Proceedings of the XXVI International Symposium on Nuclear Electronics & Computing (NEC'2017)
Becici, Budva, Montenegro, September 25 - 29, 2017

STATUS OF THE GEOMETRY DATABASE FOR THE CBM EXPERIMENT

E.P. Akishina[1], E.I. Alexandrov[1], I.N. Alexandrov[1], I.A. Filozova[1,2,3,a], V. Friese[4], V.V. Ivanov[1,5]

[1] *Laboratory of Information Technologies, Joint Institute for Nuclear Research, 6 Joliot-Curie, Dubna, Moscow region, 141980, Russia*

[2] *Dubna State University, Institute of system analysis and management, 19, Universitetskaya., Dubna, Moscow region, 141980, Russia*

[3] *Plekhanov Russian University of Economics, 36 Stremyanny per., Moscow, 117997, Russia*

[4] *GSI, 1 Planckstraße, Darmstadt, 64291, Germany*

[5] *National Research Nuclear University "MEPhI", 31, Kashirskoe shosse, Moscow, 115409, Russia*

E-mail: [a] fia@jinr.ru

This paper presents the current state of the development of the information system "Geometry Database" (Geometry DB) for the CBM experiment. The Geometry DB supports the CBM geometry, which describes the CBM experimental setup at the detail level required for simulation of particles transport through the setup using the software GEANT3. The Geometry DB is aimed to provide the storage the CBM geometry and the tools for the geometry modules management, assembling various versions of the CBM setup and providing support of various versions of the CBM setup. The Geometry DB development is carried out on the basis of analyzed users' requirements. Both Graphical User Interface (GUI) and Application Programming Interface (API) are available for users of the Geometry Database.

Keywords: CBM, Geometry Database, setup, setup module, PostgeSQL, SQLite.

1. Introduction

The Compressed Baryonic Matter (CBM) experiment is one of major scientific program of the Facility for Antiproton and Ion Research (FAIR). The goal of the research program is to explore the QCD phase diagram in the region of high baryon densities using high-energy nucleus-nucleus collisions [1].

The need to develop CBM Geometry DB is caused by the low efficiency of the Geometry Files management in the current approach. The most essential of them are the operational complexity (low adaptability) and the tendency to the errors due to the absence of an approval procedure and rules of naming of versions.

Thus, the distribution of the geometry modules through a database appears a desirable solution. The expected benefits from a successful implementation of Geometry DB are usability and minimizing of coding for CBM employees; the business transparency and providing the monitoring basis for CBM software management.

2. Requirements to Geometry Database

The initial user requests were analyzed, and the User Requirements Document (URD) was developed [2]. This document formulates some essential concepts:

Geometry Module — File in ROOT [3] format with content of detector geometry.

Setup Module — Geometry module, link to the mother geometry module, its placement in the mother module (transformation matrix).

Setup - Combination of setup modules that represents the full CBM geometry.

URD defines three types of users: Lead Developer, Developer and CBM user. A *Lead Developer* is a coordinator and a responsible person for the entire Geometry DB. Only a Lead Developer is able to delete any setup, create new one and approve it. The approval means that the setup is tested and can be used by CBM users. Lead Developer can add new versions of magnetic field and/or materials.

A *Developer* is a person responsible for only one of the setup modules. Both Developer and Lead Developer can create, edit or delete the setup module.

A *CBM user* can view the content of the Geometry DB; download into local file system the full setup or one setup module that belongs to this setup; load setup into CBM ROOT framework.

Also, URD describes the keynote use cases defining the system architecture. The full functionality of Geometry DB is in the Table 1.

Table. 1. Register of keynote Use Cases

Code	*Main actor*	*Name*	*Brief Description*
U1	CBM User	Load Geometry	Allows loading a selected CBM setup or setup subset for the use in their own program.
U2	CBM User	Download Geometry	Allows downloading a selected setup or setup subset to the local disk.
U3	CBM User	Web View	Allows viewing a description of existing setups and setup and geometry modules.
D1	Developer	Add/Edit/ Delete Geometry Module	A Developer shall create a new version of setup and geometry modules; update or delete the available one.
D2	Lead Developer Developer	Add/Edit/ Delete Setup Module	Allows creation a new setup module: select one of the existing geometry modules and add the name of a setup module and the transformation matrix and select the mother module. They can update or delete setup module.

D2	Lead Developer	Add/Edit/Approve/ Delete Setup	Allows creation a new setup: select a set of existing setup modules; updating, approving or deleting of available setups.
D3	Lead Developer	Add/Edit/Delete Field	Allows inserting a new field, updating or deleting of existing records.
D4	Lead Developer	Add/Edit/ Delete Material	Allows inserting a new material, updating or deleting of existing records.

