



# СИСТЕМА УПРАВЛЕНИЯ ПРОЦЕССАМИ ОБРАБОТКИ В SPD ONLINE FILTER

Студент

Научный руководитель

к.ф.-м.н.

Научный консультант

к.т.н

А. В. Плотников

Е. Ю. Солдатов

Д. А. Олейник

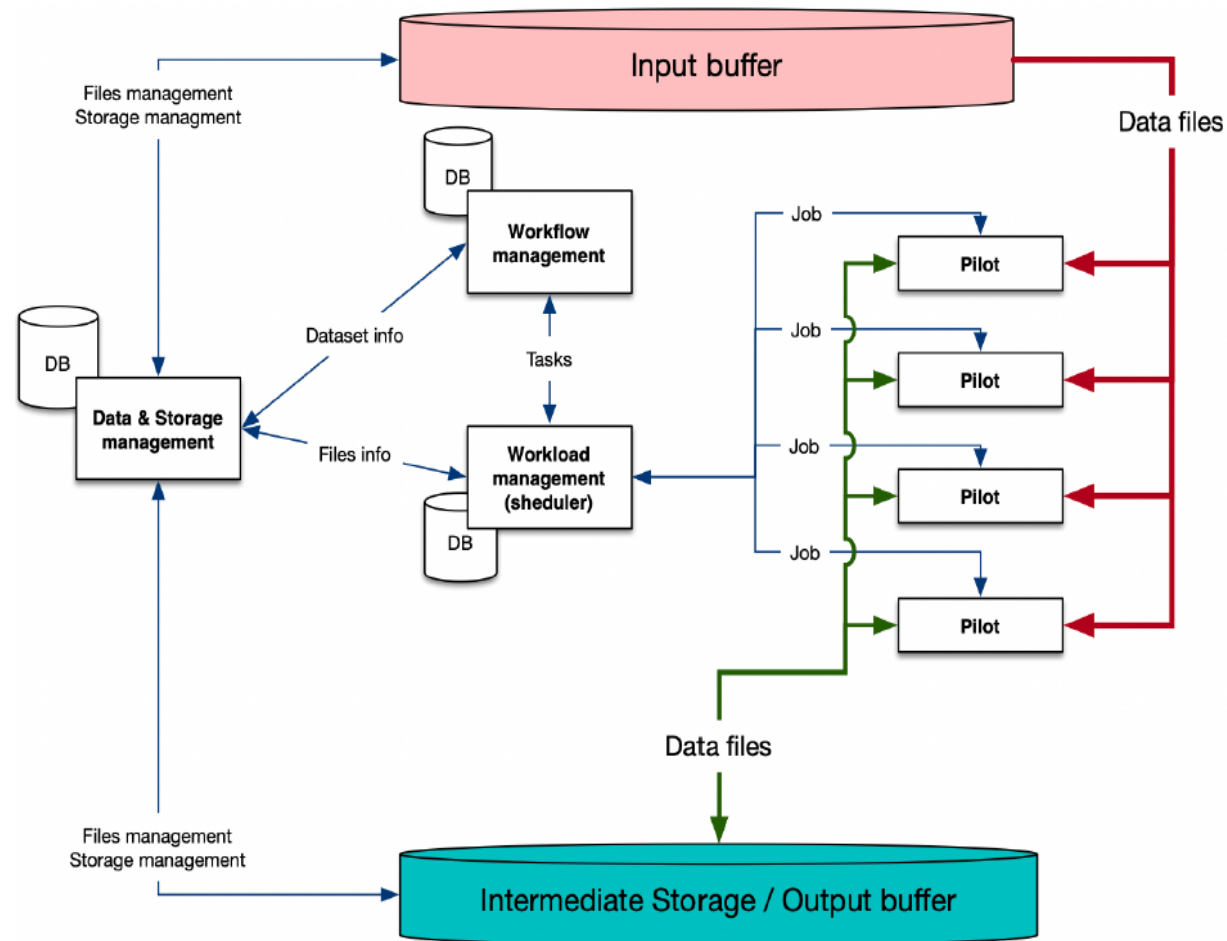


# SPD Online Filter

**SPD Online Filter** — специализированная программно-аппаратная система, предназначенная для предварительной обработки данных эксперимента SPD.

