

ClearML

для обучения нейронных сетей:
почему он и как мы
с ним подружились

Бурмистров
Владимир инженер-разработчик в отделе
компьютерного зрения

Чем я занимаюсь

EVOCARGO



ML Ops разработка в отделе Восприятия Эвокарго:

- Пайплайн создания датасетов
- Пайплайн log-файлов ВАТС
- Оптимизация гиперпараметров
- Системы визуализации
- Алгоритмы сегментации облаков точек
- Инфраструктура обучения нейронных сетей

Логистика Эвокарго

EVOCARGO

- Автономный логистический сервис на закрытых/контролируемых территориях
- Собственные автономные электрические грузовики
- Свой центр управления флотом



Наши машины

EVOCARCO

Автономные, электрические

Максимальная
грузоподъемность

2 ТОННЫ

Средняя
скорость

20 км/ч

Зарядка
от розетки

6 ЧАСОВ

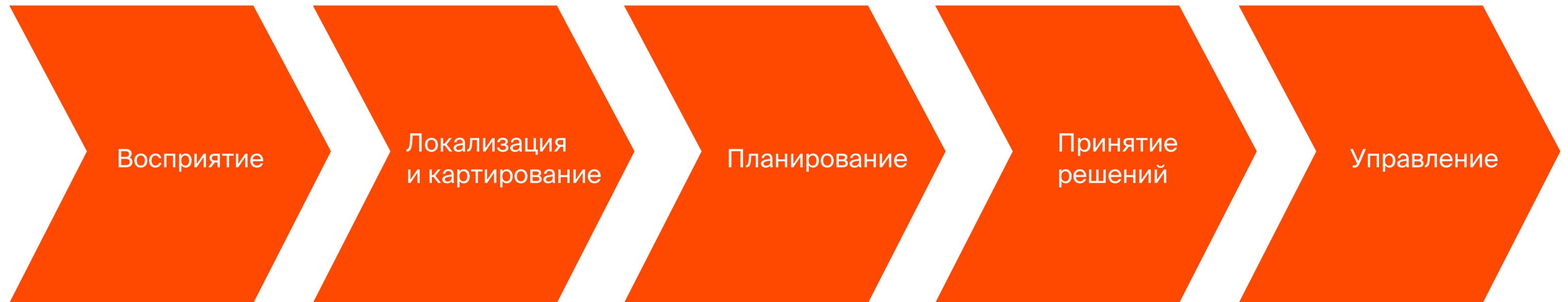
Пробег на одном заряде

до **200** км



АВТОПИЛОТ

EVOCARGO



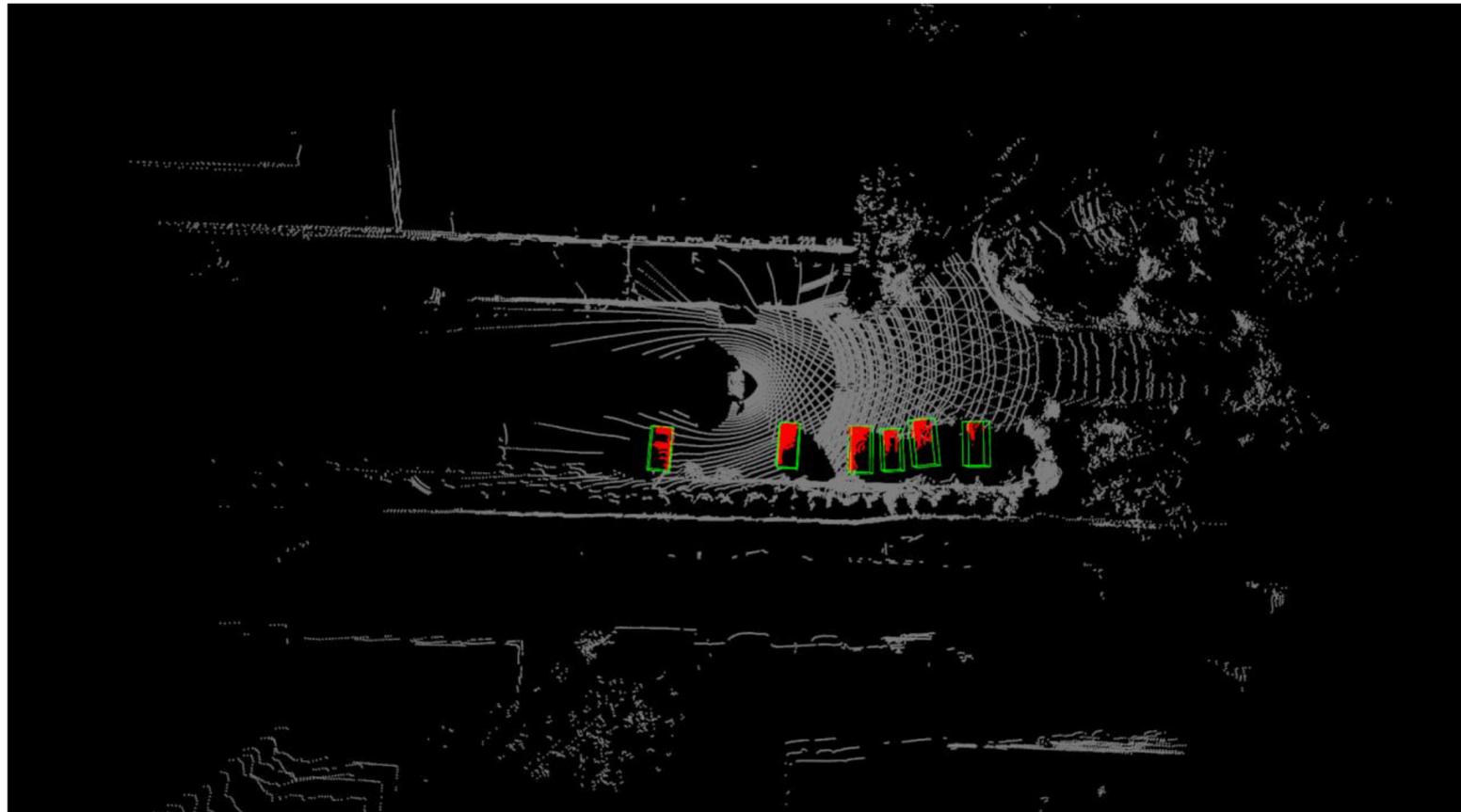
Сенсоры

EVOCARGO



Датасеты

EVOCARCO



- снимки с камер и лидаров
- метаданные (время, координаты, калибровки и т.п.)

С какими сложностями мы столкнулись

EVOCARGO

- как организовать обучение нейросетевых моделей целой командой на одном кластере
- как автоматизировать процесс обучения
 - хранить параметры экспериментов
 - хранить соответствие «модель–датасет»
- как работать с моделями, написанными на разных версиях Python
- как организовать хранение и версионирование датасетов

С какими сложностями мы столкнулись

- как организовать обучение нейросетевых моделей целой командой на одном кластере
- как работать с моделями, написанными на разных версиях Python
- как автоматизировать процесс обучения
 - хранить параметры экспериментов
 - хранить соответствие «модель–датасет»
- как организовать хранение и версионирование датасетов

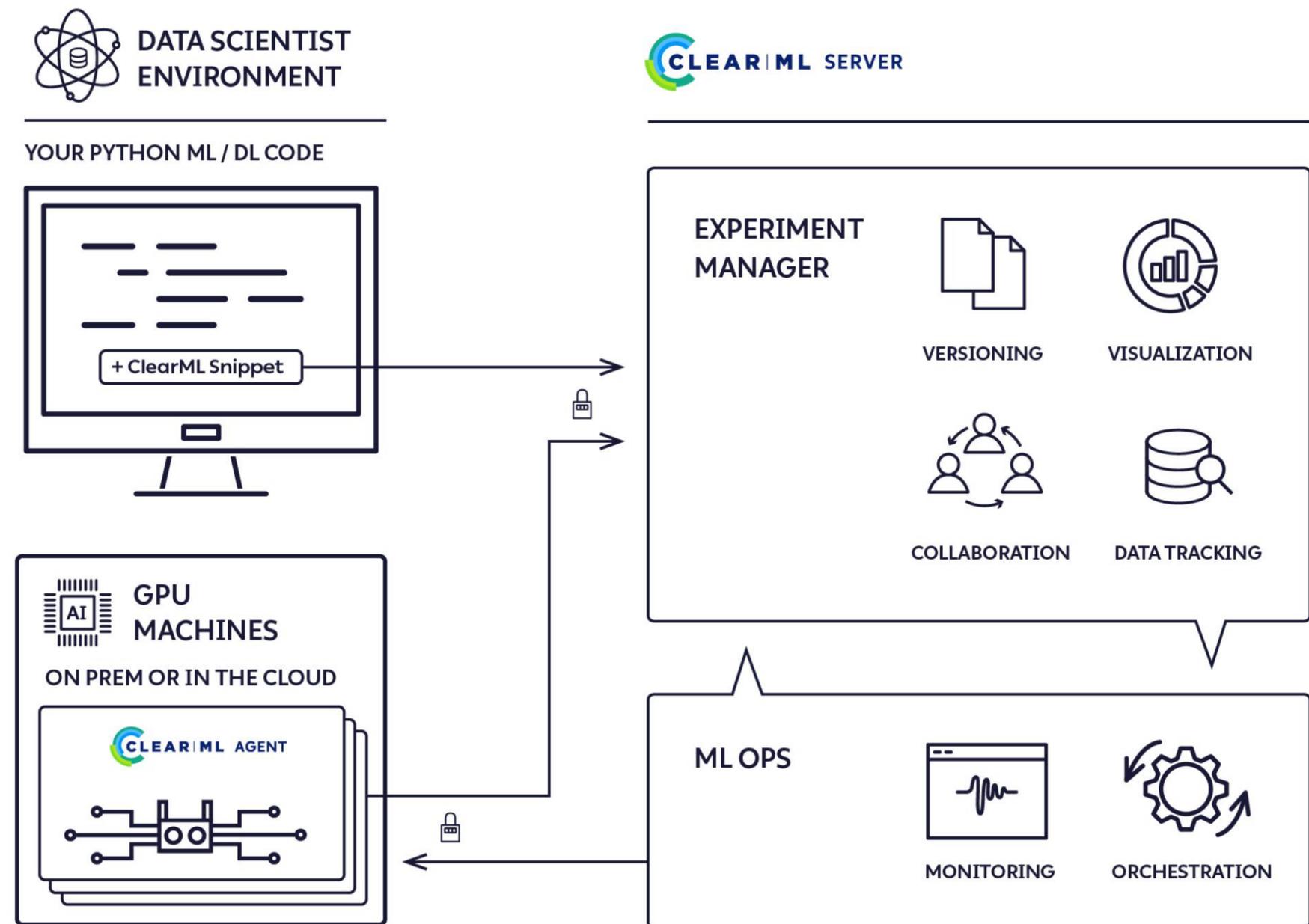
**Эти сложности можно решить
с помощью ClearML**

Сегодня я расскажу

- 1** что такое ClearML
- 2** как он устроен
- 3** его плюсы и минусы
- 4** как мы улучшили процесс обучения нейросетевых моделей с помощью ClearML

Что такое ClearML?

ClearML – платформа для обучения нейросетевых моделей



Функции ClearML

EVOCARGO

- менеджер экспериментов
- хранилище артефактов моделей
- версионирование данных
- оркестратор ресурсов
- отчеты
- динамическое выделение ресурсов кластера
- планировщик экспериментов
- управление доступом учетных записей

github



дока



ClearML web UI

EVOCARCO

The screenshot displays the ClearML web UI dashboard. At the top, there is a navigation bar with the ClearML logo, a search icon, a help icon, and a user profile icon. The main content area is divided into two sections: 'RECENT PROJECTS' and 'RECENT EXPERIMENTS'.

RECENT PROJECTS (VIEW ALL) includes a '+ NEW PROJECT' button and three project cards:

- 2dDetection**: 5K TOTAL, 6 RUNNING, 11 COMPLETED (24 hrs). COMPUTE TIME: 833 DAYS 03:41:06.
- 3dDetection**: 59 TOTAL, 1 RUNNING, 0 COMPLETED (24 hrs). COMPUTE TIME: 07:51:33.
- 3dSegmentation**: 33 TOTAL, 0 RUNNING, 0 COMPLETED (24 hrs). COMPUTE TIME: 21 DAYS 21:08:16.

RECENT EXPERIMENTS (MANAGE WORKERS AND QUEUES) includes a table with the following data:

TYPE	TITLE	PROJECT	STARTED	UPDATED	STATUS
Training	YOLOP_XL_onlyseg_wo_rot	2dDetection	Nov 8 2023 9:16	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_2fps	2dDetection	Nov 8 2023 22:09	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_1fps	2dDetection	Nov 8 2023 21:53	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_10fps	2dDetection	Nov 8 2023 22:21	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_3fps	2dDetection	Nov 8 2023 22:19	Nov 9 2023 19:19	▶ Running

ClearML web UI

EVOCARCO

The screenshot displays the ClearML web UI dashboard. At the top, there is a navigation bar with the ClearML logo, a search icon, a help icon, and a user profile icon. The main content area is divided into two sections: 'RECENT PROJECTS' and 'RECENT EXPERIMENTS'.

