



NUVOLA

ПРОГРАММНЫЙ КОМПЛЕКС

Версия 7.1

Руководство администратора

Функциональные характеристики

Подготовка модулей к эксплуатации

Аннотация

Настоящее Руководство администратора входит в комплект эксплуатационной документации по Программному комплексу Нувола (далее – ПК Нувола) и предназначено сотрудникам предприятия, отвечающим за подготовку к работе и администрирование ПК Нувола.

Руководство содержит общие сведения о ПК Нувола, его функциональные характеристики и принципы работы, расшифровки и объяснения используемых терминов и понятий.

В документе описана технология установки и подготовки к эксплуатации ПК Нувола, приведены действия по настройке его модулей, доступные администратору посредством командной строки и в Web-интерфейсе.

© ООО «Венто Технологии», 2020



<https://www.ventocloud.ru/>

info@ventocloud.ru

Документ является собственностью ООО «Венто Технологии» и защищен законодательством о правах на результаты интеллектуальной деятельности. Никакая информация, содержащаяся в данном документе, не может быть воспроизведена, искажена, переработана, переведена на иностранный язык, записана или скопирована для любых коммерческих целей. Не допускается передача данного документа третьей стороне без письменного согласия ООО «Венто Технологии». Графические изображения и названия продуктов, упоминаемые в данном документе, могут быть зарегистрированными товарными знаками, охраняемыми законодательством о правах на результаты интеллектуальной деятельности. ООО «Венто Технологии» оставляет за собой право вносить изменения в содержание данного документа без предварительного уведомления.

Условные обозначения

В документе приняты следующие шрифты и обозначения:

Примеры использования	Тип информации
Дерево конфигурации – это... кнопка Сохранить пункт меню Конфигурировать	Термины, названия компонент ПК Нувола, наименования элементов пользовательского интерфейса (поля, кнопки, пункты меню).
... выбрать значение off ...	Значения полей, отображаемых в пользовательском интерфейсе.
<code>konfne --devlist</code> ... директория <code>C:\Program Files (x86)\ ...</code>	Команды, фрагменты кода программы, текст файлов.
См. Справочники ^[49] ... с помощью дополнительной настройки ^[12] ...	Ссылка на другой раздел документа. Значок возле ссылки содержит номер страницы, на которой расположен целевой раздел.

Сокращения

ОС – операционная система.

ПК – программный комплекс.

ПМ – программный модуль.

СУБД – система управления базой данных.

Содержание

1. Введение	13
1.1. Общие сведения о ПК Нувола	14
1.1.1. Назначение ПК Нувола	14
1.1.2. Пользовательские роли и требования к квалификации персонала	15
1.1.3. Требования к техническим средствам и программному обеспечению	16
1.1.4. Компоненты ПК Нувола	17
1.2. Назначение и архитектура модуля Нувола Топаз	17
1.3. Назначение и архитектура модуля Нувола ESB	18
1.4. Назначение и архитектура модуля Нувола Виста	19
1.5. Назначение и архитектура модуля Нувола Аналитика	20
1.6. Назначение и архитектура модуля Нувола Координатор	20
1.7. Назначение модуля Нувола Логистика	21
2. Понятия и концепции	23
2.1. Дерево конфигурации и его объекты	24
2.1.1. Шаблоны объектов	25
2.2. Системные процессы	26
2.3. Сигналы и оповещения	27
2.3.1. Основные определения	27
2.3.2. Глобальные и локальные значения параметров	28
2.3.3. Вычисление статусов сигналов и отправка оповещений	28
2.3.4. Параметры сигналов и алгоритмы определения статуса	29
2.3.4.1. Функция и период для определения значения сигнала	29
2.3.4.2. Алгоритм определения статуса сигнала типа Gauge	30
2.3.4.3. Алгоритм определения статуса сигнала типа InvertGauge	31
2.3.4.4. Алгоритм определения статуса сигнала типа SmartGauge	33
2.3.4.5. Алгоритм определения статуса сигнала типа Availability	34
2.3.4.6. Признак Лидер	35
2.3.5. Класс оповещений Simple	35
2.3.6. Методы и типы оповещений	36
2.4. Информационные панели	36
2.5. Пользователи	36
2.5.1. Глобальные политики	37
2.5.2. Права доступа к узлам дерева конфигурации	37
2.5.3. Права пользователя, принадлежащего нескольким группам пользователей	38
2.5.4. Администратор (пользователь root) и другие пользовательские роли	38

2.5.5. Использование политик безопасности для внутренней аутентификации	39
3. Установка и удаление модуля Нуволы Топаз.....	41
3.1. Подготовка к установке: настройка сервера Linux.....	42
3.2. Установка модуля Нуволы Топаз.....	43
3.3. Проверка правильности функционирования.....	45
3.4. Удаление модуля Нуволы Топаз.....	46
4. Установка и удаление Web-интерфейса.....	47
4.1. Команды для установки и удаления инсталляционного файла.....	48
4.2. Проверка правильности установки.....	48
4.3. Настройки рабочей станции для соединения с сервером Нуволы.....	50
5. Установка и удаление модуля Нуволы Аналитика.....	53
5.1. Команды для установки и удаления инсталляционного файла.....	54
5.2. Настройка сервера Нуволы Аналитика.....	54
6. Установка и удаление модуля Нуволы Координатор.....	57
6.1. Команды для установки и удаления инсталляционного файла.....	58
6.2. Настройка платформы Camunda BPM.....	59
6.2.1. Настройка порта	59
6.2.2. Перезапуск сервера Координатор	59
6.3. Настройка конфигурационных файлов Нуволы Координатор.....	59
6.4. Настройка скриптов для схем бизнес-процессов.....	60
7. Установка и удаление модуля Нуволы Логистика.....	63
8. Настройка и использование модуля Нуволы Топаз.....	65
8.1. Запуск и остановка ПМ Нуволы Топаз.....	66
8.2. Настройки для администрирования пользователей.....	66
8.2.1. Настройка дополнительных параметров учетной записи	67
8.2.2. Настройка политик безопасности	68
8.2.3. Создание глобальных политик и прав доступа	70
8.2.4. Просмотр лог-файлов о действиях пользователей	73
8.3. Системные процессы ПМ Нуволы Топаз.....	73
8.3.1. Лог-файлы	74
8.3.2. Статусы системных процессов	75
8.3.3. Используемые порты	76
8.4. Утилиты ПМ Нуволы Топаз.....	76

8.5.	Настройка и обслуживание базы данных конфигурации.....	77
8.5.1.	Сервер конфигурации	77
8.5.2.	Резервное копирование и восстановление конфигурации	78
8.5.2.1.	Параметры резервного копирования (конфигурационный файл backup.conf).....	78
8.5.2.2.	Действия с резервными копиями	79
8.5.2.3.	Восстановление базы данных конфигурации из резервной копии.....	80
8.5.3.	Обновление ПМ Нувола Топаз	80
8.6.	Настройка дерева конфигурации.....	80
8.7.	Создание шаблонов объектов	82
8.7.1.	Правила создания нового шаблона объекта	82
8.7.2.	Пример шаблона объекта HPPrinter	83
8.7.2.1.	Device Header (Заголовок объекта).....	84
8.7.2.2.	Device Program (Программа объекта).....	84
8.7.2.3.	Device DATA (Данные объекта).....	86
8.7.3.	Правила написания шаблона объекта	90
8.7.3.1.	Получение информации с удалённого хоста, snmpget	90
8.7.3.2.	Использование функции обработки func	90
8.7.3.3.	Функция обработки целевого объекта collector-hook	91
8.7.3.4.	Правила маппинга.....	91
8.7.4.	Масштабирование графиков	92
8.7.5.	Универсальный плагин объектов	93
8.7.5.1.	Использование универсального плагина объектов.....	93
8.7.5.2.	Формат XML-файла.....	94
9.	Администрирование ПК Нувола через Web-интерфейс.....	99
9.1.	Управление пользователями.....	100
9.2.	Настройка дерева конфигурации.....	100
9.2.1.	Добавление нового объекта в дерево конфигурации	100
9.2.2.	Просмотр и изменение параметров объекта	102
9.2.3.	Настройка конфигурационных инструментов	103
9.2.3.1.	Настройка папки.....	103
9.2.3.2.	Настройка отчета.....	104
9.2.3.3.	Настройка сводного графика.....	109
9.2.4.	Конфигурирование устройств	111
9.2.4.1.	Стандартные параметры устройства.....	112
9.2.4.2.	Конфигурирование устройства с автоматическим выбором шаблона объектов	113
9.2.5.	Переконфигурация объекта	114
9.2.6.	Удаление объектов	114
9.2.6.1.	Удаление с помощью команды Удалить.....	115
9.2.6.2.	Удаление с помощью команды Удалить везде.....	116
9.2.6.3.	Удаление с помощью команды Уничтожить.....	117
9.3.	Редактор сигналов и оповещений (обзор приложения).....	117
9.3.1.	Панель инструментов и контекстное меню	118

9.3.2.	Вкладка Сигналы	119
9.3.3.	Вкладка Оповещения	121
9.4.	Настройка сигналов.....	122
9.4.1.	Настройка наборов сигналов	124
9.4.1.1.	Создание нового набора сигналов.....	125
9.4.1.2.	Добавление сигнала в набор сигналов	126
9.4.1.3.	Удаление сигнала из набора сигналов.....	129
9.4.1.4.	Копирование набора сигналов	130
9.4.1.5.	Переименование набора сигналов.....	130
9.4.1.6.	Изменение глобальных значений параметров сигнала.....	132
9.4.1.7.	Удаление набора сигналов.....	133
9.4.2.	Назначение наборов сигналов целевым объектам	134
9.4.2.1.	Привязка набора сигналов к целевым объектам.....	134
9.4.2.2.	Открепление набора сигналов от целевого объекта	137
9.4.2.3.	Изменение локальных значений параметров сигнала.....	139
9.5.	Настройка оповещений.....	141
9.5.1.	Настройка наборов оповещений	142
9.5.1.1.	Создание набора оповещений и добавление класса оповещений.....	143
9.5.1.2.	Добавление метода оповещений.....	144
9.5.1.3.	Удаление метода/класса оповещений из набора оповещений.....	146
9.5.1.4.	Копирование набора оповещений	147
9.5.1.5.	Переименование набора оповещений.....	147
9.5.1.6.	Изменение глобальных значений параметров метода / класса оповещений.....	148
9.5.1.7.	Удаление набора оповещений.....	149
9.5.2.	Назначение наборов оповещений целевым объектам	150
9.5.2.1.	Привязка набора оповещений к целевым объектам.....	150
9.5.2.2.	Открепление набора оповещений от целевого объекта	152
9.5.2.3.	Изменение локальных значений параметров оповещений.....	153
9.6.	Комплекс внутренней диагностики ПМ Нувола Топаз.....	154
9.6.1.	Структура комплекса внутренней диагностики	155
9.6.2.	Настройка комплекса внутренней диагностики	156
9.7.	Диспетчер процессов	156
9.7.1.	Основное окно	157
9.7.2.	Создание и изменение системного процесса	157
9.7.3.	Параметры системных процессов	158
9.7.3.1.	Настройка расписания системного процесса.....	160
9.7.4.	Удаление системного процесса	163
9.7.5.	Управление системными процессами	163
9.7.6.	Работа с журналами (лог-файлами) системных процессов	165
9.7.6.1.	Структура окна Журнал.....	165
9.7.6.2.	Просмотр журналов системных процессов.....	166
9.7.6.3.	Сохранение информации из журнала	167
9.7.6.4.	Ротация лог-файла.....	167
10.	Редактор информационных панелей.....	169

10.1.	Вход и выход из Web-приложения.....	170
10.2.	Основное окно.....	171
10.3.	Порядок разработки информационной панели.....	171
10.3.1.	Создание информационной панели	172
10.3.2.	Сохранение информационной панели	172
10.3.2.1.	Окно ввода общих параметров инф. панели.....	173
10.3.3.	Редактирование информационной панели	174
10.4.	Классификация элементов информационной панели.....	174
10.5.	Обзор параметров элементов информационных панелей.....	177
10.5.1.	Раздел Инфо	179
10.5.2.	Раздел Позиция и размер	179
10.5.3.	Раздел Отображение	180
10.5.4.	Раздел Общие	180
10.5.5.	Окно Параметры виджета	182
10.6.	Настройка параметров элементов информационной панели.....	183
10.6.1.	Изменение параметров элементов	183
10.6.2.	Привязка сигналов к элементам	184
10.6.2.1.	Привязка одного сигнала к элементу.....	184
10.6.2.2.	Привязка нескольких сигналов к элементу.....	185
10.6.2.3.	Привязка однотипных сигналов к элементу.....	188
10.6.2.4.	Копирование назначений элементов.....	190
10.6.2.5.	Проверка отображения статуса сигнала на элементе	191
10.6.3.	Настройка ссылки с элемента	191
10.6.3.1.	Составляющие параметра Ссылка	192
10.6.4.	Настройка действий для элемента	192
10.6.5.	Настройка звукового сигнала для элемента	195
10.7.	Инструментарий Редактора информационных панелей.....	196
10.7.1.	Изменение размеров информационной панели	196
10.7.2.	Расположение элементов на информационной панели	197
10.7.2.1.	Добавление элементов на канву.....	197
10.7.2.2.	Использование точек привязки.....	197
10.7.2.3.	Использование направляющих.....	198
10.7.2.4.	Выделение элементов на канве	199
10.7.2.5.	Перемещение и изменение размера элемента	202
10.7.2.6.	Группировка и разгруппировка элементов.....	203
10.7.3.	Управление слоями информационной панели	205
10.7.3.1.	Панель инструментов и контекстное меню окна Система слоёв.....	206
10.7.3.2.	Действия со слоями и элементами	207
10.7.3.3.	Изменение свойств слоев и элементов.....	208
10.7.3.4.	Перемещение слоев и элементов.....	209

11. Администрирование модуля Нувола Координатор.....213

11.1.	Обзор приложений Camunda BPM.....	214
11.2.	Настройка схем бизнес-процессов.....	214
11.3.	Особенности использования скриптов в схемах бизнес-процессов.....	216
11.3.1.	ReportStatus.js	216
11.3.2.	SendMail.js	217
11.3.3.	SendToSAC.js	219
12.	Приложение Координатор.....	221
12.1.	Вход и выход из приложения.....	222
12.2.	Основное окно.....	223
12.2.1.	Заголовок и главное меню	223
12.2.2.	Окно Фильтры	224
12.2.3.	Таблица задач	225
12.2.3.1.	Параметры задачи.....	226
12.2.4.	Окно выполнения бизнес-процесса	226
12.2.5.	Режимы работы (текущие / завершённые)	229
12.3.	Действия администратора.....	229
12.3.1.	Загрузка схемы бизнес-процесса на сервер Координатор	229
13.	Настройка и использование модуля Нувола Аналитика.....	231
13.1.	Запуск сервера Нувола Аналитика.....	232
13.2.	Утилита командной строки.....	234
13.3.	Настройка сервера Нувола Аналитика.....	235
13.4.	Настройка сигналов.....	235
13.5.	Сигнал типа Trend	236
13.5.1.	Параметры сигнала	236
13.6.	Сигнал типа TrendDeviation.....	237
13.6.1.	Параметры сигнала	237
13.7.	Графики при использовании модуля Нувола Аналитика.....	240
13.7.1.	Вычисление предполагаемого диапазона данных	240
13.7.2.	Предсказание перехода сигнала за определённый порог	241
13.7.3.	Обнаружение отклонений в реальном времени	242
14.	Справочники.....	245
14.1.	Термины и определения.....	246
14.2.	Утилиты ПМ Нувола Топаз.....	248
14.2.1.	Настройка сигналов (alertsedit.pl)	248
14.2.2.	Управление резервным копированием конфигурации (backup)	251

14.2.3.	Проверка состояния сервиса ESB (esbClient)	253
14.2.4.	Устранение всплесков (killspike.pl)	253
14.2.5.	Компиляция базы данных (kompile)	254
14.2.6.	Управление деревом конфигурации (konfne)	255
14.2.7.	Конвертация БД из HDB в LDAP и обратно (ldapConfigManager)	257
14.2.8.	Ручная проверка корректности лицензий (license.pl)	258
14.2.9.	Запуск, проверка и завершение работы сервиса баз данных (mysqld)	258
14.2.10.	Настройка оповещений (notifyedit.pl)	259
14.2.11.	Управление системными процессами ПМ Нувола Топаз (overlord.pl)	261
14.2.12.	Проведение нагрузочных тестов дисковой подсистемы (randomwrite.pl)	262
14.2.13.	Создание комплекса внутренней диагностики ПМ Нувола Топаз (selfMonitoringConfig.pl)	263
14.2.14.	Проверка целостности кода (sign)	264
14.2.15.	Просмотр значений параметров узла дерева конфигурации (target.pl)	266
14.2.16.	Проверка возможности доступа по SNMP (test_access.pl)	267
14.2.17.	Асинхронный опрос по ping (X3PingAsync.pl)	267
14.3.	Системные процессы ПМ Нувола Топаз.....	267
14.3.1.	Стандартные параметры системных процессов	268
14.3.2.	Формирование сигналов (alerts)	268
14.3.3.	Хранение данных (dataserver)	271
14.3.4.	Выборка данных для подготовки XML-фрагментов (de_providerd)	272
14.3.5.	Сервисная шина предприятия (esb)	272
14.3.6.	Отображение изменений в реальном времени (jrtgd)	275
14.3.7.	Сбор данных (kollector)	275
14.3.8.	Удаленная настройка ПМ Нувола Топаз (konfigd)	277
14.3.9.	Проверка лицензии (licdaemon)	277
14.3.10.	Мониторинг системных процессов ПМ Нувола Топаз (monit)	278
14.3.11.	Формирование оповещений (notifs)	278
14.3.12.	Подготовка отчетов (querymaker)	278
14.3.12.1.	pinger	279
14.3.13.	API для Web-интерфейса (rest.pl)	279
14.4.	Типы сигналов.....	279
14.4.1.	Параметры сигнала типа Gauge	279
14.4.2.	Параметры сигнала типа InvertGauge	281
14.4.3.	Параметры сигнала типа SmartGauge	283
14.4.4.	Параметры сигнала типа Availability	285
14.5.	Статусы сигналов	286
14.6.	Классы оповещений.....	287
14.6.1.	Параметры класса оповещений Simple	287
14.7.	Типы оповещений.....	288

14.7.1.	Параметры метода оповещений типа Email	288
14.7.2.	Параметры метода оповещений типа EsbNotify	289
14.7.3.	Параметры метода оповещений типа Exec	290
14.7.4.	Параметры метода оповещений типа Log	290
14.7.5.	Параметры метода оповещений типа MYSQL	291
14.7.6.	Параметры метода оповещений типа SSH	292
14.7.7.	Параметры метода оповещений типа SMS	292
14.7.8.	Параметры метода оповещений типа Syslog	293
14.8.	Элементы информационных панелей.....	294
14.8.1.	Простые элементы для рисования линий	294
14.8.1.1.	Кривая линия	294
14.8.1.2.	Прямоугольные ломаные линии.....	295
14.8.1.3.	Параметры элементов	295
14.8.2.	Простой элемент Прямоугольник	297
14.8.3.	Простой элемент Эллипс	299
14.8.4.	Простой элемент Текст	301
14.8.5.	Простой элемент Изображение	304
14.8.6.	Простой элемент Дата и время	305
14.8.7.	Виджет Бегущий текст	308
14.8.8.	Виджет Объект	308
14.8.9.	Виджет Лампочка	309
14.8.10.	Виджет Шкала	310
14.8.11.	Виджет Измерительный прибор	313
14.8.12.	Виджет Многофункциональный измерительный прибор (SmartGauge)	313
14.8.12.1.	Панель инструментов окна настройки виджета	314
14.8.12.2.	Глобальные настройки виджета.....	315
14.8.12.3.	Настройки компонент виджета.....	316
14.8.13.	Виджет Стекло	323
14.8.14.	Виджет Список	325
14.8.15.	Виджет Список с лампочками	327
14.8.16.	Виджет График	329
14.8.17.	Виджет Динамический график	329
14.8.18.	Виджет Таблица	332
14.8.19.	Виджет Диаграмма	332
14.8.20.	Виджет Круговая диаграмма	334
14.8.21.	Виджет Рамочный контейнер	336
14.8.22.	Виджет Табло событий	338
14.8.22.1.	Специфичные параметры виджета.....	338
14.8.22.2.	Настройка табло событий.....	340
14.9.	Предопределенные переменные (теги) для текстовых строк.....	341
14.10.	Функции для настройки действий для элементов информационной панели.....	342

1. Введение

1.1. Общие сведения о ПК Нувола

Программный комплекс «Нувола» представляет собой набор взаимодействующих программных модулей, согласованных по функциям и форматам; имеющих единообразные, точно определенные программные интерфейсы; и составляющих полноценное средство для решения комплексных задач (оперативный мониторинг, контроль и управление технологическими и бизнес-процессами, сервисами, внешними информационными системами, инфраструктурой предприятия). Модули обеспечивают сбор, обработку и хранение информации о контролируемых объектах, представление обобщенного состояния объектов, оповещение персонала о возникших проблемах и нарушениях.

1.1.1. Назначение ПК Нувола

ПК Нувола реализует следующие функции:

- ❖ формирование единого информационного пространства, позволяющего в режиме реального времени и динамике видеть достоверную картину работы оборудования предприятия, оценивать результативность этой работы и принимать оперативные решения по устранению ситуаций, связанных с авариями и простоем оборудования, регулировать грузопотоки;
- ❖ автоматизация и информационной поддержка производственных процессов и планирования деятельности предприятия;
- ❖ обеспечение безопасности данных;
- ❖ мониторинг и анализ ключевых параметров объектов контроля как для оперативного проактивного контроля в режиме реального времени, так и для анализа исторических данных для целей прогнозирования, выявления взаимосвязей и уточнения параметров использования ресурсов для долгосрочного планирования.
- ❖ получение данных об инфраструктуре и технологии работы предприятия;
- ❖ учет загруженности объектов инфраструктуры предприятия;
- ❖ обобщенное представление состояний, нарушений и сигнализация о деградации состояния контролируемых процессов в соответствии с определенной логической структурой объектов;
- ❖ отображение качественных параметров услуг/сервисов, предоставляемых предприятием;
- ❖ предоставление исторических отчетов о параметрах, характеризующих состояние контролируемых систем и качество предоставления услуг/сервисов;
- ❖ контроль и сигнализация на экранах обобщенного состояния центра контроля нарушений на объектах инфраструктуры;
- ❖ оперативное оповещение должностных лиц и персонала предприятия о произошедших нарушениях функционирования контролируемых объектов;
- ❖ графическое отображение состояния бизнес-процессов предприятия, предоставляет возможность в реальном времени контролировать операционные риски предприятия;
- ❖ отображение информации в удобной для пользователя форме, т.е. визуализирует бизнес-процессы предприятия и отражает влияние на них элементов сети.

Областью применения ПК Нувола являются объекты железнодорожной инфраструктуры предприятия, информационно-телекоммуникационные ресурсы, технологические процессы обработки служебной информации; объекты и технологические процессы транспортировки и отгрузки продукции.

Основными пользователями программного комплекса являются следующие категории персонала предприятия:

- ❖ сотрудники, в задачи которых входит контроль состояния производственной инфраструктуры предприятия в зоне ответственности ПК Нувола, координация и проведение работ по устранению неисправностей;

- ❖ руководители разного уровня, которым для выполнения должностных обязанностей необходимо своевременное получение информации о работе объектов инфраструктуры предприятия.

1.1.2. Пользовательские роли и требования к квалификации персонала

В ПК Нувола понятие *пользовательской роли* имеет условный характер, т.к. спектр функциональных возможностей, необходимых разным категориям пользователей, зависит от распределения должностных обязанностей между работниками предприятия. ПК Нувола обладает гибкой системой разграничения прав, позволяющей настроить доступ пользователей к информации и функционалу в соответствии с требованиями проектного решения.

Пользователи ПК Нувола должны иметь квалификацию инженера и обладать знаниями в области сетевых технологий.

Можно выделить три принципиальные пользовательские роли: **администратор** (или **администратор root**), **локальный администратор** и **конечный пользователь** (далее – **пользователь**).

Администратор (пользователь root)

Функциональное назначение – управление и общая настройка ПК Нувола, обеспечение его штатного функционирования. Администратор root использует в работе командную строку и специализированный функционал модулей ПК Нувола.

Требования к квалификации:

- ❖ знание ОС UNIX (Linux) на уровне администратора системы, сетевого администратора, владение командной строкой;
- ❖ глубокое знание сетевых технологий: хорошее понимание работы TCP/IP, маршрутизируемых и коммутируемых сетей, опыт поиска сетевых неисправностей, понимание принципов работы межсетевых экранов (ACL, NAT, PAT, AAA);
- ❖ понимание принципов работы основных сетевых сервисов: IMAP, POP3, SMTP, RADIUS, DNS, FTP, HTTP, SQL, VNC, DHCP, ICMP;
- ❖ понимание технологий и принципов построения систем сетевого управления и мониторинга: SNMP, SNMP-trap, WMI, syslog, NetFlow, Small Services.

Локальный администратор

Функциональное назначение – настройка сегмента инфраструктуры, выделенного администратором, посредством Web-интерфейса.

Требования к квалификации:

- ❖ знание операционных систем MS Windows в объеме, необходимом для пользователя системы;
- ❖ знание основ сетевых технологий.

Конечный пользователь

Функциональное назначение – мониторинг сегмента инфраструктуры предприятия через Web-интерфейс.

Требования к квалификации – аналогично локальному администратору.

1.1.3. Требования к техническим средствам и программному обеспечению

Требования к АРМ

Автоматизированные рабочие места (АРМ) персонала, использующего ПК Нувол в своей деятельности, должны оборудоваться в соответствии с Санитарными Правилами и Нормами 2.2.2/2.4.1340-03 – Гигиенические требования к видеодисплейным терминалам, персональным электронно-вычислительным машинам и организации работ.

Технические средства АРМ обеспечиваются в рамках стандартных процедур организации рабочих мест сотрудников, используемых на предприятии. В качестве АРМ используется компьютер с операционной системой Microsoft Windows 7 (64 бит) и выше.

Для эксплуатации ПК Нувол необходимо одно АРМ администратора. Количество АРМ локального администратора и конечного пользователя определяется на этапе разработки проектного решения.

Оснащение АРМ зависит от [функциональной пользовательской роли](#)^[15], для которой оно предназначено.

Рекомендуемое аппаратное обеспечение АРМ для администратора, локальных администраторов и конечных пользователей:

- ❖ CPU 2,6 ГГц (Intel I3 или аналогичный AMD);
- ❖ оперативная память – 8 Гб или больше (свободная память для конечных пользователей – не менее 2 Гб, для администраторов – не менее 3 Гб);
- ❖ свободный объём памяти на жестком диске – 20 Гб или больше;
- ❖ привод DVD-RW (не обязательно);
- ❖ USB-интерфейсы (обязательно);
- ❖ сетевой интерфейс FastEthernet 1000 Мбит/с;
- ❖ ЖК-монитор 19" с разрешением Full HD или более;
- ❖ клавиатура стандартная USB;
- ❖ 3- или 2-кнопочная мышь с колесом прокрутки, USB.

Рекомендуемое программное обеспечение АРМ администратора:

- ❖ web-браузер Chrome (версия 78 и выше);
- ❖ PuTTY и/или telnet;
- ❖ WinSCP и/или FTP.

Рекомендуемое программное обеспечение АРМ локального администратора / конечного пользователя:

- ❖ web-браузер Chrome (версия 78 и выше), Microsoft IE (версия 11 и выше), Edge (версия 42 и выше).

Требования к конфигурации сервера Нувол

Конфигурация серверного оборудования, необходимого для функционирования ПК Нувол, зависит от целей проектного решения, его планируемой мощности и т.п. Поэтому технические требования к используемым серверам согласовываются с Заказчиком.

Сервер Нувол – сервер, на котором установлен ПМ **Нувол Топаз**. На этот сервер (в соответствии с согласованными техническими условиями проектного решения) также могут быть установлены серверные компоненты других программных

1.1.4. Компоненты ПК Нувола

Далее перечислены программные модули ПК Нувола и кратко описаны их назначения. Комплектация, поставляемая Заказчику, зависит от проектного решения.

Модуль **Нувола Топаз** отвечает за сбор данных с объектов контроля, создание отчетов, отправку оповещений о возникших проблемах в инфраструктуре предприятия, а также обеспечивает возможность удалённого управления и настройкой ПК Нувола с помощью Web-интерфейса.

Модуль **Нувола ESB** – компонент для построения событийно-управляемой архитектуры на базе шины информационного взаимодействия (ESB). Модуль реализует архитектуру EDA, обеспечивает унифицированный обмен данными между программными компонентами – сервисами (независимо от платформы, языка программирования) и представляет единую точку управления объектами и компонентами системы, а также реализует информационно-телекоммуникационное взаимодействие со сторонними системами. Модуль **Нувола ESB** интегрирован в модуль **Нувола Топаз** и устанавливается из того же инсталляционного файла, что и модуль **Нувола Топаз**.

Модуль **Нувола Виста** позволяет настраивать и просматривать информационные панели, обеспечивающие наглядное графическое отображение состояния контролируемой инфраструктуры предприятия.

Модуль **Нувола Координатор** предназначен для автоматизации контроля за выполнением технологических процессов предприятия в соответствии с их формализованными схемами.

Модуль **Нувола Логистика** предназначен для контроля подвижных составов посредством отображения и контроля на цифровом паспорте станции, который включает в себя гео-сервер. Модуль рассчитывает оптимальную скорость движения составов, генерирует предупреждения об изменении скоростного режима, фиксирует местоположение подвижного состава, направление движения, позволяет управлять логистикой подвижного состава, составлять задания на выполнение в автоматическом, автоматизированном и ручном режимах.

Модуль **Нувола Аналитика** предназначен для уточнения удельных параметров использования ресурсов на основании анализа накопленных исторических данных мониторинга, предоставляемых подсистемой мониторинга эффективности, в которой ведется расчет эффективности согласно формализованным методикам расчетов.

1.2. Назначение и архитектура модуля Нувола Топаз

Модуль **Нувола Топаз** отвечает за сбор данных с объектов контроля, создание отчетов, отправку оповещений о возникших проблемах в инфраструктуре предприятия, а также обеспечивает возможность удалённого управления и настройкой ПК Нувола с помощью Web-интерфейса.

ПМ **Нувола Топаз** построен на принципах событийно-ориентированной архитектуры EDA (Event-Driven Architecture), реализованной модулем **Нувола ESB**.

[Архитектура EDA](#)^[18] заключается во взаимодействии всех приложений через единую точку, которая, при необходимости, обеспечивает транзакции, преобразование данных, сохранность обращений. Суть концепции – обмен событиями, когда один программный компонент порождает сообщение, другой его перехватывает и обрабатывает, при необходимости вызывая для этого третьи компоненты. Данный подход обеспечивает большую гибкость и простоту масштабирования.

Компоненты ПМ **Нувола Топаз**:

- ❖ **ESB** ([esb](#)^[272]) – сервисная шина предприятия, реализованной модулем **Нувола ESB** и обеспечивающая связь компонент ПМ **Нувола Топаз**.

1. Введение

- ❖ **База данных производительности** – база данных, обеспечивающая хранение и отображение динамических параметров, полученных от объектов контроля. Для организации хранения собранных данных используется специальный системный процесс – `dataserver` (см. ниже).
- ❖ **База данных о состоянии сигналов** – база данных, содержащая текущие значения и статусы сигналов, рассчитанные ПМ **Нуволы Топаз**, а также историю их изменений. В качестве такой базы данных используется MySQL.
- ❖ **Системный процесс хранения данных** (`dataserver`^[271]) – системный процесс, обеспечивающий организацию хранения данных на локальном или удаленном сервере.
- ❖ **База данных конфигурации** включает в себя дерево конфигурации, шаблоны объектов, настройки сигналов и оповещений, информацию о системных процессах ПМ **Нуволы Топаз**. Для упрощения схемы на ней показаны не все компоненты базы данных конфигурации.
- ❖ **Дерево конфигурации** – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления. Дерево конфигурации может быть реализовано средствами базы данных LDAP.
- ❖ **Шаблоны объектов** – библиотека [шаблонов объектов](#)^[25], поддерживаемых ПК Нуволы.
- ❖ **Системный процесс управления конфигурацией** (`konfigd`^[271]) – системный процесс, обеспечивающий возможность удалённого управления конфигурацией ПК Нуволы.
- ❖ **Системные процессы сбора данных (коллекторы)** – системные процессы, отвечающие за сбор данных от объектов контроля и сохранение их в базе данных (по умолчанию используется системный процесс `kollector`^[273]).
- ❖ **Системный процесс обработки сигналов** (`alerts`^[268]) – системный процесс формирования сообщений о произошедших событиях (сбоях в сети, изменениях состояний объектов контроля).
- ❖ **Системный процесс формирования оповещений** (`notifs`^[278]) – системный процесс формирования и отправки оповещений о произошедших событиях.
- ❖ **Системные процессы обработки инф.панелей** (`de_builderd`, `de_providerd`^[272], `de_svg`, `de_svg_js`) – системные процессы, отвечающие за формирование информационных панелей – объектов Web-интерфейса, обеспечивающих наглядное графическое отображение состояния контролируемой инфраструктуры. Указанные системные процессы являются компонентами модуля **Нуволы Виста**.
- ❖ **Системные процессы генерации отчетов** (`querymaker`^[278], `querysender`) – системные процессы, отвечающие за формирования отчётов.

1.3. Назначение и архитектура модуля Нуволы ESB

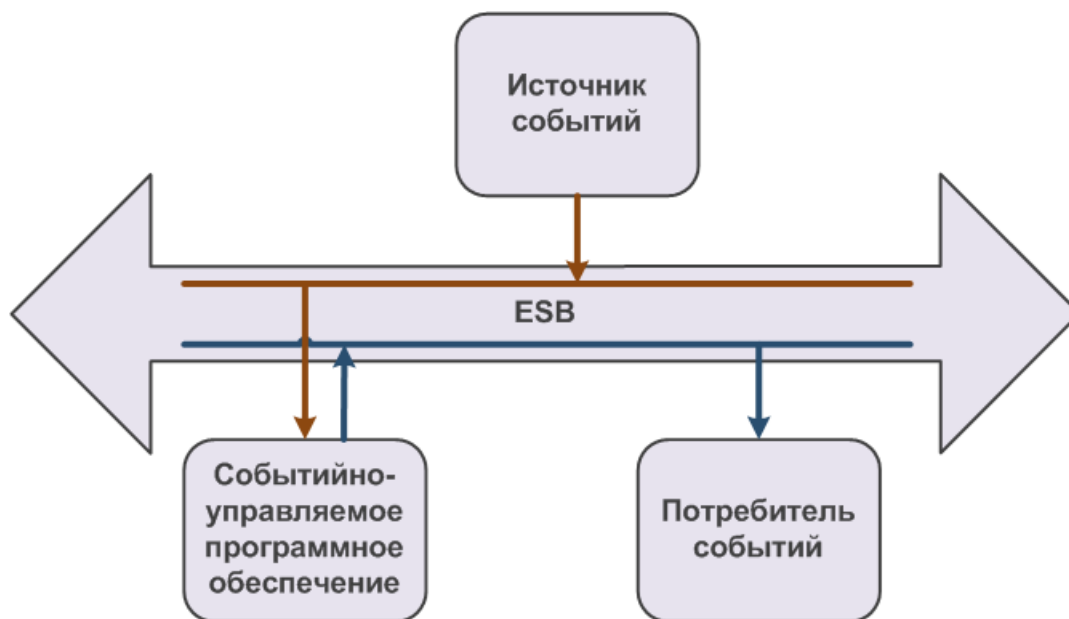
В модуле **Нуволы Топаз** используются принципы событийно-ориентированной архитектуры (EDA – Event-Driven Architecture), реализованные модулем **Нуволы ESB**. Приложение, разработанное на основе этой архитектуры, состоит из компонентов, которые сообщаются и исполняются путем отправки и приема событий. Суть концепции – обмен событиями, когда один программный компонент порождает сообщение, другой его перехватывает и обрабатывает, при необходимости вызывая для этого третьи компоненты. В данном контексте под событиями понимаются неизменяемые записи произошедшего изменения состояния или выполненного действия.

EDA предполагает управление посредством асинхронного шаблона проектирования PUB/SUB, который определяет дисциплину передачи сообщений и включает в себя Источник и Подписчик. EDA состоит из нескольких основных компонентов: источников событий, потребителей событий и программного обеспечения обработки событий. На общей схеме EDA показаны следующие компоненты:

- ❖ **Источник событий** – Источник, который инициирует события путем обработки входящих сообщений, присвоения им признака класса и передачи событий определенного класса в ESB.
- ❖ **Потребитель событий** – Подписчик, отслеживающий определенные события, подписываясь на определенный класс сообщений.

1.3. Назначение и архитектура модуля Нуволы ESB

- ❖ **Событийно-управляемое программное обеспечение** (сервис ESB) – компонент, в котором источники событий публикуют события, а потребители регистрируются на их получение.
- ❖ **ESB** (Enterprise Service Bus) – сервисная шина предприятия, обеспечивающая информационное взаимодействие сервисов программного комплекса.



Архитектура ESB заключается во взаимодействии всех приложений через единую точку, которая, при необходимости, обеспечивает транзакции, преобразование данных, сохранность обращений. Данный подход обеспечивает большую гибкость, простоту масштабирования и переноса: при замене одного приложения, подключенного к шине, нет необходимости перенастраивать остальные.

Основные характеристики ESB:

- ❖ поддержка синхронного и асинхронного способа вызова сервисов;
- ❖ использование защищённого транспорта, с гарантированной доставкой сообщений, поддерживающего транзакционную модель;
- ❖ статическая и алгоритмическая маршрутизация сообщений;
- ❖ доступ к данным из сторонних информационных систем с помощью готовых или специально разработанных адаптеров;
- ❖ обработка и преобразование сообщений.

В модуле **Нуволы Топаз** за функционирование ESB отвечает системный процесс [esb](#)^[272].

При адресации сообщений системный процесс `esb` использует следующие схемы: **Address**, **Round-robin** и **Broadcast**. Схема **Address** позволяет обеспечить адресную передачу данных. Схема **Round-robin** позволяет передавать данные свободному клиенту из заданной группы. Схема **Broadcast** позволяет передавать сообщения нескольким клиентам. Каждый подключившийся клиент должен прислать системному процессу `esb` сообщение с параметрами подписки. В соответствии с заданными параметрами для клиентов создаются очереди сообщений. Если в сообщении от отправляющей стороны указана несуществующая очередь, то данное сообщение не будет доставлено.

1.4. Назначение и архитектура модуля Нуволы Виста

Модуль **Нуволы Виста** позволяет настраивать и просматривать информационные панели, обеспечивающие наглядное графическое отображение состояния контролируемой инфраструктуры предприятия.

1. Введение

Модуль **Нуволa Вистa** предназначен для обеспечения интерактивного визуального взаимодействия пользователей с системой средствами тонкого Web-клиента. Обеспечивает непрерывный визуальный контроль состояния объектов контроля в режиме реального времени и оперативное управление объектами контроля посредством графического интерфейса пользователя. Взаимодействие с функциональными и сервисными подсистемами осуществляется средствами REST-протокола при взаимодействии в режиме запрос/ответ и средствами протокола Websockets при передаче потоков данных в режиме реального времени.

Модуль **Нуволa Вистa** включает в себя следующие компоненты:

- ❖ библиотеку информационных панелей;
- ❖ базу данных (SQLite или MySQL) хранит статусы контролируемых объектов. Актуальность данных поддерживается модулем **Нуволa Топаз**;
- ❖ системные процессы, отвечающие за формирование информационных панелей;
- ❖ Приложение **Редактор информационных панелей** позволяет создавать информационные панели в формате SVG и обеспечивает их хранение сервере Нуволa.

Компоненты модуля **Нуволa Вистa** интегрированы в состав других модулей:

- ❖ клиентская часть: Web-приложение [Редактор информационных панелей](#)¹⁷⁰ – является структурной компонентой Web-интерфейса;
- ❖ серверная часть обеспечивается средствами ПМ **Нуволa Топаз**.

1.5. Назначение и архитектура модуля Нуволa Аналитика

Модуль **Нуволa Аналитика** предназначен для уточнения удельных параметров использования ресурсов на основании анализа накопленных исторических данных мониторинга, поставляемых подсистемой мониторинга эффективности, в которой ведется расчет эффективности согласно формализованным методикам расчетов.

В качестве входных данных модуль **Нуволa Аналитика** использует массив значений параметров, собранных с целевых объектов системными процессами ПМ **Нуволa Топаз**. По этим данным модуль строит прогноз поведения параметров, а именно: общую тенденцию изменения значений параметров, сезонные колебания (часовые, суточные, месячные, годовые и пр.) и в зависимости от этих изменений устанавливает доверительные интервалы возможных колебаний значений.

В состав программного модуля **Нуволa Аналитика** входят следующие компоненты:

- ❖ сервер **Нуволa Аналитика** – серверный компонент, выполняющий математические расчеты для предсказания продолжений временных рядов, выявления аномалий и сравнения сигналов;
- ❖ набор сценариев, написанных на языке perl, дополняющих программный модуль **Нуволa Топаз** и использующих возможности клиентской библиотеки и сервера для задач мониторинга.

1.6. Назначение и архитектура модуля Нуволa Координатор

Модуль **Нуволa Координатор** предназначен для автоматизации контроля за выполнением технологических процессов предприятия в соответствии с их формализованными схемами.

Подсистема управления процессами состоит из модулей:

- ❖ построения моделей процессов;
- ❖ управления исполнением процессов в нотации;

- ❖ средств разработки автоматизации исполнения задач процессов.

Инструментом пользователей для выполнения задач является Web-приложение **Координатор**.

ПМ **Нувола Координатор** реализован с использованием компонент открытой платформы Camunda, предназначенной для автоматизации бизнес-процессов и документооборота:

- ❖ **Camunda BPM** – программная платформа, устанавливаемая на сервер с Apache Tomcat;
- ❖ **Camunda Modeler** – приложение для разработки схем бизнес-процессов.

См. документацию на сайте <http://docs.camunda.org/manual/latest/>.

Платформа Camunda BPM использует вспомогательную базу данных, которая может быть установлена как на сервер Нувола (этот вариант показан на схеме), так и на другой сервер. Вид СУБД (MySQL или др.) и вариант размещения базы данных определяется проектным решением.

1.7. Назначение модуля Нувола Логистика

Модуль **Нувола Логистика** предназначен для контроля подвижных составов посредством отображения и контроля на цифровом паспорте станции, который включает в себя гео-сервер. Модуль рассчитывает оптимальную скорость движения составов, генерирует предупреждения об изменении скоростного режима, фиксирует местоположение подвижного состава, направление движения, позволяет управлять логистикой подвижного состава, составлять задания на выполнение в автоматическом, автоматизированном и ручном режимах.

Модуль **Нувола Логистика** позволяет:

- ❖ в режиме реального времени и динамике получать достоверную картину работы оборудования, оценивать ее результативность и принимать оперативные решения по устранению ситуаций, связанных с авариями и простоем оборудования, регулировать грузопотоки;
- ❖ осуществлять мониторинг и анализ ключевых параметров (КП) объектов контроля железнодорожного транспорта как для оперативного проактивного контроля в режиме реального времени, так и для анализа исторических данных для целей прогнозирования, выявления взаимосвязей и уточнения параметров использования ресурсов для долгосрочного планирования.

Основными целями модуля **Нувола Логистика** являются:

- ❖ обеспечение транспортировки требуемого объема груза;
- ❖ ускорение оборота вагонов в процессах отгрузки готовой продукции;
- ❖ увеличение производительности работы железнодорожного транспорта, снижение себестоимости эксплуатации.

2.

Понятия и концепции

2. Понятия и концепции

В данном разделе введены термины и определения, используемые в ПК Нувола, описаны принципы его функционирования.

2.1. Дерево конфигурации и его объекты

Дерево конфигурации – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления.

Объект дерева конфигурации – компонента дерева конфигурации, созданная администратором или локальным администратором на основе *шаблона объекта* и выполняющая конкретную задачу в реализации функционального назначения ПК Нувола. Классификация объектов дерева конфигурации приведена в таблице.

Объекты дерева конфигурации	Объекты контроля	Устройства
		Другие (базы данных, ПО и т.п.)
	Конфигурационные инструменты	Папки
		Информационные панели
		Отчеты
		Сводные графики
		Ссылки

Объект контроля – любая компонента или другой сервисный актив, который необходимо контролировать для того, чтобы предоставлять ИТ-услугу и/или услугу мониторинга. В широком понимании, объектом контроля может быть, например, ИТ инфраструктура, отдельные устройства, предприятие и его технологические процессы, внешние ИТ системы, программное обеспечение. В дереве конфигурации в качестве объектов контроля регистрируются физические устройства (компьютеры, серверы, маршрутизаторы, коммутаторы, принтеры, периферийные устройства и т.п.), виртуальные объекты (программное обеспечение, базы данных и др.).

Конфигурационный инструмент – компонента дерева конфигурации, посредством которого осуществляется структурирование и хранение информации об объектах контроля, визуализация результатов обработки данных.

В корне дерева конфигурации размещена *учетная запись* пользователя. Её настройки определяют структуру дерева конфигурации, доступную конечному пользователю ПК Нувола, и набор разрешенных ему действий. Детально принципы разграничения прав пользователей описаны в разделе [Пользователи](#)^[36].

Структура объектов контроля в дереве конфигурации

Объекты контроля состоят из целевых объектов.

Целевой объект – минимальная неделимая компонента (логическая или физическая) объекта контроля, которая является источником данных, для последующего формирования обобщенного состояния объекта контроля.

Например, для компьютера целевыми объектами являются процессор, жесткие диски, сетевые интерфейсы. В дереве конфигурации целевые объекты отображаются как дочерние узлы объектов.

Тип целевого объекта – атрибут, классифицирующий целевой объект и определяющий его функциональную роль в составе объекта дерева конфигурации.

Источник данных представляет собой характеристику целевого объекта.

Для устройств источниками данных могут быть, например, переменные SNMP.

Модуль **Нуволы Топаз**, опрашивая устройства, получает от источников данных информацию о состоянии целевых объектов.

Конфигурационные инструменты

Кратко рассмотрим конфигурационные инструменты, реализованные в ПК Нуволы.

Папка (Folder) – это конфигурационный инструмент, позволяющий структурировать объекты контроля. Группируя объекты в папки, администратор создает структуру дерева конфигурации, отвечающую требованиям и задачам проектного решения.

Отчёт (Report) – конфигурационный инструмент, обеспечивающий форматированное представление статистической информации о состоянии объектов контроля или целевых объектов определенного типа. Данные для отчета отбираются по критериям, заданным администратором при настройке отчета, например, интерфейсы с наибольшим объемом трафика.

Аналитический отчёт (Analysis Report) – конфигурационный инструмент, обеспечивающий форматированное представление аналитической информации о состоянии объектов контроля. Аналитические отчеты доступны для настройки и просмотра при установленном модуле **Нуволы Отчёты**.

Сводный график (Total) – это конфигурационный инструмент, позволяющий представить в Web-интерфейсе в графической форме обобщенную информацию об изменении параметров целевых объектов, отобранных по заданным правилам.

Информационная панель (Dashboard) – конфигурационный инструмент, обеспечивающий в режиме реального времени наглядное графическое web-представление информации о состоянии объектов контроля, данных об обобщенном состоянии инфраструктуры предприятия или ее сегмента, а также предоставляемых им услуг. Информационная панель позволяет интегрировать данные из производственных и информационных систем и является средством для мониторинга, анализа и визуализации информации и управления организационными изменениями.

Рабочий стол (Desktop) – это конфигурационный инструмент, позволяющий в соответствии с задачами пользователя скомпоновать на одной экранной форме элементы (объекты) визуализации информации о состоянии объектов контроля. Экранная форма может состоять из одной или нескольких секций. Конфигурация экранной формы и содержимое секций настраивается администратором в зависимости от задач конкретного пользователя.

Ссылка (Link) – это конфигурационный инструмент, который указывает на другой узел дерева конфигурации – неконцевой (т.е. на объект дерева конфигурации, например на папку или устройство) или концевой (например, на целевой объект).

2.1.1. Шаблоны объектов

Как было отмечено выше, объекты контроля и конфигурационные инструменты создаются в дереве конфигурации на основе шаблонов. Шаблон объекта – это внутренний шаблон ПК Нуволы, определяющий свойства объекта дерева конфигурации.

Шаблон объекта контроля – это набор правил, определяющих логику сбора данных с объектов контроля определенного типа; также шаблон может содержать логику первичной обработки полученной информации и ее визуализации. Шаблон объекта контроля описывает его возможные целевые объекты и источники данных, содержит набор параметров, которые задаются при добавлении соответствующего устройства в дерево конфигурации.

2. Понятия и концепции

Каталог шаблонов объектов контроля – библиотека преднастроенных шаблонов объектов, или модулей информационно-технического взаимодействия. Шаблоны объектов хранятся в директории `/usr/local/lancelot/lib/Devices`.

Структура имени шаблона объекта

В общем случае имя шаблона объекта состоит из трех частей: `<Device Name>=Devices::<Type>::<Module>`, где

- ❖ `Devices` – это фиксированная строка;
- ❖ `Type` – категория объекта:
 - `Access` – доступ;
 - `Analysis` – анализ;
 - `Computers` – компьютеры;
 - `Firewall` – брандмауэры;
 - `Mailers` – программа работы с электронной почтой;
 - `PacketShapers` – сортировщики пакетов;
 - `Printers` – принтеры;
 - `Proxies` – прокси;
 - `Routers` – маршрутизаторы;
 - `Satellite` – спутники;
 - `Storage` – хранилища;
 - `Switches` – коммутаторы;
 - `UPS` – источники бесперебойного питания;
 - `Virtual` – виртуальные объекты;
- ❖ `Module` – модуль Perl, который реализует шаблон объекта.

Примечание. Файл модуля Perl имеет имя: `<Module>.pm`. Например, файл модуля для шаблона объекта `Devices::Routers::Cisco` расположен в `/usr/local/lancelot/lib/Devices/Routers/Cisco.pm`.

2.2. Системные процессы

Функционирование ПМ **Нувол** **Топаз** и ПК Нувол в целом обеспечивают системные процессы, отвечающие за выполнение различных задач: сбор данных с объектов контроля и обработку полученной информации, построение информационных панелей, создание отчетов, поддержку работы пользовательского Web-интерфейса, контроль целостности конфигурации, интеграцию компонент ПК Нувол и др.

Системный процесс – это процесс (daemon) на сервере Нувол, запускаемый с определенными значениями параметров. **Процесс** (daemon) является встроенным объектом ПМ **Нувол** **Топаз**. Системные процессы создаются и настраиваются администратором в зависимости от задач, решаемых с помощью ПК Нувол.

При создании системного процесса администратор указывает процесс (демон), вводит имя системного процесса и задает значения параметров для запуска процесса. Имя системного процесса может как совпадать с именем процесса, так и отличаться от него. Запуск нескольких системного процесса, в основе которых лежит один и тот же процесс (демон), фактически означает запуск нескольких экземпляров процесса. Информация о настроенных системных процессах хранится в базе данных конфигурации.

При инсталляции ПМ **Нувола Топаз** создается и запускается первоначальный набор системных процессов.

2.3. Сигналы и оповещения

ПК Нувола предоставляет следующие функциональные возможности:

- ❖ формирование оповещений о проблемах с использованием различных методов. Например, одно и то же оповещение может быть отправлено по электронной почте, SMS, а также записано в лог для последующего анализа;
- ❖ отправка оповещений о проблемах на каждом целевом объекте для разных групп пользователей. Например, группе администраторов необходимо получать оповещения от всех сетевых интерфейсов, но отдельному оператору достаточно получать оповещения только о проблемах непосредственно на его линиях;
- ❖ контроль времени отправки сообщения. Например, можно настроить повторную отправку оповещения через 4 часа после первой отправки, если проблема так и не была решена.

2.3.1. Основные определения

В настоящем разделе назначение сущностей, которыми оперирует системный процесс `alerts`.

Сигнал и набор сигналов

Сигнал – правило обработки значений параметров целевых объектов и определения статуса сигнала.

Параметры целевых объектов, которые анализируются правилом, также называются *источниками данных*. Ключевым атрибутом сигнала является тип сигнала.

Тип сигнала – алгоритм вычисления статуса сигнала, ограничивающий множество возможных статусов сигнала.

Значение сигнала – числовое значение, полученное в результате обработки значений параметров целевых объектов за определенный интервал времени (например: среднее значение, максимальное или минимальное значение). Значение сигнала используется для вычисления статуса сигнала.

Статус сигнала – характеристика сигнала, отражающая наличие проблемы на целевом объекте и уровень ее критичности. Статус сигнала идентифицируется цветом и числовой константой.

См. также справочник [Статусы сигналов](#) ^[266].

Событие – изменение статуса сигнала.

Длительность события – интервал времени между двумя последовательными изменениями статуса сигнала.

При настройке сигнала задаются:

- ❖ период времени и функция для определения **значения сигнала**;
- ❖ параметры алгоритма для вычисления **статуса сигнала** (пороговые значения).

Набор сигналов – совокупность сигналов, которые применимы к источникам данных, принадлежащим целевым объектам определенного типа. Количество сигналов в наборе ограничивается количеством источников данных, поддерживаемых ПК Нувола для выбранного типа целевого объекта.

Таким образом, для создания набора сигналов необходимо ввести его общие параметры (наименование и тип целевого объекта) и составить множество сигналов.

2. Понятия и концепции

Локальный администратор привязывает наборы сигналов к целевым объектам. К конкретному целевому объекту можно привязать только один набор сигналов. Однако один и тот же набор сигналов можно привязать к нескольким целевым объектам.

Класс оповещений, метод оповещений, набор оповещений

Оповещение – это уведомление (пользователей) о произошедшем событии (например, сбое, изменении статуса параметра объекта контроля) и /или фиксирование таких фактов в системе (например, логирование).

Класс оповещений – совокупность параметров, определяющих условия, при возникновении которых формируются оповещения. В частности, в классе оповещения указывается, по каким сигналам и при каких статусах сигналов формируется оповещение, а также алгоритм формирования повторных оповещений.

Метод оповещений – совокупность параметров, определяющих способ формирования оповещений. Метод оповещений содержит ключевой параметр **тип оповещения** и параметры, конкретизирующие выполняемое действие, например: тип оповещения «отправка сообщения по электронной почте», адреса получателей сообщений, сетевой адрес почтового сервера.

Тип оповещения – вид уведомления, например, отправка сообщения по электронной почте или SMS, создание записи в лог-файле или базе данных, передача сообщения о событии в стороннюю систему, выполнение иной программной процедуры (скрипта). Тип оповещения определяет способ формирования и доставки оповещений.

Набор оповещений – совокупность класса и метода (или нескольких методов) оповещений. Набор оповещений определяет, при каких условиях должно выполняться то или иное действие.

Набор оповещений обязательно должен содержать только один класс оповещений. В состав одного набора могут быть включены несколько методов оповещений, причем в любой комбинации способов создания и отправки. Например, набор может содержать 2 метода для отправки оповещения по электронной почте, 2 метода для отправки sms и один метод для записи оповещения в лог.

Локальный администратор привязывает наборы оповещений к целевым объектам, также как и наборы сигналов. Однако, в отличие от наборов сигналов, к конкретному целевому объекту можно привязать несколько наборов оповещений. Один и тот же набор оповещений можно привязать к нескольким целевым объектам.

2.3.2. Глобальные и локальные значения параметров

При настройке набора сигналов и набора оповещений для каждого его структурного элемента (т.е. для сигнала, класса оповещений, метода оповещений) администратор задает **глобальные значения** параметров.

При привязке набора к конкретному целевому объекту для каждого структурного элемента пользователь может задать **локальные значения** параметров. Локальное значение действует только в рамках конкретного целевого объекта и имеет повышенный приоритет по сравнению с глобальным значением:

- ❖ если для параметра задано локальное значение, то системный процесс `alerts` использует именно его, игнорируя глобальное значение;
- ❖ если для параметра локальное значение не определено, то системный процесс `alerts` использует глобальное значение.

2.3.3. Вычисление статусов сигналов и отправка оповещений

В модуле **Нувола Топаз** за обработку данных, собранных с объектов контроля, и вычисление статусов сигналов отвечает системный процесс `alerts`^[268], за формирование оповещений – системный процесс `notifs`^[278].

Значение сигнала и его статус пользователь может просмотреть в Web-интерфейсе. Оба параметра используются также модулем **Нувола Виста** при построении информационных панелей.

2.3.4. Параметры сигналов и алгоритмы определения статуса

При добавлении сигнала в набор сигналов администратор задаёт следующие параметры:

- ❖ наименование сигнала;
- ❖ тип сигнала – определяет алгоритм вычисления статуса сигнала и множество возможных статусов сигнала; реализованы следующие типы сигналов:

[Gauge](#)^[30];

[InvertGauge](#)^[31];

[SmartGauge](#)^[33];

[Availability](#)^[34];

- ❖ наименование источника данных;
- ❖ [функция и период для определения значения сигнала](#)^[29];
- ❖ параметры алгоритма вычисления статуса сигнала;
- ❖ [признак лидирующего сигнала](#)^[35] (для сигналов типа [Gauge](#)^[30] и [Availability](#)^[34]).

В приведенном списке первые три параметра являются ключевыми, т.к. именно они определяют принципы работы сигнала. Поэтому эти параметры имеют только [глобальные](#)^[28] значения. Для остальных параметров сигнала могут быть заданы [локальные](#)^[28] значения.

2.3.4.1. Функция и период для определения значения сигнала

Значение сигнала определяется с помощью функции, которая выбирает из базы данных производительности одно из значений, полученных от источника данных за заданный период времени **Диапазон** (*range*).

Доступны следующие функции:

- ❖ **среднее** (*avg*) – выбирает из базы данных значения за интервал **Диапазон** (*range*) и вычисляет среднее значение;
- ❖ **минимальное** (*min*) – выбирает из базы данных минимальное значение за интервал **Диапазон** (*range*);
- ❖ **максимальное** (*max*) – выбирает из базы данных максимальное значение за интервал **Диапазон** (*range*);
- ❖ **последнее** (*last*) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (*range*). Это значение может быть только числом;
- ❖ **текущее** (*cur*) – выбирает из базы данных последнее значение за интервал **Диапазон** (*range*). Это значение не обязательно будет числом;

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция *last* возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция *cur* возвращает значение **NAN**;

- ❖ **последнее известное** (*ldef* – от last defined) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (*range*). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

Период времени **Диапазон** (*range*) задаётся в секундах. Чем он больше, тем больше времени требуется для вычисления значения сигнала. По умолчанию – 600 секунд.

2.3.4.2. Алгоритм определения статуса сигнала типа Gauge

Сигнал типа **Gauge** характеризует уровень использования/загрузки интерфейса или другого ресурса (общий принцип: чем больше значение сигнала, тем "хуже" его статус).

Сигналу присваивается статус в соответствии с диапазоном, в который попадает значение сигнала. Диапазоны определяются пороговыми значениями, которые задает администратор в качестве глобальных или локальных значений параметров сигнала:

- ❖ **Нижний порог** (*low*) – по умолчанию равен 0. Как правило, низкий порог не указывают в явном виде, но при необходимости пользователь может его задать в абсолютных единицах.
- ❖ **Нормальный порог** (*normal*) – задается пользователем в процентах от максимального порога или в абсолютных единицах;
- ❖ **Высокий порог** (*high*) – задается пользователем в процентах от максимального порога или в абсолютных единицах;
- ❖ **Максимальный порог** (*max*) – по умолчанию равен значению параметра `Bandwidth` (пропускная способность), определенному для целевого объекта при регистрации устройства в дереве конфигурации. Если параметр `Bandwidth` не определен, то по умолчанию максимальный порог принимается равным **100**. Как правило, максимальный порог не указывают в явном виде, но при необходимости пользователь может задать его в абсолютных единицах.

В таблице приведен перечень статусов сигналов типа `Gauge`, и для каждого статуса указано условие его присвоения – диапазон значений сигнала в привязке к пороговым величинам.

Цветовой код статуса (по умолчанию)	Описание статуса	Условие присвоения статуса (x – значение сигнала)
Серый (grey)	Состояние неизвестно. Устройство не отвечает на запросы.	x – не определено (NaN)
Синий (blue)	Значение сигнала меньше нормы.	$x < low$
Зелёный (green)	Нормальное состояние.	$low \leq x < normal$
Жёлтый (yellow)	Предупреждение: значение сигнала превысило норму.	$normal \leq x < high$
Красный (red)	Проблема: значение сильно превысило норму.	$high \leq x < max$
Бордовый (crimson)	Критическое состояние: значение сигнала превышает пропускные возможности.	$max \leq x$

Алгоритм определения статуса сигнала проиллюстрирован на рисунке.



См. также:

[Параметры сигнала типа Gauge](#) ²⁷⁹

2.3.4.3. Алгоритм определения статуса сигнала типа `InvertGauge`

Сигнал типа `InvertGauge` характеризует уровень неиспользованного ресурса, например тонера для принтера, объема свободного места на диске (общий принцип: чем меньше значение сигнала, тем "хуже" его статус).

Сигналу присваивается статус в соответствии с диапазоном, в который попадает значение сигнала. Диапазоны определяются пороговыми значениями, которые задает администратор в качестве глобальных или локальных значений параметров сигнала:

- ❖ **Высокий порог** (`high`) – по умолчанию принимается равным **100**;

2. Понятия и концепции

- ❖ **Нормальный порог** (*normal*) – задается пользователем в процентах от верхнего порога или в абсолютных единицах;
- ❖ **Низкий порог** (*low*) – по умолчанию равен 0. Как правило, низкий порог не указывают в явном виде, но при необходимости пользователь может его задать в абсолютных единицах;
- ❖ **Минимальный порог** (*min*) – по умолчанию принимается равным 0. Как правило, минимальный порог не указывают в явном виде, но при необходимости пользователь может задать его в процентах от верхнего порога или в абсолютных единицах.

В таблице приведен перечень статусов сигналов типа *InvertGauge*, и для каждого статуса указано условие его присвоения – диапазон значений сигнала в привязке к пороговым величинам.

Цветовой код статуса (по умолчанию)	Описание статуса	Условие присвоения статуса (x – значение сигнала)
Серый (<i>grey</i>)	Состояние неизвестно. Устройство не отвечает на запросы.	x – не определено (NaN)
Синий (<i>blue</i>)	Значение превышает норму.	$x \geq \text{high}$
Зелёный (<i>green</i>)	Нормальное состояние.	$\text{normal} \leq x < \text{high}$
Жёлтый (<i>yellow</i>)	Предупреждение: значение приближается к критическому.	$\text{low} \leq x < \text{normal}$
Красный (<i>red</i>)	Проблема: значение опустилось ниже критического.	$\text{min} \leq x < \text{low}$
Бордовый (<i>crimson</i>)	Критическая проблема: значение ниже минимального.	$x < \text{min}$

Алгоритм определения статуса сигнала проиллюстрирован на рисунке.



См. также:

[Параметры сигнала типа *InvertGauge*](#)^[28]

2.3.4.4. Алгоритм определения статуса сигнала типа SmartGauge

Сигнал типа **SmartGauge** характеризует состояние сигнала в соответствии с заданными диапазонами значений. Администратор может настроить произвольное количество диапазонов (в этом принципиальное отличие от сигнала типа **Gauge** и **InvertGauge**).

Сигналу присваивается статус в соответствии с диапазоном, в который попадает значение сигнала. Бордовый статус не предусмотрен. Для каждого статуса может быть задано несколько диапазонов. Если значение сигнала не попадает ни в один из заданных диапазонов, то сигналу присваивается синий статус.

ВНИМАНИЕ! Логика разграничения диапазонов целиком и полностью лежит в зоне ответственности администратора. Он настраивает границы диапазонов в соответствии с техническими характеристиками контролируемых объектов и особенностями технологических процессов предприятия.

Таким образом, для сигнала типа *SmartGauge* задаются следующие параметры:

- ❖ **Нижний порог** – диапазон значений для синего статуса сигнала;
- ❖ **Нормальный диапазон** – диапазон значений для зеленого статуса сигнала;
- ❖ **Незначительный порог** – диапазон значений для желтого статуса сигнала;
- ❖ **Высокий порог** – диапазон значений для красного статуса сигнала;
- ❖ **Максимальный порог** – значение, от которого берутся проценты при вычислении границ диапазонов. По умолчанию равен значению параметра *Bandwidth* (пропускная способность), определенному для целевого объекта при регистрации устройства в дереве конфигурации. Если параметр *Bandwidth* не определен, то по умолчанию максимальный порог принимается равным **100**. Как правило, максимальный порог не указывают в явном виде, но при необходимости пользователь может задать его в абсолютных единицах.

При настройке диапазонов следует руководствоваться правилами:

- ❖ диапазон вводится в формате: **[a:b]**, причем нижняя граница принадлежит диапазону ($a \leq x < b$);
- ❖ для каждого статуса можно задать несколько диапазонов, перечислив их через запятую;
- ❖ диапазон можно задать в абсолютных единицах или в процентах от **Максимального порога**. В последнем случае символ % указывается однократно после списка диапазонов, например: **[10:20]**, **[70:80]%**;
- ❖ для указания бесконечности следует использовать обозначения **-inf** и **inf**. Один из диапазонов обязательно должен быть не ограничен справа, например: **[90:inf]%**.

2. Понятия и концепции



См. также:

[Параметры сигнала типа SmartGauge](#) ²⁸³

2.3.4.5. Алгоритм определения статуса сигнала типа Availability

Сигнал типа **Availability** показывает доступность целевого объекта, а именно: интерфейса, и используется для выявления недоступных интерфейсов.

Алгоритм определения статуса сигнала основан на булевой логике и использует эталонную величину (*ideal*):

- ❖ если значение сигнала совпадает с эталонной величиной то, сигнал получает **зеленый** статус;
- ❖ при любом другом значении сигнала – **красный** статус.

В таблице приведен перечень статусов сигналов типа Availability, и для каждого статуса указано условие его присвоения.

Цветовой код статуса (по умолчанию)	Описание статуса	Условие присвоения статуса (x – значение сигнала)
Зелёный (green)	Проблем нет. Интерфейс работает.	$x = ideal$
Красный (red)	Проблема требует внимания. Интерфейс не работает.	$x \neq ideal$
Серый (grey)	Состояние неизвестно. Устройство не отвечает на запросы.	x – не определено

Таким образом, для сигнала типа Availability предусмотрены следующие настраиваемые параметры:

- ❖ **Эталон (ideal)** – эталонная величина;

- ❖ **Описание для "хороший"** (green legend) – текст, отображаемый в оповещении при **зеленом** статусе сигнала;
- ❖ **Описание для "плохой"** (red legend) – текст, отображаемый в оповещении при **красном** статусе сигнала.

См. также:

[Параметры сигнала типа Availability](#)^[285]

2.3.4.6. Признак Лидер

Лидер (Leader) – признак, свидетельствующий о повышенном приоритете сигнала в наборе сигналов. Признак является параметром сигналов типов Gauge, InvertGauge, SmartGauge и Availability.

По умолчанию ни один сигнал не является лидером (лидирующим). Признак **Лидер** может быть установлен для одного или нескольких сигналов в наборе.

Если в наборе, привязанном к целевому объекту, нет лидирующих сигналов, то всегда обрабатываются все сигналы, т.к. они считаются равнозначными. В этом случае, может возникнуть ситуация, при которой сразу все сигналы получают «плохой» статус, например, если целевой объект стал недоступен. При соответствующих настройках набора оповещений по каждому сигналу может быть сформировано оповещение (или несколько оповещений); а если спустя некоторое время проблемы не были устранены, то могут отправляться еще и повторные оповещения.

Чтобы избежать возможного шквала оповещений, целесообразно выделять в наборе ключевые (лидирующие) сигналы. Тогда в случае возникновения серьезных проблем или аварий в сети будут формироваться оповещения только по таким наиболее важным сигналам.

Если в наборе, привязанном к целевому объекту, имеются сигналы с признаком **Лидер**, то сначала обрабатываются именно такие лидирующие сигналы, и в зависимости от их статусов выявляется необходимость обработки остальных сигналов набора:

- ❖ если хотя бы один из лидирующих сигналов в наборе имеет «плохой» статус (т.е. отличный от зеленого), то остальные сигналы набора не обрабатываются;
- ❖ если все лидирующие сигналы имеют зеленый статус, то обрабатываются остальные сигналы набора.

Лидирующие сигналы обрабатываются всегда, вне зависимости от статусов других лидирующих сигналов.

2.3.5. Класс оповещений Simple

Класс оповещений – совокупность параметров, определяющих условия, при возникновении которых формируются оповещения.

Логику отправки оповещений определяет тип класса оповещений (параметр класса оповещений). В текущей версии реализован один тип класса оповещений – **Simple**, алгоритм которого предполагает анализ изменения статуса сигнала. В классе оповещений типа **Simple** администратор задаёт:

- ❖ настройки множеств "хороших", "плохих" и "игнорируемых" статусов сигнала;
- ❖ изменения статусов, при которых формируются оповещения;
- ❖ множество сигналов, по которым будут формироваться оповещения (по умолчанию – по всем сигналам);
- ❖ параметры алгоритма формирования повторных оповещений (по умолчанию повторные оповещения не формируются).

2.3.6. Методы и типы оповещений

Метод оповещений – совокупность параметров, определяющих способ формирования оповещений. Метод оповещений содержит ключевой параметр **тип оповещения** и параметры, конкретизирующие выполняемое действие, например: тип оповещения «отправка сообщения по электронной почте», адреса получателей сообщений, сетевой адрес почтового сервера.

Метод оповещения задается следующими параметрами:

- ❖ имя метода;
- ❖ тип оповещений;
- ❖ признак активности метода:
 - on** – метод включен, оповещения формируются;
 - off** – метод отключен, оповещения не формируются (можно использовать для временной блокировки отправки оповещений);
- ❖ параметры формирования и доставки оповещения – набор этих параметров определяется типом оповещений.

В приведенном списке первые два параметра имеют только глобальные значения, т.е. они задаются при создании метода оповещений и не подлежат последующему изменению. Для остальных параметров метода оповещения могут быть заданы локальные значения.

Тип оповещения – вид уведомления, например, отправка сообщения по электронной почте или SMS, создание записи в лог-файле или базе данных, передача сообщения о событии в стороннюю систему, выполнение иной программной процедуры (скрипта).

См. также

[Справочник Типы оповещений](#)^[288]

2.4. Информационные панели

Информационная панель (Dashboard) – конфигурационный инструмент, обеспечивающий в режиме реального времени наглядное графическое web-представление информации о состоянии объектов контроля, данных об обобщенном состоянии инфраструктуры предприятия или ее сегмента, а также предоставляемых им услуг.

На информационной панели сигнал отображается в виде графического элемента, цвет которого изменяется в зависимости от статуса сигнала. По умолчанию для цветовой индикации элементов применяется палитра, предусмотренная для сигналов различных [типов](#)^[289].

ПК Нувола поддерживает информационные панели в формате SVG. Формат SVG позволяет пользователю настраивать внешний вид информационных панелей (включая порядок значков, шрифты, цвет и размер, изображения, рамки и формы). Для создания информационных панелей предназначено Web-приложение [Редактор информационных панелей](#)^[170].

2.5. Пользователи

В ПК Нувола реализован гибкий механизм разграничения прав пользователей, позволяющий настроить доступ к информации и функционалу в соответствии с требованиями проектного решения.

Для идентификации пользователей ПК Нувола и разграничения прав доступа используются учетные записи и группы пользователей. Пользователь может входить в состав одной или нескольких групп пользователей. Группа пользователей может содержать одного или нескольких пользователей.

Учетная запись – совокупность данных о пользователе, необходимая для его аутентификации в ПК Нувола и предоставления прав доступа. Учетная запись содержит следующие параметры:

- ❖ отображаемое имя пользователя;
- ❖ логин для входа в Web-интерфейс;
- ❖ пароль;
- ❖ дополнительная справочная информация (телефоны, электронная почта, должность);
- ❖ способ аутентификации и [политика безопасности](#)^[39], если используется внутренняя аутентификация;
- ❖ перечень *групп пользователей*, определяющих набор разрешенных действий и доступ пользователя к информации.

Группа пользователей – это совокупность прав для работы в Web-интерфейсе и учетных записей пользователей, обладающих этими правами. Группа пользователей включает в себя следующие компоненты:

- ❖ [глобальные политики](#)^[37];
- ❖ [пути к узлам дерева конфигурации с указанием прав доступа](#)^[37] – разрешенных пользователю внутри заданной ветви дерева конфигурации;
- ❖ ссылки на хранимые на сервере Нувола файлы конфигурации, в соответствии с которыми Web-интерфейс отображается на персональном компьютере и мобильных устройствах пользователя;
- ❖ перечень учетных записей пользователей, входящих в группу.

2.5.1. Глобальные политики

Глобальная политика – доступный пользователю функционал Web-интерфейса, направленный на решение задач определенной категории.

При настройке прав пользователей администратор устанавливает (включает) глобальные политики для групп пользователей.

ПК Нувола поставляется с набором глобальных политик, созданных разработчиками. Если его недостаточно для реализации проектного решения, то администратор через командную строку может [создать](#)^[70] другие глобальные политики.

2.5.2. Права доступа к узлам дерева конфигурации

Право доступа – действие, которое пользователь может выполнять в рамках заданной ветви дерева конфигурации.

Администратор добавляет в группу пользователей пути на узлы дерева конфигурации и для каждого пути задает *права доступа* – разрешенные действия пользователя.

ПК Нувола поставляется с набором прав доступа, созданных разработчиками. Если его недостаточно для реализации проектного решения, то администратор через командную строку может [создать](#)^[70] другие права доступа.

Пользователи получают доступ ко всей ветви дерева конфигурации, на который указан путь (т.е. пользователю доступны все нижележащие узлы, начиная с заданного).

Если задан путь без указания прав доступа, то соответствующая ветвь дерева конфигурации доступна только для просмотра.

2.5.3. Права пользователя, принадлежащего нескольким группам пользователей

Пользователь может входить в состав нескольких групп пользователей, т.е. учетная запись пользователя может быть прикреплена к нескольким группам пользователей. В этом случае действуют следующие принципы предоставления прав:

- ❖ если в группах пользователей установлены разные глобальные политики, то пользователь получает объединенные права, регламентированные политиками всех прикрепленных групп (по принципу объединения множеств);
- ❖ если в группах пользователей для одного и того же пути разрешены разные действия, то пользователь получает права на действия из всех прикрепленных групп (по принципу объединения множеств);
- ❖ если в группах пользователей для вложенных путей разрешены разные действия, то для конкретного узла дерева конфигурации будут доступны действия, которые заданы для пути на родительский узел, ближайший к запрошенному пользователем.

2.5.4. Администратор (пользователь root) и другие пользовательские роли

Понятие *пользовательской роли* в ПК Нувола имеет условный характер, т.к. логика настройки прав доступа зависит от распределения задач и должностных обязанностей между работниками предприятия – пользователями ПК Нувола в рамках конкретного проектного решения.

В эксплуатационной документации для удобства изложения выделены три принципиальные пользовательские роли.

Администратор координирует действия пользователей, осуществляет общую настройку ПК Нувола и обеспечивает ее штатное функционирование. Администратор работает под учетной записью `root` и обладает максимальными правами.

Учетная запись `root` жестко заведена в ПК Нувола. Допустимо изменять только ее отображаемое имя и пароль. Учетная запись `root` прикреплена к группе пользователей **Администраторы**, в которой установлены все глобальные политики, и отключить их нельзя. Учетную запись `root` и группу пользователей **Администраторы** удалить невозможно.

Локальный администратор – пользовательская роль, позволяющая конфигурировать в ПК Нувола объекты контроля в рамках своей зоны ответственности (сегмента инфраструктуры), а также осуществлять настройку вывода информации о состоянии объектов контроля (обобщенного состояния инфраструктуры и пр.) в наглядной форме.

В группах пользователей, предназначенных для локальных администраторов, допустимы любые глобальные политики и права доступа для путей.

Конечный пользователь ПК Нувола – пользовательская роль, позволяющая получать информацию о состоянии объектов контроля в наглядной форме посредством Web-интерфейса. Конечный пользователь обладает минимальным набором прав.

В группах пользователей, предназначенных для конечных пользователей, установлена глобальная политика **(l) Вход в систему**, допустимы также политики **(r) Просмотр графика в режиме реального времени** и **(с) Подтверждение уведомлений**. Для путей права доступа не указаны (или установлено право доступа **(h) Запрет на доступ**), т.к. данная пользовательская роль предполагает доступ только на просмотр информации.

Функциональные возможности, доступные конечному пользователю, описаны в **Руководстве пользователя**. Остальные функциональные возможности (т.е. доступные администратору `root` и локальным администраторам) описаны в **Руководстве администратора**.

2.5.5. Использование политик безопасности для внутренней аутентификации

Для аутентификации пользователей при подключении к серверу Нувола через Web-интерфейс предусмотрены два способа.

Внешняя аутентификация осуществляется с использованием сторонних средств.

Внутренняя аутентификация учетных записей выполняется в базе данных LDAP, которая является компонентой модуля **Нувола Топаз**, с применением политик безопасности.

Политика безопасности учетной записи – это совокупность параметров, определяющих порядок использования пароля и правила блокировки учетной записи. Администратор имеет право создавать и изменять политики безопасности, причем только через командную строку. В политике безопасности администратор может задать следующие параметры:

- ❖ срок действия пароля;
- ❖ необходимость предупреждения пользователя о предстоящем окончании срока действия пароля;
- ❖ необходимость сохранения старых паролей в истории в базе данных LDAP;
- ❖ необходимость блокировки учетной записи при достижении максимального количества попыток ввода пароля (в случае блокировки учетной записи в Web-интерфейсе пользователю выдается сообщение);
- ❖ время блокировки учетной записи при достижении максимального количества попыток ввода пароля;
- ❖ минимальное количество символов в пароле.

При настройке учетной записи пользователя администратор выбирает способ аутентификации и в случае использования внутренней аутентификации указывает политику безопасности.

3.

Установка и удаление модуля Нувола Топаз

3. Установка и удаление модуля Нувола Топаз

Установку и удаление модуля **Нувола Топаз** следует выполнять пользователю с правами администратора root.

Вместе с установкой и удалением модуля **Нувола Топаз** осуществляется установка и удаление модулей **Нувола ESB** и **Нувола Виста**.

3.1. Подготовка к установке: настройка сервера Linux

Перед установкой ПМ **Нувола Топаз** следует убедиться, что используемые АРМ и сервер удовлетворяют [требованиям к техническим средствам и программному обеспечению](#)^[16].

До начала установки ПМ **Нувола Топаз** на подходящей аппаратной платформе должна быть полностью установлена серверная операционная система.

Чтобы настроить сервер, следует:

1. убедиться, что на сервере установлено следующее программное обеспечение:

- ❖ библиотека Libstdc++ (либо из RPM-пакета, либо из другого источника);
- ❖ sh-совместимый интерпретатор команд bash;
- ❖ проектная среда C/C++ (gcc/gmake/gdb/gprof/glibc);
- ❖ язык программирования Python версии 3.6.8 и выше (python36);
- ❖ библиотеки Python версии 3.6.8 и выше (python36-libs);
- ❖ библиотека Libtool-ldl;
- ❖ утилиты архивации gzip, zip/unzip;
- ❖ утилиты GNU grep, Lynx, lsk/lsof;
- ❖ динамические библиотеки Apache:
 - APR версии 1.4.8 и выше;
 - APR-util версии 1.5.2 и выше;
- ❖ утилита strace/truss/tusc (в зависимости от операционной системы);

2. обеспечить доступ к необходимым портам и сервисам.

Должны быть доступны следующие порты и сервисы, в зависимости от способа мониторинга:

- ❖ 161/UDP (сервис SNMP), если наблюдение за устройством ведётся через сервис SNMP (большая часть устройств наблюдается через SNMP);
- ❖ ICMP-пинг (для проверки RTT);
- ❖ для проверки доступности какого-либо сервиса также должен быть открыт доступ к этому сервису.

Следующим сервисам, запускаемым на сервере, может потребоваться доступ извне:

- ❖ сервис Apache HTTP (по умолчанию порт 80);
- ❖ сервис Apache HTTPS (по умолчанию порт 443);
- ❖ сервис удалённой настройки ПМ **Нувола Топаз** (по умолчанию порт 5566);

3. настроить область подкачки (Swap Space).

Объём области подкачки, доступный в системе, всегда должен быть не меньше, чем объём имеющейся оперативной памяти плюс 2 Гб. Следуйте также рекомендациям производителя ОС.

4. подготовить разделы диска.

Диск должен быть разделён таким образом, чтобы объём каждого раздела был достаточным для размещения приложений и данных системы.

По умолчанию ПМ **Нуволы Топаз** устанавливается в директории: `/usr/local/lancelot` и `/usr/local/avalon`. Объем доступного свободного места определяется на этапе планирования проектного решения совместно с представителями компании-разработчика;

5. проверить системные часы.

Точность системных часов имеет принципиальное значение, так как от неё в значительной степени зависит работа системы наблюдения и синхронизации ПМ **Нуволы Топаз**. Неправильные настройки времени могут привести к нестабильной работе ПМ **Нуволы Топаз** и невозможности обмена данными с клиентской оболочкой системы. Кроме того, из-за неверных настроек системных часов могут возникнуть проблемы с идентификацией пользователей, желающих использовать сетевой интерфейс ПК Нуволы.

На серверах и клиентах, которые будут включены в ПК Нуволы, необходимо правильно установить следующие значения системных часов:

- ❖ текущее время;
- ❖ текущая дата;
- ❖ часовой пояс;
- ❖ режим перехода на летнее время;
- ❖ значение системных часов (UTC) и соответствие между системным временем и часами устройства.

Эти значения всегда должны быть синхронизированными;

6. проверить настройку часового пояса.

- ❖ Windows: дважды щёлкнуть мышью на часах, расположенных в правой части панели задач Windows, затем в появившемся окне на вкладке **Часовой пояс** установить часовой пояс и, при необходимости, переход на летнее время и обратно;
- ❖ Linux OS: рекомендуется установить часовой пояс в зависимости от географического положения и использовать протокол `ntp` для синхронизации системного времени;

7. синхронизировать время всех систем, на которых установлена ПК Нуволы, например, с помощью сервиса `ntpd` или утилиты `ntpdate`, используемых в Unix;

8. обеспечить доступ по протоколу SNMP к объектам контроля.

Для обеспечения сбора данных и получения информации необходимо обеспечить доступ по протоколу SNMP к сетевым устройствам, следуя соответствующей технической документации;

9. организовать резервный архив.

Резервные копии конфигурации создаются автоматически ежедневно. Необходимо создать резервные копии для папки данных ПМ **Нуволы Топаз** (по умолчанию: `/usr/local/lancelot/var/lancelot-data`). Ее размер может достигать сотен гигабайт, тем не менее, необходимо поддерживать её резервную копию. Файлы БД (Round Robin Database) отличаются высокой степенью сжатия, поэтому сжатые файлы занимают ~ 20%-30% от исходного размера.

Следует контролировать резервное копирование папки данных ПМ **Нуволы Топаз**, так как эта операция создаёт высокую нагрузку на контроллеры памяти, дисков и сети.

3.2. Установка модуля Нуволы Топаз

Для выполнения этапов установки ПМ **Нуволы Топаз**, описанных далее, следует подключиться к серверу Linux под пользователем `root`.

Формат имени инсталляционного файла

Имя инсталляционного файла ПМ **Нуволы Топаз** имеет следующий формат:

3. Установка и удаление модуля Нувола Топаз

nuvola-topaz-X-X.V.elZ.YYYYYY.rpm,

где:

X-X.V – номер версии ПМ **Нувола Топаз** (v – номер сборки; может содержать от 1 до 10 цифр и/или букв, а также точку);

elZ – номер версии Linux, для которой предназначен ПМ **Нувола Топаз** (el7 – CentOS 7);

YYYYYY – обозначение платформы (x86_64).

Пример имени файла: nuvola-topaz-7-1.0.el7.x86_64.rpm.

Проверка информации об инсталляционном файле

Чтобы получить информацию об инсталляционном файле, следует ввести в командной строке:

```
# rpm -qip <install_file_name>
```

где <install_file_name> – имя инсталляционного файла.

Например:

```
# rpm -qip nuvola-topaz-7-1.0.el7.x86_64.rpm
```

В результате будут выведены данные о версии. Дата выпуска версии и сборки должна быть актуальной.

Выполнение процедуры установки

Чтобы установить модуль **Нувола Топаз**, следует запустить процедуру инсталляции с помощью команды:

```
# rpm -ivh <install_file_name>
```

где <install_file_name> – имя инсталляционного файла.

Например:

```
# rpm -ivh nuvola-topaz-7-1.0.el7.x86_64.rpm
```

Установка лицензии

1. Файлы лицензий, предоставленные представителем ООО «Венто Технологии», сохранить на сервере в директории /usr/local/lancelot/etc/license;

2. для смены пользователя выполнить команду:

```
su - avalon
```

3. от имени пользователя avalon выполнить команду:

```
license.pl
```

В результате будет выведено сообщение об успешном выполнении команды, например:

```
signaling lancelotHdbServer [pid: 2821]
```

Локализация Web-интерфейса

Рекомендуется выполнять локализацию до создания дерева конфигурации. Чтобы переключить язык Web-интерфейса на английский, следует выполнить команду:

```
konfne --langselect en
```

Чтобы переключить язык Web-интерфейса на русский, следует выполнить команду:

```
konfne --langselect ru
```

3.3. Проверка правильности функционирования

При установке ПМ **Нувола Топаз** автоматически запускаются его системные процессы. В таблице приведены команды для проверки параметров функционирования ПМ **Нувола Топаз**.

Параметр	Команда для проверки	Комментарий
Объем используемых ресурсов жестких дисков	для HP-UX: df -k для Linux: df -h	Объем используемых ресурсов жестких дисков не должен превышать 60-70%. Необходимо удалить лишние сетевые интерфейсы.
Размер лог-файлов системных процессов	ls - al /usr/local/lancelot/var/lancelot-logs /usr/local/avalon/logs	Лог-файл системного процесса не должен превышать 2 Гб.
Статус системных процессов ПМ Нувола Топаз	overlord.pl ping	Команда <code>overlord.pl ping</code> показывает состояние системных процессов. Запущенные системные процессы имеют статус ОК. Если один или несколько системных процессов имеют статус <code>down</code> , рекомендуется выполнить команду <code>overlord.pl check</code> . Если при выполнении этой команды выявлены ошибки, то необходимо проверить лог соответствующего системного процесса.

Комплекс внутренней диагностики

Администратор `root` может контролировать работу ПМ **Нувола Топаз** с помощью *комплекса внутренней диагностики*, который создается автоматически в папке **Tree/Внутренняя диагностика** при установке модуля.

Комплекс внутренней диагностики – совокупность предустановленных объектов, предоставляющих информацию о функционировании ПМ **Нувола Топаз**: состояния и статистику системных процессов, состояния основных портов, состояния компонент сервера, на котором установлен ПМ **Нувола Топаз**.

Данные, получаемые от объектов комплекса, доступны администратору `root` для просмотра в [Web-интерфейсе](#)^[154].

Для создания/переконфигурации комплекса предназначена утилита [selfMonitoringConfig.pl](#)^[263].

После установки ПМ **Нувола Топаз** рекомендуется переконфигурировать *комплекс внутренней диагностики*.

Команда для создания/переконфигурации комплекса с именами на русском языке:

```
./selfMonitoringConfig.pl
```

3. Установка и удаление модуля Нувола Топаз

или

```
./selfMonitoringConfig.pl -f 'Внутренняя Диагностика' -n 'Мониторинг Сервисов'
```

Команда для создания/переконфигурации комплекса с именами на английском языке:

```
./selfMonitoringConfig.pl -f 'Self Monitoring' -n 'Services Monitoring'
```

При необходимости администратор root может изменить предустановленные настройки сигналов и оповещений в [Редакторе сигналов и оповещений](#)^[117].

3.4. Удаление модуля Нувола Топаз

Чтобы удалить ПМ Нувола Топаз, следует:

1. получить имя пакета Nuvola с помощью команды:

```
# rpm -qa | grep nuvola
```

В результате будет выдано имя пакета, например:

```
nuvola-topaz-7-1.0.el7.x86_64.rpm
```

2. запустить процедуру удаления с помощью команды:

```
# rpm -ev <имя_пакета>
```

Например:

```
# rpm -ev nuvola-topaz-7-1.0.el7.x86_64
```

По мере выполнения процедуры удаления выводятся сообщения о ее этапах, например:

```
...  
Preparing packages...  
Saving Crontab to /home/avalon/avalon.crontab  
Removing Crontab  
Erasing Cache  
Erasing Limits  
Stopping running processes  
...
```

По окончании удаления выдается сообщение:

```
Shutting down Nuvola Topaz - completed
```

4.

Установка и удаление Web-интерфейса

4. Установка и удаление Web-интерфейса

Установку и удаление Web-интерфейса следует выполнять пользователю с правами администратора root.

4.1. Команды для установки и удаления инсталляционного файла

Формат имени инсталляционного файла

Имя инсталляционного файла Web-интерфейса имеет следующий формат:

```
nuvola-web-X-X.V.YYYYYY.rpm,
```

где:

X-X.V – номер версии (V – номер сборки; может содержать от 1 до 10 цифр и/или букв, а также точку);

YYYYYY – обозначение платформы (x86_64).

Пример имени файла: `nuvola-web-7-1.0.x86_64.rpm`.

Установка Web-интерфейса

Чтобы установить Web-интерфейс, на сервере Нувола от имени пользователя root следует запустить процедуру инсталляции с помощью команды:

```
rpm -ivh nuvola-web-7-1.0.x86_64.rpm
```

После окончания установки следует убедиться, что на сервере Нувола существует и запущен системный процесс `rest_api`^[278]. Для этого от имени пользователя `avalon` выполнить команду:

```
overlord.pl ping
```

Команда для создания системного процесса `rest_api`:

```
overlord.pl create rest_api script=rest.pl esb=127.0.0.1  
logfile=/usr/local/lancelot/var/lancelot-logs/rest_api.log
```

Команда для запуска системного процесса `rest_api`:

```
overlord.pl start rest_api
```

Удаление Web-интерфейса

Чтобы удалить Web-интерфейс из 64-битной ОС, следует в командной строке ввести:

```
# rpm -e <имя_пакета>
```

Например:

```
# rpm -e nuvola-web-7-1.0.x86_64
```

4.2. Проверка правильности установки

Служебная информация о выполнении процедуры инсталляции записывается в файл:
`/usr/local/lancelot/install.log`.

Чтобы убедиться в том, что ПМ Нувола **Топаз** и Web-интерфейс установлены правильно, следует:

1. в Internet Explorer в адресной строке ввести адрес сервера Нувола. Появится страница входа в Web-интерфейс:

Логин:

Пароль:

2. ввести имя пользователя **operator** и пароль **operator** (первоначальные настройки). После первого входа в систему настоятельно рекомендуется сменить пароль. В случае успешного подключения открывается страница Web-интерфейса.

Если лицензия не установлена, выдается сообщение об ошибке.

3. через командную строку от имени пользователя `avalon` запустить неработающие системные процессы ПМ **Нувола Топаз** с помощью команды:

```
overlord.pl check
```

Пример результата выполнения команды:

```
konfne      - ok
konfigd     - ok
maintenance - ok
rtg         - ok
rest_long_api - ok
rest_api    - ok
querymaker  - ok
pinger     - ok
fnmp       - ok
esbevent   - ok
kollector  - ok
thaw       - ok
alerts     - ok
esb        - ok
de_svg-simple - ok
de_svg-js  - ok
de_providerd - ok
de_builderd - ok
dataserver - ok
```

Проверить состояние системных процессов с помощью команды:

```
overlord.pl ping
```

Системные процессы должны иметь статус UP. Пример результата выполнения команды:

```
alerts          (alerts.pl      ) [16077] [ UP ]
dataserver      (generic        ) [16293] [ UP ]
de_builderd     (de_builderd.pl ) [16263] [ UP ]
de_providerd    (de_providerd.pl) [16233] [ UP ]
de_svg-js       (de_svg-js.pl   ) [16220] [ UP ]
de_svg-simple   (de_svg.pl      ) [16209] [ UP ]
esb             (generic        ) [16100] [ UP ]
esbevent        (kollector      ) [16051] [ UP ]
fnmp            (kollector      ) [16044] [ UP ]
kollector       (kollector      ) [16057] [ UP ]
konfigd         (konfigd        ) [15998] [ UP ]
konfne          (konfigd        ) [15993] [ UP ]
maintenance     (maintenance.pl ) [16003] [ UP ]
pinger          (querymaker.pl  ) [16030] [ UP ]
querymaker      (querymaker.pl  ) [16026] [ UP ]
rest_api        (rest.pl        ) [16020] [ UP ]
rest_long_api   (rest.pl        ) [16013] [ UP ]
rtg             (rtg.pl         ) [16008] [ UP ]
thaw            (thawne.pl     ) [16068] [ UP ]
```

4.3. Настройки рабочей станции для соединения с сервером Нувола

В ряде случаев (в зависимости от условий применения ПК Нувола) может потребоваться дополнительная настройка рабочих станций пользователей / администраторов, использующих Web-интерфейс. Приведенные ниже рекомендации направлены на обеспечение соединения рабочей станции с сервером Нувола через Web-интерфейс.

Для выполнения описанных настроек необходимы права администратора рабочей станции.

Отключение спящего режима

Зачастую распорядок работы пользователя не предусматривает ежедневную перезагрузку браузера или компьютера. Как правило, наиболее важные информационные панели и страницы Web-интерфейса пользователь выводит на экран однократно, чтобы просматривать их постоянно в течение длительного срока.

Соединение рабочей станции пользователя с сервером Нувола поддерживается встроенной в web-приложение функцией автоматического обновления аутентификационных данных. Согласно настройке по умолчанию обновление происходит каждые 3 часа. Если на рабочей станции планом энергоснабжения предусмотрен спящий режим, то после выхода рабочей станции из этого режима возможны следующие ситуации:

- ❖ функция автоматического обновления может не сработать, и пользователь будет просматривать на ранее открытых страницах Web-интерфейса неактуальные данные;
- ❖ может произойти отключение (logout) web-приложения от сервера Нувола, и пользователю придется вновь вводить имя и пароль для подключения.

Поэтому во избежание некорректных ситуаций и для удобства работы пользователей рекомендуется отключать спящий режим на рабочих станциях, обеспечивающих работу с Web-интерфейсом.

Увеличение количества соединений по WebSocket

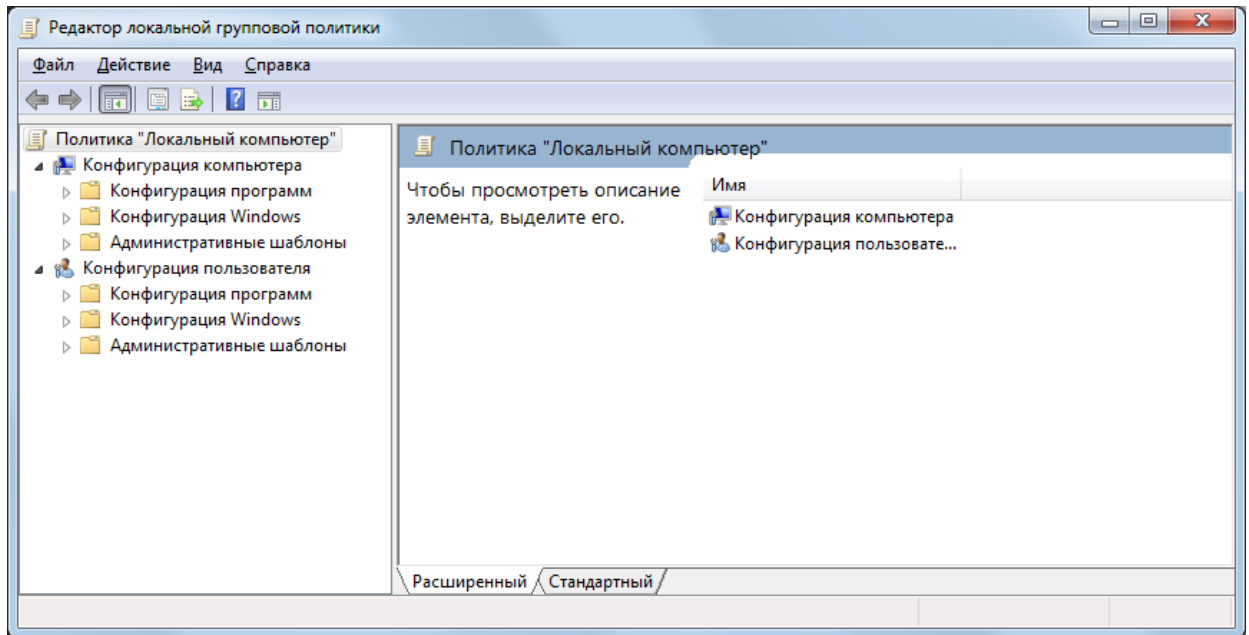
Приведенная далее рекомендация актуальна при использовании браузера **Internet Explorer**. В других браузерах описанная проблема не возникает.

При работе пользователя в Web-интерфейсе между браузером и сервером может устанавливаться одновременно несколько соединений по протоколу WebSocket, например: при просмотре графика JRTG, изменении каких-либо параметров или выполнении других команд управления. Для браузера Internet Explorer по умолчанию количество таких соединений ограничено. В процессе работы с Web-интерфейсом установленный лимит быстро исчерпывается и выдается ошибка о потере соединения с сервером. Чтобы избежать подобных ситуаций, на рабочей станции пользователя следует увеличить допустимое количество соединений по протоколу WebSocket. Данная настройка особенно необходима для рабочих станций администраторов.

