



山东大学
SHANDONG UNIVERSITY



高能宇宙辐射探测设施(HERD)离线数据处理软件

Offline data processing software for the High Energy cosmic-Radiation Detection facility

汇报人：时倩倩

On behalf of the HERD offline software team

2024/8/15

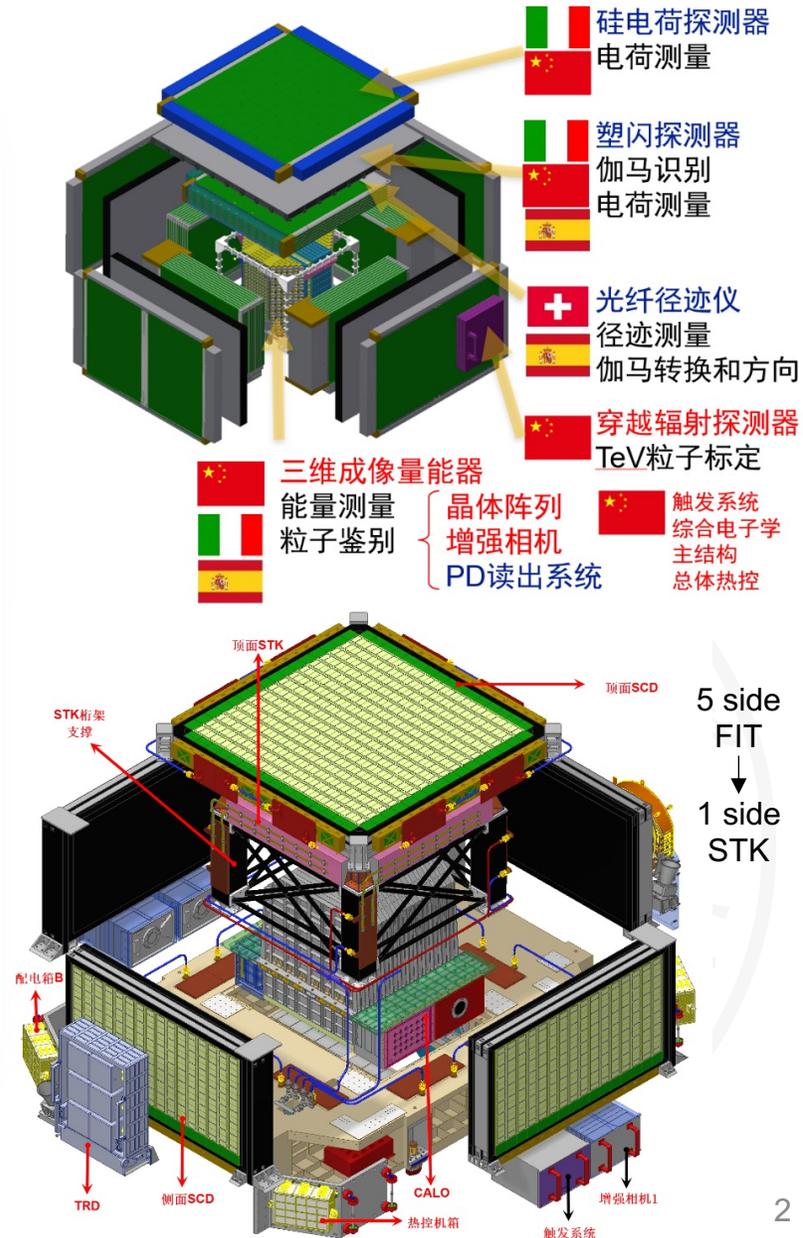
2024-08-15

1

高能宇宙辐射探测设施(HERD)

High Energy cosmic-Radiation Detection facility

- ❖ HERD实验是计划2027年发射安装在中国空间站上的空间天文和粒子天体物理实验，计划运行时间十年以上。
- ❖ 其主要科学目标是宇宙线电子能谱精细测量及高灵敏度暗物质信号搜寻，探究宇宙线起源和高灵敏度的高能伽马射线巡天观测。
- ❖ HERD探测器包含五个子探测器：Calo(三维成像量能器)、FIT(光纤径迹仪)、PSD(塑料闪烁体探测器)、SCD(硅电荷探测器)、TRD(穿越辐射探测器)。
- ❖ 作为下一代空间粒子物理探测器，HERD的科学能力将长时间保持大幅度国际领先，将成为中国空间站标志性的旗舰级重大科学实验和具有重大影响的大型国际合作项目。

