~~FairDB~~ TGenBase - universal database for FAIR experiments

Evgeny Lavrik,

Facility for Antiproton and Ion Research in Europe, for CBM Collaboration

Introduction

C++/ROOT based FairDb solution turned out to be not very fast

FairDb is a ROOT based library for direct database communication

- was used for parameter storage
- no user role management, no relations between classes, no query building, slow, only specific logic

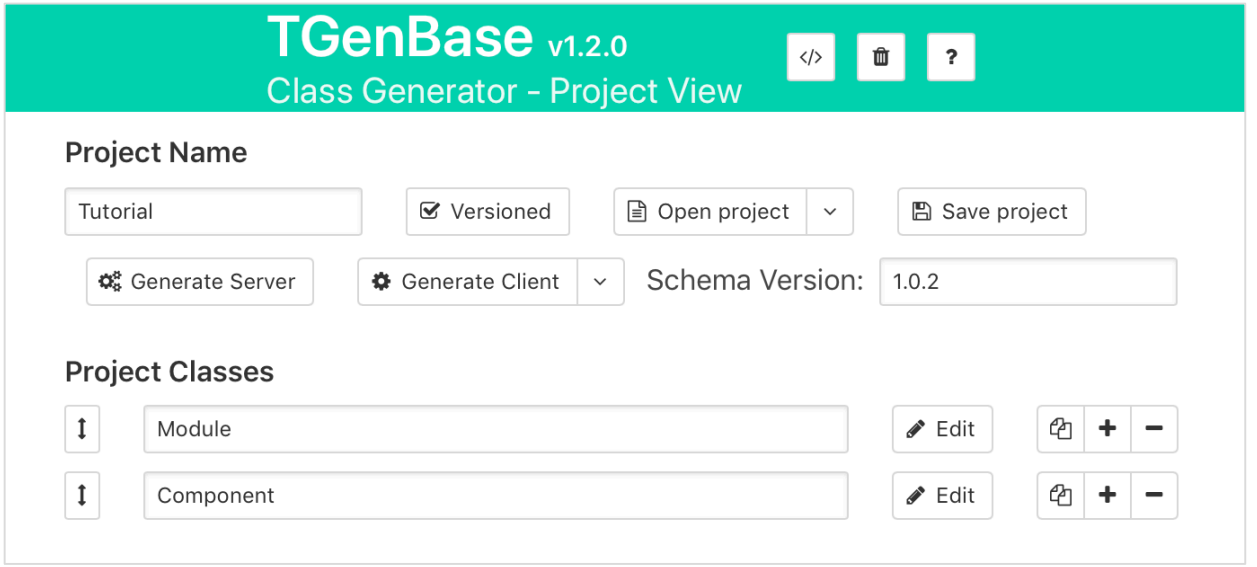
TGenBase overcomes these limitations + more

- general purpose, templated generation of code
- Client-Server architecture, clients, deployments, …

Why not to serve faster?

