# Продукт


- Обзор для клиента
- Концепции
- SLA и индекс здоровья
- Пороги
- FAQ
- Глоссарий

# Обзор для клиента

# wiSLA 6 — материалы для клиента (пресейл)

Краткий обзор продукта, перечень возможностей и готовый текст для письма клиенту.
Для ИБ-формулировок см. Безопасность. Для терминов — Концепции.

## Что есть «из коробки»

| Артефакт | Где | Для чего |
|---|---|---|
| **Документация** | /docs/ (Docusaurus) | Пользовательская + админская; можно дать ссылку на стенд |
| **Безопасность для пресейла** | security.md | Готовые формулировки «что говорить клиенту» по ИБ |
| **Калькулятор сайзинга** | Калькулятор сайзинга | KE, retention, топология; есть **экспорт в PDF** |
| **Каталог шаблонов** | Golden templates DoD | 61 golden template (59 Supported) — технический DoD, не маркетинг, но подтверждает глубину мониторинга |
| **Чеклист требований клиента** | client-requirements-gap-analysis · PDF | Сопоставление обязательных требований заказчика с wiSLA 6 (есть / частично / нет) |
| **Эпики wiSLA VI** | wisla6-vi-epics-what-exists · PDF | Что уже есть по OTel, единой топологии и цифровому опыту |

## Что можно выслать клиенту прямо сейчас

1. **Ссылку на демо** + короткий walkthrough по порталу (или screencast).
2. **Краткий one-pager** — раздел Готовый текст для письма ниже.
3. **Ссылку на docs** — Концепции, Быстрый старт, Безопасность.
4. **PDF из калькулятора сайзинга** — если обсуждаете объём (KE, retention).
5. **PDF по эпикам wiSLA VI** — что уже есть (OTel, топология, цифровой опыт).

## Позиционирование

**wiSLA 6** — единый портал мониторинга инфраструктуры и доступности услуг: инвентарь, метрики, SLA, события и эскалации в одном окне (без «зоопарка» poller + Grafana + Grafana + CMDB + ITSM).

- **Проблема:** инвентаризация, доступность и каналы реакции разорваны между wiSLA, Prometheus, Grafana, CMDB, ITSM и мессенджерами.

- **Решение:** полный контур в одном окне — инвентарь → доступность и SLA → каналы реакции.
- **Эффект:** минуты вместо недель на онборд; один портал вместо зоопарка; ноль прыжков «куда эскалировать».

# Перечень возможностей

## Инфраструктура (гибкость уровня enterprise-мониторинга)

- Агенты **Linux / Windows** с удалённым обновлением флота и эталоном конфигурации
- Шаблоны с макросами и кнопкой **«Настроить»** (без YAML с нуля)
- Импорт и runtime шаблонов wiSLA (items, LLD, triggers, preprocessing) — YAML / JSON / XML
- Коллекторы: SNMP, HTTP / JMX / ODBC-опросы, SNMP trap, syslog
- **Каталог из коробки:** ОС, СУБД, Docker, сеть (Cisco / MikroTik / Huawei / Juniper…), middleware (Apache, RabbitMQ, ClickHouse, Proxmox, HAProxy, OpenSearch, Redpanda, Tomcat и др.) — см. Golden templates DoD
- **Редактор своих шаблонов узлов** — node-templates-builder

## Онбординг и автоматизация

- **Обнаружение** ICMP / SNMP / TCP + **Action Flows** («найти → сразу в мониторинг»)

## Доступность и SLA

- **Индекс здоровья** + **SLA за период** + пороги на одном экране
- **Канальный мониторинг** (ICMP / HTTP / TLS / traceroute) между зондами — Профили каналов
- **Карта сети** и **топология** — maps

## Реакция на инциденты

- Problems, ack / bulk, **воркфлоу уведомлений** (email, Telegram, webhook) — workflows
- **Дашборды** — свои виджеты + интеграция Grafana — dashboard

## Корпоративный контур

- Роли: **ADMIN / OPERATOR / VIEWER / INFOSEC**
- **LDAP / AD**, **OIDC SSO**, **MFA**, аудит с hash-chain, SIEM — security.md
- **Масштаб:** от пилота до 10 000–100 000 КЕ (NAT, HA, георезерв) — см. калькулятор сайзинга

# Про «гибкость шаблонов и LLD»

Это не только маркетинг — в продукте заложено:

- **wiSLA export-compatible runtime** — items, LLD, triggers, preprocessing, import YAML / JSON / XML — zabbix-template-runtime
- **Template Builder** + golden catalog с UI «Настроить» для кредов
- `GET /api/v1/services/{id}/unsupported-features` — честно показывает, что ещё не поддерживается (не «тихий no-op») — zabbix-compatibility-matrix

Для технически подкованного клиента можно приложить ссылку на матрицу совместимости (внутренняя docs, не для всех).

