

## АННОТАЦИЯ

Настоящее руководство содержит сведения, необходимые пользователю для установки, настройки и эксплуатации программной платформы «СКАДА А-СОФТ» (далее по тексту СКАДА), предназначенной для реализации процесса обработки информации в составе работающей системы верхнего уровня (СВУ) автоматизированной системы управления технологическим процессом (АСУ ТП).

## СОДЕРЖАНИЕ

1.	Назначение программы .....	4
2.	Условия применения программы .....	5
2.1	Состав аппаратных средств .....	5
2.2	Состав программных средств.....	5
2.3	Требования к квалификации персонала .....	5
3.	Установка программы .....	6
3.1	Установка дополнительных пакетов в ОС Astra Linux 1.5 .....	6
3.2	Установка дополнительных пакетов на ОС Astra Linux 1.6 .....	9
3.3	Установка ПП «СКАДА А-СОФТ».....	11
4.	Обращение к программе.....	13
4.1	Запуск программы .....	13
4.2	Останов программы .....	14
5.	Конфигурация и настройка системы.....	15
5.1	Конфигурация системных параметров.....	15
6.	Установка конвертера Getdb.exe .....	22
7.	Установка дополнительных библиотек элементов в ПП «СКАДА А-СОФТ».....	22
8.	Настройка запуска ПП «СКАДА А-СОФТ» в безопасном режиме .....	23
9.	Установка интегрированной справочной системы.....	29
10.	Входные и выходные данные .....	30
10.1	Входные данные .....	30
10.2	Выходные данные .....	30
11.	Сообщения.....	31
11.1	Описание журнала активных тревог и журнала тревог и событий.....	31
11.1.1	Функциональное назначение .....	31
11.1.2	Описание журнала активных тревог и журнал тревог и событий.....	31
11.1.3	Взаимодействие журнала активных тревог и журнала тревог и событий с модулями СКАДА .....	31
11.1.4	Элементы интерфейса журнала активных тревог и журнал тревог и событий.....	33
	ПРИЛОЖЕНИЕ Конфигурационный файл СКАДА и параметры командной строки вызова СКАДА.....	37
	Перечень принятых сокращений.....	47

## **1. Назначение программы**

ПП «СКАДА А-СОФТ» предназначена для разработки и исполнения прикладного программного обеспечения, предназначенного для сбора, архивирования, визуализации информации, выдачи управляющих воздействий, а также других родственных операций, характерных для полнофункциональной системы управления и сбора данных.

## 2. Условия применения программы

### 2.1 Состав аппаратных средств

Аппаратные требования СКАДА-системы для её исполнения в зависимости от сложности проекта приведены в таблице 1.

Таблица 1

	Требования к аппаратной части
<i>Проекты более 30000 точек</i>	
АРМ	Процессор: Intel Core i3 2 ГГц; ОЗУ: 8 Гб; Жесткий диск: 40 Гб
Сервер	Процессор: Intel Xeon E5; ОЗУ: 128 Гб; Жесткий диск: 40 Гб
Сервер БД	Процессор: Intel Xeon E5; ОЗУ: 64 Гб; Жесткий диск: 1 Тб, RAID 10
Шлюз	Процессор: Intel Core i3 2 ГГц; ОЗУ: 2 Гб; Жесткий диск: 40 Гб
<i>Проекты до 5000 точек</i>	
АРМ-сервер	Процессор: Intel Core i7 3 ГГц; ОЗУ: 16 Гб; Жесткий диск: 300 Гб, RAID 5
Шлюз	Процессор: Intel Core i3 2 ГГц; ОЗУ: 2 Гб; Жесткий диск: 40 Гб
<i>Проекты до 500 точек</i>	
АРМ-сервер	Процессор: Intel Core i5 2 ГГц; ОЗУ: 16 Гб; Жесткий диск: 300 Гб

### 2.2 Состав программных средств

Для обеспечения работоспособности ПП «СКАДА А-СОФТ» необходимо наличие операционной системы Astra Linux SE Smolensk 1.5 или Astra Linux SE Smolensk 1.6 (x64).

### 2.3 Требования к квалификации персонала

Специалисты, занимающиеся настройкой ПП «СКАДА А-СОФТ», должны иметь знания для работы с системой Astra Linux на уровне пользователя.

### 3. Установка программы

#### 3.1 Установка дополнительных пакетов в ОС Astra Linux 1.5

Для установки СКАДА из исходных кодов требуется установить дополнительные пакеты. Далее будет рассмотрен алгоритм установки дополнительных пакетов на примере ОС Astra Linux 1.5.

Для этого необходимо войти в систему под суперпользователем «root» и запустить «Менеджер пакетов Synaptic» из меню «Пуск» → «Настройки».

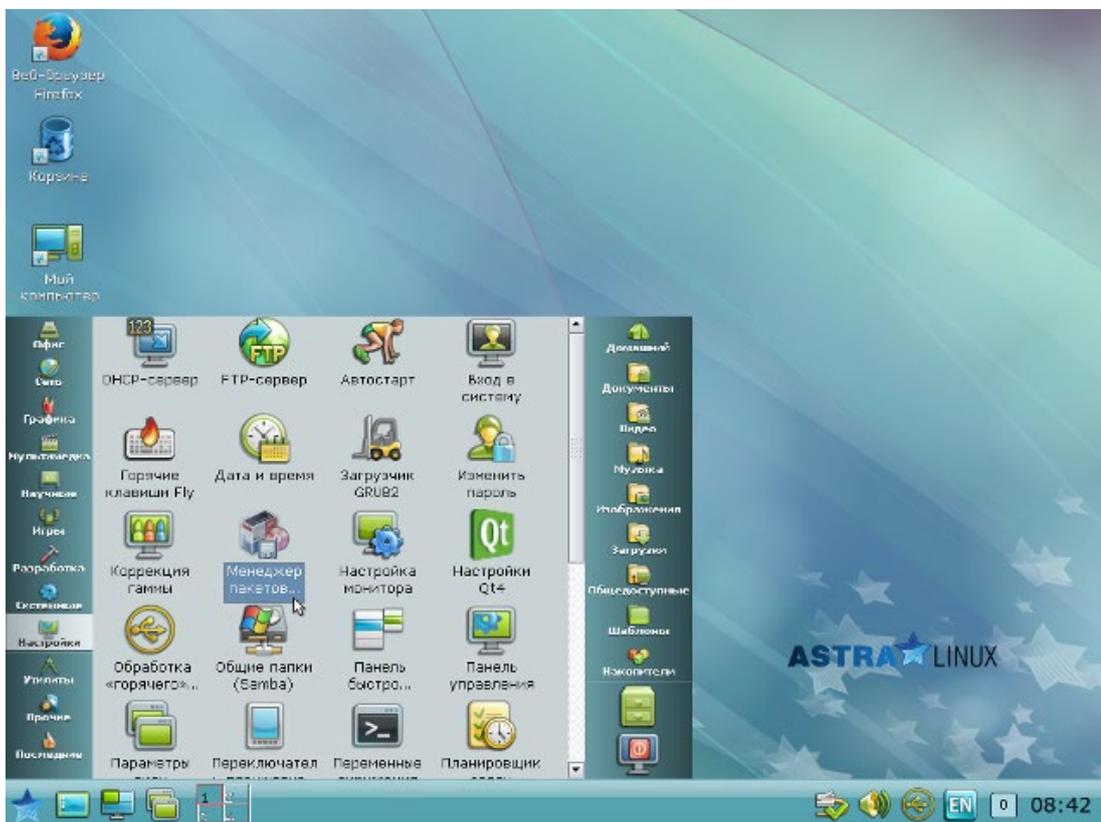


Рисунок 1

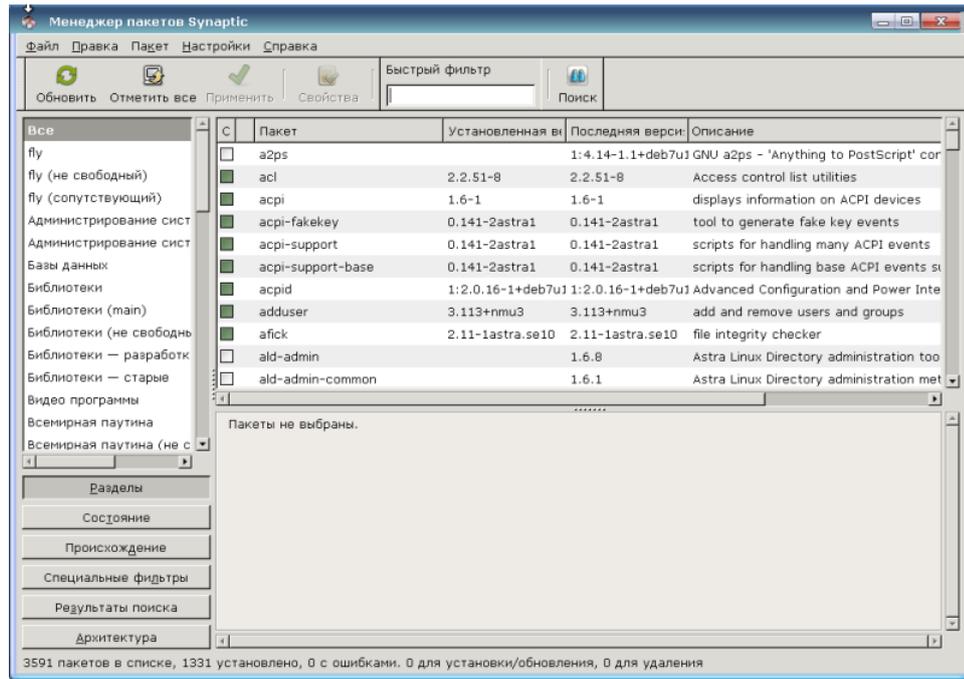


Рисунок 2

В строке «быстрого фильтра» необходимо ввести название нужного пакета. Затем с помощью мыши пометить его для установки и нажать кнопку «Применить».

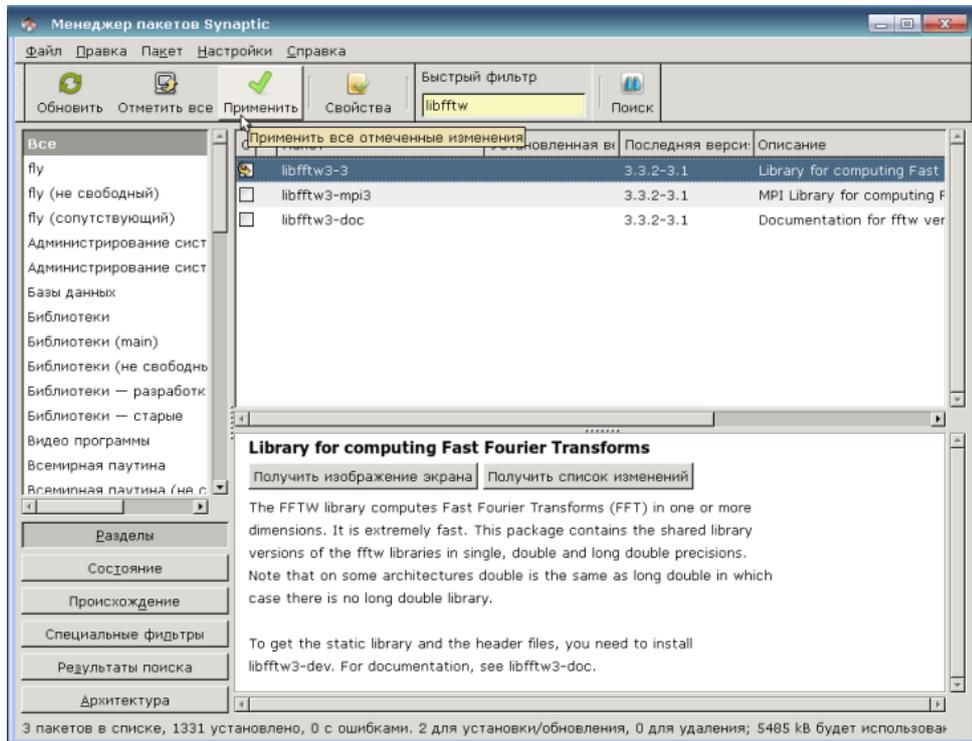


Рисунок 3

Затем появится окно проверки списка выполняемых изменений (рисунок 4), в котором необходимо нажать кнопку «Применить». При этом вместе с выбранными пакетами будут автоматически выбраны и установлены дополнительные необходимые пакеты.

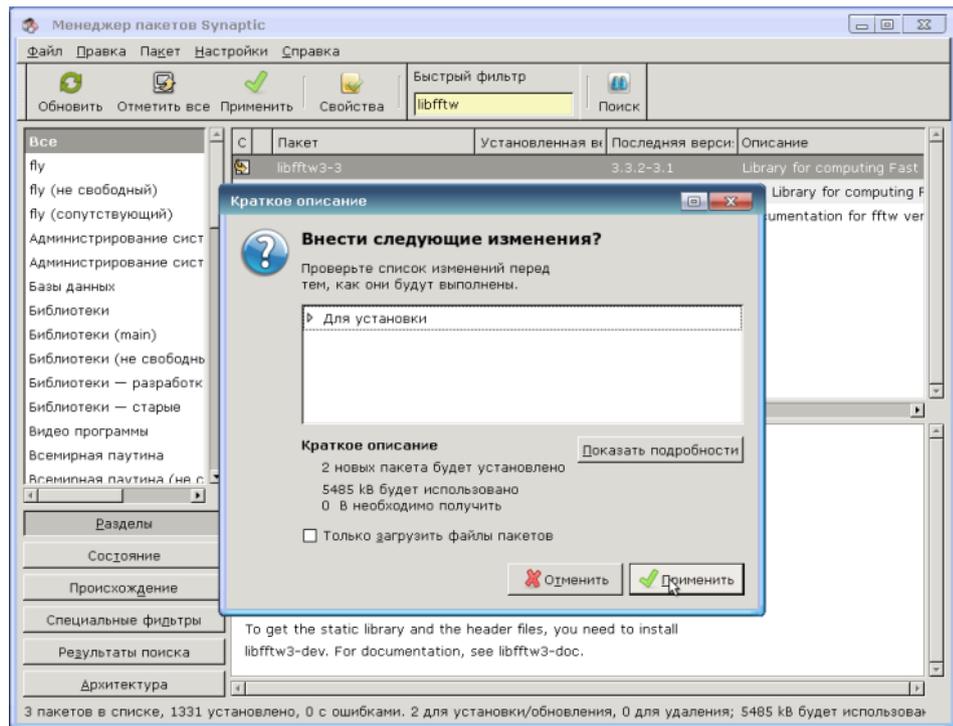


Рисунок 4

Затем система потребует вставить диск с дистрибутивом ОС Astra Linux 1.5 релиз «Смоленск» (рисунок 5). Вставьте диск в дисковод и нажмите кнопку «ОК».

