

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ

«СКАДИ»

(ООО «СКАДИ»)

УТВЕРЖДАЮ

**СРЕДСТВА ВЕБ-ДОСТУПА К ДАННЫМ ДЛЯ МОНИТОРИНГА
И МОДЕЛИРОВАНИЯ АЭС И ПРОМЫШЛЕННЫХ ПРИМЕНЕНИЙ**

Руководство системного программиста

34185873.425510.004.32.М

(На 29 листах)

2023

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Руководство системного программиста	Версия 1 10.12.2023
-------------	--	------------------------

РАЗРАБОТАЛ

СОДЕРЖАНИЕ

1 ОБЩИЕ СВЕДЕНИЯ О ПО.....	4
1.1 НАИМЕНОВАНИЕ.....	4
1.2 НАЗНАЧЕНИЕ И ФУНКЦИИ.....	4
1.3 НЕОБХОДИМЫЕ ТЕХНИЧЕСКИЕ И ПРОГРАММНЫЕ СРЕДСТВА	4
2 СТРУКТУРА ПО	6
2.1 Состав ПО	6
2.2 СВЯЗИ СОСТАВНЫХ ЧАСТЕЙ ПО.....	6
3 НАСТРОЙКА ПРОГРАММЫ.....	8
3.1 ИНСТАЛЛЯЦИЯ.....	8
3.1.1 Инсталляция для ОС Astra Linux, ОС Linux	8
3.1.2 Инсталляция для ОС Windows	8
3.2 НАСТРОЙКА ПРОЕКТА	8
3.2.1 Базовые файлы проекта.....	8
3.2.2 Настройка вне конфигурации СКАДИ.....	9
3.2.3 Настройка групповых ресурсов.....	10
3.2.4 Линейное масштабирование проекта.....	11
3.2.5 Настройка режимов записи-воспроизведения	12
3.2.6 Настройка режима управления.....	13
3.3 НАСТРОЙКА ПЕРЕМЕННЫХ ОКРУЖЕНИЯ.....	13
3.4 УСТАНОВКА ЛИЦЕНЗИИ ПРОЕКТА	13
3.5 ГЕНЕРАЦИЯ ГРАФИЧЕСКИХ ДАННЫХ	14
4 ПРОВЕРКА ПРОГРАММЫ	15
4.1 ДОСТУП ИЗ БРАУЗЕРА.....	15
5 ДОПОЛНИТЕЛЬНЫЕ ВОЗМОЖНОСТИ	16
5.1 ОБЛАЧНЫЕ ТЕХНОЛОГИИ.....	16
6 СООБЩЕНИЯ СИСТЕМНОМУ ОПЕРАТОРУ	17
ПРИЛОЖЕНИЯ	18
Приложение 1 – ОПИСАНИЕ КОНФИГУРАЦИИ DSCONFIG.....	18
Приложение 2 - ДАННЫЕ ТАБЛИЦЫ TASK КОНФИГУРАЦИИ	19
Приложение 3 – ОПЦИИ КОМАНДНОЙ СТРОКИ	22
Приложение 4 – ЗНАЧЕНИЯ УРОВНЯ ТРАССИРОВКИ	23
Приложение 5 – СКРИПТ ГЕНЕРАЦИИ ВИДЕОКАДРОВ	24
Приложение 6 – ФАЙЛ ОПИСАНИЯ ИСХОДНЫХ ДАННЫХ ДЛЯ ГЕНЕРАЦИИ ВИДЕОКАДРОВ	24
Приложение 7 – ФАЙЛ НАСТРОЕК СИСТЕМЫ ГЕНЕРАЦИИ	26
Приложение 8 – СКРИПТ ПЕРЕНОСА СГЕНЕРИРОВАННЫХ ГРАФИЧЕСКИХ ДАННЫХ	27
ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ.....	28
ЛИСТ ИЗМЕНЕНИЙ.....	29

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Руководство системного программиста	Версия 1 10.12.2023
-------------	--	------------------------

1 ОБЩИЕ СВЕДЕНИЯ О ПО

1.1 НАИМЕНОВАНИЕ

Наименование изделия — Средства веб-доступа к данным для мониторинга и моделирования АЭС и промышленных предприятий «СКАДИВЕБ 1.0».

Сокращенное наименование: «СКАДИВЕБ».

Программа «СКАДИВЕБ» («SCADIWEB») зарегистрирована в реестре программ для ЭВМ №2023685251 от 24.11.2023.

Наименование документа 34185873.425510.004.32.М, обозначение по ГОСТ 34.201-89.

«СКАДИВЕБ 1.0» интегрировано с ПО «СКАДИ» - «Программная платформа АСУ ТП и диагностики для СВБУ АЭС и промышленных применений». Описание ПО «СКАДИ», зарегистрированное в реестре программ для ЭВМ №2021681559 от 23.12.2021, представлено в документе 34185873.425510.001.ОП.80.М от 04.07.2022.

1.2 НАЗНАЧЕНИЕ И ФУНКЦИИ

Программное обеспечение СКАДИВЕБ предназначено для построения систем мониторинга и аналитики в области АСУТП и диагностики, реализующих следующие функции:

- сбора и выдачи данных, в том числе в режиме мягкого реального времени;
- онлайн-обработка последовательностей сигналов, в том числе с применением сложных диагностических алгоритмов;
- командного управления моделью управляющей системы с выдачей сигнала на тренажер;
- регистрации и представления сигнализациями;
- обеспечения доступа к архивным данным;
- запроса и управления протоколами;
- диагностики программных и аппаратных средств системы;
- конфигурации параметров системы;
- визуализации и отображения видеокадров;
- импортной и экспортной конвертации видеокадров.

1.3 НЕОБХОДИМЫЕ ТЕХНИЧЕСКИЕ И ПРОГРАММНЫЕ СРЕДСТВА

Для выполнения ПО СКАДИВЕБ необходимо иметь установленное программное обеспечение СКАДИ.

Для функционирования «СКАДИВЕБ 1.0» требуется наличие операционной системы семейств Linux или Windows. Операционное системное программное обеспечение устанавливается на все узлы, где должно работать ПО СКАДИВЕБ.

34185873.425510.004.32.М	Стр. 4 из 29
--------------------------	--------------

ПО «СКАДИВЕБ 1.0» предназначено для работы с является операционным программным обеспечением, требующим для своей работы 64-разрядных процессоров AMD/Intel с архитектурой x86_64.

Для использования средств защиты информации операционных систем рекомендуется использование ОС Astra Linux, начиная с версии 1.6.

Операционные системы к которым, не предъявляются требования по безопасности:

- Debian Linux 10 и выше
- Ubuntu Linux 18 и выше.

В зависимости от конфигурации, серверное ПО СКАДИВЕБ может размещаться в контейнерах и управляться облачными сервисами.

Клиентское ПО СКАДИВЕБ должно загружаться в браузеры и исполняться в их программной среде. От браузеров требуется стандартная поддержка JavaScript и формата SVG.

2 СТРУКТУРА ПО

2.1 СОСТАВ ПО

Программное обеспечение СКАДИВЕБ содержит в себе следующие функциональные составляющие:

- подсистема веб-серверного доступа к данным;
- подсистема клиентских модулей взаимодействия с серверами через браузер;
- подсистема скриптов генерации данных.

Подсистема веб-серверного доступа к данным состоит из набора модулей приложений, расположенных в каталоге инсталляции СКАДИВЕБ. Имена модулей приложений СКАДИВЕБ имеют шаблон Ws* для Linux-систем и Ws*.exe для Windows-систем. Модули приложений СКАДИВЕБ реализуют серверную функциональность. Они сведены в таблицу 1.