Система реализует многоступенчатую, высокопропускную методику обработки.

Основная, но не единственная, **цель обработки**: существенно уменьшить объём данных для последующего анализа и долговременного хранения.

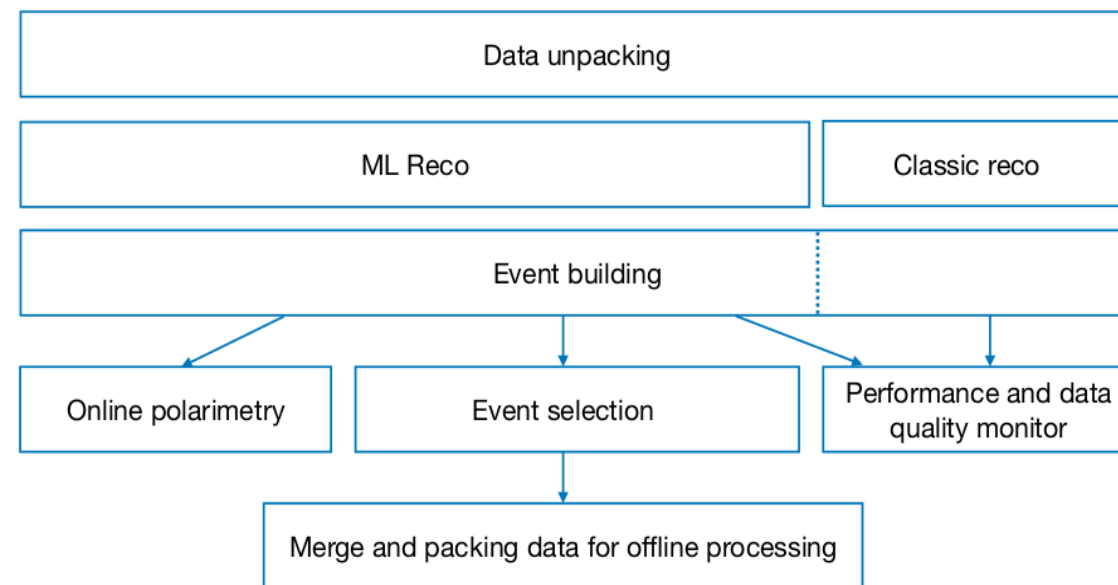


Архитектура Online Filter

# Управление процессами обработки данных

**Многоступенчатая обработка** — набор последовательных этапов обработки данных. При этом на каждом этапе может обрабатываться достаточно большой объем данных.

Каждый шаг, кроме первого, принимает данные, обработанные на предыдущем этапе, и передает результаты следующему, обеспечивая преобразование информации из одного представления в другое.