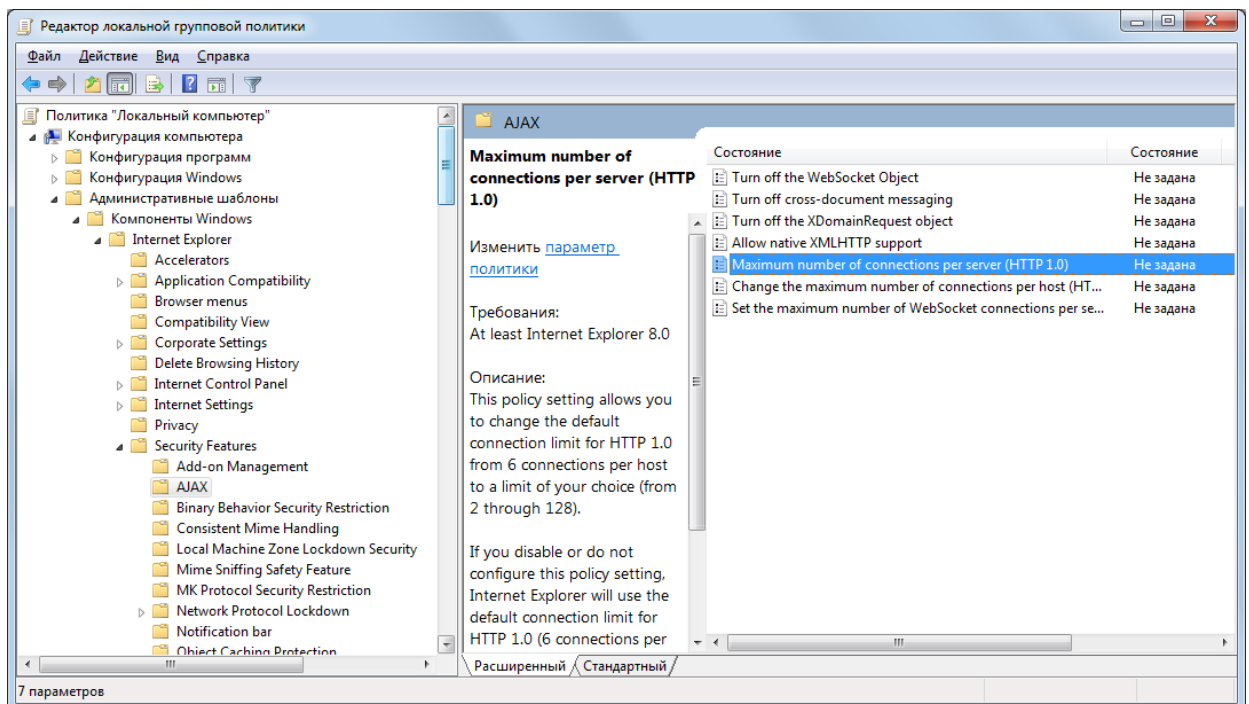
Итак, чтобы на рабочей станции увеличить количество допустимых соединений по протоколу WebSocket между браузером Internet Explorer и сервером, следует:

1. в меню **Пуск** выбрать пункт **Выполнить**;
2. ввести `gpedit.msc` и нажать кнопку **ОК** (вызываемый файл `C:\Windows\System32\gpedit.msc`). В результате откроется **Редактор локальной групповой политики**;

4.3. Настройки рабочей станции для соединения с сервером Нувола

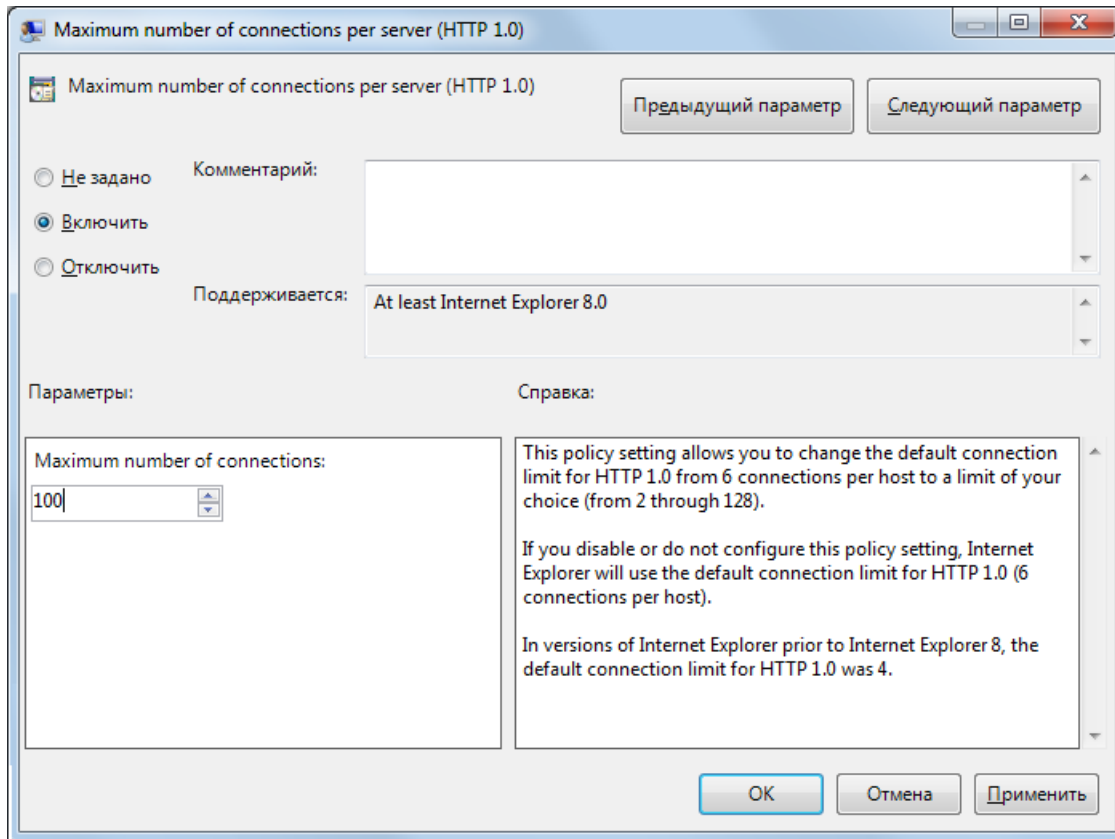


3. выбрать папку **Административные шаблоны / Компоненты Windows / Internet Explorer / Security Features / AJAX**;
4. дважды кликнуть на наименовании параметра **Maximum number of connections per server**;



5. в открывшемся окне выбрать вариант **Включить** и ввести максимальное количество соединений, например: 100;

4. Установка и удаление Web-интерфейса



6. для сохранения изменений нажать кнопку **Применить**, затем нажать кнопку **ОК**.

5.

Установка и удаление модуля Нувола Аналитика

5. Установка и удаление модуля Нуволы Аналитика

Установку и удаление модуля **Нуволы Аналитика** должен выполнять пользователь с правами администратора `root`.

5.1. Команды для установки и удаления инсталляционного файла

Формат имени инсталляционного файла

Имя инсталляционного файла модуля **Нуволы Аналитика** имеет следующий формат:

```
nuvola-analytics-X-X.V.elZ.YYYYYY.rpm,
```

где:

`X-X.V` – номер версии модуля **Нуволы Аналитика** (`V` – номер сборки; может содержать от 1 до 10 цифр и/или букв, а также точку);

`elZ` – номер версии Linux, для которой предназначен модуль **Нуволы Аналитика** (`el7` – CentOS 7);

`YYYYYY` – обозначение платформы (`x86_64`).

Пример имени файла: `nuvola-analytics-7-1.75.15379.el7.x86_64.rpm`.

Установка модуля Нуволы Аналитика

Чтобы установить модуль **Нуволы Аналитика**, на сервере Нуволы от имени пользователя `root` следует запустить процедуру инсталляции с помощью команды:

```
# rpm -ivh <install_file_name>
```

где `<install_file_name>` – имя инсталляционного файла.

Например:

```
# rpm -ivh nuvola-analytics-7-1.75.15379.el7.x86_64.rpm
```

Удаление модуля Нуволы Аналитика

Чтобы удалить модуль **Нуволы Аналитика**, следует в командной строке ввести:

```
# rpm -e <имя_пакета>
```

Например:

```
# rpm -e nuvola-analytics-7-1.75.15379.x86_64
```

5.2. Настройка сервера Нуволы Аналитика

Настройка сервера **Нуволы Аналитика** сводится к указанию адреса сервиса имен CORBA и имени CORBA-объекта, под которым сервер должен зарегистрироваться. Для настройки сервера в сценарии `/etc/init.d/sybillad` следует отредактировать строки:

```
NAMESERVER='NameService=corbaname::<IP>:<Port>'
```

```
ОБЪЕКТ='<имя_объекта>'
```

```
LOGLEVEL=<1-5>
```

где:

- ❖ NAMESERVER устанавливает настройки подключения, а именно сетевой адрес и номер порта компьютера, на котором запущен процесс сервиса имен CORBA. Слова NameService=corbaname являются частью строки подключения. Следует изменять только поля:
 - ▶ <IP> – IP-адрес сервера, на котором запущен сервис имен CORBA; по умолчанию **127.0.0.1** (т.е. сервер, на котором установлен сервер **Нуволы Аналитика**). Если конфигурация установки **Нуволы Аналитика** требует использование сервиса имен, запущенного вместе с модулем **Нуволы Топаз**, то в этом поле необходимо указать IP-адрес сервера Системы Нуволы;
 - ▶ <Port> – номер порта, на который настроен сервис имен CORBA; по умолчанию **2809**;
- ❖ OBJECT – имя объекта, с которым сервер **Нуволы Аналитика** регистрируется в сервисе имен CORBA. По умолчанию mainServer.
- ❖ LOGLEVEL – уровень детализации протоколов. Возможные значения: от **1** до **5**, где **1** – записывать только ошибки, **5** – записывать полную отладочную информацию.

Если требуется запустить несколько серверов предсказаний, то для каждого нужно определить уникальное имя объекта, под которым каждый сервер будет регистрироваться в службе имен CORBA. О настройке сервиса имен CORBA на отдельном компьютере см. п. [Запуск и завершение работы сервера Нуволы Аналитика](#)²³².

6.

Установка и удаление модуля Нувола Координатор

6. Установка и удаление модуля Нуволы Координатор

Установку и удаление ПМ **Нуволы Координатор** должен выполнять пользователь с правами администратора `root`.

Порядок установки и настройки ПМ **Нуволы Координатор**:

1. установка компонент платформы Camunda:
 - ❖ установка платформы Camunda BPM на сервер с Apache Tomcat – инсталляционный файл и инструкции по установке на сайте: <http://camunda.com/download/>;
 - ❖ установка приложения Camunda Modeler на рабочую станцию администратора – инсталляционный файл и инструкции по установке на сайте: <http://camunda.com/download/modeler/>;
2. [установка инсталляционного файла Нуволы Координатор](#)^[58] на сервер Нуволы;
3. [настройка платформы Camunda BPM](#)^[59];
4. [настройка конфигурационных файлов Нуволы Координатор](#)^[59];
5. [Настройка скриптов для схем бизнес-процессов](#)^[60].

В настоящем разделе используются следующие обозначения путей по умолчанию:

- ❖ `lancelot_home` – `/usr/local/lancelot/`
- ❖ `bpm_home` – `/usr/local/bpm/`
- ❖ `avalon_home` – `/usr/local/avalon`
- ❖ `CAMUNDA_DIRECTORY` – путь до директории с установленной платформой Camunda BPM.

6.1. Команды для установки и удаления инсталляционного файла

Формат имени инсталляционного файла

Имя инсталляционного файла ПМ **Нуволы Координатор** имеет следующий формат:

```
nuvola-coordinator-X-X.V.e1Z.YYYYYY.rpm,
```

где:

`X-X.V` – номер версии ПМ **Нуволы Координатор** (`V` – номер сборки; может содержать от 1 до 10 цифр и/или букв, а также точку);

`e1Z` – номер версии Linux, для которой предназначен ПМ **Нуволы Координатор** (`e17` – CentOS 7);

`YYYYYY` – обозначение платформы (`x86_64`).

Пример имени файла: `nuvola-coordinator-7-1.73.13479.e17.x86_64.rpm`.

Установка ПМ Нуволы Координатор

Чтобы установить ПМ **Нуволы Координатор**, на сервере Нуволы от имени пользователя `root` следует запустить процедуру инсталляции с помощью команды:

```
# rpm -ivh <install_file_name>
```

где `<install_file_name>` – имя инсталляционного файла.

Например:

```
# rpm -ivh nuvola-coordinator-7-1.73.13479.e17.x86_64.rpm
```

Удаление модуля Нуволы Координатор

Чтобы удалить модуль **Нуволы Координатор**, следует в командной строке ввести:

```
# rpm -e <имя_пакета>
```

Например:

```
rpm -e nuvola-coordinator-7-1.73.13479.x86_64
```

6.2. Настройка платформы Camunda BPM

В настоящем разделе описаны настройки, которые необходимо выполнить после того, как платформа Camunda BPM установлена на сервер и подготовлена к работе.

Далее будем использовать следующую терминологию.

Сервер Координатор – установленная платформа Camunda BPM с настроенным для нее интерфейсом REST Camunda API для взаимодействия с компонентами ПМ **Нуволы Координатор**.

Следует удалить демонстрационный пример camunda-invoice с помощью команды:

```
rm -rf CAMUNDA_DIRECTORY/server/apache-tomcat-<version>/webapps/camunda-invoice
```

6.2.1. Настройка порта

Порт, на котором будет работать сервер Координатор, необходимо указать в файле CAMUNDA_DIRECTORY/server/apache-tomcat-<version>/conf/server.xml. Порт по умолчанию необходимо заменить на требуемый в строке:

```
<Connector port=8080 protocol="HTTP/1.1"
```

6.2.2. Перезапуск сервера Координатор

Для завершения настройки необходимо перезапустить сервер Координатор.

Команда останова сервера:

```
Stop: CAMUNDA_DIRECTORY/server/apache-tomcat-<version>/bin/shutdown.sh
```

Команда запуска сервера:

```
Start: CAMUNDA_DIRECTORY/server/apache-tomcat-<version>/bin/startup.sh
```

Web-приложение Camunda Admin будет доступно примерно через 30 секунд после запуска сервера по пути `http://<server_name>:<camunda_port>/camunda`.

6.3. Настройка конфигурационных файлов Нуволы Координатор

Следует убедиться, что конфигурационные файлы, используемые модулем **Нуволы Координатор**, заполнены верно.

Файл `bpm_home/conf/bpm_camunda.json` содержит информацию о серверах:

```
{
  "url": "localhost:9000", # url сервера camunda bpm
  "mailhost": "mail:4323" # путь для подключения почтового сервера
}
```

6. Установка и удаление модуля Нувола Координатор

Файл `bpm_home/conf/mysql_bpm_camunda.json` содержит данные для подключения к базе данных Camunda BPM:

```
{
  "dbname": "camundabpm",
  "host": "127.0.0.1",
  "drivername": "mysql+pymysql",
  "port": 1906,
  "pw": "PASSWORD",
  "user": "USERNAME"
}
```

Файл `avalon_home/conf/monitoring/http_bpm.conf` содержит настройки rest api:

```
LoadModule wsgi_module modules/mod_wsgi.so

<Directory /usr/local/avalon/wsgi-bin/>
  Require all granted
  WSGIProcessGroup p_rest
  WSGIApplicationGroup %{GLOBAL}
</Directory>

WSGISocketPrefix /var/run/wsgi
WSGIDaemonProcess p_rest user=avalon python-
path=/usr/local/bpm/lib64/python3.4/site-packages/
:/usr/local/avalon/lib64/python3.4/site-packages/
:/usr/local/avalon/lib/python3.4/site-packages:/usr/local/lancelot/python/lib/
:/usr/local/bpm/python/lib/
WSGIScriptAlias /api /usr/local/avalon/wsgi-bin/rest.wsgi

Alias /bpm/ui/ /usr/local/bpm/front/bpm/
Alias /bpm/ui /usr/local/bpm/front/bpm/
Alias /bpm/ /usr/local/bpm/front/

<Directory /usr/local/bpm/front>
AllowOverride None
Options -Indexes
Require all granted
</Directory>
```

6.4. Настройка скриптов для схем бизнес-процессов

В комплектацию ПМ **Нувола Координатор** входят скрипты, предназначенные для использования в схемах бизнес-процессов, создаваемых администратором в приложении Camunda Modeler: [ReportStatus.js](#)^[216], [SendMail.js](#)^[217], [SendToSAC.js](#)^[219].

Для подготовки скриптов к использованию необходимо выполнить следующие настройки:

- ❖ перенести файлы скриптов из папки
`bpm_home/examples/incident-management/scripts/`
в папку
`CAMUNDA_DIRECTORY/server/apache-tomcat-<version>/webapps/engine-rest/WEB-INF/classes/;`
- ❖ в тексте каждого скрипта изменить значение переменной `rest_url` на реальный путь до сервера Нувола.

См. также

[Особенности использования скриптов в схемах бизнес-процессов](#) ²¹⁶

7.

Установка и удаление модуля Нувола Логистика

7. Установка и удаление модуля Нуволы Логистика

Установку и удаление модуля **Нуволы Логистика** должен выполнять пользователь с правами администратора `root`.

Формат имени инсталляционного файла

Имя инсталляционного файла модуля **Нуволы Логистика** имеет следующий формат:

```
nuvola-logistics-X-X.V.elZ.YYYYYY.rpm,
```

где:

`X-X.V` – номер версии модуля **Нуволы Логистика** (`V` – номер сборки; может содержать от 1 до 10 цифр и/или букв, а также точку);

`elZ` – номер версии Linux, для которой предназначен модуль **Нуволы Логистика** (`el7` – CentOS 7);

`YYYYYY` – обозначение платформы (`x86_64`).

Пример имени файла: `nuvola-logistics-7-1.74.13349.el7.x86_64.rpm`.

Установка модуля Нуволы Логистика

Чтобы установить модуль **Нуволы Логистика**, на сервере Нуволы от имени пользователя `root` следует запустить процедуру инсталляции с помощью команды:

```
# rpm -ivh <install_file_name>
```

где `<install_file_name>` – имя инсталляционного файла.

Например:

```
# rpm -ivh nuvola-logistics-7-1.74.13349.el7.x86_64.rpm
```

Удаление модуля Нуволы Логистика

Чтобы удалить модуль **Нуволы Логистика**, следует в командной строке ввести:

```
# rpm -e <имя_пакета>
```

Например:

```
rpm -e nuvola-logistics-7-1.74.13349.x86_64
```

8.

Настройка и использование модуля Нувола Топаз

8. Настройка и использование модуля Нуволы Топаз

Исполняемые утилиты модуля **Нуволы Топаз** находятся в директории `/usr/local/lancelot/bin`. Далее сделаны следующие предположения:

- ❖ установлено соединение с сервером от имени пользователя `avalon`;
- ❖ указанная выше директория прописана в переменной `PATH`, или пользователь перешел в эту директорию;
- ❖ на сервер [установлен](#)^[43] модуль **Нуволы Топаз**.

В ПМ **Нуволы Топаз** следует работать под пользователем `avalon`, а не под `root`. Переключение пользователя осуществляется с помощью команды `su - avalon`.

8.1. Запуск и остановка ПМ Нуволы Топаз

При установке ПМ **Нуволы Топаз** автоматически запускаются его системные процессы. Однако при необходимости администратор может выполнить запуск и остановку ПМ **Нуволы Топаз** вручную.

Настройка проверки целостности кода

При запуске ПМ **Нуволы Топаз** в процессе его установки целостность кода не проверяется. Такая проверка может производиться только при ручном запуске ПМ **Нуволы Топаз** (см. ниже).

По умолчанию режим проверки целостности кода отключен. Для его включения администратору (после установки ПМ **Нуволы Топаз**) следует выполнить команду:

```
kommander sign lock
```

В дальнейшем при попытке запустить вручную ПМ **Нуволы Топаз** с нарушенной целостностью кода появляется сообщение и запуск прерывается.

Запуск и остановка ПМ Нуволы Топаз под Linux

Для запуска ПМ **Нуволы Топаз** следует под пользователем `root` выполнить команду:

```
systemctl start nuvola
```

Для остановки ПМ **Нуволы Топаз** следует выполнить команду:

```
systemctl stop nuvola
```

Для перезапуска ПМ **Нуволы Топаз** следует использовать команду:

```
systemctl restart nuvola
```

8.2. Настройки для администрирования пользователей

Основной функционал по управлению [пользователями](#)^[36] ПК Нуволы (настройка групп пользователей и учетных записей пользователей и т.п.) реализован в [Web-интерфейсе](#)^[100]. В настоящем разделе описаны вспомогательные настройки и действия, связанные с администрированием пользователей, выполняемые через командную строку.

8.2.1. Настройка дополнительных параметров учетной записи

В Web-интерфейсе в окне ввода и изменения учетной записи блок **Дополнительная информация** (после установки ПК Нувола) содержит следующие поля для ввода справочных сведений о пользователе: **Доб. номер, Эл. почта, Моб. тел., Тел., Должность**. Набор отображаемых полей настраивается в конфигурационном файле `/usr/local/lancelot/conf/profile.conf` и при необходимости может быть изменен администратором.

Конфигурационный файл `profile.conf`

Конфигурационный файл `/usr/local/lancelot/conf/profile.conf` представляет собой словарь, в котором ключами являются имена атрибутов в LDAP (без "L-", если такая группа символов имеется в названии). Первоначальный (после установки ПК Нувола) текст файла:

```
my $cfg = {
  title => {
    displayName => "title",
    show => 1,
    editable => 1
  },
  mail => {
    displayName => "mail",
    show => 0,
    editable => 1
  },
  telephoneNumber => {
    displayName => "telephoneNumber",
    show => 1,
    editable => 1
  },
  ipPhone => {
    displayName => "ipPhone",
    show => 1,
    editable => 1
  },
  mobile => {
    displayName => "mobile",
    show => 1,
    editable => 1
  }
};
```

Каждый атрибут в файле описывается параметрами:

- ❖ `displayName` – имя атрибута, используемое для выбора записи из файла локализации `locale.json`:
 - Δ `/usr/local/lancelot/var/lancelot-data/locales/monitoring/ru` – директория хранения файла русской локализации;
 - Δ `/usr/local/lancelot/var/lancelot-data/locales/monitoring/en/locale.json` – директория хранения файла английской локализации.

Наименования полей, отображаемые в блоке **Дополнительная информация**, задаются в файлах локализации в секции `User Permissions` (в конце файла). Если в нужном файле локализации соответствующая запись отсутствует, то в качестве наименования поля в блоке **Дополнительная информация** используется значение параметра `displayName` из соответствующей секции файла `profile.conf`;

- ❖ `show` – признак отображения поля в блоке **Дополнительная информация**:

8. Настройка и использование модуля Нуволы Топаз

0 – поле не выводится,

1 – поле выводится;

- ❖ `editable` – возможность редактировать значение поля в блоке **Дополнительная информация**:

0 – значение не доступно для редактирования,

1 – значение доступно для редактирования.

Пример добавления дополнительного параметра

Чтобы добавить поле **Подразделение** в блок **Дополнительная информация** в окне ввода и изменения учетной записи, следует:

1. в конфигурационный файл `/usr/local/lancelot/conf/profile.conf` добавить секцию с параметрами атрибута `department`:

```
department => {
  displayName => "department",
  show => 1,
  editable => 1
}
```

2. в файл локализации `/usr/local/lancelot/var/lancelot-data/locales/monitoring/ru/locale.json` в секцию `User Permissions` (в конце файла) добавить запись для атрибута `department` с его отображаемым именем на русском языке:

```
"department": "Подразделение"
```

8.2.2. Настройка политик безопасности

Политики безопасности, применяемые при внутренней аутентификации пользователей, хранятся в базе данных LDAP. Администратор (пользователь `root`) может создавать и изменять политики безопасности, а также просматривать перечень существующих политик посредством командной строки.

На сервере Нуволы в директории `/usr/local/lancelot/data` хранятся файлы-заготовки с параметрами политики безопасности (поставляются с модулем **Нуволы Топаз**):

- ❖ `policyModify.ldif` – для изменения существующей политики безопасности;

- ❖ `policyAdd.ldif` – для создания новой политики безопасности.

Для создания или изменения политики безопасности администратору следует откорректировать файл и отправить соответствующую команду в базу данных LDAP. Подробности ниже.

Просмотр перечня существующих политик безопасности

Чтобы получить перечень существующих политик безопасности, следует выполнить команду:

```
ldapsearch -vv -e ppolicy -x -D 'cn=Manager,dc=lancelot' -w secret -h localhost -p 1989 -b "ou=policies,dc=lancelot" dn
```

Пример возможного ответа (наименования политик безопасности подчеркнуты):

```
ldap_initialize( ldap://localhost:1989 )
filter: (objectclass=*)
requesting: dn
# extended LDIF
#
# LDAPv3
# base <ou=policies,dc=lancelot> with scope subtree
# filter: (objectclass=*)
# requesting: dn
#
```

```
# policies, lancelot
dn: ou=policies,dc=lancelot

# admin, policies, lancelot
dn: cn=admin,ou=policies,dc=lancelot

# default, policies, lancelot
dn: cn=default,ou=policies,dc=lancelot

# strict, policies, lancelot
dn: cn=strict,ou=policies,dc=lancelot

# test, policies, lancelot
dn: cn=test,ou=policies,dc=lancelot

# search result
search: 2
result: 0 Success

# numResponses: 6
# numEntries: 5
```

Создание политики безопасности

Для создания политики безопасности следует:

1. в директории `/usr/local/lancelot/data` изменить файл `policyAdd.ldif`, указав в нем нужные параметры создаваемой политики безопасности; детальное описание параметров см. ниже;
2. выполнить команду:

```
/usr/local/avalon/bin/ldapadd -h localhost -p 1989 -x -D
'cn=Manager,dc=lancelot' -w 'secret' -f policyAdd.ldif
```

Изменение существующей политики безопасности

Для изменения существующей политики безопасности следует:

1. в директории `/usr/local/lancelot/data` изменить файл `policyModify.ldif`, указав в нем нужные параметры политики безопасности; детальное описание параметров см. ниже;
2. выполнить команду:

```
/usr/local/avalon/bin/ldapmodify -vv -x -D 'cn=Manager,dc=lancelot' -w secret -
h localhost -p 1989 -f policyModify.ldif
```

Параметры файлов `policyAdd.ldif` и `policyModify.ldif`

Ниже приведены примеры строк файлов `policyAdd.ldif` и `policyModify.ldif` с описаниями параметров, которые администратор может изменить. В примерах значения таких параметров подчеркнуты.

ВНИМАНИЕ! Администратору разрешено изменять только значения параметров, перечисленных ниже. После значения параметра не должно быть пробелов.

```
dn: cn=admin,ou=policies,dc=lancelot
```

Название политики безопасности. Оно должно совпадать с названием в следующей строке.

```
cn: admin
```

Название политики безопасности. Должно совпадать с названием в предыдущей строке.

8. Настройка и использование модуля Нувола Топаз

`pwdMaxAge: 0`

Срок действия пароля (в секундах) с момента последнего изменения. **0** – пароль бессрочный.

`pwdExpireWarning: 324000`

Временной интервал (в секундах), за который до истечения срока действия пароля пользователю выдается предупреждение. **0** – предупреждение не выдается.

`pwdInHistory: 0`

Количество старых паролей, которые должны храниться в истории в базе данных LDAP.

`pwdMaxFailure: 3`

Максимальное количество попыток ввода пароля. Если при такой настройке, как в приведённом примере, пользователь введет неправильный пароль 3 раза подряд, то учетная запись может быть заблокирована.

`pwdLockout: TRUE`

Необходимость блокировки пользователя при достижении лимита на количество попыток ввода пароля (`pwdMaxFailure`). Возможные значения : **TRUE, FALSE**.

`pwdLockoutDuration: 120`

Временной интервал (в секундах), на который блокируется учетная запись, если достигнут лимит на количество попыток ввода пароля (`pwdMaxFailure`) и блокировка разрешена (`pwdLockout`).

`pwdMinLength: 5`

Минимальное количество символов в пароле.

8.2.3. Создание глобальных политик и прав доступа

ПК Нувола поставляется с набором [глобальных политик](#)^[37] и [прав доступа](#)^[37], созданных разработчиками. Если предустановленных настроек недостаточно для реализации проектного решения, то администратор может создать кастомизированные глобальные политики и права доступа.

Общий порядок настройки:

1. в конфигурационном файле сервера конфигурации `/usr/local/avalon/etc/permissions.conf` описать новые глобальные политики и права доступа;
2. в кастомизированном конфигурационном файле модуля **Нувола Топаз** `/usr/local/lancelot/custom/conf/permissions.conf` описать новые web-запросы или переопределить существующие в соответствии с новыми глобальными политиками и правами доступа;
3. [перезапустить сервер конфигурации](#)^[77];
4. с помощью утилиты [overlord.pl](#)^[261] перезапустить системные процессы [rest-api](#)^[279], [konfigd](#)^[277].

Конфигурационный файл сервера конфигурации

В конфигурационном файле `/usr/local/avalon/etc/permissions.conf` хранятся описания создаваемых глобальных политик.

8.2. Настройки для администрирования пользователей

Файл содержит 4 блока, каждый из которых начинается именем блока и заканчивается тегом END. Между этими тегами необходимо ввести данные. Строки вне блоков можно использовать для комментариев.

Блоки содержат пары "ключ=значение":

Имя блока	Ключ	Значение
PERMISSIONS	Системное имя права доступа	Номер бита (от 1 до 12) данного права доступа в виде числа в шестнадцатеричной системе счисления. Уникальный в пределах файла.
POLICIES	Системное имя глобальной политики	Номер бита (от 1 до 12) данной политики в виде числа в шестнадцатеричной системе счисления. Уникальный в пределах файла.
SHORT_NAMES	Системное имя глобальной политики или права доступа на путь	Буквенный код (из одной или нескольких букв) для краткого обозначения глобальной политики или права доступа в окне Управление пользователями
DISPLAY_NAMES	Системное имя глобальной политики или права доступа на путь	Наименование глобальной политики или права доступа, отображаемое в окне Управление пользователями

Конфигурационный файл модуля Нувола Топаз

Модуль **Нувола Топаз** используют предустановленный конфигурационный файл `/usr/local/lancelot/conf/permissions.conf`. Он содержит объект в формате json, описывающий зависимости "web-запрос – глобальная политика" и "web-запрос – право доступа". Поля `actionCommands` и `policyCommands` хранят пары "ключ : значение", где ключ – web-запрос или его кодовое имя, а значение – системное имя глобальной политики или права доступа.

Для настройки применения новых глобальных политик и прав доступа необходимо в директории `/usr/local/lancelot/custom/conf` нужно создать кастомизированный конфигурационный файл `permissions.conf`. В его полях `actionCommands` и `policyCommands` следует описать новые зависимости "web-запрос – глобальная политика" и "web-запрос – право доступа" и/или переопределить существующие – из предустановленного конфигурационного файла `/usr/local/lancelot/conf/permissions.conf`.

Для модуля **Нувола Топаз** настройки в кастомизированном файле `/usr/local/lancelot/custom/conf/permissions.conf` приоритетнее по отношению к аналогичным настройкам из предустановленного файла `permissions.conf`.

При составлении текста кастомизированного файла можно использовать предустановленный файл `/usr/local/lancelot/conf/permissions.conf`.

Пример

На примере продемонстрируем порядок создания новой глобальной политики и права доступа.

1. Конфигурационный файл `/usr/local/avalon/etc/permissions.conf` следует заполнить:

```
PERMISSIONS
new_permission=0x1
END
```

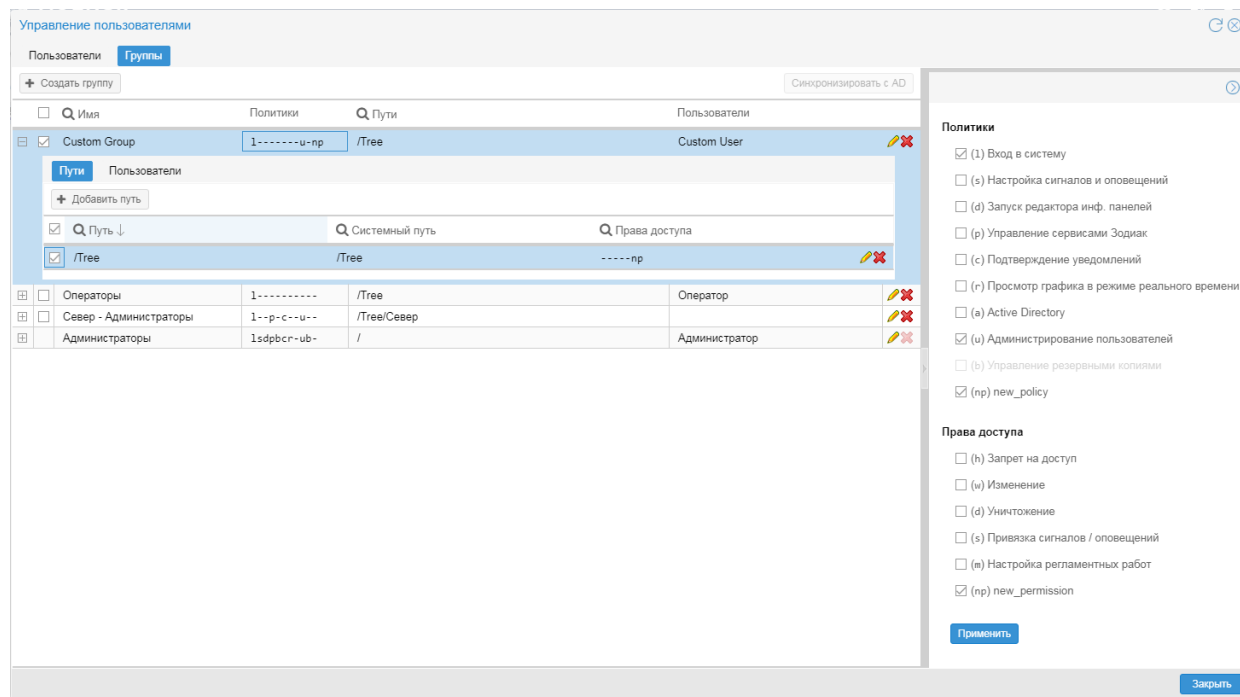
8. Настройка и использование модуля Нувола Топаз

```
POLICIES
new_policy=0x1
END

SHORT_NAMES
new_permission=np
new_policy=np
END

DISPLAY_NAMES
new_permission=new_permission
new_policy=new_policy
END
```

2. Перезапустить сервер конфигурации, выполнив от имени пользователя `root` команду:
`systemctl restart nuvola-configServer`
3. Убедиться, что новая глобальная политика и новое право доступа добавлены. Для этого в окне **Управление пользователями** перейти на вкладку **Группы** и удостовериться, что в правом окне в блоках **Группы** и **Права доступа** отображаются флаги с наименованиями добавленных глобальной политики и разрешенного действия.
4. Создать группу пользователей с именем **Custom Group** и глобальными политиками **Вход в систему**, **Администрирование пользователей** и **new_policy**. Добавить в группу путь `/` с новым правом доступа **new_permission**.



5. Прикрепить к группе **Custom Group** нового пользователя **Custom**.
6. Создать кастомизированный конфигурационный файл `/usr/local/lancelot/custom/conf/permissions.conf` следующего содержания:

```
{
"actionCommands":{
"node_GET" : "new_permission"
},
"policyCommands":{
"useradd_POST" : "new_policy"
}
}
```


Данные правила означают, что навигация по дереву и создание новых пользователей доступна только пользователям состоящим в группе, которая имеет право доступа по пути `new_permission` и глобальную политику `new_policy`.

7. Перезапустить системный процесс `rest_api`, выполнив от имени пользователя `avalon` команду:

```
overlord.pl restart rest_api
```
8. Убедиться в том, что правила работают. Для этого войти в Web-интерфейс под пользователем `root` и убедиться, что дерево больше не отображается, а в окне **Управление пользователями** при попытке создать учетную запись выдается сообщение об ошибке, связанной с недостатком прав. В то же время, при работе в web-интерфейсе под учетной записью `Custom` дерево конфигурации отображается корректно и ограничений на создание пользователей нет.

8.2.4. Просмотр лог-файлов о действиях пользователей

Лог-файлы с информацией о действиях пользователей хранятся в директории `usr/local/lancelot/var/lancelot-logs` (в числе прочих [лог-файлов](#)^[74] модуля **Нувола Топаз**).

Журнал подключений к серверу Нувола

Попытки подключения к серверу Нувола через Web-интерфейс фиксируются в лог-файле `auth.log`.

Для каждой попытки подключения записывается:

- ❖ время;
- ❖ способ подключения (`rest_api_worker` – через Web-интерфейс);
- ❖ IP-адрес пользователя;
- ❖ имя пользователя;
- ❖ статус (результат) подключения.

Пример лог-файла:

```
[04-Dec-2018 20:06:38:4349 rest_api_worker] From: 10.43.3.33, User: oper2, Login: login failed.  
[07-Dec-2018 15:19:56:4347 rest_api_worker] From: 10.43.3.33, User: oper, Login: login failed.  
[07-Dec-2018 15:20:26:4349 rest_api_worker] From: 10.43.3.33, User: guest, Login: login failed.  
[07-Dec-2018 15:20:33:4348 rest_api_worker] From: 10.43.3.72, User: guest, Login: login failed.
```

8.3. Системные процессы ПМ Нувола Топаз

Функционирование ПМ **Нувола Топаз** и ПК Нувола в целом обеспечивают системные процессы, отвечающие за выполнение различных задач: сбор данных с объектов контроля и обработку полученной информации, построение информационных панелей, создание отчетов, поддержку работы пользовательского Web-интерфейса, контроль целостности конфигурации, интеграцию компонент ПК Нувола и др.

Системный процесс – это процесс (`daemon`) на сервере Нувола, запускаемый с определенными значениями параметров. **Процесс** (`daemon`) является встроенным объектом ПМ **Нувола Топаз**. Системные процессы создаются и настраиваются администратором в зависимости от задач, решаемых с помощью ПК Нувола.

8. Настройка и использование модуля Нуволы Топаз

При создании системного процесса администратор указывает процесс (демон), вводит имя системного процесса и задает значения параметров для запуска процесса. Имя системного процесса может как совпадать с именем процесса, так и отличаться от него. Запуск нескольких системного процесса, в основе которых лежит один и тот же процесс (демон), фактически означает запуск нескольких экземпляров процесса. Информация о настроенных системных процессах хранится в базе данных конфигурации.

При инсталляции ПМ **Нуволы Топаз** создается и запускается первоначальный набор системных процессов.

Управление системными процессами осуществляется следующими средствами:

- ❖ из командной строки с помощью утилиты [overlord.pl](#)^[261];
 - ❖ с помощью приложения [Диспетчер процессов](#)^[156] в Web-интерфейсе.
- Примечание. За контроль работы системных процессов и их автоматическое восстановление в случае сбоя отвечает служебный системный процесс [monit](#)^[278].

8.3.1. Лог-файлы

Лог-файл системного процесса модуля **Нуволы Топаз** – это текстовый файл, в котором в соответствии с настроенным уровнем протоколирования регистрируются действия, выполняемые системным процессом, ошибки и предупреждения. По умолчанию лог-файлы хранятся на сервере Нуволы в директории `/usr/local/lancelot/var/lancelot-logs`.

Формат записи лог-файла:

```
[dd-Mon-yyyy hh:mm:ss :pid] <текст>
```

См. [Пример лог-файла системного процесса kollector](#)^[276].

Настройка лог-файла

Имя и содержание лог-файла настраивается с помощью параметров системного процесса:

`loglevel`

Уровень логирования – степень детализации информации в лог-файле. Возможные значения:

Info – базовая информация (наиболее эффективный уровень);

Debug – детальная информация (применяется на этапе разработки/внедрения);

Warn – информация о предупреждениях;

Error – информация об ошибках.

`logfile`

Директория и имя лог-файла для записи отладочной информации. Не рекомендуется изменять директорию хранения лог-файла по умолчанию `/usr/local/lancelot/var/lancelot-logs/`. Системный процесс создает отдельный файл для каждого потока программы с указанием в имени файла номера PID программы, порядкового номера потока и номера файла в данном потоке.

Для изменения этих параметров системного процесса следует пользоваться утилитой [overlord.pl](#)^[261].

Ротация лог-файлов

Ротация лог-файлов системного процесса модуля **Нуволы Топаз** – это одновременное закрытие и переименование текущего лог-файла и создание нового лог-файла.

Ротация лог-файлов выполняется автоматически в соответствии с настройками, заданными в конфигурационном файле `<service_name>.conf` для конкретного системного процесса. Этот файл хранится на сервере Нуволы в директории `/usr/local/lancelot/conf/lancelot_logrotate` и содержит, в частности, следующие параметры:

- ❖ период ротации (обычно ежедневно);
- ❖ максимально допустимый размер лог-файла (обычно 100 Мб);
- ❖ количество хранимых закрытых лог-файлов (обычно 1).

Ротация выполняется по окончании временного интервала и при условии, что лог-файл достиг заданного размера (обычно 100 Мб). В результате в имя текущего лог-файла добавляется цифра 1 – `<service_name>.log.1`, и создается новый лог-файл со стандартным именем `<service_name>.log`. Если в конфигурационном файле предусмотрено хранение двух закрытых лог-файлов, то в имени первого закрытого лог-файла цифра 1 изменяется на цифру 2 – `<service_name>.log.2`. Таким образом, при каждой ротации формируется новый лог-файл и перезаписывается хранимый лог-файл.

Пользователь / администратор может выполнить принудительную ротацию лог-файлов с помощью команды `rotate` утилиты [overlord.pl](#)^[261].

8.3.2. Статусы системных процессов

Системный процесс может находиться в одном из трех статусов.

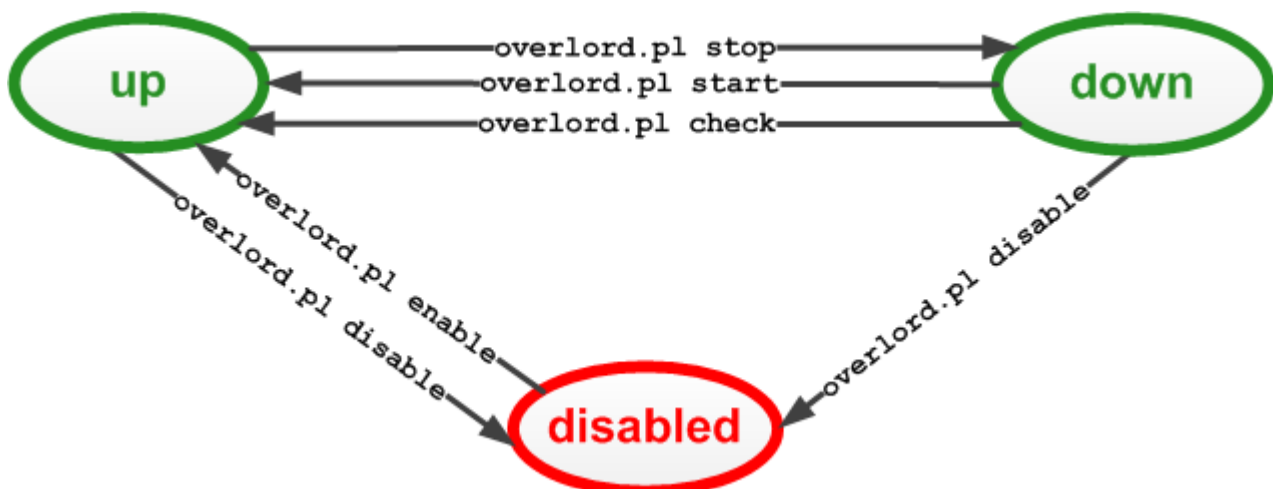
Статус **up** – системный процесс запущен.

Статус **down** – системный процесс остановлен, но доступен для ручного или автоматического запуска. Из этого состояния системный процесс может быть запущен как вручную – командами `start` и `check` утилиты [overlord.pl](#)^[261], так и автоматически – служебным сервисом `monit`.

Статус **disabled** – системный процесс отключен и не доступен для автоматического запуска. Отключенные сервисы игнорируются служебным сервисом `monit`, а также командами `start` и `check` утилиты [overlord.pl](#)^[261]. Перевод сервиса в статус **disabled** и вывод из него осуществляется администратором вручную с помощью команд `disable` и `enable` утилиты `overlord.pl`.

Системные процессы в статусах **up** и **down** имеют параметр `disabled=false`, системные процессы в статусе **disabled** – параметр `disabled=true`.

На диаграмме показаны команды утилиты [overlord.pl](#)^[261], приводящие к изменению статуса системного процесса. Работающий системный процесс можно также перезапустить, т.е. последовательно остановить, а затем запустить. Команда перезапуска (`overlord.pl restart`) на диаграмме не показана во избежание загромождения.



Например, чтобы отключить остановленный системный процесс `fnmp` (т.е. сервис в статусе **down**), администратору необходимо выполнить команду:

```
overlord.pl disable fnmp
```

8. Настройка и использование модуля Нуволы Топаз

В результате статус системного процесса `fnmp` изменится с **down** на **disabled**, значение параметра `disabled` изменится с `false` на `true`.

Для возобновления работы системного процесса `fnmp` администратору необходимо выполнить команду:

```
overlord.pl enable fnmp
```

В случае успешного завершения команды системный процесс `fnmp` будет запущен, т.е. его статус изменится с **disabled** на **up**, а значение параметра `disabled` изменится с `true` на `false`.

8.3.3. Используемые порты

Порты модуля **Нуволы Топаз**, заданные по умолчанию:

Номер порта	Процесс	Описание порта
UDP:161	<code>snmpd</code>	Стандартный порт для SNMP-агента.
UDP:162	<code>snmptrapd</code>	Стандартный порт для SNMP Trap.
TCP:80	<code>Apache</code>	Поддержка Web-интерфейса (запросы HTTP).
TCP:443	<code>Apache</code>	Доступ к WEB по протоколу HTTPS.
TCP:1906	<code>Mysqld</code>	Используется ядром СУБД MySQL.
TCP:1919	<code>server</code>	Порт для репликации данных.
TCP:1921	konfigd ^[277]	Поддержка удалённых API.
TCP:1989	<code>client</code>	Используется для LDAP-сеансов, при работе с каталогом LDAP.
TCP:2809	<code>omniNames</code>	Используется репозиторием <code>omniNames</code> .
TCP:9001	licdaemon ^[277]	Поддержка запросов о лицензии.
TCP:xxxxx	<code>lancelotHdbServer</code>	При запуске модуля Нуволы Топаз <code>omniNames</code> произвольно назначает номер порта для <code>lancelotHdbServer</code> , в диапазоне от TCP:1025 до TCP:65535, уже занятые порты не используются.

8.4. Утилиты ПМ Нуволы Топаз

Администратор осуществляет управление работой ПК Нуволы с помощью утилит. Команды, предоставляемые утилитами ПМ **Нуволы Топаз**, позволяют администратору создавать дерево конфигурации, настраивать сигналы и оповещения, контролировать работу системных процессов ПМ **Нуволы Топаз**, а также выполнять ряд операций, связанных с администрированием ПК Нуволы.

Исполняемые утилиты модуля **Нуволы Топаз** находятся в директориях: `/usr/local/lancelot/bin` и `/usr/local/lancelot/util`.

Если не указано особо, утилиты рекомендуется выполнять от имени пользователя `avalon`.

Подробнее утилиты описаны в справочнике [Утилиты ПМ Нуволы Топаз](#)^[248].

8.5. Настройка и обслуживание базы данных конфигурации

База данных конфигурации включает в себя дерево конфигурации, шаблоны объектов, настройки сигналов и оповещений, информацию о системных процессах ПМ **Нуволы Топаз**.

Дерево конфигурации – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления.

8.5.1. Сервер конфигурации

В модуле **Нуволы Топаз** в качестве хранилища данных конфигурации используется сервер, реализованный средствами LDAP. Для ускорения работы системных процессов реализован сервер конфигурации `configServer`, использующий кэш данных конфигурации.

Сервер конфигурации `configServer` – системный процесс, обеспечивающий оптимальный (быстрый) доступ системных процессов модуля **Нуволы Топаз** к данным конфигурации. Функционирование сервера конфигурации `configServer` обеспечивается процессом `lancelotHdbServer`.

Сервер конфигурации запускается автоматически при запуске модуля **Нуволы Топаз**.

В момент установки модуля **Нуволы Топаз** регистрируется служба `nuvola-configServer`, позволяющая управлять сервером конфигурации посредством `systemd` операционной системы.

Команда для проверки статуса сервера конфигурации:

```
systemctl status nuvola-configServer
```

Команда для запуска сервера конфигурации:

```
systemctl start nuvola-configServer
```

Команда для перезапуска сервера конфигурации:

```
systemctl restart nuvola-configServer
```

Команда для остановки сервера конфигурации:

```
systemctl stop nuvola-configServer
```

Аутентификация пользователей посредством сервера LDAP

При необходимости можно использовать аутентификацию пользователей ПК Нуволы посредством протокола LDAP. Для настройки такой возможности следует отредактировать файл `/usr/local/lancelot/conf/ldap_auth.conf`, заполнив следующие параметры:

- ❖ `'hostname'` – адрес контроллера домена Active Directory;
- ❖ `'port'` – порт подключения к контроллеру домена, по умолчанию 3268;
- ❖ `'domein'` – полное доменное имя.

Способ аутентификации пользователя указывается в его учетной записи.

8.5.2. Резервное копирование и восстановление конфигурации

Управление резервными копиями базы данных конфигурации ПК Нуволы осуществляется посредством утилиты [backup](#)^[251] в ручном и автоматическом режиме. Настройки утилиты `backup` задаются в конфигурационном файле `/usr/local/lancelot/conf/backup.conf`.

Автоматический режим настраивается через планировщик Linux. В соответствии с настройками по умолчанию ежедневно автоматически в 01:01 создается резервная копия базы данных конфигурации, и в 01:30 удаляются устаревшие резервные копии.

Администратор может выполнить следующие действия:

- ❖ изменить настройки конфигурационного файла [backup.conf](#)^[78];
- ❖ с помощью утилиты [backup](#)^[251]:
 - [создать и удалить резервную копию](#)^[79];
 - [очистить директорию хранения от устаревших резервных копий](#)^[79];
 - [установить/снять защиту резервной копии от удаления](#)^[80];
 - [восстановить базу данных конфигурацию из резервной копии](#)^[80].

При создании резервной копии автоматически создается файл контрольной суммы, который используется для проверки целостности данных при восстановлении. Действия утилиты `backup`, выполняемые как в ручном, так и автоматическом режиме, распространяются на оба файла: файл резервной копии и связанный с ним файл контрольной суммы.

8.5.2.1. Параметры резервного копирования (конфигурационный файл `backup.conf`)

Настройки механизма резервного копирования, используемые утилитой `backup`, хранятся в конфигурационном файле `/usr/local/lancelot/conf/backup.conf`. После установки ПК Нуволы файл содержит следующие настройки:

```
storefolder=/home/avalon/nuvola-backups
```

Директория, в которой создаются и хранятся файлы резервных копий.

```
period=90
```

Период (в днях), по истечении которого резервная копия считается устаревшей и подлежит удалению командой [cleanup](#)^[79].

```
folders_to_backup="lancelot/bin lancelot/conf lancelot/etc lancelot/data  
lancelot/es lancelot/js lancelot/webui lancelot/lib lancelot/util lancelot/scripts  
lancelot/var/lancelot-data/alerts lancelot/var/openldap-data/config avalon/cgi-bin  
avalon/conf avalon/etc avalon/htdocs lancelot/var/lancelot-data/scenarios  
lancelot/custom"
```

Перечень директорий, которые попадают в резервную копию.

```
folders_to_restore="lancelot/etc lancelot/var avalon/etc/openldap/schema"
```

Перечень директорий для [стандартного восстановления](#)^[80] базы данных конфигурации из резервной копии.

```
custom_folders_to_restore="lancelot/etc lancelot/var avalon/etc/openldap/schema  
lancelot/custom lancelot/webui/custom"
```

Перечень директорий для [нестандартного восстановления](#)^[80] базы данных конфигурации из резервной копии (может использоваться на этапе внедрения ПК Нувола).

```
exclusions="status.overlord"
```

Перечень файлов и папок, которые не попадают в резервную копию. Можно использовать регулярное выражение.

8.5.2.2. Действия с резервными копиями

В настоящем разделе описаны действия, которые администратор может осуществлять с помощью утилиты `backup`.

Создание резервной копии

Команда для создания резервной копии базы данных конфигурации:

```
kommander backup save <backup_name>
```

В качестве имени `<backup_name>` рекомендуется вводить дату и время создания резервной копии в формате `YYYY-MM-DD_HHMM`.

Вместе с файлом резервной копии создается файл контрольной суммы, например:

```
2016-10-25-0101 – файл резервной копии;  
.2016-10-25_0101.chk – файл контрольной суммы.
```

Команда для получения списка сохранённых резервных копий:

```
kommander backup list
```

См. также Создание резервной копии через Web-интерфейс.

Удаление резервной копии

Возможно удаление только незащищенных резервных копий, т.е. не помеченных признаком `safe`. Вместе с файлом резервной копии автоматически удаляется и соответствующий ему файл контрольной суммы.

Команда для удаления резервной копии с заданным именем `<backup_name>`:

```
kommander backup delete <backup_name>
```

См. также Удаление резервной копии через Web-интерфейс.

Очистка директории хранения от устаревших резервных копий

Под очисткой понимается удаление из директории хранения устаревших незащищенных резервных копий. Другими словами, в результате выполнения процедуры очистки из директории, заданной параметром `storefolder` в файле [backup.conf](#)^[78], удаляются файлы резервных копий (и связанные с ними файлы контрольных сумм), которые:

- ❖ старше срока, заданного параметром `period` в файле [backup.conf](#)^[78],
- ❖ не помечены признаком `safe`.

Команда для очистки директории хранения:

```
kommander backup cleanup
```

Защита резервных копий от удаления

Для файла резервной копии можно установить признак защиты от удаления – `safe`. Такие защищенные файлы не удаляются командами `cleanup` и `delete`.

Команда для установки признака защиты от удаления на резервную копию:

```
kommander backup safe <backup_name>
```

Команда для снятия признака защиты от удаления с резервной копии:

```
kommander backup unsafe <backup_name>
```

См. также Защита резервных копий через Web-интерфейс.

8.5.2.3. Восстановление базы данных конфигурации из резервной копии

База данных конфигурации, восстановленная из резервной копии, может не содержать объектов контроля, по которым на момент восстановления имелись собранные данные. Такие объекты будут отсутствовать в дереве конфигурации, сбор данных по ним осуществляться не будет, однако ранее собранные данные останутся.

При восстановлении резервной копии осуществляется проверка соответствующего файла контрольной суммы. Контрольная сумма резервной копии должна совпадать с контрольной суммой, вычисляемой для текущей версии ПК Нувола, иначе – восстановление резервной копии отклоняется. При несовпадении контрольных сумм возможно только принудительное восстановление без проверки контрольной суммы.

Команды для восстановления базы данных конфигурации из резервной копии:

- ❖ стандартное восстановление базы данных конфигурации (в соответствии с параметром `folders_to_restore` конфигурационного файла [backup.conf](#)^[78]):

```
kommander backup restore <backup_name>
```

8.5.3. Обновление ПМ Нувола Топаз

Перед обновлением ПМ Нувола Топаз настоятельно рекомендуется [сохранить резервную копию данных](#)^[78].

Чтобы обновить ПМ Нувола Топаз из RPM, следует выполнить действия:

1. удалить установленный RPM:

```
rpm -e nuvola-topaz
```

При этом не будут удалены конфигурация и данные. Появится сообщение о том, что папка не пуста;

2. установить новый RPM:

```
rpm -ihv nuvola-topaz-X-X.X.elZ.YYYYYY.rpm
```

См. также

[Формат имени инсталляционного файла](#)^[43]

8.6. Настройка дерева конфигурации

Дерево конфигурации – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления.

Администратор может настраивать дерево конфигурации следующими способами:

- ❖ через командную строку, а именно с помощью утилиты [konfne](#)^[255];
- ❖ через [Web-интерфейс](#)^[100].

Автоматическая настройка сетевых устройств

ПМ **Нувола Топаз** может автоматически распознавать различные типы сетевых устройств и применять для их конфигурирования подходящий шаблон объекта. В большинстве случаев автоматическое распознавание представляет собой добавление нового оборудования. Недостатком автоматического распознавания можно считать невозможность указать для утилиты `konfne` специфические опции соответствующего устройства. Функция автоматического распознавания не применима к виртуальным объектам.

Чтобы добавить в дерево устройство с использованием автоматического распознавания типа, следует выполнить команду:

```
konfne --autotype --ip <ip> --community <secret> cfg /Tree/Routers/myrouter
```

Просмотр справки о параметрах устройства

Чтобы просмотреть справку об индивидуальных параметрах конкретного объекта, следует использовать команду:

```
konfne help <Devices Name in Нувола>
```

где `<Devices Name in Нувола>` – это название соответствующего шаблона объекта. Например:

```
konfne help Devices::Routers::Cisco
```

Загрузка объектов из файла (пакетный режим)

Для добавления в дерево конфигурации нескольких новых объектов можно использовать пакетный режим. Он предусматривает загрузку параметров объектов из файла, который предварительно создан и размещен на сервере Нувола. Пакетный режим не применим для добавления учетных записей.

Загружаемый файл должен содержать строки следующего формата:

```
<folder>,<IP_address>,<community>,[<device_type>]
```

- ❖ `folder` – папка в дереве конфигурации или папка и имя устройства. Трактовка подстроки `folder` при загрузке файла зависит от наличия символа слеш «/» в последней позиции:
 - Δ если подстрока заканчивается слешем, то она воспринимается как путь на папку в дереве конфигурации. Например, подстрока `/Tree/batch/` будет воспринята как путь на папку `/Tree/batch` для размещения устройства; в этом случае при загрузке файла следует использовать функцию получения имени с устройства (`-f`);
 - Δ если в конце подстроки слеш отсутствует, то слово после последнего слеша трактуется как имя устройства, а часть подстроки до последнего слеша воспринимается как путь на папку в дереве конфигурации. Например, из подстроки `/Tree/batch/ServerLinux` будет выделено имя устройства `ServerLinux` и путь на папку `/Tree/batch`;
- ❖ `IP_address` – IP-адрес устройства;
- ❖ `community` – сообщество SNMP устройства;
- ❖ `device_type` – соответствующий шаблон объекта в стандартном формате, например, `Devices::Routers::Cisco`. Если шаблон объекта не указан, то при загрузке файла следует использовать функцию автоматического распознавания (`-a`).

Пример содержания файла `devices.txt` для загрузки:

```
#####
#config_file_begin
/Tree/batch/,10.42.3.13,public,
```

8. Настройка и использование модуля Нувола Топаз

```
/Tree/batch/,10.42.0.31,public,  
#config_file_end  
#####
```

Чтобы загрузить файл, следует выполнить следующие действия:

1. разместить файл на сервере Нувола, например, поместить файл `devices.txt` в директорию `/usr/local/lancelot`;
2. выполнить команду:

```
konfne -a -f --script /usr/local/lancelot/devices.txt
```

В результате загрузки файла `devices.txt` (см. пример выше) в папку `/Tree/batch` дерева конфигурации будут добавлены два устройства под своими системными именами.

Получение списка шаблонов объектов

Список всех шаблонов объектов можно посмотреть, запустив команду `konfne --devlist`. Например:

```
konfne --devlist  
konfne 1.1 Nuvola Topaz version 6.1 built on lnx32build at Mon Sep 15 12:50:55 MSK  
2014 by root features: ALL locale: ru  
Currently supported devices are:  
    Devices::Access::Lucent  
    Devices::Access::MPLS  
    ...
```

8.7. Создание шаблонов объектов

Создание шаблонов объектов в ПМ **Нувола Топаз** заключается в создании модуля Perl, который реализует объект, компиляции этого модуля с помощью команды `kompile` и конфигурировании шаблона объекта с помощью команды `konfne`.

Для создания более сложных шаблонов объектов рекомендуется применять профессиональные сервисы ООО «Венто Технологии».

Используемый в [примере](#)^[83] шаблон объекта это принтер HP. Имя шаблона объекта для этого принтера: `Devices::Printers::HP`, а модуль Perl, который реализует этот шаблон объекта, находится в директории `/usr/local/lancelot/lib/Devices/Printers/HP.pm`.

Следует отметить, что для разработки и отладки модулей шаблонов объектов для ПМ **Нувола Топаз** необходимы навыки работы с Perl.

8.7.1. Правила создания нового шаблона объекта

Набор правил, которые необходимо выполнять при создании нового шаблона объектов (`*.pm`):

- ❖ Запрещается в одном файле несколько раз использовать секции с одинаковыми типами (`dictionary name`) и названиями (`instance name`). Например: `target --default--` может использоваться в файле только один раз.
- ❖ Запрещается в одной секции дважды использовать атрибут с одинаковым именем.
- ❖ В именах секций запрещается использовать следующие символы: `","`, `"+"`, `""`, `"\"`, `"<"`, `">"`, `";"`, `"#"`, `"="`.
- ❖ В именах атрибутов можно использовать только латинские буквы и символ `"-"`, все остальные символы приведут к ошибке.

- ❖ Имена секций и ссылки на них должны быть чувствительны к регистру символов. На данный момент они не чувствительны к регистру, поэтому нельзя использовать похожие имена (например: Name и nAme).
- ❖ Имена атрибутов должны быть написаны прописными символами (верхний регистр).
- ❖ Имена атрибутов, должны быть из списка атрибутов перечисленных в файле `avalon.schema`, и данный атрибут должен находиться в соответствующем `objectClass`. Если данного атрибута нет в файле, то его необходимо добавить в файл секцией:

```

attributetype ( 1.3.6.1.4.1.12251.9999.1.5808
  NAME ( 'L-console-params' 't5808' )
  DESC 'console-params'
  EQUALITY caseIgnoreMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
)

```

Где:

5808 – уникальный номер атрибута;

'L-console-params' – имя атрибута в LDAP базе,

'console-params' – имя атрибута в файлах конфигурации HDB сервере.

8.7.2. Пример шаблона объекта HPPrinter

Модуль, отвечающий за работу с устройством, состоит из трех частей:

- [Device Header \(Заголовок объекта\)](#)^[84]
- [Device Program \(Программа объекта\)](#)^[84]
- [Device DATA \(Данные объекта\)](#)^[86]

В качестве примера приведён код Perl, который реализует работу с принтером HP. Цифры слева обозначают номера строк кода.

По завершении написания конфигурационного файла шаблона объекта следует скомпилировать его с помощью команды `kompile`. Если компиляция прошла успешно, надо проверить, что объект сконфигурирован правильно в базе данных модуля **Нувола Топаз** с помощью следующей команды:

```
konfne -devlist
```

Это не гарантирует, что устройство работает правильно. Это только показывает, что соответствующий объект сконфигурирован правильно. Окончательная проверка – это просмотр устройства и его интерфейсов в Web-интерфейсе модуля Нувола Топаз.

8.7.2.1. Device Header (Заголовок объекта)

Device Header модуля Perl, который реализует работу с принтером:

```
1.   package Devices::Printers::HP;
2.   use base 'Devices::NetCom::IfMib';
3.   use strict 'vars';
4.   use Common::OID;
5.   use Server::Edit;
6.   use snmpUtils;
7.   use Data::Dumper;
8.   use Common::Log;
9.   use Common::Util;
10.  $Devices::enterprises{11} = "Devices::Printers::HP";
```

Строка 1: Первая строка раздела заголовка должна содержать определение пакета (package). Имя пакета должно совпадать с именем шаблона объекта, который требуется реализовать. В данном случае это: `Devices::Printers::HP`. Напомним, что общий вид имени шаблона объекта: `Devices::<Type>::<Module>`.

Строка 2: Директива `use` определяет использование необходимых подпрограмм и параметров из базового класса. Например, если требуется использовать шаблоны интерфейсов трафика в новом объекте, и известно, что объект имеет MIB, именуемый IF-MIB, можно вызвать функцию `ifMib` без параметров; система возьмет необходимые параметры из базового класса `/lancelot/lib/Devices/NetCom` и сконфигурирует необходимые интерфейсы трафика.

Строки 3-9: Это объявления модулей Perl. Можно будет использовать подпрограммы и параметры из этих модулей для реализации данного модуля.

Строка 10: Эта строка определяет номер производителя определенного устройства для облегчения идентификации устройства во время конфигурирования.

Другими словами, система будет знать, как получить доступ к определенному устройству на основании его номера производителя.

```
$Devices::enterprises{'enterprise number'} = Devices::<Type>::<Module>;
```

Или, в данном случае:

```
$Devices::enterprises{11} = 'Devices::Printers::HP';
```

8.7.2.2. Device Program (Программа объекта)

Device Program модуля Perl, который реализует принтер:

```
11.  sub cfg_impl {
12.  my ($self) = @_;
13.
14.  my $snmp = "$$self{community}@$$self{ip}";
15.  my $path = "/Devices/Printers/HP";
16.  print ">>> Configuring HP Printer Device ... \n";
17.  my $ldesc = $$self{sysDescr};
18.  $self->put_defaults($path, 'long-desc' => $ldesc );
19.  $self->add_target($path, "st ats", {
20.                                'target-type'    => 'prn-HP-stats',
21.                                'short-desc'     => 'Status',
22.                                'display-name'   => 'Status',
23.                                'order'         => '1000',
24.  });
25.  $self->cfg_ifmib;
```

```

26.     return 1;
27.   }
28.   sub describe {
29.     <<"EOD";
30.     HP Printers
31.     The device supports HP printers with a network card. In additi on
32.     to information about network interface, it also shows statistics
33.     about paper usage.
34.     EOD
35.   }
36.   1;

```

Строка 11: Каждый модуль шаблона объекта должен содержать подпрограмму `cfg_impl`. Эта подпрограмма используется для конфигурирования или обновления объекта. В системе мониторинга **Нувола Топаз** данная подпрограмма получает объект со всеми необходимыми параметрами и дескрипторами.

Строка 12: Параметр `$self` – это объект устройства со всеми присущими дескрипторами, которые унаследованы от вызывающего оператора этой функции.

Строка 14: Первый параметр, который должен быть объявлен – это `$snmp`. Следует использовать именно такую строку, как в приведённом примере.

Строка 15: Второй параметр, который должен быть объявлен – это `$path`, он должен соответствовать имени пакета, определенному ранее. Например: мы назвали пакет `Devices::Printers::HP`, и, следовательно, `$path = /Devices/Printers/HP`.

Строка 16: Директива `print` будет использоваться во время конфигурирования объекта. Она служит для вывода информации.

Строка 17: Описание устройства может быть извлечено из объекта `$self`.

Строка 18: Во время конфигурирования модуль **Нувола Топаз** должен создать файл с именем `targets` в поддиректории `etc/lancelot-config/Device/`; файл содержит параметры для процесса `collector` модуля **Нувола Топаз**. Этот файл `targets` по умолчанию имеет заголовок и несколько параметров:

```

target --default--
device-name          = <имя устройства, которое надо отслеживать>
long-desc            = <Длинное описание устройства>
snmp-community       = public
snmp-host            = <IP адрес устройства>

```

Для создания файла `targets` в поддиректории `etc/lancelot-config/Device/` используется следующий вызов функции `put_defaults`:

```
$self->put_defaults($path, 'long-desc' => $ldesc);
```

Значение `<IP>` берется из команды `konfne`. Следует обратить внимание, что этот вызов также устанавливает параметр `long-desc` в файле `targets`.

Строка 19: Следующий шаг – это вызов функции `add_target`. Функция принимает три параметра: `$path`, `interface-name` (имя интерфейса) и хэш свойств для интерфейса. Например:

```

$self->add_target($path, "stats", {
  'target-type' =>'prn-HP-stats',
    'short-desc' =>'Status',
  'display-name' =>'Status',
'order'        => '1000',
});

```

Функция `add_target` выполняет следующие операции:

8. Настройка и использование модуля Нуволы Топаз

- функция обновляет файл с именем `targets` в директории `etc/lancelot-config/Tree/` (точное расположение файла берется из команды `konfne`), и на основе параметра `interface-name` добавляет директиву `link-target` в файл. Например:

```
target stats
    link-target = /Devices/Printers/HP/<IP>/stats
```

где `<IP>` берется из команды `konfne`;

- функция обновляет файл с именем `targets` в директории `etc/lancelot-config/$path/<IP>` в поддиректории `Devices/` и, для определенного интерфейса `stats`, добавляет хэш свойств интерфейса в файл. В данном примере следующие строки добавляются к файлу `targets`:

```
targetstats
display-name = "Status"
order       = 1000
short-desc  = "Status"
target-type = prn-HP-stats
```

Эта информация необходима для процесса `kollector` модуля **Нуволы Топаз**. Кроме того, здесь описывается информация, которая будет отображаться на соответствующей странице Web-интерфейса **Нуволы Топаз**. Следует отметить, что при каждом вызове `add_target` должен указываться хэш параметров интерфейса, как в примере:

- ▶ `target-type` – уникальное имя, которое будет использоваться в разделе `DATA` для дальнейшего описания объекта;
- ▶ `display-name` – имя интерфейса в Web-интерфейсе **Нуволы Топаз**;
- ▶ `order` – порядок, в котором интерфейс будет отображаться на Web-странице. Интерфейс с самым высоким значением приоритета будет отображаться первым. Интерфейс с самым низким значением будет отображаться последним.

`interface-name` для функции `add_target` должен быть уникальным. Возможен случай, когда одно устройство имеет несколько интерфейсов одного типа. Например, машина Linux с двумя процессорами. И, хотя существует один `interface-type`, а именно CPU, все равно следует определить два отдельных интерфейса: по одному для каждого процессора. Другими словами, необходимо вызвать функцию `add_target` дважды, с разными значениями `interface-name`. Установившейся практикой является добавление индекса к имени, например, присвоить имена интерфейсов `cpu1` и `cpu2`.

Следует отметить, что функция `add_target` может принимать дополнительный параметр `inst`. Это будет описано в описании части `Device DATA`.

Строка 25: `$self->cfg_ifmib` автоматически добавляет целевые объекты, связанные с интерфейсами трафика.

Строка 26: Возвращает значение или индицирует успешное завершение функции `cfg_impl`.

Строки 28-35: Функция `describe` предоставляет сообщение для устройства. Сообщение может быть отображено с помощью команды `konfne`. Например:

```
konfne help Devices::Printers::HP
```

Строка 36: Отмечает конец пакета Perl и завершает конфигурацию шаблона объекта. Следующая часть – `Device DATA` – используется командой `kcompile`.

8.7.2.3. Device DATA (Данные объекта)

Часть `Device DATA` обрабатывается командой `kcompile`. Часть `Device DATA` следует за частью `Device Program` и состоит из следующих блоков:

- блок `Target` (цель)
- блок `OID`
- блок `DataSource` (источник данных)

- блок Target-Type (тип целевого объекта)
- блок Graphic (график)

Ниже приведён пример Device DATA для принтера HP.

```

37.  _DATA_
38.  ### Defaults-Section Devices/Printers/HP ###
39.  Target --default--
40.      computer          = %auto-target-name%
41.      snmp-host         = %computer%
42.      target-type       = undef
43.  OID prnHPonLine      1.3.6.1.4.1.11.2.3.9.1.1.2.1.0
44.  OID prnHPpaperInMin  1.3.6.1.4.1.11.2.3.9.1.1.2.8.0
45.  datasource --default--
46.      rrd-ds-type       = GAUGE
47.      rrd-min           = 0
48.      rrd-max           = undef
49.  datasource prnHPonLine
50.      ds-source         = snmp://%snmp%/prnHPonLine
51.  datasource prnHPpaperInMin
52.      ds-source         = snmp://%snmp%/prnHPpaperInMin
53.      rrd-ds-type       = GAUGE
54.  targetType prn-HP-stats
55.      ds                 = "prnHPonLine,prnHPpaperInMin"
56.      view               = "OnLine: prnHPonLine,
57.                          PaperSpeed:prnHPpaperInMin"
58.  graph prnHPonLine
59.      color              = green
60.      legend             = "On-Line Status"
61.      draw-as            = AREA2
62.      y-min              = 0
63.      scale              = 0,EQ,100,*
64.
65.  graph prnHPpaperInMin
66.      color              = green
67.      legend             = "Paper speed"
68.      draw-as            = AREA2
69.      y-min              = 0

```

Строка 37: Определяет начало Раздела DATA.

Строка 38: Во время конфигурирования модуль **Нуволы Топаз** создаст файл с именем `default` в директории `etc/lancelot-config/Devices/Printers/HP/` и скопирует в него раздел DATA.

Строки 39-42: `Targetdefault` должна быть записана, как указано в примере.

```

Target --default--
computer          = %auto-target-name%
snmp-host         = %computer%
target-type       = undef

```

Оба параметра (`%auto-target-name%` и `%computer%`) получаются во время выполнения процесса `kollector` модуля **Нуволы Топаз**. Параметр `target-type` не определен, поэтому необходимо определить его позже для каждого типа интерфейса.

Параметры `targetdefault` наследуются последующими определениями целевых объектов.

8. Настройка и использование модуля Нувола Топаз

Строки 43-44: Эти две строки определяют OID для принтера. Каждое устройство должно иметь определение OID, которое предоставляется производителем устройства. Записи OID в строках 42-43 служат в качестве макроопределений, поэтому можно использовать имена OID вместо длинных последовательностей цифр.

Имена OID, используемые в данном примере, это: `prnHPonLine` и `prnHPpaperInMin`.

Следует обратить внимание, что OID индивидуален для каждого интерфейса. Другими словами, каждый интерфейс имеет собственный OID.

Строки 45-48: После определения целевых объектов и OID переходим к определению источников данных. Источники данных необходимы для построения RRD-файлов объекта. RRD-файл – это бинарный файл, который содержит данные, собранные процессом `collector` модуля **Нувола Топаз**, и каждый интерфейс объекта ассоциируется со своим RRD-файлом. Значения по умолчанию для источника данных:

```
datasource --default--
rrd-ds-type      = GAUGE
rrd-min          = 0
rrd-max         = undef
```

`rrd-ds-type` – может принимать одно из следующих значений: `GAUGE`, `INTEGER` или `DERIVE`.

Значение `rrd-ds-type` для определенного OID может быть установлено с помощью команды `snmp walk`.

`rrd-min` и `rrd-max` определяют диапазон значений OID, собранных протоколом SNMP. Чтобы установить неограниченное значение, используйте `undef`.

Следующим шагом является установка источника данных для каждого OID.

Строки 49-50: Это определение источника данных для OID `prnHPonLine`.

```
Datasource prnHPonLine
  ds-source = snmp://%snmp%/prnHPonLine
```

Общая форма определения `ds-source`: `snmp://%snmp%/<OID-name>`.

Обратите внимание, что значения `rrd` типа: `rrd-ds-type`, `rrd-min`, `rrd-max` наследуются из исходного источника данных.

Строки 51-53: Это определение источника данных для OID `prnHPpaperInMin`.

```
Datasource prnHPpaperInMin
  ds-source = snmp://%snmp%/prnHPpaperInMin
rrd-ds-type = GAUGE
```

Обратите внимание, что в строке 52 назначается значение `rrd-ds-type`. В данном примере оно идентично значению по умолчанию. Однако в примере эта строка оставлена, чтобы показать, что значения `rrd` могут быть изменены.

Параметр `snmp:/` используется для запроса SNMP версии v1. Для запроса SNMP версии v2c следует писать `snmp2:/`. Например:

```
ds-source = snmp2://%snmp%/cpqHoCpuUtilMin.%inst%
```

Параметр `.%inst%` будет присоединен к определению `ds-source`, если результат SNMP запроса с данным OID является таблицей, но нужно только одно значение для данного интерфейса. Например, определение объекта `Compaq`:

```
ds-source = snmp://%snmp%/cpqHoCpuUtilMin.%inst%
```

Значение параметра `.%inst%` для каждого интерфейса берется из хэша свойств каждого интерфейса. Функция `add_target`, в программном разделе, для объекта `Compaq`:

```
foreach my $index (keys %cpus) {
    $self->add_target($path, "cpu$index" => {
```



```

        'show-bandwidth'    => 'false',
        'short-desc'       => "CPU $index",
        'target-type'      => 'compaq-cpu-info',
'inst'
        => $index,
    });
}

```

Строки 54-57: Следующий шаг – это назначение типов целей для каждого целевого объекта, определенной функцией `add_target`, в программном разделе. Типы целей указывают коллектору, какие источники данных использовать и как отображать их на Web-странице.

```

targetTypeprn-HP-stats
ds
    = "prnHPonLine,prnHPpaperInMin"
view
    = "OnLine: prnHPonLine,
PaperSpeed: prnHPpaperInMin"

```

`ds` – определяет список источников данных, которые будут использоваться. Не вставляйте пробелы в этот список.

`view` – определяет какие графики отображать на экране. Каждый график, который планируется отобразить, должен быть связан с текстовой меткой, которая действует как гиперссылка; нажатие кнопки мыши на этой текстовой метке отобразит график. Не следует использовать одну и ту же текстовую метку для различных графиков.

В примере с принтером на экране появятся две гиперссылки справа от названия интерфейса: `Online` и `PaperSpeed`. При нажатии на гиперссылку `Online` вызывается график `prnHPonLine`, а при нажатии на гиперссылку `PaperSpeed` вызывается график `prnHPpaperInMin`.

Строки 58-63: Эти строки описывают, как строить график `prnHPonLine`. Общее описание графика должно соответствовать синтаксису и правилам, показанным в примере:

```

Graph prnHPonLine
color      = green
legend    = "On-Line Status"
draw-as   = AREA2
            y-min = 0
scale     = 0,EQ,100,*

```

Имя графика `prnHPonLine` должно фигурировать в определении `targetType`. Обязательные поля:

- `color` – определяет цвет графика. Это может быть один из следующих цветов: `green` (зеленый), `red` (красный), `blue` (синий), `cyan` (голубой), `magenta` (пурпурный), `yellow` (желтый), `dark-red` (темно-красный), `dark-blue` (темно-синий), `cornflower` (васильковый), `lightpurple` (светло-пурпурный), `pink` (розовый), `purple` (пурпурный), `deep-blue` (глубокий синий), `brown` (коричневый), `black` (черный), `neogrn` (неоновый зеленый), `skyblue` (небесно-голубой), `orange` (оранжевый), `burntorng` (огненный оранжевый), `purply` (с пурпурным оттенком), `white` (белый);
- `legend` – название графика;
- `draw-as` – форма графика. Может принимать одно из следующих значений:
 - ▶ `AREA`, `AREA2` – обе формы заполняют область графика выбранным цветом;
 - ▶ `STACK` – нарисованная область находится над другим графиком, если такой график существует; в противном случае ведет себя как `AREA`;
 - ▶ `LINE1`, `LINE2` и `LINE3` – все три формы рисуют линию различной толщины. `LINE1` соответствует самой малой толщине, а `LINE3` соответствует самой большой толщине.

Опциональные поля:

- `y-min` – минимальное значение для оси Y графика;
- `scale` – определяет алгоритм масштабирования. См. [Масштабирование графиков](#)^[92].

8. Настройка и использование модуля Нуволы Топаз

Строки 65-69: Эти строки описывают, как строить график `prnHPpaperInMin`. Общее описание графика должно соответствовать такому же синтаксису и правилам, как и для графика `prnHPonLine`, выше.

8.7.3. Правила написания шаблона объекта

Далее приведены правила, которые следует принимать во внимание при написании шаблона объекта. Несоблюдение этих правил может привести к некорректной работе Системы Нуволы и/или к серьезному замедлению в сборе данных.

При написании нового шаблона объекта следует помнить, что в Системе Нуволы используется два метода сбора данных:

- напрямую, то есть путём прямого обращения к устройствам по протоколу SNMP;
- сбор данных посредством модуля **Нуволы Приамус**, к которому сервис сбора данных обращается по протоколу FNMP.

8.7.3.1. Получение информации с удалённого хоста, `snmpget`

В шаблоне объекта НЕ следует ИСПОЛЬЗОВАТЬ функцию `snmpUtils::get`, а также всё семейство `snmpUtils::*`, так как эти функции напрямую обращаются к устройству по SNMP. В шаблоне объекта следует использовать `$self->snmp_get` и семейство `snmp_*`, описанное в `NetCom.pm` (`snmp_get`, `snmp_get_table` и т.д.). Эти функции по наличию либо отсутствию тега `fnmp` определяют используемый метод сбора данных (напрямую или посредством модуля **Нуволы Приамус**, соответственно).

Примеры правильной настройки получения информации с удалённого объекта:

```
lib/Devices/Servers/Citrix.pm:111:
my(%procTable) = $self->snmp_get_table('1.3.6.1.4.1.3845.3.1.1.12.1.1');
lib/Devices/Servers/Citrix.pm:113:
my $procName=$self->snmp_get("1.3.6.1.4.1.3845.3.1.1.12.1.2.$index");
```

8.7.3.2. Использование функции обработки `func`

Если в шаблоне объекта при описании сбора данных напрямую используется функция обработки (`func`), то для обеспечения правильного сбора данных посредством модуля **Нуволы Приамус** следует использовать функцию `fnmp-oid-cb`.

При необходимости дополнительной обработки результатов, полученных от источника данных, используемого в новом шаблоне объекта, следует указать в этом шаблоне объекта имя функции, которая будет вызвана для вычисления значения от источника данных. Имя функции указывается в шаблоне объекта, раздел `datasource`, тег `ds-source` для сервиса сбора данных, использующего SNMP, и тег `fnmp-oid-cb` для сервиса сбора данных, использующего FNMP. Кроме того:

- для сервиса сбора данных, использующего FNMP, необходимо указывать список OID, значения которых будут переданы в функцию `func`. См. строку 435 в приведённом ниже примере;
- если задано правило мапинга, то это правило будет также применено для заданных OID;
- в теге `ds-source` необходимо указывать ключевое слово `func`: перед именем функции;
- в теге `fnmp-oid-cb` нельзя указывать ключевое слово `func`: перед именем функции.

Пример правильной настройки функции обработки:

```
lib/Devices/Computers/Compaq.pm:433:
datasource cpqHoFileSysSpaceFree
lib/Devices/Computers/Compaq.pm:434:
```

```
ds-source = func:Devices::Computers::Compaq::HddFree(%inst%,%snmp%)
lib/Devices/Computers/Compaq.pm:435:
fnmp-oid-cb =
1.3.6.1.4.1.232.11.2.4.1.1.3,1.3.6.1.4.1.232.11.2.4.1.1.4,Devices::Computers::Compa
q::HddFreeFnmp(%snmp%)
```

8.7.3.3. Функция обработки целевого объекта collector-hook

При наличии отдельной функции обработки данных целевого объекта `collector-hook`, используемой при сборе данных напрямую, при настройке сбора данных посредством модуля **Нувола Приамус** следует воспользоваться функцией `fnmp-collector-hook` для указания обработчика и задать список OID с помощью `fnmp-collector-hook-oids`.

Пример правильной настройки функции обработки целевого объекта:

```
lib/Devices/Computers/Dagonet.pm:536:
collector-hook = Devices::Computers::Dagonet::SoftProber::soft_property_fetch
lib/Devices/Computers/Dagonet.pm:537:
fnmp-collector-hook =
Devices::Computers::Dagonet::SoftProber::soft_property_fetch_FNMP
lib/Devices/Computers/Dagonet.pm:538:
fnmp-collector-hook-oids = 1.3.6.1.4.1.12251.27.1.1,1.3.6.1.4.1.12251.27.1.2
```

8.7.3.4. Правила маппинга

При использовании сложного маппинга при сборе данных напрямую, по протоколу SNMP, используется конструкция, состоящая из `base-oid` и `match`. Для обеспечения правильного маппинга при сборе данных посредством модуля **Нувола Приамус** следует дополнительно использовать конструкцию `fnmp-select`.

Пример правильной настройки сложного маппинга:

```
map storage-descr
base-oid = hrStorageDescr
match = %storage-descr%
fnmp-select = "WHERE table(1.3.6.1.2.1.25.2.3.1.3) eq $storage-descr"
```

Операторы маппинга

Правило маппинга представляет собой правило выбора конкретного атрибута из списка значений. Операторы, используемые при создании правила маппинга для сбора данных посредством модуля **Нувола Приамус**, приведены ниже:

- `'...'` – заданное значение параметра;
- `[0]` (`COND_INST`) – значение первого добавленного числа в строке OID. Например, если запрашивалась таблица с OID `1.3.6.1.2.1.2.2.1`, то результатом будут значения с OID `1.3.6.1.2.1.2.2.1.2.1`. Значение `2.1` – это часть, добавленная к исходно заданному OID. Значение параметра `[0] = 2`, а значение параметра `[1] = 1`.
- `=` – проверка равенства полученных значений и заданного значения для типа `INTEGER`;
- `~` – сравнить строковые значения (стандартный формат строки);
- `eqs` – проверяет наличие подстроки A в строке B (строка в формате `lstr`).
- `eq` – сравнивает строковые значения A и B. Строки должны быть полностью идентичны (строка в формате `lstr`).
- `table(OID)` – получить таблицу значений для заданного OID;

8. Настройка и использование модуля Нувола Топаз

- AND, OR – логические операции выполнения условий;
- MAX[0] – выбор OID с максимальным значением числа в добавленной части OID с номером 0, нумерация слева направо. Например, если запрашивалась таблица с OID 1.3.6.1.2.1.2.2.1, то результатом будут значения с OID 1.3.6.1.2.1.2.2.1.1.1...1.3.6.1.2.1.2.2.1.23.4. Значение 1.1...23.4 – это часть, добавленная к исходно заданному OID. При MAX[0] будет выбран первый OID с максимальным значением 23.

Примеры правил:

```
table(1.3.6.1.2.1.25.2.3.1.1) = 'disk-name'
```

```
table(1.3.6.1.2.1.31.1.1.1.18) ~ 'interface-name'
```

```
'disk-name' eqs table(1.3.6.1.2.1.25.2.3.1.3)
```

```
table(1.3.6.1.2.1.25.4.2.1.2) eq 'process-name' AND table(1.3.6.1.2.1.25.4.2.1.5) eq 'process-args'
```

```
table(1.3.6.1.4.1.9.9.166.1.6.1.1.1, vec(table(1.3.6.1.4.1.9.9.166.1.5.1.1.2, vec([0], [0]))) eq 'policy-name' -AND table(1.3.6.1.4.1.9.9.166.1.7.1.1.1, vec(table(1.3.6.1.4.1.9.9.166.1.5.1.1.2, vec([0], [1]))) eq 'class-name'
```

8.7.4. Масштабирование графиков

Графики **Нувола Топаз** строятся по данным бинарных файлов – RRD-файлов. Данные RRD-файлов – значения, сохранённые процессом `collector` модуля **Нувола Топаз**. Они соответствуют данным, которые изменялись с течением времени (где время откладывается по оси X).

Для отображения и масштабирования данных RRD-файла необходимо управлять значениями, зависящими от времени. Один из алгоритмов масштабирования **Нувола Топаз** использует RPN-синтаксис.

Для создания сложных функций масштабирования, которые не могут быть получены с помощью синтаксиса RPN, следует проконсультироваться с представителями ООО «Венто Технологии».

Алгоритм масштабирования RPN

Например, рассмотрим обычное математическое выражение $a+b*c$. Оно читается как: «к результату умножения $b*c$ прибавить a ». Эквивалентное RPN выражение имеет вид $a,b,c*,+$, что вычисляется на основе стека следующим образом:

1. поместить 'a' наверх `stack[1]`;
2. поместить 'b' наверх `stack[2]`;
3. поместить 'c' наверх `stack[3]`;
4. извлечь значения 'b' и 'c';
5. вычислить $p = b*c$ и поместить 'p' наверх `stack[2]`;
6. извлечь значения 'a' и 'p';
7. вычислить $r = a + p$;
8. поместить r наверх `stack[1]`;
9. закончить выражение, вернуть значение `stack[1]`.

Простое RPN выражение: $10,*$. Это выражение говорит алгоритму масштабирования умножить на 10 каждое значение, полученное из RRD файла.

Выражение для масштабирования для принтера HP, приведённого в примере, будет: $0,EQ,100,*$, что означает осуществление следующих шагов над каждым значением из RRD файла:

1. поместить значение **RRD** наверх `stack[1]`;

2. поместить **0** наверх **stack[2]**;
3. извлечь **stack[1]** и **stack[2]**;
4. выполнить функцию **EQ**, **p = equal (stack[1], stack[2])**. Обратите внимание, что **p** имеет значение **1**, если значение **RRD= 0**;
5. поместить **'p'** наверх **stack[1]**;
6. поместить **100** наверх **stack[2]**;
7. извлечь **stack[1]** и **stack[2]**;
8. вычислить **p = stack[1] * stack[2]**;
9. поместить **p** наверх **stack[1]**;
10. закончить выражение, вернуть значение **stack[1]**.

Конечный результат выражения для масштабирования будет: **0,EQ,100,***, то есть преобразовать каждое нулевое значение RRD в 100.

Модуль **Нувола Топаз** поддерживает следующие операторы в выражениях RPN:

- **EQ** – равно;
- **GT** – больше чем;
- **GE** – больше или равно;
- **LE** – меньше или равно;
- **LT** – меньше чем.

8.7.5. Универсальный плагин объектов

Реализация универсального плагина объектов осуществляется DLL-модулем `LctVCustom.dev`. Этот модуль используется аналогично модулям, которые существуют для каждого шаблона объектов. Существенное отличие заключается в следующем: этот модуль может поддерживать различные шаблоны объектов и динамически формировать пользовательские диалоги, используемые для настройки параметров объекта, в соответствии со свойствами конкретного устройства. Информация о каждом из этих устройств может быть предоставлена сервером, но теперь все данные сохраняются в файлах формата XML. Эти XML-файлы, по одному для каждого из объектов, поддерживаемых модулем, хранятся в определенном каталоге, и в первый раз могут быть созданы вручную. Для этого модуля реализован механизм проверки правильности ввода: производится проверка на наличие непустых значений в строках, а также правильность формата введённого IP-адреса. В случае ввода неправильного значения текст становится красным, а кнопка сохранения/обновления отключена.

8.7.5.1. Использование универсального плагина объектов

При добавлении нового шаблона объекта можно вместо создания нового проекта использовать универсальный плагин объектов. Для реализации нового шаблона объекта следует выполнить следующие действия:

1. Создать XML-файл для нового объекта, соответствующий описанию, приведённому в [Формат XML-файла](#)^[94]. Имя этого файла должно иметь расширение `.xml` и соответствовать шаблону объекта, где все пары символов двоеточия ":" заменены на символ точки ".". То есть если объект имеет шаблон `devices::category::name`, имя `.xml` файла должно быть `devices.category.name.xml`.
2. Создать файл значка файла с тем же именем, но с расширением `.ico`. Файл значка для примера, приведённого выше, должен быть назван `devices.category.name.ico`. В этом файле должны быть иконки для нового объекта с размерами 16x16 и 32x32 пикселя.
3. Оба файла следует положить в директорию `<ApplicationDirectory>/Devices`.

8. Настройка и использование модуля Нувола Топаз

Также существует возможность заменить существующие объекты, если требуется изменить их функциональность. Для этого следует выполнить шаги, описанные выше, и после этого удалить файл `.dev` этого объекта из директории приложения `<ApplicationDirectory>/Devices`.

8.7.5.2. Формат XML-файла

Для работы универсального плагина объектов требуется наличие двух файлов, `.ico` и `.xml`. Первый из этих файлов содержит иконки объекта, второй представляет собой модуль объекта в XML-формате. Ниже описаны правила создания XML-файла.

Файл должен содержать теги `header`, `device` и хотя бы один тег `dialog` внутри тега `device`. Тег `dialog` должен содержать теги `control`, это описание элементов управления. Тег `control` может содержать теги `value` и `dependency`.

Теги, которые могут быть или должны быть использованы, с их родительскими тегами

Тег	Родительский тег	Возможное число тегов
<code>device</code>	-	1
<code>dialog</code>	<code>device</code>	1 или больше
<code>control</code>	<code>dialog</code>	1 или больше
<code>value</code>	<code>control</code>	0, 1 или больше
<code>dependency</code>	<code>control</code>	0, 1 или больше

Возможные атрибуты тегов

Атрибут	Тег	Значения	Требуется	Значение по умолчанию	Описание
<code>type</code>	<code>device</code>	<code>a::b::c</code>	✓		Шаблон объекта.
<code>version</code>	<code>device</code>	1	✓		Версия универсального плагина объектов (номер текущей версии "1").
<code>icon</code>	<code>device</code>			<code>###.ico</code>	Имя файла иконки объекта.

Атрибут	Тег	Значения	Требуется	Значение по умолчанию	Описание
category	device	{a-b-c-d-e}	V		Идентификатор CLSID категории, которая содержит этот объект (для списка в Web-интерфейсе).
description	device		V		Информация об объекте, отображаемая в Web-интерфейсе.
title	device		V		Заголовок объекта, отображаемый в списке устройств в Web-интерфейсе.
nickname	device		V		Краткое название объекта, используемое в Web-интерфейсе.
status	device	notused, used		used	Если установлено значение notused , то объект может быть скрыт в списке устройств.
tag	dialog		V		Уникальный идентификатор диалога.
name	dialog		V		Имя диалога.
main	dialog	true, false		false	Отображается ли диалог при запуске.
type	control	edit, checkbox, combobox	V		Тип элемента управления.
tag	control		V		Уникальный идентификатор элемента управления.
label	control		V		Описание элемента управления, выводимое как текст около текстового поля ввода или около флажка.

8. Настройка и использование модуля Нувола Топаз

Атрибут	Тег	Значения	Требуется	Значение по умолчанию	Описание
validation	control	ip, notempty			Тип проверки содержимого элемента управления. По умолчанию проверка не производится.
default	control				Значение по умолчанию, используемое для только что созданных объектов.
password	control	yes,no		no	Отображение текста с заменой символов звёздочками (*).
data	value		Только для типа combobox		Выбирается одно значение из списка возможных.
tag	dependency		V		Идентификатор элемента управления, который может выполнить действие над данным элементом (например, сделать его доступным или недоступным).
value	dependency	true/false etc.	V		Значение/состояние элемента управления, указанного в tag.
kind	dependency	disable	V		Тип действия, выполняемого над текущим элементом управления, когда элемент tag принимает value.
voffset	dependency	<integer>		0	Отступ элемента управления от первоначального положения по вертикали.
hoffset	dependency	<integer>		0	Отступ элемента управления от первоначального положения по горизонтали.

Атрибут	Тег	Значения	Требуется	Значение по умолчанию	Описание
distance	dependency	<integer>			Расстояние между элементом управления и его описанием (label). По умолчанию все элементы управления выравниваются по вертикали.
revert	dependency	true/false		false	Обеспечение возможности откатить результат kind элемента управления (например, снова сделать его доступным).

Дополнительные ограничения:

- `validation` – этот тег используется для предотвращения попыток пользователя сохранить неверные данные. Если хотя бы у одного из элементов управления недопустимое значение, его метка отображается красным цветом, кнопка для сохранения/обновления объекта с выбранными значениями параметров отключается. Проверка `IP` означает, что содержимое элемента управления должно включать в себя допустимый IP-адрес. Проверка `notempty` позволяет предотвратить использование пустых значений элементов управления. По умолчанию элементы управления не проверяются. Проверка `host` ограничивает длину строки до 255 символов и используется для имен хостов.
- `value` – этот тег используется только для типа `combobox`.
- `password` – отображает вводимое в поле значение с заменой символов звёздочками `"*`". Используется только для типа `edit`.

Пример структуры XML-файла и результат его работы

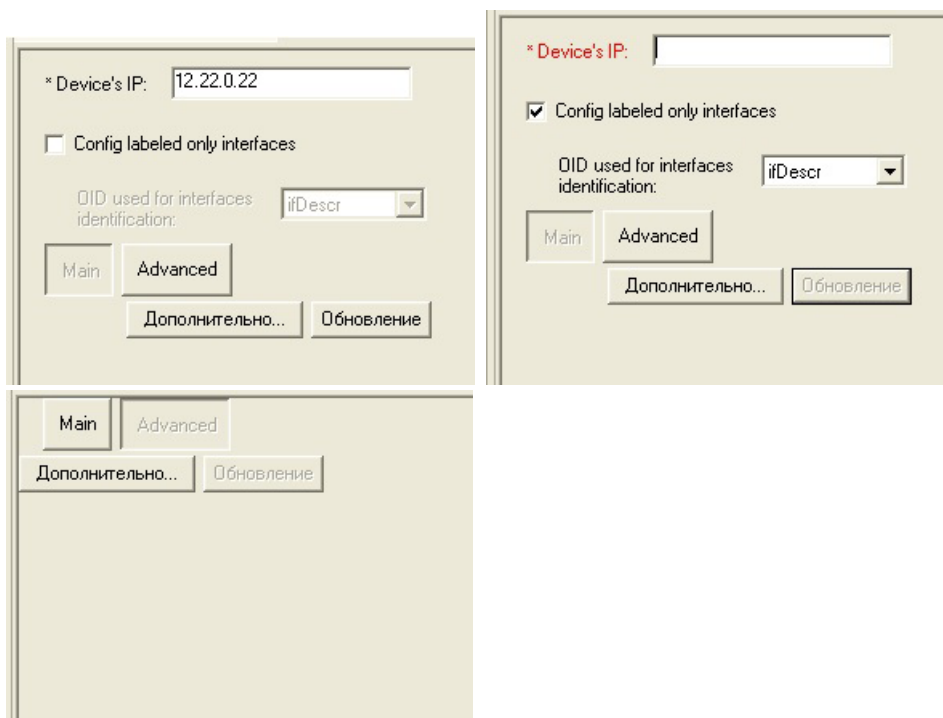
Ниже приведён пример структуры XML-файла:

```
<?xml version="1.0" ?>
<device type="Devices::Computers::Linux" version="1"
icon="Devices.Computers.Linux.ico">
  <dialog tag="dlg_main" name="Main" main="true">
    <control type="edit" tag="ip" label="* Device's IP"
validation="ip" />
    <control type="checkbox" tag="if.named-only"
label="Config labeled only interfaces" default="false" />
    <control type="combobox" tag="if.id-source"
label="OID used for interfaces identification">
      <value data="ifDescr" />
      <value data="ifName" />
      <value data="ifPhysAddress" />
      <value data="ifAlias" />
      <value data="ifIndex" />
    <dependency tag="if.named-only" value="false" kind="disable" voffset="-10" />
  </control>
</dialog>
<dialog tag="dlg_advanced" name="Advanced">
```

8. Настройка и использование модуля Нувола Топаз

```
</dialog>
</device>
```

Результат работы этого модуля показан на скриншотах:



Пример XML-файла

Ниже приведён пример XML-файла для объекта `Devices::Computers::Dagonet::FileProber`:

```
<?xml version="1.0"?>
<device type="Devices::Computers::Dagonet::FileProber" version="1"
icon="Devices.ico"
category="{8842C748-195D-4528-BCE1-4682C2DDEFB2}" status="notused">
  <dialog tag="dlg_main" name="Main" main="true">
    <control type="edit" tag="ip" label="* IP Address" validation="ip"/>
    <control type="edit" tag="community" label="* Community"
default="public" validation="notempty"/>
    <control type="edit" tag="display-name" label="* Display name"
validation="notempty"/>
    <control type="checkbox" tag="hr.show-cpu" label="Collect and Show
CPU stats"/>
    <control type="checkbox" tag="hr.show-storage" label="Collect and
Show Storage stats"/>
    <control type="checkbox" tag="hr.show-dev" label="Collect and Show
Plug-n-Play stats"/>
    <control type="checkbox" tag="hr.show-printers" label="Collect and
Show Printer stats"/>
    <control type="checkbox" tag="hr.show-proctab" label="Collect and
Show Process Table"/>
    <control type="checkbox" tag="hr.show-softtab" label="Collect and
Show Installed Software Table"/>
    <control type="edit" tag="custom-view-type" label="Change custom
view"/>
  </dialog>
</device>
```

9.

