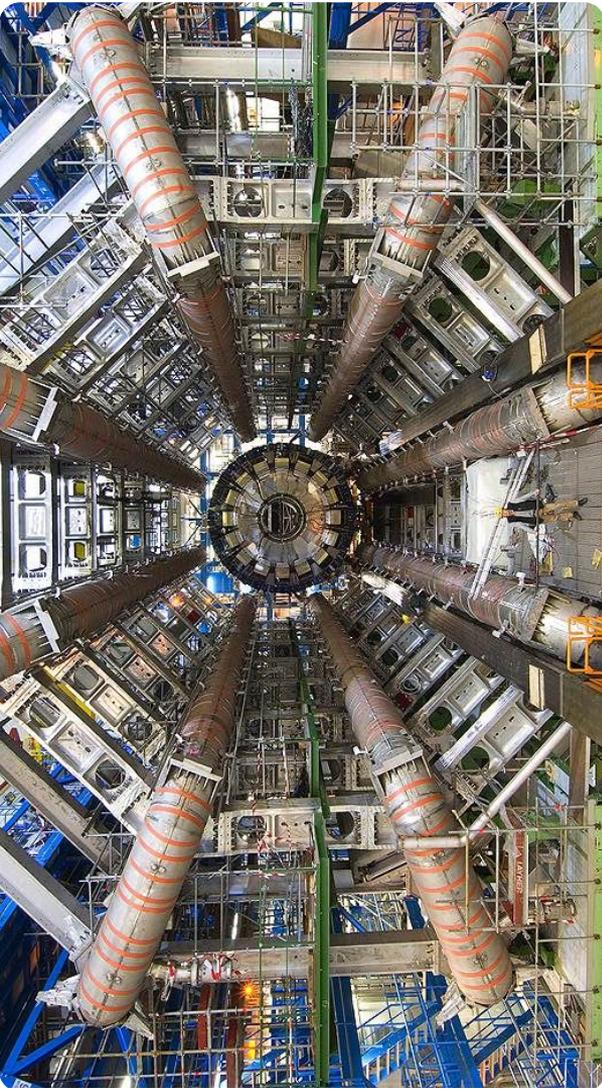


Eventindex

Ричард Гурциев, 3 курс бакалавриата,
Алан Газзаев, 4 курс бакалавриата,
Замира Будтуева, 3 курс бакалавриата,
направление подготовки “Физика”

Северо- Осетинский Государственный Университет
им. Коста Левановича Хетагурова

Научные руководители:
Прокошин Федор Валерьевич (ОИЯИ),
Тваури Инга Васильевна (СОГУ)



EventIndex

Atlas

SPD



При разработке EventIndex использовался уже имеющийся опыт создания аналогичной системы на установке ATLAS, с дальнейшим использованием в установке SPD на коллайдере NICA.

EventIndex будет использоваться на детекторе SPD, с которого ожидается получение 30 миллиардов событий в год.

Оба эксперимента имеют сходную модель данных событий и организацию хранения и обработки данных. Основное отличие: Online Filter вместо триггера.

ПОЛЯ В ТАБЛИЦЕ EVENTS

```
run_number | event_number | olf_result | dsid_raw | dsid_raw |  
dsid_aod [| dsid_aod_v2 [| dsid_aod_v3 ...]] | var_1 [var_2] ...
```

Для разработки системы каталога событий достаточно определить предварительный набор параметров которые мы будем индексировать.