**Пример последовательных шагов обработки**

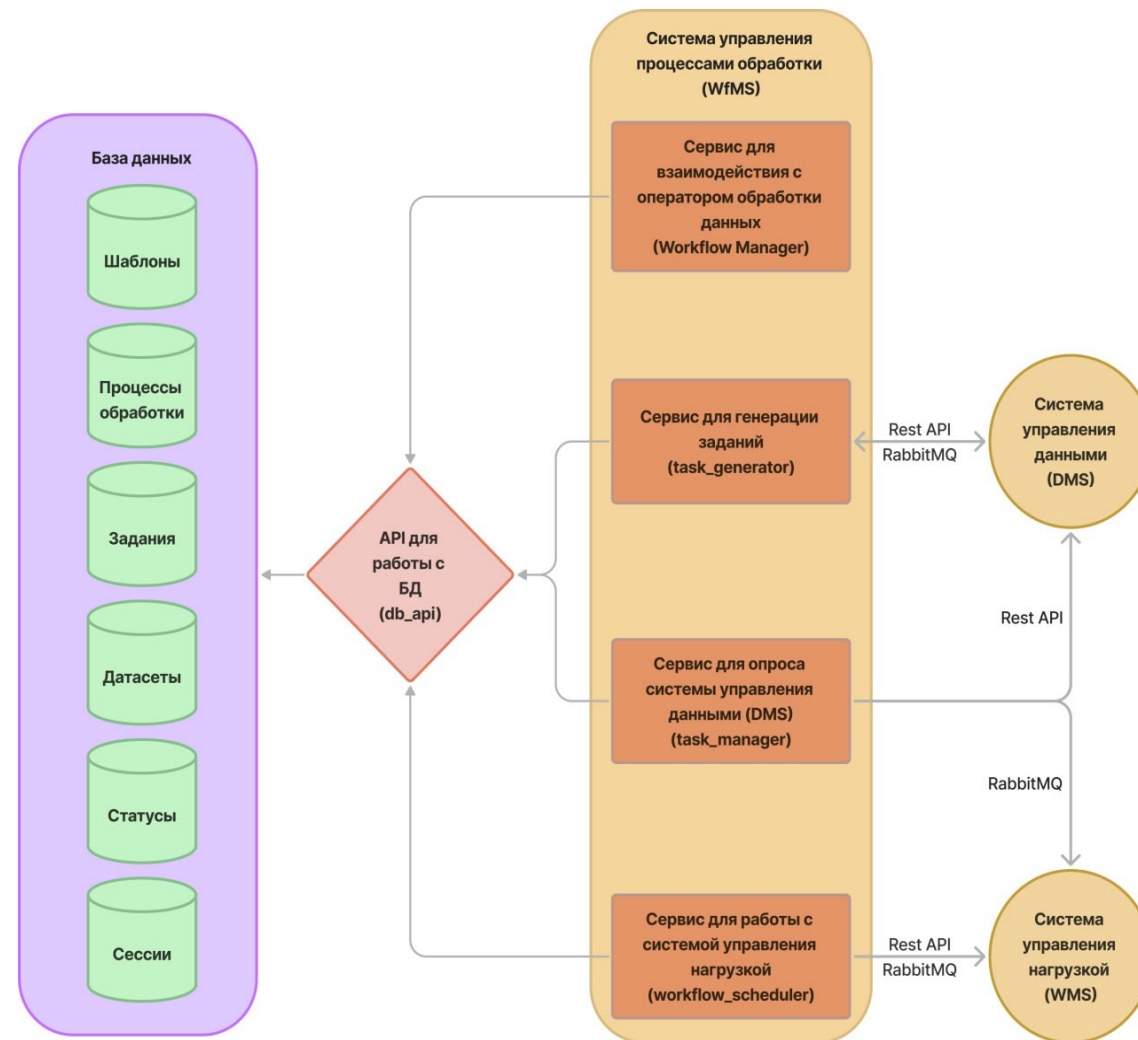
# Система управления процессами обработки (WfMS)

