Nuclear Inst. and Methods in Physics Research, A 908 (2018) 43–48

Contents lists available at ScienceDirect

Nuclear Inst. and Methods in Physics Research, A

journal homepage: www.elsevier.com/locate/nima

Technical Notes

# GDML based geometry management system for offline software in JUNO

Kaijie Li [a], Zhengyun You [a], Yumei Zhang [a,*], Jiang Zhu [a], Tao Lin [b], Ziyan Deng [b], Weidong Li [b]

[a] Sun Yat-sen University, Guangzhou 510275, China
[b] Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, China

ARTICLE INFO

*Keywords:*
JUNO
Offline software
Geometry
GDML
ROOT

ABSTRACT

In the JUNO experiment, a geometry management system has been developed in offline software. The system is based on Geometry Description Markup Language (GDML) to provide consistent detector description for different applications, such as simulation, reconstruction, visualization and analysis. It works for different detector designs with flexibility and has been successfully used for detector optimization and performance evaluation with Monte Carlo data.

## 1. Introduction

The geometry management system is indispensable in any high-energy physics (HEP) experiment. In offline software of an experiment, the basic function of the geometry management system is to describe the detector properties, such as the geometrical structure, shape, dimensions, materials, positions and rotations of detector components in the coordinate system. It also provides interfaces for applications in the offline software where geometry information is necessary, such as simulation, reconstruction, event display and data analysis.

GDML (Geometry Description Markup Language) [1] is a detector description language based on XML(eXtensible Markup Language) [2]. GDML describes all detector information through a set of tags and attributes in plain text format to provide persistent detector description for an experiment. The automatic detector data conversion interfaces between GDML and the commonly used HEP software, such as Geant4[3] and ROOT [4], are also available for users to develop applications based on them in a consistent way.

The Jiangmen Underground Neutrino Observatory (JUNO) [5] is a multi-purpose neutrino experiment. It is currently under construction in Jiangmen, Guangdong, China. Located at the distance of 53 km from both of Yangjiang and Taishan Nuclear Power Plants, JUNO is designed to determine neutrino mass hierarchy by precisely measuring the energy spectrum of anti-neutrinos from the reactors. Besides, JUNO also has rich physics goals, such as studying properties of solar neutrinos, atmospheric neutrinos, geoneutrinos and supernova burst neutrinos, as well as exploring new physics beyond the standard model [6].

In this paper, we start with an introduction of the JUNO detector design and the requirements of the geometry management. Then we will describe how to design the GDML based geometry management system in JUNO offline software to work for different detector designs. Next, the implementation of the geometry system and its applications will be introduced. Finally, we will discuss its performance in offline software.

## 2. JUNO detector and geometry management requirements

As shown in Fig. 1, the JUNO detector consists of three sub-detectors: Central Detector (CD), Water Pool (WP) and Top Tracker (TT).

The Central Detector is the main part of JUNO detector, its inner layer is an acrylic sphere with radius of 17.7 m and thickness of 12 cm, the outer layer is made up of stainless-steel trusses. The acrylic sphere is filled with about 20,000 tons pure liquid scintillator (LS) to detect neutrinos. The surface of CD is surrounded by about 18,000 20-inch photomultiplier tubes (PMT) and about 25,000 3-inch PMTs for photon detection. The CD is placed in a water pool with more than 2,000 20-inch PMTs in it, which works as a water Cherenkov detector to veto cosmic ray muons and radioactive backgrounds from the air and rocks surrounding the detector. On top of the water pool is the Top Tracker, which is made of plastic scintillators for muon tracks identification and veto.

The geometry management system in offline software needs to work with different detector designs. In conceptual design of the JUNO detector, four different options of the Central Detector designs had been proposed [7]. Sketches of the four CD designs are shown in Fig. 2, The top left plot is the acrylic design option, an spherical acrylic tank with the diameter of 35.5 m to hold 20 kilotons of LS, which is supported by stainless-steel trusses. This is also the design option that has been adopted in the current JUNO construction. The top right plot is the balloon design option, in which the container is made of balloon to separate LS and buffer materials, with a steel tank in the water pool to hold them and the PMTs. The bottom two plots are small module option (bottom left) and large module option (bottom right), respectively. By dividing the spherical surface into triangle modules, the PMTs are arranged in each individual module to separate them from each other, with the gap between the modules filled with LS.

* Corresponding author.
*E-mail address:* zhangym26@mail.sysu.edu.cn (Y. Zhang).

https://doi.org/10.1016/j.nima.2018.08.008
Received 26 April 2018; Received in revised form 3 August 2018; Accepted 4 August 2018
Available online xxxx