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

`olf_result` - результат онлайн фильтра, критерии по которым отобрано событие

`dsid_raw`- id датасета, в который объединены файлы с raw-данными, внешний ключ для таблицы `datasets`

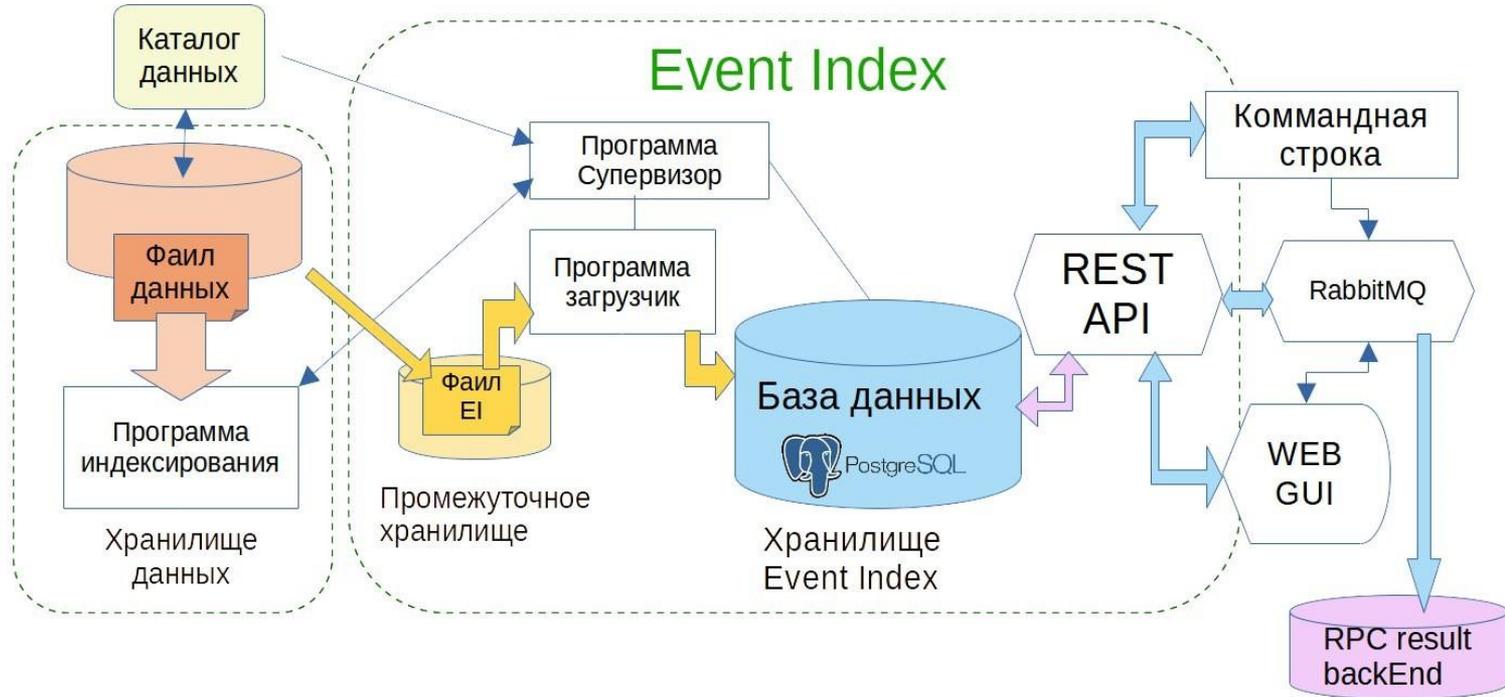
`fuid_raw`- UUID для файла в котором хранится это событие, в файле будет храниться от тысячи до 10 тысяч событий, по `uuid` к файлу можно получить доступ посредством распределенной системы хранения

ГЕНЕРАЦИЯ И ЗАПИСЬ

gen_data.py - генератор псевдоданных event index для последующего импорта

store_events.py- запись данных событий в таблицу events и датасета в таблицу datasets. Таблица датасетов имеет в качестве (primary) ключа id датасета, и содержит информацию о датасетах

существующая архитектура



database

POSTGRES

open-source

поддерживает сложную логику и
связи между таблицами

На начальном этапе использовался 1 млн событий, разбитый на 100 частей. Во время загрузки столкнулись с медленным выполнением записи данных, поэтому было решено использовать метод “массовой загрузки”.