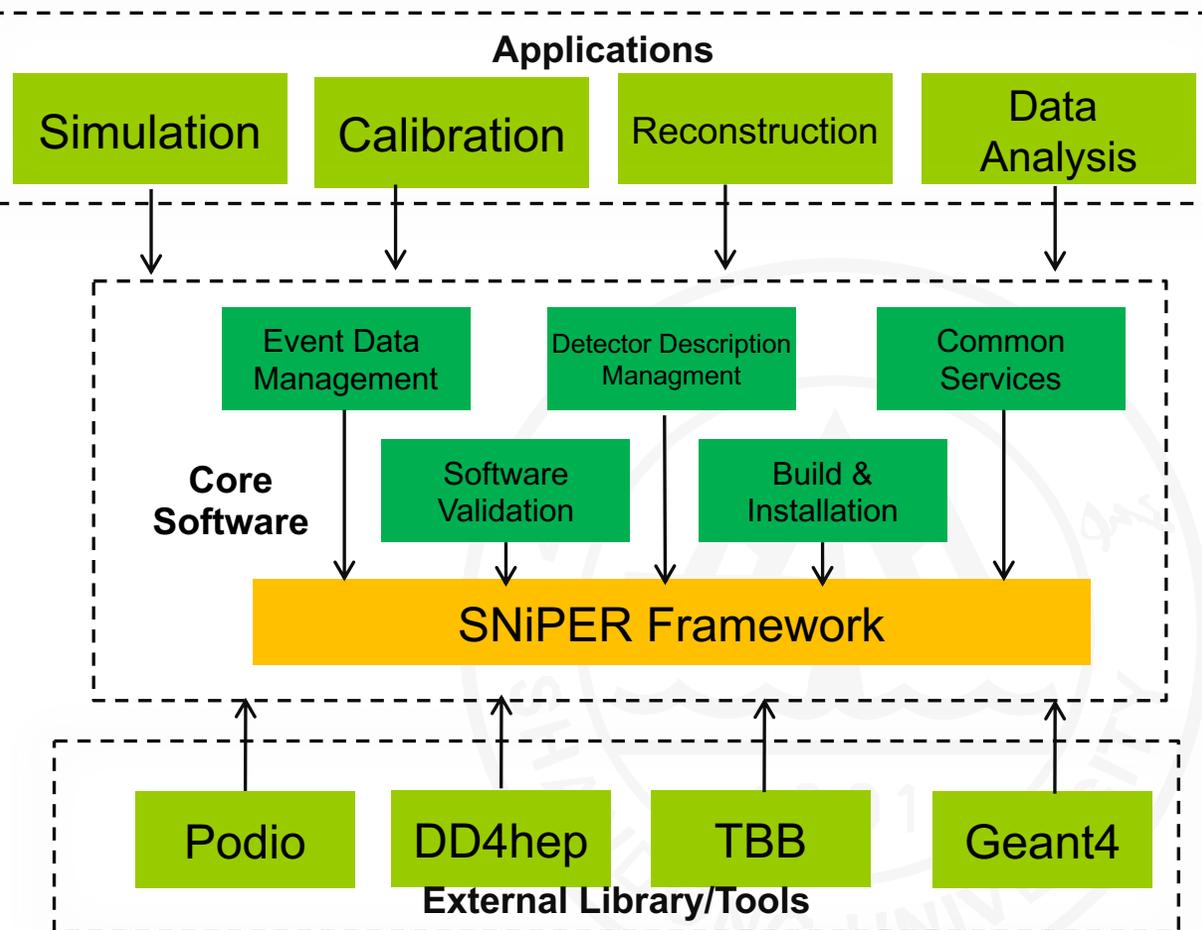


HERD离线软件系统的整体架构

Overview of HERD Offline Software System

- ❖ HERD离线软件(HERDOS)是整个实验的重要组成部分，主要负责离线数据的处理包括MC模拟、刻度、重建和物理分析等，并为用户提供一个共同的平台，方便开发和执行分析任务。
- ❖ 应用软件基于核心软件开发，并且很大程度上依赖于核心软件提供的功能。
- ❖ 核心软件提供应用软件所需要的公共核心功能和关键技术：

- 底层框架
- 事例数据管理系统
- 几何数据管理系统
- 事例显示软件
- 几何模拟接口
- 通用服务
- 并行计算和机器学习
-



底层框架：SNiPER

Underlying Framework: SNiPER

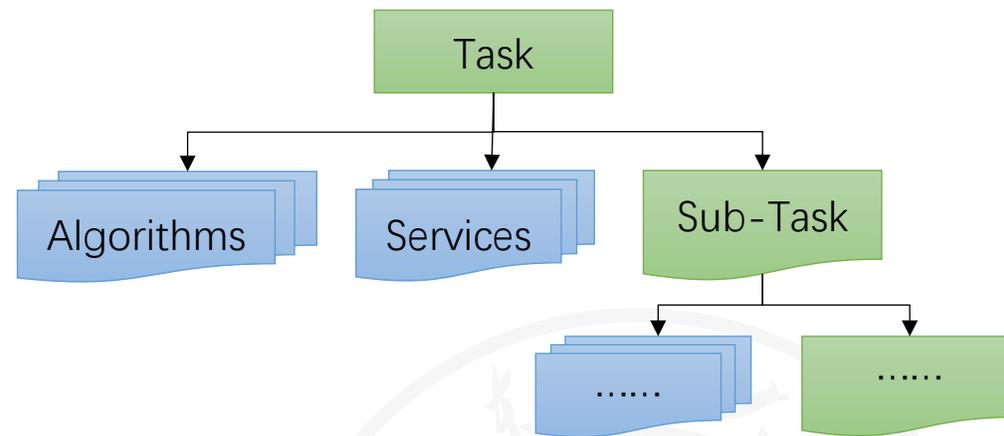
❖ 高能物理实验领域较为流行的通用软件框架有Gaudi、Marlin等，为加速器设计，广泛应用于LHCb、ATLAS、BESIII等实验。

❖ Software for non-collider experiments (SNiPER)

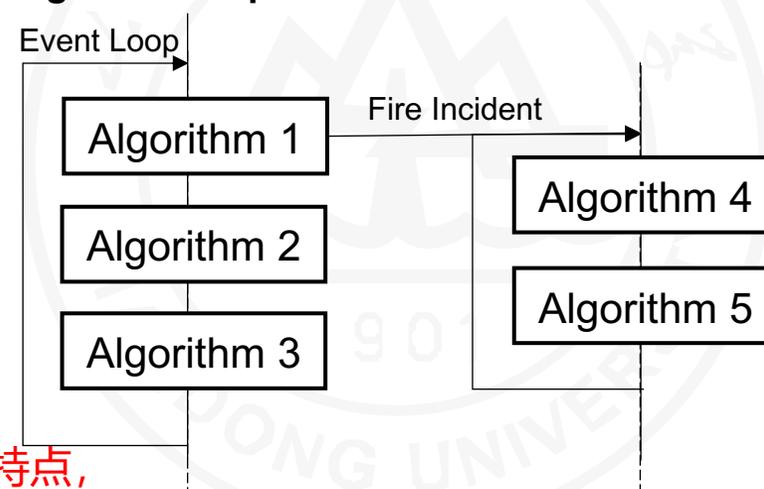
- 为非加速器实验设计和开发 (2012)
- 应用于JUNO、LHAASO、nEXO、STCF等
- 提供数据处理的基本功能：事例循环、应用接口、作业配置等等

❖ SNiPER的优势

- 轻量级，最小程度依赖外部库
- 模块化，高内聚低耦合
- 灵活的事例循环控制
- 灵活的与其他软件整合，如podio, ROOT, ...
- 利用Python Binding, C++和python的混合编程
- 多Task, 支持多线程



An algorithm sequence in Task Sub-Task



结合HERD实验运行时间长、数据量大和数据处理任务复杂度高的特点，HERDOS选择SNiPER作为底层框架。

多线程需求

Requirement for Multithreading

❖ HERD实验对于高能量 (\sim PeV) 重原子核的模拟会耗费大量的时间和内存等计算资源:

- 模拟1个 \sim 1PeV的proton花费大概5小时
- 模拟1个 \sim 3Pev的helion花费大概20小时
- 巨大的内存消耗, 导致作业被杀死

❖ 采用并行计算技术模拟高能重原子核:

