

УДК 004.75

СОЗДАНИЕ ШАБЛОНА DEVOPS ПРИМЕНИТЕЛЬНО К ИНДУСТРИАЛЬНОЙ CONTINUOUS INTELLIGENCE ПЛАТФОРМЕ

Малых Максим Александрович, студент, Базовая кафедра «Аналитика больших данных и методы видеоанализа», Уральский федеральный университет имени первого Президента России Б.Н. Ельцина

Предеин Никита Сергеевич, студент, Базовая кафедра «Аналитика больших данных и методы видеоанализа», Уральский федеральный университет имени первого Президента России Б.Н. Ельцина

Бурдин Дмитрий Евгеньевич, студент, Базовая кафедра «Аналитика больших данных и методы видеоанализа», Уральский федеральный университет имени первого Президента России Б.Н. Ельцина

Медведев Максим Александрович, к.т.н., доцент Базовой кафедры «Аналитика больших данных и методы видеоанализа», Уральский федеральный университет имени первого Президента России Б.Н. Ельцина

Аннотация

В статье рассматривается разработка шаблона DevOps для платформы Object Relations Technique/Continuous Intelligence (ORT/CI), предназначенной для поддержки концепции Continuous Intelligence в промышленных приложениях. Платформа ORT/CI включает инструменты Apache NiFi, JupyterLab и Totum, что обеспечивает интеграцию больших данных и аналитики непосредственно в операционные процессы. Цель работы – создать универсальный шаблон версионирования, развертывания и сопровождения пользовательских приложений на этой платформе, а также настроить сбор и визуализацию метрик и логов с помощью инструментов Prometheus, Grafana и Loki. В рамках работы был реализован конвейер CI/CD на основе GitLab CI с контейнеризацией (Docker, Kubernetes), реализовано автоматическое развертывание функций, созданных в JupyterLab, с помощью без серверной платформы Nuclio. Разработанный шаблон упрощает вход в среду ORT/CI, обеспечивая стандартизацию процессов разработки и эксплуатации. Результаты показывают, что интеграция современных DevOps-практик позволяет повысить автоматизацию и прозрачность работы платформы, сохраняя при этом гибкость настройки и масштабируемость.

КЛЮЧЕВЫЕ СЛОВА: DevOps, ORT/CI, Continuous Intelligence, CI/CD, JupyterLab, Nuclio, Apache NiFi, Prometheus, Grafana, Loki, платформы интеграции, мониторинг, автоматизация.

CREATING A DEVOPS TEMPLATE FOR AN INDUSTRIAL PLATFORM OF CONTINUOUS INTELLIGENCE

Malykh Maksim Aleksandrovich, Student, Department of Big Data Analytics and Video Analysis Methods, Institute of Radioelectronics and Information Technology, Ural Federal University named after the first President of Russia B.N. Yeltsin

Predein Nikita Sergeevich, Student, Department of Big Data Analytics and Video Analysis Methods, Institute of Radioelectronics and Information Technology, Ural Federal University named after the first President of Russia B.N. Yeltsin

Buridin Dmitriy Evgen'evich, Student, Department of Big Data Analytics and Video Analysis Methods, Institute of Radioelectronics and Information Technology, Ural Federal University named after the first President of Russia B.N. Yeltsin

Medvedev Maksim Aleksandrovich, Associate Professor, Department of Big Data Analytics and Video Analysis Methods, Institute of Radioelectronics and Information Technology, Ural Federal University named after the first President of Russia B.N. Yeltsin

Abstract

This article describes the development of a DevOps template for the ORT/CI platform, which supports the concept of Continuous Intelligence in industrial applications. The ORT/CI platform integrates tools like Apache NiFi, JupyterLab, and Totum, enabling data integration and analytics within operational processes. The project's goal was to create a general template for version control, deployment, and maintenance of user applications on this platform, and to set up collection and visualization of metrics and logs using Prometheus, Grafana, and Loki. A CI/CD pipeline based on GitLab CI was implemented, leveraging containerization (Docker, Kubernetes), automating deployment of functions developed in JupyterLab via the Nuclio serverless platform. The developed template simplifies onboarding to the ORT/CI environment by standardizing development and operations workflows. Results indicate that integrating modern DevOps practices increases automation and visibility of the platform's operations while retaining configuration flexibility and scalability.

KEYWORDS: DevOps, ORT/CI, Continuous Intelligence, CI/CD, JupyterLab, Nuclio, Apache NiFi, Prometheus, Grafana, Loki, integration platforms, monitoring, automation.