## Основная цель системы:

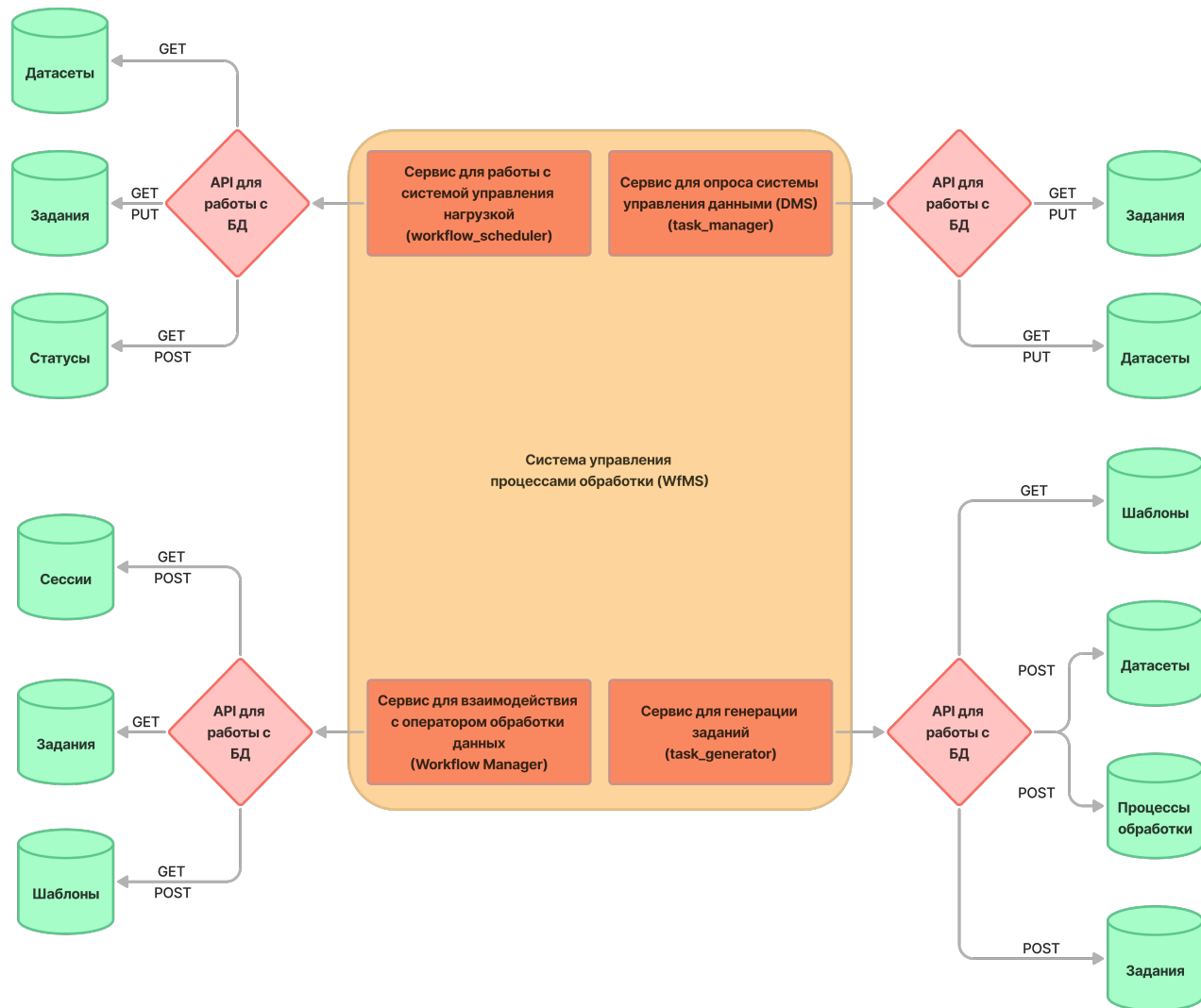
параллельное и наиболее результативное управление большим количеством процессов обработки данных и контроль статусов выполнения цепочек обработки для физического эксперимента SPD.

## Основные задачи системы:

- Создание формального описания процесса;
- Контроль за выполнением обработки.