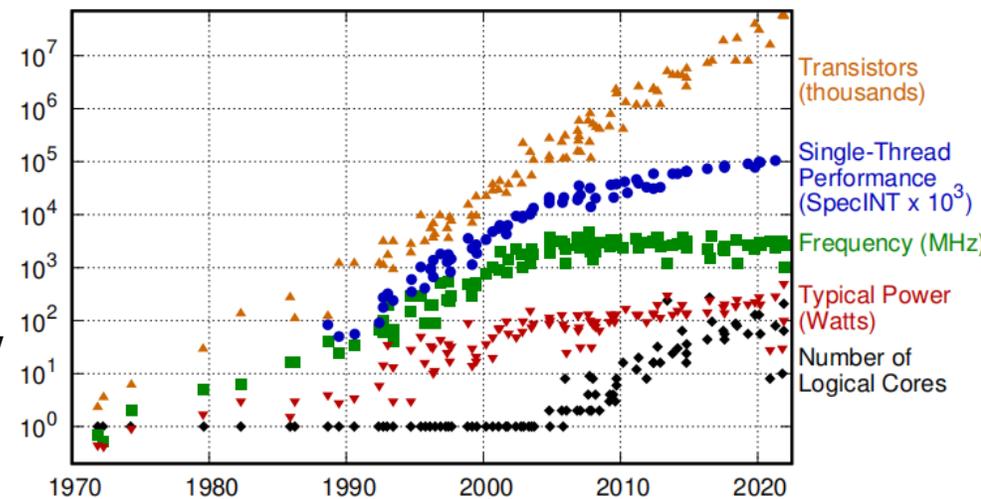
- 减少模拟重粒子所消耗的时间
- 通过在内存中共享一些对象降低总的内存消耗:

Geometry(TGeoManager costs \sim 1GB memory), physics list, common services, I/O buffer ...

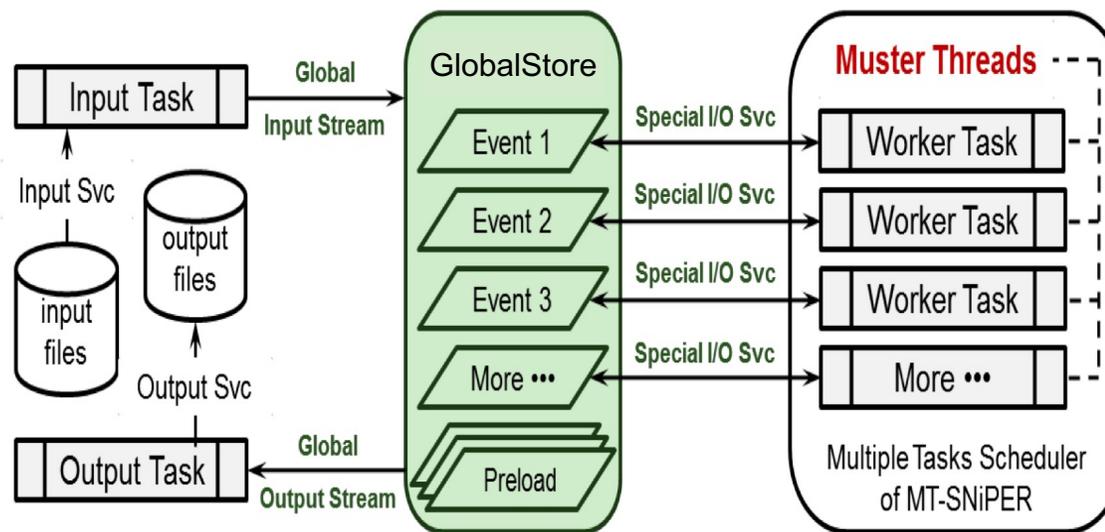
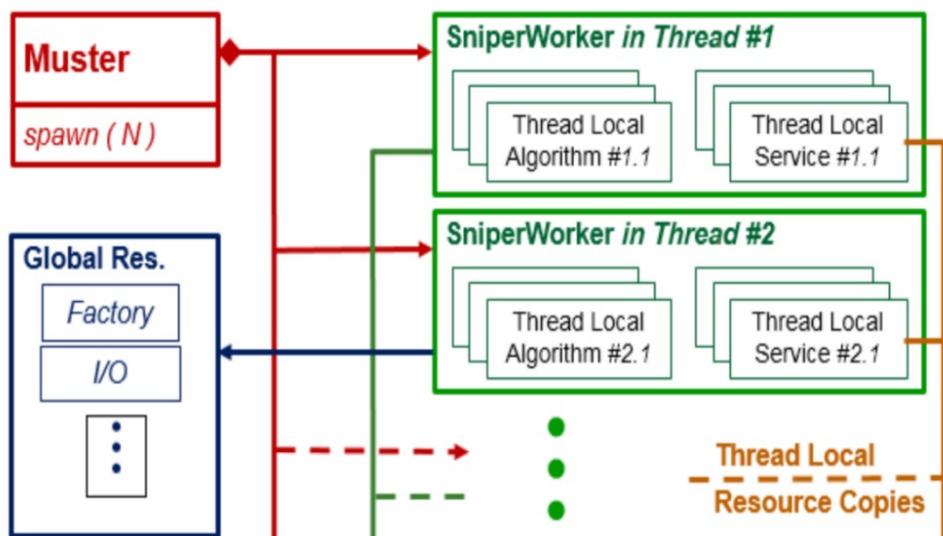
❖ 应用多级的并行模拟算法:

- **Event-level** (事例之间) : 多个线程并行独立地处理多个物理事例
- **Track-level** (事例内部) : 多个线程同时处理一个事例 (比如: 多个线程并行模拟单个物理事例内的多条径迹)

50 Years of Microprocessor Trend Data

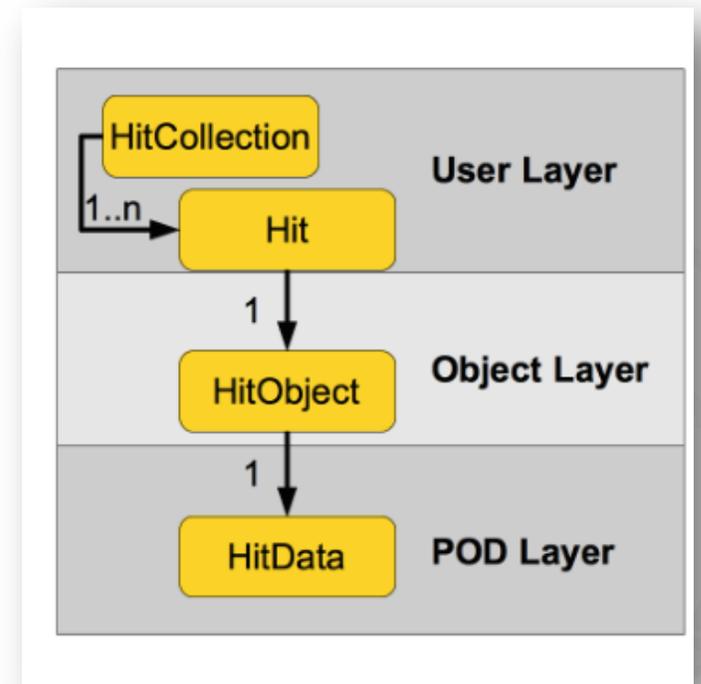


- ❖ SNiPER为构建多线程应用程序提供了非常简单的接口
 - ✓ 基于Intel TBB开发**Muster**（SNiPER多任务调度器）作为调度器
 - ✓ 开发**GlobalStore**支持并行地事例数据管理
 - ✓ 数据的输入输出服务放置在两个独立线程中以保证数据输入输出的性能和灵活性
 - ✓ Worker线程作为SNiPER Task副本保证应用代码串行与并行执行基本一致



