

# JUNO 探测器模拟软件

林韬

IHEP

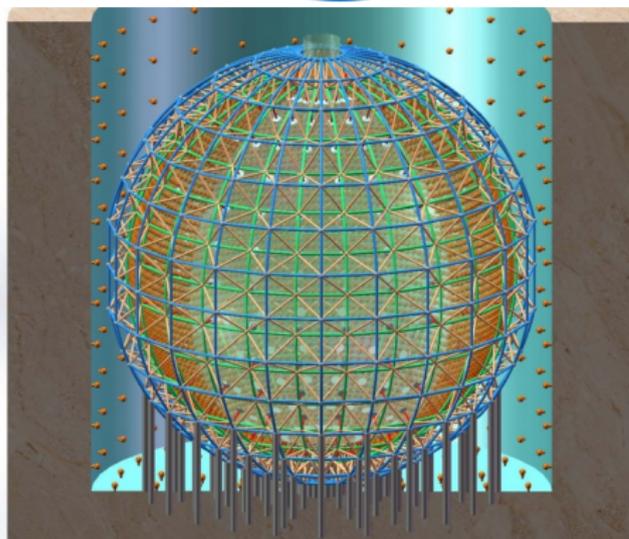
2015 年 8 月 18 日

第十七届全国科学计算与信息化会议暨智慧科研论坛

<http://indico.ihep.ac.cn/event/4928/>

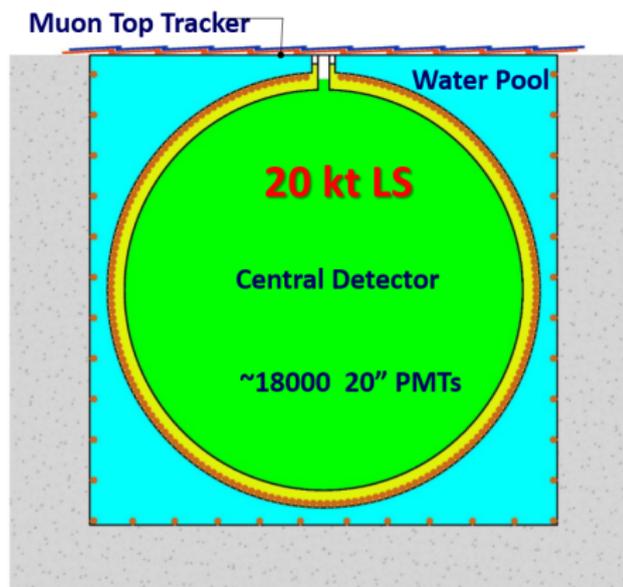
# OUTLINE

- 1 JUNO 实验及探测器简介
- 2 SNIKER 框架及数据处理流程
- 3 探测器模拟软件架构
- 4 JUNO 模拟软件的实现
- 5 结论



# JUNO 实验及探测器简介

- 江门中微子实验 (JUNO) 位于广东省江门市, 距离阳江和台山核电站 53 公里.
- 通过测量反应堆中微子振荡, 确定中微子质量顺序, 并测量中微子混合参数.
- 中心探测器 (Central Detector) 含有 2 万吨液体闪烁体, 约 18,000 个 20 英寸光电倍增管 (PMT), 能量分辨率可到  $3\%/\sqrt{E(\text{MeV})}$ .



外水池探测器 (Water Pool) 和顶部径迹探测器 (Top Tracker) 用于反符合本底事例.