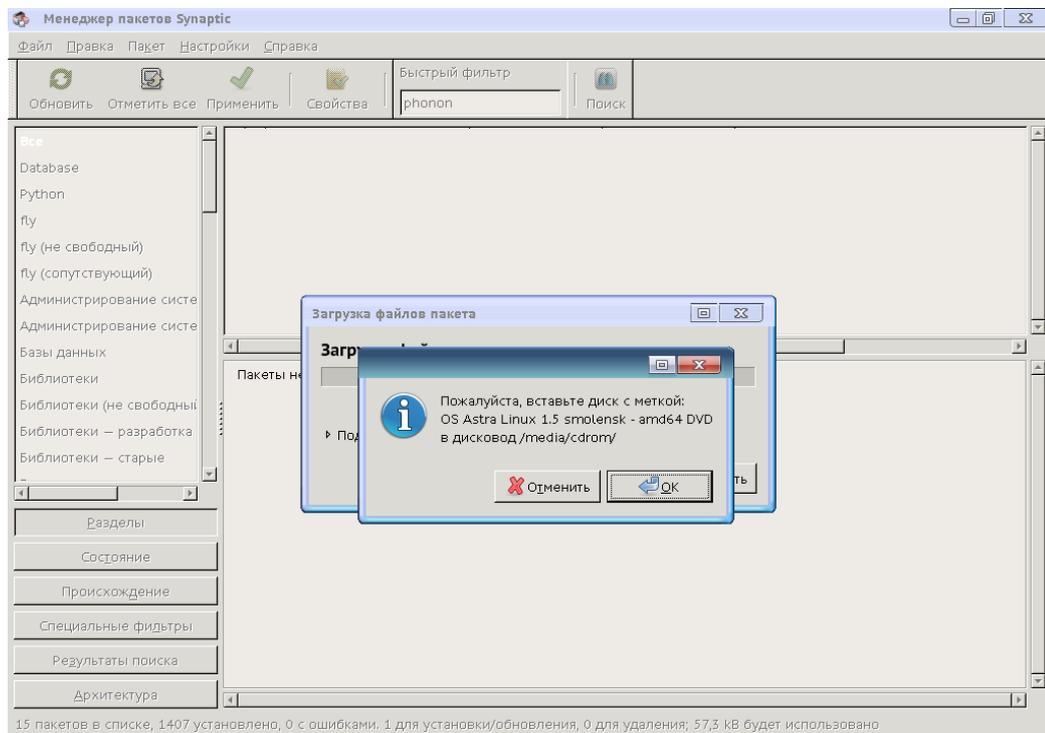


Рисунок 5

Далее на экране будет отображаться процесс установки пакетов до появления окна «Изменения применены» (рисунок 6). Для окончания установки в данном окне следует нажать кнопку «Закрыть».

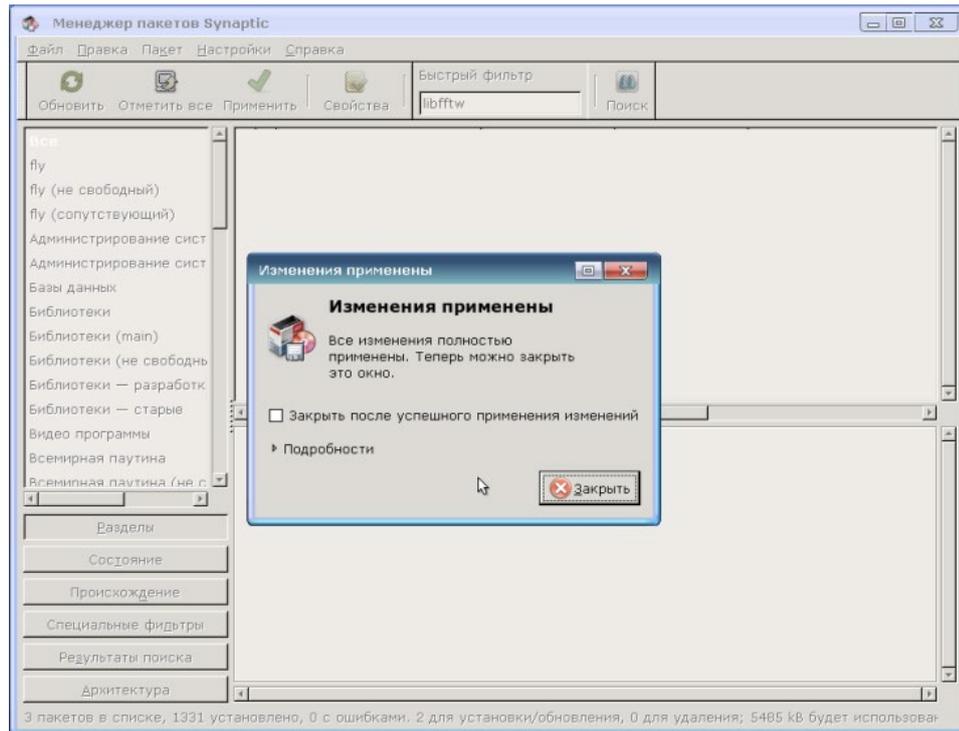


Рисунок 6

Описанным выше образом, для ОС «Astra Linux 1.5» необходимо установить следующие пакеты:

- libfftw3-3
- phonon
- libqscintilla2-11
- libodbc1
- libmysqlclient18
- libfbclient2

### 3.2 Установка дополнительных пакетов на ОС Astra Linux 1.6

Алгоритм установки дополнительных пакетов аналогичен установке на ОС Astra Linux 1.5.

В меню «Пуск» выбрать «Панель управления» или «Настройки»(в зависимости от вида рабочего стола) → «Программы» → «Менеджер пакетов Synaptic» (рисунок 7).

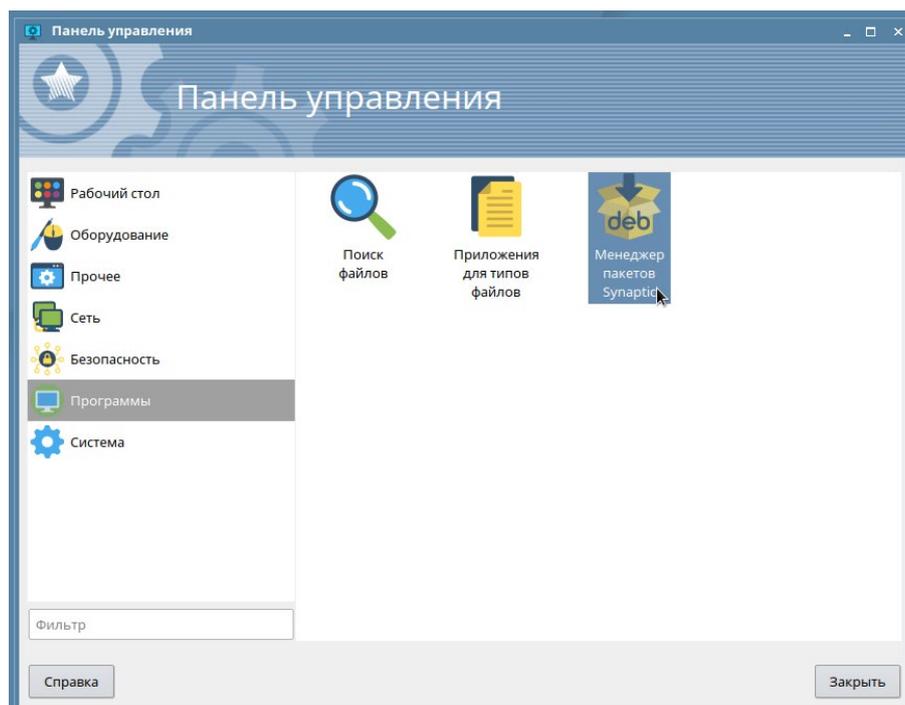


Рисунок 7

Далее в окне «Менеджера пакетов Synaptic» необходимо нажать на иконку «Поиск» (рисунок 8) и в появившемся окне ввести название устанавливаемого пакета:

- libqscintilla2-12v5.

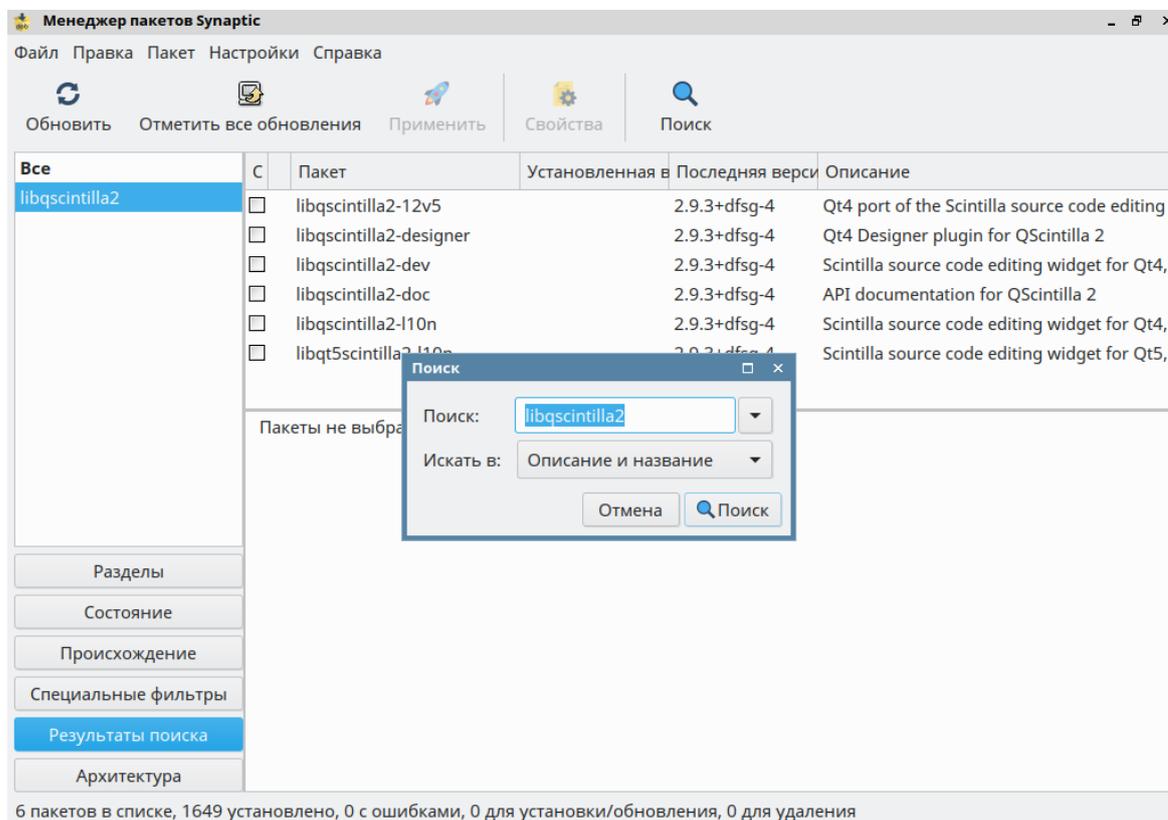


Рисунок 8

После этого, в отобразившемся списке выбрать устанавливаемый пакет двойным нажатием мыши, и согласиться с установкой дополнительных библиотек, нажав на кнопку «Применить». Проверив список изменений, нажать кнопку «Применить». Система потребует вставить диск с дистрибутивом ОС Astra Linux 1.6. После чего начнется установка выбранных пакетов.

Также требуется установить пакет:

- libqtwebkit4.

### 3.3 Установка ПП «СКАДА А-СОФТ»

Перед первой установкой ПП «СКАДА А-СОФТ» необходимо установить три deb-пакета с установочного диска. Для этого с установочного диска необходимо скопировать файлы *omniorb\_4.1.7\_amd64.deb*, *zeromq\_3.2.5\_amd64.deb* и *integrationplatform\_2.1.3\_amd64.deb* в каталог */var/opt/SCADA/*. Далее, запустив терминал Fly, перейти в каталог */var/opt/SCADA/* командой:

```
cd /var/opt/SCADA
```

и запустить установку deb-пакетов, поочередно выполнив команды:

```
dpkg -i omniorb_4.1.7_amd64.deb
```

```
dpkg -i zeromq_3.2.5_amd64.deb
```

```
dpkg -i integrationplatform_2.1.3_amd64.deb
```

Для установки ПП «СКАДА А-СОФТ» необходимо в зависимости от версии ОС, с установочного диска скопировать deb-пакет с соответствующим названием СКАДА:

для ОС «Astra Linux 1.5» – *scada\_x.xx-rx\_4.20\_amd64.deb*,

для ОС «Astra Linux 1.6» – *scada\_x.xx-rx\_4.15\_amd64.deb*

(где *x*- номер версии СКАДА) в каталог */var/opt/SCADA/*.

Затем, в терминале Fly запустить установку СКАДА, набрав команду:

```
dpkg -i scada_x.xx-rx_4.20_amd64.deb
```

До выполнения команды *dpkg* должен быть закрыт «Менеджер пакетов Synaptic» (рисунок 9).

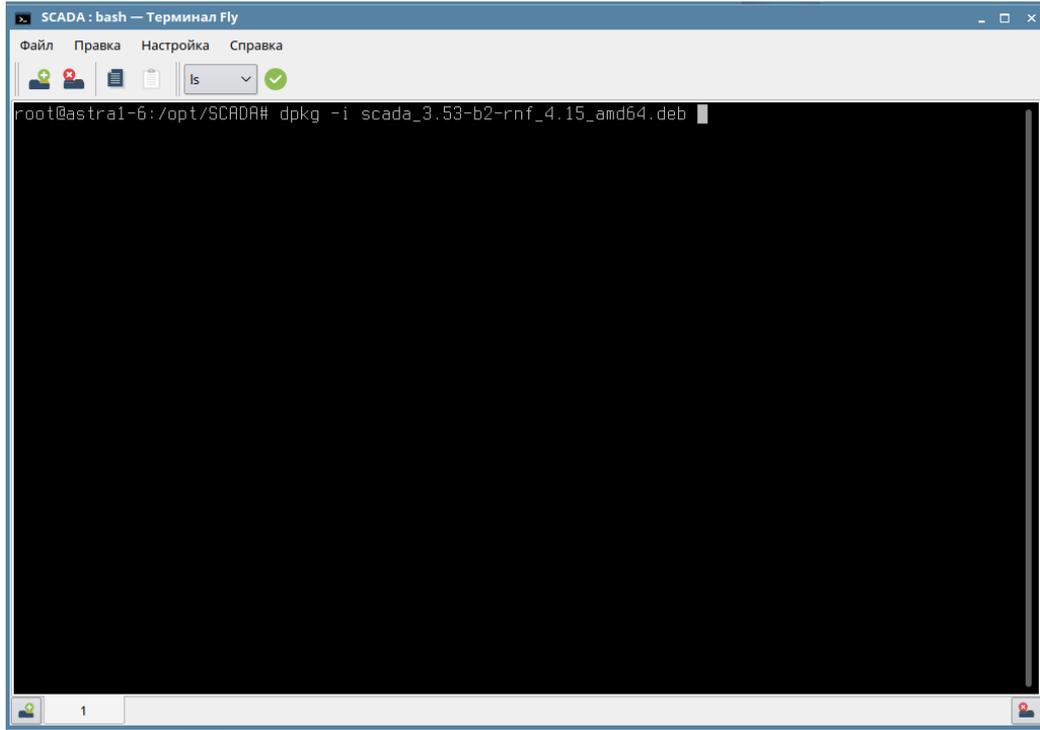


Рисунок 9

После завершения установки СКАДА окно терминала будет иметь вид, показанный на рисунке 10.

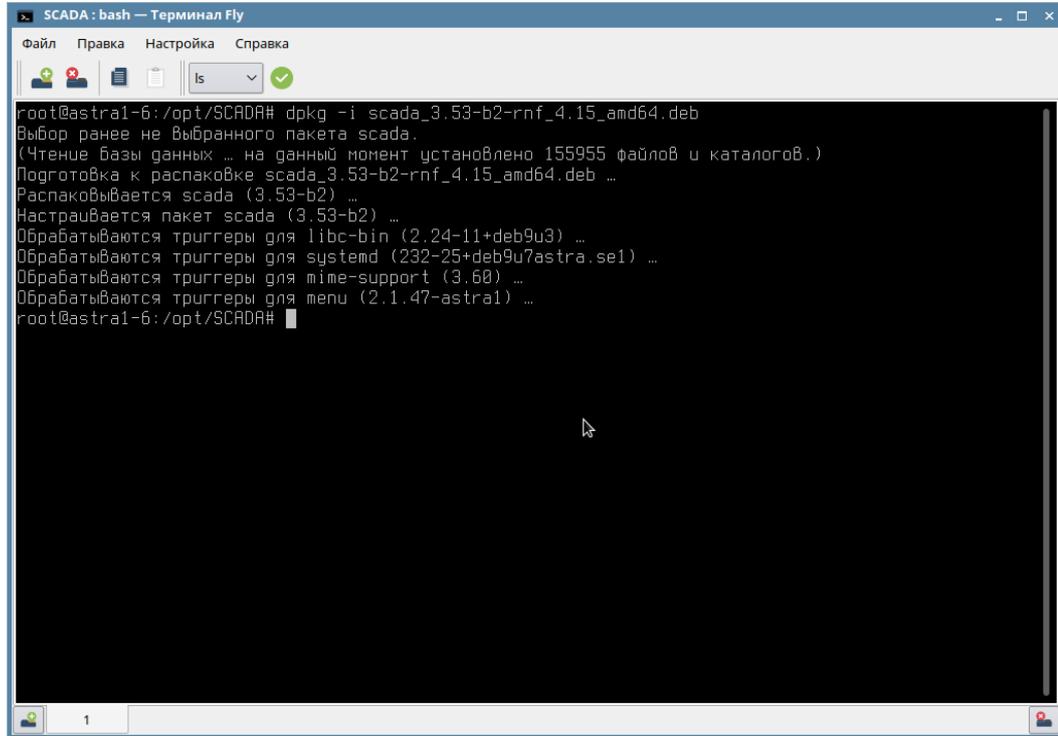


Рисунок 10

После установки СКАДА необходимо изменить права на доступ к основному конфигурационному файлу - scada.xml командой:

```
chmod a+w /etc/scada.xml.
```

## 4. Обращение к программе

### 4.1 Запуск программы

Запуск программы осуществляется командой `scada` из командной строки терминала `Fly` или из графического меню «Пуск» → «Графика» → «SCADA» с помощью мыши.