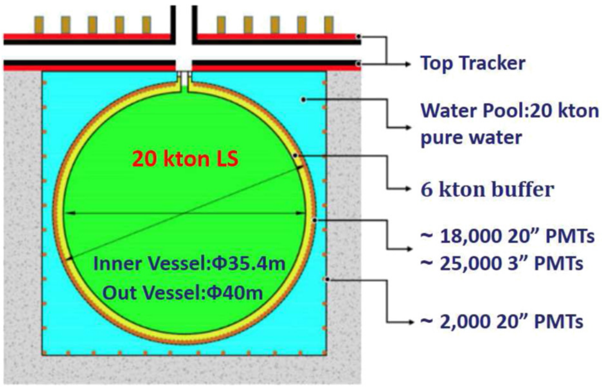


**Fig. 1.** Schematic view of the JUNO detector.

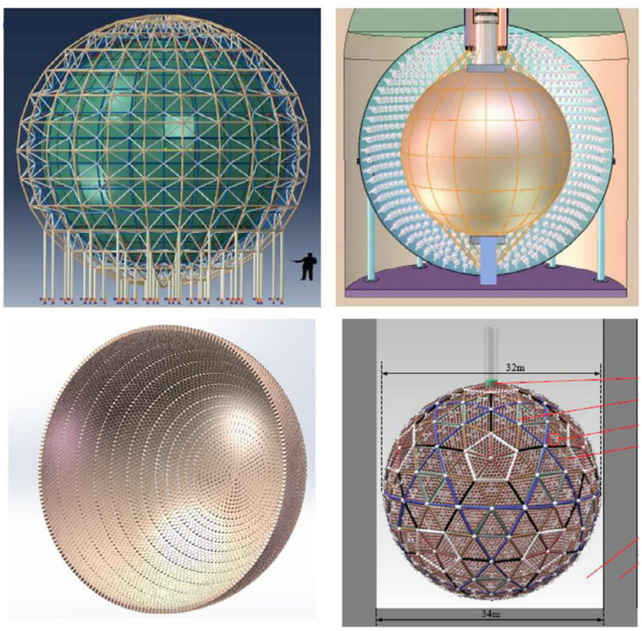

**Fig. 2.** Sketches of the four designs of JUNO Central Detector.

To study the performance of different detector designs, such as energy resolution and background identification power, users need to run detector simulation, reconstruction and analyze the Monte Carlo data in JUNO offline software with different detector geometry. Basically, the JUNO geometry management system has to meet the following requirements:

(1) Work with different detector geometry and can switch between them easily.

(2) For any specific detector design, provide consistent detector description for different applications, including simulation, reconstruction and analysis.

(3) Help calculation of the PMT coverage and optimization of their arrangements, so that the detector can have maximum geometrical acceptance of PMTs to receive as many photons as possible for precise neutrino energy measurement.

## 3. Implementation

### 3.1. Design idea

In offline software of HEP experiments, the geometry management system tries to describe properties of the detector as precisely as possible. On the other hand, a modern HEP detector is a complicated large system with huge number of components. The requirements of detector description precision and software performance are usually contradictory and a balance point needs to be found between them.

Different applications like simulation and reconstruction usually focus on different aspects of detector properties and have their own specific requirements, the geometry management system needs to provide an identical detector description for different applications to guarantee the consistency between them. Besides, the interfaces of the geometry management system also need to be user-friendly, easy to use, and provide functions for detector design, software development and data analysis at different stages of an experiment.

Some popular geometry description tools in high energy physics have been studied, including TGeo [8], GDML and other XML-based geometry description tools. Since GDML offers all the JUNO experiment needs without any additional complexity introduced by more elaborate packages, we chose GDML to develop the geometry management system in JUNO offline software. The advantages of GDML are as follows:

(1) GDML provides an efficient way to use a common source of detector information for different programs. It is platform independent and programming language independent. A set of common attributes have been defined by GDML for general usage, meanwhile it also allows users to extend with their own definitions based on requirements.

(2) The JUNO detector simulation package is based on Geant4, while the reconstruction, visualization and analysis packages are heavily dependent on ROOT. The GDML format detector description can be automatically translated into the corresponding geometry models in Geant4 and ROOT with the existing Geant4–GDML interface and GDML–ROOT interface, which makes GDML very suitable for geometry management in JUNO offline.

(3) GDML has full support and maintenance from its development team. It has also been successfully applied in some previous HEP experiments, such as BESIII [9] and PHENIX [10].

### 3.2. Architecture

The JUNO offline software [11] is developed in the SNiPER (Software for Non-collider Physics ExpeRiment) framework [12]. SNiPER is a light-weight, efficient, flexible framework with dependencies on several external packages such as Geant4, ROOT, XML and Python, etc.

The geometry service is a module in JUNO offline software to provide detector information for all applications. As shown in Fig. 3, it is developed in the SNIPER framework and is based on GDML and ROOT geometry, which makes it possible to initialize the detector geometry from either GDML or ROOT. The geometry service has three sub-detector geometry classes (CDGeom, WPGeom, TTGeom) and a general geometry utility class to provide corresponding detector information for applications. It is designed to be a service type plug-in module in the framework so that it can be dynamically loaded by applications when necessary.

The design of the geometry management system in JUNO offline software is shown in Fig. 4. The geometry of all sub-detectors is first constructed in Geant4 detector construction with C++ coding. They are exported into GDML files automatically with the G4-GDML writer and serve as input of the geometry service. Meanwhile, the GDML files can also be converted into ROOT geometry objects to initialize the geometry service, construct the correlations between detector identifiers and corresponding objects, and provide the requested detector information for other applications including reconstruction, event display and analysis.

In the design of the geometry management system, it is an important feature to keep a consistent detector description between different applications, such as in simulation and reconstruction. With the automatic conversion interface between GDML and Geant4, GDML and ROOT, the consistency of the detector description between different application is guaranteed.