# Готовый текст для письма клиенту

> **wiSLA 6** — единый портал мониторинга инфраструктуры и доступности услуг: инвентарь, метрики, SLA, события и эскалации в одном окне (без «зоопарка» poller + Grafana + Grafana + CMDB + ITSM).
>
> **Ключевые возможности:**
>
> **Инфраструктура (гибкость уровня enterprise-мониторинга):**
>
> - Агенты Linux/Windows с удалённым обновлением флота
> - Шаблоны с макросами и кнопкой «Настроить» (без YAML с нуля)
> - Импорт и runtime шаблонов wiSLA (items, LLD, triggers, preprocessing)
> - SNMP, HTTP/JMX/ODBC-опросы, SNMP trap, syslog
> - Каталог из коробки: ОС, СУБД, Docker, сеть (Cisco/MikroTik/ Huawei…), middleware (Apache, RabbitMQ, ClickHouse, Proxmox и др.)
> - Редактор своих шаблонов узлов
>
> **Онбординг и автоматизация:**
>
> - Обнаружение ICMP/SNMP/TCP + Action Flows («найти ⟶ сразу в мониторинг»)
>
> **Доступность и SLA:**
>
> - Индекс здоровья + SLA за период + пороги на одном экране
> - Канальный мониторинг (ICMP/HTTP/TLS/traceroute) между зондами
> - Карта сети и топология
>
> **Реакция на инциденты:**
>
> - Problems, ack/bulk, воркфлоу уведомлений (email, Telegram, webhook)
> - Дашборды (свои виджеты + интеграция Grafana)
>
> **Корпоративный контур:**
>
> - Роли (ADMIN / OPERATOR / VIEWER / INFOSEC)
> - LDAP/AD, OIDC SSO, MFA, аудит с hash-chain, SIEM

> • Масштаб: от пилота до 10 000–100 000 КЕ (NAT, HA, георезерв)
>
> Могу организовать **демо на стенде** (живой портал + docs) или прислать **оценку по сайзингу** под ваш объём.

# Чего пока нет

| Нет | Комментарий |
|---|---|
| Готового **PDF-брошюры / PPT** в репозитории | Можно собрать из этого документа + screencast демо |
| Отдельного **commercial one-pager** в marketing-формате | Можно собрать из этого документа + PDF сайзинга |
| Публичного **landing page** продукта | Демо и docs — на стенде заказчика / пилота |

# Практический next step

Если есть доступный стенд (альфа / демо):

1. Дать ссылку на демо-портал и `/docs/`.
2. Предложить **30-минутный walkthrough** по цепочке: *обнаружение* → *шаблон* → *SLA* → *событие* → *воркфлоу*.
3. При обсуждении объёма — калькулятор сайзинга с экспортом PDF.

Это сильнее любого PDF без живого показа.

# Концепции

# Концепции wiSLA

## Узел (host / service)

Объект мониторинга в инвентаре: сервер, устройство или логический сервис. Имеет интерфейсы, шаблоны, макросы, статус, метрики и события.

## Канал

Сервис канального мониторинга между двумя зондами. На канал вешают **один профиль измерений** (ICMP *или* UDP *или* TCP+HTTP), а не весь каталог: пробы конкурируют за путь и портят SLA.

Как выбрать набор: Профили каналов. Кто может быть зондом: SLAMon ≥ 1.14.10.

## Шаблон

Набор item'ов, триггеров, LLD и макросов, который применяется к узлу или каналу. Ускоряет типовой мониторинг (OS, ICMP, SNMP…). Для канала шаблон = тип пробы (P-Test, HTTP, traceroute…).

## Агент

Процесс `wisla-wiprobe-agent` на хосте: heartbeat, метрики, опционально channel jobs, remote update.

## Item / метрика

Единица сбора (ключ, интервал, тип collector: agent или server). Значения хранятся в ClickHouse и показываются на обзоре.

## Триггер / Problem

Условие на метриках. При срабатывании создаётся **Problem** в `/events`. Операции по проблеме (открытие, взятие, снятие) кормят воркфлоу уведомлений. Keep-alive проблему не дублирует.

## LLD

Низкоуровневое обнаружение: динамические item'ы (диски, интерфейсы…) на основе prototypes шаблона.

# Maintenance

Окно обслуживания узла: подавляет новые проблемы и связанные уведомления.