После загрузки системы появится окно регистрации (рисунок 11).

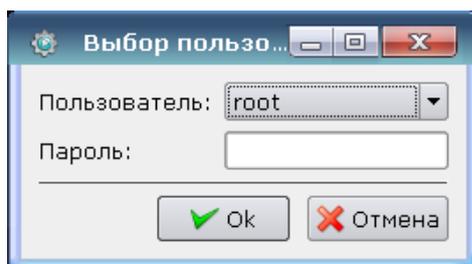


Рисунок 11

После регистрации в СКАДА (по умолчанию пользователь: `root`, пароль: `root`) на экране появится окно системного конфигуратора СКАДА (рисунок 12). Загрузка ПП «СКАДА А-СОФТ» считается корректной, если в ходе ее запуска не выдается ошибок и на экране появляется окно системного конфигуратора.

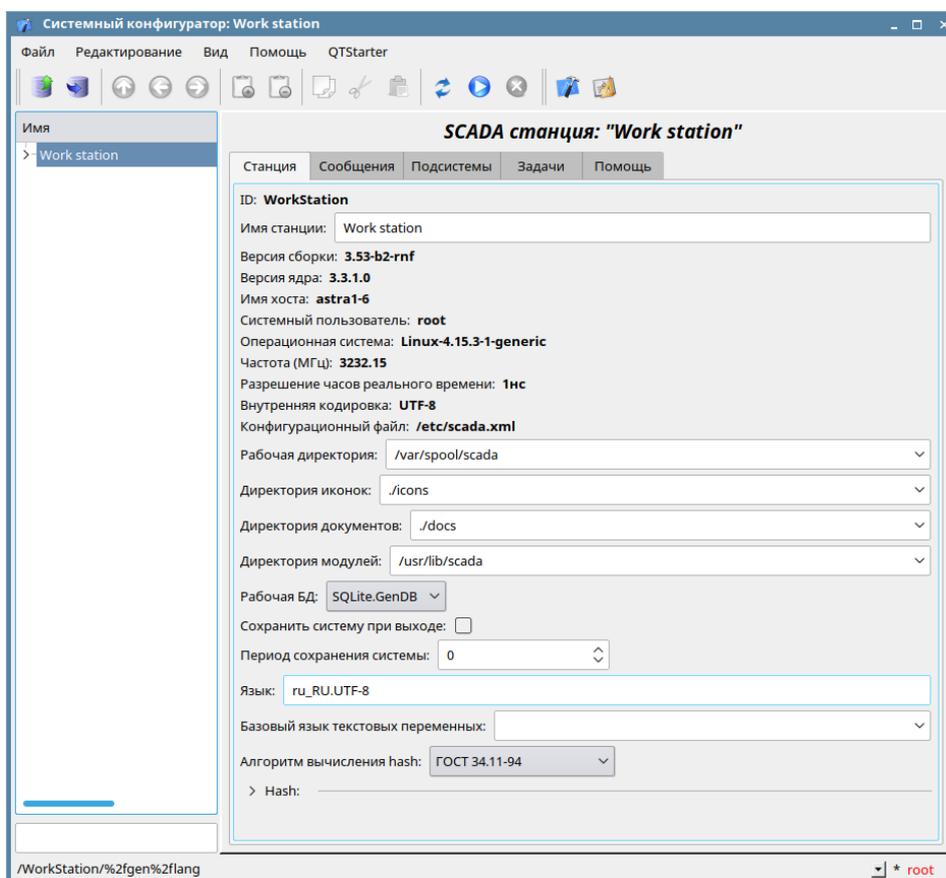


Рисунок 12

Конфигуратор является основным и достаточным средством для конфигурации любого компонента ПП «СКАДА А-СОФТ».

Из окна конфигуратора можно открыть окно разработки графических интерфейсов пользователя или запустить проект пользовательского интерфейса на исполнение.

#### 4.2 Останов программы

Для выхода из программы необходимо в пункте меню «Файл» выбрать строку «Выход». ПП «СКАДА А-СОФТ» будет остановлена.

## 5. Конфигурация и настройка системы

Конфигурирование СКАДА осуществляется при помощи модуля конфигурации - UI.QTCfg. Данный модуль предоставляет развитый интерфейс конфигурации позволяющий управлять, как локальной станцией, так и удалёнными станциями в локальной и глобальной сетях, включая безопасное соединение.

Значения конфигурации, изменённые в конфигураторе, а также большинство данных сохраняются в базах данных (БД). Учитывая модульность подсистемы "БД", ими могут быть различные БД. Причём предоставляется возможность хранения разных частей СКАДА как в разных БД одного типа, так и в БД разных типов.

Кроме БД данные о конфигурации могут содержаться в конфигурационном файле СКАДА, а также передаваться посредством параметров командной строки при вызове СКАДА. Сохранение конфигурации в конфигурационном файле осуществляется наравне с БД. Типовым именем конфигурационного файла является /etc/scada.xml. Формат конфигурационного файла и параметры командной строки рассматриваются в приложении.

Многие настройки и конфигурация объектов СКАДА, которые исполняются или уже включены, не применяются сразу же по внесению изменений, поскольку конфигурация читается/применяется обычно только при включении или запуске. Поэтому, для применения изменений в таких случаях, достаточно включить/выключить включенный объект или перезапустить исполняющийся — остановить/запустить.

### 5.1 Конфигурация системных параметров

Конфигурация системных параметров размещается на пяти вкладках корневой страницы станции.

Вкладка "Станция" содержит основные информационные и конфигурационные параметры станции. Ее вид показан на рисунке 13.

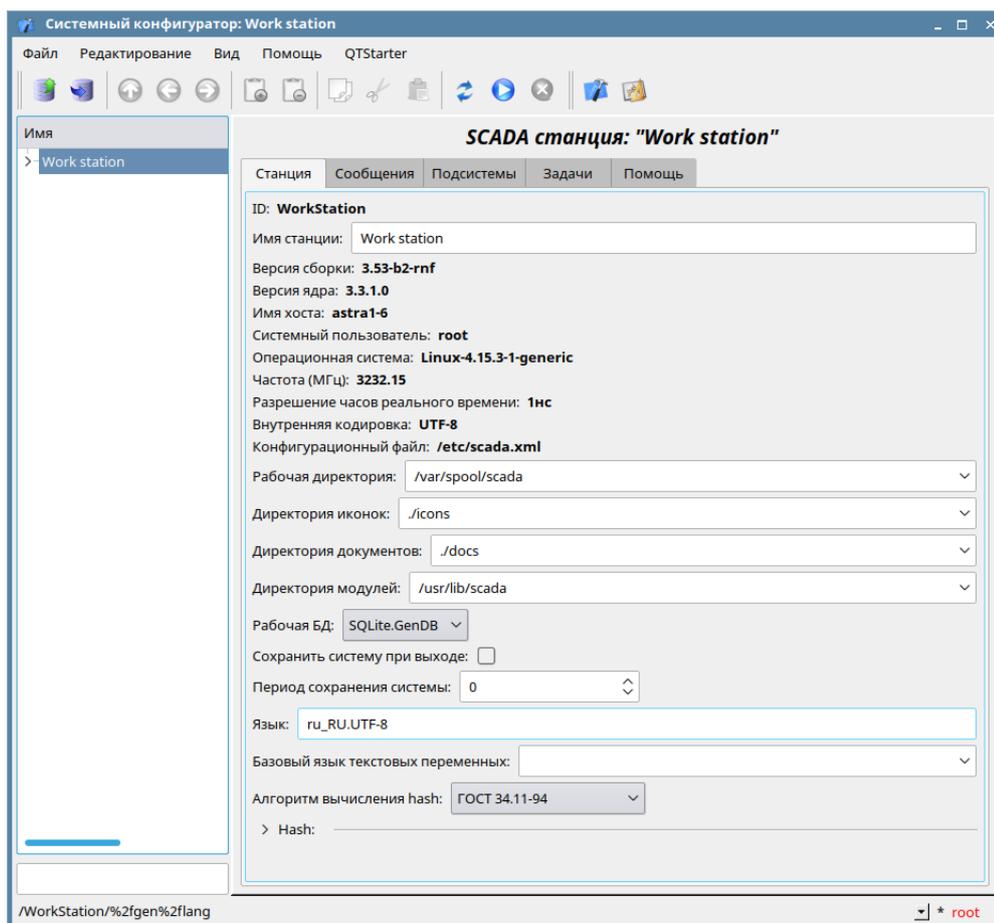


Рисунок 13

На вкладке "Станция" представлены следующие поля:

- *ID* - содержит информацию об идентификаторе станции. Указывается параметром командной строки - Station. При загрузке ищется соответствующий идентификатору станции раздел в конфигурационном файле и, если не обнаруживается, то используется первый доступный;
- *Имя станции* - указывает локализованное имя станции;
- *Версия сборки* - содержит информацию о текущей версии программы;
- *Версия ядра* - содержит информацию о текущей версии ядра системы;
- *Имя хоста* - содержит информацию об имени машины, на которой запущена станция.
- *Системный пользователь* - содержит информацию о пользователе, от имени которого выполняется программа в ОС;
- *Операционная система* - содержит информацию об имени и версии ОС, ядре ОС, на которой исполняется программа;

- *Частота (МГц)* - содержит информацию о частоте центрального процессора, которым исполняется программа. Значение частоты проверяется раз в 10 секунд и позволяет отслеживать её изменение, например, механизмами управления питанием;
- *Разрешение часов реального времени (нс)* - содержит информацию о возможности или разрешении часов реального времени ОС. Позволяет сориентироваться с минимальным интервалом времени периодических задач, например, для задач сбора данных;
- *Внутренняя кодировка* - содержит информацию о кодировке, в которой хранятся текстовые сообщения внутри программы;
- *Конфигурационный файл* - содержит информацию о конфигурационном файле, используемом программой. Устанавливается параметром командной строки – Config;
- *Рабочая директория* - указывает на рабочую директорию станции. Используется в относительной адресации объектов на файловой системе, например, файлов БД. Допускает изменение пользователем для сохранения данных системы в другую БД. При этом значение этого поля не сохраняется в БД, а может быть изменено только в секции "WorkDB" конфигурационного файла;
- *Директория иконок* - указывает на директорию, содержащую иконки программы. Если в дереве навигации конфигуратора отсутствуют иконки, то неправильно указано значение этого поля;
- *Директория документов* - указывает на директорию, содержащую справочную документацию по СКАДА;
- *Директория модулей* - указывает на директорию модулей для СКАДА. Если значение этого поля некорректно, то при запуске на экране будет отображена только информация в консоли о корректном запуске ядра СКАДА;
- *Рабочая БД* - указывает на рабочую базу данных (БД), а именно на БД, используемую для хранения основных данных программы. Изменение этого поля отмечает все объекты программы как модифицированные, что позволяет сохранить или загрузить данные станции из указанной основной БД;
- *Сохранять систему при выходе* - указывает на необходимость сохранения изменённых данных при завершении работы;
- *Период сохранения системы* - указывает период в секундах, с которым необходимо сохранять изменённые данные станции;

- *Язык* - указывает на язык сообщений программы. Изменение этого поля допустимо, однако приводит к изменению языка сообщений только для интерфейса и динамических сообщений;

- *Базовый язык текстовых переменных* - используется для включения режима поддержки многоязыковых текстовых переменных. Значение базового языка выбирается из списка двухсимвольных кодов языков, обычно только текущий и базовый языки в списке. Далее для текстовых переменных на небазовом языке в таблицах БД будут создаваться отдельные колонки. Под текстовыми переменными подразумеваются все текстовые поля конфигулятора, которые могут быть переведены на другой язык. Числа и другие символьные значения к их числу не относятся и не переводятся;

- *Алгоритм вычисления hash* – позволяет выбрать алгоритм вычисления контрольной суммы элемента. Для выбора доступны алгоритмы в соответствии с ГОСТ 34.11-94 и ГОСТ 34.11-2012 (описание настройки приведено в Руководстве программиста);

- *Hash* – значение вычисленной контрольной суммы.

Вкладка "Сообщения" - это раздел группы параметров, управляющих работой с сообщениями станции (рисунок 14), включает в себя:

- *Таблица «Уровни сообщений»* – указывает на уровень сообщений, начиная с которого необходимо их обрабатывать. Сообщения ниже этого уровня будут игнорироваться. Необходимо, например, для исключения из обработки отладочных сообщений уровня 0. Для каждого уровня сообщения возможно выбрать логирование – указав «Тгуе» в соответствующей ячейке;

- *В системный логер (syslog)* – указывает на необходимость направления сообщений в системный логер, механизм ОС для работы с сообщениями системы и ПО. При включении этого параметра появляется возможность управлять и контролировать сообщения СКАДА механизмами ОС;

- *На стандартный выход (stdout)* – указывает на использование стандартного механизмы вывода в консоль. Выключение этого свойства исключит весь вывод в консоль, если не указан следующий параметр;

- *На стандартный выход ошибок(stderr)* – указывает на использование стандартного механизма вывода ошибок, обычно тоже направляется в консоль;

- *В архив* – указывает на необходимость вывода сообщений в архив сообщений СКАДА. Этот параметр обычно включен, а его выключение приводит к фактическому отключению архивирования сообщений на станции.

