



**АКЦИОНЕРНОЕ ОБЩЕСТВО  
«СИСТЕМНЫЙ ОПЕРАТОР ЕДИНОЙ ЭНЕРГЕТИЧЕСКОЙ СИСТЕМЫ»**

## **ОБЩЕЕ ОПИСАНИЕ**

**программного обеспечения мониторинга функционирования устройств  
и комплексов системы мониторинга переходных режимов**

Москва, 2022

## СОДЕРЖАНИЕ

Перечень сокращений.....	3
1 Назначение системы.....	4
2 Структура и связи компонентов системы.....	5
3 Программное обеспечение .....	8
4 Техническое обеспечение .....	9
5 Описание интерфейсов системы.....	10
6 Описание правил нумерации версий системы .....	11

## Перечень сокращений

Таблица 1. Перечень сокращений

Сокращение	Описание или расшифровка
API	Интерфейс программирования приложений (англ. Application programming interface) - набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) или операционной системой для использования во внешних программных продуктах
SPA	Single Page Application, одностраничное веб-приложение
АИП	Автоматизированная интеграционная платформа
АО "СО ЕЭС"	Акционерное общество "Системный оператор единой энергетической системы России"
АС СИ СМНР	Автоматизированная система сбора информации от регистраторов системы мониторинга переходных режимов
БД	База данных
ИА	Исполнительный аппарат
КСВД	Концентратор синхронизированных векторных данных
ПО МФУК	Программное обеспечение мониторинга функционирования устройств и комплексов системы мониторинга переходных режимов
НСИ	Нормативно-справочная информация
ОС	Операционная система
ПО	Программное обеспечение
СУБД	Система управления базами данных
УСВИ	Устройство синхронизированных векторных измерений

## 1 Назначение системы

Система МФУК предназначена для обеспечения процесса учёта комплексов СМПР, их настроек и эксплуатационного состояния, получения информации, ведения и предоставления технологическому персоналу и автоматизированным системам НСИ о действующих устройствах и комплексах СМПР.

Один экземпляр системы внедрен в ИА и к ней предоставлен доступ на рабочих местах из филиалов АО "СО ЕЭС".

Система предназначена для ведения структурированного информационного хранилища в рамках деловых процессов по организации эксплуатации СМПР ЕЭС России.

Система МФУК предназначена для автоматизации следующих процессов:

- автоматизации технического учёта и анализа функционирования комплексов СМПР;
- ведение протоколов настройки УСВИ и КСВД;
- мониторинг функционирования устройств и комплексов СМПР ЕЭС России и их эксплуатационного состояния;
- регистрация информации обо всех этапах жизненного цикла технических средств;
- систематизация настроек конфигурирования устройств и комплексов СМПР;
- поиск хранящейся информации;
- подготовка отчетной информации;
- автоматизации ведения НСИ об устройствах и комплексах СМПР;
- передача автоматизированным системам актуальной НСИ о действующих устройствах и комплексах СМПР.