# SLA и индекс здоровья (ИЗ)

- **SLA** — готовность (доля времени UP) за период; считается из рядов метрик.
- **ИЗ** — мгновенный score узла по CPU/памяти/ICMP/агенту; история — только после снимков на сервере.

Где смотреть в UI, API и какой код править: SLA и индекс здоровья.

# Дашборд

Персональная раскладка виджетов (KPI, топы, проблемы, графики). Режим полного экрана прячет меню и шапку; заголовки виджетов — по наведению.

# Аудит

Журнал значимых действий и security-событий для расследований и compliance.

# Безопасность (ИБ)

Контроли доступа и аудита продукта: роли (включая INFOSEC), MFA, lockout, SIEM, LDAP.
Обзор и формулировки для клиента: Безопасность; план усиления: Roadmap ИБ.

# SLA и индекс здоровья

# SLA и индекс здоровья (ИЗ)

Краткий справочник: **чем отличаются** показатели, **где смотреть** в UI, **как считаются** и **какой код трогать**, если нужно переделать.

## Два разных сигнала

| | **SLA (готовность)** | **ИЗ (индекс здоровья)** |
|---|---|---|
| Смысл | Доля времени, когда объект **доступен** (reachability) за период | Мгновенный **score 0–100** по открытым проблемам правил шаблона здоровья |
| Где в списке Мониторинг | Чип **SLA** (у всех kind) | Чип **ИЗ** только у **узлов** (`POST /api/v1/health-index/bulk`) |
| Где на обзоре | Узел: блок SLA и полоска **Готовность скрыты** (решение не принято); лента **ИЗ** остаётся. Канал/доступность — SLA в шапке + лента. | Gauge ИЗ (узел) + контекст шаблона здоровья; история — строки `service_health_snapshots` |
| Источник | Ряды в ClickHouse (ICMP / agent / loss канала) | Открытые trigger-problems + effective health config |
| Можно ли восстановить «назад» | Да — из уже хранимых точек | Нет честно без снимков |

**Не путать:** высокий ИЗ ≠ выполненный SLA за сутки, и наоборот.

## Где в интерфейсе

1. **Список** `/services` — колонка **SLA / ИЗ**
   - узел: чип **ИЗ** из `POST /api/v1/health-index/bulk`
   - канал / доступность: один чип `SLA p%`
2. **Обзор** `/services/:id` — у **узла** блок SLA и полоска **Готовность скрыты**; лента **Индекс здоровья** остаётся; у канала/доступности — SLA в шапке + лента; ИЗ gauge справа от шапки
3. **Каталог** `/sla-sli/health-templates` — Шаблоны здоровья
4. **Над графиком** — лента **История статусов** (на узле сейчас только ИЗ; на канале/доступности — готовность). Маркеры смены шаблона/привязки/override — в блоке ИЗ

Подробнее для оператора: Сервисы и узлы, Шаблоны здоровья.

## Как считается SLA

- **Узел:** ICMP (`icmpping`) и/или агент (`agent.ping`). Если нет ни ICMP, ни агента, но есть **ODBC ping** (`db.odbc.select[pgsql.ping...]` / mysql) — готовность считается по нему.

**Готовность %** при ICMP+агент — OR. На ленте SLA: оба UP → ok; один UP → **warn**; оба DOWN → bad.
Сервис: `NodeAvailabilitySlaService` + `segmentsFromNodeReachability`.

- **Канал (по loss + RTT):** пороги шаблона **warn / crit** → банды (худший из loss и RTT **avg/max**):
  - **ok / warn / bad** — quality; **готовность % = доля времени не в bad** (жёлтое не сжигает).
    Полоска и SLA берут `max(rtt.avg, rtt.max)` — один медленный пакет в пробе красит ленту, даже если AVG на KPI ещё «норма». Карточки KPI показывают **последний AVG** (и тон задержки учитывает max). График в режиме AVG — среднюю; MAX — пики.
    `ChannelAvailabilitySlaService` + `fromLossTimeWeighted` / `fromLossAndRttTimeWeighted`.
- **Доступность услуги:** готовность = успех проверки (`*.ok`, иначе loss < crit как UP). RTT **не** входит в %; на ленте может дать **warn** (медленно), **bad** только при провале проверки.
  `segmentsFromAvailabilityCheck` / `availabilityReadySamples`.
- API сводки: `GET /api/v1/services/{id}/sla/summary?hours=...` (+ `template_key`, `direction` для канала).

Тон блока SLA относительно цели (по умолчанию 99%): **ok** ≥ цели, **warn** чуть ниже, **bad** сильнее — это про **итог %**, отдельно от цвета сегментов на timeline.