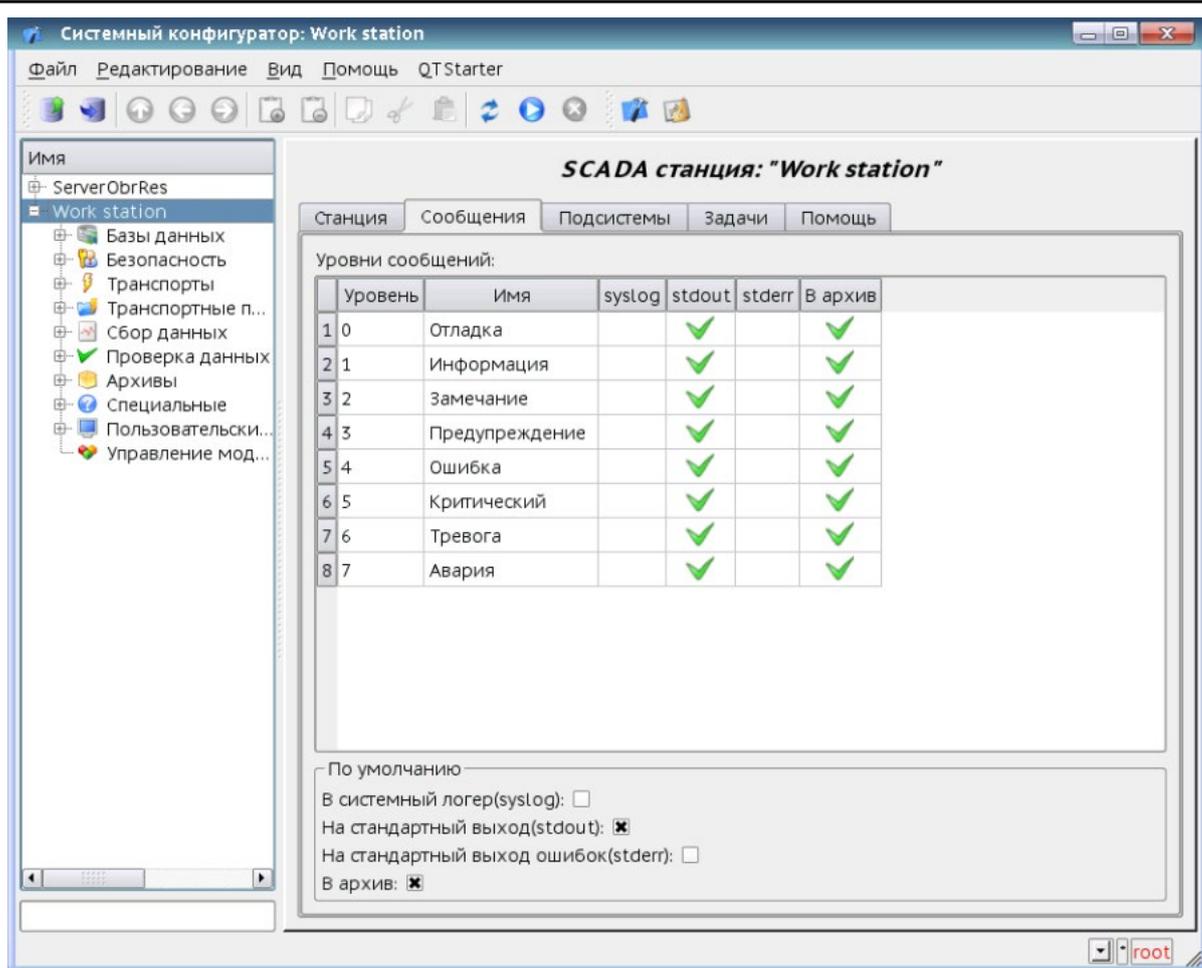


Рисунок 14

Вкладка "Подсистемы" содержит список подсистем и позволяет выполнять прямые переходы к ним с помощью контекстного меню. Ее вид показан на рисунке 15.

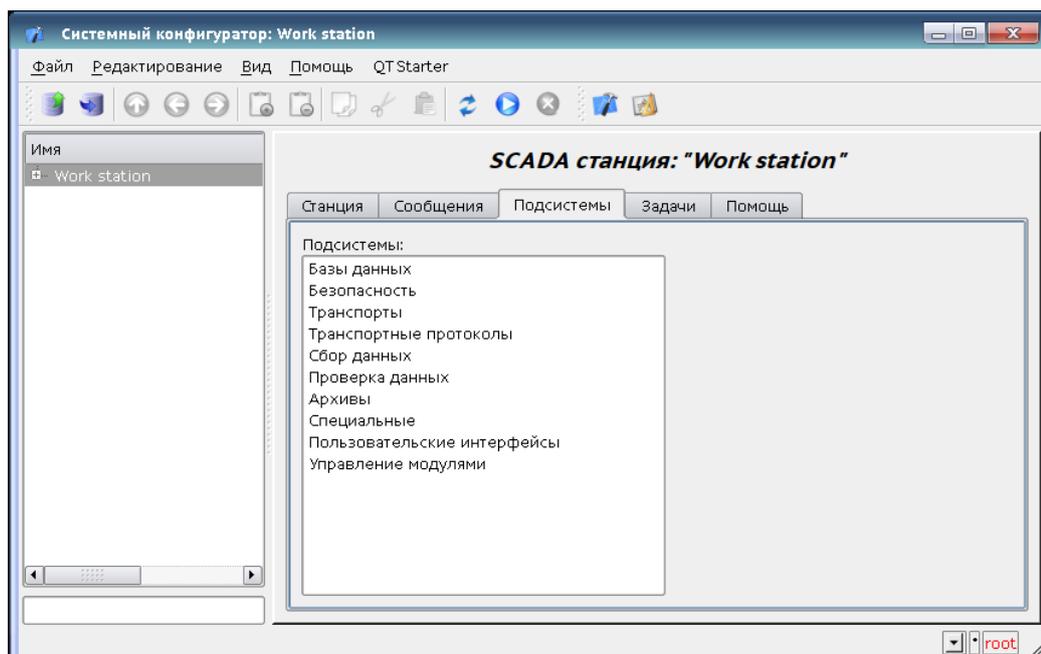


Рисунок 15

Вкладка "Задачи" содержит таблицу со списком задач открытых различными компонентами СКАДА. Из таблицы можно получить различную информацию о задачах, а также в колонке «УСТ.СРУ» назначить процессоры для задач, на многопроцессорных системах.

Вид вкладки "Задачи" показан на рисунке 16.

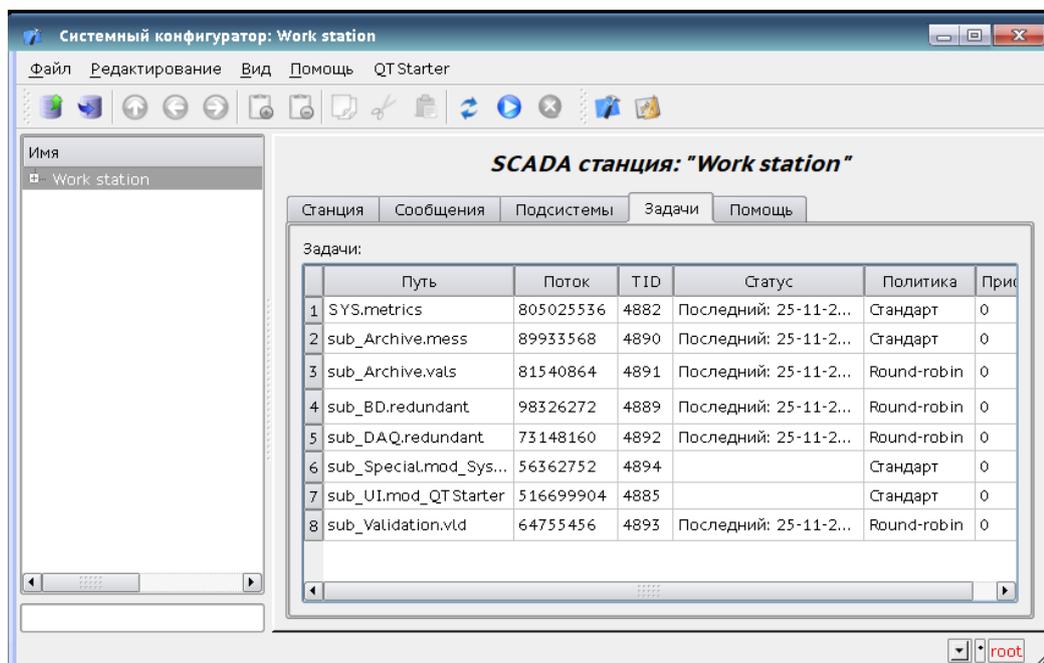


Рисунок 16

Вкладка "Помощь" содержит краткую помощь для данной страницы. Ее вид показан на рисунке 17.

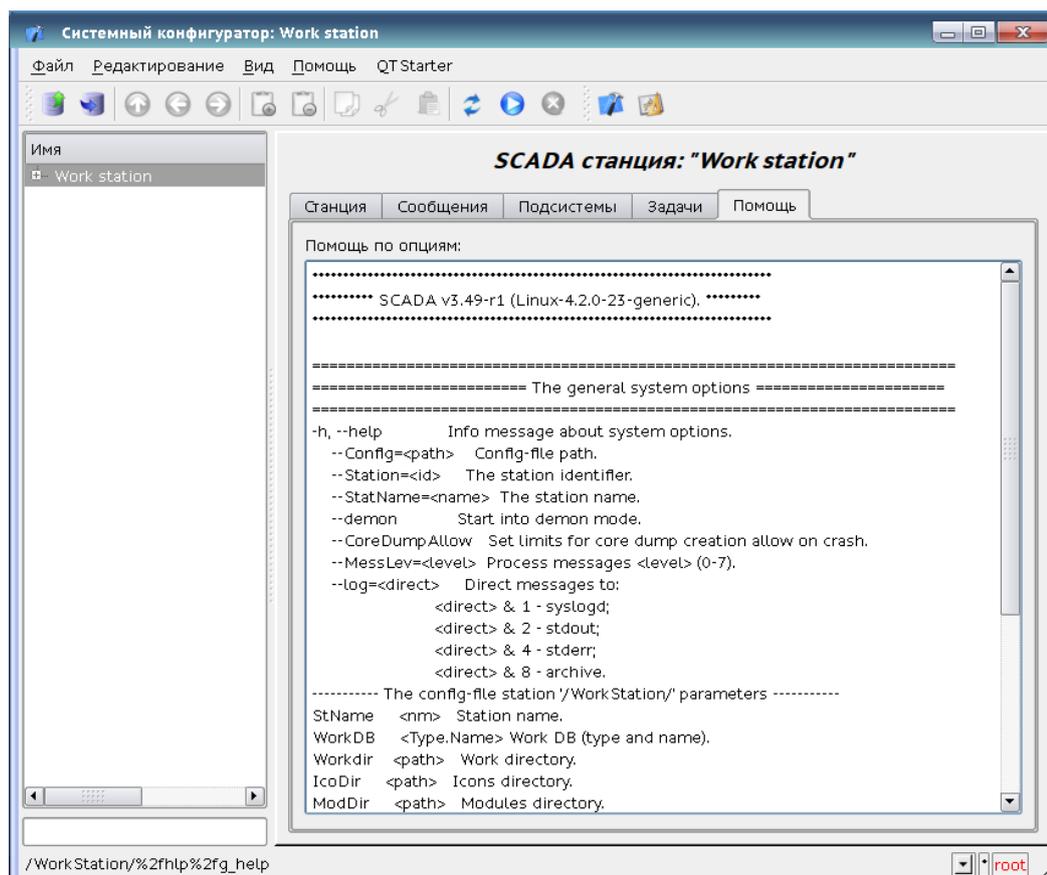


Рисунок 17

Кроме того, справочная информация о работе с системой, конфигурировании и настройкам модулей подсистемы доступны для чтения в пункте меню «Помощь» → «Справка» или нажатием клавиши «F1». Для использования данной опции необходимо предварительно установить интегрированную справочную систему в соответствии с пунктом 9.

**Примечание.** Для модификации полей корневой страницы станции могут потребоваться права привилегированного пользователя. Получить такие права можно, включив пользователя в группу суперпользователя "root", или, войдя на станцию от имени суперпользователя "root".

Поля идентификаторов всех объектов СКАДА недопустимы для прямого редактирования, поскольку являются ключом для хранения данных объектов в БД. Поменять идентификатор объекта можно с помощью команды переноса и последующей вставки объекта в конфигураторе.

Подробно описание объектов СКАДА описано в документе «Руководство оператора.».

## **6. Установка конвертера Getdb.exe**

Формирование файлов баз данных из GET-проекта производится программой *getdb.exe*, которая реализована для ОС Windows и ОС Astra Linux 1.6. Результатом работы программы являются базы данных INFO.db, Protocol.db, ProjectBase.db в формате СУБД SQLite.

Дистрибутивы программы содержатся в каталоге *Translator* на поставляемом диске в папках соответствующих ОС. Для установки программы на ПК независимо от ОС необходимо скопировать и разархивировать в рабочую директорию ПК файл *getdb\_<версия><имя объекта>.rar*. При этом на ПК под управлением ОС Windows 7 (Windows XP) должна быть предустановлена СУБД PostgreSQL версии 9.x, а для ПК под управлением ОС Astra Linux 1.6 необходимо проверить в «Менеджере пакетов Synaptic» установлены ли пакеты qt5-default, qtcreator, libqt5sql5-psql, libqt5sql5-sqlite и установить их в случае отсутствия (см. 3.2).

Дальнейшие действия по работе с программой описаны в Руководстве оператора.

## **7. Установка дополнительных библиотек элементов в ПП «СКАДА А-СОФТ»**

Для упрощения конфигурирования в ПП «СКАДА А-СОФТ» используются библиотеки с шаблонами элементов «Основные элементы», «Электроэлементы», «System-NT», «NT-tmp» и др. Описание элементов указанных библиотек приведено в части 2 руководства оператора. Подключение библиотек производится по мере необходимости.

Для установки библиотеки необходимо:

- 1) с поставляемого диска скопировать определенные библиотеки \*.db или весь каталог «Библиотека виджетов» в директорию /var/spool/scada/DATA;
- 2) запустить ПП «СКАДА А-СОФТ»;
- 3) в модуле БД создать объект БД (ID должен совпадать с названием скопированной библиотеки);
- 4) ввести для нее корректные параметры в поле «Адрес» (последовательный выбор адреса в одноименном поле);
- 5) сохранить сделанные изменения;
- 6) в строке состояние поставить галочку «Включен»;
- 7) нажать на кнопку «Загрузить систему из этой БД».

## 8. Настройка запуска ПП «СКАДА А-СОФТ» в безопасном режиме

Под безопасным режимом функционирования понимается настройка ОС и ПП таким образом, что после авторизации пользователя в ОС Astra Linux происходит автоматическая загрузка ПП «СКАДА А-СОФТ».

### 8.1 Настройка безопасного режима для ОС Astra Linux 1.5

Настройка безопасного режима осуществляется следующим образом:

- 1) Если не создан пользователь, для которого будет настроен безопасный режим, создать его командой в терминале Fly:

```
sudo useradd имя пользователя
```

или добавить пользователя средствами графического интерфейса ОС Astra Linux: «Настройки» → «Политика безопасности» → «Пользователи» → «Создать». Для добавления пользователя необходимо войти в систему в качестве суперпользователя **root**.

- 2) Войти в ОС под именем пользователя и провести отключение блокировки экрана:

- в контекстном меню рабочего стола выбрать строку «Свойства» (рисунок 18).

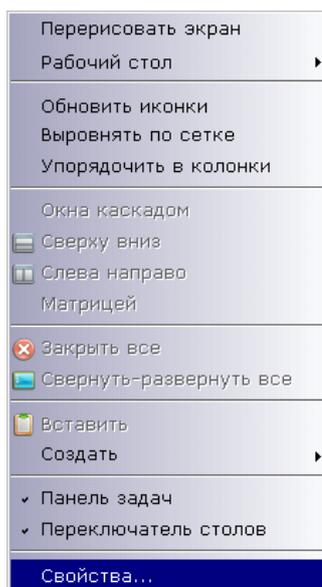


Рисунок 18

- в появившемся окне выбрать раздел «Блокировка» и снять галочку «Блокировать экран через» (рисунок 19). Нажать кнопку «Применить». Закрывать окно.

