

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ

«СКАДИ»

(ООО «СКАДИ»)

УТВЕРЖДАЮ

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

Описание программы

34185873.425510.004.ОП.80.М

(На 25 листах)

2023

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

РАЗРАБОТАЛ

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

СОДЕРЖАНИЕ

1 ОБЩИЕ СВЕДЕНИЯ	5
1.1 ОБОЗНАЧЕНИЕ И НАИМЕНОВАНИЕ ПРОГРАММ	5
1.2 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ФУНКЦИОНИРОВАНИЯ ПРОГРАММЫ	5
1.3 ЯЗЫКИ ПРОГРАММИРОВАНИЯ, НА КОТОРЫХ НАПИСАНА ПРОГРАММА	5
2 ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ	6
2.1 КЛАССЫ РЕШАЕМЫХ ЗАДАЧ	6
2.2 НАЗНАЧЕНИЕ ПРОГРАММЫ И ОСНОВНЫЕ ФУНКЦИИ	6
2.3 СВЕДЕНИЯ О ФУНКЦИОНАЛЬНЫХ ОГРАНИЧЕНИЯХ НА ПРИМЕНЕНИЕ.....	7
3 ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ.....	8
3.1 ОСНОВНЫЕ АЛГОРИТМЫ И МЕХАНИЗМЫ ВЗАИМОДЕЙСТВИЯ.....	8
3.1.1 Взаимодействие с возможностью запросов	8
3.1.2 Взаимодействие с возможностью запросов и управления	9
3.1.3 Взаимодействие в режиме только на чтение	11
3.1.3 Взаимодействие в режиме имитации только на чтение.....	12
3.2 ИСПОЛЬЗУЕМЫЕ МЕТОДЫ ОБМЕНА ДАННЫХ.....	12
3.2.1 Обмен браузера с серверами.....	12
3.3 СТРУКТУРА ПРОГРАММЫ	14
3.4 ПОСИСТЕМА ВЕБ-СЕРВЕРНОГО ДОСТУПА К ДАННЫМ	15
3.4.1 Структура приложений подсистемы.....	15
3.4.2 Состав реализаций приложений.....	16
3.4.3 Библиотека функций преобразования данных.....	16
3.5 ПОСИСТЕМА КЛИЕНТСКИХ МОДУЛЕЙ ВЗАИМОДЕЙСТВИЯ.....	16
3.5.1 Структура подсистемы	16
3.6 ПОСИСТЕМА СКРИПТОВ ГЕНЕРАЦИИ ДАННЫХ	17
3.6.1 Структура подсистемы	17
3.7 СВЯЗИ С ДРУГИМИ ПРОГРАММАМИ	18
3.7.1 Доступ к таблицам ПО СКАДИ	18
3.7.2 Циклические буферы сообщений.....	19
3.7.3 Связь с внешним источником данных	20
4 СТАТИЧЕСКИЕ ДАННЫЕ	22
4.1 СОСТАВ ФАЙЛОВ СИСТЕМЫ ОТОБРАЖЕНИЯ	22
4.2 РАСШИРЕНИЯ ФОРМАТА SVG	23
4.3 ОПИСАНИЯ ШАБЛОНОВ КОМАНД.....	28
5 ДИНАМИЧЕСКИЕ ДАННЫЕ	29
5.1 ВЕБ-СОКЕТ СООБЩЕНИЯ ФОРМАТА JSON	29
6 ИСПОЛЬЗУЕМЫЕ ПРОГРАММНО-ТЕХНИЧЕСКИЕ СРЕДСТВА	31
7 ВЫЗОВ И ЗАГРУЗКА.....	32
ПРИЛОЖЕНИЯ	33
Приложение 1 – Перечень встроенных функций преобразования данных	33
Приложение 2 – Список типов глобалов общей памяти	34
Приложение 3 – Встроенные типы данных	35
Приложение 4 – Базовые части структурных типов	37
Приложение 5 - Данные пользовательских реляционных таблиц	39
Приложение 6 – Структуры циклических буферов.....	43

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Приложение 7 – Таблицы данных динамических объектов	45
Приложение 8 – Файл шаблона команд	47
Приложение 9 – Перечень сообщений формата JSON	49
Приложение 10 – Пример клиентского веб-сокета доступа	52
Перечень терминов и сокращений.....	59
Лист изменений.....	60

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

1 ОБЩИЕ СВЕДЕНИЯ

1.1 ОБОЗНАЧЕНИЕ И НАИМЕНОВАНИЕ ПРОГРАММ

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

Сокращенное наименование: «СКАДИВЕБ».

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

Наименование документа 34185873.425510.004.ОП.80.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 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ФУНКЦИОНИРОВАНИЯ ПРОГРАММЫ

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.

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.

1.3 ЯЗЫКИ ПРОГРАММИРОВАНИЯ, НА КОТОРЫХ НАПИСАНА ПРОГРАММА

«СКАДИВЕБ 1.0» написана на языке Go в серверной части, на языке JavaScript в клиентских модулях и на языке Python в части скриптов генерации данных. Плагины взаимодействия с ПО СКАДИ написаны на языке Оберон.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

2 ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

2.1 КЛАССЫ РЕШАЕМЫХ ЗАДАЧ

СКАДИВЕБ является платформой или интеграционным средством доступа к данным технологических процессов АЭС и промышленных предприятий. С помощью СКАДИВЕБ осуществляется доступ к оперативным и архивным данным в режиме онлайн.

Использование СКАДИВЕБ позволяет решать следующие классы задач:

- разработка информационных средств мониторинга и аналитики перспективных АСУТП АЭС и промышленных объектов;
- разработка моделей действующих систем АСУТП АЭС для обучения персонала;
- проведение планового импортозамещения программных платформ, используемых в составе ряда аналитических систем действующих АЭС и промышленных объектов российской юрисдикции.

2.2 НАЗНАЧЕНИЕ ПРОГРАММЫ И ОСНОВНЫЕ 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.

Интеграция ПО СКАДИВЕБ с ПО СКАДИ означает обеспечение возможности доступа СКАДИВЕБ к данным СКАДИ в режиме чтения или в двунаправленном режиме запись-чтение.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

2.3 СВЕДЕНИЯ О ФУНКЦИОНАЛЬНЫХ ОГРАНИЧЕНИЯХ НА ПРИМЕНЕНИЕ

СКАДИВЕБ может быть использована как информационная система без функций управления в АСУ ТП.

СКАДИВЕБ может быть использована как информационно-управляющая система в тренажерах, в частности, в тренажерах АСУ ТП.

3 ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

3.1 ОСНОВНЫЕ АЛГОРИТМЫ И МЕХАНИЗМЫ ВЗАИМОДЕЙСТВИЯ

3.1.1 Взаимодействие с возможностью запросов

Онлайн-доступ к оперативным и архивным данным СКАДИВЕБ реализуется по клиент-серверной модели с подпиской. В качестве тонких клиентов используются браузеры, серверные задачи работают на хостах. Главный механизм осуществляет доступ к значениям базы данных реального времени с возможностью запросов и управления (рисунок 1).

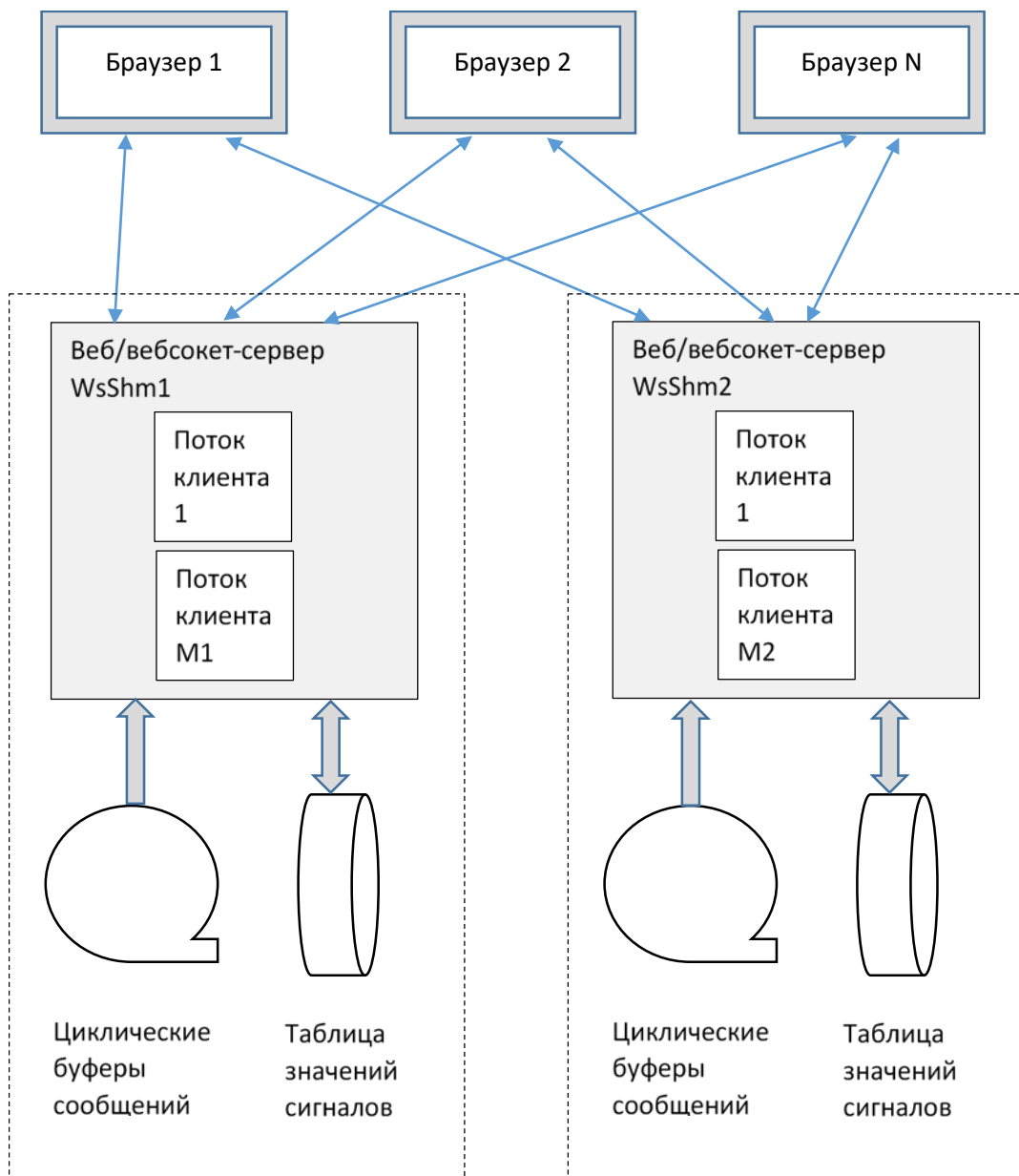


Рисунок 1 – Главный механизм взаимодействия СКАДИВЕБ

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Главный механизм использует серверные приложения с доступом на чтение и запись.

В нижней части серверные приложения связаны с базой данных реального времени (БДРВ) СКАДИ. БДРВ включает в себя таблицы текущих значений сигналов и циклические буферы сообщений. Таблицы содержат актуальные значения, буферы сообщений содержат последовательности изменений сигналов.

Технически БДРВ располагается в разделяемой памяти. Помимо значений и очередей, в БДРВ содержится множество метаданных, определяющих информацию о типах, диапазонах, единицах измерения и прочих параметрах значений сигналов.

В верхней части каждый сервер реализует веб-интерфейс http/https и веб-сокеты интерфейс wss для работы с браузерами. Веб-сокеты интерфейс используется для организации подписки при задании набора данных для мониторинга. Действия клиента по выбору видеокadra, группы, отчета приводят к изменению набора данных для подписки. После установки набора данные начинают посылаться от сервера к браузеру по веб-сокеты интерфейсу. И это будет происходить до тех пор, пока не сменится подписка.

Клиент может обращаться к другому веб-серверу, например, если оперативные и архивные данные размещены на разных узлах. При этом контроль и управление подпиской переходит к приложению, расположенному на новом узле.

Помимо подписки, клиент через веб-сокеты-сервер может запрашивать выполнение удаленной встроенной процедуры, например, запроса списка файлов на сервере или запроса смены страницы отображаемых данных.

Главный механизм имплементируется серверной задачей WsShm.

3.1.2 Взаимодействие с возможностью запросов и управления

Кроме запросов, возможна реализация выдачи команд управления в тренажеры технологических процессов. Команды управления записываются в циклические буферы сообщений и передаются в качестве команд в модель тренажера.

Механизм взаимодействия с выдачей команд управления представлен на рисунке 2.

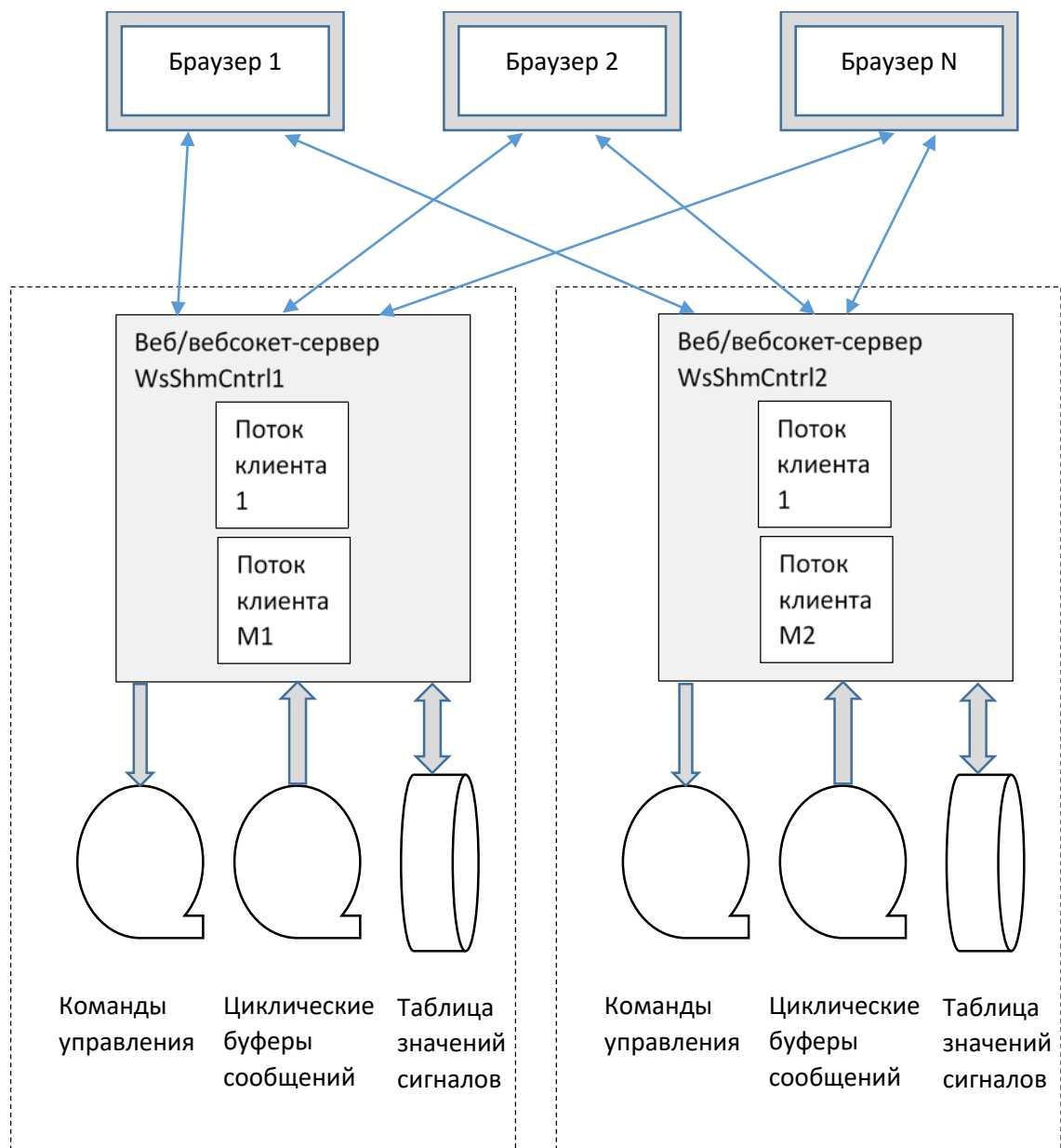


Рисунок 2 – Механизм взаимодействия СКАДИВЕБ с выдачей команд управления

Каждому экземпляру вебсокета-сервера ставится в соответствие циклический буфер, в который записываются команды управления. Распространение и архивирование буфера с командами управления осуществляется средствами серверного программного обеспечения СКАДИ.

