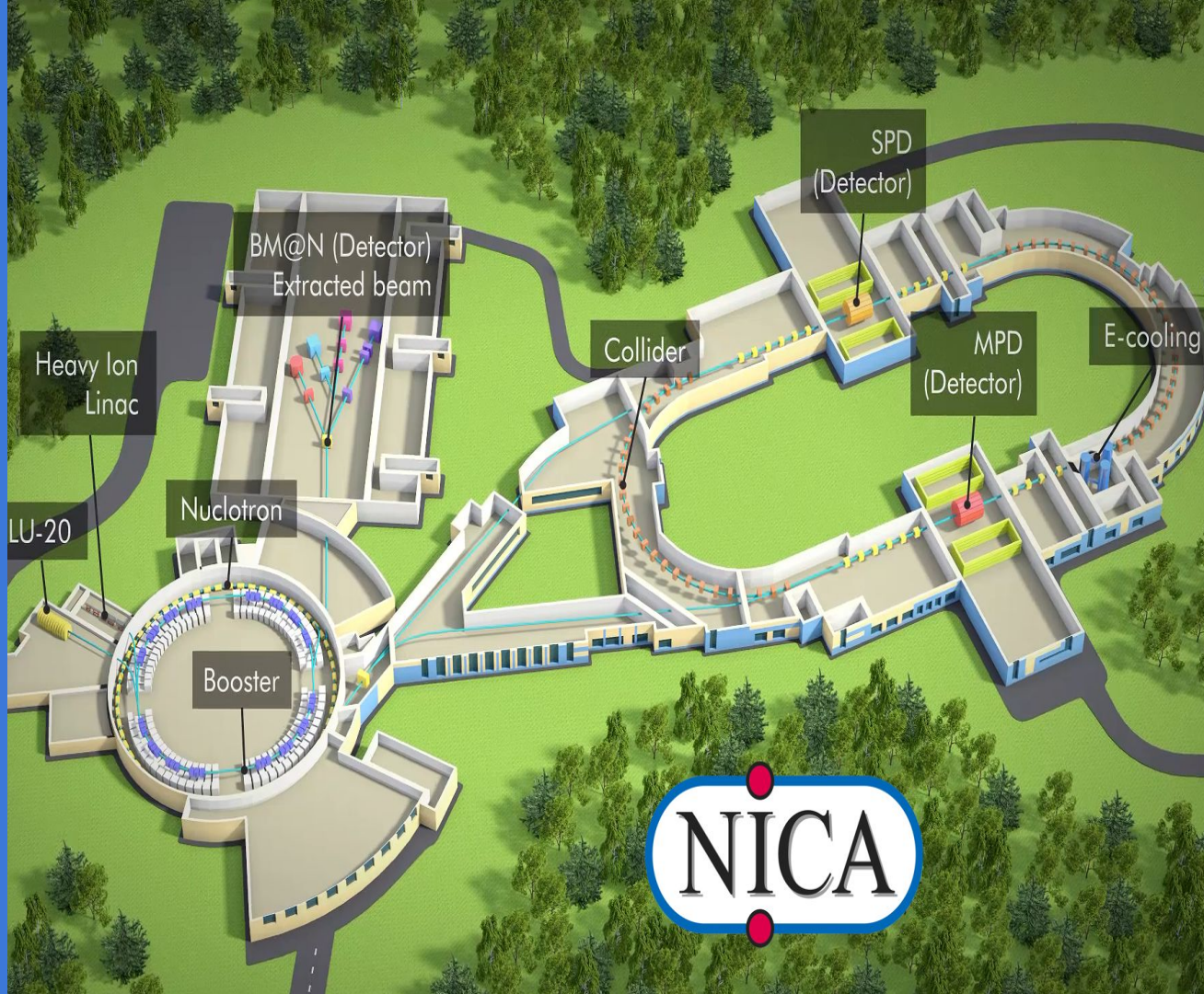
Eventindex

Алан Газзаев
1 курс магистратуры
Университет ИТМО

Научный руководитель:
Прокошин Федор Валерьевич (ОИЯИ)

EventIndex for SPD



SPD EventIndex

SPD EventIndex - это каталог физических событий, полученных с детектора SPD на коллайдере NICA или смоделированных для анализа и обработки данных.