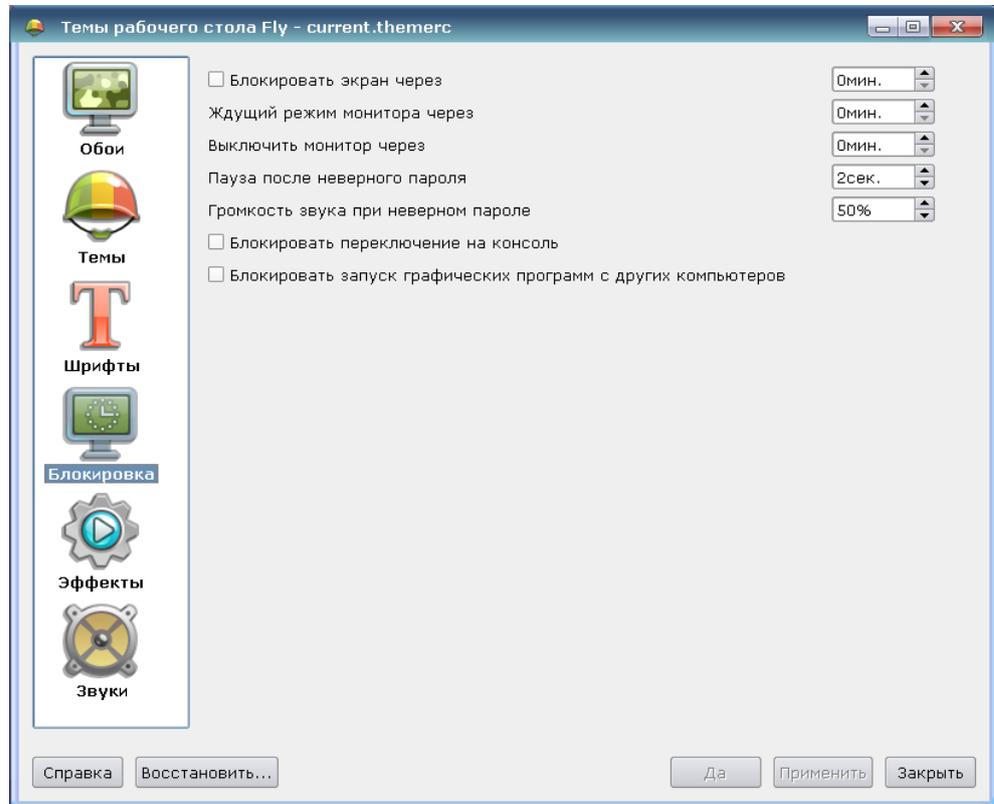


Рисунок 19

- в меню Пуск в разделе «Настройки» выбрать «Настройка монитора». В закладке «Управление питанием» снять галочку «Включить энергосбережение дисплея» (рисунок 20).

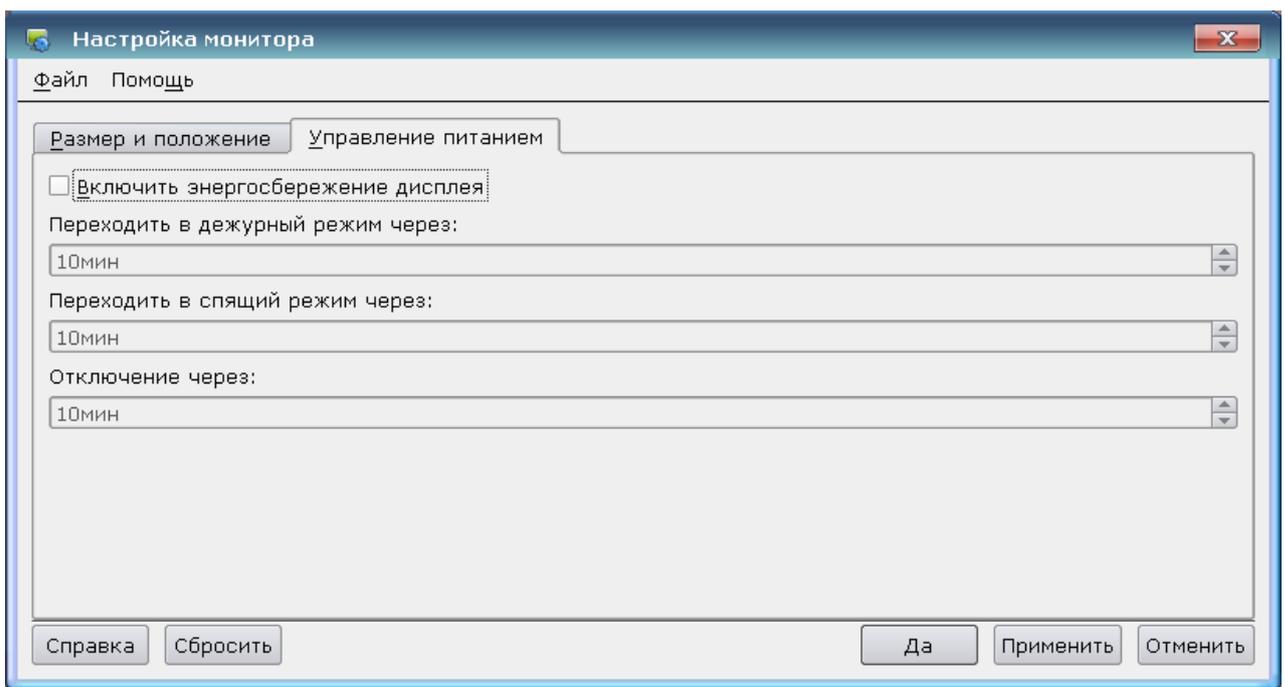


Рисунок 20

- в пункте меню «Файл» необходимо перейти в пункт «Настройки автостарта...», и поставить галочку «Применять настройки при запуске Fly» (рисунок 21). Подтвердить выбор нажатием кнопки «Применить» и закрыть окно.

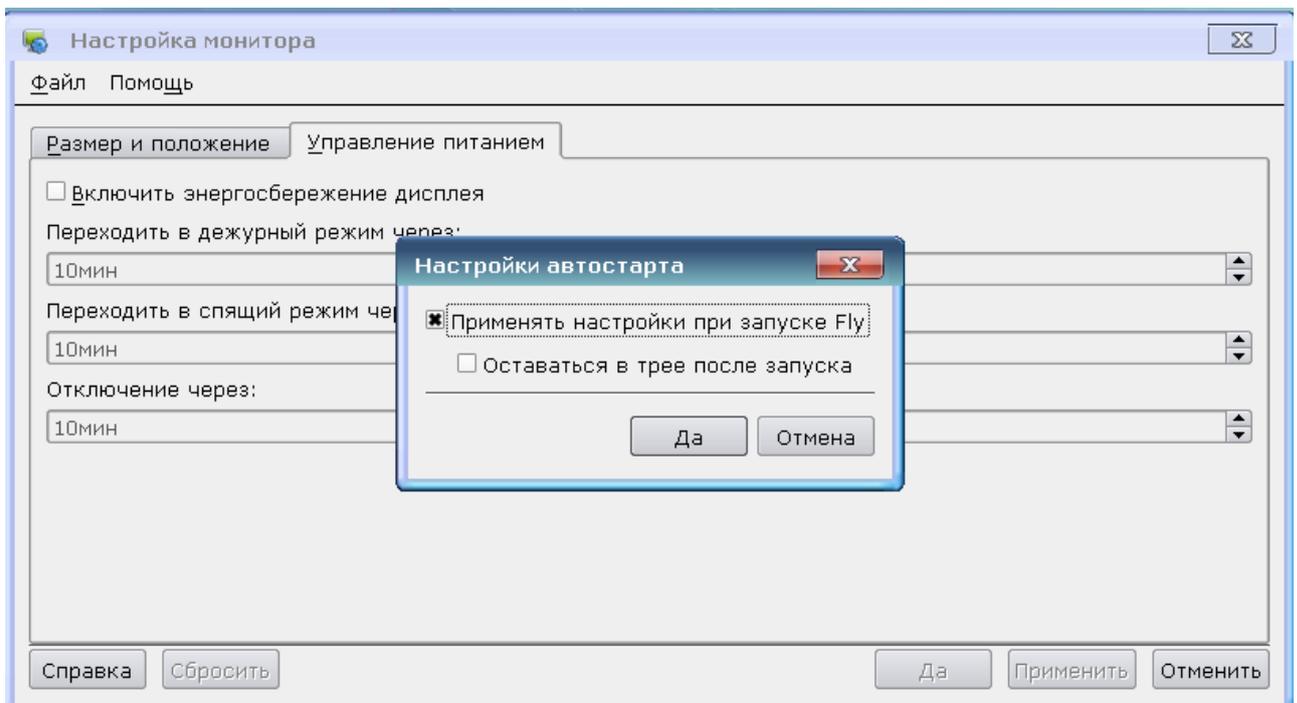


Рисунок 21

3) Войти в систему под суперпользователем.

4) Установить ПП «СКАДА А-СОФТ» согласно 3.3 (если не установлена).

5) С поставляемого изделия программного необходимо распаковать в рабочую директорию файл `scada-fs-helper-gen-1.0.tar` командой:

```
tar -xvf scada-fs-helper-gen-1.0.tar,
```

6) Запустить скрипт из распакованного архива командой:

```
./build.pkg -u user , где user - имя пользователя оператора.
```

7) Установить сгенерированный `deb`-пакет командой:

```
dpkg -i arm-cfg-user.deb, где user - имя пользователя оператора.
```

8) Перезагрузить компьютер и войти под логином пользователя оператора.

В результате на экране появится окно загрузки ПП «СКАДА А-СОФТ». При выходе из нее пользователь вернется в окно авторизации пользователя в ОС Astra Linux.

## 8.2 Настройка безопасного режима для ОС Astra Linux 1.6

Настройка безопасного режима осуществляется следующим образом:

- 1) Войти в систему под пользователем `root` и создать пользователя, для которого будет настроен безопасный режим, средствами графического интерфейса Astra Linux 1.6, например, `user_test`.
- 2) Из стартового меню выбрать строку «Панель управления»→ «Безопасность» → «Политика безопасности».
- 3) В открывшемся окне «Управление политикой безопасности» в дереве объектов выбрать пользователя, для которого настраивается безопасный режим и в окне настройки пользователя на вкладке «Графический киоск Fly» поставить галочку «Режим графического киоска Fly» (рисунок 22).

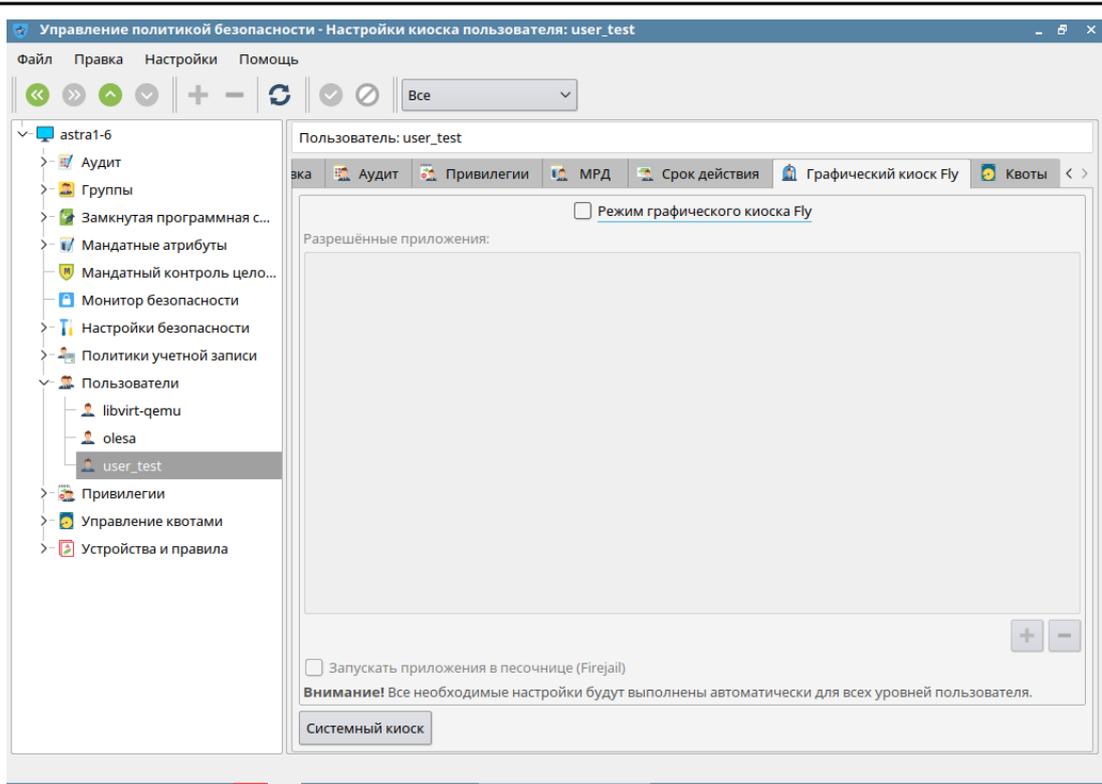


Рисунок 22

- 4) В окне запроса системы на блокировку консоли дать утвердительный ответ.
- 5) После чего, добавить приложения, разрешенные для запуска, нажав пиктограмму + и указать путь к программе (рисунок 23).

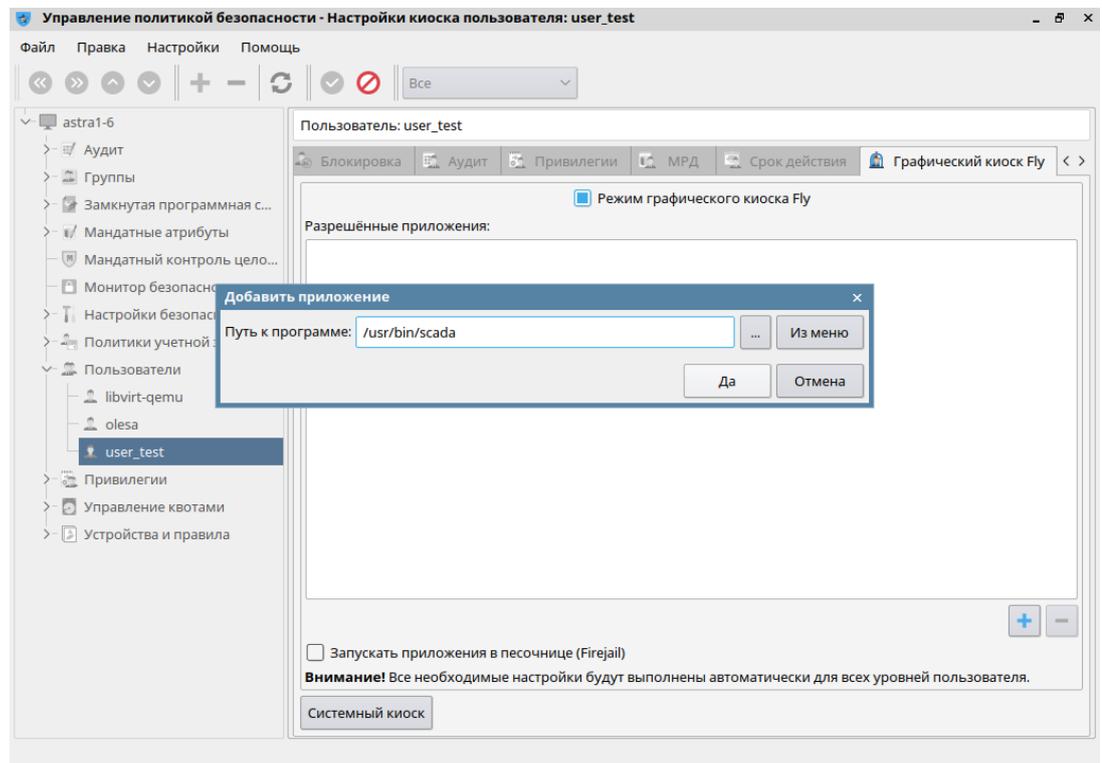


Рисунок 23

- 6) Применить сделанные изменения, нажав на пиктограмму вверху окна Панели управления (рисунок 24).