Последовательность команд управления, формируемая в циклическом буфере, содержит, помимо информации непосредственно о командах, данные о клиенте, который указанные команды инициировал.

Механизм взаимодействия СКАДИВЕБ с выдачей команд управления имплементируется серверной задачей WsShmCntrl.

3.1.3 Взаимодействие в режиме только на чтение

Доступ только на чтение к оперативным данным СКАДИВЕБ реализуется также по клиент-серверной модели с подпиской (рисунок 3).

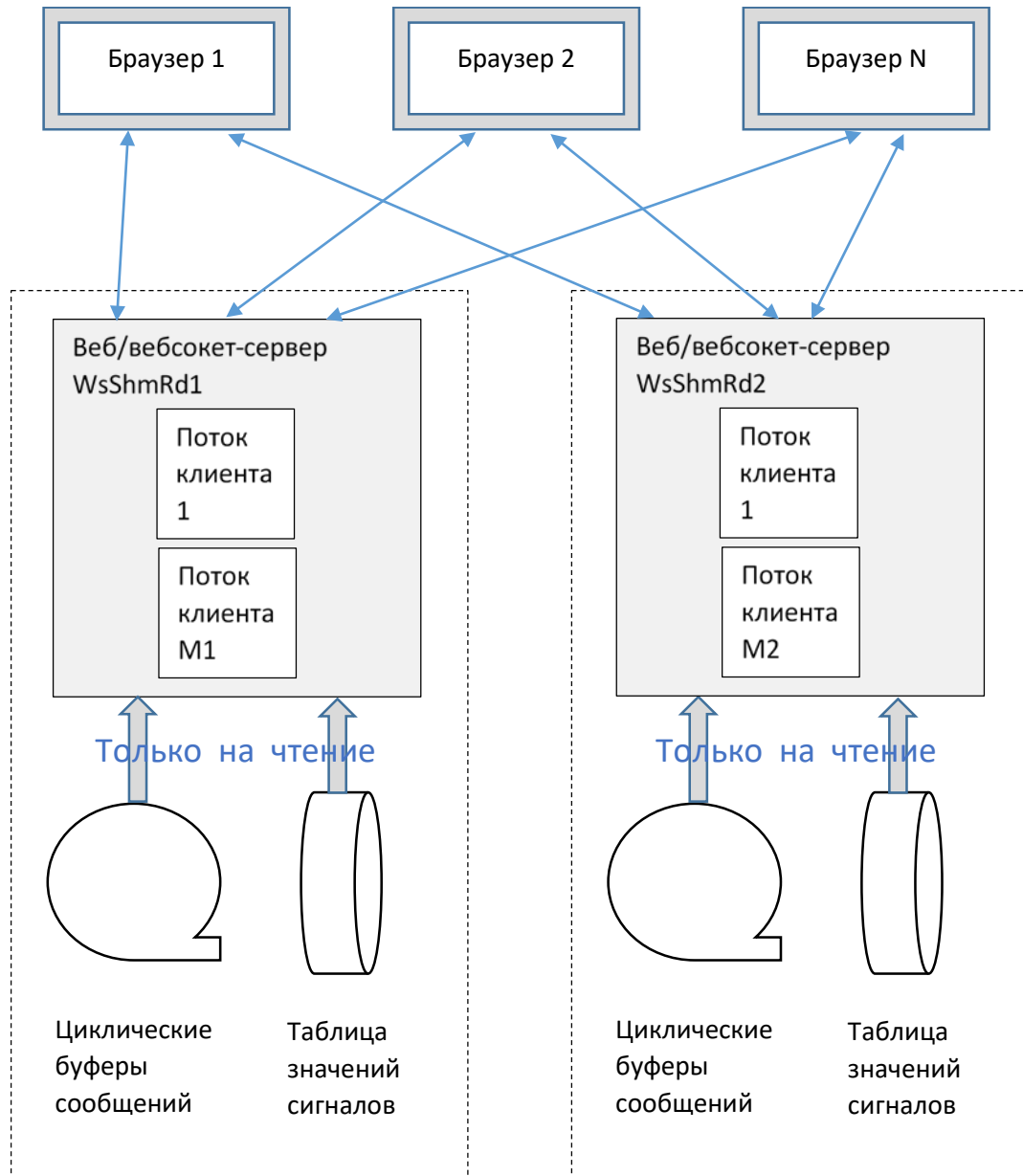


Рисунок 3 – Взаимодействие СКАДИВЕБ только на чтение

При таком механизме серверное приложение подключается к общей памяти только на чтение. Никаких записей в общую память серверная задача осуществить не может. И, следовательно, серверная задача не может оказывать влияния на процесс работы системы СКАДИ.

Механизм взаимодействия в режиме только на чтение имплементируется серверной задачей WsShmRd. Серверная задача WsShmRd запускается автономно, не в составе задач СКАДИ.

3.1.3 Взаимодействие в режиме имитации только на чтение

Доступ в режиме имитации является доступом только на чтение. В режиме имитации серверная задача работает автономно, данные СКАДИВЕБ имитируются также по клиент-серверной модели с подпиской (рисунок 4).

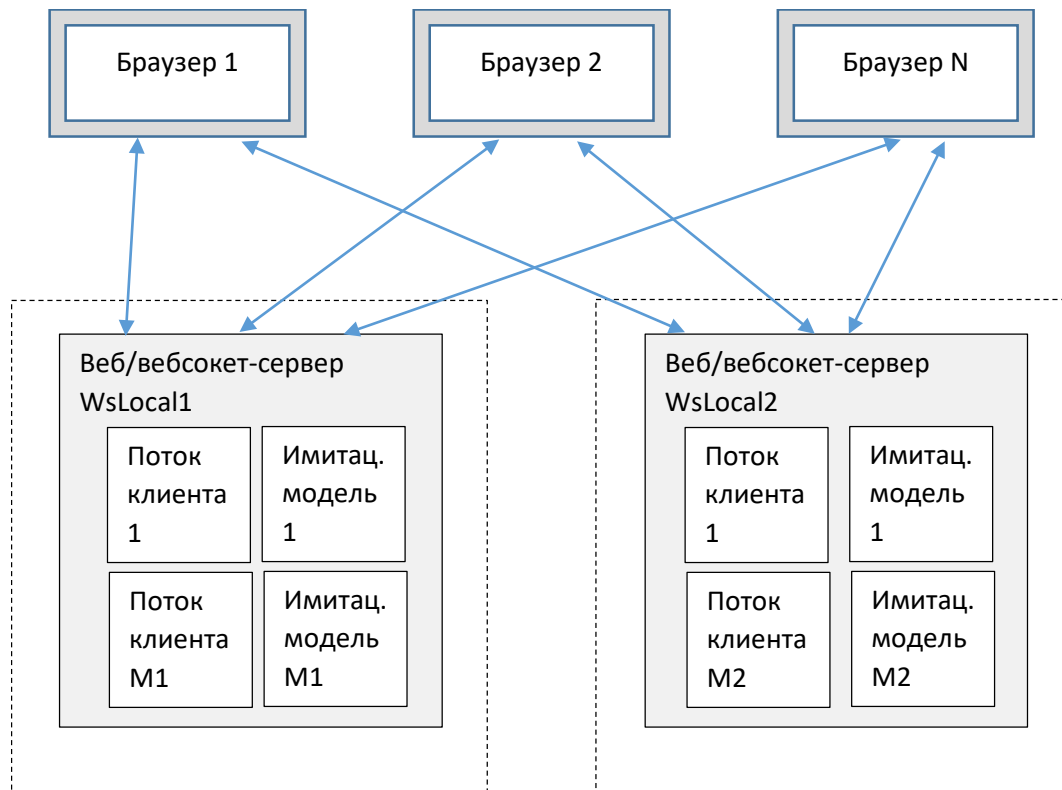


Рисунок 4 – Взаимодействие СКАДИВЕБ в режиме имитации

При таком механизме серверное приложение не присоединяется ни к каким ресурсам. По идентификаторам данных, запрашиваемых на подписку, устанавливаются диапазоны значений имитации. В каждом цикле осуществляется генерация случайных значений в диапазоне имитации.

Механизм взаимодействия в режиме имитации имплементируется серверной задачей WsLocal. Серверная задача WsLocal запускается автономно, не в составе задач СКАДИ.

3.2 ИСПОЛЬЗУЕМЫЕ МЕТОДЫ ОБМЕНА ДАННЫХ

3.2.1 Обмен браузера с серверами

Обмен данными браузера с серверами осуществляется по протоколам http/https для веб-сервера и wss для веб-сокета сервера. Рисунке 5 представлена последовательность включения указанных методов обмена.

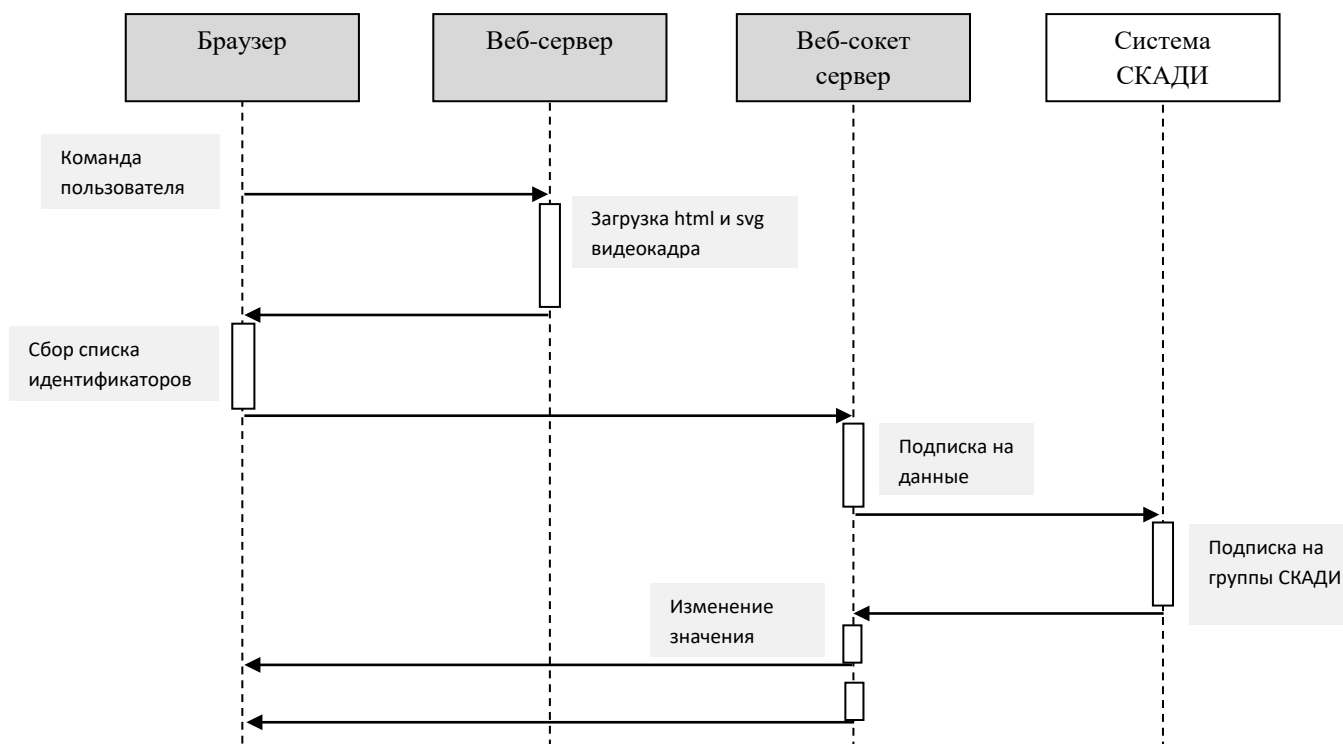


Рисунок 5 – Последовательность методов обмена данными

При пользовательском запросе в браузере реализуется команда загрузки веб-контента html и видеокadra, передаваемая веб-серверу. Полученные от веб-сервера страницы и видеокadры отображаются в браузере, после чего осуществляется сбор списка идентификаторов с динамической привязкой к переменным процесса. Перечень идентификаторов и строка параметров подписки передаются на веб-сокет сервер. Веб-сокет сервер устанавливает подписку на перечень идентификаторов. При наличии строки параметров подписки последняя передается в систему СКАДИ, где устанавливается как подписка на группы СКАДИ. Строка параметров подписки определяется в меню. Она не является пустой для подписок на групповое отображение, сигнализации, графики, отчеты. Строка параметров подписки в рамках подписки на группы использует ресурсы СКАДИ.

В свою очередь, обычные видеокadры не имеют строки параметров подписки и не используют ресурсы СКАДИ.

На данные, соответствующие выбранным группам и идентификаторам, устанавливается подписка для выбранного клиента. Это означает, что серверная задача периодически контролирует изменение значений, и, при фиксации таких изменений, передает значение, код качества и метку времени клиентскому приложению. Следующий цикл обновления повторяет клиентскую итерацию контроля и передачи значений.

3.3 СТРУКТУРА ПРОГРАММЫ

СКАДИВЕБ является платформой для построения систем АСУ ТП АЭС и промышленных предприятий. Все системы, реализованные на основе СКАДИВЕБ, наследуют ее клиент-серверную архитектуру с механизмами подписки данных.

В СКАДИВЕБ входят следующие функциональные составляющие (рисунок 6):

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

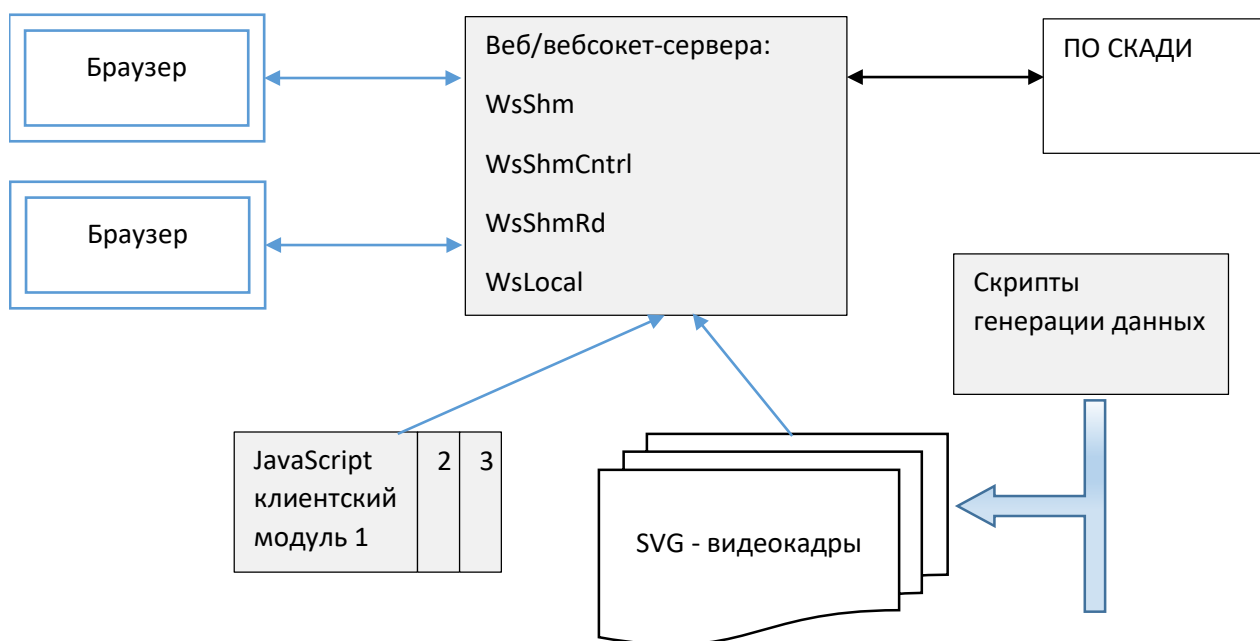


Рисунок 6 – Структура программы СКАДИВЕБ

Подсистема веб-серверного доступа к данным состоит из приложений – веб+веб-сокет серверов, через которые осуществляется доступ к данным из браузеров.

Подсистема клиентских модулей взаимодействия состоит из программных скриптов JavaScript, загружаемых с серверного каталога в браузер.

Подсистема скриптов генерации данных состоит из программных компонент преобразования текстовых данных из формата СКАДИ в формат SVG.

Данные видеокадры, а также подмодели, изображения, равно как и скрипты, располагаются в проектных каталогах, определенных для каждой системы на основе СКАДИВЕБ. Веб-данные проектного каталога структурируются в корневом каталоге web/, доступном для работы серверных задач.