Приложения	Описание	Примечание
WsShm	Стандартный доступ с групповыми запросами	Чтение+запись в ПО СКАДИ
WsShmCntrl	Доступ с возможностью управления	Чтение+запись+управление в ПО СКАДИ
WsShmRd	Доступ только на чтение	Чтение+запись в ПО СКАДИ
WsLocal	Доступ на чтение	Автономный имитатор

Таблица 1. Реализация серверных приложений

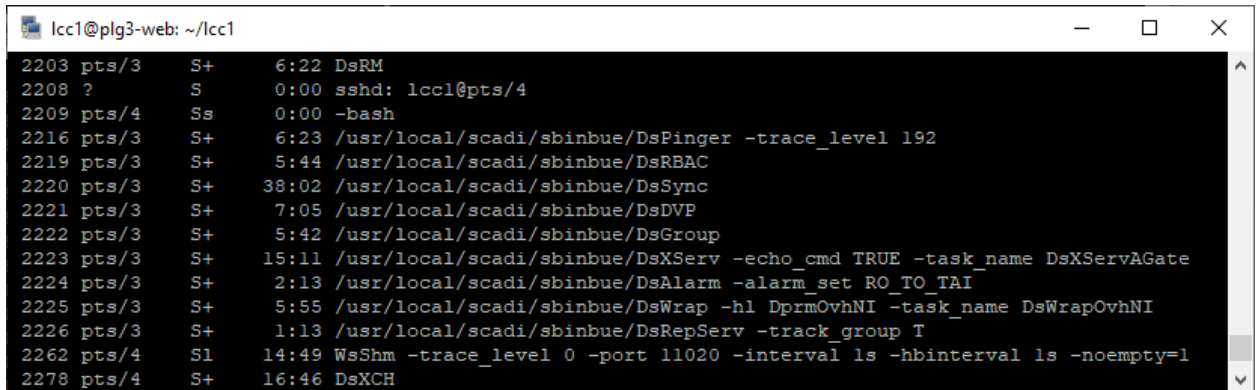
Модули приложений WsShm, WsShmCntrl запускаются в конфигурации ПО СКАДИ. Модуль приложений WsShmRd запускается автономно, но требует путь к файлам общей памяти СКАДИ. Модуль WsLocal является полностью автономным.

Подсистема клиентских модулей взаимодействия с серверами через браузер располагается в подкаталоге web корневого каталога.

Подсистема скриптов генерации данных состоит из утилит преобразования форматов и расположена в подкаталоге scripts каталога инсталляции СКАДИВЕБ.

2.2 СВЯЗИ СОСТАВНЫХ ЧАСТЕЙ ПО

Серверные модули приложений СКАДИВЕБ взаимодействуют с загрузкой ПО СКАДИ. Модули приложений WsShm, WsShmCntrl включаются в проектную загрузку СКАДИ и управляются рантайм-менеджером DsRM. Рантайм-менеджер при старте запускает задачи СКАДИВЕБ, частности модули приложений WsShm, WsShmCntrl. Набор задач для загрузки определяется конфигурацией СКАДИ. На рисунке 1 приведен пример сформированной загрузки, включающей в себя модуль WsShm СКАДИВЕБ.

A terminal window titled 'lcc1@plg3-web: ~/lcc1' displays a list of system logs. The logs show the start and execution of various services, including DsRM, sshd, DsPinger, DsRBAC, DsSync, DsDVP, DsGroup, DsXServ, DsAlarm, DsWrap, DsRepServ, and WsShm. The WsShm entry at 14:49 shows it was started with specific configuration options: -trace_level 0 -port 11020 -interval 1s -hbinterval 1s -noempty=1.

```
lcc1@plg3-web: ~/lcc1
2203 pts/3 S+ 6:22 DsRM
2208 ? S 0:00 sshd: lcc1@pts/4
2209 pts/4 Ss 0:00 -bash
2216 pts/3 S+ 6:23 /usr/local/scadi/sbin/bue/DsPinger -trace_level 192
2219 pts/3 S+ 5:44 /usr/local/scadi/sbin/bue/DsRBAC
2220 pts/3 S+ 38:02 /usr/local/scadi/sbin/bue/DsSync
2221 pts/3 S+ 7:05 /usr/local/scadi/sbin/bue/DsDVP
2222 pts/3 S+ 5:42 /usr/local/scadi/sbin/bue/DsGroup
2223 pts/3 S+ 15:11 /usr/local/scadi/sbin/bue/DsXServ -echo_cmd TRUE -task_name DsXServAGate
2224 pts/3 S+ 2:13 /usr/local/scadi/sbin/bue/DsAlarm -alarm_set RO_TO_TAI
2225 pts/3 S+ 5:55 /usr/local/scadi/sbin/bue/DsWrap -hl DprmOvhNI -task_name DsWrapOvhNI
2226 pts/3 S+ 1:13 /usr/local/scadi/sbin/bue/DsRepServ -track_group T
2262 pts/4 S1 14:49 WsShm -trace_level 0 -port 11020 -interval 1s -hbinterval 1s -noempty=1
2278 pts/4 S+ 16:46 DsXCH
```

Рисунок 1 – WsShm в загрузке рантайм-менеджера

Рантайм-менеджер контролирует жизнеспособность процессов, в частности WsShm. При отсутствии обновлений за период, определяемый конфигурационными настройками, модуль может быть перезапущен менеджером.

Модуль приложения WsShmRd запускается автономно. Он имеет доступ только по чтению. Контроль и поддержка его работы должна осуществляться средствами ОС.

3 НАСТРОЙКА ПРОГРАММЫ

3.1 ИНСТАЛЛЯЦИЯ

3.1.1 Инсталляция для ОС Astra Linux, ОС Linux

Разместить пакет для соответствующей архитектуры в домашнем каталоге текущего пользователя:

- scadiweb_1.0.0_amd64.deb для архитектуры x86-64

Установить пакет ПО СКАДИВЕБ средствами ОС:

```
sudo dpkg -i install scadiweb_1.0.0_amd64.deb (* x86-64*)
```

Типовые каталоги инсталляции подсистем:

/usr/local/scadiweb/sbinur для подсистемы веб-серверного доступа к данным;

/usr/local/scadiweb/web для подсистемы клиентских модулей взаимодействия с серверами через браузер;

/usr/local/scadiweb/scripts для подсистемы скриптов генерации.

3.1.2 Инсталляция для ОС Windows

Разместить инсталлятор ПО СКАДИВЕБ Scadiweb_1_0_0_x86.msi для 32-bit на жестком диске.

Установить ПО СКАДИВЕБ запуском инсталлятора.

3.2 НАСТРОЙКА ПРОЕКТА

3.2.1 Базовые файлы проекта

Поскольку СКАДИВЕБ предназначена для построения систем автоматизации на основе базовой платформы, для ее использования необходимо создание проекта системы автоматизации. Для проекта системы автоматизации создается корневой каталог проекта. При дальнейшей работе все данные и конфигурация, относящиеся к проекту, должны находиться в корневом каталоге проекта.

Установить корневой каталог проекта (вместо названия project подставляется конкретное имя разрабатываемого проекта):

```
mkdir project
```

```
cd project
```

Корневой каталог проекта используется для размещения проектных данных и программного обеспечения, относящихся к конкретному проекту. Он должен иметь определенную структуру для обеспечения доступа к указанным данным.

Проектный каталог для каждого проекта должен иметь структуру для размещения файлов ПО СКАДИ и файлов СКАДИВЕБ.

project> - место для конфигурационных и индексных файлов;

project/web> - корневой каталог клиентского ПО и контента;

project/web/images> - каталог изображений графических форматов;

project/web/vk - каталог видеокладов проекта.