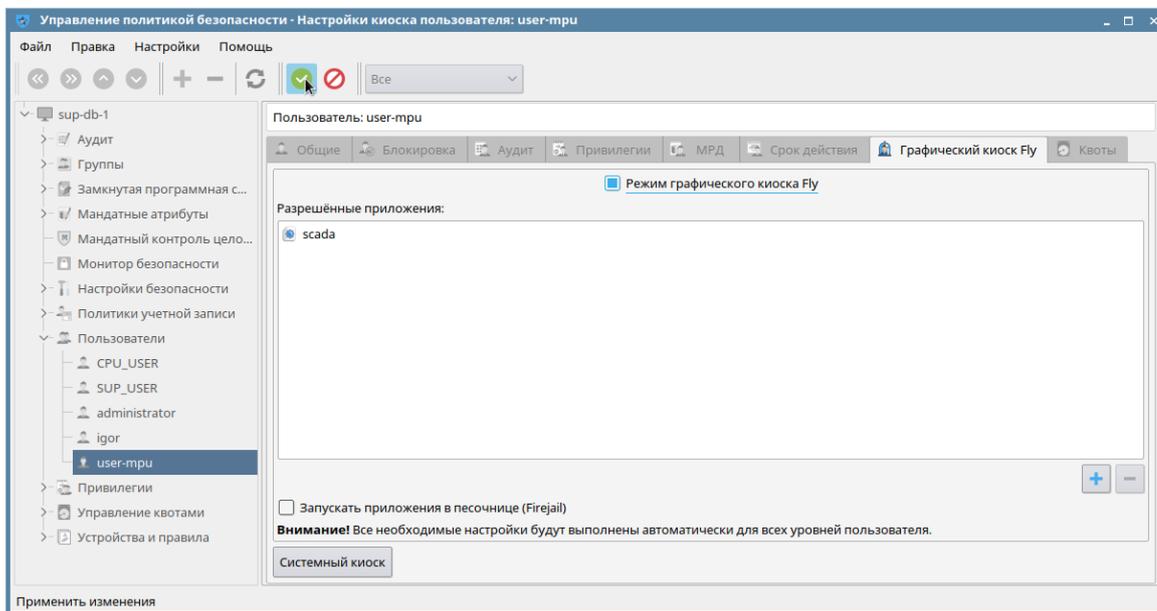


Рисунок 24

- 7) Завершить сессию и войти под пользователем, для которого настроен безопасный режим (в нашем случае, test\_user).

В результате на экране появится окно авторизации пользователя в ОС Astra Linux, после идентификации пользователя на черном экране появится окно загрузки ПП «СКАДА А-СОФТ» (рисунок 25).

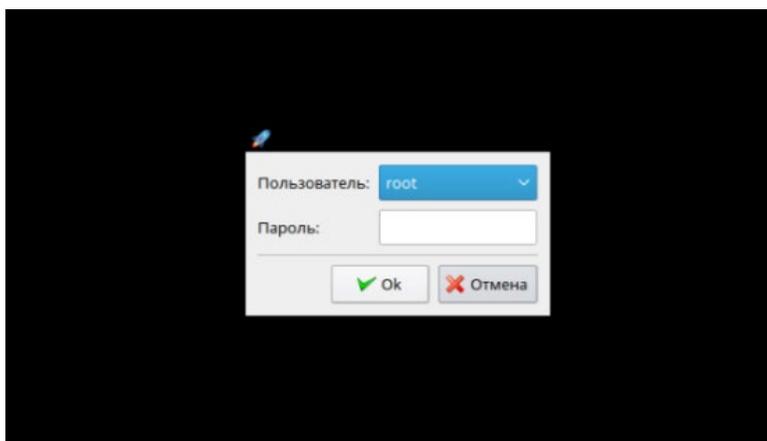


Рисунок 25

## 9. Установка интегрированной справочной системы

Для обеспечения возможности вызова из пользовательского интерфейса справочной информации о работе с ПП «СКАДА А-СОФТ», ее составных частях, способах конфигурирования и настройках, предусмотрена справочная система, позволяющая вызывать справку нажатием на кнопку F1.

Для установки данной справочной системы необходимо:

- 1) с изделия программного скопировать файлы справочных статей, проекта справки и проекта коллекции справки: *1.html* , *2.html* , *3.html* , *qthelp.qhp* , *qthelp.qhcp* в директорию *var/spool/scada/docs*;
- 2) в «Менеджере установки пакетов Sinaptic» убедиться в наличии установленной библиотеки *qt4-dev-tools.lib*. Если библиотека не установлена, то установить в соответствии с п. 3.1 настоящего руководства;
- 3) в директории *var/spool/scada/docs* перейти в окно терминала *Fly* и сгенерировать справку командой:

```
qcollectiongenerator qthelp.qhcp -o qthelp.qhc
```

При следующей загрузке ПП «СКАДА А-СОФТ» после нажатия на кнопку F1 появляется окно справки.

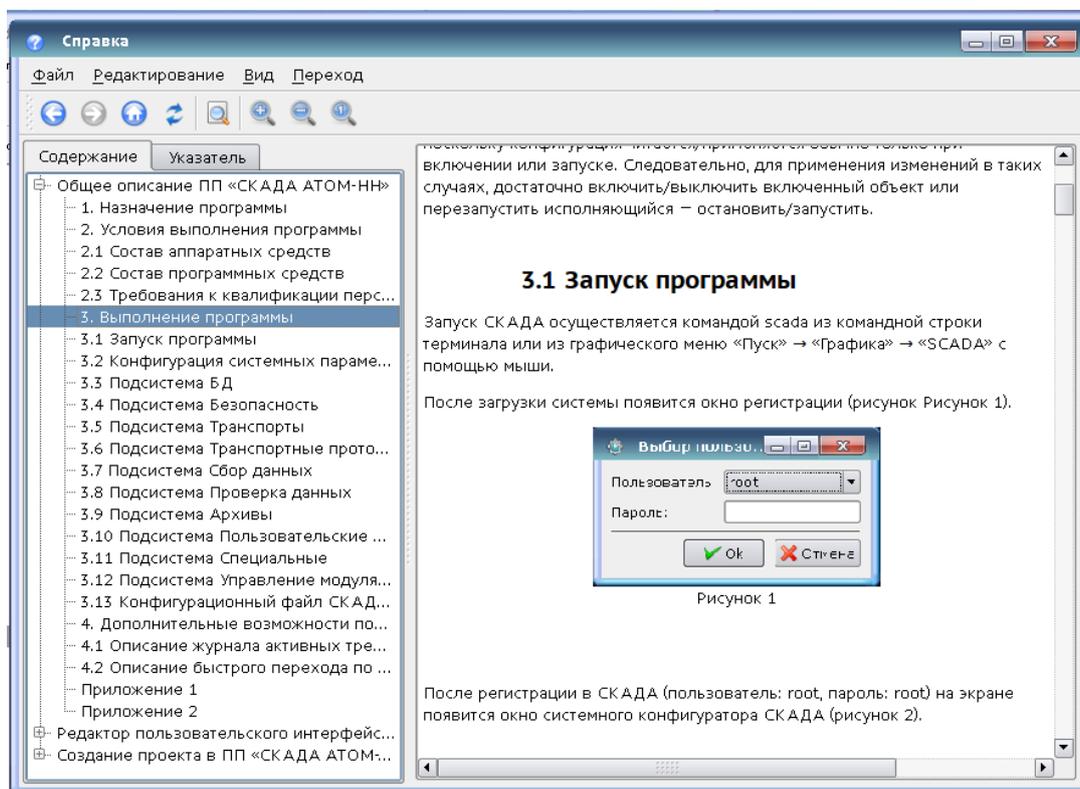


Рисунок 26

## **10. Входные и выходные данные**

### **10.1 Входные данные**

Входными данными для ПП «СКАДА А-СОФТ» являются сигналы от низовых ПТК.

### **10.2 Выходные данные**

Выходными данными ПП «СКАДА А-СОФТ» являются пиктограммы элементов на мнемосхемах (видеокадрах), отображающие состояние технологического оборудования, архивы технологических параметров и команды оператора.

Для обеспечения поддержки источников динамических данных в СКАДА предназначена подсистема «Сбор данных». В функции этой подсистемы входит предоставление полученных входных данных в структурированном виде и обеспечение управления этими данными, например, их модификация. Подключение подсистемы «Сбора данных» осуществляется с помощью системного конфигуратора ПП «СКАДА А-СОФТ». Описание составных частей подсистемы «Сбор данных» представлено в Руководстве оператор.

## 11. Сообщения

ПП «СКАДА А-СОФТ» содержит в себе журнал активных тревог и журнал тревог и событий, в которые помещаются соответствующие сообщения, для отображения тревог и событий в удобном для пользователя виде.

### 11.1 Описание журнала активных тревог и журнала тревог и событий

#### 11.1.1 *Функциональное назначение*

Разработанное ПО обеспечивает выполнение следующих функций:

- отображение всех активных тревог в окне «Журнала активных тревог»;
- отображение все тревог и событий в окне «Журнале тревог и событий» с использованием определенных критериев выбора;
- выполнение квитирования аварийных или предупредительных тревог, имеющих в «Журнале активных тревог»;
- сохранение информации о квитировании тревог в «Журнал тревог и событий».

#### 11.1.2 *Описание журнала активных тревог и журнал тревог и событий*

Журнал тревог и событий содержит таблицу с сообщениями, полученными за заданный пользователем промежуток времени. Журнал активных тревог отображает активные квитированные и не квитированные тревоги до тех пор, пока атрибуты, вызвавшие сообщение, не войдут в норму.

Существует возможность применения фильтра по типам отображаемых сообщений и установка максимального количество отображаемых сообщений. В журнале активных тревог реализована возможность квитирования выбранного пользователем сообщения и перехода к видеокадру, содержащему элемент с вызвавшим тревогу атрибутом.

#### 11.1.3 *Взаимодействие журнала активных тревог и журнала тревог и событий с модулями СКАДА*

Журнал активных тревог и журнал тревог и событий входят в модуль отображения электронно-диспетчерского журнала (ЭДЖ). Взаимодействие журнала активных тревог и журнала тревог и событий с модулями СКАДА представлено на рисунке 27.

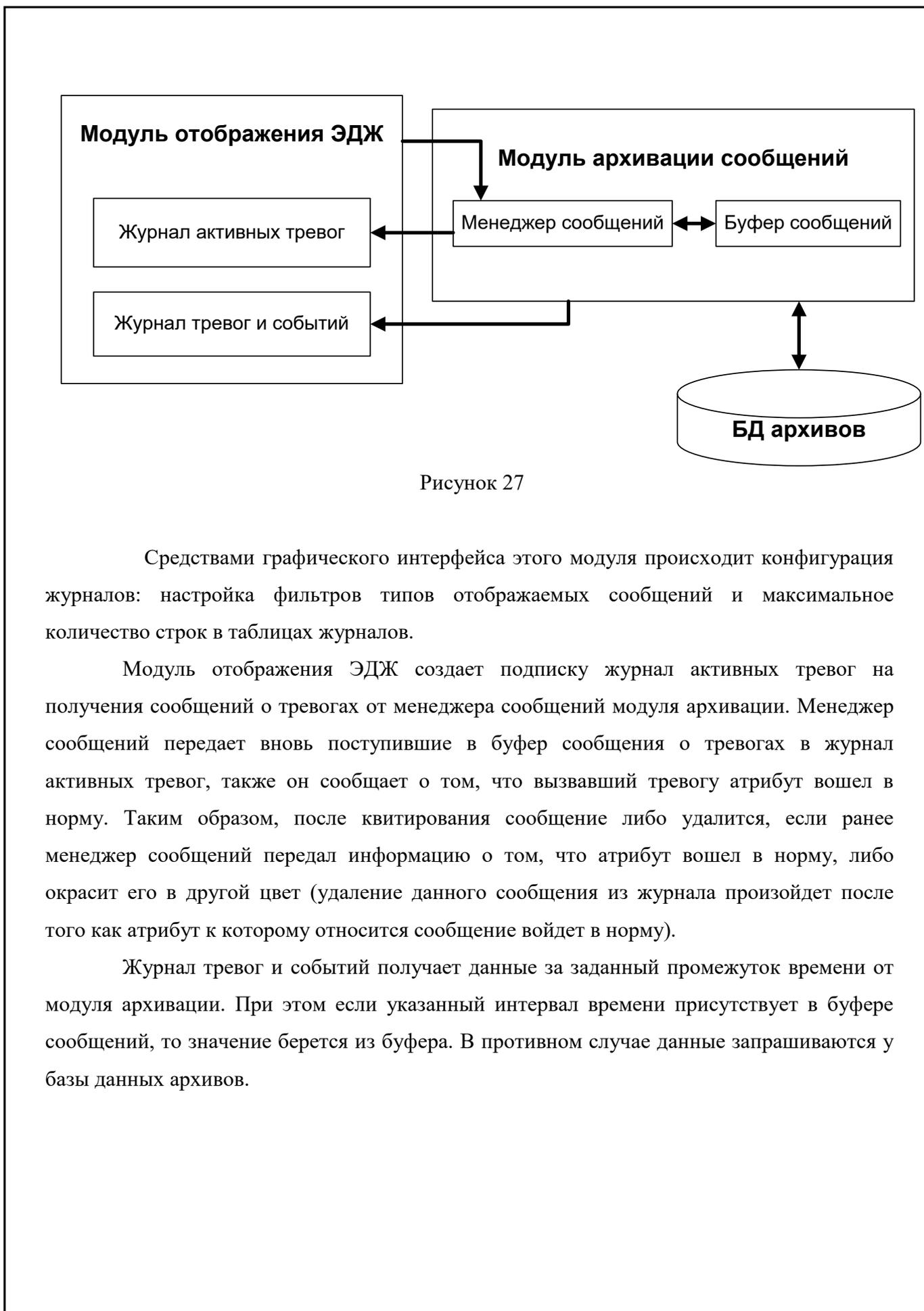


Рисунок 27

Средствами графического интерфейса этого модуля происходит конфигурация журналов: настройка фильтров типов отображаемых сообщений и максимальное количество строк в таблицах журналов.

Модуль отображения ЭДЖ создает подписку журнал активных тревог на получения сообщений о тревогах от менеджера сообщений модуля архивации. Менеджер сообщений передает вновь поступившие в буфер сообщения о тревогах в журнал активных тревог, также он сообщает о том, что вызвавший тревогу атрибут вошел в норму. Таким образом, после квитирования сообщение либо удалится, если ранее менеджер сообщений передал информацию о том, что атрибут вошел в норму, либо окрасит его в другой цвет (удаление данного сообщения из журнала произойдет после того как атрибут к которому относится сообщение войдет в норму).

Журнал тревог и событий получает данные за заданный промежуток времени от модуля архивации. При этом если указанный интервал времени присутствует в буфере сообщений, то значение берется из буфера. В противном случае данные запрашиваются у базы данных архивов.

### 11.1.4 Элементы интерфейса журнала активных тревог и журнал тревог и событий

Для работы с журналом активных тревог и журналом тревог и событий интерфейс ЭДЖ содержит кнопки открытия журналов («1» на рисунке 28) и вкладки, содержащие журнал активных тревог и журнал тревог и событий («2» на рисунке 28).