Окна ППР и исключений SLA вычитаются из знаменателя (`maintenance_excluded_ms`, `exception_excluded_ms`); на ленте — отдельные тона. Исключение SLA **не** глушит Events.

# Как считается ИЗ (узел)

Score строится **только из открытых проблем триггеров** и effective-конфига шаблона здоровья. Выражения триггеров и окна метрик в этом пути не вычисляются.

```
activeRulesInSeverity = count(enabled effective rules for severity)
ruleImpact = severityWeight / activeRulesInSeverity
penalty = min(100, sum(impact of firing enabled effective rules))
baseScore = 100 - penalty
finalScore = min(baseScore, lowest score ceiling among firing rules)
```

- Веса важности (Not classified, Information, Warning, Average, High, Disaster) в сумме 100%. Если у важности нет активных правил, её вес **перераспределяется** на остальные (в каталоге после seed такие оси хранятся как 0%). Если есть хотя бы одно активное правило, вес этой важности **не меньше 1%**.
- По умолчанию **info** и **not_classified** в шаблоне здоровья выключены. Каталог создаётся только если у шаблона мониторинга есть включённый триггер.
- Критичность `key_function` (потолок по умолчанию 60) и `service_outage` (0) не дают размыть отказ средним по многим правилам. В редакторе поле потолка видно, когда его можно править или когда значение уже задано.
- **Недоступность системы** — это отказ услуги, даже если в шаблоне важность Average/Warning/High. Иначе один AVERAGE среди ~8 правил даёт штраф ~3% (ИЗ **97%**) или HIGH даёт потолок 60% на мёртвом объекте. Сюда входят агент («agent is not available», `agent.offline`), **SNMP** («No SNMP data collection», ICMP ping unavailable), **ODBC** («Service is down / unreachable / unavailable», «Failed to fetch info data», порт БД), JMX/HTTP без агента (нет данных JMX, status/stats page unavailable, connection

unavailable). LLD-экземпляры (один backend, один интерфейс) не считаются отказом всей системы. Переопределение критичности **на узле** не трогается.
- LLD: одно правило на прототип; режимы `any_instance` / `affected_share` / `each_instance_capped`.
- Несколько шаблонов мониторинга на узле: правила **сливаются**; переопределения сервиса — разреженный overlay.
- Каталог: `/sla-sli/health-templates`. API: `GET /api/v1/services/{id}/health`, `GET /api/v1/services/{id}/health/breakdown` (разложение для Sankey/радара, только при открытии модалки; у правил есть `tags`; **без** записи снимка — persist только у worker после open/close события), `POST /api/v1/health-index/bulk`.
- На обзоре узла клик по блоку **индекса здоровья** и на списке **Мониторинг** клик по чипу **ИЗ** открывают модалку: справа от процента переключатель **Потоки** / **Радар**. Потоки — Sankey от узла **Здоровье** (цвет как у текущего ИЗ) ⟶ остаток по осям важности, штраф от открытых проблем, при необходимости срез потолка критичности. Спокойные правила одной оси сверх 12 сворачиваются в узел с числом («14 правил»); клик открывает полный список. Радар — вершины по тегам `scope:*` выбранных правил шаблона (локализованные имена), центр 0%, край 100% по категории. Bulk latest на списке по-прежнему без firing — разложение не префетчится.
- Снимки истории: партиционированная таблица Postgres `service_health_snapshots` — **только** смена тона (ok/warn/bad) или процента. Пересчёт после открытия/закрытия события на узле (система или вручную); смена шаблона здоровья / override тоже ставит узел в очередь. Нет периодического обхода флота.
- **BREAKING:** колонка `node_templates.health_index_keys` удалена (Flyway V141).

# История статусов (timeline)

## API

```
GET /api/v1/services/{id}/status-history?kind=sla|health|both&hours=24
```

Опционально: `template_key`, `direction` (канал).

Ответ: `tracks[]` с полями `kind`, `available`, `segments: [{ from, to, tone, value, issues? }]`.

`issues[]` на **warn/bad**: `{ key, tone, value }` — какие метрики сломали срез (ICMP/агент, loss/RTT, CPU/RAM/диски для ИЗ). Нужны для тултипа ленты и доп. линий на общем графике узла.

| kind | Поведение |
|---|---|
| `sla` | Узел: ok/warn(partial)/bad; **канал** loss+RTT; **доступность** ok/warn(RTT)/bad(fail); пробелы ⟶ **unknown**; пересечение ППР ⟶ **maintenance**, исключение SLA ⟶ **exception** |
| `health` | Только **node**; снимки score ⟶ ok/warn/bad. Если текущий ИЗ критический (0 / tone `bad`) и открыта недоступность системы (`agent.offline`, ICMP, SNMP/ODBC/JMX/HTTP «нет сбора / сервис недоступен»), лента **перекрашивает этот интервал в красный** — старые снимки ~97% не оставляют жёлтую «деградацию». Ручки выбора интервала на ленте ИЗ красятся по **этой** колбасе, не по SLA. Пока точек нет ⟶ `available: false` |

| kind | Поведение |
|---|---|
| `both` | Оба трека (обзор узла) |

Канал / доступность на FE запрашивают `kind=sla`; узел — `both`.

