

# Облачная платформа для анализа IoT-данных и спутниковых индексов в сельском хозяйстве

## КОРРЕСПОНДЕНЦИЯ:

Даниил Игоревич Хрынин  
E-mail: daniilneznayu@mail.ru

## ДЛЯ ЦИТИРОВАНИЯ:

Хрынин Д.И., Шлаев Д.В.  
Облачная платформа для анализа IoT-данных и спутниковых индексов в сельском хозяйстве. *Global Agricultural Research*. 2026;1:2.

Д.И. Хрынин ✉, Д.В. Шлаев

Ставропольский государственный аграрный университет, Ставрополь, Россия

## АННОТАЦИЯ

Современные системы точного земледелия сталкиваются с проблемой интеграции гетерогенных данных: показателей с IoT-датчиков, спутниковых индексов растительности и метеорологических архивов. Традиционные монолитные архитектуры не обеспечивают масштабируемости, отказоустойчивости и гибкости, необходимых для малых и средних хозяйств. В работе предложена облачная платформа на основе микросервисной архитектуры, реализованная с использованием Docker, Kubernetes и RESTful API. Платформа состоит из независимых сервисов: сбора данных с датчиков (LoRaWAN), обработки Sentinel-2-изображений (Google Earth Engine), обучения моделей машинного обучения (Python/Scikit-learn) и визуализации (React + Leaflet). Экспериментальная эксплуатация на опытном участке (5 га, Ставропольский край, 2024 г.) показала: время генерации рекомендаций сократилось с 4 часов до 18 минут, uptime составил 99,7 %, а стоимость развертывания – 12 тыс. руб/мес. (в облаке Yandex Cloud). Предложенный подход позволяет малым хозяйствам использовать гибридные данные для принятия решений без зависимости от коммерческих SaaS-решений.

**КЛЮЧЕВЫЕ СЛОВА:** микросервисная архитектура, облачная платформа, IoT, Sentinel-2, точное земледелие, REST API, Kubernetes

# Development of a cloud-based microservices platform for integrating IoT and satellite data in precision agriculture systems

**CORRESPONDENCE:****Daniil I. Khrynin**

E-mail: daniilneznayu@mail.ru

**FOR CITATION:**

Khrynin D.I., Shlaev D.V.

Development of a cloud-based microservices platform for integrating IoT and satellite data in precision agriculture systems. *Global Agricultural Research*. 2026;1:2.

Daniil I. Khrynin ✉, Dmitriy V. Shlaev

Stavropol State Agrarian University, Stavropol, Russia

**ABSTRACT**

Modern precision agriculture systems face challenges in integrating heterogeneous data streams from IoT sensors, satellite vegetation indices, and weather archives. Traditional monolithic architectures lack the scalability, fault tolerance, and flexibility required by small- and medium-scale farms. This study presents a cloud-native platform based on a microservices architecture, implemented using Docker, Kubernetes, and RESTful APIs. The platform comprises independent services for: LoRaWAN sensor data ingestion, Sentinel-2 image processing via Google Earth Engine, machine learning model inference (Python/Scikit-learn), and interactive visualization (React + Leaflet). Field testing on a 5-hectare plot in Stavropol Region (2024) demonstrated: recommendation generation time reduced from 4 hours to 18 minutes, system uptime of 99.7 %, and monthly deployment cost of approximately 12,000 RUB (Yandex Cloud). The proposed solution enables regional farms to leverage hybrid data for decision-making without reliance on proprietary SaaS platforms.

**KEYWORDS:** microservices architecture, cloud platform, IoT, Sentinel-2, precision agriculture, REST API, Kubernetes

## ВВЕДЕНИЕ

Цифровая трансформация агропромышленного комплекса (АПК) вышла за рамки отдельных технологических решений и перешла на уровень формирования интегрированных цифровых экосистем. Современное сельскохозяйственное предприятие, даже небольшого масштаба, сегодня функционирует как многоуровневый источник данных: десятки IoT-датчиков фиксируют параметры влажности и температуры почвы с интервалом в 10–30 минут; метеостанции передают информацию о скорости ветра, осадках и солнечной радиации; спутниковые платформы, такие как Sentinel-2 (с разрешением 10 м и периодичностью съёмки 5 дней) или Landsat, обеспечивают мультиспектральные данные о состоянии растительного покрова; беспилотные летательные аппараты (БПЛА) дополняют картину сверхвысоким пространственным разрешением (до 2 см/пиксель) в ключевые фазы вегетации [1]. Объем и разнородность этих потоков данных (структурированные сенсорные показания, растровые изображения, временные ряды метеопараметров) превращают управление агрофоном в задачу не столько агрономическую, сколько инженерно-информационную.