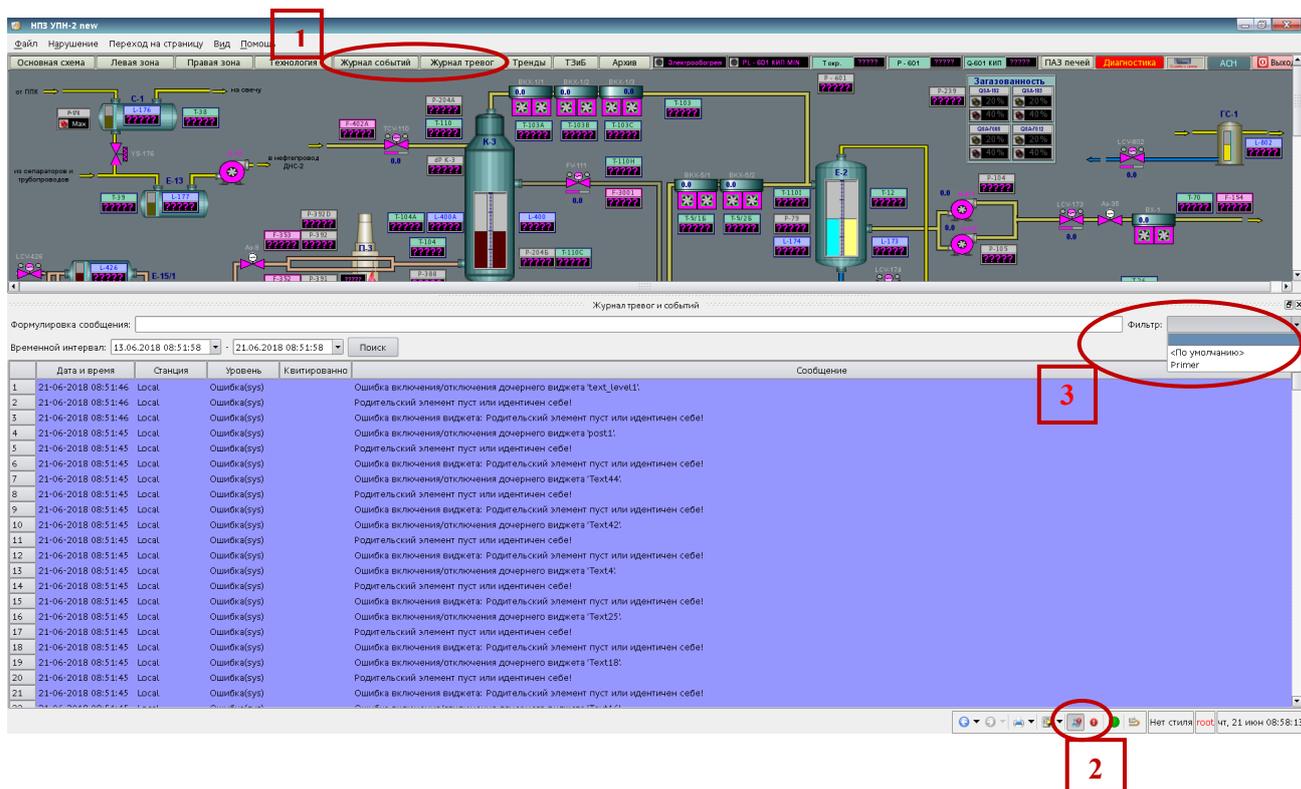


Рисунок 28

В строке журнала тревог и событий отображается следующая информация:

- дата и время;
- станция;
- уровень (ошибка, критическое сообщение, предупреждение, информационное сообщение);
- оборудование;
- отметка квитирования;
- пользователь;
- текст сообщения.

В верхней части окна журнала тревог и событий находятся фильтры, используя которые можно сделать выборку сообщений по дате и по формулировке сообщения.

Для определения порядка отображения столбцов в окне журнала тревог и событий из контекстного меню необходимо выбрать строку «Настройка» и проставить крестик в чек-боксе напротив выводимых столбцов (рисунок 29).

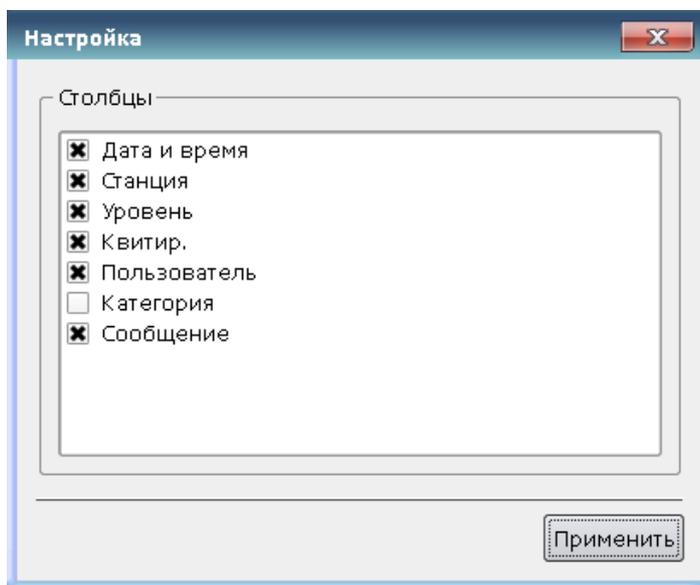


Рисунок 29

В правой части окна журнала тревог и событий находится поле «Фильтр» (см. "3" на рисунке 28), в котором можно настроить типы выводимых сообщений. На рисунке 30 показана настройка фильтра "Primer". Добавление фильтра производится выбором строки «Добавить» контекстного меню и вводом ID и имени нового фильтра. После чего необходимо выбрать уровни (типы) воспроизводимых сообщений (поставить крестик в соответствующем чек-боксе), сохранить изменения и закрыть окно редактирования фильтра. Для просмотра результата работы фильтра необходимо выбрать временной интервал и нажать кнопку «Поиск».

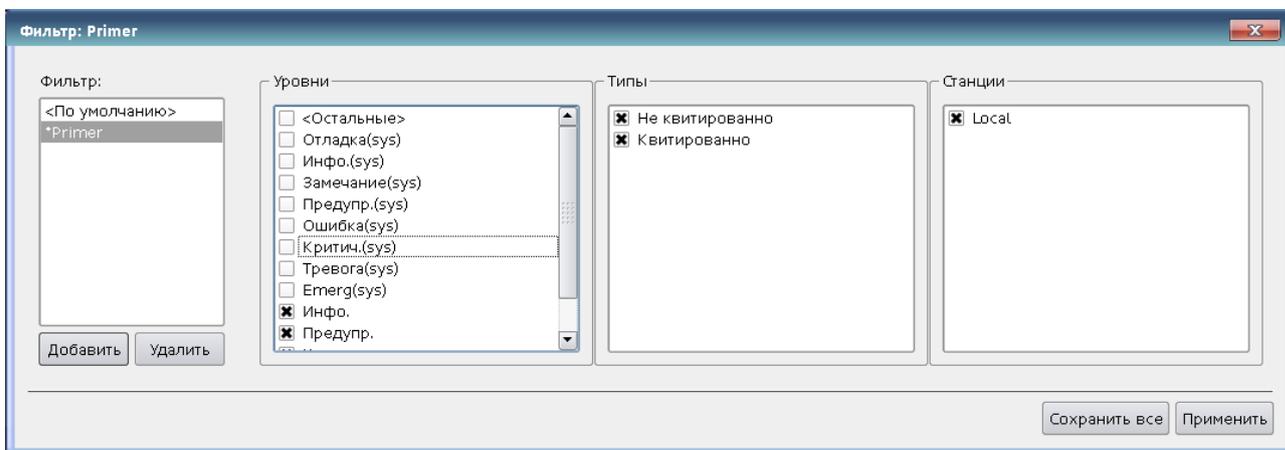


Рисунок 30

Журнал активных тревог служит для просмотра активных тревог и квитирования любой из них. Тревоги от одного источника отображаются вложенной строкой, которую при необходимости можно развернуть.

В строке тревоги отображаются:

- дата и время;
- уровень (ошибка, критическое сообщение);
- оборудование;
- текст сообщения о тревоге.

Тревоги отображаются цветом следующим образом:

тревога активна и не квитирована	красный фон с черной мигающей рамкой;
тревога активна и квитирована	красный фон;
тревога вошла в норму, но не квитирована	белый фон с черной мигающей рамкой.

Квитированная тревога, вошедшая в норму, удаляется из журнала активных тревог.

Квитирование тревоги производится из контекстного меню квитируемой строки.

Для квитирования сообщения необходимо нажать на него правой кнопкой мыши и в появившемся меню выбрать пункт «Квитировать» (рисунок 31). Квитированное сообщение либо изменит цвет, если значение атрибута не вернулось к моменту квитирования в норму, либо удалится из таблицы в противном случае. Если строка имеет вложенные тревоги, то можно в контекстном меню выбрать строку «квитировать все» и будут квитированы все вложенные тревоги.

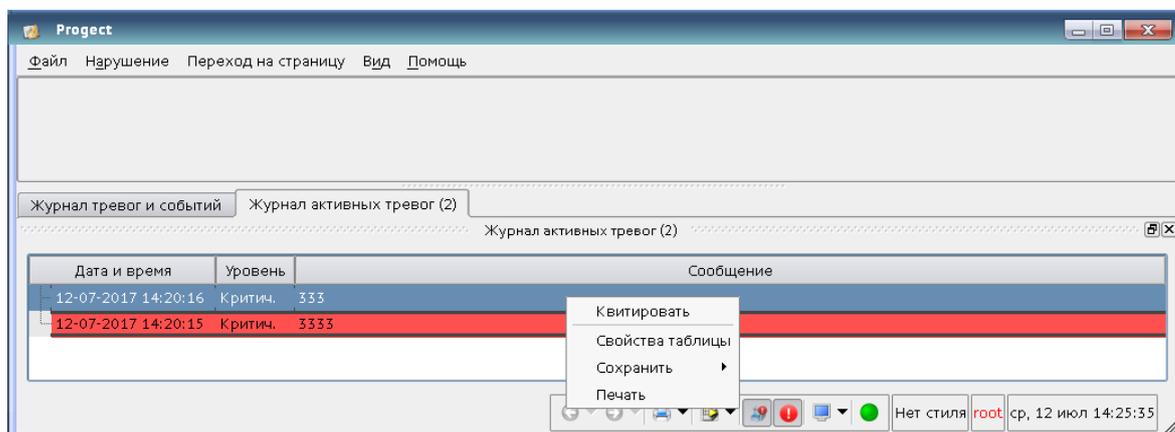


Рисунок 31

Кроме того, обеспечена возможность сохранения данных о тревоге в log-файл, при выборе соответствующего пункта в контекстном меню.

Конфигурирование журнала активных тревог и журнала тревог и событий осуществляется средствами редактора пользовательского интерфейса в окне редактирования свойств визуального элемента «Проект» – «Журнал сообщений».

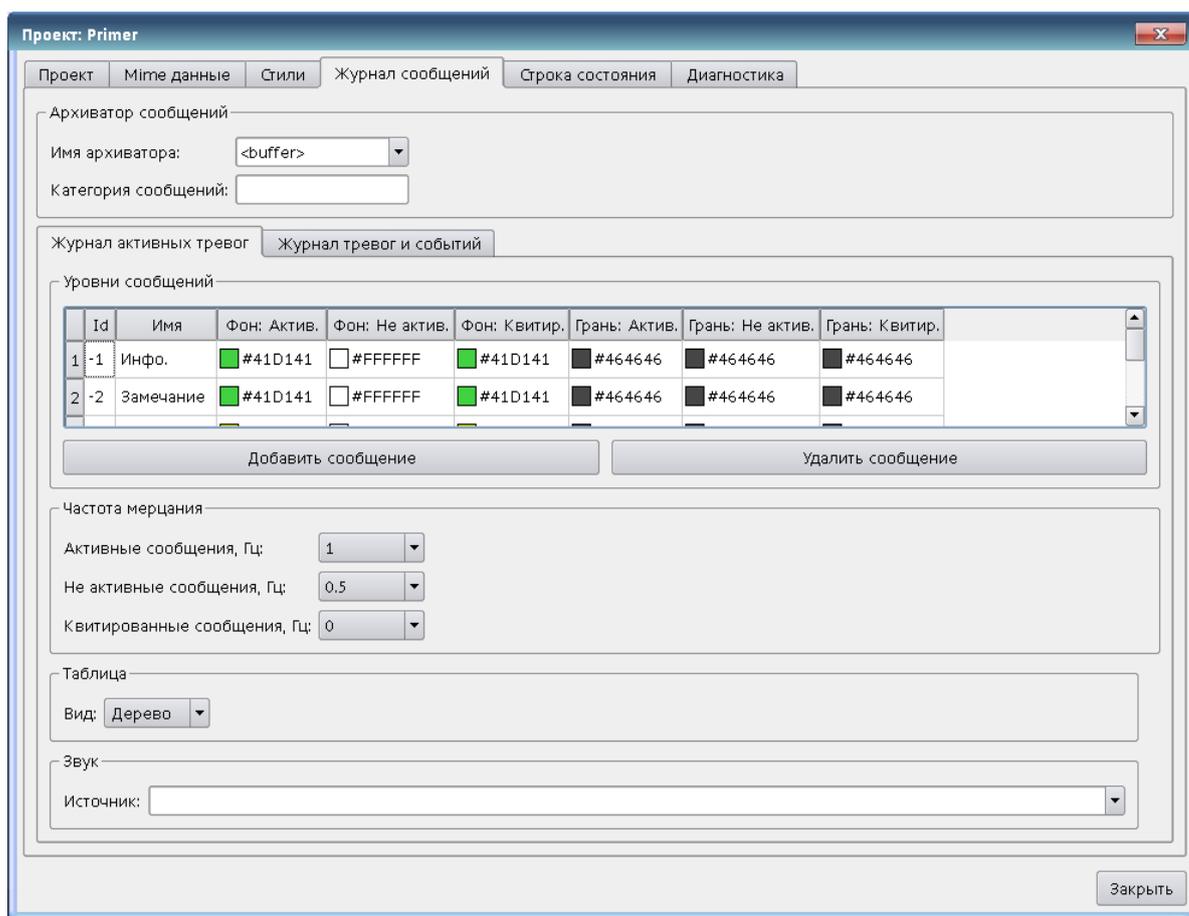


Рисунок 32

Для этого необходимо выбрать вкладку «Журнал сообщений» и настроить атрибуты «Журнала активных тревог» и «Журнала тревог и событий» в зависимости от поставленной задачи, например, добавить сообщения, изменить частоту и цвет мерцания сообщения (рисунок 32).

## ПРИЛОЖЕНИЕ

### Конфигурационный файл СКАДА и параметры командной строки вызова СКАДА

Конфигурационный файл системы СКАДА предназначен для хранения системной и общей конфигурации СКАДА-станции. Только в конфигурационном файле и через параметры командной строки можно указать часть ключевых системных параметров станции.

Называться конфигурационный файл системы СКАДА может произвольно, однако принято название `scada.xml` и производные от него. Конфигурационный файл может быть указан при запуске станции параметром командной строки:

```
--Config=/path/scada.xml
```

где *path* – путь к конфигурационному файлу.

Если конфигурационный файл не указан, то используется стандартный конфигурационный файл: `/etc/scada.xml`.

Структурно конфигурационный файл организован на расширяемом языке разметки текста XML. Следовательно, требуется жёсткое соблюдение правил синтаксиса XML. Пример образца типового конфигурационного файла СКАДА, с узлами конфигурации большинства компонентов СКАДА, приведен ниже.

Один конфигурационный файл может содержать конфигурацию нескольких станций в секциях `<station id="DemoStation"/>`.

Атрибутом указывается идентификатор станции. Использование той или иной секции станции, при вызове, указывается параметром командной строки `--Station=DemoStation`. Секция станции непосредственно содержит параметры станции и секции подсистем. Параметры конфигурации секции записываются в виде `<prm id="StName">Demostation</prm>`. Где в атрибуте `<id>` указывается идентификатор атрибута, а в теле тега указывается значение параметра "Demo station".