# Взаимодействия микросервисов WfMS с БД

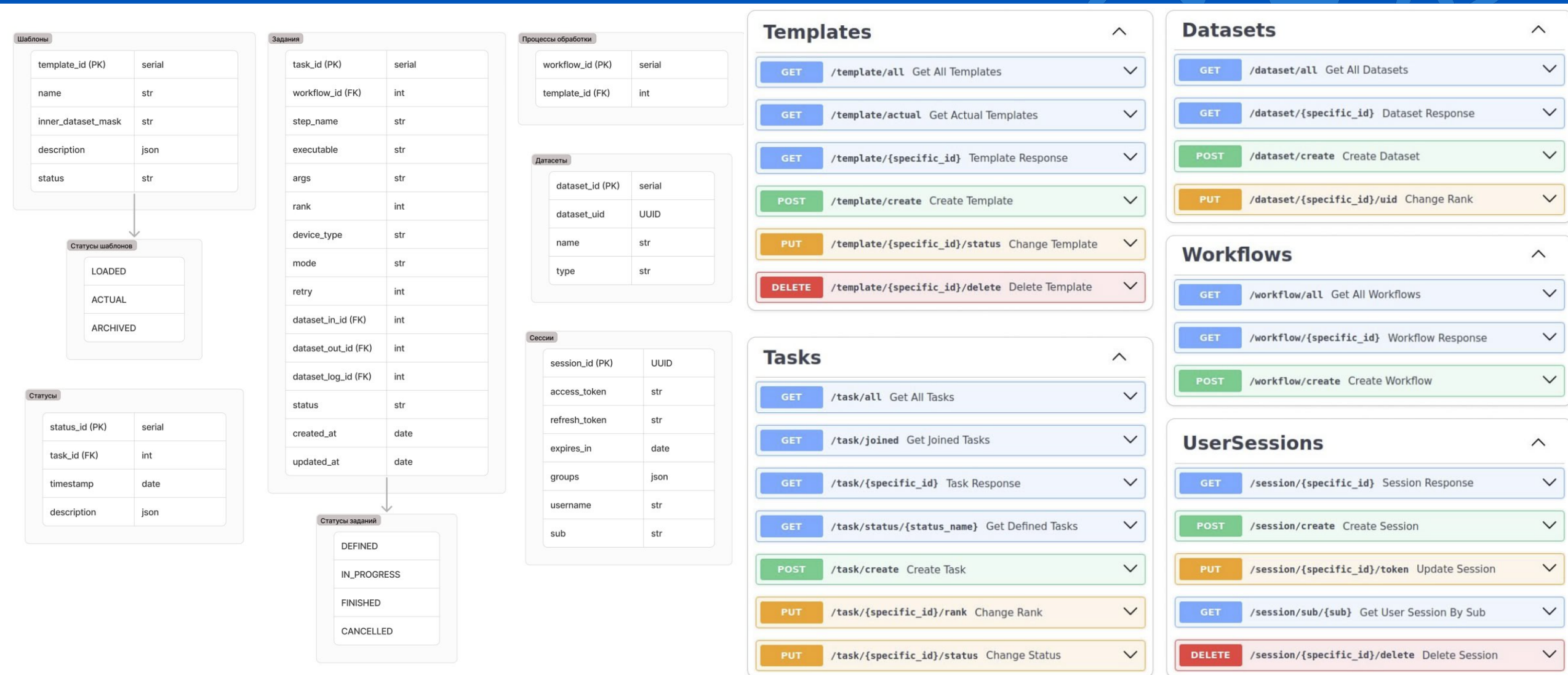


- База данных PostgreSQL развернута на отдельной виртуальной машине.
- Взаимодействие с базой данных осуществляется через REST API.

SQLAlchemy ORM  
+  
асинхронные сессии и asynprg  
+  
миграции Alembic

Схема взаимодействия микросервисов с БД

# Структура базы данных и API к ней



# Сервис для взаимодействия с оператором обработки данных

## Основной функционал сервиса

1. Авторизация пользователей с помощью SPD-IAM;
2. Вывод CWL-шаблонов;
3. Вывод заданий;
4. Создание CWL-шаблонов суперпользователями (ввод в специальное поле / загрузка из файла / клонирование существующих шаблонов);
5. Предварительная валидация и запись в БД CWL-шаблонов;
6. Изменение статусов шаблонов суперпользователями на "ACTUAL" и "ARCHIVED";
7. Удаление суперпользователями шаблонов в статусе "LOADED".

- Пользовательский интерфейс
- Backend: FastAPI
- Шаблонизатор: Jinja2
- Frontend: Bootstrap (HTML + CSS + JS)

## Workflow Manager

Templates ▾

Tasks

[admin@jinr.ru](#)

Logout

id	wflow_id	step	template	exec	args	priority	type	mode	retry	in_ds_name	out_ds_name	log_ds_name	status
<a href="#">2</a>	1	reconstruction	<a href="#">Decoding &amp; Reco</a>	processing_program	callback	1	CPU	map	5	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw.output.1	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw.output.2	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw.log.2	DEFINED
<a href="#">1</a>	1	decoding	<a href="#">Decoding &amp; Reco</a>	processing_program	callback	1	CPU	map	5	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw.output.1	input.test.4b5f78b1-2412-4058-9a7e-f9b09012ec9d.raw.log.1	IN_PROGRESS