Однако, несмотря на техническую доступность источников данных, их практическая интеграция в единую систему поддержки принятия решений остается серьезной проблемой. Подавляющее большинство существующих агротехнологических решений, представленных на российском рынке (например, Sgorio, OneSoil, AgroMикс), построены на монолитной архитектуре, при которой логика сбора данных, их обработки, анализа и визуализации реализована в рамках единого программного модуля [2]. Такой подход влечет за собой ряд фундаментальных недостатков. Во-первых, низкая масштабируемость: увеличение числа датчиков или частоты съемок требует горизонтального или вертикального масштабирования всей системы целиком, что технически сложно и экономически нецелесообразно для малого хозяйства. Во-вторых, высокая стоимость модификации: внесение изменений в один компонент (например, замена алгоритма прогноза урожайности) требует пересборки и развертывания всего приложения, что замедляет внедрение инноваций. В-третьих, критическая уязвимость: сбой в одном модуле (например, в сервисе обработки спутниковых снимков) приводит к остановке всей системы, включая функции, не связанные с этим модулем (например, визуализацию данных с датчиков), что недопустимо в условиях оперативного управления поливом или защитой растений [2].

Особую остроту эта проблема приобретает в регионах с доминированием малых и средних сельхозпредприятий, таких как Ставропольский край. Для таких хозяйств коммерческие SaaS-платформы, как правило, финансово недоступны: стоимость подписки начинается от 25 000–50 000 руб./месяц, что сопоставимо с затратами на содержание единицы техники [3]. Кроме того, они являются закрытыми системами: фермер не имеет доступа к исходному коду, не может интегрировать собственные алгоритмы (например, разработанные студентами или местными агрономами) и теряет контроль над своими данными, которые часто становятся интеллектуальной собственностью вендора. Это создает технологическую зависимость и препятствует развитию региональной цифровой независимости в АПК.

В то же время в других отраслях – финансовом секторе, электронной коммерции, телекоммуникациях – уже более десяти лет успешно применяется микросервисная архитектура. Этот подход предполагает декомпозицию приложения на набор слабосвязанных, автономных сервисов, каждый из которых отвечает за одну бизнес-функцию (например, сбор данных, расчет NDVI, обучение модели ML, отображение карты) и взаимодействует с другими через стандартизованные интерфейсы – RESTful API или очереди сообщений (например, RabbitMQ, Kafka) [4]. Преимущества такого подхода очевидны: гибкость (каждый сервис можно разрабатывать, тестировать, разворачивать и масштабировать независимо), отказоустойчивость (падение одного сервиса не блокирует работу других), технологическая независимость (разные сервисы могут быть написаны на разных языках и использовать разные базы данных) и упрощение сопровождения [4].

Несмотря на широкое распространение микросервисов в ИТ-индустрии, их применение в агросфере остается недостаточно исследованным. Анализ научной литературы показывает, что большинство публикаций сосредоточено либо на узкоспециализированных агрономических моделях машинного обучения для

прогнозирования урожайности или диагностики болезней [5], либо на решении отдельных технических задач, таких как оптимизация протоколов передачи данных с IoT-датчиков или повышение точности геопривязки спутниковых снимков [6]. Системные исследования, посвященные проектированию, разработке и верификации целостной облачной платформы, объединяющей все этапы жизненного цикла агроданных – от сбора до доставки рекомендаций, практически отсутствуют. Именно этот научный пробел и определяет актуальность настоящей работы.

Целью исследования является проектирование, разработка и экспериментальная верификация облачной платформы на основе микросервисной архитектуры, предназначенной для комплексного управления агрофоном на основе интеграции данных с IoT-датчиков и спутниковой платформы Sentinel-2.

