MESHCHERYAKOV LABORATORY of INFORMATION TECHNOLOGIES

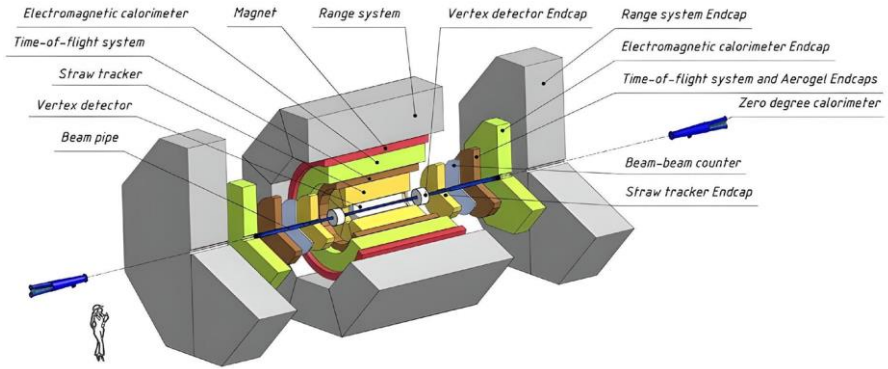
SPD NICA

Система управления нагрузкой специализированной вычислительной системы SPD On-line Filter

Гребень Никита
стажер-исследователь, НТО ВКиРИС

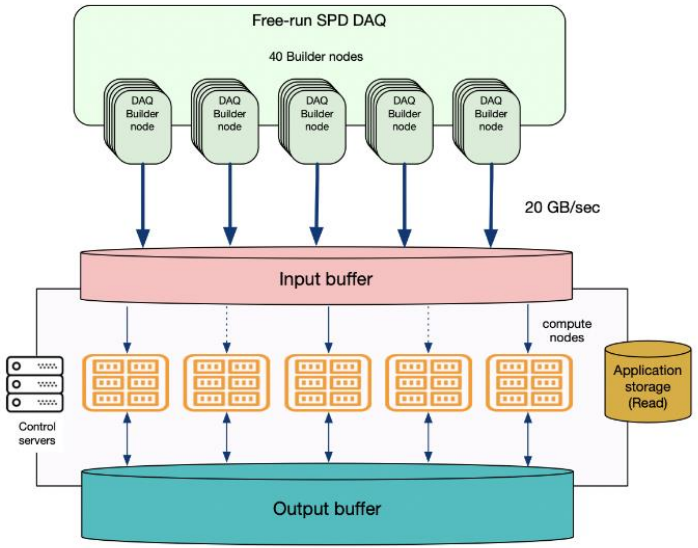
Эксперимент SPD

- Количество регистрирующих каналов в SPD ~ 500000 – 700000
- ~ 3 MHz ожидаемая частота событий (на максимальной светимости)
- Безтритггерная система снятия данных
  - ~ 20 GB/s (200PB//"год") “сырых” данных



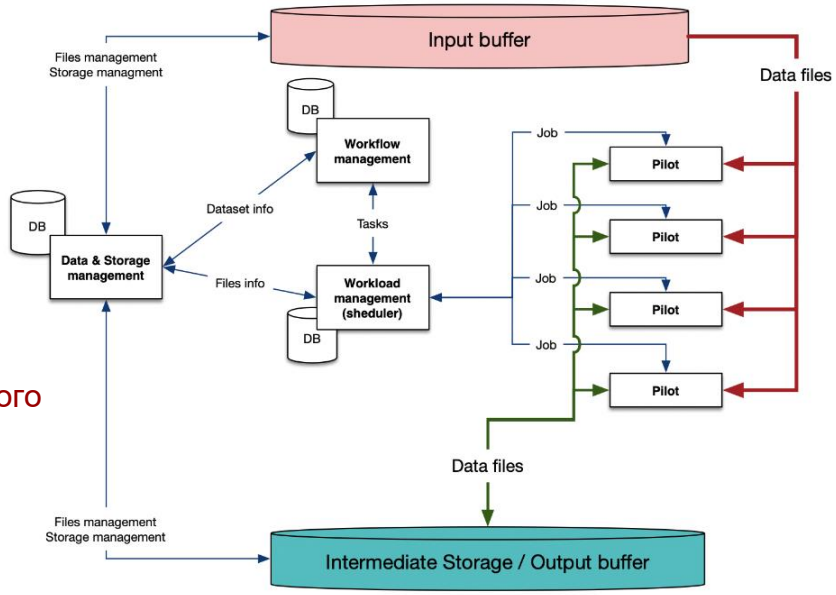
SPD OnLine Filter – высокопропускная вычислительная система