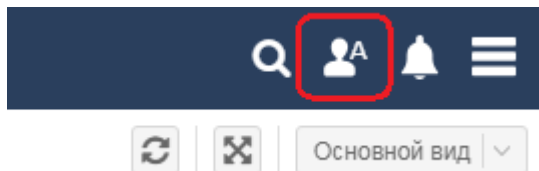
Администрирование ПК Нувола через Web-интерфейс

9. Администрирование ПК Нувола через Web-интерфейс

Web-интерфейс позволяет пользователям и администраторам конфигурировать объекты в дереве конфигурации, а также настраивать сигналы и оповещения. Набор разрешенных действий и доступ к данным определяется правами пользователя, настроенными администратором.

Общий порядок работы с Web-интерфейсом и функционал, доступный конечным пользователям, описан в разделе Web-интерфейс.

Для администратора (пользователя root) на иконке в заголовке Web-страницы отображается буква **A**.




Меню администратора (пользователя root), помимо стандартных пунктов, содержит пункты, не доступные другим пользователям:

- ❖ **Инструменты --> Управление Integrity** – проверка целостности дерева конфигурации и исправление нарушений;
- ❖ **Обновить лицензию** – актуализация доступных в Web-интерфейсе данных о лицензионных ограничениях в соответствии с установленным [файлом лицензии](#)^[43]. Информацию об установленной лицензии и текущие значения счетчиков по параметрам, ограниченному лицензионным соглашением, могут просмотреть все пользователи в пункте меню **Информация о лицензии**.

9.1. Управление пользователями

Настройка и администрирование [пользователей](#)^[36] осуществляется в окне **Управление пользователями**.

Чтобы его открыть, следует в заголовке Web-интерфейса нажать кнопку  и в меню приложений нажать иконку **Управление пользователями**.

Порядок настройки пользователей строго не регламентирован. Желательно, чтобы к моменту настройки пользователей в [дереве конфигурации](#)^[100] уже была создана структура папок, к которым будут выдаваться права доступа. Далее можно сначала создать учетные записи пользователей, а затем настроить группы, прикрепив к ним учетные записи пользователей. Либо наоборот: сначала настроить группы пользователей, а затем создать учетные записи, привязав их к группам пользователей.

См. также

[Администрирование пользователей \(вспомогательные настройки и действия\)](#)^[66]

9.2. Настройка дерева конфигурации

В этом разделе описаны методы настройки объектов дерева конфигурации.

ВНИМАНИЕ! Перед внесением изменений в дерево конфигурации рекомендуется создать резервную копию базы данных конфигурации.

9.2.1. Добавление нового объекта в дерево конфигурации

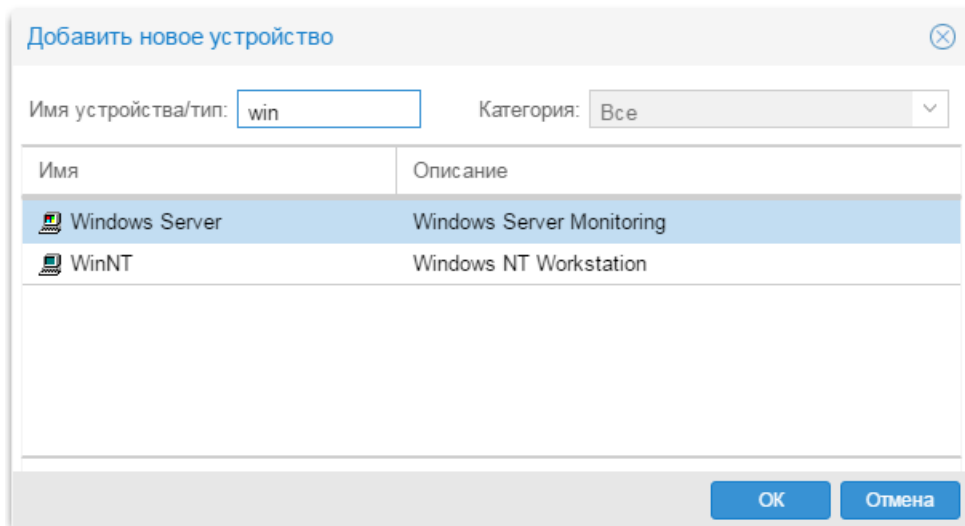
В настоящем разделе изложен порядок создания объекта в дереве конфигурации. В целом технология универсальна и не зависит от вида объекта. Однако следует учесть, что:

- ❖ конфигурирование некоторых объектов имеет ряд особенностей, в частности, необходимо заполнить специфические поля или ввести дополнительные настройки, предусмотренные шаблоном объекта. См. [Конфигурирование устройств](#)^[111];
- ❖ при конфигурировании устройств с автоматическим определением выбором шаблона объекта (Autotype) порядок действий несколько отличается от стандартного, изложенного в данном разделе.

Чтобы добавить объект в дерево конфигурации, следует:

1. в окне **Дерево конфигурации** выбрать папку, в которой планируется разместить новый объект;
Примечание. В корневой узел **root** дерева конфигурации можно добавлять только папки (Folder).
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить устройство** или нажать клавишу **Ins**;
3. в появившемся окне **Добавить новое устройство** выбрать шаблон объекта. Для поиска можно использовать поля фильтра: **Имя устройства/тип** (фрагмент имени шаблона) и **Категория** (группа шаблонов). Найденные шаблоны отображаются в таблице по мере ввода критерия поиска.
Примечание. При конфигурировании нескольких однотипных объектов подряд для ускорения поиска можно в поле **Категория** выбрать вариант **Недавно сконфигурированные устройства**. В результате будут отобраны последние шаблоны, использованные текущим администратором.

Для выбора шаблона поместить курсор на нужную строку и нажать кнопку **OK**;



4. в результате откроется окно **Добавить...**, в котором в соответствии с выбранным шаблоном будут сформированы поля и кнопки для настройки параметров объекта. Следует заполнить поля и с помощью кнопок ввести предусмотренные дополнительные настройки. См. [Настройка конфигурационных инструментов](#)^[103], [Конфигурирование устройств](#)^[111];

Добавить Windows Server в 'devs'

* IP-адрес:

* Группа (community):

Получить имя с устройства:

* Отображаемое имя:

Вести лог событий:

Отслеживать изменения дисков:

Отслеживать изменения аппаратного обеспечения:

Отслеживать изменения процессов:

Отслеживать изменения программного обеспечения:

Отслеживать изменения ARP:

Отслеживать изменения сетевого окружения:

Собирать ICMP статистику:

Собирать TCP статистику:

Собирать UDP статистику:

^ FNMP

Использовать FNMP:

Имя объекта:

5. нажать кнопку **Сохранить**.

В результате созданный объект будет добавлен как дочерний узел к выбранной папке в окне **Дерево конфигурации**.

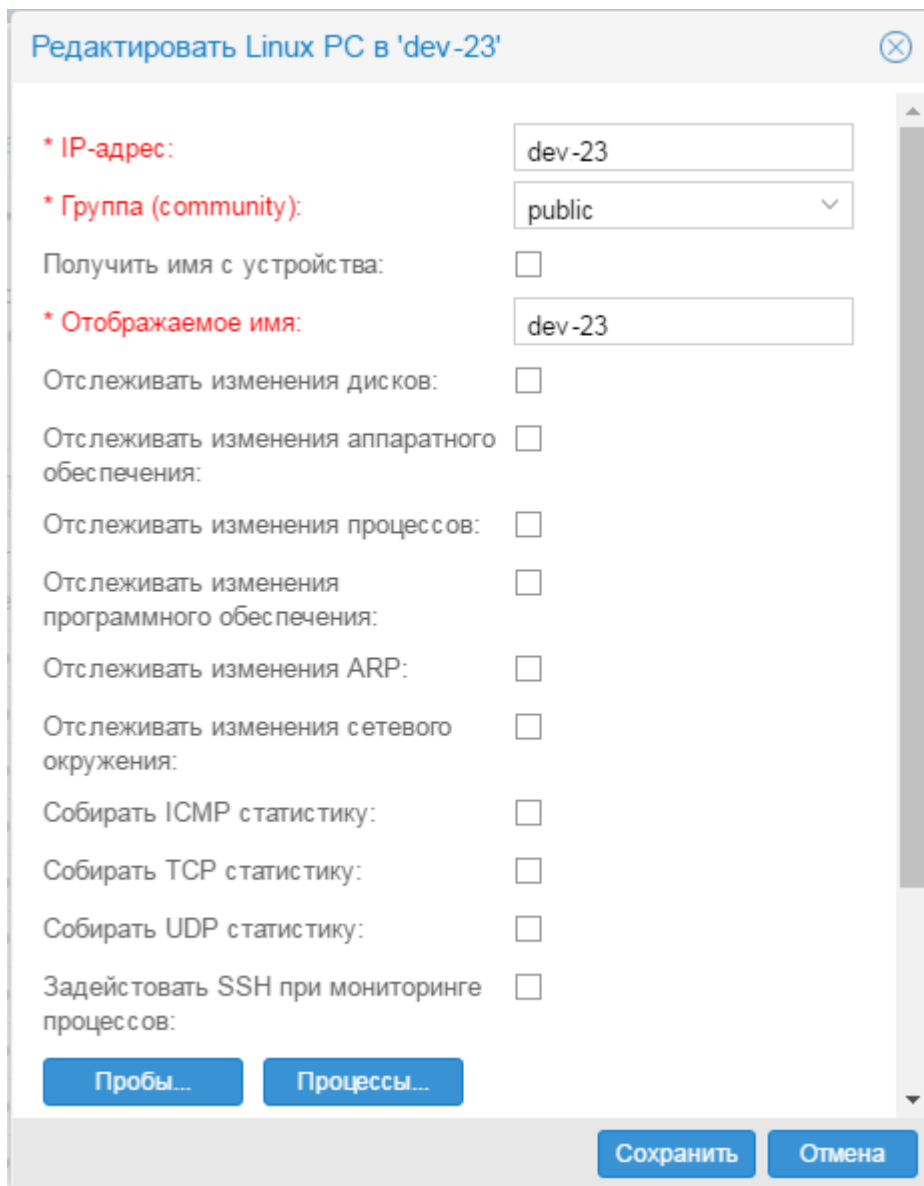
9.2.2. Просмотр и изменение параметров объекта

Открыть окно для просмотра или изменения параметров объекта, размещенного в дереве конфигурации, можно одним из способов:

- ❖ в окне **Дерево конфигурации** дважды кликнуть мышью на нужном объекте;
- ❖ в окне **Дерево конфигурации** поместить курсор на нужный объект и с помощью правой кнопки мыши в контекстном меню выбрать пункт **Редактировать** или нажать клавишу **Enter**.

В открывшемся окне **Редактировать...** будут отображены текущие параметры выбранного объекта.

После внесения изменений следует нажать кнопку **Сохранить**.



Редактировать Linux PC в 'dev-23'

* IP-адрес:

* Группа (community):

Получить имя с устройства:

* Отображаемое имя:

Отслеживать изменения дисков:

Отслеживать изменения аппаратного обеспечения:

Отслеживать изменения процессов:

Отслеживать изменения программного обеспечения:

Отслеживать изменения ARP:

Отслеживать изменения сетевого окружения:

Собирать ICMP статистику:

Собирать TCP статистику:

Собирать UDP статистику:

Задействовать SSH при мониторинге процессов:

9.2.3. Настройка конфигурационных инструментов

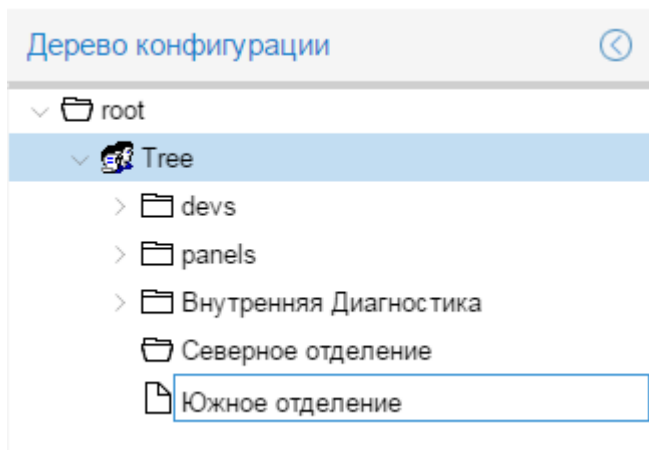
В настоящем разделе описаны особенности настройки [конфигурационных инструментов](#)^[25].

9.2.3.1. Настройка папки

Папка (Folder) – это конфигурационный инструмент, позволяющий структурировать объекты контроля.

Для создания папки в дереве конфигурации можно воспользоваться [стандартной технологией](#)^[100] или упрощенным методом:

1. в окне **Дерево конфигурации** выбрать папку, в которой планируется разместить новую папку;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Новая папка** или нажать клавишу **N**;
3. в появившемся поле ввести имя новой папки.



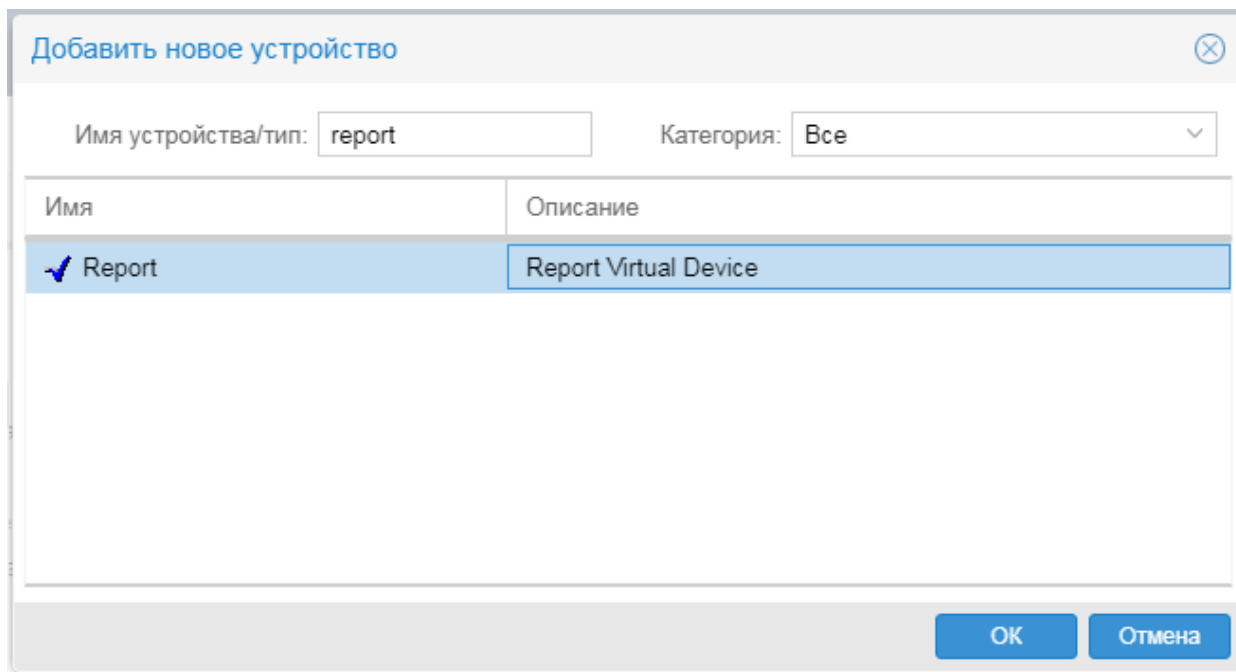
Существующую папку можно переименовать с помощью пункта **Переименовать** контекстного меню или клавиши **F2**.

9.2.3.2. Настройка отчета

Отчёт (Report) – конфигурационный инструмент, обеспечивающий форматированное представление статистической информации о состоянии объектов контроля или целевых объектов определенного типа.

Для создания отчета следует выполнить:

1. в окне **Дерево конфигурации** выбрать папку, в которой планируется разместить отчет;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить устройство** или нажать клавишу **Ins**;
3. в появившемся окне **Добавить новое устройство** выбрать шаблон объекта **Report** и нажать кнопку **ОК**;



4. в открывшемся окне **Добавить Отчет...** ввести поля:

Отображаемое имя | `display-name`

Имя объекта, отображаемое в дереве конфигурации. Должно быть уникальным в пределах одного уровня конфигурации (например, в рамках одной папки).

Описание | `long-desc`

Развернутое описание объекта. Может содержать HTML-теги и пробелы.

Тип отчёта | `type`

Тип отчёта. Наименование типа отчета следует выбрать из списка.

Диапазон (сек.) | `range`

Интервал времени (в секундах), за который берутся данные для формирования отчёта. По умолчанию – **3600**.

Количество строк в отчёте | `limit`

Количество отображаемых в отчете строк с результатами (например: 20 интерфейсов с максимальным объемом трафика). Целое положительное число, по умолчанию – **10**.

Функция отбора данных | `query-cfunc`



Функция отбора данных для построения отчета. Возможные варианты:

- ❖ **Среднее** (`AVERAGE`) – выбирает из базы данных RRD значения за интервал **Диапазон** (`range`) и вычисляет среднее значение;
- ❖ **Максимальное** (`MAX`) – выбирает из базы данных RRD максимальное значение за интервал **Диапазон** (`range`).

Автообновление | `auto-refresh`

Способ обновления информации.

- ❖ если флаг не установлен (`false`), то отчет формируется при создании (сохранении изменений) объекта в дереве конфигурации. В дальнейшем объект обновляется в соответствии с расписанием, а если оно не задано, то объект не актуализируется и хранится на сервере в первоначальном виде;
- ❖ если флаг установлен (`true`), то информация обновляется каждый раз по запросу пользователя, т.е. при выборе отчета/информационной панели в Web-интерфейсе.

5. в поле **Запланированные задания** необходимо настроить расписание выполнения системного процесса [querymaker](#)^[278], отвечающего за формирование отчета. Чтобы вызвать окно **Планировщик задач** для [настройки расписания](#)^[106], следует нажать кнопку **Редактировать**.
6. в таблице слева указать ветви дерева конфигурации, из которых будут отбираться целевые объекты для формирования отчёта. Чтобы добавить запись в таблицу, достаточно отметить нужный узел в дереве конфигурации справа от таблицы, либо в поле над таблицей ввести путь на узел дерева и нажать кнопку . Для удаления записи из таблицы следует нажать кнопку  в строке;
7. нажать кнопку **Сохранить**.

9. Администрирование ПК Нувола через Web-интерфейс

Добавить Отчет в 'Operator1'

* Отображаемое имя: Загруженные интерфейсы

Описание: Максимально загруженны

Тип отчета: Packets In

* Диапазон (сек.): 3601

* Количество строк в отчете: 10

* Функция отбора данных: Среднее Максимальное

Запланированные задания:

Каждый день в: 10:30

Редактировать

/Tree/Operator1

Путь
0 ✓ /Tree/Operator1 ✗

- root
 - LocalAdmin
 - Tree
 - Operator1
 - Внутренняя Диагностика

Сохранить Отмена

9.2.3.2.1. Настройка расписания формирования отчета (окно Планировщик задач)

Расписание формирования отчета настраивается в универсальном окне **Планировщик задач**, которое вызывается из окна **Добавить.../Редактировать...** с помощью кнопки **Редактировать**, расположенной возле поля **Запланированные задания**.

Планировщик задач

Минута 0

Час 0

День 0

Месяц 0

Добавить

Редактировать

Удалить

Удалить всё

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Запланированные задания:

Минута	Час	Месяц	День недели
--------	-----	-------	-------------

ОК Отмена

Структура универсального окна Планировщик задач

Таблица **Запланированные задания** содержит строки, отображающие периодичность выполнения процесса.

В верхней части окна расположены управляющие элементы для создания строк расписания.

Положение каждой из кнопок **Минута**, **Час**, **День**, **Месяц** определяет функциональное назначение соседнего с ней поля:

- ❖ если кнопка не нажата, то введенное в поле значение указывает на момент времени;
- ❖ если кнопка нажата, то введенное в поле значение указывает на периодичность. В название кнопки добавляется слово (**шаг**), а введенное в поле значение указывает на временной интервал, через который выполняется процесс.

Значения в поля рядом с кнопками **Минута**, **Час**, **День**, **Месяц** можно ввести вручную или с помощью бегунка.

Флаги **День недели** показывают дни недели, в которые выполняется процесс.

В правой верхней части окна расположены кнопки для управления таблицей **Запланированные задания**:

- ❖ **Добавить** – создание новой строки;
- ❖ **Редактировать** – сохранение изменений для выбранной строки;
- ❖ **Удалить** – удаление выбранной строки;
- ❖ **Удалить всё** – удаление всех строк.

Планировщик задач ✕

Минута

Час

День

Месяц

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Запланированные задания:

Минута	Час	Месяц	День недели
*	10	*	2
*	22	*	0

Добавить

Редактировать

Удалить

Удалить всё

OK
Отмена

Создание расписания и изменение его строк

Чтобы создать расписание выполнения процесса, в универсальном окне следует:

1. на панели настройки с помощью управляющих элементов задать периодичность;

☐ Пример 1: процесс выполняется "Каждый день в 13:15"

В этом случае следует:

- ▶ вручную или с помощью бегунка установить значения **Час=13** и **Минута=15** (кнопки **Час** и **Минута** не должны быть нажаты),
- ▶ отметить флаг **Все** (все дни недели).

Планировщик задач

Минута: 15

Час: 13

День: 0

Месяц: 0

Добавить

Редактировать

Удалить

Удалить всё

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Запланированные задания:

Минута	Час	День	Месяц	День недели
15	13	*	*	*

ОК Отмена

для сохранения заданных параметров нажать кнопку **Добавить**; в результате в таблице **Запланированные задания** будет создана строка расписания;

3. при необходимости с помощью кнопки **Добавить** создать новую строку и повторить пп. 1 и 2;
4. чтобы сохранить данные в универсальном окне настройки расписания и вернуться в предыдущее окно, нажать кнопку **ОК**.

Изменение существующего расписания осуществляется аналогичным образом. Чтобы в окне **Планировщик задач** изменить строку в расписании, следует:

1. в таблице **Запланированные задания** поместить курсор на строку;
2. с помощью управляющих элементов изменить параметры периодичности;
3. для сохранения нажать кнопку **Редактировать**.

Чтобы сохранить данные в универсальном окне настройки расписания и вернуться в предыдущее окно, следует нажать кнопку **ОК**.

Удаление строк из расписания

В универсальном окне настройки расписания для удаления одной выбранной строки из расписания предназначена кнопка **Удалить**, для удаления сразу всех строк – кнопка **Удалить всё**.

Например, чтобы удалить строку из расписания, необходимо выполнить действия:

1. открыть универсальное окно настройки расписания;
2. поместить курсор на строку расписания;
3. нажать кнопку **Удалить**;
4. чтобы сохранить изменение и закрыть окно, нажать кнопку **ОК**.

9.2.3.3. Настройка сводного графика

Сводный график (Total) – это конфигурационный инструмент, позволяющий представить в Web-интерфейсе в графической форме обобщенную информацию об изменении параметров целевых объектов, отобранных по заданным правилам.

Для создания сводного графика следует выполнить:

1. в окне **Дерево конфигурации** выбрать папку, в которой планируется разместить сводный график;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить устройство** или нажать клавишу **Ins**;
3. в появившемся окне **Добавить новое устройство** выбрать шаблон объекта **Total** и нажать кнопку **ОК**;

Имя	Описание
Total	Total Virtual Device

4. в открывшемся окне **Добавить...** ввести поля:

Отображаемое имя | `display-name`

Имя объекта, отображаемое в дереве конфигурации. Должно быть уникальным в пределах одного уровня конфигурации (например, в рамках одной папки).

Описание | `long-desc`

Развернутое описание объекта. Может содержать HTML-теги и пробелы.



9. Администрирование ПК Нувола через Web-интерфейс

Тип сводного графика | type

Тип сводного графика.

Фильтр по регулярному выражению | regexp

Маска описания целевого объекта (регулярное выражение UNIX). В соответствии с заданной маской из заданных ветвей дерева конфигурации отбираются целевые объекты, участвующие в построении сводного графика.

5. в таблице слева указать ветви дерева конфигурации, из которых будут отбираться целевые объекты для формирования сводного графика. Чтобы добавить запись в таблицу, достаточно отметить нужный узел в дереве конфигурации слева от таблицы, либо в поле над таблицей ввести путь на узел дерева и нажать кнопку . Для удаления записи из таблицы следует нажать кнопку  в строке;
6. если требуется сохранять данные по элементам сводного графика в долговременный файл, отметить флаг **Хранить информацию сводного графика в базе данных**. Может использоваться только для сводных графиков типа **Sum** или **Average**.
7. нажать кнопку **Сохранить**.





Добавить Итого в 'Сводные графики'




* **Отображаемое имя:** Средний трафик Ethernet

Описание: Средний трафик Ethernet

Тип сводного графика: average

Фильтр по регулярному выражению: eth*

	Путь	
0	/Tree/Operator1/WorkSt...	 
1	/Tree/Operator1/ws-002	 

/Tree/Operator1/ws-002   

- WorkStation-057
- ws-002
 - CPU 1
 - CPU 2
 - Average CPU
 - Memory
 - Disk L: utilization
 - Disk C: utilization
 - Disk D: utilization
 - Netstat Information
 - Realtek PCIe GBE Family Controller
 - Realtek PCIe GBE Family Controll...
 - Realtek PCIe GBE Family Controll...
 - Realtek PCIe GBE Family Controll...
 - WAN Miniport (IP)

Хранить информацию сводного графика в базе данных:

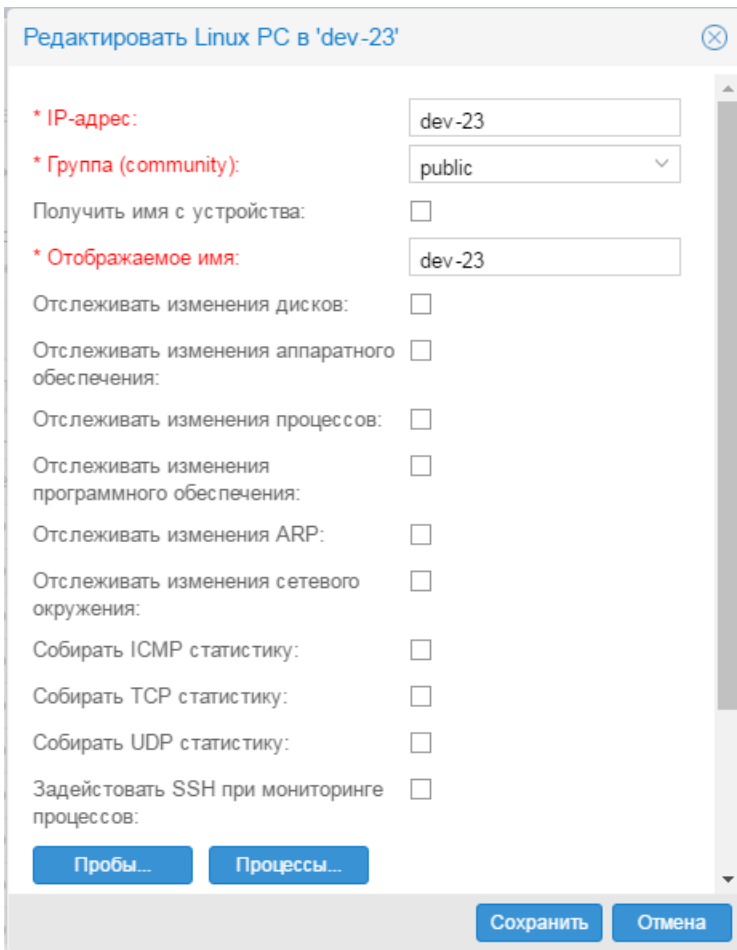
Сохранить **Отмена**

9.2.4. Конфигурирование устройств

В настоящем разделе описаны особенности настройки объектов дерева конфигурации, соответствующих устройствам.

В окне [Добавить...](#)^[100] / [Редактировать...](#)^[102] формируются поля для ввода параметров объекта в соответствии с шаблоном объекта. Для большинства устройств необходимо ввести [стандартные параметры](#)^[112].

Окна для ввода дополнительных настроек, предусмотренных шаблоном объекта, вызываются из окна [Добавить.../Редактировать](#) с помощью кнопок [Пробы...](#), [Процессы...](#), [Сервисы...](#), [Дополнительно...](#)



Редактировать Linux PC в 'dev-23'

* IP-адрес: dev-23

* Группа (community): public

Получить имя с устройства:

* Отображаемое имя: dev-23

Отслеживать изменения дисков:

Отслеживать изменения аппаратного обеспечения:

Отслеживать изменения процессов:

Отслеживать изменения программного обеспечения:

Отслеживать изменения ARP:

Отслеживать изменения сетевого окружения:

Собирать ICMP статистику:

Собирать TCP статистику:

Собирать UDP статистику:

Задействовать SSH при мониторинге процессов:

Пробы... Процессы...

Сохранить Отмена

9.2.4.1. Стандартные параметры устройства

При конфигурировании устройства необходимо, как правило, задать стандартные параметры:

Добавить Cisco IOS Switch в 'Север'

* IP-адрес:

* Группа (community):

Получить имя с устройства:

* Отображаемое имя:

SNMP порт:

SNMP версия:

Дополнительно...

Задействовать реплику:

Выбор коллектора:

FNMP

Использовать FNMP:

Имя объекта:

Сохранить Отмена

IP адрес | ip

IP-адрес устройства.

Группа (community) | community

Сообщество с доступом на чтение данных SNMP этого объекта (по умолчанию – public). Необязательный параметр для устройств без SNMP.

Получить имя с устройства

Установленный флаг означает, что при сохранении объекта в поле **Отображаемое имя** будет автоматически подставлено имя (sysname), запрошенное непосредственно с устройства.

Отображаемое имя | display name

Имя объекта, отображаемое в дереве конфигурации. Должно быть уникальным в пределах одного уровня конфигурации (например, в рамках одной папки). Поле недоступно для редактирования, если установлен флаг **Получить имя с устройства**.

SNMP порт | snmp-port-number

Номер порта UDP. По умолчанию – 161. Параметр используется для SNMP-устройств.

SNMP версия | snmp-version

Версия протокола SNMP. Параметр используется для SNMP-устройств. Возможные варианты:

auto – версия определяется автоматически последовательной проверкой 1, 2с: первый полученный ответ означает принятие соответствующей версии;

1 – версия SNMPv1;

2c – версия SNMPv2c.

Если при первоначальной конфигурации устройства версия SNMP не задана (значение **auto**), то при последующей переконфигурации устройства (`konfne -r cfg`) версия SNMP может измениться (в случае если изменились настройки агента на устройстве). Жёстко заданная версия (**1** или **2c**) при переконфигурации не изменится.

Выбор коллектора | `collector-name`

Имя системного процесса, который собирает данные с устройства. По умолчанию используется [kollector](#)^[275].

Системные процессы создает администратор в приложении [Диспетчер процессов](#)^[156]. Если в списке для выбора отсутствует нужный системный процесс, то следует очистить кеш браузера и перезагрузить страницу.

FNMP

Флаг **Использовать FNMP** предназначен для настройки работы модуля **Нуволы Приамус**. Флаг следует установить, если необходимо осуществлять сбор данных с устройства по протоколу FNMP. В этом случае также в поле **Имя объекта** следует ввести уникальный идентификатор используемого модуля **Нуволы Приамус** (значение параметра `server-name` системного процесса `priamUSD`), по умолчанию `defaultdc`.

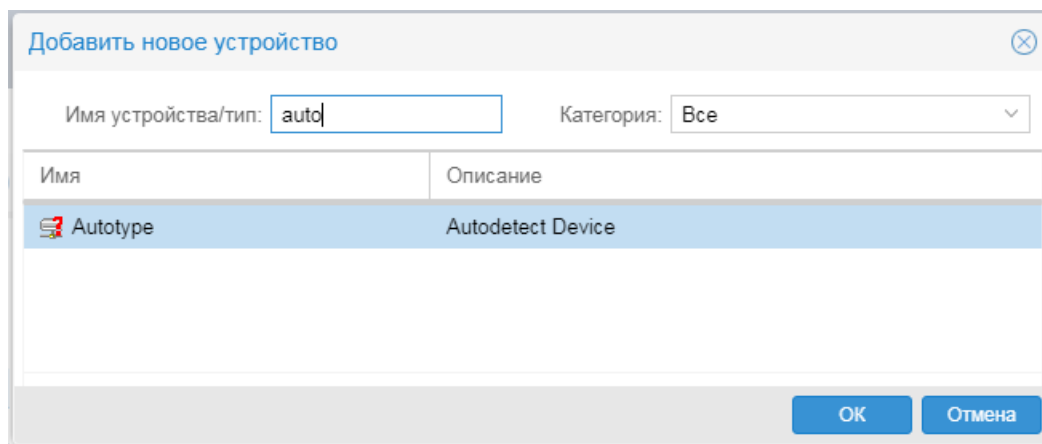
9.2.4.2. Конфигурирование устройства с автоматическим выбором шаблона объектов

Для ускорения и удобства конфигурирования устройств, доступных по SNMP, можно использовать механизм автоматического определения шаблона объектов **Autotype**. Общий принцип его применения таков:

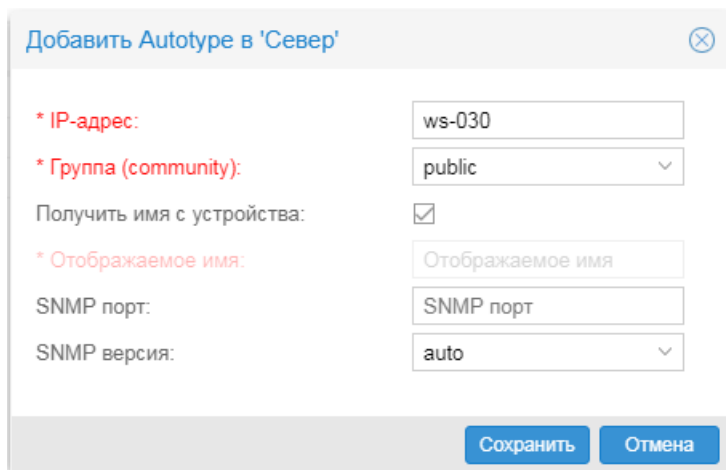
1. администратор, вместо конкретного шаблона объектов, выбирает **Autotype** и вводит ключевые параметры устройства: **IP-адрес** и **Группа (community)**;
2. ПМ **Нуволы Топаз** по этим параметрам обращается к устройству, подбирает в каталоге подходящий шаблон объектов и в соответствии с ним конфигурирует устройство.

Итак, чтобы добавить устройство в дерево конфигурации с использованием механизма **Autotype**, следует:

1. в окне **Дерево конфигурации** выбрать профиль или папку, в которую будет добавляться новый объект;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить устройство** или нажать клавишу **Ins**;
3. в появившемся окне **Добавить новое устройство** выбрать **Autotype** и нажать кнопку **ОК**;



4. ввести [стандартные](#)^[112] параметры устройства и нажать кнопку **Сохранить**:



В результате шаблон объекта будет определен автоматически и объект будет добавлен в выбранную папку в окне **Дерево конфигурации**;

5. при необходимости в окне [Редактировать...](#) задать дополнительные настройки объекта, предусмотренные шаблоном.

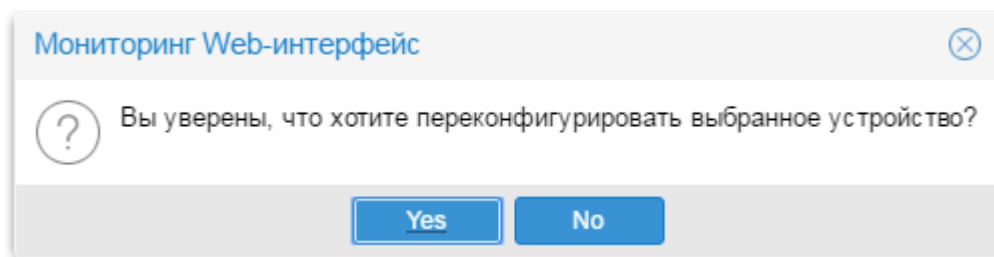
9.2.5. Переконфигурация объекта

При переконфигурации объекта осуществляется запрос конфигурационных параметров с устройства и сохранение их в дереве конфигурации вместе с текущими настройками, заданными в полях окна **Редактировать**.

Переконфигурацию следует выполнять после изменения параметров функционирования устройства, например: после включения на маршрутизаторе новых интерфейсов.

Для переконфигурации объекта следует:

1. в окне **Дерево конфигурации** поместить курсор на объект;
2. в контекстном меню выбрать пункт **Переконфигурировать**;
3. для подтверждения действия в появившемся окне нажать кнопку **Yes**.



Действие пункта **Переконфигурировать** распространяется также и на дочерние узлы выбранного объекта.

9.2.6. Удаление объектов

Удаление объектов из дерева конфигурации осуществляется с помощью команд контекстного меню. Команды удаления доступны только администратору (пользователю `root`). В таблице обозначены действия команд и приведены их описания.

Команда	Результат выполнения команды		
	Удаление объекта из дерева конфигурации	Прекращение сбора данных с объекта	Удаление ранее собранных данных
<p>Удалить ^[115]</p> <p>Удаление объекта только из конкретного местоположения дерева конфигурации без прекращения сбора данных и с сохранением данных, собранных ранее данных.</p>	✓		
<p>Удалить везде ^[116]</p> <p>Удаление объекта из конкретного местоположения дерева конфигурации с прекращением сбора данных, но сохранением данных, собранных ранее.</p>	✓	✓	
<p>Уничтожить ^[117]</p> <p>Удаление объекта из конкретного местоположения дерева конфигурации вместе с собранными данными и прекращение сбора данных с объекта.</p>	✓	✓	✓

Устройство может быть удалено из дерева конфигурации только вместе с целевыми объектами. Невозможно удалить целевые объекты устройства выборочно.

9.2.6.1. Удаление с помощью команды Удалить

Назначение команды

Удаление объекта только из конкретного местоположения дерева конфигурации без прекращения сбора данных и с сохранением данных, собранных ранее данных.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то удаление таким способом затронет только заданный объект, а остальные объекты, соответствующие этому устройству, не изменятся.

Применительно к реальным устройствам этот способ удаления целесообразен, если требуется прекратить возможность просмотра данных по объекту для одной из нескольких учетных записей, которым этот объект доступен.

Если в дереве конфигурации объект зарегистрирован только один раз, то после удаления сбор данных продолжается, однако для просмотра они недоступны. Для выявления таких "невидимых" объектов можно воспользоваться отчетом, который формируется системным процессом *integrity*.

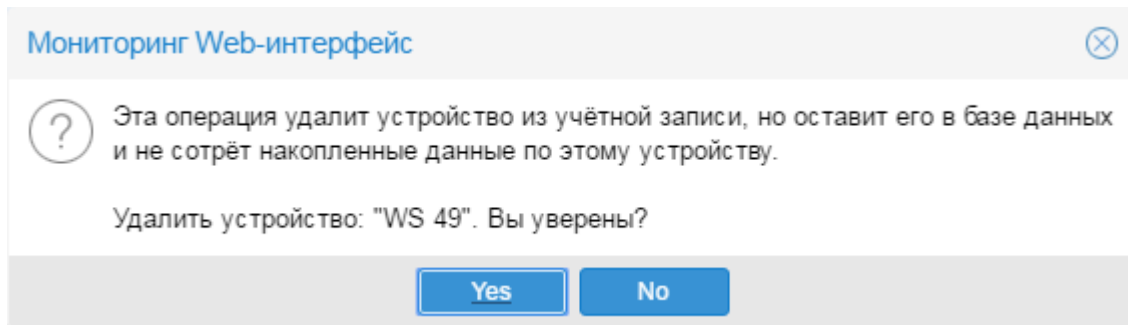
Выполнение команды

Для удаления объекта следует:

1. в окне **Дерево конфигурации** поместить курсор на объект, который требуется удалить (кроме целевого объекта);

9. Администрирование ПК Нувола через Web-интерфейс

2. выполнить одно из действий:
 - ❖ по нажатию правой кнопки мыши в контекстном меню выбрать пункт **Удалить**;
 - ❖ нажать клавишу **Del**;
3. в появившемся окне прочитать предупреждение и подтвердить удаление, нажав кнопку **Yes**.



9.2.6.2. Удаление с помощью команды Удалить везде

Назначение команды

Удаление объекта из конкретного местоположения дерева конфигурации с прекращением сбора данных, но сохранением данных, собранных ранее.

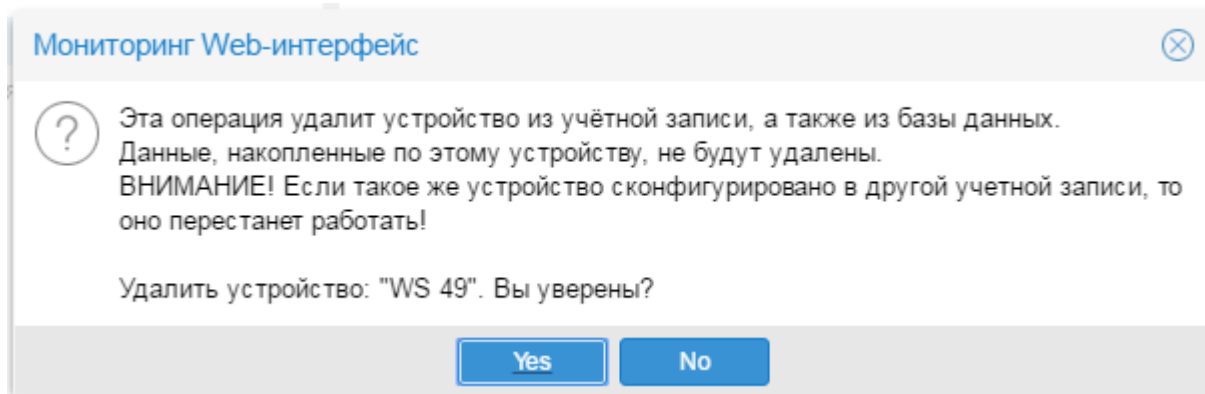
Собранные данные по объектам, удаленным таким способом, могут быть восстановлены: при повторной регистрации устройства в дереве конфигурации с теми же параметрами собранные данные попадут в графики.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то команда удаления затронет только заданный объект. Остальные объекты, соответствующие этому устройству, по-прежнему будут доступны через Web-интерфейс, однако информация по ним будет отображаться в усеченном виде.

Выполнение команды

Для удаления объекта следует:

1. в окне **Дерево конфигурации** выбрать объект, который требуется удалить (кроме целевого объекта);
2. по нажатию правой кнопки мыши в контекстном меню выбрать пункт **Удалить везде**;
3. в появившемся окне прочитать предупреждение и подтвердить удаление, нажав кнопку **Yes**.



9.2.6.3. Удаление с помощью команды Уничтожить

Назначение команды

Удаление объекта из конкретного местоположения дерева конфигурации вместе с собранными данными и прекращение сбора данных с объекта.

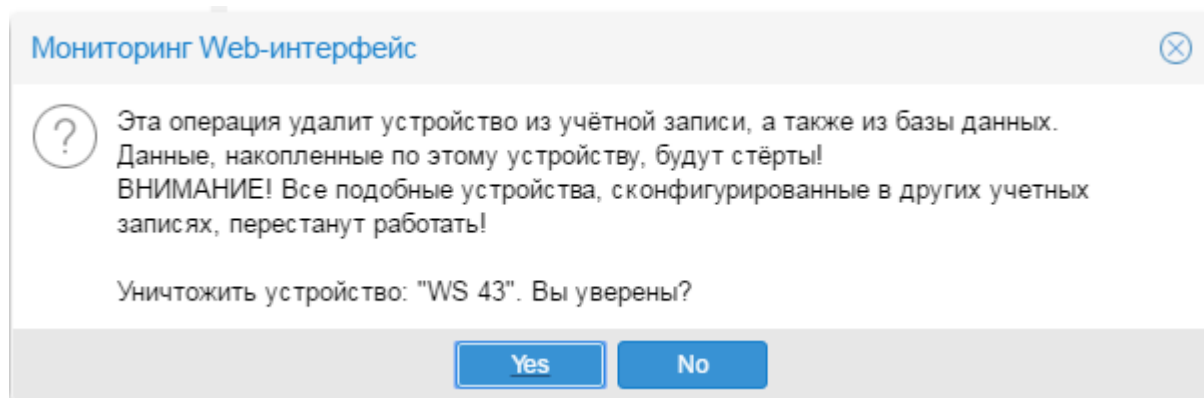
Собранные данные по объектам, удаленным таким способом, восстановлению не подлежат.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то команда удаления затронет только заданный объект. Остальные объекты, соответствующие этому устройству, по-прежнему будут доступны через Web-интерфейс, однако информация по ним будет отображаться в усеченном виде.

Выполнение команды


Для удаления объекта следует:

1. в окне **Дерево конфигурации** выбрать объект, который требуется удалить (кроме целевого объекта);
2. по нажатию правой кнопки мыши в контекстном меню выбрать пункт **Уничтожить**;
3. в появившемся окне прочитать предупреждение и подтвердить удаление, нажав кнопку **Yes**.



9.3. Редактор сигналов и оповещений (обзор приложения)

Приложение **Редактор сигналов и оповещений** позволяет настраивать наборы сигналов и наборы оповещений. В настоящем разделе руководства приведено общее описание основного окна редактора. Подробности о настройке сигналов и оповещений см. в разделах: [Настройка сигналов](#)^[122] и [Настройка оповещений](#)^[141].

Для вызова **Редактора сигналов и оповещений** следует в заголовке Web-интерфейса нажать кнопку  и в меню приложений нажать иконку с названием приложения.

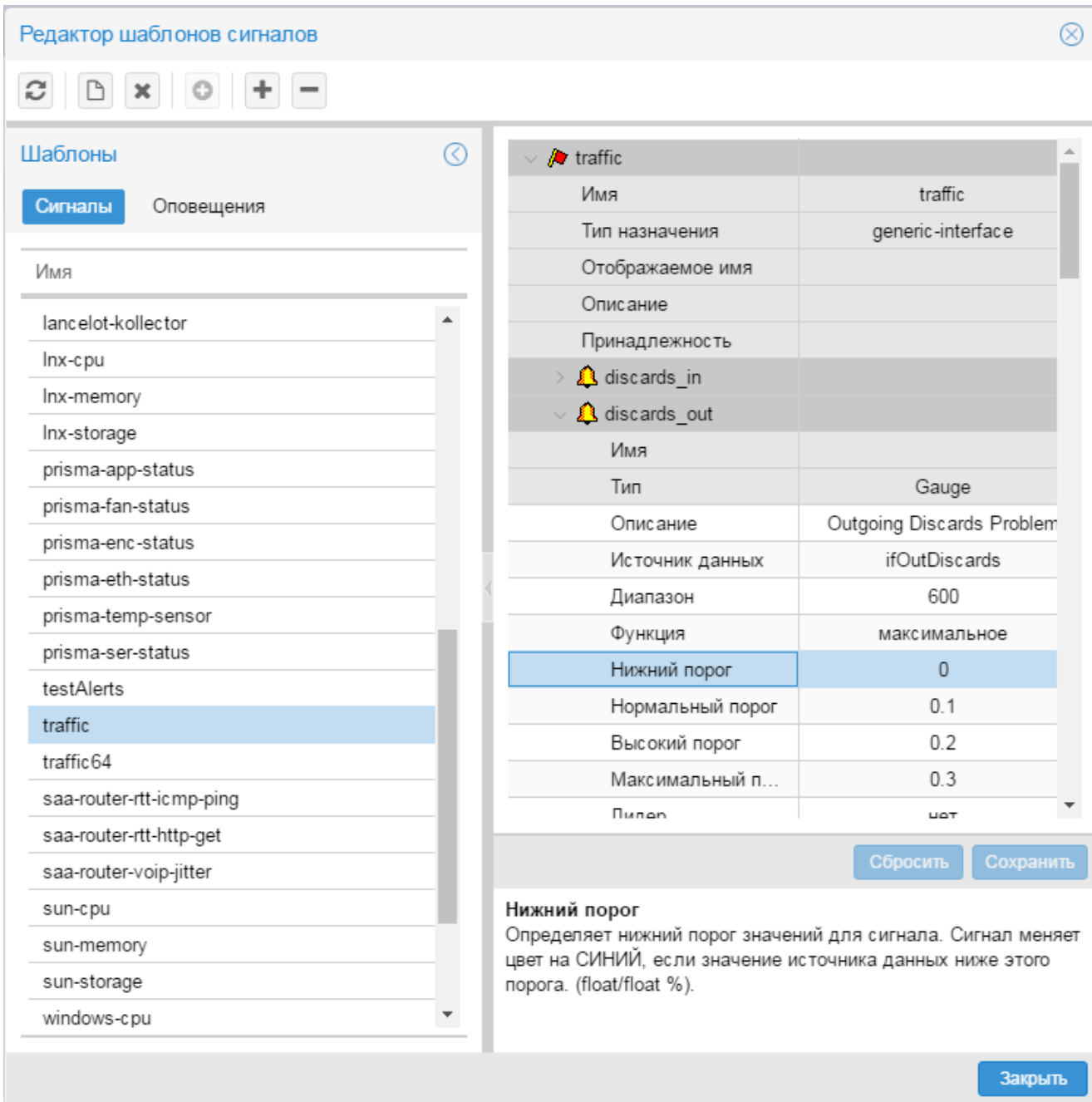
Примечание. Иконка доступна, если пользователь входит в группу пользователей с установленной глобальной политикой **Настройка сигналов и оповещений**.

Основное окно приложения имеет следующую структуру:

- ❖ под заголовком окна размещена [панель инструментов](#)^[118];
- ❖ слева находится окно **Шаблоны** с вкладками [Сигналы](#)^[119] и [Оповещения](#)^[121];

9. Администрирование ПК Нувола через Web-интерфейс

- ❖ справа располагается рабочая область, в которой выводится информация о наборе, выбранном на вкладке **Сигналы** или **Оповещения**.



Имя	traffic
Тип назначения	generic-interface
Отображаемое имя	
Описание	
Принадлежность	
> 📢 discards_in	
▼ 📢 discards_out	
Имя	
Тип	Gauge
Описание	Outgoing Discards Problem
Источник данных	ifOutDiscards
Диапазон	600
Функция	максимальное
Нижний порог	0
Нормальный порог	0.1
Высокий порог	0.2
Максимальный п...	0.3
Цвета	цвет

Для выхода из окна **Редактор шаблонов сигналов** можно использовать кнопку  в заголовке окна или кнопку **Закреть** в правом нижнем углу окна.

9.3.1. Панель инструментов и контекстное меню

В окне **Редактор шаблонов сигналов** для выполнения действий предусмотрены кнопки на панели инструментов, комбинации клавиш и пункты контекстного меню окна **Шаблоны**. Некоторые действия можно выполнить несколькими способами. Чтобы вызвать контекстное меню, следует в окне **Шаблоны** поместить курсор на строку списка на вкладках **Сигналы** или **Оповещения** и кликнуть правой кнопкой мыши.

Ниже перечислены кнопки панели инструментов и соответствующие им пункты контекстного меню и клавиши.



Обновление информации в окне **Редактор сигналов и оповещений** в соответствии с данными сервера Нувола.



Новый шаблон... | Ins

Создание нового набора сигналов или набора оповещений.



Удалить шаблон | Del

Удаление выбранного набора сигналов или набора оповещений.



Добавить класс оповещения

Добавление класса оповещения в набор оповещений.



Добавить подсигнал / Добавить метод

Добавление сигнала / метода оповещений в выбранный набор сигналов / набор оповещений.



Удалить подсигнал / Удалить метод

Удаление выбранного сигнала / метода из выбранного набора сигналов / набора оповещений.

Копировать шаблон...

Создание копии выбранного набора сигналов / набора оповещений.

Переименовать шаблон... | F2

Изменение отображаемого имени выбранного набора сигналов / набора оповещений.

9.3.2. Вкладка Сигналы

Вкладка **Сигналы** предназначена для настройки наборов сигналов и содержит список существующих наборов сигналов, как предустановленных, так и созданных администратором.

Для набора сигналов, выбранного на вкладке **Сигналы**, в рабочей области справа формируется двухуровневая иерархия раскрываемых блоков с параметрами.

Головной блок содержит параметры набора сигналов:

- ❖ **Имя** – наименование набора сигналов;
- ❖ **Тип назначения** – тип целевого объекта, с которым связан набор сигналов;
- ❖ **Отображаемое имя** – имя, под которым набор сигналов отображается в Web-интерфейсе, в том числе и в списке на вкладке **Сигналы**. Если параметр **Отображаемое имя** не задан, то на Web-страницах выводится значение параметра **Имя**.
- ❖ **Описание** – описание набора сигналов (комментарий);
- ❖ **Принадлежность** – имя сервиса, отвечающего за обработку сигналов. По умолчанию используется предустановленный сервис [alerts](#)^[268].

Вложенные блоки соответствуют сигналам в составе набора. Для раскрытия и сворачивания этих блоков можно использовать кнопки со стрелками (слева от пиктограммы). Каждый блок содержит:

- ❖ общие параметры сигнала, не подлежащие редактированию (на затененном фоне);
- ❖ глобальные значения параметров.

9. Администрирование ПК Нувола через Web-интерфейс

В поле под иерархией блоков выводится описание параметра, выбранного в данный момент.

Если пользователь изменил значение параметра, но еще не сохранил его, то соответствующая ячейка помечается красным треугольником в левом верхнем углу, а в нижней части рабочей области активизируются кнопки:

- ❖ **Сбросить** – отказ от внесенных изменений и возврат к прежним значениям;
- ❖ **Сохранить** – сохранение изменений на сервере Нувола.

Действие обеих кнопок распространяется сразу на все измененные параметры.

Блок с параметрами набора сигналов

Блок с параметрами сигнала

Глобальные значения параметров сигнала

Измененное, но ещё не сохраненное значение

Inx-cpu	
Имя	Inx-cpu
Тип назначения	linux-cpu-info
Отображаемое имя	
Описание	Alert for Linux station cpu
Принадлежность	
load	
Имя	
Тип	Gauge
Описание	
Источник данных	linuxGeneric Cpu
Диапазон	600
Функция	максимальное
Нижний порог	0
Нормальный порог	1
Высокий порог	4
Максимальный по...	5
Лидер	нет

Сбросить Сохранить

Нормальный порог
Определяет нормальный порог для сигнала. Сигнал меняет цвет на ЗЕЛЁНЫЙ, если значение источника данных ниже этого порога или на ЖЁЛТЫЙ, если значение выше. (float/float %).

Закрыть

Администратор (пользователь `root`) имеет следующие возможности:

- ❖ [создать новый набор сигналов](#)^[125];
- ❖ [добавить сигнал в набор сигналов](#)^[126];
- ❖ [удалить сигнал из набора сигналов](#)^[128];
- ❖ [скопировать набор сигналов](#)^[130];
- ❖ [переименовать набор сигналов](#)^[130];
- ❖ [изменить глобальные значения параметров сигнала](#)^[132];
- ❖ [удалить набор сигналов](#)^[133].


9.3.3. Вкладка Оповещения

Вкладка **Оповещения** предназначено для настройки наборов оповещений и содержит список существующих наборов оповещений, как предустановленных, так и созданных администратором.

Для набора оповещений, выбранного на вкладке **Оповещения**, в рабочей области справа формируется двухуровневая иерархия раскрываемых блоков с параметрами.

Головной блок с пиктограммой  содержит параметры набора оповещений:

- ❖ **Имя** – наименование набора оповещений;
- ❖ **Описание** – описание набора оповещений (комментарий);
- ❖ **Отображаемое имя** – имя, под которым набор сигналов отображается в Web-интерфейсе.

Вложенные блоки соответствуют классу оповещений и методам оповещений в составе набора. Блок класса оповещений обозначен пиктограммой , блоки методов оповещений – пиктограммами, соответствующими типу оповещения. Для раскрытия и сворачивания этих блоков можно использовать кнопки со стрелками (слева от пиктограммы). Каждый блок содержит:

- ❖ общие параметры, не подлежащие редактированию;
- ❖ глобальные значения параметров.

В поле под иерархией блоков выводится описание параметра, выбранного в данный момент.

Параметры, недоступные для редактирования, отображаются на затененном фоне.

Если администратор изменил значение параметра, то соответствующая ячейка в левом верхнем углу помечается красным треугольником, а в нижней части рабочей области активизируются кнопки:

- ❖ **Сбросить** – отказ от внесенных изменений и возврат к прежним значениям;
- ❖ **Сохранить** – сохранение изменений на сервере Нувола.

Действие обеих кнопок распространяется сразу на все измененные параметры.

9. Администрирование ПК Нувола через Web-интерфейс

Редактор шаблонов сигналов

Шаблоны

Сигналы Оповещения

Имя

AlertsConsole
EsbEvent
EsbNotification
mailer
logger

logger	logger
Имя	logger
Описание	Log notification in the file
Отображаемое имя	
filter	filter
Имя	filter
Тип	Simple
Сигналы	all
Исключительный	off
Описание	
Состояния 'плохо'	
Состояния 'игнорировать'	
Состояния 'хорошо'	
Разрешить напоминания	off
Козэф. увел. интервала нап...	1
Базовый интервал напомина...	3600
Ограничение времени нап...	0
Отправка всегда	off
log	log
Имя	log
Тип	Log
Включено	on
Путь	/usr/local/lancelot/var/lancelot-logs/eve...
Степень детальности	low

Сбросить Сохранить

Включено
Включен метод оповещения <написание в log-файл>. (Допустимые значения: on/ off)

Заккрыть

Блок с параметрами набора оповещений

Блок с параметрами класса оповещений

Глобальные значения параметров класса оповещений

Блок с параметрами метода оповещений

Глобальные значения параметров метода оповещений

Изменённое, но ещё не сохранённое значение

Администратор (пользователь `root`) имеет следующие возможности:

- ❖ [создать новый набор оповещений и добавить в него класс оповещений](#)^[143];
- ❖ [добавить метод оповещений в набор оповещений](#)^[144];
- ❖ [удалить метод/класс оповещений из набора оповещений](#)^[146];
- ❖ [скопировать набор оповещений](#)^[147];
- ❖ [переименовать набор оповещений](#)^[147];
- ❖ [изменить глобальные значения параметров метода / класса оповещений](#)^[148];
- ❖ [удалить набор оповещений](#)^[149].

9.4. Настройка сигналов

Процесс настройки сигналов включает в себя два этапа:

- ❖ [настройка наборов сигналов](#)^[124] в **Редакторе сигналов и оповещений** выполняется пользователями, входящими в группу с установленной глобальной политикой **Настройка сигналов и оповещений**;

Редактор шаблонов сигналов

Шаблоны

Сигналы Оповещения

Имя

- lancelot-kollector
- lnx-cpu
- lnx-memory
- lnx-storage
- prisma-app-status
- prisma-fan-status
- prisma-enc-status
- prisma-eth-status
- prisma-temp-sensor
- prisma-ser-status
- testAlerts
- traffic**
- traffic64
- saa-router-rtt-icmp-ping
- saa-router-rtt-http-get
- saa-router-voip-jitter
- sun-cpu
- sun-memory
- sun-storage
- windows-cpu

Имя	traffic
Тип назначения	generic-interface
Отображаемое имя	
Описание	
Принадлежность	
> 🛎️ discards_in	
▼ 🛎️ discards_out	
Имя	
Тип	Gauge
Описание	Outgoing Discards Problem
Источник данных	ifOutDiscards
Диапазон	600
Функция	максимальное
Нижний порог	0
Нормальный порог	0.1
Высокий порог	0.2
Максимальный п...	0.3
Цвет	цвет

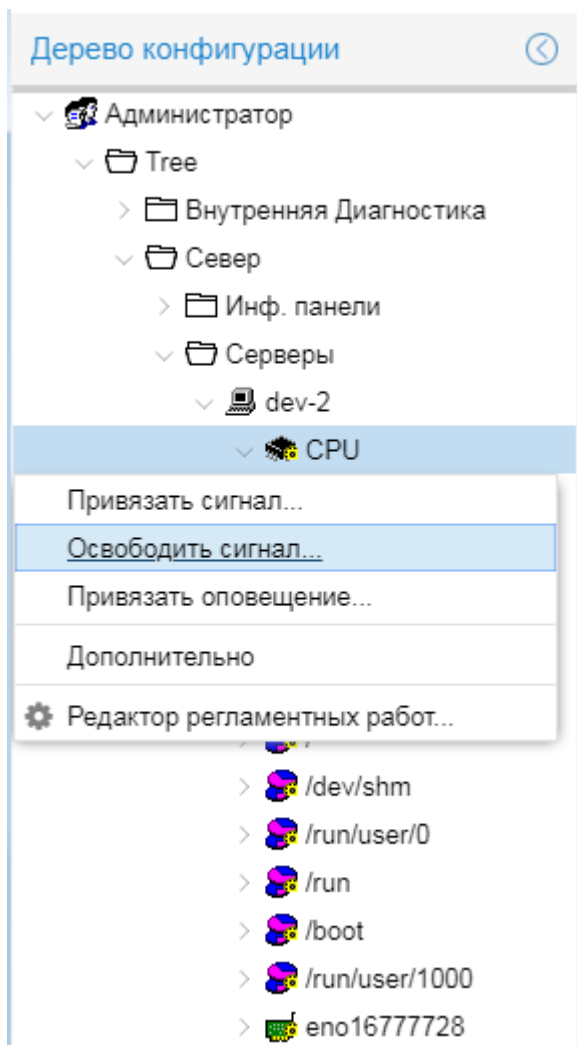
Сбросить Сохранить

Нижний порог
 Определяет нижний порог значений для сигнала. Сигнал меняет цвет на СИНИЙ, если значение источника данных ниже этого порога. (float/float %).

Закреть

9. Администрирование ПК Нувола через Web-интерфейс


- ❖ [назначение наборов сигналов целевым объектам](#)^[134] – выполняется через контекстное меню окна **Дерево конфигурации**.



При настройке сигналов следует руководствоваться правилами:

- ❖ набор сигналов может содержать один или несколько сигналов;
- ❖ к целевому объекту можно прикрепить только один набор сигналов.

9.4.1. Настройка наборов сигналов

Настройка наборов сигналов осуществляется в окне **Редактор шаблонов сигналов**, которое вызывается из меню приложений (кнопка  в заголовке Web-интерфейса).

Чтобы настроить новый набор сигналов, администратору следует:

- ❖ сначала [создать новый набор сигналов](#)^[125];
- ❖ затем [добавить в него сигналы](#)^[126].

Для ускорения настройки можно [скопировать](#)^[130] существующий набор сигналов и внести в него необходимые изменения.


Администратор также имеет следующие возможности:

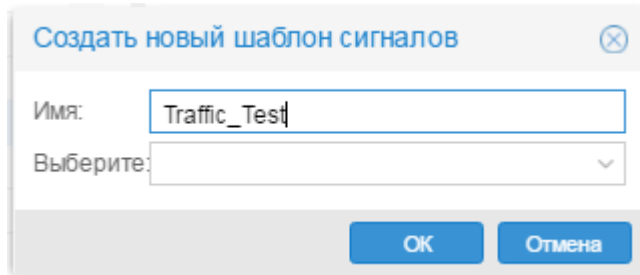
- ❖ [удалить сигнал из набора сигналов](#)^[129];
- ❖ [переименовать набор сигналов](#)^[130];


- ❖ [изменить глобальные значения параметров сигнала](#)¹³²;
- ❖ [удалить набор сигналов](#)¹³³.

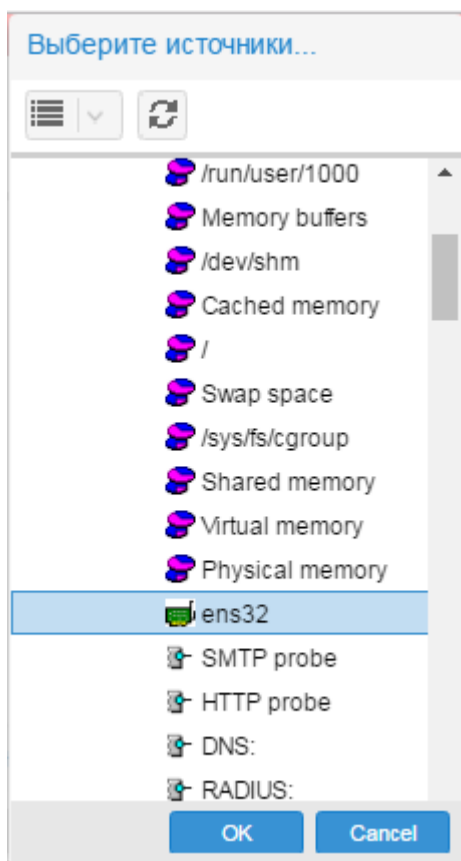
9.4.1.1. Создание нового набора сигналов

Чтобы создать новый набор сигналов, следует:

1. в окне **Шаблоны** открыть вкладку **Сигналы** и в контекстном меню выбрать пункт **Новый шаблон**, либо нажать клавишу **Ins** или кнопку  на панели инструментов;
2. в появившемся окне в поле **Имя** ввести имя нового набора сигналов;



3. в поле **Выберите** следует указать тип целевого объекта, к которому будет применяться создаваемый набор; для этого нажать кнопку . Откроется окно **Выбор элемента** с деревом конфигурации;



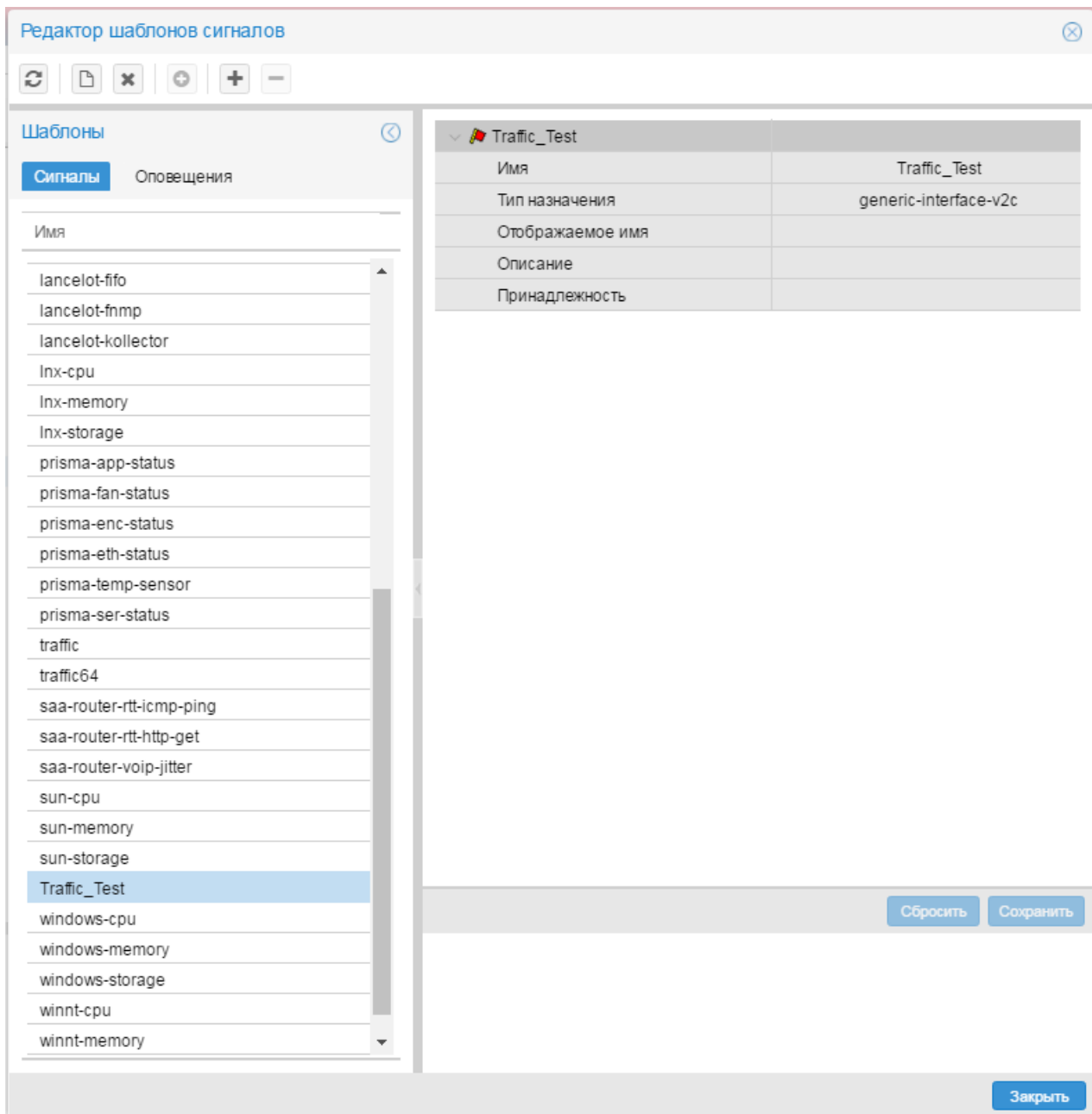
4. выбрать целевой объект нужного типа и нажать **ОК**.

В результате созданный набор сигналов будет добавлен в список вкладки **Сигналы**. В рабочей области будут заполнены поля **Имя шаблона** и **Тип назначения**. При необходимости следует заполнить поля:

9. Администрирование ПК Нувола через Web-интерфейс

Принадлежность

Имя (или шаблон имени) системного процесса, отвечающего за обработку сигналов. По умолчанию сигналы обрабатывает предустановленный системный процесс [alerts](#)²⁶⁸ и заполнять поле **Принадлежность** не нужно.



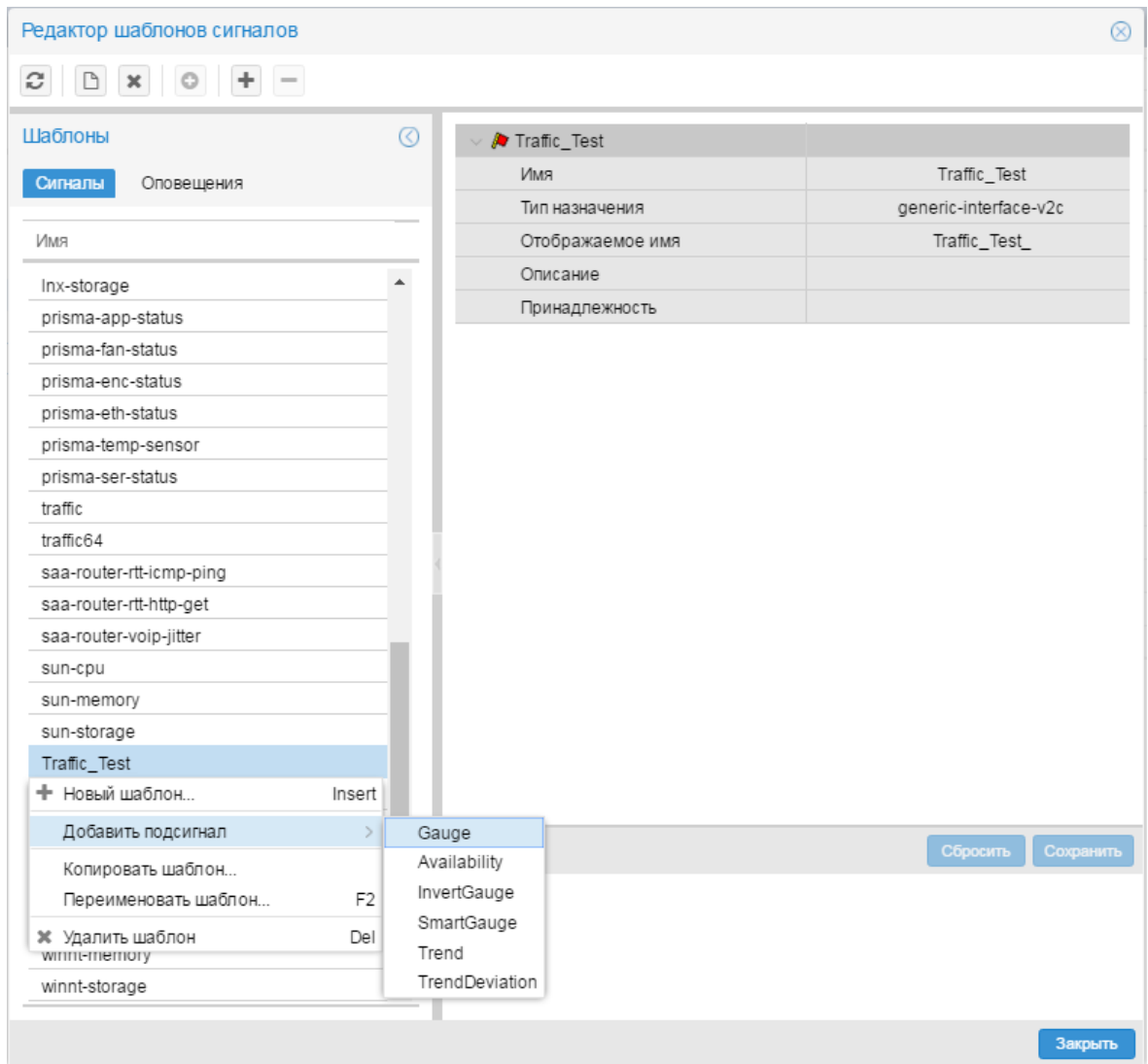
Далее в созданный набор следует [добавить сигналы](#)¹²⁶.


9.4.1.2. Добавление сигнала в набор сигналов

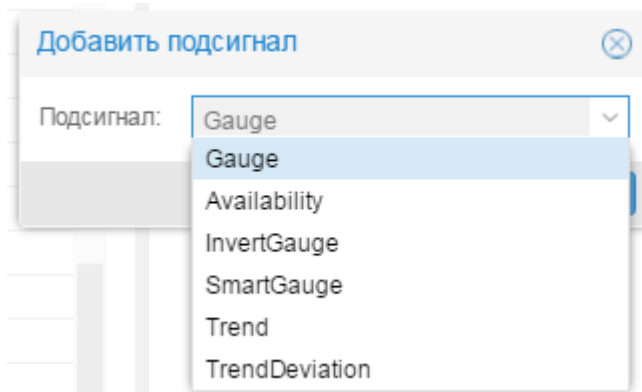
Для добавления сигнала в набор следует:

1. в окне **Сигналы** поместить курсор на набор сигналов, к которому нужно добавить сигнал;
2. выполнить одно из действий:

- ❖ в контекстном меню выбрать пункт **Добавить подсигнал** и далее – подпункт, соответствующий типу сигнала;

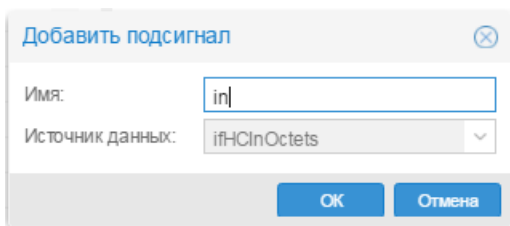


- ❖ на панели инструментов нажать кнопку  и в появившемся окне в списке **Подсигнал** выбрать тип сигнала;



3. в появившемся окне в поле **Имя подсигнала** ввести имя сигнала:

9. Администрирование ПК Нувола через Web-интерфейс



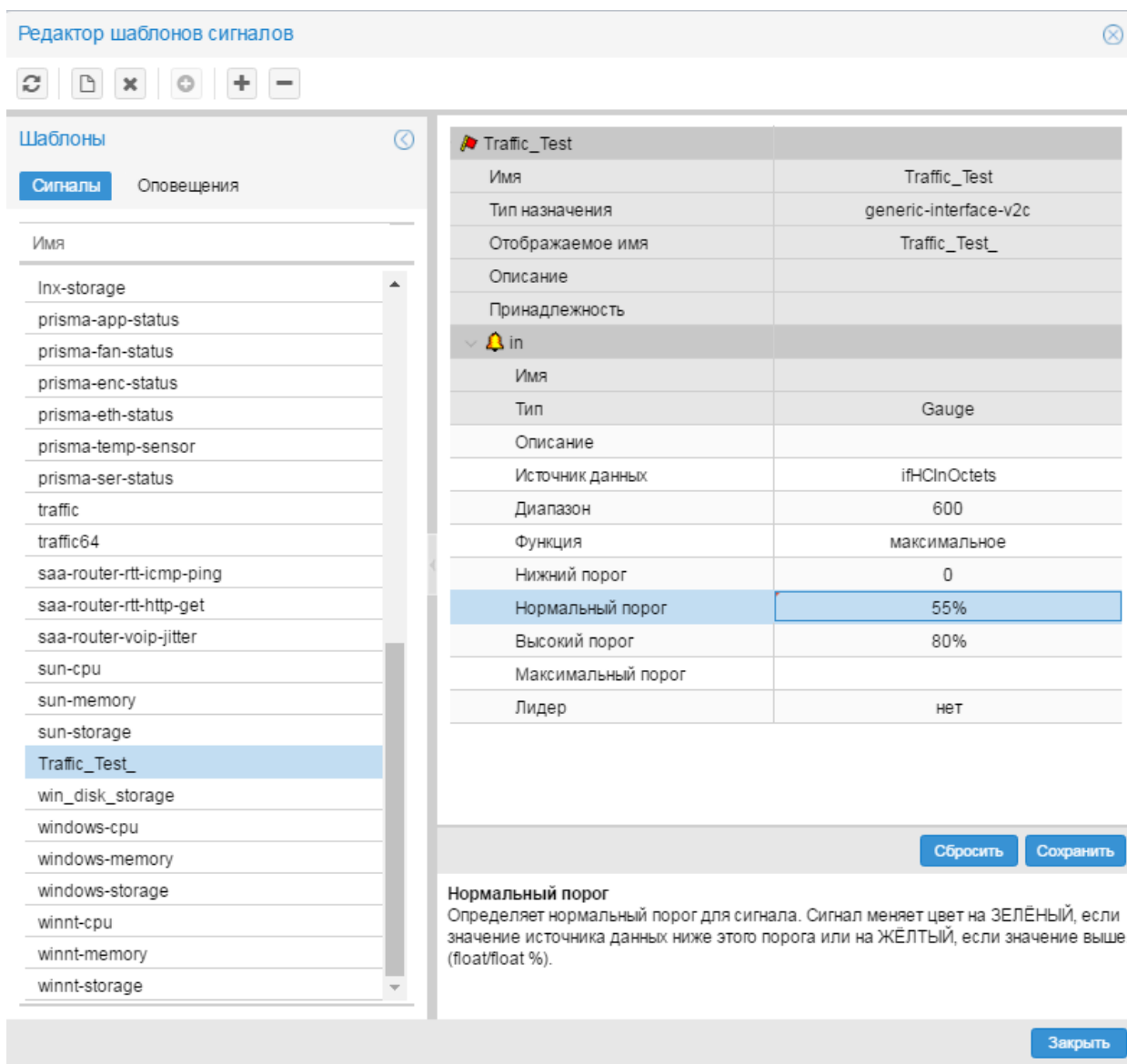
Добавить подсигнал

Имя:

Источник данных:

OK Отмена

4. в списке **Источник данных** указать источник данных. Рекомендуется проверить доступные источники данных в MIB устройства;
5. нажать кнопку **OK**. В результате в рабочую область справа будет добавлен блок с параметрами сигнала;
6. в рабочей области ввести глобальные значения параметров сигнала или оставить значения по умолчанию. Поля формируются в соответствии с типом сигнала. Серые поля недоступны для редактирования. Описание полей см. в справочнике [Типы сигналов](#)²⁷⁹;
7. после заполнения полей рабочей области нажать кнопку **Сохранить**.



Редактор шаблонов сигналов

Шаблоны

Сигналы Оповещения

Имя

Inx-storage
prisma-app-status
prisma-fan-status
prisma-enc-status
prisma-eth-status
prisma-temp-sensor
prisma-ser-status
traffic
traffic64
saa-router-rtt-icmp-ping
saa-router-rtt-http-get
saa-router-voip-jitter
sun-cpu
sun-memory
sun-storage
Traffic_Test_
win_disk_storage
windows-cpu
windows-memory
windows-storage
winnt-cpu
winnt-memory
winnt-storage

Traffic_Test	
Имя	Traffic_Test
Тип назначения	generic-interface-v2c
Отображаемое имя	Traffic_Test_
Описание	
Принадлежность	
in	
Имя	
Тип	Gauge
Описание	
Источник данных	ifHCInOctets
Диапазон	600
Функция	максимальное
Нижний порог	0
Нормальный порог	55%
Высокий порог	80%
Максимальный порог	
Лидер	нет

Сбросить Сохранить

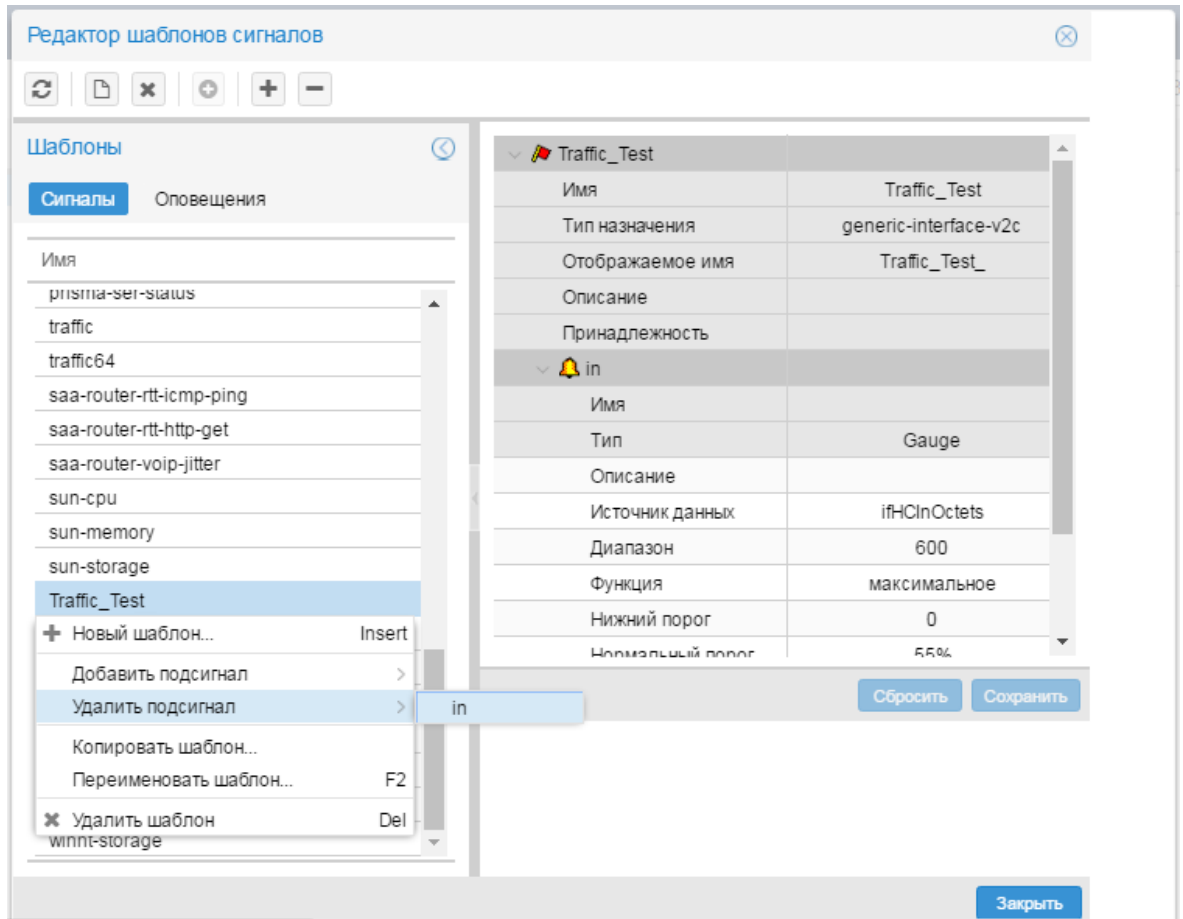
Нормальный порог
Определяет нормальный порог для сигнала. Сигнал меняет цвет на ЗЕЛЁНЫЙ, если значение источника данных ниже этого порога или на ЖЁЛТЫЙ, если значение выше. (float/float %).


Закрыть

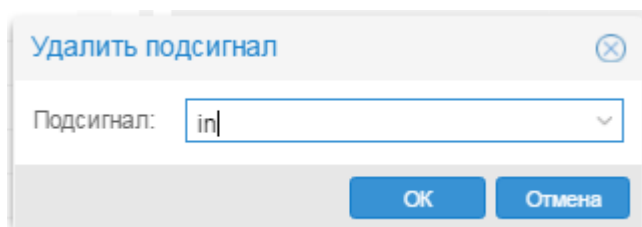
9.4.1.3. Удаление сигнала из набора сигналов

Для удаления сигнала из набора сигналов следует:

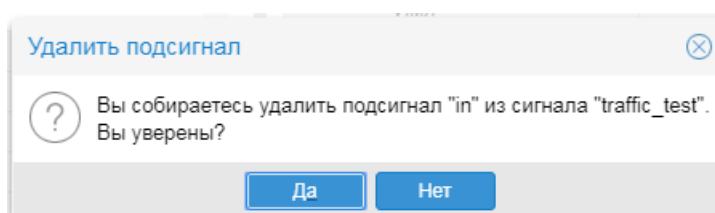
1. на вкладке **Сигналы** поместить курсор на набор сигналов, из которого требуется удалить сигнал;
2. выполнить одно из действий:
 - ❖ в контекстном меню выбрать пункт **Удалить подсигнал** и далее – подпункт с наименованием сигнала;



- ❖ на панели инструментов нажать кнопку  и в появившемся окне в списке **Подсигнал** выбрать сигнал;



3. в появившемся окне для подтверждения удаления нажать кнопку **Да**, для отказа – кнопку **Нет**.



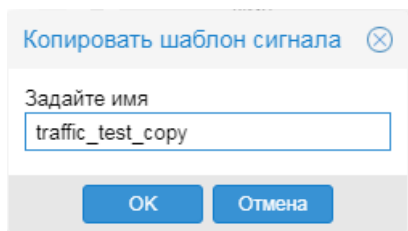
После подтверждения блок с параметрами сигнала будет удален из рабочей области.

9.4.1.4. Копирование набора сигналов

Для ускорения и упрощения настройки нового набора сигналов можно создать копию существующего набора сигналов и затем внести в него изменения.

Чтобы создать копию набора сигналов, следует:

1. в окне **Сигналы** поместить курсор на набор сигналов, который нужно скопировать;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Копировать шаблон**;
3. в появившемся окне ввести имя нового набора сигналов; по умолчанию предлагается **<имя исходного набора сигналов>_copy**;



4. нажать кнопку **ОК**.

В результате будет создан новый набор сигналов, идентичный исходному набору сигналов.

Для дальнейшей настройки набора сигналов можно выполнить следующие действия:

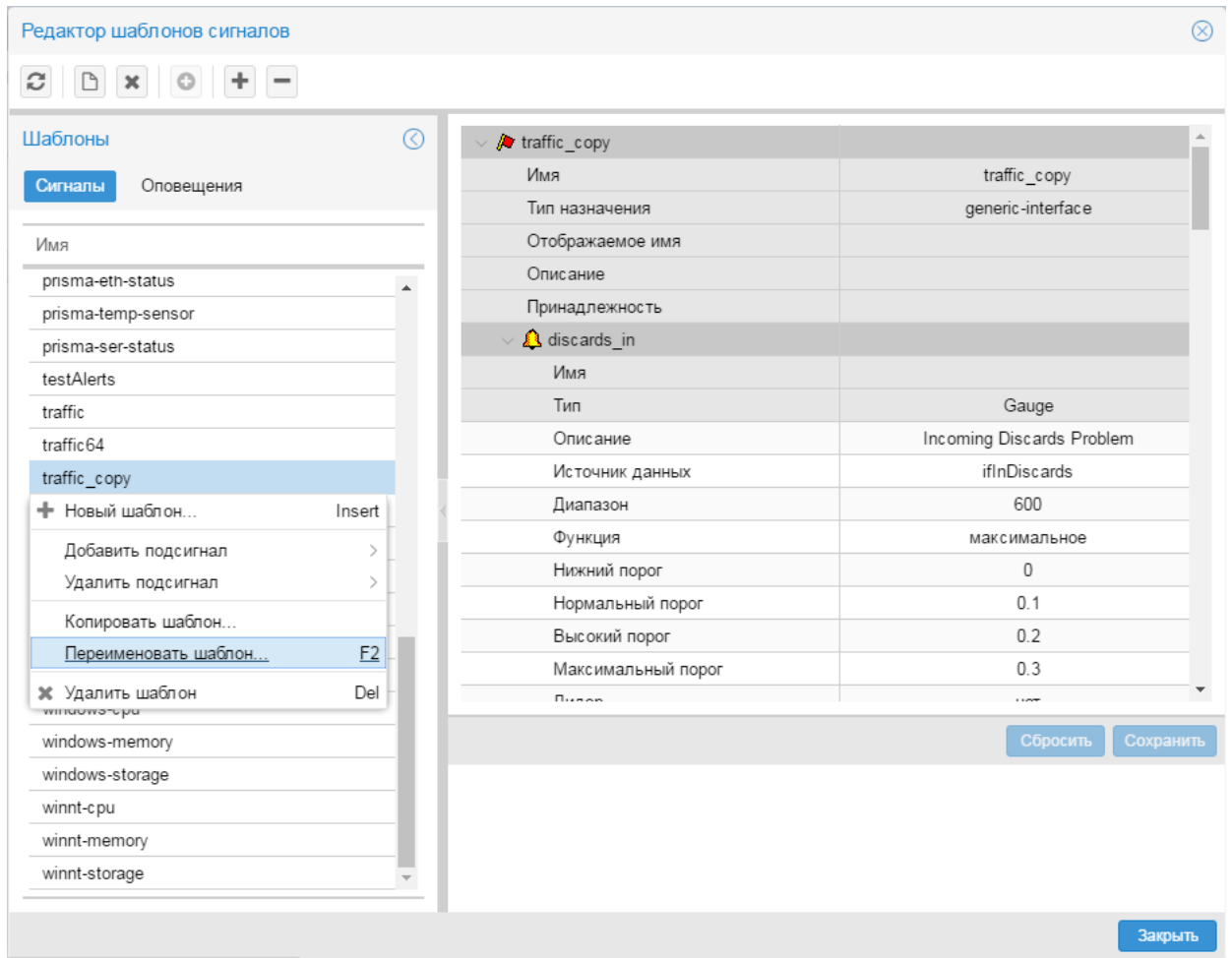
- ❖ [Переименование набора сигналов](#)^[130]
- ❖ [Добавление сигнала в набор сигналов](#)^[126]
- ❖ [Изменение глобальных значений параметров сигнала](#)^[132]

9.4.1.5. Переименование набора сигналов

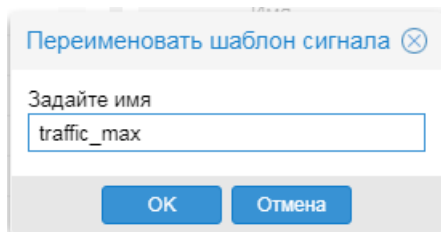
Переименование набора сигналов – это изменение его отображаемого имени.

Чтобы переименовать набор сигналов, следует:

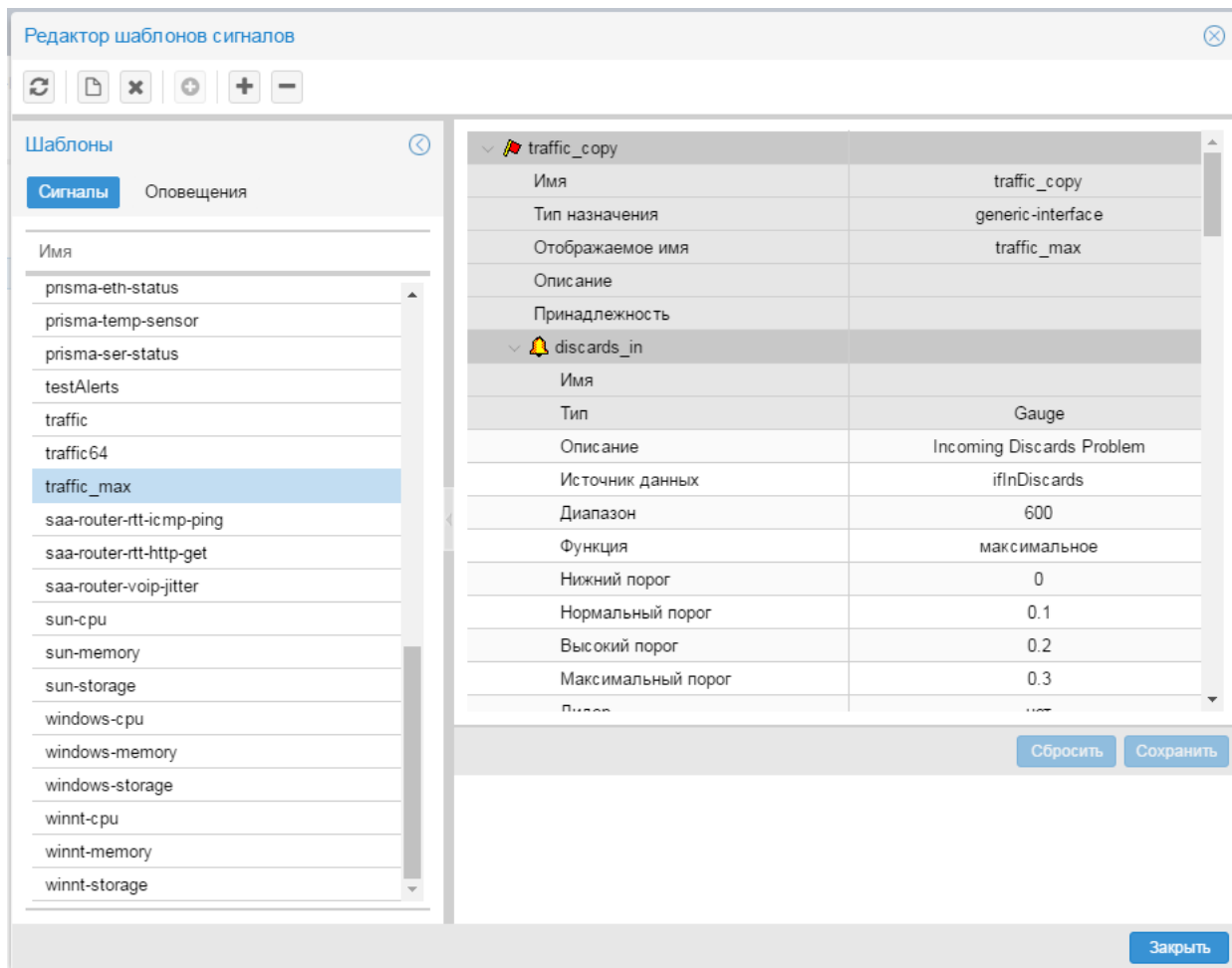
1. на вкладке **Сигналы** поместить курсор на нужный набор сигналов;



- с помощью правой кнопки мыши в контекстном меню выбрать пункт **Переименовать шаблон** (или нажать клавишу **F2**);
- в появившемся окне ввести новое название набора сигналов;



- нажать кнопку **ОК**.



9.4.1.6. Изменение глобальных значений параметров сигнала

Измененное глобальное значение параметра сигнала распространяется на целевые объекты, к которым сигнал (в составе набора) привязан без указания локального значения.

Для изменения глобальных значений параметров сигнала следует:

1. в окне **Сигналы** поместить курсор на нужный набор сигналов;
2. в рабочей области раскрыть блок с параметрами нужного сигнала и изменить значения полей во втором столбце. Поля формируются в соответствии с типом сигнала. Поля на затененном фоне недоступны для редактирования. Описание полей см. в справочнике [Типы сигналов](#) ^[279];
3. для сохранения изменений нажать кнопку **Сохранить**, для отказа от изменений – кнопку **Сбросить**.