## Workflow Manager

Templates ▾

Tasks

[admin@jinr.ru](#)

Logout

template_id	name	input_dataset_mask	status
<a href="#">2</a>	Decoding&Reco	.test.	<a href="#">ACTUAL ▾</a>
<a href="#">4</a>	Data_Decoding_Clone	.test.	<a href="#">ARCHIVED ▾</a>
<a href="#">1</a>	Data_Decoding	.test.	<a href="#">ARCHIVED ▾</a>
<a href="#">5</a>	RAW data decoding	RAW2024	<a href="#">LOADED ▾</a>
			<a href="#">Delete</a>

# Создание шаблонов

- Создание шаблонов доступно только суперпользователям (в частности, отсутствует кнопка в интерфейсе)
- Предварительная валидация шаблонов с помощью инструментов cwltools
- Сохранение шаблона в БД

Workflow Manager

Templates ▾

Tasks

[admin@jinr.ru](#)

Logout

## CWL Template

Template name

Inner dataset mask

Enter CWL here...

Complete



# Сервис для генерации заданий

1. Получение зарегистрированных датасетов от DMS из RabbitMQ;
2. Сопоставление датасета по маске имени с нужным шаблоном;
3. Регистрация входного датасета в системе;
4. Создание процесса обработки по шаблону;
5. Создание выходного датасета и датасета логов в системе;
6. Создание заданий.

## Шаблон

```
{
  "name": "template1",
  "description": "(CWL)",
  "inner_dataset_mask": ".test.",
  "template_id": 1,
  "status": "ACTUAL"
}
```

## Датасеты и процесс обработки

```
[
  {
    "type": "input",
    "dataset_id": 26,
    "dataset_uid": "5a2775f5-73c5-4328-bdf3-166cff26a594",
    "name": "input.test.368d6f34-3925-4272-9d40-3059295a7fcd.raw"
  },
  {
    "type": "output",
    "dataset_id": 27,
    "dataset_uid": "37222a88-6bde-4183-ba4f-06d164ae689b",
    "name": "input.test.368d6f34-3925-4272-9d40-3059295a7fcd.raw.output.1"
  },
  {
    "type": "output",
    "dataset_id": 28,
    "dataset_uid": "410dcf92-7534-4009-ba3d-55222161dfc8",
    "name": "input.test.368d6f34-3925-4272-9d40-3059295a7fcd.raw.log.1"
  },
  {
    "type": "output",
    "dataset_id": 29,
    "dataset_uid": "2a81e72d-5368-4c57-a752-6a74cb787e2a",
    "name": "input.test.368d6f34-3925-4272-9d40-3059295a7fcd.raw.output.2"
  },
  {
    "type": "output",
    "dataset_id": 30,
    "dataset_uid": "391baf82-5fdb-482a-8736-276844452f45",
    "name": "input.test.368d6f34-3925-4272-9d40-3059295a7fcd.raw.log.2"
  }
]
```

```
{
  "template_id": 1,
  "workflow_id": 6
}
```

## Задания

```
[
  {
    "task_id": 11,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 28,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 26,
    "dataset_out_id": 27,
    "status": "DEFINED"
  },
  {
    "task_id": 12,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 30,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 27,
    "dataset_out_id": 29,
    "status": "DEFINED"
  }
]
```

От шаблона к заданиям

# Сервис для опроса системы управления данными (DMS)

1. Итерация по заданиям в статусе "DEFINED";
2. Опрос DMS о статусе входного датасета ("CLOSED");
3. Создание выходных датасетов и датасетов логов в DMS;
4. Отправление задания в RabbitMQ для последующей обработки в WMS;
5. Смена статуса задания на "IN\_PROGRESS".

Задания до

```
[
  {
    "task_id": 11,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 28,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 26,
    "dataset_out_id": 27,
    "status": "DEFINED"
  },
  {
    "task_id": 12,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 30,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 27,
    "dataset_out_id": 29,
    "status": "DEFINED"
  }
]
```

RabbitMQ

Messages: 1  
Get Message(s)  
Message 1  
The server reported 5 messages remaining.