- Тип вычислений, при котором одновременно выполняется множество простых и независимых друг от друга вычислительных задач для выполнения задания по обработке данных.
- Поскольку каждый элемент данных может быть обработан одновременно, это можно применять к данным, агрегированным системой сбора данных (DAQ).
- Обработка данных должна быть многоэтапной:
  - Один этап обработки → Задание/Task
  - Обработка блока данных (файла) → Задача/Job



SPD OnLine Filter – промежуточное программное обеспечение

- **Data management system**
  - каталогизация данных и организация данных
- **Workflow Management System:**
  - определение и выполнение цепочек обработки путем генерации необходимого количества вычислительных заданий
- **Workload management system:**
  - Генерация необходимого количества задач для выполнения одного задания;
  - Отправка задач на рабочие узлы посредством пилотного приложения;
  - Контроль выполнения задач;
  - Контроль пилотов (выявление "мертвых" пилотов);
  - Эффективное управление ресурсами



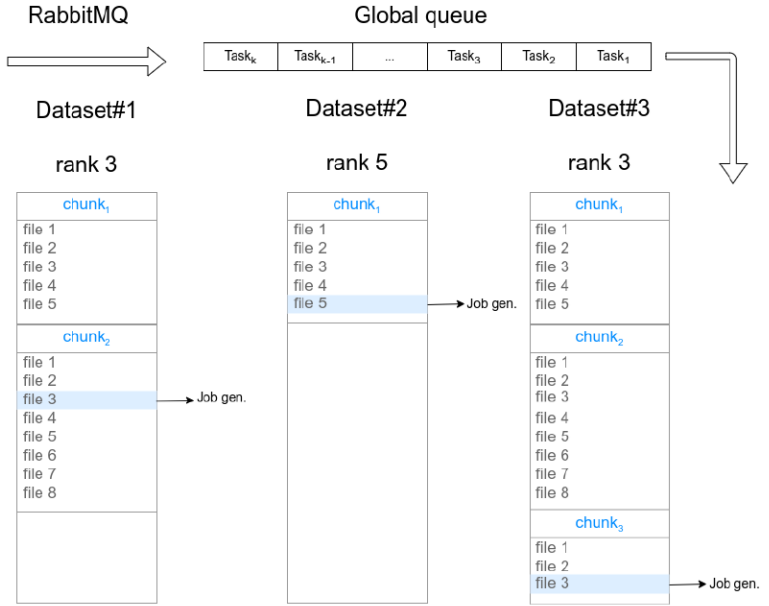
Определение заданий/задач

- Задание - единица рабочей нагрузки, отвечающая за обработку блока однородных данных.
- Запрос на обработку - это набор входных данных, который может состоять из множества файлов, и обработчик.
- Критерием завершения задания является обработка блока данных.

- Задача (полезная нагрузка) - это единица работы, которая обрабатывает единицу данных (файл).
- Блок, отвечающий за обработку одного файла с точки зрения рабочей нагрузки, называется задачей.
- **Ответственность за генерацию задач, их отправку на вычислительные узлы и выполнение лежит на системе управления рабочей нагрузкой (Workload Management System, WMS).**