# Offline Software Stack



Final User-Friendly Scripts

tut\_detsim.py

tut\_sim2calib.py

tut\_calib2rec.py

Generator

DetSim

ElecSim

WaveformFit

Rec

Analysis

Algorithms

Common Services

Root IO

Buffer Mgr

Data Model

Geometry Service

Random Service

Framework

SNiPER ( Software for Non-collider Physics Experiments )

CMT

SVN

CMake

Useful tools

Python

Boost

CLHEP

Geant4

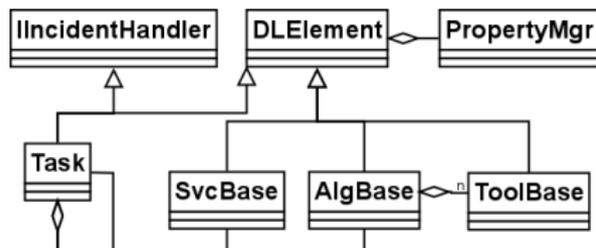
HepMC

ROOT

Common external libraries

# SNiPER (Software for Non-collider Physics ExpeRiments)

- SNiPER 是整个离线软件系统的基础.
- 设计遵循“弱耦合, 强内聚”原则.
- 模块间通过“接口”通讯.
- 重要的类
  - DLElement, 动态加载元素的基础类.
  - Task, 任务.
  - AlgBase, 算法.
  - SvcBase, 服务.
  - ToolBase, 工具.
- 这些组件都具有动态加载以及运行时配置的功能.



- 任务由算法以及子任务构成.
- 算法是针对特定阶段的数据处理过程.
- 服务则提供了公共的软件功能.
- 工具与服务类似, 但是由算法调度管理, 提供辅助功能.

## SNiPER 数据处理流程

- 创建顶层任务.
- 配置算法, 服务以及次级任务.
- 初始化服务, 算法.
- 运行顶层任务中的算法链.
- 完成算法, 服务的清理.

## JUNO 数据处理流程

- 产生子
- 探测器模拟
- 电子学/数字化模拟
- 波形重建
- 顶点/能量/径迹重建

# 产生子接口介绍 (I)

## 使用产生子接口的原因

- 高能物理实验中使用的产生子使用方式及输出格式并不相同。
- JUNO 的物理目标丰富, 含有多种产生子, 包括
  - 反  $\beta$  衰变产生子,
  - 天然放射性产生子,
  - 宇宙线产生子,
  - 刻度源产生子,
  - 基于 GENIE 的大气中微子产生子,
  - 以及超新星爆发产生子。
- 为了简化并统一最终用户的使用习惯, 借鉴了大亚湾中微子实验中 GenTools 的设计。

## 实现

- 设计思想: 根据流水线的工作机制, 使用符合统一接口的“工具”, 设置粒子的原初四动量信息, 产生位置, 时间等。
- 流程
  - 顶层算法创建 `HepMC::GenEvent` 对象, 依次传递给“工具”链。
  - 调用“工具”1, 设置粒子类型及四动量
  - 调用“工具”2, 设置粒子的位置信息
  - 调用“工具”3, 设置粒子的时间信息
  - ...

## 产生子接口介绍 (II)

- 粒子枪，设定多种原初粒子的四动量。
- 束流枪，设定粒子束在特定的面元上产生，产生的动量可以服从某一特定的分布。
- 位置选择工具，可以设置固定位置或者在物理体内均匀产生。
- 时间选择工具，对时间进行调整。例如放射性产生子衰变产生的次级粒子的时间是相对于母粒子而言，因此通过时间选择工具，将次级粒子的时间零点进行调整。
- HepEvt 格式转换工具，运行输出格式为 HepEvt 的独立产生子，将数据重定向到转换工具中，以此读入粒子信息。
- GST 转换工具，用于读取 GENIE 的 GST 格式的数据，
- Supernova 数据载入工具，此工具除了读入四动量外，还需要载入时间关联信息。
- GenDecay，从 NNDC (National Nuclear Data Center) 核数据库中读入核子的衰变信息，产生衰变的次级粒子。GenDecay 的原理是从母核开始构建完整的衰变链，计算每条衰变链所占的比率，最后根据该比率抽样得出所需的次级粒子。