Thus, Geometry Database is the information system to store the modules of CBM and provide the set of tools for the collaboration. The Geometry DB is used:

- ✓ to provide interfaces to view, retrieve and update modules and setups;
- ✓ to store setups as combination of setup modules, magnetic fields and materials;
- ✓ to store setup modules as ROOT files and transformation matrix.

3. Assembling various options of the CBM setup

Assembling various options of the CBM setup is one of the keynote work processes of the Lead Developer. The CBM geometry is described in ROOT format and realized as a tree of nodes. The top level node Cave contains the entire geometry. The second level is consists of modules, each standing for a CBM subdetector — MVD, STS, RICH, MUCH, TRD, RPC, ECAL, PSD - or passive system (Magnet, Beam Pipe). In addition to the module geometries, the magnetic field map and a list of materials used in the setup are maintained.

Thus, the procedure for assembling various options of the CBM setup takes place according to certain rules. Each setup module involves the root-file and the transformation matrix (Figure.1). The root-file contains detailed information about the geometry of the module. The transformation matrix defines the module location inside the CBM setup. The additional files include information on the distribution of the magnetic field inside and around the dipole magnet, and a list of materials that the CBM installation consists of.

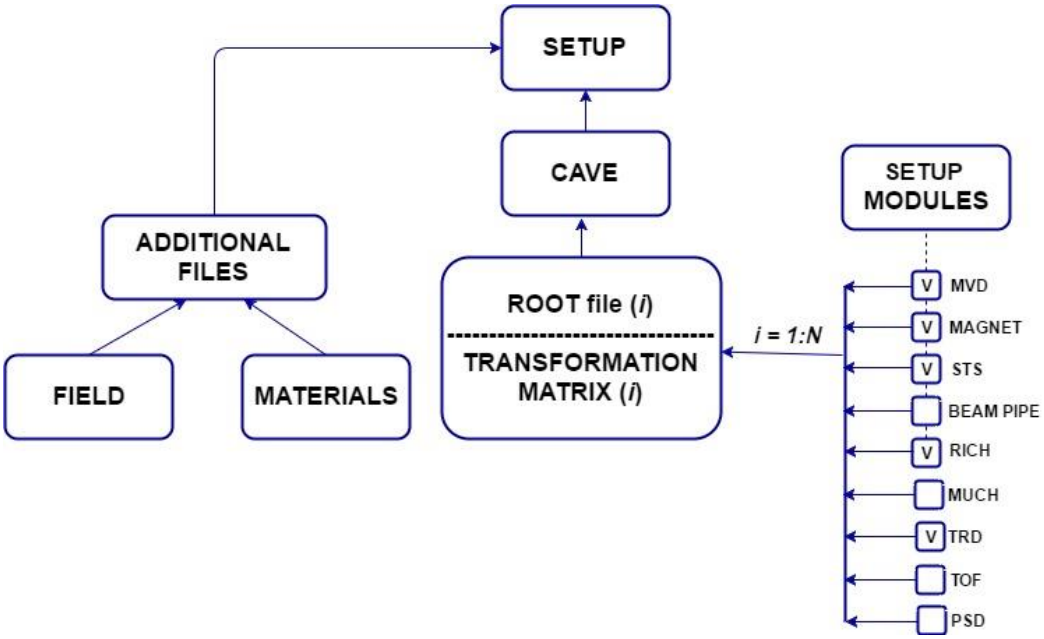

Figure 1. The Setup Structure

4. Architecture of the Geometry DB

The set of UML diagrams presented on figures 2-4 describes the system architecture.

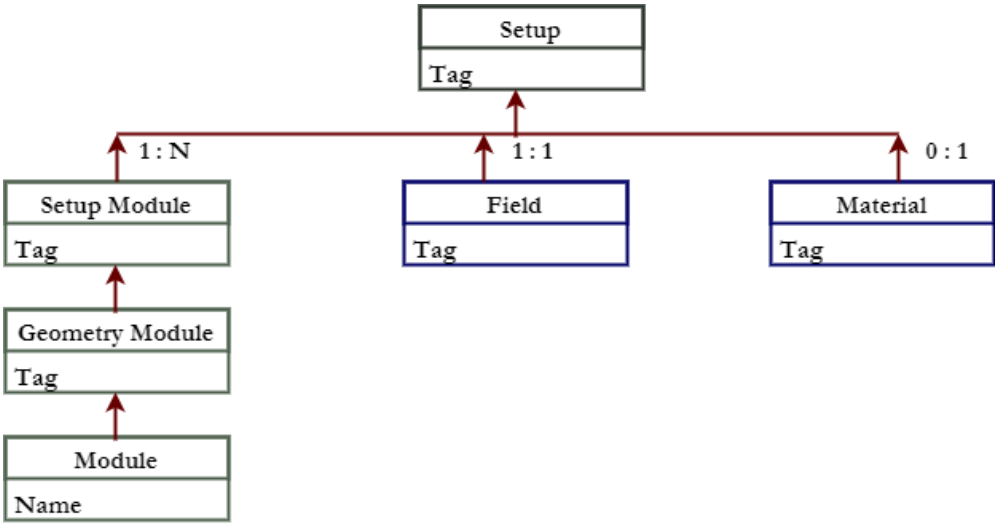

Figure 2. The general view and relationship classes of system

The keynote objects with their interrelationships are shown on the figure 3.