## Снимки ИЗ

- Пересчёт: `HealthIndexRecomputeWorker` (очередь, drain ~1.5 с) после **open/close** проблемы на сервисе, в том числе ручного закрытия; также после правки шаблона здоровья / override. При **удалении сервиса** ID снимается с очереди; persist не пишет summary/ snapshot для несуществующей строки `services`.
- История: `service_health_snapshots` (Postgres, RANGE по месяцу). Строка пишется **только** если сменились `%` или тон. INSERT снимка — отдельная транзакция, чтобы FK на удалённый сервис не abort'ил основной persist и не давал `25P02` в логах.
- GET попапа / breakdown **не** пишет снимок
- Retention партиций: `wisla.health.snapshot-retention-months` (по умолчанию 12); DDL-only job `HealthSnapshotPartitionMaintainer`
- ClickHouse / `metric_last_values` для ИЗ **не** используются

Пока снимков нет, UI показывает: *«История индекса здоровья появится после накопления снимков»* (после первого события или первого заполнения списка).

# Карта кода (чтобы переделать)

## Backend

| Что | Где |
|---|---|
| REST | `ServiceController` ⟶ `GET …/status-history`, `…/sla/summary` |
| Сборка истории | `ServiceStatusHistoryService` |
| Сегменты | `StatusHistoryMath` (+ тест `StatusHistoryMathTest`) |
| Причины warn/bad | `StatusHistoryIssueAnnotator` |
| Счёт ИЗ | `HealthScoringEngine` + `HealthIndexService` |
| Снимки истории | `ServiceHealthSnapshotStore` (`service_health_snapshots`) |
| Очередь пересчёта | `HealthIndexRecomputeWorker` (не флот-таймер) |
| Партиции | `HealthSnapshotPartitionMaintainer` |
| SLA узла | `NodeAvailabilitySlaService` |
| Кэш SLA | `SlaSummaryCache` (`wisla.sla.summary-cache-ms`) |
| SLA канала | `ChannelAvailabilitySlaService`, `SlaSummaryMath` |
| DTO | `ServiceDtos.StatusHistory*View`, `ChannelSlaSummaryView` |
| Док API | services-icmp.md |

## Frontend

| Что | Где |
|---|---|
| Список: два чипа | `service-list/service-list-page.*` (`loadHealth`, `showsHealth` / `showsSla`) |
| Клиент API | `services/data/services-api.service.ts` → `slaSummary()`, `statusHistory()` |
| Timeline UI | `shared/components/status-history-timeline/` (тултип на warn/bad/unknown) |
| Линии инцидента | `host-dynamics-chart` + `status-history-issues.ts` (доп. серии на общем графике) |
| Обзор узла | `hosts/host-overview/` (+ `host-overview-data.service.ts`, период графика) |
| Обзор канала / доступности | `services/channel-overview/`, `services/availability-overview/` |
| i18n | `services.badge.*`, `services.statusHistory.*` в `core/i18n/messages.ts` |
| Док UI | services-list-page.md, host-overview-page.md |

# Типовые вопросы

**В списке у узла «—» у SLA, а ИЗ есть.**
Так и задумано на compact-стенде: список не сканирует ClickHouse за 24 ч по каждому узлу. SLA узла — на обзоре. ИЗ живёт от latest.

**На timeline ИЗ пусто, SLA уже цветной.**
Нормально, пока не было смены ИЗ (open/close события) и нет строки в `service_health_snapshots`. Первый проход списка может записать стартовый снимок для узлов без summary.

**Почему на timeline у канала есть жёлтый сегмент, а у узла нет?**
У узла жёлтый есть, если жив только один из сигналов (ICMP **или** агент). Канал — quality loss+RTT. Доступность — жёлтый от медленного RTT. Цвета SLA и ИЗ одинаковые (норма / деградация / недоступно / нет данных). Жёлтый **не** снижает % готовности.

**Почему сегменты SLA только ok/bad, без warn?**
Устарело: у канала/доступности/узла (при двух сигналах) — ok/warn/bad. Warn/bad у **цифры** SLA vs цель — отдельный слой.

