

????????

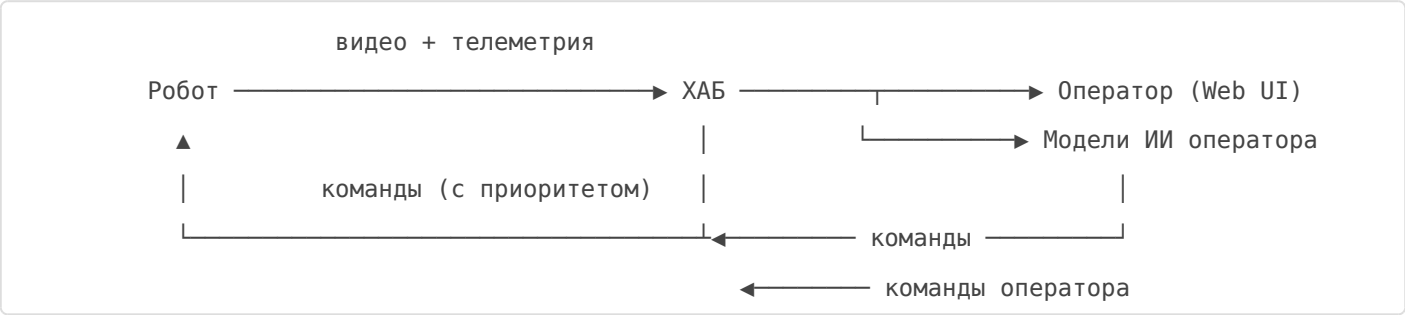
Как ИИ управляет роботами и режимы состязаний

- [Управление роботами с помощью ИИ: обзор](#)
- [Режимы управления и состязаний](#)

Вопросы и ответы

Платформа ARTEX позволяет управлять роботом человеку, модели ИИ или обоим одновременно. Модели ИИ работают в распределённой вычислительной среде платформы (или на борту робота), получают от ХАБа видео и телеметрию и возвращают команды управления — в том же формате, что и оператор.

Схема взаимодействия



- **Основной источник команд — модель ИИ.** Оператор управляет роботом при обучении модели или для «подстраховки» (перехватывает управление в опасной ситуации).
- **Одна команда — один формат.** Для робота нет разницы, кто отправил команду: различается только автор (`author`), по которому ХАБ применяет приоритет.
- **Каждый оператор может подключить свои модели** (Модели ИИ оператора 1 ... N) — они запускаются в контейнерах вычислительной среды.

Таблица ролей

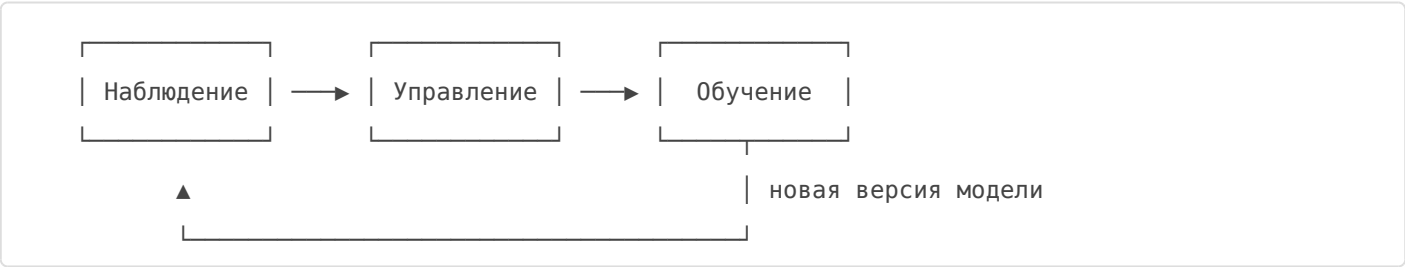
Уровень	Кто управляет	Роль ИИ	Роль оператора
Ручное управление	Оператор	Не участвует или только анализирует (детекция объектов поверх видео)	Полное управление: джойстик, клавиатура, VR-контроллер

Уровень	Кто управляет	Роль ИИ	Роль оператора
Assist (гибридный режим)	Оператор + ИИ	«Второй пилот»: подсказки, стабилизация, удержание курса, автоматическая остановка перед препятствием	Задаёт цель и основное направление
Полная автономия	ИИ	Принимает все решения по видео и телеметрии	Наблюдает, может в любой момент вмешаться (приоритет оператора)

?????? ?????????????? ??????????

Платформа поддерживает полный цикл: от наблюдения и ручного управления до сбора датасета и обучения следующей модели.

Этап	Технологии	Что происходит
Наблюдение	WebRTC + WebSocket	FPV-видео; телеметрия в JSON (GPS, скорость, батарея, IMU); дашборд реального времени
Управление	WebSocket-команды	Ручной режим → Assist → полная автономия ИИ; джойстик / клавиатура / VR-контроллер
Обучение	Запись в S3	Каждое действие оператора записывается в датасет; Imitation Learning для обучения следующей модели



????? ?????????? ?????????????????????