Наличие общего конфигурационного файла должно обеспечивать старт системы командой:

```
project> DsRM -cfg_file DsConfig_<conftype>.cfg
```

В качестве аргументов команды задается имя конфигурационного файла. Если имя конфигурационного файла не задано, то берется по умолчанию DsConfig.cfg. Описание содержимого конфигурационного файла приведено в приложении 1. Описание таблицы Task для заполнения по задачам СКАДИВЕБ приведено в приложении 2. Для модулей WsShm WsShmCtrl необходима конфигурация параметров запуск как задач СКАДИ. Ниже приведен пример конфигурации двух задач WsShm и WsShmArch (задача WsShm для доступа к архивным данным).

```
TASK[11].name=WsShm
TASK[11].exe_path=WsShm
TASK[11].command_line=-port 11020 -interval 100ms -ast_invalid "*****"
TASK[11].run_at=STDSEVER
TASK[11].restart_limit=-1
#TASK[11].init_trace_level=-2
TASK[12].name=WsShmArch
TASK[12].exe_path=WsShm
TASK[12].command_line=-port 11030 -interval 100ms -ast_invalid "*****"
TASK[12].run_at=AUXSEVER
TASK[12].restart_limit=-1
```

В аргументы командной строки command_line заносятся параметры конфигурации задач. В примере заданы порты веб-сервера 11020 и 11030, интервал 100миллисекунд и значение недоверенного сигнала как «***». Перечень всех опций командной строки приведен в приложении 3.

3.2.2 Настройка вне конфигурации СКАДИ

Конфигурация задач WsShmRd и WsLocal должна производиться вне конфигурационного файла DsConfig.cfg. Задача WsLocal запускается полностью автономно. Для обеспечения непрерывной работы WsLocal необходимо включить службу wslocal.service в автозагрузку ОС от имени пользователя root следующей командой:

```
sudo systemctl enable wslocal.service
```

Для запуска необходимо запустить службу wslocal.service от имени пользователя root следующей командой

```
sudo systemctl start wslocal.service
```

Для обеспечения непрерывной работы WsShmRd необходимо включить службу wsshmrd.service в автозагрузку ОС от имени пользователя root следующей командой:

```
sudo systemctl enable wsshmrd.service
```

Для запуска необходимо запустить службу wsshmrd.service от имени пользователя root следующей командой

```
sudo systemctl start wsshmrd.service
```

Конфигурация серверов передается в качестве опций командной строки. Для сервиса WsShmRd необходимо указать полный путь к каталогу проекта СКАДИ, либо запускать его из каталога project проекта СКАДИ. Пример такого вызова будет следующим.

```
WsShmRd -trace_level 0 -port 11010 -interval 1s -hbinterval 1s -noempty=1
```

3.2.3 Настройка групповых ресурсов

Использование приложения WsShm требует конфигурирования областей для групповых запросов. Каждый клиентский экземпляр (даже на одном компьютере может запускаться несколько браузеров и несколько экземпляров клиентских программ в каждом браузере) выделяет для работы область в таблицах CSets и Cells. При доступе к архивным данным также должны выделяться области Reports и Repdatas.

Количество клиентских областей работы устанавливается параметром Host.maxdpr для каждого хоста. Конфигурация для 2 серверных хостов будет выглядеть следующим образом.

```
HOST[0].name=plg3-web
HOST[0].ip_address=192.168.2.3
HOST[0].is_std_server=TRUE
HOST[0].is_server=TRUE
HOST[0].maxdpr=100
#
HOST[1].name=plg3-srv2
HOST[1].ip_address=192.168.2.2
HOST[1].is_std_server=FALSE
HOST[1].is_server=TRUE
HOST[1].maxdpr=100
#
HOST[2].name=comp-dagaevdv
HOST[2].is_std_server=FALSE
HOST[2].is_server=FALSE
HOST[2].maxdpr=1
#
HOST[3].name=comp-ovchinnikov
HOST[3].is_std_server=FALSE
HOST[3].is_server=FALSE
HOST[3].maxdpr=1
```

При старте СКАДИ будет выделено суммарное количество в качестве размера Reports. Сервера добавляют 200 областей, рабочие станции – по одной.

```
s 15 Reports n_total_objs= 202, n_used_objs= 202, sz_used_mem= 131704, usr_data= 29
```

Размер области CSets определяется с учетом умножения на параметр ldr_sets_per_host=1. Для доступа только через веб достаточно иметь значение 1, и размер CSets, равный размеру Reports. Для конфигурирования рабочих станций данный параметр увеличивается до значения числа фреймов на клиентское приложение рабочей станции (в нашем примере 2, диапазон 2..6). В любом случае $size(CSets) = size(Reports) * ldr_sets_per_host$.

```
s 13 CSets n_total_objs= 404, n_used_objs= 404, sz_used_mem= 892032, usr_data= 27
```

Значение числа объектов Repdatas определяет размеры областей отчетных данных. Параметр ldr_repdata_per_report (по умолчанию 300) устанавливает число строк в буфере на отчет. При размере страницы до 60 строк буферизируются по 5 страниц отчета одновременно. Размер области Repdatas будет равен произведению $size(Repdatas) = size(Reports) * ldr_repdata_per_report$.

```
s 16 Repdatas n_total_objs= 60600, n_used_objs= 60600, sz_used_mem= 20361600, usr_data= 28
```

Значение числа объектов Cells определяет размеры областей групповых данных. Для Cells определены параметры ldr_cells_in_set и ldr_cells_small_set. Основной фрейм аллоцируется количеством ldr_cells_in_set ячеек, а остальные ldr_sets_per_host-1 фреймов аллоцируются количеством ldr_cells_small_set ячеек. В нашей конфигурации заданы параметры.

```
ldr_cells_in_set=60
ldr_cells_small_set=1
ldr_sets_per_host=2
```

Число объектов Cells будет определено как $\text{size}(\text{Cells}) = \text{size}(\text{Reports}) * (\text{ldr_cells_in_set} + (\text{ldr_sets_per_host} - 1) * \text{ldr_cells_small_set})$.

```
s 14 Cells n_total_objs= 12322, n_used_objs= 12322, sz_used_mem= 13455624,
usr_data= 26
```

Опция командной строки `-cset false` гарантирует неиспользование групповых ресурсов данным экземпляром сервиса WsShm. При этом сохраняется доступ к переменным по подписке.

3.2.4 Линейное масштабирование проекта

Линейное масштабирование проекта используется при требованиях увеличенного числа пользователей, имеющих доступ к данным через браузер. Линейное масштабирование используется для доступа к ресурсам одной загрузки одного проекта. Для доступа к ресурсам нескольких загрузок или нескольких проектов используются средства контейнеризации.

Масштабирование веб-сокет серверов осуществляется запуском нескольких приложений WsShm* с разными номерами портов.

```
WsShmRd -port 12000 -interval 1s -hbinterval 1s -noempty=1
WsShmRd -port 12001 -interval 1s -hbinterval 1s -noempty=1
WsShmRd -port 12002 -interval 1s -hbinterval 1s -noempty=1
```

Аналогично может прописываться СКАДИ-конфигурация.

```
TASK[11].exe_path=WsShm
TASK[11].command_line=-port 12000 -interval 100ms -ast_invalid "*****"
TASK[12].exe_path=WsShm
TASK[12].command_line=-port 12001 -interval 100ms -ast_invalid "*****"
TASK[13].exe_path=WsShm
TASK[13].command_line=-port 12003 -interval 100ms -ast_invalid "*****"
```

Масштабирование серверов доступа к архивным данным осуществляется запуском нескольких экземпляров DsRepServ с разными именами.

```
TASK[14].name=DsRepServ
TASK[14].command_line=-track_group T -idle_task T
TASK[16].name=DsRepServ2
TASK[16].command_line=-track_group T -idle_task T
```

Для корректной работы в режиме масштабирования каждому экземпляру должны задаваться параметры `track_group`, `idle_task`. Параметр `track_group` включает трассировку по изменениям задания группового отображения. Параметр `idle_task` выбирает наименее загруженный сервер доступа к архивным данным.