Ожидаемый поток данных с детектора SPD может достигать 30 миллиардов событий в год.

При разработке EventIndex использовался опыт создания аналогичной системы на установке ATLAS.

Оба эксперимента имеют сходную модель данных событий и организацию хранения и обработки данных. Основное отличие: В SPD используется бестригерная система сбора данных, отбор событий с помощью Online Filter

Каталог будет содержать ссылки на все экземпляры событий во всех форматах и версиях, хранящихся на серверах распределенной вычислительной системы.

Для разработки системы каталога событий определяется предварительный набор параметров которые мы будем индексировать. Набор параметров может быть изменен

Информация о событиях содержится в таблице **events**

run_number | event_number | olf_result | fuid_raw | dsid_raw
[| fuid_aod | dsid_aod [| fuid_aod_v2 | dsid_aod_v2 …]]] | var_1 [var_2] …

olf_result - результат онлайн фильтра, критерии по которым отобрано событие

fuid_xxx - UUID для файла в котором хранится это событие, в файле хранится несколько тысяч событий. позволяет получить доступ к файлу в распределенной системе хранения

dsid_xxx - UUID для датасета, содержащего файл с информацией о событи. внешний ключ для дополнительной таблице **datasets**

dsid_raw | name | date | events | …
dsid_aod | name | date | events | …

запись события также может содержать несколько важных параметров, используемых при отборе для анализа

Для оптимизации размеров таблицы предполагается изменить схему организации данных добавив таблицу файлов

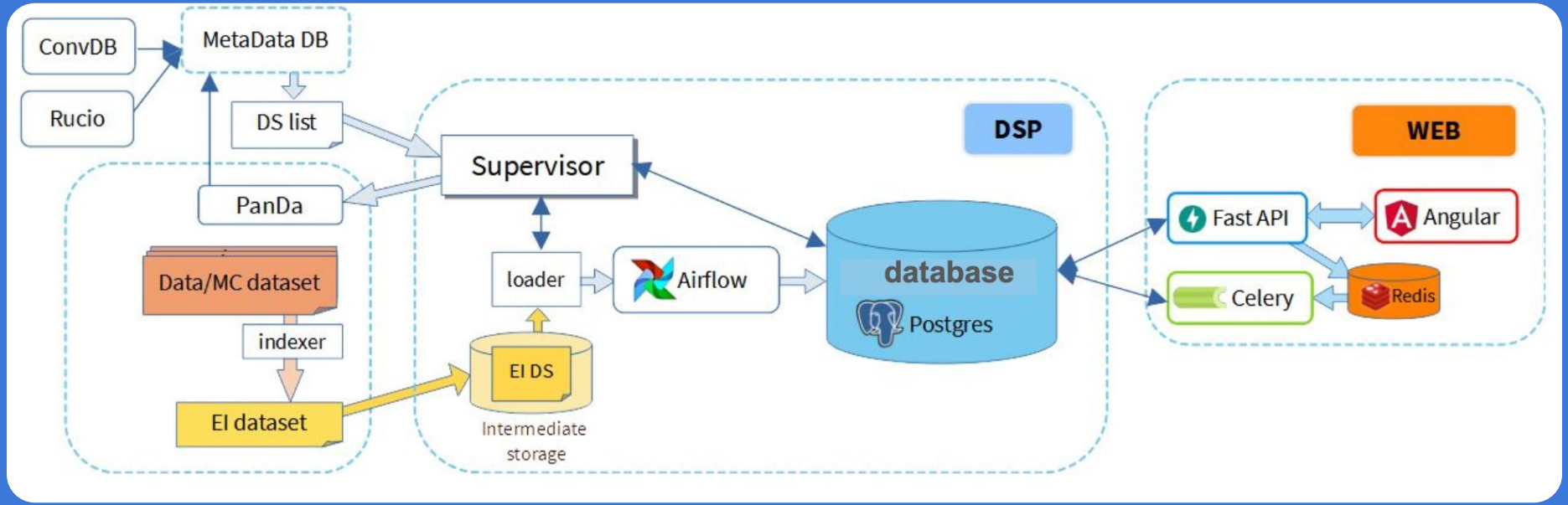
Новая таблица **events** с внешним ключом **fuid_xxx**

run_number | event_number | olf_result
| fuid_raw | fuid_aod [| fuid_aod_v2 [| fuid_aod_v3 …]]] | var_1 [var_2] …