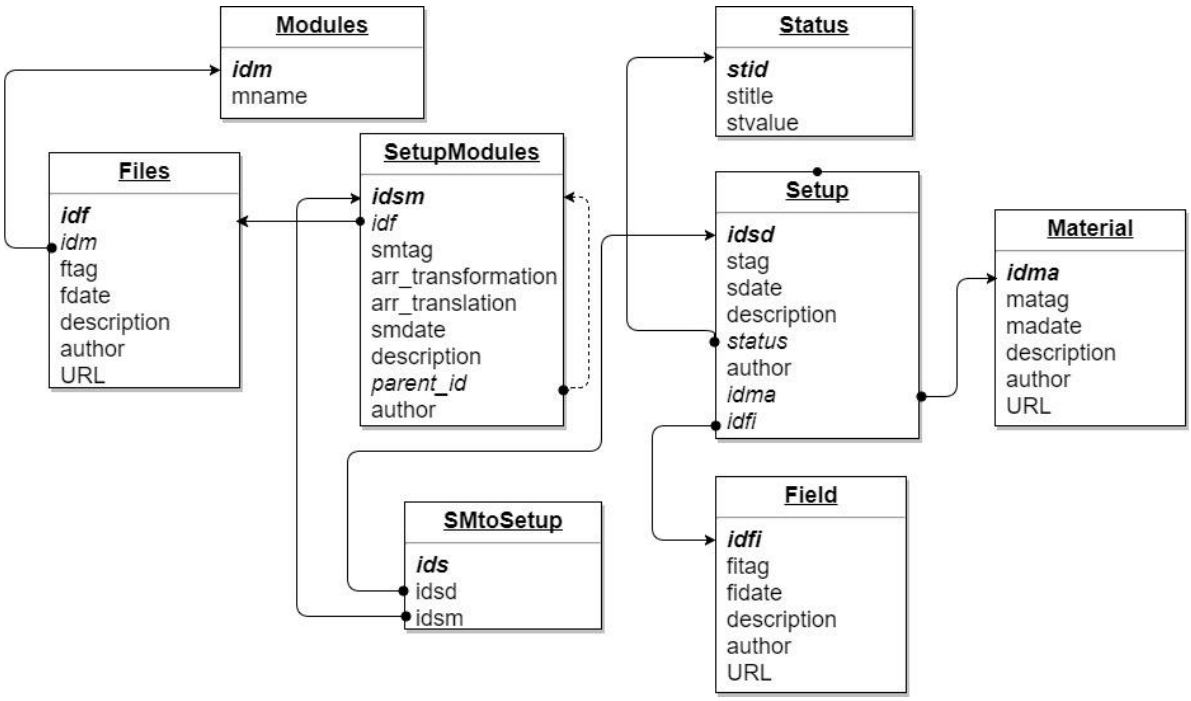

Figure 3. Logical Entity-Relationship Diagram

The entities *Setup, SetupModules, Material, Field* and *Files* has the field named *stag, smtag, matag, fitag* and *ftag* respectively. This is a string value identifying the entity instance by the software version, context and running version uniquely. *Description* is a brief description of the object, *Author* is the user name or email who created the object, and *Date* is the time of the object creation. *URL* is a string with a full path to the data file. The object *Setup* has a specific field *Status* which may have two possible values "Created" and "Approved". This object can be used by the *CBM user* when its *Status* has the value "Approved". The entity *SetupModules* has the recursive reference by the field *parent_id*. It's need to reflect the link to the mother detector in which the system corresponding to the setup module should be. The field *arr_transformation* stores the values of transformation matrix to calculate the coordinates of the *Setup module* (entity *SetupModules* on figure 3) in the coordinate system of the parent module. The *SetupModules* instance is associated with one *Files* object. The field *URL* is a reference to the file, which contains the geometry data of the detector in the ROOT format. Each *Files* object has association with only one *Modules* object. Only one *SetupModules* object of the specific type of the detectors can be included in one *Setup*. The *Setup* object has association with a set of *SetupModules*, only one *Field* object and only one *Material* object.

The deployment diagram (Figure 4) illustrates the physical deployment of system artifacts on the server host and user workstations.