Event Data Model Based on Podio

- ❖ Event Data Model (EDM) 是软件框架的重要组成部分：
 - 定义物理事例数据在内存和数据文件中的格式
 - 构建不同数据对象之间的关联关系（重建径迹与模拟粒子入射径迹、tracks与hits）
- ❖ HERDOS选择Podio作为EDM定义的工具
- ❖ Podio：新兴的定义和管理数据模型的C++工具
 - 基于YAML文件自动产生C++代码
 - 支持多种数据格式：ROOT，提供C++和Python接口
 - 支持多线程
 - 处理EDM对象之间的相互关联关系



目前官方已经设计出一套完整的EDM，但用户可以根据需求自己定义扩展

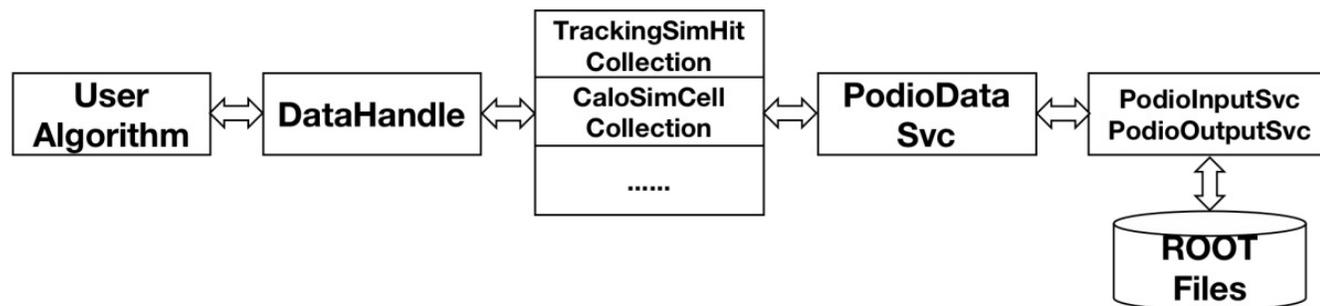
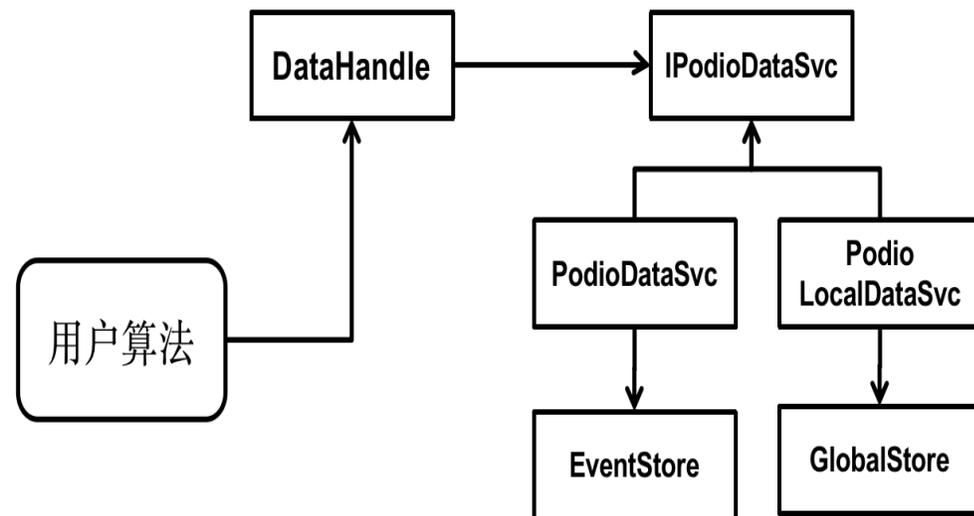
Event Data Management

❖ Event Data Management System负责管理数据在内存中的布局 and 事例数据模型的生命周期，提供读取和存放数据的接口，并实现数据文件的输入输出等功能。

❖ 基于Podio扩展SNIKER的数据管理系统

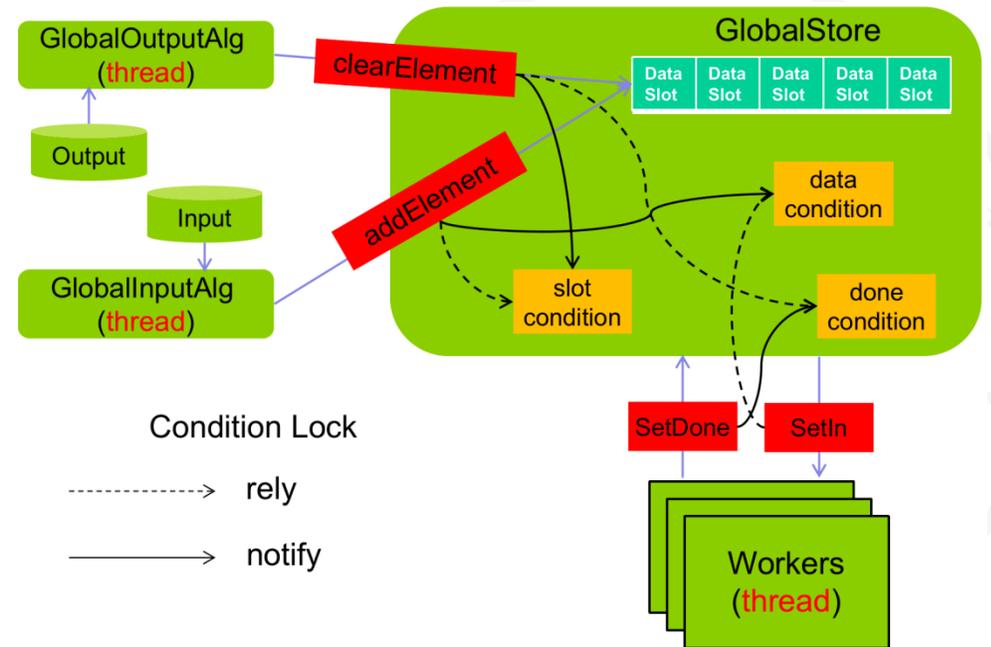
- PodioDataSvc: 管理podio::EventStore (串行)
- PodioLocalDataSvc: 访问GlobalStore (并行)
- PodioInputSvc: 数据输入
- PodioOutputSvc: 数据输出
- **DataHandle**: 用户获取数据接口