PostgreSQL позволяет гибкую настройку параметров записи для ускорения массовой загрузки огромных объемов данных:

1. COPY вместо INSERT
2. Оптимизация контрольных точек
3. Отключенные триггеры
4. Оптимизация порядка столбцов для массовой загрузки

database

open-source

поддерживает сложную логику и
связи между таблицами

POSTGRES

для оптимизации
скорости загрузки и
чтения данных было
решено
протестировать
различные драйвера

asyncpg

pg8000

SQLAlchemy

psycopg

PyGreSQL

asyncio

web-app

EVENT SEARCH



SEARCH

Run number

Event number

Data format: RAW

Data version: r2345

request

request id	Client username	Client e-mail	format	version	created	fuid
33	wcsgio	ftemf@jinr.ru	RAW	r2345	2022-07-04 08:26:34	3d5db645-6e79-48a9-8c99-385a0df217c0
32	icoluorbom	aqiyzcvcem@jinr.ru	RAW	r2345	2022-07-03 00:50:00	acc453d8-6460-47a0-9889-aa3003f55933
31	ggmiqbv	rtxktn@jinr.ru	RAW	r2345	2022-01-29 03:43:25	c3969954-0de4-4df8-81df-2e18a6837660
30	ybpqxaqti	qynjffdj@jinr.ru	RAW	r2345	2022-06-29 18:22:07	10c170f3-45e6-4e91-bd14-4ada2763849b
29	weedn	jmfulzdja@jinr.ru	RAW	r2345	2022-01-12 01:51:06	17753d28-7c58-46ed-afb5-2d0a06d8ffee
28	tqnbaeltjn	fqgtyhdaz@jinr.ru	RAW	r2345	2022-04-02 11:14:07	2cfa616d-6807-4756-9310-686cd2ff73b1
27	nhiukre	epyjuvonbr@jinr.ru	RAW	r2345	2022-10-22 01:07:41	58918511-2cf7-4561-ac1a-40cb7e26b1ba
26	xoxqzk	ohujj@jinr.ru	RAW	r2345	2022-08-14 12:52:56	908203e9-d2f7-4729-84a5-815fb5df9958
25	dylobfb	pezjnvul@jinr.ru	RAW	r2345	2022-11-04 02:41:33	625cc4af-44ec-41f4-94bd-498f962b20d7
24	wddcsve	gllkqi@jinr.ru	RAW	r2345	2022-07-16 02:26:25	f70676c0-ae90-401f-b428-c04e8dfe0a95
23	regytd	zwtnejrerb@jinr.ru	RAW	r2345	2022-08-16 21:30:32	8d8b72f4-0d61-4abc-8d5b-49f8f85a9ebb
22	pprxentoyz	hertkoctx@jinr.ru	RAW	r2345	2022-10-25 04:40:23	7fc980ce-29fa-4e04-951e-8aea6478d0b3
21	hhkbuodgp	utjqr@jinr.ru	RAW	r2345	2022-10-15 22:31:50	cc2f4602-1938-4544-b7e9-ed0209d5f503
20	flppmwktye	wxunowxm@jinr.ru	RAW	r2345	2022-09-13 00:21:05	51e392df-dfe2-49e9-8ef5-63dd3bf04376
19	zgrqw	fgiwsqpv@jinr.ru	RAW	r2345	2022-03-18 19:51:04	18744b2b-a97b-4a04-950a-00fe08ab8632
18	cbmtghopqp	dtlzkjhjn@jinr.ru	RAW	r2345	2022-09-19 00:08:00	f9e9faa7-f449-4a0e-963b-4fd637b8a4ec
17	hxvofyrwr	vmzozewh@jinr.ru	RAW	r2345	2022-12-20 16:38:41	35f7230d-974f-42ff-9435-2b19b7b12c2f
16	oifztyu	dzfjuw@jinr.ru	RAW	r2345	2022-02-11 22:52:10	b3a0c64b-776d-4ffe-80f1-3b1996e7a379
15	toctroufe	yqdluvkvr@jinr.ru	RAW	r2345	2022-01-25 11:43:11	76d5459e-a82b-49cc-bcf4-0fee828b816a
14	unzquetuck	btwhryvir@jinr.ru	RAW	r2345	2022-11-27 13:17:54	3c355001-8b0d-420e-98c4-98a238d64f40