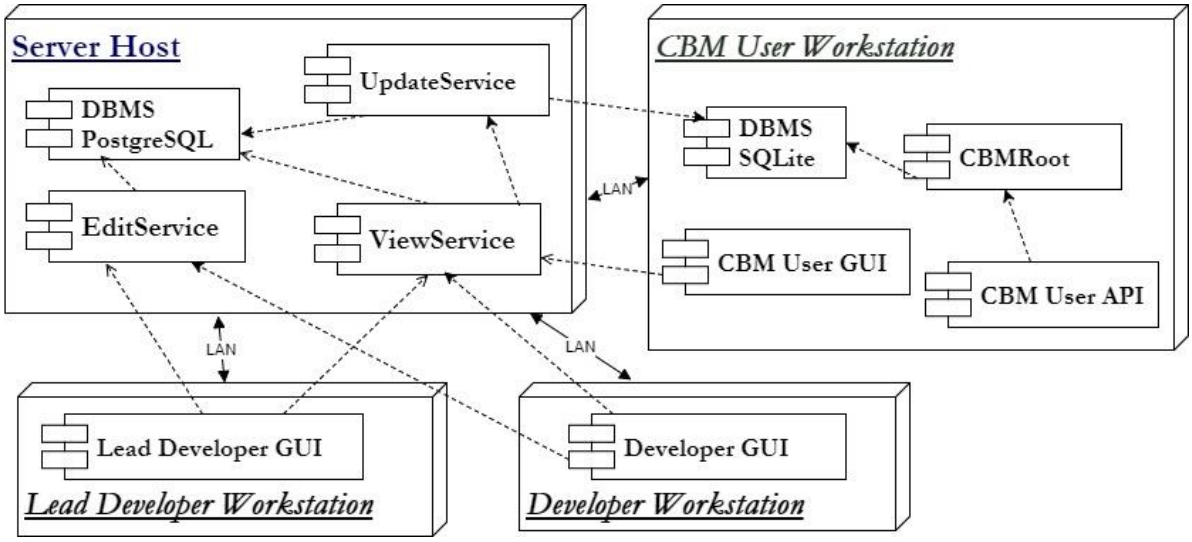

Figure 4. Deployment Diagram

The system supports two databases: centralized and local. The centralized database is implemented at the *Server Host* under DBMS PostgreSQL [4]. *Lead Developer* and *Developer* use the centralized database to view and update data corresponding to any object of the Geometry DB by Graphical User Interface (GUI). The local database is located at *CBM User* workstation into the CBM ROOT environment and implemented under DBMS SQLite [5] as a replica of centralized database. *UpdateService* (Figure 4) provides updating of the local database. The replica is used for simulation of particles transport through the setup using the software GEANT3.

All types of Users can view the information from Geometry Database. *Lead Developer* and *Developer* use Database to modify data files corresponding to any object of Geometry DB by GUI.

5. Implementation

5.1. Graphical User Interface

GUI was implemented as a standard web-interface. The user interface has a compact form and allows getting the information by drilling down. So, from the list of available setups, the user can go to the level of detailed description of the setup, and further to the description of setup modules (Figure 5). The menu item «Download GeometryDB» allows to download the archive of the full database to the user local disk. Also, the user may to download the setup as archived file to the local disk by the click on the button «Download» in the row of need setup. Only approved setups are included in the local database.

The subsystem *Configure Access* was developed to provide the access rights for other users' types. *Configure Access* defines the role-based access control system to all Geometry DB services and implements the full functionality to manage, grant and revoke any rights (Figure 6).

After authorization in the system, the user can work with edit menu (see Figure 7). To approve Setup, one needs to press the button ✔, to delete – ✖.