Необходимым условием работы серверов доступа к архивным данным является наличие циклических буферов сообщений, по одному для каждого сервера. Типовая их конфигурация будет следующей.

```
GLOBAL[15].globtype=q
GLOBAL[15].globname=RepSq
GLOBAL[15].max_count=15000
GLOBAL[15].reftask=DsRepServ
GLOBAL[15].flags=Q
GLOBAL[15].run_at=SERVER
GLOBAL[25].globtype=q
GLOBAL[25].globname=RepSq2
GLOBAL[25].max_count=15000
GLOBAL[25].reftask=DsRepServ2
GLOBAL[25].flags=Q
GLOBAL[25].run_at=SERVER
```

3.2.5 Настройка режимов записи-воспроизведения

При реализации режимов записи воспроизведения СКАДИВЕБ используется система архивации СКАДИ. Система архивации требует соответствующей настройки. Как минимум, один сервер должен иметь задачу архивации и задачу воспроизведения архива. Данный сервер назовем архивным. Архивный сервер должен получать данные и записывать их в архивные файлы. Приходящие на сервер данные не должны попадать в ячейки значений текущих данных. Ячейки значений текущих данных должны заполняться задачей воспроизведения архива. Задача воспроизведения архива должна отслеживать устанавливаемый в определенной переменной исторический диапазон. При смене диапазона ненулевым значением задача воспроизведения архива должна стартовать. При установке нулевого значения задача воспроизведения архива должна останавливаться.

Ниже приведен вариант конфигурации архивного сервера.

```
TASK[10].name=DsWrapOvhNI
TASK[10].exe_path=DsWrap
TASK[10].command_line=-hl DprmOvhNI
TASK[10].watch_limit=60
TASK[10].restart_limit=-1
TASK[10].update_ms=1000
TASK[10].init_trace_level=0
TASK[10].run_at=SERVER
TASK[13].name=DsArchServ
TASK[13].exe_path=DsArchServ
TASK[13].watch_limit=60
TASK[13].restart_limit=5
TASK[13].update_ms=1000
TASK[13].run_at=AUXSERVER
TASK[15].name=DsArchServPlay
TASK[15].exe_path=DsArchServ
TASK[15].watch_limit=60
TASK[15].restart_limit=5
TASK[15].update_ms=1000
TASK[15].run_at=AUXSERVER
```

Серверная задача DsArchServ используется как задача архивации. Задачу воспроизведения архива реализует DsArchServPlay. Задача DsWrapOvhNI содержит алгоритмику управления диапазоном воспроизведения архива. Заданный диапазон содержится в специальной переменной NI:~4SY0PLY00_CNST.nv. Задание значения интервала производится скриптом ниже.

```
DsAccess -setvars NI:~4SY0PLY00_CNST.nv -sync val -val $1
```

Для возможностей воспроизведения должна задаваться конфигурация проигрывателей для каждого глобала. Например, для глобала аналоговых сигналов AI должен быть определен проигрыватель следующего вида.

```
GLOBAL[22].globtype=a
GLOBAL[22].globname=PLPAI
GLOBAL[22].otype=AI
GLOBAL[22].max_count=30000
GLOBAL[22].ref_count=15000
GLOBAL[22].reftask=DsArchServPlay
GLOBAL[22].aclass=CPNGLOB
GLOBAL[22].artype=NGLOB
GLOBAL[22].afmt=FBA
GLOBAL[22].alength=48
GLOBAL[22].slice_time=* * -1 0
GLOBAL[22].flags=
GLOBAL[22].run_at=AUXSERVER
```

3.2.6 Настройка режима управления

Для реализации режима управления необходимо использовать задачу на основе приложения WsShmCntrl. Задача управления требует отдельного определения в лицензии. Для конфигурирования задачи управления требуется аналогичная WsShm структура.

```
TASK[11].exe_path=WsShmCntrl
TASK[11].command_line=-port 14000 -interval 100ms -ast_invalid "*****"
TASK[11].run_at=SERVER
```

Необходимым условием работы задачи управления является наличие циклических буферов сообщений, по одному для каждой задачи. Типовая их конфигурация будет следующей.

```
GLOBAL[12].globtype=q
GLOBAL[12].globname=Cntrlq
GLOBAL[12].max_count=2000
GLOBAL[12].reftask=WsShmCntrl
GLOBAL[12].run_at=SERVER
```

Прохождение команды определяется конфигурацией СКАДИ и ее агентами обмена.

3.3 НАСТРОЙКА ПЕРЕМЕННЫХ ОКРУЖЕНИЯ

При инсталляции ПО СКАДИВЕБ в каталоги программ. В Linux-системах записывается в каталоги вида /usr/local/scadiweb.

При этом бинарные файлы располагаются в каталогах:

- sbinbur – для 64-битных Linux-систем x64;
- sbinbwr – для 64-битных Windows-систем x64.

Необходимо прописать пути доступа к исполняемым модулям СКАДИ. Например, для Linux.

```
export PATH=${PATH}:/usr/local/scadiweb/sbinbur/
```

Для возможности запуска приложений WsShm WsShmCntrl из СКАДИ необходимо прописать переменную окружения SCADIWEB_PATH.

```
export SCADIWEB_PATH=/usr/local/scadiweb/sbinur
```

3.4 УСТАНОВКА ЛИЦЕНЗИИ ПРОЕКТА

Приложения СКАДИВЕБ WsShm* использует файл лицензии как необходимое условие запуска для проекта. Автономное приложение WsLocal не требует лицензии. Лицензия создается разработчиками ПО СКАДИВЕБ и передается как разрешение на использование ПО СКАДИВЕБ в рамках данного проекта <project>. Файл лицензии должен находиться в каталоге <project>. Файл лицензии должен иметь формат Wss_<hostname>.lic, где hostname является именем хоста стандартного сервера. Ниже пример для хоста sd1.

```
Devel12@sd1:~/plg1$ file Wss_sd1.lic
Wss_sd1.lic: ASCII text
```

При отсутствии валидной лицензии будет выведено сообщение с указанием ошибки лицензии вида: “SCADIWEB License error file=Wss_sd1.lic”. Работа ПО СКАДИВЕБ возможна только при наличии валидной лицензии.

При просроченной лицензии приложением будет выведено сообщение: “SCADIWEB License file=Wss_sd1.lic expired=14/10/2022”. Работа ПО СКАДИВЕБ возможна только при наличии актуальной лицензии.

Лицензия вводит ограничения на возможность использования функций управления. При запуске приложения WsShmCntrl и наличии лицензии без управления выводится

сообщение вида: “SCADIWEB License process control not authorized”. Работа ПО СКАДИВЕБ в режиме управления возможна только при наличии необходимой лицензии.

Валидная лицензия проекта должна устанавливаться при инсталляции проекта.

3.5 ГЕНЕРАЦИЯ ГРАФИЧЕСКИХ ДАННЫХ

Подсистема скриптов генерации данных СКАДИВЕБ использует средства Python3 для собственно скриптов и конфигурационных файлов.

Состав скриптов генерации зависит от проекта. Как правило, сюда входят следующие скрипты. Все эти скрипты размещаются в каталоге vis/gen данного проекта.

Имя скрипта (или шаблон имени)	Назначение	Пример
convall.py	Скрипт генерации видеокладов	Приложение 5
dvlist.py	Файл описания исходных данных для генерации видеокладов. Используется скриптом convall.py	Приложение 6
srcfg.py	Файл настроек системы генерации	Приложение 7
<шаблон>_gen.py	Скрипты генерации исходных данных видеокладов из шаблона	

Генерация видеокладов выполняется командой:
python convall.py

Эта команда запускается в каталоге vis/gen данного проекта.

Сгенерированные таким способом графические данные переносятся в каталог web/vk видеокладов проекта локально или на удаленный хост вручную или с помощью скрипта копирования, специфического для данного проекта. Пример такого скрипта приведен в Приложении 8.