# Связанные документы

- Пользователь: Сервисы и узлы
- Админ / сайзинг: Системные требования
- Backend API: Services + ICMP
- Patch notes: changelog

# Пороги

# Пороги (узлы, каналы, доступность)

Единая картина: **где править**, **куда сохраняется**, **почему вкладка пустая**, **куда смотреть в коде**.

Техническая глубина (expression engine, skip reasons, smoke): triggers-thresholds.md.

## Три модели — не путать

| Kind | Где UI | Что сохраняется | Scope |
|---|---|---|---|
| **Узел** (`node`) | Обзор → **Показатели сервера** → вкладка **Пороги показателей** | Макросы **на сервисе** (`PUT /api/v1/services/{id}/macros`), перекрывают шаблон | Только этот узел |
| **Канал** (`channel`) | Обзор → **Показатели** \| **Пороги** (и каталог `/templates/channels`) | `channel_templates.thresholds_json` (`PUT /api/v1/channel-templates/{id}`) | **Все** каналы с этим шаблоном |
| **Доступность** (`availability`) | Обзор → **Показатели** \| **Пороги** (и каталог `/templates/availability`) | `availability_service_templates.thresholds_json` | **Все** сервисы с этим шаблоном |

На канале/доступности в UI есть баннер: пороги шаблона, не override на один сервис (override — вне текущей волны).

```
flowchart TB
  subgraph nodeHost [Node]
    TabN[Tab Porogi] --> API_N["GET threshold-macros / PUT macros"]
    API_N --> HostMacros[service macros override template]
    HostMacros --> Hints[threshold-hints bands]
    HostMacros --> Triggers[TriggerThresholdScheduler]
  end
  subgraph chAv [Channel and Availability]
    TabC[Tab Porogi] --> API_T["PUT channel/availability-templates"]
    API_T --> ThrJson[thresholds_json warn/crit/op]
    ThrJson --> KPI[KPI tone + chart lines]
    ThrJson --> ChProb[ChannelThresholdProblemScheduler]
  end
```

## Узел: как заполняется вкладка

1. Берём effective macros сервиса (шаблон + host overrides).
2. `ThresholdMacrosService.discover` оставляет только **threshold-like** имена.
3. UI группирует (cpu / memory / disks / network / other), пары WARN+CRIT или single-level.
4. Save merge'ит в host macros (не затирает connection-макросы вроде `{$PVE.TOKEN.*}`).

## Что считается порогом (discovery)

Попадают:

- `WARN` / `CRIT` / `WARNING` / `TRIGGER`
- `UTIL.MAX`, `PCT.MAX`, `PUSED`, `LOSS`, `PFREE.MIN`, `AVAILABLE.MIN`, `FREE.PCT.MIN`, …
- bare окончание `.MAX` / `.MIN` / `.HIGH` / `.LOW`
  примеры: `JVM.HEAP.USED.PCT.MAX`, `MSSQL.DEADLOCKS.MAX`, `HAPROXY.PX.SCUR.MAX`

Не попадают (намеренно):

- connection / LLD filters: `USER`, `PASSWORD`, `URI`, `DSN`, `PATH`, `URL`, `MATCHES`, `INTERVAL`, …
- `KERNEL.MAXFILES` / `FUZZYTIME` / `HOST.` / `AGENT.` / non-trigger `PVE.*`
- шаблоны **только** с connection-макросами (Jenkins URL, Vault token, …) — вкладка «Всего 0 порогов» нормальна, пока в seed нет threshold-макросов

Состояния в таблице **Показатели** (`Норма` / `Внимание` / `Критично` / `Не поддерживается`) считаются отдельно на клиенте из value + hints/bands — см. host-overview-page.md.

# Канал и доступность

- Карта порогов: ключ метрики ⟶ `{ warn, crit, op: gt|lt }`.
- Defaults: `ChannelMetricThresholds` (backend) / `channel-metric-thresholds.ts` (FE).
- KPI и линии на графике читают map активного шаблона.
- Problems по каналу: crit из map ⟶ `channel.thr.*` (см. OpenSpec `channel-threshold-problems`).
- Shared UI grid: `frontend/…/shared/components/metric-thresholds-editor/`
- Панель overview: `service-template-thresholds-panel` (`kind=channel|availability`)

Каталог шаблонов каналов: форма «книга» (слева параметры, справа тот же grid).

# Если «Всего 0 порогов» на узле

1. Какие шаблоны привязаны? Есть ли в seed `{$….WARN/CRIT/MAX}`?
2. Перезапущен ли backend после смены seed / discovery?
3. `GET /api/v1/services/{id}/threshold-macros` — пустой массив?
4. Не connection-only шаблон ли это?