## 为什么需要设计?

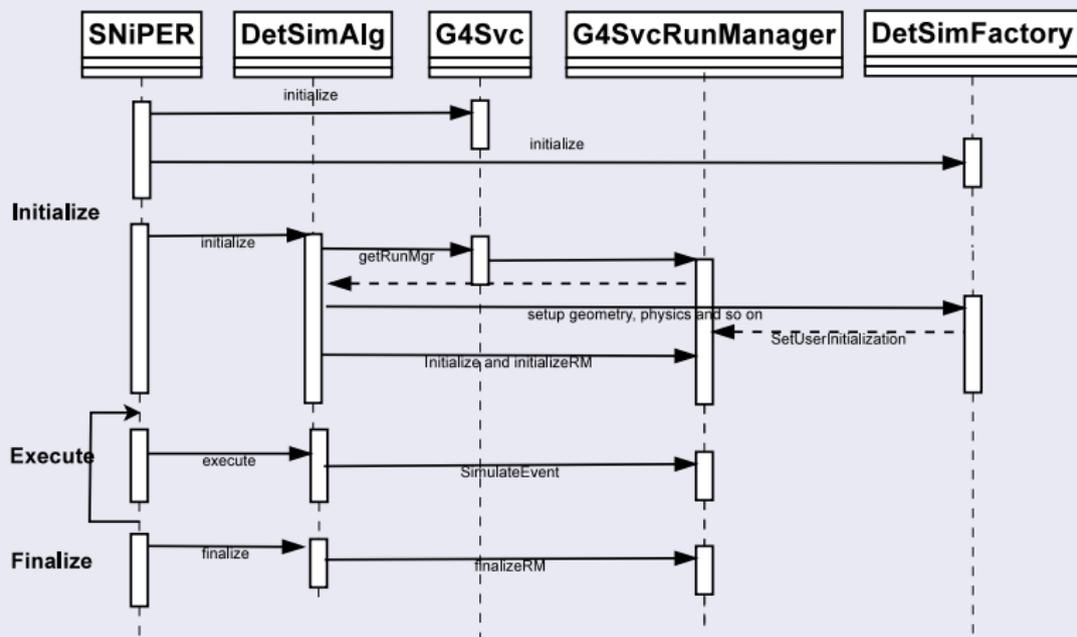
- 在独立的 Geant4 模拟程序中，探测器、物理过程、产生子、User Action 以及事例循环都由 G4RunManager 管理。
- 为了让程序能够在 SNIPEr 框架中运行，需要重新构建探测器模拟软件运行的流程。
- 同时，为了减少对 Geant4 源代码的修改，在实践中增加间接层，完成对整套软件的设计。
- 将模拟软件的设计分成两部分
  - 对 Geant4 的初始化.
  - 对 Geant4 的事例循环.

## 如何设计？

- 用于事例循环的算法 `DetSimAlg`。它与 `G4SvcRunManager` 以及 `IDetSimFactory` 的派生类共同协作，完成 `SNiPER` 与 `Geant4` 的连接。
- 与 `Geant4` 相关的 `G4SvcRunManager`。该类从 `Geant4` 的 `G4RunManager` 派生，修改了基类中事例循环的策略。由 `DetSimAlg` 转派调用。
- 提供获取 `G4SvcRunManager` 对象的 `G4Svc`。
- 与特定实验相关的 `IDetSimFactory`。该接口中定义了探测器构建、物理过程构建、产生子构建、`User Action`。

# 探测器模拟软件架构

## 如何运行?



# 探测器模拟软件架构

- 为了充分利用 SNIper 提供的特性, 包括动态加载以及运行时配置, 遵循 SNIper 的设计原则, 设计并实现了常用的接口.

## 重要的接口

- 随机数产生子服务, RandomSvc.
- 描述探测器逻辑体构建的 IDetElement.
- 描述逻辑体摆放的 IDetElementPos.
- 用于灵敏探测器的 ISensitiveDetectorElement.
- 用于模拟信息获取及分析的 IAnalysisElement.
- 针对 PMT 设计的 IPMTElement.