4 ПРОВЕРКА ПРОГРАММЫ

4.1 ДОСТУП ИЗ БРАУЗЕРА

Для удаленного доступа по сети к веб-серверу и веб-сокет серверу необходимо иметь видимость имен и/или адресов удаленного сервера. Такой доступ может быть проверен командой ping.

```
ping plg3-srv2
```

При наличии видимости имен/адресов проверяется доступ из браузера. Для этого в корневом каталоге проекта должен быть файл local.html. Проверка доступа из браузера осуществляется через данную тестовую страницу.

Для проверки доступа из браузера необходимо набрать адрес вида 192.168.2.3:11020/local.html (вместо 192.168.2.3 подставляется имя или адрес серверного хоста). Доступ осуществляется к тестовому файлу local.html с вариантами проверок команд, рисунок 1.

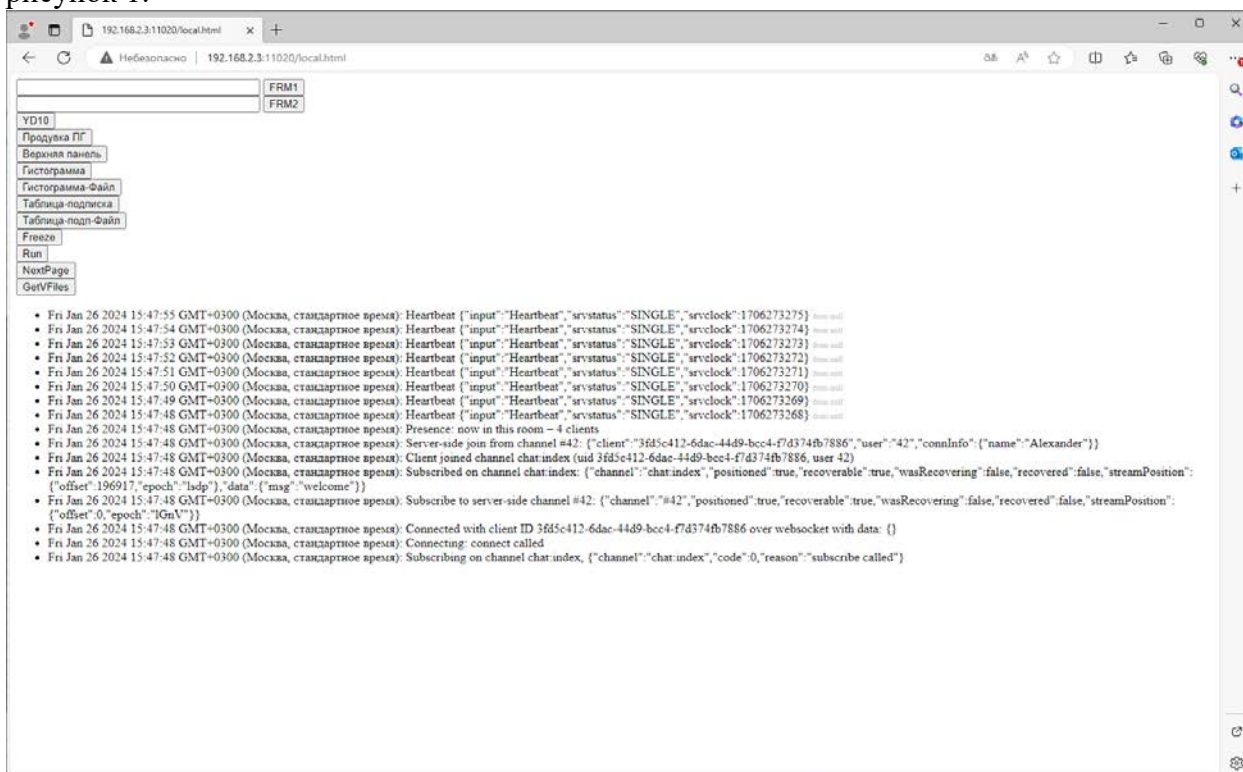


Рисунок 1 – Доступ из браузера к тестовой странице

Файл local.html содержит варианты подписок для проверки работоспособности. Возможны варианты:

- Подписка типового видеокadra;
- Подписка гистограммы с набором идентификаторов;
- Подписка гистограммы с файлом набора идентификаторов;
- Управление режимами останова (freeze) и работы (run);
- Листание страниц подписки;
- Получение списка файлов.

5 ДОПОЛНИТЕЛЬНЫЕ ВОЗМОЖНОСТИ

5.1 ОБЛАЧНЫЕ ТЕХНОЛОГИИ

Для реализации проектов на основе СКАДИВЕБ в облачной инфраструктуре используется механизм докеров. ПО СКАДИВЕБ и конфигурация проекта записывается в контейнеры. Управление и оркестрация контейнеров осуществляется средствами облачной инфраструктуры. Связи между контейнерами с компонентами СКАДИВЕБ осуществляются в локальной части сети.

Связь контейнеров с внешним миром осуществляется через механизмы веб- и веб-сокет серверов по портам, адресованным конечному пользователю.

6 СООБЩЕНИЯ СИСТЕМНОМУ ОПЕРАТОРУ

Задание уровня трассировки осуществляется опцией `-trace_level`. Возможные значения уровня трассировки приведены в приложении 4. Значение по умолчанию равно 0.

Уровень трассировки 1 (TRAT_RUNTIME) приводит к периодической выдаче сообщений статуса следующего вида.

```
2024/01/31 16:06:56 GetStatusMsg [123 34 105 5 110 112 117 116 34 58 34 72 101
97 114 116]
```

Уровень трассировки 2 (TRAT_TRANSPORT) приводит к печати сообщений пользовательской активности.

```
2024/01/31 16:16:54 [user 42] connection is still active
```

Уровень трассировки 8 (TRAT_MESSAGE) приводит к печати сообщений о передаче пользователем данных/запросов на подписку.

```
2024/01/31 16:18:56 [user 42] publishes into channel chat:index:
{"input": "NI:~4SY0PLY01_MDST.nv VAL_DATE_STR11,NI:~4SY0PLY01_MDST.nv
VAL_TIME_STR10,NI:~4SY0PLY01_MDST.nv ACTIVITYEXT14,NI:~4SY0PLY01_MDST.nv
DO_INT,Host:@H.current_time/f", "sframe": "frm0"}
```

Уровень трассировки 32 (TRAT_ERRORS) печатает ошибки коммуникации.

Уровень трассировки 64 (TRAS_ERRORS) печатает ошибки задачи.

Уровень трассировки 128 (TRAS_RUNTIME) приводит к печати сообщений о ходе обработки.

```
2024/01/31 16:37:01 VarParams {GBV [] -report DprepRvp -pagesize 60 -rep_kind
5 -artype STA -globnames . -codes '40YDR10CP001_XQ01 40YDR20CP001_XQ01
40YDR30CP001_XQ01 40YDR40CP001_XQ01 40TKR30CQ001_XQ01' -delta_sec 120 lf hf lw
hw la ha units descr
```

Уровень трассировки 256 (TRAS_AUX) приводит к печати процесса обработки значений переменных. Дает слишком много данных в логе, не стоит использовать без необходимости.

```
2024/01/31 16:41:25 vltf 31/01/24 16:35:53.339 16.60 0.21 124.80
124.80 22.813 0 0 > 31/01/24 16:35:53.339 16.60 0.21
124.80 124.80 22.813 10
```

Уровень трассировки 512 (TRAS_DATA) приводит к выдаче значений переменных. Дает слишком много данных в логе, не стоит использовать без необходимости.

```
2024/01/31 16:43:01 GetValQTS @58.srec: 31/01/24 16:35:53.339 16.60 0.21
124.80 124.80 22.813 , 0, 0
```

Уровень трассировки 1024 (TRAS_TIMES) дает возможность оценки временных характеристик сервиса при работе.

```
2024/01/31 16:44:51 time elapsed: 2.810565ms, msg len:79021
```

