Архитектура и состав платформы wiSLA

ПАК **wiSLA** — корпоративная платформа для мониторинга качества услуг, управления SLA и оперативного учёта неисправностей. Платформа объединяет сбор измерений с сетевого оборудования и зондов, оценку состояния сервисов, расчёт показателей SLA, ведение паспортов неисправностей и доставку уведомлений заинтересованным пользователям.

На схеме показаны основные части платформы и связи между ними: роли пользователей, портал оператора, центральная обработка, функциональные подсистемы, внешние источники данных и хранилища. Опциональные компоненты (агрегатор зондов, коллектор NetFlow) выделены пунктирной рамкой.

1. Основные компоненты платформы

Центральная платформа (`wisla-engine`)

Ядро системы, которое координирует работу всех подсистем: принимает данные, рассчитывает показатели, управляет жизненным циклом сервисов и событий, обслуживает запросы портала.

Параметр	Значение
Артефакт	WAR, WildFly 14.0.1
Стек	Java 11, Spring 5.2, Hibernate 5.4
Включает модули	audit, data-collection, inventory-management, performance-management, sla-management, party-management, communicator
Развёртывание	Обязательный, основной процесс на WildFly

Портал оператора (`wisla-operator-portal`)

Веб-интерфейс для работы с платформой. Доступ разграничивается по ролям:

- **Пользователь** — просмотр состояния сервисов, отчётов и уведомлений в рамках предоставленных прав;
- **Оператор SLA** — мониторинг качества, управление паспортами неисправностей, формирование отчётов SLA;
- **Системный администратор** — настройка инфраструктуры, пользователей, ролей и параметров мониторинга.

Параметр	Значение
Технология	Angular 22 (статическое SPA)
Развёртывание	Статика на WildFly, API → `wisla-engine`

Сбор данных (`wisla-data-collection`)

Подсистема взаимодействия с измерительным оборудованием и источниками телеметрии. Собирает показатели с зондов wiProbe, сетевых устройств (SNMP, CLI), веб-ресурсов, а также принимает данные NetFlow. При необходимости данные могут поступать через внешние интеграционные интерфейсы.

Параметр	Значение
Размещение	Модуль внутри engine
Протоколы	wiProbe, SNMP, CLI, HTTP
Шина сообщений	ActiveMQ Artemis (JMS)

Учёт инфраструктуры (`wisla-inventory-management`)

Ведение каталога ресурсов и сервисов: точки доступа, зонды, тесты, связи между сервисами и показателями качества. Это «единый источник правды» о том, что именно мониторится и как устроена инфраструктура заказчика.

Параметр	Значение
Размещение	Модуль внутри engine
База данных	PostgreSQL (Hibernate)

Мониторинг качества сервисов (`wisla-performance-management`)

Непрерывная оценка состояния сервисов на основе поступающих метрик. Система отслеживает пороговые значения, фиксирует отклонения и формирует актуальный статус каждого сервиса.

Параметр	Значение
Размещение	Модуль внутри engine
Обработка	SQM-монитор на JMS

Управление SLA (`wisla-sla-management`)

Расчёт показателей готовности, времени неготовности и компенсаций с учётом согласованных перерывов. Формирование периодических отчётов SLA и печатных форм по договорным требованиям.

Параметр	Значение
Размещение	Модуль внутри engine
База данных	PostgreSQL (конфигурация SLA)
Метрики	TSDB (исторические показатели)

Учёт неисправностей (`wisla-sla-management`)

Ведение **паспортов неисправностей** — от регистрации инцидента до закрытия. Поддерживаются этапы обработки, приостановка и возобновление, эскалация и связь с изменением статусов сервисов.

Параметр	Значение
Размещение	Модуль внутри engine
База данных	PostgreSQL (паспорта неисправностей)

Управление контрагентами и доступом (`wisla-party-management`)

Учёт контрагентов, пользователей, ролей и прав доступа. Настройка каналов подписок на уведомления (электронная почта, портал, мобильные каналы).

Параметр	Значение
Размещение	Модуль внутри engine
База данных	PostgreSQL (схема `party_management`)

Уведомления (`wisla-notification-service`)

Доставка сообщений о событиях заинтересованным пользователям по выбранным каналам. Уведомления формируются при изменении статусов сервисов, открытии или закрытии неисправностей, достижении порогов и других значимых событиях.

Параметр	Значение
Тип	Отдельный микросервис
Стек	Spring Boot 4.1, Java 25
Артефакт	JAR (standalone)
Интеграция	Kafka (consumer событий от engine)
База данных	PostgreSQL read-only (схема `party_management`) для подписок
Каналы	email, Telegram, portal push

Аудит (`wisla-audit`)

Журналирование действий пользователей и системных событий для контроля и расследования.

Параметр	Значение
Размещение	Модуль внутри engine
База данных	PostgreSQL

Отчёты и печатные формы (`wisla-sla-management` + `print-form-plugins`)

Генерация отчётов SLA, отчётов о покрытии и других документов по настраиваемым шаблонам.

Параметр	Значение
Логика	В engine
Плагины печатных форм	JAR в WildFly `wisla_plugins/`

Коллектор NetFlow (`wisla-netflow-collector`)

Приём и обработка данных сетевого трафика от NetFlow-источников.

Параметр	Значение
Артефакт	Опциональный WAR на WildFly

Агрегатор зондов (`wisla-wiprobe-aggregator`)

Опциональный компонент для сценариев с большим числом удалённых зондов wiProbe. Агрегирует HTTP-данные с нескольких зондов и пересылает их в центральную платформу, снижая нагрузку на каналы связи.

Параметр	Значение
Тип	Опциональный standalone JAR
Стек	Spring Boot 2.7, Java 11
Назначение	Агрегирует HTTP-данные с нескольких wiProbe, пересылает в wiSLA

---

2. Внешние источники и интеграции

Платформа получает данные не только от собственных зондов, но и от внешних систем:

Источник	Назначение
Измерительное оборудование (wiProbe, SNMP-устройства)	Сбор метрик качества и доступности
Агрегатор зондов (опционально)	Промежуточная агрегация данных с нескольких wiProbe перед передачей в платформу
NetFlow-источники	Анализ сетевого трафика
Внешние OSS/BSS-системы	Импорт данных о сервисах и ресурсах, обмен событиями
Специализированные интеграции	Подключение к отраслевым системам заказчика (по сценарию поставки)

Состав интеграций определяется **конкретной инсталляцией** и может включать только те компоненты, которые нужны заказчику.

---

3. Хранение данных

Платформа использует несколько типов хранилищ:

Хранилище	Технология	Содержание
База конфигурации и учётных данных	PostgreSQL 42.x	Сервисы, пользователи, настройки, паспорта неисправностей, аудит, схема `party_management`
Хранилище метрик	HBase 2.x **или** TimescaleDB (PostgreSQL)	Измерения и показатели качества за длительный период
Журнал событий	Kafka	События уведомлений (engine → notification-service)
Внутренние очереди	ActiveMQ Artemis (в WildFly)	Очереди сбора данных и обработки метрик

Хранилище	Технология	Содержание
Кластерный кэш	Hazelcast 4.x	Кэширование в кластере engine

Конкретная технология хранения метрик (`HBASE_V2`, `TSDB_PG` или `IN_MEM`) выбирается при развёртывании и не влияет на пользовательские сценарии работы с платформой.

Примечание

Описание отражает архитектуру платформы wiSLA: логический состав компонентов и их технические характеристики. Параметры конфигурации серверов, сетевые настройки и детали развёртывания конкретной инсталляции приведены в эксплуатационной документации поставки.