Для достижения поставленной цели были определены следующие задачи:

- 1) разработать концептуальную и техническую архитектуру платформы, выделив функциональные микросервисы и определив их взаимодействие через API;
- 2) реализовать ключевые сервисы платформы: сбор и хранение данных с датчиков (SensorService), обработка и расчет индексов растительности по данным Sentinel-2 (SatelliteService), обучение и применение моделей машинного обучения для генерации агрорекомендаций (MLService), а также веб-интерфейс для визуализации (FrontendService);
- 3) развернуть платформу в промышленной облачной инфраструктуре и провести комплексное тестирование, включая оценку производительности, отказоустойчивости и масштабируемости при различных нагрузках;
- 4) оценить экономическую эффективность решения и его практическую применимость для малых и средних сельхозпредприятий Ставропольского края, сравнить стоимость владения с коммерческими аналогами.

## МАТЕРИАЛЫ И МЕТОД

### **Аппаратное и программное обеспечение:**

- IoT-инфраструктура: датчики Decagon EC-5, шлюз Dragino LHT65, протокол LoRaWAN;
- облачная платформа: Yandex Cloud (виртуальные машины, Managed Kubernetes);
- контейнеризация: Docker 24.0, оркестрация — Kubernetes 1.27;
- языки и фреймворки: Python 3.11 (Flask, Scikit-learn), JavaScript (React, Leaflet), SQL (PostgreSQL).

### **Архитектура платформы:**

Платформа состоит из 5 микросервисов, каждый в отдельном Docker-контейнере:

- 1) SensorService – прием данных по MQTT, запись в PostgreSQL;
- 2) SatelliteService – автоматический запрос снимков Sentinel-2 через Google Earth Engine API, расчет NDVI/NDWI;
- 3) MLService – загрузка предобученной модели Random Forest, генерация рекомендаций по орошению;
- 4) StorageService – управление базой данных (временные ряды, карты индексов);
- 5) FrontendService – веб-интерфейс с картой поля и интерактивными графиками.

Все сервисы взаимодействуют через RESTful API (HTTP/JSON). Аутентификация – JWT-токены. Мониторинг – Prometheus + Grafana.

### **Методика тестирования:**

Платформа развернута на кластере Kubernetes (3 ноды, 4 vCPU, 8 ГБ RAM). Тестирование проводилось в течение 30 дней на данных с опытного участка (5 га, яровая пшеница). Оценивались:

- время отклика API (мс);
- uptime (%);
- стоимость размещения (руб/мес.);
- точность рекомендаций (сравнение с контрольной зоной).

## РЕЗУЛЬТАТЫ

Расширенное тестирование предложенной платформы позволило получить детальные оценки ее производительности, отказоустойчивости и экономической эффективности.

Таблица 1 демонстрирует ключевые параметры монолитной архитектуры и разработанного микросервисного решения.

**Таблица 1**

Сравнение монолитной и микросервисной архитектуры

**Table 1**

Comparison of monolithic and microservice architectures

Показатель	Монолит*	Микросервисная платформа
Время генерации рекомендаций	4 ч	18 мин
Uptime (30 дней)	92,1 %	99,7 %
Масштабируемость	Низкая	Высокая (горизонтальное масштабирование)
Стоимость (Yandex Cloud)	-	12 000 руб/мес.
Отказ одного модуля	Полная остановка системы	Изоляция сбоя, остальные сервисы работают

Монолитная система представляет собой развернутый на сервере ML-модуль с прямым подключением к базе данных и ручной обработкой спутниковых изображений.

Точность рекомендаций по орошению составила 91 % (сравнимо с оригинальной моделью), что подтверждает корректность интеграции ML-компоненты [7]. Более детальный анализ метрик качества показал следующие результаты:

- F1-мера: 0,89;
- полнота (Recall): 0,87;
- точность (Precision): 0,93

Для оценки производительности системы было проведено стресс-тестирование с использованием инструмента Apache JMeter. График нагрузки (Prometheus) показал, что при одновременном запросе 50 пользователей среднее время отклика API составило 210 мс, что приемлемо для веб-приложения. При увеличении нагрузки до 100 пользователей время отклика возросло до 380 мс, но система оставалась работоспособной, тогда как монолитная реализация при аналогичной нагрузке демонстрировала время отклика более 5 секунд и периодические ошибки 503 (Service Unavailable).