Уровень трассировки 2048 (TRAS_HISTORY) дает возможность печати исторических данных.

```
2024/01/31 16:46:15 GotTrendData report_id=105 subscr_id=983660 nset_id=1
amount=144 more=false
```

```
2024/01/31 16:46:15 MsgVars 144 [{RAWVAL,@1._ 16.98;48;1706708739 R4QTS}
{RAWVAL,@2._ 119.3;48;1706708739
```

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1 – ОПИСАНИЕ КОНФИГУРАЦИИ DSCONFIG

Конфигурация представляет собой файл вида `DsConfig_<conftype>.cfg`. Данный файл используется рантайм-менеджером DsRM при создании глобалов общей памяти на этапе старта. Конфигурация состоит из одиночных опций и массивов опций. Одиночные опции могут задаваться в командной строке или в конфигурационном файле. Пример одиночной опции.

```
trace_level=940
```

Массивы опций задаются только в конфигурационном файле. Пример массива опций.

```
CONSTA[0].name=pers_tsd_delay_sec
CONSTA[0].val=10
```

Одиночные опции приведены в таблице 1-1. Сокращения в поле «Диап» означают минимальное значение, значение по умолчанию, максимальное значение.

Опция	Тип	Диап	Описание	Примечание
trace_level	I4	-	Уровень трассировки	
ldr_n_enum s	I4	-	Размер массива констант Enumerations	Рекомендуется 200
ldr_n_types	I4	-	Размер массива констант Types	Рекомендуется 50
ldr_n_attrs	I4	-	Размер массива констант Attributes	Рекомендуется 500
ldr_task_ms gs	I4	-	Размер массива констант Task Messages	Рекомендуется 400
ldr_syslogs _persist_sec	I4	-	Интервал сохранения данных Syslogs	Рекомендуется 60

Таблица 1-1 – Список одиночных опций

Массивы опций приведены в таблице 1-2. Все ссылки на таблицы даны для приложения 2.

Опция	Размер	Тип	Описание	Примечание
CHARACTERISTIC	32	Characteristic	Структура описания характеристики	--
CONSTA	256	S32+I4	Строка описания константы	--
TASK	100	Task	Структура описания задания	Таблица 2-1
HOST	256	Host	Структура описания хоста	--
HGROUP	32	HGroup	Структура описания группы	--
GLOBAL	512	Global	Структура описания глобала	--
XGROUP	32	XGroup	Структура описания коммуникационной группы	--
XCH	256	Xch	Структура узла коммуникации	--

Таблица 2-2 – Список массивов опций

ПРИЛОЖЕНИЕ 2 - ДАННЫЕ ТАБЛИЦЫ TASK КОНФИГУРАЦИИ

Параметрам в таблице конфигурации соответствуют поля структур объектов базы данных реального времени. В приведенных таблицах имя колонки «Параметр» относится к параметру конфигурации, а имя колонки «Поле» относится к полю структуры объекта базы данных реального времени. В конфигурационном файле могут быть только поля с заполненной колонкой «Параметр». Остальные поля инициализируются при старте системы по установленным алгоритмам.

Конфигурационный параметр TASK соответствует типу Task, предназначенному для хранения задач СКАДИ. Структура Task - это запись следующего вида, наследуемая от OGeneric.

```

type Task struct {
    OGeneric
    Process_id, Program_id, N_program int32
    Exe_path String64
    Command_line String128
    Dependencies STag
    Watch_limit, Watch_count, Restart_limit, Restart_count int32
    Is_disabled, Is_debug, Is_master, Must_start bool
    Run_at int32
    Command_result int32
    Task_status SrvStat
    Update_ms int32
    Start_time, Stop_time, Current_time, Cycle_period, Delta_time Time
    Error_count, Error_code_last, Init_trace_level, Trace_level,
    Init_priority, Priority int32
    Error_time_first, Error_time_last, Process_start, Process_end Time
    Current_load, Maximum_load, Cpu_load, Statistics_interval int16
    Write_q, Counter_n int32
    For_recv_q, Recv_q, Read_gap_q, For_send_q, Send_q, Resend_q int32
    I0, I1, I2, I3, I4, I5, I6, I7, I8, I9, IA, IB, IC, ID, IE, IF, IG int32
}

TASK[].name=<string64>
TASK[].exe_path=<string64>
TASK[].dependencies=<string64>
TASK[].command_line=<string128>
TASK[].watch_limit=<int>
TASK[].restart_limit=<int>
TASK[].is_disabled=<bool>
TASK[].is_debug=<bool>
TASK[].init_trace_level=<int>
TASK[].init_proirity=<int>
TASK[].update_ms=<int>
TASK[].run_at=<string64>
TASK[].statistics_interval=<int>

```

Список полей структуры приведен в таблице 2-1.

Параметр	Поле	Наименование	Значения	Примечание
name		Имя	<Строка32>	OGeneric.name
	process_id	Идентификатор процесса	<Число>	getpid()
	program_id	Идентификатор задачи	<Роли>	
exe_path	exe_path	Имя исполняемой задачи	<Строка>	

command_line	command_line	Аргументы командной строки	<Строка>	Запуск: <exe_path> <command_line>
dependencies	dependencies	Зависимости – загруженные задачи	<Строка>	
watch_limit	watch_limit	Предел счетчика исправности	<Число>	
	watch_count	Значение счетчика исправности	<Число>	Перезапуск при 0
restart_limit	restart_limit	Предел счетчика рестартов	<Число>	
	restart_count	Значение счетчика рестартов	<Число>	Прекращение при 0
is_disabled	is_disabled	Задача отключена	BOOL	
is_debug	is_debug	Задача в отладке	BOOL	
	is_master	Задача является основной	BOOL	
	must_start	Задача стартует автоматически	BOOL	
run_at	run_at	Уровень запуска	<Число>	
	command_result	Результат выполнения команды	<Число>	Код возврата процесса
	task_status	Статус задачи	0-OFF 1-INIT 2-LOAD 3- ERROR 4- END 5- WAITING 6- STARTING 7- SINGLE 8- PRIMARY 9- BACKUP 10- FAILOVER	Enum SRV_STAT
update_ms	update_ms	Время обновления	<Интервал>	Миллисекунд
	start_time	Время старта	<Время>	
	stop_time	Время останова	<Время>	
	current_time	Текущее время	<Время>	
	cycle_period	Период цикла	<ИнтервалВрем>	
	delta_time	Время обработки	<ИнтервалВрем>	
	error_count	Счетчик ошибок	<Число>	
init_trace_level	init_trace_level	Начальное значение уровня трассировки	<Число>	
	trace_level	Текущее значение уровня трассировки	<Число>	
init_priority	init_priority	Начальное значение приоритета	<Число>	

	priority	Текущее значение приоритета	<Число>	
	error_time_first	Время первой ошибки	<Время>	
	error_time_last	Время последней ошибки	<Время>	
	process_start	Время старта процесса	<Время>	
	process_end	Время окончания процесса	<Время>	
	current_load	Текущая загрузка процесса	<Число>	
	maximum_load	Максимальная загрузка процесса	<Число>	
	cpu_load	Загрузка ЦПУ	<Число>	
statistics_interval	statistics_interval	Интервал сбора статистики	<Число>	
	write_q	Очередь сообщений для записи	<Номер>	Номер глобала
	counter_n	Счетчик изменений глобала n	<Число>	
	for_recv_q	Счетчик изменений для очереди приема	<Число>	
	recv_q	Счетчик изменений очереди приема	<Число>	
	for_send_q	Счетчик изменений для очереди передачи	<Число>	
	send_q	Счетчик изменений очереди передачи	<Число>	
	resend_q	Счетчик изменений очереди повторной передачи	<Число>	

Таблица 2-1. Список полей структуры Task

Значения TASK представляют данные, относящиеся к функционированию текущей задачи. Идентификация задачи осуществляется по имени, заданном в поле name структуры OGeneric.