Для JMX после расширения discovery должны появиться `JVM.*.PCT.MAX` / `JVM.THREADS.MAX`.

**JVM MemoryPool:** высокий % у Eden/Survivor/Metaspace — **не** авария (HotSpot `max=-1`, считаем от `committed`). Триггер и UI не красят их в критичность; алерты — только пулы с `Usage.max>0` (Old Gen, CodeCache, …). Подробно: services-jmx-ipmi.md.

# Куда лезть при доработке

| Задача | Файлы |
|---|---|
| Шире/уже discovery макросов узла | `backend/…/template/ThresholdMacrosService.java` + `ThresholdMacrosServiceTest` |

| Задача | Файлы |
| --- | --- |
| Seed-макросы шаблона | `backend/src/main/resources/seed/zabbix/**` |
| Вкладка узла | `host-metric-thresholds-panel.*`, `host-metrics-panel` |
| Вкладка канала/доступности | `service-template-thresholds-panel.*`, `channel-overview-page.*`, `availability-overview-page.*` |
| Grid warn/crit/op | `metric-thresholds-editor.*`, `channel-metric-thresholds.ts` |
| Problems по триггерам узла | `TriggerThresholdScheduler`, expression engine — triggers-thresholds.md |
| JMX pool % / ложные алерты Eden·Metaspace | seed `java_jmx` + `isInformationalJvmPoolPercentKey` — services-jmx-ipmi.md |
| Problems по порогам канала | `ChannelThresholdProblemScheduler` |

# Вне текущей волны (не сломано — просто не сделано)

- Override порогов **на один** канал/availability-сервис (как макросы узла)
- Правка порога из строки таблицы метрик
- Полный выражения во expression engine

# FAQ

# Частые вопросы

## Вход и доступ

**Не пускает после нескольких ошибок.**
Сработала блокировка brute-force (HTTP 429). Подождите TTL lockout или обратитесь к администратору.

**После пароля просит код.**
Включена MFA. Введите код из authenticator или backup code.

**Аккаунт «ожидает подтверждения».**
Статус PENDING — нужен approve администратора.

## Мониторинг

**Узел создан, метрик нет.**
Проверьте агент online, шаблон, интерфейс/`access_point`, подождите интервал опроса.
Админу: Диагностика.

На обзоре «Доступен» + **Критично** у ICMP и «—» у потерь/RTT — шаблон Linux может быть живой: график и Min/Avg/Max читают `/metrics/stats` (ClickHouse), а «Сейчас» — `/metrics/latest`. Latest берётся из Postgres (`metric_last_values`) сразу по `service:` и агенту; ClickHouse — только keyed, если ключа нет в снимке (полный `latestAll` не используется). На CH 24.8 старый SQL с `max(ts) AS ts` давал `ILLEGAL_AGGREGATION` → API `[]`. Если в логах `cannot execute INSERT in a read-only transaction` на `metric_last_values` — снимок не писался из-за dependent ingest в read-only TX (фикс: upsert в `REQUIRES_NEW`). Если CPU всё равно пустой после деплоя и одного цикла опроса — нет last-value и keyed CH не ответил.

**Нужно ли вешать все тесты на один канал?**
Нет. Один канал — один профиль (ICMP *или* UDP *или* TCP+HTTP). RFC и traceroute — отдельно. Подробнее: Профили каналов.

**Поддерживается ли SLAMon 1.14.10?**
Софтовый аналог измерений — `wisla-wiprobe-agent` (каналы и availability с агента). Классический SLAMon/wiProbe ≥ 1.14.10 регистрируется в списке (паспорт, online); jobs на него не выдаются. Железо WPE/Berkut — позже. Матрица: SLAMon ≥ 1.14.10.

**Проблема не приходит в Telegram/MAX/почту.**
Уведомления выключены, низкая severity, узел в maintenance, неверные credentials.

**Зачем maintenance?**
Плановые работы без шума в Events и каналах алертов. Со списка Мониторинг → выделение → **ППР** (интервал и тип окна).

**Чем SLA отличается от индекса здоровья (ИЗ)?**
SLA — доля времени доступности за период; ИЗ — мгновенный score узла. В списке у узла виден чип **ИЗ**; чип SLA на списке узлов часто «—» (процент — на обзоре). У каналов/

доступности в списке — SLA %. История ИЗ на ленте — после снимков (~5 мин). Подробнее: SLA и индекс здоровья.

**Где править пороги и почему на узле вкладка пустая?**
Узел — макросы на этом сервисе; канал и доступность — `thresholds_json` **шаблона** (влияет на все с ним). Пустая вкладка на узле часто значит connection-only шаблон. Сводка: Пороги.