## 2 Структура и связи компонентов системы

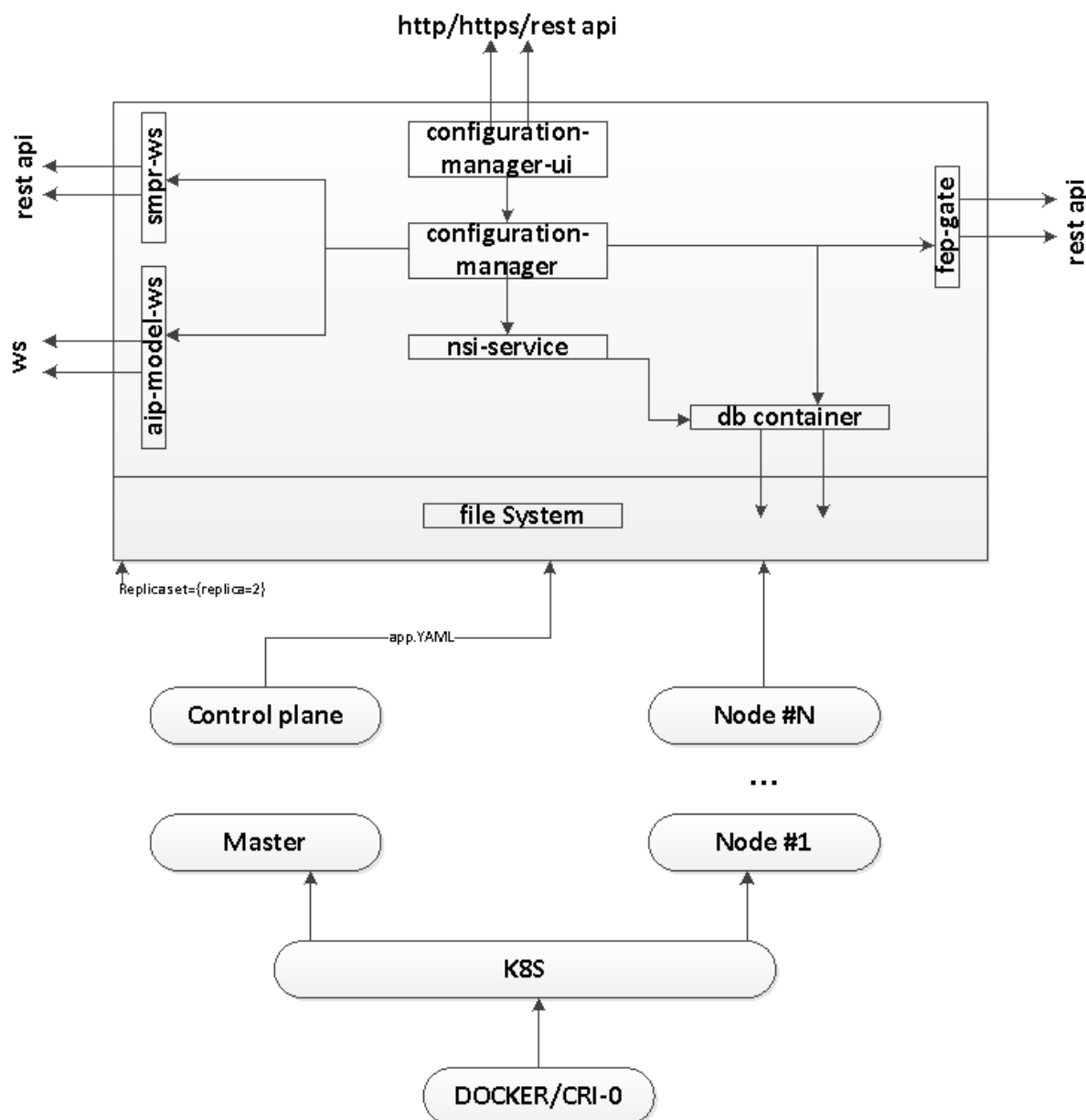


Рис. 1 Схема структуры и связей компонентов системы

ПО МФУК функционирует на серверах общего назначения под управлением ОС Astra Linux, среды выполнения приложения в виде контейнеров Docker и ПО оркестровки (взаимодействия и порядка запуска контейнеров) и обеспечения работы контейнеров как единого приложения – Kubernetes.

Инсталляция kubernetes в рамках МФУК предполагает установку собственного кластера, а также назначение для сервера всех ролей Kubernetes – master (точка обращения к kubernetes утилитами администрирования), control plane (планировщик запуска приложений и распределения ресурсов) и Node (среда выполнения контейнеров приложения). Для ПО МФУК выделен

отдельный namespace в kubernetes – mfuk, в рамках которого запускаются все контейнеры и их связи.