Редактор шаблонов сигналов

Шаблоны

Сигналы Оповещения

Имя

- prisma-eth-status
- prisma-temp-sensor
- prisma-ser-status
- testAlerts
- traffic
- traffic64
- traffic_max**
- saa-router-rtt-icmp-ping
- saa-router-rtt-http-get
- saa-router-voip-jitter
- sun-cpu
- sun-memory
- sun-storage
- windows-cpu
- windows-memory
- windows-storage
- winnt-cpu
- winnt-memory
- winnt-storage

Имя	traffic_copy
Тип назначения	generic-interface
Отображаемое имя	traffic_max
Описание	
Принадлежность	
Имя	discards_in
Тип	Gauge
Описание	Incoming Discards Problem
Источник данных	ifInDiscards
Диапазон	600
Функция	максимальное
Нижний порог	0
Нормальный порог	0.1
Высокий порог	0.2
Максимальный порог	0.9

Сбросить Сохранить


Максимальный порог
Граница максимального уровня критичности. Инцидент (подсигнал) принимает статус RED (красный), если значение SNMP переменной пересекло границу высокого уровня критичности, но не пересекло границу максимального уровня критичности. Если значение SNMP переменной пересекло границу максимального уровня критичности, инцидент (подсигнал) принимает статус CRIMSON. По умолчанию граница максимального уровня критичности равна

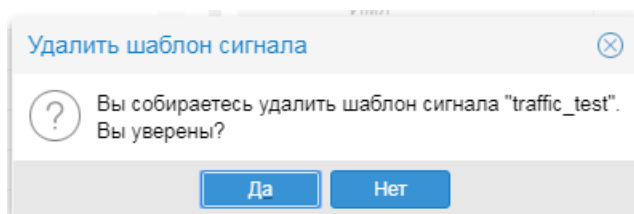
Закреть

9.4.1.7. Удаление набора сигналов

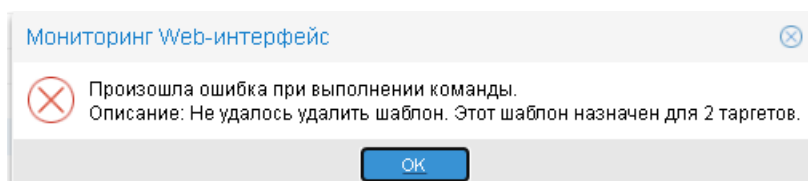
Допускается удаление наборов сигналов, не прикрепленных к целевым объектам.

Чтобы удалить набор сигналов, следует:

1. на вкладке **Сигналы** поместить курсор на нужный набор сигналов и с помощью правой кнопки мыши в контекстном меню выбрать пункт **Удалить шаблон** либо нажать клавишу **Del** или кнопку  на панели инструментов;
2. в появившемся окне для подтверждения удаления нажать кнопку **Да**, для отказа – кнопку **Нет**.



Если набор сигналов не прикреплен ни к одному целевому объекту, то он будет удален из списка на вкладке **Сигналы**. В противном случае выдается сообщение об ошибке.



Для удаления такого набора сигналов следует сначала [открепить](#) ¹³⁷ его от целевых объектов.

9.4.2. Назначение наборов сигналов целевым объектам

Прикрепление наборов сигналов к целевым объектам и изменение настроек привязки (т.е. изменение локальных значений параметров) пользователь осуществляет через контекстное меню окна **Дерево конфигурации**.

Примечание. В контекстном меню формируются соответствующие пункты, только если для ветви дерева конфигурации в группе пользователей установлено право доступа **Привязка сигналов/оповещений**.

Пользователь может выполнять следующие действия:

- ❖ [Привязка набора сигналов к целевым объектам](#)^[134]
- ❖ [Открепление набора сигналов от целевого объекта](#)^[137]
- ❖ [Изменение локальных значений параметров сигнала](#)^[139]

9.4.2.1. Привязка набора сигналов к целевым объектам

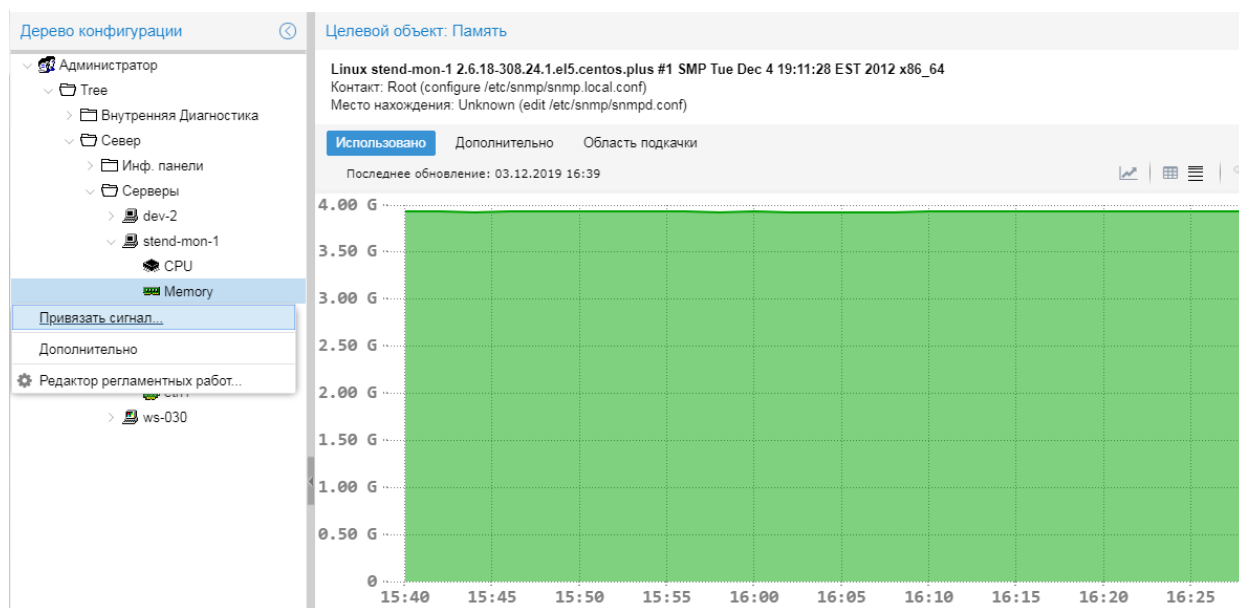
К одному целевому объекту можно привязать только один набор сигналов. После того как набор сигналов привязан к целевому объекту, при необходимости можно установить специфические настройки применения набора сигналов, для этого следует [задать локальные значения параметров сигналов](#)^[139].

Привязывать наборы сигналов к целевым объектам можно двумя методами.

Привязка набора сигналов к конкретному целевому объекту

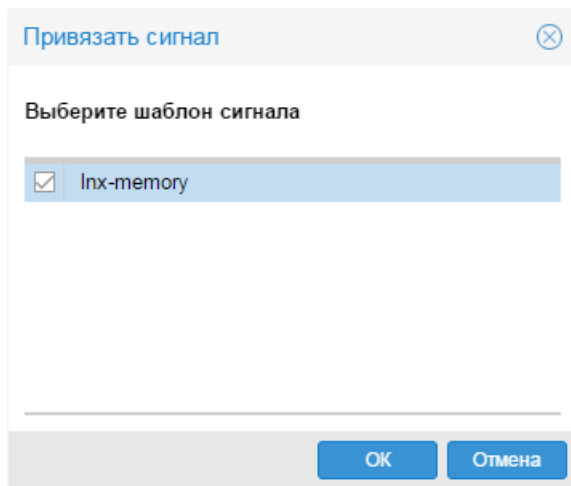
Для выполнения этой операции следует:

1. в окне **Дерево конфигурации** поместить курсор на целевой объект, к которому нужно привязать набор сигналов;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Привязать сигнал**;



в результате открывается окно **Привязать сигнал** со списком наборов сигналов, у которых тип назначения совпадает с типом целевого объекта;

3. пометить один набор сигналов;



4. нажать кнопку **ОК**.

В результате в окне **Дерево конфигурации** привязанный набор сигналов отображается как дочерний узел целевого объекта, а на пиктограмме целевого объекта, к которому был привязан набор сигналов, появляется желтый значок.

Шаблон сигнала: Inx-memory

Используется для объекта типа: linux-memory-info
Alert for Linux station memory

Состояние сигналов

Имя	* Источник...	Ст...	Значение	Пороговые значения	Время последней проверки
swap_avail	linuxSwap...	●	NaN	●:0.00 bytes ●:2.45 Gbytes ●:3.43 Gbytes ●:4.91 Gbytes	0
использов...	linuxUsed...	●	NaN	●:0.00 bytes ●:5.46 Gbytes ●:7.02 Gbytes ●:7.80 Gbytes	0

Локальные параметры сигналов

Имя	Класс	* Источник данных	Интервал	Функция	Пороговые значения	Описание
swap_avail	Gauge	linuxSwapUsed	600 сек.	максима...	●:0 ●:50% ●:70% ●:	
использ...	Gauge	linuxUsedMemory	600 сек.	максима...	●:0 ●:70% ●:90% ●:	

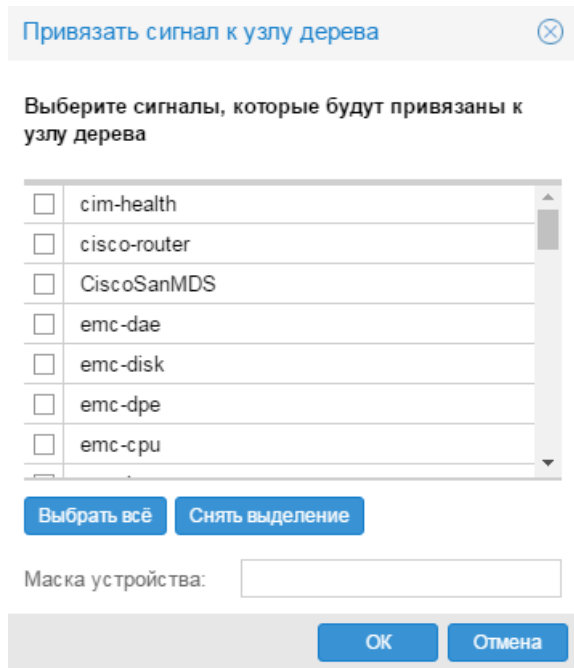
Глобальные параметры сигналов

Имя	Класс	* Источник данных	Интервал	Функция	Пороговые значения	Описание
swap_avail	Gauge		600 сек.	максима...	●:0 ●:50% ●:70% ●:	
использ...	Gauge		600 сек.	максима...	●:0 ●:70% ●:90% ●:	

Привязка наборов сигналов к целевым объектам в рамках выбранного узла дерева конфигурации

Для выполнения этой операции следует:

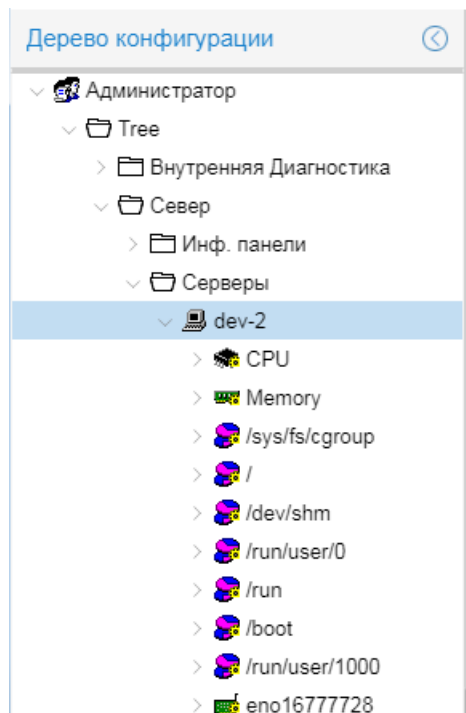
1. в окне **Дерево конфигурации** поместить курсор на устройство или папку;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Привязать сигнал**; в результате открывается окно **Привязать сигнал к узлу дерева** со списком существующих наборов сигналов;
3. отметить один или несколько нужных наборов. С помощью кнопки **Выбрать всё** можно отметить сразу все записи, чтобы применить наборы сигналов, подходящие по типу, ко всем целевым объектам в рамках указанного узла дерева конфигурации:



При необходимости можно настроить привязку набора сигналов по маске имени объекта. Для этого в поле **Маска устройства** следует задать требуемый фильтр в виде стандартного регулярного выражения Perl. Например, маска `cpu.*` позволит привязать сигналы к объектам `cpu`, `cpu1`, `cpu2`, `cpu_total` и т.д. Маска `cpu[0-9]+` позволит привязать сигналы к нумерованным процессорам, то есть `cpu1`, `cpu2` .. `cpu99`;

4. нажать кнопку **ОК**.

В результате выбранные наборы сигналов, подходящие по типу, будут привязаны к целевым объектам в рамках выбранного объекта дерева конфигурации. В окне **Дерево конфигурации** на иконках целевых объектов, к которым привязаны наборы сигналов, появляются желтые значки.



При выборе целевого объекта в окне **Дерево конфигурации** информация о привязанном наборе сигналов отображается в области вывода данных справа.

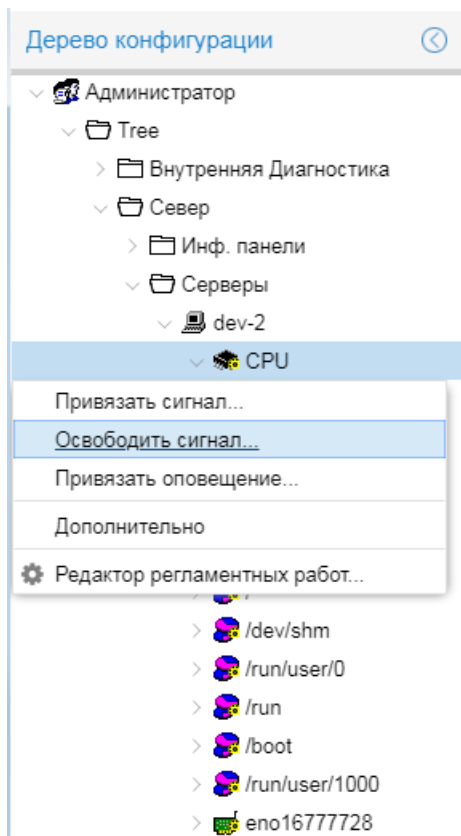
9.4.2.2. Открепление набора сигналов от целевого объекта

Для открепления наборов сигналов от целевых объектов можно использовать два метода.

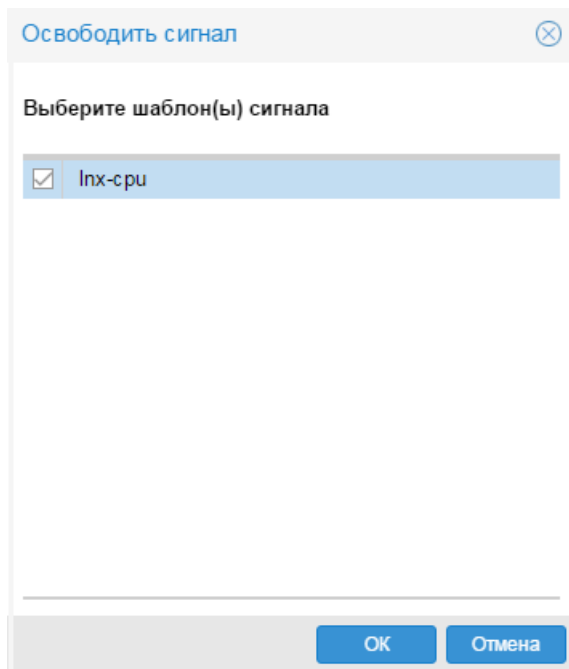
Открепление набора сигналов от заданного целевого объекта

Для выполнения этой операции следует:

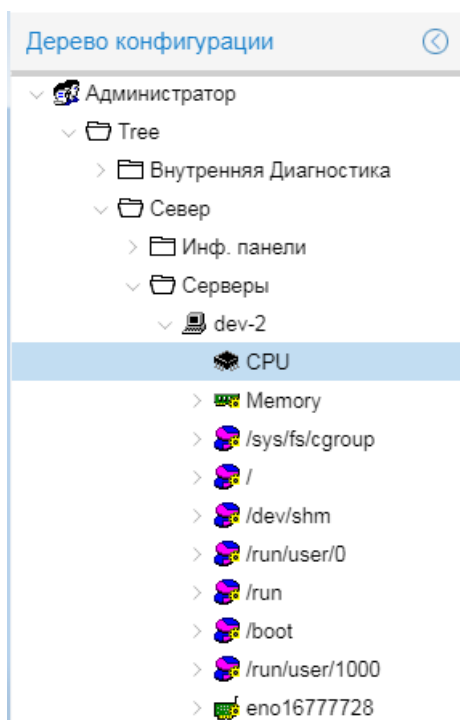
1. в окне **Дерево конфигурации** поместить курсор на целевой объект, от которого нужно открепить набор сигналов;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Освободить сигнал**;



3. убедиться, что в появившемся окне нужный набор сигналов отмечен, и нажать кнопку **ОК**.



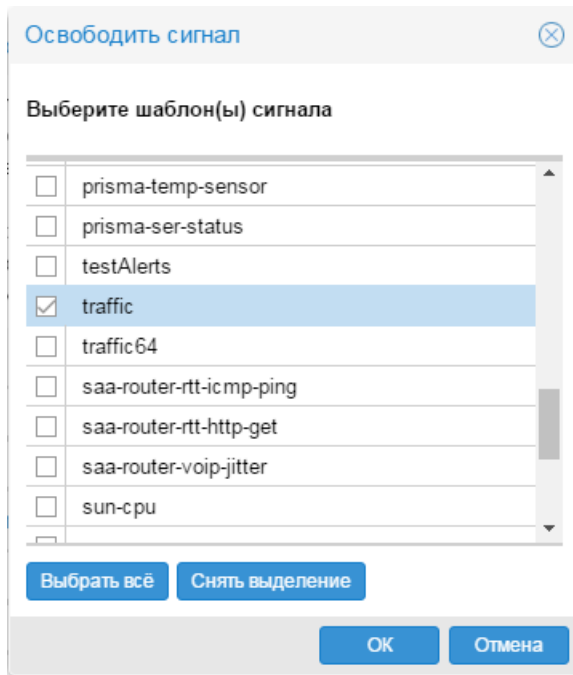
В результате в окне **Дерево конфигурации** будет удален узел, соответствующий набору сигналов, а с пиктограммы целевого объекта исчезнет желтый значок.



Открепление наборов сигналов от целевых объектов в рамках заданного узла дерева конфигурации

Для выполнения этой операции следует:

1. в окне **Дерево конфигурации** поместить курсор на устройство или папку;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Освободить сигнал**;
3. в появившемся окне пометить один или несколько наборов сигналов, которые необходимо открепить от целевых объектов, либо нажать кнопку **Выбрать все** для открепления наборов сигналов ото всех целевых объектов в рамках выбранного узла дерева конфигурации;



4. нажать кнопку **OK**.

В результате наборы сигналов будут откреплены от всех целевых объектов в рамках выбранного объекта дерева конфигурации. В окне **Дерево конфигурации** с пиктограмм целевых объектов исчезнут желтые значки.

9.4.2.3. Изменение локальных значений параметров сигнала

В наборе сигналов администратор устанавливает [глобальные значения](#)^[132] параметров сигнала. После того как набор сигналов привязан к целевому объекту, оператор (пользователь с правами администрирования) имеет возможность задать для параметров сигнала локальные значения, которые будут применяться только для данного целевого объекта.

Для изменения локальных значений параметров сигнала следует:

1. в окне **Дерево конфигурации** поместить курсор на сигнал, входящий в набор, привязанный к целевому объекту;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Редактировать** (или нажать клавишу **Enter** или дважды кликнуть мышью на выбранном узле дерева конфигурации).

9. Администрирование ПК Нувола через Web-интерфейс

В результате в области вывода данных будет показана таблица с параметрами сигнала:

первый столбец – наименование параметра;

второй столбец – локальное значение параметра (доступны для изменения);

третий столбец – глобальное значение параметра, заданное администратором (недоступны для изменения);

Поля формируются в соответствии с типом сигнала. Серые поля недоступны для редактирования. Детальное описание параметров см. в справочнике [Типы сигналов](#)²⁷⁹;

- изменить локальные значения во втором столбце (по нажатию кнопки **Сбросить** во второй столбец возвращаются последние сохраненные значения);

Имя	used	used
Тип	Gauge	Gauge
Описание		Used Memory
Источник данных	linuxUsedMemory	linuxUsedMemory
Диапазон	600	600
Функция	максимальное	максимальное
Нижний порог		0
Нормальный порог	68%	70%
Высокий порог	92%	90%
Максимальный порог		
Лидер	нет	нет
state	2	

Высокий порог
Граница высокого уровня критичности. Инцидент (под сигнал) принимает статус YELLOW (желтый), если значение SNMP переменной пересекло границу нормального уровня критичности, но не пересекло границу высокого уровня критичности. (Единицы измерения: %, действительное число).

Локальные значения Глобальные значения

- по окончании ввода значений для сохранения внесенных изменений нажать кнопку **Сохранить**. В результате в области вывода данных будет показана обновленная информация о сигнале.

Статус: ●

Состояние сигналов

Имя	Источ...	С...	Значе...	Пороговые значения	Время последней...	Время послдене...
used	linuxU...	●	3.040 ...	●:0.00 bytes ●:2.52 Gbytes ●:3.41 Gby...	24.05.2017 04:14:15	24.05.2017 03:34:14

Локальные параметры сигналов

Имя	Класс	Источник данн...	Инте...	Функ...	Пороговые значения	Описание
used	Gauge	linuxUsedMemo...	600 ...	макс...	●:0 ●:68% ●:92% ●:	

Глобальные параметры сигналов

Имя	Класс	Источник данн...	Инте...	Функ...	Пороговые значения	Описание
used	Gauge		600 ...	макс...	●:0 ●:70% ●:90% ●:	Used Memory

9.5. Настройка оповещений

Процесс настройки оповещений состоит из двух этапов:

- ❖ [настройка наборов оповещений](#)^[142] в **Редакторе сигналов и оповещений** выполняется пользователями, входящими в группу с установленной глобальной политикой **Настройка сигналов и оповещений**;

Редактор шаблонов сигналов

Шаблон: Оповещения

Имя: logger

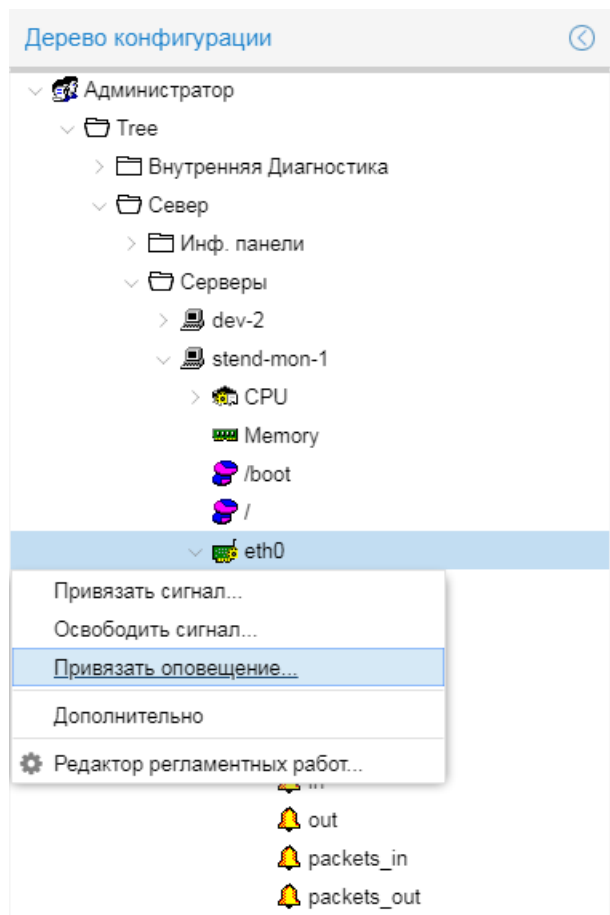
Имя шаблона	logger
Описание	Log notification in the file
Отображаемое имя	
Имя	filter
Тип	Simple
Сигналы	all
Исключительный	off
Описание	
Статус 'плохой'	
Статус 'игнорировать'	
Статус 'хороший'	
Включено	off
Кэфф. увел. интервала напом...	1
Базовый интервал напоминания	3600
Ограничение времени напоми...	0
Отправка всегда	off

Сбросить Сохранить

Закреть

9. Администрирование ПК Нувола через Web-интерфейс


- ❖ [назначение наборов оповещений целевым объектам](#)^[150] – выполняется через контекстное меню окна **Дерево конфигурации**.



При настройке сигналов следует руководствоваться правилами:

- ❖ набор оповещений обязательно должен содержать один класс оповещений;
- ❖ набор оповещений может содержать один или несколько методов оповещений;
- ❖ к целевому объекту можно прикрепить несколько наборов оповещений.

9.5.1. Настройка наборов оповещений

Настройка наборов оповещений осуществляется в окне **Редактор шаблонов сигналов**, которое вызывается из меню приложений (кнопка  в заголовке Web-интерфейса)

Для настройки наборов оповещений следует перейти на вкладку **Оповещения**.

Чтобы настроить новый набор оповещений, администратору следует:

- ❖ сначала [создать набор оповещений и добавить в него класс оповещений](#)^[143];
- ❖ затем [добавить в него методы оповещений](#)^[144].

Для ускорения настройки можно [скопировать](#)^[147] существующий набор оповещений и внести в него необходимые изменения.


Администратор также имеет следующие возможности:

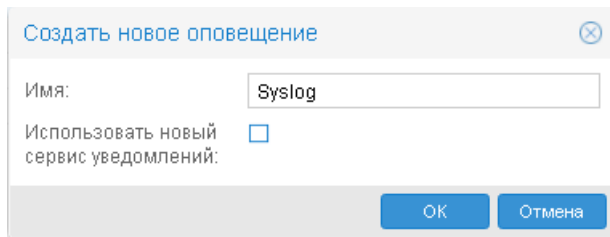
- ❖ [удалить метод/класс оповещений из набора оповещений](#)^[146];
- ❖ [переименовать набор оповещений](#)^[147];

- ❖ [изменить глобальные значения параметров метода / класса оповещений](#) ¹⁴⁸;
- ❖ [удалить набор оповещений](#) ¹⁴⁹.

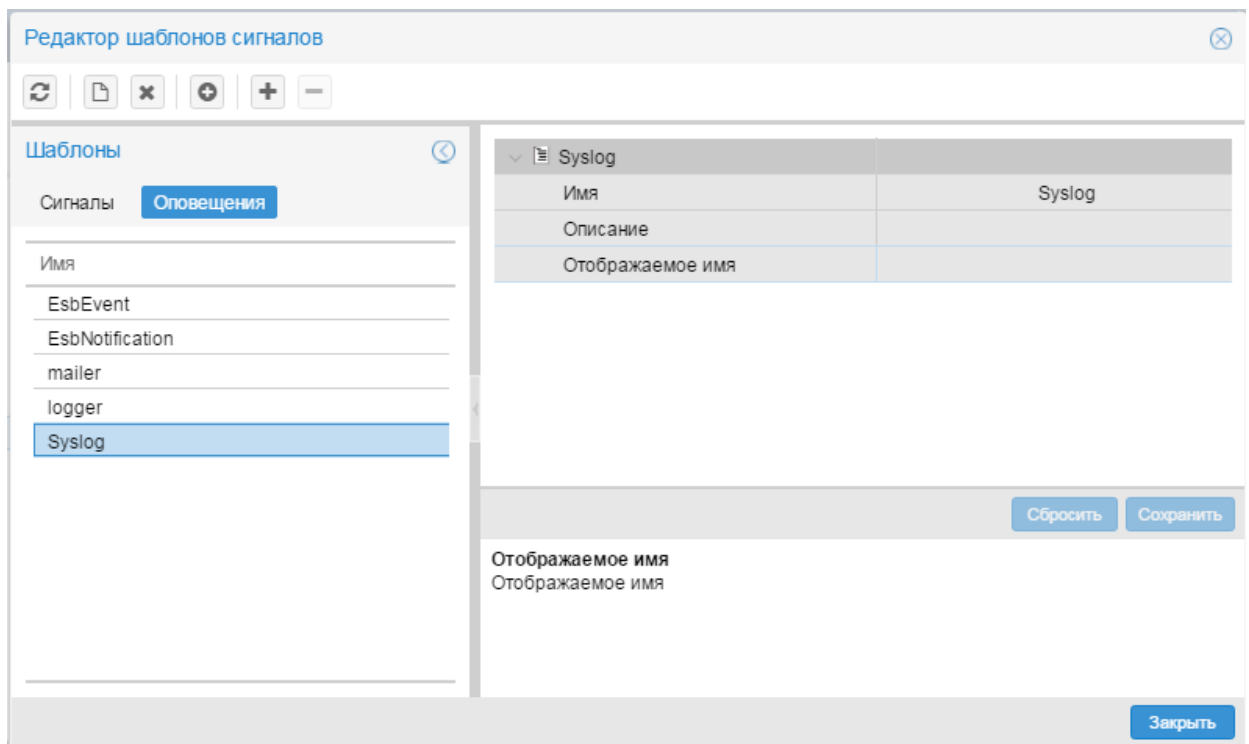
9.5.1.1. Создание набора оповещений и добавление класса оповещений

Для создания нового набора оповещения следует:


1. в окне **Шаблоны** открыть вкладку **Оповещения** и в контекстном меню выбрать пункт **Новый шаблон**, либо нажать клавишу **Ins** или кнопку  на панели инструментов;
2. в появившемся окне ввести имя нового набора и нажать **ОК**.



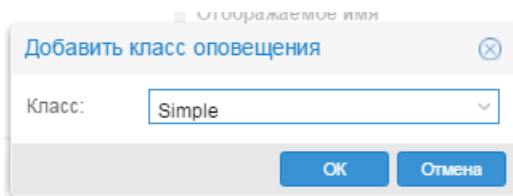
В результате новый пустой набор появится в списке на вкладке **Оповещения**, а рабочая область будет содержать только головной блок параметров;



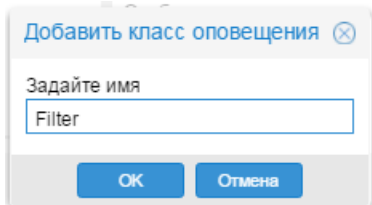
Имя	Syslog
Описание	
Отображаемое имя	

3. чтобы добавить класс оповещений, на вкладке **Оповещения** поместить курсор на только что созданный пустой набор оповещений и выполнить одно из действий:
 - ❖ с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить класс** → **Simple**;
 - ❖ на панели инструментов нажать кнопку , в появившемся окне в списке выбрать класс **Simple** и нажать кнопку **ОК**;

9. Администрирование ПК Нувола через Web-интерфейс

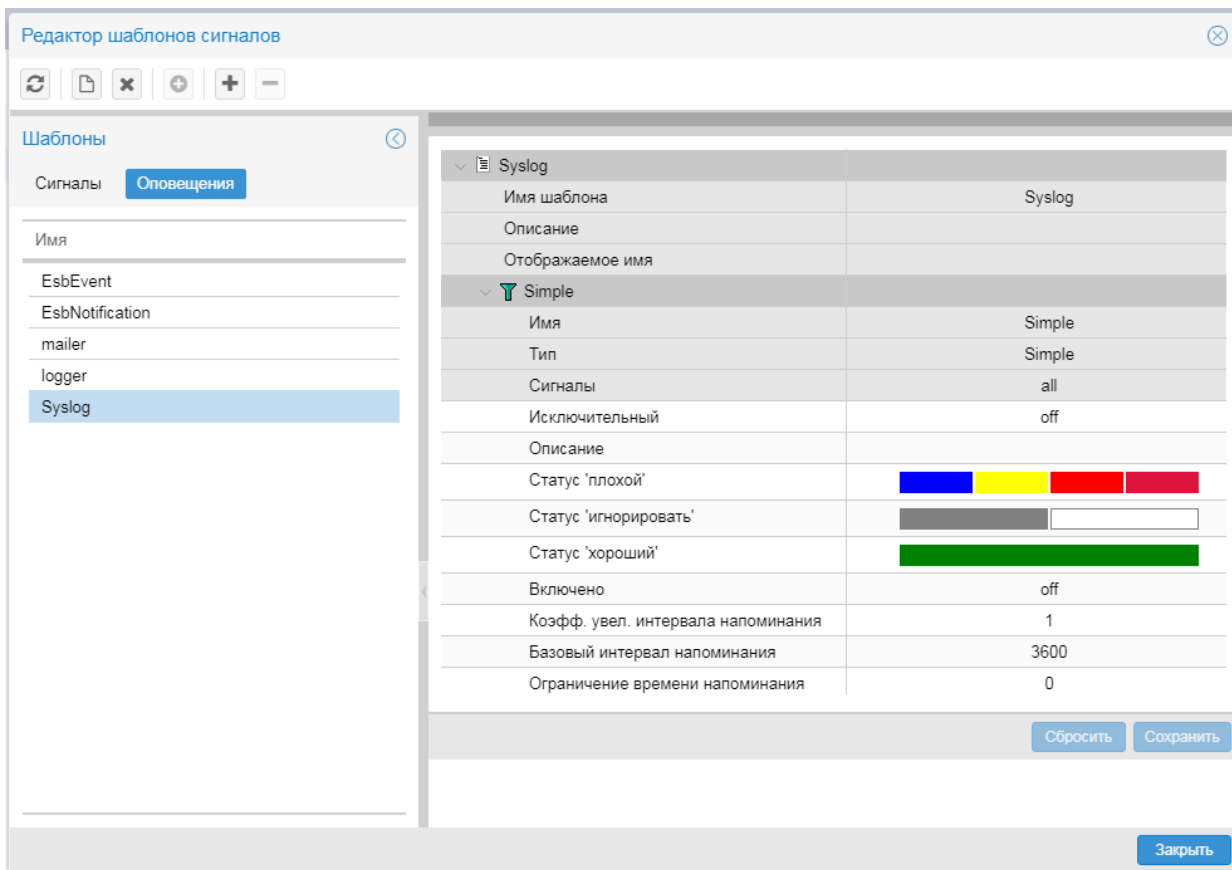


4. в появившемся окне ввести название класса оповещений и нажать **OK**;



в результате в рабочей области сформируется вложенный блок с параметрами класса оповещений;

5. задать глобальные значения параметров класса оповещений либо оставить значения по умолчанию без изменений; подробное описание параметров см. в справочнике [Параметры класса оповещений Simple](#)²⁸⁷;
6. нажать кнопку **Сохранить**.




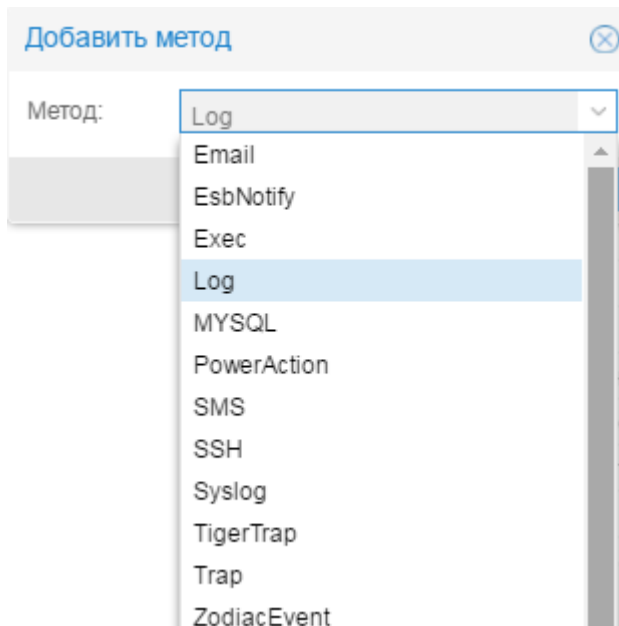
Затем необходимо в набор оповещений [добавить методы оповещения](#)¹⁴⁴.

9.5.1.2. Добавление метода оповещений

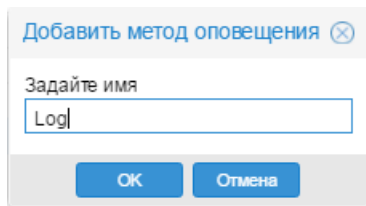
Для добавления метода к набору оповещений следует:

1. на вкладке **Оповещения** поместить курсор на набор оповещений, к которому нужно добавить метод оповещений;
2. выполнить одно из действий:

- ❖ в контекстном меню выбрать пункт **Оповещения** → **Добавить метод** и далее – подпункт, соответствующий типу оповещений;
- ❖ на панели инструментов нажать кнопку , в появившемся окне в списке **Метод** выбрать тип оповещений и нажать кнопку **ОК**;



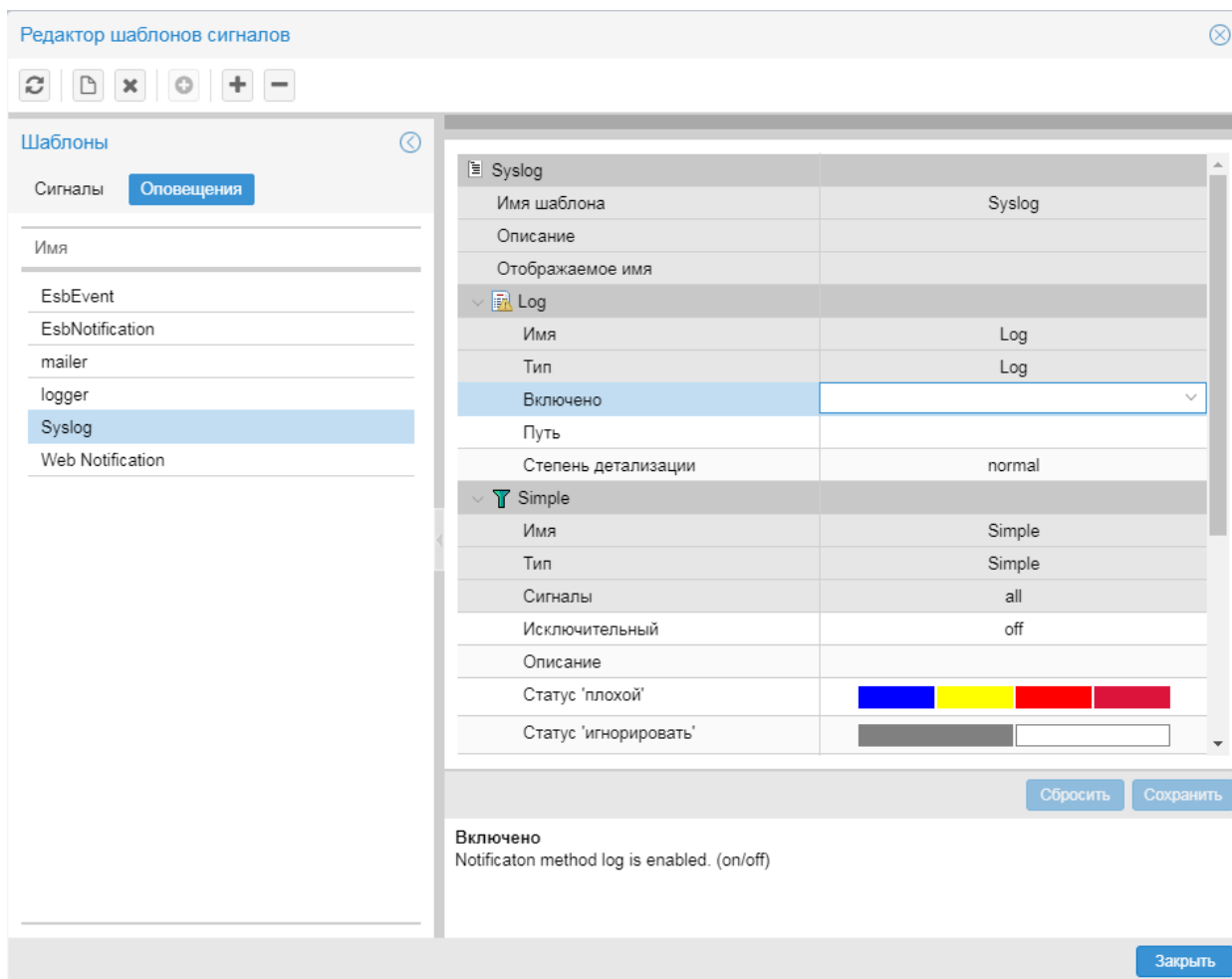
3. в появившемся окне ввести имя метода и нажать кнопку **ОК**;



в результате в рабочей области сформируется вложенный блок с параметрами метода оповещений;

4. ввести глобальные значения параметров метода оповещений. Набор полей зависит от выбранного типа оповещений. Подробное описание параметров см. в справочнике [Типы оповещений](#)^[288].


9. Администрирование ПК Нувола через Web-интерфейс

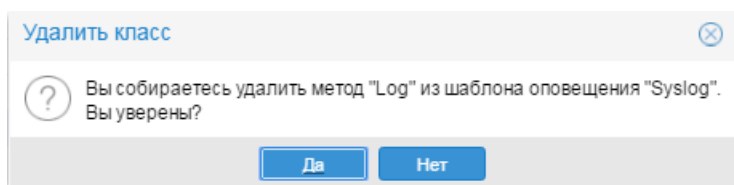


5. после ввода параметров метода нажать кнопку **Сохранить**.

9.5.1.3. Удаление метода/класса оповещений из набора оповещений

Для удаления метода/класса из набора оповещений следует:

1. на вкладке **Оповещения** поместить курсор на набор оповещений, из которого требуется удалить метод или класс;
2. выполнить одно из действий:
 - ❖ с помощью правой кнопки мыши в контекстном меню выбрать пункт **Удалить метод / Удалить класс** и далее – подпункт с названием метода / класса оповещений;
 - ❖ на панели инструментов нажать кнопку  и в появившемся окне в списке выбрать метод / класс;
3. в появившемся окне для подтверждения удаления нажать кнопку **Да**, для отказа – кнопку **Нет**.



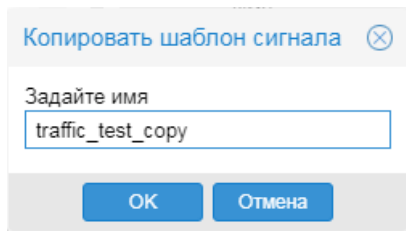
В результате блок с параметрами метода / класса оповещений будет удален из рабочей области.

9.5.1.4. Копирование набора оповещений

Для ускорения и упрощения настройки нового набора оповещений можно создать копию существующего набора оповещений и затем внести в него изменения.

Чтобы создать копию набора оповещений, следует:

1. на вкладке **Оповещения** поместить курсор на набор сигналов, который нужно скопировать;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Копировать шаблон**;
3. в появившемся окне ввести имя нового набора сигналов; по умолчанию предлагается **<имя исходного набора сигналов>_copy**;



4. нажать кнопку **ОК**.

В результате будет создан новый набор оповещений, идентичный исходному набору оповещений.

Для дальнейшей настройки набора сигналов можно выполнить следующие действия:

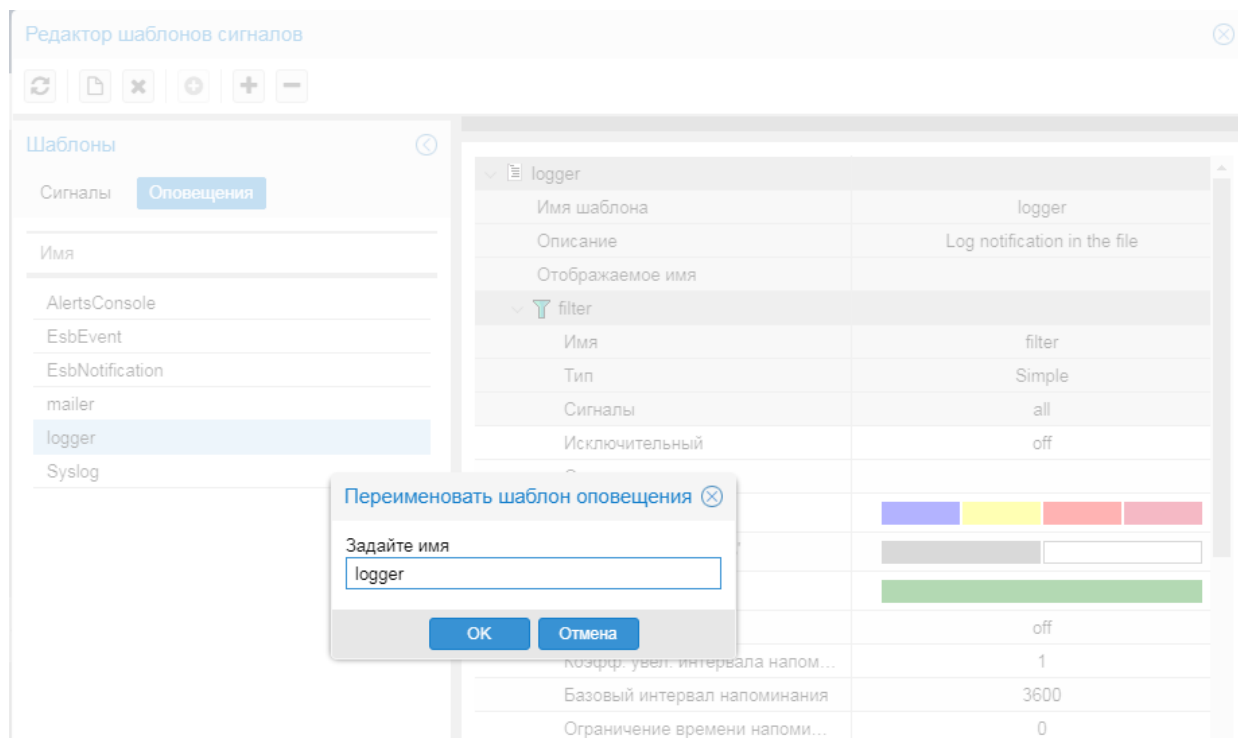
- ❖ [Переименование набора оповещений](#)^[147]
- ❖ [Добавление метода оповещений](#)^[144]
- ❖ [Изменение глобальных значений параметров метода / класса оповещений](#)^[148]

9.5.1.5. Переименование набора оповещений

Переименование набора оповещений – это изменение его отображаемого имени.

Чтобы переименовать набор оповещений, следует:

1. на вкладке **Оповещения** поместить курсор на нужный набор сигналов;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Переименовать шаблон** или нажать клавишу **F2**;
3. в появившемся окне ввести новое название набора сигналов;
4. нажать кнопку **ОК**.

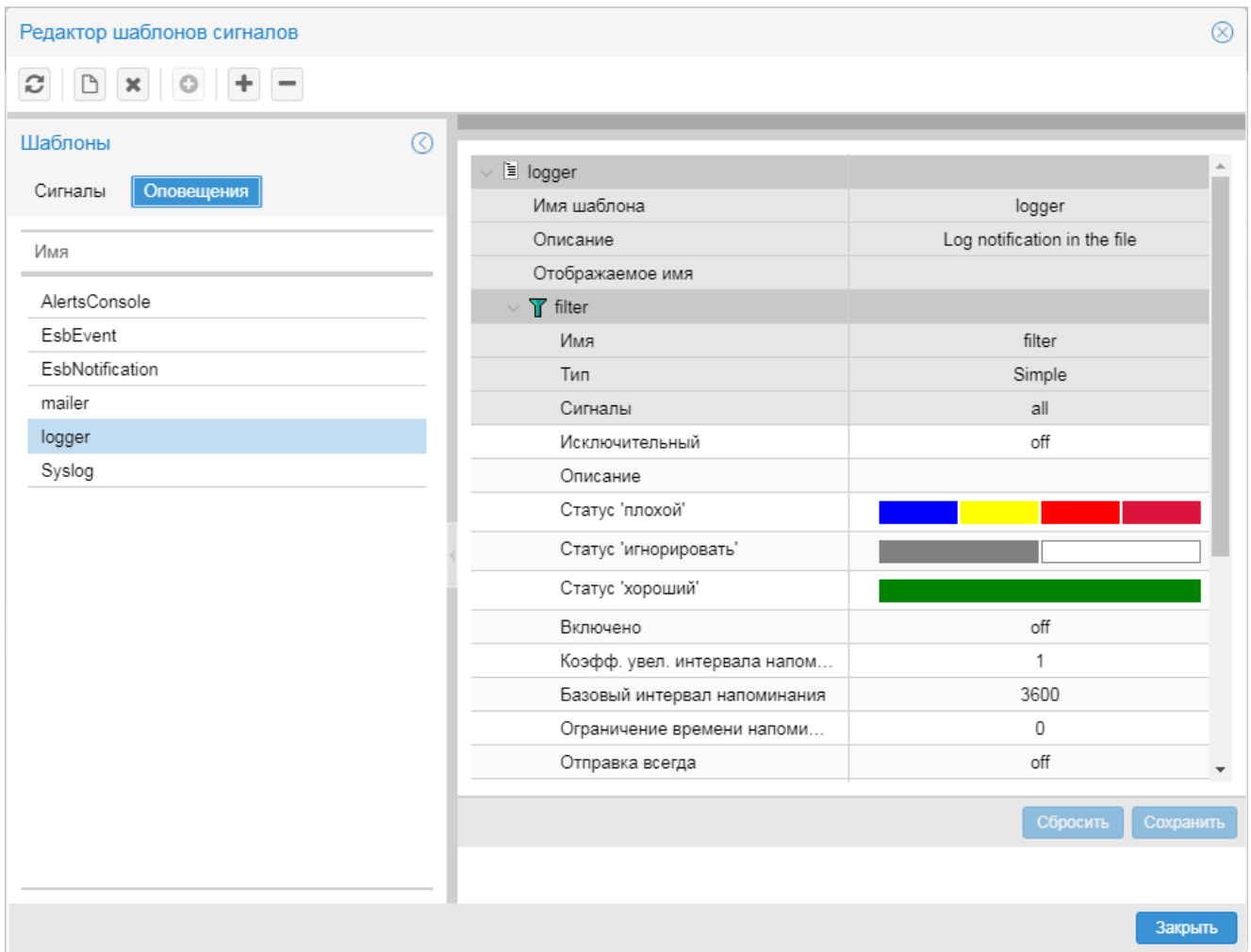


9.5.1.6. Изменение глобальных значений параметров метода / класса оповещений

Измененные глобальные значения параметров метода/класса оповещений распространяется на целевые объекты, к которым привязан соответствующий набор оповещений без указания локальных значений.

Для изменения глобальных значений параметров метода / класса оповещений следует:

1. на вкладке **Оповещения** поместить курсор интересующий набор оповещений;
2. в рабочей области раскрыть блок с параметрами нужного метода / класса оповещений и изменить значения полей во втором столбце. Поля в блоке метода оповещений формируются в соответствии с типом оповещений. Поля на затененном фоне недоступны для редактирования. Подробное описание полей см. в справочниках: [Параметры класса оповещений Simple](#)^[287] и [Типы оповещений](#)^[288];
3. для сохранения изменений нажать кнопку **Сохранить**, для отказа от изменений – кнопку **Сбросить**.

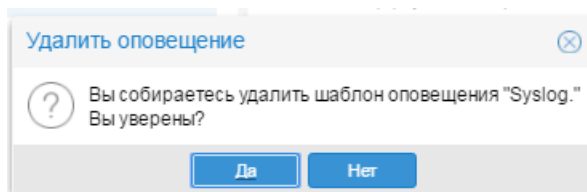


9.5.1.7. Удаление набора оповещений

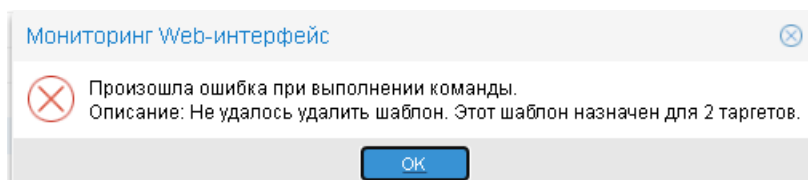
Допускается удаление наборов оповещений, не прикрепленных к целевым объектам.

Чтобы удалить набор оповещений, следует:

1. на вкладке **Оповещения** поместить курсор на нужный набор сигналов и с помощью правой кнопки мыши в контекстном меню выбрать пункт **Удалить шаблон** либо нажать кнопку на панели инструментов или клавишу **Del**;
2. в появившемся окне для подтверждения удаления нажать кнопку **Да**, для отказа – кнопку **Нет**.



Если набор оповещений не прикреплен ни к одному целевому объекту, то он будет удален из списка на вкладке **Оповещения**. В противном случае выдаётся сообщение об ошибке.



Для возможности удаления такого набора оповещений следует сначала [открепить](#)^[152] его от целевых объектов.

9.5.2. Назначение наборов оповещений целевым объектам

Прикрепление наборов оповещений к целевым объектам и изменение настроек привязки (т.е. изменение локальных значений параметров) пользователь осуществляет через контекстное меню окна **Дерево конфигурации**.

Примечание. В контекстном меню формируются соответствующие пункты, только если для ветви дерева конфигурации в группе пользователей установлено право доступа **Привязка сигналов/оповещений**.

Администратор может выполнять следующие действия:

- ❖ [Привязка набора оповещений к целевым объектам](#)^[150]
- ❖ [Открепление набора оповещений от целевого объекта](#)^[152]
- ❖ [Изменение локальных значений параметров оповещений](#)^[153]

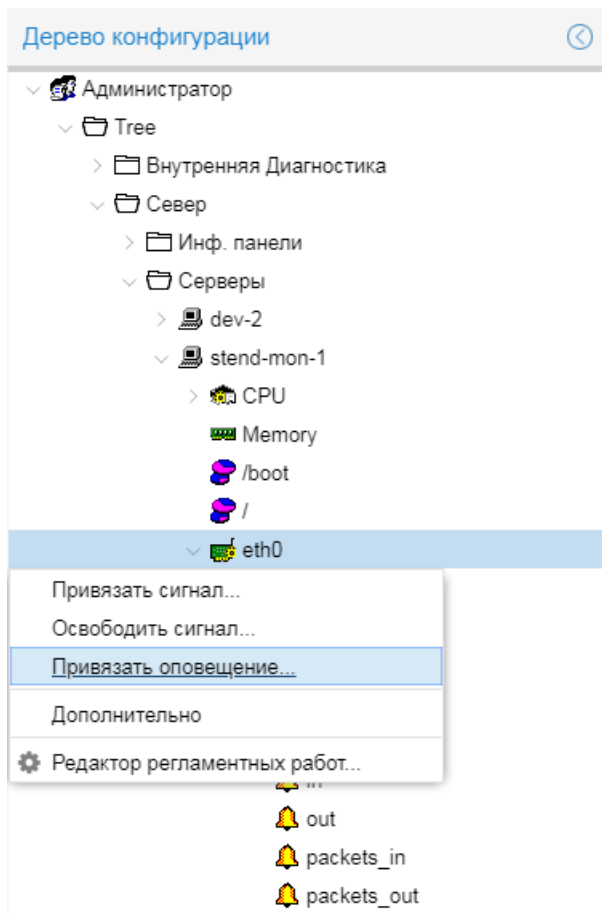
9.5.2.1. Привязка набора оповещений к целевым объектам

К одному целевому объекту можно привязать несколько наборов оповещений. После того как набор оповещений привязан к целевому объекту, при необходимости можно установить специфические настройки применения набора оповещений, для этого следует задать локальные значения параметров оповещений.

Осуществлять привязку наборов оповещений можно как к конкретному целевому объекту, так и сразу ко всем целевым объектам в рамках выбранного устройства, папки или учетной записи.

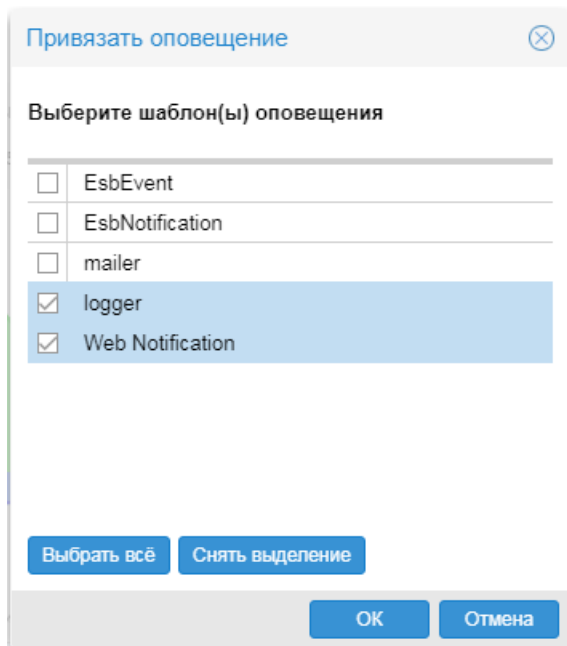
Для выполнения этой операции следует:

1. в окне **Дерево конфигурации** поместить курсор на узел дерева, для которого нужно осуществить привязку наборов оповещений; таким узлом может быть учетная запись, папка, устройство, целевой объект.
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Привязать оповещение**;



в результате открывается окно **Привязать оповещение** со списком наборов оповещений;

3. отметить флагом один или несколько наборов оповещений для привязки к целевым объектам в рамках узла, выбранного на шаге 1;



4. нажать кнопку **ОК**.

В результате в окне **Дерево конфигурации** привязанные наборы оповещений будут показаны как дочерние узлы целевых объектов. На пиктограммах целевых объектов, к которым были привязаны наборы оповещений, появится белый значок.

9. Администрирование ПК Нувола через Web-интерфейс

Дерево конфигурации

Шаблон оповещения: Web Notification

WebNotification	
Имя шаблона	WebNotification
Описание	
Отображаемое имя	Web Notification
filter	
Имя	filter
Тип	Simple
Сигналы	all
Исключительный	off
Описание	
Статус 'плохой'	
Статус 'игнорировать'	
Статус 'хороший'	
Включено	on
Козфф. увел. интервала напоминания	1
Базовый интервал напоминания	3600
Ограничение времени напоминания	0
Отправка при любом изменении статуса	off
Отправка при переходе хороший-плохой	on
Отправка при переходе плохой-хороший	off
Отправка при переходе плохой-плохой	off
Задержка при переходе в 'плохой'	0
Задержка при переходе в 'хороший'	0
method	
Имя	method
Тип	WsNotify
Включено	on
Шаблон	{{ target_display_name }}: {{ gettext(state_str) }} {{ state_b...
Автоматическое сокрытие	on
Задержка	3000

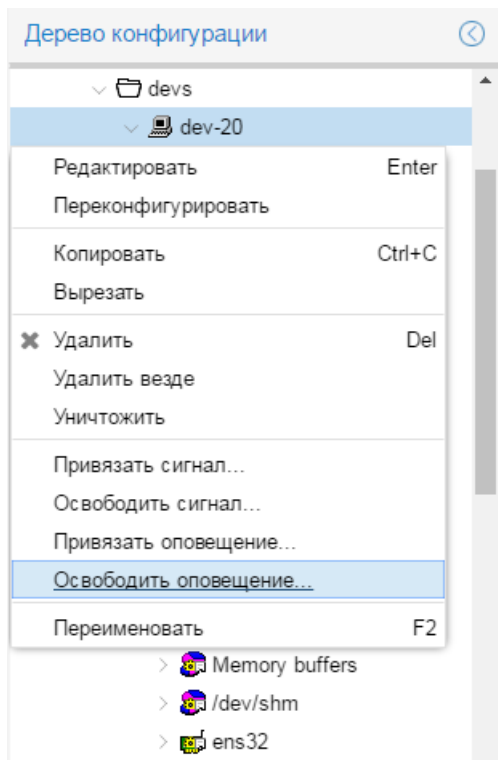
Проверить Сбросить Сохранить

9.5.2.2. Открепление набора оповещений от целевого объекта

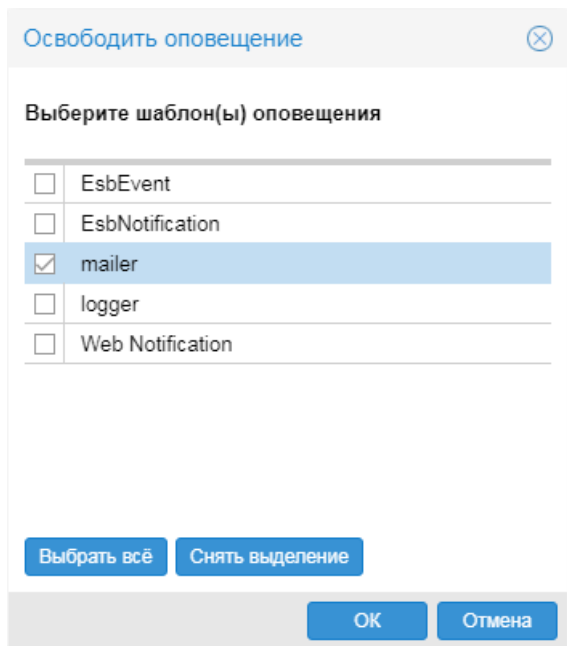
Осуществлять открепление наборов оповещений можно как от конкретного целевого объекта, так и сразу от всех целевых объектов в рамках выбранного устройства, папки или учетной записи.

Для выполнения этой операции следует:

1. в окне **Дерево конфигурации** поместить курсор на объект дерева, для которого нужно осуществить открепление наборов оповещений; таким узлом может быть учетная запись, папка, устройство, целевой объект;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Освободить оповещение**;



3. в появившемся окне отметить один или несколько набор оповещений и нажать кнопку **ОК**.



В результате отмеченные наборы оповещений будут откреплены от всех целевых объектов в рамках выбранного объекта дерева конфигурации. В окне **Дерево конфигурации** с пиктограмм целевых объектов, от которых откреплены все наборы оповещений, исчезнут белые значки.

9.5.2.3. Изменение локальных значений параметров оповещений

В наборе оповещений администратор устанавливает [глобальные значения](#)^[148] параметров класса / метода оповещений. После того как набор оповещений привязан к целевому объекту, пользователь с правами администрирования имеет возможность задать локальные значения, которые будут применяться только для данного целевого объекта.

Для изменения локальных значений параметров класса / метода оповещений следует:

9. Администрирование ПК Нуволы через Web-интерфейс

1. в окне **Дерево конфигурации** поместить курсор набор оповещений, привязанный к целевому объекту, либо на входящий в его состав класс или метод оповещений. В зависимости от выбранного узла в области вывода данных будет показана таблица с параметрами набора оповещений и/или входящих в его состав класса и метода (или нескольких методов):

- ❖ первый столбец – наименование параметра;
- ❖ второй столбец – локальное значение параметра (доступны для изменения);
- ❖ третий столбец – глобальное значение параметра, заданное администратором (недоступны для изменения);

Поля формируются в соответствии с типом оповещений. Серые поля недоступны для редактирования. Детальное описание и назначение параметров см. в справочниках: [Параметры класса оповещений Simple](#)^[287] и [Типы оповещений](#)^[288];

2. изменить локальные значения во втором столбце (по нажатию кнопки **Сбросить** во второй столбец возвращаются последние сохраненные значения);

Имя шаблона	WebNotification
Описание	
Отображаемое имя	Web Notification
filter	
Имя	
Тип	
Сигналы	
Исключительный	
Описание	
Статус 'плохой'	
Статус 'игнорировать'	
Статус 'хороший'	
Включено	
Козфф. увел. интервала напоминания	
Базовый интервал напоминания	
Ограничение времени напоминания	
Отправка при любом изменении статуса	
Отправка при переходе хороший-плохой	
Отправка при переходе хороший-хороший	
Отправка при переходе плохой-хороший	
Отправка при переходе плохой-плохой	
Задержка при переходе в 'плохой'	
Задержка при переходе в 'хороший'	
method	
Имя	
Тип	
Включено	
Шаблон	
Автоматическое сокрытие	
Задержка	

3. по окончании ввода значений для сохранения внесенных изменений нажать кнопку **Сохранить**.

9.6. Комплекс внутренней диагностики ПМ Нуволы Топаз

Администратор root может контролировать работу ПМ Нуволы Топаз с помощью *комплекса внутренней диагностики*.

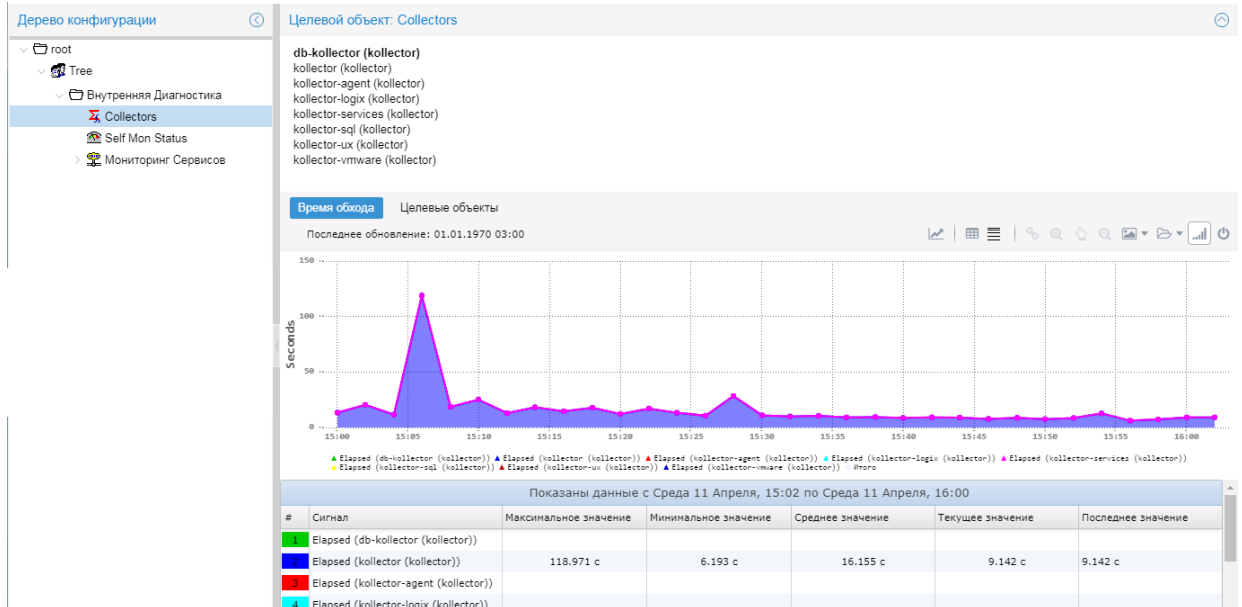
Комплекс внутренней диагностики – совокупность предустановленных объектов, предоставляющих информацию о функционировании ПМ Нуволы Топаз: состояния и статистику системных процессов, состояния основных портов, состояния компонент сервера, на котором установлен ПМ Нуволы Топаз.

Комплекс создается автоматически при установке модуля Нуволы Топаз. Информация, предоставляемая объектами комплекса, доступна администратору root в Web-интерфейсе.

9.6.1. Структура комплекса внутренней диагностики

По умолчанию комплекс размещается в папке **Tree/Внутренняя Диагностика** и включает в себя три объекта:

- ❖ **Collectors** – сводный график с информацией о состоянии системных процессов, созданных на основе процесса collector;



- ❖ **Self Mon Status** – информационная панель, отображающая состояния компонент сервера, на котором установлен модуль **Нувола Топаз**, состояния основных портов, состояния и статистику основных системных процессов;



9. Администрирование ПК Нуволы через Web-интерфейс

- ❖ **Мониторинг Сервисов** – виртуальный объект (`Devices::Analysis::Lancelot`), целевыми объектами которого являются системные процессы модуля **Нуволы Топаз** и компоненты сервера с установленным модулем **Нуволы Топаз**.

The screenshot shows the web interface for 'Мониторинг Сервисов'. On the left is a 'Дерево конфигурации' (Configuration Tree) with a tree view showing the hierarchy: root > Tree > Внутренняя Диагностика > Мониторинг Сервисов. The right pane shows the configuration for the device 'Мониторинг Сервисов'. It includes system information (Linux dev-2ad.cc.ru 3.10.0-020.18.0.el7.x86_64 #1 SMP Thu May 12 11:03:55 UTC 2016 x86_64) and several monitoring sections:

- Thaw**: Table with columns: Объект контроля, Интервал, Время обхода (послед.), Up (послед.), Down (послед.). Row: thaw (thawne.pl), 600.00, 48.81 сек., 153, 4.
- Rest**: Table with columns: Объект контроля, Rest Workers (Максимальное), Busy Workers (Максимальное). Rows: rest_long_api (rest.pl) with 3 Rest Workers and 0 Busy Workers; rest_api (rest.pl) with 3 Rest Workers and 1 Busy Worker.
- Query makers**: Table with columns: Объект контроля, Интервал, Целевые объекты (послед.), Время обхода (послед.). Rows: querymaker (querymaker.pl) with 3.60 k interval and NaN targets; pinger (querymaker.pl) with 120.00 interval and NaN targets.
- Regular collectors**: Table with columns: Объект контроля, Интервал, Время обхода (послед.), Целевые объекты (послед.), Ошибки (послед.). Row: kollector-vmware (kollector) with 120.00 interval and NaN targets.

9.6.2. Настройка комплекса внутренней диагностики

Для создания/переконфигурации комплекса предназначена утилита [selfMonitoringConfig.pl](#)^[263]. При установке модуля **Нуволы Топаз** она запускается автоматически и создает комплекс в уменьшенном виде: у виртуального объекта **Мониторинг Сервисов** отсутствуют некоторые основные процессы, а также все диски, кроме `\` и `Swap space`.

Поэтому после установки модуля **Нуволы Топаз** рекомендуется переконфигурировать комплекс внутренней диагностики.

Команда для создания/переконфигурации комплекса с именами на русском языке:

```
./selfMonitoringConfig.pl
```

или

```
./selfMonitoringConfig.pl -f 'Внутренняя Диагностика' -n 'Мониторинг Сервисов'
```

Команда для создания/переконфигурации комплекса с именами на английском языке:


```
./selfMonitoringConfig.pl -f 'Self Monitoring' -n 'Services Monitoring'
```

При необходимости администратор `root` может изменить настройки сигналов и оповещений, предустановленные для целевых объектов виртуального объекта **Мониторинг Сервисов**, в [Редакторе сигналов и оповещений](#)^[117].

9.7. Диспетчер процессов

Приложение **Диспетчер процессов** – компонента Web-интерфейса, позволяющая администратору посредством графического пользовательского интерфейса управлять системными процессами, функционирующими на сервере Нуволы и обеспечивающими работу ПМ **Нуволы Топаз** и других модулей ПК Нуволы.

Функции приложения могут быть также выполнены с помощью утилиты [overlord.pl](#)^[261] ПМ **Нуволы Топаз**.

Для вызова **Диспетчера процессов** следует в заголовке Web-интерфейса нажать кнопку  и в меню приложений нажать иконку с названием приложения.

Примечание. Приложение доступно, если пользователь входит в группу пользователей с глобальной политикой **Управление сервисами**.

Приложение открывается в новой вкладке браузера.

9.7.1. Основное окно

Основное окно приложения **Диспетчер процессов**, открываемое в отдельной вкладке браузера, содержит следующие компоненты:

- ❖ заголовок;
- ❖ панель инструментов;
- ❖ таблицу системных процессов;
- ❖ окно **Свойства** с параметрами системных процессов;
- ❖ окно **Журнал** ¹⁶⁵ для просмотра лог-файлов системных процессов; по умолчанию свернуто.

Имя системного процесса ↑	Статус	Процесс (daemon)	Id системного процесса
alerts	UP	alerts.pl	26261
dataserver	DOWN	generic
db-kollector	DSBL	kollector
de_builderd	UP	de_builderd.pl	25755
de_providerd	UP	de_providerd.pl	25742
de_svg-js	UP	de_svg-js.pl	25683
de_svg-simple	UP	de_svg.pl	25665
esb	DOWN	generic
esbevent	UP	kollector	25614
etld	DSBL	bistored.pl
fnmp	UP	kollector	25603
kollector	UP	kollector	25594
kollector-agent	DSBL	kollector
kollector-logix	DSBL	kollector
kollector-services	UP	kollector	25583
kollector-sql	DSBL	kollector
kollector-ux	DSBL	kollector
kollector-vmware	DSBL	kollector
kollector1	UP	kollector	25571
konfigd	UP	konfigd	26142
konfne	DOWN	konfigd
maintenance	UP	maintenance.py	25542
nms	DSBL	konfigd
notifs	UP	notifier.py	25533

Свойства	
Системный процесс	alerts
Процесс (daemon)	alerts.pl
Основные	
Путь к файлам репликации	
Хранение статистики	
Использовать файлы репл...	
Общие	
Уровень журналирования	
Журнал	/usr/local/lancelot/var/lancelot-logs/ale...
Интервал опроса, сек	210
Ветви	
Список	
Дополнительно	
enabled-notifications	^AlertsConsole\$
esb	127.0.0.1
no-esb-alerts	false

Добавить параметр Сохранить Сбросить


Системный процесс
Системный процесс

Журнал

9.7.2. Создание и изменение системного процесса

Создание нового системного процесса

Чтобы создать новый системный процесс, следует:

1. нажать кнопку  на панели инструментов;
2. в появившемся окне выбрать процесс (демон) и нажать кнопку **ОК**;

в результате в таблицу будет добавлена строка с именем, состоящим из имени процесса и цифры (порядкового номера); в столбце **Id системного процесса** до первого сохранения отображается значение **Creating**;

9. Администрирование ПК Нувола через Web-интерфейс

The screenshot shows a web interface with a table of system processes and a 'Свойства' (Properties) panel on the right. The table has columns for 'Имя системного процесса' (Process Name), 'Статус' (Status), 'Процесс (daemon)' (Process (daemon)), and 'Id системного процесса' (Process ID). The 'alerts1' process is highlighted in blue. The 'Свойства' panel shows the 'Системный процесс' (System process) field set to 'alerts1' and other parameters like 'Уровень журналирования' (Logging level) set to 'Error' and 'Интервал опроса, сек' (Polling interval, sec) set to '210'.

Имя системного процесса	Статус	Процесс (daemon)	Id системного процесса
alerts	UP	alerts.pl	26261
alerts1	DOWN	alerts	Creating
dataserver	DOWN	generic	----
db-kollector	DSBL	kollector	----
de_builderd	UP	de_builderd.pl	25755
de_providerd	UP	de_providerd.pl	25742
de_svg-js	UP	de_svg-js.pl	25683
de_svg-simple	UP	de_svg.pl	25665
esb	DOWN	generic	----
esbevent	UP	kollector	25614
etlid	DSBL	bistored.pl	----
fnmp	UP	kollector	25603
kollector	UP	kollector	25594
kollector-agent	DSBL	kollector	----
kollector-logix	DSBL	kollector	----
kollector-services	UP	kollector	25583

3. в окне **Свойства** в поле **Системный процесс** отредактировать имя системного процесса;

ВНИМАНИЕ! После сохранения системного процесса изменить его имя невозможно!

4. в окне **Свойства** в остальных полях ввести значения параметров системного процесса или оставить значения по умолчанию без изменений;
5. в окне **Свойства** нажать кнопку **Сохранить**.

Изменение параметров системного процесса

Для изменения доступны все параметры системного процесса, за исключением его имени (см. параметр **Системный процесс** в окне **Свойства**). Параметры можно изменять вне зависимости от статуса системного процесса. При изменении параметров у запущенного системного процесса автоматически выполняется его перезапуск.

Чтобы изменить параметры системного процесса, необходимо выполнить следующие действия:

1. поместить курсор на нужный системный процесс;
2. в окне **Свойства** ввести значения параметров;
3. в окне **Свойства** нажать кнопку **Сохранить**.

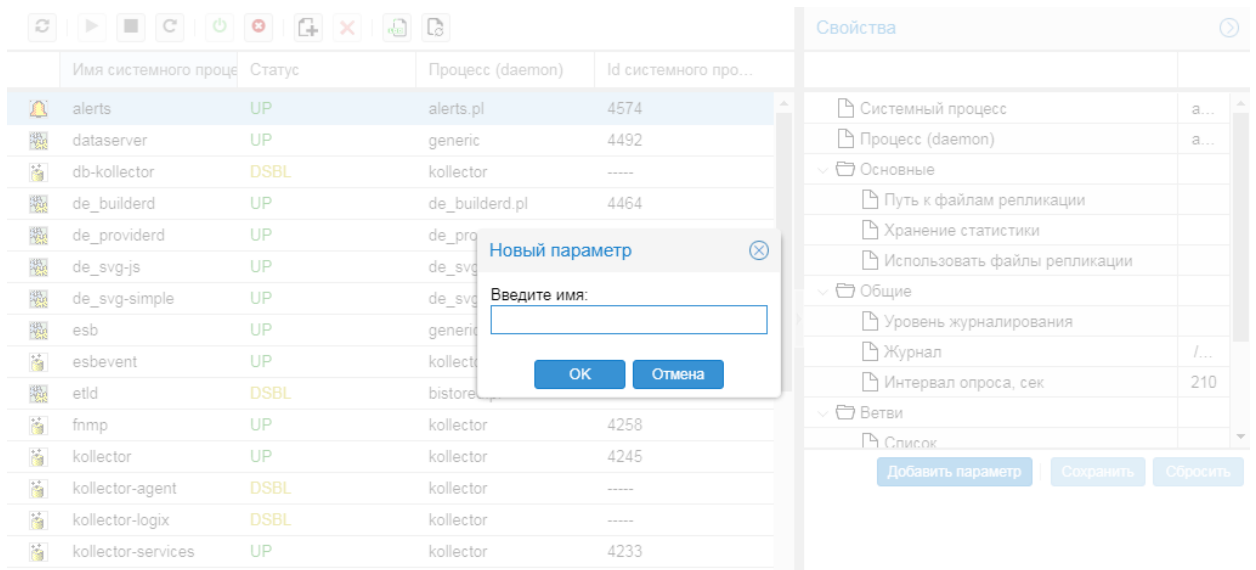
9.7.3. Параметры системных процессов

Окно **Свойства** позволяет ввести параметры нового системного процесса и изменить параметры существующего системного процесса.

В окне **Свойства** отображаются параметры системного процесса, выбранного в списке. Параметры объединены в блоки. В нижней части окна **Свойства** выводится описание параметра, выбранного в данный момент. Если пользователь изменил значение параметра, но еще не сохранил его, то соответствующая ячейка помечается красным треугольником в левом верхнем углу.

Параметры системных процессов можно условно разделить на две группы:

- ❖ стандартные – задаются для большинства системных процессов;
- ❖ специфические – задаются для одного или нескольких системных процессов. Специфические параметры описаны в справочнике [Системные процессы модуля Нувола Топаз](#)²⁶⁷. При необходимости (по согласованию с разработчиками) можно добавить параметры в блок **Дополнительно** с помощью кнопки **Добавить параметр**.



Стандартные параметры системных процессов

Над блоками параметров находятся поля, которые задаются при создании системного процесса и в дальнейшем для редактирования недоступны.

Системный процесс

Имя системного процесса.

Процесс (даемон)

Процесс (даемон), определяющий функциональность системного процесса. Тип `generic` означает, что системный процесс реализован средствами языка C++.

В блоке **Общие** содержатся параметры, стандартные для всех системных процессов.

Уровень журналирования | `loglevel`

Уровень логирования – степень детализации информации в лог-файле. Возможные значения:

Info – базовая информация (наиболее эффективный уровень);

Debug – детальная информация (применяется на этапе разработки/внедрения);

Warn – информация о предупреждениях;

Error – информация об ошибках.

Журнал* | `logfile`

Директория и имя лог-файла для записи отладочной информации. Не рекомендуется изменять директорию хранения лог-файла по умолчанию `/usr/local/lancelot/var/lancelot-logs/`. Системный процесс создает отдельный файл для каждого потока программы с указанием в имени файла номера PID программы, порядкового номера потока и номера файла в данном потоке.

Интервал опроса, сек | `interval`

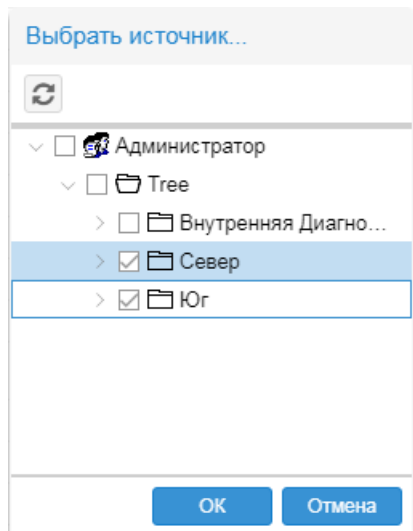
Периодичность работы системного процесса, в секундах.

В блоке **Ветви** в поле **Список** (`list`) можно указать ветви дерева конфигурации, с которыми будет работать системный процесс. По умолчанию системный процесс применяется ко всему дереву конфигурации.

Чтобы указать ветви дерева, следует:

9. Администрирование ПК Нувола через Web-интерфейс

1. дважды кликнуть мышью в поле **Список**;
2. в появившемся окне **Выбрать источник...** отметить нужные узлы дерева или снять отметки;



3. нажать кнопку **ОК**. В результате выбранные узлы дерева будут добавлены в поле **Список**.



Для удаления ветви дерева из списка также следует использовать окно **Выбрать источник...**, как описано выше.

Блок **Дополнительно**, помимо специфичных параметров, также содержит и стандартные параметры (они отсутствуют только у системных процессов `generic`).

esb | `esb`

IP-адрес хоста, на котором работает шина `esb` (по умолчанию **127.0.0.1**). Чтобы отключить системный процесс от шины [ESB](#)^[18], необходимо ввести значение **false**.

no-esb-alerts | `no-esb-alerts`

Параметр влияет на работу модуля **Нувола Топаз** только при использовании сервисной шины [ESB](#)^[18]. Возможные значения:

- ❖ `false` (по умолчанию) – конфигурация сигналов хранится на сервере конфигураций, а значения сигналов – на сервере данных; системный процесс `alerts.pl` получает данные из шины ESB, и данные по изменению состояния сигнала передает обратно в ESB;
- ❖ `true` – конфигурация сигналов хранится в виде файлов на диске; системный процесс `alerts.pl` обменивается данными через шину ESB.

9.7.3.1. Настройка расписания системного процесса

Для некоторых системных процессов (в частности, для `querymaker`) можно настроить расписание выполнения. Настройка осуществляется в универсальном окне **Планировщик задач**, которое вызывается из окна **Свойства** по двойному клику мыши в поле **Расписание работы** (блок **Основные**).

Планировщик задач

Минута 0

Час 0

День 0

Месяц 0

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Запланированные задания:

Минута	Час	Месяц	День недели

Добавить

Редактировать

Удалить

Удалить всё

OK Отмена

Структура универсального окна Планировщик задач

Таблица **Запланированные задания** содержит строки, отображающие периодичность выполнения процесса.

В верхней части окна расположены управляющие элементы для создания строк расписания.

Положение каждой из кнопок **Минута**, **Час**, **День**, **Месяц** определяет функциональное назначение соседнего с ней поля:

- ❖ если кнопка не нажата, то введенное в поле значение указывает на момент времени;
- ❖ если кнопка нажата, то введенное в поле значение указывает на периодичность. В название кнопки добавляется слово (**шаг**), а введенное в поле значение указывает на временной интервал, через который выполняется процесс.

Значения в поля рядом с кнопками **Минута**, **Час**, **День**, **Месяц** можно ввести вручную или с помощью бегунка.

Флаги **День недели** показывают дни недели, в которые выполняется процесс.

В правой верхней части окна расположены кнопки для управления таблицей **Запланированные задания**:

- ❖ **Добавить** – создание новой строки;
- ❖ **Редактировать** – сохранение изменений для выбранной строки;
- ❖ **Удалить** – удаление выбранной строки;
- ❖ **Удалить всё** – удаление всех строк.

9. Администрирование ПК Нувола через Web-интерфейс

Планировщик задач

Минута:

Час:

День:

Месяц:

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Добавить

Редактировать

Удалить

Удалить всё

Запланированные задания:

Минута	Час		Месяц	День недели
*	10	*	*	2
*	22	*	*	0

OK Отмена

Создание расписания и изменение его строк

Чтобы создать расписание выполнения процесса, в универсальном окне следует:

- на панели настройки с помощью управляющих элементов задать периодичность;
 - Пример 1: процесс выполняется "Каждый день в 13:15"

В этом случае следует:

- вручную или с помощью бегунка установить значения **Час=13** и **Минута=15** (кнопки **Час** и **Минута** не должны быть нажаты),
- отметить флаг **Все** (все дни недели).

Планировщик задач

Минута:

Час:

День:

Месяц:

День недели: Все Пн Вт Ср Чт Пт Сб Вс

Добавить

Редактировать

Удалить

Удалить всё

Запланированные задания:

Минута	Час		Месяц	День недели
15	13	*	*	*

OK Отмена

для сохранения заданных параметров нажать кнопку **Добавить**; в результате в таблице **Запланированные задания** будет создана строка расписания;

3. при необходимости с помощью кнопки **Добавить** создать новую строку и повторить пп. 1 и 2;
4. чтобы сохранить данные в универсальном окне настройки расписания и вернуться в предыдущее окно, нажать кнопку **ОК**.

Изменение существующего расписания осуществляется аналогичным образом. Чтобы в окне **Планировщик задач** изменить строку в расписании, следует:

1. в таблице **Запланированные задания** поместить курсор на строку;
2. с помощью управляющих элементов изменить параметры периодичности;
3. для сохранения нажать кнопку **Редактировать**.

Чтобы сохранить данные в универсальном окне настройки расписания и вернуться в предыдущее окно, следует нажать кнопку **ОК**.

Удаление строк из расписания

В универсальном окне настройки расписания для удаления одной выбранной строки из расписания предназначена кнопка **Удалить**, для удаления сразу всех строк – кнопка **Удалить всё**.

Например, чтобы удалить строку из расписания, необходимо выполнить действия:



1. открыть универсальное окно настройки расписания;
2. поместить курсор на строку расписания;
3. нажать кнопку **Удалить**;
4. чтобы сохранить изменение и закрыть окно, нажать кнопку **ОК**.

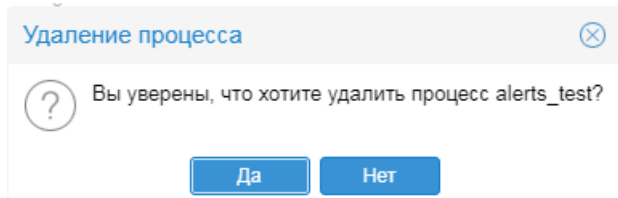
9.7.4. Удаление системного процесса

Системный процесс можно удалить, только если он находится в статусе **down** или **disabled**.

Системные процессы на основе процесса [konfigd^{\[277\]}](#) удалению не подлежат.

Для удаления системного процесса следует:

1. убедиться, что системный процесс остановлен или отключен (в столбце **Статус** указано значение **down** или **disabled**). Если системный процесс запущен, остановить его с помощью кнопки ;
2. поместить курсор на строку с остановленным системным процессом;
3. нажать кнопку ;
4. в появившемся окне для подтверждения действия нажать кнопку **Да**.



В результате системный процесс будет удален.

9.7.5. Управление системными процессами

Действия по управлению системными процессами сопряжены, в первую очередь, с изменением их статусов. Текущие статусы системных процессов отображаются в столбце **Статус**.

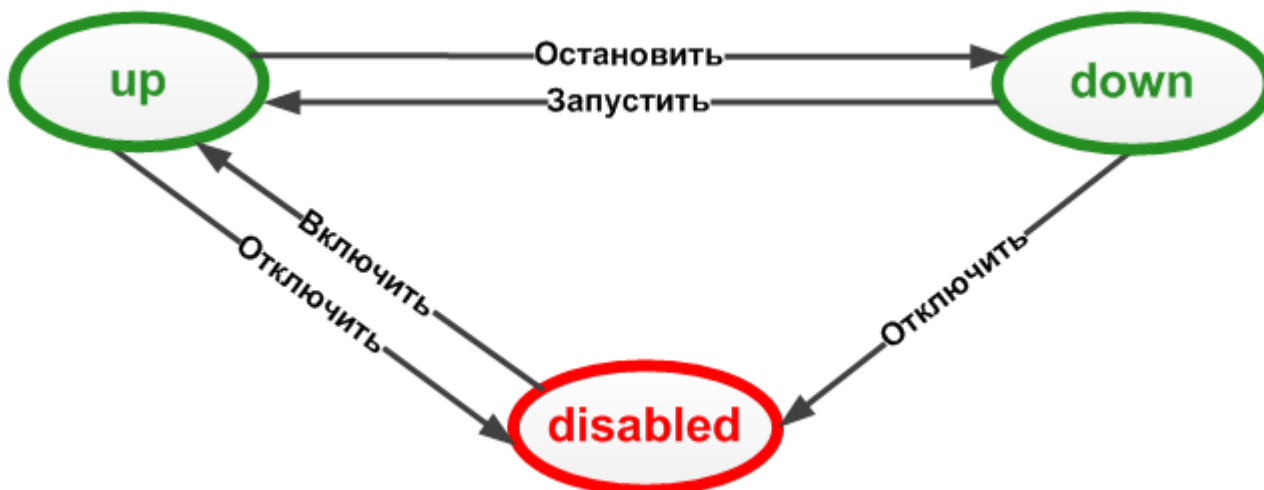
9. Администрирование ПК Нувола через Web-интерфейс

Статус **up** – системный процесс запущен.

Статус **down** – системный процесс остановлен, но доступен для ручного или автоматического запуска.


Статус **disabled** – системный процесс отключен и не доступен для автоматического запуска.

На диаграмме схематично показаны операции, доступные пользователю/администратору в **Диспетчере процессов**. Работающий системный процесс можно также перезапустить, т.е. последовательно остановить его, а затем запустить (операция перезапуска на диаграмме не показана во избежание загромождения).



Запуск системного процесса


Чтобы запустить системный процесс, необходимо выполнить следующие действия:

1. в таблице поместить курсор на нужный системный процесс со статусом **down**;
2. нажать кнопку  на панели инструментов.

В результате на сервер Нувола отправляется команда для запуска системного процесса. Успешно запущенный системный процесс получает статус **up**.

Остановка системного процесса


Остановка системного процесса означает перевод его из статуса **up** в статус **down**. Чтобы остановить системный процесс, следует:

1. в таблице поместить курсор на нужный системный процесс со статусом **up**;
2. нажать кнопку  на панели инструментов.

Перезапуск системного процесса


Перезапуск – это последовательное выполнение команд остановки и запуска системного процесса. После перезапуска системный процесс находится в статусе **up**. Перезапуск системных процессов следует выполнять, например, после обновления лицензии.

Чтобы перезапустить системный процесс, следует:

1. в таблице поместить курсор на нужный системный процесс со статусом **up**;
2. нажать кнопку  на панели инструментов.

Отключение системного процесса


Отключение системного процесса означает перевод его из статуса **up** или **down** в статус **disabled**. Чтобы отключить системный процесс, следует:

1. в таблице поместить курсор на нужный системный процесс со статусом **up** или **down**;
2. нажать кнопку  на панели инструментов.

В результате системный процесс переводится в статус **disabled**, а в окне **Свойства** в блоке **Дополнительно** появляется параметр `disabled=true`.

Включение системного процесса

Включение системного процесса означает перевод его из статуса **disabled** в статус **up**. Чтобы включить системный процесс, следует:

1. в таблице поместить курсор на нужный системный процесс со статусом **disabled**;
2. нажать кнопку  на панели инструментов.

9.7.6. Работа с журналами (лог-файлами) системных процессов


Лог-файл системного процесса модуля **Нувола Топаз** – это текстовый файл, в котором в соответствии с настроенным уровнем протоколирования регистрируются действия, выполняемые системным процессом, ошибки и предупреждения. По умолчанию лог-файлы хранятся на сервере Нувола в директории `/usr/local/lancelot/var/lancelot-logs`.

Содержание и имя лог-файла настраивается с помощью параметров системного процесса: **Уровень журналирования** (`loglevel`) и **Журнал** (`logfile`).

Диспетчер процессов позволяет администратору:

- ❖ [просматривать последние записи лог-файла](#)^[166] (количество отображаемых строк и интервал их автоматического обновления настраивается локальным администратором);
- ❖ [сохранить информацию из журнала](#)^[167], выведенную на вкладку, в отдельный файл на рабочей станции;
- ❖ [выполнить ротацию лог-файла](#)^[167], т.е. закрыть текущий лог-файл и открыть новый.


9.7.6.1. Структура окна Журнал


Окно **Журнал** располагается в нижней части основного окна **Диспетчера процессов**. Первоначально оно свернуто, отображается лишь его заголовок. Чтобы развернуть окно **Журнал**, следует нажать кнопку  на его заголовке.

Лог-файлы по выбранным системным процессам отображаются на отдельных вкладках. Первоначально вкладки отсутствуют.

Окно **Журнал** имеет собственную панель инструментов, содержащую следующие поля и кнопки:


- ❖ **линий** – количество последних строк лог-файла, отображаемых на вкладках;
- ❖ **интервал, сек** – периодичность (в секундах) автоматического обновления отображаемой информации.

Чтобы изменить количество отображаемых строк, следует установить другое значение поля **линий** (первоначально – 10). Тогда при следующем обновлении (в автоматическом режиме или по нажатию кнопки ) будет выведено заданное количество последних строк, содержащихся в лог-файле.

Для удаления информации с вкладки следует нажать кнопку . Информация из лог-файла на сервере Нувола при этом не удаляется.



Режим автоматического обновления


Чтобы установить режим автоматического обновления выводимой информации из журнала, необходимо:

1. в поле **интервал, сек** ввести период обновления в секундах;
2. нажать кнопку .

В результате с заданной периодичностью информация на вкладке будет автоматически обновляться. Поле **интервал, сек** при этом недоступно для редактирования и выполняет функцию таймера, т.е. показывает количество секунд, оставшихся до момента обновления.


Чтобы изменить интервал автоматического обновления, необходимо:

1. нажать кнопку , чтобы отключить режим автоматического обновления;
2. в поле **интервал, сек** ввести нужное значение;
3. еще раз нажать кнопку , чтобы включить режим автоматического обновления.

Для отключения режима автоматического обновления следует нажать кнопку .


Закрытие окна просмотра журнала

Чтобы закрыть вкладку с журналом системного процесса, следует нажать крест в заголовке вкладки.

Чтобы свернуть окно **Журнал**, следует нажать кнопку  в заголовке окна. При следующем разворачивании окна **Журнал** открытые вкладки будут восстановлены.

9.7.6.3. Сохранение информации из журнала

Информацию, отображаемую в данный момент на открытой вкладке в окне **Журнал**, можно сохранить в текстовый файл на локальной рабочей станции. Для этого следует:

1. нажать кнопку  на панели инструментов окна **Журнал**;
2. в появившемся стандартном окне Windows выбрать директорию на локальной рабочей станции и ввести имя файла (по умолчанию предлагается <имя системного процесса>.log);
3. нажать кнопку **Сохранить**.

9.7.6.4. Ротация лог-файла

Ротация лог-файлов системного процесса модуля **Нувола Топаз** – это одновременное закрытие и переименование текущего лог-файла и создание нового лог-файла.


9. Администрирование ПК Нувола через Web-интерфейс

Ротация лог-файлов выполняется автоматически в соответствии с настройками, заданными в конфигурационном файле `<service_name>.conf` для конкретного системного процесса. Этот файл хранится на сервере Нувола в директории `/usr/local/lancelot/conf/lancelot_logrotate` и содержит, в частности, следующие параметры:

- ❖ период ротации (обычно ежедневно);
- ❖ максимально допустимый размер лог-файла (обычно 100 Мб);
- ❖ количество хранимых закрытых лог-файлов (обычно 1).

Ротация выполняется по окончании временного интервала и при условии, что лог-файл достиг заданного размера (обычно 100 Мб). В результате в имя текущего лог-файла добавляется цифра 1 – `<service_name>.log.1`, и создается новый лог-файл со стандартным именем `<service_name>.log`. Если в конфигурационном файле предусмотрено хранение двух закрытых лог-файлов, то в имени первого закрытого лог-файла цифра 1 изменяется на цифру 2 – `<service_name>.log.2`. Таким образом, при каждой ротации формируется новый лог-файл и перезаписывается хранимый лог-файл.

Однако, при необходимости администратор может принудительно выполнить ротацию лог-файла. Для этого нужно выполнить следующие действия:

1. в таблице поместить курсор на нужный системный процесс;
2. нажать кнопку  на панели инструментов основного окна **Диспетчера процессов**.

10.

Редактор информационных панелей

10. Редактор информационных панелей

Web-приложение **Редактор информационных панелей** (компонента модуля **Нувола Виста**) интегрировано в Web-интерфейс.

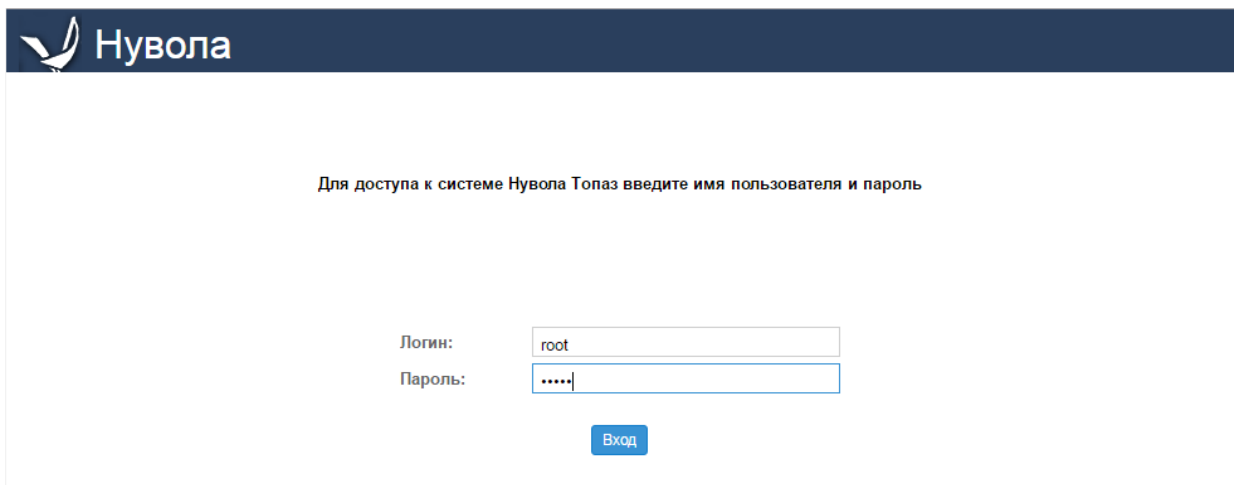
Web-приложение позволяет создавать информационные панели в формате SVG и обеспечивает их хранение на сервере Нувола.