Все платформа СКАДИВЕБ и подсистемы, на основании ее сделанные, работают в веб-инфраструктуре.

3.4 ПОСИСТЕМА ВЕБ-СЕРВЕРНОГО ДОСТУПА К ДАННЫМ

3.4.1 Структура приложений подсистемы

Серверные приложения подсистемы имеют общую объектную структуру, различаясь способами реализации модулей. Объектная структура построена вокруг программного обеспечения веб/веб-сокет сервера Centrifuge, рисунок 7.

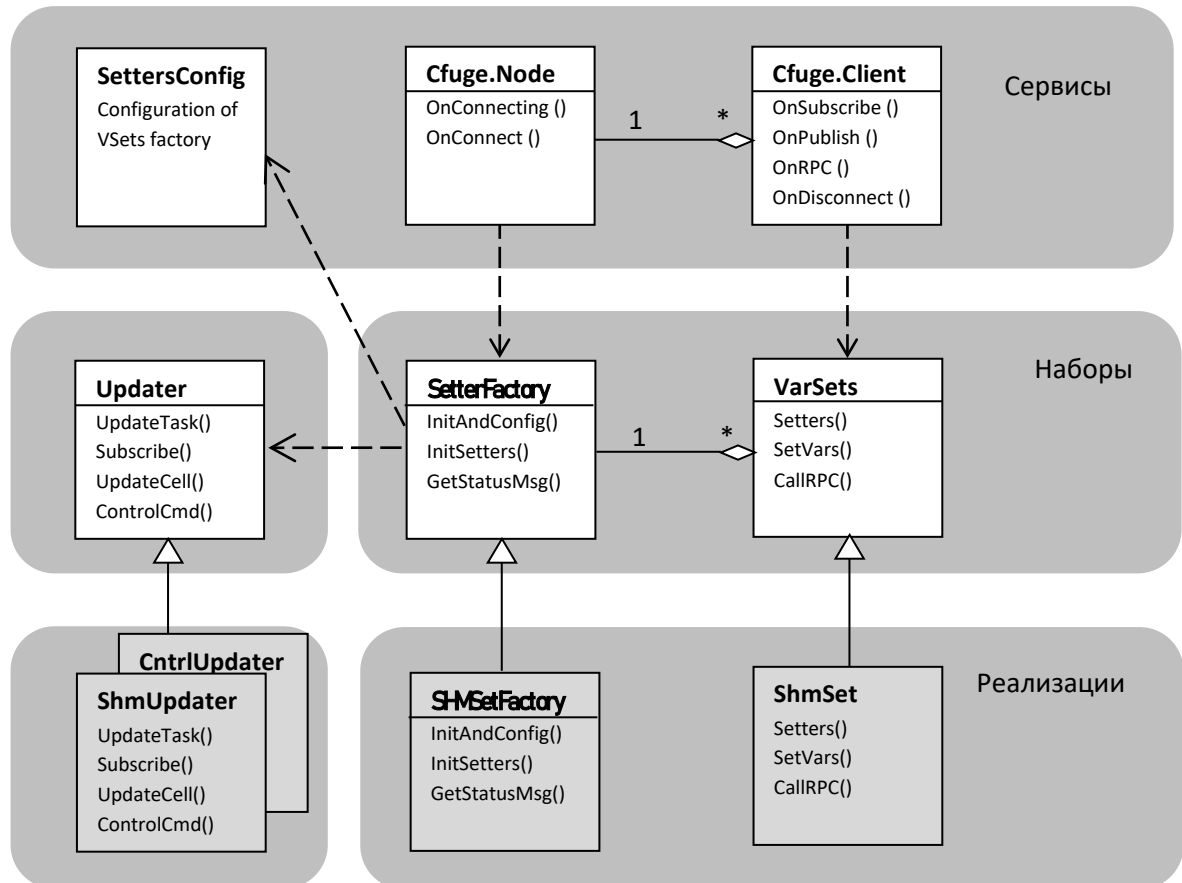


Рисунок 7 – Объектная структура сервисных приложений

Сервер Svc реализует функции веб-сервера, передавая размещенные в локальных каталогах html, svg, png и др. файлы клиентскому приложению. Сервер Svc также реализует функции веб-сокет-сервера, создавая для каждого коннекта экземпляр типа Client, который в дальнейшем осуществляет взаимодействие с приложением браузера в своем потоке.

Клиент имеет структуру VSets, содержащую несколько наборов данных, по одному для каждого фрейма видеокадра. Экземпляры VSets для каждого клиента создаются фабричным объектом VSetsFactory.

Помимо наборов данных, в системе используется также интерфейс Updater, регламентирующий доступ к серверным данным, необходимым для отображения. Например, метод Subscribe() устанавливает строку и номер подписки для задачи группового отображения СКАДИ. Последующие данные группового отображения устанавливают соответствие запрошенных групповых параметров реальным идентификаторам сигналов. При изменении такого соответствия выдается положительная информация на метод

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

UpdateCell(). Команда ControlCmd() инициирует выдачу команды управления, если объект реализации поддерживает такую команду (поддерживает только CntrlUpdater).

3.4.2 Состав реализаций приложений

Реализации интерфейсов Updater, VSets, VSetsFact зависят от требуемой функциональности серверного приложения, таблица 1.

Приложения	Updater	SetterFactory
WsShm	shmUpdater	SHMSetFactory
WsShmCntrl	cntrlUpdater	SHMSetFactory
WsShmRd	zeroUpdater	SHMSetFactory
WsLocal	zeroUpdater	RandFactory

Таблица 1. Реализация серверных приложений

Указанный подход в реализации обеспечивает гарантию отсутствия функциональности сверх требуемой. Например, приложение WsLocal не имеет доступа к данным СКАДИ, поэтому все экземпляры объектов VSets, создаваемые фабрикой RandFactory, способны работать лишь со случайными данными. Приложение WsShmRd, имеющее пустой объект обновления общей памяти zeroUpdater, не способно обновлять общую память СКАДИ, из которой считываются данные. Связь с данными СКАДИ для приложения WsShmRd осуществляется в режиме «только-на-чтение». Приложение WsShm способно создавать запросы на выполнение групповых операций СКАДИ, но не способна выдавать команды управления. Реализация shmUpdater не имеет доступ на запись к глобалам q очередей сообщений. В свою очередь, приложение WsShmCntrl с помощью реализации cntrlUpdater выдает команды управления в сконфигурированные для данного сервера глобалы q.

3.4.3 Библиотека функций преобразования данных

При подписке на данные клиенты СКАДИВЕБ могут указывать для каждого сигнала функцию преобразования. Функция преобразования должна реализовывать следующий прототип.

```
type Converter func (value string, quality int16, timestamp int64, bobj
[]byte, offset int32, votype VOType, atype int16, clientdata string)
(string, int16)
```

Каждая функция преобразования должна иметь свое уникальное в рамках библиотеки имя, по которому она извлекается при запросе указанного имени. Перечень встроенных функций преобразования приведен в приложении 1.

При необходимости расширения номенклатуры функций преобразования дополнительно используется проектно-зависимая библиотека функций. Проектно-зависимая библиотека представляет собой динамически загружаемый модуль so/dll и подключается посредством `-dlconv <library>`.

3.5 ПОСИСТЕМА КЛИЕНТСКИХ МОДУЛЕЙ ВЗАИМОДЕЙСТВИЯ

3.5.1 Структура подсистемы

Подсистема клиентских модулей взаимодействия состоит из модулей на языке JavaScript. Назначение отдельных модулей приведено в таблице 2.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 01.12.2023	1
-------------	---	----------------------	---

Имя модуля	Назначение
centrifuge.js	Обмен данными с веб-сокет сервером
cfuge.js	Реализация шаблонов взаимодействия с веб-сокет сервером для однооконного веб-интерфейса
xcfuge.js	Реализация шаблонов взаимодействия с веб-сокет сервером для многооконного веб-интерфейса
svd.js	Библиотека классов, реализующих элементы веб-интерфейса и динамические объекты
dialog.js	Реализация диалогов

Таблица 2. Назначение клиентских модулей взаимодействия

Назначение классов, реализованных в модуле svd.js, приведено в таблице 3.

Имя класса	Назначение
Svd	Реализация подмодели
HLDyn	Реализация глобального объекта динамики
Trend	Реализация графиков
XAxis	Реализация оси абсцисс для гистограмм
YAxis	Реализация оси ординат для гистограмм и графиков
TAxis	Реализация оси времени для графиков
Tip	Реализация всплывающей подсказки
ScadiDoc	Реализация контейнера для обработки загруженного SVG-документа

Таблица 3. Назначение классов библиотеки svd.js

В модуле dialog.js реализован класс ControlDialog, используемый для создания окон диалогов, в том числе окон управления в управляющих системах. Также в этом модуле определена глобальная переменная controls, представляющая собой ассоциативный массив, содержащий открытые в данный момент диалоги.

3.6 ПОСИСТЕМА СКРИПТОВ ГЕНЕРАЦИИ ДАННЫХ

3.6.1 Структура подсистемы

Подсистема скриптов генерации данных включает в себя скрипт csaver.py на языке Python 3. Этот скрипт реализует функцию сохранения файлов видеок кадров в расширенном формате SVG. Он не вызывается непосредственно, для обращения к нему используются скрипты генерации данных платформы СКАДИ.

Головная функция сохранения файла определена в этом скрипте следующим образом:

```
def save_file(fname, tl, edict, eprops, pdict):
    fname – имя файла, в котором сохраняется видеок кадр.
```

tl – словарь данных, содержащий внутреннее представление исходного видеок кадра, сгенерированное скриптами генерации данных платформы.

Остальные параметры не используются при сохранении данных и нужны для совместимости прототипа функции с системой генерации данных платформы СКАДИ.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

3.7 СВЯЗИ С ДРУГИМИ ПРОГРАММАМИ

3.7.1 Доступ к таблицам ПО СКАДИ

Связь СКАДИВЕБ с ПО СКАДИ осуществляется путем доступа серверных задач СКАДИВЕБ к распределенной общей памяти СКАДИ. Распределенная общая память состоит из так называемых глобалов – выделенных при старте системы блоков фиксированного размера. Каждый глобал состоит из заголовка и массива данных определенной структуры.

Заголовок глобала представляет собой 24-байтную структуру в составе:

```
type SHDR struct {
    N_total_objs int32
    N_used_objs int32
    Sz_used_mem int32
    Usr_data int32
    Length int64
}
```

Заголовок каждого глобала содержит поля:

- максимальное число объектов N_total_objs;
- текущее число объектов N_used_objs;
- размер используемой памяти Sz_used_mem;
- поле переменного значения Usr_data, обычно используемое как номер типа или индекс в очереди;
- длину Length, используемую в зависимости от типа глобала.

Массив данных определенной структуры заполняется в зависимости от типа глобала. Список типов глобалов общей памяти приведен в приложении 2.

Все типы глобалов СКАДИ подразделяются на 2 категории: таблицы и циклические буферы. Таблицы используются для хранения текущих данных, циклические буферы используются для хранения последовательностей данных.

Текущие данные представляют собой объекты определенного типа. Типы данных подразделяются на встроенные и структурные, представляющие собой композицию встроенных. Встроенные типы данных приведены в приложении 3.

Структурные типы данных представляют собой записи, состоящие из встроенных типов. Они имеют корни общего наследования, определенные в таблицах OGeneric, OConfigurable, OManagable (приложение 4).

Основные структурные типы приведены в приложении 5.

Массив данных определенного структурного типа записывается в глобал после заголовка. Например, для глобала AIs, состоящего из объектов AI с номером типа usr_data=37 получится следующий массив данных.

```
type ArrayAI = [N_total_objs]AI;
```

При этом объекты AI, описанные в приложении 5, будут представлять собой структурные типы с полями Units, Lf, Hf, La, Na, ...

Доступ к полям объектов осуществляется при наличии информации о структуре данных объектов. Значения индексов объекта Oid и полей Name, Code, Pin структур OGeneric и OConfigurable позволяют осуществлять привязку адресов объектов к именам с последующим быстрым доступом к данным.

Доступ к объектам пользовательских таблиц AC, DI, DC, NI осуществляется аналогично.

Доступ к объектам системных таблиц осуществляется аналогично для таблиц:

- Task – задачи СКАДИ;

- CSet – параметры групп отображения;
- Cell – ячейки групп отображения;
- VFile – таблица файлов выборки;
- FVar - переменные файлов выборки.

3.7.2 Циклические буферы сообщений

Глобал q представляет собой циклические буферы сообщений, расположенные в общей памяти. Общая память для очереди сообщений представляет собой массив фиксированной длины SHDR.N_total_objs. Массив состоит из элементов фиксированного размера 48 байт. Как структура данных массив используется в качестве циклического буфера, рисунок 8.

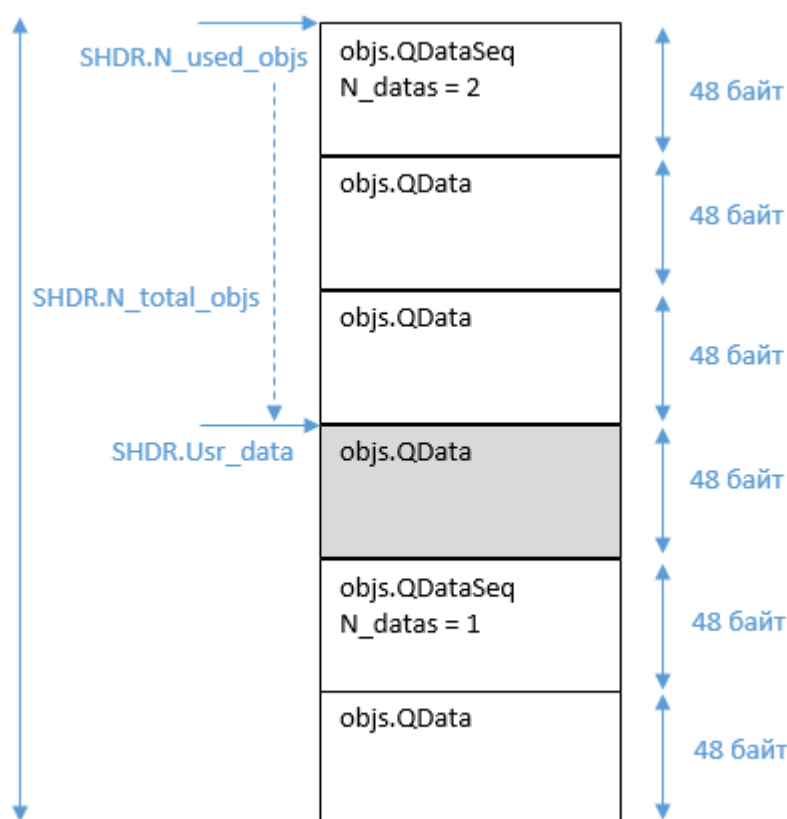


Рисунок 8 – Циклический буфер очереди сообщений

Циклический буфер очереди сообщений перезаписывается в направлении сверху-вниз, по возрастанию индекса. Перед записью индекс записывается в параметры SHDR.N_used_objs, SHDR.Usr_data. При записи ячейки переписываются новыми данными, и индекс SHDR.Usr_data увеличивается. Перед окончанием записи наблюдается картина, показанная на рисунке, когда shdr.usr_data указывает на следующую ячейку «старых» данных, а SHDR.N_used_objs еще указывает на начало последней записи – элемент objs.QDataSeq. В памяти указанные типы различаются по параметру ref.ature, который равен -1 для типа objs.QDataSeq. При достижении индексом конца массива SHDR.N_total_objs индекс обнуляется, и начинается движение с начала буфера.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Циклический буфер очереди сообщений считывается, начиная с индекса SHDR.Usr_data. При этом старые данные могут пропускаться до первого элемента последовательности objs.QDataSeq.

Последовательность команд управления, формируемая в циклическом буфере, содержит, помимо информации непосредственно о командах, данные о клиенте, который указанные команды инициировал.

3.7.3 Связь с внешним источником данных

Связь СКАДИВЕБ с внешним источником данных осуществляется через ПО СКАДИ. ПО СКАДИВЕБ реализует связь с ПО СКАДИ, ПО СКАДИ реализует связь с внешним источником (рисунок 9).

