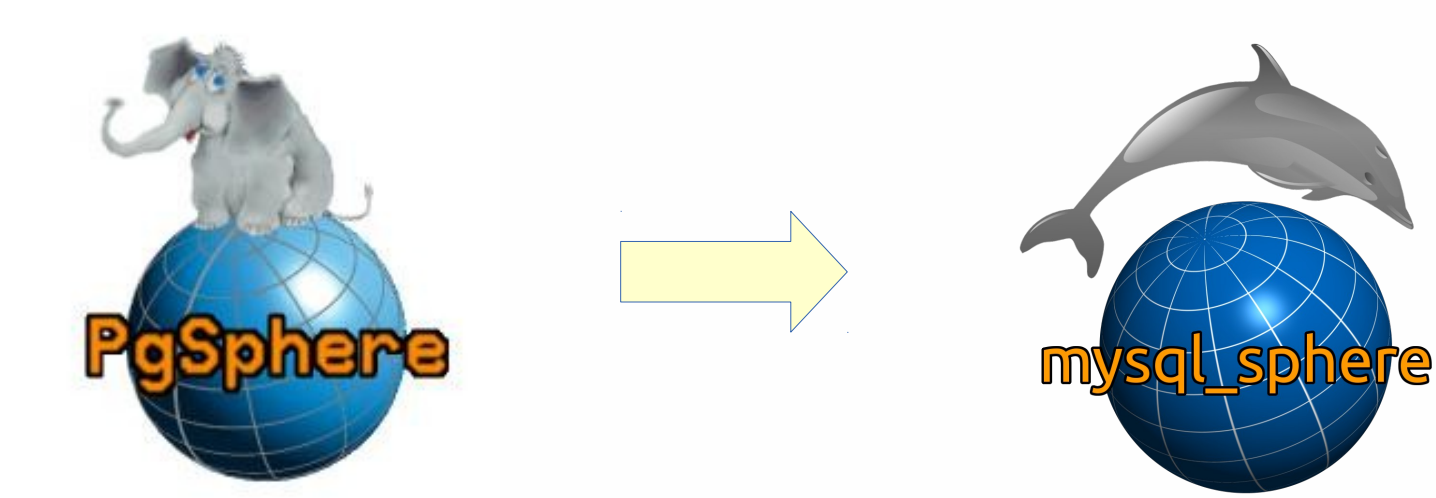


- MySQL sprng
  - Implements several random number generators
  - Better random sampling for large numbers than with built-in function



mysql_sphere

- Functions of pgSphere converted to mysql_sphere
- Allows queries on a spherical surface (cut outs, range in angles)
- Especially important for observational databases



- … now also ported to SQLite!

Data download: VOTable dump

- Plugin for MySQL, fork of mysqldump
- dumps VOTable format 1.3, ASCII or binary format, directly from MySQL database table
- => especially useful for large tables, no additional conversion on server needed
- Download from https://github.com/adrpar/mysqldump-vo

New portal: www.cosmosim.org

Web application: Daiquiri

- Developed by Jochen Klar und Adrian Partl
- http://escience.aip.de/daiquiri/
- Web application for publishing data
- Modular, highly customizable
- Using PHP, Zend-framework
- Modern interface using bootstrap, jQuery
- Authentication, Query Interface
- Wordpress integration
- One code base to serve most needs, open source, (easily) extendable

Database access

- SAMP for sending results to VO clients
- UWS implementedclient
  - Python client to access UWS services
  - Create, execute, abort or delete jobs
  - see https://github.com/adrpar/uws-client

- Package for „astroquery“ (developed by astroquery-contributors)
  - https://github.com/astropy/astroquery, maintained by Adam Ginsburg, Thomas Robitaille
  - affiliated to astropy
  - Provides access to astronomical web services (e.g. Simbad, UKIDSS)

Summary

- Publishing data of cosmological simulations
- DBIngestor library for data upload and conversion, for any kind of database, also for migrations
- MySQL cluster using Spider engine
- Own additions: PaQu, QueryQueue
- Libhilbert, MySQL sprng for random numbers
- Mysqldump for VOTable
- UWS client
- Daiquiri: web application with SAMP and UWS support

- All developments available on GitHub!
  => easy to share and contribute!