Требования к системе управления нагрузкой

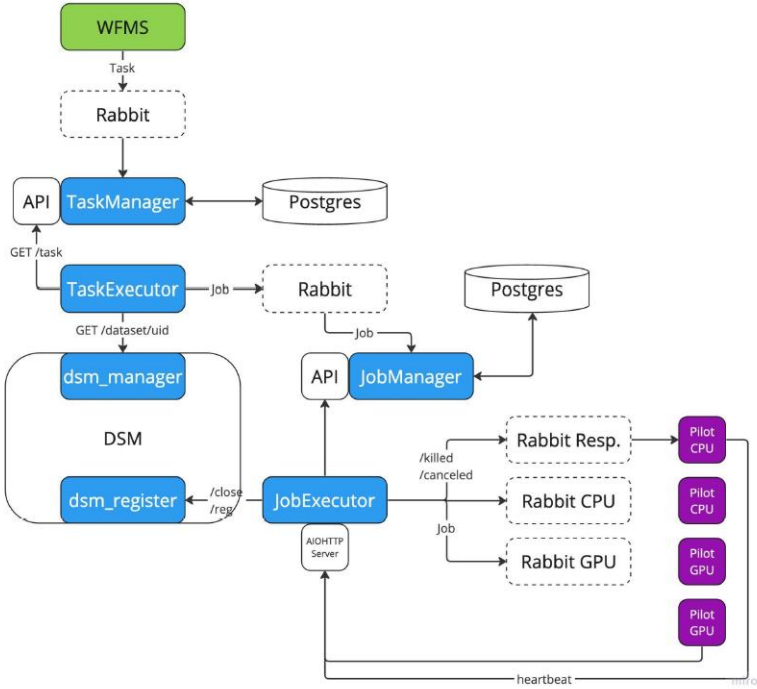
- **Регистрация заданий –** формализованное описание заданий, включая параметры задач и набор метаданных для регистрации.
- **Определение и генерация задач по шаблону задания –** генерация необходимого количества задач для выполнения прикладного ПО путем контролируемой загрузки доступных вычислительных ресурсов.
- **Управления ходом выполнения задач** – постоянный мониторинг состояния задач посредством связи с пилотом, повторные попытки выполнения задач в случае сбоев, завершение выполнения задач.



Выполнено предметно-ориентированное проектирование системы управления нагрузкой

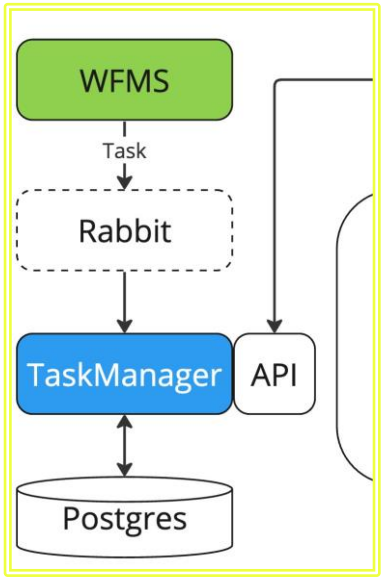
Была определена следующая архитектура и функциональность **Workload Management System**:

- **task-manager –** отвечает за взаимодействие с системой управления рабочими процессами, обеспечивая функциональность регистрации заданий, их отмены, отслеживания хода выполнения и формирования итоговых отчетов
- **task-executor –** отвечает за формирование задач в зависимости от содержания набора данных и типа заданий, управление очередью заданий
- **job-manager –** отвечает за получение задач из брокера, регистрацию задач, передачу готовых задач исполнителю задач, предоставляет внутренние API для работы с задачами и метаданными файлов
- **job-executor –** направляет задачи пилотам и следит за их выполнением, регистрирует данные.



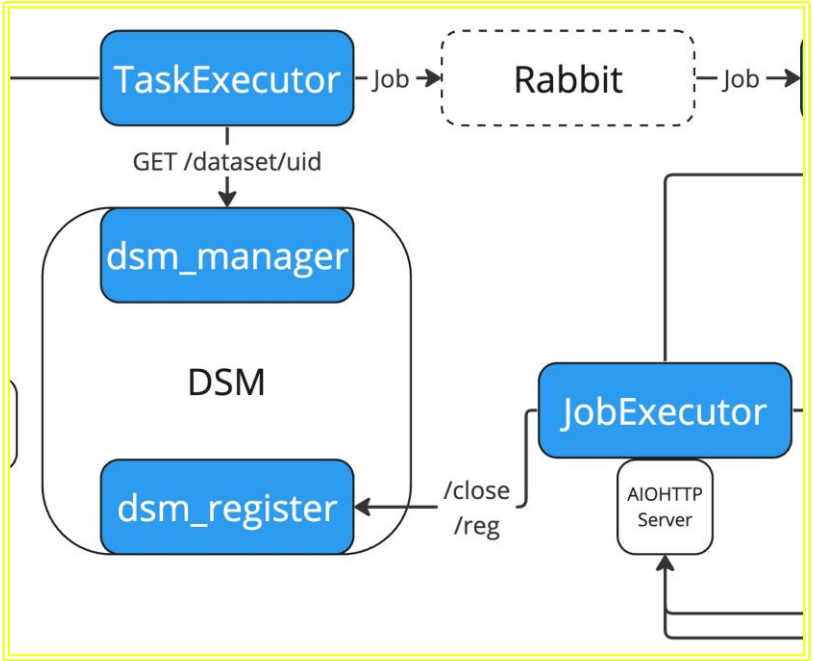
Взаимодействие с системой управления процессом обработки данных – *WFMS*

1. Регистрация задания на обработку.
2. Отмена, изменение приоритета задания.
3. Сводка о текущих задачах в системе.
4. Сводка о текущих выходных файлах в системе.