1. Введение

Быстрое развитие информационных технологий стимулирует цифровизацию промышленных процессов и внедрение современных методов сбора и анализа данных. Концепция Continuous Intelligence (CI) – непрерывной аналитики в реальном времени – становится ключевой для предприятий, позволяя адаптировать операции к изменяющимся условиям практически мгновенно [1].

Цифровизация производства является одной из наиболее актуальных и значимых тем, поскольку информационно-коммуникационные технологии трансформируют экономические и производственные процессы. Внедрение Интернета вещей (IoT), больших данных (Big Data) и искусственного интеллекта (AI) способствует оптимизации производственных процессов, снижению издержек, повышению качества продукции и открывает новые возможности для инноваций [2].

Однако, несмотря на очевидные преимущества, предприятия сталкиваются с серьезными проблемами в процессе цифровизации. Одной из ключевых проблем является разрозненность данных. Специфические модели и форматы данных в различных системах определяются бизнес-процессами конкретных участков производства, что делает их индивидуальными для каждой системы и непригодными для обобщенного глобального анализа.

Попытки агрегировать данные в так называемые «озера данных» с использованием технологий Big Data часто не решают проблему разнородности, а лишь переносят ее на общий сервер, формируя коллекцию таблиц и файлов. Это приводит к необходимости создания дополнительных ИТ-команд для подготовки сырых данных к анализу, что снижает оперативность принятия решений. Таким образом, работа направлена на преодоление этой проблемы, предлагая унифицированный подход к управлению данными и процессами. Проектируемая платформа

Object Relations Technique/Continuous Intelligence (ORT/CI) ориентирована на реализацию принципов CI: она объединяет механизмы интеграции данных и аналитики в единое решение, предоставляя возможности искусственного интеллекта и машинного обучения для обработки больших объёмов информации. Для успешной эксплуатации такой платформы важно разработать стандартизированные процессы разработки и развертывания приложений [3].

В данном контексте DevOps-подход играет решающую роль, поскольку способствует слаженной работе команд разработки и эксплуатации, автоматизации сборки, тестирования и развертывания программных компонентов. Целью работы является создание шаблона DevOps, применительно к ORT/CI, включающего средства контроля версий, конвейер CI/CD, контейнеризацию и инструменты мониторинга, что позволит ускорить разработку пользовательских решений и повысить надежность их работы.

2. Платформа ORT/CI и концепция Continuous Intelligence

Платформа ORT/CI представляет собой интегрированное решение для промышленной аналитики и управления данными. Она включает несколько ключевых компонентов: Apache NiFi, предназначенную для построения конвейеров обработки данных; JupyterLab – среду разработки интерактивных вычислений и анализа данных; а также Totum – систему создания информационных панелей и управления бизнес-процессами.

Apache NiFi – это мощная система обработки потоков данных на основе концепции flow-based programming. NiFi позволяет создавать визуальные схемы маршрутизации и преобразования данных с гибкими возможностями мониторинга «происхождения» данных (data provenance).

JupyterLab – это веб среда для написания и запуска кода, предоставляющая гибкие интерфейсы для работы с блокнотами, данными и расширениями. Она служит основным инструментом разработки аналитических модулей и моделей в составе платформы. Totum обеспечивает адаптивные интерфейсы и дашборды, позволяя настраивать решения под нужды конечных пользователей.

Все эти компоненты объединены общей концепцией Continuous Intelligence, при использовании которой данные из различных источников обрабатываются в реальном времени и немедленно используются для принятия управленческих решений. Так, платформа ORT/CI поддерживает непрерывный цикл «сбор-анализ-действие»: входящие бизнес-события и

показатели обрабатываются конвейерами данных, результаты анализа визуализируются в дашбордах, а при необходимости автоматически инициируются управляющие воздействия [4].

3. Методы

3.1 Инструментарий проекта

Разрабатываемый шаблон DevOps интегрируется с основными инструментами платформы ORT/ CI. В таблице ниже приведены ключевые компоненты инфраструктуры и их роль в шаблоне:

Таблица 1. Описание основных компонентов инфраструктуры проекта

Компонент	Описание
Apache NiFi	Используется для построения потоков обработки и интеграции данных между системами. Благодаря визуальному интерфейсу NiFi позволяет легко настраивать конвейеры получения, преобразования и передачи информации. В шаблоне предусмотрены типовые процессоры NiFi (например, GetFile, SplitJson, InvokeHTTP) для работы с входящими данными и API внешних сервисов [5].
JupyterLab	Среда разработки аналитических функций. На её базе создаются Python скрипты и Jupyter-блокноты, реализующие бизнес-логику (например, моделирование или предобработку). Для включения таких функций в CI/CD-процесс используется расширение nuclio-jupyter и утилита конвертации jupyter nbconvert для экспорта блокнота в конфигурацию Nuclio [6].
Nuclio	Nuclio – это платформа для создания и запуска без серверных функций (FaaS). Как указано на сайте проекта, «Nuclio – это без серверная платформа с открытым исходным кодом, используемая для минимизации затрат на разработку и обслуживание, и автоматизации развертывания приложений, основанных на науке о данных». В рамках шаблона DevOps функции, разработанные в JupyterLab, автоматически разворачиваются в Nuclio. Это обеспечивает мгновенное масштабирование и упрощает управление версиями функций [7].
Prometheus & Grafana	Для мониторинга приложения и сбора метрик задействован стек Prometheus+Grafana. Prometheus – это открытая система мониторинга и хранения временных рядов, которая получает метрики из контейнеров и сервисов ORT/CI. Grafana – это популярный инструмент визуализации и аналитики показателей, широко используемый для построения дашбордов. В шаблоне настроены экспортеры метрик Prometheus и готовые дашборды Grafana для отображения состояния функций, сервисов и производительности системы [8,9].
Grafana Loki	Для агрегирования и поиска логов используется лог-система Loki от Grafana Labs. Loki – «горизонтально масштабируемая, высокодоступная система агрегации логов, вдохновленная Prometheus» (или проще: система хранения и поиска логов из приложений). Логи контейнеров и функций отправляются в Loki, что позволяет проводить корреляцию с метриками и анализировать поведение сервисов [10].

3.2 Архитектура платформы ORT/CI

Концепция serverless используется в рамках ORT/CI для упрощения развертывания вычислений. Nuclio обеспечивает высокую производительность без серверных функций: на официальном сайте отмечается, что Nuclio способен выполнять до сотен тысяч вызовов функций в секунду. Такая архитектура позволяет отделить стадию разработки алгоритмов (в JupyterLab) от механизма их исполнения в продуктивной среде [11].

В шаблоне DevOps настройка без серверной среды включает преобразование Jupyter-блокнотов в конфигурационные файлы Nuclio и разметку функций метками для управления состоянием и запросами. Предусмотрена интеграция с системой CI: при каждом обновлении кода функции конвейер автоматически вызывает `jupyter nbconvert --to nuclio`, собирает Docker-образ и деплоит функцию в Nuclio. После успешного развёртывания функция становится доступна по REST API и сразу же попадает под наблюдение Prometheus/Grafana и сбор логов в Loki. Такой подход обеспечивает быстрое преобразование аналитических наработок в производственные микросервисы без ручной операционной работы.

3.3 Система непрерывной интеграции

Шаблон DevOps предусматривает использование систем CI/CD для управления жизненным циклом разработки. В качестве основы используется GitLab CI – популярная платформа автоматизации сборки и развертывания. Непрерывная интеграция (CI) предполагает частое слияние изменений в общий репозиторий и автоматическую проверку каждой новой версии (сборка и тестирование). Такая практика помогает быстро обнаруживать дефекты на ранних этапах и обеспечивает качество кода. Непрерывная доставка (CD) дополняет CI автоматическим развертыванием каждой версии в тестовом или продуктивном окружении [12].

В разработанном шаблоне конвейер CI/CD включает следующие этапы: сборка образов Docker на основе кода функции/сервиса, прогон модульных и интеграционных тестов, пуш образов в репозиторий артефактов, и развертывание в Nuclio. Важной особенностью является автоматический сбор метрик и логов на каждой стадии: GitLab CI генерирует отчёты сборки и тестов, а Prometheus-провайдер отслеживает работоспособность развернутых сервисов. Благодаря этой автоматизации достигается высокая прозрачность и управляемость процессов, что существенно ускоряет выпуск новых версий ПО и снижает число ошибок.

4. Результаты

4.1 Демонстрационная функция и её развёртывание

Для проверки работоспособности шаблона была реализована демонстрационная функция-микросервис в JupyterLab и развернута в Nuclio. Функция имитирует обработку данных «двойника» устройства из внешнего сервиса ORT/CI. В ходе работы разработчиками были выполнены следующие основные шаги: создание и отладка функции в Jupyter Notebook (с использованием nuclio-jupyter и API библиотеки dca-sm-sdk для состояния модели), конвертация блокнота в Nuclio YAML-конфигурацию с помощью jupyter nbconvert, и ручной (или CI-скриптом) деплой функции через веб-интерфейс или CLI Nuclio. После развёртывания функция автоматически регистрируется в системе наблюдения Prometheus и появляется в списке сервисов. На этапе тестирования подтвердилось, что при помощи шаблона DevOps можно быстро повторить процесс: внесение правок в код функции сопровождается обновлением Docker-образа и заменой развернутой функции, а новая версия начинает сбор метрик и логов в существующую инфраструктуру мониторинга. Таким образом, демонстрация показала целостность подхода и возможность масштабирования его на произвольные пользовательские решения.