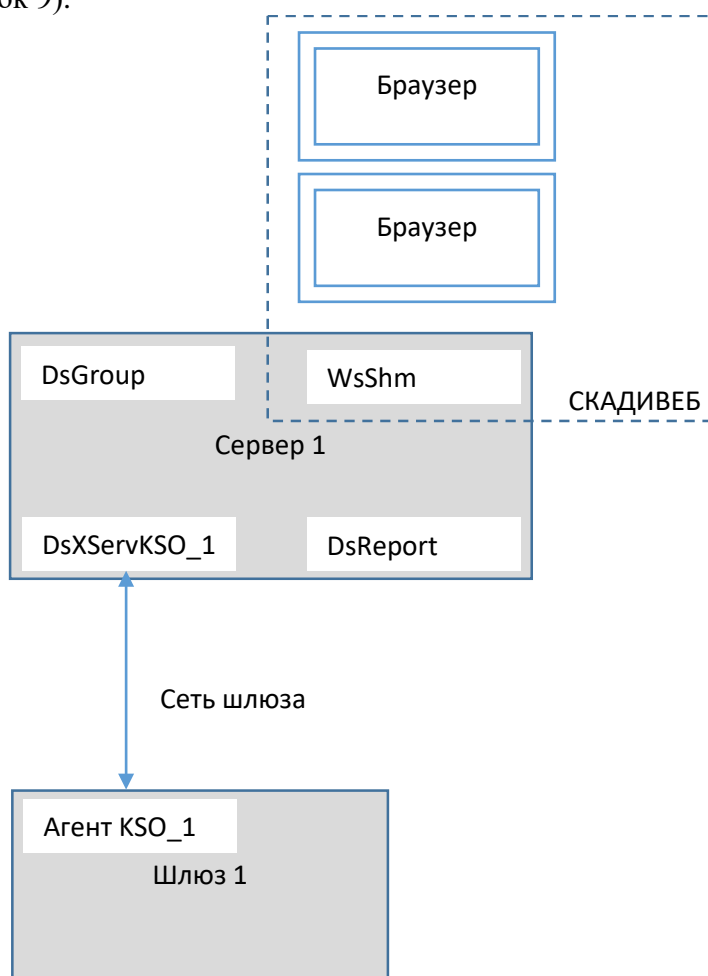


Рисунок 9 – Получение данных от внешнего источника

На приведенном рисунке данные от шлюза 1 через связь агента обмена – сервер обмена DsXServKSO_1 поступают в СКАДИ.

Данные записываются в таблицы СКАДИ и в циклические буферы сообщений. По запросу серверного приложения WsShm при необходимости включаются в работу модули DsGroup при подписке на групповое отображение и DsReport при запросе данных отчета или архивного графика.

Серверное приложение WsShm направляет изменившиеся данные клиентским браузерам согласно запрошенным подпискам каждого клиента.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

4 СТАТИЧЕСКИЕ ДАННЫЕ

4.1 СОСТАВ ФАЙЛОВ СИСТЕМЫ ОТОБРАЖЕНИЯ

Серверный корневой каталог web содержит статические данные, запрашиваемые и отображаемые браузером. Эти статические данные устанавливаются для каждого разрабатываемого проекта. Корневой каталог web вставляется в проектный каталог. Он содержит данные следующих типов:

- Веб-страницы *.html;
- Видеокадры vk/*.svg;
- Изображения images/*.png;
- Контейнеры видеокадров *wgap.svg;
- Скрипты, загружаемые в браузер.

На рисунке 10 приведен пример организации фреймов и контейнеров для отображения в браузере.

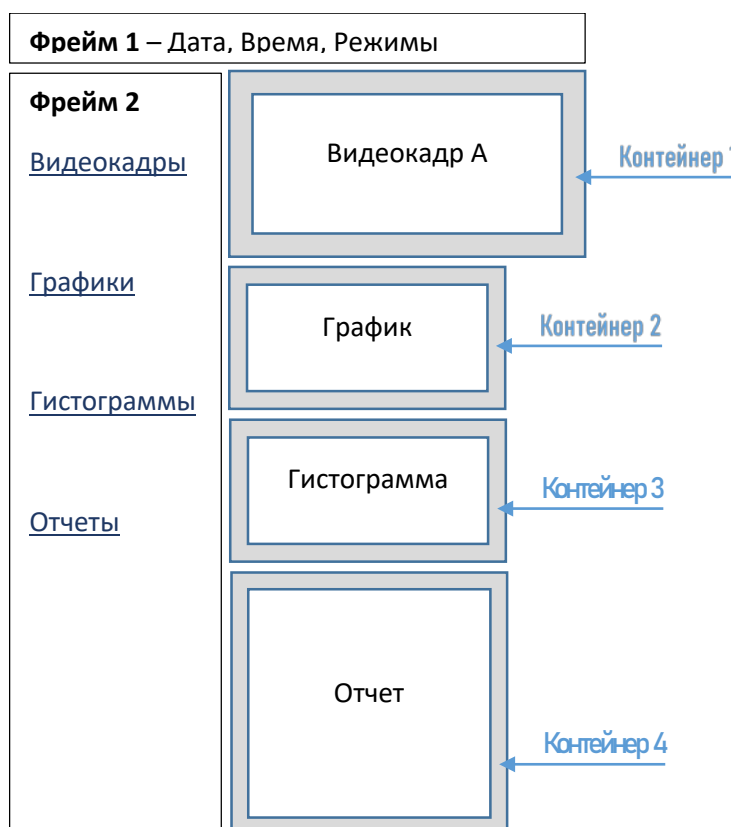


Рисунок 10 – Фреймы и контейнеры браузера

Во фреймы браузера загружаются html-страницы или svg-контейнеры. Контейнеры определяют область загрузки видеокадра указанного типа с элементами управления контейнером.

Все векторные графические файлы соответствуют стандарту Scalable Vector Graphics (SVG) 1.1 (Second Edition), за исключением видеокадров, которые используют расширения SVG, описанные в следующем разделе.

4.2 РАСШИРЕНИЯ ФОРМАТА SVG

Расширения SVG нужны для привязки анимации данных к переменным процесса СКАДИВЕБ. Все расширения формата SVG начинаются со слова scadiweb-, их перечень приведен в таблице 4.

Имя СКАДИВЕБ	Имя СКАДИ	Описание
scadiweb-thr	tt	Таблица данных
scadiweb-var	Var	Переменная процесса
scadiweb-dmisc	Dmisc	Динамика цвета кисти dforeground, фона dbackground и видимости dvisible
scadiweb-dtext	Dtext	Динамика текста
scadiweb-dcoord	Dcoord	Динамика заполнения координат dfill_left (right, up, down) и движения dmove_h, dmove_v
scadiweb-dconf	Dconf	Динамика конфигурации
scadiweb-dseq	Dseq	Динамика обработки последовательностей

Таблица 4. Перечень расширений формата SVG

Таблица данных представляет собой массив из объектов границ данных scadiweb-thr. Размерность массива определена параметром ntt. Объекты scadiweb-thr также имеют именованные параметры с установленными значениями. Например, так записывается динамика цвета фона с таблицей данных

```
<scadiweb-dmisc objtype="Dmisc" dyn="dbackground" refoid="#gav"
groid="#name_4" dt="t" ntt="13" imax="64" fmax="64.0">
  <scadiweb-thr ind="1" tr="0" idat="12632256" />
  <scadiweb-thr ind="2" tr="4096" idat="16711680" />
  <scadiweb-thr ind="3" tr="17920" idat="255" />
  <scadiweb-thr ind="4" tr="18432" idat="65535" />
  <scadiweb-thr ind="5" tr="18944" idat="16776960" />
  <scadiweb-thr ind="6" tr="19456" idat="16711680" />
  <scadiweb-thr ind="7" tr="24576" idat="32768" />
  <scadiweb-thr ind="8" tr="25088" idat="16776960" />
  <scadiweb-thr ind="9" tr="25600" idat="16711680" />
  <scadiweb-thr ind="10" tr="26112" idat="16711680" />
  <scadiweb-thr ind="11" tr="26624" idat="16776960" />
  <scadiweb-thr ind="12" tr="27136" idat="16776960" />
  <scadiweb-thr ind="13" tr="27648" idat="16711680" />
</scadiweb-dmisc>
```

С объектом динамики Dmisc связывается таблица данных размером ntt=13. Первый элемент таблицы содержит запись с параметрами ind="1" tr="0" idat="12632256". Описание таблицы данных приведено в приложении 7.

Объект типа scadiweb-var представляет собой переменную процесса. Типовое задание переменной следующее.

```
<scadiweb-var objtype="Var" name="plink" vname="plink" vartype="10" gflags="6"
feval="$ModName" id="plink" />
```

Параметр vname задает имя переменной привязки в формате func,type:name.attr. Значение type должно соответствовать реальному типу из глобала T=Types. Вместо name подставляется имя объекта привязки, вместо attr подставляется атрибут объекта привязки. Значение attr должно соответствовать реальному атрибуту глобала A=Attrs. Необязательный параметр func задает функцию обработки значения, при этом возвращается результат работы функции, а не само значение. Параметр vartype задает тип значения, возвращаемой переменной или функцией, если последняя используется.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Параметр `vartype` соответствует встроенному типу данных СКАДИВЕБ, приложение 3. Как атомарные, так и структурные встроенные типы могут возвращаться переменной или функцией объекта `scadiweb-var`. Объект `scadiweb-var` получает значения, качества и метки времени, которые используются непосредственно или обрабатываются функциями. Функция обработки значения выбирается из таблицы по имени. Она может быть встроенной или взятой из плагина обработчика. Перечень встроенных функций приведен в приложении 1.

Имя переменной может содержать специальные символы, используемые для идентификации объектов в базе данных реального времени СКАДИ.

Идентификация объекта по имени `name` осуществляется по правилу:

Идент = `name`

Имя соответствует полю `name` объекта в базе данных реального времени.

Идентификация объекта по коду `code` осуществляется по правилу:

Идент = `~code`

Код соответствует полю `code` объекта в базе данных реального времени.

Идентификация объекта по контакту `pin` осуществляется по правилу:

Идент = `+pin`

Контакт соответствует полю `pin` объекта в базе данных реального времени.

Идентификация объекта по макросу осуществляется по правилу:

Идент = `@macro`

Возможные значения макроса зависят от типа таблиц:

- `@H` означает подстановку индекса хоста в таблице `Hosts`;
- `@T` означает подстановку индекса текущей задачи в таблице `Tasks`;
- `@[a-f]<num>` означает подстановку номера строки в групповом отображении в таблице `Cells`. При этом символ в имени типа `&` обозначает ссылку на объект, а не поле. Наличие ссылки на объект позволяет подключить любой адресуемый атрибут, относящийся к данному объекту;
- `@[a-f]` означает подстановку номера группы в групповом отображении в таблице `Csets`.

При групповом отображении данных вместо имен используются шаблоны вида `@a1.pagam`, что соответствует заданию `name.pagam` для объекта 1, и так далее для всех объектов из группы.

Параметр `feval` объекта `scadiweb-var` задает шаблон подстановки для генерации имени переменной или константы. Шаблон имеет вид текстовой строки `aaabbb$PARccss`, в которой осуществляется подстановка параметра `PAR` значением параметра, определенным привязкой. В качестве параметра `PAR` могут выступать имена `oid`, `name`, `code`, `type`, `cntr`, а также имена из параметризации `type-cntr`. Например, для параметризации `-type-cntr {-X T}` должен быть шаблон, использующий данную параметризацию, например, `-feval $X=`. При подстановке имя константы получит значение «T».

Параметр `gflags` объекта `scadiweb-var` задает флаги вида используемой переменной, таблица 5.

Бит	gflags	Имя флага	Описание
0	1	GFLA_THENAME	Явное задание имени, отменяет подстановку в подмоделях <code>obj -objtype Var -vname VAL_DATE_STR,Task:DsRM.current_time -vartype 10 -gflags 1</code>
1	2	GFLA_FEVAL	Указание на то, что параметр <code>feval</code> используется как подстановка

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

			obj -objtype Var -feval SVALX,AI:%\$code.av -gflags 2 -name av -vname av -vartype 10
2	4	GFLA_CONST	Задано имя не переменной, а константа со строковым значением
			obj -objtype Var -feval \$X= -gflags 6 -name rxc -vname par -vartype 10
3	8	GFLA_MNUARGS	Осуществляется подстановка передаваемых аргументов командной строки
			obj -objtype Var -name vtag -vname vtag -feval TAG,DI:~\$code_Z0.name -gflags 10 -vartype 10
4	16	GFLA_ICONST	Задано имя не переменной, а константа с целочисленным значением
			obj -objtype Var -feval \$ival -gflags 18 -name iv12 -vartype 3
5	--	GFLA_CONNECT	Устанавливается системой, если необходима привязка переменной к процессу

Таблица 5. Флаги вида переменной с примерами

Нулевой бит устанавливается кодом gflags=1, при этом имя в подмодели используется явно без подстановки. В примере таким именем является VAL_DATE_STR,Task:DsRM.current_time – текущее время. Бит 1 означает подстановку с feval, вместо SVALX,AI:%\$code.av в имени будет подставлен конкретный технологический код, например, SVALX,AI:%RL71T01.av. Бит 2 означает константу со строковым значением, например подстановка из “\$X=” строки “T=” при параметризации -type-cntr {-X T}. Бит 3 означает, что в подстановке используются не объекты параметризации подмодели, а аргументы командной строки. Это происходит при формировании параметров окон управления. Вызов командной строки представляет собой запрос окна вида @VIDEOCADR ?-name name? ?-code code? ... Например, @VLV -name 1DI0005 -code RL71T01 -objtype DI -x 579 -y 526. Бит 4 означает константу с целочисленным значением, например подстановка из “\$ival” значения “4” при параметризации -type-cntr {-ival 4}.

Параметр var_names объекта scadiweb-var задает строковое имя или перечень строковых имен переменных/параметров, используемых функциями преобразования func.

Динамические объекты scadiweb-d* предназначены для анимации графических объектов значениями переменных процесса. Они вызываются системой при изменении выходных значений объектов scadiweb-var, полученных из значений сигналов, кодов качества, результатов функций-обработчиков и т.д. Динамические объекты содержат привязку объектов scadiweb-var к конкретным типам анимации: цвету, видимости, тексту, положению, миганию и т.д.

Конкретный тип объекта может охватывать несколько типов динамик. Это позволяет добавлять новые типы динамик к неизменному перечню динамических объектов.

Объект типа scadiweb-dseq реализует динамику последовательной передачи трендам значений, качества и метки времени. Также содержит информацию об используемом текущем временном интервале вида (начало, конец, страница, флаг изменения). Пример использования объекта.

```
<scadiweb-var objtype="Var" name="rav1" vname="RAWVAL,@1._" vartype="15"
oid="12" id="rav1" />
<svg foreground="7206159" qcl="0" line_width="2" prolong="1" ylimit="1"
qdash="2" prodash="5" min="0" max="1" isrect="1" stroke="#6df50f"
oname="graph1" dlname="DpdgTrends.Trend" dlx="70" dly="17" dlwidth="1740"
dlheight="773" taxref="taxis1" />
<scadiweb-dseq objtype="Dseq" refoid="#rav1" groid="#graph1" oid="15" />
```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Данный пример осуществляет необходимую привязку последовательности событий Dseq, формируемой текущими значениями сигнала RAWVAL,@1._. Данная последовательность передается для отображения в графический объект DpdgTrends.Trend с именем graph1. Графический объект, реализованный плагином DpdgTrends.Trend, отображает пришедшие данные в виде динамического тренда.

Объект типа Dtext реализует динамику изменения текста, в зависимости от переменных привязки. Предполагается, что на вход поступает изменяемая строка символов UTF-8. Размер строки задается начальной длиной параметра str привязанного к динамике графического объекта Text. Пример использования объекта.

```
<scadiweb-var objtype="Var" name="vsval8" vname="VFX,@8._" vartype="10"
oid="386" id="vsval8" />
<scadiweb-dtext objtype="Dtext" refoid="#vsval8" groid="#tsval8" oid="387" />
<text name="tsval8" font-align="6" x="1655.555555555557" y="853"
fill="#ffffff" oid="385" stroke="#1000000" xml:space="preserve" font-
family="Arial" font-size="18" dy="-7.0">SVAL</text>
```

Данный пример осуществляет привязку отображаемого значения переменной vsval8 к текущему значению Var в общей памяти. Текущее значение VFX,@8._ форматируется в строку посредством функции VFX. Функция VFX учитывает число значащих цифр для аналоговой переменной определенного диапазона и преобразовывает значение в текстовый формат. При изменении текстового значения данные от динамического объекта Dtext передаются в связанный с ним посредством groid текстовый объект text, и отображаемая строка меняется.