ПРИЛОЖЕНИЕ 3 – ОПЦИИ КОМАНДНОЙ СТРОКИ

Опции командной строки СКАДИВЕБ доступны вызовом приложения с флагом – help, рисунок 3-1.

```
C:\DVDagaev\lcc1>WsShm --help
Usage of WsShm:
  -ast_invalid string
    Symbols for invalid substitution
  -cset
    Use CSet booking for queries (default true)
  -hbinterval duration
    Time interval of heartbeat (default 1s)
  -interval duration
    Time interval of update (default 1s)
  -ndigits int
    Number of digits for analog conversion (default 4)
  -noempty
    Send only not empty messages (default true)
  -port int
    Port to bind app to (default 8010)
  -projpath string
    Project path to be connected via shm
  -sendall
    Send all values in cycle
  -task_name string
    Project path to be connected via shm (default "WsShm")
  -trace_level int
    Level of printing trace info
```

Рисунок 3-1 – WsShm с опциями командной строки

СКАДИВЕБ использует следующие опции командной строки, таблица 3-1.

Опция	Описание	Возможные значения
-ast_invalid	Символ недостоверного значения	*** ???
-cset	Возможность групповых запросов	true(по умолчанию) false
-hbinterval	Интервал сердцебиения	1s(по умолчанию), 100ms, 300ms, ...
-interval	Интервал опроса	1s(по умолчанию), 100ms, 300ms, ...
-ndigits	Число значащих цифр	4(по умолчанию) 1-10
-noempty	Посылка только непустых сообщений	true(по умолчанию) false
-port	Номер порта	8010(по умолчанию) число
-projpath	Путь к проекту	Для WsShmRd
-sendall	Посылка всех данных, а не только измененных	false(по умолчанию) true
-task_name	Имя задачи	Задается DsRM при запуске
-trace_level	Уровень трассировки	Число

Таблица 3-1 – Перечень опций командной строки

Возможные значения уровня трассировки приведены в приложении 4. Значение по умолчанию равно 0.

ПРИЛОЖЕНИЕ 4 – ЗНАЧЕНИЯ УРОВНЯ ТРАССИРОВКИ

Для всех модулей приложений СКАДИВЕБ можно динамически устанавливать и менять уровень трассировки. Изменение переменной `Task:@T.trace_level` также отслеживается динамически задачами СКАДИ. Трассировка для задач СКАДИВЕБ задается следующими способами:

- из командной строки указанием `-trace_level 34`
- динамически из отладчика СКАДИ. Добавить переменную `Task:#0.trace_level`, задать значение `1=36`

Значение величины трассировки представляет собой сумму значений бит, согласно таблице 4-1.

Бит	Значение	Коммуникация	Наименование	Обозначение
0	1	да	Цикл обновления	<code>objs.TRAT_RUNTIME</code>
1	2	да	Транспортный уровень	<code>objs.TRAT_TRANSPORT</code>
2	4	да	--	<code>objs.TRAT_POLLING</code>
3	8	да	Логический уровень сообщений	<code>objs.TRAT_MESSAGE</code>
4	16	да	--	<code>objs.TRAT_MESSAGE_DATA</code>
5	32	да	Ошибки коммуникации	<code>objs.TRAT_ERRORS</code>
6	64	нет	Ошибки задачи	<code>objs.TRAS_ERRORS</code>
7	128	нет	Уровень обработки в задаче	<code>objs.TRAS_RUNTIME</code>
8	256	нет	Детализация уровня обработки	<code>objs.TRAS_AUX</code>
9	512	нет	Сообщения уровня обработки	<code>objs.TRAS_DATA</code>
10	1024	нет	Профилирование по времени	<code>objs.TRAS_TIMES</code>
11	2048	нет	Архивные сообщения	<code>objs.TRAS_HISTORY</code>

Таблица 4-1. Список значений трассировки

Опции приведены к значением опций ПО СКАДИ. Опции, обозначенные TRAT*, относятся к транспортному уровню и имеют значение «да» в графе коммуникация. Опции, обозначенные TRAS*, относятся к сервисному уровню и имеют значение «нет».

Для примера, значение 1002 означает установленные биты {1,3,5..9} из таблицы. Все проверки на вхождение бита завершатся включением консольной печати для `objs.TRAT_TRANSPORT`, `objs.TRAT_POLLING`, `objs.TRAT_MESSAGE`, `objs.TRAT_MESSAGE_DATA`, `objs.TRAL_ERRORS`, `objs.TRAS_ERRORS`, `objs.TRAS_RUNTIME`, `objs.TRAS_AUX`, `objs.TRAS_DATA`.

Консольная печать, требуемая для отладки, включается после установки соответствующих опций трассировки. Следует заметить, что при необходимости

трассировки инициализации задачи, нужно установить опцию `-trace_level` в параметрах командной строки.

ПРИЛОЖЕНИЕ 5 – СКРИПТ ГЕНЕРАЦИИ ВИДЕОКАДРОВ

Типичный скрипт генерации видеокадров имеет вид:

```
#!/usr/bin/python3

import os, glob, dvlist

for xg in dvlist.XGEN:
    for xgi in dvlist.XGEN[xg]:
        cmd = dvlist.PYCMD + " " + xg + " " + xgi
        print(cmd)
        os.system(cmd)

ind = 1

while ind in dvlist.CPATH:
    cpl = dvlist.CPATH[ind]
    ind += 1
    if cpl[1].startswith("glob:"):
        cpll = glob.glob(cpl[0] + cpl[1][len("glob:"):])
        for cpi in cpll:
            cmd = dvlist.CCMD + " -svg " + cpi
            print(cmd)
            os.system(cmd)
    else:
        for cpi in cpl[1].split():
            cmd = dvlist.CCMD + " -svg " + cpl[0] + cpi
            print(cmd)
            os.system(cmd)
```