Exchange wfms.manager  
Routing Key wfms.manager.tasks.key

Redelivered ☐

Properties

Payload

375 bytes

Encoding: string

```
{
  "task_id": 29,
  "executable": "processing_program",
  "args": "cable_map",
  "rank": 1,
  "device_type": "CPU",
  "mode": "map",
  "retry": 5,
  "dataset_in_id": 26,
  "dataset_out_id": 27,
  "status": "IN_PROGRESS"
}
```

Задания после

```
[
  {
    "task_id": 11,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 28,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 26,
    "dataset_out_id": 27,
    "status": "IN_PROGRESS"
  },
  {
    "task_id": 12,
    "args": "cable_map",
    "device_type": "CPU",
    "retry": 5,
    "dataset_log_id": 30,
    "workflow_id": 6,
    "executable": "processing_program",
    "rank": 1,
    "mode": "map",
    "dataset_in_id": 27,
    "dataset_out_id": 29,
    "status": "DEFINED"
  }
]
```

Отправление заданий на обработку

# Сервис для работы с системой управления нагрузкой (WMS)

## Основной функционал сервиса

1. Завершение заданий с закрытием обработанных датасетов в DMS;
2. Отмена заданий на основе данных о заполненности выходного хранилища с оповещением WMS;
3. Изменение приоритетов заданий в процессе их выполнения.

## 1 этап

1. Итерация по заданиям в статусе "IN\_PROGRESS";
2. Опрос WMS о состоянии задания;
3. Запись состояния;
4. Для успешно завершенных заданий:
  - а) Смена статуса выходных датасетов на CLOSED в DMS;
  - б) Смена статуса задания на "FINISHED";
5. Удаление входного и промежуточных датасетов.

# Логирование, контейнеризация и оркестрация

- Логирование: custom logger

```
task_generator 2024-09-24 20:59:44,380 INFO Logger started
task_generator 2024-09-24 20:59:44,665 INFO Templates successfully initialised.
```

- Контейнеризация и оркестрация:  
Docker и Docker Compose

docker-compose

```
services:
  db_api:
    ...
  task_generator:
    ...
  task_manager:
    ...
  workflow_manager
    ...
  workflow_scheduler
    ...

networks:
  wfms:
```

# Результаты и дальнейшие планы

## Результаты

- 1) разработан API для работы с базой данных, открывающий доступ для сервисов WfMS к определенному набору функций;
- 2) создан сервис для работы с оператором обработки данных, позволяющий просматривать задания и шаблоны, а также предоставляющий возможность суперпользователям создавать шаблоны и управлять их статусами, произведена интеграция с SPD-IAM;
- 3) разработан сервис для генерации заданий, сопоставляющий датасеты с шаблонами по маске имени и создающий по ним цепочки обработки и задания;
- 4) создан сервис для опроса DMS, опрашивающий его о готовности входного датасета для каждого готового задания, создающий выходные датасеты для таких заданий и отправляющий задания на обработку в WMS;
- 5) произведена контейнеризация всех приложений и настроена их оркестрация с помощью docker-compose;
- 6) внедрен logger, сохраняющий информацию об актуальном состоянии системы в каждый момент времени.

## Дальнейшие планы

- 1) Разработка сервиса для взаимодействия с WMS;
- 2) Переход к полной асинхронности;
- 3) Тестирование системы.



**Спасибо за  
внимание!**

# Дополнительные слайды

Планы на сервис для генерации заданий и сервис для опроса DMS

## Перейти:

1. к полной асинхронности;
2. от скрипта к полноценному демону.



Планы на сервис для взаимодействия с оператором обработки данных

## Добавить:

1. возможность изменения статусов шаблонов суперпользователями;
2. возможность изменения шаблонов в статусе "LOADED";
3. загрузка шаблона из файла;
4. ~~возможность работать с аккаунтом пользователя.~~ → интеграция с SPD-IAM



Приступить к созданию сервиса для работы с системой управления нагрузкой (WMS). 