Объект типа Dmisc реализует динамику изменения цвета фона, кисти и видимости, определяемую значением параметра dyn. Видимость dvisible принимает 2 значения, 0 (FALSE) относится к невидимой области. Пример использования объекта.

```
<line name="mha8" stroke="#ff0000" x1="1740.0" y1="63" x2="1760.0" y2="63"
oid="372" />
<scadiweb-var objtype="Var" name="vmha8" vname="NVAL,@8.ha" vartype="5"
oid="373" id="vmha8" />
<scadiweb-dmisc objtype="Dmisc" dyn="dvisible" dt="t" ntt="2" fmin="-1"
fmax="1" groid="#mha8" refoid="#vmha8" oid="375">
<scadiweb-thr ind="1" tr="0" idat="0" />
<scadiweb-thr ind="2" tr="16384" idat="1" />
</scadiweb-dmisc>
```

Данный пример задает таблицу данных видимости mha8 по значениям верхней аварийной границы параметра в базе. При отсутствии нижней аварийной границы нормализованное значение NVAL,@8.ha становится нулевым, и выбирается значение idat=0. При наличии нижней аварийной границы нормализованное значение NVAL,@8.ha становится единичным, и выбирается значение idat=1 из таблицы. Таблицы данных описаны в приложении 7.

Цвета кисти и фона объекта Dmisc задаются значениями dforeground и dbackground соответственно. Цвета могут задаваться по таблице стандартных значений для кода качества или посредством цветового RGB-типа.

```
<scadiweb-var objtype="Var" name="barcolor8" vname="QUALITY,@8._" vartype="3"
oid="353" id="barcolor8" />
<scadiweb-dmisc objtype="Dmisc" dyn="dbackground" dt="r" refoid="#barcolor8"
groid="#bar8" oid="354" />
<rect name="bar8" stroke="#-00001" fill="#8080ff" x="1660.555555555557" y="64"
width="84.44444444444446" height="548" oid="350" />
```

Данный пример задает цвета кода качества, по которым отображается цвет фона прямоугольника гистограммы.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Объект типа Dconf реализует динамику изменения конфигурации параметра объекта в зависимости от переменных процесса. Значение переменной любого типа конвертируется в строку и передается объекту типа Dconf. Измененное значение передается ассоциированному с ним графическому объекту для изменения конфигурации его параметра pname. Если значение изменилось, предполагается, что Dconf выполнит команду <grobj> config -<pname> <value>. В качестве функции обработки значения обычно используется преобразование в строку, чтобы избежать преобразования по умолчанию, не регулирующего числа значащих цифр.

Примером может служить конфигурация верхнего предела графика, зависящего от диапазона параметра.

```
<svg background="0xffffffff" foreground="0xffffffff" dash_size="94.44444444444446"
fill="#ffffff" stroke="#ffffff" font-family="Times" font-size="18"
oname="yaxis8" dlname="DpdgTrends.YAxis" dlx="1605.5555555555557" dly="64"
dlwidth="50" dlheight="548">
<scadiweb-var objtype="Var" name="shf8" vname="VFX,@8.hf" vartype="10"
dimlen="6" oid="357" id="shf8" />
<scadiweb-dconf objtype="Dconf" groid="#yaxis8" refoid="#shf8" pname="max"
oid="359" />
```

Объект типа Dcoord реализует динамику изменения координат. Тип динамики определяется параметром dyn и подразделяется на две группы, первая означает заполнение, вторая – позицию. Группа с заполнением включает типы динамики dfill_left, dfill_right, dfill_up, dfill_down. При этом форма ассоциированного с динамикой графического объекта заполняется соответственно влево, вправо, вверх, вниз. Ниже приведен пример динамики заполнения гистограммы вправо.

```
<scadiweb-var objtype="Var" name="nval8" vname="NVAL,@8._" vartype="5"
oid="365" id="nval8" />
<rect name="bar8" stroke="#-00001" fill="#8080ff" x="841" y="935.375"
width="908" height="39.8125" oid="364" />
<scadiweb-dcoord objtype="Dcoord" dyn="dfill_right" refoid="#nval8"
groid="#bar8" oid="366" />
```

Для работы подобной динамики обычно применяется функция нормализации значения NVAL, возвращающая действительное число в диапазоне (0,1). Нулевому изменению диапазона соответствует нулевое заполнение прямоугольника, значению 0.5 соответствует заполнение от левой границы на 50%. Значению 1 соответствует полное заполнение.

Группа Dcoord с позицией включает типы динамики dmove_h, dmove_v. При этом ассоциированный с динамикой графический объект сдвигается по горизонтали или вертикали соответственно. Ниже приведен пример динамики позиции.

```
<line name="mha8" stroke="#ff0000" x1="840" y1="945.375" x2="840" y2="925.375"
oid="386" />
<scadiweb-var objtype="Var" name="vmha8" vname="NVAL,@8.ha" vartype="5"
oid="387" id="vmha8" />
<scadiweb-dcoord objtype="Dcoord" dyn="dmove_h" refoid="#vmha8" fmin="0"
fmax="910" groid="#mha8" oid="388" />
<scadiweb-dmisc objtype="Dmisc" dyn="dvisible" dt="t" ntt="2" fmin="-1"
fmax="1" groid="#mha8" refoid="#vmha8" oid="389">
<scadiweb-thr ind="1" tr="0" idat="0" />
<scadiweb-thr ind="2" tr="16384" idat="1" />
</scadiweb-dmisc>
```

Данный пример демонстрирует перемещение вертикальных линий, отмечающих пределы на гистограмме. Также применяется функция нормализации значения NVAL, но к параметру, означающему верхний аварийный предел NVAL,@8.ha. Если этот предел имеется, то нормализованное значение будет больше nval>0, и графический объект

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

(вертикальная линия $mha8$) сдвинется на расстояние, определяемой $fmax*nval$. Если верхний аварийный предел отсутствует, то линия не сдвигается. Кроме этого, предусмотрена специальная динамика видимости, скрывающая линию при отсутствии предела.

4.3 ОПИСАНИЯ ШАБЛОНОВ КОМАНД

Взаимодействие пользователя с системой через браузер меняет правила интерактивного вызова информационных кадров. Вместо общего меню и контекстного меню используются веб-страницы с возможностью вызова команд нажатием кнопки манипулятора. Вместо мыши может использоваться тач-скрин или трекбол. Вызываемая команда может определяет видеокادر, тип контейнера, параметры группового отображения. Описания шаблонов команд задаются в файлах $NMI_*.ру$, используемых в скриптах генерации данных.

Пример файла шаблона команд приведен в приложении 8.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

5 ДИНАМИЧЕСКИЕ ДАННЫЕ

5.1 ВЕБ-СОКЕТ СООБЩЕНИЯ ФОРМАТА JSON

ПО СКАДИВЕБ загружает статические данные через веб-сервер по протоколу http/https.

Динамические данные СКАДИВЕБ запрашиваются, устанавливаются по подписке и принимаются браузером по протоколу веб-сокета wss. Все динамические данные представляют собой сообщения формата JSON. Перечень сообщений представлен в приложении 9.

Запрос на установку подписки изменения текущих данных видеокладов может осуществляться следующим примером. Запрос представляет собой структуру clientMessage с заполненными полями.

```
sub.publish({"sframe": "frml", "input": "AI:1AI001343.av AI:1AI001343.units
AI:1AI001343.descr SVALX,AI:~40YDR10CF001_XQ01.av AI:~40YDR10CF001_XQ01.units
AI:~40YDR10CF001_XQ01.descr SVALX,AI:~40YDR10CP001_XQ01.av
NVAL,AI:~40YDR10CP001_XQ01.av AI:~40YDR10CP001_XQ01.units
AI:~40YDR10CP001_XQ01.descr SVALX,DI:~40YDR10CL003_XH03.dv
DI:~40YDR10CL003_XH03.descr"})
```

Для фрейма frml устанавливается набор переменных подписки input. По установленному набору передаются сообщения об изменениях значений параметров от предыдущих величин. Для данного набора сообщение dataMessage выглядит следующим образом.

```
Message:
{"sframe": "frml", "nvars": 4, "vars": [{"vname": "AI:1AI001343.av", "val": "0.498;48
;1704975166", "atype": "R4QTS"}, {"vname": "SVALX,AI:~40YDR10CF001_XQ01.av", "val":
"0.4904", "atype": "R4"}, {"vname": "SVALX,AI:~40YDR10CP001_XQ01.av", "val": "62.0
9", "atype": "R4"}, {"vname": "NVAL,AI:~40YDR10CP001_XQ01.av", "val": "0.24834582",
"atype": "R4"}]}
```

Запрос на данные группового отображения осуществляется заполнением дополнительно поля inips структуры clientMessage. Пример ниже иллюстрирует запрос на данные группового отображения 20-ти параметров из файла CHANGING.

```
sub.publish({"sframe": "frml", "input": "SPARAM0,CSet:@a.dispcmd @a.fp @a.cp
@a.np DO_BOOL,Host:@H.current_time;r NVAL,@1._ NVAL,@2._ QUALITY,@1._
QUALITY,@2._ VFX,@1.lf VFX,@1.hf VFX,@2.lf VFX,@2.hf QUALITY_TEXT,@1._
TS_TIME,@1._ @3.code", "fragname": "GBV", "inips": "-file CHANGING -pagesize
20", "cp": 1})
```

Для фрейма frml устанавливается набор переменных подписки input. Также устанавливается значение группового запроса inips. Переменные с шаблонами @ в именах динамически ассоциируются с тэгами в группе, причем @1 соответствует параметрам 1 тэга, 2 – второго, и т.д. Сообщение dataMessage для группового набора выглядит следующим образом.

```
Message:
{"sframe": "frml", "nvars": 3, "vars": [{"vname": "NVAL,@1._", "val": "0.30716908", "a
type": "R4"}, {"vname": "NVAL,@2._", "val": "0.31126034", "atype": "R4"}, {"vname": "T
S_TIME,@1._", "val": "11/01/2024 15:29:31", "atype": "STR"}]}
```

Команда остановки обновлений freeze реализуется как клиентское сообщение RpcMessage.

```
centrifuge.rpc("RunStatus", {"sframe": "frml", "rstatus": "freeze"})
```

В ответ сервером посылается сообщение runStatusMessage об остановке.

Run Status is:

```
{"data": {"sframe": "frml", "rmethod": "RunStatus", "result": "freeze"}}
```

Команда старта обновлений run реализуется как клиентское сообщение RpcMessage

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```
centrifuge.rpc("RunStatus", {"sframe": "frml", "rstatus": "run"})
```

В ответ сервером посылается сообщение `rpcStatusMessage` о старте.

Run Status is:

```
{"data":{"sframe":"frml","rmethod":"RunStatus","result":"run"}}
```

Команда перевода страницы в подписке также реализуется как клиентское сообщение `RpcMessage`.

```
centrifuge.rpc("NextPage", {"sframe": "frml", "pageset": 2})
```

В ответ сервером посылается сообщение `rpcPagesMessage` об изменившихся страницах.

Next Page is:

```
{"data":{"sframe":"frml","rmethod":"NextPage","result":"2","pages":{"cp":1,"fp":1,"np":2}}}
```

Команда получения списка файлов общего доступа также реализуется как клиентское сообщение `RpcMessage`.

```
centrifuge.rpc("GetVFiles", {"sframe": "frml"})
```

В ответ сервером посылается сообщение `rpcVFilesMessage` со списком файлов.

VFiles are:

```
{"data":{"sframe":"frml","rmethod":"GetVFiles","result":"vpgg","vfiles":[{"fname":"BINDATA","descr":"Дискретные сигналы","nvars":20}, {"fname":"CHANGING","descr":"Меняющиеся сигналы","nvars":40}, {"fname":"STABLE","descr":"Мало меняющиеся сигналы","nvars":9}, {"fname":"TSTGRP","descr":"Тестовая серверная группа","nvars":2}, {"fname":"SG1REGIM","descr":"НМИ","nvars":4}]}}
```

Как показано, данный интерфейс сообщений является общим для всех видов серверных команд и запросов.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

6 ИСПОЛЬЗУЕМЫЕ СРЕДСТВА

ПРОГРАММНО-ТЕХНИЧЕСКИЕ

Требование к техническим средствам минимальной конфигурации:

- процессор X64 не менее 1ГГц;
- оперативная память не менее 1Гбайт;
- место на диске не менее 1Гбайт;
- возможность установки ОС Astra Linux (не менее 1.4), Linux (ядро не менее 4.0), Windows (не менее Windows 7).

Для доступа к локальным сетям через WiFi необходимы программно-технические устройства роутеров.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

7 ВЫЗОВ И ЗАГРУЗКА

Серверное ПО конфигурируется на хостах распределенной системы.

На клиентской стороне первоначально осуществляется загрузка в браузер.

Загруженная страница осуществляет установку клиентского соединения через веб-сокеты. Структура веб-контента вызываемого ПО является полностью проектно-зависимой, при этом общие механизмы доступа используются для работы.

Для веб-сокета доступа к данным необходима ПО centrifuge. Код доступа осуществляется средствами JavaScript.

```
<script type="text/javascript" src="web/centrifuge.js"> </script>
```

Корневой элемент доступа представляет собой объект типа Centrifuge.

```
const centrifuge = new Centrifuge('ws://' + location.host +
'/connection/websocket', {
  data: {"user-agent": navigator.userAgent}
});
```

Функции доступа реализуются от корневого элемента и привязываются к одноименным событиям.

```
centrifuge.on('connected', function(ctx){
  drawText('Connected with client ID ' + ctx.client + ' over ' +
ctx.transport + ' with data: ' + JSON.stringify(ctx.data));
  input.removeAttribute('disabled');
});
```

```
centrifuge.on('message', function(ctx) {
  drawText('Message: ' + JSON.stringify(ctx.data));
});
```

```
centrifuge.on('publication', function(ctx) {
  drawText('Server-side publication from channel ' + ctx.channel + ": " +
JSON.stringify(ctx.data));
});
```

Пример клиентского веб-сокета доступа приведен в приложении 10.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ 1 – ПЕРЕЧЕНЬ ВСТРОЕННЫХ ФУНКЦИЙ ПРЕОБРАЗОВАНИЯ ДАННЫХ

Встроенные функции преобразования размещены в пакете vsonv. Пакет vsonv входит в состав веб/веб-сокета сервисов доступа к данным. Перечень функций преобразования данных пакета vsonv приведен в таблице 1-1.

Имя	Описание	Тип значения	Примечание
VAL	Возврат значения	Тип без QTS	
RAWVAL	Равен входным данным	Тип с QTS	
QUALITY	Код качества	I2	
QUALITY_TEXT	Текст кода качества	строка	
SVALX	Строковое значение	Тип без QTS	
VFX	Строковое значение	Тип без QTS	
NVAL	Нормализованное значение	Тип без QTS	-1 для несуществующих
VAL_TIME_STR	Значение как время в строку	строка	hh:mm:ss
VAL_TIME_STR10	Метка времени в строку	строка	hh:mm:ss
VAL_DATE_STR	Значение как дата в строку	строка	DD-MM-YY
VAL_DATE_STR1	Значение как дата в строку	строка	DD-MM-YYYY
VAL_DATE_STR11	Метка времени в строку	строка	DD-MM-YYYY
TS_TIME	Метка времени в строку	строка	dd-mm-yyyy hh:mm:ss
ACTIVITY4	Движение времени	I4	2 фазы
ACTIVITYEXT12	Движение метки времени	I4	2 фазы
ACTIVITYEXT14	Движение метки времени	I4	4 фазы
DO_INT	Целочисленные состояния клиента	I4	clientData доп инфо
DO_BOOL	Булевские состояния клиента	BOOL	clientData доп инфо
STATCLR	Цвет статуса	I4	RGB
SPARAM0	Выбор слова по номеру	строка	
PROJECT1	Имя проекта	строка	

Таблица 1-1. Перечень встроенных функций преобразования данных

ПРИЛОЖЕНИЕ 2 – СПИСОК ТИПОВ ГЛОБАЛОВ ОБЩЕЙ ПАМЯТИ

Список типов глобалов общей памяти СКАДИ приведен в таблице 2-1.

Тип	Описание	Мета/Данные	Структура
G	Список глобалов	Мета	КонстТаблица
E	Список имен констант	Мета	КонстТаблица
T	Типы объектов	Мета	КонстТаблица
A	Типы атрибутов объектов	Мета	КонстТаблица
S	Лог системных сообщений	Сообщения	ЦиклБуфер
s	Системные реляционные таблицы	Данные	Таблица
n	Пользовательские реляционные таблицы	Данные	Таблица
i	Индексы реляционных таблиц	Индексы	Таблица
q	Очереди сообщений	Очереди	ЦиклБуфер
a	Архивные данные	Записи	ЦиклБуфер
c	Заголовочные записи архива	Заголовки	ЦиклБуфер

Таблица 2-1. Список типов глобалов общей памяти

ПРИЛОЖЕНИЕ 3 – ВСТРОЕННЫЕ ТИПЫ ДАННЫХ

Перечень встроенных типов данных приведен в таблице 3-1.

№	Имя	Описание	Определение	Байты
0		Пустой тип		
1	I1	Целый 1 байт	int8	1
2	I2	Целый 2 байта	int16	2
3	I4	Целый 4 байта	int32	4
4	I8	Целый 8 байт	int64	8
5	R4	Действительный 4 байта	float32	4
6	R8	Действительный 8 байт	float64	8
7	C8	Командная структура	<pre>type C8B struct { Content, Command_status, Program_id, Ugroup_id int8 Ival int32 }</pre>	8
8	E16	Структура сигнализации	<pre>type E16B struct { Content, Dst_mask, Sev_group, Alm_type int8 Evcs, Prevcs, Condition, Pre_condition int8 U int64 }</pre>	16
9	BOOL	Булевский	bool	1
10	STR	Строка (размер кратен 16)	type String16 [16]byte	16
11	VREF	Ссылка на объект	<pre>type VariableRef struct { Atype, Otype int16 Dim int16 Index int16 Oid, Attrid int32 }</pre>	16
12	CHAR	Символ	byte	1
13	I2QTS	Целый 2 с качеством и меткой времени	<pre>type I2QTS struct { I2val int16 d3, d4, d5, d6 byte Quality Qual Timestamp Time }</pre>	16
14	I4QTS	Целый 4 с качеством и меткой времени	<pre>type I4QTS struct { I4val int32 d5, d6 byte Quality Qual Timestamp Time }</pre>	16
15	R4QTS	Действительный 4 с качеством и меткой времени	<pre>type R4QTS struct { R4val float32 d5, d6 byte Quality Qual Timestamp Time }</pre>	16
16	C8TS	Командная структура с качеством и меткой времени	<pre>type C8TS struct { C8val C8B Timestamp Time }</pre>	16

Таблица 3-1. Перечень встроенных типов

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Встроенные типы с номерами 1-6, 9, 12 являются атомарными. Встроенные типы 7-8, 10-11, 13-16 представляют собой структуры.

Встроенные типы используются соответствуют типам данных, используемых в СКАДИ.

ПРИЛОЖЕНИЕ 4 – БАЗОВЫЕ ЧАСТИ СТРУКТУРНЫХ ТИПОВ

Параметр в определении типа `var_t` определяет, к какой группе типов относится данный. Значение `var_t=1` означает встроенный тип.

Значение `var_t=8` означает наследование только от структуры `OGeneric`. `OGeneric` является базовым для всех наследуемых типов.