**Почему Metaspace / Eden ~90–98% — это авария?**
Нет. У этих пулов HotSpot часто нет лимита (`max=-1`); % считается от `committed` и бывает высоким «по дизайну». Проблема и красная шкала — только для пулов с реальным max (Old Gen, CodeCache) и для heap %. Подробно: JMX / IPMI.

**Почему на обзоре «История статусов» пустая у ИЗ?** Пока не было смены индекса (open/close события на узле) — в `service_health_snapshots` нет точек. SLA-полоска может уже быть — она из ICMP/agent рядов.

**Почему кольцо ИЗ жёлтое, а полоса зелёная?**
Кольцо — сейчас (в т.ч. диски). Полоса — снимки раз в ~5 мин; в снимок входят те же тома. После деплоя жёлтый сегмент появится со следующего снимка. Подробнее: SLA и индекс здоровья.

# Интерфейс

**Как сменить тему / язык?**
В подвале бокового меню (или справа на верхней/нижней рейке). Тема — **восемь** вариантов (`classic`, `neon`, `air`, `ocean`, `dark`, `cosmos`, `gray`, `wow`), ключ `wisla.theme` (тот же в docs; палитры совпадают с порталом). Язык UI — **Ru / En / Kz / Uz** (`wisla.locale`); в docs переключатель в шапке синхронизируется с порталом, но текст статей пока в основном на русском.
Подробнее: Быстрый старт → Язык.

**Как переместить меню наверх?**
Иконка раскладки в блоке инструментов — перетащите меню вверх, вниз, влево или вправо. На верхней/нижней рейке кнопка плотности рядом с аватаром включает подписи пунктов.

**Где документация и API?**
Справка (?) → Документация / Документация API.

**Как развернуть дашборд на весь экран?**
На дашборде кнопка **Полный экран** (иконка в панели, рядом с меню Поделиться/Экспорт) скрывает меню с логотипом и шапку. Заголовок виджета — при наведении на карточку. Выход — та же кнопка или Esc.

# Документация не открывается локально

Нужен процесс Docusaurus (`docs-site` на :3400) и frontend proxy. См. docs-and-swagger.

# Глоссарий

# Глоссарий

| Термин | Значение |
|---|---|
| ADMIN | Роль полного управления |
| Acknowledge (ack) | Подтверждение проблемы оператором |
| Agent | Агент сбора на хосте |
| Audit | Журнал действий |
| Brute-force | Подбор пароля; порог → lockout |
| Channel | Канальный сервис мониторинга |
| Channel profile | Готовый набор 1–2 проб на канал (качество IP, голос, приложения, маршрут, ёмкость) |
| ClickHouse | Хранилище метрик |
| Контрагент | Организация или филиал в каталоге `/counterparties`: иерархия, связанные и ответственные пользователи |
| Discovery | Сканирование сети / LLD |
| Flyway | Миграции схемы PostgreSQL |
| INFOSEC | Роль «Администратор ИБ» (read-only admin surfaces) |
| Item | Элемент данных / метрика |
| LLD | Low-level discovery |
| Lockout | Временная блокировка входа |
| Maintenance | Окно обслуживания |
| MFA / TOTP | Двухфакторная аутентификация |
| API token (personal) | Долгоживущий Bearer пользователя (`wsl.pat....`) |
| API token (system) | Service token инстанса (`wsl.sys....`) |
| OPERATOR | Роль «Оператор» (эксплуатация) |
| Воркфлоу | Холст уведомлений (триггер → wait/switch → notify) |
| Retention | Срок хранения данных |
| Severity | Критичность проблемы |
| SIEM | Внешняя система, webhook аудита |
| SLA | Готовность: доля времени доступности за период (не путать с ИЗ) |
| SLAMon / wiProbe | Линейка Wellink: софт ≥ 1.14.10 — инвентарь в wiSLA 6; измерения каналов и availability — `wisla-wiprobe-agent`. Железо — отложено. Матрица |

| Термин | Значение |
| --- | --- |
| ИЗ / HI | Индекс здоровья узла (score 0–100 по текущим метрикам) |
| Status history | Timeline сегментов ok/warn/bad на обзоре (`/status-history`) |
| Threshold / порог | Правило warn/crit для метрики; у узла — макросы шаблона/хоста, у канала и доступности — `thresholds_json` шаблона. См. Пороги |
| VIEWER | Роль «Пользователь» (только просмотр) |
| ZBXD | Passivecсивный протокол опроса агента |
| wiSLA | Система мониторинга (этот продукт) |