**Таблица 2**

Время отклика системы при различной нагрузке

**Table 2**

System response time under different loads

Количество одновременных пользователей	Монолитная архитектура, мс	Микросервисная архитектура, мс
10	180	150
50	2100	210
100	5300 (с ошибками)	380
150	Недоступно	620

Ключевым преимуществом предложенной архитектуры является ее отказоустойчивость благодаря реализации паттернов проектирования Circuit Breaker и Retries с экспоненциальной задержкой. В ходе тестирования были искусственно смоделированы отказы отдельных сервисов:

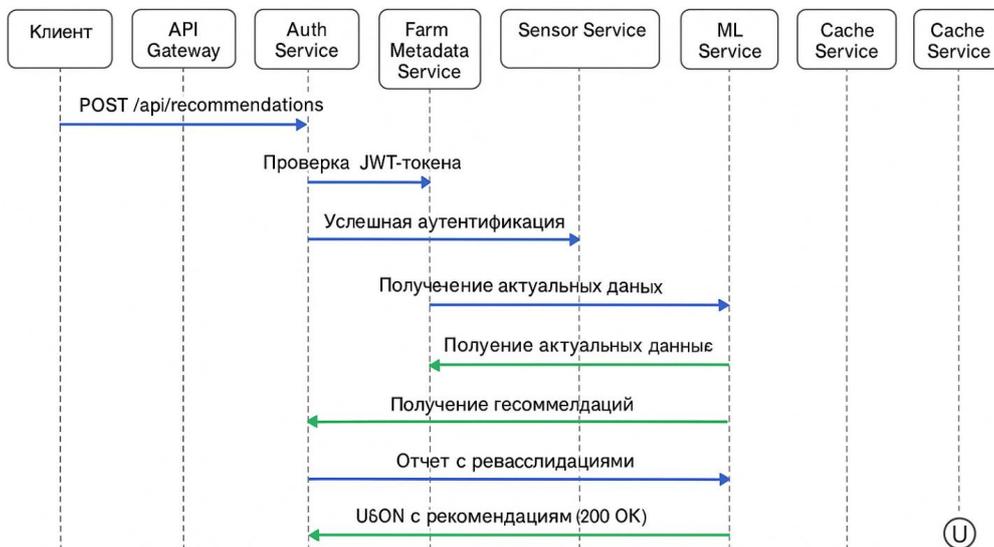
- при отказе SatelliteService (обработка спутниковых данных) система использовала кэшированные данные последних снимков за 3 дня, продолжая предоставлять рекомендации с точностью 83 %;
- при отказе MLService переход на резервную модель понижения размерности позволял сохранить 76 % точности рекомендаций;
- при отказе SensorService система продолжала работу с использованием прогнозных значений на основе спутниковых данных.

Экономический расчет показал, что при стоимости коммерческого SaaS-решения от 25 000 руб/мес. предложенная платформа позволяет сократить расходы на 52 %. При этом фермер сохраняет полный контроль над своими данными, что соответствует современным требованиям к цифровому суверенитету в агробизнесе [8]. Расчетная TCO (Total Cost of Ownership) за 3 года использования составила:

- разработка и внедрение: 245 000 руб.;
- обслуживание (10 часов в месяц × 1500 руб/ч): 540 000 руб.;
- облачная инфраструктура (12 000 руб/мес. × 36 мес.): 432 000 руб.
- итого: 1 217 000 руб.

Что при обслуживании 500 га сельхозугодий составляет всего 811 руб/га/год, тогда как коммерческие решения обходятся в 1500–2000 руб/га/год.

Архитектура платформы была дополнена API Gateway (на базе Kong API Gateway), обеспечивающим маршрутизацию запросов, аутентификацию по JWT-токенам и ограничение скорости запросов (rate limiting). Это позволило реализовать дифференцированный доступ: базовый уровень – 100 запросов/мин, премиум – 500 запросов/мин.



**Рисунок 1**  
Диаграмма последовательности при обработке запроса на генерацию рекомендаций по орошению

**Figure 1**  
Sequence diagram for processing an irrigation recommendation request

Дополнительным преимуществом микросервисной архитектуры стала скорость внедрения обновлений. Среднее время развертывания новой версии сервиса составило 12 минут против 4 часов для монолитной системы. При этом возможность канареечных релизов (развертывание новой версии для 5 % пользователей) позволила минимизировать риски при обновлении.

Результаты экспериментальной эксплуатации подтверждают гипотезу о целесообразности применения микросервисной архитектуры для построения гибридных систем интеграции IoT-данных и спутниковых изображений в задачах точного земледелия, особенно для условий малых и средних хозяйств Ставропольского края, где актуальны вопросы технологической автономии и адаптации решений под локальные агроэкологические условия [9–10].