RECENT PROJECTS (VIEW ALL) includes a '+ NEW PROJECT' button and three project cards:

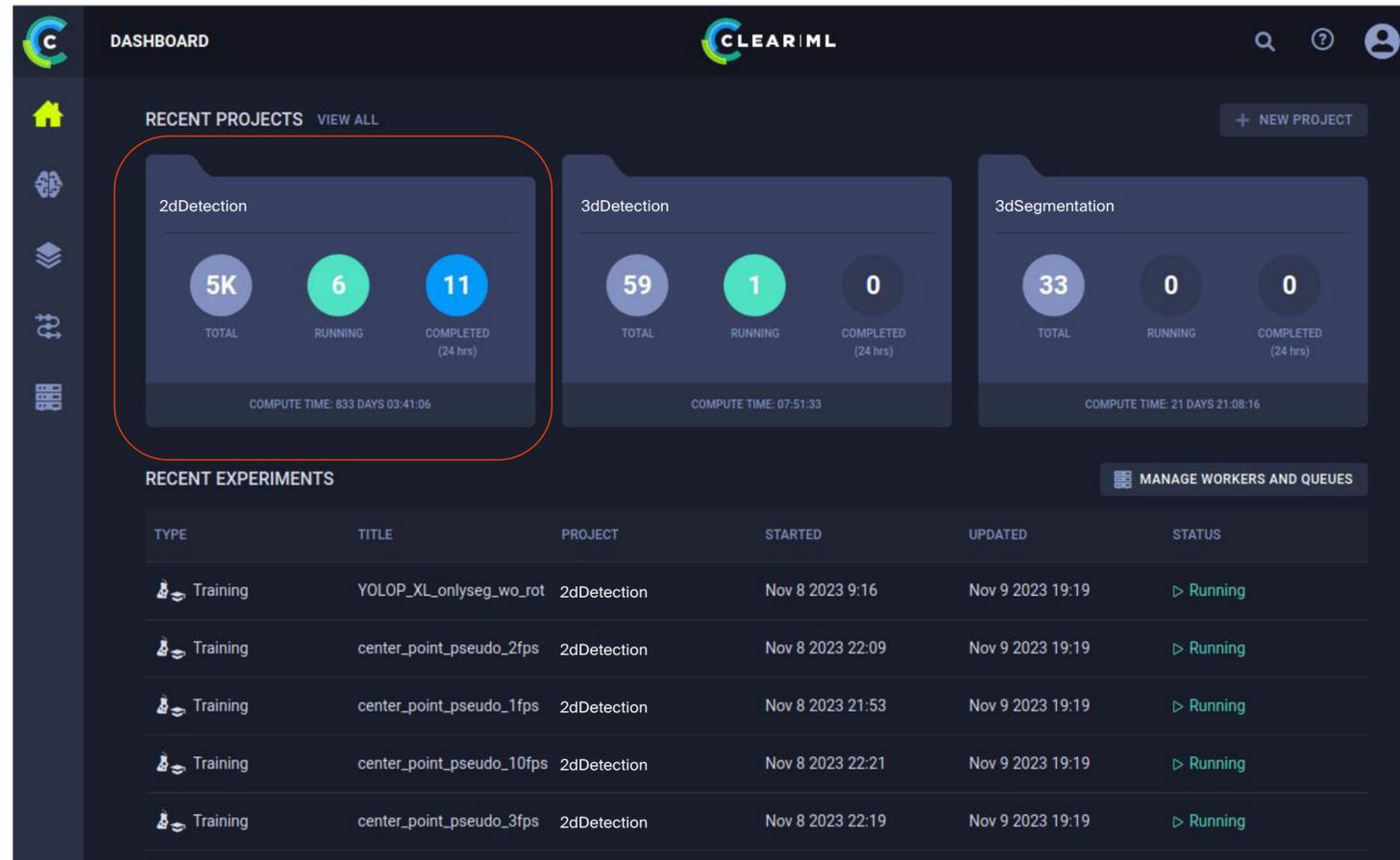
- 2dDetection**: 5K TOTAL, 6 RUNNING, 11 COMPLETED (24 hrs). COMPUTE TIME: 833 DAYS 03:41:06.
- 3dDetection**: 59 TOTAL, 1 RUNNING, 0 COMPLETED (24 hrs). COMPUTE TIME: 07:51:33.
- 3dSegmentation**: 33 TOTAL, 0 RUNNING, 0 COMPLETED (24 hrs). COMPUTE TIME: 21 DAYS 21:08:16.

RECENT EXPERIMENTS (MANAGE WORKERS AND QUEUES) includes a table with the following data:

TYPE	TITLE	PROJECT	STARTED	UPDATED	STATUS
Training	YOLOP_XL_onlyseg_wo_rot	2dDetection	Nov 8 2023 9:16	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_2fps	2dDetection	Nov 8 2023 22:09	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_1fps	2dDetection	Nov 8 2023 21:53	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_10fps	2dDetection	Nov 8 2023 22:21	Nov 9 2023 19:19	▶ Running
Training	center_point_pseudo_3fps	2dDetection	Nov 8 2023 22:19	Nov 9 2023 19:19	▶ Running

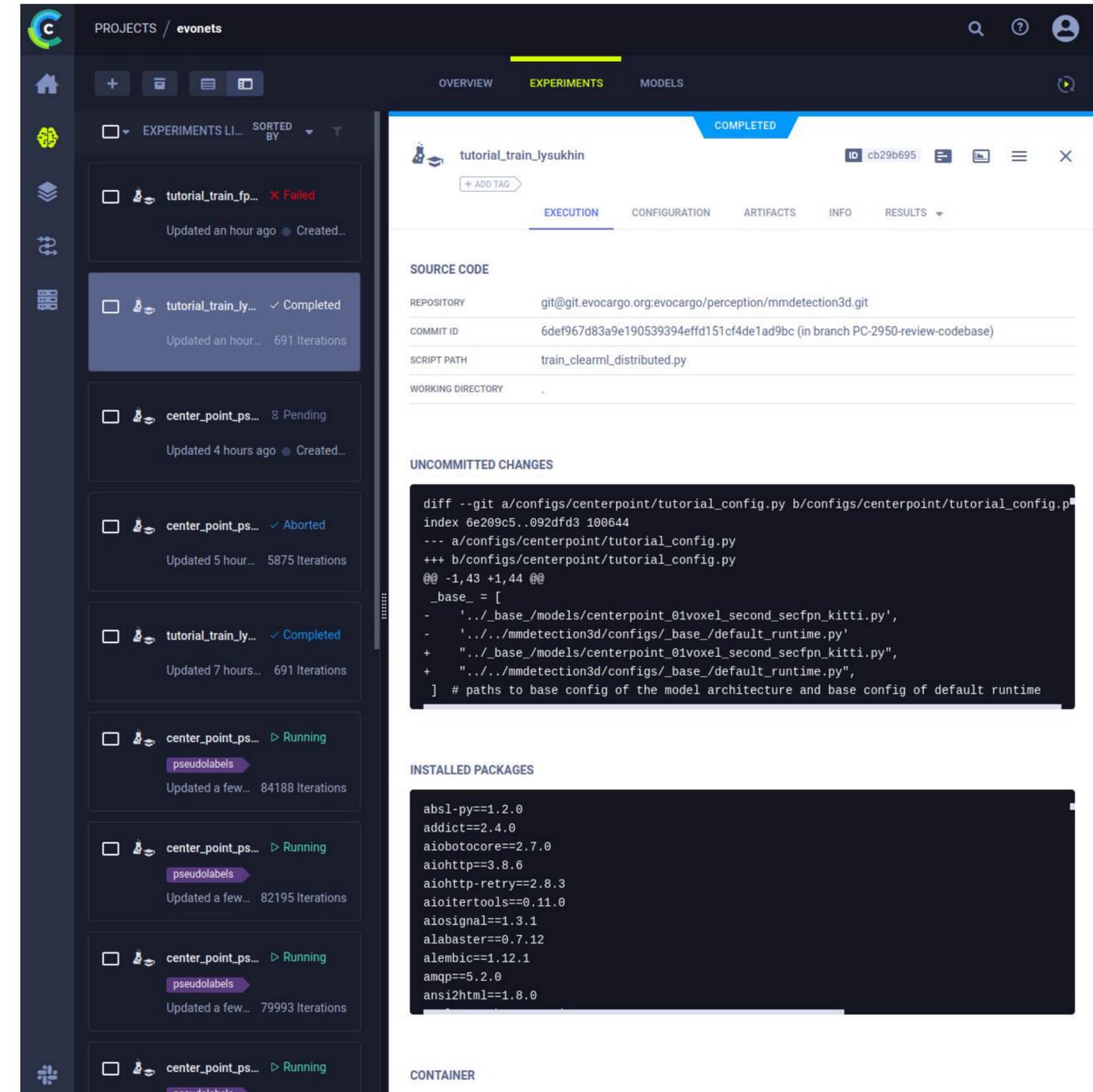
ClearML web UI

EVOCARGO



The dashboard shows a sidebar with navigation icons and a main area with three project cards: 2dDetection, 3dDetection, and 3dSegmentation. Each card displays counts for Total, Running, and Completed items, along with a compute time. Below the cards is a 'RECENT EXPERIMENTS' table.

TYPE	TITLE	PROJECT	STARTED	UPDATED	STATUS
Training	YOLOP_XL_onlyseg_wo_rot	2dDetection	Nov 8 2023 9:16	Nov 9 2023 19:19	Running
Training	center_point_pseudo_2fps	2dDetection	Nov 8 2023 22:09	Nov 9 2023 19:19	Running
Training	center_point_pseudo_1fps	2dDetection	Nov 8 2023 21:53	Nov 9 2023 19:19	Running
Training	center_point_pseudo_10fps	2dDetection	Nov 8 2023 22:21	Nov 9 2023 19:19	Running
Training	center_point_pseudo_3fps	2dDetection	Nov 8 2023 22:19	Nov 9 2023 19:19	Running



The screenshot shows the 'EXPERIMENTS' view for a project named 'evonets'. It lists several experiments with their status (Failed, Completed, Pending, Aborted, Running) and iteration counts. A modal window is open for a specific experiment, showing source code, uncommitted changes, and installed packages.

```
diff --git a/configs/centerpoint/tutorial_config.py b/configs/centerpoint/tutorial_config.py
index 6e209c5..092dfd3 100644
--- a/configs/centerpoint/tutorial_config.py
+++ b/configs/centerpoint/tutorial_config.py
@@ -1,43 +1,44 @@
_base_ = [
- '../_base_/models/centerpoint_01voxel_second_secfpn_kitti.py',
- '../_base_/mmdetection3d/configs/_base_/default_runtime.py'
+ '../_base_/models/centerpoint_01voxel_second_secfpn_kitti.py',
+ '../_base_/mmdetection3d/configs/_base_/default_runtime.py',
] # paths to base config of the model architecture and base config of default runtime
```

```
abs1-py==1.2.0
addict==2.4.0
aiobotocore==2.7.0
aiohttp==3.8.6
aiohttp-retry==2.8.3
aiointertools==0.11.0
aiosignal==1.3.1
alabaster==0.7.12
alembic==1.12.1
amqp==5.2.0
ansi2html==1.8.0
```

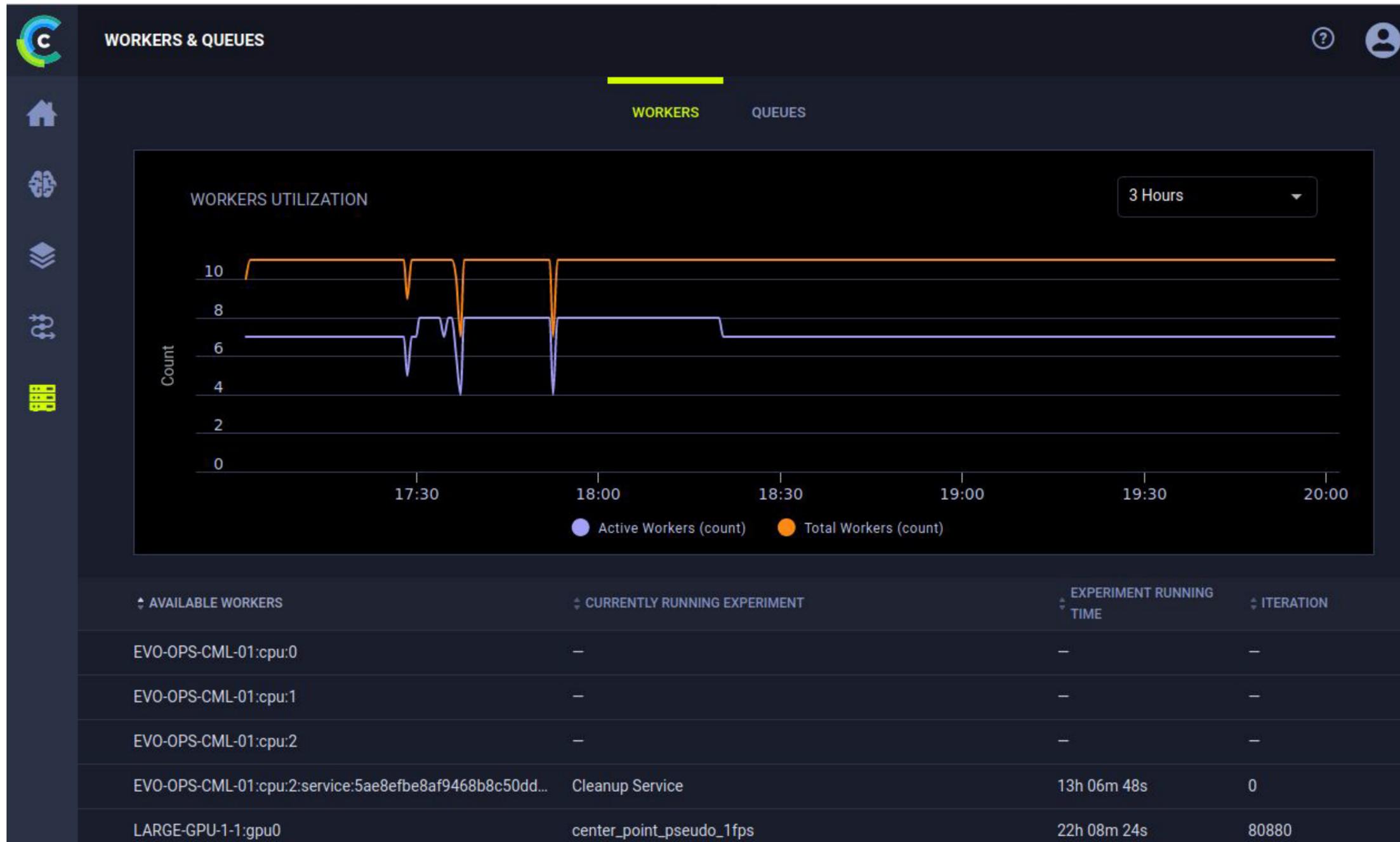
ClearML web UI

EVOCARCO

The screenshot displays the ClearML web interface for managing datasets. The main area shows a grid of dataset cards, each representing a different dataset. Each card includes the dataset name, the time it was last updated, the number of versions, and the file size of the latest version.