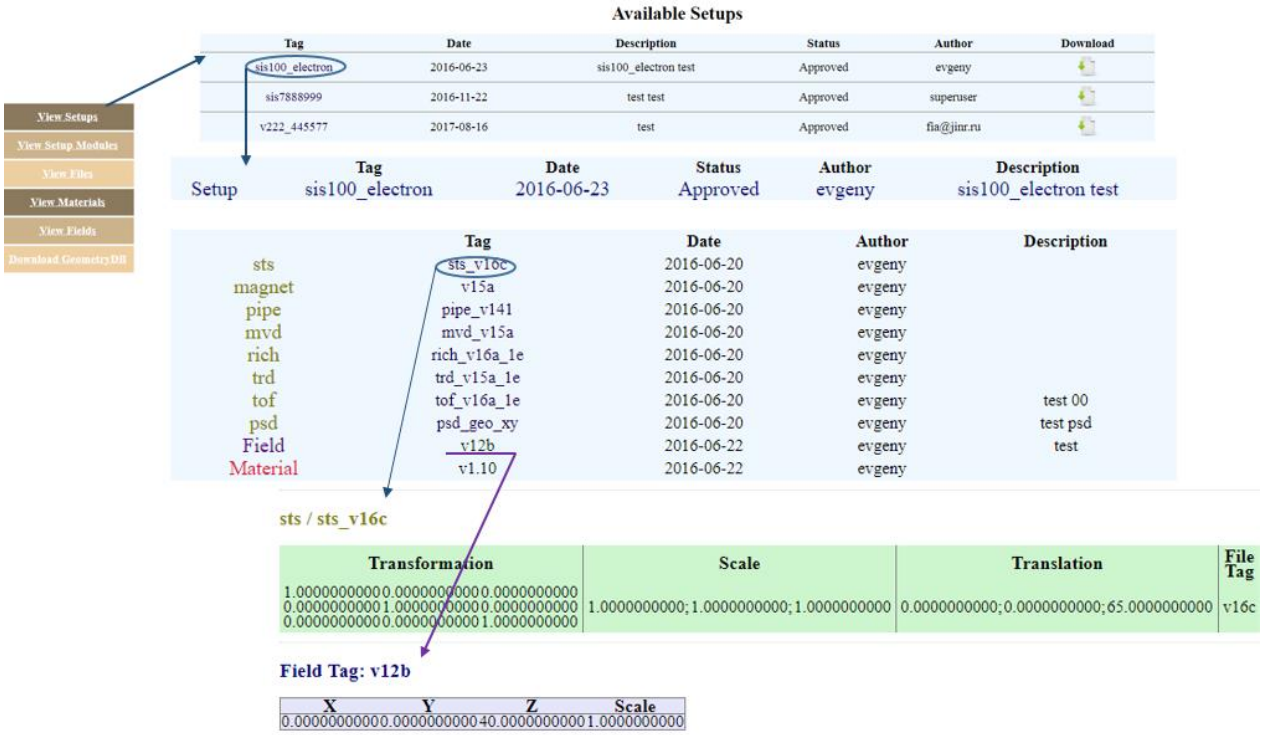

Figure 5. Web-interface. View mode

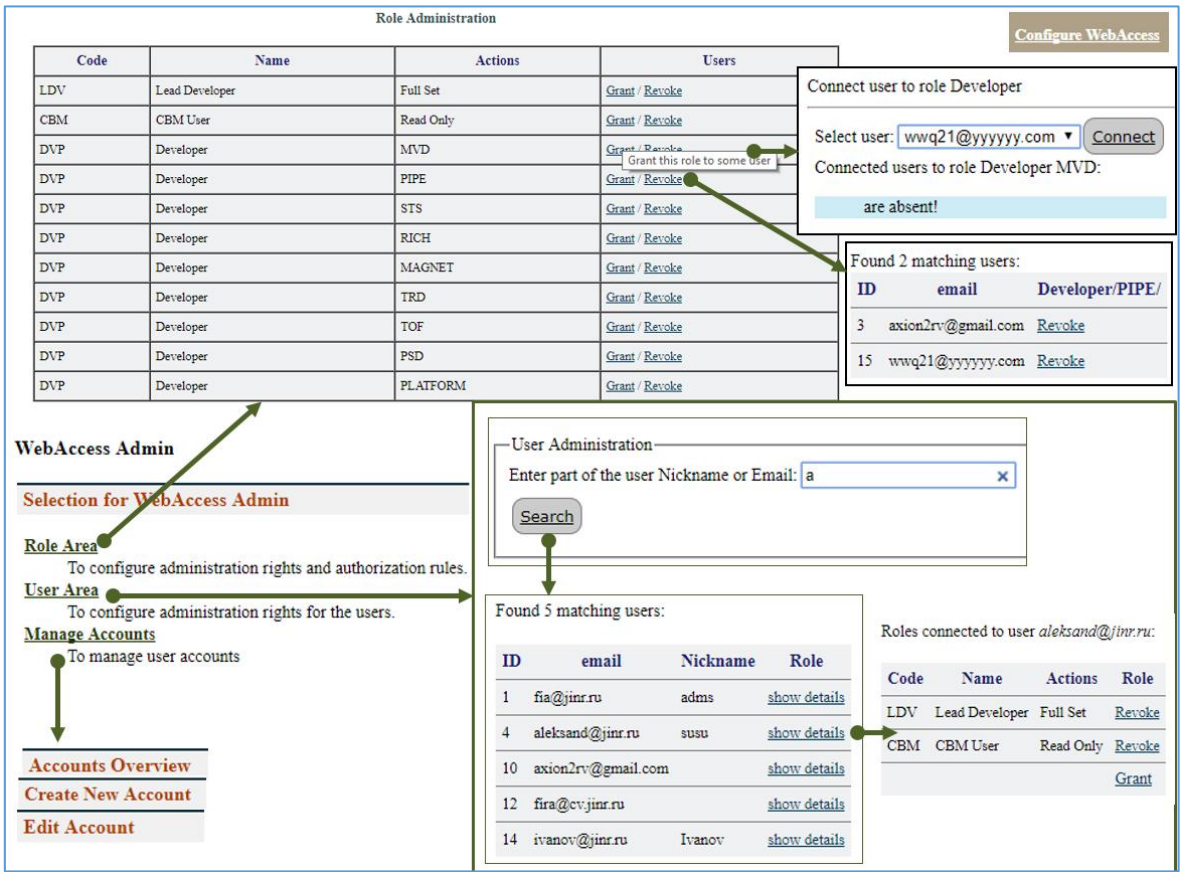

Figure 6. Web-interface. Configure Access

The value of *Description* may be changed directly on the form. This text field is editable. A new input value must be saved by click the button «OK».