Примечание. Приложение доступно, если пользователь входит в группу пользователей с глобальной политикой **Запуск редактора инф. панелей**.

10.1. Вход и выход из Web-приложения

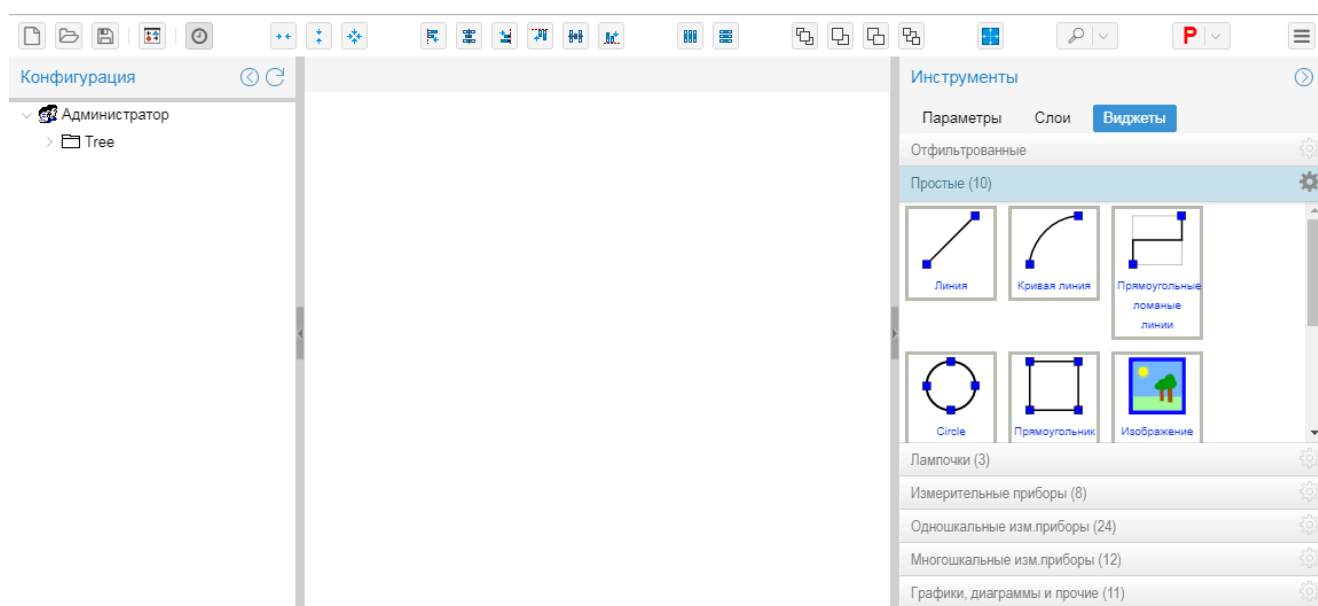
Чтобы войти в **Редактор информационных панелей**, следует:

1. войти в Web-интерфейс: в браузере ввести адрес сервера Нувола, затем на странице приглашения модуля Нувола Виста ввести имя пользователя (логин), пароль и нажать кнопку **Вход**;



2. в заголовке страницы Web-интерфейса нажать кнопку  и в меню приложений нажать иконку **Редактор информационных панелей**.

В результате в новой вкладке браузера откроется [основное окно](#) приложения.

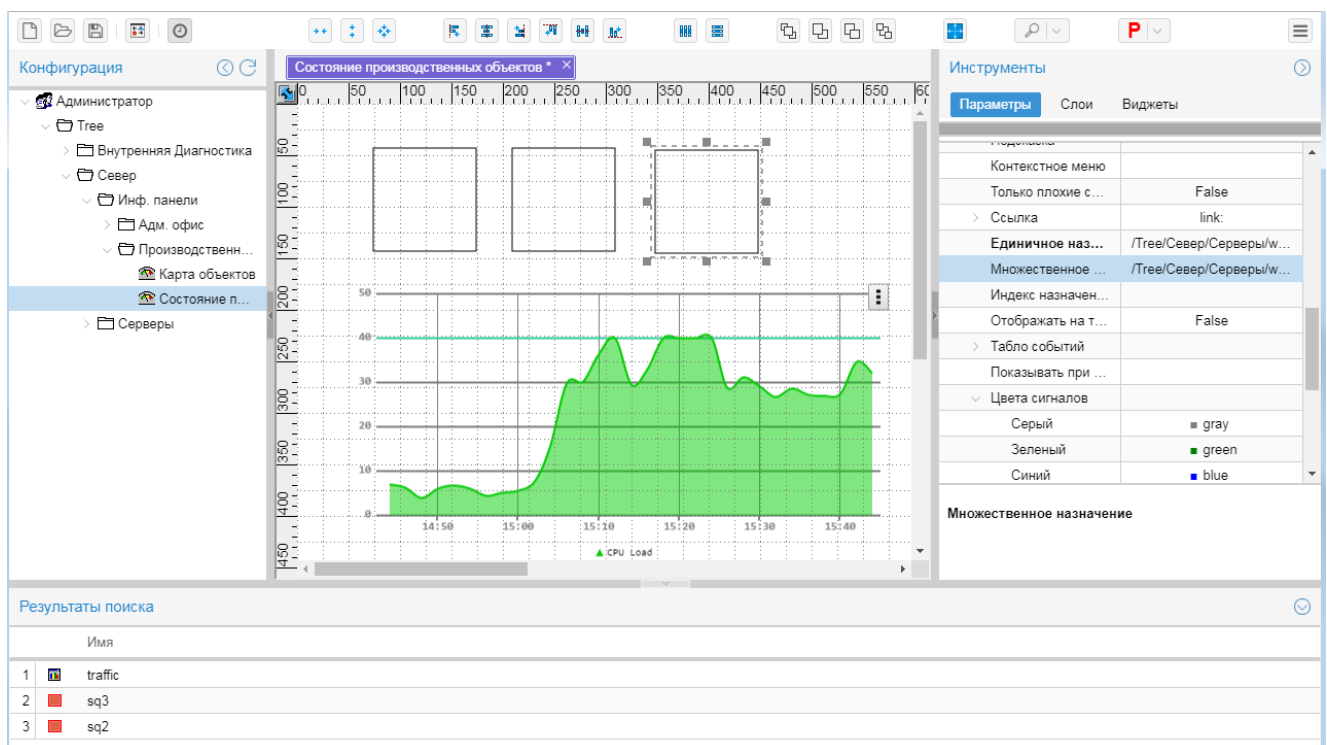


Для выхода из приложения следует закрыть вкладку браузера.

10.2. Основное окно

Основное окно **Редактора информационных панелей** содержит следующие компоненты (некоторые из них, как правило, скрыты при входе в приложение):

- ❖ панель инструментов с кнопкой главного меню;
- ❖ **окно Конфигурация** – дерево конфигурации;
- ❖ область редактирования (канва) – рабочая область для создания информационной панели;
- ❖ окно **Инструменты** – предоставляет средства для создания и настройки информационной панели:
 - вкладка **Параметры** – область свойств элементов информационной панели;
 - вкладка **Система слоёв** – предназначена для управления слоями и элементами информационной панели;
 - вкладка **Виджеты** – набор простых элементов и виджетов, доступных для размещения в области редактирования;
- ❖ панель недавно использованных виджетов – содержит последние элементы / виджеты, размещенные пользователем на канве.




10.3. Порядок разработки информационной панели

Разработка информационной панели представляет собой многошаговый процесс, в котором можно выделить следующие основные этапы:


- ❖ [Создание информационной панели](#)^[172].
- ❖ Размещение элементов на канве и настройка их параметров, см. [Обзор элементов информационной панели](#)^[174], [Инструментарий Редактора информационных панелей](#)^[196].
- ❖ [Привязка сигнала или сигналов к элементам информационной панели](#)^[184].
- ❖ Сохранение информационной панели на сервер или локально, см. [Сохранение информационной панели](#)^[172].

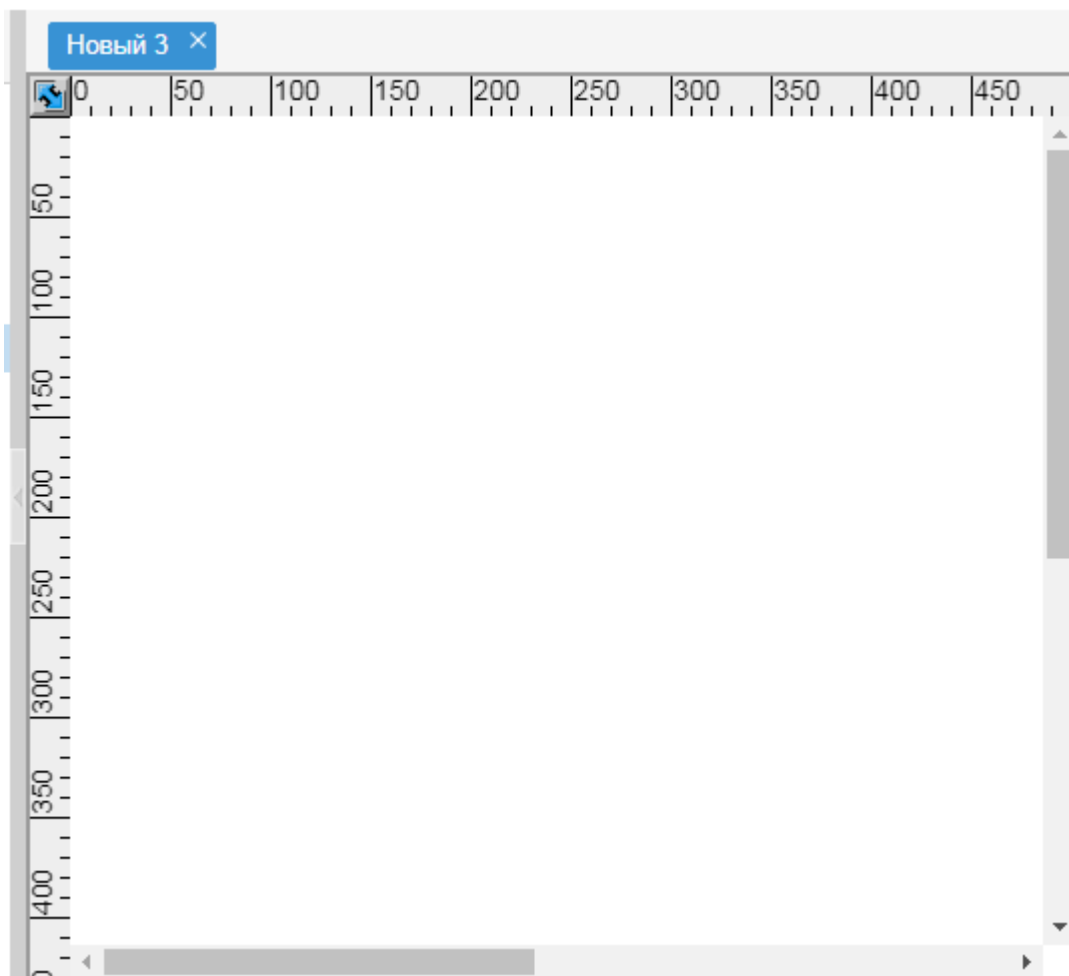
10. Редактор информационных панелей

В процессе разработки информационной панели рекомендуется:

- ❖ периодически сохранять информационную панель локально на рабочей станции или на сервере;
- ❖ контролировать создаваемую информационную панель, просматривая ее в браузере. Чтобы открыть редактируемую информационную панель в отдельной вкладке браузера, следует нажать кнопку  на панели инструментов.

10.3.1. Создание информационной панели

Чтобы создать информационную панель на чистой канве, необходимо выбрать пункт меню **Файл** → **Новый** или нажать кнопку . В результате в области редактирования будет открыта вкладка **Новый <номер>** с чистой канвой.



Далее на канве можно размещать элементы и задавать настройки. См. [Инструментарий Редактора информационных панелей](#)^[196].

10.3.2. Сохранение информационной панели


Сохранять информационную панель необходимо не только по завершении, но и в процессе разработки.

Для сохранения предусмотрены следующие способы:

- ❖ на сервере Нуволы в дереве конфигурации; следует использовать этот вариант, если информационная панель полностью подготовлена; рекомендуется каждую информационную панель сохранять в отдельную папку; поэтому предварительно следует создать ее в дереве конфигурации;

- ❖ локально на рабочей станции или удаленной машине локальной сети; следует использовать этот вариант для сохранения черновика (эскиза) или резервной копии информационной панели.

Чтобы сохранить новую информационную панель или существующую информационную панель в другое месторасположение, следует выбрать пункт меню **Файл-->Сохранить как** и в появившемся окне ввести [общие параметры](#)^[173]. После выполнения этой операции в области редактирования вкладка с сохраненной информационной панелью останется активной.

В дальнейшем в процессе разработки информационной панели для ее сохранения (без изменения месторасположения) следует использовать пункт меню **Файл --> Сохранить** (кнопку  или комбинацию клавиш **Ctrl+S**).

Можно также создать резервную копию информационной панели. Для этого следует выбрать пункт меню **Файл --> Сохранить копию...** и в аналогичном окне ввести общие параметры информационной панели. При этом в области редактирования останется активной вкладка с исходной информационной панелью, а вкладка с сохраненной копией не открывается.

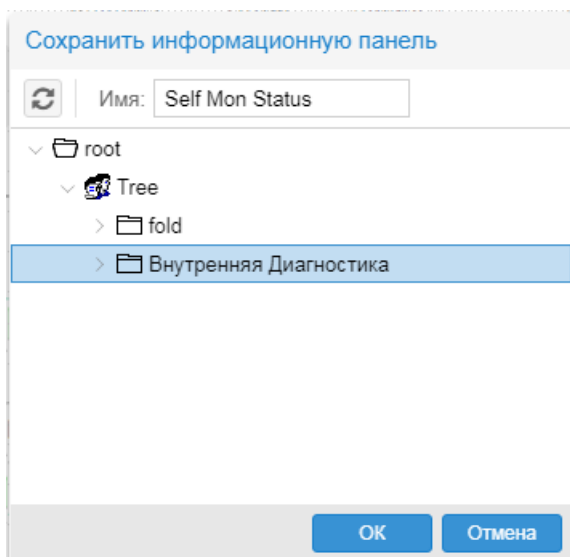
10.3.2.1. Окно ввода общих параметров инф. панели

При первом сохранении новой информационной панели или при сохранении ее копии открывается окно для ввода общих параметров информационной панели.

Порядок действий следующий:

1. задать общие параметры:
 - ❖ **Протокол обновления** – протокол передачи данных от сервера Нувола на клиентскую машину (компьютер пользователя) для обновления информационной панели, открытой в браузере. Возможные варианты:
 - WebSocket** – рекомендуемое предустановленное значение. По мере получения от ПМ **Нувола Топаз** новых данных сервер автоматически отправляет их на клиентскую машину;
 - http** – клиентская машина запрашивает данные у сервера с периодичностью, указанной в поле **Интервал обновления (мс)**. Интервал задается в миллисекундах; предустановленное значение – **5000** (5 секунд). Использование протокола http может увеличить нагрузку на сервер;
 - ❖ **Открывать ссылки в новых окнах** – способ открытия ссылок, заданных для элементов информационной панели (глобальная настройка для информационной панели в целом). Если флаг установлен, то ссылки открываются в новых окнах; если не установлен – в одном окне с информационной панелью в новых вкладках. Однако, каждый элемент может иметь персональную настройку (параметр **Открыть в** в блоке [Ссылка](#)^[180] раздела **Общие**), которая приоритетнее глобальной настройки;
3. установить способ сохранения информационной панели, нажав одну кнопку **Сервер** или **Файл**;
4. нажать кнопку **Сохранить**; в результате в зависимости от выбранного способа открывается окно для ввода имени и выбора места сохранения информационной панели;
5. ввести параметры сохранения информационной панели:
 - ❖ при сохранении на сервер вызывается окно, в котором необходимо ввести **Название информационной панели** и в дереве конфигурации выбрать папку для сохранения информационной панели. При выборе в дереве конфигурации существующей информационной панели ее наименование автоматически отображается в поле **Название информационной панели**;

10. Редактор информационных панелей



- ❖ при сохранении в файл вызывается стандартное окно Windows, в котором необходимо выбрать директорию хранения и ввести имя svg-файла;


Следует помнить, что название информационной панели чувствительно к регистру;

6. нажать кнопку **ОК**.

10.3.3. Редактирование информационной панели

Информационную панель, сохраненную в дереве конфигурации или на рабочей станции, можно загрузить на канву для дальнейшей доработки.


Открытие информационной панели на канве

Чтобы открыть информационную панель с диска на рабочей станции или доступного сетевого ресурса, следует на панели инструментов нажать кнопку  и в стандартном окне Windows выбрать svg-файл.

Чтобы открыть информационную панель, размещенную в дереве конфигурации на сервере Нуволы, следует в окне **Конфигурация** дважды кликнуть мышью на значок с информационной панелью.

Каждая информационная панель открывается в рабочей области в отдельной вкладке.

Сохранение информационной панели

Чтобы сохранить отредактированную информационную панель без изменения ее месторасположения, следует выбрать пункт меню **Файл** → **Сохранить** (либо нажать комбинацию клавиш **Ctrl+S** или кнопку ).

См. также

[Сохранение информационной панели](#) 

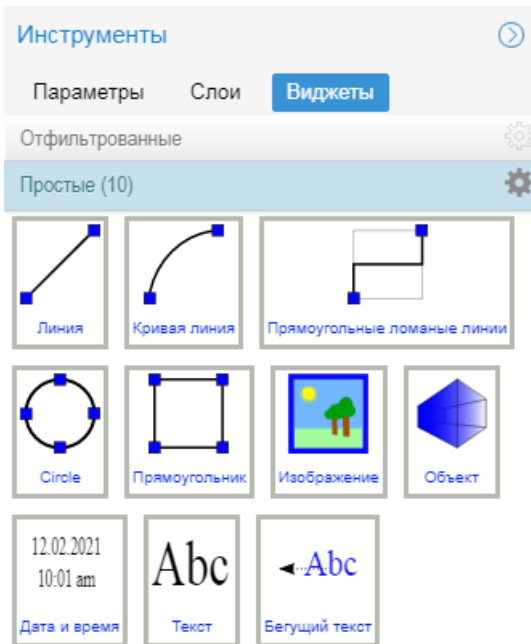
10.4. Классификация элементов информационной панели

При разработке информационной панели наиболее трудоемким этапом является размещение графических элементов в области редактирования и настройка их свойств.

Иконки доступных графических элементов расположены на вкладке **Виджеты** окна **Инструменты**. Графические элементы можно разделить на простые элементы и виджеты.

Простые элементы

К простым элементам относятся **Линия**, **Кривая линия**, **Прямоугольные ломаные линии** (см. [Простые элементы для рисования линий](#)^[294]), **Прямоугольник**^[297], **Эллипс**^[299], **Текст**^[301], **Дата и время**^[305], **Изображение**^[304]. На вкладке **Виджеты** их иконки находятся в разделе **Простые**.



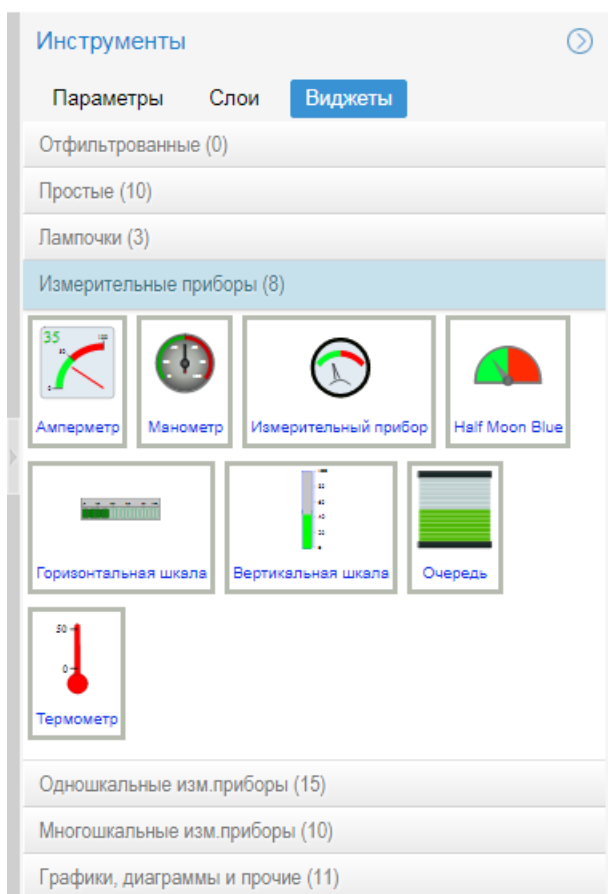
Настройка внешнего вида простых элементов, как правило, сводится к установке их размеров на канве с помощью мыши и заданию параметров на вкладке **Параметры** окна **Инструменты**.

Виджеты

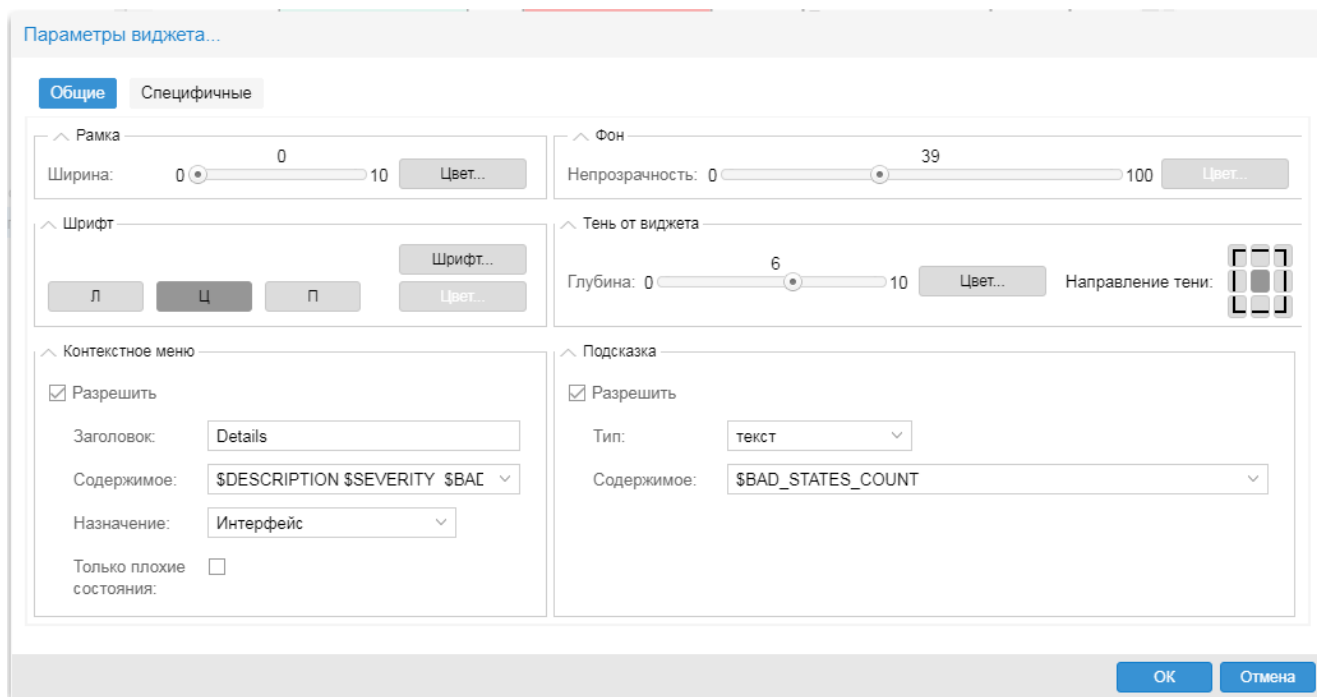
Виджет (widget) – это сложный графический элемент, отображающий состояние объекта контроля или его компонентов в специфичной форме, например в виде измерительного прибора, лампочки, диаграммы. Для каждой категории виджетов, помимо стандартных настроек, предусмотрены специфические параметры.

На вкладке **Виджеты** иконки виджетов сгруппированы по категориям.

10. Редактор информационных панелей



Специфические параметры виджета задаются в окне [Параметры виджета](#)¹⁸², которое вызывается по двойному щелчку мыши на элементе, размещенном на канве.



См. также

[Справочник Элементы информационных панелей](#)²⁹⁴

10.5. Обзор параметров элементов информационных панелей

Параметры элементов информационной панели можно условно разделить на стандартные – имеются у большинства элементов, и специфичные – характерны только для элементов определенного типа.

Параметры простых элементов задаются на вкладке **Параметры**. Для ввода специфичных параметров в зависимости от типа элемента на вкладке **Параметры** формируются дополнительные поля или разделы.



Для виджетов стандартные параметры задаются на вкладке **Параметры**, а специфичные – в специальном окне [Параметры виджета](#)^[182], которое открывается по двойному клику на виджете, размещенном на канве, или с помощью контекстного меню **Параметры виджета**. Некоторые параметры (стандартные и специфичные) доступны для ввода как на вкладке **Параметры**, так и в окне **Параметры виджета**.

В настоящем разделе руководства приведен обзор стандартных параметров на вкладке **Параметры**. Детальное описание специфических параметров для каждого элемента см. в справочнике [Элементы информационных панелей](#)^[294].

10. Редактор информационных панелей

Инструменты ⌵

Параметры Слои Виджеты

Инфо	
Номер	6006
Тип	rect
Имя	Подложка
Позиция и размер	
Слева	565
Сверху	48
Ширина	100
Высота	100
Угол	0
Отображение	
Видимость	True
Форма	
Скругление угла по x	7
Скругление угла по y	27
Ширина рамки	1
Непрозрачность, %	100
Фон	■ D2691E
Цвет линий	■ 000000
Градиент	false
Стиль линии	 сплошная
Соединение линий	 квадратное
Общие	
Подсказка	
Контекстное меню	
Только плохие сост...	false
> Ссылка	link:

Непрозрачность, %

Непрозрачность фона, в процентах

10.5.1. Раздел Инфо

Раздел **Инфо** содержит параметры, идентифицирующие элемент на конкретной информационной панели.

Номер

Идентификатор элемента. Формируется автоматически при добавлении элемента на канву и для редактирования недоступен.

Тип

Тип элемента, значение формируется автоматически и для редактирования недоступно. Возможные значения:

- connecting_line** – прямоугольная ломаная линия;
- line** – прямая линия;
- spline** – кривая линия;
- datetime** – дата и время;
- ellipse** – эллипс;
- group** – группа элементов (произвольный набор сгруппированных элементов);
- image** – изображение;
- text** – текст;
- rect** – прямоугольник;
- widget** – виджет.

Имя

Наименование элемента. По умолчанию отсутствует, но рекомендуется его ввести, т.к. в ряде случаев (например, при [назначении действия для элемента](#)^[192]) наличие имени необходимо. Имя элемента должно быть уникальным в пределах одной информационной панели, уникальность имени проверяется автоматически. Если имя не задано, то на вкладке [Слой](#)^[205] вместо имени элемента отображается его тип. При копировании и вставке именованных элементов для новых элементов имена формируются автоматически.

См. также [Переименование нескольких элементов](#)^[183].

10.5.2. Раздел Позиция и размер

Раздел **Позиция и размер** содержит параметры, определяющие положение элемента на информационной панели и его размер. Размер элемента определяется габаритами прямоугольника, в который вписывается элемент. Положение элемента определяется в системе координат канвы позицией верхней левой точки прямоугольника, в который вписывается элемент. Значения параметров данного раздела задаются в пикселах. Если пользователь с помощью мыши на канве перемещает элемент или изменяет его размеры, то значения полей раздела **Позиция и размер** корректируются автоматически.

Слева

Координата по горизонтали.

Сверху

Координата по вертикали.

Ширина

Ширина прямоугольника, в который вписывается элемент.

10. Редактор информационных панелей

Высота

Высота прямоугольника, в который вписывается элемент.

Угол

Угол наклона прямоугольника, в который вписывается элемент. См. также [Поворот элемента](#)^[203].

Для некоторых элементов в разделе **Позиция и размер** формируются дополнительные параметры.


10.5.3. Раздел Отображение

Видимость

Признак видимости элемента на информационной панели. Возможные значения:

True – элемент видим (по умолчанию). В этом случае на отображение элемента также влияют значения параметров **Показывать при уровне** и **Локализация** (раздел **Общие**), также [назначенные действия](#)^[192].

False – элемент скрывается; можно использовать, например, для временной блокировки элемента.

Изменить признак видимости элемента на информационной панели можно также на вкладке [Слои](#)^[209] с помощью переключателя  в последнем столбце.

10.5.4. Раздел Общие

Раздел **Общие** содержит параметры, применимые к большинству типов элементов.

Подсказка

Настройка всплывающей подсказки..

Контекстное меню

Настройка контекстного меню.

Только плохие состояния

Параметр используется для настройки контекстного меню для элементов с [множественным назначением](#)^[185]. Возможные значения:

False (по умолчанию) – в контекстное меню включаются записи обо всех сигналах, привязанных к элементу;

True – в контекстное меню включаются только записи о сигналах с "плохим" статусом, т.е. отличным от зеленого.

Аналогичный параметр (флаг) имеется в окне **Параметры контекстного меню**.

Ссылка

Ссылка, которая открывается по щелчку мыши на элементе при просмотре инф. панели. См. [Настройка ссылки с элемента](#)^[191].

Единичное назначение

Привязка к элементу информационной панели одного сигнала. См. [Привязка одного сигнала к элементу информационной панели](#)^[184], [Привязка однотипных сигналов к элементу](#)^[188].

Множественное назначение

Привязка к элементу информационной панели нескольких сигналов. Если не задан **Индекс назначения**, элемент отображает состояние сигнала с наихудшим статусом. См. также [Привязка нескольких сигналов к элементу информационной панели](#)^[185].

Индекс назначения

Для элемента с множественным назначением можно указать индекс конкретного сигнала, статус которого должен отображать элемент. Индекс сигнала соответствует его порядковому номеру (начиная с 0) в списке привязанных к элементу сигналов. См. также [Использование Индекса назначения](#)^[187].

Отображать на табло

Необходимость отображать сообщения, связанные с данным элементом информационной панели, на виджете [Табло событий](#)^[338]. Если параметр установлен (**True**), то настройки отображения сообщений следует задать в блоке параметров **Табло событий**. По умолчанию параметр не установлен (**False**).

Табло событий

Настройки отображения сообщений на виджете [Табло событий](#)^[338]. Их необходимо задать, если параметр **Отображать на табло=True**. Поле **Табло событий** заполняется автоматически по мере ввода значений его составляющих. См. [Настройка табло событий](#)^[340].

Показывать при уровне

Перечень статусов сигнала, при которых элемент отображается. В раскрывающемся списке следует отметить нужные статусы. Если ни один флаг не отмечен (по умолчанию), элемент отображается при любых статусах сигнала.

Цвета сигналов

Соответствие между цветом окраски элемента и статусом сигнала, назначенного элементу. Раздел состоит из полей с цветовыми кодами для каждого статуса. По умолчанию используются стандартные цветовые коды сигналов.

Черный цвет не является статусом сигнала. Элемент окрашивается в черный цвет, если его назначение некорректно, например, если объект контроля был удален или переименован.

Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

▼ Цвета сигналов	
Серый	■ gray
Зеленый	■ green
Синий	■ blue
Желтый	■ yellow
Красный	■ red
Темно-красный	■ crimson
Черный	■ black
Белый	■ white

Разрешить звуки

Возможность воспроизведения звукового сигнала в зависимости от цвета элемента.

True – при окрашивании элемента в определенный цвет воспроизводится звуковой файл в соответствии с настройкам, заданными в поле **Звук при состоянии**;

False – звуковое сопровождение отключено.

Звук при состоянии

10. Редактор информационных панелей

Настройка звуковых эффектов в зависимости от статусов сигнала. См. [Настройка звукового сигнала для элемента](#)^[195].

Действия

Действия, которые будут выполняться при определенных манипуляциях пользователя во время просмотра информационной панели. См. [Настройка действий для элемента](#)^[192].

Анимация

Эффекты анимации, используемые для оживления информационной панели.

Локализация

Язык Web-интерфейса (**Русский** или **Английский**), при котором элемент отображается на информационной панели. По умолчанию элемент отображается в Web-интерфейсе вне зависимости от выбранной локализации. Параметр **Локализация** можно задать также для группы элементов.

10.5.5. Окно Параметры виджета

Окно **Параметры виджета** вызывается по двойному щелчку мыши на элементе, размещенном на канве.

На вкладке **Общие** находятся параметры, которые можно задать для каждого виджета.

The screenshot shows the 'Widget Parameters' dialog box with the 'General' tab selected. The dialog is divided into several sections:

- Рамка (Border):** Width slider set to 0, with a 'Color...' button.
- Шрифт (Font):** Buttons for font face (l, Ц, П), font color ('Цвет...'), and font style ('Шрифт...').
- Фон (Background):** Opacity slider set to 39, with a 'Color...' button.
- Тень от виджета (Widget Shadow):** Depth slider set to 6, 'Color...' button, and a 'Direction of shadow' icon.
- Контекстное меню (Context Menu):** 'Allow' checkbox checked, title 'Details', content '\$DESCRIPTION \$SEVERITY \$BAC', and assignment 'Интерфейс'. There is also a 'Only bad states' checkbox.
- Подсказка (Tooltip):** 'Allow' checkbox checked, type 'текст', and content '\$BAD_STATES_COUNT'.

Buttons for 'OK' and 'Отмена' (Cancel) are at the bottom right.


На вкладке **Специфичные** находятся параметры, характерные для конкретного виджета.

Для некоторых категорий виджетов указывается параметр **Тип**, определяющий внешний вид виджета.

Параметры виджета...

Общие Специфичные

Queue



Поведение

Линейный Триггер

3х-мерный эффект

Назначение

/Tree/Operator1/ws-057/CPU 1;windows-cpu;load

Список пороговых значений

Цвет...

Значение	Цвет

Определение шкалы

Цена осн. деления: 20

Цена всп. деления: 5

Текст: процент

Единицы: нет

Множитель: 1

Шкала

Слева Справа Сверху По низу Нет

Координирование

Горизонтальная Вертикальная

OK Отмена

10.6. Настройка параметров элементов информационной панели

В настоящем разделе описаны особенности настройки параметров, характерных для большинства элементов.

10.6.1. Изменение параметров элементов

Чтобы изменить свойства элемента, размещенного на канве, следует сначала выделить его, а затем задать нужные значения в окне **Параметры**.

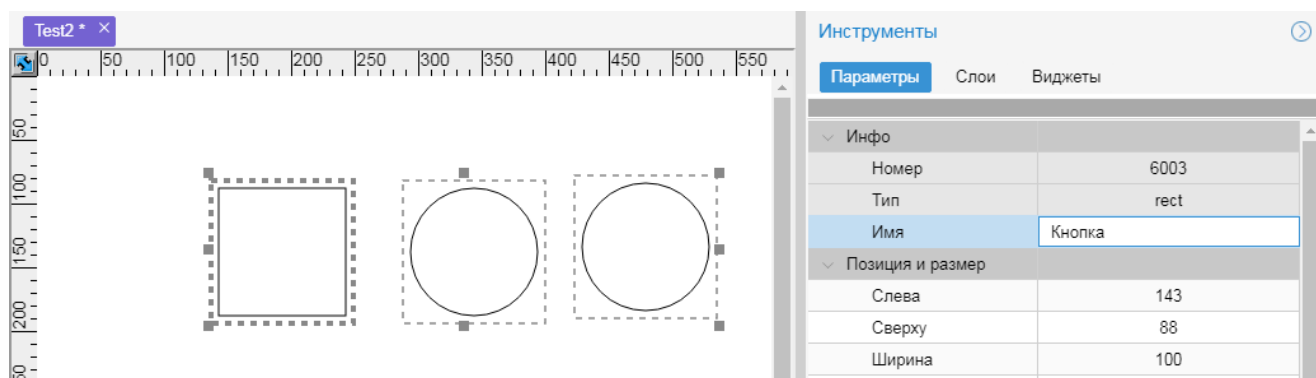
Изменять значения в окне **Параметры** можно сразу для нескольких выделенных элементов.

Переименование нескольких элементов

Переименование нескольких элементов имеет особенности. Если выделить на канве несколько элементов (однотипных или разнотипных) и ввести значение в поле **Имя**, то основной элемент получит имя в том виде, в каком оно было введено, а к именам остальных элементов через дефис будут добавлены индексы, начиная с нуля.

Пример. На канве выделено 3 элемента, прямоугольник – основной. В поле **Имя** введено значение **Кнопка**. Тогда прямоугольник будет именоваться как **Кнопка**, а остальные элементы – **Кнопка-0** и **Кнопка-1**.

10. Редактор информационных панелей



10.6.2. Привязка сигналов к элементам

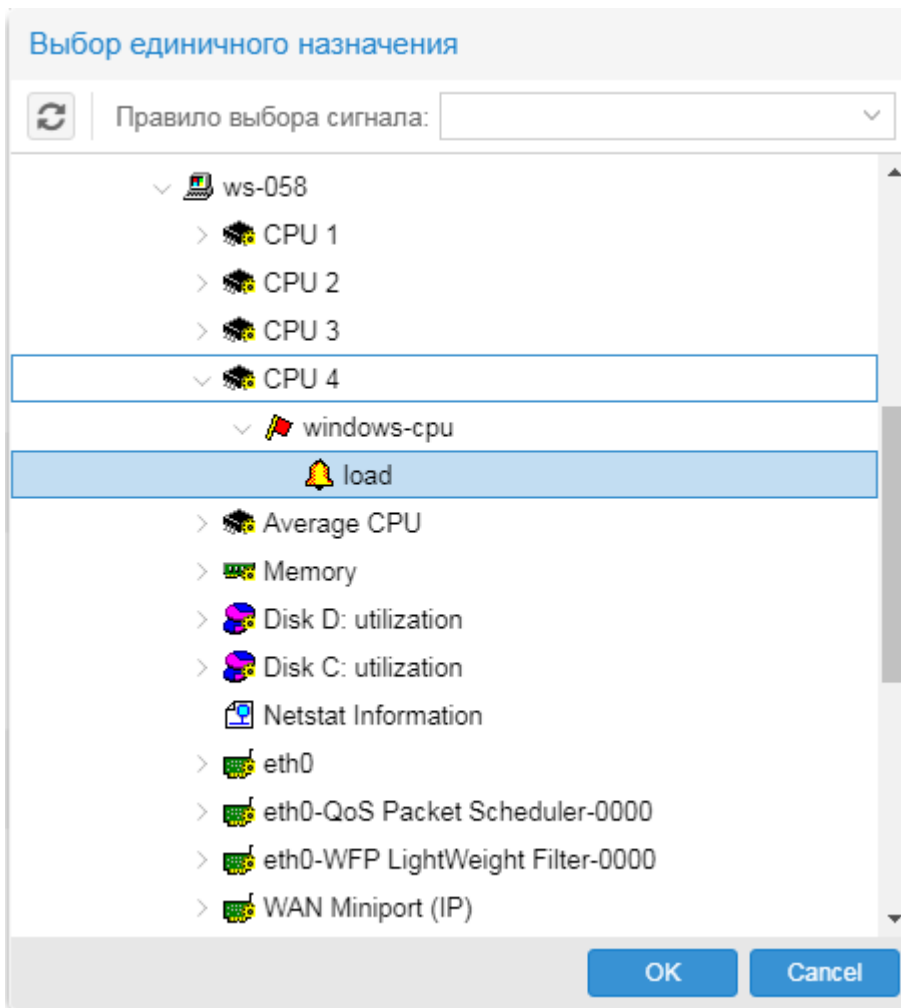
Назначение (параметр элемента информационной панели) – привязка к сигналам, влияющим на поведение элемента, например: изменение цвета заливки элемента в зависимости от статусов сигналов, показания шкалы (на виджетах) и т.п. **Единичное назначение** – привязка к одному сигналу. **Множественное назначение** – совокупность привязок к нескольким сигналам.

К каждому элементу, размещённому на канве, можно привязать один или несколько сигналов. Исключение – виджеты [Рамочный контейнер](#)³³⁶ и [Динамический график](#)³²⁹, к ним можно привязать только по одному сигналу. Если к элементу привязаны несколько сигналов, то по умолчанию (т.е. если не задан параметр **Индекс назначения**) в Web-браузере элемент отображает наихудший из статусов привязанных сигналов.

10.6.2.1. Привязка одного сигнала к элементу

Чтобы привязать сигнал к элементу информационной панели, следует:

1. на вкладке **Параметры** открыть раздел **Общие** и в поле для ввода значения параметра **Единичное назначение** нажать кнопку
2. в открывшемся дереве конфигурации выбрать сигнал и нажать кнопку **ОК**.



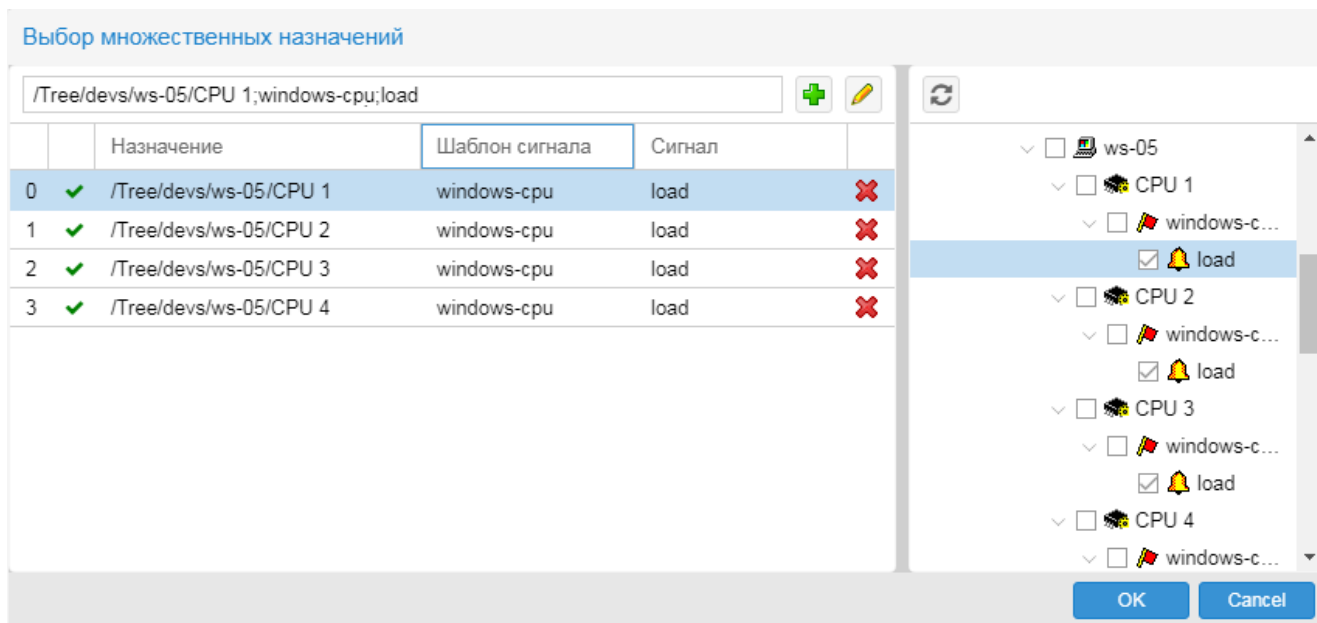
В результате путь на выбранный сигнал будет указан в поле **Единичное назначение**.

Примечание. В поле **Множественное назначение** будет автоматически введена та же текстовая строка, что и в поле **Единичное назначение**.

10.6.2.2. Привязка нескольких сигналов к элементу

Привязка нескольких сигналов к элементу информационной панели осуществляется в окне **Выбор множественных назначений**.


10. Редактор информационных панелей



Окно вызывается следующими способами:

- ❖ для простых элементов и виджетов – по нажатию кнопки в поле **Множественное назначение** (раздел **Общие** на вкладке **Параметры**);
- ❖ для виджетов – по нажатию кнопки в поле **Назначение** в окне **Параметры виджета**. Для некоторых виджетов данное поле называется **Источник получения данных для входящей строки**.

Структура окна Выбор множественных назначений

В окне **Выбор множественных назначений** справа расположено дерево конфигурации. Флаги в его узлах предназначены для выбора сигналов и настройки привязок. Кнопка  позволяет обновить структуру в соответствии с текущими данными на сервере.

Слева находится таблица с параметрами настроенных привязок, а над ней – панель для редактирования записей таблицы. Каждая строка содержит следующие параметры:

Первый столбец

Индекс назначения. Формируется автоматически. См. также [Использование Индекса назначения](#)¹⁸⁷.

Второй столбец

Индикатор корректности записи.

✓ – запись корректна;

⚠ – запись некорректна.

Назначение

Путь в дереве конфигурации до целевого объекта.


Шаблон сигнала

Имя набора сигналов, в который входит сигнал.

Сигнал

Имя сигнала, привязанного


Последний столбец

Кнопка  для удаления привязки.

Если поместить курсор на строку таблицы, то в поле над таблицей выводится текст ссылки.

Привязка нескольких сигналов к элементу

Чтобы привязать несколько сигналов к элементу, следует:

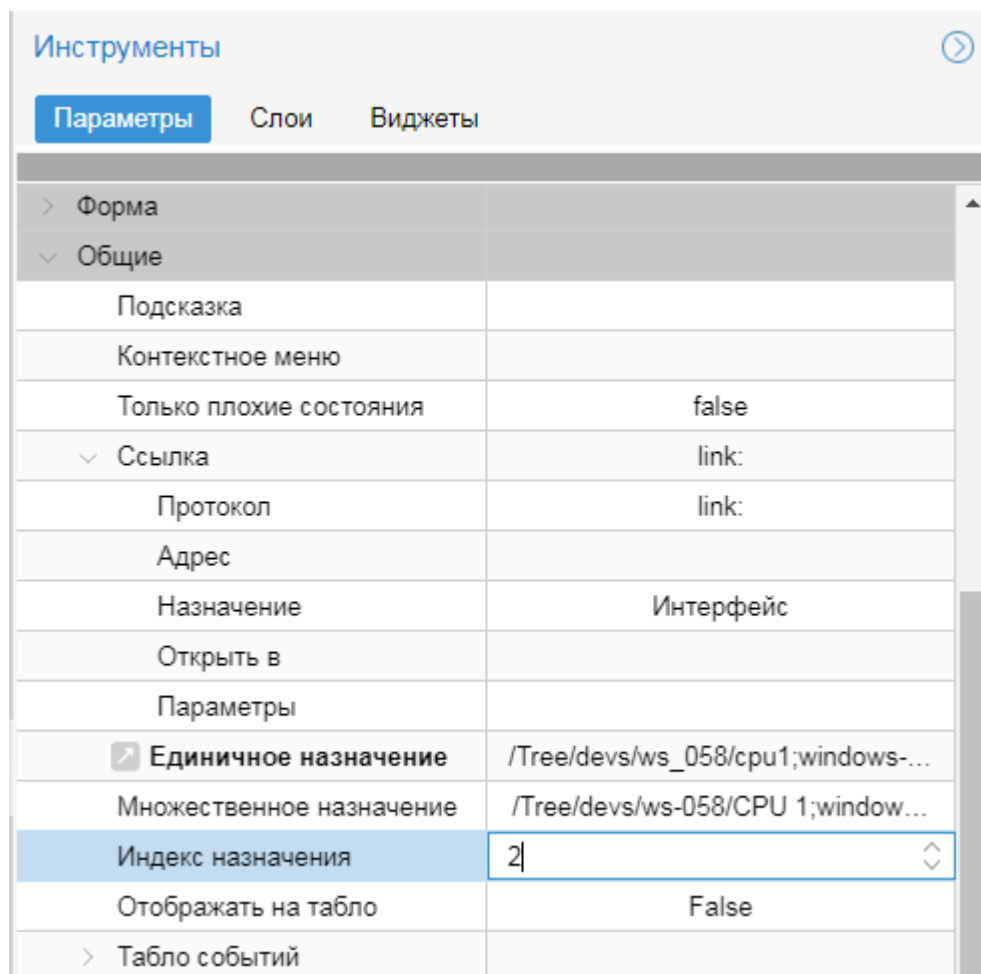
1. на вкладке **Параметры** открыть раздел **Общие** и в поле для ввода значения параметра **Множественное назначение** нажать кнопку . Откроется окно **Выбор множественных назначений**;
2. в дереве конфигурации отметить флаги рядом с нужными сигналами. По мере установки флагов, в таблице в левой части окна будут автоматически добавляться строки;
3. после того, как список сигналов сформирован, для сохранения настроек и возврата в основное окно нажать кнопку **ОК**.

Использование Индекса назначения

По умолчанию элемент с множественным назначением отображает в Web-интерфейсе наихудший статус среди привязанных сигналов. Однако можно в явном виде указать сигнал, статус которого должен отображать элемент. Для этого на вкладке **Параметры** в поле **Индекс назначения** (раздел **Общие**) необходимо указать индекс соответствующего сигнала.

Индексы привязанных сигналов отображаются в окне **Выбор множественных назначений** (см. выше) в первом столбце таблицы: первая строка имеет индекс **0**, вторая строка – индекс **1**, третья строка – индекс **2**, и т.д.

Пример. К элементу информационной панели привязаны три сигнала. Требуется, чтобы элемент отображал статус сигнала, расположенного на третьей строке в окне **Выбор множественных назначений**. В этом случае в поле **Индекс назначения** следует ввести **2**.



Индексы также используются при настройке для элемента [действия](#)^[192] с функцией [Change Target Index of Specified Element](#)^[343].

10.6.2.3. Привязка однотипных сигналов к элементу

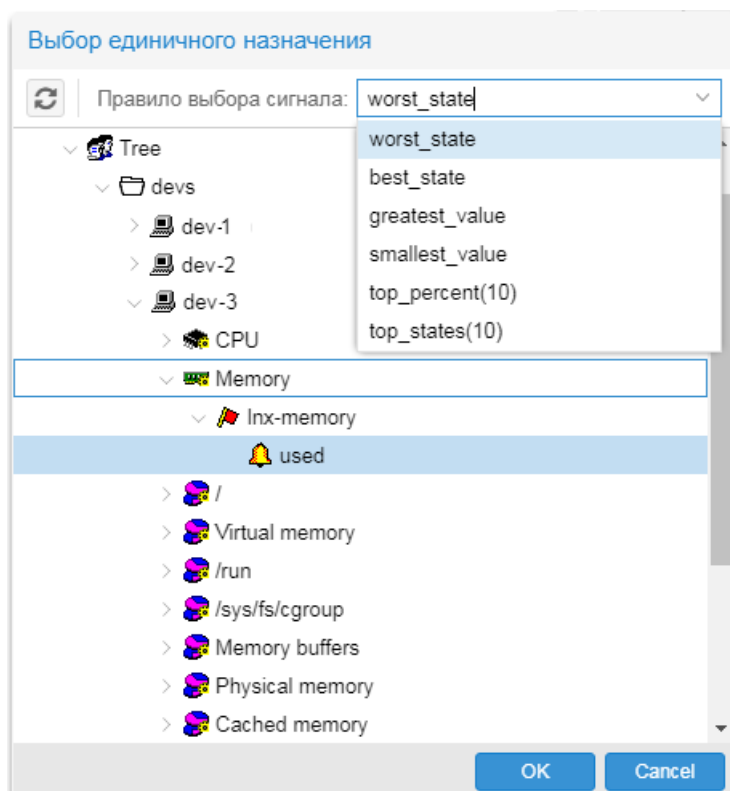
К элементу информационной панели можно привязать несколько однотипных сигналов в рамках ветви дерева конфигурации и задать правило выбора сигнала для отображения на элементе. В частности, элемент может показывать сигнал с наихудшим/наилучшим статусом или наибольшим/наименьшим значением среди всех привязанных сигналов.

Для привязки нескольких однотипных сигналов следует:

1. в области редактирования выделить элемент информационной панели;
2. нажать кнопку в поле **Единичное назначение**;
3. в открывшемся дереве конфигурации выбрать сигнал нужного типа в нужной ветви дерева конфигурации;
4. в поле **Правило выбора сигнала** выбрать функцию отбора сигнала для отображения на элементе информационной панели. Доступны следующие функции:
 - worst_state** – сигнал с наихудшим статусом;
 - best_state** – сигнал с наилучшим статусом;
 - greatest_value** – сигнал наибольшим значением;
 - smallest_value** – сигнал с наименьшим значением;
 - top_percent(10)** – сигналы с наибольшими значениями в процентах;

top_states(10) – сигналы с наихудшими статусами (используется по умолчанию).

В перечисленных функциях используется установленная в ПК Нувола [классификация статусов](#)^[286] по степени критичности: бордовый – наихудший, зеленый – наилучший. Последние две функции списка (**top_percent** и **top_states**) имеют один аргумент – количество отбираемых сигналов, по умолчанию – **10**. Отсортированный список из отобранных сигналов будет отображаться по нажатию правой кнопки на элементе при просмотре инф. панели. В случае функции **top_percent** вверху списка выводится сигнал с наибольшим значением в процентах, в случае функции **top_states** – сигнал с наихудшим статусом (т.е. с бордовым).



5. нажать кнопку **ОК**;
6. на вкладке **Параметры** в поле **Единичное назначение** (раздел **Общие**) из пути для выбранного сигнала удалить названия узлов, конкретизирующих целевой объект (таким образом будет задан обобщенный путь к сигналам ветви дерева конфигурации);

Если на шаге 4 была выбрана функция **top_percent(10)** или **top_states(10)**, то можно изменить ее аргумент, т.е. количество сигналов отображаемых в списке. Для этого достаточно в **ПОЛЕ Единичное назначение** отредактировать число в скобках после названия функции.


После внесения изменений в строку в поле **Единичное назначение** нажать клавишу **Enter**;

Пример. После выбора конкретного сигнала в дереве конфигурации (на шаге 4) в поле **Единичное назначение** формируется следующая запись:

```
/Tree/Self Monitoring/localhost/CPU;lnx-cpu(top_states(10));load
```

Для того чтобы элемент отображал 7 сигналов с наихудшими статусами во всем дереве, строку необходимо отредактировать следующим образом:

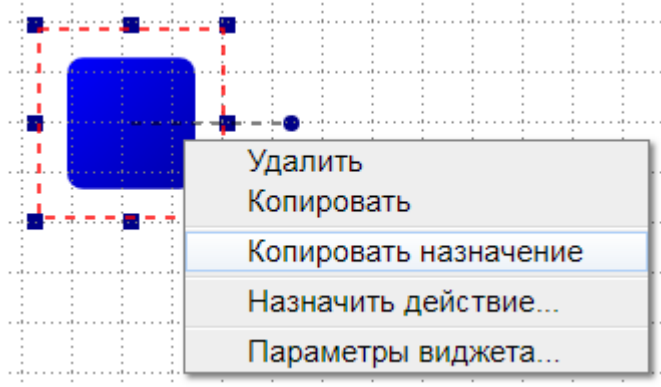
```
/Tree;lnx-cpu(top_states(7));load
```

7. сохранить информационную панель с помощью кнопки  на панели инструментов.

10.6.2.4. Копирование назначений элементов

Для удобства и ускорения процесса разработки информационной панели можно копировать привязки к сигналам от одних элементов к другим. Для выполнения функции копирования предназначен пункт контекстного меню:

- **Копировать назначение / назначения** – скопировать в буфер обмена привязки к сигналам, установленные для выбранного элемента / нескольких элементов.
- **Установить назначение / назначения** – установить для выбранного элемента / нескольких элементов привязки к сигналам из буфера обмена.



Общий порядок использования данной функции заключается в следующем. На канве пользователь выбирает один или несколько элементов, от которых нужно скопировать назначения. В результате копирования в буфер обмена попадает строка, содержащая путь на сигнал, назначенный выбранному элементу, или несколько строк (если выделенный на канве элемент имеет множественное назначение, или если пользователь выделил несколько элементов). Чтобы скопированные пути назначить другому элементу (или сразу нескольким элементам), следует выделить на канве этот элемент (или несколько элементов) и вставить содержимое буфера обмена. Предварительно при необходимости содержимое буфера обмена можно изменить в текстовом редакторе.

Копирование множественного назначения с использованием стороннего текстового редактора

Для копирования назначений, содержащих однотипные имена удобно пользоваться сторонним текстовым редактором. Чтобы скопировать множественное назначение от одного или нескольких элементов к другому элементу, следует:

1. на канве выделить один или несколько элементов, от которых нужно скопировать назначения. В контекстном меню выбрать пункт **Копировать назначения**. В результате в буфер обмена попадут строки, содержащие пути на сигналы, назначенные выбранным элементам;
2. вставить содержимое буфера обмена в текстовый редактор (например, в "Блокнот"). Каждое назначение будет представлено в виде отдельной строки:

```
/Tree/devs/ws-03/CPU 1;windows-cpu;load,  
/Tree/devs/ws-03/CPU 2;windows-cpu;load
```

3. отредактировать текст в соответствии с решаемой задачей. Например, можно изменить пути на целевые объекты, имена сигналов и т.д. Допускается добавлять и удалять строки.

ВНИМАНИЕ! Между строками необходимо удалить знаки абзаца (символы перевода строки), оставив в качестве разделителя только запятую (без пробела).

Пример отредактированной строки:

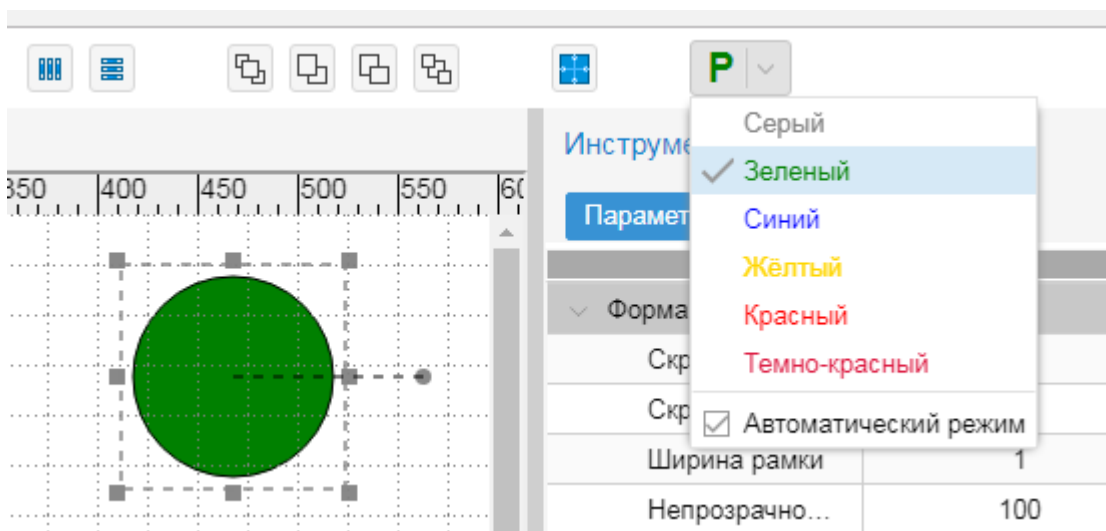
```
/Tree/devs/ws-04/CPU 2;windows-cpu;load,/Tree/devs/ws-04/CPU 3;windows-  
cpu;load,/Tree/devs/ws-04/CPU 4;windows-cpu;load
```

4. скопировать подготовленную строку в буфер обмена (с помощью клавиш **Ctrl+C**);
5. на канве выделить элемент, к которому нужно привязать сигналы;
6. на вкладке **Параметры** в поле **Множественное назначение** вставить содержимое буфера обмена;
7. чтобы убедиться в корректности выполненной привязки, нажать кнопку в поле **Множественное назначение** и просмотреть список назначений в окне **Выбор множественных назначений**^[185]. Если строки помечены символом **✓**, то назначения корректны. Если список пуст или строки помечены символом **▲**, то привязка не выполнена.

10.6.2.5. Проверка отображения статуса сигнала на элементе

Некоторые элементы информационной панели, размещенные на канве, можно протестировать на предмет того, как они будут выглядеть при изменении статуса сигнала. Такая проверка возможна для простых графических элементов: прямая линия, кривая линия, прямоугольная ломаная линия, прямоугольник, эллипс. Наличие привязки к сигналу не требуется.

Проверка выполняется с помощью кнопки **P** (probe) на панели инструментов. Кнопка активна, если в области редактирования выделен один из перечисленных элементов. Цвет буквы **P** на кнопке соответствует цвету, в который окрашивается элемент при тестировании. По нажатию стрелки на кнопке открывается меню, с помощью которого можно изменить цвет, а также включить и отключить автоматический режим.



Для выполнения проверки следует в области редактирования выделить нужный элемент и нажать кнопку

P на панели инструментов или выбрать пункт меню **Правка** -> **Проба** (или нажать комбинацию клавиш **Alt+L**). В результате на несколько секунд элемент окрасится в заданный цвет. Если включен автоматический режим, то изменение цвета происходит сразу при выборе элемента на канве или на вкладке **Слои** (без дополнительных манипуляций пользователя).

10.6.3. Настройка ссылки с элемента


При просмотре информационной панели по клику мыши на элементе пользователь осуществляет переход по ссылке, настроенной с помощью параметра **Ссылка** (вкладка **Параметры**, раздел **Общие**). Ссылка может открывать страницу Web-интерфейса ПК Нувола или внешний сайт.

Значение параметра **Ссылка** формируется автоматически из значений **составляющих его полей**^[192]. Пример формата: `link:/url;parameter`, где `parameter` – дополнительный параметр (поле **Параметры**), например, `parameter=if_avail`.

10. Редактор информационных панелей

Как правило, для настройки ссылки администратору следует заполнить составляющие поля параметра **Ссылка**. Однако в некоторых случаях целесообразно ввести строку непосредственно в поле **Ссылка**.

10.6.3.1. Составляющие параметра Ссылка

Непосредственно поле **Ссылка** доступно для редактирования и позволяет вводить строку ссылки вручную. Также можно выбрать назначение ссылки из дерева конфигурации, нажав кнопку .

Раздел параметров **Ссылка** включает в себя следующие параметры:

Протокол

Список значений содержит допустимые виды ссылок:

[no link] – ссылка отсутствует (используется по умолчанию для элементов, не привязанных к сигналам);

link – ссылка на какой-либо узел дерева конфигурации ПК Нувола (используется по умолчанию для элементов, привязанных к сигналам);

http:// – ссылка на произвольный сайт или Web-страницу;

ftp:// – ссылка на какой-либо объект по протоколу ftp;

mailto: – вызов приложения Microsoft Outlook для отправки письма по заданному адресу.

Адрес

url или адрес электронной почты.

Назначение

Параметр указывает web-страницу, которая открывается по ссылке. Возможны два варианта:

Интерфейс – открывается страница целевого объекта;

Устройство – открывается страница объекта контроля (устройства). При выборе данного варианта в поле **Параметры** автоматически проставляется значение **&elink=LINK2DEV**.

Открыть в

Способ отображения содержимого ссылки:

новом окне – в новом окне браузера;

новой закладке – в новой вкладке текущего окна браузера;

в текущем окне браузера – в текущем окне браузера (вместо страницы, с которой ссылка вызвана).

Параметры

Параметры ссылки, вводятся при необходимости. См. также [Предопределенные переменные \(теги\) для текстовых строк](#)^[34]. В поле **Ссылка** параметры начинаются после точки с запятой (символ ; формируется автоматически).

10.6.4. Настройка действий для элемента

Для элемента информационной панели можно настроить действия, которые будут выполняться при определенных манипуляциях пользователя во время просмотра информационной панели. Например, по щелчку мыши на одном элементе может быть изменен цвет другого элемента.

Выполняемое действие описывается функцией. Одним из аргументов функции является имя элемента, к которому будет применено действие. Поэтому перед настройкой следует убедиться, что участвующие в действии элементы поименованы, т.е. для них заполнено поле **Имя**^[179] (раздел **Инфо** на вкладке **Параметры**).

Для вызова окна **Назначение действия** следует:

- ❖ нажать правую кнопку мыши на элементе, размещенном на канве, и в контекстном меню выбрать пункт **Назначить действие...**
либо
- ❖ на вкладке **Параметры** в разделе **Общие** в поле **Действия** нажать кнопку .

Настроенные действия выводятся в таблице, состоящей из четырёх столбцов.

Событие

Манипуляция пользователя или событие, при котором выполняется действие. Возможные варианты:

OnClick – щелчок левой кнопки мыши на элементе;

OnMouseOver – размещение курсора мыши на элементе;

OnMouseOut – смещение курсора мыши с элемента;

OnChangeState – переход сигнала, привязанного к элементу, в одно из заданных состояний (в поле **Состояние**).

Функция

Выполняемое действие. Функцию следует выбрать из списка. См. справочник [Функции для настройки действий для элементов информационной панели](#)^[342].

Параметры

Перечень параметров, принимаемых функцией. Значения параметров вводятся через запятую без пробелов.

Для параметров, значениями которых являются имена элементов, допускаются следующие способы ввода:

- ❖ для указания только одного элемента, достаточно ввести его имя;
- ❖ для указания нескольких элементов можно перечислить их имена через точку с запятой, например: **a1;a2;a3**, либо задать регулярное выражение.

Значения некоторых параметров вводятся во вспомогательном окне (например, выбираются из дерева конфигурации). Если такая возможность предусмотрена, то для вызова вспомогательного окна следует в поле **Параметры** нажать кнопку .

Перечень параметров выбранной функции можно просмотреть в подсказке (см. ниже).

Состояния


Состояния, при переходе в которые выполняется действие. Поле активно, если в поле **Событие** указано **OnChangeState**. Для выбора состояния следует нажать кнопку и в окне отметить соответствующие флаги.

В примере на рисунке ниже настроено следующее действие: по щелчку мыши на элементе, для которого назначено действие, переключается видимость элемента с именем **square**. Необходимо помнить, что видимость сгруппированного элемента зависит от видимости группы.

10. Редактор информационных панелей

	Событие	Функция	Параметры	Состояния	
+	OnClick	Show hidden element, or hide shown	square		✖

OK Отмена

По нажатию кнопки , расположенной в первом столбце, раскрываются две строки с подсказкой:

- ❖ в первой строке отображается формат вызова выбранной функции;
- ❖ во второй строке – вызов функции с указанными значениями параметров; данная строка заполняется автоматически по мере ввода параметров.


	Событие	Функция	Параметры	Состояния	
+	OnClick	Show hidden element, or hide shown	Frame		✖
-	OnMouseOver	Set Fill Color	Button1, #262f2d		✖
-	OnMouseOut	Set Fill Color	element_name,color		✖

function setcolor(element_name,color) {...}
function setcolor(Button1, #262f2d) {...}


function setcolor(element_name,color) {...}
function setcolor(element_name,color) {...}

OK Отмена

Чтобы назначить действие элементу информационной панели, следует:

1. на канве поместить курсор на элемент, с которым пользователь будет выполнять манипуляцию;
2. в контекстном меню выбрать пункт **Назначить действие...**;
3. в окне **Назначение действия** нажать кнопку  в правом верхнем углу окна;
4. в появившейся строке заполнить поля **Событие**, **Функция**, **Параметры** и **Состояния** (при необходимости);
5. если для элемента необходимо настроить несколько действий, то пп. 3 и 4 повторить для каждого действия;
6. для сохранения введенной информации и возврата на канву нажать кнопку **ОК**.

Чтобы удалить действие, следует:

1. в таблице поместить курсор на строку, подлежащую удалению;
2. нажать кнопку  в последнем столбце строки;
3. для подтверждения действия и возврата в канву нажать кнопку **ОК**.

При [сохранении](#)^[172] информационной панели, содержащей элементы с настроенными действиями, имеет смысл устанавливать флаг [Сохранять текущее состояние](#)^[173].

10.6.5. Настройка звукового сигнала для элемента

Для элемента информационной панели можно настроить звуковой сигнал, который будет воспроизводиться при окрашивании элемента в определенный цвет.

Для настройки необходимо на вкладке **Параметры** в разделе **Общие** заполнить следующие поля:

Разрешить звуки




Возможность воспроизведения звукового сигнала в зависимости от цвета элемента.

True – при окрашивании элемента в определенный цвет воспроизводится звуковой файл в соответствии с настройкам, заданными в поле **Звук при состоянии**;

False – звуковое сопровождение отключено.

Звук при состоянии

Настройка звуковых эффектов в зависимости от статусов сигнала. После включения флага **Разрешить звуки** в полях активизируются предустановленные настройки для статусов – звуковые файлы с указанием продолжительности воспроизведения в секундах. Пользователь может их изменить. Головное поле **Звук при состоянии** заполняется автоматически на основании настроек, заданных для статусов.

Разрешить звуки	True
Звук при состоянии	blue.mp3;1 sec,green.mp3;1 sec,yellow.mp...
Синий	blue.mp3;1 sec
Зеленый	green.mp3;1 sec
Желтый	mp3;1 sec   
Красный	red.mp3;1 sec
Темно-красный	crimson.mp3;1 sec
Серый	gray.mp3;1 sec
Черный	black.mp3;1 sec
Белый	white.mp3;1 sec

В полях с названиями статусов размещены функциональные кнопки:



– вызов окна **Звук по состоянию** для настройки звукового эффекта;




– сохранение файла, указанного в поле, на локальном диске или доступном сетевом ресурсе;

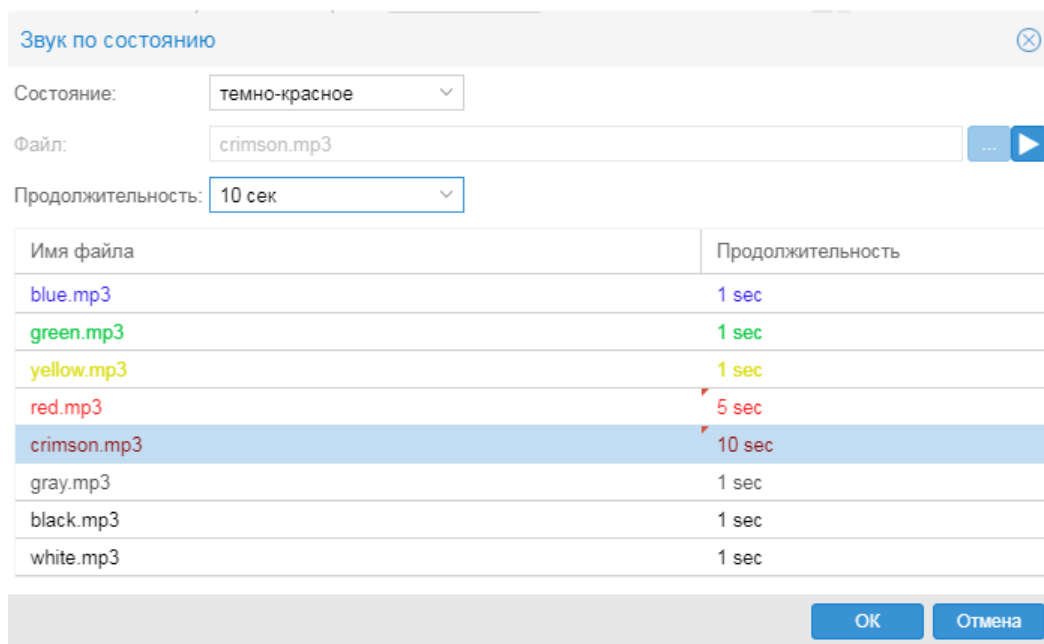


– тестирование звукового эффекта (воспроизведение файла в течение заданного интервала).

Чтобы изменить настройки звуковых эффектов, следует:

1. нажать кнопку  в любом из полей со статусом;
2. в открывшемся окне **Звук по состоянию** для нужных статусов сигнала выбрать звуковой файл, хранимый на локальном диске рабочей станции или на доступном сетевом ресурсе, и задать продолжительность воспроизведения;
3. для сохранения настроек и возврата в основное окно **Редактора инф. панелей** нажать кнопку **ОК**.

10. Редактор информационных панелей




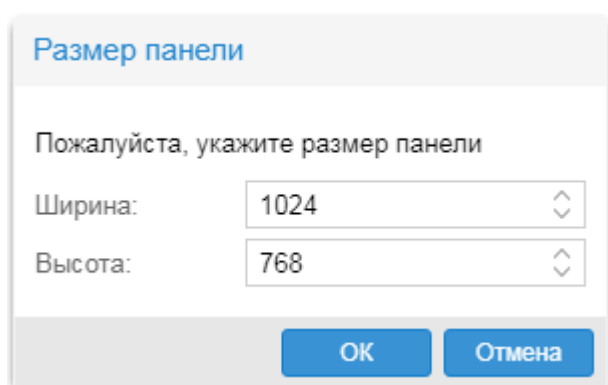
10.7. Инструментарий Редактора информационных панелей


В настоящем разделе изложены методы и приемы разработки информационной панели, описаны инструменты и возможности Web-приложения.

10.7.1. Изменение размеров информационной панели

При создании новой информационной панели размеры канвы устанавливаются по размерам последней сохраненной информационной панели. Для изменения размеров канвы можно использовать два способа.

- ❖ Чтобы задать размеры канвы вручную, следует нажать кнопку , расположенную в левом верхнем углу канвы, в появившемся окне ввести ширину и высоту в пикселах и нажать **ОК**.



- ❖ Чтобы установить оптимальные размеры канвы в соответствии с расположенными на ней элементами, следует нажать кнопку  на панели инструментов. В результате размеры автоматически скорректируются:

по ширине – к горизонтальной координате самой правой точки самого правого элемента добавляются 10 пикселей;

по высоте – к вертикальной координате самой нижней точки самого нижнего элемента добавляются 10 пикселей.

После изменения размеров канвы информационную панель следует сохранить.

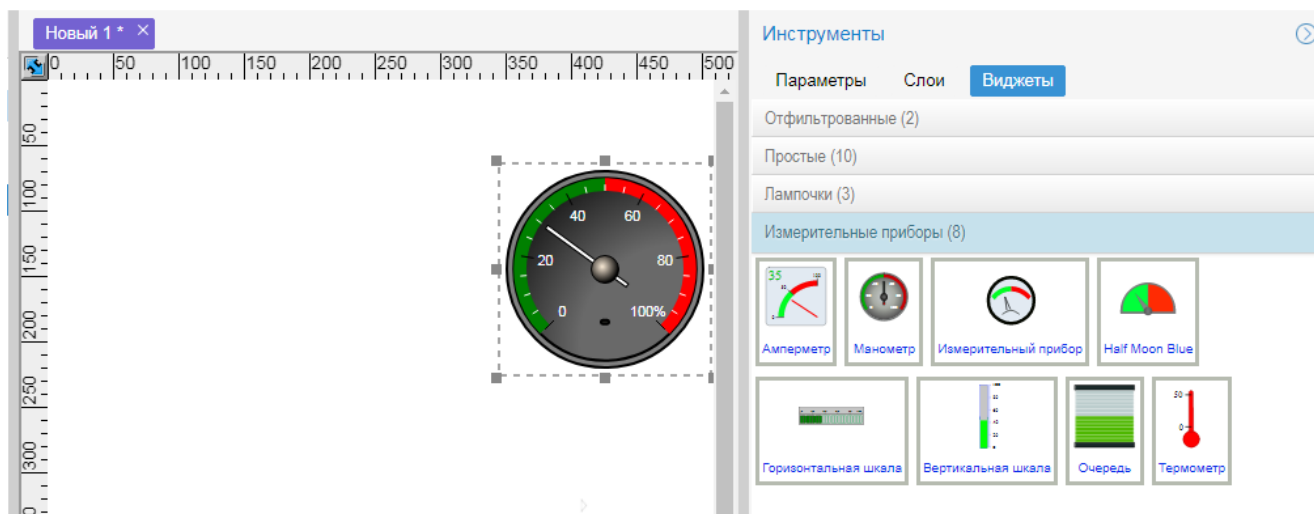
10.7.2. Расположение элементов на информационной панели

В данном разделе описаны приемы и методы размещения элементов информационной панели на канве.

10.7.2.1. Добавление элементов на канву

Чтобы добавить элемент на канву, следует:

1. на вкладке **Виджеты** открыть нужный раздел;
2. с помощью мыши перетащить иконку элемента на канву.



Иконки последних элементов, размещенных на канве, автоматически попадают на *панель недавно использованных виджетов*. Иконки с этой панели также можно перетаскивать на канву для добавления элемента.

10.7.2.2. Использование точек привязки

Точки привязки (ТП) – логические объекты, используемые для реализации связи между линиями и другими элементами информационной панели, а также для автоматического изменения размера или положения линии при изменении размера или положения связанного элемента.

Режим использования точек привязки можно установить и отключить с помощью пункта меню **Правка** → **Точки привязки** (клавиша **P**).

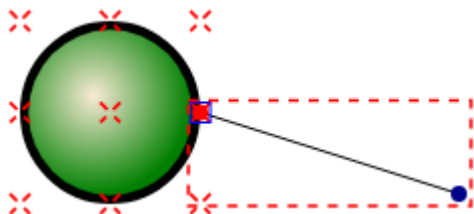
Элементы информационной панели (за исключением линий) имеют стандартные точки привязки: восемь по периметру и одну в центре. Положение точки привязки автоматически изменяется при смещении элемента на величину смещения и при изменении его размера – пропорционально изменению. Изменение положения точки привязки вызывает изменение размера или положения привязанных к ней линий.

Точка привязки линии к элементу автоматически добавляется при наведении конца линии на элемент. При этом действуют следующие правила:

- ❖ конец линии “прилипает” к уже имеющимся точкам привязки;
- ❖ конец линии “прилипает” к границе элемента;
- ❖ выбор элементов при наведении линии осуществляется из доступных для выделения, в обратном порядке их видимости;
- ❖ к одной точке привязки можно подключить любое количество линий;

10. Редактор информационных панелей

- ❖ положение точки привязки произвольное, в пределах границы элемента;
- ❖ при наведении линии на элемент все имеющиеся точки привязки этого элемента подсвечиваются крестиком;
- ❖ красным квадратом выделяется точка, к которой будет привязан конец линии, синим или черным – уже имеющиеся точки привязки линии.



Отсоединение линии от точки привязки элемента управления происходит автоматически в следующих случаях:

- ❖ перемещение конца линии за границу элемента;
- ❖ при создании группы, если один из элементов, линия или связанный элемент управления, не входит в группу.

При группировке элементов связи между элементами внутри группы сохраняются, но без возможности изменения. Точки привязки снова активизируются после разгруппировки.

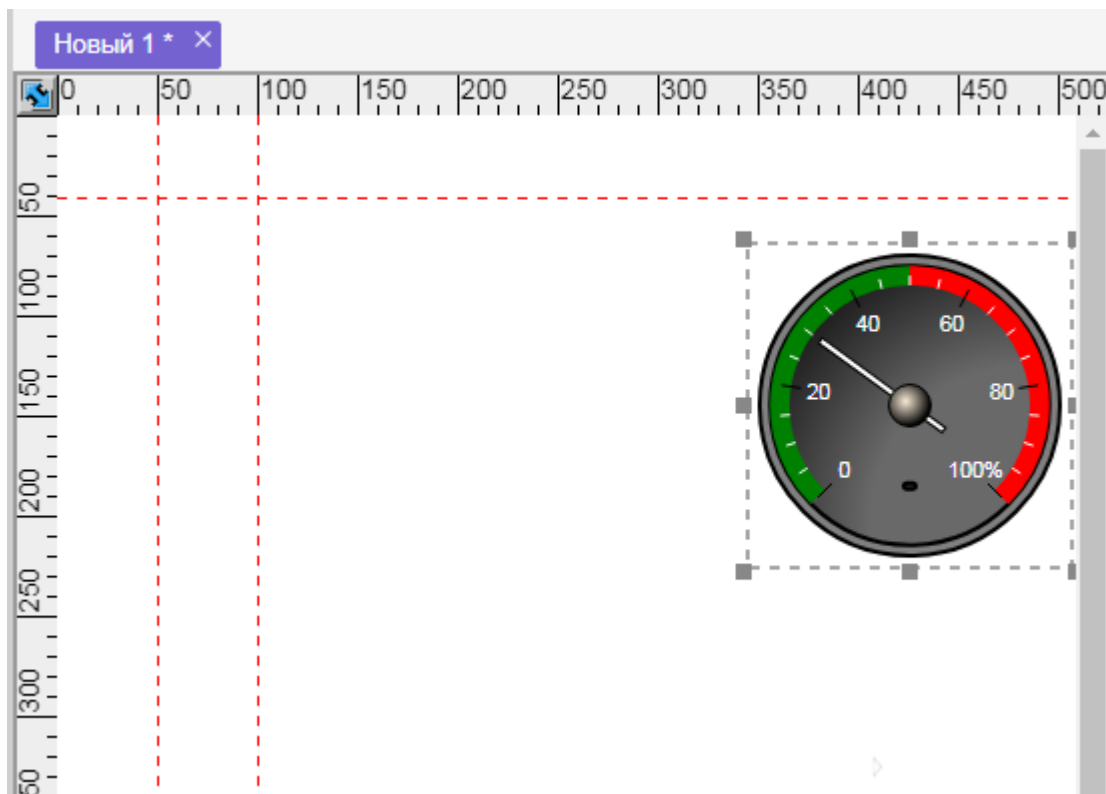
Примечание. В группе работает алгоритм пропорционального вычисления координаты точки (т.е. при изменении размера группы новая координата точки привязки скорректируется пропорционально).

10.7.2.3. Использование направляющих

Для удобства выравнивания элементов на канве можно использовать направляющие – горизонтальные или вертикальные пунктирные линии красного цвета.

Чтобы добавить направляющие, следует нажать мышью на линейке сверху или слева от области редактирования и потянуть в область редактирования. Указатель мыши изменит вид на крестик со стрелками на двух противоположных концах, и результате в области редактирования появится направляющая, параллельная выбранной линейке.

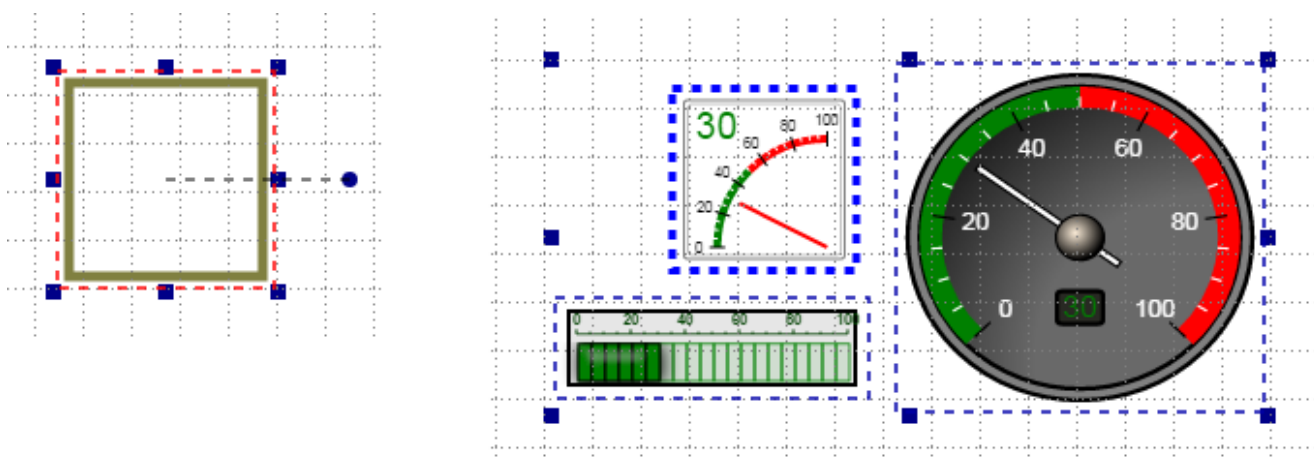
Настройки направляющих сохраняются для каждой информационной панели.



10.7.2.4. Выделение элементов на канве

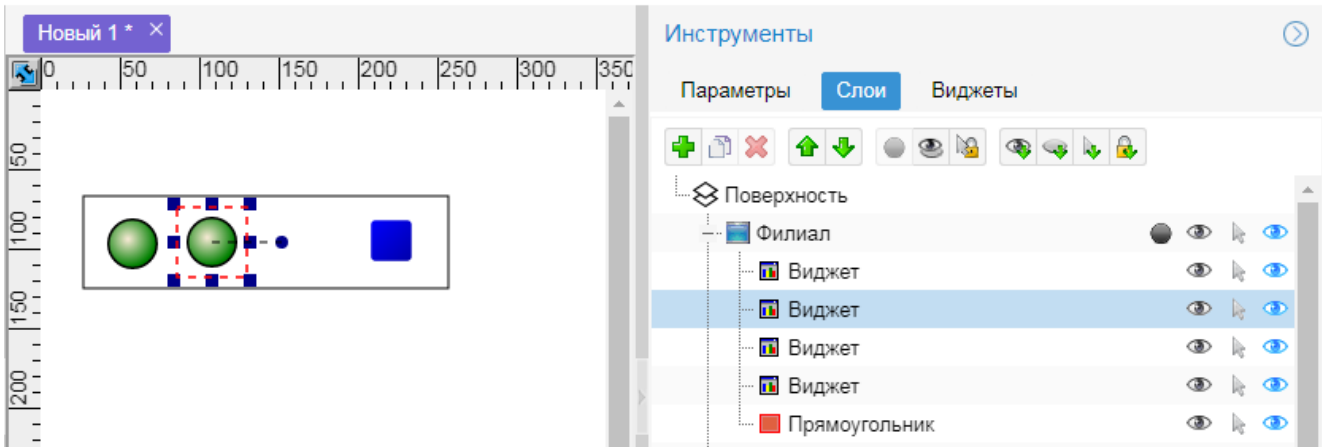
На канве при выделении одного элемента вокруг него появляется красная рамка с габаритными маркерами (темно-синими квадратиками), а также отображается направляющая и маркер поворота. Маркер поворота отсутствует у виджетов [Рамочный контейнер](#)^[336] и [Динамический график](#)^[329].

Синими рамками отмечаются несколько выделенных элементов, причем **основной** среди них обозначен жирной синей рамкой.



Чтобы выбрать (выделить) один элемент, достаточно щелкнуть на нем мышью в области редактирования или на вкладке **Слой**. Если элемент входит в группу, то для выбора элемента следует щелкнуть на нем мышью, удерживая при этом клавишу **Alt**.

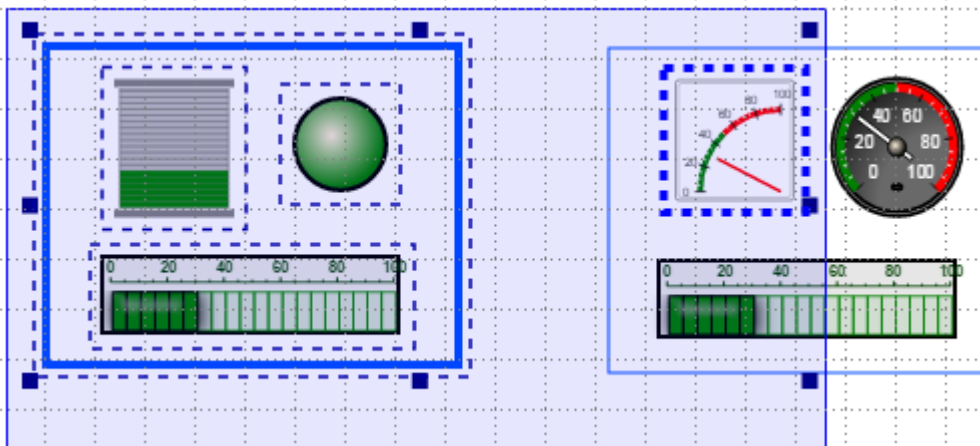
10. Редактор информационных панелей



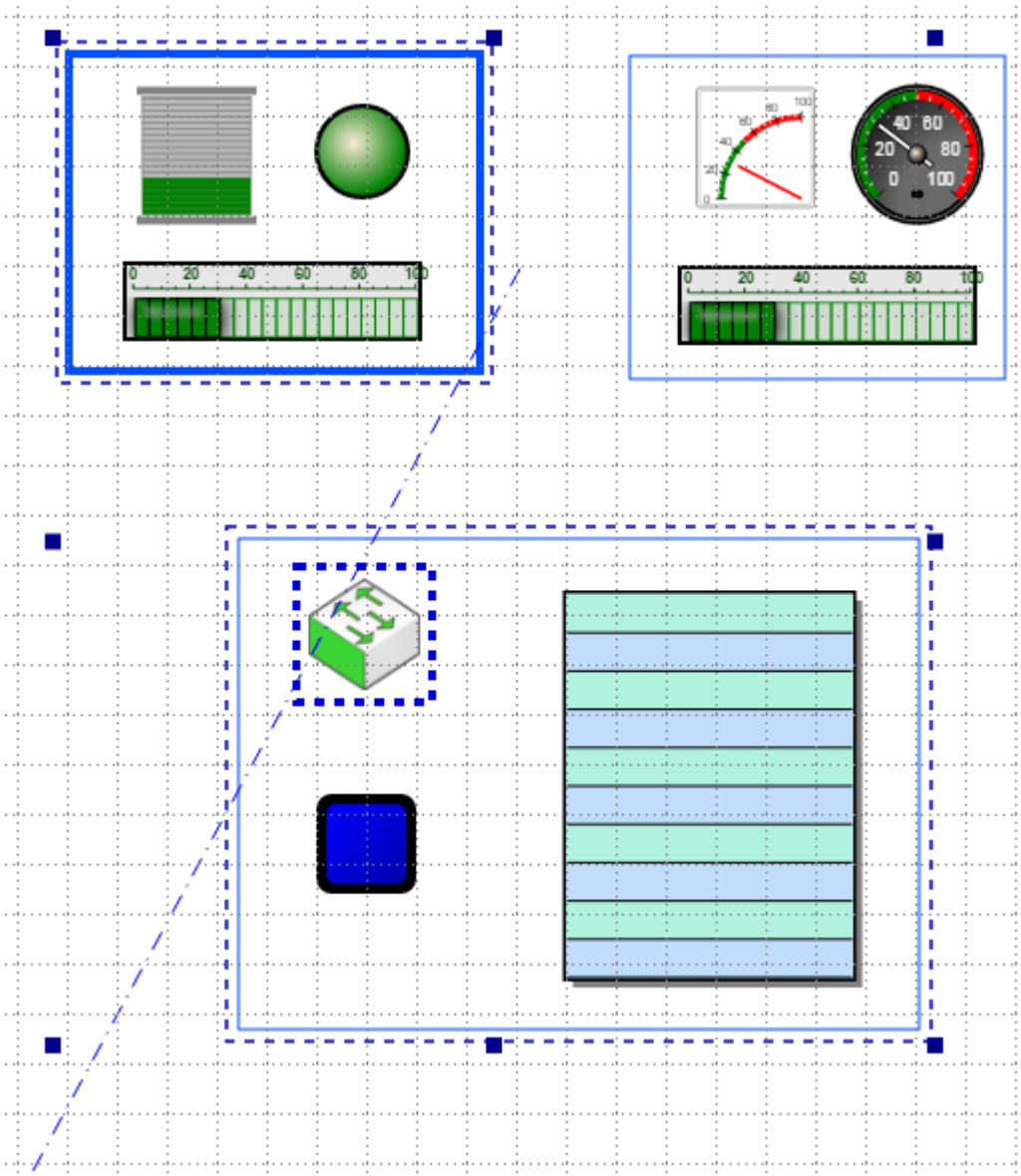
Для выделения нескольких элементов или групп элементов в области редактирования можно воспользоваться одним из методов:

10.7. Инструментарий Редактора информационных панелей

- ❖ удерживая левую кнопку мыши, растянуть область выделения, и затем отпустить кнопку мыши; в результате выделяются элементы, рамки которых полностью попадают в область выделения;





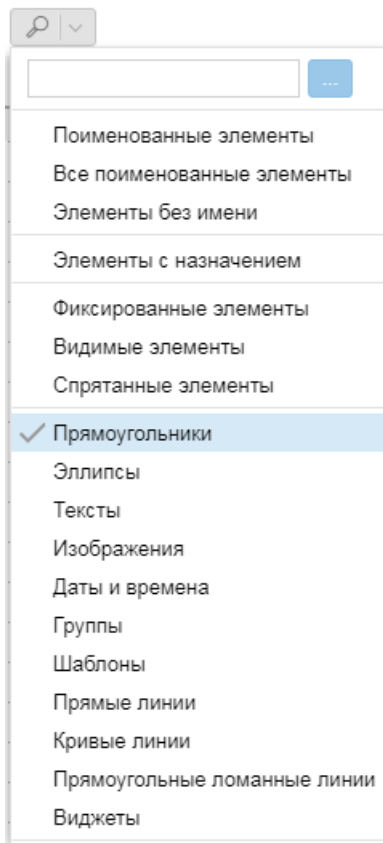
- ❖ удерживая клавишу **Ctrl** и левую кнопку мыши, указателем мыши нарисовать прямую пунктирную линию – в результате выделяются элементы, которые пересекает линия;



- ❖ выделить мышью один элемент и затем, удерживая клавишу **Shift**, выделить мышью еще несколько элементов (последний из них станет основным);

10. Редактор информационных панелей

- ❖ удерживая клавишу **Ctrl**, в окне **Слои** отметить с помощью мыши названия требуемых объектов (последний из них станет основным);
- ❖ на панели инструментов нажать на стрелку в правой части кнопки , отметить признак для поиска или ввести фрагмент имени и нажать на лупу в левой части кнопки .



В результате найденные по заданному признаку элементы будут выделены на канве, а также перечислены в окне **Результаты поиска**.

Для выделения нескольких элементов, входящих в состав группы, следует использовать перечисленные выше приемы с одновременным нажатием клавиши **Alt**.

10.7.2.5. Перемещение и изменение размера элемента

Для изменения размеров элемента и перемещения его на канве следует сначала выделить элемент, а затем – использовать один из способов:

- ручной ввод значений параметров (вкладка **Параметры**, поля в разделе [Позиция и размер](#)^[179]);
- растяжение/сжатие элемента и перемещение с помощью мыши (значения в окне **Параметры** при этом изменяются автоматически). Подробности далее.

Перемещение и растяжение/сжатие элемента

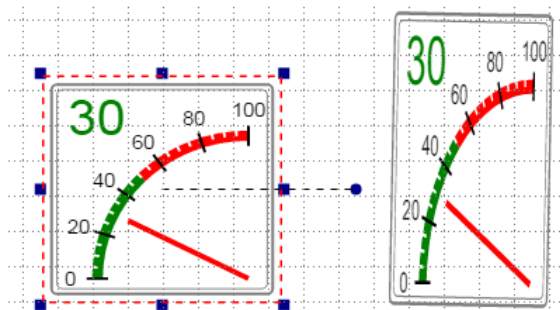
Для растяжения/сжатия элемента по вертикали и горизонтали следует потянуть мышью за габаритный маркер (темно-синий квадратик на рамке) либо использовать клавиши со стрелками, удерживая при этом клавишу **Ctrl**.

Для перемещения элемента по канве можно использовать:

- ❖ левую кнопку мыши (стандартный способ);
- ❖ клавиши со стрелками – каждое нажатие клавиши смещает элемент на 1 пиксел;

- ❖ клавиши со стрелками с одновременным нажатием клавиши **Alt** – каждое нажатие комбинации клавиш смещает элемент на 10 пикселей.

Для растяжения/сжатия элемента по вертикали и горизонтали следует потянуть мышью за габаритный маркер (темно-синий квадратик на рамке) либо использовать клавиши со стрелками, удерживая при этом клавишу **Ctrl**.

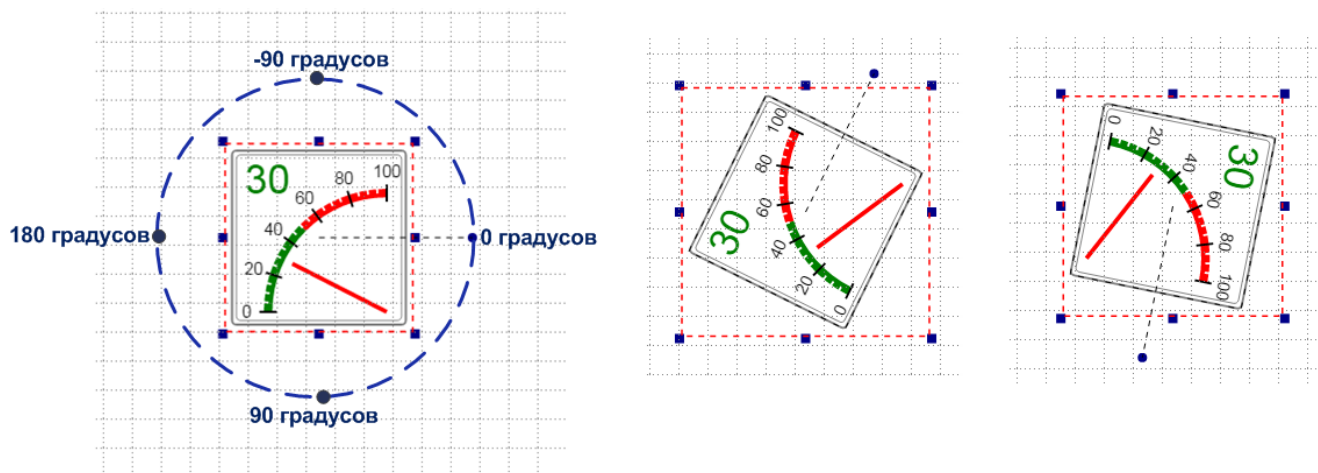


Изменять размеры и местоположение можно сразу у несколько выделенных элементов.

Поворот элемента

Чтобы включить возможность вращения элементов с помощью мыши на канве, следует отметить пункт меню **Правка** → **Направляющие поворота**. Тогда при выделении элемента отображается его направляющая поворота – пунктирная линия от центра элемента до маркера поворота.

Чтобы повернуть элемент, следует мышью перетащить маркер поворота в нужном направлении. Шкала измерения угла поворота показана на рисунке ниже. По мере вращения элемента изменяется значение параметра **Угол** (раздел **Позиция и размер** на вкладке **Параметры**).