❖ 事例数据与用户应用完全解耦，开发者仅需要通过DataHandle访问内存中的数据。



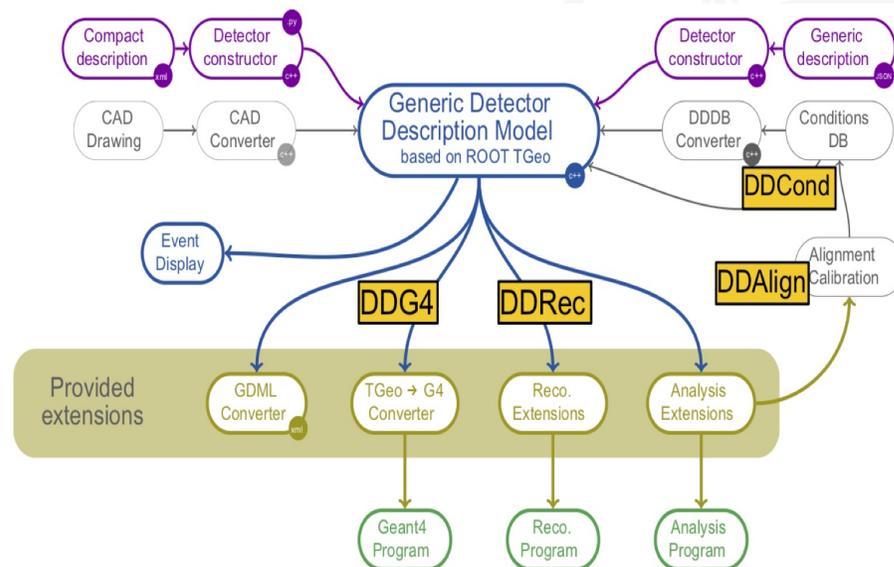
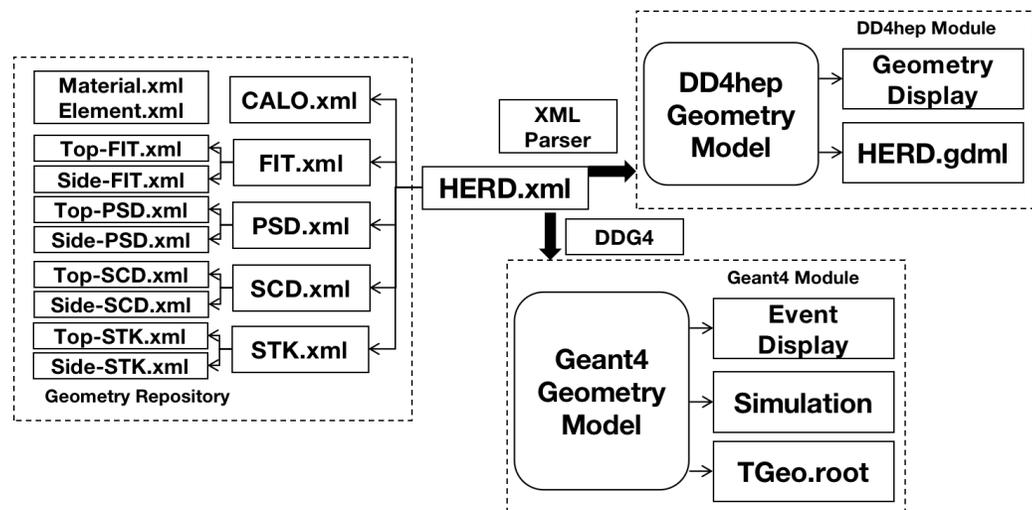
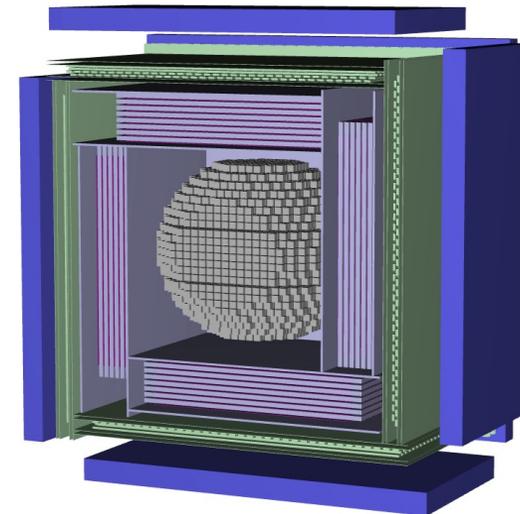
Parallelized Event Data Management

- ❖ 为了支持并行的数据处理，基于Podio开发和实现了全局内存池GlobalStore
 - 重现实现podio::EventStore同时缓存多个物理事例对应的数据
 - 输入输出服务与两个专用的线程绑定在一起从而确保性能和灵活性
 - 使用多个condition lock来保证线程之间复杂的数据交换的安全
- ❖ 基于并行化的事例数据管理系统，并行的探测器模拟和重建也可以实现
- ❖ 用户仅需通过简单更改作业的配置就可以实现串行或者并行的选择



Detector Geometry Description

- ❖ 在HERDOS中探测器几何管理系统基于DD4hep实现
 - 使用XML文件定义几何，采用TGeo对象作为统一的内存存储格式，提供多个插件
- ❖ HERD的全探测器几何与束流测试几何由XML文件定义
 - 各子探测器定义在不同的XML文件中，且具有独立的版本方案
 - 根据版本号选择实现版本切换，无需编译
 - Elements, materials定义为共用的文件，与其他子探测器文件组合构建成完成探测器模型
 - 复杂的探测器几何（包括空间站）通过CAD格式实现



❖ 实现 **GeometrySvc**, 为各种应用提供获取各种格式的探测器几何描述信息:

- 不同几何格式的转换 (XML, CAD, Geant4, ROOT, GDML, ...)
- 全局坐标和局部坐标的获取与相互转换 (cellID)
- 编码方式的转化 (cellID, volumeID, cellcode)
- 所有探测器尺寸和位置的计算
- 获取physical volume, placed volume, logical volume 的接口
- 计算粒子经过physical volumes的径迹长度