### Типовой конфигурационный файл СКАДА:

```
<SCADA>
<!-- This is the SCADA configuration file. -->
<station id="DemoStation">
  <!-- Discribe internal parameter for station. Station this only SCADA
programm. -->
  <prm id="StName">Demo station</prm>
  <prm id="StName_ru">Демо станция</prm>
  <prm id="WorkDB">SQLite.GenDB</prm>
  <prm id="Workdir">~/SCADA</prm>
  <prm id="IcoDir">./icons</prm>
  <prm id="ModDir">/usr/lib/SCADA</prm>
  <prm id="LogTarget">10</prm>
  <prm id="MessLev">0</prm>
  <prm id="Lang2CodeBase">en</prm>
  <prm id="SaveAtExit">0</prm>
  <prm id="SavePeriod">0</prm>
  <node id="sub_BD">
    <prm id="SYSStPref">0</prm>
    <tbl id="DB">
      <fld ID="GenDB" TYPE="SQLite" NAME="Generic DB" NAME_ru="Основная БД"
ADDR="./DEMO/DemoSt.db" CODEPAGE="UTF-8"/>
    </tbl>
  </node>
  <node id="sub_Security">
    <!--
    <tbl id="Security_user">
      <fld NAME="root" DESCR="Super user" DESCR_ru="Супер пользователь"
PASS="SCADA"/>
      <fld NAME="user" DESCR="System user" DESCR_ru="Системный пользователь"
PASS=""/>
    </tbl>
    <tbl id="Security_grp">
      <fld NAME="root" DESCR="Super users groups" DESCR_ru="Группа
суперпользователей" USERS="root;user"/>
    </tbl>-->
  </node>
  <node id="sub_ModSched">
    <prm id="ModAllow">*</prm>
    <prm id="ModDeny"></prm>
```

```
<prm id="ChkPer">0</prm>
</node>
<node id="sub_Transport">
<!--
<tbl id="Transport_in">
  <fld ID="WEB_1" MODULE="Sockets" NAME="Generic WEB interface"
NAME_ru="Основной WEB интерфейс" DESCRIPT="Generic transport for WEB
interface." DESCRIPT_ru="Основной транспорт для WEB интерфейса."
ADDR="TCP::10002:0" PROT="HTTP" START="1"/>
  <fld ID="WEB_2" MODULE="Sockets" NAME="Reserve WEB interface"
NAME_ru="Резервный WEB интерфейс" DESCRIPT="Reserve transport for WEB
interface." DESCRIPT_ru="Резервный транспорт для WEB интерфейса."
ADDR="TCP::10004:0" PROT="HTTP" START="1"/>
</tbl>
  <tbl id="Transport_out">
  <fld ID="testModBus" MODULE="Sockets" NAME="Test ModBus" NAME_ru="Тест
ModBus" DESCRIPT="Data exchange by protocol ModBus test." DESCRIPT_ru="Тест
обмена по протоколу ModBus." ADDR="TCP:localhost:10502" START="1"/>
</tbl>-->
</node>
<node id="sub_DAO">
<!--
<tbl id="tmplib">
  <fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" NAME_DESCR=""
DESCR_ru="" DESCR_DB="tmplib_test2"/>
</tbl>
  <tbl id="tmplib_test2">
  <fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
DB="test2" PROGRAM="JavaLikeCalc.JavaScript&#010;cnt=5*i"/>
</tbl>
  <tbl id="tmplib_test2_io">
  <fld TMPL_ID="test2" ID="i" NAME="I" NAME_ru="I" TYPE="4" FLAGS="160"
VALUE="" POS="0"/>
  <fld TMPL_ID="test2" ID="cnt" NAME="Cnt" NAME_ru="Cnt" TYPE="4"
FLAGS="32" VALUE="" POS="0"/>
</tbl>-->
<node id="mod_LogicLev">
<!--
<tbl id="DAQ">
  <fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="test2prm" PERIOD="1000" PRIOR="0"/>
```

```
</tbl>
<tbl id="test2prm">
  <fld SHIFR="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
EN="1" MODE="2" PRM="test2.test2"/>
</tbl>-->
</node>
<node id="mod_System">
  <!--
  <tbl id="DAQ">
    <fld ID="DataOS" NAME="Data OS" NAME_ru="Данные ОС" DESCR="Data of
services and subsystems OS." DESCR_ru="Данные сервисов и подсистем ОС."
ENABLE="1" START="1" AUTO_FILL="0" PRM_BD="DataOSprm" PERIOD="1000"
PRIOR="0"/>
  </tbl>
  <tbl id="DataOSprm">
    <fld SHIFR="CPU" NAME="CPU load" NAME_ru="Нагрузка CPU" DESCR=""
DESCR_ru="" EN="1" TYPE="CPU" SUBT="gen"/>
    <fld SHIFR="MEM" NAME="Memory" NAME_ru="Память" DESCR="" DESCR_ru=""
EN="1" TYPE="MEM"/>
  </tbl> -->
</node>
<node id="mod_DiamondBoards">
  <!--
  <tbl id="DAQ">
    <fld ID="Athena" NAME="Athena board" NAME_ru="Плата Athena" DESCR=""
DESCR_ru="" ENABLE="1" START="0" BOARD="25" PRM_BD_A="AthenaAnPrm"
PRM_BD_D="AthenaDigPrm" ADDR="640" INT="5" DIO_CFG="0" ADMODE="0"
ADRANGE="0" ADPOLAR="0" ADGAIN="0" ADCONVRATE="1000"/>
  </tbl>
  <tbl id="AthenaAnPrm">
    <fld SHIFR="ai0" NAME="AI 0" NAME_ru="AI 0" DESCR="" DESCR_ru=""
EN="0" TYPE="0" CNL="0" GAIN="0"/>
  </tbl>
  <tbl id="AthenaDigPrm">
    <fld SHIFR="di0" NAME="DI 0" NAME_ru="DI 0" DESCR="" DESCR_ru=""
EN="0" TYPE="0" PORT="0" CNL="0"/>
  </tbl> -->
</node>
<node id="mod_BlockCalc">
  <!--
  <tbl id="DAQ">
```

```
<fld ID="Model" NAME="Model" NAME_ru="Модель" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="Model_prm" BLOCK_SH="Model_blcks"
PERIOD="1000" PRIOR="0" PER_DB="0" ITER="1"/>
</tbl>
<tbl id="Model_blcks">
<fld ID="Klap" NAME="Klapan" NAME_ru="Клапан" DESCR="" DESCR_ru=""
FUNC="DAQ.JavaLikeCalc.lib_techApp.klap" EN="1" PROC="1"/>
</tbl>
<tbl id="Model_blcks_io">
<fld BLK_ID="Klap" ID="l_kl1" TLNK="0" LNK="" VAL="50"/>
<fld BLK_ID="Klap" ID="l_kl2" TLNK="0" LNK="" VAL="20"/>
</tbl>
<tbl id="Model_prm">
<fld SHIFR="l_kl" NAME="Klap lev" NAME_ru="Полож. клапана" DESCR=""
DESCR_ru="" EN="1" BLK="Klap" IO="l_kl1"/>
</tbl> -->
</node>
<node id="mod_JavaLikeCalc">
<!--
<tbl id="DAQ">
<fld ID="CalcTest" NAME="Calc Test" NAME_ru="Тест вычисл." DESCR=""
DESCR_ru="" ENABLE="1" START="1" PRM_BD="Cal FUNC="TemplFunc.d_alarm"
PERIOD="1000" PRIOR="0" PER_DB="0" ITER="1"/>
</tbl>
<tbl id="CalcTest_val">
<fld ID="in" VAL="0"/>
<fld ID="alrm" VAL=""/>
<fld ID="alrm_md" VAL="1"/>
<fld ID="alrm_mess" VAL="Error present."/>
</tbl>
<tbl id="CalcTest_prm">
<fld SHIFR="alrm" NAME="Alarm" NAME_ru="Авария" DESCR="" DESCR_ru=""
EN="1" FLD="alrm"/>
</tbl>
<tbl id="lib">
<fld ID="TemplFunc" NAME="" NAME_ru="" DESCR="" ESCR_ru=""
DB="lib_TemplFunc"/>
</tbl>
<tbl id="lib_TemplFunc">
```

```
<fld ID="d_alarm" NAME="Digit alarm" NAME_ru="Авария по дискр."
DESCR="" FORMULA="alm=(in==alm_md)?&quot;1:&quot;+alm_mess:&quot;0&quot;;
"/>
</tbl>
<tbl id="lib_TemplFunc_io">
<fld F_ID="d_alarm" ID="in" NAME="Input" NAME_ru="Вход" TYPE="3"
MODE="0" DEF="" HIDE="0" POS="0"/>
<fld F_ID="d_alarm" ID="alm" NAME="Alarm" NAME_ru="Авария" TYPE="0"
MODE="1" DEF="" HIDE="0" POS="1"/>
<fld F_ID="d_alarm" ID="alm_md" NAME="Alarm mode" NAME_ru="Режим
аварии" TYPE="3" MODE="0" DEF="" HIDE="0" POS="2"/>
<fld F_ID="d_alarm" ID="alm_mess" NAME="Alarm message"
NAME_ru="Сообщ. аварии" TYPE="0" MODE="0" DEF="" HIDE="0" POS="3"/>
</tbl>---->
</node>
<node id="mod_Siemens">
<!--
<tbl id="DAQ">
<fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="test2prm" PERIOD="1000" PRIOR="0" CIF_DEV="0"
ADDR="5" ASINC_WR="0"/>
</tbl>
<tbl id="test2prm">
<fld SHIFR="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
EN="1" TMPL="S7.ai_man"/>
</tbl>--->
</node>
<node id="mod_SNMP">
<!--
<tbl id="DAQ">
<fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="test2prm" PERIOD="1000" PRIOR="0"
ADDR="localhost" COMM="public" PATTR_LIM="20"/>
</tbl>
<tbl id="test2prm">
<fld SHIFR="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
EN="1" OID_LS="system"/>
</tbl>--->
</node>
<node id="mod_ModBus">
<!--
```

```
<tbl id="DAQ">
  <fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="test2prm" PERIOD="1000" PRIOR="0"
TRANSP="Sockets" ADDR="exlar.diya.org" NODE="1"/>
</tbl>
<tbl id="test2prm">
  <fld SHIFR="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
EN="1" ATTR_LS="321:0:tst:Test"/>
</tbl>-->
</node>
<node id="mod_Transporter">
  <!--
  <tbl id="DAQ">
    <fld ID="test2" NAME="Test 2" NAME_ru="Тест 2" DESCR="" DESCR_ru=""
ENABLE="1" START="1" PRM_BD="test2prm" PERIOD="1000" PRIOR="0" SYNCPER="60"
STATIONS="loop" CNTRPRM="System.AutoDA"/>
  </tbl>-->
</node>
</node>
<node id="sub_Archive">
  <prm id="MessBufSize">1000</prm>
  <prm id="MessPeriod">5</prm>
  <prm id="ValPeriod">1000</prm>
  <prm id="ValPriority">10</prm>
  <!--
  <tbl id="Archive_mess_proc">
    <fld ID="StatErrors" MODUL="FSArch" NAME="Errors" NAME_ru="Ошибки"
DESCR="Local errors\' archive" DESCR_ru="Архив локальных ошибок" START="1"
CATEG="/DemoStation*" LEVEL="4" ADDR="ARCHIVES/MESS/stError/"
FSArchMSize="300" FSArchNFiles="10" FSArchTmSize="30" FSArchXML="1"
FSArchPackTm="10" FSArchTm="60"/>
    <fld ID="NetRequsts" MODUL="FSArch" NAME="Net requests"
NAME_ru="Сетевые запросы" DESCR="Requests to server through transport
Sockets." DESCR_ru="Запросы к серверу через транспорт Sockets." START="1"
CATEG="/DemoStation/Transport/Sockets*" LEVEL="1" ADDR="ARCHIVES/MESS/Net/"
FSArchMSize="300" FSArchNFiles="10" FSArchTmSize="30" FSArchXML="1"
FSArchPackTm="10" FSArchTm="60"/>
  </tbl>
  <tbl id="Archive_val_proc">
    <fld ID="1h" MODUL="FSArch" NAME="1hour" NAME_ru="1час"
DESCR="Averaging for hour" DESCR_ru="Усреднение за час" START="1"
```

```
ADDR="ARCHIVES/VAL/1h/" V_PER="360" A_PER="60" FSArchTmSize="8640"
FSArchNFiles="10" FSArchRound="0.1" FSArchPackTm="10" FSArchTm="60"/>
  </tbl>
  <tbl id="Archive_val">
    <fld ID="test1" NAME="Test 1" NAME_ru="Tect 1" DESCR="Test 1"
DESCR_ru="Tect 1" START="1" VTYPE="1" BPER="1" BSIZE="200" BHGRD="1"
BHRES="0" SrcMode="0" Source="" ArchS=""/>
  </tbl>-->
</node>
<node id="sub_Protocol">
</node>
<node id="sub_UI">
<node id="mod_QTStarter">
<prm id="StartMod">QTCfg</prm>
</node>
<node id="mod_WebCfg">
<prm id="SessTimeLife">20</prm>
</node>
<node id="mod_VCAEngine">
<!--
<tbl id="LIB">
  <fld ID="test2" NAME="Test 2" NAME_ru="Tect 2" DESCR="" DESCR_ru=""
DB_TBL="wlib_test2" ICO="" USER="root" GRP="UI" PERMIT="436"/>
  </tbl>
  <tbl id="wlib_test2">
    <fld ID="test2" ICO="" PARENT="/wlb_originals/wdg_Box" PROC=""
PROC_ru="" PROC_PER="-1" USER="root" GRP="UI" PERMIT="436"/>
  </tbl> <tbl id="wlib_test2_io">
    <fld IDW="test2" ID="name" IO_VAL="Test 2" IO_VAL_ru="Tect 2"
SELF_FLG="" CFG_TMPL="" CFG_TMPL_ru="" CFG_VAL=""/>
    <fld IDW="test2" ID="dscr" IO_VAL="Test module 2" IO_VAL_ru="Tect
модуля 2" SELF_FLG="" CFG_TMPL="" CFG_TMPL_ru="" CFG_VAL=""/>
  </tbl>
  <tbl id="PRJ">
    <fld ID="test2" NAME="Test 2" NAME_ru="Tect 2" DESCR="" DESCR_ru=""
DB_TBL="prj_test2" ICO="" USER="root" GRP="UI" PER </tbl> <tbl
id="prj_test2">
    <fld OWNER="/test2" ID="pg1" ICO="" PARENT="/wlb_originals/wdg_Box"
PROC="" PROC_ru="" PROC_PER="-1" USER="root" GRP="UI" PERMIT="436"
FLGS="1"/>
```

```
<fld OWNER="/test2/pg1" ID="pg2" ICO=""
PARENT="/wlb_originals/wdg_Box" PROC="" PROC_ru="" PROC_PER="-1"
USER="root" GRP="UI" PERMIT="436" FLGS="0"/>
</tbl>
<tbl id="prj_test2_incl">
<fld IDW="/prj_test2/pg_pg1" ID="wdg1"
PARENT="/wlb_originals/wdg_Box"/>
</tbl>-->
</node>
</node>
<node id="sub_Special">
<node id="mod_SystemTests">
<prm id="PARAM" on="0" per="5" name="LogicLev.experiment.F3"/>
<prm id="XML" on="0" per="10" file="/etc/oscada.xml"/> <prm id="MESS"
on="0" per="10" categ="" arhtor="DBArch.test3"/>
<prm id="SOAttDet" on="0" per="20"
name="../../lib/SCADA/daq_LogicLev.so" full="1"/>
<prm id="Val" on="0" per="1" name="LogicLev.experiment.F3.var"
arch_len="5" arch_per="1000000"/>
<prm id="Val" on="0" per="1" name="System.AutoDA.CPULoad.load"
arch_len="10" arch_per="1000000"/>
<prm id="BD" on="0" per="10" type="MySQL"
bd="server.diya.org;roman;123456;oscadaTest" table="test" size="1000"/>
<prm id="BD" on="0" per="10" type="DBF" bd="./DATA/DBF"
table="test.dbf" size="1000"/>
<prm id="BD" on="0" per="10" type="SQLite" bd="./DATA/test.db"
table="test" size="1000"/>
<prm id="BD" on="0" per="10" type="FireBird"
bd="server.diya.org:/var/tmp/test.fdb;roman;123456" table="test"
size="1000"/>
<prm id="TrOut" on="0" per="1" addr="TCP:127.0.0.1:10001"
type="Sockets" req="time"/>
<prm id="TrOut" on="0" per="1" addr="UDP:127.0.0.1:10001"
type="Sockets" req="time"/>
<prm id="TrOut" on="0" per="1" addr="UNIX:./oscada" type="Sockets"
req="time"/>
<prm id="TrOut" on="0" per="1" addr="UDP:127.0.0.1:daytime"
type="Sockets" req="time"/>
<prm id="Func" on="0" per="10"/> <prm id="SysContrLang" on="0"
per="10" path="/Archive/FSArch/mess_StatErrors/%2fprm%2fst"/>
```

```
<prm id="ValBuf" on="0" per="5"/> <prm id="Archive" on="0" per="30"  
arch="test1" period="1000000"/>  
<prm id="Base64Code" on="0" per="10"/>  
</node>  
</node>  
</station>  
</SCADA>
```

### Перечень принятых сокращений

БД	база данных
ОЗУ	оперативное запоминающее устройство
ОС	операционная система
ПО	программное обеспечение
ПП	программная платформа
ПТК	программно-технический комплекс
ЭДЖ	электронно-диспетчерский журнал
SCADA	диспетчерское управление и сбор данных (Supervisory Control And Data Acquisition)