Data description made easy

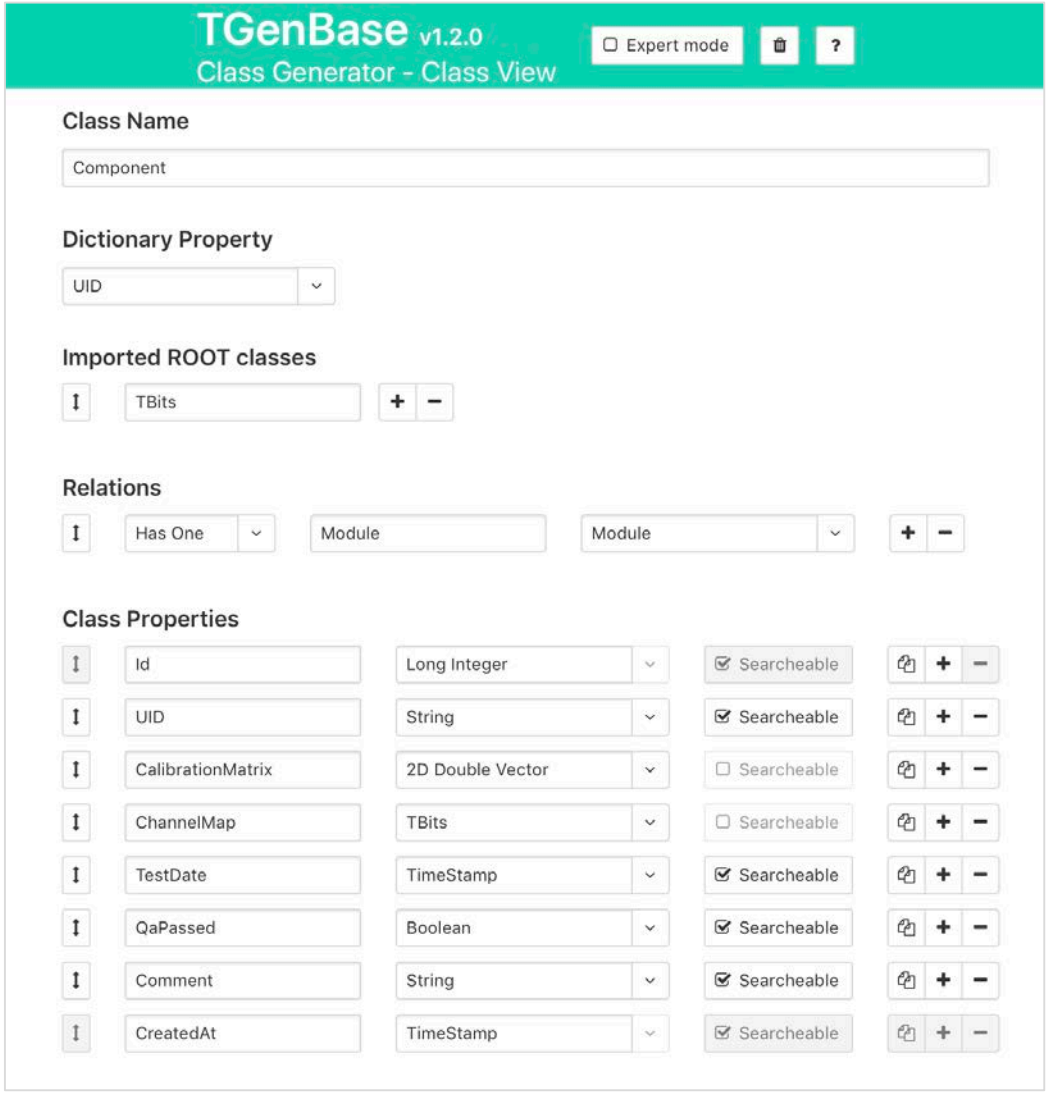
- We provides a web application to describe the data
- Provides usual CRUD (create, read, update, delete) logic
- And versioned insert-only logic -> no need to update, data will not be overwritten, all history is available



Name and configure your project, add classes to it
Or open sample projects ready to us

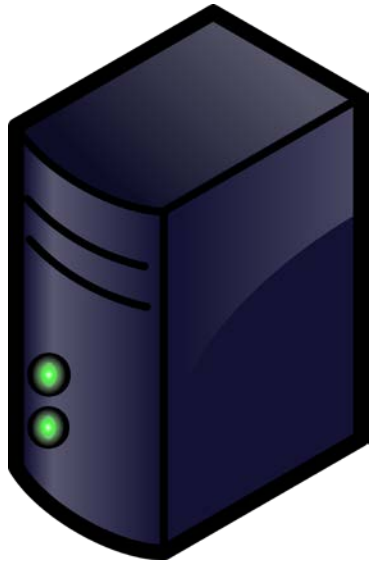
Data description made easy (cont.)

- In this workplace one defines the properties of the data, their types, relations with other classes, etc.
- Property configurations are predefined, with possibility to customize their behavior, validation rules, etc.
- Support for STL containers and ROOT classes



Database server

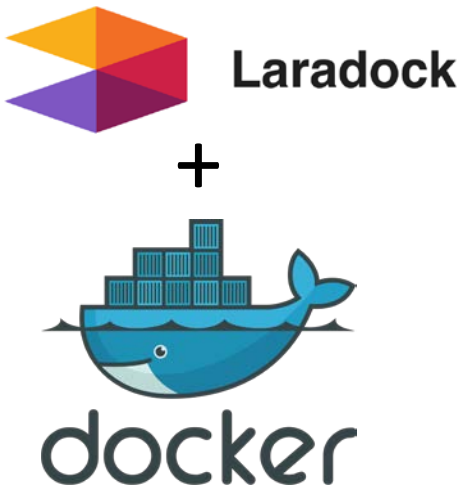
- Is in fact an HTTP server with secure data exchange by SSL
- Provides standard RESTful API to access data with JSON message exchange with clients
- Provides configurable role-based user permission management
- Caches data for frequent access
- Exchanges data with underlying DB Management Systems