❖ 这些功能在模拟、数字化和重建算法中得到积极地应用

```
// Get geant4 geometry information
dd4hep::sim::Geant4GeometryInfo* getGeoInfo();
// Get geant4 physical volume
G4VPhysicalVolume* getPhyVol();
// Get geant4 magnetic field
G4MagneticField* getMagField();
// Get dd4hep detector instance
dd4hep::Detector* getDetDesc();

// Get the global position of cell by its volumeid
dd4hep::Position getPosition(dd4hep::VolumeID &volId);
// Get the global position of cell by its cellcode and systemid
dd4hep::Position getPosition(SubDetector systemId, int cellcode);

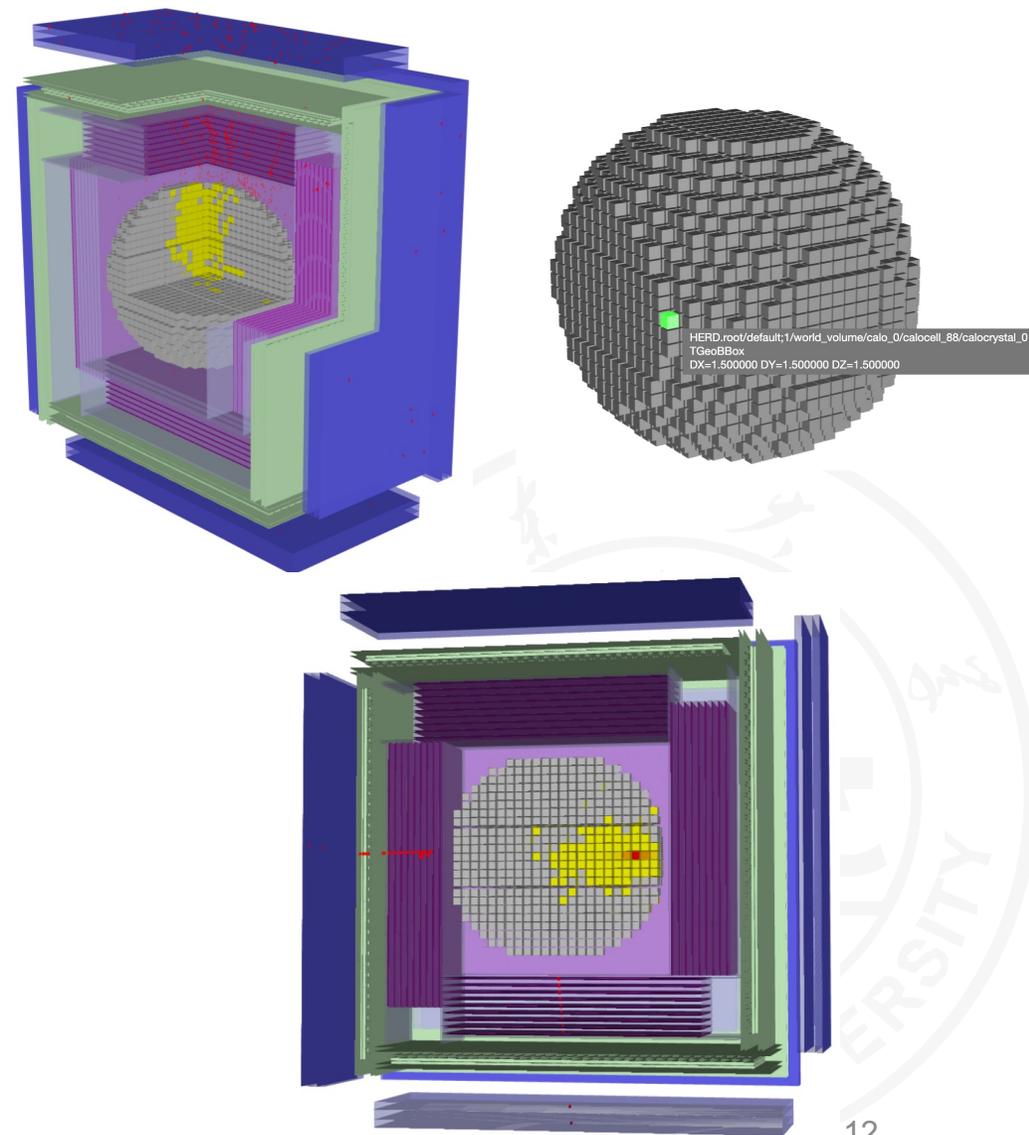
// Get the dimensions of cell by its volumeid
std::vector<double> dimension(dd4hep::VolumeID &volId);
// Get the dimensions of cell by its systemid and cellcode
std::vector<double> dimension(SubDetector systemId, int cellcode);

// Get the physical node of cell by its volumeid
TGeoPhysicalNode *getPhyNode(dd4hep::VolumeID &volId);

// Transform from world coordinates to local ones at giving level
dd4hep::Position globalToLocal(const dd4hep::Position &global, int level=-1);
// Transform a point from local coordinates of a given level to global coordinates
dd4hep::Position localToGlobal(const dd4hep::Position &local, dd4hep::VolumeID &volId, int level=-1);
```

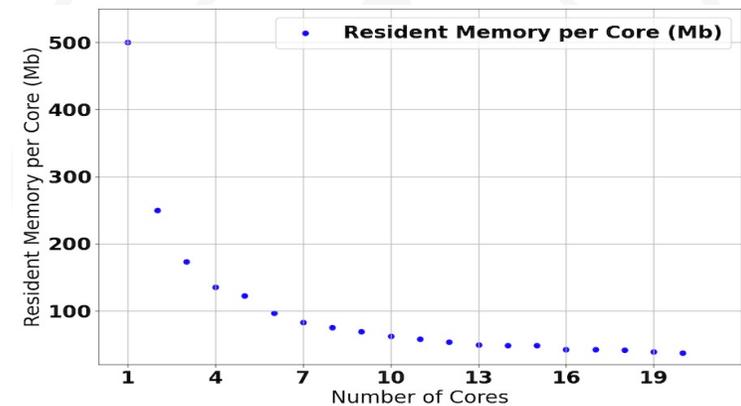
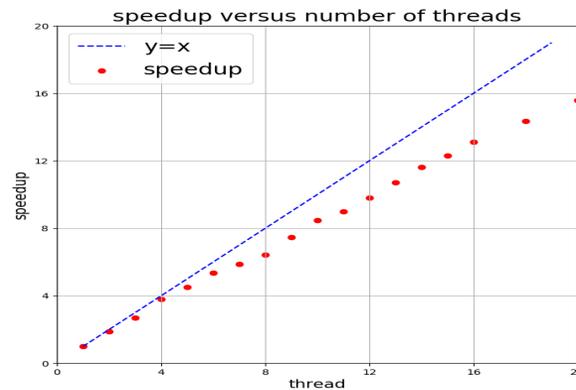
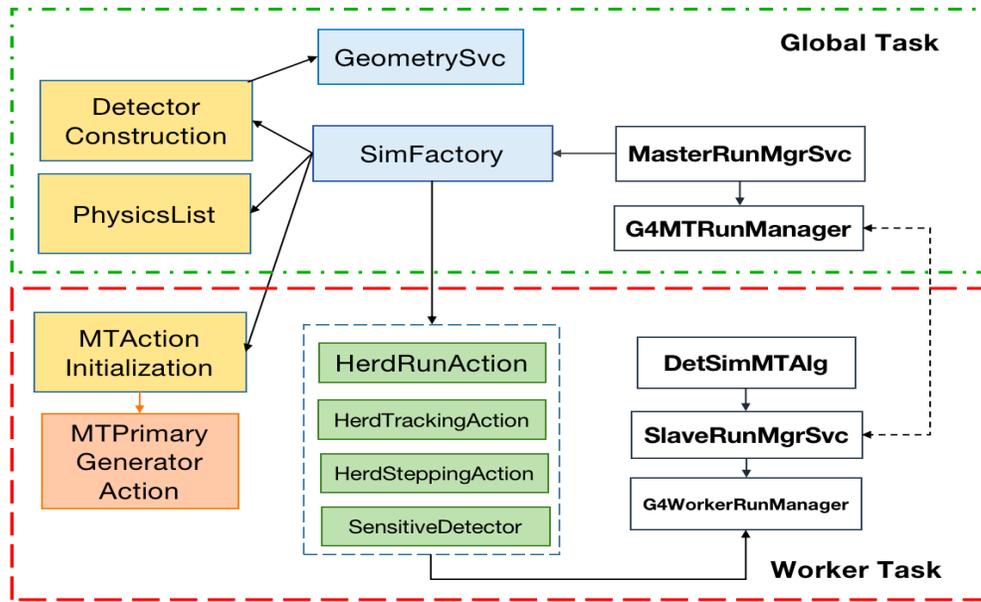
Detector and Event Display