ПРИЛОЖЕНИЕ 6 – ФАЙЛ ОПИСАНИЯ ИСХОДНЫХ ДАННЫХ ДЛЯ ГЕНЕРАЦИИ ВИДЕОКАДРОВ

Пример файла описания исходных данных для генерации видеокадров:

```
import sys

XGEN = dict()
XGEN["GRPDIS_gen.py"] = ["A20", "A30", "A40", "ALM20"]
XGEN["BARV_gen.py"] = ["4", "8", "12"]
XGEN["BARH_gen.py"] = ["4", "8", "12"]
XGEN["XITREND_gen.py"] = ["1 TC 2", "2 TC 2", "3 TC 2", "4 TC 2", "1 TCX 2",
"2 TCX 2", "3 TCX 2", "4 TCX 2"]
XGEN["XTTREND_gen.py"] = ["2 TT", "3 TT m", "4 TT m", "6 TT", "8 TT", "2 TTX",
"3 TTX m", "4 TTX m", "6 TTX", "8 TTX"]
XGEN["XMTREND_gen.py"] = ["10 TM", "20 TM", "30 TM", "40 TM", "10 TMX", "20
TMX", "30 TMX", "40 TMX"]
XGEN["REPORT_gen.py"] = ["1", "2"]
XGEN["SELGRP_gen.py"] = [""]
```



```
CPATH = dict()
CPATH[1] = ["../genicon/", "pan_DATASRC2.dov pan_DATETIME2.dov pan_PAGING2.dov
pan_VKNAME2.dov pan_CRTGRP2.dov pan_SECRANGE.dov pan_SECRANGEA.dov
pan_DIAPA.dov pan_ALME1.dov pan_ALME2.dov pan_ALMW1.dov pan_ALMW2.dov"]
CPATH[2] = ["../genicon/", "test_ANA_175_25.bgd test_LINK.bgd arrow_LINK.bgd
arrow_LINK_VK.bgd arrow_LINK_TC.bgd arrow_LINK_TT.bgd arrow_LINK_TM.bgd
arrow_LINK_ALME.bgd arrow_LINK_ALMW.bgd arrow_LINK_SELGRP.bgd"]
CPATH[3] = ["../genicon/", "arrow_LINK_BARH.bgd arrow_LINK_BARV.bgd
arrow_LINK_TAB.bgd arrow_LINK_REP.bgd"]
CPATH[4] = ["/", "_AGRP20.dov _AGRP30.dov _AGRP40.dov _ALMGRP20.dov _BARV4.dov
_BARV8.dov _BARV12.dov _BARH4.dov _BARH8.dov _BARH12.dov _REPLAND.dov
_REPPORT.dov _SELGRP.dov"]
CPATH[5] = ["/", "DEFS.dov UPPER.dov UPPERA.dov LWBACKGROUND.dov LEFT1.bgd
LEFT1A.bgd _TC1.dov _TC2.dov _TC3.dov _TC4.dov _TCX1.dov _TCX2.dov _TCX3.dov
_TCX4.dov _TT2.dov _TT3.dov _TT4.dov _TT6.dov _TT8.dov _TTX2.dov _TTX3.dov
_TTX4.dov _TTX6.dov _TTX8.dov _TM10.dov _TM20.dov _TM30.dov _TM40.dov
_TMX10.dov _TMX20.dov _TMX30.dov _TMX40.dov"]
CPATH[6] = ["/", "SCADI0.bgd 1_KONT.bgd 2_KONT.bgd YB10.bgd YB20.bgd YB30.bgd
YB40.bgd YD10.bgd YD20.bgd YD30.bgd YD40.bgd BOU1NN.bgd"]
CPATH[7] = ["../../nweb/", "LPAN.bgd WTC20.bgd"]

if sys.platform == "win32":
    PYCMD = "py"
    CCMD = PYCMD + " ../../../../scadi0/scripts/vconv.py"
else:
    PYCMD = "python3"
    CCMD = "vconv.py"
```

Приведенный файл исходных данных определяет следующие словари:

- XGEN – словарь, задающий исходные данные видеок кадров, генерируемых по шаблонам. Ключом в нем служит имя скрипта шаблона, а значением – список, элементы которого задают параметры командной строки, с которым последовательно вызывается данный скрипт.
- CPATH – словарь, задающий набор генерируемых видеок кадров. Ключом в нем служит порядковый номер в наборе, а значением – список из двух элементов. Первый элемент – директория, в которой лежат исходные данные для генерации видеок кадров. Второй элемент – список видеок кадров.

Также определяются:

- PYCMD – команда, вызывающая интерпретатор языка Python3.
- CCMD – команда вызова головного скрипта генерации графических данных. Обычно это скрипт vconv.py, являющийся частью платформы СКАДИ.

ПРИЛОЖЕНИЕ 7 – ФАЙЛ НАСТРОЕК СИСТЕМЫ ГЕНЕРАЦИИ

```
SAVER_PATH="../../scadiweb/scripts"

ICON_PRELOAD_DIR = "./PORTAL_subviews/"
ICON_PRELOAD_EXT = ".svg"
ICON_PRELOAD_DXY = dict()

ICON_PRELOAD_DXY["oisc_Button"] = "-1,-1"

ICON = dict()
#ICON["obj_Button_icon"] = "oisc_Button"

ICON_DYN = dict()
#ICON_DYN["test_ANA_175_25"] = "../genicon/test_DS_ANA.dyn"

ICON_LINK = dict()
#ICON_LINK["test_ANA_175_25"] = "title,AI,XQ01"

ICON_CLINK = dict()
#ICON_CLINK["test_ANA_175_25"] = "X"

ICON_TCLINK = dict()
#ICON_TCLINK["test_VLV"] = "title,DI"

ICON_CMD = dict()
#ICON_CMD["test_ANA_175_25"] = "@ANADATA"

COMMENT = dict()
#COMMENT["YD10"] = "LFARROW YD40;RTARROW YD20"

VK_DEF_SIZE = "1920,980"
VK_SIZE = dict()
#VK_SIZE["LWBACKGROUND"] = "1920,1080"

VK_DEF_SCALE = "100%"
VK_SCALE = dict()
#VK_SCALE["YB10"] = VK_SCALE["YB20"] = VK_SCALE["YB30"] = VK_SCALE["YB40"] =
"96%"

VK_XGROUP = []
#VK_XGROUP = ["_AGRP20", "_AGRP30", "_AGRP40", "_ALMGRP20", "_TC1", "_TC2",
"_TC3", "_TC4", "_TT2", "_TT3", "_TT4", "_TT6", "_TT8"]

BASE_SUBSCR = "attrnames lf hf lw hw la ha units descr;fragname GBV"

VK_XSUBSCR = dict()
for vxg in VK_XGROUP:
    VK_XSUBSCR[vxg] = BASE_SUBSCR

VK_HMI = dict()
VK_HMI["LEFT1"] = "oper"
VK_HMI["LEFT1A"] = "arch"

VA_H = dict()
VA_H["16"] = 12
VA_H["24"] = 17

VA_DY0 = dict()
VA_DY0["18,test_ANA_175_25"] = 0
```

```
VA_DY0["18"] = -7
VA_DY0["10"] = -5

ICON_HREF_PATH = dict()

ICON_HREF_DXY = dict()
#ICON_HREF_DXY["arrow_LINK"] = "0.0,2.0"

ICON_CALLBACK = dict()
#ICON_CALLBACK["test_LINK"] = "onclick,goLink"

BTN_CALLBACK = dict()
#BTN_CALLBACK["goto_online"] = "onclick,goOnline"
#BTN_CALLBACK["!addint"] = "onclick,parent.parent.window.addInt"
#BTN_CALLBACK["!clrSel"] = "onclick,parent.parent.window.clrSel"

GRP_CALLBACK = "onclick,parent.parent.window.addGrp"

IMAGE_DIR = "./images/"

IMAGE_HREF_DXY = dict()

SCRIPT = dict()
#SCRIPT["icon"] = "function goLink(lName) {window.parent.frames[2].open(lName
+ '.svg', '_self')} function frmLink(lName) {window.parent.frmAdd(lName)}"

XSCRIPT = dict()

ICON_NO_BG = dict()
ICON_NO_BG["oisc_Button"] = 1

FONT_CFG_PATH = "./DsFonts.cfg"

MENU_FONT = "Arial,18"
#MENU_WITH_BG = ["LWBACKGROUND"]
MENU_WITH_BG = []

VK_CONV_COLOR = dict()
```

Большинство настроек, задаваемых в этом скрипте, являются проектно-зависимыми и в общем случае могут представлять собой пустые словари и списки. Необходимым является задание следующих параметров:

- SAVER_PATH – путь к подсистеме скриптов генерации платформы СКАДИВЕБ.

ПРИЛОЖЕНИЕ 8 – СКРИПТ ПЕРЕНОСА СГЕНЕРИРОВАННЫХ ГРАФИЧЕСКИХ ДАННЫХ

```
rm -rf web/vk
mkdir web/vk
cp vis/gen/*.svg web/vk

for h in plg3-web; do
    for p in web; do
        tar cvf - $p | ssh lcc1@$h "cd lcc1;tar xvf -;chmod 775
$p"
    done
done
```

ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

АЭС	- Атомная Электростанция
КСА	- Комплекс средств автоматизации
ОС	- Операционная система
ПО	- Программное обеспечение
LLVM	- Low Level Virtual Machine – Виртуальная низкоуровневая машина
СКАДИ	- Средства комплексной автоматизации и диагностики
ЧМИ	- Человеко-машинный интерфейс
ICMP	- Internet Control Message Protocol
TCP	- Transmission Control Protocol
UDP	- User Datagram Protocol