# 随机数产生子接口

- `RandomSvc` 用于控制一个作业的起始随机数种子数。
- 并且可以将每个事例运行前的状态保存下来。
- 同时, 也可以通过读入完整的随机数引擎的状态, 重复特定的一个事例。
- 当前默认的随机数引擎是 CLHEP 中的 `HepJamesRandom`。
- 用户在使用随机数产生子时, 可以使用随机数产生子接口, 也可以使用 CLHEP 库。因为目前底层的实现是一样的。

## 例子

- 初始的随机数种子一致, 可以保证一次作业的可重复性。
- 完整的随机数引擎状态一致, 可以保证重复特定的事例。
- 例如, 一次作业运行 10000 个事例, 可能需要重复第 9000 个事例。通过恢复运行 9000 个事例前随机数引擎的状态, 可以跳过前面的事例。

- 首先介绍原先独立的模拟程序对探测器不同方案的处理方式.
- 之后将介绍移入框架后的方式.

## 独立 Geant4 应用时的开发模式

- 由于在中心探测器设计阶段存在多种方案，因此针对每种探测器方案，都建立一套完整的拷贝，包括探测器构建、物理过程、产生子、灵敏探测器、User Action 等。
- 其中最主要的差异在探测器构建的代码中。通过建立完整的拷贝，不同方案的开发人员不会因为代码的修改而发生冲突。
- 但这里最突出的问题，就是**不一致性**。
  - 例如，在中心探测器中使用相同排布的光电倍增管，那么就需要在多份拷贝中同时修改相关的代码。
  - 这不仅增加了维护的成本，也增加不一致的概率。

## 移入 SNIPEr 框架的考虑

- 在将独立的模拟程序移入框架时, SNIPEr 框架的模块化设计为探测器模拟程序的设计提供了指导.
- 其基本的原则是, 将各个探测器组件进行模块化, 每个组件有具体实现, 各个组件通过接口互相结合起来, 最后完成探测器的构建。

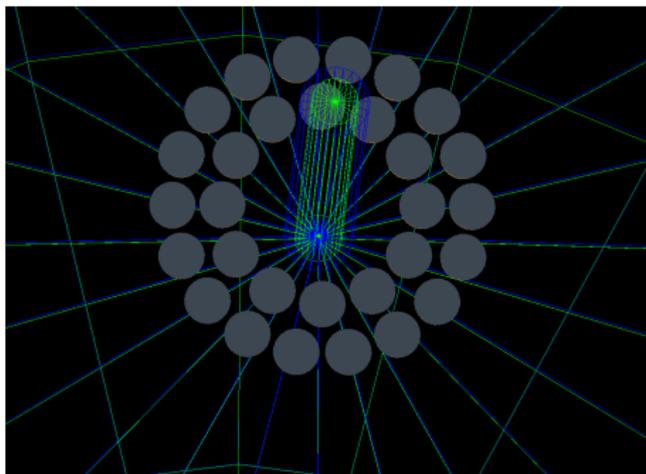
## 设计与实现

- 为了模块化各个探测器, 每个不同的探测器都从 `IDetElement` 中派生。
- 通过派生 `IDetElementPos` 设置子探测器模块的摆放。
- 在运行模拟程序时, 可以动态地配置是否需要将子探测器放入模拟中。
- 这利于刻度装置的选择与摆放、探测器子模块的调试。