Blinks with lights, spins itsfans, sends us stuff

DB server software stack

HTTPS Thin Clients



Container deployment

Server / load balancer

User Management
REST API

Fast Caching

Database Layer

Development/testing Server

- Light version of the full software stack
- Can be installed on Linux and macOS systems
- Has minimal dependencies, which are easily installed with a script provided
- Perfect solution for prototyping or small-sized data hosting

"Thin" clients to access data

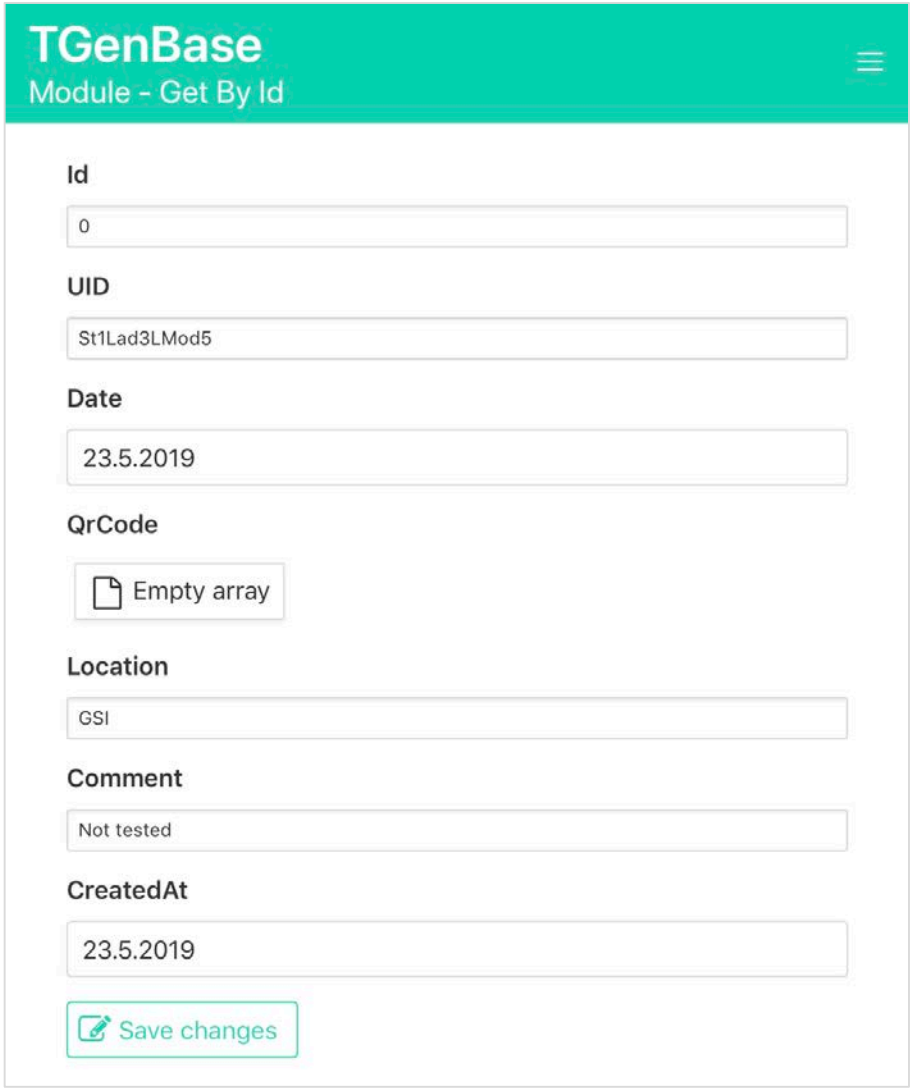
- There are following clients for FairDb
  - C++ library
    - requires only basic ROOT to be installed
    - minimal dependencies
  - Web based application - CMS
  - Python module for scripting
  - LabVIEW library
  - Matlab library?