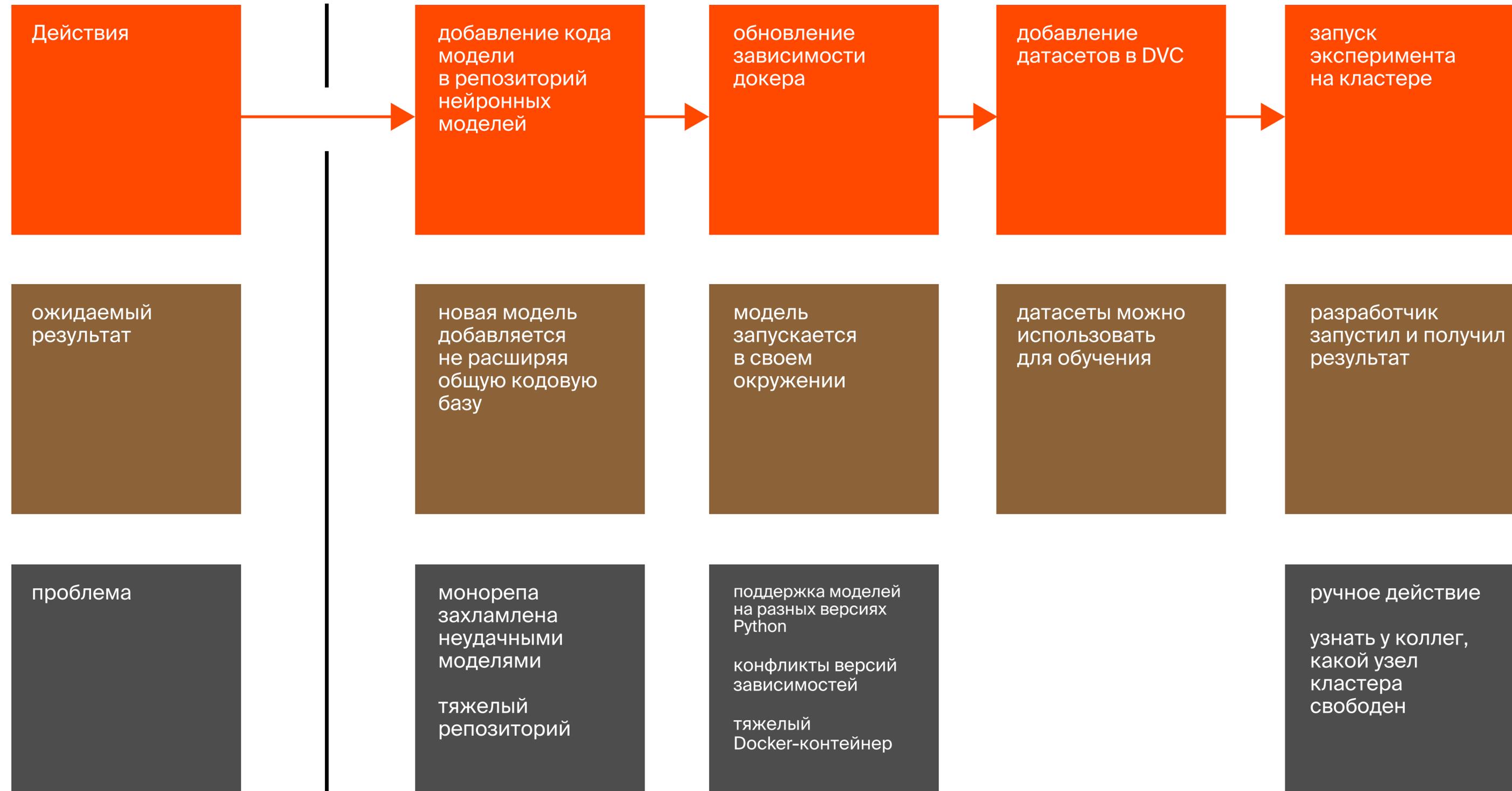
Dataset Name	Updated	Versions	Files	Size
bags_historical	Updated a year ago	3	49K	12GB
bags_historical_debag	Updated a year ago	1	40K	8.5GB
bags_historical_updated	Updated a year ago	1	40K	8.5GB
bags_side_cameras	Updated a year ago	2	56K	11GB
bags_side_cameras_new	Updated a year ago	1	8K	1.3GB
bdd100k	Updated a year ago	1	340K	7.9GB

ClearML web UI

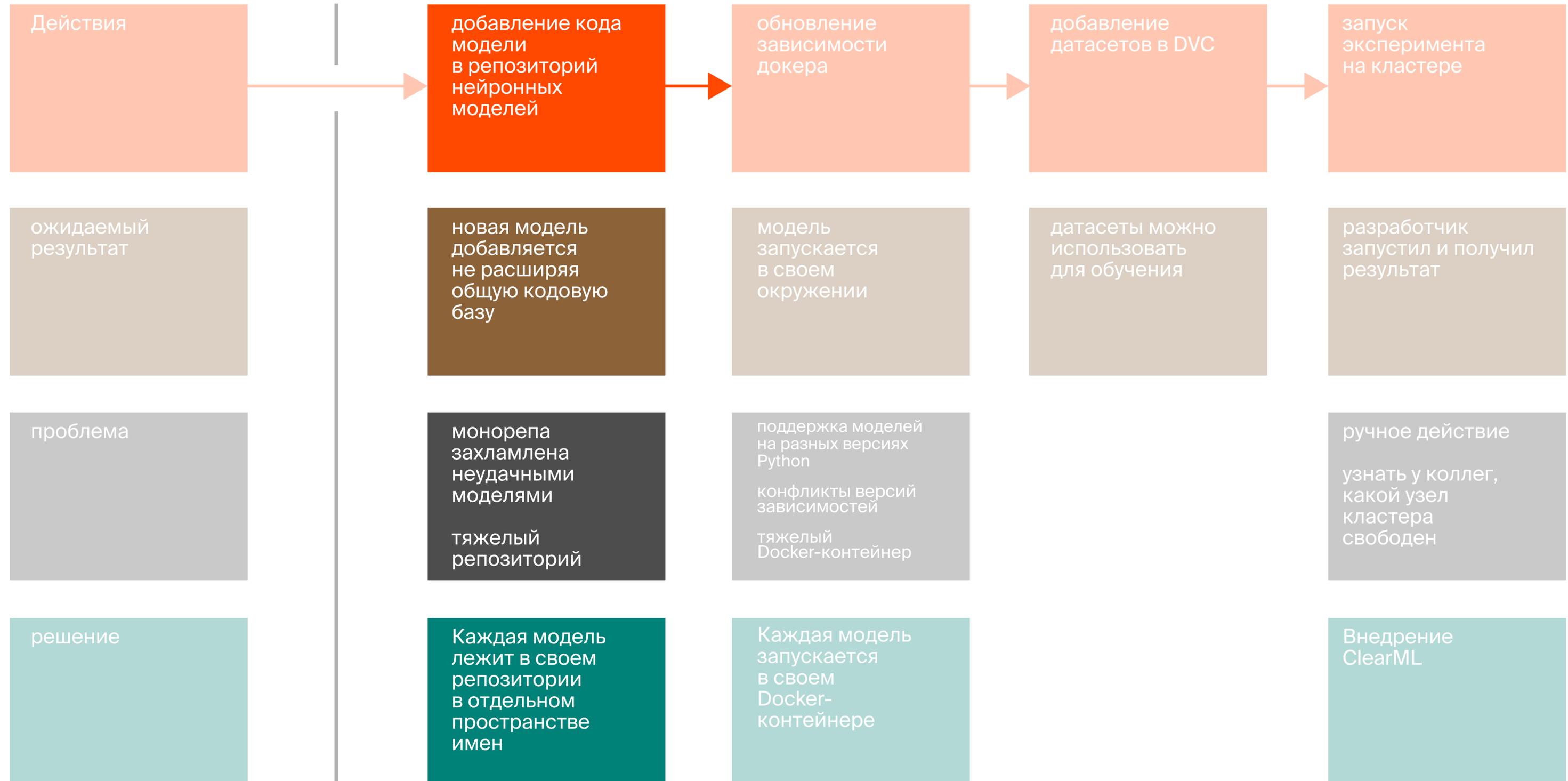


Обучение нейросетевых моделей

На компьютере разработчика



На компьютере разработчика



1. Разделили монорепозиторий

Было



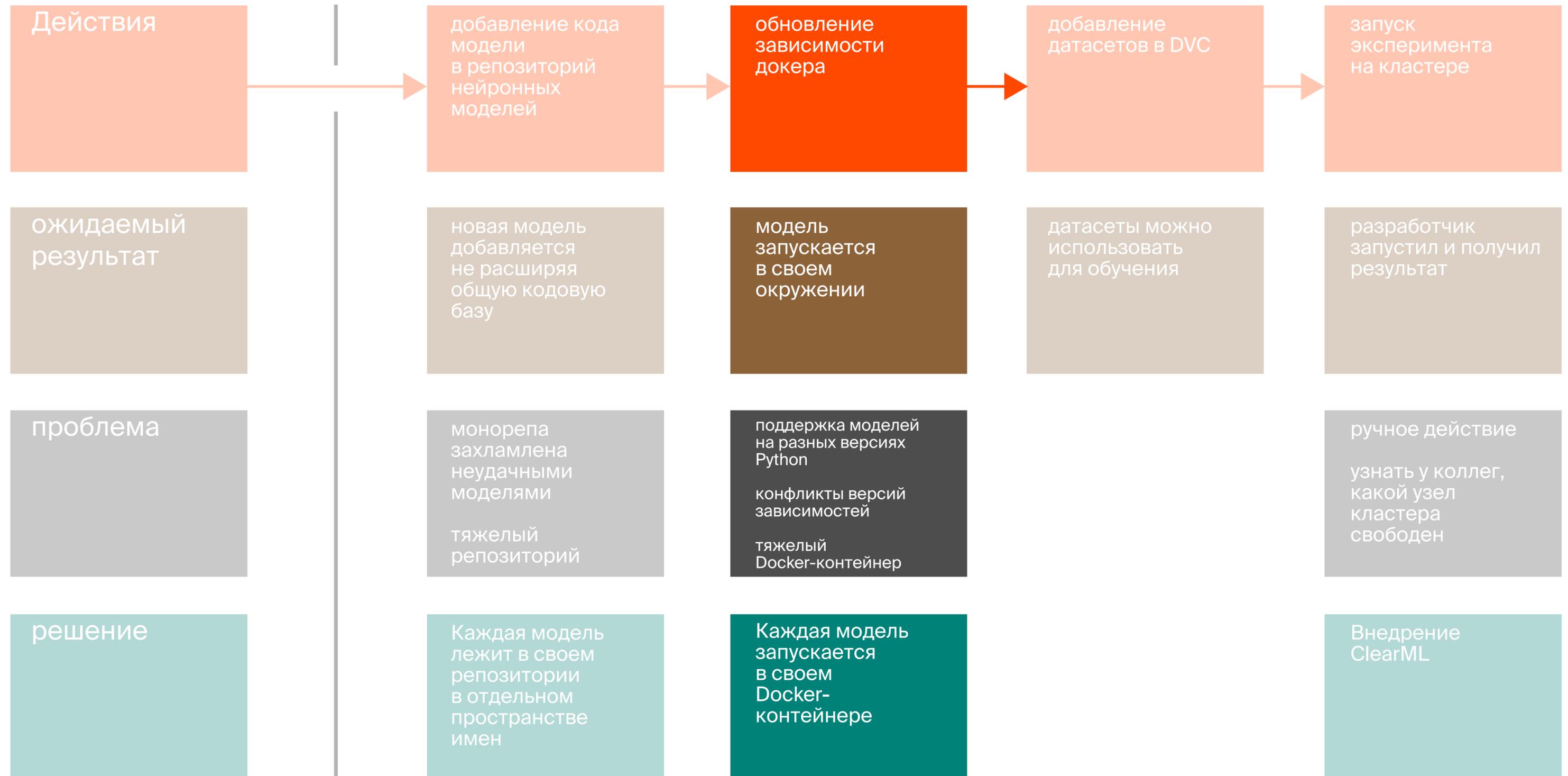
- nn-module (repository)
 - utils (subpackage)
 - ml_model_1 (subpackage)
 - ml_model_2 (subpackage)
 - data (folder)
 - DockerFile (file)