Примечание. Возможность поворота отсутствует у виджетов **Рамочный контейнер** и **Динамический график**.

10.7.2.6. Группировка и разгруппировка элементов

При разработке информационной панели иногда удобно сгруппировать элементы. Управление группами возможно как непосредственно на канве, так и с помощью дерева элементов, отображаемого на вкладке **Слой**.

Сгруппированные элементы можно передвигать по канве только совместно. Группы элементов можно перемещать с одного слоя на другой.

10. Редактор информационных панелей

Для элементов, объединенных в группу, можно установить единое значение параметра **Видимость** в разделе **Отображение**. Изменять значения других параметров сразу для группы элементов не допускается.

По умолчанию для группы устанавливается признак видимости, при этом видимость отдельных элементов не меняется. При переключении признака видимости для группы аналогичным образом изменяется признак видимости для входящих в нее элементов, а при расформировании группы признак видимости сохраняется.

Если элемент входит в состав группы, то чтобы выбрать его в области редактирования, следует щелкнуть по нему мышью мышью, удерживая при этом клавишу **Alt**, либо пользоваться вкладкой **Слои** ^[205].

Группировка элементов

Для того, чтобы сгруппировать элементы, необходимо выделить несколько элементов и в контекстном меню выбрать пункт **Сгруппировать**.

Чтобы создать группу элементов с помощью окна **Система слоёв**, следует:

1. в окне **Слои** выделить несколько элементов и/или групп;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Создать группу**. Будет создана группа в самом верхнем слое из тех, в которых выбраны элементы.

Назначение имени группе элементов

Для того чтобы задать имя всей группе, следует:

1. в окне **Слои** выделить группу;
2. с помощью правой кнопкой мыши в контекстном меню выбрать пункт **Изменить имя...**;
3. в открывшемся окне ввести имя группы и нажать **ОК**. Группа будет переименована, её имя появится в **дереве слоёв**.

Добавление элементов в группу

В существующую группу можно добавить элемент или группу элементов в качестве вложенной группы. Для этого следует:

1. на канве выделить группу элементов, в которую требуется вставить элементы, и, удерживая клавишу **Shift**, выделить добавляемый элемент или группу элементов; добавляемый элемент (или группа элементов) должен быть выделен жирной пунктирной рамкой, т.е. быть основным элементом; можно воспользоваться и другими способами выделения элементов, см. Выделение элементов на канве;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Добавить в группу <Имя группы>**.

Разгруппировка элементов

Для расформирования группы можно воспользоваться одним из способов:

- выделить группу на канве и с помощью правой кнопки мыши в контекстном меню выбрать пункт **Разгруппировать**;
- выделить группу в окне **Система слоёв** и с помощью правой кнопки мыши в контекстном меню выбрать пункт **Раскрыть группу**.

10.7.3. Управление слоями информационной панели

Использование функций со слоями упрощает процесс создания информационной панели. Для работы с системой слоев необходимо в окне **Инструменты** открыть вкладку **Слой**. Она имеет собственную панель инструментов.

Рабочая область вкладки отображает структуру информационной панели в виде дерева, в узлах которого находятся ее элементы, группы элементов и слои. В качестве названия элемента или группы используется значение поля **Имя** на вкладке **Параметры**. Если оно не задано, то в дереве в качестве названия элемента выводится его тип (**Прямоугольник**, **Эллипс**, **Виджет** и т.д.).

Возможны вложенные группы.

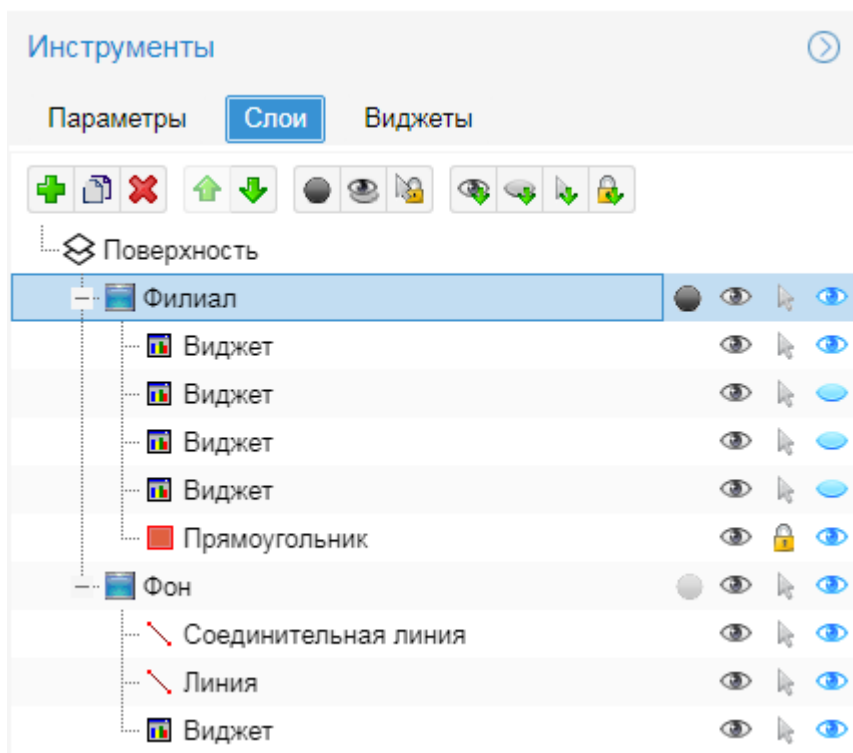
В четырех столбцах справа с помощью значков отображаются свойства слоев и элементов:

1-й столбец – признак слоя: ● – основной слой, ○ – остальные слои;

2-й столбец – видимость слоя/элемента на канве: 👁 – видимый, 👁 – невидимый;

3-й столбец – доступность слоя-элемента для выбора и изменения: 🖱 – доступен, 🔒 – заблокирован;













4-й столбец – видимость слоя/элемента на информационной панели (при просмотре конечным пользователем): 👁 – видимый, 👁 – невидимый.



В дереве слоев можно выбрать элемент для просмотра или изменения параметров в области свойств. Для выбора элемента достаточно щёлкнуть мышью на соответствующем названии. Для выбора нескольких элементов следует при щелчке мыши удерживать клавишу **Ctrl**. При выделении одного или нескольких элементов в области редактирования эти элементы также выделяются в области слоёв.

10.7.3.1. Панель инструментов и контекстное меню окна Система слоёв

На вкладке **Слои** на панели инструментов расположены следующие функциональные кнопки:

-  – [добавление нового слоя](#) ^[207];
-  – [копирование выбранного слоя](#) ^[207];
-  – [удаление выбранного слоя](#) ^[208];
-  – [перемещение выбранного объекта или слоя на одну позицию выше](#) ^[209];
-  – [перемещение выбранного объекта или слоя на одну позицию ниже](#) ^[209];
-  – [установка выбранного слоя основным \(рабочим\)](#) ^[208];
-  – [переключение видимости выбранного слоя или элемента](#) ^[208];
-  – [блокировка/разблокировка \(переключение доступности\) выбранного слоя или элемента](#) ^[209];
-  – [сделать видимыми все слои](#) ^[208];
-  – [скрыть все слои](#) ^[208];
-  – [разблокировать все слои \(включить доступность\)](#) ^[209];
-  – [заблокировать все слои \(отключить доступность\)](#) ^[209].

Дерево слоев имеет контекстное меню. Каждый его пункт доступен, если в дереве слоев выделен соответствующий узел (элемент, группа элементов или слой).

Переместить на...

[Переместить выбранный элемент информационной панели на указанный слой](#) ^[209].

Создать группу

Создать группу из выбранных элементов или групп.

Раскрыть группу

Разгруппировать выбранную группу.

Изменить виджет...

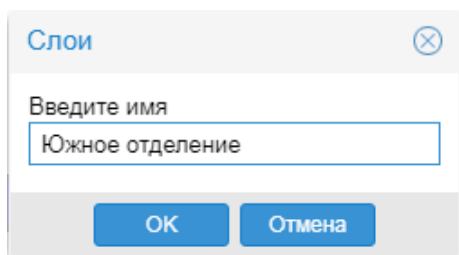
Изменить параметры выделенного виджета. Открывается окно со специфическими параметрами виджета (если для виджета такое окно реализовано).

10.7.3.2. Действия со слоями и элементами

Далее рассматриваются основные действия со слоями и элементами, доступные на вкладке **Слои**.

Создание нового слоя

Для создания нового слоя следует нажать кнопку  и в появившемся окне ввести его имя.




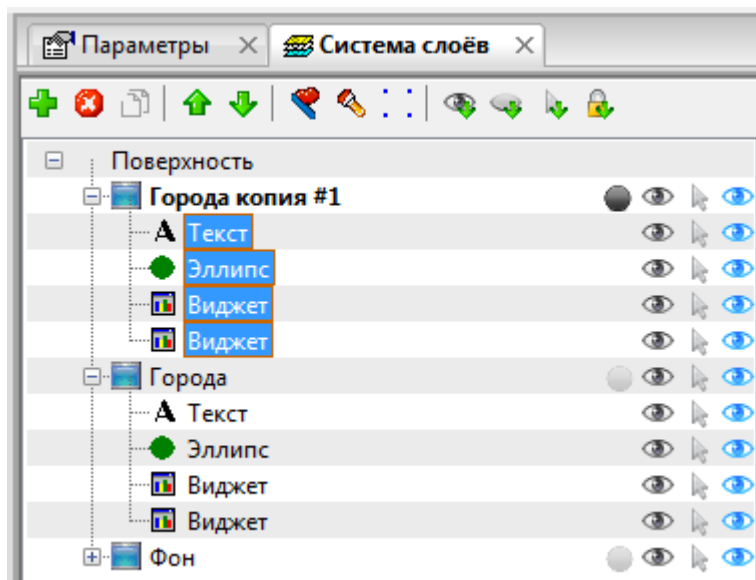
В дальнейшем можно изменить имя слоя, кликнув на нем мышью.

В дереве слоёв каждый добавляемый слой располагается выше остальных.

Копирование слоя

При копировании слоя создается новый слой с тем же набором объектов, что и в исходном слое.

Чтобы скопировать слой, следует в дереве слоёв выделить исходный слой и нажать кнопку . В результате в дереве появится копия выделенного слоя.




Примечание. При копировании невидимого или недоступного слоя новый слой будет виден и доступен для редактирования.

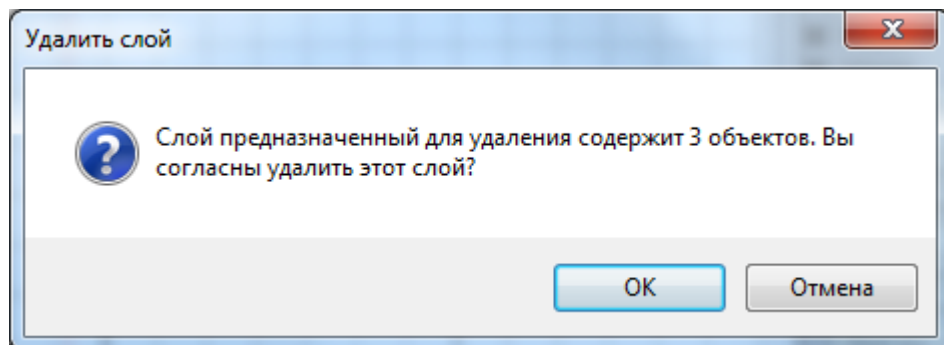
Созданные объекты отображаются в области редактирования со смещением относительно объектов исходного слоя.

Удаление слоя

Удаление слоя приводит к удалению расположенных на нем элементов. Если на информационной панели всего один слой, то его нельзя удалить.


Чтобы удалить слой, следует

1. поместить курсор на соответствующую строку в списке слоев;
2. нажать кнопку ;
3. для подтверждения удаления в появившемся окне нажать кнопку **ОК**.






10.7.3.3. Изменение свойств слоев и элементов



Установка основного слоя




На вкладке **Слой** основной слой помечен черной точкой . Его невозможно сделать невидимым. Элементы, добавляемые на канву путем переноса со вкладки **Виджеты**, попадают в основной слой.



По умолчанию при создании информационной панели автоматически создается фоновый слой, который является основным.

Чтобы установить основной слой, следует нажать на серую точку  напротив имени слоя либо поместить курсор на соответствующую строку и нажать кнопку . В результате точка станет черной .

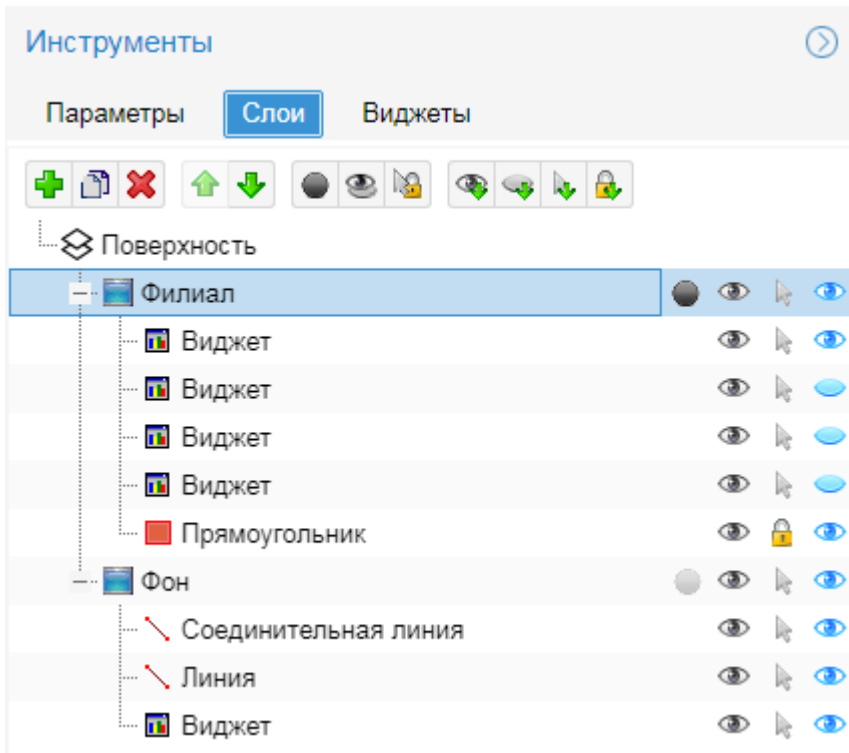
Изменение признака видимости на канве для слоя или элемента

Признак видимости устанавливается как для элементов, так и для слоев. Он отображается в виде значков:  (видимый) и  (скрытый).

Для переключения признака видимости слоя или элемента достаточно щелкнуть мышью на значке признака ( или ) напротив элемента или слоя либо поместить курсор на строку слоя и нажать кнопку .

Чтобы сделать видимыми сразу все элементы и слои, следует нажать кнопку . Чтобы скрыть сразу все элементы и слои, за исключением основного, следует нажать кнопку .

Скрытые элементы и слои не отображаются в области редактирования. Невидимость для слоя приоритетнее, чем для элемента, т.е. если скрыт целый слой, то скрываются и его элементы, вне зависимости от их признаков видимости.



Изменение признака видимости на Web-интерфейсе для слоя или элемента

Для переключения признака видимости в Web-интерфейсе достаточно щелкнуть мышью на значке признака (👁️ или 👁️ – голубой глаз в последнем столбце) напротив элемента или слоя.

Изменение доступности слоя или элемента

При разработке информационной панели отдельные элементы или целые слои можно блокировать от возможности выбора и случайного изменения и перемещения.

Признак доступности отображается в виде значков: 🖱️ (доступен для выбора и редактирования) и 🔒 (установлена блокировка).

Для переключения признака видимости достаточно щелкнуть мышью на значке признака (👁️ или 🔒) напротив элемента/слоя либо поместить курсор на строку элемента/слоя и нажать кнопку 🖱️.

Чтобы сделать доступными все заблокированные слои и элементы, следует нажать кнопку 📁. Для блокировки сразу всех слоев и элементов следует нажать кнопку 🔒.

10.7.3.4. Перемещение слоев и элементов



Окно Система слоёв позволяет перемещать элементы информационной панели внутри слоя, между слоями, а также перемещать слои между собой.

Перемещение элементов внутри слоя и перемещение слоев

Как объекты можно перемещать внутри слоя, так и слои можно перемещать относительно друг друга.

10. Редактор информационных панелей

Чтобы изменить местоположение одного или нескольких объектов внутри слоя, следует:

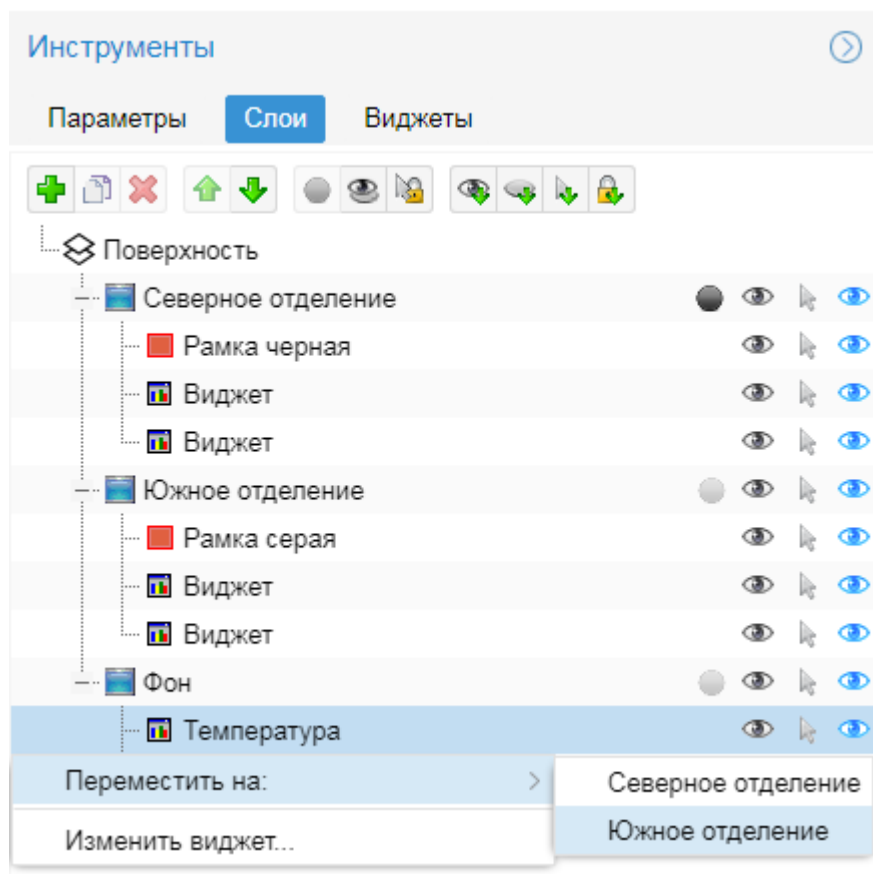
1. в дереве окна **Система слоёв** следует выделить один или несколько соседних объектов (с помощью мыши и клавиши **Shift**);
2. последовательно нажимать кнопку  (или клавишу **[-]**) для перемещения на одну позицию вверх или кнопку  (или клавишу **[+]**) для перемещения на одну позицию вниз.

Аналогичным образом можно изменить местоположение слоев.

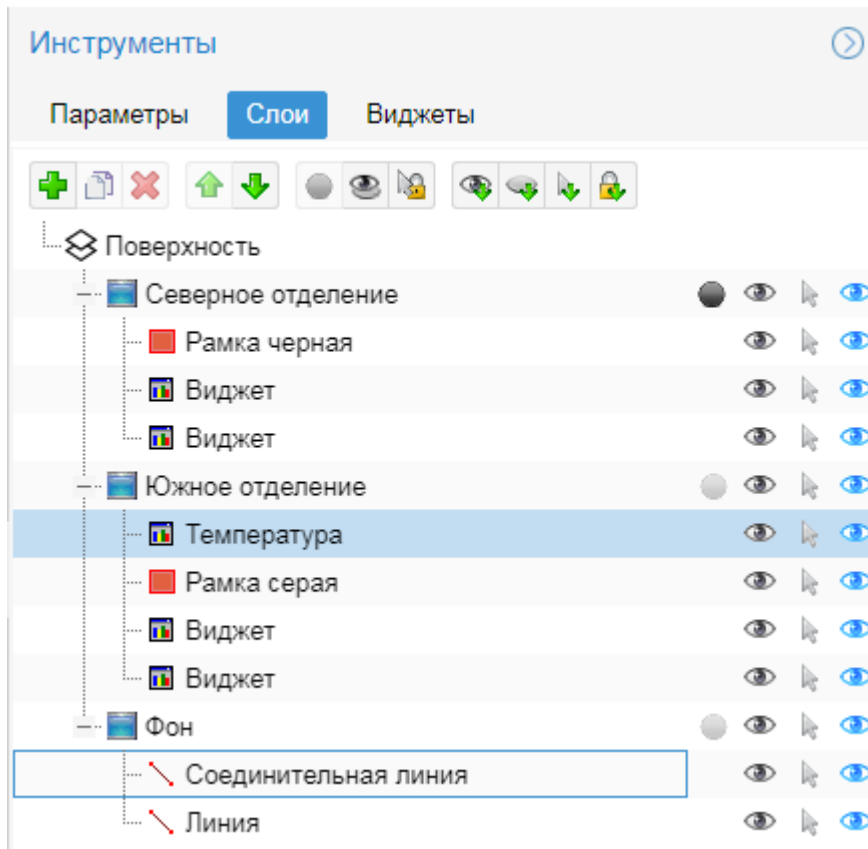
Перемещение элементов между слоями

Чтобы поместить элемент на нужный слой, следует:

1. выделить элемент в дереве вкладки **Слой** или в области редактирования;
2. в контекстном меню выбрать пункт **Переместить на:** и далее – название слоя:



В результате объект привязывается к выбранному слою и появляется в дереве окна **Система слоёв** в соответствующем слое.



11.

Администрирование модуля Нувола Координатор

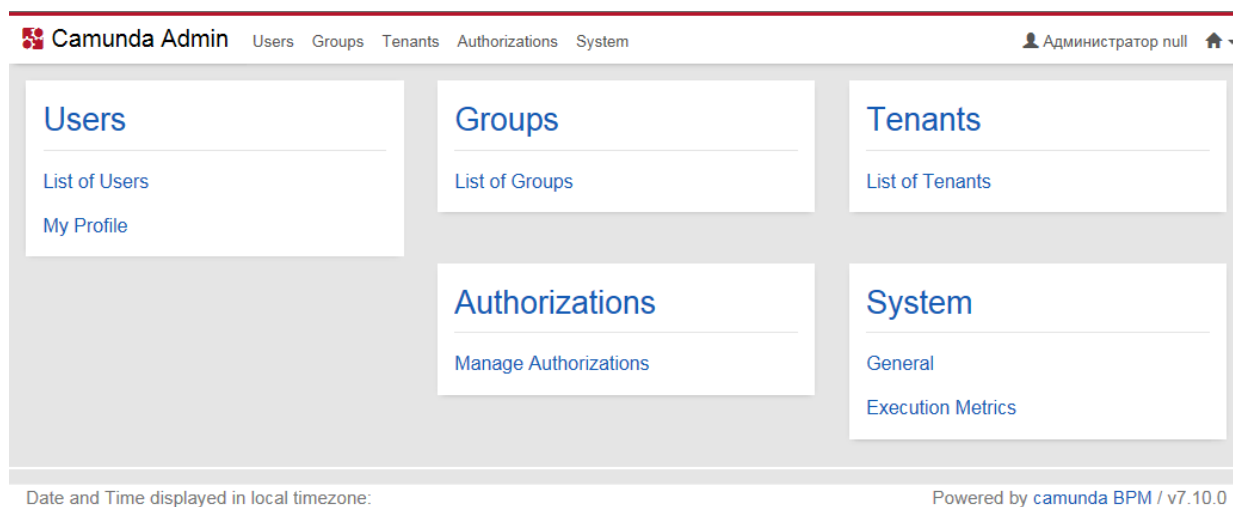
11. Администрирование модуля Нуволы Координатор

В настоящем разделе кратко описаны действия администратора по подготовке к работе ПМ **Нуволы Координатор** с указанием используемых инструментов.

11.1. Обзор приложений Camunda BPM

Платформа Camunda BPM предоставляет администратору следующие инструменты:

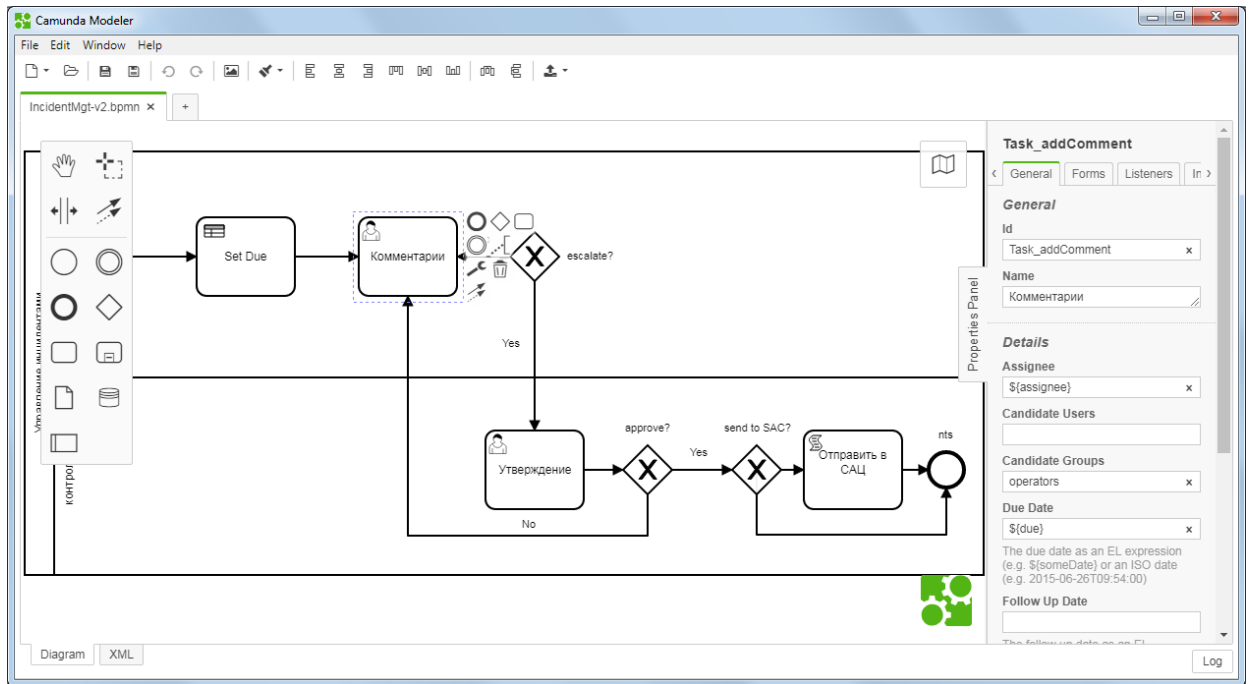
- ❖ приложение Camunda Modeler – конструктор схем бизнес-процессов; см. документацию на сайте <http://camundarus.ru/products/modeler/>;
- ❖ web-приложение Camunda Cockpit – позволяет управлять схемами бизнес-процессов; см. документацию на сайте <http://camundarus.ru/products/cockpit/>;
- ❖ web-приложение Camunda Admin – предназначено для администрирования пользователей; документация на сайте <http://camundarus.ru/products/admin/>.



11.2. Настройка схем бизнес-процессов

Общий порядок настройки схемы бизнес-процессов:

1. создание схемы бизнес-процесса в приложении Camunda Modeler, установленном на рабочей станции администратора, см. также [Установка и удаление модуля Нуволы Координатор](#)^[58];



2. загрузка схемы на сервер Координатор через приложение [Координатор](#) ²²⁹.

Для управления схемами бизнес-процессов, введенными в действие, рекомендуется использовать Web-приложение Camunda Cockpit. См. документацию на сайте: <https://camundarus.ru/products/cockpit/>.

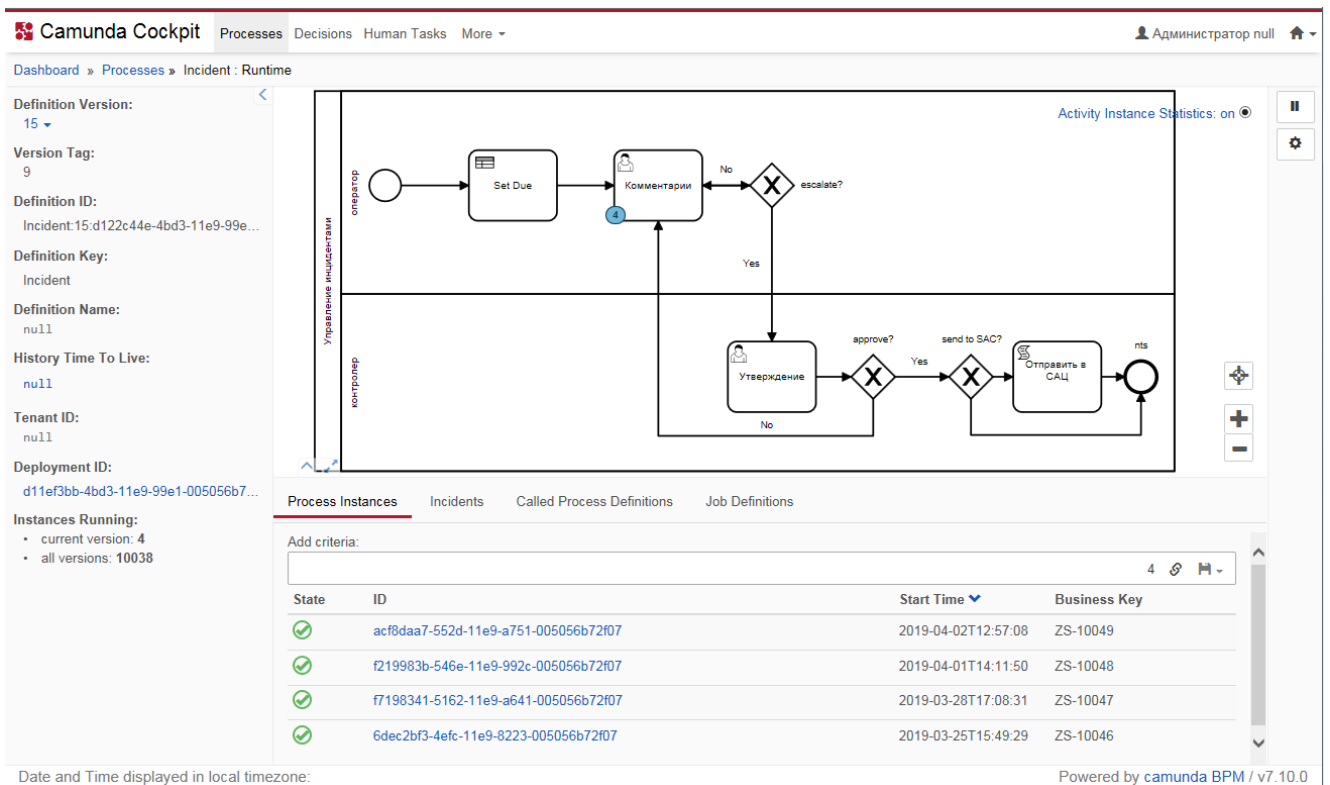


Рисунок 1. Схема бизнес-процесса в Web-приложении Camunda Cockpit

11.3. Особенности использования скриптов в схемах бизнес-процессов

При разработке схем бизнес-процессов в приложении Camunda Modeler администратор может использовать скрипты, поставляемые в комплекте с ПМ **Нувола Координатор**. В настоящем разделе описаны особенности их применения.

Общие настройки скриптов приведены в разделе [Настройка скриптов кастомизации платформы Camunda BPM](#).

В настоящем разделе используются следующие обозначения путей по умолчанию:

- ❖ lanceLOT_home – /usr/local/lancelot/
- ❖ bpm_home – /usr/local/bpm/

11.3.1. ReportStatus.js

Скрипт ReportStatus.js предназначен для отправки сообщения о смене статуса бизнес-процесса в ПК Нувола.

Для успешной работы скрипта необходимы следующие условия:

- ❖ в схеме бизнес-процесса процессе существуют переменные alert_ph и alert_time_state;
- ❖ существует таблица данных lanceLOT_alerts.bpm;
- ❖ в конфигурационном файле bpm_home/conf/camunda_process_states.json описаны пары "ключ - значение", где ключом является activityId бизнес-процесса, а значением – числовое значение желаемого статуса. Предусмотренные значения:

статус 0 – состояние не определено;

статус 1 – процесс начат, но не завершен;

статус 2 – процесс завершен;

The screenshot shows the configuration for a StartEvent_1 in Camunda Modeler. The 'General' tab is selected, and the 'Id' field is highlighted with a red box, containing the value 'StartEvent_1'. The 'Name' field is empty. The 'Details' section shows the 'Initiator' field set to 'demo'. The 'Asynchronous Continuations' section has two unchecked options: 'Asynchronous Before' and 'Asynchronous After'. The 'Documentation' section has an empty 'Element Documentation' field.

- ❖ скрипт прикреплен к задаче бизнес-процесса следующим образом:

The screenshot shows the configuration for an execution listener on the 'end' event of a task named 'StartEvent_1'. The listener is of type 'Script' and uses the 'JavaScript' format. It is an external resource named 'ReportStatus.js'. The configuration is shown in a tabbed interface with 'Listeners' selected.

- ❖ после изменений схема сохранена и выгружена на сервер.

11.3.2. SendMail.js

Скрипт `SendMail.js` инициирует в ПК Нувола отправку сообщения о текущем состоянии задачи по электронной почте указанным пользователям.

Письма отправляются в следующих случаях:

- ❖ изменен пользователь в поле **Ответственный**;
- ❖ задача создана и в поле **Ответственный** указана группа пользователей (`candidateGroups`).

Для успешной работы скрипта необходимы следующие условия:

- ❖ в задаче, соответствующей указанному `taskId`, будет заполнено поле `assignee` или в схеме будут указаны `candidateGroups`;
- ❖ для пользователей, которым предназначены сообщения, в ПК Нувола заведены учетные записи с заполненным полем **Эл.почта**;
- ❖ в директории `lancelot_home/themes/camunda` (при отсутствии папки, её следует создать) существует шаблон сообщения, именем которого является конструкция `definitionKey.j2`. Шаблон следует создавать с помощью языка шаблонов [Jinja2](http://jinja.pocoo.org/docs/2.10/) (<http://jinja.pocoo.org/docs/2.10/>);

11. Администрирование модуля Нувола Координатор

Task_addComment

< General Forms Listeners In >

General

Id **definitionKey**

Task_addComment x

Name

Комментарии

Details

Assignee

#{assignee} x

Candidate Users

Candidate Groups

operators x

Due Date

#{due} x

The due date as an EL expression (e.g. #{someDate}) or an ISO date (e.g. 2015-06-26T09:54:00)

Follow Up Date

The follow up date as an EL expression (e.g. #{someDate}) or an ISO date (e.g. 2015-06-26T09:54:00)

Priority

- ❖ заполнена логика обработки шаблона в функции `createContent` в модуле `bpm_home/python/rest/camunda/custom.py`;
- ❖ заполнен конфигурационный файл `bpm_home/conf/bpm_camunda.json`;
- ❖ скрипт дважды прикреплен к задаче бизнес-процесса следующим образом:

The image displays two side-by-side screenshots of the configuration interface for a task named 'Task_addComment'. Both screenshots show the 'Listeners' tab selected. The left screenshot shows the 'Task Listener' list with 'assignment : Script' selected. The right screenshot shows the 'Task Listener' list with 'create : Script' selected. Below the list, the 'Task Listener' configuration panel shows the 'Event Type' dropdown set to 'assignment' in the left view and 'create' in the right view. Other settings like 'Listener Type' (Script), 'Script Format' (JavaScript), 'Script Type' (External Resource), and 'Resource' (SendMail.js) are consistent between both views. The 'Field Injection' section is empty in both.

11.3.3. SendToSAC.js

Скрипт `SendToSAC.js` предназначен для отправки сообщений в стороннюю систему. Данный скрипт запускается отдельной скриптовой задачей.

11. Администрирование модуля Нуволы Координатор

Task_0e5jc9x

< General Listeners Input/Output >

General

Id
Task_0e5jc9x x

Name
Отправить в САЦ

Details

Script Format
JavaScript x

Script Type
External Resource ▾

Resource
SendToSAC.js x

Result Variable

Asynchronous Continuations

Asynchronous Before

Asynchronous After

Documentation

Element Documentation

Для использования данного скрипта необходимы дополнительные настройки, предусмотренные проектным решением.

12.

Приложение Координатор

12. Приложение Координатор

Приложение **Координатор** является компонентой ПМ **Нувола Координатор** и позволяет выполнять задачи в соответствии со схемами бизнес-процессов, настроенными администратором с помощью Camunda Modeler.

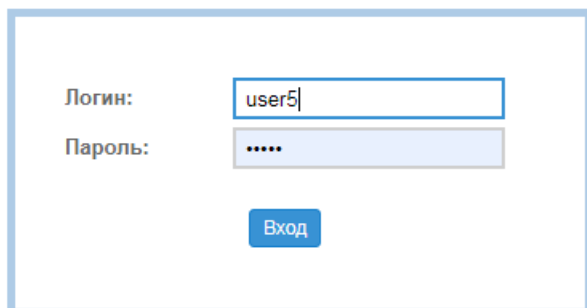
12.1. Вход и выход из приложения

Вход в приложение

Для входа в приложение **Координатор** можно использовать разные способы.

В общем случае, чтобы войти в приложение, следует:

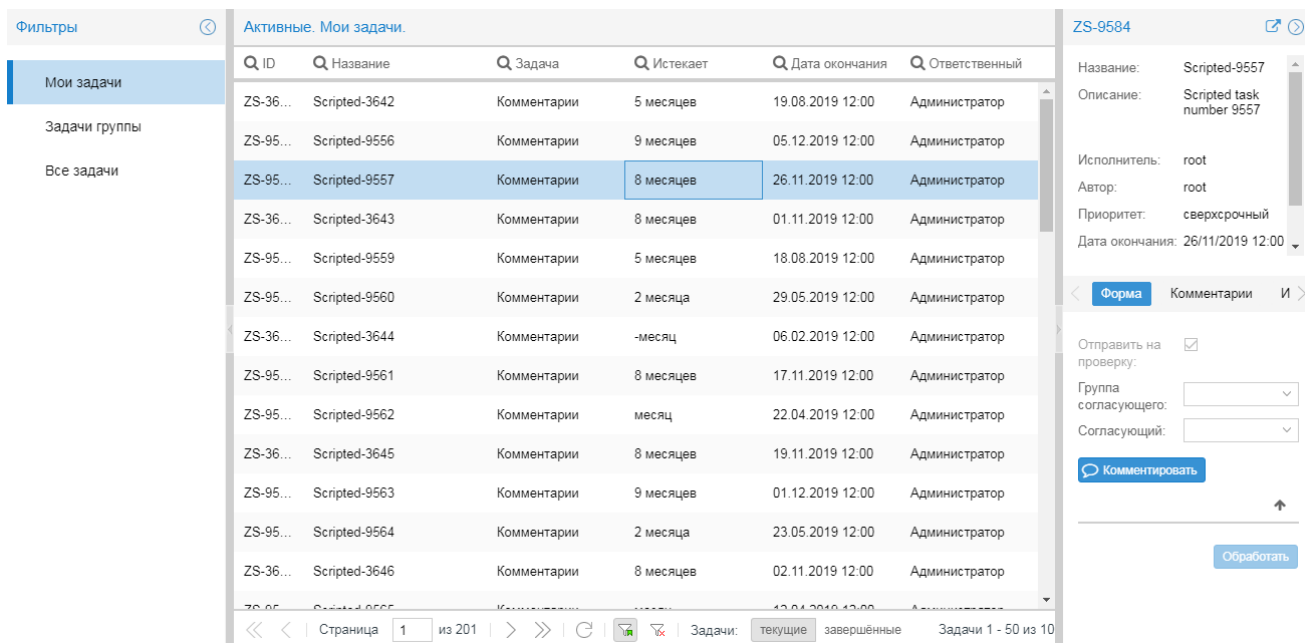
1. в браузере ввести адрес сервера Координатор. В результате откроется приглашение:



2. ввести имя пользователя (логин), пароль и нажать кнопку **Вход**.

Если соединение с сервером не установлено по какой-либо причине (например, введен неверный пароль или учетная запись заблокирована), будет выведено соответствующее сообщение.

В случае успешного подключения откроется основное окно приложения.



Q ID	Q Название	Q Задача	Q Истекает	Q Дата окончания	Q Ответственный
ZS-36...	Scripted-3642	Комментарии	5 месяцев	19.08.2019 12:00	Администратор
ZS-95...	Scripted-9556	Комментарии	9 месяцев	05.12.2019 12:00	Администратор
ZS-95...	Scripted-9557	Комментарии	8 месяцев	26.11.2019 12:00	Администратор
ZS-36...	Scripted-3643	Комментарии	8 месяцев	01.11.2019 12:00	Администратор
ZS-95...	Scripted-9559	Комментарии	5 месяцев	18.08.2019 12:00	Администратор
ZS-95...	Scripted-9560	Комментарии	2 месяца	29.05.2019 12:00	Администратор
ZS-36...	Scripted-3644	Комментарии	-месяц	06.02.2019 12:00	Администратор
ZS-95...	Scripted-9561	Комментарии	8 месяцев	17.11.2019 12:00	Администратор
ZS-95...	Scripted-9562	Комментарии	месяц	22.04.2019 12:00	Администратор
ZS-36...	Scripted-3645	Комментарии	8 месяцев	19.11.2019 12:00	Администратор
ZS-95...	Scripted-9563	Комментарии	9 месяцев	01.12.2019 12:00	Администратор
ZS-95...	Scripted-9564	Комментарии	2 месяца	23.05.2019 12:00	Администратор
ZS-36...	Scripted-3646	Комментарии	8 месяцев	02.11.2019 12:00	Администратор

Настройка задачи ZS-9584:

- Название: Scripted-9557
- Описание: Scripted task number 9557
- Исполнитель: root
- Автор: root
- Приоритет: сверхсрочный
- Дата окончания: 26/11/2019 12:00

Действия: Форма, Комментарии, И


Отправить на проверку:

Группа согласующего:

Согласующий:


Кнопки: **Комментировать**, **Обработать**

ПМ **Нувола Координатор**, согласно проектному решению, может быть интегрирован с другими компонентами ПК Нувола. В этом случае войти в приложение **Координатор** можно также через Web-интерфейс. Возможные способы:

- ❖ в заголовке страницы Web-интерфейса ПК Нувола следует нажать кнопку  и в меню выбрать иконку с названием приложения;

- ❖ через **Консоль событий** при создании бизнес-процесса.

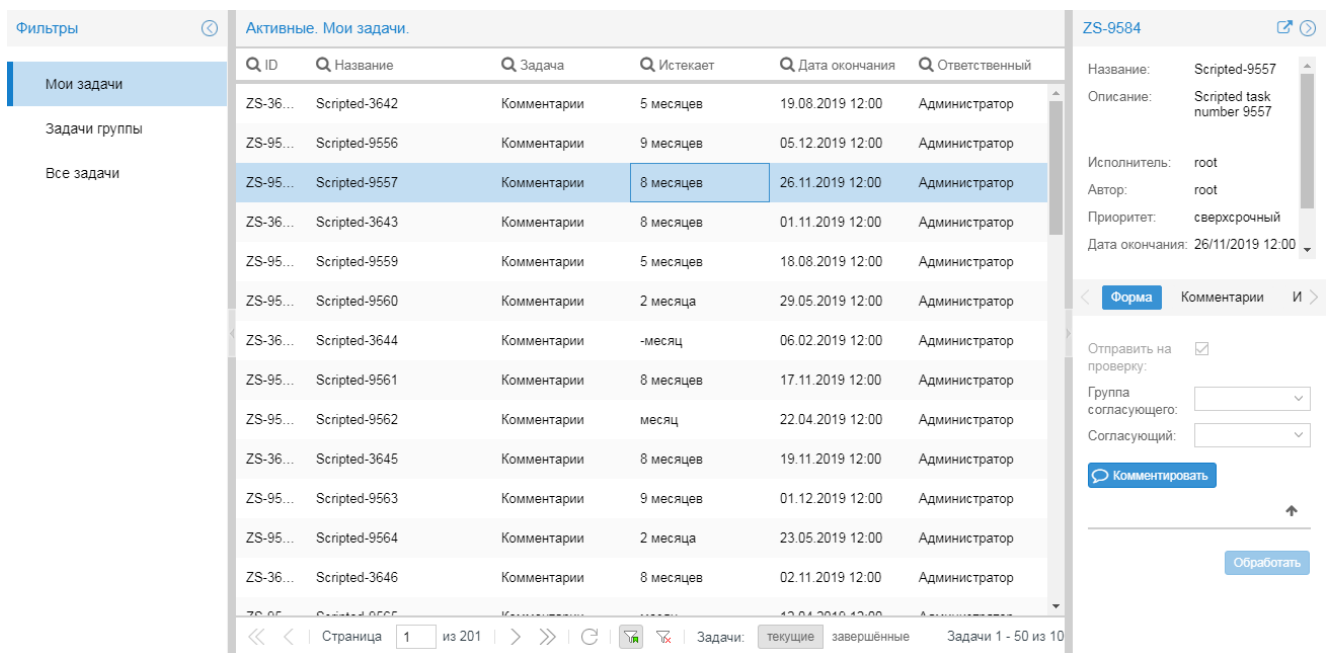
Выход из приложения

Чтобы выйти из приложения **Координатор**, следует в заголовке страницы нажать кнопку  и в появившемся меню выбрать пункт **Выход**. В результате соединение с сервером будет завершено, и откроется страница с приглашением в Web-интерфейс.

12.2. Основное окно

Основное окно приложения **Координатор** состоит из следующих компонентов:

- ❖ [заголовок](#) ²²³;
- ❖ окно [Фильтры](#) ²²⁴;
- ❖ [таблица задач](#) ²²⁵;
- ❖ [окно выполнения бизнес-процесса](#) ²²⁶.



The screenshot displays the main interface of the 'Координатор' application. On the left, there is a sidebar with 'Фильтры' and 'Мои задачи'. The main area contains a table of active tasks. The table has columns for ID, Name, Task, Expires, Completion Date, and Responsible. The task with ID ZS-9557 is highlighted. On the right, a detailed view for task ZS-9584 is shown, including its name, description, executor, author, priority, and completion date. Below the details, there are options to send for review, assign a group, and a 'Обработать' button.

ID	Название	Задача	Истекает	Дата окончания	Ответственный
ZS-36...	Scripted-3642	Комментарии	5 месяцев	19.08.2019 12:00	Администратор
ZS-95...	Scripted-9556	Комментарии	9 месяцев	05.12.2019 12:00	Администратор
ZS-95...	Scripted-9557	Комментарии	8 месяцев	26.11.2019 12:00	Администратор
ZS-36...	Scripted-3643	Комментарии	8 месяцев	01.11.2019 12:00	Администратор
ZS-95...	Scripted-9559	Комментарии	5 месяцев	18.08.2019 12:00	Администратор
ZS-95...	Scripted-9560	Комментарии	2 месяца	29.05.2019 12:00	Администратор
ZS-36...	Scripted-3644	Комментарии	-месяц	06.02.2019 12:00	Администратор
ZS-95...	Scripted-9561	Комментарии	8 месяцев	17.11.2019 12:00	Администратор
ZS-95...	Scripted-9562	Комментарии	месяц	22.04.2019 12:00	Администратор
ZS-36...	Scripted-3645	Комментарии	8 месяцев	19.11.2019 12:00	Администратор
ZS-95...	Scripted-9563	Комментарии	9 месяцев	01.12.2019 12:00	Администратор
ZS-95...	Scripted-9564	Комментарии	2 месяца	23.05.2019 12:00	Администратор
ZS-36...	Scripted-3646	Комментарии	8 месяцев	02.11.2019 12:00	Администратор

12.2.1. Заголовок и главное меню

В левой части заголовка отображаются логотип и название ПМ **Нувола Координатор**.

В правой части заголовка расположены иконки:



– поиск записей в [таблице задач](#) ²²⁵;



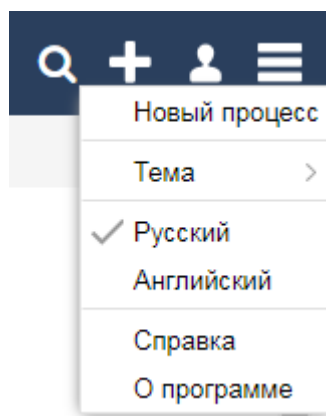
– создание бизнес-процесса на основе схемы;




или – меню, содержащее имя пользователя и команду для выхода из Web-интерфейса; буква А на иконке выводится для администратора root;



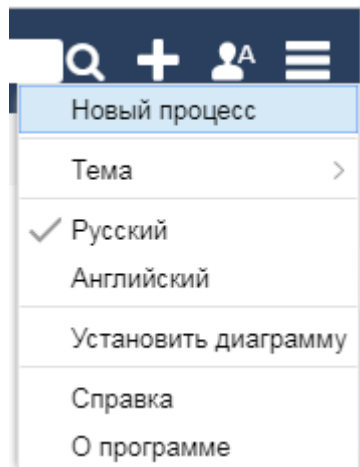
– главное меню.



Состав главного меню зависит от прав пользователя. Конечному пользователю доступны следующие пункты:

- ❖ **Новый процесс** – создание бизнес-процесса на основе схемы (аналог кнопки  в заголовке);
- ❖ **Тема** – установка темы для оформления Web-интерфейса; выбранная тема применяется сразу и сохраняется в cookies браузера;
- ❖ **Русский / Английский** – переключение языка Web-интерфейса (выбор локализации);
- ❖ **Справка** – вызов справки по Web-интерфейсу;
- ❖ **О программе** – просмотр информации о версии ПМ **Нувола Координатор**.



Главное меню администратора root содержит также пункт **Установить диаграмму**, позволяющий [загрузить схему бизнес-процесса](#)²²⁹ на сервер Координатор.



12.2.2. Окно Фильтры

Окно **Фильтры** в левой части основного окна содержит глобальные фильтры для вывода информации в таблицу задач. Пользователь может отбирать записи в зависимости от ответственного исполнителя. Реализованы следующие варианты:

- ❖ **Мои задачи** – задачи, в которых ответственным исполнителем является текущий пользователь;
- ❖ **Задачи группы** – задачи, назначенные на группу, в которой состоит текущий пользователь;
- ❖ **Все задачи** – все задачи.

Окно **Фильтры** можно скрывать / раскрывать с помощью кнопки  /  в заголовке окна или стрелки на рамке окна.

The screenshot shows the main window of the software. On the left, there is a sidebar with filters and task groups. The main area contains a table of tasks with columns for ID, Name, Task, Expires, Completion Date, and Responsible. The task with ID ZS-9557 is highlighted. On the right, a details panel for task ZS-9584 is visible, showing its name, description, executor, author, priority, and completion date. Below the table is a toolbar with navigation and filtering options.

ID	Название	Задача	Истекает	Дата окончания	Ответственный
ZS-36...	Scripted-3642	Комментарии	5 месяцев	19.08.2019 12:00	Администратор
ZS-95...	Scripted-9556	Комментарии	9 месяцев	05.12.2019 12:00	Администратор
ZS-95...	Scripted-9557	Комментарии	8 месяцев	26.11.2019 12:00	Администратор
ZS-36...	Scripted-3643	Комментарии	8 месяцев	01.11.2019 12:00	Администратор
ZS-95...	Scripted-9559	Комментарии	5 месяцев	18.08.2019 12:00	Администратор
ZS-95...	Scripted-9560	Комментарии	2 месяца	29.05.2019 12:00	Администратор
ZS-36...	Scripted-3644	Комментарии	-месяц	06.02.2019 12:00	Администратор
ZS-95...	Scripted-9561	Комментарии	8 месяцев	17.11.2019 12:00	Администратор
ZS-95...	Scripted-9562	Комментарии	месяц	22.04.2019 12:00	Администратор
ZS-36...	Scripted-3645	Комментарии	8 месяцев	19.11.2019 12:00	Администратор
ZS-95...	Scripted-9563	Комментарии	9 месяцев	01.12.2019 12:00	Администратор
ZS-95...	Scripted-9564	Комментарии	2 месяца	23.05.2019 12:00	Администратор
ZS-36...	Scripted-3646	Комментарии	8 месяцев	02.11.2019 12:00	Администратор

12.2.3. Таблица задач

В центре основного окна размещена таблица задач. В заголовке таблицы указывается выбранный режим просмотра (**Активные / Завершённые**) и глобальный фильтр, установленный в окне справа **Фильтры**.

В таблицу загружаются записи о задачах в соответствии с критериями, заданными пользователем. См. Применение фильтров для отбора записей о задачах. Отобранные записи автоматически разбиваются на страницы. Диапазон порядковых номеров записей, выведенных на текущей странице, и общее количество отобранных записей показывается на панели инструментов под таблицей справа

Набор отображаемых столбцов и порядок их расположения в таблице настраивается пользователем. Некоторые столбцы доступны только в режиме **текущие** или только в режиме **завершённые**. Описание столбцов см. в разделе [Параметры задачи](#) ^[226].

Для наглядного выделения просроченных задач и задач с истекающим сроком выполнения используется цветовая маркировка строк. См. [Режимы работы \(текущие / завершённые\)](#) ^[226].

Панель инструментов

The toolbar contains navigation buttons (back, forward), a page number input field (currently 3 of 18), refresh, filter, and reset buttons. It also shows the current task mode (текущие) and the range of tasks displayed (Задачи 101 - 150 из 879).

Панель инструментов под таблицей задач содержит следующие элементы (слева направо):

- ❖ кнопки навигации по таблице; для перехода на конкретную страницу достаточно в поле **Страница** ввести ее номер и нажать клавишу **Enter**;
- ❖ кнопка – обновление данных в таблице в соответствии с текущим фильтром;
- ❖ кнопка – применение фильтра для отбора задач;
- ❖ кнопка – сброс параметров текущего фильтра и возврат к первоначальным настройкам;
- ❖ переключатель **Задачи: текущие / завершённые** устанавливает [режим работы](#) ^[226] с задачами;
- ❖ диапазон отображаемых задач и общее количество задач.

12.2.3.1. Параметры задачи

Ниже описаны поля, доступные в таблице задач. Некоторые столбцы доступны только в одном из режимов: **текущие** или **завершённые**.

ID	Идентификатор бизнес-процесса. Формируется автоматически.
Название	Название бизнес-процесса.
Задача (только в режиме текущие)	Имя задачи.
Истекает (только в режиме текущие)	Период времени, в течение которого задача должна быть выполнена. Период с отрицательным знаком означает, что задача просрочена.
Дата окончания	Плановая дата окончания задачи.
Завершено (только в режиме завершённые)	Реальная дата выполнения задачи.
Просрочено (только в режиме завершённые)	Период, на который выполнение задачи просрочено, например: 5 дней . Период с отрицательным знаком означает, что задача выполнена в срок.
Создано	Дата создания задачи.
Длительность (только в режиме завершённые)	Временной интервал выполнения задачи.
Процесс	Наименование процесса.
Ответственный	Пользователь, ответственный за задачу.

12.2.4. Окно выполнения бизнес-процесса

Окно справа от таблицы задач называется окном выполнения бизнес-процесса. Оно содержит подробную информацию о выбранной задаче и бизнес-процессе, к которому относится задача. В данном окне пользователь может осуществлять выполнение задачи и вводить комментарии по бизнес-процессу.

В заголовке окна указан идентификатор выбранной задачи и кнопки управления окном.

ZS-9
🔗 ↻

Название: Инцидент 6

Описание: Описание

Исполнитель: Администратор

Автор: Администратор

Приоритет:

Дата окончания: ✔ ✖



Событие:




Форма Комментарии История

↓

- ✓ Общие переменные
- ✓ Комментарии: 01.03.19 13:59, 3 дня
- ✓ Утверждение: 04.03.19 09:58, 8 дней
- ✓ Утверждение: 04.03.19 09:58
- ✓ Комментарии: 12.03.19 15:26

Кнопки управления окном

С помощью кнопки  /  (а также стрелки на рамке окна) окно можно скрывать / раскрывать.

По нажатию кнопки  окно становится независимым. В заголовке появляется кнопка , с помощью которой можно раскрыть окно на весь браузер. Для уменьшения окна до исходных размеров следует нажать кнопку .

Для выхода из режима независимого окна следует нажать кнопку .

12. Приложение Координатор

The screenshot shows a window titled 'ZS-9' with a close button. The main content area displays the following details:

- Название: Инцидент 6
- Описание: Описание
- Исполнитель: User 1
- Автор: Администратор
- Приоритет: низкий
- Дата окончания: 02/03/2019 13:59
- Событие:

Below the details is a tabbed interface with three tabs: 'Форма', 'Комментарии', and 'История'. The 'История' tab is active. It contains two expandable sections:

Общие переменные

Время	Пользователь	Поле	Исходное значение
11.03.2019 16:50	Администратор	reporter	user1
12.03.2019 15:35	User 1	описание	Описание инцидента
12.03.2019 15:35	User 1	описание	Описание инцидента
12.03.2019 15:36	User 1	приоритет	5
20.03.2019 15:31	Администратор	reporter	user10

Комментарии: 01.03.19 13:59, 3 дня

Время	Пользователь	Поле	Исходное значение
01.03.2019 14:00	User 1	исполнитель	
04.03.2019 9:58	User 1	задача завершена	нет
04.03.2019 9:58	User 1	задача завершена	нет

Утверждение: 04.03.19 09:58, 8 дней

Утверждение: 04.03.19 09:58

A 'Закрыть' button is located at the bottom right of the window.

Структура окна

В верхней части окна отображаются атрибуты бизнес-процесса: **Название** и **Описание**.

В нижней части размещены вкладки:

- ❖ **Форма** – параметры задачи; набор полей и кнопок формируется в соответствии со схемой процесса; ; вкладка **Форма** не формируется, если в таблице задач включен режим **завершённые**;
- ❖ **Комментарии** – комментарии по процессу;;
- ❖ **История** – журнал операций, выполненных в ходе бизнес-процесса. В верхнем блоке **Общие параметры** отображается история изменения параметров бизнес-процесса, в нижних блоках – задачи, выполненные пользователями в рамках данного бизнес-процесса.

12.2.5. Режимы работы (текущие / завершённые)

В [таблице задач](#)²²⁵ реализованы два режима работы с задачами: **текущие** и **завершённые**. Для перехода из одного режима в другой предназначен переключатель **Задачи** на панели инструментов под таблицей. Набор доступных столбцов таблицы задач зависит от выбранного режима.

The screenshot shows a web application interface for task management. On the left, there is a sidebar with filters: 'Мои задачи', 'Задачи группы', and 'Все задачи'. The main area displays a table of tasks under the heading 'Активные. Все задачи'. The table has columns for ID, Name, Task, Expires, Completion Date, and Responsible. The first three rows are highlighted in red, and the fourth row (ID I-6, Name Тест 3) is highlighted in yellow. Below the table, there is a pagination bar showing 'Страница 1 из 1' and a task status filter set to 'текущие'. On the right, a detailed view for task 'I-6' is shown, including fields for Name, Description, Assignee, Author, Priority, and Completion Date. Below these fields are tabs for 'Форма', 'Комментарии', and 'История'. The 'Комментарии' tab is active, showing a form to add a comment and a list of existing comments.

ID	Название	Задача	Истекает ↑	Дата окончания	Ответственный
I-1	I-345	Комментарии	-6 дней	26.11.2019 10:09	Operator 1
I-3	I-34523	Комментарии	-6 дней	26.11.2019 12:57	Администратор
I-4	I-5343	Комментарии	-3 дня	29.11.2019 11:29	Администратор
I-6	Тест 3	Комментарии	44 минуты	02.12.2019 18:15	Оператор

В режиме **текущие** в таблице отображаются задачи, которые еще не выполнены. Жёлтым цветом выделяются задачи, у которых срок выполнения (поле **Дата окончания**) истекает менее, чем через час; красным цветом – просроченные задачи (дата в поле **Завершено** позднее, чем **Дата окончания**).

В этом режиме пользователю доступны следующие действия:

- ❖ выполнение задачи;
- ❖ ввод комментария по процессу;
- ❖ изменение параметров процесса.

В режиме **завершённые** в таблице отображаются уже выполненные задачи. Красным цветом выделяются просроченные задачи. В этом режиме информация в таблице задач, как и в окне выполнения бизнес-процесса, доступна только для просмотра.

Примечание. Цвета для маркировки просроченных задач и задач с истекающим сроком, а также период маркировки задач с истекающим сроком выполнения, могут быть изменены в конфигурационном файле. Выше приведены стандартные настройки.

12.3. Действия администратора

В настоящем разделе рассматриваются действия, доступные администратору root в приложении Координатор.

12.3.1. Загрузка схемы бизнес-процесса на сервер Координатор

Администратор создает схемы бизнес-процессов в приложении Camunda Modeler и сохраняет их на своей рабочей станции или доступном сетевом ресурсе. Приложение **Координатор** позволяет администратору root загрузить готовую схему процесса на сервер Координатор.

Чтобы загрузить схему процесса на сервер Координатор, следует:

12. Приложение Координатор

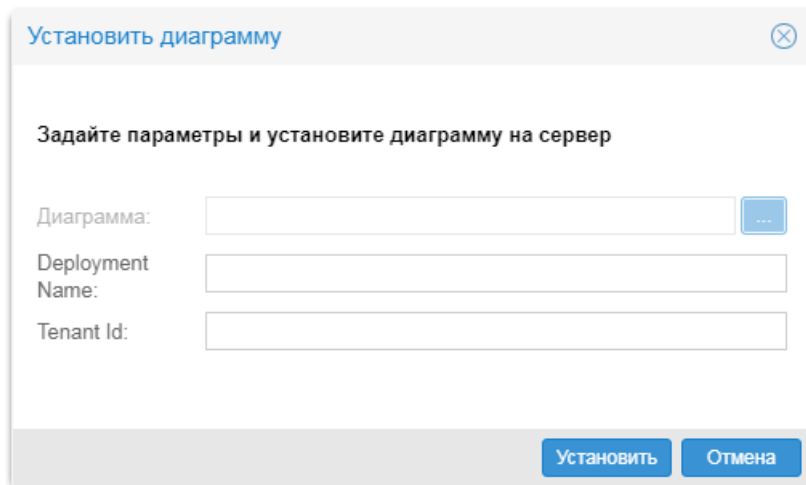
1. в главном меню выбрать пункт **Установить диаграмму**;
2. в появившемся окне заполнить поля:

Диаграмма – файл схемы; выбрать с помощью кнопки 

Deployment Name;


Tenant ID;

3. нажать кнопку **Установить**.



Установить диаграмму

Задайте параметры и установите диаграмму на сервер

Диаграмма: 

Deployment Name:

Tenant Id:

Установить Отмена

13.

Настройка и использование модуля Нувола Аналитика

13.1. Запуск сервера Нуволы Аналитика

Запуск сервера Нуволы Аналитика

В штатном режиме сервер **Нуволы Аналитика** запускается вместе с операционной системой посредством командного сценария `/etc/init.d/sybilla`. Возможен запуск сервера **Нуволы Аналитика** в ручном режиме с помощью команды:

```
$ /usr/local/sybilla/bin/sybilla-service [options]
```

Аргументы `[options]` приведены ниже.

`--help`

Вывод справки по опциям командной строки.

`--version`

Вывод номера версии.

`--object-name=<имя объекта CORBA>`

Указание имени объекта CORBA.

`--nameserver=<спецификатор сервиса имен>`

Указание спецификатора сервиса имен. По умолчанию используется `NameService=corbaname::127.0.0.1:2809`

`--chdir=<рабочая директория>`

Указание рабочей директории программы. В эту директорию будут записываться протоколы работы сервера предсказаний **Нуволы Аналитика**. По умолчанию используется `/usr/local/sybilla`.

`--daemonize`

Запуск приложения в режиме сервиса.

`--log-file=<префикс>`

Этот параметр задает постоянный префикс имени файла протокола, сервер автоматически дополняет имя файла датой и временем его создания.

`--log-level=<2-5>`

Настройка уровня детализации протоколов. Возможные значения от **2** до **5**, где **2** – записывать только ошибки, **5** – записывать полную отладочную информацию.

`--debug-memory`

Запись в лог-файл ошибок выделения памяти.

`--debug-dump-core`

Сохранение образа адресного пространства процесса («дампа») в случае аварийного завершения работы.

`--debug-channels=<RegExp>`

Регулярное выражение `RegExp` задаёт имена каналов отладочной информации.

`--username=<user>`

Смена EUID на EUID указанного пользователя в режиме сервиса.

Реализация отладочной информации

Отладочная печать реализована таким образом, что каждая выводимая строка может быть озаглавлена некоторым именем. Имена выбраны разработчиком исходя из внутренней логики выводимого сообщения. Например: «Сообщение о выделении памяти», «Сообщение математической библиотеки» и т.д. Использовать режим отладочного протоколирования следует только по рекомендации разработчика для передачи ему полученных файлов протокола.

Имеющийся список не полон, разработчик оставляет за собой право изменить его в последующих версиях без предварительного уведомления. В текущей версии реализованы следующие имена каналов:


```
svrtk_thread_pool  
svrtk_alloc  
svrtk_dump  
svrtk_hash  
svrtk_service_queue  
svrtk_hugedb  
svrtk_svc  
svrtk_main  
svrtk_daemon  
svrtk_bstack
```

Для дальнейшего использования зарезервированы следующие имена каналов:

```
SYBLOGCTRL  
SYBNOCHANNEL  
SYBPROCIO  
SYBEXTIO  
SYBMEM  
SYBMATH  
SYBYO  
SYBDEBUG
```

13.2. Утилита командной строки

В процессе установки утилита командной строки копируется в каталог `/usr/local/sybilla/bin`.

Запуск утилиты командной строки осуществляется командой:

```
$ /usr/local/sybilla/bin/sybillacl
    [-p <params_file>]
    [-o <serverObject>]
    [-n <corbaNameServerSpec>]
    [-r <repeat_count>]
    <command> <command args>
```

Аргументы для утилиты командной строки

`-p params_file`

Файл с параметрами, необходимыми для указанного в команде типа расчета.

`-o serverObject`

Имя объекта CORBA, под которым зарегистрирован сервер (необязательно в случае, если клиент и сервер **Нуволы Аналитика** запущены на одной машине).

`-n corbaNameServerSpec`

Спецификатор адреса сервиса имен CORBA в формате `NameService=corbaname::<IP>:<Port>` (необязательно, если клиент и сервер **Нуволы Аналитика** запущены на одной машине).

`-r repeat_count`

Счетчик повторений, задает количество повторов команды `<command>`. По умолчанию равен 1.

Примечание. Параметр `-r` не используется для команд `version` и `echo`.

`command <command args>`

Команда с аргументами.

Команды, выполняемые утилитой командной строки

`version`

Вывод номера версии сервера **Нуволы Аналитика**, к которому подключается утилита.

`echo <input_vector> <output_vector>`

Передать на сервер входной вектор из внешнего файла `<input_vector>` и сохранить результат в файле с именем `<output_vector>`.

`predict <input_vector> <predicted_vectot> <treshholds_vector>`

Построить продолжение вектора, заданного в файле `<input_vector>`; сохранить вычисленный результат в файле с именем `<output_vector>`, сохранить вычисленный доверительный интервал в файле `<treshholds_vector>`.

`proximity <input1> <input2>`

Сравнить два входных вектора, заданных в файлах `<input1>` и `<input2>`, распечатать результат на консоль.

`anomalies <input_vector> <anom1> <anom2>`

Проверить исходный вектор, заданный в файле `<input_vector>`, на наличие аномалий; сохранить результаты в файлах `<anom1>` и `<anom2>`. В файле `<anom1>` сохраняются все значения, аномальные по амплитуде, а в файле `<anom2>` – значения, «особо аномальные» с точки зрения алгоритма.

13.3. Настройка сервера Нувола Аналитика

Сервис **Нувола Аналитика** для запуска использует сервис `xinetd`. Конфигурация сервера сохранена в `/etc/xinetd.d/Sybilla` и выглядит следующим образом:

```
Service sybilla
{
socket_type = stream
user = avalon
server = /usr/local/sybilla/bin/sybilla-service
wait = no
server_args = --log-level=5 --debug-channels=syb|deb
--chdir=/usr/local/sybilla/logs --debug-dump-core
log_on_failure += USERID
port = 1933
type = UNLISTED
protocol = tcp
env = LD_LIBRARY_PATH=/usr/local/sybilla/lib
}
```

Важные параметры:

- `User` – пользователь для запуска процесса;
- `Server` – исполняемый файл;
- `Server-args` – аргументы модуля **Нувола Аналитика**;
- `Port` – порт для соединения;
- `Env` – технологическое оборудование.

13.4. Настройка сигналов

Настройка клиента модуля **Нувола Аналитика** заключается в настройке сигналов типа **Trend** или **TrendDeviation**.

Сигналы типа **Trend** и **TrendDeviation** позволяют предсказать тенденцию поведения сигнала, получаемого от целевого объекта, и определить доверительные интервалы (динамические границы) для будущих значений сигнала. Доверительные интервалы определяются в зависимости от поведения сигнала в соответствии с заданными параметрами. Сигнал типа **Trend** показывает прогнозируемую тенденцию изменения значения показателя, получаемого от целевого объекта, и позволяет определить доверительные интервалы (динамические границы) для будущих значений показателя. Доверительные интервалы определяются в зависимости от поведения сигнала в соответствии с заданными параметрами. Сигнал типа **TrendDeviation** показывает отклонение реально полученного значения от предсказанной величины (в процентах от предсказанной величины). Эти методы предсказания довольно грубые, однако они позволяют спрогнозировать периодические (сезонные) всплески сигнала. Преимуществами этих методов по сравнению с другими методами предсказаний являются экономия вычислительных ресурсов и быстродействие.

Чтобы настроить прогнозирование поведения какого-либо показателя, получаемого от целевого объекта, следует:

1. в [Редакторе сигналов и оповещений](#)^[117] создать набор сигналов, состоящий из одного сигнала типа **Trend** или **TrendDeviation**;
2. привязать набор сигналов к целевому объекту.

Если к целевому объекту уже привязан набор из сигналов типа **Availability**, **Gauge** и т.п., то рекомендуется в дереве конфигурации создать ссылку на целевой объект и к ней привязать набор сигналов, включающий в себя один сигнал типа **Trend** или **TrendDeviation**.

Результат прогнозирования отображается на графике на web-странице целевого объекта.

13.5. Сигнал типа Trend

Чтобы настроить прогнозирование поведения параметра, получаемого от целевого объекта, следует:

1. в приложении [Редактор сигналов и оповещений](#)^[117] создать набор сигналов, состоящий из сигналов типа [Trend](#)^[236];
2. в окне [Дерево конфигурации](#) [привязать набор сигналов](#)^[134] к целевому объекту.

Если к целевому объекту уже привязан набор из сигналов типа **Availability**, **Gauge** и т.п., то рекомендуется в дереве конфигурации создать ссылку на целевой объект и к ней привязать набор сигналов, включающий в себя один сигнал типа **Trend**.

Результат прогнозирования отображается на графике на web-странице целевого объекта.

13.5.1. Параметры сигнала

Сигнал типа **Trend** показывает прогнозируемую тенденцию изменения значения показателя, получаемого от целевого объекта, и позволяет определить доверительные интервалы (динамические границы) для будущих значений показателя. Доверительные интервалы определяются в зависимости от поведения сигнала в соответствии с заданными параметрами.

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

Источник данных

Имя источника данных в терминологии SNMP.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (*avg*) – выбирает из базы данных значения за интервал **Диапазон** (*range*) и вычисляет среднее значение;
- ❖ **минимальное** (*min*) – выбирает из базы данных минимальное значение за интервал **Диапазон** (*range*);
- ❖ **максимальное** (*max*) – выбирает из базы данных максимальное значение за интервал **Диапазон** (*range*);
- ❖ **последнее** (*last*) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (*range*). Это значение может быть только числом;
- ❖ **текущее** (*cur*) – выбирает из базы данных последнее значение за интервал **Диапазон** (*range*). Это значение не обязательно будет числом;

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция *last* возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция *cur* возвращает значение **NAN**;

- ❖ **последнее известное** (`ldef` – от last defined) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (`range`). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

Диапазон (`range`)

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Диапазон расчёта тренда

Интервал времени (в секундах), для которого осуществляется прогноз. По умолчанию **604800** (неделя).

Показать линию тренда

Отображение линии с прогнозируемыми значениями на графике исследуемого параметра на странице целевого объекта. Возможные значения: **да**, **нет**.

В следующих поля вводятся краткие описания диапазонов значений исследуемого параметра. Введённый текст отображается в столбце **Метка** в представлении **Консоль событий**.

Метка для зеленого коридора

Описание диапазона значений, ограниченного зелёным коридором.

Метка для жёлтого коридора

Описание диапазона значений, ограниченного жёлтым коридором.

Метка для красного коридора

Описание диапазона значений, ограниченного красным коридором.

13.6. Сигнал типа TrendDeviation

Сигнал типа **TrendDeviation** показывает отклонение реально полученного значения от предсказанной величины (в процентах от предсказанной величины).

13.6.1. Параметры сигнала

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Источник данных

Имя источника данных в терминологии SNMP. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (`avg`) – выбирает из базы данных значения за интервал **Диапазон** (`range`) и вычисляет среднее значение;

13. Настройка и использование модуля Нувола Аналитика

- ❖ **минимальное** (*min*) – выбирает из базы данных минимальное значение за интервал **Диапазон** (*range*);
- ❖ **максимальное** (*max*) – выбирает из базы данных максимальное значение за интервал **Диапазон** (*range*);
- ❖ **последнее** (*last*) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (*range*). Это значение может быть только числом;
- ❖ **текущее** (*cur*) – выбирает из базы данных последнее значение за интервал **Диапазон** (*range*). Это значение не обязательно будет числом;
Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция *last* возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция *cur* возвращает значение **NAN**;
- ❖ **последнее известное** (*ldef* – от last defined) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (*range*). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

Диапазон

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

Режим работы

Режим работы системы расчёта трендов. Могут использоваться следующие режимы:

- ❖ **Average base line** – среднее арифметическое. Производится расчёт среднего значения за интервал **Диапазон расчёта тренда** и отображение его на графике в виде горизонтальной линии;
- ❖ **Linear trend** – линейное приближение. Производится расчёт линейного приближения к значениям, входящим в интервал **Диапазон расчёта тренда**, и отображение результата на графике в виде наклонной прямой;
- ❖ **Sliding window trend** – усреднение показателя методом скользящего окна. Производится расчёт средних значений в каждой точке некоторого диапазона значений, задаваемого параметром **Sliding Window**, для всего интервала скольжения; результат отображается на графике в виде кривой.

Нижний порог

Пороговое значение между диапазонами синего и зелёного статусов сигнала. Если значение сигнала меньше **Нижнего порога**, то сигналу присваивается синий статус, если больше или равно – зелёный.

Нижний порог задётся в абсолютных единицах. Допустимы приставки k, m, g. По умолчанию – 0. Как правило, **Нижний порог** не задают.

Нормальный порог

Пороговое значение между диапазонами зелёного и жёлтого статусов сигнала. Если значение сигнала меньше **Нормального порога**, то сигналу присваивается зелёный статус, если больше или равно – жёлтый.

Нормальный порог задаётся в абсолютных единицах или процентах от **Максимального порога**. Допустимы приставки k, m, g. Обязательный параметр.

Высокий порог

Пороговое значение между диапазонами красного и бордового статусов сигнала. Если значение сигнала меньше **Максимального порога**, то сигналу присваивается красный статус, если больше или равно – бордовый.

Максимальный порог задаётся в абсолютных единицах. Допустимы приставки k, m, g. По умолчанию равен значению параметра Bandwidth (пропускная способность) целевого объекта. Если параметр Bandwidth не определен, то по умолчанию **Максимальный порог** принимается равным 100. Как правило, **Максимальный порог** не задают.

Максимальный порог

Пороговое значение между диапазонами жёлтого и красного статусов сигнала. Если значение сигнала меньше **Высокого порога**, то сигналу присваивается жёлтый статус, если больше или равно – красный.

Высокий порог задаётся в абсолютных единицах или процентах от **Максимального порога**. Допустимы приставки k, m, g. Обязательный параметр.

Диапазон расчёта тренда*

Временной интервал (в секундах), используемый для вычисления тренда. По умолчанию **604800** (неделя).

Показать линию тренда

Отображение вычисленной линии тренда на графике целевого объекта. Возможные значения: **да**, **нет**.

Интервал обучения

Элементарный интервал обучения (в секундах). Данный параметр используется только в режиме **Sliding window trend**. По умолчанию **86700** (сутки).

Число интервалов обучения

Число элементарных интервалов обучения. Данный параметр используется только в режиме **Sliding window trend**. По умолчанию **7**.

Размер скользящего окна

Размер скользящего окна (в секундах). Данный параметр используется только в режиме **Sliding window trend**. По умолчанию **900** (15 минут).

Границы доверительного интервала вычисляются по следующим формулам:

$$\text{upper_threshold} = \text{trend} + \text{threshold_pct_factor}(\%) + \text{threshold_value}$$

$$\text{lower_threshold} = \text{trend} - \text{threshold_pct_factor}(\%) - \text{threshold_value}$$

где

upper_threshold и lower_threshold – верхний и нижний пороги доверительного интервала, соответственно;

trend – вычисленное значение тренда;

threshold_pct_factor(%) – значение параметра **Относительная погрешность (в процентах)**;

threshold_value – значение параметра **Абсолютная погрешность**.

13. Настройка и использование модуля Нувол Аналитика

▼ 🔔 trend-dev	
Имя	
Тип	TrendDeviation
Описание	
Источник данных	emc-ps-health
Функция	максимальное
Диапазон	600
Режим работы	Average base line
Нижний порог	0
Нормальный порог	60%
Высокий порог	80%
Максимальный порог	100%
Метка для нижнего порога	
Метка для нормального порога	
Метка для высокого порога	
Метка для максимального порога	
Метка выше максимального порога	
Диапазон расчета тренда	604800
Показать линию тренда	1
Интервал обучения	86400
Число интервалов обучения	7
Размер скользящего окна	900
Условные параметры	
extend_range	86400

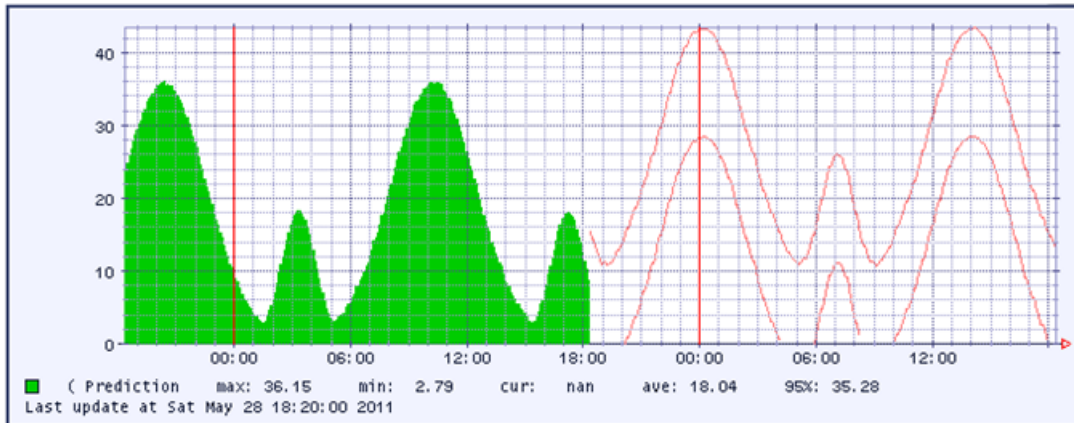
13.7. Графики при использовании модуля Нувол Аналитика

При использовании модуля **Нувол Аналитика** у целевых объектов, к которым привязаны соответствующие наборы сигналов, изменяется внешний вид графиков и могут появиться дополнительные параметры сигналов.

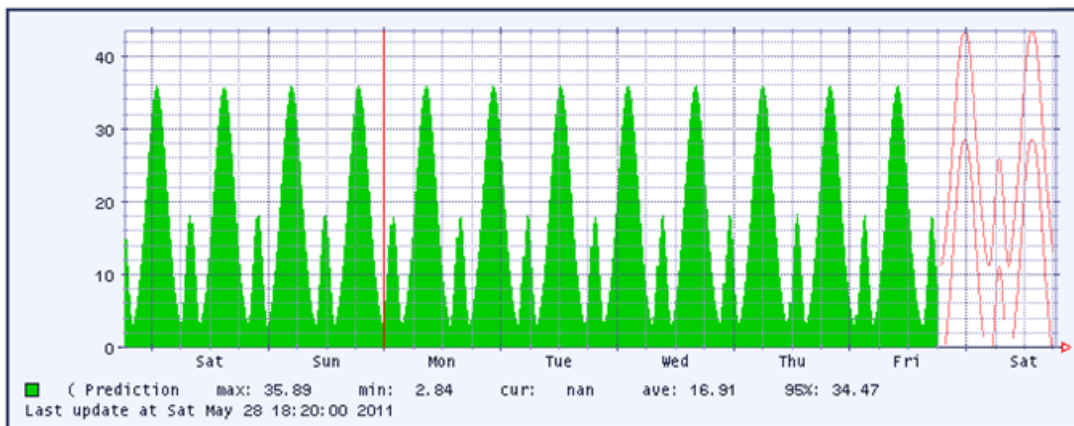
13.7.1. Вычисление предполагаемого диапазона данных

В режимах `Simple Thresholds` и `Delta Thresholds` на основании полученных данных модуль **Нувол Аналитика** вычисляет диапазон, в который данные попадут с наибольшей вероятностью. Верхняя и нижняя границы предполагаемого диапазона данных называются динамическими порогами. Динамические пороги отображаются на графиках в виде тонких красных линий.

Часовой



Дневной



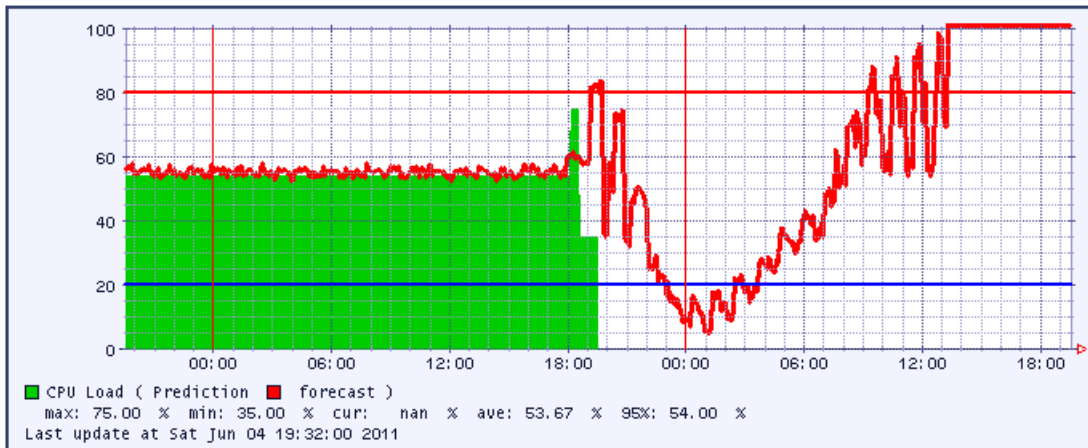
13.7.2. Предсказание перехода сигнала за определённый порог

В режиме Capacity Planning модуль **Нувола Аналитика** вычисляет и отображает период времени до того момента, когда значение прогнозируемого сигнала пересечёт определенные пользователем пороги. На графике верхний порог отображается горизонтальной красной жирной линией, нижний порог – горизонтальной синей жирной линией.

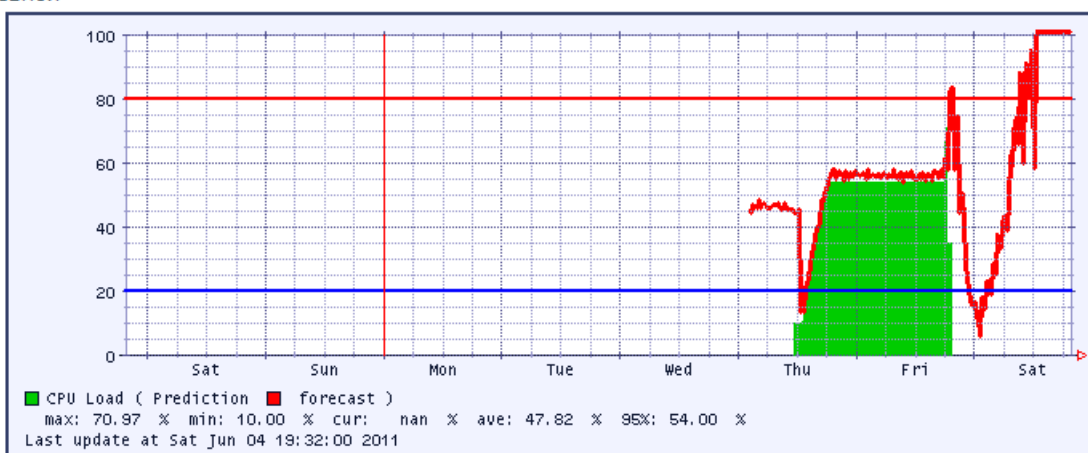
Вычисленный период пересечения предсказанного сигнала и какого-либо порога отображается в поле **Описание** в окне шаблона сигнала. Обычно пересечения нет, и в этом поле стоит знак бесконечности. Ниже показаны скриншоты случая, когда пересечение есть, и в поле **Описание** при этом указано конечное время до пересечения.

13. Настройка и использование модуля Нуволы Аналитика

Часовой



Дневной

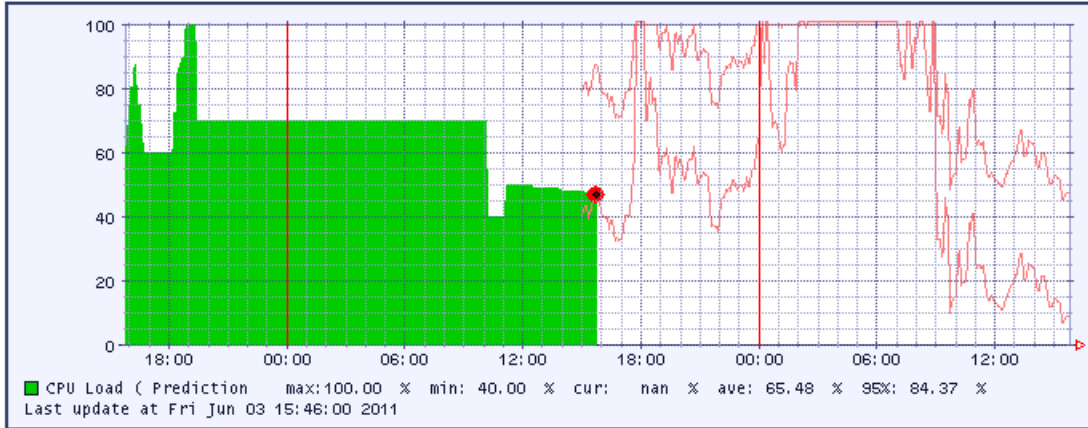


13.7.3. Обнаружение отклонений в реальном времени

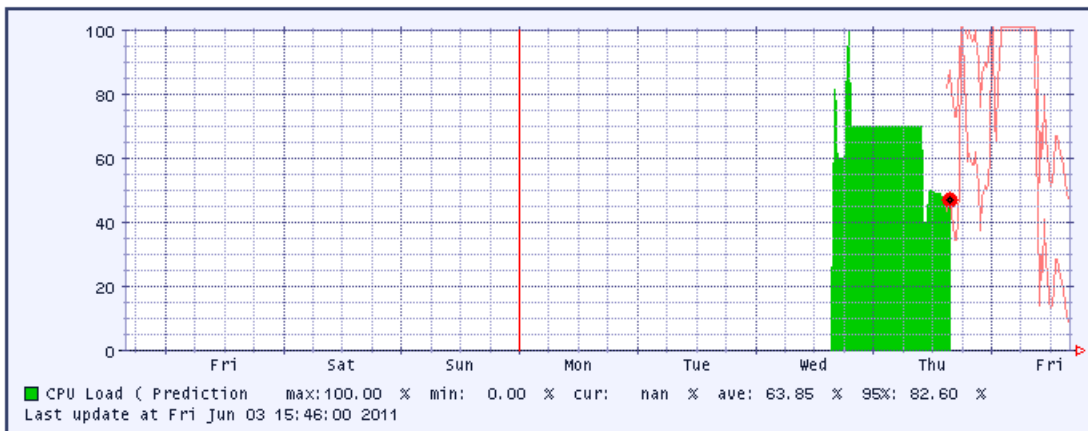
В режиме *RT Anomalies* модуль **Нуволы Аналитика** обеспечивает обнаружение отклонений в реальном времени, основываясь на сделанном прогнозе. После выполнения прогноза и сбора новых реальных данных на их основе осуществляется обнаружение отклонений в реальном времени. Когда новые данные выходят за границы динамических порогов, эта точка передается ядру **Нуволы Аналитика**, которое решает, является она аномалией, или нет. На графиках выявленные аномалии отображаются в виде красных точек.

13.7. Графики при использовании модуля Нувола Аналитика

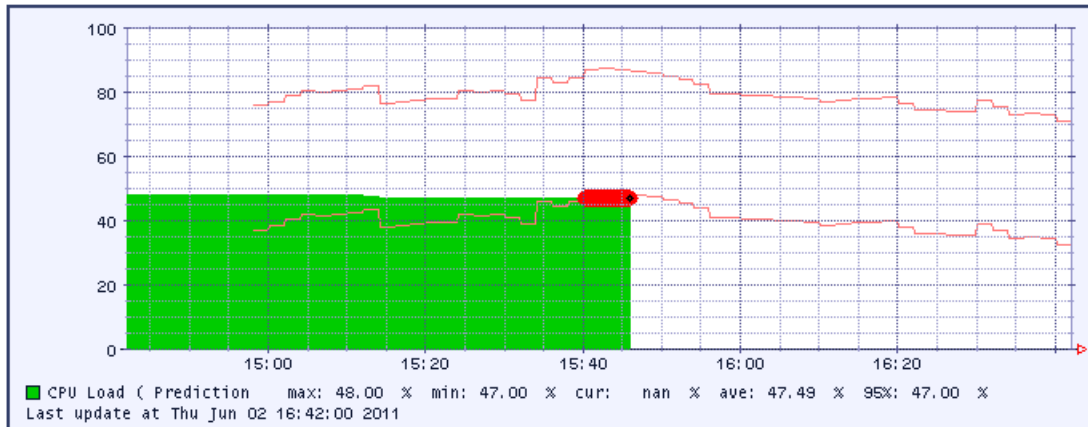
Часовой



Дневной

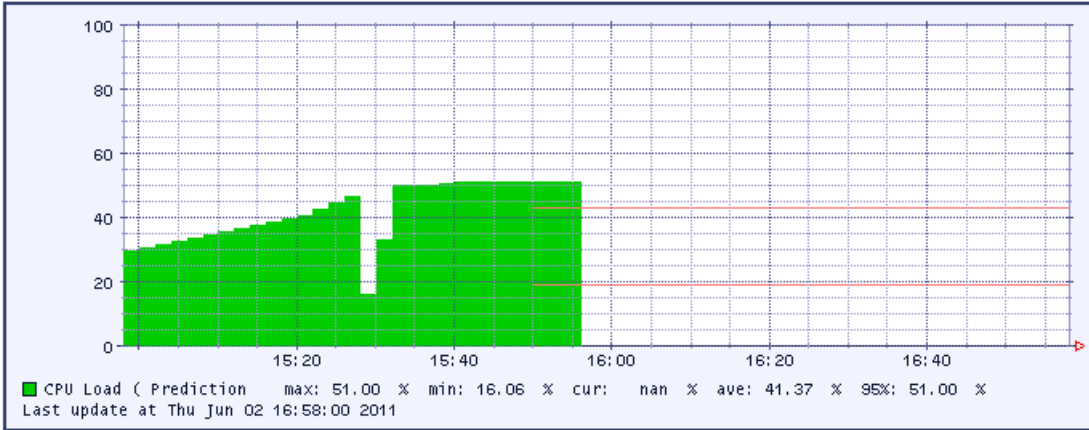


Jun 2 14:42:00 2018 - Jun 2 16:42:00 2018

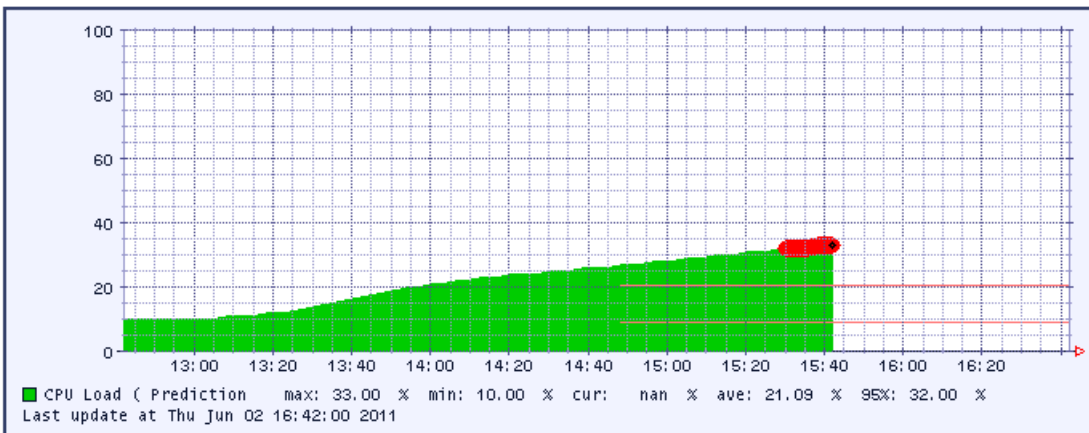


13. Настройка и использование модуля Нувол Аналитика

Jun 2 14:58:00 2018 - Jun 2 16:58:00 2018



Jun 2 12:43:00 2018 - Thu Jun 2 16:43:00 2018



14.

Справочники

14.1. Термины и определения

В данном разделе в алфавитном порядке приведены используемые в документе термины, определения и сокращения.

Camunda – платформа для автоматизации бизнес-процессов, позволяющая создавать модели бизнес-процессов и контролировать их выполнение.

RRD (Round Robin Database) – циклическая база данных. Объем хранимых данных не увеличивается со временем (ячейки хранения используются циклически).

Администратор координирует действия пользователей, осуществляет общую настройку ПК Нувола и обеспечивает ее штатное функционирование. Администратор работает под учетной записью `root` и обладает максимальными правами.

База данных конфигурации включает в себя дерево конфигурации, шаблоны объектов, настройки сигналов и оповещений, информацию о системных процессах ПМ Нувола Топаз.

База данных производительности – база данных, обеспечивающая хранение и отображение динамических параметров, полученных от объектов контроля.

Бизнес-процесс – это совокупность взаимосвязанных мероприятий или задач, направленных на создание определенного продукта или услуги для потребителей.

Виджет (widget) – это сложный графический элемент, отображающий состояние объекта контроля или его компонентов в специфичной форме, например в виде измерительного прибора, лампочки, диаграммы. Для каждой категории виджетов, помимо стандартных настроек, предусмотрены специфические параметры.

Виртуальный объект выполняет вспомогательные функции, необходимые для организации мониторинга.

Датчик сервиса – функция, которая без подключения к устройству по SNMP контролирует работу одного из стандартных сервисов (SSH, DNS, SQL, pcAnywhere, Ping, VNC, SMTP, RADIUS, HTTP, FTP, POP3) или произвольного TCP-порта и в структуре устройства представляет собой целевой объект.

Дерево конфигурации – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления.

Приложение **Диспетчер процессов** – компонента Web-интерфейса, позволяющая администратору посредством графического пользовательского интерфейса управлять системными процессами, функционирующими на сервере Нувола и обеспечивающими работу ПМ Нувола Топаз и других модулей ПК Нувола.

Значение сигнала – числовое значение, полученное в результате обработки значений параметров целевых объектов за определенный интервал времени (например: среднее значение, максимальное или минимальное значение). Значение сигнала используется для вычисления статуса сигнала.

Информационная панель (Dashboard) – конфигурационный инструмент, обеспечивающий в режиме реального времени наглядное графическое web-представление информации о состоянии объектов контроля, данных об обобщенном состоянии инфраструктуры предприятия или ее сегмента, а также предоставляемых им услуг.

Источник данных представляет собой характеристику целевого объекта.

Каталог шаблонов объектов контроля – библиотека преднастроенных шаблонов объектов, или модулей информационно-технического взаимодействия.

Класс оповещений – совокупность параметров, определяющих условия, при возникновении которых формируются оповещения.

Коллектор – обобщённое название системных процессов, основанных на процессе `collector`, осуществляющих сбор данных с объектов контроля и сохранение полученной информации в базе данных.

Комплекс внутренней диагностики – совокупность предустановленных объектов, предоставляющих информацию о функционировании ПМ **Нувола Топаз**: состояния и статистику системных процессов, состояния основных портов, состояния компонент сервера, на котором установлен ПМ **Нувола Топаз**.

Конечный пользователь ПК Нувола – пользовательская роль, позволяющая получать информацию о состоянии объектов контроля в наглядной форме посредством Web-интерфейса. Конечный пользователь обладает минимальным набором прав.

Консоль событий – графический пользовательский интерфейс, отображающий в виде таблицы информацию о состоянии объектов контроля.

Конфигурационный инструмент – компонента дерева конфигурации, посредством которого осуществляется структурирование и хранение информации об объектах контроля, визуализация результатов обработки данных.

Лидер (Leader) – признак, свидетельствующий о повышенном приоритете сигнала в наборе сигналов. Признак является параметром сигналов типов `Gauge`, `InvertGauge`, `SmartGauge` и `Availability`.