- Allow to browse, query and modify data easily

Web Client

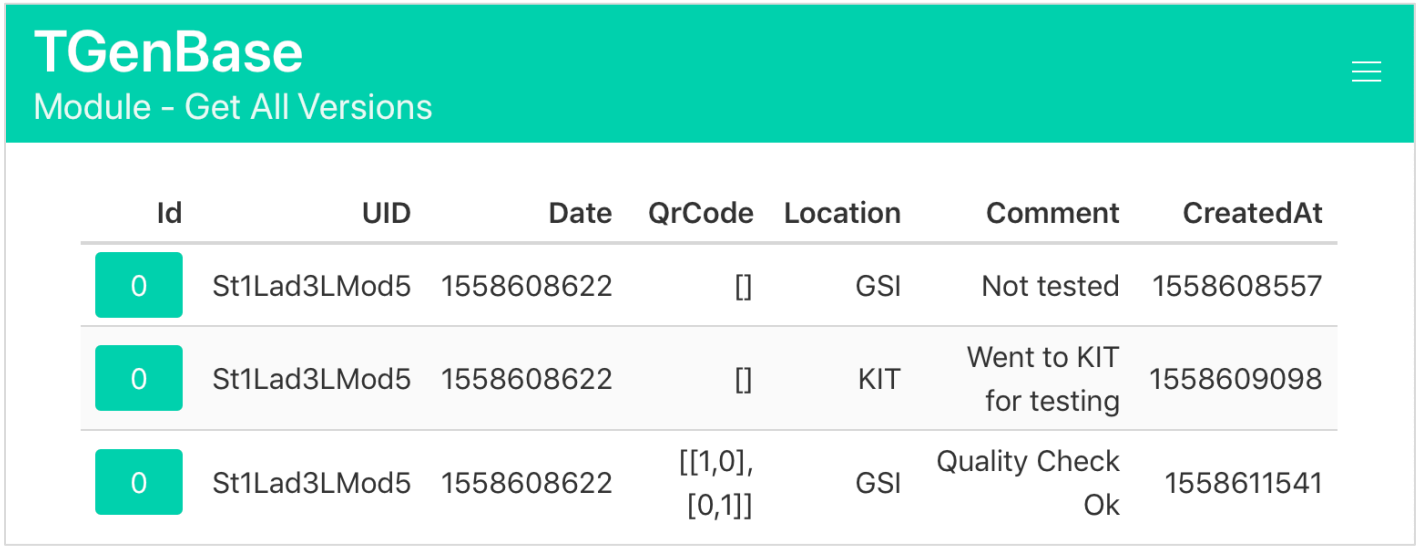
- You get, out of the box, a content management system ready to be used
- Provides the workplaces for data input, querying, editing, etc.
- Users can register with their email or use their existing LDAP accounts (GSI-weblogin)

- Image: data input/editing interface



Web Client, cont.

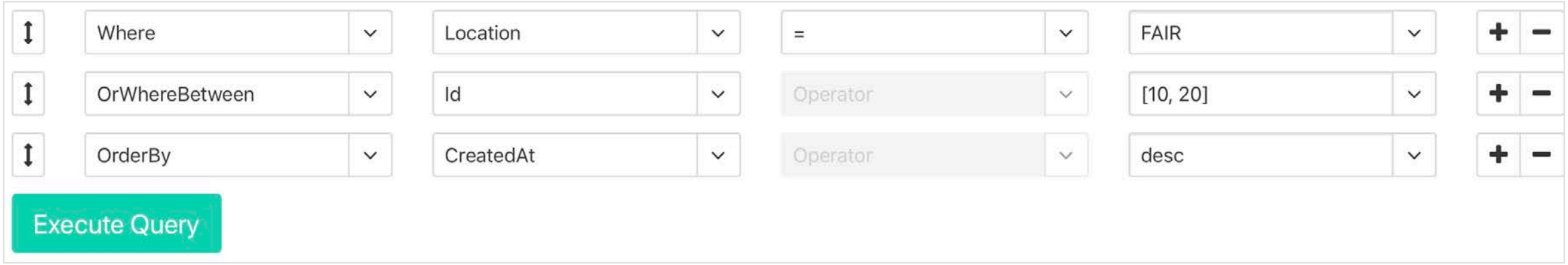
- Web client provides the administrator UI to manage users and their access rights
- For versioned logic, the history of the entries can be viewed
- Image: logistics out of the box ☺



Id	UID	Date	QrCode	Location	Comment	CreatedAt
0	St1Lad3LMod5	1558608622	[]	GSI	Not tested	1558608557
0	St1Lad3LMod5	1558608622	[]	KIT	Went to KIT for testing	1558609098
0	St1Lad3LMod5	1558608622	[[1,0], [0,1]]	GSI	Quality Check Ok	1558611541