```
type OGeneric struct {
    Oid int32
    Ncadr int32
    Name STag
    Proflags, Alloflags int8
    Host_local int8
    Fsm_state int8
    Dst_mask int32
}
```

Все системные типы вначале включают поля `OGeneric`, а затем специфические для определенного типа поля. Перечень полей `OGeneric` приведен в таблице 4-1.

Поле	Наименование	Значения	Размер, байт
Oid	Идентификатор объекта	<Число>	4
Ncadr	Номер кадра	<Число>	4
Name	Имя объекта	<Имя>	32
Proflags	Флаги обработки	<Число>	1
Alloflags	Флаги аллокирования	<Число>	1
Host_local	Номер хоста для локальных данных	<Число>	0 означает отсутствие атрибута
Fsm_state	Номер состояния объекта	<Число>	1
Dst_mask	Адресат сигнализации	<Число>	4

Таблица 4-1. Поля типа `OGeneric`

Расширяемый пользовательский тип `var_t = 13` наследуется от `OConfigurable`. Расширение `OConfigurable` является структурой, наследуемой от `OGeneric` и добавляющей дополнительные поля для индексации и версионности.

```
type OConfigurable struct {
    OGeneric
    Revnum int32
    Revres int32
    Revdate Time
    Code, Pin STag
}
```

Все пользовательские типы глобалов `n`, не являющиеся переменными процесса, имеют `var_t = 13` и наследуется от `OConfigurable`. Перечень полей `OConfigurable` приведен в таблице 4-2.

Поле	Наименование	Значения	Размер, байт
Revnum	Номер версии	<Число>	4
Revres	Результат ревизии	<Число>	4
Revdate	Дата модификации	<Число>	8
Code	Технологический код в форматах KKS, PTM, ...	<Имя>	32
Pin	Контакт или адрес устройства связи с объектом в формате, определяемом этим устройством	BOOL	32

Таблица 4-2. Поля типа `OConfigurable`

Объекты с `var = 13` могут описывать какие-то функциональные блоки (регуляторы, блоки ФГУ), не являющиеся переменными процесса.

Управляемые пользовательские типы с `var = 23` наследуются от `OManagable`. Расширение `OManagable` является структурой, наследуемой от `OConfigurable` и добавляющей дополнительные поля, характеризующие запись как переменную процесса.

```

type OManagable struct {
    OConfigurable
    Descr SDescr
    Xarea_id, Fun_type int16
    Almpri int16
    Chatter bool
    Std_server int8
    Block, Ablock VariableRef
    Evcu E16B
    Imcmd C8B
}

```

Перечень полей `OManagable` приведен в таблице 4-3.

Поле	Наименование	Значения	Размер, байт
Descr	Описание параметра	<Строка>	80
Xarea_id	Номер области агента обмена	<Число>	2
Fun_type	Тип функции обработчика	<Число>	2
Almpri	Флаги приоритета сигнализации	<Множество>	2
Chatter	Дребезг	BOOL	1
Std_server	Номер стандартного сервера	<Число>	1
Block	Ссылка на объект	<Ссылка>	16
Ablock	Альтернативная ссылка	<Ссылка>	16
Evcu	Значение сигнализации	<Сигнализация>	16
Imcmd	Имитируемая команда		8

Таблица 4-3. Поля типа `OConfigurable`

Полям типов глобала `n` соответствуют структурные типы глобала `n` с `var=13` или `23`.

ПРИЛОЖЕНИЕ 5 - ДАННЫЕ ПОЛЬЗОВАТЕЛЬСКИХ РЕЛЯЦИОННЫХ ТАБЛИЦ

Тип AI предназначен для обработки аналоговых сигналов. Объекты AI представляют собой переменные процесса и наследуются от OGeneric, OConfigurable, OManagable. Объекты включают в себя поля OGeneric, OConfigurable, OManagable и поля, относящиеся к непосредственно AI. Структура AI – это запись следующего вида, наследуемая от OManagable.

```

type AI struct {
    OManagable
    Units String16
    Lf, Hf, La, Ha, Lw, Hw, Lr, Hr float32
    Ainp, Av R4QTS
    Av_sim, Aper, Zone float32
    Ndigits int32
}

```

Список полей структуры приведен в таблице 5-1.

Поле	Наименование	Значения	Примечание
Units	Единицы измерения	<Строка>	
Ff	Нижний предел	<Предел>	Значение или ~ если не задано
Hf	Верхний предел	<Предел>	Аналогично
La	Нижняя аварийная граница	<Предел>	Аналогично
Ha	Верхняя аварийная граница	<Предел>	Аналогично
Lw	Нижняя предупредительная граница	<Предел>	Аналогично
Hw	Верхняя предупредительная граница	<Предел>	Аналогично
Lr	Нижняя регламентная граница	<Предел>	Аналогично
Hr	Верхняя регламентная граница	<Предел>	Аналогично
Ainp	Входное аналоговое значение	<Значение, Качество, МеткаВремени>	
Av	Текущее аналоговое значение	<Значение, Качество, МеткаВремени>	
Av_sim	Имитируемое аналоговое значение	<Значение>	
Aper	Апертура архивации	<Значение>	
Zone	Зона возврата гистерезиса	<Значение>	
Ndigits	Число значащих цифр	<Число>	

Таблица 5-1. Список полей структуры AI

Значения AI представляют собой поля объектов, определяющих аналоговые переменные. Переменные имеют значение, входное значение, а также диапазон и границы сигнализации. Доступ к значениям структуры AI осуществляется по именам, заданным в поле name структуры OGeneric.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВБЕБ - Описание программы	Версия 1 01.12.2023
-------------	--	------------------------

Тип DI предназначен для обработки дискретных сигналов. Объекты DI представляют собой переменные процесса и наследуются от OGeneric, OConfigurable, OManagable. Объекты включают в себя поля OGeneric, OConfigurable, OManagable и поля, относящиеся к непосредственно DI. Структура DI – это запись следующего вида, наследуемая от OManagable.

```
type DI struct {
    OManagable
    Dinp, Dv I2QTS
    Dv_sim int8
    Dcmd C8TS
    Dtype byte
    Inver bool
}
```

Список полей структуры приведен в таблице 5-2.

Поле	Наименование	Значения	Примечание
Dinp	Входное дискретное значение	<Число1, Качество, МеткаВремени>	Число1 размером 1 байт, булевские и целые значения
Dv	Текущее дискретное значение	<Число1, Качество, МеткаВремени>	Аналогично
Dv_sim	Имитируемое дискретное значение	<Число1>	Аналогично
Dv_cmd	Команда управления	Структура C8	
Dtype	Тип дискрета	<Символ>	
Inver	Прямой или инверсный	BOOL	

Таблица 5-2. Список полей структуры DI

Значения DI представляют собой поля объектов, определяющих дискретные переменные. Переменные имеют значение, входное значение, а также параметры типизации. Доступ к значениям структуры DI осуществляется по именам, заданным в поле name структуры OGeneric.

Тип AC предназначен для обработки аналоговых вычислений. Объекты AC представляют собой вычисляемые переменные и наследуются от OGeneric, OConfigurable, OManagable. Объекты включают в себя поля OGeneric, OConfigurable, OManagable и поля, относящиеся к непосредственно AC. Структура AC – это запись следующего вида, наследуемая от OManagable.

```
type AC struct {
    OManagable
    Units String16
    Lf, Hf, La, Ha, Lw, Hw, Lr, Hr float32
    Acv R4QTS
    Aper float32
    Ndigits int32
    Arg1, Arg2, Arg3, Arg4 VariableRef
    Sexpr, Cexpr String256
}
```

Список полей структуры приведен в таблице 5-3.

Поле	Наименование	Значения	Примечание
Units	Единицы измерения	<Строка>	
Ff	Нижний предел	<Предел>	Значение или ~ если не задано

Hf	Верхний предел	<Предел>	Аналогично
La	Нижняя аварийная граница	<Предел>	Аналогично
Ha	Верхняя аварийная граница	<Предел>	Аналогично
Lw	Нижняя предупредительная граница	<Предел>	Аналогично
Hw	Верхняя предупредительная граница	<Предел>	Аналогично
Lr	Нижняя регламентная граница	<Предел>	Аналогично
Hr	Верхняя регламентная граница	<Предел>	Аналогично
Acv	Вычисляемое аналоговое значение	<Значение, Качество, МеткаВремени>	
Aper	Апертура архивации	<Значение>	
Ndigits	Число значащих цифр	<Число>	
Arg1	Ссылка на вход1	<ссылка>	
Arg2	Ссылка на вход2	<ссылка>	
Arg3	Ссылка на вход3	<ссылка>	
Arg4	Ссылка на вход4	<ссылка>	
Sexpr	Строковое выражение	<Строка>	
Cexpr	Строка компиляции	<Строка>	Компилированное Sexpr

Таблица 5-3. Список полей структуры AC

Значения AC представляют собой поля объектов, определяющих аналоговые переменные. Переменные имеют значение, входное значение, а также диапазон и границы сигнализации. Доступ к значениям структуры AC осуществляется по именам, заданным в поле name структуры OGeneric.

Тип DC предназначен для обработки дискретных вычислений. Объекты DC представляют собой вычисляемые переменные и наследуются от OGeneric, OConfigurable, OManagable. Объекты включают в себя поля OGeneric, OConfigurable, OManagable и поля, относящиеся к непосредственно DC. Структура DC – это запись следующего вида, наследуемая от OManagable.

```

type DC struct {
    OManagable
    Dcv I2QTS
    Dtype byte
    Arg1, Arg2, Arg3, Arg4 VariableRef
    Sexpr, Cexpr String256
}

```

Список полей структуры приведен в таблице 5-4.

Поле	Наименование	Значения	Примечание
Dcv	Вычисляемое дискретное значение	<Число1, Качество, МеткаВремени>	Аналогично
Dtype	Тип дискрета	<Символ>	
Arg1	Ссылка на вход1	<ссылка>	
Arg2	Ссылка на вход2	<ссылка>	
Arg3	Ссылка на вход3	<ссылка>	

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

Arg4	Ссылка на вход4	<ссылка>	
Sexpr	Строковое выражение	<Строка>	
Sexpr	Строка компиляции	<Строка>	Компилированное Sexpr

Таблица 5-4. Список полей структуры DC

Значения DC представляют собой поля объектов, определяющих дискретные переменные. Переменные имеют значение, входное значение, а также параметры типизации. Доступ к значениям структуры DC осуществляется по именам, заданным в поле name структуры OGeneric.

Тип NI предназначен для обработки целочисленных сигналов. Объекты NI представляют собой переменные процесса и наследуются от OGeneric, OConfigurable, OManagable. Объекты включают в себя поля OGeneric, OConfigurable, OManagable и поля, относящиеся к непосредственно NI. Структура NI – это запись следующего вида, наследуемая от OManagable.

```

type NI struct {
    OManagable
    Ninp, Nv I4QTS
    Fun_kind, Fun_ind int32
}

```

Список полей структуры приведен в таблице 5-5.

Поле	Наименование	Значения	Примечание
Ninp	Входное целочисленное значение	<Число, Качество, МеткаВремени>	Число размером 4 байт содержит целые значения
Nv	Текущее целочисленное значение	<Число, Качество, МеткаВремени>	Аналогично
Fun_kind	Подтип функции обработки	<Число>	
Fun_ind	Индекс элемента обработки	<Число>	

Таблица 5-5. Список полей структуры NI

Значения NI представляют собой поля объектов, определяющих аналоговые переменные. Переменные имеют текущее и входное значения. Доступ к значениям структуры NI осуществляется по именам, заданным в поле name структуры OGeneric.

ПРИЛОЖЕНИЕ 6 – СТРУКТУРЫ ЦИКЛИЧЕСКИХ БУФЕРОВ

Тип `objs.QDataSeq` представляет собой следующую структуру, за которой следует `n_datas` объектов типа `objs.QData`, которая, в свою очередь, представляет собой структуру:

```

type QDataSeq struct {
    Atype int16
    Srvstatus int16
    Source_id, Subscr_id, Flags, Ncadr int32
    N_datas int32
    Clock, Srvclock, Oldclock Time
}
type QData struct {
    Ref VariableRef
    Val ValueData
    Oldval ValueData
}

```

Список полей структур `objs.QData` приведен в таблице 6-1.

Поле	Наименование	Значения	Примечание
Ref	Ссылка на объект	<Ссылка>	Стандартный тип ссылки VREF
Val	Текущее значение	<Значение>	Универсальный контейнер значений
Oldval	Предыдущее значение	<Значение>	Аналогично

Таблица 6-1. Список полей структур `objs.QData`

Список полей структур `objs.QDataSeq` приведен в таблице 6-2.

Поле	Наименование	Значения	Примечание
N_datas	Число данных последовательности	<Число>	
Atype	Тип атрибута	<Тип>	
Srvstatus	Статус сервера	<Число>	Enum SRV_STAT
Source_id	Идентификатор источника данных	<Число>	
Subscr_id	Идентификатор подписки	<Число>	
Ncadr	Номер кадра	<Число>	
Flags	Флаги передачи	<Число>	Множество
Clock	Текущее время	<Время>	
Srvclock	Время сервера	<Время>	
Oldclock	Предыдущее время	<Время>	

Таблица 6-2. Список полей структур `objs.QDataSeq`

Флаги передачи данных являются целочисленным представлением множества, каждый элемент которого есть определенный флаг, согласно таблице 6-3.

№	Имя	Описание
1	SEQF_TRIM	Неполные данные
2	SEQF_FIRST	Первый элемент последовательности

3	SEQF_GAP	Пропуск данных
4	SEQF_HEAD	Данные заголовков
5	SEQF_REFRESH	Посылка уже отправлявшихся данных
6	SEQF_RESEND	Повторная пересылка данных
7	SEQF_UPDATED	Обновление данных
8	SEQF_VAR	Значения переменных процесса
9	SEQF_PARM	Значения параметров процесса
10	SEQF_LOCAL	Значения локальных величин
11	SEQF_CMD	Значения команд управления
12	SEQF_TEST	Тестовый режим
13	SEQF_CONTINUED	Продолжение данных в последовательности
21	SEQF_GROUP	Данные группы
22	SEQF_XCH	Данные агентов обмена
23	SEQF_CRC_REF	Наличие контрольных сумм
24	SEQF_AREC	Данные в формате архивной записи
25	SEQF_ECHO	Эхо последовательность принятой команды
26	SEQF_NOECHO	Эхо последовательность отвергнутой команды
27	SEQF_ATINY	Минимальная архивная запись 8 байт
28	SEQF_ASMALL	Малая архивная запись 16 байт
29	SEQF_AMIDDLE	Средняя архивная запись 24 байт
30	SEQF_AFULL	Полная архивная запись 48 байт

Таблица 6-3. Таблица флагов передачи данных

Данные очередей сообщений и передачи по сети группируются в последовательности с источником и временной меткой. Во флаги передачи данных записываются дополнительная информация о характере описываемой последовательности данных. Эта информация содержит флаги, из которых следует, что передаваемая последовательность содержит переменные или параметры или команды управления, и т.д.

ПРИЛОЖЕНИЕ 7 – ТАБЛИЦЫ ДАННЫХ ДИНАМИЧЕСКИХ ОБЪЕКТОВ

Таблицы данных используются для задания правил динамики объектов в табличном виде. Табличные данные задаются с помощью общего для динамик набора параметров и массива параметров scadiweb-thr. Общий набор параметров для динамик приведен в таблице 7-1.

Имя	Описание	Значения	Примечания
dt	Тип таблицы	i-Index t-Table s-Set r-RGB b-Blink	Задаёт алгоритм отображения динамики по таблице или без
ntt	Число элементов в массиве tt	>0	
imin	Нижняя целочисленная граница	I4	imin<imax
imax	Верхняя целочисленная граница	I4	
fmin	Нижняя действительная граница	R4	fmin<fmax
fmax	Верхняя действительная граница	R4	

Таблица 7-1. Параметры таблиц динамик

Массив параметров tt динамических данных имеет ntt элементов, каждый со своим набором параметров. Набор параметров элемента массива tt приведен в таблице 7-2.

Имя	Описание	Значение	Примечания
ind	Индекс элемента таблицы		
tr	Правый предел сравнения	I4	0-32768 для Table Битовая маска для Set
idat	Целочисленное значение	I4	Возвращаемое целое значение
fdat	Действительное значение	R4	Возвращаемое действительное значение

Таблица 7-2. Параметры таблиц динамик

Для индексного Index типа –dt i алгоритм отображения динамики следующий. Входное значение val должно быть целочисленного типа I1, I2, I4, BOOL, I2QTS, I4QTS. Если значение val < imin – ошибка. Если значение val > imin+ntt, то значение индекса ограничивается ind := imin+ntt, иначе ind := val. Возвращаемое значение будет равно tt[ind].idat.

Для табличного Table типа –dt t алгоритм отображения динамики следующий. Должны быть заданы границы fmin и fmax, причем fmin<fmax. Входное значение val должно быть действительного R4, R4QTS или целочисленного типа I1, I2, I4, BOOL, I2QTS. Входное значение val, изменяемое в пределах (fmin, fmax), нормируется на диапазон (0,32768). Формула преобразования равна $v := (val - fmin) / (fmax - fmin) * 32768$. Таким образом, изменению значения val в диапазоне (fmin, fmax) соответствует изменение значения v в диапазоне (0,32768). При этом значению val, меньшему fmin, соответствует отрицательное значение v. А значению val, большему fmax, соответствует значение v, превышающее 32768. При этом таблица tt разбивает весь диапазон (0,32768) на области. Примером такой разбивки служит цветовая динамика, представленная на рисунке 7-1.

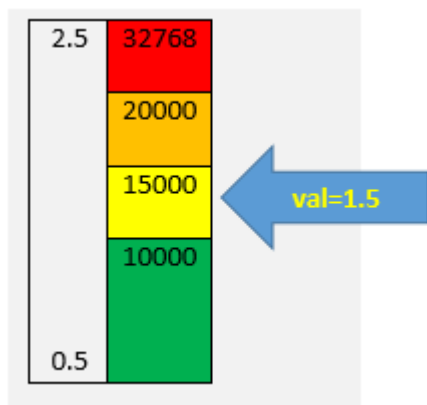


Рисунок 7-1 – Табличное задание цветовой динамики

Для приведенной таблицы имеем

```
<scadiweb-dmisc objtype="Dmisc" dyn="dbackground" refoid="#vaa" groid="#aa"
dt="t" ntt="4" fmin="0.5" fmax="2.5">
  <scadiweb-thr ind="1" tr="10000" idat="0x00ff00" />
  <scadiweb-thr ind="2" tr="15000" idat="0xffff00" />
  <scadiweb-thr ind="3" tr="20000" idat="0xff8000" />
  <scadiweb-thr ind="4" tr="32768" idat="0xff0000" />
</scadiweb-dmisc>
```

Алгоритм выборки индекса по таблице следующий. При движении снизу вверх, если $v < thr[ind].tr$, то это значение выбирается. В приведенном примере это индекс 2 – желтый с цветом 0xffff00 (scadiweb-thr ind="2" tr="15000" idat="0xffff00"). Выход за диапазон приравнивается к первому и последнему индексам соответственно.

Для множественного Set типа –dt s алгоритм отображения динамики следующий. Входное значение val должно быть целочисленного типа I1, I2, I4, BOOL, I2QTS, I4QTS. Табличные значения tr задают тестируемые биты в val, при этом значению по умолчанию соответствует значение tr = -1. При совпадении тестируемого бита с пришедшем значением выбирается индекс тестируемого бита. Приведенный пример показывает цветовую динамику черным цветом при наличии бита 2 и серым 0xa0a0a0 в противном случае.

```
<scadiweb-dmisc objtype="Dmisc" dyn="dforeground" refoid="#r" groid="#g" dt="s"
ntt="4" fmin="0.5" fmax="2.5">
  <scadiweb-thr ind="1" tr="2" idat="1" />
  <scadiweb-thr ind="2" tr="-1" idat="0xa0a0a0" />
</scadiweb-dmisc>
```

Для цветового RGB типа –dt r алгоритм отображения динамики следующий. При этом в объектах Var могут использоваться функции обработки QUALITY, QUALITY_COLOR.

```
<scadiweb-var objtype="Var" name="var" vname="QUALITY_COLOR,@1.av" vartype="3"
/>
<scadiweb-dmisc objtype="Dmisc" dyn="dforeground" refoid="#var" groid="#gvar"
dt="r"/>
```

В приведенном примере по значению кода качества переменной av вычисляется цвет, который используется в динамике.

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

ПРИЛОЖЕНИЕ 8 – ФАЙЛ ШАБЛОНА КОМАНД

Приведенный файл шаблона команд определяет следующие таблицы:

- FRM – фреймы экранной области;
- CONTAINER – типы контейнеров;
- MENU – Логические группы команд;
- ITEM – описание команды;
- Начальные размеры экрана WIDTH, HEIGHT

```

WIDTH=1920
HEIGHT=1040
FRM = dict()
# FRM[1].s=2 -initial_tvс SCADIO -x 100 -y 30 -w 1820 -h 1010
FRM[1]=[2, "SCADIO", 100, 30, 1820, 1010]
FRM[2]=[1, "UBACKGROUND", 0, 0, -1, 30]
FRM[3]=[3, "LBACKGROUND", 0, 30, 100, 1010]
MENU = dict()
ITEM = dict()
CONTAINER = dict()
CONTAINER["trend"] = [1200, 750, "atrendwrap"]
# + Кнопки вперед-назад страница
CONTAINER["rland"] = [1600, 1000, "rlandwrap"]
CONTAINER["rport"] = [800, 900, "rportwrap"]
MENU["07"] = "Тренды"
ITEM["07", "01"] = ["График 1 аналогового сигнала", "_TCX1", "-codes '%g' -
pagesize 1 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "02"] = ["График 2 аналоговых сигналов", "_TCX2", "-codes '%g' -
pagesize 2 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "03"] = ["График 3 аналоговых сигналов", "_TCX3", "-codes '%g' -
pagesize 3 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "04"] = ["График 4 аналоговых сигналов", "_TCX4", "-codes '%g' -
pagesize 4 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "10"] = ["_"]
ITEM["07", "12"] = ["Совмещенный график 2 сигналов", "_TTX2", "-codes '%g' -
pagesize 2 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "13"] = ["Совмещенный график 3 сигналов", "_TTX3", "-codes '%g' -
pagesize 3 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "14"] = ["Совмещенный график 4 сигналов", "_TTX4", "-file SG1REGIM
-pagesize 4 -report . -artype STA -globnames . -delta_sec 360", "trend"]
ITEM["07", "16"] = ["Совмещенный график 6 сигналов", "_TTX6", "-codes '%g' -
pagesize 6 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "18"] = ["Совмещенный график 8 сигналов", "_TTX8", "-codes '%g' -
pagesize 8 -report . -artype STA -globnames . -delta_sec 600 -avg_points 500",
"trend"]
ITEM["07", "20"] = ["_"]
ITEM["07", "22"] = ["Мультиграфик 20 сигналов", "_TMX20", "-file CHANGING -
pagesize 20 -report . -artype STA -globnames . -delta_sec 120 -avg_points 500",
"trend"]
ITEM["07", "23"] = ["Мультиграфик 30 сигналов", "_TMX30", "-file CHANGING -
pagesize 30 -report . -artype STA -globnames . -delta_sec 120 -avg_points 500",
"trend"]

```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

ИТЕМ["07", "24"] = ["Мультиграфик 40 сигналов", "_TMX40", "-file CHANGING -
pagesize 40 -report . -artype STA -globnames . -delta_sec 120 -avg_points 500",
"trend"]
МЕНУ["08"] = "Отчеты"
# Актуальных значений
ИТЕМ["08", "01"] = ["Отчет значений по текущей группе Альбом", "_REPLAND", "-
report DprepValsAt -pagesize 42 -artype STA -globnames . -codes '%g' -delta_sec
300", "rland"]
ИТЕМ["08", "02"] = ["Отчет значений по текущей группе Альбом7", "_REPLAND", "-
report DprepValsAt -pagesize 42 -rep_kind 7 -artype STA -globnames . -codes
'%g' -delta_sec 300", "rland"]
ИТЕМ["08", "03"] = ["Отчет значений по текущей группе Альбом8", "_REPLAND", "-
report DprepValsAt -pagesize 42 -rep_kind 8 -artype STA -globnames . -codes
'%g' -delta_sec 300", "rland"]
ИТЕМ["08", "04"] = ["Отчет значений по текущей группе Портрет", "_REPPORT", "-
report DprepValsAt -pagesize 60 -rep_kind 5 -artype STA -globnames . -codes
'%g' -delta_sec 300", "rport"]
#Свободного формата
ИТЕМ["08", "11"] = ["Формата L,P 1 контура", "_REPLAND", "-report DprepFf -
pagesize 42 -artype STA -globnames . -file LP1K.txd -delta_sec 300", "rland"]
ИТЕМ["08", "12"] = ["Формата P,L,T парогенераторов", "_REPLAND", "-report
DprepFf -pagesize 42 -artype STA -globnames . -file PLTSG.txd -delta_sec 300",
"rland"]
#Событий
ИТЕМ["08", "21"] = ["Отчет событий по текущей группе Альбом", "_REPLAND", "-
report DprepArch -pagesize 42 -artype STA -globnames . -codes '%g' -delta_sec
300", "rland"]
ИТЕМ["08", "22"] = ["Отчет событий по текущей группе Альбом7", "_REPLAND", "-
report DprepArch -pagesize 42 -rep_kind 7 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rland"]
ИТЕМ["08", "23"] = ["Отчет событий по текущей группе Альбом8", "_REPLAND", "-
report DprepArch -pagesize 42 -rep_kind 8 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rland"]
ИТЕМ["08", "24"] = ["Отчет событий по текущей группе Портрет", "_REPPORT", "-
report DprepArch -pagesize 60 -rep_kind 5 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rport"]
#Табличный
ИТЕМ["08", "35"] = ["Табличный отчет по текущей группе Портрет5", "_REPPORT",
"-report DprepRvp -pagesize 60 -rep_kind 5 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rport"]
ИТЕМ["08", "36"] = ["Табличный отчет по текущей группе Портрет7", "_REPPORT",
"-report DprepRvp -pagesize 60 -rep_kind 6 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rport"]
ИТЕМ["08", "38"] = ["Табличный отчет по текущей группе Альбом8", "_REPLAND", "-
report DprepRvp -pagesize 42 -rep_kind 8 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rland"]
ИТЕМ["08", "39"] = ["Табличный отчет по текущей группе Альбом9", "_REPLAND", "-
report DprepRvp -pagesize 42 -rep_kind 9 -artype STA -globnames . -codes '%g'
-delta_sec 300", "rland"]

```


ПРИЛОЖЕНИЕ 9 – ПЕРЕЧЕНЬ СООБЩЕНИЙ ФОРМАТА JSON

Сообщение от клиента clientMessage устанавливает подписку для данного фрейма клиента.

```

type clientMessage struct {
    Timestamp int64 `json:"timestamp"`
    SFrame    string `json:"sframe"`
    Input     string `json:"input"`
    FragName  string `json:"fragname"`
    Inips     string `json:"inips"`
    VarNames  string `json:"varnames"`
    VFile     string `json:"vfile"`
    AttrNames string `json:"attrnames"`
}

```

Перечень параметров сообщения приведен в таблице 9-1.

Имя	Описание	Значение	Примечания
timestamp	Метка времени	время	64бит
sframe	Имя фрейма	строка	Уникальное имя
input	Набор переменных подписки	строка	Пробелы как разделители
fragname	Имя фрагмента	строка	
inips	Строка инициализации группового отображения	строка	По спецификации СКАДИ
varnames	Имена переменных	строка	Пробелы как разделители
vfile	Имя файла	строка	Файл из разделяемой памяти
attrnames	Имена атрибутов	строка	Пробелы как разделители

Таблица 9-1. Параметры сообщения clientMessage

Сообщение от клиента RpcMessage запрашивает вызов клиентом удаленной процедуры на стороне сервера.

```

type RpcMessage struct {
    Timestamp int64 `json:"timestamp"`
    SFrame    string `json:"sframe"`
    RStatus   string `json:"rstatus"`
    PageSet   int   `json:"pageset"`
}

```

Перечень параметров сообщения приведен в таблице 9-2.

Имя	Описание	Значение	Примечания
timestamp	Метка времени	время	64бит
sframe	Имя фрейма	строка	Уникальное имя
rstatus	Статус выполнения	строка	run/freeze
pageset	Уставка номера страницы	число	

Таблица 9-2. Параметры сообщения RpcMessage

Сообщение от сервера dataMessage передает клиенту очередную порцию данных по подписке. Сообщение содержит динамический массив структур varMessage по одной структуре на каждую передаваемую переменную.

```

type varMessage struct {
    Vname string `json:"vname"`
    Val    string `json:"val"`
    Atype string `json:"atype"`
}
type dataMessage struct {

```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```
SFrame      string `json:"sframe"`
NVars int   `json:"nvars"`
Vars []varMessage `json:"vars"`
}
```

Перечень параметров сообщений приведен в таблице 9-3.

Имя	Описание	Значение	Примечания
vname	Имя переменной	строка	
val	Значение переменной	строка	
atype	Тип переменной	строка	Поле «Имя» таблицы 3.1
sframe	Имя фрейма	строка	Уникальное имя
nvars	Число переменных	число	
vars	Массив переменных	[]varMessage	Массив varMessage

Таблица 9-3. Параметры сообщения dataMessage

Сообщение от сервера statusMessage передает клиентам телеграммы жизни, включающие состояние и текущее время удаленного сервера.

```
type statusMessage struct {
    TInput      string `json:"input"`
    SrvStatus   string `json:"srvstatus"`
    SrvClock    int64  `json:"srvclock"`
}
```

Перечень параметров сообщения приведен в таблице 9-4.

Имя	Описание	Значение	Примечания
input	Имя телеграммы	строка	Heartbeat
srvstatus	Статус сервера	число	
srvclock	Время сервера	время	64бит Unix

Таблица 9-4. Параметры сообщения statusMessage

Сообщение от сервера rpcStatusMessage является ответом на запрос клиента RpcMessage с методом «RunStatus».