Стало

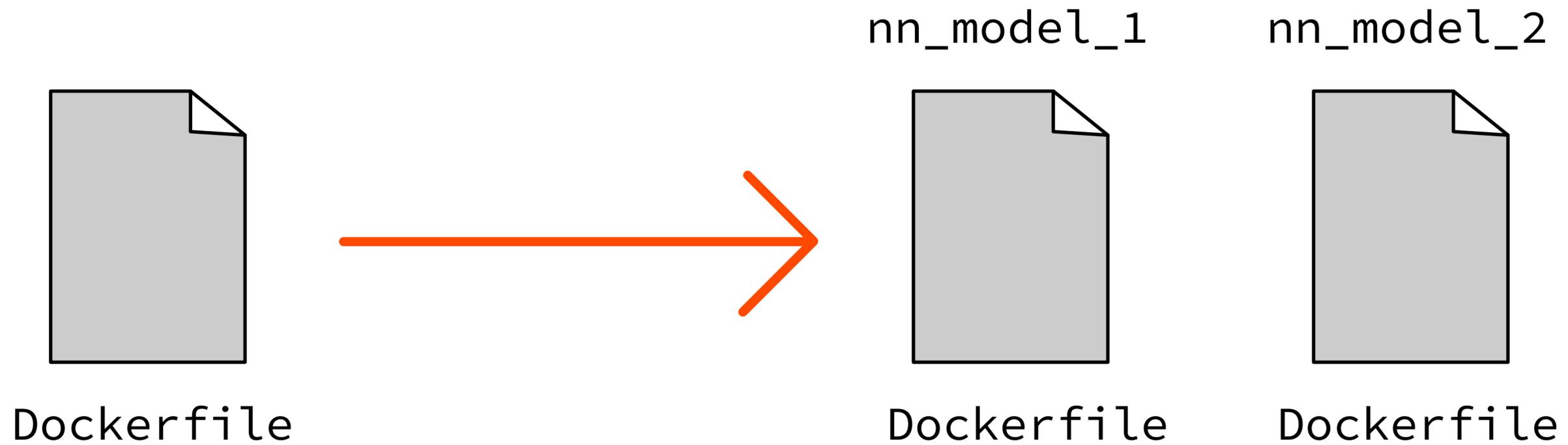


- perception (namespace)
 - neural_network (namespace)
 - ml_model_1 (repository)
 - ml_model_2 (repository)
 - utils (namespace)
 - nn-utils (repository)
 - clearml-client (repository)
 - perception-docker (repository)
 - perception-datasets (repository)