Состав логических компонентов в рамках МФУК следующий:

- configuration-manager – центральный серверный компонент, который обеспечивает основную логику системы и предоставляет REST API для пользовательского интерфейса. Реализован в виде docker-образа configuration-manager;
- configuration-manager-ui – пользовательский интерфейс МФУК, реализованный в виде SPA-вебприложения. Реализован в виде docker-образа configuration-manager-ui;
- air-model-ws – компонент для взаимодействия с АИП в части приема НСИ. Реализован в виде docker-образа air-model-ws;
- smpr-ws – компонент для взаимодействия с АС СИ СМПП в части приема топологии. Реализован в составе docker-образа configuration-manager;
- fer-gate – компонент для взаимодействия с ФЭП в части приема текущих классов качества КСВД и УСВИ. Реализован в составе docker-образа configuration-manager;
- nsi-service – компонент обработки НСИ в составе МФУК, обеспечивающий API предоставления НСИ и ее централизованное хранение. Реализован в виде docker-образа nsi-service;
- db container - БД на основе PostgreSQL для хранения НСИ, конфигураций и журналов работ. Реализована в виде docker-образа postgresql.

Компонент configuration-manager обеспечивает централизованный доступ к функциональности Системы в виде REST API и взаимодействует со всеми остальными компонентами системы. Пользовательский интерфейс в виде веб-приложения обращается за данными к configuration-manager по REST API.

Компонент air-model-ws обеспечивает взаимодействие по веб-сервисам с АИП; компонент smpr-ws - взаимодействие по REST API с АС СИ СМПП; компонент fer-gate - взаимодействие по REST API с ФЭП.

Компоненты configuration-manager и nsi-service работают с БД для хранения информации (НСИ, конфигурации, журналы работы) и предоставления ее по REST API. Сама БД реализована в виде отдельного контейнера с сохранением данных на дисках хостового сервера.

Также в рамках системы фигурируют компоненты администрирования в следующем составе:

- kubernetes – утилита администрирования сервера kubernetes. Используется для установки, обновления и конфигурирования самого kubernetes;

- `kubectl` – клиент `kubernetes` для настройки и управления задачами. Используется для установки, обновления и конфигурирования приложений в рамках `kubernetes`;
- `k9s` – еще один клиент `kubernetes`, использующий `kubectl` и предоставляющий более высокоуровневый интерфейс взаимодействия по сравнению с `kubectl`. Используется как утилита для просмотра текущего состояния и режимов работы контейнеров, их связей, а также логов отдельных компонентов;

### 3 Программное обеспечение

Для обеспечения работоспособности ПО МФУК необходимо установить следующее ПО:

Сервер приложений:

- ОС Astra Linux;
- среда выполнения приложения Docker последней версии;
- Kubernetes (ПО оркестровки и обеспечение работы контейнеров как единого приложения) последней версии. Установка осуществляется согласно официальной документации <https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/> ;
- K9s актуальной версии;
- СУБД Postgres Pro Ent;
- ПО Keepalived.

Клиент:

- ОС Windows;
- браузер с поддержкой HTML5 и разрешенным Javascript (Mozilla Firefox версии 5 или выше, Google Chrome версии 18 и выше, Microsoft Edge).

Процесс установки ПО описан в документе "Инструкция по установке и настройке".



## **4 Техническое обеспечение**

Для функционирования ПО МФУК требуется три сервера (виртуальные или физические) в следующем составе:

1. Сервер БД/приложений:
  - 4х ядерный CPU с архитектурой x86-64;
  - 16 Гб RAM;
  - 800 Гб дискового пространства;
2. Сервер БД/приложений:
  - 4х ядерный CPU с архитектурой x86-64;
  - 16 Гб RAM;
  - 800 Гб дискового пространства.
3. Сервер БД:
  - 4х ядерный CPU с архитектурой x86-64;
  - 8 Гб RAM;
  - 400 Гб дискового пространства.

## **5 Описание интерфейсов системы**

Доступ к функциям системы реализуется через REST API.

Пользовательский интерфейс системы более подробно описан в Руководстве пользователя.

## **6 Описание правил нумерации версий системы**

Версия системы состоит из 4 уровней, первые два обозначают версию программы, остальные два - дополнительная информация о версии (фиксы и хэш коммита), например, - v.3.4.65.24с67, где:

- 3 - основной номер версии;
- 4 - минорный номер внутри основной версии;
- 65 - версия изменений, связанных с исправлениями в версии;
- 24с67 - дополнительный номер - хэш коммита (не привязан к глобальности изменений).