Локальный администратор – пользовательская роль, позволяющая конфигурировать в ПК Нувола объекты контроля в рамках своей зоны ответственности (сегмента инфраструктуры), а также осуществлять настройку вывода информации о состоянии объектов контроля (обобщенного состояния инфраструктуры и пр.) в наглядной форме.

Метод оповещений – совокупность параметров, определяющих способ формирования оповещений. Метод оповещений содержит ключевой параметр **тип оповещения** и параметры, конкретизирующие выполняемое действие, например: тип оповещения «отправка сообщения по электронной почте», адреса получателей сообщений, сетевой адрес почтового сервера.

Набор оповещений – совокупность класса и метода (или нескольких методов) оповещений. Набор оповещений определяет, при каких условиях должно выполняться то или иное действие.

14. Справочники

Набор сигналов – совокупность сигналов, которые применимы к источникам данных, принадлежащим целевым объектам определенного типа. Количество сигналов в наборе ограничивается количеством источников данных, поддерживаемых ПК Нувола для выбранного типа целевого объекта.

Назначение (параметр элемента информационной панели) – привязка к сигналам, влияющим на поведение элемента, например: изменение цвета заливки элемента в зависимости от статусов сигналов, показания шкалы (на виджетах) и т.п. **Единичное назначение** – привязка к одному сигналу. **Множественное назначение** – совокупность привязок к нескольким сигналам.

Объект дерева конфигурации – компонента дерева конфигурации, созданная администратором или локальным администратором на основе *шаблона объекта* и выполняющая конкретную задачу в реализации функционального назначения ПК Нувола.

Объект контроля – любая компонента или другой сервисный актив, который необходимо контролировать для того, чтобы предоставлять ИТ-услугу и/или услугу мониторинга.

Оповещение – это уведомление (пользователей) о произошедшем событии (например, сбое, изменении статуса параметра объекта контроля) и /или фиксирование таких фактов в системе (например, логирование).

Проектное решение – совокупность программных модулей, созданная на базе ПК Нувола с учетом особенностей информационной среды заказчика и настроенная для решения согласованного круга задач в рамках определенного договора (проекта).

Приложение **Редактор сигналов и оповещений** – компонента Web-интерфейса. Позволяет администратору настраивать наборы сигналов и наборы оповещений.

Сервер Координатор – установленная платформа Camunda BPM с настроенным для нее интерфейсом REST Camunda API для взаимодействия с компонентами ПМ **Нувола Координатор**.

Сервер Нувола – сервер, на котором установлен ПМ **Нувола Топаз**. На этот сервер (в соответствии с согласованными техническими условиями проектного решения) также могут быть установлены серверные компоненты других программных

14.2. Утилиты ПМ Нувола Топаз

В настоящем разделе описан синтаксис и примеры использования утилит, доступных администратору `root` через командную строку. Если не указано особо, утилиты рекомендуется выполнять от имени пользователя `avalon`.

14.2.1. Настройка сигналов (`alertsedit.pl`)

Утилита `alertsedit.pl` предназначена для настройки наборов сигналов и прикрепления их к целевым объектам.

Эта утилита позволяет создавать и удалять наборы сигналов, а также настраивать уже существующие наборы сигналов. С помощью этой утилиты можно выполнять следующие действия:

- ❖ переименование набора сигналов;
- ❖ копирование набора сигналов в новый;

- ❖ добавление к набору сигналов нового сигнала;
- ❖ привязка набора сигналов к целевым объектам заданного поддерева;
- ❖ отсоединение набора сигналов от целевых объектов заданного поддерева;
- ❖ вывод данных о настройках набора сигналов или параметров набора сигналов;
- ❖ настройка параметров набора сигналов или сигнала.

Синтаксис

```
alertsedit.pl [OBJECT] [COMMAND] [OPTIONS]
```

[OBJECT]

Набор сигналов или сигнал, к которому применяется команда.

```
<templatename>
```

Имя набора сигналов.

```
<templatename>.<alertname>
```

Сигнал с именем `alertname` в составе набора сигналов с именем `templatename`.

```
all
```

Все наборы сигналов.

[COMMAND]

Выполняемое действие.

```
create <target>
```

Создание нового набора сигналов.

`<target>` – путь на целевой объект для определения `target-type` создаваемого набора сигналов (т.е. набор сигналов будет иметь тот же `target-type`, что и заданный целевой объект).

```
remove
```

Удаление набора сигналов. Эта операция отключает обработку сигналов для целевых объектов, которые были прикреплены к этому набору сигналов.

```
rename <newname>
```

Переименование набора сигналов. Эта операция отключает обработку сигналов для целевых объектов, к которым был прикреплен этот набор сигналов. Синтаксис команды:

```
alertsedit.pl <templatename> rename <newname>
```

```
copy <newname>
```

Копирование набора сигналов в новый. Синтаксис команды:

```
alertsedit.pl <templatename> copy <new_templatename>
```

```
add <alertname> <alertclass> <data source> [parameters]
```

Добавление к набору сигналов нового сигнала. Синтаксис команды:

```
alertsedit.pl <templatename> add <alertname> <alertclass> <data source> [parameters]
```

где:

- ❖ `templatename` – имя существующего набора сигналов;
- ❖ `alertname` – имя сигнала в наборе сигналов. Должно быть уникальным внутри набора;
- ❖ `alertclass` – типа сигнала ([Gauge](#)^[30] / [Availability](#)^[34]);
- ❖ `datasource` – источник данных для сигнала;
- ❖ `parameters` – глобальные значения параметров сигнала (множество допустимых параметров зависит от типа сигнала).

14. Справочники

`del <alertname>`

Удаление сигнала из набора сигналов. Синтаксис команды:

```
alertsedit.pl <templatename> del <alertname>
```

`assign`

Привязка набора сигналов к конкретным целевым объектам или ко всем подходящим объектам в заданных поддеревьях. Набор сигналов может быть привязан к целевому объекту, только если тип целевого объекта совпадает с типом, указанным в наборе сигналов (параметр `target-type` набора сигналов). Одной командой можно привязать только один набор сигналов. Синтаксис команды:

```
alertsedit.pl <templatename> assign <subtree1> <subtree2> ...
```

`<alertname> assignre <regularexpression> <subtree>`

Привязка к целевым объектам поддерева набора сигналов с использованием регулярного выражения. Синтаксис команды:

```
alertsedit.pl <alertname> assignre <regularexpression> <subtree>
```

где:

- ❖ `alertname` – имя сигнала в наборе сигналов. Должно быть уникальным внутри набора;
- ❖ `regularexpression` – стандартное регулярное выражение Perl;
- ❖ `subtree` – поддерево.

`release <subtree1> <subtree2> ...`

Открепление набора сигналов от целевых объектов заданного поддерева. Синтаксис аналогичен синтаксису команды `assign`.

`notify`

Не используется.

`dump`

Вывод информации о настройках набора сигналов: параметров набора сигналов и глобальных значений параметров сигналов в составе набора.

`print`

Вывод информации о настройках набора сигналов (аналогично `dump`) в удобном для пользователя виде.

`setup "<param>=<value>"`

Изменение значений параметров набора сигналов или сигнала.

Для набора сигналов можно изменить только его описание. Синтаксис команды:

```
alertsedit.pl <templatename> setup "description=<value>"
```

Синтаксис команды для изменения глобального значения параметра сигнала:

```
alertsedit.pl <templatename>.<alertname> setup "<param>=<value>"
```

[OPTIONS]

`--classes`

Вывод списка существующих типов сигналов.

`--list`

Вывод списка существующих наборов сигналов.

`--cached`

Не используется.

`--nohdb`

Не используется.

`--version`

Вывод информации о версии ПМ **Нувола Топаз**.

`--help`

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

Вывод списка существующих наборов сигналов:

```
alertsedit.pl --list
```

Вывод существующих типов сигналов:

```
alertsedit.pl --classes
```

Создание набора под названием `myalert` с типом целевого объекта `target-type=generic-interface` (заведомо известно, что такой тип имеет целевой объект `/Tree/Head_Office/Main_server/eth0`):

```
alertsedit.pl myalert create /Tree/Head_Office/Main_server/eth0
```

Изменение нижнего порогового значения для сигнала `in`:

```
alertsedit.pl traffic.in setup "thresholds.low=10%"
```

Добавление сигнала `out` (с типом `Gauge`, источником данных `ifOutOctets` и пороговыми значениями 0 до 70%) к набору сигналов `traffic`:

```
alertsedit.pl traffic add out Gauge ifOutOctets 0 70%
```

Прикрепление набора сигналов `traffic` к подходящим целевым объектам во всем дереве конфигурации `/Tree`:

```
alertsedit.pl traffic assign /Tree
```

Прикрепление набора сигналов `traffic` к целевым объектам, подходящим к маске `eth.*`, в ветви дерева конфигурации `/Tree/Head_Office`:

```
alertsedit.pl traffic assignre eth.* /Tree/Head_Office
```

Открепление набора сигналов `traffic` от целевых объектов во всем дереве конфигурации `/Tree`:

```
alertsedit.pl traffic release /Tree
```

Удаление сигнала `out` из набора `traffic`:

```
alertsedit.pl traffic del out
```

Удаление набора сигналов `myalert`:

```
alertsedit.pl myalert remove
```

14.2.2. Управление резервным копированием конфигурации (backup)

Утилита `backup` предназначена для управления резервными копиями конфигурации ПК Нувола.

Утилита использует настройки, заданные в конфигурационном файле [backup.conf](#)^[78].

Синтаксис

```
kommander backup <ACTION> [OPTION]
```

<ACTION>

Выполняемое действие.

```
list
```

14. Справочники

Получение списка сохраненных файлов резервных копий, размещенных в директории, заданной в конфигурационном файле [backup.conf](#)^[78].

```
save <backup_file_name>
```

Создание резервной копии.

```
restore <backup_name>
```

Стандартное восстановление базы данных конфигурации из указанного файла резервной копии. Восстанавливаются директории, заданные параметром `folders_to_restore` в конфигурационном файле [backup.conf](#)^[78].

```
restorecustom <backup_name>
```

Нестандартное восстановление базы данных конфигурации и файлов, не включенных в стандартную поставку Системы Нувол, из указанного файла резервной копии. Восстанавливаются директории, заданные параметром `custom_folders_to_restore` в конфигурационном файле [backup.conf](#)^[78]. Команда используется на этапе внедрения Системы Нувол.

```
restoreall <backup_name>
```

Восстановление всех данных, сохранённых в резервной копии, в том числе исполняемых файлов, системных настроек, шаблонов объектов и др.

ВНИМАНИЕ! Команду `restoreall` не следует использовать без крайней необходимости! В результате ее выполнения восстанавливается не только дерево конфигурации и его настройки, но и исполняемые файлы, системные настройки, шаблоны устройств и др., т.е. происходит откат ПК Нувол на момент создания резервной копии.

```
delete <backup_name>
```

Удаление незащищенной резервной копии.

```
safe <backup_name>
```

Установка для заданной резервной копии признака защиты от удаления. Такие защищенные файлы невозможно удалить командами `delete` и `cleanup`.

```
unsafe <backup_name>
```

Снятие с резервной копии признака защиты от удаления.

```
cleanup
```

Очистка директории хранения. Под очисткой понимается удаление из директории хранения устаревших незащищенных резервных копий. Другими словами, в результате выполнения процедуры очистки из директории, заданной параметром `storefolder` в файле [backup.conf](#)^[78], удаляются файлы резервных копий (и связанные с ними файлы контрольных сумм), которые:

- ❖ старше срока, заданного параметром `period` в файле [backup.conf](#)^[78],
- ❖ не помечены признаком `safe`.

[OPTION]

```
force
```

Принудительное выполнение команды без проверки контрольной суммы.

ВНИМАНИЕ! Отключение проверки контрольной суммы позволяет восстановить базу данных конфигурации, даже если версия ПМ **Нувол Топаз**, в которой была создана резервная копия, отличается от текущей версии. Такое восстановление может пройти некорректно, поэтому его не рекомендуется выполнять без предварительной консультации со специалистами ООО «Венто Технологии».

Примеры

Восстановление конфигурации без проверки контрольной суммы:

```
kommander backup restore 2020-05-25_0102 force
```

См. также

[Резервное копирование и восстановление конфигурации](#)^[78]

14.2.3. Проверка состояния сервиса ESB (esbClient)

Утилита `esbClient` предназначена для прямого подключения к сервисной шине ESB с целью проверки состояния сервиса ESB (список подключенных клиентов и размера очередей).

Синтаксис

```
client [OPTIONS]
```

[OPTIONS]

```
--net-ip=string
```

Сетевой адрес подключения к брокеру для обмена сообщениями (по умолчанию **127.0.0.1**).

```
--net-port=number
```

Прослушиваемый сетевой порт для создания нового соединения (по умолчанию **5565**).

```
--name=string
```

Уникальный идентификатор клиента (по умолчанию **TestClient**).

```
--recv=number
```

Прием сообщений с сервера.

```
--messages=file
```

Отправка сообщений на сервер из файла.

```
--queues
```

Возвращение перечня и размеров очередей брокера ESB.

```
--delay-queues
```

Возвращение задержек сообщений в очереди (в секундах).

```
--destroy=queue name
```

Уничтожить очереди брокера ESB. Не рекомендуется уничтожать очереди брокера ESB без консультации со специалистами ООО «Венто Технологии».

14.2.4. Устранение всплесков (killspike.pl)

Всплески представляют собой резкое изменение наблюдаемых данных, отмеченное во время каких-либо системных изменений и по этой причине не соответствующее реальности.

Имеется несколько факторов появления всплесков:

- ❖ системное время было изменено без остановки сервиса [collector](#)^[275];
- ❖ тип счётчика SNMP был установлен или изменён с 32-бит на 64-бит;
- ❖ произошли физические изменения в наблюдаемом оборудовании.

Данные об обнаруженных всплесках можно удалить с помощью `killspike.pl`. Эта утилита находится в директории `util` модуля **Нувола Топаз** и позволяет удалить данные, собранные от устройств, в указанном временном диапазоне. Таким образом, её можно использовать для удаления из базы любых некорректных данных.

Синтаксис

```
killspike.pl [OPTIONS]
```

14. Справочники

[OPTIONS]

`--file <filename>`

Название обрабатываемого файла.

`--date <yyyy-mm-dd>`

Дата всплеска.

`--start <hh:mm>`

Начало всплеска.

`--end <hh:mm>`

Конец всплеска.

Примеры

Наилучшие результаты достигаются при совместном использовании утилиты `killspike.pl` с другими утилитами UNIX, например `find`. Команда удаления данных за 25 июля 2012 с 10:00 до 11:00 из всех файлов:

```
find /usr/local/lancelot/var/lancelot-data \  
-name "*.rrd" \  
-exec killspike.pl --file \{\} \  
--date 2012-07-25 \  
--start 10:00 \  
--end 11:00 \;
```

14.2.5. Компиляция базы данных (kompile)

Утилита `kompile` выполняет следующие функции:

- ❖ уведомляет БД API о необходимости обновить дерево конфигурации в соответствии с изменениями в текстовых файлах иерархической базы данных;
- ❖ пересоздает файлы `Defaults` в `/usr/local/lancelot/etc/lancelot-config/Devices` в соответствии с шаблонами объектов (т.е. в соответствии с файлами `<Module>.pm`, где `Module` – модуль `Perl`, реализующий шаблон объекта).

Утилиту `kompile` следует запускать после каждого ручного изменения файлов иерархической базы данных.

Синтаксис

`kompile [OPTIONS]`

[OPTIONS]

`--loglevel`

Глубина логов (уровень отладки).

`--logfile`

Файл логов.

`--version`

Вывод информации о версии ПМ **Нувола Топаз**.

`--help`

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

`--nohdb`

Не используется.

`--devices`

Присвоить параметры по умолчанию для каждого шаблона объектов.

14.2.6. Управление деревом конфигурации (konfne)

Утилита `konfne` представляет собой основной инструмент управления деревом конфигурации. При настройке нового или уже существующего объекта выполняются следующие действия:

- ❖ проверяется доступность соответствующего устройства по SNMP;
- ❖ общая конфигурация объекта помещается в `/Devices` в соответствии с классом объекта. Например, у маршрутизатора можно контролировать и интерфейсы, и само устройство. Интерфейсы помещаются в `/Devices/Interfaces/<routername>` или `<router ip>/`. Конфигурация маршрутизатора находится в `/Devices/Routers/`. Некоторые объекты, такие как отчёт, могут не иметь общей конфигурации;
- ❖ если общая конфигурация у объекта уже существует, то она обновляется;
- ❖ создаются ссылки и прочие необходимые элементы объекта.

Синтаксис

```
konfne [OPTIONS] [COMMAND] [OBJECT]
```

[OPTIONS]

```
--devlist
```

Вывод списка поддерживаемых шаблонов объектов.

```
-a, --autotype
```

Шаблон объекта назначается в соответствии с его системной информацией. Если используется `--autotype`, то опция `--device` игнорируется.

```
-i, --ip <ip>
```

IP-адрес или имя хоста устройства (значение параметра `ip`).

```
-c, --community <community>
```

Сообщество устройств SNMP (значение параметра `community`). По умолчанию используется `public`.

```
-f, --fetchname
```

Извлечение имени устройства из `sysName`. Символы после последнего слеша (`/`) в названии пути заменяются извлечённым именем.

```
-d, --device <device-name>
```

Шаблон объекта. Эта опция несовместима с автоматическим определением шаблона объекта `--autotype`.

```
-r, --recursively
```

Применение команды ко всем объектам в выбранном поддереве. Обычно используется для автоматического изменения конфигурации уже настроенных объектов.

```
-t, --tag <tagname>=<tagvalue>
```

Изменение значений параметров объекта. Большинство реальных объектов имеют следующие параметры:

`ip` – IP-адрес устройства.

`community` – Сообщество с доступом на чтение данных SNMP этого объекта (по умолчанию – `public`). Необязательный параметр для устройств без SNMP.

`display-name` – Имя объекта, отображаемое в дереве конфигурации. Должно быть уникальным в пределах одного уровня конфигурации (например, в рамках одной папки).

Для получения перечня параметров (тегов) следует использовать команду `help`.

См. также Шаблоны объектов ПК Нуволы.

14. Справочники

`--help`

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

`--aghost`

Опция для отладки. Вместо исполнения команды только печатает вызываемые команды.

`--langselect <lang>`

Смена языка (локализация) Web-интерфейса модуля **Нувол** **Топаз** на язык, заданный параметром `lang`. В настоящее время доступны опции **en** (английский язык Web-интерфейса) и **ru** (русский язык Web-интерфейса).

Рекомендуется выполнять локализацию до создания дерева конфигурации.

`--script <filename>`

[Загрузка объектов из файла](#) ^[81].

`-p, --problems <file>`

Имя файла для записи ошибок.

[COMMAND]

Список команд является расширяемым и зависит от шаблона объекта. Далее приведены команды, допустимые для большинства объектов.

`cfg`

Настройка новых объектов или обновление уже настроенных объектов.

`add`

Аналогична `cfg`, настройка новых объектов. Если объект с такими параметрами уже существует, он будет переименован.

`del`

Удаление объекта только из конкретного местоположения дерева конфигурации без прекращения сбора данных и с сохранением данных, собранных ранее данных.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то удаление таким способом затронет только заданный объект, а остальные объекты, соответствующие этому устройству, не изменятся.

Применительно к реальным устройствам этот способ удаления целесообразен, если требуется прекратить возможность просмотра данных по объекту для одной из нескольких учетных записей, которым этот объект доступен.

Если в дереве конфигурации объект зарегистрирован только один раз, то после удаления сбор данных продолжается, однако для просмотра они недоступны. Для выявления таких "невидимых" объектов можно воспользоваться отчетом, который формируется системным процессом `integrity`.

`delete`

Удаление объекта из конкретного местоположения дерева конфигурации с прекращением сбора данных, но сохранением данных, собранных ранее.

Собранные данные по объектам, удаленным таким способом, могут быть восстановлены: при повторной регистрации устройства в дереве конфигурации с теми же параметрами собранные данные попадут в графики.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то команда удаления затронет только заданный объект. Остальные объекты, соответствующие этому устройству, по-прежнему будут доступны через Web-интерфейс, однако информация по ним будет отображаться в усеченном виде.

DEMOLISH

Удаление объекта из конкретного местоположения дерева конфигурации вместе с собранными данными и прекращение сбора данных с объекта.

Собранные данные по объектам, удаленным таким способом, восстановлению не подлежат.

Если одно и то же устройство зарегистрировано в двух и более ветвях дерева конфигурации, то команда удаления затронет только заданный объект. Остальные объекты, соответствующие этому устройству, по-прежнему будут доступны через Web-интерфейс, однако информация по ним будет отображаться в усеченном виде.

`probe`

Проверка, отвечает ли соответствующее устройство на запросы SNMP.

`help`

Вывод справочной информации о заданном шаблоне объектов: список поддерживаемых параметров (тегов) и список применимых команд.

[ОБЪЕКТ]

Путь к объекту в дереве конфигурации или имя шаблона объектов, к которому применяется команда.

Примеры

Добавление в дерево конфигурации нового маршрутизатора Cisco с SNMP `secret@192.168.0.2` в `/Tree/Routers/c800`:

```
konfne --ip 192.168.0.2 --community secret --device Devices::Routers::Cisco
cfg /Tree/Routers/c800
```

или

```
konfne -i 192.168.0.2 -c secret -af cfg /Tree/Routers/
```

Обновление интерфейсов на маршрутизаторе:

```
konfne cfg /Tree/Routers/c800
```

Обновление конфигурации всех объектов в корневой папке:

```
konfne -r cfg /Tree
```

Получение информации для `Devices::Routers::Cisco`:

```
konfne help Devices::Routers::Cisco
```

Вывод списка поддерживаемых шаблонов объектов:

```
konfne --devlist
```

14.2.7. Конвертация БД из HDB в LDAP и обратно (ldapConfigManager)

Утилита `ldapConfigManager` позволяет конвертировать базу данных конфигурации из HDB-формата в LDAP и обратно.

Синтаксис

```
ldapConfigManager <OPTIONS>
```

<OPTIONS>

```
--import
```

14. Справочники

Импорт иерархической БД в LDAP.

```
--export
```

Экспорт иерархической БД из LDAP.

```
--path
```

Путь к базе данных конфигурации. Путь по умолчанию: `/usr/local/lancelot/etc/lancelot-config`

```
--debug
```

Не останавливаться на ошибках.

```
--help
```

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

Импорт иерархической БД в LDAP:

```
ldapConfigManager --import
```

Экспорт иерархической БД из LDAP:

```
ldapConfigManager --export
```

14.2.8. Ручная проверка корректности лицензий (license.pl)

Утилита `license.pl` предназначена для ручной проверки корректности файлов лицензий, размещенных в директории `/usr/local/lancelot/etc/license`.

Синтаксис

```
license.pl <OPTIONS>
```

<OPTIONS>

```
--list
```

Список файлов лицензий (`*.lic`) в директории `/usr/local/lancelot/etc/license`.

```
--verify <licfile>
```

Проверка корректности файла лицензии с заданным именем `<licfile>`.

```
-h, --help
```

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

Вывод списка файлов лицензий, размещенных директории `/usr/local/lancelot/etc/license`:

```
license.pl --list
```

Проверка файла лицензии `sybilla-01-01-201202.lic`:

```
license.pl verify sybilla-01-01-201202.lic
```

14.2.9. Запуск, проверка и завершение работы сервиса баз данных (mysqld)

Утилита `mysqld` позволяет запускать, останавливать, перезапускать и проверять состояние сервиса баз данных MySQL.

Синтаксис

```
/usr/local/lancelot/bin/mysqld <COMMAND> [MySQL server options]
```

<COMMAND>

Действие, выполняемое при данном вызове утилиты.

start

Запуск сервиса баз данных mysql.

stop

Остановка сервиса баз данных mysql.

restart

Перезапуск сервиса баз данных mysql.

reload

Применить изменения в конфигурации только когда все клиенты отключатся.

force-reload

Применить изменения в конфигурации без ожидания когда все клиенты отключатся.

status

Проверка состояние сервиса баз данных mysql.

14.2.10. Настройка оповещений (notifyedit.pl)

Утилита `notifyedit.pl` предназначена для настройки наборов оповещений и прикрепления их к целевым объектам.

Синтаксис

```
notifyedit.pl [OBJECT] [COMMAND] [OPTIONS]
```

[ОБЪЕКТ]

Набор оповещений или метод, к которому применяется команда.

<templatename>

Имя набора оповещений.

<templatename>.<method-class>

Метод или класс оповещений с именем <method-class> в составе набора оповещений с именем <templatename>.

all

Все наборы оповещений.

[COMMAND]

Действие, выполняемое при данном вызове утилиты.

assign <subtrees>

Прикрепление набора оповещений к заданным поддеревьям.

release <subtrees>

Открепление набора оповещений от заданных поддеревьев.

create

Создание нового набора оповещений с заданным именем.

rename <newname>

14. Справочники

Переименование набора оповещений. Если набор оповещений уже был прикреплен к целевым объектам, то его переименование приведёт к прекращению отправки оповещений.

`copy <newname>`

Копирование набора оповещений в новый.

`add <method+class>`

Добавление к набору оповещений метода оповещения или класса.

`setup <method+class.parameter=value>`

Изменение глобальных значений параметров набора, класса или метода оповещения.

`dump`

Вывод информации о настройках набора оповещений: параметров набора оповещений и глобальных значений параметров класса и методов оповещений в составе набора.

`print`

Вывод информации о настройках набора сигналов (аналогично `dump`) в удобном для пользователя виде.

`remove`

Удаление набора оповещений.

`del <method+class>`

Удаление указанного метода или класса из указанного набора оповещений.

`notify`

Установление метода оповещений (НЕ РЕКОМЕНДУЕТСЯ использовать).

[OPTIONS]

`--classes`

Вывод списка доступных классов оповещений.

`--methods`

Вывод списка доступных методов оповещений.

`--list`

Вывод списка существующих наборов оповещений.

`--cached`

Не используется.

`--nohdb`

Не используется.

`--version`

Вывод информации о версии ПМ **Нувола Топаз**.

`--help`

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

Создание нового набора оповещений:

```
notifyedit.pl logger create
```

Вывод списка доступных классов оповещений:

```
notifyedit.pl --classes
```

Вывод списка доступных методов оповещений:

```
notifyedit.pl --methods
```

Вывод списка существующих наборов оповещений:

```
notifyedit.pl --list
```

14.2.11. Управление системными процессами ПМ Нувола Топаз (overlord.pl)

Утилита `overlord.pl` позволяет создавать новые системные процессы на основе [процессов](#)^[267], реализованных в ПМ **Нувола Топаз**, изменять параметры существующих системных процессов, запускать, останавливать и перезапускать системные процессы.

Синтаксис

```
overlord.pl [COMMAND] [DAEMONS] [PARAMS] [OPTIONS]
```

[COMMAND]

Выполняемое действие.

`ping`

Вывод [статуса системного процесса](#)^[75].

`check`

Вывод статуса каждого настроенного системного процесса и запуск остановленных системных процессов, т.е. перевод системного процесса из статуса **down** в статус **up**.

`list`

Вывод списка параметров системного процесса.

`tail <name>`

Вывод последних строк лог-файла системного процесса.

`modify`

Изменение параметров запуска системного процесса. После установки модуля **Нувола Топаз** из RPM или DEPOТ параметрам присвоены значения по умолчанию. Не рекомендуется изменять их без предварительной консультации с представителем ООО «Венто Технологии».

`start`

Аналог `check`. Запуск системного процесса.

`kill`

Остановка системного процесса.

`stop`

Аналог `kill`. Остановка системного процесса.

`restart`

Перезапуск системного процесса, т.е. последовательное выполнение остановки и запуска системного процесса.

`disable`

Отключение системного процесса, т.е. перевод его в статус **disabled** (при этом для системного процесса устанавливается параметр `disabled=true`).

`enable`

Включение и запуск системного процесса, т.е. перевод системного процесса из статуса **disabled** в статус **up** (при этом для системного процесса устанавливается параметр `disabled=false`).

`create <name>`

Создание нового системного процесса.

`remove`

Удаление системного процесса. Не рекомендуется использовать эту команду без предварительной консультации с представителем ООО «Венто Технологии».

`rotate`

Ротация лог-файлов системного процесса.

14. Справочники

[DAEMONS]

Регулярные выражения, соответствующие именам системных процессов. По умолчанию – все системные процессы (*).

[PARAMS]

Значения параметров системных процессов в формате "<param_name>=<value>".

[OPTIONS] --version

Вывод информации о версии ПМ **Нувола Топаз**.

--help

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

Создание нового системного процесса сбора данных:

```
overlord.pl create client "interval=120" "mtargets=true" "list=/Devices"
"script=kollector"
```

Просмотр статусов системных процессов и запуск остановленных системных процессов (в статусе **down**):

```
./overlord.pl check
```

Перезапуск всех системных процессов:

```
overlord.pl restart
```

Отключение системного процесса ping:

```
overlord.pl disable ping
```

Просмотр значений параметров всех системных процессов:

```
overlord.pl list
```

Просмотр значений параметров системного процесса alerts:

```
overlord.pl list alerts
```

Просмотр информации о версии ПМ **Нувола Топаз**:

```
overlord.pl --version
```

14.2.12. Проведение нагрузочных тестов дисковой подсистемы (randomwrite.pl)

Утилита `randomwrite.pl` предназначена для проведения нагрузочных тестов дисковой подсистемы с использованием типичного шаблона поведения при записи данных.

Синтаксис

```
randomwrite.pl [OPTIONS]
```

[OPTIONS]

```
-b, --block N
```

Размер блока файла в байтах (по умолчанию **1024**).

```

-r, --record N
    Размер тестового отчета в байтах (по умолчанию 128).
-f, --files N
    Количество файлов для теста (по умолчанию 4096).
-t, --tests N
    Количество тестов (по умолчанию 5).
-p, --path <path>
    Путь к директории для тестов (по умолчанию /tmp/).
--pfx <prefix>
    Префикс для тестовых файлов (по умолчанию test).
-r, --remove <y|n|1|0|true|false>
    Удаление файлов после каждого теста (по умолчанию true). Возможные значения:
    y|1|true – удалять файлы;
    n|0|false – не удалять файлы.
-w, --write N
    Количество тестовых отчетов в файле (по умолчанию 1).
-h, --help
    Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

```

14.2.13. Создание комплекса внутренней диагностики ПМ Нувола Топаз (selfMonitoringConfig.pl)

Утилита `selfMonitoringConfig.pl` предназначена для создания и переконфигурации [комплекса внутренней диагностики](#)^[154].

Утилита запускается автоматически при установке ПМ **Нувола Топаз** и создает комплекс внутренней диагностики в уменьшенном виде. Администратор `root` с помощью данной утилиты может переконфигурировать комплекс.

Синтаксис

```
selfMonitoringConfig.pl [OPTIONS]
```

[OPTIONS]

```

-l, --lancelot
    Путь до папки lancelot. По умолчанию – '/usr/local/lancelot'.
-f, --folder
    Отображаемое имя папки, в которой хранится комплекс. По умолчанию – 'Внутренняя
    Диагностика'.
-n, --name
    Отображаемое имя объекта, создаваемого по шаблону Devices::Analysis::Lancelot. По
    умолчанию – 'Мониторинг Сервисов'.
-d, --dashboard
    Путь до архива self_mon_status_blank.db.tpl.gz, в котором находится информационная
    панель Self Mon Status (компонента комплекса внутренней диагностики). По умолчанию –
    '/usr/local/lancelot/var/lancelot-data/Tree/SelfMon'.
-t, --templated
    Возможные значения:
    true – удаляет у объекта, созданного по шаблону Devices::Analysis::Lancelot, все
    диски, кроме '\ ' и 'Swap space'. Использовать НЕ рекомендуется;

```

`false` – по умолчанию.

```
-, --help
```

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Примеры

При установке ПМ **Нувола Топаз** комплекс внутренней диагностики создается автоматически в уменьшенном виде: отсутствуют некоторые основные процессы, а также все диски, кроме `\` и `'Swap space'`. Для этого используется команда:

```
./selfMonitoringConfig.pl -t true
```

Команда для создания/переконфигурации комплекса с именами на русском языке:

```
./selfMonitoringConfig.pl
```

или

```
./selfMonitoringConfig.pl -f 'Внутренняя Диагностика' -n 'Мониторинг Сервисов'
```

Команда для создания/переконфигурации комплекса с именами на английском языке:

```
./selfMonitoringConfig.pl -f 'Self Monitoring' -n 'Services Monitoring'
```

14.2.14. Проверка целостности кода (sign)

Проверка целостности кода ПК Нувола.

Утилита `sign` используется для контроля целостности кода установленного ПК Нувола. Утилита позволяет создавать снимки контрольных сумм кода, сравнивать их между собой и со снимком текущего состояния. Для вычисления контрольной суммы используется любая из программ MD5, SHA-1, SHA-2 или совместимая по формату командной строки с `md5sum`. По умолчанию используется `sha256sum`. Можно использовать другую доступную программу, указав её имя, как параметр утилиты `sign`.

Снимки контрольных сумм сохраняются в директории `$HOME/lancelot-backups` и имеют формат имени `<год-месяц-день_часминуты>.chk`. Первый снимок контрольных сумм создаётся автоматически при инсталляции файла `*.rpm`. В дальнейшем при [создании резервной копии](#)^[78] также выполняется проверка снимка контрольных сумм (`kommander sign watch`).

Синтаксис

```
kommander sign <ACTION> [--cmd PROGRAM] [FILE]
```

<ACTION>

Выполняемое действие.

```
refresh
```

Вычисление контрольных сумм файлов и их сохранение.

Если снимок контрольных сумм успешно сохранён, то сообщение не выдаётся. В случае возникновения проблем выдаётся сообщение об ошибке.

```
check
```

Вычисление контрольных сумм файлов и сверка их с ранее сохранённым снимком.

Если снимки совпадают, то сообщение не выдаётся. Если снимки не совпадают, то выдаётся сообщение об ошибке сверки:

```
[avalon@devel-lnx-6 lancelot]$ kommander sign check
sha256sum: WARNING: 2 of 613 computed checksums did NOT match
/usr/local/lancelot/lib/Command/.Sign.pm.swp: FAILED
/usr/local/lancelot/lib/Command/Sign.pm: FAILED
```


watch

Проверка контрольных сумм файлов в текущем состоянии по сравнению с последним сохранённым снимком. Создание снимка, если снимков не обнаружено или обнаружены расхождения по сравнению с последним сохранённым снимком.

lock

Включение режима проверки целостности кода при запуске модуля **Нувола Топаз**, создание и обновление файла с контрольными суммами. Файл с контрольными суммами `.lock.chk` помещается в `$Common::global::gDataRoot`.

lcheck

Проверка контрольных сумм по файлу, созданному по команде `kommander sign lock`.

unlock

Отключение режима проверки при запуске модуля **Нувола Топаз**.

diff <FILE1> <FILE2>

Сравнение двух указанных файлов снимков (подобно утилите `diff`, используемой в Unix), возвращение списка файлов, чьи контрольные суммы не совпадают в разных снимках.

help | --help | -h

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

[--cmd PROGRAM]

Программа, используемая для вычисления контрольной суммы. По умолчанию используется `sha256sum`.

[FILE]

Имя файла со снимком контрольных сумм. Можно указать свой файл, например, расположенный на инженерном флеш-носителе.

Примеры

Проверка текущих контрольных сумм относительно самого последнего файла `.chk` в директории `~/lancelot-backups`:

```
kommander sign check
```

Проверка текущих контрольных сумм относительно конкретного файла `/mydisk/myfile.chk`:

```
kommander sign check /mydisk/myfile.chk
```

Проверка текущих контрольных сумм относительно конкретного «автоматического» снимка:

```
kommander sign check /home/avalon/lancelot-backups/2013-11-22_1729.chk
```

Сравнение двух автоматических снимков, созданных с помощью `watch`:

```
kommander sign diff /home/avalon/lancelot-backups/2013-11-22_1729.chk /home/avalon/lancelot-backups/2013-11-26_0101.chk
```

Проверка контрольных сумм относительно самого последнего снимка, создание нового снимка в случае обнаружения несоответствия:

```
kommander sign watch
```

Использование утилиты `md5sum` для вычисления и проверки контрольных сумм:

```
kommander sign refresh --cmd md5sum
kommander sign check --cmd md5sum
```

14.2.15. Просмотр значений параметров узла дерева конфигурации (target.pl)

Утилита `target.pl` позволяет просмотреть значения параметров заданного узла дерева конфигурации.

Утилиту можно использовать при отладке базы данных конфигурации.

Синтаксис

```
target.pl [OPTIONS]
```

[OPTIONS]

```
--target <path>
```

Уникальный путь, идентифицирующий узел дерева конфигурации.

```
--dictionary
```

Название существующего каталога. По умолчанию используется каталог `target`.

```
--dictionary name
```

Название элемента каталога. Например, `ifInOctets` – подходящее название для каталога `graph`. Этот параметр не нужно указывать для целевого каталога.

```
--show-children
```

Показать дочерние узлы указанного узла дерева конфигурации.

```
--nolinks
```

Не отслеживать ссылки в базе данных.

```
--loglevel
```

Глубина логов (уровень отладки).

```
--logfile
```

Файл логов.

```
--nohdb
```

Не используется.

```
--version
```

Вывод информации о версии ПМ **Нувола Топаз**.

```
--help
```

Вывод справки об утилите (выводятся описания команд, параметров, примеры использования).

Пример

Вывод значений параметров неконцевого узла `/Tree/Routers/rothut`:

```
target.pl --show-children /Tree/Routers/rothut
```

Ориентировочный ожидаемый результат:

```
target: /Tree/Routers/rothut
--heritage-- = :--default--:--default--:--default--
--theme-- = gpl
auto-base = /usr/local/lancelot/etc/lancelot-config
auto-is-dir = true
auto-real-name = /Tree/Routers/rothut
auto-root = ../../
auto-target-name = rothut
auto-target-name-linked = rothut
auto-target-name-unlinked = rothut
auto-target-path = /Tree/Routers
auto-target-path-linked = /Tree/Routers
```

```

auto-target-path-unlinked = /Tree/Routers
community = bigsecret
datadir = /usr/local/lancelot/var/lancelot-data//Tree/Routers
device = Devices::Computers::Linux
ip = rothut
password = guest
profile = guest
rrd-datafile = /usr/local/lancelot/var/lancelot-data//Tree/Routers/rothut.rrd
tree-background = cccff

children:
/Tree/Routers/rothut/cpu-lnk
/Tree/Routers/rothut/eth0-lnk
/Tree/Routers/rothut/memory-lnk
/Tree/Routers/rothut/storage1-lnk
/Tree/Routers/rothut/storage3-lnk
/Tree/Routers/rothut/storage4-lnk
/Tree/Routers/rothut/storage5-lnk

```

14.2.16. Проверка возможности доступа по SNMP (test_access.pl)

Утилита `test_access.pl` предназначена для проверки возможности доступа по протоколу SNMP к заданным объектам. Список проверяемых объектов передается в файле.

Синтаксис

```
test_access.pl <device_list_file>
```

где:

```
<device_list_file>
```

Файл со списком объектов.

14.2.17. Асинхронный опрос по ping (X3PingAsync.pl)

Утилита `X3PingAsync.pl` позволяет опрашивать устройства с помощью `ping`, используя асинхронный алгоритм опроса.

14.3. Системные процессы ПМ Нуволы Топаз

В настоящем справочнике приведено формализованное описание системных процессов ПМ **Нуволы Топаз**, необходимых для функционирования модуля **Нуволы Аналитика**. Пункты отсортированы по именам процессов в алфавитном порядке.

Для управления системными процессами (если не указано особо) можно использовать утилиту [overlord.pl](#)^[261] или приложение [Диспетчер процессов](#)^[156] в Web-интерфейсе.

Параметры системных процессов можно условно разделить на две группы:

- ❖ стандартные – имеются у большинства системных процессов; они описаны в разделе [Стандартные параметры системных процессов](#)^[268] и не упоминаются в пунктах, посвященных конкретным системным процессам;
- ❖ специфические – описаны в специализированных разделах о системных процессах.

14.3.1. Стандартные параметры системных процессов

В настоящем разделе приведены параметры, которые имеются у большинства системных процессов. Особенности ввода значений некоторых стандартных параметров содержатся в пунктах с описанием системных процессов.

`name`

Имя системного процесса.

`loglevel`

Уровень логирования – степень детализации информации в лог-файле. Возможные значения:

Info – базовая информация (наиболее эффективный уровень);

Debug – детальная информация (применяется на этапе разработки/внедрения);

Warn – информация о предупреждениях;

Error – информация об ошибках.

`logfile`

Директория и имя лог-файла для записи отладочной информации. Не рекомендуется изменять директорию хранения лог-файла по умолчанию `/usr/local/lancelot/var/lancelot-logs/`. Системный процесс создает отдельный файл для каждого потока программы с указанием в имени файла номера PID программы, порядкового номера потока и номера файла в данном потоке.

`interval`

Периодичность работы системного процесса, в секундах.

`esb`

IP-адрес хоста, на котором работает шина `esb` (по умолчанию **127.0.0.1**). Чтобы отключить системный процесс от шины [ESB](#)^[18], необходимо ввести значение **false**.

`fastcache`

Отключение от сервера конфигурации, использование отдельного кеша.

`list`

Ветви дерева конфигурации (через запятую).

14.3.2. Формирование сигналов (alerts)

Системный процесс `alerts`, основанный на процессе `alerts.pl`, обрабатывает данные, собранные системным процессом `collector`, и вычисляет статусы сигналов, фиксируя, таким образом, наличие проблем.

См. [Вычисление статусов сигналов и отправка оповещений](#)^[28].

Параметры

`spooled`

Настройка высокопроизводительного режима обработки сигналов. При установленном флаге системный процесс `alerts` извлекает данные из буферных файлов, созданных системным процессом [collector](#)^[27] (при этом у системного процесса `collector` должен быть установлен флаг `spooler`).

`list`

Ветви дерева конфигурации (через запятую).

`spooldir`

Настройка высокопроизводительного режима обработки сигналов. Директория, из которой извлекаются собранные данные.

`reports`

Хранение статистики для ускорения создания отчетов.

fifo

Путь на директорию на сервере Нуволы и имя размещенного в ней файла, используемого системным процессом `omniorb-eventservice` при построении очереди событий. Параметр используется только в режиме высокопроизводительной обработки сигналов. Заданное значение должно совпадать со значением параметра `fifo` системного процесса `fnmpunp`.

qsize

Размер очереди событий. Значение параметра следует задавать только при использовании высокопроизводительного режима обработки сигналов. По умолчанию параметр не задан.

external-plug

Для включения высокопроизводительного режима обработки сигналов следует ввести значение FIFO, а также задать параметры `fifo` и `qsize`.

По умолчанию значение не задано и высокопроизводительный режим не используется.

no-esb-ds

Получение данных из RRD-файлов. По умолчанию отключено (**false**).

no-esb-alerts

Параметр влияет на работу модуля **Нуволы Топаз** только при использовании сервисной шины [ESB](#)^[18]. Возможные значения:

- ❖ `false` (по умолчанию) – конфигурация сигналов хранится на сервере конфигураций, а значения сигналов – на сервере данных; системный процесс `alerts.pl` получает данные из шины ESB, и данные по изменению состояния сигнала передает обратно в ESB;
- ❖ `true` – конфигурация сигналов хранится в виде файлов на диске; системный процесс `alerts.pl` обменивается данными через шину ESB.

enabled-notifications

Шаблоны [оповещений](#), обрабатываемые системным процессом `alerts`. Значение параметра – регулярное выражение для отбора шаблонов оповещений по имени. Оповещения по отобранным шаблонам будет формировать системный процесс `alerts` (вместо системного процесса [notifs](#)^[278]).

Начиная с версии 7.4 функционал формирования оповещений вынесен в отдельный системный процесс [notifs](#)^[278]. Поэтому изменять значение данного параметра не нужно.

Распределение нагрузки между системными процессами alerts для оптимизации производительности ПК Нуволы

Первоначальные настройки ПМ **Нуволы Топаз** предусматривают один системный процесс на основе процесса `alerts.pl` – с именем `alerts`. С увеличением количества целевых объектов очереди к нему могут возрастать, что может отрицательно сказаться на производительности ПК Нуволы в целом.

```
[avalon@lnx-2 ~]$ /usr/local/lancelot/util/esb.queue.pl
```

```
-----
2019-05-23 15:57:20
S ALERTS 231152 +596
-----
```

Для решения данной проблемы рекомендуется создать несколько системных процессов на основе процесса `alerts.pl` и распределить нагрузку между ними.

При настройке системных процессов следует принимать во внимание значение поля [Принадлежность](#)^[125] (`belongs_to`) в наборах сигналов:

Если поле **Принадлежность** не заполнено, то обрабатывать сигналы данного набора могут системные процессы с именами вида `alertsN`, где N – необязательный порядковый номер. Например, системный процесс `alerts` или `alerts42` может обработать сигналы, а системный процесс `alerts-mega` – не сможет.

Аналогичный принцип соблюдается, если поле **Принадлежность** заполнено. Например, если в нем указан системный процесс `sybilla_alerts`, то сигналы могут быть обработаны системными процессами с именами `sybilla_alerts` и `sybilla_alerts98`, но и не могут быть обработаны системными процессами `alerts` и `alerts_sybilla`.

Такую жесткую привязку набора сигналов к системному процессу с именем (шаблоном имени), указанным в поле **Принадлежность**, можно отключить, если для системного процесса на основе процесса `alerts.pl` установить параметр `ignore-belongs-to` (но обычно в этом нет необходимости).

Возможны следующие варианты:

- ❖ Создать несколько системных процессов с именами `alerts`, `alerts2`, `alerts3`, ... и настроить автоматическое распределение нагрузки с помощью параметра `balancing`. Например:

```
[avalon@lnx-2 ~]$ overlord.pl list alerts
Create new cHdb connection ...

alerts3(alerts.pl)
  balancing=Hash 3/3
  esb=127.0.0.1
  interval=210
  logfile=/usr/local/lancelot/var/lancelot-logs/alerts3.log
  no-esb-alerts=false
alerts2(alerts.pl)
  balancing=Hash 2/3
  esb=127.0.0.1
  interval=210
  logfile=/usr/local/lancelot/var/lancelot-logs/alerts2.log
  no-esb-alerts=false
alerts(alerts.pl)
  balancing=Hash 1/3
  esb=127.0.0.1
  interval=210
  logfile=/usr/local/lancelot/var/lancelot-logs/alerts.log
  no-esb-alerts=false
```

- ❖ В дополнение к системному процессу `alerts` создать еще несколько – под разными именами, например `alerts-param1`, `alerts-param2`. Чтобы распределить нагрузку, в нескольких наборах сигналов задать параметр **Принадлежность**^[125] (`belongs-to`), указав имя одного из созданных системных процессов.
- ❖ Создать несколько системных процессов с разными именами, например `alerts1`, `alerts2`, а для распределения нагрузки в параметре `list` указать для разных системных процессов различные ветви дерева конфигурации (`--list=/Sub-tree1`, `--list=/Sub-tree2`). Таким образом, необходимо разделить дерево конфигурации между системными процессами на основе `alerts.pl`. Например:

```
[avalon@qa-lnx-2 ~]$ overlord.pl list alerts
Create new cHdb connection ...

alerts2(alerts.pl)
  disabled=false
  esb=127.0.0.1
  interval=210
  list=/Ros/South/Objects,/Ros/North/Objects
  logfile=/usr/local/lancelot/var/lancelot-logs/alerts-param-2.log
  no-esb-alerts=false
alerts1(alerts.pl)
  disabled=false
  esb=127.0.0.1
  interval=210
  list=/Ros/East/Objects,/Ros/West/Objects
```

```
logfile=/usr/local/lancelot/var/lancelot-logs/alerts-param-1.log
no-esb-alerts=false
```

14.3.3. Хранение данных (dataserver)

Системный процесс хранения данных `dataserver` (исполняемый файл `dataServer`) обеспечивает организацию хранения данных в RRD-файлах на локальной или удаленной машине, а также обеспечивает выгрузку во внешние базы данных (Oracle, MySQL).

Для конфигурирования системного процесса `dataserver` необходимо задать значения входных аргументов при запуске или настроить конфигурационный файл.

Параметры

`daemonize`

Сервер будет запущен в режиме сервиса. По умолчанию включен.

`log-volume-size`

Максимальный размер одного лог-файла.

При превышении размера файла данного значения, будет создан еще один файл для данного потока, с увеличением значения номера файла в данном потоке. Значение по умолчанию равно 10 Мб.

`log-history-size`

Максимальное число файлов с сообщениями сохраняемых для каждого потока программы. Значение по умолчанию равно 4.

`log-debug-channels`

Задать имена типов `debug` сообщений. Данный фильтр применяется только при `--log-level=5`.

Аргумент может принимать значения: **table**, **dataserver**, **pds**, **zmqsocket**, **zmqtransfer**.

`log-show-time`

Включить/отключить запись времени в сообщениях от сервера.

`log-show-filename`

Включить/отключить запись имени файла в сообщениях от сервера.

`configuration`

Задается имя файла с конфигурацией.

В файле конфигурации могут задаваться все предыдущие аргументы и списки сервисов с идентификаторами общих данных.

`name`

Идентификатор сервера, который будет передаваться в сообщениях от него. При использовании нескольких серверов, идентификаторы серверов не должны совпадать. Значение по умолчанию равно **DS**.

`prefix-to-path`

Задать префикс к пути расположения RRD файлов. Значение по умолчанию **/usr/local/lancelot/var/lancelot-data**.

`path-to-data`

Задать путь, указывающий, где расположены RRD файлы. Данный аргумент используется при создании таблицы существующих RRD файлов. От заданного пути производится сканирование диска на наличие RRD файлов. Полный путь равен сумме `prefix-to-path` + `path-to-data`. Значение по умолчанию равно **/Devices**.

`net-ip`

IP-адрес, с которого разрешено подключение к данному серверу. Адрес ***** разрешает подключение с любого IP-адреса. При использовании аргумента `broker` в файле конфигурации задается IP-адрес брокера, к которому необходимо подключиться.

`net-port`

14. Справочники

Сетевой порт сокета, к которому будут подключаться клиенты для передачи сообщений, по умолчанию – **5566**. При использовании аргумента `broker` в файле конфигурации значение данного порта будет использоваться для подключения к брокеру.

`version`

Просмотр данных о версии.

`help`

Просмотр справки по использованию системного процесса.

Параметры файла конфигурации

Конфигурируемые параметры в файле конфигурации разбиты на логические группы. В группе `LOGGING` задаются параметры для логирования работы программы:

- ❖ `file` – аналогичен параметру `--log-file` в командной строке;
- ❖ `level` – аналогичен параметру `--log-level` в командной строке;
- ❖ `volume-size` – аналогичен параметру `--log-volume-size` в командной строке;
- ❖ `history-size` – аналогичен параметру `--log-history-size` в командной строке;
- ❖ `show-filename` – аналогичен параметру `--log-show-filename` в командной строке;
- ❖ `show-time` – аналогичен параметру `--log-show-time` в командной строке;
- ❖ `debug-channels` – аналогичен параметру `--log-debug-channels` в командной строке.

В группе `CONFIG` задаются основные параметры программы необходимые для работы брокера:

- ❖ `name` – аналогичен параметру `--name` в командной строке;
- ❖ `net-IP` – аналогичен параметру `--net-IP` в командной строке;
- ❖ `net-port` – аналогичен параметру `--net-port` в командной строке;
- ❖ `broker` – указать, что данный сервер будет подключаться к брокеру или нет.

В группе `SERVICES` задается список системных процессов для обмена данными посредством ESB. Для каждого системного процесса требуется задать список общих данных, которые будут получены от системного процесса `esb`. Список типов данных и правил фильтрации задаются в группах с названиями системных процессов. В группе должны присутствовать параметры `typeN`, где `N` номер по порядку начиная с 1. В этих параметрах задается имя публикуемых данных.

14.3.4. Выборка данных для подготовки XML-фрагментов (`de_providerd`)

Системный процесс `de_providerd`, основанный на процессе `de_providerd.pl`, отбирает данные для подготовки XML-фрагментов.

14.3.5. Сервисная шина предприятия (`esb`)

Сервисная шина предприятия `esb` (исполняемый файл `esbBroker`) обеспечивает связь компонент модуля **Нувола Топаз**.

Параметры системного процесса `esb` задаются при его запуске или в конфигурационном файле.

Параметры

`daemonize`

Сервер будет запущен в режиме сервиса. По умолчанию включен.

log-volume-size

Максимальный размер одного лог-файла.

При превышении размера файла данного значения, будет создан еще один файл для данного потока, с увеличением значения номера файла в данном потоке. Значение по умолчанию равно 10 Мб.

log-history-size

Максимальное число файлов с сообщениями сохраняемых для каждого потока программы. Значение по умолчанию равно 4.

log-debug-channels

Задать имена типов debug сообщений. Данный фильтр применяется только при `--log-level=5`.

Аргумент может принимать значения: **table**, **dataserver**, **pds**, **zmqsocket**, **zmqtransfer**.

log-show-time

Включить/отключить запись времени в сообщениях от сервера.

log-show-filename

Включить/отключить запись имени файла в сообщениях от сервера.

configuration

Задается имя файла с конфигурацией.

В файле конфигурации могут задаваться все предыдущие аргументы и списки сервисов с идентификаторами общих данных.

net-ip

IP-адрес, с которого разрешено подключение к данному серверу. Адрес ***** разрешает подключение с любого IP-адреса. При использовании аргумента `broker` в файле конфигурации задается IP-адрес брокера, к которому необходимо подключиться.

net-port

Сетевой порт сокета, к которому будут подключаться клиенты для передачи сообщений, по умолчанию – **5566**. При использовании аргумента `broker` в файле конфигурации значение данного порта будет использоваться для подключения к брокеру.

configServer-ip

IP адрес машины, где работает служба `omniNames`, в которой зарегистрирован сервер конфигурации. Значение по умолчанию равно **127.0.0.1**.

configServer-port

Сетевой порт для подключения к службе `omniNames`. Значение по умолчанию равно **2809**.

configServer-update

Задается промежуток времени через который будет производиться обновление таблицы соответствия путей к `gtd` файлам по данным сервера конфигурации.

check

Команда, выполняемая утилитой `overlord.pl` при получении команды `check` (запуск остановленного системного процесса).

esb

IP-адрес хоста, на котором работает шина `esb` (по умолчанию **127.0.0.1**). Чтобы отключить системный процесс от шины [ESB](#)¹⁸⁾, необходимо ввести значение **false**.

pid_regexp_string

Параметр, используемый утилитой `overlord.pl` для идентификации процесса.

restart

Команда, выполняемая утилитой `overlord.pl` для перезапуска сервиса.

start

Команда, выполняемая утилитой `overlord.pl` для запуска сервиса.

stop

Команда, выполняемая утилитой `overlord.pl` для остановки сервиса.

Параметры файла конфигурации

Конфигурируемые параметры в файле конфигурации разбиты на логические группы. В группе `LOGGING` задаются параметры для логирования работы программы:

- ❖ `file` – аналогичен параметру `--log-file` в командной строке;
- ❖ `level` – аналогичен параметру `--log-level` в командной строке;
- ❖ `volume-size` – аналогичен параметру `--log-volume-size` в командной строке;
- ❖ `history-size` – аналогичен параметру `--log-history-size` в командной строке;
- ❖ `show-filename` – аналогичен параметру `--log-show-filename` в командной строке;
- ❖ `show-time` – аналогичен параметру `--log-show-time` в командной строке;
- ❖ `debug-channels` – аналогичен параметру `--log-debug-channels` в командной строке.

В группе `CONFIG` задаются основные параметры программы, необходимые для работы брокера:

- ❖ `net-ip` – аналогичен параметру `--net-ip` в командной строке;
- ❖ `net-port` – аналогичен параметру `--net-port` в командной строке;
- ❖ `swap-file-path` – аналогичен параметру `--swap-file-path` в командной строке;
- ❖ `hwm` – задать максимальное число сообщений, которые будут храниться в файле для передачи клиентам (серверам) при отсутствии связи. Значение по умолчанию равно **1000000**.

В группе `CONFIG_SERVER` задаются параметры программы необходимые для подключения к серверу конфигурации:

- ❖ `net-ip` – IP адрес машины, где работает служба `omniNames`, в которой зарегистрирован сервер конфигурации. Значение по умолчанию равно **127.0.0.1**;
- ❖ `net-port` – сетевой порт для подключения к службе `omniNames`. Значение по умолчанию равно **2809**;
- ❖ `time_update` – задается промежуток времени, через который будет производиться обновление таблицы соответствия путей к `gtd` файлам по данным сервера конфигурации.

В группах `CONNECT_N` (где `N` – число от **1** до **100**) задаются параметры подключения текущего сервиса `esb` к другому сервису `esb`:

- ❖ `name` – идентификатор сервиса `esb`, который будет передаваться в сообщениях от него. При использовании нескольких брокеров, их идентификаторы не должны совпадать. Значение по умолчанию равно **Broker**;
- ❖ `net-ip` – задать IP адрес сервиса `esb`, к которому будет подключаться данный брокер;
- ❖ `net-port` – задать сетевой порт сокета сервиса `esb`, к которому будем подключаться для передачи сообщений.

В группе `SERVICES` задается список сервисов, которые необходимо создать на брокере, к которому подключаемся. Для каждого сервиса необходимо задать список общих данных, которые необходимо получать от брокера. Список типов данных и правил фильтрации задаются в группах с названиями сервисов. В группе должны присутствовать параметры:

- ❖ `socket` – имя сервиса `esb`, на котором создать данный сервис;

- ❖ `filter` – задается ветка поддерева конфигурации, которой принадлежит устройство, данные от которого передаются. Значение параметра `filter` может быть пустым, что означает передавать все сообщения без применения фильтрации;
- ❖ `typeN` (где N номер по порядку начиная с 1) - задается имя публикуемых данных.

14.3.6. Отображение изменений в реальном времени (jrtgd)

Системный процесс `jrtg`, основанный на процессе `jrtgd.pl`, отвечает за отображение изменений в реальном времени.

14.3.7. Сбор данных (collector)

Коллектор – обобщённое название системных процессов, основанных на процессе `collector`, осуществляющих сбор данных с объектов контроля и сохранение полученной информации в базе данных.

Системный процесс `collector`, основанный на процессе `collector`, выполняет сбор данных с объектов контроля, опрашивая их по протоколу SNMP и используя линейный метод, полученную информацию о состоянии целевых объектов записывает в базу данных.

По умолчанию предустановленный в ПМ **Нувола Топаз** системный процесс `collector` собирает данные со всех объектов контроля, зарегистрированных в дереве конфигурации. Исключение составляют объекты контроля, для которых предусмотрены специализированные коллекторы в составе других модулей ПК Нувола, требующие дополнительной настройки (см. далее).

Опрос большого числа объектов контроля может занять продолжительное время. Поэтому для оптимизации производительности ПМ **Нувола Топаз** и снижения нагрузки на системный процесс `collector` администратор может создать и запустить дополнительные коллекторы.

При [конфигурировании устройства](#)^[117] в поле **Выбрать коллектор** можно явно указать коллектор, который будет осуществлять сбор данных с устройства.

Функционирование некоторых модулей ПК Нувола обеспечивают специализированные коллекторы: `fnmp`, `collector-services`, `rmf-collector`, `collector-logix`, `collector-ux`. Они настраиваются в соответствии с проектным решением.

Параметры

`server`

Запуск сбора данных в серверном режиме. Этот режим необходим для репликации данных, а также для графических сервисов, работающих в режиме реального времени, в том числе JRTG (Java Real Time Grapher).

`port <port>`

Прослушивание подключений заданного порта.

`host <ip-name>`

IP-адрес или хост, который прослушивается сервисом. По умолчанию ведется наблюдение за входящими подключениями к **0.0.0.0**.

`allowed <ip regex>`

Разрешение подключений только с IP адресов, которые удовлетворяют регулярному выражению. По умолчанию разрешены подключения со всех IP адресов.

`spooler`

Если флаг установлен, то данные сохраняются на диске без обновления базы данных. Данные могут использоваться для дублирования или для очень быстрого создания сигнала или отчёта.

`spooldir`

Директория для хранения буфера собранных данных.

`localcopy`

14. Справочники

Флаг используется, только если установлен флаг `spooler`. Если флаг `localcopy` установлен, то данные сохраняются в базе данных. Если нет, то данные сохраняются на диске без обновления базы данных.

`client`

Если флаг установлен, то `collector` берет данные с удалённых серверов. Удалённые сервера должны быть описаны в каталоге серверов в корневой папке конфигурации. Сервис на удалённом сервере должен иметь флаг `client=true`. Синхронизация между клиентом и сервером должна осуществляться каким-либо внешним механизмом, таким как CVS, rsync или rdist.

`sqlcache`

При установленном флаге `sqlcache=true` в качестве временного хранилища данных используется база данных MySQL, а не оперативная память. Использование этого позволяет оптимизировать процесс репликации данных в случае нестабильного соединения между серверами.

`mtargets`

Обработка данных по многоэлементным отчётам.

`count <n>`

Выполнение `n` итераций и остановка. Используется для поиска и устранения неполадок.

`autotune`

Если флаг установлен, то выполняется автоматическая перегруппировка файлов БД в случае, если число источников данных файла не соответствует числу источников данных в определении целевого объекта.

`lwupdate <pwrite | mmap | fullmmap>`

Флаг включает режим высокопроизводительного обновления БД, при котором файл постоянно открыт, а его метаданные хранятся в памяти. Этот режим может применяться для очень большого числа интерфейсов. Перед использованием такой функциональной возможности НЕОБХОДИМО проконсультироваться с представителем ООО «Венто Технологии».

`nomap`

При установленном флаге схему интерфейсов составляет `thawne.pl`, по умолчанию и при неустановленном флаге – `collector`.

`fnmp`

Сервис с установленным флагом `fnmp` запрашивает данные только о тех устройствах, для которых в свойствах соответствующих объектов указана возможность использования FNMP.

`esbevent`

Сервис с установленным флагом `esbevent` запрашивает данные только о тех устройствах, для которых в свойствах соответствующих объектов указан префикс `esb`.

`suspend-on-reconf={false|<num_of_seconds>}`

Приостановка сбора данных при проведении переконфигурации.

`<num_of_seconds>` – время (в секундах), в течение которого `collector` ожидает от `configServer` извещения об изменении конфигурации. Если в течение заданного интервала времени извещение не получено, то считается, что изменения завершены и `collector` возобновляет сбор данных. Значение `false` отключает режим ожидания, то есть `collector` проводит сбор данных, не обращая внимания на проведение изменения конфигурации. По умолчанию режим приостановки сбора данных отключен (`false`).

Лог-файл

Каждое измерение, выполненное процессом `collector`, записывается в следующем формате:

```
[time] Retrieved data for <path>(inst)[error] : <ds1>@timestamp[low_bound-upper_bound],<ds2>,...,<dsn>
```

`time` Время записи в формате `[dd-Mon-yyyy hh:mm:ss :pid]`.

<code>path</code>	Расположение объекта в дереве конфигурации.
<code>inst</code>	Привязка элемента, определяемая схемой. Может быть пустой.
<code>error</code>	Классификация сообщения (<code>Error</code> , <code>Warn</code> , <code>Info</code> или <code>Debug</code>).
<code>timestamp</code>	Реальное время в миллисекундах, записываемое в базу данных.
<code>low_bound</code>	В дробных секундах. Показывает, когда началось измерение.
<code>upper_bound</code>	В дробных секундах. Показывает, когда закончилось измерение.
<code>dsl..dsn</code>	Источники данных, такие как <code>ifInOctets</code> , <code>ifOutOctets</code> и т.д. Значение <code>U</code> показывает, что по какой-то причине данные от источника отсутствуют. Возможными причинами этого могут быть сбой в сети, проблемы с устройством и т.д. Все источники данных описаны в шаблонах объектов в <code>/usr/local/lancelot/Devices/Device Name</code> .

Например:

```
[14-Sep-2011 13:45:16:23166] Retrieved data for
@/Devices/Interfaces/10.0.0.1/bri0_0_1(2) [0.31%] :
1264586560@1126698316818[1126698316.81177-
1126698316.8238],1630681937,9,1,0,0,1,27085441,16622472,0,189758653
```

14.3.8. Удаленная настройка ПМ Нувола Топаз (`konfigd`)

Системный процесс `konfigd` обеспечивает возможность удаленной настройки ПМ Нувола Топаз с помощью Web-интерфейса.

Параметры

`check`

Команда, выполняемая утилитой `overlord.pl` при получении команды `check` (запуск остановленного системного процесса).

`pid_regexp_string`

Параметр, используемый утилитой `overlord.pl` для идентификации процесса.

`restart`

Команда, выполняемая утилитой `overlord.pl` для перезапуска сервиса.

`start`

Команда, выполняемая утилитой `overlord.pl` для запуска сервиса.

`stop`

Команда, выполняемая утилитой `overlord.pl` для остановки сервиса.

14.3.9. Проверка лицензии (`licdaemon`)

Системный процесс `licensed`, основанный на процессе `licdaemon.pl`, выполняет проверку корректности лицензии.

Параметры

`port`

14. Справочники

Номер используемого порта.

`check-interval`

Интервал проверки валидности лицензии. Не рекомендуется менять без консультации с представителями ООО «Венто Технологии».

14.3.10. Мониторинг системных процессов ПМ Нуволы Топаз (monit)

Служебный системный процесс `monit` осуществляет мониторинг работы системных процессов ПМ **Нуволы Топаз** и обеспечивает их автоматическое восстановление в случае сбоя.

Среднее время восстановления работоспособности системных процессов – 3 минуты.

14.3.11. Формирование оповещений (notifs)

Системный процесс `notifs`, основанный на процессе `notifier.py`, в соответствии с заданными администратором настройками формирует оповещения о возникшей проблеме одним или несколькими способами: email, sms, прерывание snmp или запись в лог.

Параметры

`digest-interval`

Периодичность отправки дайджестов по электронной почте (используется только методом оповещений [Email](#)^[288]). Значение параметра – интервал в секундах.

`disable`

Шаблоны оповещений, игнорируемые системным процессом `notifs`. Значение параметра – регулярное выражение для отбора шаблонов оповещений по имени. Оповещения по отобранным шаблонам формироваться не будут.

14.3.12. Подготовка отчетов (querymaker)

Системный процесс `querymaker`, основанный на процессе `querymaker.pl`, выполняет подготовку отчетов по собранным данным: сканирует определённые поддеревья или всю наблюдаемую сеть и формирует отчёты.

На процессе `querymaker` также основан сервис [pinger](#)^[278].

Параметры

`now`

Признак блокировки расписания выполнения сервиса. Если признак установлен или расписание не задано, то сервис работает по расписанию, заданному в конфигурационном файле.

`cron`

Расписание выполнения процесса.

В случае разветвлённой сети рекомендуется указывать конкретные пути хранения отчётов, чтобы у `querymaker.pl` не было необходимости искать их по всему дереву (по умолчанию сервис начинает поиск с корневого каталога дерева конфигурации).

14.3.12.1. pinger

Системный процесс `pinger`, основанный на процессе `querymaker.pl`, выполняет асинхронный пинг списка устройств.

14.3.13. API для Web-интерфейса (rest.pl)

Системный процесс `rest_api`, основанный на процессе `rest.pl`, обеспечивает взаимодействие Web-интерфейса с ПМ Нуволы Топаз.

Системный процесс `rest_long_api`, основанный на процессе `rest.pl`, обеспечивает передачу и обработку длинных запросов (например, на поиск) от Web-интерфейса в ПМ Нуволы Топаз.

14.4. Типы сигналов

В настоящем справочнике описаны параметры типов сигналов, используемых модулем Нуволы Аналитика.

14.4.1. Параметры сигнала типа Gauge

Сигнал типа **Gauge** характеризует уровень использования/загрузки интерфейса или другого ресурса (общий принцип: чем больше значение сигнала, тем "хуже" его статус).

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Источник данных

Имя источника данных в терминологии SNMP.

Диапазон

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (`avg`) – выбирает из базы данных значения за интервал **Диапазон** (`range`) и вычисляет среднее значение;
- ❖ **минимальное** (`min`) – выбирает из базы данных минимальное значение за интервал **Диапазон** (`range`);
- ❖ **максимальное** (`max`) – выбирает из базы данных максимальное значение за интервал **Диапазон** (`range`);
- ❖ **последнее** (`last`) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (`range`). Это значение может быть только числом;
- ❖ **текущее** (`cur`) – выбирает из базы данных последнее значение за интервал **Диапазон** (`range`). Это значение не обязательно будет числом;

14. Справочники

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция `last` возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция `cur` возвращает значение **NAN**;

- ❖ **последнее известное** (`ldef` – от `last defined`) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (*range*). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

В режиме ввода локальных значений список для выбора также содержит вариант **undef**, который эквивалентен неопределенному значению, т.е. указывает на использование глобального значения параметра.

Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

Нижний порог

Пороговое значение между диапазонами синего и зелёного статусов сигнала. Если значение сигнала меньше **Нижнего порога**, то сигналу присваивается синий статус, если больше или равно – зелёный.

Нижний порог задётся в абсолютных единицах. Допустимы приставки *k*, *m*, *g*. По умолчанию – 0. Как правило, **Нижний порог** не задают.

Метка для нижнего порога

Описание диапазона значений для синего статуса сигнала.

Данное и последующие поля **Метка...** предназначены для ввода кратких описаний диапазонов значений, соответствующих статусам сигнала. Описания отображаются в консоли событий в поле **Метка**.

Нормальный порог

Пороговое значение между диапазонами зелёного и жёлтого статусов сигнала. Если значение сигнала меньше **Нормального порога**, то сигналу присваивается зелёный статус, если больше или равно – жёлтый.

Нормальный порог задаётся в абсолютных единицах или процентах от **Максимального порога**. Допустимы приставки *k*, *m*, *g*. Обязательный параметр.

Метка для нормального порога

Описание диапазона значений для зелёного статуса сигнала.

Высокий порог

Пороговое значение между диапазонами жёлтого и красного статусов сигнала. Если значение сигнала меньше **Высокого порога**, то сигналу присваивается жёлтый статус, если больше или равно – красный.

Высокий порог задаётся в абсолютных единицах или процентах от **Максимального порога**. Допустимы приставки *k*, *m*, *g*. Обязательный параметр.

Метка для высокого порога

Описание диапазона значений для жёлтого статуса сигнала.

Максимальный порог

Пороговое значение между диапазонами красного и бордового статусов сигнала. Если значение сигнала меньше **Максимального порога**, то сигналу присваивается красный статус, если больше или равно – бордовый.

Максимальный порог задаётся в абсолютных единицах. Допустимы приставки k, m, g. По умолчанию равен значению параметра `Bandwidth` (пропускная способность) целевого объекта. Если параметр `Bandwidth` не определен, то по умолчанию **Максимальный порог** принимается равным 100. Как правило, **Максимальный порог** не задают.

Метка для максимального порога

Описание диапазона значений для красного статуса сигнала.

Метка выше максимального порога

Описание диапазона значений для бордового статуса сигнала.

Лидер

Признак [лидирующего](#)^[35] сигнала – показывает приоритет сигнала в наборе сигналов. Возможные значения:

нет (по умолчанию) – сигнал не является лидирующим;

да – сигнал является лидирующим.

14.4.2. Параметры сигнала типа `InvertGauge`

Сигнал типа `InvertGauge` характеризует уровень неиспользованного ресурса, например тонера для принтера, объема свободного места на диске (общий принцип: чем меньше значение сигнала, тем "хуже" его статус).

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Источник данных

Имя источника данных в терминологии SNMP.

Диапазон

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (`avg`) – выбирает из базы данных значения за интервал **Диапазон** (`range`) и вычисляет среднее значение;
- ❖ **минимальное** (`min`) – выбирает из базы данных минимальное значение за интервал **Диапазон** (`range`);
- ❖ **максимальное** (`max`) – выбирает из базы данных максимальное значение за интервал **Диапазон** (`range`);
- ❖ **последнее** (`last`) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (`range`). Это значение может быть только числом;
- ❖ **текущее** (`cur`) – выбирает из базы данных последнее значение за интервал **Диапазон** (`range`). Это значение не обязательно будет числом;

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция `last` возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция `cur` возвращает значение **NAN**;

- ❖ **последнее известное** (`ldef` – от `last defined`) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (*range*). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

В режиме ввода локальных значений список для выбора также содержит вариант **undef**, который эквивалентен неопределенному значению, т.е. указывает на использование глобального значения параметра.

Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

Высокий порог

Пороговое значение между диапазонами синего и зелёного статусов сигнала. Если значение сигнала больше или равно **Высокого порога**, то сигналу присваивается синий статус, если меньше – зелёный.

Высокий порог задаётся в абсолютных единицах. Допустимы приставки *k*, *m*, *g*. По умолчанию – **100**.

Метка для высокого порога

Описание диапазона значений для синего статуса сигнала.

Данное и последующие поля **Метка...** предназначены для ввода кратких описаний диапазонов значений, соответствующих статусам сигнала. Описания отображаются в консоли событий в поле **Метка**.

Нормальный порог

Пороговое значение между диапазонами зелёного и жёлтого статусов сигнала. Если значение сигнала больше или равно **Нормального порога**, то сигналу присваивается зелёный статус, если меньше – жёлтый.

Нормальный порог задаётся в абсолютных единицах или процентах от **Высокого порога**. Допустимы приставки *k*, *m*, *g*. Обязательный параметр.

Метка для нормального порога

Описание диапазона значений для зелёного статуса сигнала.

Нижний порог

Пороговое значение между диапазонами жёлтого и красного статусов сигнала. Если значение сигнала больше или равно **Нижнего порога**, то сигналу присваивается жёлтый статус, если меньше – красный.

Нижний порог задётся в абсолютных единицах или процентах от **Высокого порога**. Допустимы приставки *k*, *m*, *g*. Обязательный параметр.

Метка для нижнего порога

Описание диапазона значений для жёлтого статуса сигнала.

Минимальный порог

Пороговое значение между диапазонами красного и бордового статусов сигнала. Если значение сигнала больше или равно **Минимального порога**, то сигналу присваивается красный статус, если меньше – бордовый.

Минимальный порог задаётся в абсолютных единицах. Допустимы приставки *k*, *m*, *g*. По умолчанию – 0. Как правило, **Минимальный порог** не задают.

Метка для минимального порога

Описание диапазона значений для красного статуса сигнала.

Метка ниже минимального

Описание диапазона значений для бордового статуса сигнала.

Лидер

Признак [лидирующего](#)^[35] сигнала – показывает приоритет сигнала в наборе сигналов. Возможные значения:

- нет** (по умолчанию) – сигнал не является лидирующим;
- да** – сигнал является лидирующим.

14.4.3. Параметры сигнала типа SmartGauge

Сигнал типа **SmartGauge** характеризует состояние сигнала в соответствии с заданными диапазонами значений. Администратор может настроить произвольное количество диапазонов (в этом принципиальное отличие от сигнала типа **Gauge** и **InvertGauge**).

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Источник данных

Имя источника данных в терминологии SNMP.

Диапазон

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (*avg*) – выбирает из базы данных значения за интервал **Диапазон** (*range*) и вычисляет среднее значение;
- ❖ **минимальное** (*min*) – выбирает из базы данных минимальное значение за интервал **Диапазон** (*range*);
- ❖ **максимальное** (*max*) – выбирает из базы данных максимальное значение за интервал **Диапазон** (*range*);
- ❖ **последнее** (*last*) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (*range*). Это значение может быть только числом;
- ❖ **текущее** (*cur*) – выбирает из базы данных последнее значение за интервал **Диапазон** (*range*). Это значение не обязательно будет числом;

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция *last* возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция *cur* возвращает значение **NAN**;

14. Справочники

- ❖ **последнее известное** (`ldef` – от last defined) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (`range`). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

В режиме ввода локальных значений список для выбора также содержит вариант **undef**, который эквивалентен неопределенному значению, т.е. указывает на использование глобального значения параметра.

Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

Нижний порог

Данное значение игнорируется. Сигналу присваивается синий статус, если значение сигнала НЕ находится в диапазонах для зеленого, желтого или красного статусов.

Метки для нижних порогов

Описания диапазонов значений для синего статуса сигнала.

Данное и последующие поля **Метка...** предназначены для ввода кратких описаний диапазонов значений, соответствующих статусам сигнала. Описания отображаются в консоли событий в поле **Метка**. Описания вводятся в той же последовательности, что и диапазоны в соответствующем поле, и разделяются запятой и пробелом.

Нормальный порог

Диапазоны значений для зеленого статуса сигнала.

При настройке диапазонов следует руководствоваться правилами:

- ❖ диапазон вводится в формате: **[a:b]**, причем нижняя граница принадлежит диапазону ($a \leq x < b$);
- ❖ для каждого статуса можно задать несколько диапазонов, перечислив их через запятую;
- ❖ диапазон можно задать в абсолютных единицах или в процентах от **Максимального порога**. В последнем случае символ % указывается однократно после списка диапазонов, например: **[10:20],[70:80]%**;
- ❖ для указания бесконечности следует использовать обозначения **-inf** и **inf**. Один из диапазонов обязательно должен быть не ограничен справа, например: **[90:inf]%**.

Метки для нормальных порогов

Описания диапазонов значений для зеленого статуса сигнала.

Незначительный порог (Threshold minor)

Диапазоны значений для желтого статуса сигнала.

Метки для минорных порогов

Описания диапазонов значений для желтого статуса сигнала.

Высокий порог

Диапазоны значений для красного статуса сигнала.

Метки для высоких порогов

Описания диапазонов значений для красного статуса сигнала.

Максимальный порог

Значение, используемое для вычисления процентных величин, заданных в предыдущих полях. По умолчанию максимальный порог равен значению параметра `Bandwidth` (пропускная способность) целевого объекта. Если параметр `Bandwidth` не определен, то по умолчанию **Максимальный порог** принимается равным 100. Как правило, параметр **Максимальный порог** не указывают. Возможно использование приставок `k`, `m`, `g`.

Метки для максимальных порогов

Не используется.

Лидер

Признак [лидирующего](#)^[35] сигнала – показывает приоритет сигнала в наборе сигналов. Возможные значения:

нет (по умолчанию) – сигнал не является лидирующим;

да – сигнал является лидирующим.

14.4.4. Параметры сигнала типа **Availability**

Сигнал типа **Availability** показывает доступность целевого объекта, а именно: интерфейса, и используется для выявления недоступных интерфейсов.

Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

Источник данных

Имя источника данных в терминологии SNMP.

Диапазон

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд.

Функция

Функция для определения значения сигнала.

Доступны следующие функции:

- ❖ **среднее** (`avg`) – выбирает из базы данных значения за интервал **Диапазон** (`range`) и вычисляет среднее значение;
- ❖ **минимальное** (`min`) – выбирает из базы данных минимальное значение за интервал **Диапазон** (`range`);
- ❖ **максимальное** (`max`) – выбирает из базы данных максимальное значение за интервал **Диапазон** (`range`);
- ❖ **последнее** (`last`) – выбирает из базы данных последнее известное значение за интервал **Диапазон** (`range`). Это значение может быть только числом;
- ❖ **текущее** (`cur`) – выбирает из базы данных последнее значение за интервал **Диапазон** (`range`). Это значение не обязательно будет числом;

14. Справочники

Например, если в течение интервала **Диапазон** (*range*) целевой объект стал недоступным, то функция `last` возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция `cur` возвращает значение **NAN**;

- ❖ **последнее известное** (`ldef` – от last defined) – выбирает из базы данных RRD последнее известное значение за интервал **Диапазон** (*range*). Если за указанный интервал значения отсутствуют или они не определены, то функция возвращает значение **NAN**.

В режиме ввода локальных значений список для выбора также содержит вариант **undef**, который эквивалентен неопределенному значению, т.е. указывает на использование глобального значения параметра.

Эталон

Эталонное значение сигнала, используемое [алгоритмом определения статуса сигнала](#)^[34]. Настоятельно рекомендуется задавать данный параметр. Например, для статуса трафика (`ifOperStatus`) эталонное значение равно 100. Эталонное значение можно узнать опытным путем – посмотреть значение сигнала через Web-интерфейс.

Описание для 'хороший'

Текст, отправляемый в оповещении при переходе сигнала в зеленый статус.

Описание для 'плохой'

Текст, отправляемый в оповещении при переходе сигнала в красный статус.

Лидер

Признак [лидирующего](#)^[35] сигнала – показывает приоритет сигнала в наборе сигналов. Возможные значения:







нет (по умолчанию) – сигнал не является лидирующим;


да – сигнал является лидирующим.

14.5. Статусы сигналов

Статусу сигнала соответствует цвет и числовая константа, выполняющая роль идентификатора.

В ПМ **Нувола Топаз** для сортировки статусов сигналов по степени критичности используются приоритеты статусов, которые задаются параметром `states_priorities` в файле `usr/local/lancelot/conf/svgd.conf`. Ниже в таблице статусы сигналов отсортированы от наилучшего (серый) к наихудшему (белый) в соответствии с предустановленными настройками.

Цвет статуса	Идентификатор статуса	Примечание
 серый	5	
 синий	0	
 зелёный	1	
 жёлтый	2	
 красный	3	
 бордовый	4	

	чёрный	6	Чёрный статус означает, что целевой объект отсутствует в дереве конфигурации. Используется на информационных панелях.
	белый	7	В Web-интерфейсе для обозначения белого статуса используется также иконка  (серая шестерёнка).

14.6. Классы оповещений

В ПК Нувола реализован один класс оповещений **Simple**.

14.6.1. Параметры класса оповещений Simple

Класс оповещений **Simple** имеет следующие параметры:

Имя

Имя класса оповещения. Параметр недоступен для редактирования.

Тип

Тип класса оповещения, *Simple*. Параметр недоступен для редактирования.

Сигналы

Список названий сигналов, по которым формируются оповещения.

Исключительный

Настройка событий для отправки оповещений. Возможные значения:

- on** – оповещение отправляется только для первого сработавшего сигнала из набора привязанного к целевому объекту;
- off** – оповещения отправляются по каждому сработавшему сигналу.

Описание

Комментарий.

Статус 'плохой'

Множество "плохих" статусов сигналов.

По клику мыши в поле появляется список идентификаторов [статусов сигналов](#)²⁸⁶, включенных в множество.

Статус 'игнорировать'

Множество статусов сигналов, которые игнорируются и не требуют формирования оповещений.

Способы ввода значений аналогичны предыдущему параметру.

Статус 'хороший'

Множество "хороших" статусов сигналов.

Способы ввода значений аналогичны предыдущему параметру.

Включено

Признак отправки повторных оповещений. Возможные значения:

on – оповещение отправляется, если статус сигнала изменился с "хорошего" на "плохой";

off – отправка оповещения не производится.

Для уменьшения количества повторных оповещений можно настроить интервал между их отправкой. Интервал вычисляется по экспоненциальной формуле:

$I_{n+1} = \text{Backoff.base} * \text{Backoff.power}^n$, где

n – порядковый номер оповещения;

Backoff.base – базовый интервал, см ниже;

Backoff.power – вещественное неотрицательное число, см. ниже.

Козфф. увел. интервала напоминания

Вещественное неотрицательное число; по умолчанию **0**. Этот параметр используется, если **Backoff.enabled=on**.

Базовый интервал напоминания

Базовый интервал между повторными оповещениями, в секундах; по умолчанию **7200** секунд (2 часа). Этот параметр используется, если **Разрешить напоминания / Backoff.enabled=on**.

Ограничение времени напоминания

Временной интервал, по истечении которого прекращается отправка повторных оповещений о проблеме; по умолчанию **604800** секунд (1 неделя).

Отправка при любом изменении статуса

Если признак установлен (**on**), то оповещения отправляются при любом изменении статуса сигнала, т.е. множества статусов, заданные в полях **Состояния 'плохо'**, **Состояния 'хорошо'**, **Состояния 'игнорировать'**, не влияют на отработку оповещения. Предустановленная настройка – **off**.

14.7. Типы оповещений

В настоящем справочнике описаны параметры типов оповещений.

14.7.1. Параметры метода оповещений типа Email

Метод оповещений типа **Email** позволяет отправлять сообщения по электронной почте. Сообщение доставляется через локальный sendmail (или через сервер, указанный в mailhost целевого объекта), либо через почтовый сервер, адрес которого указан в переменной окружения LANCELOT_MAILHOST.

Тело письма создаётся по шаблону (`alerts/single.tpl`), в который передаются параметры сигнала.

Механизм объединения оповещений Alerts Digesting даёт возможность сократить количество отправляемых писем. Вместо многочисленных писем по каждому сработавшему сигналу, формируются дайджесты – письма с пакетами сообщений. Периодичность отправки писем задается параметром `digest-interval` системного процесса [notifs](#)^[278].

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **Email** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

- on** – метод включен, оповещения формируются;
- off** – метод отключен, оповещения не формируются.

Получатели e-mail | `email.to`

Адрес электронной почты, на который направляется оповещение. Можно указать несколько адресов через запятую.

Отправитель e-mail | `email.from`

Адрес отправителя, который отображается в письме в поле **От / From**. По умолчанию используется `alerts@hostname`.

Хост сервера e-mail | `email.host`

IP-адрес или имя хоста, используемого в качестве шлюза электронной почты.

Хост Нувола Топаз | `lancelot.host`

IP-адрес или имя сервера Нувола.

Порт Нувола Топаз | `lancelot.port`

Порт, используемый модулем **Нувола Топаз**.

Игнорировать | `ignored`

Множество объектов, по которым рассылка оповещений не производится. Указывается полный путь на объект (устройство или целевой объект) и его имя или маска имени. Можно указать несколько путей через запятую. При срабатывании сигнала от целевого объекта из заданного множества, письма не отправляются.

 **Пример**

Предположим, в папке `/Profile1/Moscow/HeadOffice` размещены три устройства – компьютеры `Workstation1`, `Workstation2` и `Workstation3`.

Если **Игнорировать** (`ignored`) = `/Profile1/Moscow/HeadOffice/Workstation`, то при срабатывании сигнала от целевого объекта на любом из трех компьютеров оповещения отправляться не будут.

Если **Игнорировать** (`ignored`) = `/Profile1/Moscow/HeadOffice/Workstation1`, то оповещения не будут отправляться при срабатывании сигналов от целевых объектов, принадлежащих устройству `Workstation1`.

Интервал отправки | `send.interval`

Интервал времени (в секундах) для отправки группы оповещений (только для сжатых оповещений).

Компоновать сообщения | `digest`

Включение механизма Alerts Digesting. Возможные значения:

- **no** – каждое письмо содержит только одно сообщение;
- **yes** – в одном письме объединяются несколько сообщений.

14.7.2. Параметры метода оповещений типа **EsbNotify**

Метод оповещений типа **EsbNotify** позволяет посредством шины ESB отправлять сообщения модулю `Nuvola CEP`.

Имя класса

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **EsbNotify** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

on – метод включен, оповещения формируются;

off – метод отключен, оповещения не формируются.

14.7.3. Параметры метода оповещений типа **Exec**

Метод оповещений типа **Exec** позволяет инициировать запуск сторонней программы. При этом ей не передаются значения параметров сигнала.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **Exec** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

on – метод включен, оповещения формируются;

off – метод отключен, оповещения не формируются.

Путь | `script.path`

Путь к выполняемой программе.

Exec.params | `exec.params`

Значения параметров программы, с которыми она должна быть запущена. Текстовое значение параметра следует заключать в одинарные кавычки. Формат ввода проиллюстрирован на примерах:

--type=server --message='Disk Space problem on first server'

--type=station --message='Workstation free space is low'

14.7.4. Параметры метода оповещений типа **Log**

Метод оповещений типа **Log** позволяет создавать записи в лог-файле.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **Log** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

on – метод включен, оповещения формируются;

off – метод отключен, оповещения не формируются.

Путь | `log.file`

Путь и имя лог-файла. Можно ввести абсолютный путь к файлу (начиная с '/'). По умолчанию файл `events.log` создается в стандартной директории модуля **Нувола** **Топаз** `/usr/local/lancelot/var/lancelot-logs`.

Степень детальности | `log.severity`

Уровень важности, с которым сообщение записывается в лог-файл. Возможные значения:

- **Error** – ошибка;
- **Warning** – предупреждение;
- **Info** – информация;
- **Debug** – отладочное сообщение.

14.7.5. Параметры метода оповещений типа MySQL

Метод оповещений типа **MYSQL** позволяет сохранять сообщения в базе данных MySQL. Перед использованием этого метода следует создать базу данных MySQL и настроить её.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **MYSQL** и для редактирования недоступно.

Разрешить работу MYSQL | `enabled`

Признак активности метода. Возможные значения:

- on** – метод включен, оповещения формируются;
- off** – метод отключен, оповещения не формируются.

Сетевой адрес сервера MYSQL | `mysql.ip`

IP-адрес базы данных.

Порт сервера MYSQL | `mysql.port`

Порт для обращения к базе данных.

Имя базы данных MYSQL | `mysql.dbname`

Имя базы данных.

Имя пользователя MYSQL | `mysql.user`

Имя пользователя для обращения к базе данных.

Пароль сервера MYSQL | `mysql.pass`

Пароль пользователя. Хранится в зашифрованном виде.

14.7.6. Параметры метода оповещений типа SSH

Метод оповещений типа **SSH** позволяет передать команду на удаленный ресурс по протоколу SSH.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **SSH** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

on – метод включен, оповещения формируются;

off – метод отключен, оповещения не формируются.

В режиме ввода локальных значений список для выбора также содержит вариант **undef**, который эквивалентен неопределенному значению, т.е. указывает на использование глобального значения параметра.

Адрес IP | `ssh.host`

Имя хоста или IP-адрес для подключения к удаленному ресурсу.

Имя пользователя | `ssh.username`

Имя пользователя для подключения к удаленному ресурсу.

Пароль | `ssh.pass`

Пароль для подключения к удаленному ресурсу.

Команда | `ssh.exec`

Команда для выполнения на удалённом ресурсе.

14.7.7. Параметры метода оповещений типа SMS

Метод оповещений типа **SMS** позволяет отправлять SMS-сообщения через SMS-шлюз. Используется формат SMS2Email, позволяющий отправлять и получать SMS-сообщения без использования мобильного телефона.

Тело письма задаётся по шаблону, `alerts/single.tpl`, в шаблон передаются параметры сигнала. Тема создаётся автоматически, на основе переменных сигнала.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **SMS** и для редактирования недоступно.

Разрешить работу сервера SMS | `enabled`

Признак активности метода. Возможные значения:

on – метод включен, оповещения формируются;

off – метод отключен, оповещения не формируются.

Хост сервера SMS | `sms.host`

IP-адрес или имя шлюза передачи сообщений SMS2Email.

Сервер SMS | `sms.mailhost`

Электронный адрес шлюза передачи сообщений SMS2Email, который отображается в письме в поле **От / From**. Например: `localhost@127.0.0.1`.

Получатель SMS | `sms.to`

Номер мобильного телефона получателя. Можно ввести несколько номеров через запятую.

14.7.8. Параметры метода оповещений типа Syslog

Метод оповещений типа **Syslog** позволяет создавать записи в системном журнале.

Имя

Имя метода оповещений. Поле для редактирования недоступно.

Тип

Тип оповещений. Поле имеет значение **Syslog** и для редактирования недоступно.

Включено | `enabled`

Признак активности метода. Возможные значения:

- on** – метод включен, оповещения формируются;
- off** – метод отключен, оповещения не формируются.

Получатель Syslog | `syslog.destination`

Размещение системного журнала, путь к файлу.

Приоритет Syslog | `syslog.priority`

Приоритет, который присваивается записи в журнале. Возможные значения:

- **0** – система недоступна;
- **1** – требуется немедленное вмешательство;
- **2** – критическая ситуация;
- **3** – ошибка;
- **4** – предупреждение;
- **5** – уведомление;
- **6** – информационное сообщение;
- **7** – сообщения, используемые для отладки.

Принадлежность Syslog | `syslog.facility`

Принадлежность системного журнала. Используется для разделения сообщений в зависимости от каких-либо характеристик источника сигнала (территориально, по типу устройства и т.д.). Рекомендуется использовать значения **local0** – **local7**.

14.8. Элементы информационных панелей

В настоящем разделе рассмотрены особенности настройки простых элементов и виджетов информационных панелей, описаны специфические параметры некоторых виджетов.

Параметры, универсальные для всех элементов, приведены в разделе [Обзор параметров элементов информационных панелей](#)¹⁷⁷.

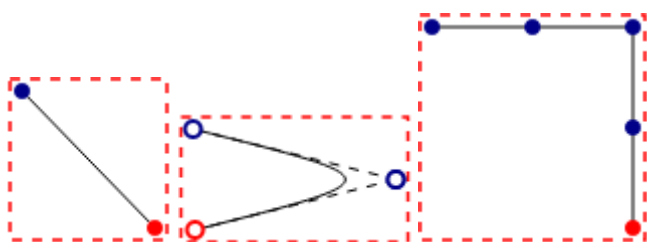
14.8.1. Простые элементы для рисования линий

Для рисования линий реализованы три простых элемента: **Линия**, **Кривая линия**, **Прямоугольные ломаные линии**. Цвет линии изменяется в зависимости от статусов привязанных сигналов.

Иконки для рисования линий на канве находятся на вкладке **Виджеты** в разделе **Простые**:



Соответственно, размещенные на канве элементы выглядят следующим образом:



Красным цветом выделена активная точка.

Вне зависимости от типа линии для изменения ее конфигурации можно использовать как мышь, так и клавиатуру:

- ❖ перемещение активной точки (красного цвета) – удерживая клавишу **Ctrl**, нажимать клавишу со стрелкой, соответствующую выбранному направлению;
- ❖ переключение активных точек – удерживая клавишу **Shift**, нажимать клавишу со стрелкой. Порядок переключения точек изменяется в зависимости от нажатых клавиш со стрелками. Так, нажатие клавиш **«Вверх»** или **«Вправо»** циклически переключает точки в последовательности «Т1, Т2, Т3, ... Тn, Т1, ...», а нажатие клавиш **«Влево»** или **«Вниз»** циклически переключает точки в последовательности «Тn, Тn-1, ... Т1, Тn, ...».

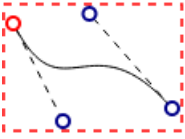
Рисование прямой линии (элемента **Линия**) не представляет сложности. Поэтому далее рассматриваются особенности рисования элементов **Кривая линия** и **Прямоугольные ломаные линии**.

14.8.1.1. Кривая линия

Чтобы добавить на канву элемент **Кривая линия**, следует:

1. со вкладки **Виджеты** из раздела **Простые** мышью перетащить иконку  на канву;

- с помощью мыши выделить элемент. На линии появятся круглые метки, обозначающие концы линии и направляющие;
- для искривления линии потянуть мышью за любую метку. Кривизна задается с помощью двух направляющих.



14.8.1.2. Прямоугольные ломаные линии

Элемент **Прямоугольные ломаные линии** представляет собой набор вертикальных и горизонтальных отрезков перпендикулярных друг другу, причем для любой пары отрезков концевая точка первого является началом второго.

Чтобы добавить на канву элемент **Прямоугольные ломаные линии**, следует со вкладки **Виджеты** из



раздела **Простые** мышью перетащить иконку на канву. Будет создана соединительная линия из двух перпендикулярных отрезков. При последующем выборе этой линии будут выделены точки сгиба, концевые точки и середины отрезков. При помощи данных точек можно изменять конфигурацию линии.

Алгоритм создания линии: если изменение по оси Y больше изменения по оси X, через концевую точку пройдет вертикальная линия, а через начальную – горизонтальная, в противном случае – наоборот. Алгоритм меняется на обратный при нажатии и удержании клавиши **Shift**.

Для изменения конфигурации линии её надо выделить и переместить одну из точек. Концевые точки и точки изгиба перемещаются в любом направлении. Срединные точки отрезков перемещаются только по горизонтали или вертикали, в зависимости от типа отрезка. При перемещении срединной точки крайнего отрезка появляется новый отрезок, перпендикулярный перемещаемому, с сохранением положения крайней концевой точки соединительной линии.

При совпадении двух и более соседних отрезков соединительной линии они трансформируются в один. Данного эффекта также можно добиться при перемещении точек при помощи клавиатуры.

Вид угла между линиями задается с помощью параметра **Соединение линий** ²⁹⁶ в разделе **Форма** на вкладке **Параметры**.

14.8.1.3. Параметры элементов

В настоящем разделе рассмотрены специфические параметры линий.

Раздел **Позиция и размер**

Параметры, определяющие координаты концевых и активной (красной) точек элементов:

x1

Горизонтальная координата точки начала линии.

y1

Вертикальная координата точки начала линии.

14. Справочники

x2

Горизонтальная координата точки конца линии.

y2

Вертикальная координата точки конца линии.

x

Горизонтальная координата активной (красной) точки.

y

Вертикальная координата активной (красной) точки.

Раздел Форма

Раздел **Форма** содержит параметры, определяющие внешний вид элемента.

Ширина рамки

Толщина линии.

Фон

Цвет заливки по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Цвет линий

Цвет рамки (если она имеется) по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Градиент

Использование градиентной заливки элемента. Возможные значения:

False – градиент не используется;

True – градиент используется.

Стиль линии

Настройка стиля линии. Возможные значения: **сплошная**, **пунктирная**, **штриховая**, **штрихпунктирная**.

Конец линии

Внешний вид конца линии. Возможные значения: **квадратное окончание**, **закруглённое окончание**.

Соединение линий

Настройка типа соединения линий. Возможные значения:

квадратное;

закруглённое;

срезанное.

Раздел Общие

Изменять цвет границы

Необходимость окрашивать границу элемента в цвет привязанного сигнала. Возможные значения:

False (по умолчанию) – граница имеет постоянный цвет, заданный параметром Цвет линий в разделе Форма;

True – цвет границы элемента изменяется в соответствии с цветом сигнала.

Изменять цвет заливки

Необходимость окрашивать элемент в цвет привязанного сигнала. Возможные значения:

False – элемент имеет постоянный цвет, заданный параметром Фон в разделе Форма;

True (по умолчанию) – цвет границы элемента изменяется в соответствии с цветом сигнала.