```
type rpcStatusMessage struct {
    SFrame      string `json:"sframe"`
    RMethod     string `json:"rmethod"`
    Result      string `json:"result"`
}
```

Перечень параметров сообщения приведен в таблице 9-5.

Имя	Описание	Значение	Примечания
sframe	Имя фрейма	строка	Уникальное имя
rmethod	Имя метода	строка	RunStatus
result	Результат выполнения	строка	run freeze

Таблица 9-5. Параметры сообщения rpcStatusMessage

Сообщение от сервера rpcPagesMessage является ответом на запрос клиента RpcMessage с методом «NextPage». Сообщение содержит вложенную структуру rpcPages с номерами страниц.

```
type rpcPages struct {
    Cp int32 `json:"cp"`
    Fp int32 `json:"fp"`
    Np int32 `json:"np"`
}
type rpcPagesMessage struct {
    SFrame      string `json:"sframe"`
}
```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

RMethod string `json:"rmethod"`
Result string `json:"result"`
Pages rpcPages `json:"pages"`
}

```

Перечень параметров сообщения приведен в таблице 9-6.

Имя	Описание	Значение	Примечания
ср	Текущая страница	число32	
фр	Первая страница	число32	
нр	Число страниц	число32	
sframe	Имя фрейма	строка	Уникальное имя
rmethod	Имя метода	строка	NextPage
result	Результат выполнения	строка	
pages	Структура страниц	rpcPages	

Таблица 9-6. Параметры сообщения rpcPagesMessage

Сообщение от сервера rpcVFilesMessage является ответом на запрос клиента RpcMessage с методом «GetVFiles». Сообщение содержит динамический массив структур rpcVFile по одной структуре на каждый передаваемый файл.

```

type rpcVFile struct {
    Fname string `json:"fname"`
    Descr string `json:"descr"`
    Nvars int `json:"nvars"`
}
type rpcVFilesMessage struct {
    SFrame string `json:"sframe"`
    RMethod string `json:"rmethod"`
    Result string `json:"result"`
    Vfiles []rpcVFile `json:"vfiles"`
}

```

Перечень параметров сообщения приведен в таблице 9-7.

Имя	Описание	Значение	Примечания
fname	Имя файла	число32	
descr	Описание файла	число32	
nvars	Число переменных в нем	число	
sframe	Имя фрейма	строка	Уникальное имя
rmethod	Имя метода	строка	GetVFiles
result	Результат выполнения	строка	
vfiles	Массив файлов	[]rpcVFile	Массив rpcVFile

Таблица 9-7. Параметры сообщения rpcVFilesMessage

ПРИЛОЖЕНИЕ 10 – ПРИМЕР КЛИЕНТСКОГО ВЕБ-СОКЕТ ДОСТУПА

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title></title>
    <style type="text/css">
      input[type="text"] { width: 300px; }
      .muted {color: #CCCCCC; font-size: 10px;}
    </style>
    <script type="text/javascript" src=" web/centrifuge.js"></script>
    <script type="text/javascript">
      // helper functions to work with escaping html.
      const tagsToReplace = {'&': '&amp;', '<': '&lt;', '>': '&gt;'};
      function replaceTag(tag) {return tagsToReplace[tag] || tag;}
      function safeTagsReplace(str) {return str.replace(/[\<>]/g,
replaceTag);}

      const channel = "chat:index";

      window.addEventListener('load', function() {
        const input = document.getElementById("input");
        const dinput = document.getElementById("dinput");
        const container = document.getElementById('messages');

        const centrifuge = new Centrifuge('ws://' + location.host +
'/connection/websocket', {
          data: {"user-agent": navigator.userAgent}
        });

        centrifuge.on('connecting', function(ctx){
          drawText('Connecting: ' + ctx.reason);
          input.setAttribute('disabled', 'true');
        });

        centrifuge.on('disconnected', function(ctx){
          drawText('Disconnected: ' + ctx.reason);
          input.setAttribute('disabled', 'true');
        });

        // bind listeners on centrifuge object instance events.
        centrifuge.on('connected', function(ctx){
          drawText('Connected with client ID ' + ctx.client + '
over ' + ctx.transport + ' with data: ' + JSON.stringify(ctx.data));
          input.removeAttribute('disabled');
        });

        centrifuge.on('message', function(ctx) {
          drawText('Message: ' + JSON.stringify(ctx.data));
        });

        centrifuge.on('publication', function(ctx) {
          drawText('Server-side publication from channel ' +
ctx.channel + ": " + JSON.stringify(ctx.data));
        });

        centrifuge.on('subscribing', function(ctx) {
```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

        drawText('Subscribing on server-side channel ' +
ctx.channel);
    });

    centrifuge.on('unsubscribed', function(ctx) {
        drawText('Unsubscribe from server-side channel ' +
ctx.channel);
    });

    centrifuge.on('subscribed', function(ctx) {
        drawText('Subscribe to server-side channel ' +
ctx.channel + ': ' + JSON.stringify(ctx));
    });

    // show how many users currently in channel.
    function showPresence(sub) {
        sub.presence().then(function(result) {
            let count = 0;
            for (let key in result.clients){
                count++;
            }
            drawText('Presence: now in this room - ' + count + '
clients');
        }, function(err) {
            drawText("Presence error: " + JSON.stringify(err));
        });
    }

    // subscribe on channel and bind various event listeners.
Actual
    // subscription request will be sent after client connects to
    // a server.
    const sub = centrifuge.newSubscription(channel);

    sub.on("publication", handlePublication)
        .on("join", handleJoin)
        .on("leave", handleLeave)
        .on("unsubscribed", handleUnsubscribed)
        .on("subscribed", handleSubscribed)
        .on("subscribing", handleSubscribing)
        .on("error", handleSubscriptionError);

    sub.subscribe();

    // Trigger actual connection establishing with a server.
    // At this moment actual client work starts - i.e.
subscriptions
    // defined start subscribing etc.
    centrifuge.connect();

    function handleSubscribed(ctx) {
        drawText('Subscribed on channel ' + ctx.channel + ': ' +
JSON.stringify(ctx));
        showPresence(sub);

        //centrifuge.rpc("getCurrentYear",
    {}).then(function(data){
        //    drawText("RPC response data: " +
JSON.stringify(data));

```

34185873.425510.004.ОП.80.М	Стр. 53 из 60
-----------------------------	---------------

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

    //}, function(err) {
    //    drawText("RPC error: " + JSON.stringify(err));
    //});
}

function handleUnsubscribed(ctx) {
    drawText('Unsubscribed from channel ' + ctx.channel + ', '
+ JSON.stringify(ctx));
}

function handleSubscribing(ctx) {
    drawText('Subscribing on channel ' + ctx.channel + ', '
+ JSON.stringify(ctx));
}

function handleSubscriptionError(ctx) {
    drawText('Error subscribing on channel ' +
JSON.stringify(ctx));
}

function handlePublication(message) {
    let clientID;
    if (message.info){
        clientID = message.info.client;
    } else {
        clientID = null;
    }
    const inputText = message.data["input"].toString();
    const text = safeTagsReplace(inputText+'
'+JSON.stringify(message.data)) + ' <span class="muted">from ' + clientID +
'</span>';

    drawText(text);
}

function handleJoin(ctx) {
    drawText('Client joined channel ' + this.channel + ' (uid
' + ctx.info["client"] + ', user ' + ctx.info["user"] + ')');
}

function handleLeave(ctx) {
    drawText('Client left channel ' + this.channel + ' (uid '
+ ctx.info["client"] + ', user ' + ctx.info["user"] + ')');
}

function drawText(text) {
    let e = document.createElement('li');
    e.innerHTML = [(new Date()).toString(), ' ' +
text].join(':');
    container.insertBefore(e, container.firstChild);
}

document.getElementById('form').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frml", "input":
input.value}).then(function() {
        drawText("Successfully published AI to channel");
    }, function(err) {
        drawText("Publish error: " + JSON.stringify(err));
    });
});

```

34185873.425510.004.ОП.80.M	Стр. 54 из 60
-----------------------------	---------------

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

    });
    input.value = '';
  });
  document.getElementById('dform').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frm2", "input":
dinput.value}).then(function() {
      drawText("Successfully published DI to channel");
    }, function(err) {
      drawText("Publish error: " + JSON.stringify(err));
    });
    dinput.value = '';
  });
  document.getElementById('yd10').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frm1", "input": "AI:1AI001343.av
AI:1AI001343.units AI:1AI001343.descr SVALX,AI:~40YDR10CF001_XQ01.av
AI:~40YDR10CF001_XQ01.units AI:~40YDR10CF001_XQ01.descr
SVALX,AI:~40YDR10CP001_XQ01.av NVAL,AI:~40YDR10CP001_XQ01.av
AI:~40YDR10CP001_XQ01.units AI:~40YDR10CP001_XQ01.descr
SVALX,DI:~40YDR10CL003_XH03.dv DI:~40YDR10CL003_XH03.descr"}).then(function()
{
      drawText("Successfully published YD10 to channel");
    }, function(err) {
      drawText("Publish error: " + JSON.stringify(err));
    });
  });
  document.getElementById('ry10').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frm1", "input":
"VAL,DI:~40RYB00EA102_ZV01.dv DI:~40RYB00EA102_ZV01.descr
QUALITY,DI:~40RYB00EA102_XK97.dv DI:~40RYB00EA102_XK97.descr
SVALX,AI:~40RYR21CF003_XQ01.av AI:~40RYR21CF003_XQ01.units
AI:~40RYR21CF003_XQ01.descr"}).then(function() {
      drawText("Successfully published RY10 to channel");
    }, function(err) {
      drawText("Publish error: " + JSON.stringify(err));
    });
  });
  document.getElementById('UWBACKGROUND').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frm1", "input":
"Task:DsRM.current_time VAL_TIME_STR,Task:DsRM.current_time
VAL_DATE_STR1,Task:DsRM.current_time ACTIVITY4,Host:@H.current_time
ACTIVITY4,Task:DsRM.current_time STATCLR,Host:#0.host_status
STATCLR,Host:#1.host_status STATCLR,NI:~4SY0TSK16_TSTT.nv
DO_BOOL,Host:@H.current_time;l DO_BOOL,Host:@H.current_time;r
DO_BOOL,Host:@H.current_time;U"}).then(function() {
      drawText("Successfully published UWBACKGROUND to
channel");
    }, function(err) {
      drawText("Publish error: " + JSON.stringify(err));
    });
  });

```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

        document.getElementById('GBV').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frml", "input":
"SPARAM0,CSet:@a.dispcmd DO_BOOL,Host:@H.current_time;r NVAL,@1._ NVAL,@2._
QUALITY,@1._ QUALITY,@2._ VFX,@1.lf VFX,@1.hf VFX,@2.lf VFX,@2.hf
QUALITY_TEXT,@1._ TS_TIME,@1._ @3.code", "franame": "GBV", "varnames":
"AI:~40RYR21CF003_XQ01.av AI:~40YDR10CP001_XQ01.av", "attrnames": "lf hf lw
hw la ha units descr", "cp": 1}).then(function() {
    drawText("Successfully published GBV to channel");
    }, function(err) {
    drawText("Publish error: " + JSON.stringify(err));
    });
});
        document.getElementById('GBVF').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frml", "input":
"SPARAM0,CSet:@a.dispcmd DO_BOOL,Host:@H.current_time;r NVAL,@1._ NVAL,@2._
QUALITY,@1._ QUALITY,@2._ VFX,@1.lf VFX,@1.hf VFX,@2.lf VFX,@2.hf
QUALITY_TEXT,@1._ TS_TIME,@1._ @3.code", "franame": "GBV", "vfile":
"STABLE", "attrnames": "lf hf lw hw la ha units descr", "cp":
1}).then(function() {
    drawText("Successfully published GBVF to channel");
    }, function(err) {
    drawText("Publish error: " + JSON.stringify(err));
    });
});
        document.getElementById('GRP').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frml", "input":
"SPARAM0,CSet:@a.dispcmd DO_BOOL,Host:@H.current_time;r NVAL,@1._ NVAL,@2._
QUALITY,@1._ QUALITY,@2._ VFX,@1.lf VFX,@1.hf VFX,@2.lf VFX,@2.hf
QUALITY_TEXT,@1._ TS_TIME,@1._ @3.code", "franame": "GBV", "inips": "-codes
{40YDR10CF001_XQ01 40YDR10CP001_XQ01 40YDR10CT001_XQ01} -pagesize 20", "cp":
1}).then(function() {
    drawText("Successfully published GBVF to channel");
    }, function(err) {
    drawText("Publish error: " + JSON.stringify(err));
    });
});
        document.getElementById('GRPF').addEventListener('submit',
function(event) {
    event.preventDefault();
    sub.publish({"sframe": "frml", "input":
"SPARAM0,CSet:@a.dispcmd @a.fp @a.cp @a.np DO_BOOL,Host:@H.current_time;r
NVAL,@1._ NVAL,@2._ QUALITY,@1._ QUALITY,@2._ VFX,@1.lf VFX,@1.hf VFX,@2.lf
VFX,@2.hf QUALITY_TEXT,@1._ TS_TIME,@1._ @3.code", "franame": "GBV",
"inips": "-file CHANGING -pagesize 20", "cp": 1}).then(function() {
    drawText("Successfully published GBVF to channel");
    }, function(err) {
    drawText("Publish error: " + JSON.stringify(err));
    });
});
        document.getElementById('FREEZE').addEventListener('submit',
function(event) {
    event.preventDefault();

```



```
        centrifuge.rpc("RunStatus", {"sframe": "frm1", "rstatus":
"freeze"}).then(function(data){
            drawText("Run Status is: " + JSON.stringify(data));
        }, function(err) {
            drawText("Run Status error: " + JSON.stringify(err));
        });
    });
    document.getElementById('RUN').addEventListener('submit',
function(event) {
    event.preventDefault();
    centrifuge.rpc("RunStatus", {"sframe": "frm1", "rstatus":
"run"}).then(function(data){
        drawText("Run Status is: " + JSON.stringify(data));
    }, function(err) {
        drawText("Run Status error: " + JSON.stringify(err));
    });
});

document.getElementById('NEXTPAGE').addEventListener('submit',
function(event) {
    event.preventDefault();
    centrifuge.rpc("NextPage", {"sframe": "frm1", "pageset":
2}).then(function(data){
        drawText("Next Page is: " + JSON.stringify(data));
    }, function(err) {
        drawText("Next Page error: " + JSON.stringify(err));
    });
});

document.getElementById('GETVFILES').addEventListener('submit',
function(event) {
    event.preventDefault();
    centrifuge.rpc("GetVFiles", {"sframe":
"frm1"}).then(function(data){
        drawText("VFiles are: " + JSON.stringify(data));
    }, function(err) {
        drawText("VFiles get error: " + JSON.stringify(err));
    });
});
</script>
</head>
<body>
    <form id="form">
        <label for="input"></label><input type="text" id="input"
autocomplete="off" />
        <input type="submit" id="submit" value="FRM1">
    </form>
    <form id="dform">
        <label for="input"></label><input type="text" id="dinput"
autocomplete="off" />
        <input type="submit" id="submit" value="FRM2">
    </form>
    <form id="yd10">
        <input type="submit" id="submit" value="YD10">
    </form>
    <form id="ry10">
        <input type="submit" id="submit" value="Продувка ПГ">
    </form>
```

ООО «СКАДИ»	Средства веб-доступа для мониторинга и моделирования СКАДИВЕБ - Описание программы	Версия 1 01.12.2023
-------------	---	------------------------

```

<form id="UWBACKGROUND">
  <input type="submit" id="submit" value="Верхняя панель">
</form>
<form id="GBV">
  <input type="submit" id="submit" value="Гистограмма">
</form>
<form id="GBVF">
  <input type="submit" id="submit" value="Гистограмма-Файл">
</form>
<form id="GRP">
  <input type="submit" id="submit" value="Таблица-подписка">
</form>
<form id="GRPF">
  <input type="submit" id="submit" value="Таблица-подп-Файл">
</form>
<form id="FREEZE">
  <input type="submit" id="submit" value="Freeze">
</form>
<form id="RUN">
  <input type="submit" id="submit" value="Run">
</form>
<form id="NEXTPAGE">
  <input type="submit" id="submit" value="NextPage">
</form>
<form id="GETVFILES">
  <input type="submit" id="submit" value="GetVFiles">
</form>
<ul id="messages"></ul>
</body>
</html>

```

ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

АЭС	- Атомная Электростанция
ОС	- Операционная система
ПО	- Программное обеспечение
СВБУ	- Система верхнего блочного уровня
СКАДИ	- Средства комплексной автоматизации и диагностики
НМІ	- Human Machine Interface
JSON	- JavaScript Object Notation
SVG	- Scalable Vector Graphics