5. Обсуждение результатов

Одним из ключевых преимуществ предложенного решения является высокая скорость релизов, по результатам которой автоматизация достигла $\approx 80\%$, заменив собой ручную работу. Вместе с тем, внедрение требует наличия Kubernetes-кластера и навыков работы с Docker, что может стать барьером для сотрудников без соответствующего опыта. Среди ограничений стоит отметить зависимость от конкретных FaaS-платформ (Nuclio) и инструментов мониторинга. В будущем планируется добавить поддержку альтернативных серверлес-решений и обеспечить полноценную интеграцию с системами CMDB и MES для закрытия полного цикла Continuous Intelligence.

В отличие от классических решений на базе Jenkins и Ansible, разработанный шаблон ориентирован на глубокую интеграцию с аналитической средой JupyterLab и без серверной архитектурой. Это делает его более подходящим для задач Data Science в промышленности, хотя и требует специфических навыков работы с Kubernetes и FaaS.

6. Заключение

В статье представлено решение по созданию шаблона DevOps для платформы ORT/CI, включающее стандартизированный конвейер CI/CD, контейнеризацию и механизмы

мониторинга. Разработанный шаблон опирается на инструменты Apache NiFi, JupyterLab (с Nuclio), Prometheus, Grafana и Loki, обеспечивая интеграцию обработки данных и средств визуализации метрик. Внедрение такого шаблона упрощает вход разработчиков в экосистему ORT/CI: существенно снижается порог знаний для развертывания новых сервисов, повышается повторяемость процессов и прозрачность инфраструктуры. Эксперименты показали, что применение DevOps-практик способствует ускорению доставки и повышению качества разрабатываемых решений в условиях производственных задач. В дальнейшем планируется расширить шаблон новыми инструментами и кейсами, а также автоматизировать полную интеграцию с промышленными системами (например, CMDB и MES) для реализации полного контура Continuous Intelligence.

Литература

1. Gartner Continuous Intelligence. Gartner Information Technology Glossary. – url: <https://www.gartner.com/en/information-technology/glossary/continuous-intelligence> (дата обращения: 19.04.2025).
2. Ли Цзюнь, Юй Гуанюань Актуальность внедрения процесса цифровизации в деятельность предприятий // Universum: экономика и юриспруденция. 2021. №11 (86). URL: <https://cyberleninka.ru/article/n/aktualnost-vnedreniya-protsessa-tsifrovizatsii-v-deyatelnost-predpriyatiy> (дата обращения: 25.04.2025).
3. Crosby S., Continuous Intelligence with Machine Learning, Digital Twin and Knowledge Graphs / Crosby S. // Secrets of Data & Analytics Leaders. - September 08. 2020. - url: <https://www.eckerson.com/articles/simon-crosby-continuous-intelligence-with-machine-learning-digital-twin-and-knowledge-graphs> (дата обращения: 26.04.2025).
4. M. Pratt, “What is Continuous Intelligence?”, TechTarget - url: <https://www.techtarget.com/searchbusinessanalytics/definition/continuous-intelligence> (дата обращения: 26.04.2025).
5. Что такое Apache NiFi // gitverse blog URL: <https://gitverse.ru/blog/articles/architecture/248-cto-takoe-apache-nifi> (дата обращения: 30.04.2025).
6. Project Jupyter, JupyterLab: A Next-Generation Notebook Interface - url:
7. <https://jupyter.org/> (дата обращения: 02.05.2025).
8. Nuclio: Serverless Platform for Automated Data Science - url:

9. <https://nuclio.io/> (дата обращения: 02.05.2025).
10. Prometheus, Open source metrics and monitoring for your systems and services - url: <https://prometheus.io/> (дата обращения: 02.05.2025).
11. Grafana, The open-source platform for monitoring and observability - url: <https://grafana.com/> (дата обращения: 02.05.2025).
12. Grafana Labs, Grafana Loki – Log aggregation system - url:
13. <https://grafana.com/oss/loki/> (дата обращения: 02.05.2025).
14. R. Castro Fernandez, V. F. Diaz, and M. Garijo, «Serverless Computing in the Cloud: An Architectural Review and Research Challenges», ACM Computing Surveys, vol. 54, no. 1, pp. 1-33, 2021.
15. DevOps Automation Pipeline Deployment with IaC (Infrastructure as Code) - url: <https://arxiv.org/abs/2503.16038> (дата обращения: 05.05.2025).