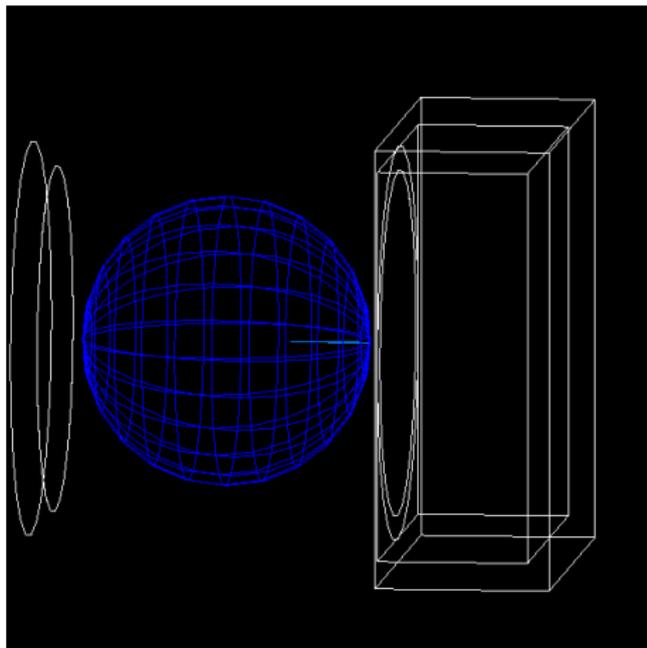
## 设计与实现

- 在模拟软件中，Central Detector、Water Pool、Top Tracker 和 Calib Unit 作为独立的模块存在。
- 它们都从 IDetElement 派生，实现构建逻辑体的代码。
- 探测器设计阶段，Central Detector 根据不同的方案有着不同的实现。由于在几何尺寸上是兼容的，不同方案的中心探测器就可以被放入相同的外水池中。
- 同时，在不同方案的中心探测器中，光电倍增管的摆放方式也可以复用。
- 通过模块化各个组成部分，使其可以独立变化，最后将各个组件通过脚本有机结合起来。
- 这在探测器方案的研究中，即保证了代码的灵活性，又保证了代码的一致性。

# Detector Element 介绍



运行时修改几何 (开启烟囱) 及载入的 PMT.



运行时加载刻度装置.

## 基于 GDML 构建探测器

除了针对特定的探测器实现不同的 DetElement，一个基于 GDML 构建探测器几何的类 GDMLDetElemConstruction 也被实现出来。它可以在运行时指定需要载入的 GDML 文件，然后通过 GDML 解析器，构建出 Geant4 的几何、逻辑体、物理体对象。该功能可以应用在刻度系统中，便于刻度装置的优化，而无需频繁修改源代码。

## 基于 GDML 构建材料属性

为了运行时构建材料，仍然采用 GDML 的构建方式，设计 GDMLMaterialBuilder 类。与前面的探测器构建非常相似，Material Builder 将会解析 GDML 中的元素及材料，然后构建出相应的材料。该功能可以用于对材料属性的优化，无需修改代码。