web-app

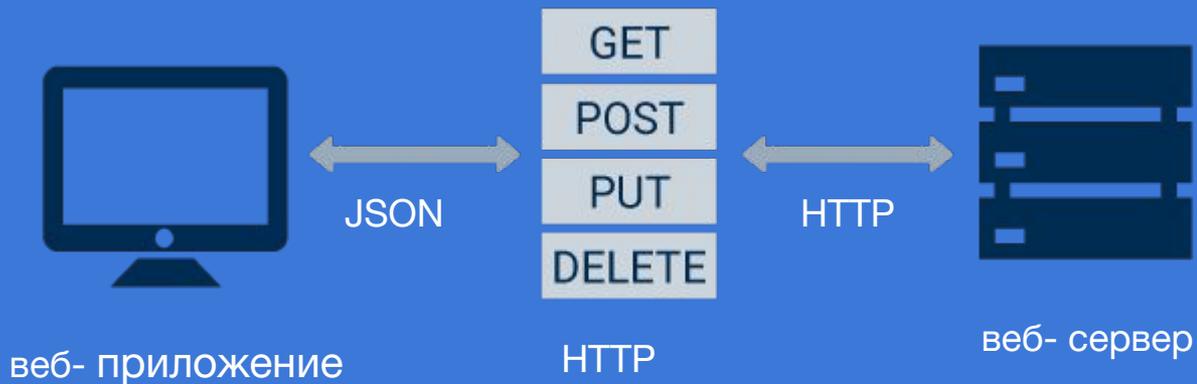
Результатом запроса является `fuid_raw` (id датасета, в который объединены файлы с `raw` данными).

Получаемые данные генерируются в тестовом режиме на стороне бэкэнда.

```
eventindex=# SELECT * FROM events;
```

run_number	event_number	olt_result	dsid_raw	fuid_raw
280308061	8	29721	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	14	24218	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	22	11794	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	37	13903	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	39	18546	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	54	27200	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	66	23844	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	71	18992	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	86	9718	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	99	15973	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	113	18546	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	129	1296	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	136	23743	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	141	10936	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	154	5041	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	160	9718	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	181	26502	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	182	11376	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	188	9144	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	190	15356	1	3ad87450-23e6-4b72-afcb-077b5184577e
280308061	197	1881	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	206	8011	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	220	86	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	228	31532	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	235	30732	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	252	11294	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	257	1881	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	264	23740	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	282	4495	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	294	14914	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	315	15356	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091
280308061	321	17183	1	6cfc4dfc-c63b-4ae2-95f3-b2e889478091

Веб- сервер (REST API)