Таблица файлов содержит dsid_xxx в качестве внешнего ключа

fuid_raw | dsid_raw | date | events | …
fuid_aod | dsid_aod | date | events | …

текущая архитектура EventIndex



Supervisor

Супервизор использует gRPC для усправления процессами. gRPC — это фреймворк с открытым исходным кодом для удаленного вызова процедур.

Это абстрактная спецификация. Она описывает абстрактную RPC (remote procedure call), то есть удаленный вызов процедуры, которая обладает определенными свойствами:

1) Поддержка одиночных вызовов и стриминга. То есть все сервисы, которые реализовывают эту спецификацию, поддерживают оба варианта.
2) Возможность передавать метаданные
3) Поддержка отмены запроса и таймаутов

Supervisor. Что есть на данный момент

gRPC- сервер

```
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  24307368569993911187
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  60587960997857435164
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  10349375620687626 95
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  7709520852446125564
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  4874141362615808731
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  247522647591238837
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  26829623955783458284
{'user-agent': 'grpc-python/1.62.1 grpc-c/39.0.0 (osx; chttp2)'}
FILE PATH:  /Users/a.gazzaev/Desktop/gRPC/test.py
START REMOTE PYTHON SCRIPT:
TOKEN:  2232124856697178198
```

EventIndex server file structure:

protobufs/remote_task.proto- служит API между клиентом и сервером. Этот файл содержит объявления для методов интерфейса и указывает параметры и возвращаемые значения. Он также определяет типы сообщений, используемые для связи клиент-сервер

eventindex/supervisor_server.py- gRPC- сервер, выполняет запуск удаленного скрипта

eventindex/remote_task_pb2_grpc.py и eventindex/remote_task_pb2.py-Генерируемый код, от которых наследуется основной класс сервера

Supervisor. Что есть на данный момент:

gRPC- клиент

```
args = parser.parse_args()

def run():
    with grpc.insecure_channel('localhost:50051') as channel:
        stub = remote_task_pb2_grpc.RunTaskServiceStub(channel)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)
        response = stub.RunTask(remote_task_pb2.RunRemoteTask(filepath=args.file_path))
        print("RESPONSE: ", response.task_id)

if __name__ == '__main__':
    logging.basicConfig()
    run()
```

Пример вызова удаленной процедуры

```
RESPONSE:  2430736856993911187
RESPONSE:  60587960997857435164
RESPONSE:  1034937562068762695
RESPONSE:  7709520852446125564
RESPONSE:  4874141362615808731
RESPONSE:  247522647591238837
RESPONSE:  26829623955783458284
RESPONSE:  2232124856697178198
```

Supervisor. Что планируется сделать:

1. Отбор и проверка датасетов на основе метаданных
2. Запуск программы индексации в виде задачи распределенной вычислительной системе PanDA. Информация о событиях записывается в датасет.
3. Отслеживание задачи
4. В случае неуспешного завершения задачи, отработка возникших ошибок, перезапуск при необходимости
5. При успешном завершении задачи, анализ лога задачи и отработка ошибок
6. Передача датасета с индексированными данными с удаленного сервера средствами Rucio и FTS
7. Если передача прошла успешно - запуск процедуры записи в БД
8. При успешной записи данных в БД - сообщение ИС метаданных об успешной проверке данных и о количестве индексированных событий.
9. В противном случае датасет помечается в ИС метаданных как дефектный

Заключение

1. Производится разработка каталога событий SPD (для реальных и Монте- Карло данных)
2. В качестве системы хранения используется СУБД Postgres
3. EventIndex производит индексацию сгенерированных и реальных данных во взаимодействии с другими ИС SPD (ИС метаданных, Panda и системой распределенного хранения и передачи данных Rucio)
4. Для оркестровки процесса будет использоваться система Supervisor
5. Для взаимодействия с клиентами разрабатывается клиент с FastAPI и reddis и фронтенд на основе angular

Спасибо!