- ❖ HERDOS开发了可视化软件(HERDEvE)
 - 基于WebGL技术实现用户界面和开源的JSROOT解析
 - 基于Three.JS的3D引擎和图形库
 - 使用Vue.js HTML5 开发框架实现Web界面
 - 通过Web Worker framework的多线程能力减少3D延迟
 - 几何信息基于DD4hep定义的几何描述 (XML) , 事例数据从podio中获取
 - 读取离线数据文件, 之后接收在线系统实时传输的数据



Parallelized Detector Simulation

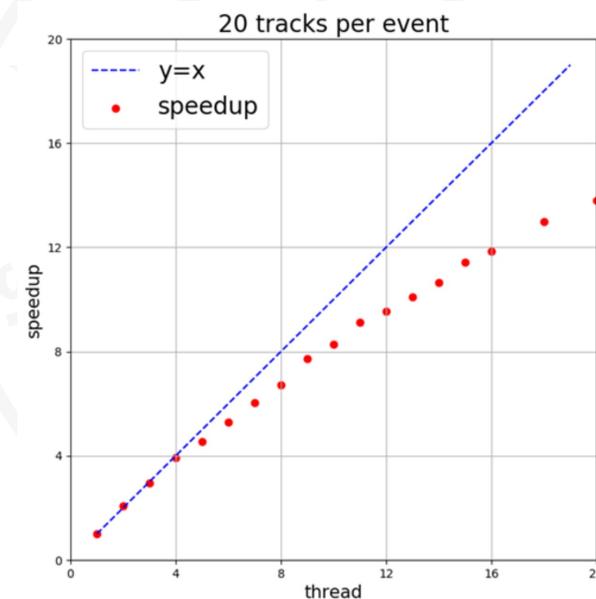
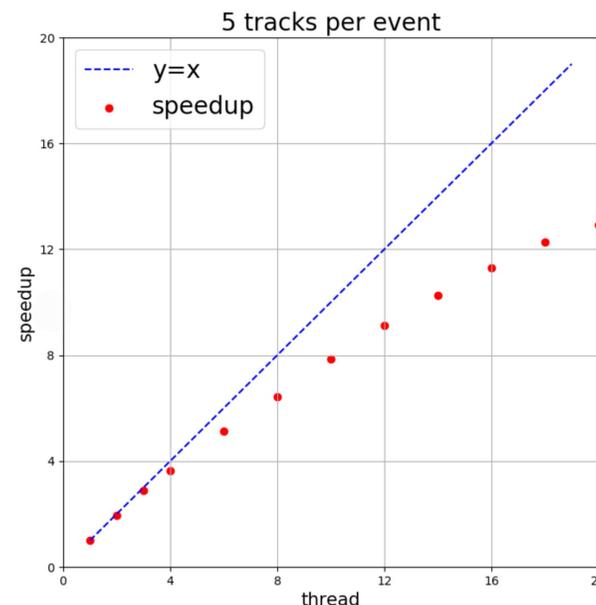
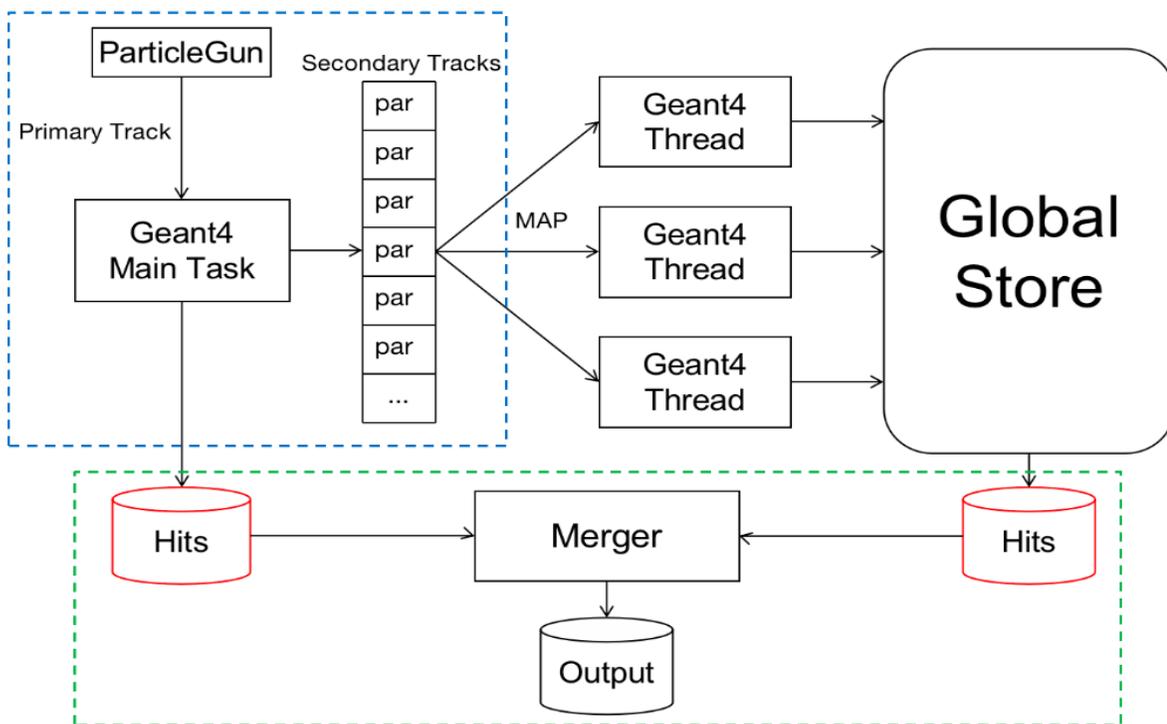
- ❖ HERDOS整合SNIper, DD4hep, Geant4和Podio提供了统一的探测器模拟接口。
- ❖ 基于SNIper实现模块化设计, 保证开发工作互不干扰
- ❖ 基于MT-SNIper和并行化的数据管理系统, HERDOS开发了事例级别的并行模拟, 最大程度地复用串行模式中实现的模块
 - 在多个线程同时模拟多个event
 - 基本的性能测试展示出了良好的可扩展性