Only a *Lead Develo*per can create a new Setup or modify the existing one. To create a new Setup, one needs to press the button «Create New Setup» and to enter a unique value for the Tag in the pop-up window (see Figure 7). After that, the form for selection of needed modules appears (Figure 8). They are placed on the block in the style of "accordion". Via radio buttons the *Lead Developer* compiles the new Setup.

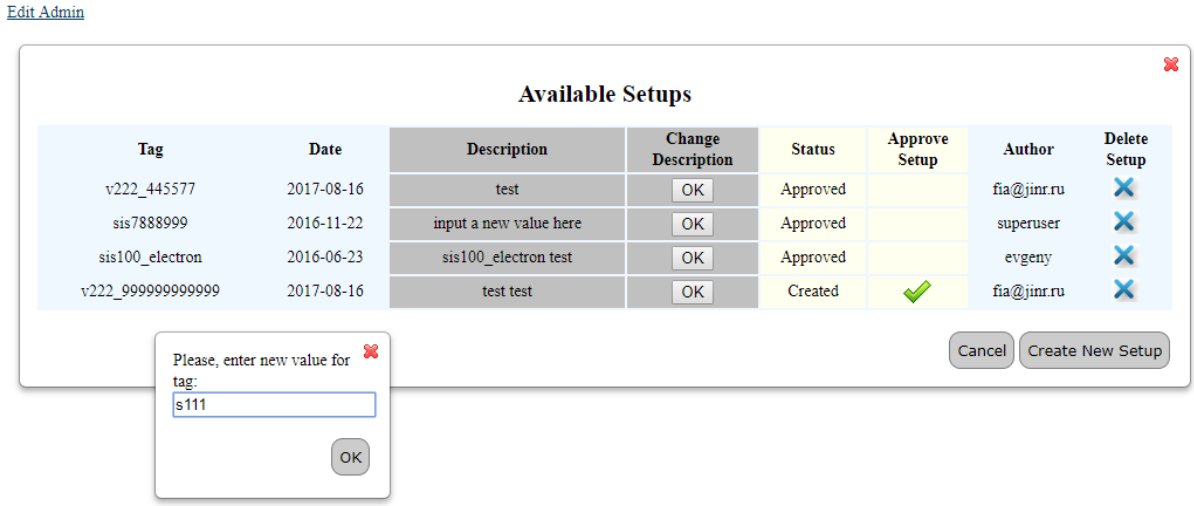

Figure 7. Web-interface. Edit mode

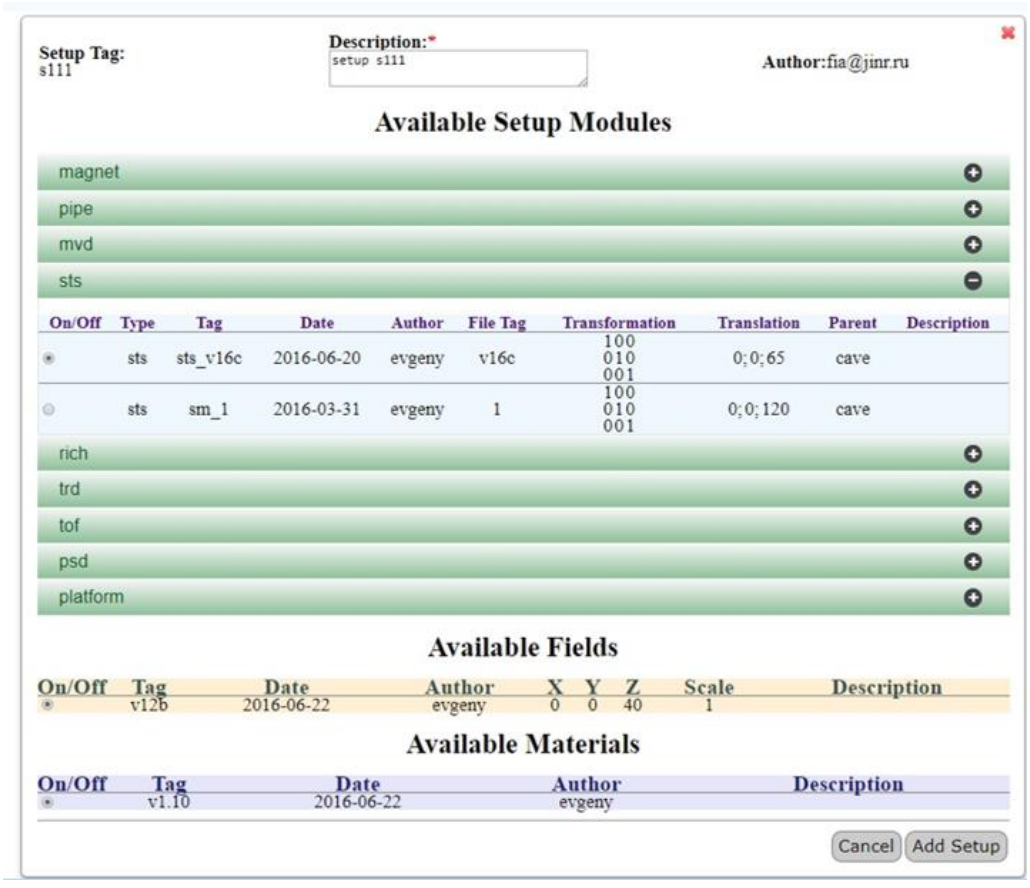

Figure 8. Web-interface. Compiling Setup

After clicking the button «Add Setup», the new Setup appears in the list of available Setups. By analogy, editing the metadata about Field, Material, File and Setup Modules (Figure 9) is possible.

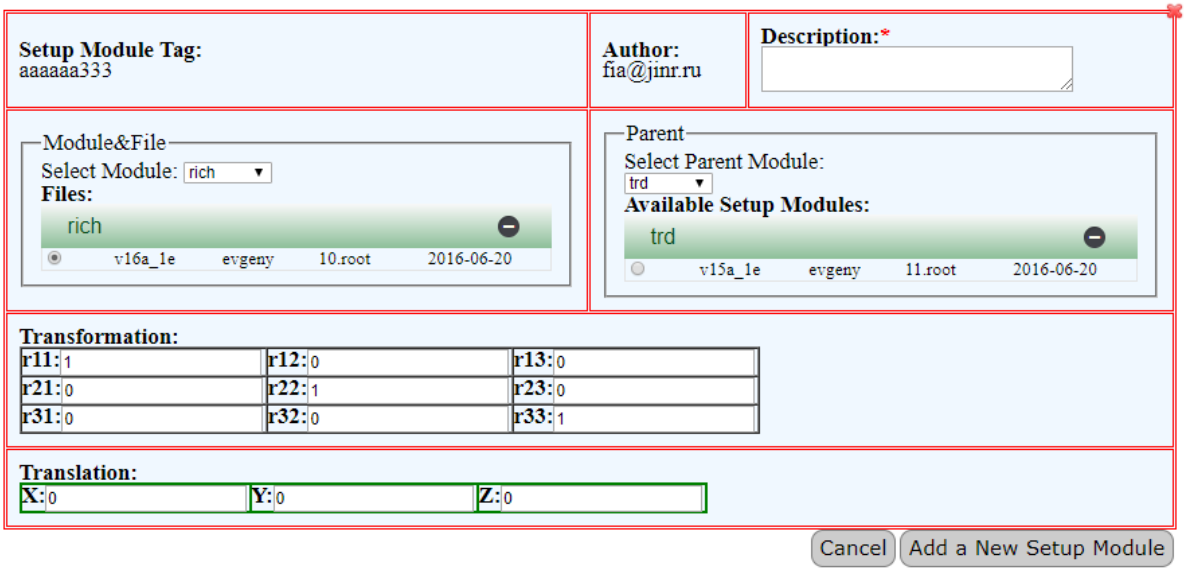

Figure 9. Web-interface. Add&Edit Setup Module

5.2. Application Programming Interface

The API is implemented as set of macros of the ROOT framework [3]. Any macro can be used as executable file or can be called from other ROOT macros. Two macros are realized: `getSetupList` and `loadSetup`. The macro `getSetupList` prints tag, date of creation, author name and value of parameter *description* for each approved setup that is stored in database Figure 10).

```
[aleksand@cbmdb geomdb]$ root -b -q getSetupList.C
   ------------------------------------------------------------
  | Welcome to ROOT 6.11/01                 http://root.cern.ch |
  |                               (c) 1995-2017, The ROOT Team |
  | Built for linuxx8664gcc                                    |
  | From heads/master@f5d2f9a, May 22 2017, 22:58:00           |
  | Try '.help', '.demo', '.license', '.credits', '.quit'/'.q' |
   ------------------------------------------------------------


Processing getSetupList.C...
sqlite://test2.db
Setup list:
Tag                 Date            Author              Description
sis100_electron     23.06.2016      evgeny              desc sis100 electron
```

Figure 10. Listing of executing `getSetupList` macro

Macro `loadSetup` load geometry of setup into CBM ROOT environment. Geometry can be used in root environment after this operation. The geometry of setup with tag `sis100_electron` is presented on Figure 11.