☐ Примеры элементов для разных значений параметров Изменять цвет границы и Изменять цвет заливки

Изменять цвет границы	Изменять цвет заливки	Внешний вид элементов
True	False	
False	True	
True	True	
False	False	

14.8.2. Простой элемент Прямоугольник

Простой элемент **Прямоугольник** может использоваться в качестве визуальной формы в дизайне информационной панели, а также как индикатор статуса сигнала – за счет изменения цвета заливки элемента.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



14. Справочники

На канве элемент принимает вид в соответствии с настройками, заданными на вкладке **Параметры**.

Далее рассмотрены специфические параметры элемента.

Раздел Форма

Раздел **Форма** содержит параметры, определяющие внешний вид элемента.

Скругление угла по x

Горизонтальная ось эллипса закругления углов рамки. Если величина закругления больше половины длины стороны рамки, то будет сделано закругление до середины этой стороны.

Скругление угла по y

Вертикальная ось эллипса закругления углов рамки. Если величина закругления больше половины длины стороны рамки, то будет сделано закругление до середины этой стороны.

Ширина рамки

Толщина линии.

Непрозрачность (%)

Степень прозрачности фона. Возможные значения: от **0** (прозрачный) до **100** (непрозрачный).

Фон

Цвет заливки по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Цвет линий

Цвет рамки (если она имеется) по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Градиент

Использование градиентной заливки элемента. Возможные значения:

False – градиент не используется;

True – градиент используется.

Стиль линии

Настройка стиля линии. Возможные значения: **сплошная**, **пунктирная**, **штриховая**, **штрихпунктирная**.

Соединение линий

Настройка типа соединения линий. Возможные значения:

квадратное;

закруглённое;

срезанное.

Раздел Общие

Изменять цвет границы

Необходимость окрашивать границу элемента в цвет привязанного сигнала. Возможные значения:

False (по умолчанию) – граница имеет постоянный цвет, заданный параметром Цвет линий в разделе Форма;

True – цвет границы элемента изменяется в соответствии с цветом сигнала.

Изменять цвет заливки

Необходимость окрашивать элемент в цвет привязанного сигнала. Возможные значения:

False – элемент имеет постоянный цвет, заданный параметром Фон в разделе Форма;

True (по умолчанию) – цвет границы элемента изменяется в соответствии с цветом сигнала.

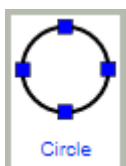
☐ Примеры элементов для разных значений параметров Изменять цвет границы и Изменять цвет заливки

Изменять цвет границы	Изменять цвет заливки	Внешний вид элементов
True	False	
False	True	
True	True	
False	False	

14.8.3. Простой элемент Эллипс

Простой элемент **Эллипс** может использоваться в качестве визуальной формы в дизайне информационной панели, а также как индикатор статуса сигнала – за счет изменения цвета заливки элемента.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



14. Справочники

Элемент, размещенный на канве, принимает вид в соответствии с настройками, заданными на вкладке **Параметры**.

Далее рассмотрены специфические параметры элемента.

Раздел Форма

Раздел **Форма** содержит параметры, определяющие внешний вид элемента.

Ширина рамки

Толщина линии.

Непрозрачность (%)

Степень прозрачности фона. Возможные значения: от **0** (прозрачный) до **100** (непрозрачный).

Фон

Цвет заливки по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Цвет линий

Цвет рамки (если она имеется) по умолчанию. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Градиент

Использование градиентной заливки элемента. Возможные значения:

False – градиент не используется;

True – градиент используется.

Стиль линии

Настройка стиля линии. Возможные значения: **сплошная**, **пунктирная**, **штриховая**, **штрихпунктирная**.

Раздел Общие

Изменять цвет границы

Необходимость окрашивать границу элемента в цвет привязанного сигнала. Возможные значения:

False (по умолчанию) – граница имеет постоянный цвет, заданный параметром **Цвет линий** в разделе **Форма**;

True – цвет границы элемента изменяется в соответствии с цветом сигнала.

Изменять цвет заливки

Необходимость окрашивать элемент в цвет привязанного сигнала. Возможные значения:

False – элемент имеет постоянный цвет, заданный параметром **Фон** в разделе **Форма**;

True (по умолчанию) – цвет границы элемента изменяется в соответствии с цветом сигнала.

☐ Примеры элементов для разных значений параметров Изменять цвет границы и Изменять цвет заливки

Изменять цвет границы	Изменять цвет заливки	Внешний вид элементов
True	False	
False	True	
True	True	
False	False	

Разрешить звуки

Возможность воспроизведения звукового сигнала в зависимости от цвета элемента.

True – при окрашивании элемента в определенный цвет воспроизводится звуковой файл в соответствии с настройкам, заданными в поле **Звук при состоянии**;

False – звуковое сопровождение отключено.

Звук при состоянии

Настройка звуковых эффектов в зависимости от статусов сигнала. См. [Настройка звукового сигнала для элемента](#)^[195].

Действия

Действия, которые будут выполняться при определенных манипуляциях пользователя во время просмотра информационной панели. См. [Настройка действий для элемента](#)^[192].

Анимация

Эффекты анимации, используемые для оживления информационной панели.

Локализация

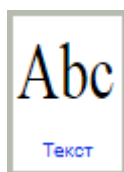
Язык Web-интерфейса (**Русский** или **Английский**), при котором элемент отображается на информационной панели. По умолчанию элемент отображается в Web-интерфейсе вне зависимости от выбранной локализации. Параметр **Локализация** можно задать также для группы элементов.

14.8.4. Простой элемент Текст

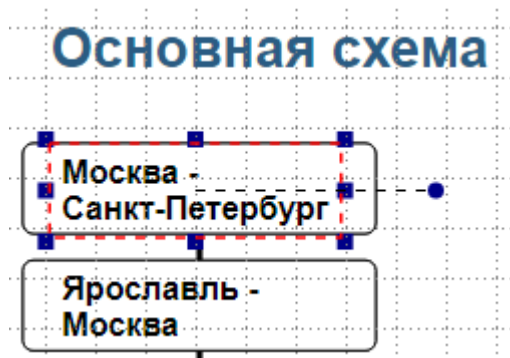
Простой элемент **Текст** позволяет вывести на информационную панель произвольную строку и значения [предопределенных переменных](#)^[341] (например, значение сигнала, наименование целевого объекта и т.п.). Элементы **Текст** используются в качестве заголовков, подписей и комментариев.

14. Справочники

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



Текст, отображаемый на элементе, можно вводить в поле **Текст** на вкладке **Параметры**.



Далее рассмотрены специфические параметры элемента.

Раздел **Позиция и размер**

Макс. ширина

Максимальная ширина текстового поля в пикселях. Если текст шире заданного значения, то содержимое обрезается до максимальной ширины, и к нему добавляется многоточие "...". Общая ширина поля с учетом многоточия не превышает максимальной ширины. Для такого элемента во всплывающей подсказке сначала выводится текст полностью, а затем – непосредственно подсказка (см. поле **Подсказка** в разделе **Общие**).

Раздел **Текст**

Текст

Текст, отображаемый на информационной панели. Может состоять из нескольких строк. Для разделения строк при вводе текста на канве следует использовать клавишу **Enter**, при вводе в поле **Текст** – тег `
`.

Допустимо использование [предопределенных переменных](#)^[341], например: `$VALUE` (значение сигнала).

Если заданы параметры **Содержимое** и **Формат**, то значение поля **Текст** отображается только в процессе загрузки информационной панели, т.е. до момента вывода строки, заданной параметрами **Содержимое** и **Формат**.

Содержимое

Аргументы для строки форматирования функции `printf`. Допустимо использование [предопределенных переменных](#)^[341].

Формат

Строка форматирования (аргумент функции `printf`). Вводится без кавычек. Например: `%s MB` – вывод значения переменной, указанной в поле **Содержимое**, и букв **MB**, обозначающих единицы измерения значения (мегабайты).

Шрифт

Параметры шрифта можно ввести в полях подраздела или в окне, которое вызывается по двойному клику на поле **Шрифт**.

Размер

Размер шрифта.

Имя

Название шрифта.

Тип

Признак курсива. Возможные значения:

Normal – обычный;

Italic – курсив.

Насыщенность

Признак толщины линий шрифта. Возможные значения:

Normal – обычный;

Light – тонкий (под Windows эквивалентно обычному);

Bold – жирный.

Подчеркивание


Признак подчеркивания. Возможные значения:

False – обычный;

True – подчеркнутый.

14. Справочники

Цвет

Цвет шрифта. Чтобы задать цвет, следует нажать кнопку  в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**)..

Выравнивание

Выравнивание текста строки. Возможные варианты: **По правому краю**, **По левому краю** (по умолчанию), **По середине**.

14.8.5. Простой элемент Изображение

Простой элемент **Изображение** позволяет отобразить на информационной панели произвольный графический файл в форматах gif, psx, ico, xpm, bmp, jpg, png, а также файл в формате svg. Если элемент привязан к сигналу, то для разных статусов сигнала можно задать специфические файлы, которые элемент будет отображать в случае получения соответствующего статуса.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:






Размещенный на канве элемент, если графический файл еще не задан, выглядит следующим образом:



Далее рассмотрены специфические параметры элемента.

Изображение

Раздел **Изображение** содержит поля для привязки графических файлов к элементу.

Изображение	
Файл	router-4-grey.png
Отображать состояние	,router-4-green.png,router-4-red.png,router
Серый	router-4-grey.png
Зеленый	router-4-green.png
Синий	
Желтый	router-4-red.png
Красный	router-4-red.png
Темно-красный	  
Черный	
Белый	

В полях размещены функциональные кнопки:



– выбор файла, хранимого на локальном диске или доступном сетевом ресурсе;



– сохранение файла, указанного в поле, на локальном диске или доступном сетевом ресурсе;

 – просмотр файла, указанного в поле, в отдельном окне:



Файл

Имя файла с изображением. Обязательный параметр. Выбранный файл отображается на канве. Следует помнить, что из анимированного gif-файла будет отображаться только первый кадр.

Отображать состояние

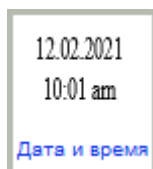
Блок полей предназначен для элементов с [назначением](#)^[184] (т.е. с привязанными сигналами). В полях с названиями статусов указываются специфические изображения, которые будут отображаться в зависимости от статуса сигнала. Поле **Отображать состояние** заполняется автоматически по мере того, как пользователь указывает графические файлы для разных статусов.

По умолчанию для всех статусов на информационной панели отображается файл, указанный в поле **Файл**. Если для каждого статуса сигнала установить свое изображение, то файл, заданный в поле **Файл**, будет отображаться только при загрузке элемента, до получения данных о статусе сигнала.

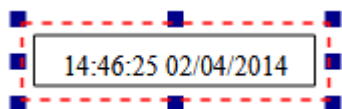
14.8.6. Простой элемент Дата и время

Простой элемент **Дата и время** позволяет отобразить текущие дату и/или время в заданном формате.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



На канве элемент принимает вид в соответствии с настройками, заданными на вкладке **Параметры**.

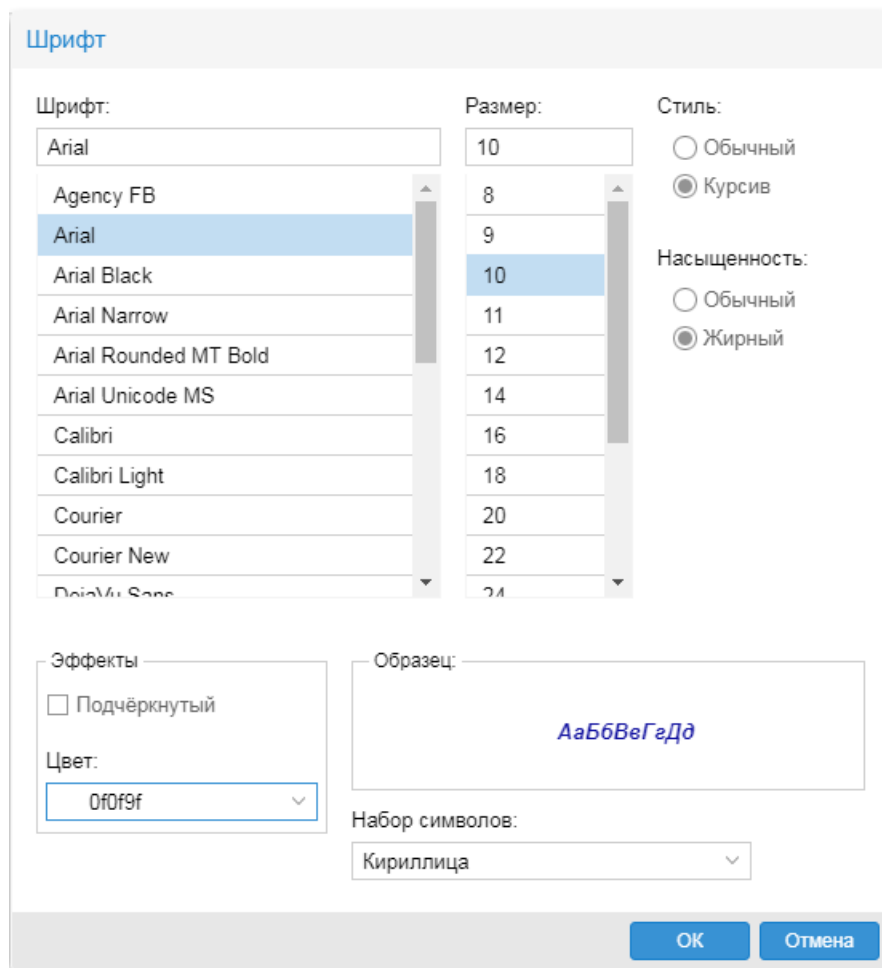


Далее рассмотрены специфические параметры элемента.

Раздел Текст

Шрифт

Параметры шрифта можно ввести в полях подраздела или в окне, которое вызывается по двойному клику на поле **Шрифт**.



Размер

Размер шрифта.

Имя

Название шрифта.

Тип

Признак курсива. Возможные значения:

Normal – обычный;

Italic – курсив.

Насыщенность

Признак толщины линий шрифта. Возможные значения:

Normal – обычный;

Light – тонкий (под Windows эквивалентно обычному);

Bold – жирный.

Подчеркивание

Признак подчеркивания. Возможные значения:

False – обычный;

True – подчеркнутый.

Цвет

Цвет шрифта. Чтобы задать цвет, следует нажать кнопку в поле ввода и выбрать цвет на палитре. Можно также вручную ввести шестнадцатеричный цветовой код либо стандартное название цвета в согласно CSS (например: **aqua**).

Выравнивание

Выравнивание текста строки. Возможные варианты: **По правому краю**, **По левому краю** (по умолчанию), **По середине**.

Дата и время

Часовой пояс

Определяет смещение времени относительно среднего времени по Гринвичу (GMT). Значение следует выбрать из раскрывающегося списка. Если на информационной панели необходимо отображать время сервера Нувола, то следует выбрать **Локальное время**. Если требуется отображать время по Гринвичу, то следует выбрать вариант с соответствующей корректировкой.

Показать дату

Настройка вывода даты: **False** – дата не выводится, **True** – дата выводится.

Показать время

Настройка вывода времени: **False** – время не выводится, **True** – время выводится.

Формат

Строка форматирования даты и времени. Может содержать текст и спецификаторы UNIX-формата. Если поле **Формат** не заполнено, то дата и время выводятся в виде **ЧЧ:ММ:СС ДД/ММ/ГГГГ**, например: **13:34:22 24/09/2020**.

Наиболее востребованные спецфикаторы:

%y – год без столетия (**00 – 99**);

%Y – год, включая столетие, например: **2020**;

%m – номер месяца (**01 – 12**);

%d – число месяца (**01 – 31**);

%a – сокращенное наименование дня недели, например: **Tue, Sun**;

%A – полное наименование дня недели, например: **Tuesday, Sunday**;

%H – час (**01 – 23**);

%M – минута (**00 – 59**);

%S – секунда (**00 – 59**);

%D – дата в формате **%m/%d/%y**, например: **12/06/09**;

%R – время в формате **%H:%M**, например: **16:27**;

%T – время в формате **%H:%M:%S**, например: **16:27:43**;

%z – часовой пояс в формате RFC-2822 (\pm zzzz), например: **+0300**.

Примеры строк форматирования и результатов вывода:

Сейчас %H:%M → **Сейчас 21:01**

%d-%m-%Y, %a → **25-06-2018, Mon**

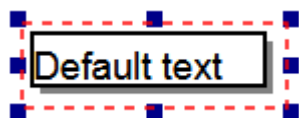
14.8.7. Виджет Бегущий текст

Виджет **Бегущий текст** позволяет отобразить на информационной панели произвольную текстовую строку.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



Размещенный на канве элемент, если графический файл еще не задан, выглядит следующим образом:







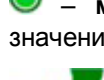

14.8.8. Виджет Объект

Виджет **Объект** отображает состояние сигнала в наглядном виде.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Простые**:



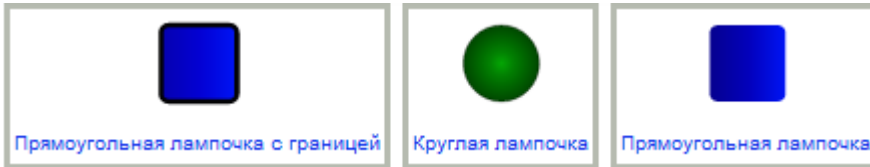
Внешний вид элемента задается параметром **Шаблон** (см. раздел **Общие** на вкладке **Параметры**).
Возможные варианты:

-  – **синий** (для отображения сигналов с синим статусом, аналог микро);
-  – **шкала** (используется, например, для представления загрузки ЦПУ, диска и т.д.);
-  – **полный** (используется для представления отдельных сигналов, например, использование ЦПУ, диска и т.д.);
-  – **микро** (используется для представления отдельных сигналов, для которых имеет значение только статус, а не значение сигнала);
-  – **мини снизу** (используется, например, для входящего трафика);
-  – **мини сверху** (используется, например, исходящего трафика).

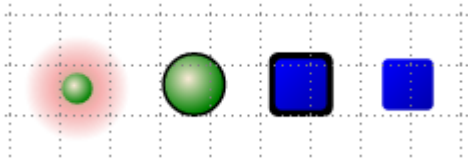
14.8.9. Виджет Лампочка

Виджеты семейства **Лампочка** позволяют настроить цветовое отображение статусов сигналов.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Лампочки**:



Примеры лампочек на канве:



Параметры виджета...

Общие Специфичные

SquaredBulb
BulbClassic
FramedBulb
SquaredBulb
FramedWb

Список пороговых значений

Цвет...

Значение	Цвет

Определение шкалы

Текст: процент

Единицы: нет

Множитель: 1

Диапазон

Мин. значение: 0

Макс. значение: 100

3х-мерный эффект

Назначение

OK Отмена

Специфичные параметры виджета

Тип

Внешний вид виджета.



– **BulbClassic** (круглая лампочка без контура);



– **FramedBulb** (круглая лампочка с контуром);




– **SquaredBulb** (прямоугольная лампочка с контуром);




– **FramedWb** (прямоугольная лампочка без контура).

14. Справочники

Список пороговых значений

Пороговые значения для разделения шкалы на диапазоны и их цветовая маркировка. Чтобы использовать настройки по умолчанию, заполнять таблицу не нужно. Кнопка  используется для добавления строк. Значения следует вводить в возрастающем порядке, в абсолютных величинах или процентах. Допускаются отрицательные значения, но в рамках допустимого диапазона, заданного в группе полей **Диапазон** (см. ниже). В отдельных случаях также могут быть использованы слова **true** и **false**.

Чтобы добавить цветовой диапазон в таблицу, следует:

1. в поле ввести значение верхнего порога диапазона;
2. с помощью кнопки **Цвет...** выбрать цветовую маркировку диапазона шкалы на виджете;
3. нажать кнопку , в результате в таблице создается строка с указанными значениями; если в таблице уже имеются записи, строка добавляется в конец списка.

Диапазон

Диапазон шкалы на виджете. В полях **Мин.Значение**, **Макс.Значение** допускаются отрицательные значения.

3x-мерный эффект

Использование градиентной заливки для создания объемного эффекта.

Назначение

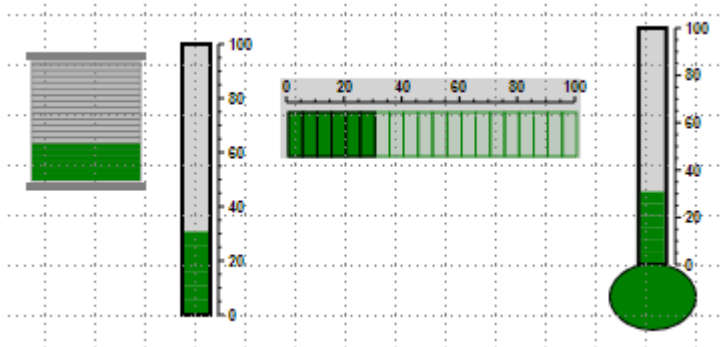
Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу информационной панели](#) ¹⁸⁵.

14.8.10. Виджет Шкала

Иконки для размещения элементов на канве находятся на вкладке **Виджеты** в разделе **Измерительные приборы**:



Примеры элементов на канве (слева направо: **Очередь**, **Вертикальная шкала**, **Горизонтальная шкала**, **Термометр**):




Специфичные параметры виджета

Параметры виджета...

Общие **Специфичные**

Queue



Поведение
 Линейный Триггер
 3х-мерный эффект

Назначение
 /Tree/Operator1/ws-057/CPU 1;windows-cpu;load

Список пороговых значений

Значение	Цвет

Диапазон
 Мин. значение: 0
 Макс. значение: 100

Определение шкалы
 Цена осн. деления: 20
 Цена всп. деления: 5
 Текст: процент
 Единицы: нет
 Множитель: 1

Шкала
 Слева Справа Сверху По низу Нет

Координирование
 Горизонтальная Вертикальная

OK Отмена

Тип

Внешний вид виджета.


Queue – очередь,

VClassic – вертикальная шкала,


HClassic – горизонтальная шкала,

Termometer – термометр.

Список пороговых значений

Пороговые значения для разделения шкалы на диапазоны и их цветовая маркировка. Чтобы использовать настройки по умолчанию, заполнять таблицу не нужно. Кнопка  используется для добавления строк. Значения следует вводить в возрастающем порядке, в абсолютных величинах или процентах. Допускаются отрицательные значения, но в рамках допустимого диапазона, заданного в группе полей **Диапазон** (см. ниже). В отдельных случаях также могут быть использованы слова **true** и **false**.

Чтобы добавить цветовой диапазон в таблицу, следует:

1. в поле ввести значение верхнего порога диапазона;
2. с помощью кнопки **Цвет...** выбрать цветовую маркировку диапазона шкалы на виджете;
3. нажать кнопку , в результате в таблице создается строка с указанными значениями; если в таблице уже имеются записи, строка добавляется в конец списка.

Определение шкалы

- ❖ **Разрешить** – флаг использования шкалы.
- ❖ **Цена осн.делений** – настройка цены основных делений.
- ❖ **Цена всп.делений** – настройка цены вспомогательных делений.
- ❖ **Текст** – единицы измерения, отображаемые у основных делений. Доступные варианты: **нет**, **цифра**, **процент**.

14. Справочники

- ❖ **Единицы** – отображаемое наименование единиц измерений. Можно выбрать из списка или ввести текст вручную.
- ❖ **Множитель** – коэффициент для вычисления отображаемого значения в соответствии с заданными единицами.

Доступно для **VClassic, HClassic, Termometer**.

Шкала

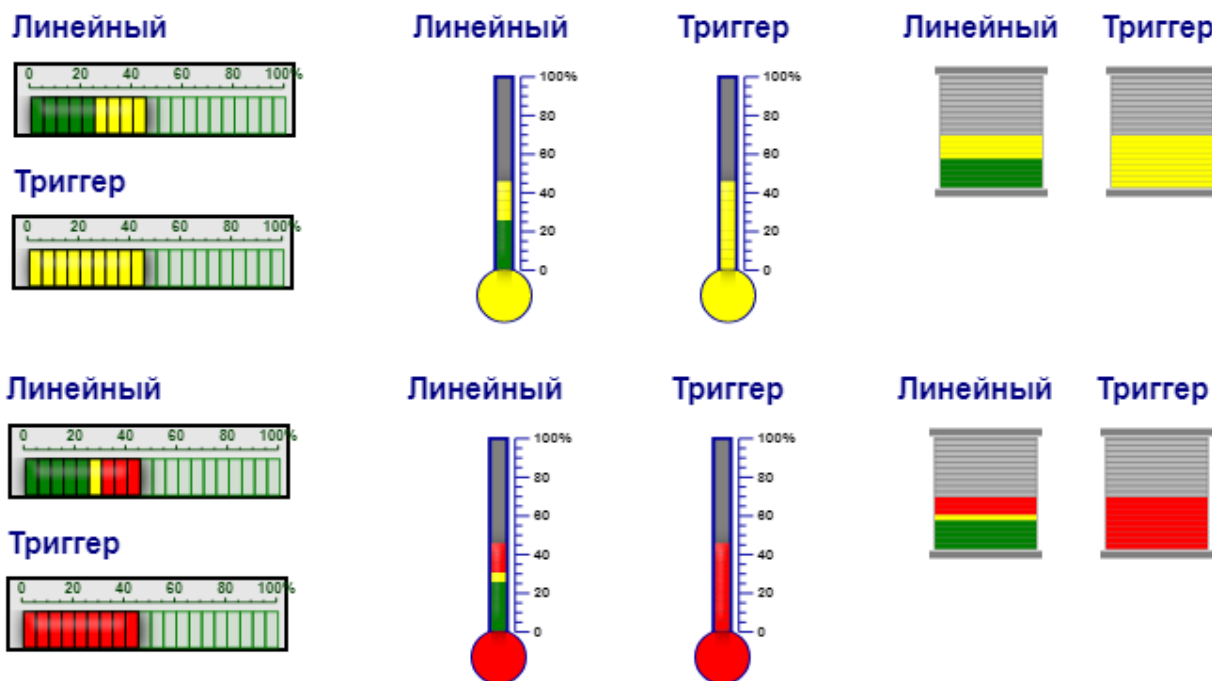
Настройка положения шкалы (для виджетов **VClassic, HClassic, Termometer**). Возможные варианты: **Слева, Справа, Сверху, По низу, Нет**.

Поведение

Вариант окраски индикатора:

- ❖ **Линейный** – дискретная заливка в соответствии с настроенными порогами;
- ❖ **Триггер** – заливка цветом текущего статуса сигнала.

Примеры. Отображение желтого и красного статусов на виджетах с различными значениями параметра **Поведение**:



3x-мерный эффект

Использование градиентной заливки для создания объемного эффекта.

Диапазон

Диапазон шкалы на виджете. В полях **Мин.Значение**, **Макс.Значение** допускаются отрицательные значения.

Координирование

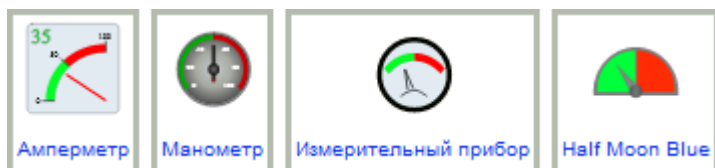
Установка ориентации виджета **Queue**. Доступные варианты: **Горизонтальная, Вертикальная**.

Назначение

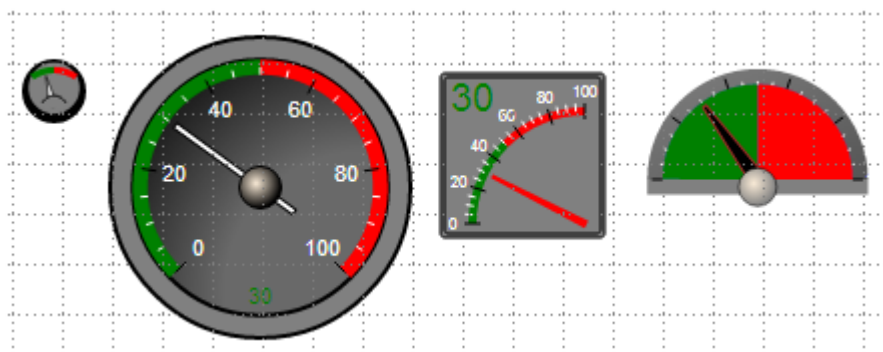
Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу информационной панели](#)¹⁸⁵.

14.8.11. Виджет Измерительный прибор

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Измерительные приборы**:



Примеры измерительных приборов на канве (слева направо: **Измерительный прибор**, **Манометр**, **Амперметр**, **HalfmoonBlue**):



14.8.12. Виджет Многофункциональный измерительный прибор (SmartGauge)

Виджет **SmartGauge** представляет собой значительно расширенную версию виджета **Измерительный прибор** и обладает разнообразными возможностями настройки не только внешнего вида, но и функционала.

Иконки для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Многошкальные изм. приборы**:

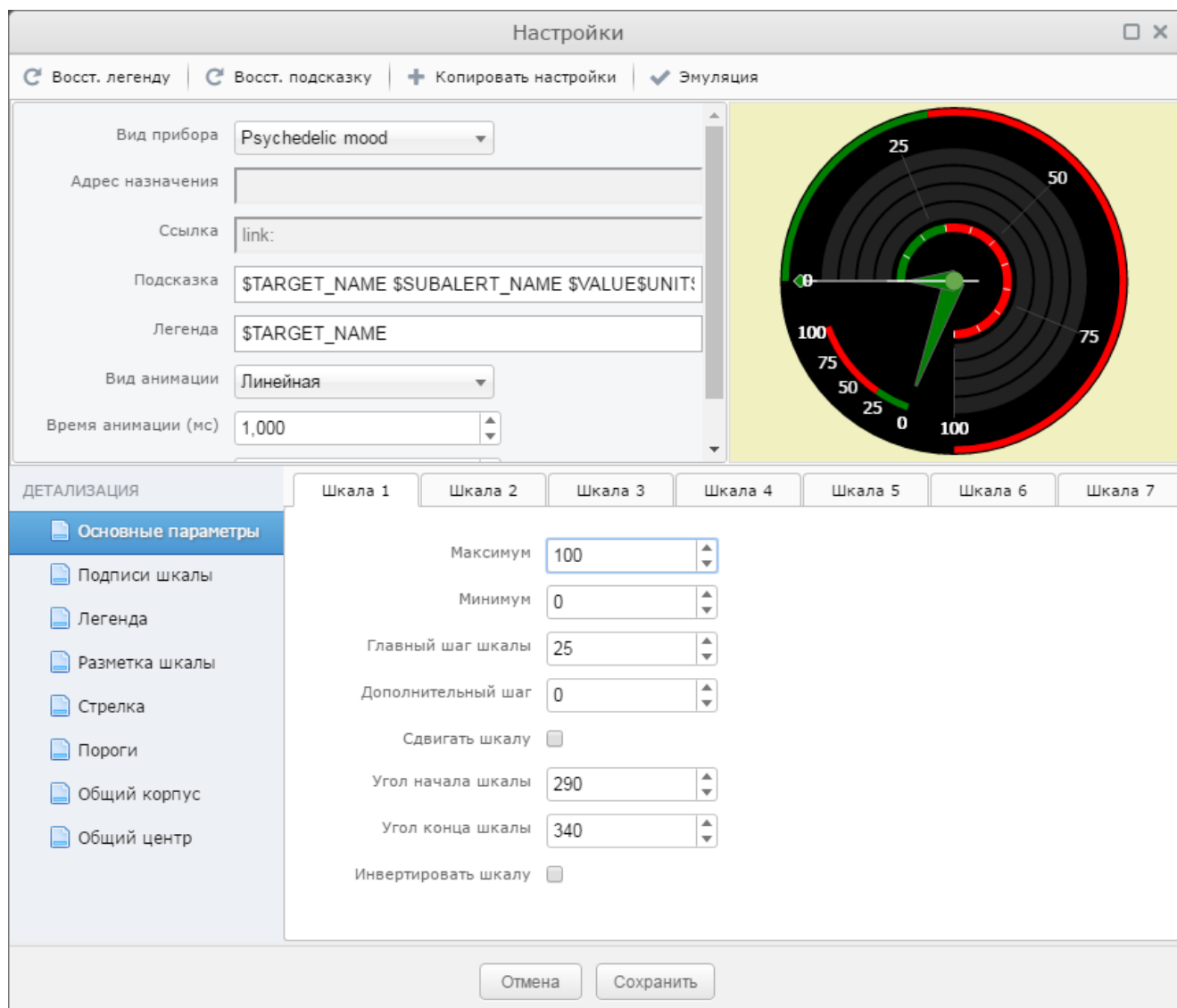
Внешний вид виджета **SmartGauge** на канве зависит от его настроек.

Окно настройки виджета **SmartGauge** вызывается по двойному клику на виджете, размещенном на канве, или с помощью контекстного меню **Параметры виджета**. Окно имеет собственную [панель инструментов с сервисными кнопками](#)³¹⁴.

Основная рабочая область окна по горизонтали разделена на две части:

- ❖ верхняя часть содержит поля для [глобальной настройки виджета](#)³¹⁵ виджета и панель демонстрации некоторых текущих настроек виджета;
- ❖ нижняя часть содержит поля для [настройки компонент виджета](#)³¹⁶;

В нижней части окна размещены кнопки **Отмена** и **Сохранить**.



Для размещения виджетов **SmartGauge** на канве можно использовать разделы **Одношкальные изм. приборы** и **Многошкальные изм. приборы** на вкладке **Виджеты**, которые содержат иконки виджетов **SmartGauge** с предустановленными настройками.

14.8.12.1. Панель инструментов окна настройки виджета

Панель инструментов содержит кнопки, реализующие сервисные функции:

- ❖ **Восст. легенду** – восстановление содержимого поля **Легенда** до предустановленного по умолчанию значения **\$TARGET_NAME**;
- ❖ **Восст. подсказку** – восстановление содержимого поля **Подсказка** до предустановленного по умолчанию **\$TARGET_NAME \$SUBALERT_NAME \$VALUE\$UNIT\$**;
- ❖ **Копировать настройки** – сохранение в буфер обмена текущих настроек виджета в виде текстовой строки в JSON-формате. Данная функция необходима для расширения набора предустановленных виджетов;
- ❖ **Эмуляция** – включение и выключение режима имитации работы виджета. Параметры виджета, для большей наглядности, можно менять при включенном режиме эмуляции.

14.8.12.2. Глобальные настройки виджета

Вид прибора

Элемент управления, позволяющий выбрать один из предустановленных видов прибора. При нажатии на стрелку открывается перечень доступных для выбора приборов. Если данный список содержит много строк, то справа появляется полоса прокрутки. При отображении списка, первоначальный выбор всегда устанавливается на первый элемент списка. Смена осуществляется только при клике указателем мыши на выбранный элемент. Нажатие в любом месте диалога вне списка приводит к его закрытию без выбора. Если открыть список и начать набирать на клавиатуре текст, то содержимое списка будет отфильтровано по синтаксическому набору.

Адрес назначения

Текстовое поле, позволяющее указать привязку элемента к сигналу целевого объекта. В режиме версии "1.0" данное поле недоступно и привязка осуществляется через окно **Параметры** (поле **Единичное назначение** или **Множественное назначение** для многошкального отображения).

Ссылка

Аналогично предыдущему, в режиме версии "1.0" запрещено к изменению.

Подсказка

Форматная строка, позволяющая изменить текст, выводимый в подсказке к данному прибору в режиме отображения контрольной панели. В отличие от привычных элементов информационной панели, данный виджет формирует подсказку для каждой привязки (шкалы) в виде списка и не реагирует на клики мышью. Для перехода по ссылке необходимо нажать клавишу **Ctrl** и кликнуть мышью по интересующей строке списка.

Легенда

Форматная строка, отображающаяся на панели прибора в режиме отображения. Легенда отображается только в том случае, если параметр **Вид легенды** установлен в значение **1** (см. ниже).

Вид анимации

Список, содержащий названия видов анимации движения стрелки прибора при изменении текущего, отображаемого значения. Поведение элемента управления аналогично элементу **Вид прибора**.

Время анимации

Элемент управления, принимающий только числовые, положительные значения. Значение по умолчанию – 100 (мс). Любое некорректное значение будет заменено на значение по умолчанию, при потере элементом фокуса ввода. Кроме непосредственного ввода числового значения с клавиатуры, допускается изменение параметра с помощью нажатия на стрелки, расположенные на правой границе элемента управления. Нажатие на стрелку вверх увеличивает значение на 100, вниз – уменьшает на 100 единиц. Также возможно изменение значений с помощью нажатия на клавиатуре клавиш **Стрелка вверх** и **Стрелка вниз**, причем однократное нажатие увеличивает (вверх), либо уменьшает (вниз) значение на 100 единиц, а продолжительное – включает режим непрерывного изменения до момента отпущения клавиши. Если одновременно с клавишей **Стрелка вниз** / **Стрелка вверх** зажать клавишу **Ctrl**, то шаг изменения изменится со 100 на 10, а при отпущении **Ctrl** – вернется к значению шага равному 100. Максимально допустимое время анимации 10000 (10 секунд).

Количество шкал

Элемент управления, отвечающий за количество одновременно отображающихся шкал на приборе. Максимальное число – 7.

Поведение самого элемента управления подобно описанному выше **Время анимации**. При увеличении значения будет создана дополнительная шкала, с параметрами, идентичными текущей шкале. Например, если прибор имеет две шкалы и текущей выбранной является шкала 1, то при установке этого параметра в значение 3, будет создана дополнительная шкала с параметрами идентичными параметрам шкалы номер 1, что отобразится в добавлении закладки с именем **Шкала 3** в нижней части диалогового окна и данная закладка станет активной. При изменении значения в меньшую сторону, все лишние шкалы, начиная с последней, и независимо от номера текущей шкалы будут уничтожены, а текущей станет последняя из оставшихся шкал.

14.8.12.3. Настройки компонент виджета

Нижняя часть рабочей области окна содержит поля для детальной настройки отдельных элементов виджета. Слева находится список **ДЕТАЛИЗАЦИЯ**, содержащий основные элементы прибора, а справа – вкладки, отвечающие за детальную настройку параметров выбранного элемента на разных шкалах прибора.

14.8.12.3.1. Вкладка Основные параметры

Позволяет настроить основные параметры шкал(ы) прибора.

Максимум

Максимальное значение входного сигнала. Данный параметр следует задавать, если требуется получить определенный диапазон отображения сигнала. В нормальных условиях, прибор получает максимальное значение сигнала от сервера и рассчитывает деления шкалы автоматически.

Минимум

Минимальное значение входного сигнала. Как и **Максимум**, это значение влияет только на минимальное значение шкалы, но не на сам сигнал.

Главный шаг шкалы

Шаг главных отметок шкалы (в процентах). Шкала размечается большими (главными) и маленькими (дополнительными) отметками. Например, если **Главный шаг шкалы = 20 (%)**, **Минимум = 0**, **Максимум = 120**, то первая главная отметка будет располагаться на значении 24 (а не 20, как может показаться на первый взгляд).

Дополнительный шаг

Шаг дополнительных отметок шкалы в процентах. Дополнительные отметки не подписываются.

Угол начала шкалы

Положение нулевой (%) отметки шкалы прибора (в градусах). Допустимый диапазон – от **0** до **359**. 0 градусов находится на 9 часах прибора. Направление разметки шкалы зависит от взаимного расположения начального и конечного углов прибора. Любое изменение углов (равно как и всех остальных параметров) приводит к мгновенному изменению текущего вида шкалы прибора, поэтому настроить желаемый вид очень просто.

Угол конца шкалы

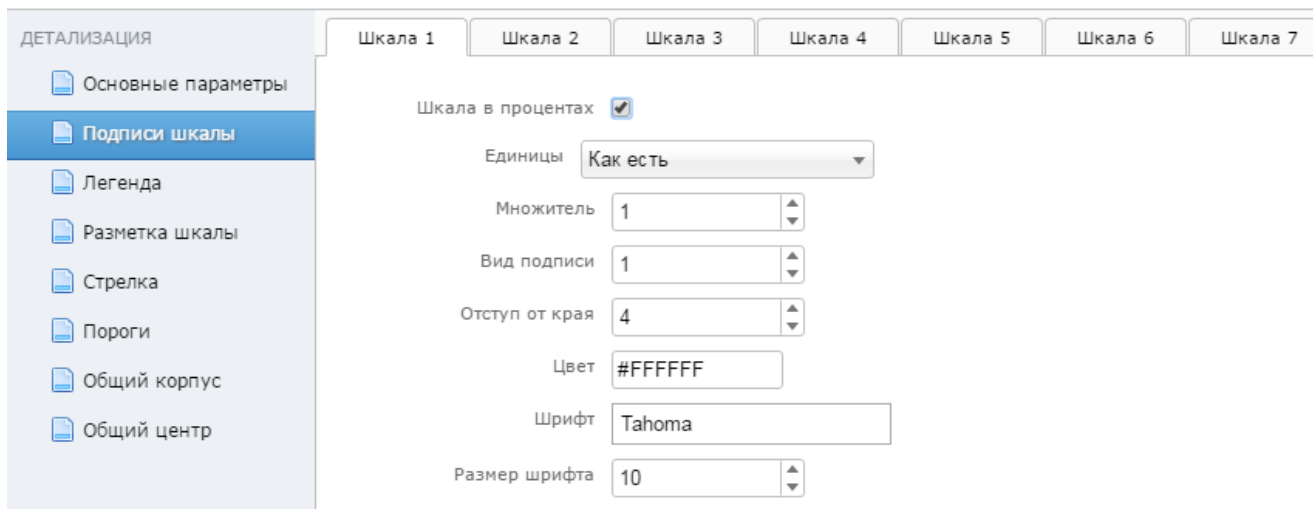
Положение конца шкалы (100%) прибора. Допустимый диапазон – от **0** до **719**. В некоторых случаях, шкала может прокручиваться два раза. Данное отображение не является ошибочным или случайным, а сделано специально, для отображения специфических шкал. Итак, изменяя любые параметры прибора и наблюдая за видом прибора на панели предпросмотра, пользователь может привести его в требуемый для конкретного применения вид.

Инвертировать шкалу

Установленный флаг изменяет направление отсчета по шкале: $0 = 100\%$, а $100\% = 0$; при этом направление движения стрелки также изменяется.

14.8.12.3.2. Вкладка Подписи шкалы

Позволяет настроить параметры отображения шкалы.



Шкала в процентах

Независимо от того, что прибор работает только в процентах и принимает значения сигнала в процентах, шкала прибора может быть подписана абсолютными числами, при этом **Максимум** и **Минимум** на вкладке **Основные параметры**, интерпретируются, соответственно, как максимальное и минимальное абсолютные значения. Так как при разметке шкалы не происходит автоматической адаптации размера шрифта, то корректность отображения подписей шкалы остается в зоне ответственности пользователя.

Вид подписи

Переключатель, позволяющий быстро изменить режим отображения подписи шкалы. Ввиду того, что настройки прибора не будут происходить слишком часто (настроил и забыл), а также учитывая простоту и наглядность процесса настройки параметров, режимы (для простоты) обозначаются цифрами:

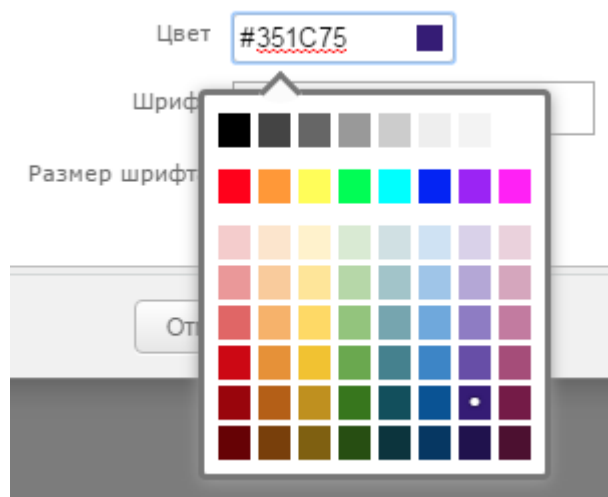
- 0 – подписи не отображаются;
- 1 – отображение в абсолютных величинах (0, 20, 40,...);
- 2 – отображение в экспоненциальном виде (0, 2.0e+1, 4.0e+1,...);
- 3 – отображение десятичными дробями в пределах от 0 до 1 (0, 0.2, 0,4,...);
- 4 – отображение обыкновенными дробями (0, 1/5, 2/5,...).

Отступ от края

Значение, определяющее отступ центра подписи (в процентах от радиуса прибора). При некоторых значениях возможно некорректное отображение. Важно, что проверка корректности отображения находится в ведении пользователя!

Цвет

Цвет подписи. Цветовой код можно ввести вручную в виде числа в шестнадцатеричном формате, например, **#FA04E3**, либо в сокращенном формате, например, **#777**, при этом символ **#** вводить не требуется. Также можно выбрать цвет из палитры с помощью мыши. Выбранный цвет на палитре отмечается серой точкой, а в поле отображается квадрат, окрашенный в выбранный цвет.



Шрифт

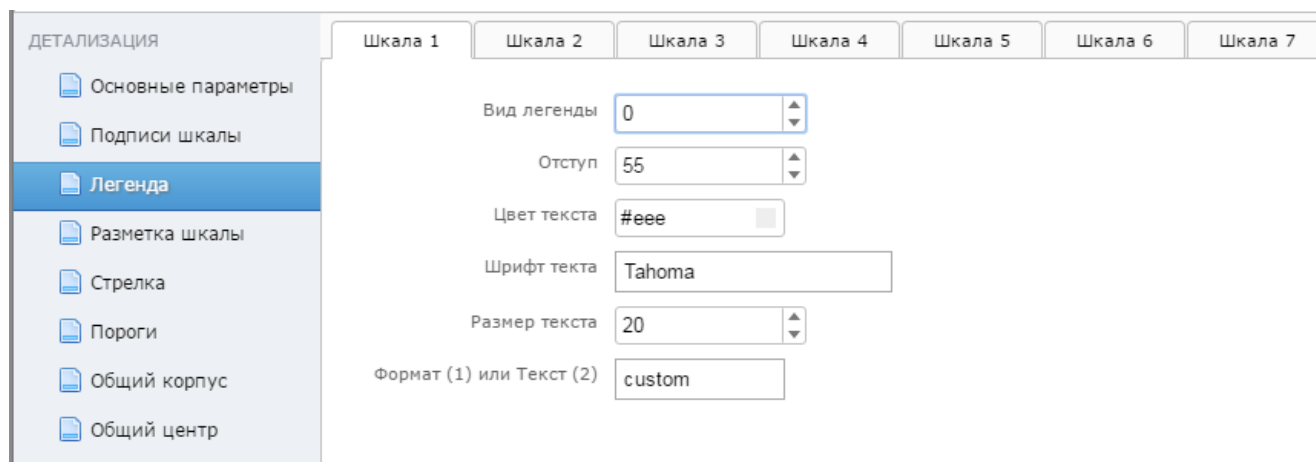
Название шрифта на английском языке.

Размер шрифта

Высота шрифта в единицах от 5 до 72.

14.8.12.3.3. Вкладка Легенда

Позволяет настроить вид и контент, отображаемый в специальном поле.



Вид легенды

Переключатель, позволяющий быстро изменить вид легенды на один из предустановленных:

0 – не отображать легенду;

1 – если в параметре **Легенда** в [глобальных настройках виджета](#)^[315] записана форматирующая строка, то в легенде будет отображаться получаемое значение. Если параметр **Легенда** пуст, но в параметре **Формат (1) или Текст (2)** (см. ниже) записана форматирующая строка, то в легенде будет также отображаться получаемое значение;

2 – строка, записанная в параметре **Формат (1) или Текст (2)**, интерпретируется как простой текст и отображается в легенде.

Отступ

Положение центра легенды по вертикали в процентах от радиуса прибора. Допустимый диапазон значений – от **0** до **200**. Значение **0** располагает центр легенды в самой верхней точке прибора, а значение **200** – в самой нижней.

Цвет текста, Шрифт текста, Размер текста

Аналогично параметрам на вкладке **Подписи шкалы**.

Формат (1) или Текст (2)

Форматирующая строка или текст для легенды. Параметр дополняет параметр **Вид легенды**.

14.8.12.3.4. Вкладка Разметка шкалы

Позволяет настроить видимость и вид разметки шкалы.

Показывать шкалу

Переключатель, имеющий два значения:

- 0** – шкала не отображается;
- 1** – шкала отображается.

Отступ главных отметок

Положение главных отметок шкалы – в процентах от радиуса прибора. Может иметь значение от **0** до **200**. Поведение элемента управления описано выше.

Длина главных отметок

Длина главных отметок шкалы – в процентах от радиуса прибора. Может иметь значение от **0** до **100**. Некорректное значение интерпретируется как **100**, при потере фокуса.

Цвет главных отметок

Цвет отображения главных отметок шкалы. Поведение элемента управления подобно описанным выше элементам выбора цвета.

Отступ вспом. отметок

Положение вспомогательных отметок. Поведение аналогично описанному выше элементу управления параметром **Отступ главных отметок**;

Длина вспом. отметок

Длина вспомогательных отметок и поведением подобен элементу управления длиной главных отметок.

Цвет вспом. отметок

Цвет отображения вспомогательных отметок и поведением подобен описанному выше параметру **Цвет главных отметок**.

14.8.12.3.5. Вкладка Стрелка

Позволяет настроить вид стрелки.

Шкала 1	Шкала 2	Шкала 3	Шкала 4	Шкала 5	Шкала 6	Шкала 7
ДЕТАЛИЗАЦИЯ						
Основные параметры						
Подписи шкалы						
Легенда						
Разметка шкалы						
Стрелка						
Пороги						
Общий корпус						
Общий центр						

Вид стрелки: 4

Отступ от края: 28

Радиус: 0

Цвет линии: #666

Цвет заполнения: #aaa

Вид стрелки

Вид стрелки. Возможные значения:

0 – стрелка не отображается;

1 – стрелка отображается в виде прямой линии заданного цвета и заполнения. Параметр **Радиус** не учитывается;

2 – стрелка отображается в виде тонкой прямой линии с квадратным маркером на конце. Цвет заполнения маркера зависит от цвета (статуса) сигнала и не зависит от установленного цвета заполнения (так называемая – активная стрелка). Параметр **Радиус** определяет длины сторон маркера;

3 – стрелка отображается в виде треугольника с заданными цветами линии и заполнения. Размер основания треугольника задается параметром **Радиус** на вкладке [Общий центр](#)³²³. Параметр **Радиус** не учитывается;

4 – активная стрелка, по виду подобная выше описанному значению 3;

5 – особый вид стрелки, в виде активного сектора. Положение, ширина и цвет заливки сектора определяется параметрами управления главными отметками, такими как **Отступ главных отметок**, **Длина главных отметок**, **Цвет главных отметок**. Параметр **Отступ от края** не учитывается. Начало анимации может сопровождаться «вспыхиванием», связанным с полной перерисовкой. Параметр **Радиус** не учитывается;

6 – подобна интерпретации значения 5, но без заливки сектора. При этом становятся видны главные отметки. Параметр **Отступ от края** не учитывается. Параметр **Радиус** не учитывается;

7 – стрелка отображается в виде активного круга. Параметр **Радиус** задает размер. Цвет заполнения не учитывается;

8 – стрелка отображается в виде неактивного круга. Параметр **Радиус** задает размер. Параметр **Цвет заполнения** определяет цвет заливки;

9 – стрелка отображается в виде активного круга с перекрестьем внутри. Параметр **Радиус** задает размер. Цвет перекрестья совпадает с цветом линии (параметр **Цвет линии**). Цвет заполнения не учитывается;

10 – стрелка отображается в виде неактивного круга с перекрестьем внутри. Параметр **Радиус** задает размер. Цвет перекрестья совпадает с цветом линии (параметр **Цвет линии**). Параметр **Цвет заполнения** определяет цвет заливки;

Начало анимации может сопровождаться «вспыхиванием», связанным с полной перерисовкой;

Отступ от края

Отступ конца стрелки от края прибора – в процентах от радиуса прибора. Актуален для видов стрелки **1, 2, 3, 4**. Допустимый диапазон значений от **0** до **100**. При определенных значениях может перекрываться общим центром, параметры которого задаются на вкладке [Общий центр](#)³²³.

Цвет линии

Цвет линии стрелок вида **1, 2, 3, 4**.

Цвет заполнения

Цвет заполнения стрелок вида **1 и 3**.

14.8.12.3.6. Вкладка Пороги

Позволяет настроить вид и отображение порогов (thresholds).

Вид порогов

Переключатель видимости и типа отображаемых порогов:

0 – пороги не отображаются;

1 – пороги отображаются в виде секторов, ширина которых определяется параметром **Ширина узкого типа**;

2 – пороги отображаются в виде секторов, положение и ширина которых связана с главными отметками шкалы. При этом параметры **Отступ от края** и **Ширина узкого типа** не учитываются.

Отступ от края

Отступ порогов от внешнего радиуса прибора в процентах.

Ширина узкого типа

Ширина порогов для вида 1 в процентах.

Ширина широкого типа

В текущей версии неактуален, зарезервирован на будущее.

14.8.12.3.7. Вкладка Общий корпус

Позволяет настроить вид корпуса прибора. При наличии двух и более шкал параметры панели остаются глобальными (одинаковыми для всех шкал) и отображаются только для удобства редактирования.

ДЕТАЛИЗАЦИЯ	Шкала 1	Шкала 2	Шкала 3	Шкала 4	Шкала 5	Шкала 6	Шкала 7
Основные параметры							
Подписи шкалы							
Легенда							
Разметка шкалы							
Стрелка							
Пороги							
Общий корпус							
Общий центр							

Вид корпуса	<input type="text" value="3"/>
Толщина корпуса	<input type="text" value="2"/>
Цвет корпуса	<input type="text" value="#000000"/>
Толщина ободка	<input type="text" value="4"/>
Цвет ободка	<input type="text" value="#000000"/>
Толщина циферблата	<input type="text" value="1"/>
Цвет обводки циферблата	<input type="text" value="#000000"/>
Цвет циферблата	<input type="text" value="#000000"/>

Вид корпуса

Переключатель, отвечающий за видимость и вид корпуса прибора:

0 – корпус не отображается;

1 – корпус отображается в виде круга, заполненного цветом ободка. Никакие другие параметры не учитываются;

2 – циферблат заполняется цветом ободка, при этом цвет корпуса и толщина корпуса определяют внешнее кольцо;

3 – наиболее полный вариант отображения корпуса, состоящего из внешнего кольца самого корпуса, последующего за ним кольца – ободка и внутреннего круга – циферблата. При определенных настройках можно добиться вида, при котором корпус состоит из 4-х разноцветных окружностей;

4 – разновидность вида 3, с активным ободком, цвет которого меняется в зависимости от статуса сигнала.

Толщина корпуса и Цвет корпуса

Используется для видов **2, 3, 4**.

Толщина ободка

Используется для видов **3 и 4**.

Цвет ободка

Используется для видов **1, 2, 3**.

Толщина циферблата

Толщина линии обводки внутреннего круга. Используется для видов **3 и 4**.

Цвет обводки циферблата

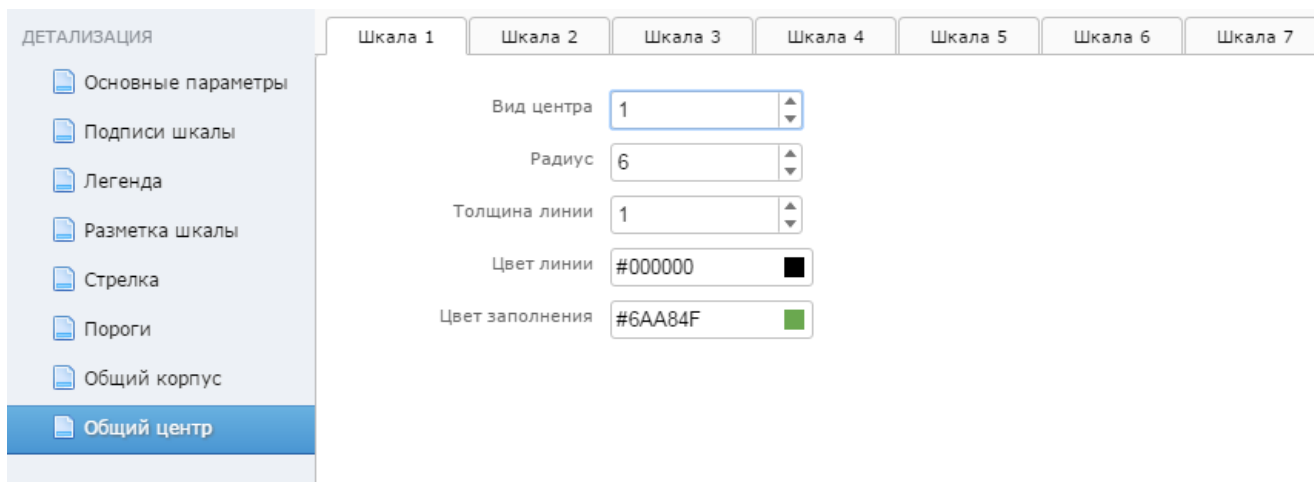
Цвет линии обводки внутреннего круга. Используется для видов **3 и 4**.

Цвет циферблата

Цвет внутреннего круга. Используется для видов **3 и 4**.

14.8.12.3.8. Вкладка Общий центр

Позволяет настроить видимость, вид и поведение центрального круга прибора.



Вид центра

Переключатель вида и видимости:

0 – центр не видим;

1 – центр отображается в виде окружности с параметрами, описываемыми ниже;

2 – центр отображается в виде активной окружности, при этом цвет заполнения не учитывается, а соответствует цвету (статусу) сигнала.

Радиус

Радиус центрального круга – в процентах от радиуса прибора. Допустимый диапазон значений от **0** до **100**. При расчете размера центра учитываются параметры корпуса прибора, и возможны ситуации, когда центр перекрывает отдельные его части, такие как циферблат и ободок. Также этот параметр определяет ширину основания треугольной стрелки вида **2** и **3**.

Толщина линии

Толщина обводки центра. Линия, обводящая элемент центр, не влияет на радиус центра и распространяется наружу. При этом возможно получение различных интересных эффектов отображения прибора. Для примера можно задать следующие параметры:

Вид центра=1, Радиус=90, Толщина линии=50, Цвет линии=#666, Цвет заполнения=#aaa, а также Вид стрелки=3, Отступ от края=0;

Цвет линии

Цвет обводки центра вида **1** и **2**.

Цвет заполнения

Цвет заполнения круга. Используется только для вида **1**.

14.8.13. Виджет Стек

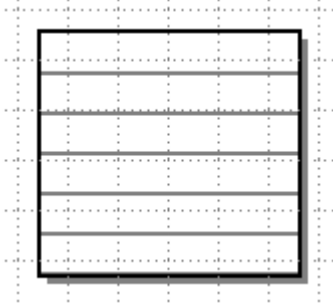
Виджет **Стек** отображает историю изменения значения сигнала в виде списка. Виджет предполагает привязку одного сигнала (т.е. единичное назначение).

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:

14. Справочники



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Параметры виджета...

Общие **Специфичные**

Число строк: 2 20

Направление:

Показывать разделители строк

Использовать цветовую схему

Закрасить фон двумя чередующимися цветами

Закрашивать фон одним цветом с изменяющейся прозрачностью

Отображать цветные квадратики с левой стороны каждой строки

Увеличение прозрачности (%) 0 100

Источник получения данных для входящей строки

Формат данных

Содержимое:

Число линий

Настройка числа линий элементов.

Направление

Направление сортировки полученных значений сигнала.

Вверх – значения перемещаются снизу вверх, т.е. нижняя строка содержит последнее значение;

Вниз – значения перемещаются сверху вниз, т.е. верхняя строка содержит последнее значение.

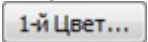
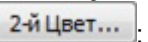
Показывать разделители строк

Признак отображения разделителей строк.

- ❖ **Цвет...** – настройка цвета разделителей строк в стандартном окне.

Использовать цветовую схему


При отключенном флаге строки списка не закрашиваются. При включенном флаге к виджету применяется настроенная цветовая схема:

- ❖ **Закрасить фон двумя чередующимися цветами** – цвета задаются с помощью кнопок , ;
- ❖ **Закрашивать фон одним цветом с изменяющейся прозрачностью** – цвет заливки фона задается с помощью кнопки **Цвет...** ниже;
- ❖ **Отображать цветные квадратики с левой стороны каждой строки** – в каждой строке списка отображаются квадратики цвета, заданного с помощью кнопки **Цвет...**;
- ❖ **Увеличение прозрачности (%)** – максимальная степень прозрачности строк или квадратики.

Источник получения данных для входящей строки

Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу](#)^[185], [Привязка одноптипных сигналов к элементу](#)^[188].

Формат данных

Шаблон выводимых данных. Допустимо использовать произвольный текст и [предопределенные переменные](#)^[341] – их список вызывается по нажатию кнопки  или по двойному клику.

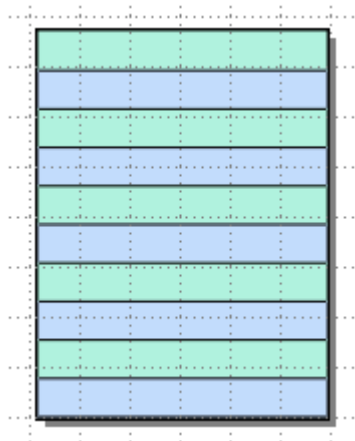
14.8.14. Виджет Список

Виджет **Список** отображает значения нескольких сигналов в виде отсортированного списка.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Параметры виджета...

Общие **Специфичные**

Число строк: 2 20

Направление:

Показывать разделители строк

Использовать цветовую схему

Закрасить фон двумя чередующимися цветами

Закрашивать фон одним цветом с изменяющейся прозрачностью

Отображать цветные квадратики с левой стороны каждой строки

Увеличение прозрачности 100 (%)

Источник получения данных для входящей строки

Сортировка строк

Первые N по возрастанию по убыванию как есть (без сортировки)

Формат данных

Содержимое:

Число линий

Настройка числа линий элементов.

Направление

Направление сортировки полученных значений сигнала.

Показывать разделители строк

Признак отображения разделителей строк.

- ❖ **Цвет...** – настройка цвета разделителей строк в стандартном окне.

Использовать цветовую схему

При отключенном флаге строки списка не закрашиваются. При включенном флаге к виджету применяется настроенная цветовая схема:

- ❖ **Закрасить фон двумя чередующимися цветами** – цвета задаются с помощью кнопок , ;
- ❖ **Закрашивать фон одним цветом с изменяющейся прозрачностью** – цвет заливки фона задается с помощью кнопки **Цвет...** ниже;
- ❖ **Отображать цветные квадратики с левой стороны каждой строки** – в каждой строке списка отображаются квадратики цвета, заданного с помощью кнопки **Цвет...**;
- ❖ **Увеличение прозрачности (%)** – максимальная степень прозрачности строк или квадратиков.

Источник получения данных для входящей строки

Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу](#)^[185], [Привязка однопипных сигналов к элементу](#)^[188].

Сортировка строк

Настройка сортировки. Возможные варианты: **Первые N**, **Сортировка по возрастанию**, **Сортировка по убыванию**, **Как есть (без сортировки)**.

Формат данных

Шаблон выводимых данных. Допустимо использовать произвольный текст и [предопределенные переменные](#)^[347] – их список вызывается по нажатию кнопки или по двойному клику.

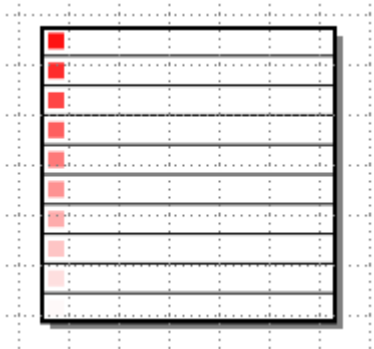
14.8.15. Виджет Список с лампочками

Виджет **Список с лампочками** позволяет просмотреть значения и статусы сигналов в виде списка с маркерами.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Параметры виджета...

Общие Специфичные

Число строк: 2 10 20

Направление: Вверх

Показывать разделители строк Цвет...

Использовать цветовую схему

Закрасить фон двумя чередующимися цветами

Закрашивать фон одним цветом с изменяющейся прозрачностью

Отображать цветные квадратики с левой стороны каждой строки

Увеличение прозрачности (%): 0 10 100

Цвет...

Источник получения данных для входящей строки

/Tree/Север/ws-002/CPU 1;windows-cpu;load,/Tree/Север/ws-002/CPU 2;windows-cpu;load,/Tree/Север/stend-mon-1/CPU;lnx-cpu;load

Формат данных

Содержимое: \$VALUE\$UNITS

OK Отмена

Число линий

Настройка числа линий элементов.

Направление

Направление сортировки полученных значений сигнала.

Показывать разделители строк

Признак отображения разделителей строк.

- ❖ **Цвет...** – настройка цвета разделителей строк в стандартном окне.

Использовать цветовую схему

При отключенном флаге строки списка не закрашиваются. При включенном флаге к виджету применяется настроенная цветовая схема:

- ❖ **Закрасить фон двумя чередующимися цветами** – цвета задаются с помощью кнопок ;
- ❖ **Закрашивать фон одним цветом с изменяющейся прозрачностью** – цвет заливки фона задается с помощью кнопки **Цвет...** ниже;
- ❖ **Отображать цветные квадратики с левой стороны каждой строки** – в каждой строке списка отображаются квадратики цвета, заданного с помощью кнопки **Цвет...**;
- ❖ **Увеличение прозрачности (%)** – максимальная степень прозрачности строк или квадратики.

Источник получения данных для входящей строки

Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу](#)^[185], [Привязка одноптиных сигналов к элементу](#)^[188].

Формат данных

Шаблон выводимых данных. Допустимо использовать произвольный текст и [предопределенные переменные](#)^[347] – их список вызывается по нажатию кнопки или по двойному клику.

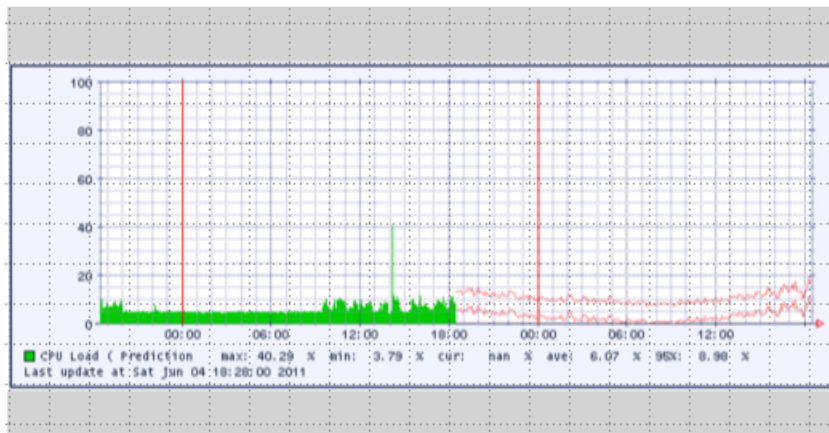
14.8.16. Виджет График

Виджет **График** позволяет просмотреть график изменения значения сигнала.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



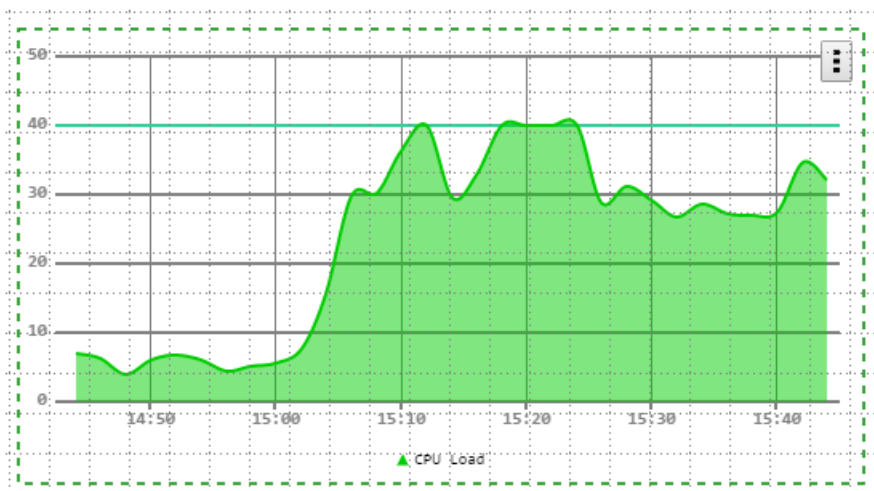
14.8.17. Виджет Динамический график

Виджет **Динамический график** позволяет отобразить область вывода графиков. К виджету можно привязать один целевой объект.

Иконка для размещения виджета на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Вводятся в окне **Настройки**, которое вызывается по двойному клику на виджете, размещенном на канве, или с помощью контекстного меню **Параметры виджета**.

Настройки

Источник: */Tree/Северный филиал/ws-002/WAN Miniport (IP) ...

График: */Трафик

Предсказания: показывать

Показывать данные за: 1 день

Режим обновления: перезагружать все, последние изменения, накапливать данные до 0 точек

Частота обновлений: постоянная, или: []

Серверное время:

Линейка инструментов: скрыть

Подсказка:


Легенда:

Статистика: разрешить панель отображения статистики по легенде
Изначально: Показывается Спрятана

Высота: переменная (но не более 35%), фиксированная: [] пикс. %

Отмена Сохранить

Источник

Целевой объект, по которому будет отображаться график. Следует нажать кнопку  и в дереве конфигурации выбрать целевой объект.

График

Представление (вид графика). Список для выбора зависит от выбранного целевого объекта (определяется шаблоном объекта).

Предсказания

Необходимость отображать график с предсказаниями (трендами). Флаг имеет действие, только если в поле **Источник** выбран целевой объект с сигналом типа **Trend**.

Показывать данные за

Период, отображаемый на графике. Доступные значения: **1 час** (за последний час), **1 день** (за последние сутки), **1 неделя** (за последние 7 суток), **1 месяц** (за последние 30 суток), **1 год** (за последние 365 суток).

Режим обновления

Способ обновления графика:

- ❖ **перезагружать все** – полное обновление графика через заданный интервал (по умолчанию 5 секунд);
- ❖ **последние изменения** – динамическое обновление графика, подгрузка только новых данных. В этом режиме ранее полученные данные не обновляются, а только сдвигаются к концу графика. При получении новых данных самые ранние точки графика смещаются за границы отображаемой области;
- ❖ **накапливать данные до ... точек** – количество точек на графике не превышает заданного числа (длины очереди). Новые данные добавляются в начало графика, старые при этом не смещаются. По достижению количества точек на графике заданного числа старые точки удаляются.

Частота обновления

Настройка обновления данных на графике:

- ❖ **постоянная** – обновление данных графика происходит при получении очередного пакета данных от модуля **Нувола Топаз**. Частота получения пакетов определяется конфигурацией сервера;
- ❖ заданная частота обновления – запрос новых данных и обновление графика в соответствии с указанным параметром. Доступные значения: **2 sec** (2 секунды), **15 sec** (15 секунд), **1 min** (1 минута), **2 min** (2 минуты), **10 min** (10 минут).

Серверное время


Если флаг установлен, то на графике показываются данные по серверному времени, если не установлен – по времени клиентской рабочей станции, на которой запущен браузер.


Линейка инструментов

Настройка панели инструментов, расположенной над графиком. Возможные варианты:

- ❖ **скрыть** – панель инструментов не показывается
- ❖ **показать** – полная панель инструментов графика;
- ❖ **частичный набор** – отображается панель инструментов с ограниченным набором функций: масштабирование графика, переход по ссылке на страницу целевого объекта и экспорт данных.

Статистика

Настройка таблицы статистики. Если флаг **Разрешить панель отображения...** установлен, то в границах виджета под графиком возможен вывод таблицы со статистическими значениями по каждому сигналу на динамическом графике, а именно: максимальное, минимальное, среднее, текущее, последнее значения. На панели инструментов виджета активизируется кнопка  для открытия/закрытия таблицы со статистикой. При установленном флаге активизируются остальные поля и флаги для настройки панели статистики.

Изначально: Показывается / Спрятна – настройка отображения таблицы со статистикой при открытии информационной панели. Вне зависимости от настройки для открытия/закрытия таблицы можно использовать кнопку  на панели инструментов динамического графика.

Высота

Высота таблицы со статистикой:

- ❖ **переменная** – высота определяется динамически в зависимости от количества строк таблицы;
- ❖ **фиксированная** – высота определена в пикселах или процентах от высоты виджета.

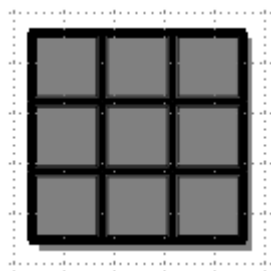
14.8.18. Виджет Таблица

Виджет **Таблица** позволяет вывести на информационную панель таблицу со значениями сигнала.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



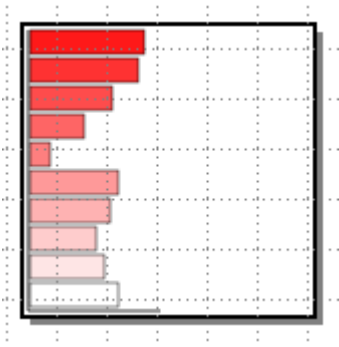
14.8.19. Виджет Диаграмма

Виджет **Диаграмма** выводит значения сигналов в виде линейной диаграммы и может содержать легенду с информацией о показанных сигналах.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Параметры виджета...

Общие Специфичные

Количество столбцов: 2 2

Ориентация столбцов: вертикальная

Легенда Л П В Н

Ось Y Ширина: 2

Ось X Ширина: 2

Отметки шкалы

Режим накопления

Автоматическое назначение контрастных цветов

Закрашивать одним цветом с изменяющейся прозрачностью

Увеличение прозрачности 0 100 (%)

Источник получения данных для входящей строки

/Tree/Север/Серверы/dev-23/eno16777728,traffic64;in,/Tree/Север/Серверы/mon-1/eth0,traffic64;in,/Tree/Север/Серверы/mon-1/eth1,traffic64

Сортировка

Первые N по возрастанию по убыванию как есть (без сортировки)

Формат данных

Содержимое: \$VALUESUNITS

Количество столбцов

Максимальное число столбцов, отображаемых на диаграмме. Возможные значения: от **2** до **20**.

Пример. **Количество столбцов=10.** Если к виджету привязано 8 сигналов, то диаграмма состоит из 8 столбцов. Если к виджету привязано 15 сигналов, то на диаграмме отображаются только 10 сигналов.

Ориентация столбцов

Установка ориентации столбцов: **вертикальная, горизонтальная.**

Легенда

При установленном флаге отображается легенда диаграммы. Переключатель задаёт местоположение легенды относительно диаграммы: **Л** – слева, **П** – справа, **В** – выше, **Н** – ниже.

14. Справочники

Ось Y / Ось X

При установленном флаге на диаграмме отображается ось Y / X с заданными параметрами: **Ширина** – ширина оси в пикселях, **Цвет...** – настройка цвета оси в стандартном окне.

Отметки шкалы

При установленном флаге вдоль столбцов отображаются деления шкалы (метки и значения).

Режим накопления

Флаг использования режима накопления.

Автоматическое назначение контрастных цветов

Автоматическое использование контрастных цветов соседних столбцов.

Закрашивать одним цветом с изменяющейся прозрачностью

Признак заливки фона указанным цветом. Параметры:

- ❖ **Цвет...** – настройка цвета в стандартном окне;
- ❖ **Увеличение прозрачности (%)** – максимальное увеличение прозрачности при закрашивании.

Источник получения данных для входящей строки

Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу](#)^[185], [Привязка одноптипных сигналов к элементу](#)^[188].

Сортировка

Настройка сортировки отображаемых данных. Возможные варианты: **Первые N**, **Сортировка по возрастанию**, **Сортировка по убыванию**, **Как задано (без сортировки)**.

Формат данных

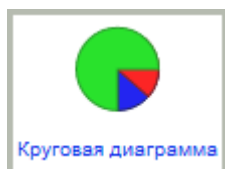
Шаблон выводимых данных. Допустимо использовать произвольный текст и [предопределенные переменные](#)^[347] – их список вызывается по нажатию кнопки или по двойному клику.

Настроенный шаблон применяется для вывода легенды.

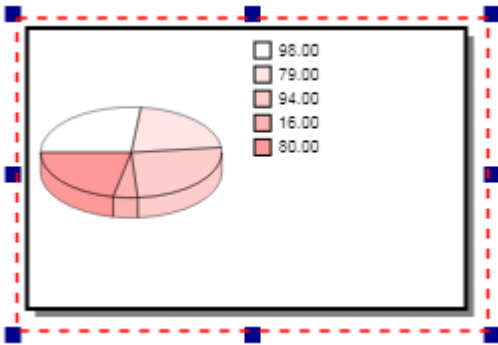
14.8.20. Виджет Круговая диаграмма

Виджет **Круговая диаграмма** выводит значения одноптипных сигналов в виде круговой диаграммы и может содержать легенду с информацией о показанных сигналах.

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Параметры виджета...

Общие **Специфичные**

Количество сегментов: 2 2

Легенда Л П В Н

Режим накопления

Автоматическое назначение контрастных цветов

Сектора автоматически окрашенные контрастными цветами

Увеличение прозрачности (%): 0 100

Источник получения данных для входящей строки

Сортировка

Первые N по возрастанию по убыванию как есть (без сортировки)

Формат данных

Содержимое: \$VALUE

Количество сегментов

Число сегментов круговой диаграммы.

Легенда

При установленном флаге отображается легенда диаграммы. Переключатель задаёт местоположение легенды относительно диаграммы: **Л** – слева, **П** – справа, **В** – выше, **Н** – ниже.

Режим накопления

Автоматическое назначение контрастных цветов

Соседние сектора на диаграмме окрашиваются в контрастные цвета, выбираемые автоматически.

Закрасить одним цветом с уменьшающейся прозрачностью

Соседние сектора на диаграмме закрашиваются одним цветом, но с разной степенью прозрачности, которая увеличивается дискретно с заданным шагом.

14. Справочники

- ❖ **Цвет...** – выбор цвета в стандартном окне;
- ❖ **Увеличение прозрачности (%)** – шаг увеличения прозрачности.

Источник получения данных для входящей строки

Привязка сигналов к виджету. См. [Привязка нескольких сигналов к элементу](#)¹⁸⁵, [Привязка однотипных сигналов к элементу](#)¹⁸⁸.

Сортировка

Настройка сортировки отображаемых данных. Возможные варианты: **Первые N**, **Сортировка по возрастанию**, **Сортировка по убыванию**, **Как задано (без сортировки)**.

Формат данных

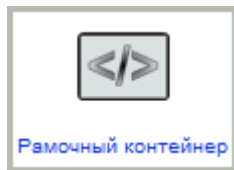
Шаблон выводимых данных. Допустимо использовать произвольный текст и [предопределенные переменные](#)³⁴¹ – их список вызывается по нажатию кнопки или по двойному клику.

Настроенный шаблон применяется для вывода легенды.

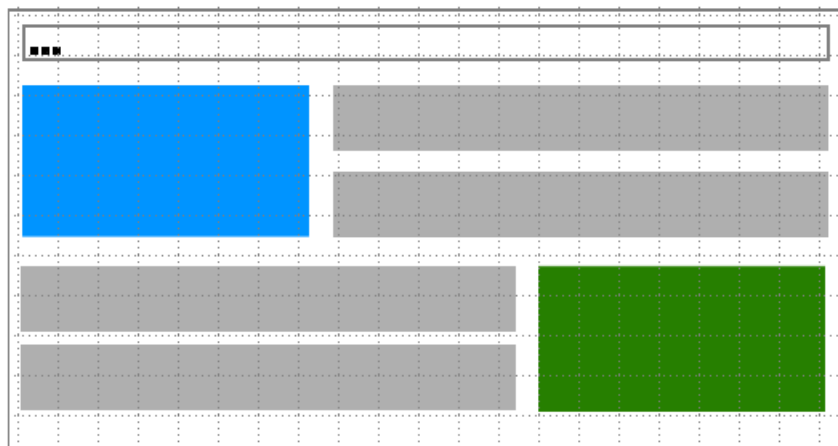
14.8.21. Виджет Рамочный контейнер

Виджет **Рамочный контейнер** позволяет встраивать в информационную панель другую информационную панель, любую страницу Web-интерфейса ПК Нувол (например, консоль событий) или внешний web-сайт (при условии, что последний допускает такую возможность).

Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



Специфичные параметры виджета

Вводятся на вкладке **Параметры** в разделе **Общие**.

Ссылка

Ссылка, которую отображает виджет **Рамочный контейнер**. Как правило, значение в данном поле редактировать не нужно, т.к. строка ссылки формируется автоматически по мере ввода значений вложенных параметров блока.

Например, чтобы настроить отображение консоли событий, следует сначала в основном окне Web-интерфейса открыть для нужного целевого объекта представление консоль событий, настроить столбцы таблицы, фильтры, сортировку, а затем скопировать (**Ctrl+C**) содержимое адресной строки браузера и вставить (**Ctrl+V**) его в поле **Адрес**.

Протокол

Список значений содержит допустимые виды ссылок:

[no link] – ссылка отсутствует (используется по умолчанию для элементов, не привязанных к сигналам);

link – ссылка на какой-либо узел дерева конфигурации ПК Нувола (используется по умолчанию для элементов, привязанных к сигналам);

http:// – ссылка на произвольный сайт или Web-страницу;

ftp:// – ссылка на какой-либо объект по протоколу ftp;

mailto: – вызов приложения Microsoft Outlook для отправки письма по заданному адресу.

Адрес

url или адрес электронной почты.

Назначение

Параметр указывает web-страницу, которая открывается по ссылке. Возможны два варианта:

Интерфейс – открывается страница целевого объекта;

Устройство – открывается страница объекта контроля (устройства). При выборе данного варианта в поле **Параметры** автоматически проставляется значение **&elink=LINK2DEV**.

Открыть в

Способ отображения содержимого ссылки:

новом окне – в новом окне браузера;

новой закладке – в новой вкладке текущего окна браузера;

в текущем окне браузера – в текущем окне браузера (вместо страницы, с которой ссылка вызвана).

Параметры

Параметры ссылки, вводятся при необходимости. См. также [Предопределенные переменные \(теги\) для текстовых строк](#)^[341]. В поле **Ссылка** параметры начинаются после точки с запятой (символ ; формируется автоматически).

Дополнительные параметры для настройки функционирования виджета следует вводить через точку с запятой в формате: **&имя параметра=<значение параметра>**, например: **&fit_to_frame=1**.

Доступные параметры:



- ❖ **fit_to_frame** – при значении **1** или **true** растягивает встраиваемый ресурс в размер рамочного контейнера, по умолчанию **1**;
- ❖ **hide_splash_screen** – при значении **1** не отображается прогресс загрузки инф. панели; актуально, если для контейнера настроена ссылка на другую инф. панель; по умолчанию **0**, т.е. и прогресс загрузки отображается;
- ❖ **overflow**, **overflow-x** или **overflow-y** – настройка показа/скрытия полос прокрутки. Допустимые значения такие же как для стандарта [html5/css](#).

Интервал обновления (сек)

14. Справочники

Интервал (в секундах) автоматического обновления встраиваемого ресурса. При значении **0** обновление отключено. По умолчанию не задано, отключено.

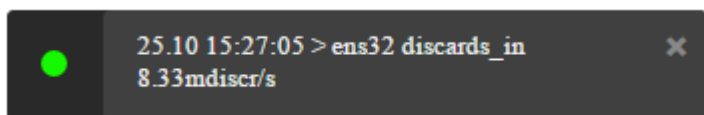
Разрешить раскрытие в размер ИП

Если установлено значение **True**, то при просмотре в Web-интерфейсе рамочный контейнер можно развернуть до размера информационной панели, в которой он содержится, и свернуть обратно. Для этого следует использовать кнопки  и , которые появляются в верхнем левом углу рамочного контейнера. По умолчанию значение параметра не задано; используется значение **False**.

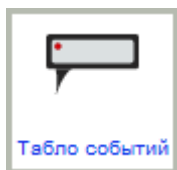
14.8.22. Виджет Табло событий

Виджет **Табло событий** представляет собой динамическую консоль, на которой появляются сообщения о сигналах, привязанных к элементам информационной панели. На информационной панели допустим только один виджет **Табло событий**, он всегда отображается в ее правом нижнем углу.

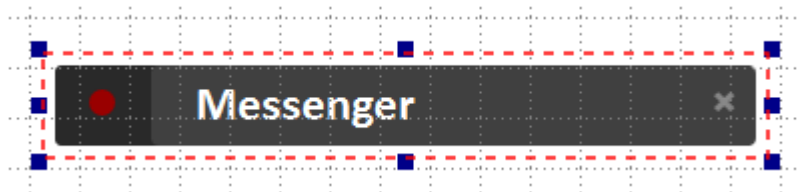
На **Табло событий** каждая строка содержит индикатор типа события и текст сообщения. Пример табло событий на информационной панели:



Иконка для размещения элемента на канве находится на вкладке **Виджеты** в разделе **Графики, диаграммы и прочие**:



Размещенный на канве элемент выглядит следующим образом:



14.8.22.1. Специфичные параметры виджета

Специфичные параметры виджета вводятся в окне **Настройки**, которое вызывается по двойному клику на виджете, размещенном на канве, или с помощью контекстного меню **Параметры виджета**.

Формат сообщения

Шаблон текстовой строки с описанием события, отображаемой на **Табло событий**. В шаблоне можно использовать [предопределенные переменные \(теги\)](#)³⁴¹. Данный формат используется, если не задан формат сообщения для конкретного элемента инф. панели (вкладка **Параметры**, раздел **Общие**, блок **Табло событий**, поле **Формат**).

Индикатор обновления

Видимое количество

Максимальное количество сообщений, отображаемых на табло (от 1 до 50).

Информация, сек.

Продолжительность (в секундах) отображения информационных сообщений типа **Info**.

Нормальные, сек.

Продолжительность (в секундах) отображения сообщений о нормальных состояниях.

Проблема, сек.

Продолжительность (в секундах) отображения сообщений типа **Error**.

Показывать для

Список состояний, для которых сообщения будут отображаться на **Табло событий**.

Переход для

Список состояний, для которых будет разрешен переход на страницу источника данных по клику на сообщении.

Скрыть для

Список состояний, для которых пользователю будет разрешено закрыть сообщение.

Цвет по состоянию

Если флаг установлен, то индикатор в сообщении будет окрашен в цвет статуса сигнала. Если флаг не установлен, то для окраски индикатора будет использоваться таблица цветов по типам сообщений: **state** – цвет индикатора соответствует статусу сигнала; **success / error / info** – цвет индикатора одинаковый при любом статусе сигнала (соответственно: зеленый / красный / зеленый).

Настройки □ ×

Формат сообщения	<input type="text" value="\$TARGET_NAME \$SUBALERT_NAME \$VALUE!"/>	<input type="button" value="Восстановить"/>	
Индикатор обновления	<input type="text" value="Updated"/>	- выводится вместе со временем, если не заданы состояния для отображения. Может быть пустым	
Видимое количество	<input type="text" value="20"/>	- максимальное количество сообщений, выводимых на экран. От 1 до 50	
Информация, сек.	<input type="text" value="5"/>	- время отображения информационного сообщения. От 1 до 600 сек	
Нормальные, сек.	<input type="text" value="2"/>	- время отображения сообщения для нормальных состояний. От 1 до 600 сек	
Проблема, сек.	<input type="text" value="20"/>	- время отображения сообщения для проблемных состояний. От 1 до 600 сек	
Показывать для	<input type="text" value="0,1,2,3,4,5"/>	- список состояний, разделенный запятыми, для которых показывать сообщение. Может быть пустым. Например: 1,2,3,4,5	
Переход для	<input type="text" value="2,3,4,5"/>	- список проблемных состояний, для которых разрешен переход к источнику. Может быть пустым.	
Скрыть для	<input type="text" value="2,3,4,5"/>	- список состояний, для которых показать кнопку 'Закрыть'. Может быть пустым.	
Цвет по состоянию	<input checked="" type="checkbox"/>	- показывать цвета, соответствующие состояниям, вместо стандартных: зеленый для 1 и красный для остальных!	

14.8.22.2. Настройка табло событий

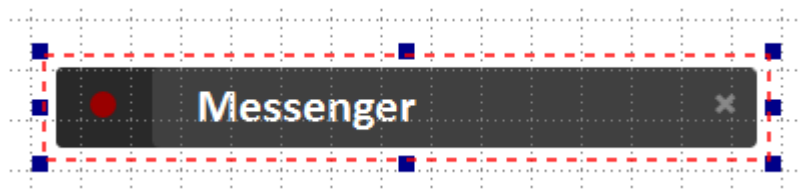
При разработке информационной панели пользователь может настроить:

- ❖ элементы информационной панели, сообщения по которым должны отображаться на **Табло событий**;
- ❖ формат вывода сообщений и продолжительность их показа;
- ❖ возможность перехода по ссылке по клику на сообщении;
- ❖ др. параметры.

Таким образом, для настройки на информационной панели виджета **Табло событий**, необходимо задать не только параметры этого виджета, но и настроить каждый элемент инф. панели, сообщения по которому должны отображаться на **Табло событий**. Параметры, заданные для элементов информационной панели, приоритетнее аналогичных параметров виджета **Табло событий**.

Чтобы на информационной панели настроить виджет **Табло событий**, следует:

1. в области редактирования в любом месте расположить виджет **Табло событий**;
Примечание. Вне зависимости от положения в области редактирования, при просмотре информационной панели в Web-интерфейсе **Табло событий** всегда отображается в правом нижнем углу;



2. настроить параметры виджета **Табло событий** (окно ввода вызывается по двойному клику на виджете, размещенном в области редактирования); см. выше;
3. для элементов, ранее размещенных на информационной панели, настроить необходимость отображения сообщений на **Табло событий** и задать параметры этих сообщений. Для этого на вкладке **Параметры** в разделе **Общие** для параметра **Отображать на табло** установить значение **True** и заполнить блок параметров **Табло событий**:

Тип

Зависимость цвета индикатора на **Табло событий** от статуса сигнала в сообщении: **state** – цвет индикатора соответствует статусу сигнала; **success / error / info** – цвет индикатора одинаковый при любом статусе сигнала (соответственно: зеленый / красный / зеленый).

Формат

Шаблон текстовой строки, отображаемой на **Табло событий**. В шаблоне можно использовать [предопределенные переменные \(теги\)](#)^[341].

Ссылка

Адрес перехода по клику на сообщении в **Табло событий**. По умолчанию – **\$LINK**. Если поле пустое, то переход недоступен.

Показывать время события

Показывать ли время события в тексте сообщения на **Табло событий**.

Продолжительность, сек

Время отображения сообщения на **Табло событий** (в секундах). По истечении заданного интервала сообщение автоматически скрывается.

Можно закрыть

Может ли пользователь принудительно закрыть сообщение на **Табло событий**. По умолчанию – **True**.

Уникальность

Показывать сообщения только по уникальным событиям (**True**) или по всем событиям (**False**).

Показывать при уровне

Статусы сигнала, при которых сообщения отображаются на **Табло событий**. В раскрывающемся списке следует отметить нужные статусы. Если поле не задано, отображаются сообщения при любых статусах.

14.9. Предопределенные переменные (теги) для текстовых строк

Приведенные ниже переменные можно использовать в шаблонах текстовых строк при настройке параметров элементов информационных панелей.

\$DEFAULT – показывать предустановленные значения;

\$IP – сетевой адрес;

\$PROFILE – логин учетной записи;

\$PATH – путь к целевому объекту;

\$DEV_NAME – имя устройства;

\$TARGET_NAME – имя целевого объекта;

\$SUBALERT_NAME – имя сигнала;

\$TEMPLATE_NAME – имя набора сигналов;

\$DESCRIPTION – описание сигнала;

\$VALUE – значение сигнала;

\$UNITS – единицы измерения значения сигнала;

\$PCT – отображать значение в процентах;

\$DATE – дата генерации сигнала;

\$TIME – время генерации сигнала;

\$GOOD_STATES_COUNT – количество сигналов в хороших статусах;

\$BAD_STATES_COUNT – количество сигналов в плохих статусах;

\$TOTAL_STATES_COUNT – общее количество сигналов;

\$SMART_STATES_COUNT(2,3,4) – количество сигналов с заданными статусами. [Числовые константы статусов](#)^[288] перечисляются в скобках через запятую. В приведенном примере значением переменной будет суммарное количество сигналов с желтым (2), красным (3) и бордовым (4) статусом.

\$LEGEND – краткое описание;

\$SEVERITY – критичность сигнала;

\$ALERT_STATUS – список статусов сигналов.

14.10. Функции для настройки действий для элементов информационной панели

В данном справочнике перечислены функции, которые можно использовать для настройки [действий](#)¹⁹², выполняемого при манипуляции пользователем с элементом информационной панели.

Show Specified Element

Показать заданный элемент.

Hide Specified Element

Скрыть заданный элемент.

Show hidden Element, or hide shown

Изменить видимость заданного элемента, т.е. показать его, если он скрыт, или скрыть, если он виден.

TurnOn Visibility Specified Element

Включить доступность заданного элемента.

TurnOff Visibility Specified Element

Выключить доступность заданного элемента.

Click on Specified Element

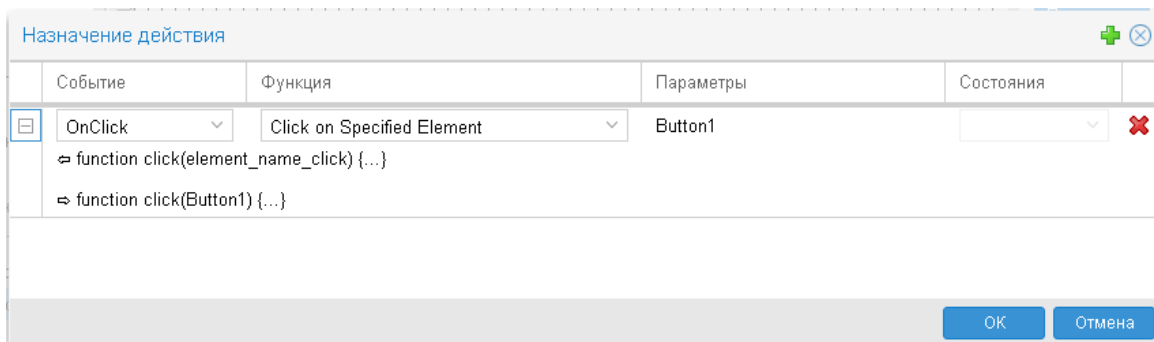
Эмулировать клик мыши на заданном элементе, т.е. выполнить действие, настроенное для события **OnClick** на другом заданном элементе. Данная функция применяется в иерархических информационных панелях, например, для первоначальной загрузки информационной панели или для управления инф. панелями нижнего уровня из инф. панели более высокого уровня через параметры url.

Пример

На информационной панели расположены элементы **Прямоугольник** с именами **Button1** и **Button2**, выполняющие функцию кнопок, и виджет **Рамочный контейнер** с именем **Frame**. Для кнопки **Button1** по событию **OnClick** настроено действие **Show hidden Element, or hide shown** – управление видимостью виджета **Frame**.

Назначение действия				
Событие	Функция	Параметры	Состояния	
OnClick	Show hidden element, or hide shown	Frame		✖
<pre>⇒ function changevis(element_name) {...} ⇒ function changevis(Frame) {...}</pre>				
			OK	Отмена

Для кнопки **Button2** по событию **OnClick** настроено действие **Click on Specified Element** – эмуляция клика мыши на кнопке **Button1**, т.е. кнопка **Button2** также управляет видимостью виджета **Frame**.



Set Fill Color

Изменить цвет заданного элемента; цвет задается наименованием (например: **red**, **green**, **blue**, **yellow**, **black**, **brown**, **white**) либо шестнадцатеричным значением в системе RGB, например: **#ab23ff** или **ab23ff** (символ # использовать не обязательно).

Set Stroke Color

Изменить цвет линии заданного элемента; цвет задается аналогично предыдущей функции.

Set Fill Opacity

Изменить прозрачность элемента; задается числом от **0** (полностью прозрачный) до **1** (не прозрачный), например: **0.5**, **0.2**.

Offset Up-Left

Сдвинуть элемент по диагонали на заданное количество пикселей); если смещение указано без знака, то элемент сдвигается на одинаковое количество пикселей вверх и влево, если со знаком "минус" – вниз и вправо.

Start Animation

Для заданного элемента запустить заданную анимацию.

Stop Animation

Для заданного элемента остановить анимацию.

Set Iframe URL

Назначение адреса для заданного виджета [Рамочный контейнер](#)³³⁶. Функция принимает в качестве параметра имя соответствующего виджета **Рамочный контейнер** и адрес для отображения в виджете. Если адрес является внутренним (ссылка на другую информационную панель или целевой объект в дереве конфигурации), при заполнении поля **Параметры** перед адресом необходимо вписать строку **link;**

Change dashboard localization

Изменить локализацию информационной панели. В поле **Параметры** следует ввести язык, на который следует переключить информационную панель: **ru** или **en**. В результате выполнения действия на информационной панели отображаются элементы, у которых параметр **Локализация** не задан или совпадает с заданным.

Change Target Index of Specified Element

Для заданного элемента с множественным назначением показать сигнал с заданным Индексом назначения.

14. Справочники

Hide dashboard

Отобразить информационную панель в заданном **Рамочном контейнере**. В поле **Параметры** следует ввести имя виджета **Рамочный контейнер**.

Show dashboard

Скрыть информационную панель в заданном **Рамочном контейнере**. В поле **Параметры** следует ввести имя виджета **Рамочный контейнер**.

Evaluate Script

Выполнить предопределённую функцию; в поле **Параметры** следует ввести имя выполняемой функции.

Set x and y position

Переместить заданный элемент в указанную позицию. В поле **Параметры** следует ввести имя элемента и значения координат x и y для его левой верхней точки (начало координат находится на канве в левой верхней точке).

Set height and width

Изменить высоту и ширину заданного элемента. В поле **Параметры** следует ввести имя элемента и новые значения для его параметров **Высота** и **Ширина**.

Set x and y position, height and width

Изменить местоположения и размеры элемента (объединение двух предыдущих действий).