Data Querying

- Web client



- Dynamic UI allows to add, remove and rearrange the search criteria

Data Querying

- C++

```
TGenBase::QueryBuilder().Where("Location", "=", "FAIR")
                        .OrWhereBetween("Id", {"10", "20"})
                        .OrderBy("CreatedAt", "desc")
                        .Execute<Tutorial::Module>()

(std::vector<Tutorial::Module>) { @0x7fd65dcb0280, @0x7fd65dcb0300, @0x7fd65dcb0380 }
```

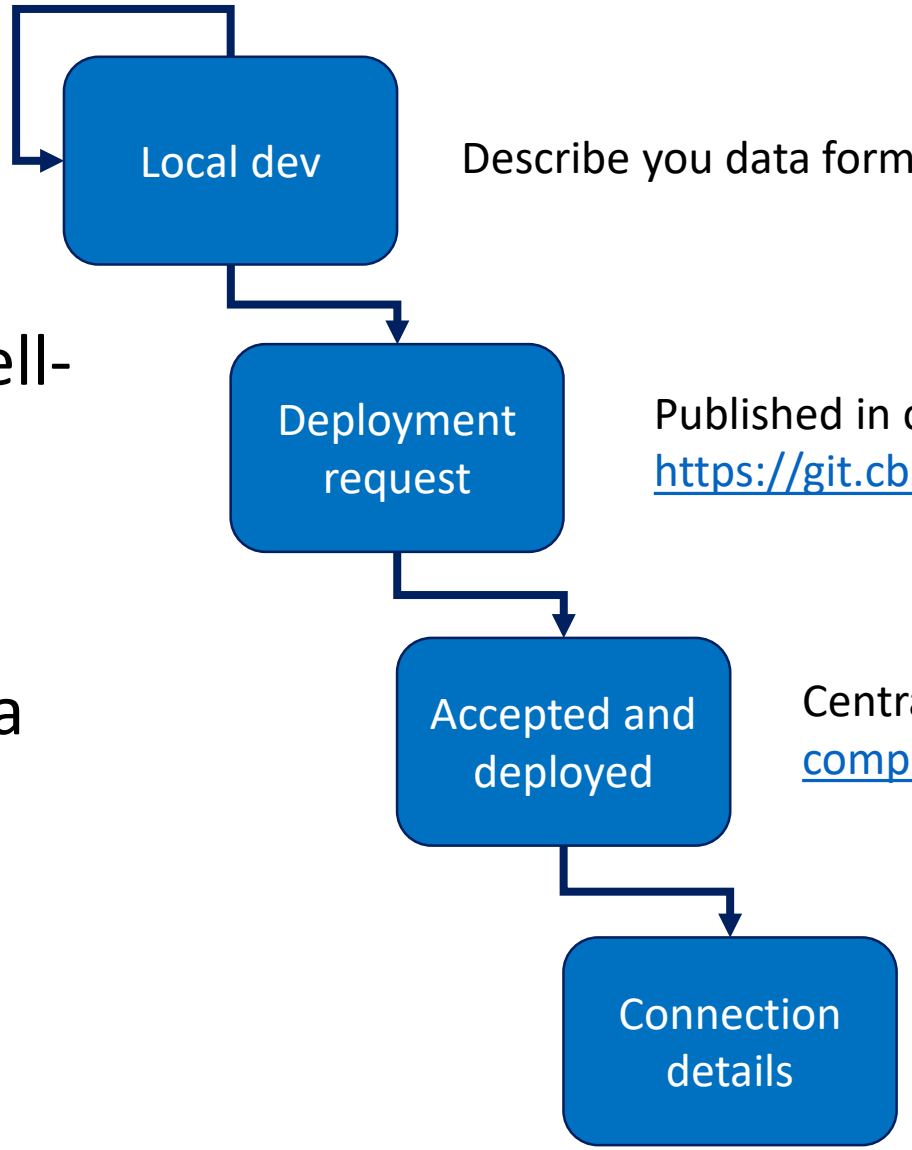
- Fluent interface which follows the ActiveRecord pattern
- Allows to chain the selection criteria and build complex queries
- Results are vectors (arrays) of objects

Schema

- Complete description of the classes and relations between them defines the data storage schema
- Ultimately it defines the contract between the client and server
- It maps easily to the database structure
- It has version which is incremented every time you generate server
- Schema evolution is done with migrations