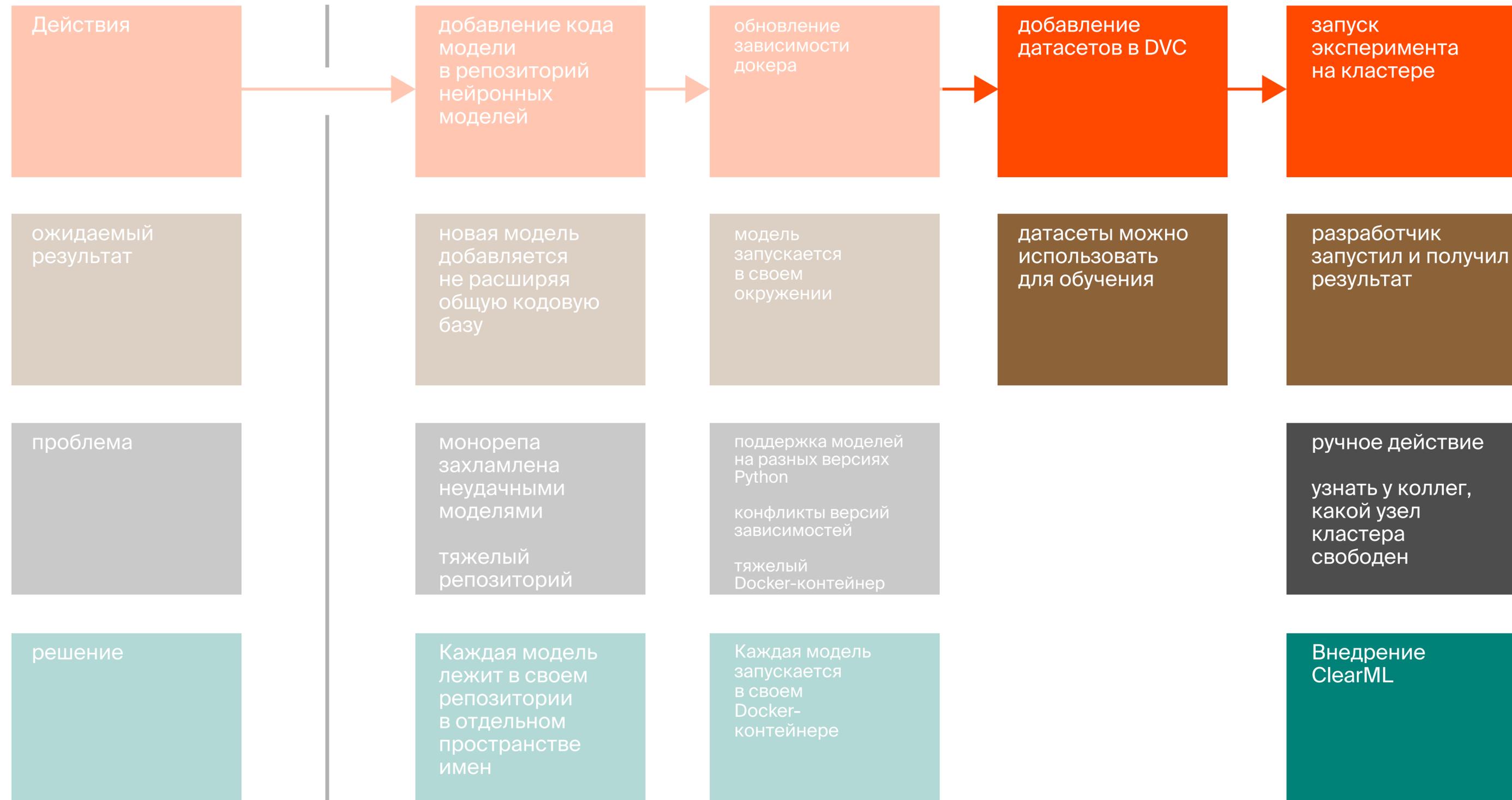
На компьютере разработчика



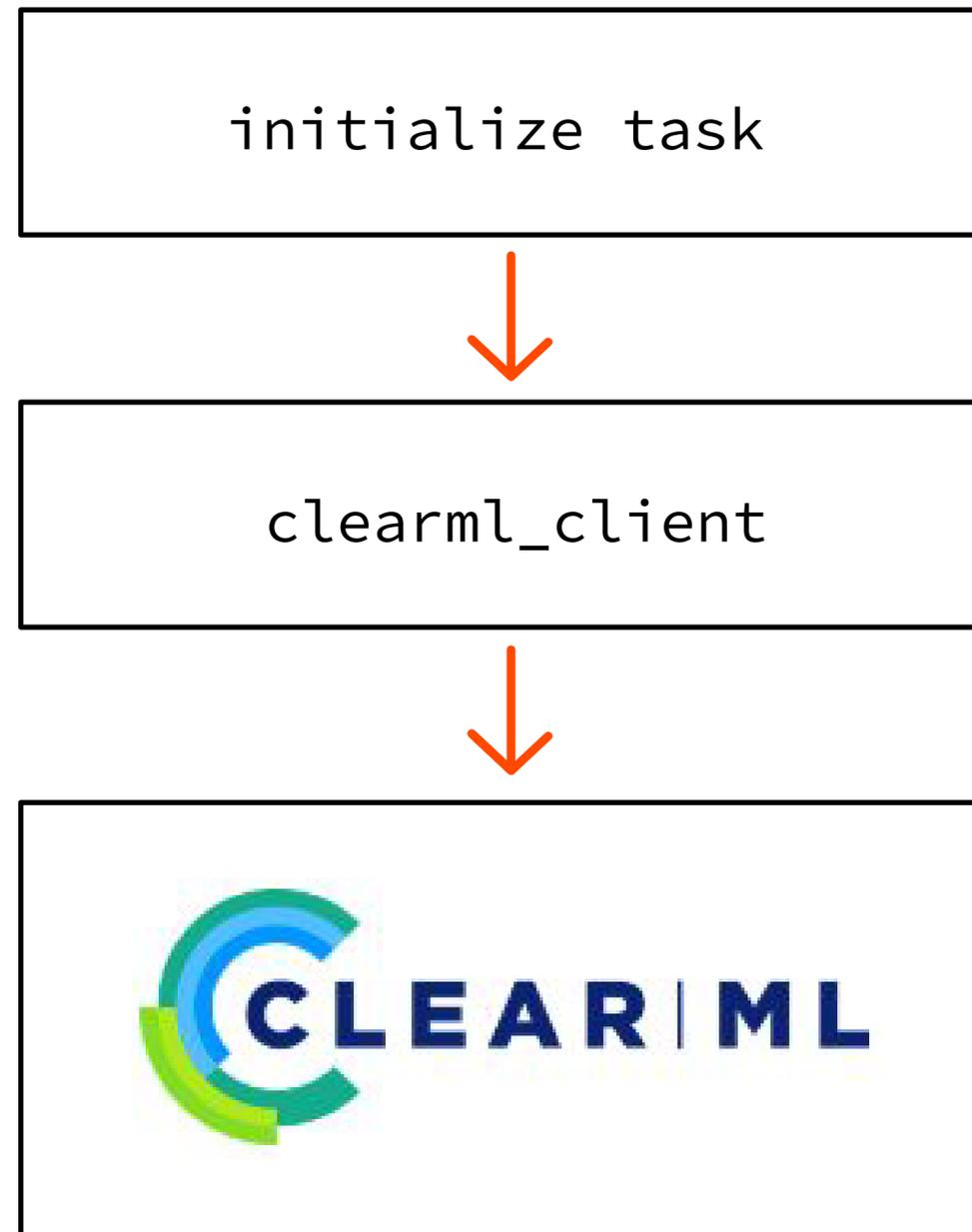
Обновление зависимостей



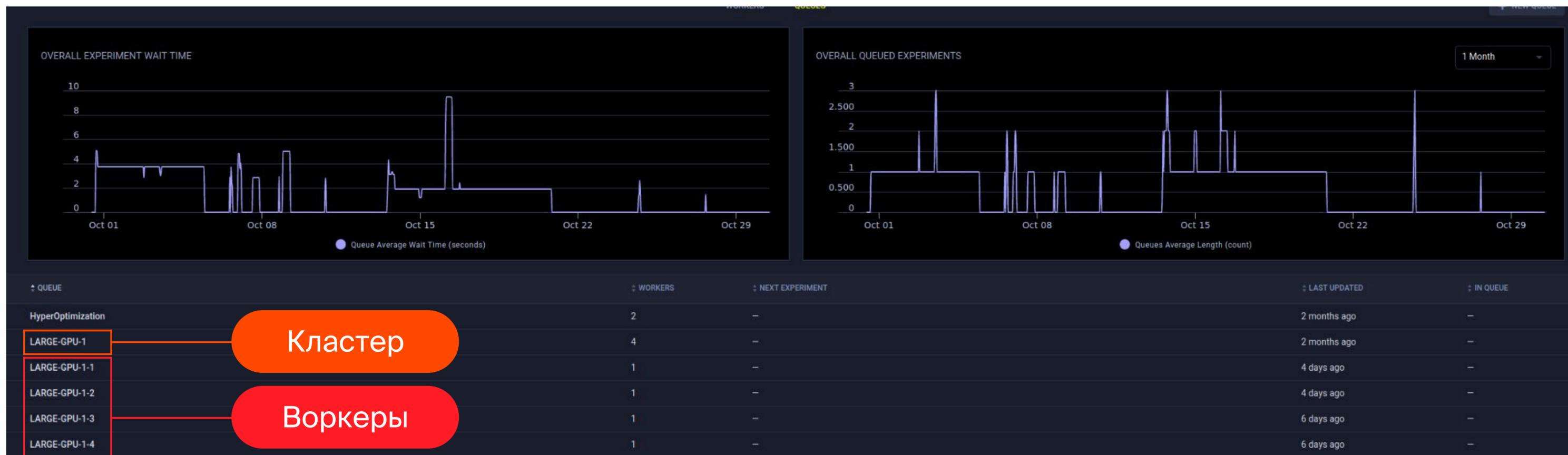
На компьютере разработчика



Сделали обертку над ClearML API

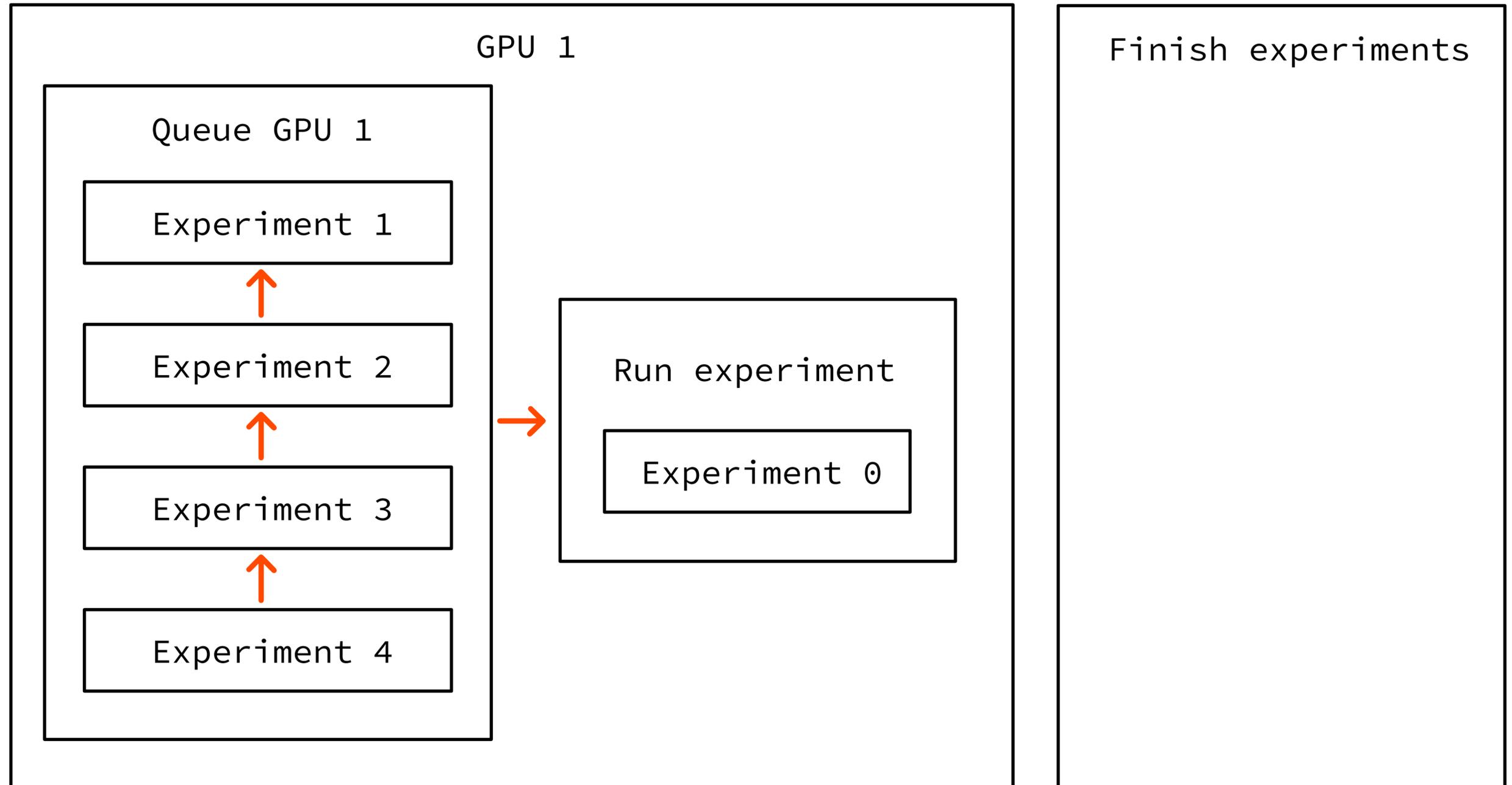


Кластеры серверов обучения



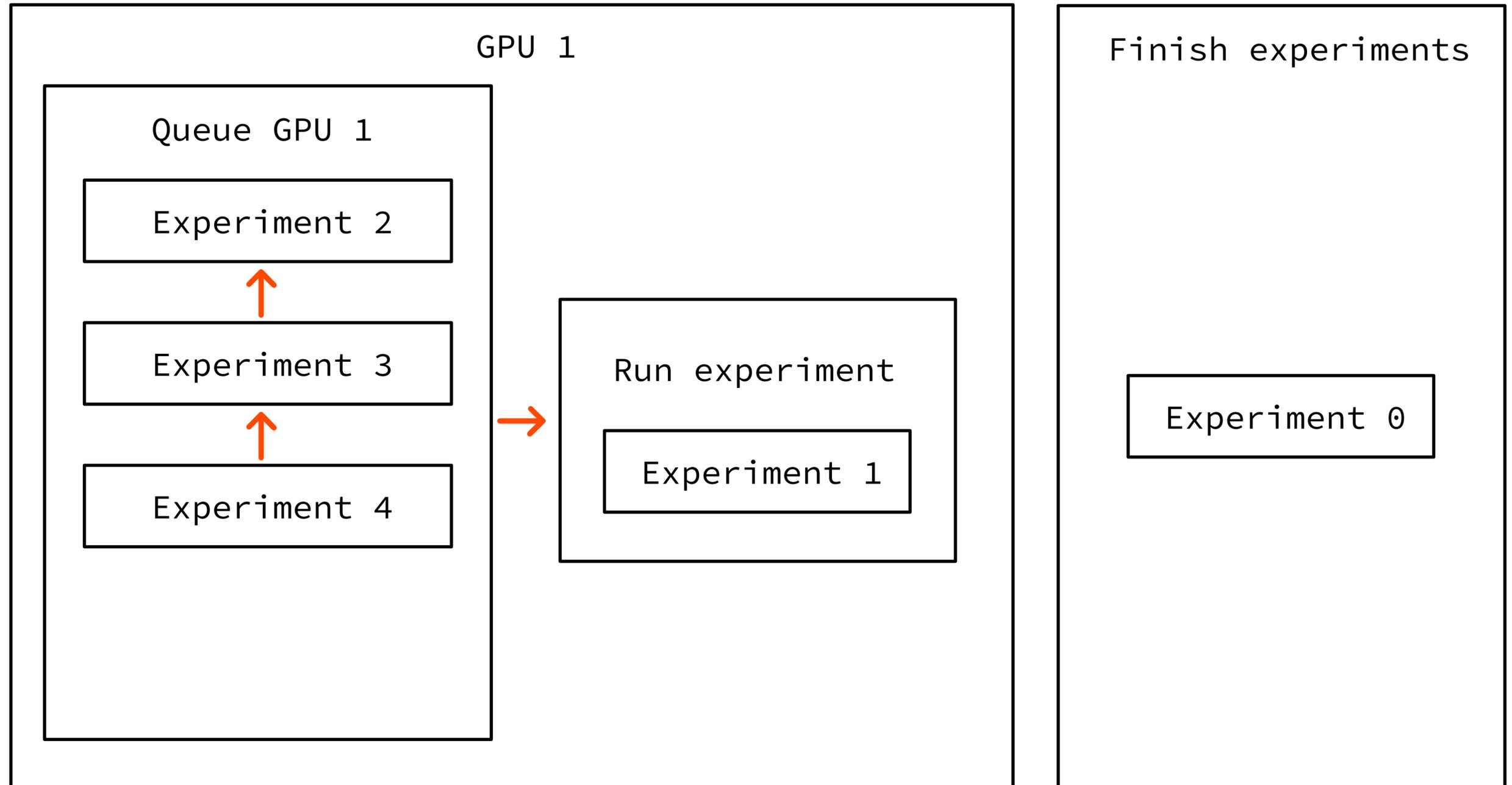
Очередь обучения

ШАГ 1



Очередь обучения

ШАГ 2



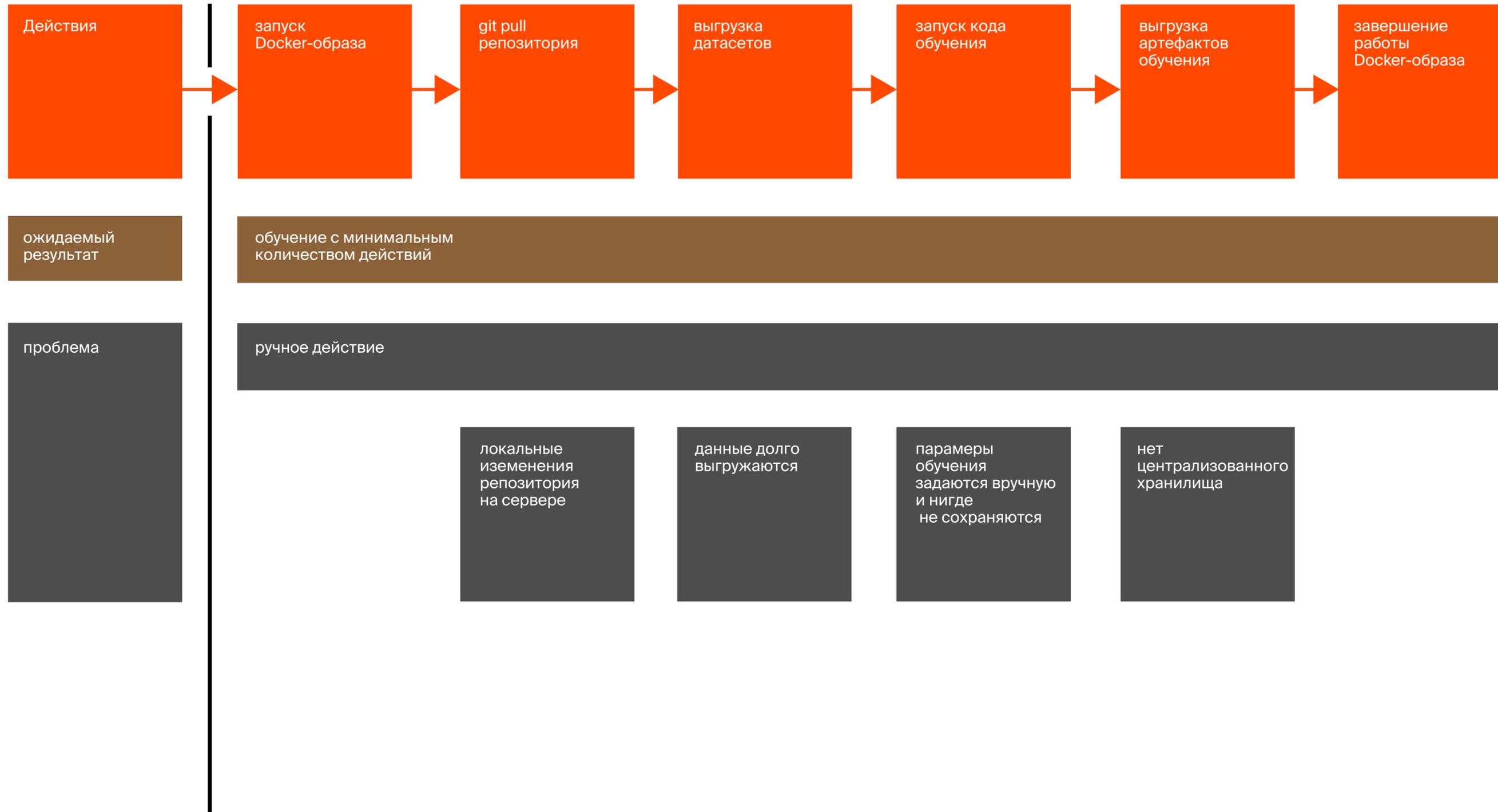
Проекты обучения

Деление экспериментов

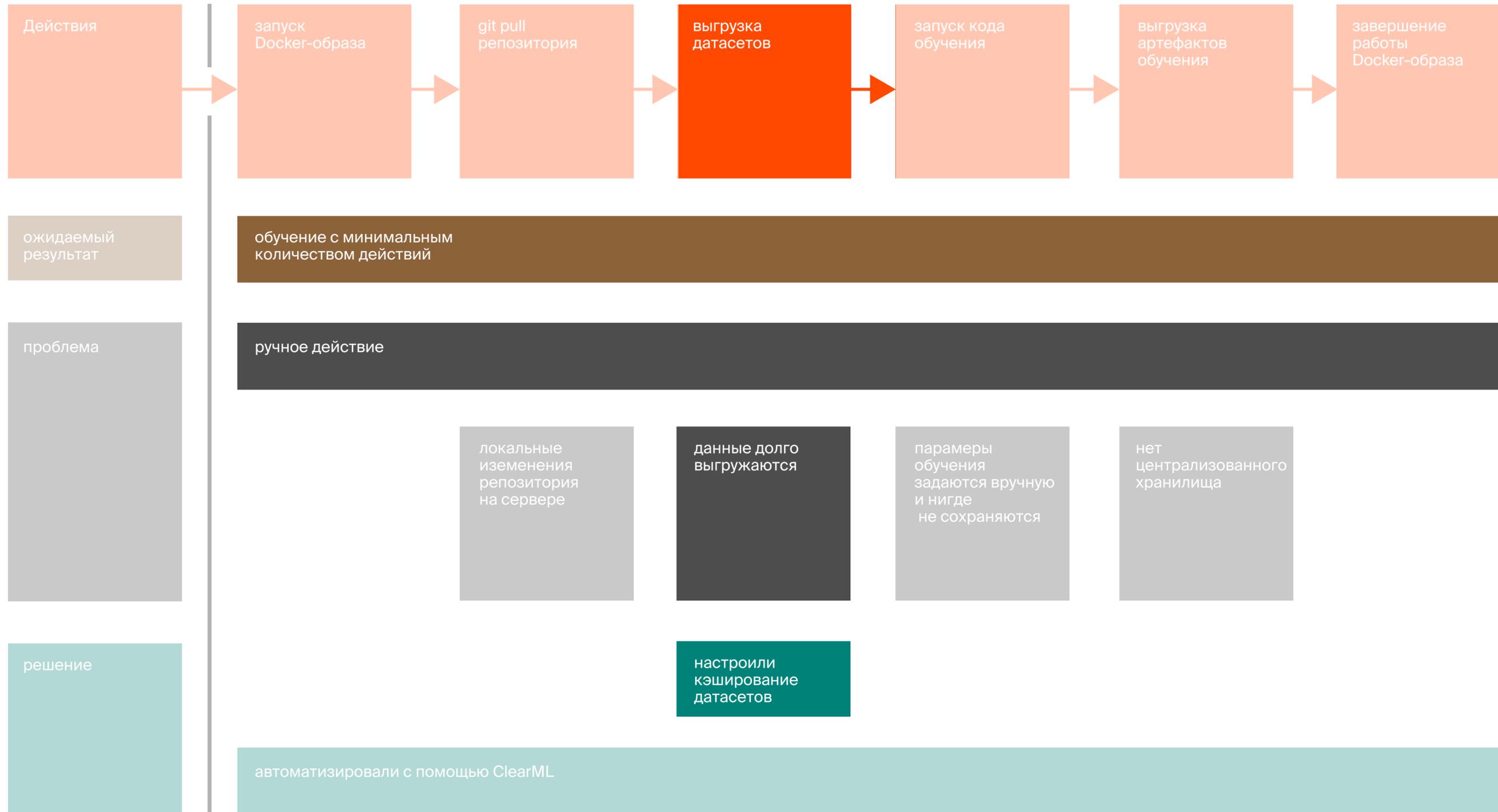
- по типам (обучение сетей, оптимизация гиперпараметров и тд.)
- по командам