6. Conclusion

The prototype of the information system Geometry DB for the CBM experiment has been developed according to User Requirements Document [2]. Geometry DB provides storing and retrieving the geometry of CBM modules and involves the databases, GUI tools and API tools.

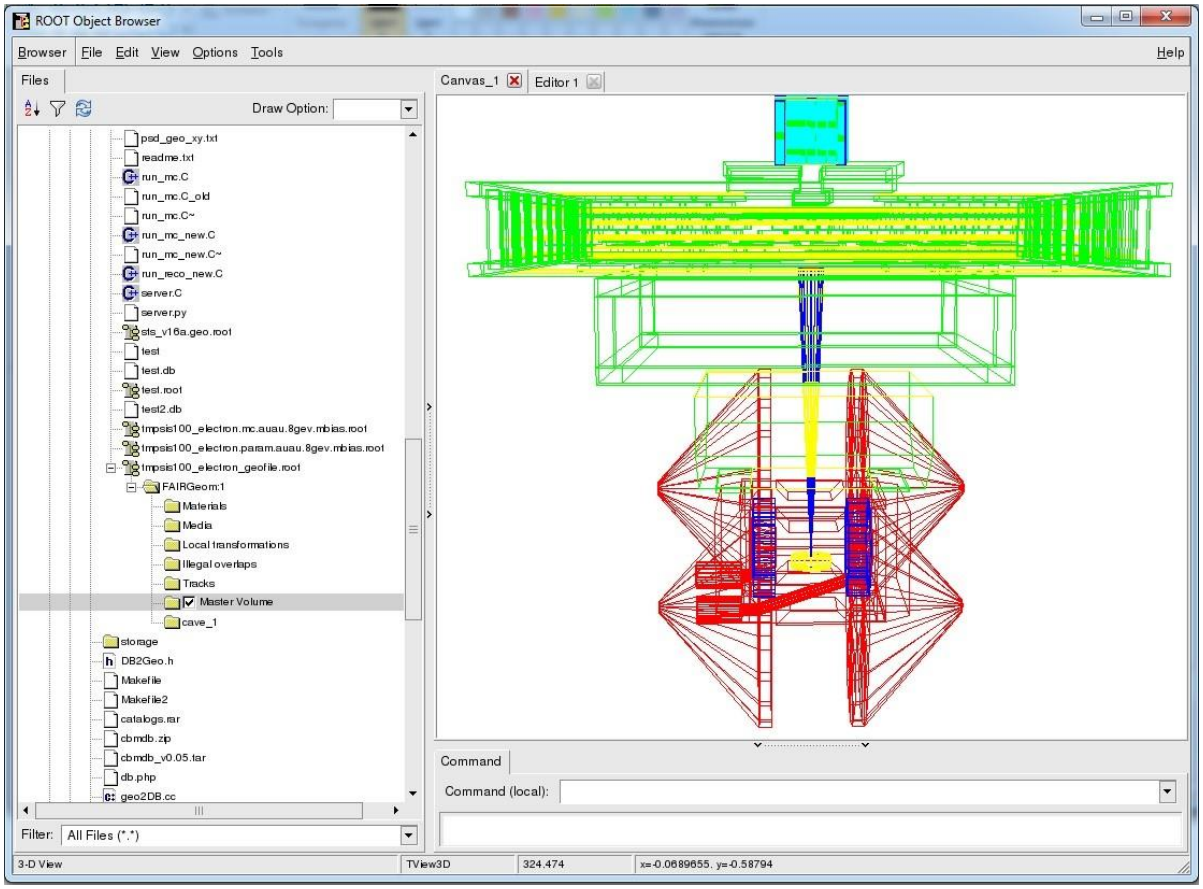

Figure 11. Geometry of setup `sis100_electron`

The system supports two databases: centralized and local. GUI tools are implemented as web-application for viewing, updating data corresponding to any object of the Geometry DB in the centralized database. The local database is used for simulation of particles transport through the setup using the software GEANT3. API tools are realized as a set of ROOT macros to load geometry of setup into CBM ROOT environment.

The actualization of the database content is the next stage of work. After receiving the feedback from users, the improvement of the functionality will be done.

References

[1] Ablyazimov, T. a.o. Challenges in QCD matter physics. The scientific programme of the Compressed Baryonic Matter experiment at FAIR//European Physical Journal A, Volume 53, Issue 3, Article:60, 2017. - p. .- DOI: 10.1140/epja/i2017-12248-y.

[2] User Requirements Document of the Geometry Database for the CBM experiment. Available at: http://lt-jds.jinr.ru/record/69336?ln=en (accessed 01.11.2017).

[3] Project official website *Data Analysis Framework ROOT*. Available at: https://root.cern.ch/ (accessed 01.11.2017).

[4] The official website of PostreSQL project. Available at: https://www.postgresql.org (accessed 02.11.2017).

[5] The official website of SQLite project. Available at: https://www.sqlite.org/ (accessed 02.11.2017).