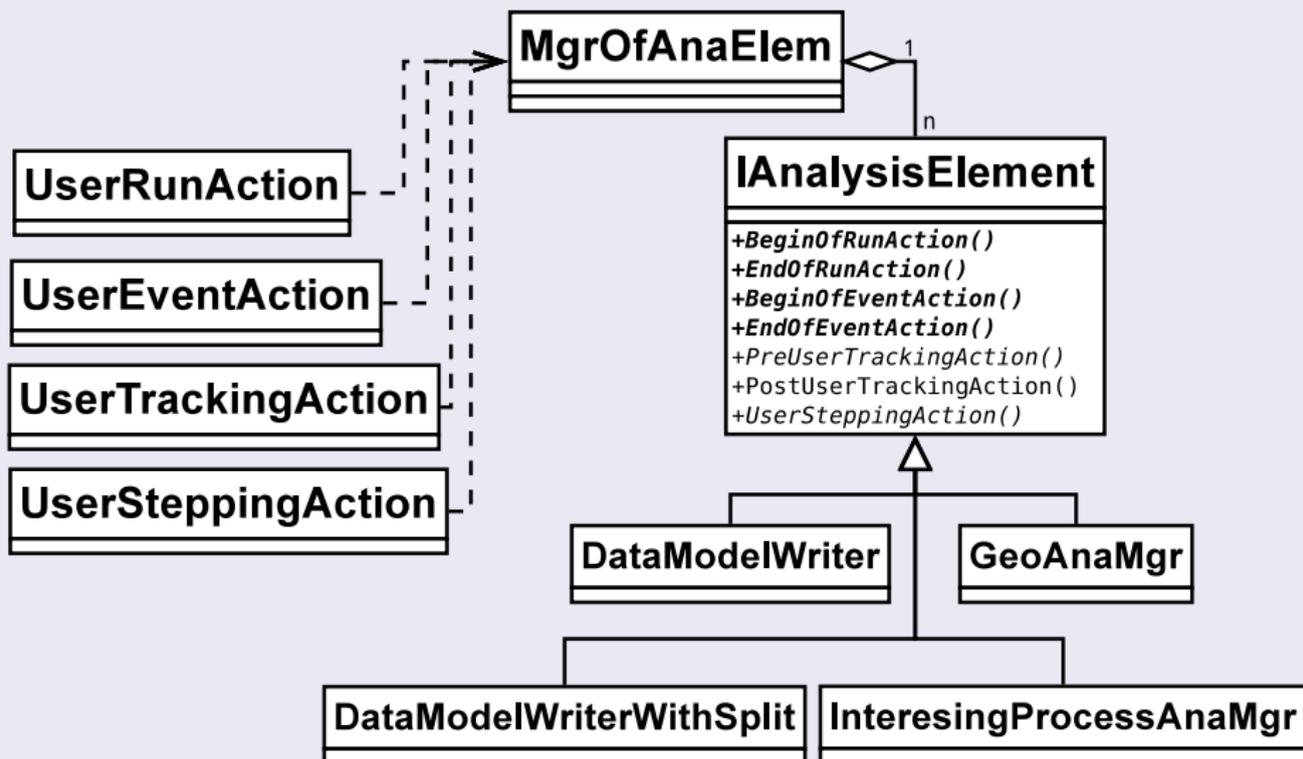
## 设计 Analysis Element 的原因

- 在 Geant4 中，为了提供用户与内核的信息交换，提供了 Run、Event、Tracking、Stepping 等级别的 User Action。
- 对于同一级别的 User Action 的操作都是在同一个类中实现的，这就不利于提高模块本身的聚合性。
- IAnalysisElement 则是对 Geant4 已有接口的重新封装。

## Analysis Element 的设计

- `IAnalysisElement` 的设计理念是，每个 `Analysis Element` 只做一个相关的事情，可以插拔。
- 这意味着调用一个特定的功能时，只需要插入这个 `Analysis Element` 即可，而无需改动代码的其他地方。
- 同时，`Analysis Element` 将不同级别的接口封装到一起，对于开发者也更加的便利。
  - 例如，开发者可以在 `Run` 开始时创建直方图，并将其关联到要输出的文件中，在 `Event` 级别对该直方图进行填充，因为是在同一个类中，直方图指针无需额外传递。

## Analysis Element 的类图



# Analysis Element 介绍

基于 Analysis Element 的设计，实现了多种实用的类。

## Analysis Element 的已有的实现

- DataModelWriter, 用于输出符合数据模型的数据。
- DataModelWriterWithSplit, 同样用于输出数据模型的数据, 但是使用拆分的方式, 将一个对象拆分输出。对于含有大量击中的宇宙线缪子事例, 尤其有用。
- DepositEnergyAnaMgr, 用于记录液闪中粒子的沉积能量。除此之外, 还有刻度系统、顶部径迹探测器对应的 AnaMgr。
- GenEvtInfoAnaMgr, 用于记录事例原初信息。
- GeoAnaMgr, 用于将 Geant4 的几何输出到文件中, 包括 GDML, ROOT, DAE。
- InterestingProcessAnaMgr, 用于挑选特定的物理过程, 例如中子俘获、米歇尔电子等。