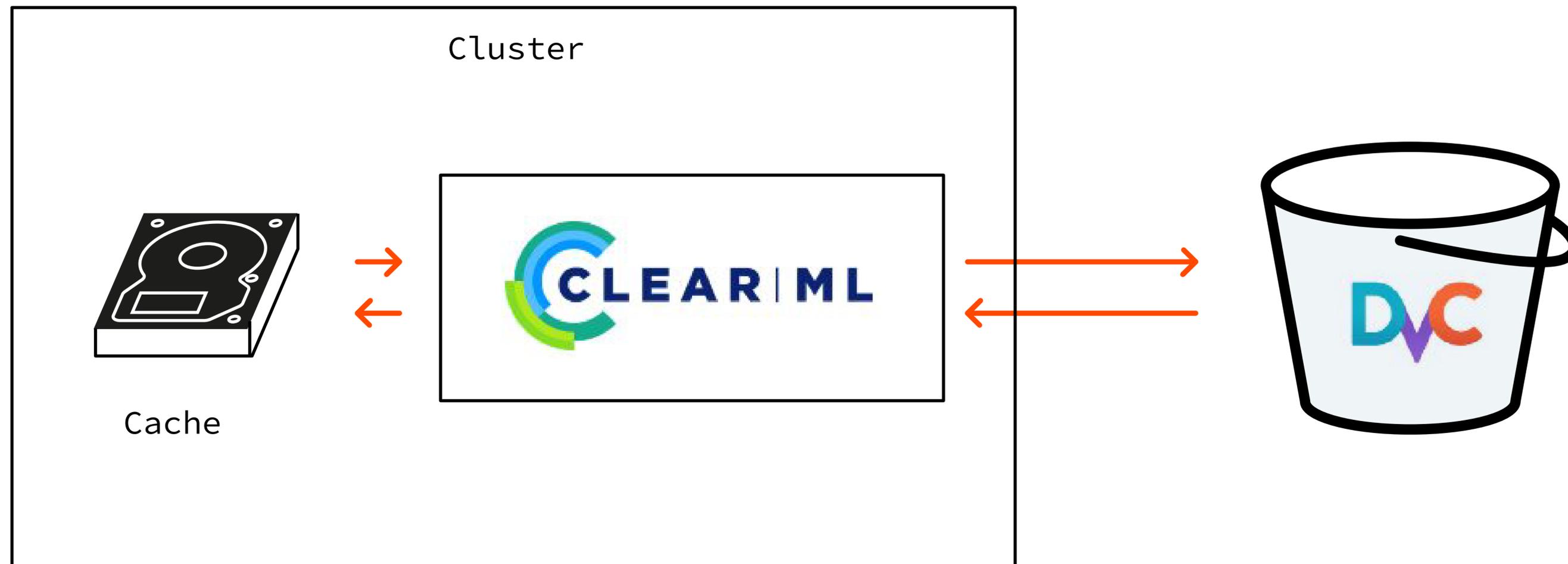
На кластере



На кластере



Настроили кэширование датасетов на сервере обучения



Работа с данными

→ хранение, тегирование и preview датасетов

The screenshot displays the Evocargo web interface for a dataset named 'coco_data'. The breadcrumb navigation at the top reads 'PROJECTS / Datasets / .datasets / coco_data'. A blue 'COMPLETED' status bar is visible. The dataset name 'coco_data' is shown with an ID of '08298917' and a '+ ADD TAG' button. Below this, a horizontal menu contains tabs for 'EXECUTION', 'CONFIGURATION', 'ARTIFACTS', 'INFO', 'CONSOLE', 'SCALARS', 'PLOTS', and 'DEBUG SAMPLES'. The 'DEBUG SAMPLES' tab is selected. A 'Metric: Images' dropdown and 'Iterations: 0' counter are present. A grid of 10 image thumbnails is shown, each with a filename below it: 00000061624.jpg, 00000102460.jpg, 00000298736.jpg, 00000314714.jpg, 00000357941.jpg, 00000385337.jpg, 00000422654.jpg, 00000432424.jpg, 00000473983.jpg, and 00000475042.jpg. The interface includes a dark sidebar on the left with various icons and a top navigation bar with a search icon and a user profile icon.

Работа с данными

EVOCARGO

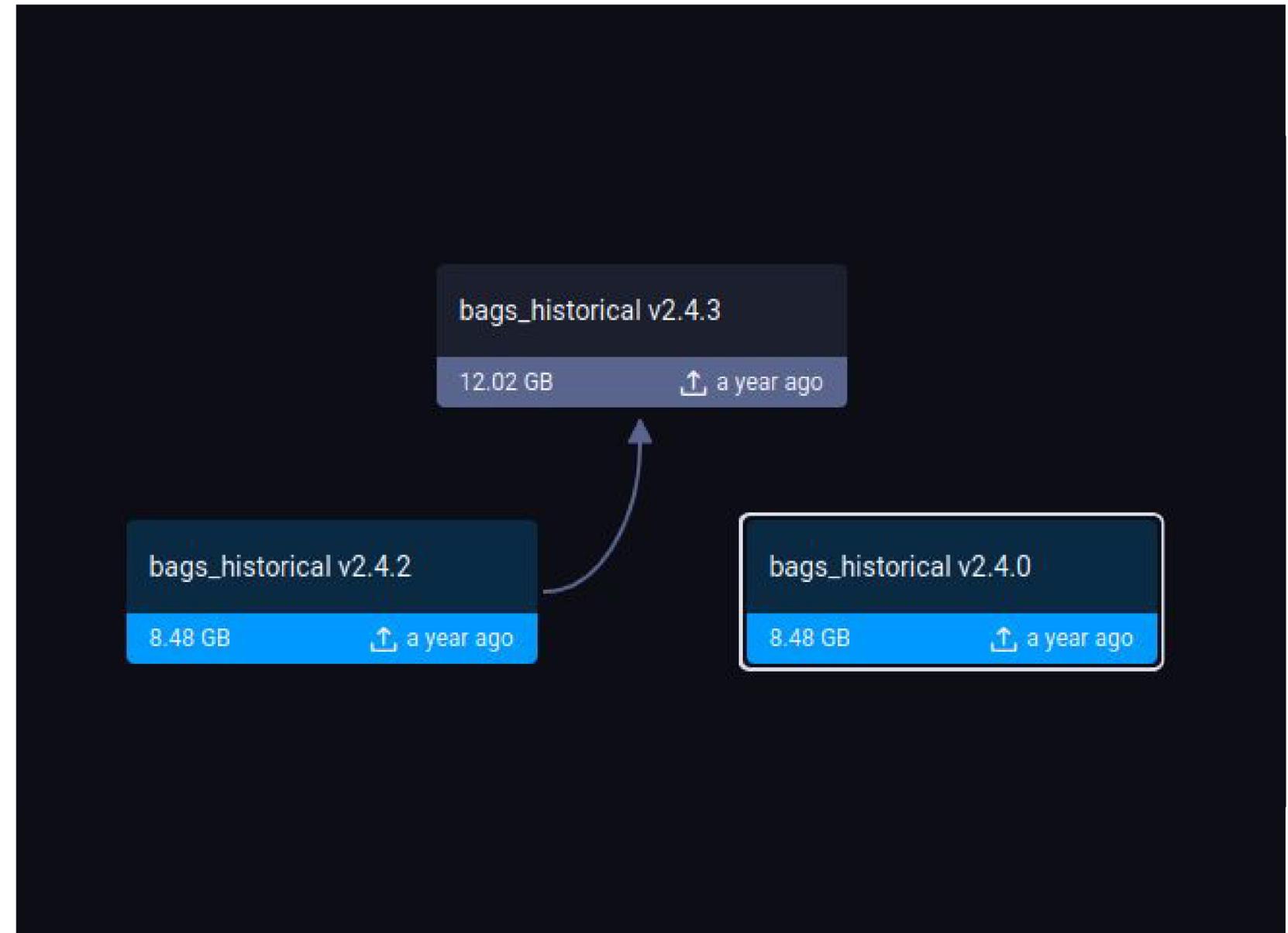
- хранение, тегирование и preview датасетов
- консоль для просмотра структуры данных без выгрузки

Работа с данными

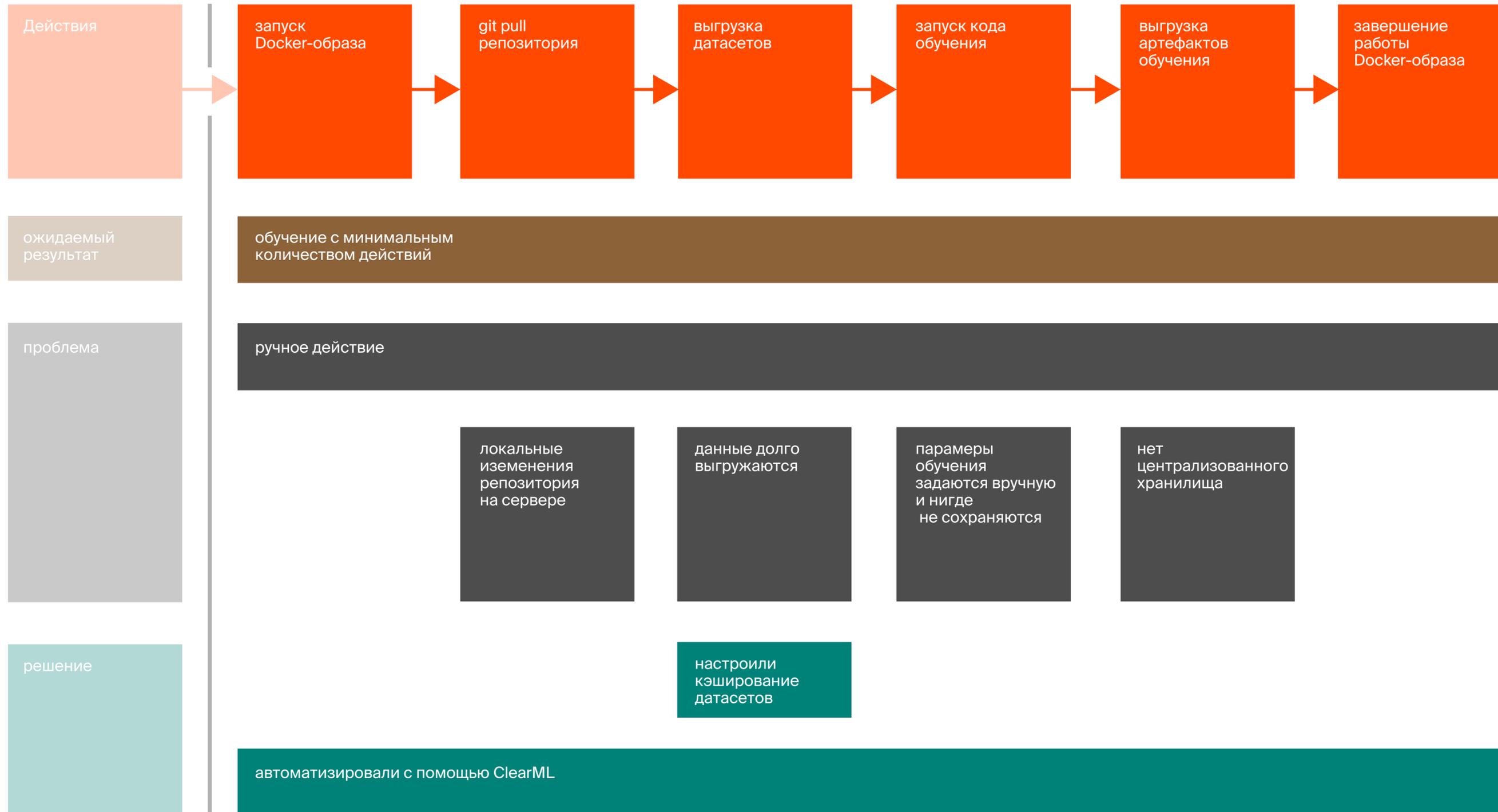
- хранение, тегирование и preview датасетов
- консоль для просмотра структуры данных без выгрузки
- интерфейс для загрузки/выгрузки датасетов