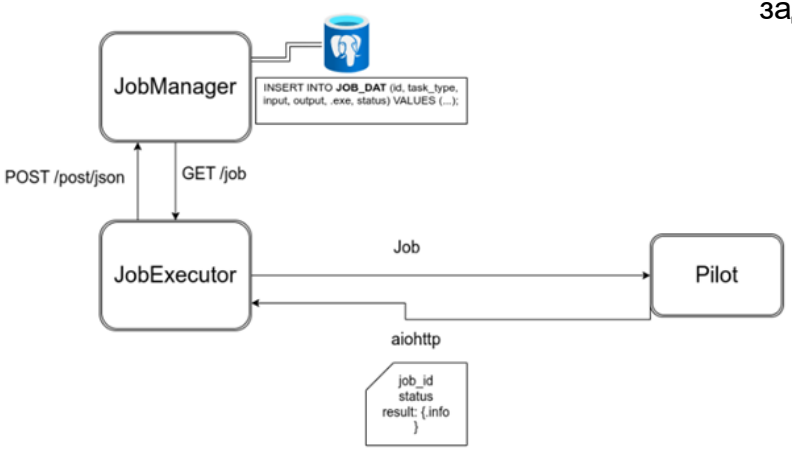
Взаимодействие с системой хранения и обработки данных – *DSM*

1. Получение информации о содержимом датасета.
2. Регистрация файлов/логов в датасете.
3. Закрытие датасета.



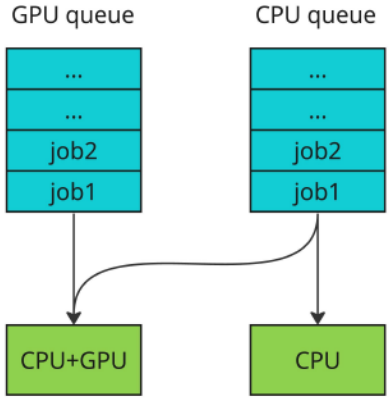
Взаимодействие с пилотным приложением

- Два канала коммуникаций:
  - REST (aiohttp)
  - Брокер сообщений (RabbitMQ)



- Два типа архитектур:
  - Multi-CPU
  - Multi-CPU + GPU

Пилот определяет тип ресурса и забирает задание из соответствующей очереди. Если нет заданий, ориентированных на GPU, пилот возьмет задание из очереди "CPU".



Технологический стэк

Основные ➢ Python 3.11 ➢ docker ➢ docker compose	Фреймворки ➢ aio-pika (RabbitMQ + asyncio) ➢ FastAPI ➢ aiohttp
Базы данных ➢ PostgreSQL СУБД ➢ Alembic (миграции БД) ➢ SQLAlchemy 2.0 ➢ asyncpg	Дополнительно ➢ uvicorn ➢ Pydantic ➢ pytest-asyncio

Текущие задачи

- Отладка взаимодействий с пилотом, обновление статуса задачи по мере выполнения.
- Формирование задач по датасету, закрытие датасета для системы хранения и обработки данных.
- Развертывание брокера сообщений RabbitMQ и PostgreSQL СУБД на виртуальной машине.

Выступления

1. Весенняя школа по информационным технологиям ОИЯИ, Лаборатория информационных технологий им. М.Г.Мещерякова, Дубна, Россия – Проектирование планировщика задач для специализированной распределенной вычислительной системы SPD Online filter
2. 10th International Conference `Distributed Computing and Grid Technologies in Science and Education` (GRID`2023), Joint Institute for Nuclear Research, Dubna, Russia – Система управления нагрузкой для специализированной вычислительной системы «SPD On-line Filter»

Публикации

1. *N. Greben, L. Romanychev, D. Oleynik, A. Degtyarev*.
SPD On-line Filter: Workload management system and Pilot Agent.
// Physics of Particles and Nuclei

Планы на следующий год

- Отладка основных алгоритмов и интерфейсов с внешними системами.
- Проработка интеграции с прикладным ПО и выход на тестирование на моделированных данных SPD-DAQ.