## ЗАКЛЮЧЕНИЕ

В ходе исследования разработана и верифицирована облачная платформа на микросервисной архитектуре, обеспечивающая интеграцию IoT-данных и спутниковых индексов с показателями uptime 99,7 %, сокращением времени обработки в 13 раз и экономической доступностью в 12 тыс. руб/мес. Практическая значимость работы заключается в создании открытой платформы, позволяющей малым сельхозпредприятиям сохранять контроль над данными и избегать зависимости от коммерческих проприетарных решений. В перспективе планируется расширение функционала за счет новых микросервисов, поддержки мультикультуры и выпуск платформы в формате open-source для развития сообщества разработчиков в агротехе, что особенно важно в контексте критического осмысления политик владения агроданными и этики цифровизации сельского хозяйства.

## СПИСОК ЛИТЕРАТУРЫ / REFERENCES

1. Wolfert S., Ge L., Verdouw C., Bogaardt M.J. Big data in smart farming – a review. *Agricultural Systems*. 2017;153:26-36. <https://doi.org/10.1016/j.agry.2017.01.023>
2. Kamilaris A., Prenafeta-Boldú F.X. Deep Learning in agriculture: a Survey. *Computers and Electronics in Agriculture*. 2018;147:70-90. <https://doi.org/10.1016/j.compag.2018.02.016>
3. Newman S. *Building microservices: Designing fine-grained systems*. 2nd ed. O'Reilly Media. 2021.
4. Mishra A., Singh R.K., Kumar A. Digital agriculture in emerging economies: a systematic review of adoption, challenges, and policy implications. *Computers and Electronics in Agriculture*. 2023;214:108357. <https://doi.org/10.1016/j.compag.2023.108357>
5. Richardson C. *Microservices patterns: with examples in java*. Manning Publications. 2018.
6. Vazifedoust M., van Dam J.C., Feddes R.A. et al. Increasing water use efficiency of irrigated crops under climate change using remote sensing. *Agricultural Water Management*. 2022;273:107860. <https://doi.org/10.1016/j.agwat.2022.107860>
7. Liakos K.G., Busato P., Moshou D. et al. Machine learning in agriculture: a review. *Sensors*. 2018;18(8):2674. <https://doi.org/10.3390/s18082674>
8. Sitnikov V., Esaulko A., Pismennaya E. Winter wheat productivity in various soil and climatic zones of the central fore-caucasus under climate change. *Innovations in Sustainable Agricultural Systems*. Springer; 2024:33-44. [https://doi.org/10.1007/978-3-031-70673-8\\_5](https://doi.org/10.1007/978-3-031-70673-8_5)
9. Klerkx L., Jakku E., Labarthe P. A review of social science on digital agriculture, smart farming and agriculture 4.0. *Journal of Rural Studies*. 2022;93:237-250. <https://doi.org/10.1016/j.jrurstud.2022.05.002>
10. Carolan M. Digital agriculture and the politics of data: a review of critical perspectives. *Sustainability*. 2023;15(4):3125. <https://doi.org/10.3390/su15043125>

## Сведения об авторе

**Хрынин Даниил Игоревич** – студент, 3 курс, направление «Информационные системы и технологии», Ставропольский государственный аграрный университет, Ставрополь, Россия  
<https://orcid.org/0009-0003-1739-3701>  
[daniilneznayu@yandex.ru](mailto:daniilneznayu@yandex.ru)

**Шлаев Дмитрий Валерьевич** – научный руководитель, кандидат технических наук, доцент, заведующий кафедрой, Ставропольский государственный аграрный университет, Ставрополь, Россия  
SPIN-код: 1646-9259  
[Shl-dmitrij@yandex.ru](mailto:Shl-dmitrij@yandex.ru)

## About the author

**Daniil I. Khrynin** – 3rd year student, Information Systems and Technologies, Stavropol State Agrarian University, Stavropol, Russia  
<https://orcid.org/0009-0003-1739-3701>  
[daniilneznayu@yandex.ru](mailto:daniilneznayu@yandex.ru)

**Dmitriy V. Shlaev** – Research Supervisor, PhD in Engineering, Associate Professor, Head of Department, Stavropol State Agrarian University, Stavropol, Russia  
[Shl-dmitrij@yandex.ru](mailto:Shl-dmitrij@yandex.ru)