Структура

`app.py` - основной файл для запуска приложения

`flask_manager.py` - создает app

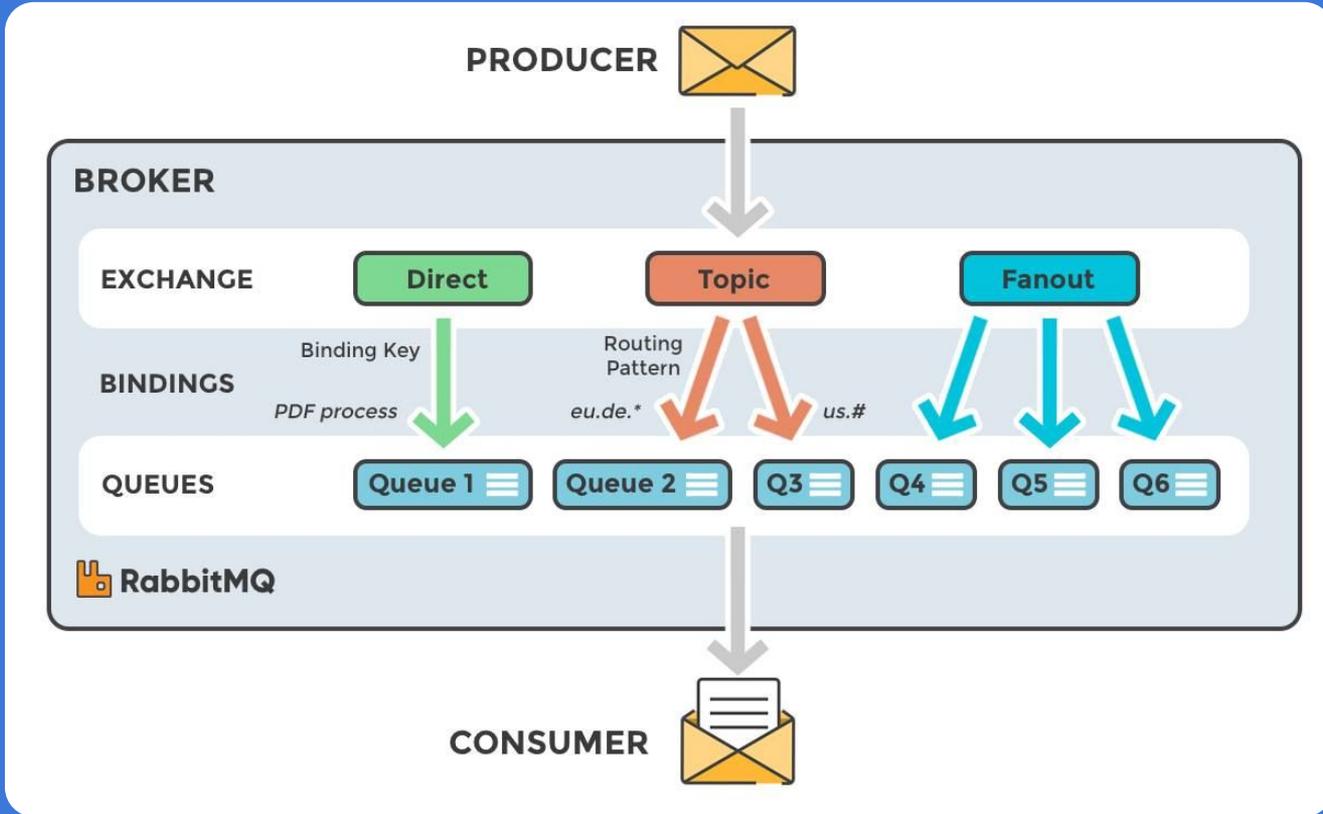
`rest_api.py` - содержит эндпоинты

Асинхронная выдача результатов

В случае если пользователь делает запрос, выполнение которого занимает много времени, результат можно будет увидеть в личном кабинете или на почте. Также в личном кабинете можно будет отслеживать статус выполнения задачи. Система асинхронное выполнение задач работает на основе RabbitMQ, задачи хранятся в RPC result backend

celery + RabbitMQ

RabbitMQ



ЗАДАЧИ:

- Оптимизация загрузки больших объемов данных в БД на основе PostgreSQL.
- Разработка механизмов авторизации и аутентификации пользователей.
- Оптимизация обработки запросов пользователя
- Разработка механизмов передачи данных EventIndex для индексирования файлов расположенных на удаленных узлах вычислительной сети
- разработка диспетчера (supervisor) - программного обеспечения для управления сбора и импорта данных в eventIndex
- Разработка системы мониторинирования компонент EventIndex

Спасибо!