# 光电倍增管的光学模型

对于江门中微子实验，光电倍增管的模拟非常重要。光电倍增管用于接收粒子在液闪中沉积能量后发出的光子。为了与真实的情况一致，需要考虑光电倍增管玻璃表面的折射反射、光阴极的量子效率、以及收集效率。该模型的几何构建分成 3 部分，水，玻璃以及真空。在玻璃和真空之间，放置一层“光学界面” (Optical Surface)，用于表示光阴极面。

## 模拟流程

- 光子从水中飞向 PMT，在水与玻璃的界面发生折射或者反射
- 进入玻璃的光子继续输运，飞到玻璃与真空之间的光学界面。
- 光子触发光学界面的模拟。在模拟中，该界面的反射率设为 0，让其 100% 吸收。
- 然后根据量子效率，决定该光子是吸收还是探测。
- 如果光子是被探测，在灵敏探测器部分，根据收集效率的不均匀性进行抽样。
- 如果被收集，产生一个击中。

# 中心探测器的光学模型

- 中心探测器是一个由液闪组成的探测器，精确的光学物理过程非常重要。
- 液闪的光学相关的属性也很重要，例如光产额，发射谱，快慢时间成分，散射长度，吸收长度以及重发射几率等。

## 重要的物理过程

- 闪烁光过程 (Scintillation)
- 契仑科夫光过程 (Cerenkov)
- 吸收过程 (Absorption)
- 吸收重发射过程 (Re-emission)
- 瑞利散射过程 (Rayleigh Scattering)
- 以及光学界面过程 (Boundary Process)

# 中心探测器的光学模型

- 当前的光学模型简化了液闪的属性，所得的结果是平均后的。
- 新的光学模型则考虑了液闪的具体组分，包括 LAB (溶剂)、bisMSB (波长位移剂) 以及 PPO (第一发光物质)。
- 使用新的光学模型可以用于对液闪组分的调节，寻找最优的比例。

## 新的光学模型模拟

- 产生时的发射谱按照 PPO 抽样
- 吸收的概率按照 LAB、bisMSB 和 PPO 抽样，这里表示光子可以被其中一个组分吸收
- 吸收后，可以根据吸收组分的重发射概率谱，计算重发射的概率，如果重发射，则再次抽取该组分的发射谱。

- 整合了 Geant4 与 SNIper 的探测器模拟软件已经正常运行。
  - 产生子, 探测器模拟软件都在 SNIper 框架中运行.
  - 探测器的构建, 数据的分析相比原先独立的程序, 更加灵活.
- 但是, 大型的液闪探测器也使我们的软件面临多方面的挑战. 例如宇宙线事例模拟:
  - 运行时间,
  - 占用内存,
  - 文件输出
- 下一步工作:
  - 考虑多线程, 多核, 及 GPU 计算等技术.
  - 研究快速模拟的方法.

- ① JUNO 合作组, Neutrino Physics with JUNO, arxiv:1507.05613
- ② Geant4 合作组, Geant4 - A Simulation Toolkit, Nuclear Instruments and Methods A 506 (2003) 250-303
- ③ SNIPEr, <http://gitlab.ihep.ac.cn/zoujh/sniper>
- ④ Rene Brun and Fons Rademakers, ROOT - An Object Oriented Data Analysis Framework, Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86.
- ⑤ C.Andreopoulos *et al.*, The GENIE Neutrino Monte Carlo Generator, Nucl. Instrum. Meth. A 614:87-104,2010.
- ⑥ M. Dobbs and J.B. Hansen, Comput. Phys. Commun. 134 (2001) 41.
- ⑦ 邓子艳, BESIII 探测器模拟软件的设计和开发, 中科院博士论文
- ⑧ CLHEP, <http://proj-clhep.web.cern.ch/proj-clhep/index.html>
- ⑨ R. Chytracsek *et al.*, Geometry Description Markup Language for Physics Simulation and Analysis Applications, IEEE Trans. Nucl. Sci., Vol. 53, Issue: 5, Part 2, 2892-2896
- ⑩ COLLADA, <https://www.khronos.org/collada/>

# Q & A