Track-level

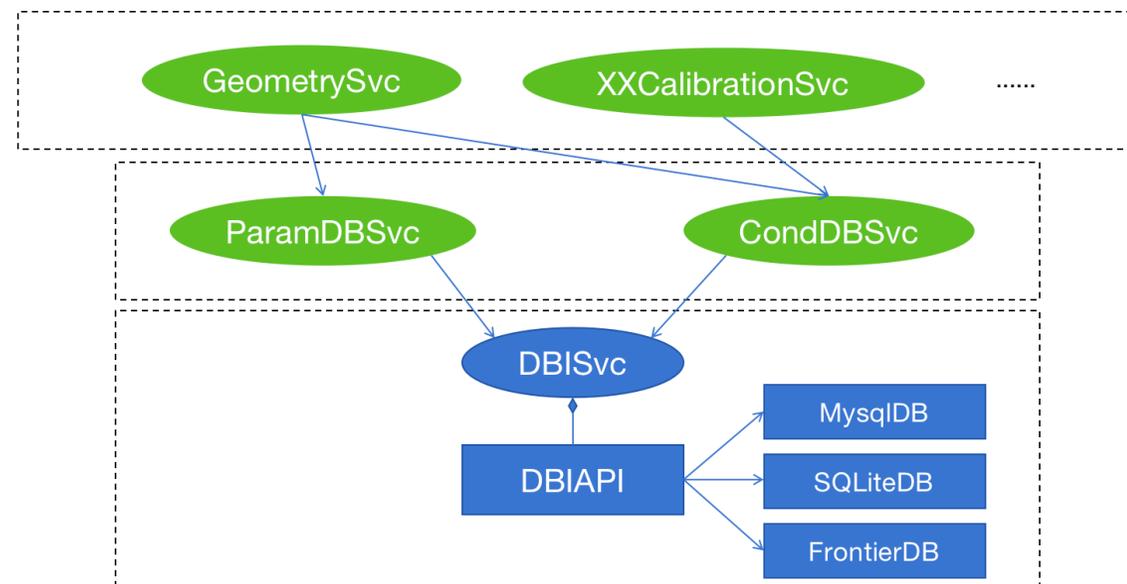
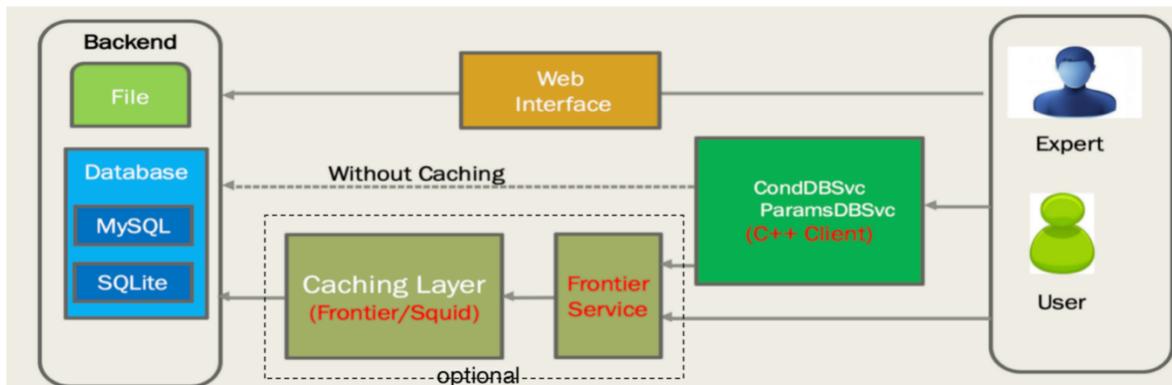
❖ 为应对超高能粒子，需要实现事例内部的并行模拟，即将单个event产生的次级粒子分发到不同的工作线程中模拟：

- 在主线程模拟primary particle（输入线程->Geant4模拟线程）
- Secondary particles被中止缓存后分发到工作线程中
- 所有track模拟完成之后，模拟击中被合并写出到输出文件中



❖ 整体设计

- 网页接口, C++客户端, Frontier服务



❖ 数据库服务接口

- **DBIAPI**为不同的数据库后端提供统一的接口(Frontier, MySQL, SQLite)
- **DBISvc**提供数据库连接和访问接口
- **ParaDBSvc**提供参数数据访问接口, 提供静态参数
- **CondDBSvc**提供状态数据访问接口, 提供随时间变化的状态参数(如探测器刻度和Alignment参数等)
- **GeometrySvc, CalibrationSvc, AlignmentSvc**等为应用服务提供具体参数

- ❖ 使用GlobalTag、Tag和IOV等字段实现刻度参数的长期有序管理

- ❖ 介绍了HERD离线软件系统(HERDOS)的基本设计和功能
 - 部分基于key4hep开发
 - 许多组件专为HERD实验开发, 但具有通用性可应用于其他实验
- ❖ HERDOS的设计与实现细节
 - 底层框架的选择: SNiPER
 - 基于podio设计的事例数据模型(EDM)和整合SNiPER及podio实现的数据管理系统(DMS)
 - 基于SNiPER和TBB的并行化数据管理系统, 重实现GlobalStore
 - 基于DD4hep开发的几何管理系统: 一致的几何描述、几何信息获取以及几何和事例显示
 - 开发了探测器模拟框架, 包括串行、并行模拟。
 - 两种模式并行模拟选择: event-level, track-level
 - 开发了数据库系统管理状态数据和参数
- ❖ HERDOS在探测器设计等方面实现有效运行, 同时在物理研究上也有着巨大潜力

为天下储人才
为国家图富强

感谢各位的聆听

Thank You For Your Listening

2024年08月15日