Проблемы с версиями датасетов

- множественное наследование версий
- нельзя задать версию самостоятельно



На кластере



Логгер обучения

```
2023-07-21 13:48:29 Train Epoch: 1 [0/30000 (0%)] worker: 1, Loss: 2.311087
Train Epoch: 1 [0/30000 (0%)] worker: 0, Loss: 2.335604
Train Epoch: 1 [640/30000 (2%)] worker: 1, Loss: 1.383415
Train Epoch: 1 [640/30000 (2%)] worker: 0, Loss: 1.362116
Train Epoch: 1 [1280/30000 (4%)] worker: 1, Loss: 0.939890
Train Epoch: 1 [1280/30000 (4%)] worker: 0, Loss: 0.697477
Train Epoch: 1 [1920/30000 (6%)] worker: 1, Loss: 0.379707
Train Epoch: 1 [1920/30000 (6%)] worker: 0, Loss: 0.380613
Train Epoch: 1 [2560/30000 (9%)] worker: 1, Loss: 0.360629
Train Epoch: 1 [2560/30000 (9%)] worker: 0, Loss: 0.458939
Train Epoch: 1 [3200/30000 (11%)] worker: 1, Loss: 0.363109
Train Epoch: 1 [3200/30000 (11%)] worker: 0, Loss: 0.418063
Train Epoch: 1 [3840/30000 (13%)] worker: 1, Loss: 0.240043
Train Epoch: 1 [3840/30000 (13%)] worker: 0, Loss: 0.196349
Train Epoch: 1 [4480/30000 (15%)] worker: 0, Loss: 0.221501
Train Epoch: 1 [4480/30000 (15%)] worker: 1, Loss: 0.113831
Train Epoch: 1 [5120/30000 (17%)] worker: 0, Loss: 0.236115
Train Epoch: 1 [5120/30000 (17%)] worker: 1, Loss: 0.115690
Train Epoch: 1 [5760/30000 (19%)] worker: 0, Loss: 0.328873
Train Epoch: 1 [5760/30000 (19%)] worker: 1, Loss: 0.070968
Train Epoch: 1 [6400/30000 (21%)] worker: 0, Loss: 0.223515
Train Epoch: 1 [6400/30000 (21%)] worker: 1, Loss: 0.251144
Train Epoch: 1 [7040/30000 (23%)] worker: 0, Loss: 0.214814
Train Epoch: 1 [7040/30000 (23%)] worker: 1, Loss: 0.104148
Train Epoch: 1 [7680/30000 (26%)] worker: 0, Loss: 0.133469
Train Epoch: 1 [7680/30000 (26%)] worker: 1, Loss: 0.163791
Train Epoch: 1 [8320/30000 (28%)] worker: 0, Loss: 0.207600
Train Epoch: 1 [8320/30000 (28%)] worker: 1, Loss: 0.135818
Train Epoch: 1 [8960/30000 (30%)] worker: 0, Loss: 0.168563
```

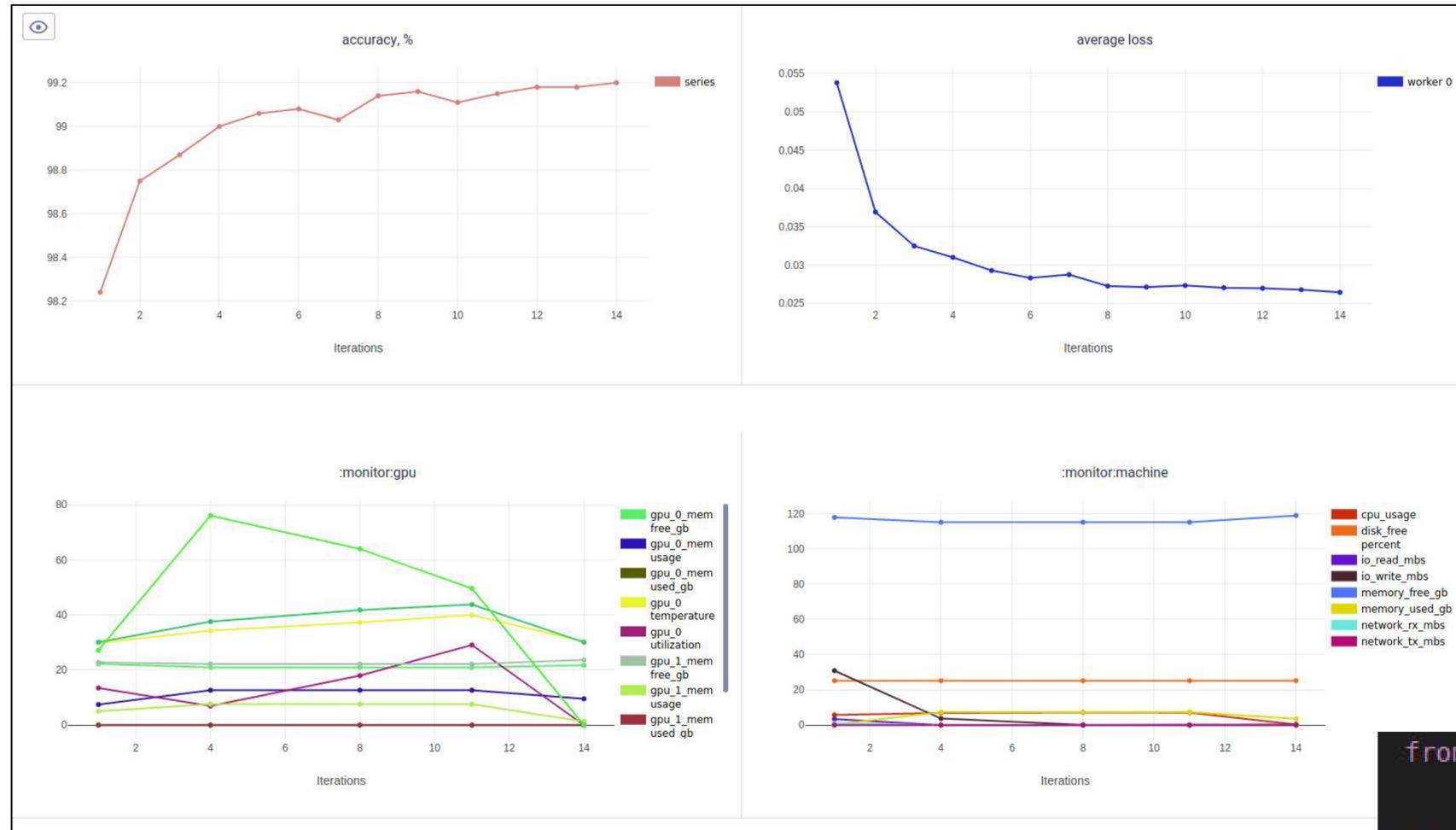
```
import logging
from clearml import Logger

|

logger = Logger.current_logger()

logger.report_text(
    log_data, level=logging.DEBUG, print_console=True
)
```

Логгер обучения



```
from clearml import Logger

logger = Logger.current_logger()

logger.report_scalar("average loss", "series", test_loss, epoch)
logger.report_scalar(
    "accuracy, %",
    "series",
    100.0 * correct / len(test_loader.dataset),
    epoch,
)
```

Результаты эксперимента

The screenshot displays the Evocargo ML experiment results interface. The left sidebar shows a list of experiments with their status (Completed, Failed, Aborted) and iteration counts. The main panel shows details for a 'trailers' experiment, including source code, uncommitted changes, installed packages, and container information.

EXPERIMENTS LIST

Experiment Name	Status	Updated	Created by	Iterations
trailers	Completed	Updated 13 days ago	Created by clearml_user	59910
trailers	Failed	Updated a month ago	Created by clearml_user	
trailers	Failed	Updated a month ago	Created by clearml_user	
pv_rcnn++	Aborted	Updated 13 days ago	Created by clearml_user	79955
YOLOP_obst_aug_new_data_coco	Failed	Updated a month ago	Created by clearml_user	
YOLOP_GAIN_coco	Failed	Updated a month ago	Created by clearml_user	
pv_rcnn++	Aborted	Updated a month ago	Created by clearml_user	9035
pv_rcnn++	Aborted	Updated a month ago	Created by clearml_user	649
pv_rcnn++	Aborted	Updated a month ago	Created by clearml_user	458

trailers (COMPLETED)

EXECUTION | CONFIGURATION | ARTIFACTS | INFO | CONSOLE | SCALARS | PLOTS | DEBUG SAMPLES

SOURCE CODE

REPOSITORY: git@git.evocargo.org:evocargo/perception/autoannotation.git
COMMIT ID: 91f3557100f0032906f2c374afaacdb1b1907619 (in branch track_class)
SCRIPT PATH: train.py
WORKING DIRECTORY: .

UNCOMMITTED CHANGES

```
diff --git a/.dvc/config b/.dvc/config
index 9c3df29..0409947 100755
--- a/.dvc/config
+++ b/.dvc/config
@@ -1,5 +1,9 @@
 [core]
   remote = models
+[cache]
+  dir = /dvc_cache
+  shared = group
+  type = symlink
```

INSTALLED PACKAGES

```
addict==2.4.0
aiobotocore==2.5.4
aiofiles==22.1.0
aiohttp==3.8.5
aiohttp-retry==2.8.3
aiointertools==0.11.0
aiosignal==1.3.1
aiosqlite==0.19.0
alabaster==0.7.13
albumintations==0.5.2
alembic==1.12.0
```

CONTAINER

USER PROPERTIES ^

Properties

HYPER PARAMETERS ^

Args

Hydra

CONFIGURATION OBJECTS ^

OmegaConf

OMEGACONF

TYPE OmegaConf YAML

DESCRIPTION Full OmegaConf YAML configuration. This is a read-only section, unless 'Hydra/_allow_omegaconf_edit_' is set to True

CONFIGURATION OBJECT

```
name: finetune_v1
build_dataset: true
target_path: data/custom
classes:
- Car
- Pedestrian
- Cyclist
- Trailer
pretrain:
  enabled: true
  weights_path: models/weights
launch:
  model_cfg: models/pv_rcnn+_finetune/model_config.yaml
  data_cfg: models/pv_rcnn+_finetune/data_config.yaml
  batch_size: 8
  epochs: 30
  workers: 8
  extra_tag: default
  ckpt: null
  pretrained_model: models/weights/pv_rcnn+_evo_finnetune/checkpoint_v2_1.pth
  launcher: none
  tcp_port: 18888
  sync_bn: false
  fix_random_seed: true
  ckpt_save_interval: 2
  local_rank: 0
  max_ckpt_save_num: 100
  merge_all_iters_to_one_epoch: false
  set: null
  max_waiting_mins: 0
  start_epoch: 0
  num_epochs_to_eval: 0
  save_to_file: false
```

- INPUT MODELS ^
- epoch_35
- OTHER ^
- epoch_1.pth
- epoch_10.pth
- epoch_100.pth
- epoch_11.pth
- epoch_12.pth
- epoch_13.pth
- epoch_14.pth
- epoch_15.pth
- epoch_16.pth
- epoch_17.pth

MODEL NAME Imported by centerpoint-evo-new-data-014 

CREATING EXPERIMENT

MODEL CONFIGURATION

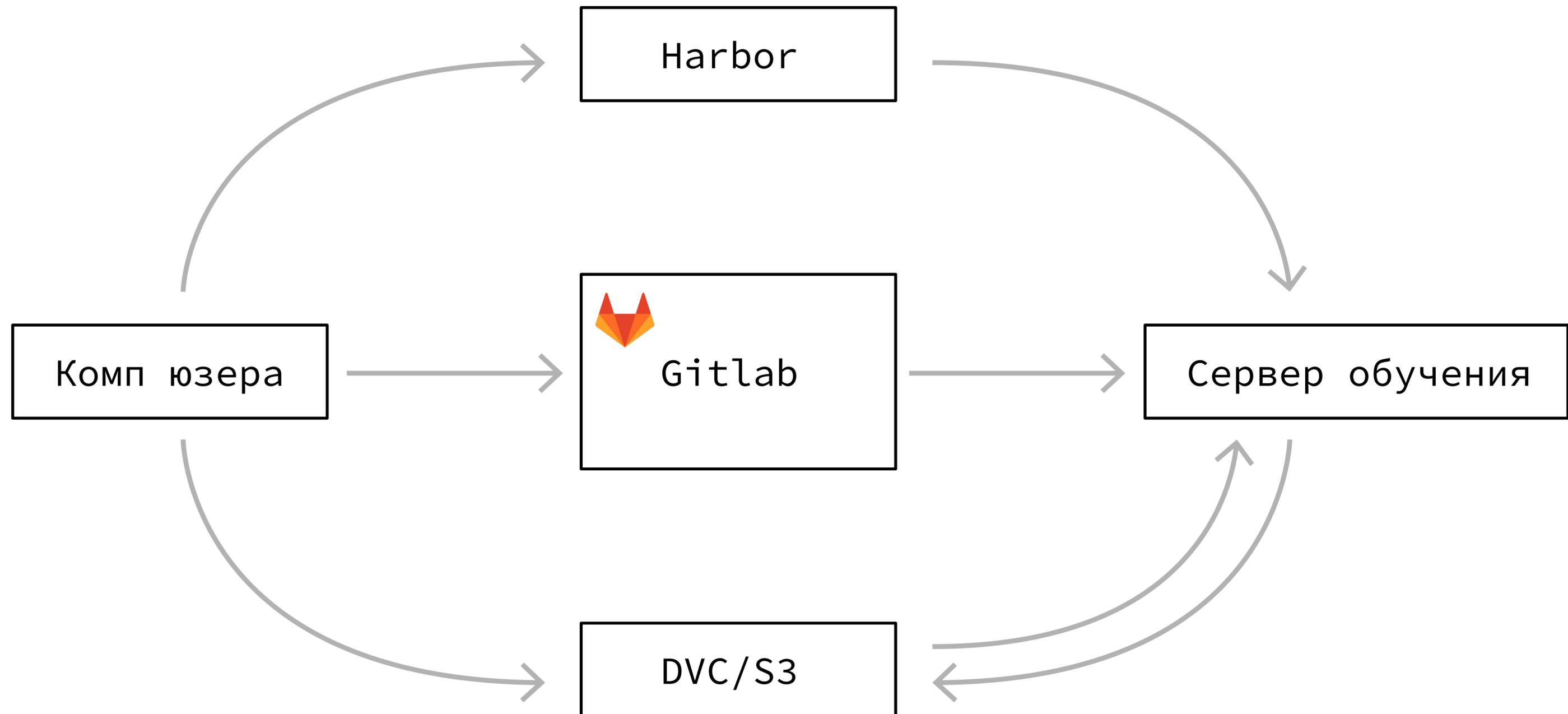
No changes logged

DESCRIPTION:

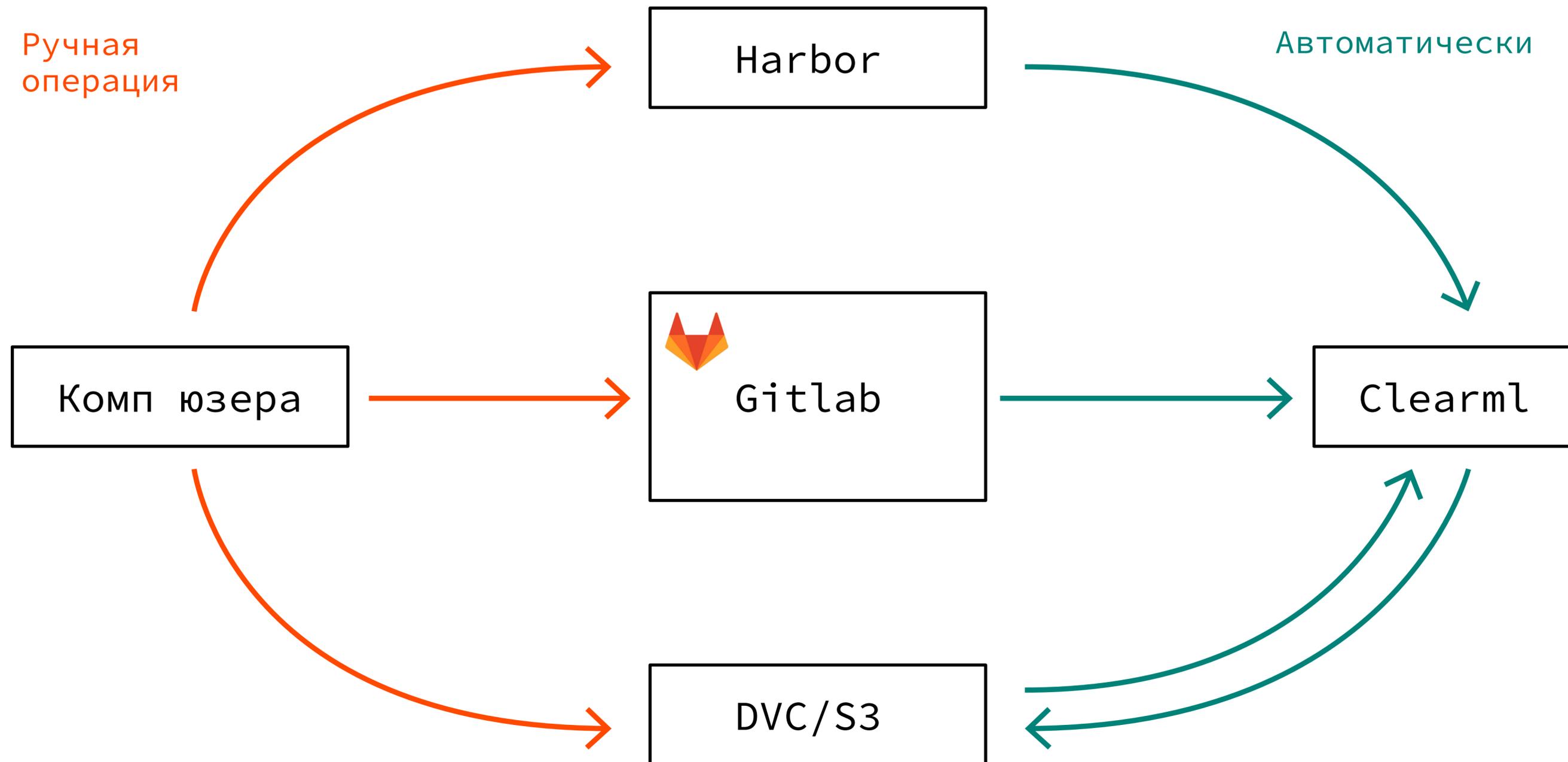
Auto-generated at 2023-09-26 15:04:40 UTC by user@arsenii.shirokov
Using model id: 97c9ff2459b6495bb672df9da077da26

ARCHIVED:	No
CHANGED AT:	Sep 27 2023 20:16
LAST ITERATION:	59910
STATUS MESSAGE:	Changed by clearml-agent LARGE-GPU-1-4:gpu3
STATUS REASON:	worker execution done
CREATED AT:	Sep 26 2023 18:04
STARTED AT:	Sep 26 2023 18:04
LAST UPDATED AT:	Oct 28 2023 1:08
COMPLETED AT:	Sep 27 2023 20:16
RUN TIME:	1:02d
QUEUE:	LARGE-GPU-1-4
WORKER:	LARGE-GPU-1-4:gpu3
CREATED BY:	clearml_user
PARENT TASK:	N/A
PROJECT:	evonets
ID:	bebe9cdc26444c909f7ef8985a89ea47
OS	Linux-5.15.0-67-generic-x86_64-with-glibc2.27
cpu_cores	64
gpu_count	1
gpu_driver_cuda_version	12.0
gpu_driver_version	525.78.01
gpu_memory	24GB

Процесс обучения до ClearML



Процесс обучения с ClearML



Итоги

EVOCARGO

- Процесс обучения нейронных сетей обрел структуру

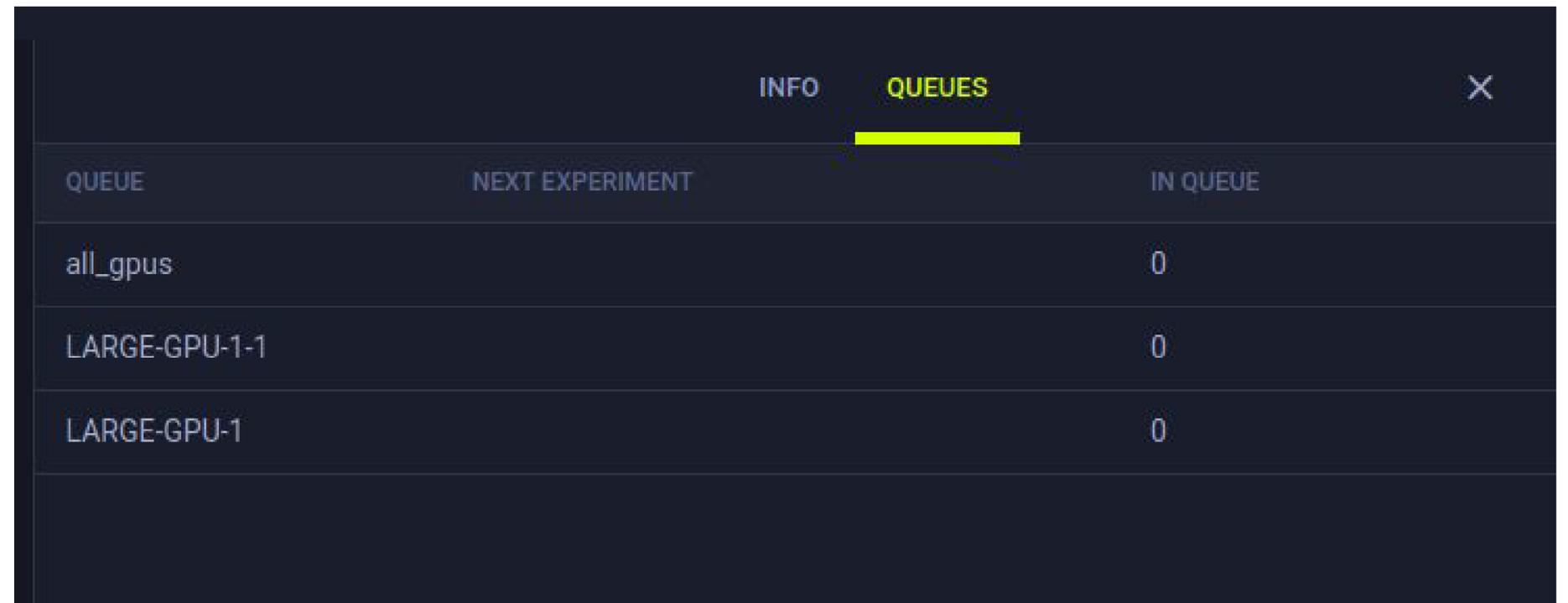
Итоги

- Процесс обучения нейронных сетей обрел структуру
- Мы можем поднять любой эксперимент из истории

<input type="checkbox"/>	 soilingnet-PC-2577-regeression-exp-v3.3.1-add-test-step	✓ Completed
	IMPORTANT soiling_L	
	Updated 4 days ago ● Created by clearml_user	39 Iterations
<input type="checkbox"/>	 tutorial_train_dlysukhin	✓ Completed
	Updated 5 days ago ● Created by clearml_user	691 Iterations
<input type="checkbox"/>	 soilingnet-PC-2577-regeression-exp-v3.2.9-conv2-maxpool2-modified	✓ Completed
	Updated 5 days ago ● Created by clearml_user	39 Iterations
<input type="checkbox"/>	 soilingnet-PC-2577-regeression-exp-v3.2.5-conv2-fc2	✓ Completed
	Updated 5 days ago ● Created by clearml_user	39 Iterations
<input type="checkbox"/>	 soilingnet-PC-2577-regeression-exp-v3.2.8-maxpool2	✓ Completed
	Updated 5 days ago ● Created by clearml_user	39 Iterations
<input type="checkbox"/>	 soilingnet-PC-2577-regeression-exp-v3.2.7-avgpool-same	✓ Completed
	Updated 6 days ago ● Created by clearml_user	39 Iterations

Итоги

- Процесс обучения нейронных сетей обрел структуру
- Мы можем поднять любой эксперимент из истории
- И отслеживать свободные ресурсы



The screenshot shows a dark-themed interface with a 'QUEUES' tab selected. Below the tab is a table with three columns: 'QUEUE', 'NEXT EXPERIMENT', and 'IN QUEUE'. The table contains three rows of data.

QUEUE	NEXT EXPERIMENT	IN QUEUE
all_gpus		0
LARGE-GPU-1-1		0
LARGE-GPU-1		0

Итоги

- Процесс обучения нейронных сетей обрел структуру
- Мы можем поднять любой эксперимент из истории
- И отслеживать свободные ресурсы
- Мы стандартизировали процесс обучения

```
from clearml_client import init_clearml
from dvc.api import DVCFileSystem

# инициализация задачи для Clearml, а не на машине где был запущен код.
# Данная команда обязательно должна быть вызвана до любой
# другой команды для взаимодействия с clearml. Иначе команды clearml, которые
# будут вызваны ранее, вернут None или выкинут ошибку. task_name - название
# задачи в clearml, принято называть именем нейронной сети которая запускается
task = init_clearml(task_name)

# Подключение к задаче clearml и выставление параметров обучения в описании
# задачи на вкладке `Configuration`. dict_params - как раз параметры, которые
# будут отображены внутри вкладки.
task.connect(dict_params)

# Создание clearml логера, который умеет прокидывать выхлоп во вкладку `Console`,
# а также при помощи его функций можно создавать графики на странице `Scalars`.
# Подробнее в документации:
# https://clear.ml/docs/latest/docs/references/sdk/logger/#class-logger.
logger = Logger.current_logger()
...

# Выкачка данных из DVC
self._dvc = DvcApi()

dataset_dvc_path = self._dvc.dvc_path / MNIST_DATASET_PATH
dataset_path = self.dataset_dvc_path.parent

self._dvc.dvc_pull(self.dataset_dvc_path)

...

# Добавление информации в график
logger.report_scalar("average loss", "series", test_loss, epoch)

# Сохранение артефактов (например, весов модели) на ClearML. После создания будут
# доступны во вкладке Artifacts для скачивания.
task.upload_artifact(file_name, file_path)
```

Итоги

EVOCARGO

- Процесс обучения нейронных сетей обрел структуру
- Мы можем поднять любой эксперимент из истории
- И отслеживать свободные ресурсы
- Мы стандартизировали процесс обучения
- Поддерживать окружения в актуальном состоянии стало удобнее

Спасибо за внимание!

EVOCARGO

ClearML Comparison Chart

CATEGORY	TensorBoard	ClearML	MLFlow	W&B	Neptune.ai	DVC	Polyaxon	Comet
Experiment Management	Better	Best	Best	Best	Best	Better	Best	Best
MLOps	Good	Best	Good	Best	Good	Better	Best	Good
Automation	N/A	Better	Better	Better	Good	Best	Paid	Good
Data Management	N/A	Best	N/A	Better	Good	Good	N/A	N/A
General	Best	Better	Best	Good	Good	Best	Best	Good

<https://clear.ml/blog/stacking-up-against-the-competition>