Модель	Задача	Входы	Выходы
YOLO	Детекция объектов на видео в реальном времени	Кадры	Список объектов: класс, уверенность, рамка

Модель	Задача	Входы	Выходы
Assist AI	Помощь оператору (co-pilot)	Кадры, телеметрия, команды оператора	Скорректированные команды, подсказки
RL Agent	Управление на основе обучения с подкреплением	Признаки из видео, телеметрия	Команды управления
Custom	Любые пользовательские модели	Любые данные ХАБа	Команды, признаки, события

???
 ?????????????
 ???????

Вариант	Когда выбирать	Пример
В вычислительной среде платформы (Docker, K8s, GPU)	Тяжёлые модели, соревнования ИИ vs ИИ, централизованный контроль	YOLO на сервере, RL-агент
В конвейере модулей сервера	Типовая обработка, собираемая без программирования	YoloModule в визуальном редакторе
На борту робота	Нужна минимальная задержка или работа при плохой связи	Лёгкая детекция на Jetson / Raspberry Pi; результаты передаются как признаки в телеметрии

?
 ?????
 ???????

1. Познакомьтесь с [веб-интерфейсом управления роботами](#).
2. Выберите [режим управления или состязаний](#).
3. Разработайте модель по руководству [Разработка модели ИИ для платформы](#).
4. Соберите конвейер обработки в визуальном редакторе — [Конвейеры модулей](#).
5. Запишите данные и обучите следующую версию — [Сбор данных и обучение](#).

?????? ?????????? ?

???????????

Битва интеллектуальных роботов — соревновательный формат платформы ARTEX. Роботы на арене (физической или виртуальной) управляются людьми, моделями ИИ или их комбинацией. Побеждает не тот, у кого мощнее железо, а тот, у кого умнее алгоритм.

??????

Режим	Кто управляет	Суть	Технологии
Человек vs Человек	Операторы вручную	Классический формат: операторы управляют роботами с джойстика или VR-контроллера по FPV-видео	WebSocket + FPV
Человек vs ИИ	Оператор против автономной модели	Тест навыков человека против алгоритма; ИИ может помогать оператору (Assist)	YOLO + Assist AI
ИИ vs ИИ	Только модели	Полностью автономные модели: алгоритм против алгоритма. Побеждает код, а не оператор	RL + Socket API
Соревнования и турниры	Команды	ELO-рейтинг команд; школьные, кадетские, вузовские лиги; публичный лидерборд	ELO + Leaderboard

???????????????? ?????????????????

?????????????

- Регистрация команд** — администратор создаёт учётные записи операторов и выдаёт им видимость своих роботов.
- Подключение роботов** — каждый робот подключается через SDK `artex-robot` и проверяется: видео, телеметрия, команды, кнопка «Стоп».

3. **Загрузка моделей** (для режимов с ИИ) — команды загружают модели в вычислительную среду; модели проходят тестовый заезд.
4. **Правила** — задаются: длительность раунда, критерии победы, ограничения скорости, допустимые команды.

????????? ??????

1. Судья запускает раунд; включается запись всех потоков (видео, телеметрия, команды).
2. Операторы и/или модели управляют роботами.
3. Судья может остановить любого робота (Стоп / DISARM).
4. Результат раунда фиксируется и влияет на рейтинг.

????? ??????

- Запись раунда доступна для разбора: видео синхронизировано с телеметрией и командами.
- Данные раундов «Человек vs Человек» — датасет для обучения моделей следующего поколения.

??????? ELO

После каждого поединка рейтинг участников A и B пересчитывается:

$$E_A = 1 / (1 + 10^{((R_B - R_A) / 400)}) \quad \# \text{ ожидаемый результат A}$$
$$R_A' = R_A + K \cdot (S_A - E_A) \quad \# S_A = 1 \text{ победа, } 0.5 \text{ ничья, } 0 \text{ поражение}$$

Рекомендуемые параметры: начальный рейтинг 1000, $K = 32$ для новых участников (менее 10 поединков) и $K = 16$ для остальных.

```
def elo_update(r_a, r_b, score_a, k=32):
    expected_a = 1 / (1 + 10 ** ((r_b - r_a) / 400))
    new_a = r_a + k * (score_a - expected_a)
    new_b = r_b + k * ((1 - score_a) - (1 - expected_a))
    return round(new_a), round(new_b)

elo_update(1000, 1100, 1) # → (1020, 1080): победа над более сильным даёт больше очков
```

Рейтинги ведутся отдельно для людей, моделей и смешанных команд.

?????

Лига	Участники	Особенности
Школьная	Школьники, кружки робототехники	Режимы «Человек vs Человек» и «Человек vs ИИ», готовые модели, геймификация
Кадетская	Кадетские корпуса, ведомственные центры	Управление БПЛА, тактические сценарии
Вузовская	Студенты, аспиранты, лаборатории	Режим «ИИ vs ИИ», собственные модели, публикация результатов