Workflow

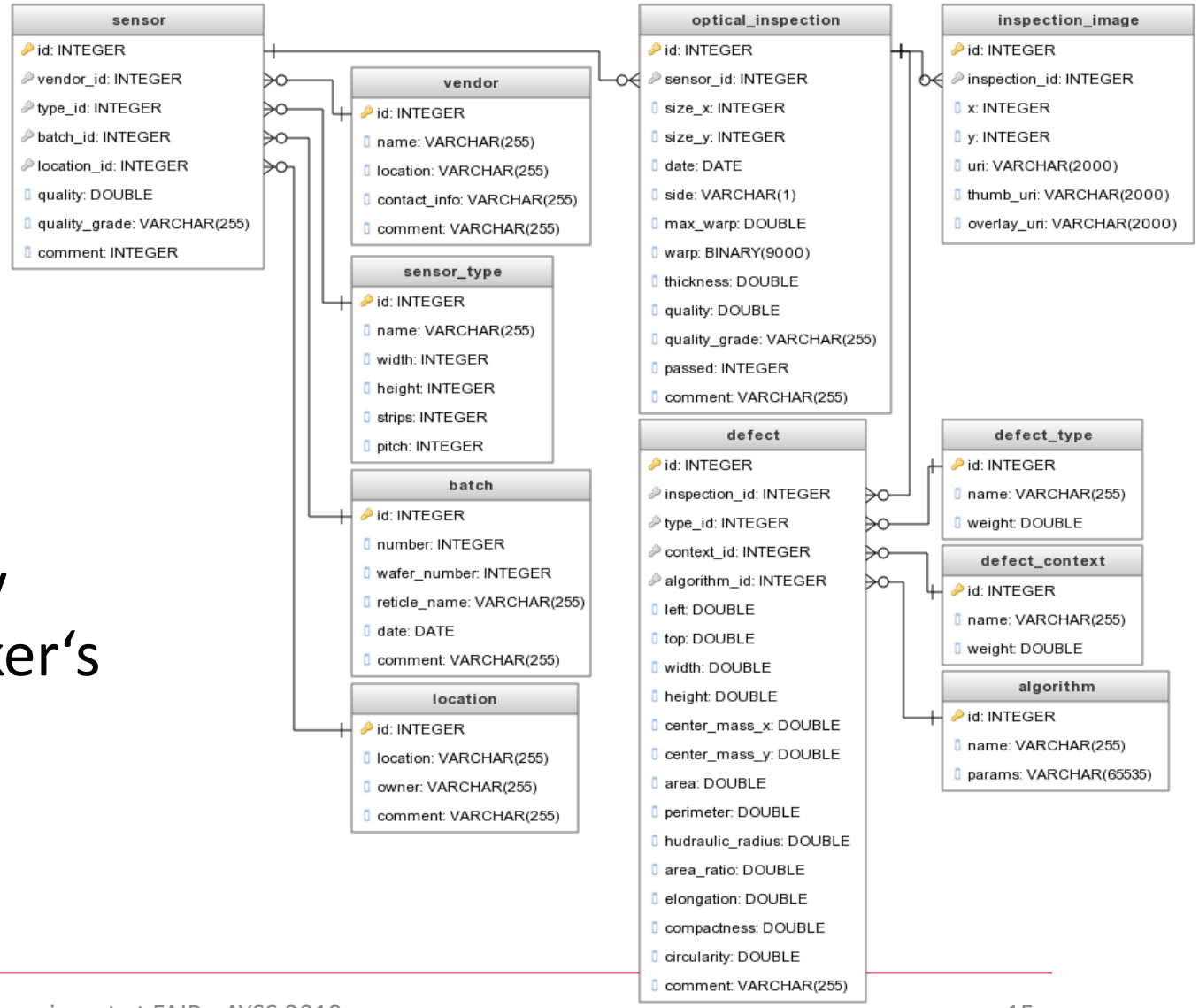
- Established and well-working for CBM
- Used for detector components and quality control data



DB Schema Example

User needs are different:
from one single class to complex hierarchies with class relations

Example: description of detector components, logistics and quality control data for CBM silicon tracker‘s sensors



Other applications

- Component database for detector components of full Silicon Tracking System (estimated row count is tens of millions)
- DCS (Detector Control System) database for hardware in collaboration with NA61 people
- Calibration data storage
- Analysis metadata storage – breakdown of data files in subsets, such as run periods, physics trigger based, etc.

Summary

- TGenBase is a powerfull set of tools to describe, host and access data with many language bindings
- It provides the data management and visualisation out of the box
- Provides two logic sets: CRUD and versioned insert-only
- General purpose: can be used for any data storage task

Check out the data description UI https://tgenbase.com

source code https://tgenbase.com/#/git

and the tutorial example online https://tgenbase.com:5050/