Федеральное государственное унитарное предприятие
Российский федеральный ядерный центр
Всероссийский научно-исследовательский институт экспериментальной физики

УТВЕРЖДЕН
07623615.00097-05 90 01-ЛУ

КОМПЛЕКС ПРОГРАММ В ЗАЩИЩЕННОМ ИСПОЛНЕНИИ
«СИСТЕМА ПОЛНОГО ЖИЗНЕННОГО ЦИКЛА ИЗДЕЛИЙ
«ЦИФРОВОЕ ПРЕДПРИЯТИЕ»

Программный модуль «Комплекс средств интеграции»

Руководство администратора

07623615.00097-05 90 01

Листов 31

Инв. № подл. | Подп. и дата | Взам. инв. № | Инв. № дубл. | Подп. и дата

2021

АННОТАЦИЯ

В документе приводится общее описание программного модуля «Комплекс средств интеграции» комплекса программ в защищенном исполнении «Система полного жизненного цикла изделий «Цифровое предприятие» и его компонентный состав.

Представлена информация о составе дистрибутивного пакета и необходимых процедурах, требуемых для установки и настройки программного модуля.

Указаны команды запуска и остановки, а также общие алгоритмы проверки доступности программного модуля.

Полное наименование: программный модуль «Комплекс средств интеграции».

Краткое наименование: КСИ.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ

1.1. Назначение программы

КСИ предназначен для получения данных из функционально-специализированных информационных систем предприятия согласно заданным алгоритмам, путем исполнения соответствующих сценариев интеграции и специализированных сценариев обработки и маршрутизации сообщений.

В КСИ реализована функциональность, позволяющая автоматизировать процедуры оперативного получения данных из смежных и внешних систем.

КСИ предоставляет работникам предприятия наборы инструментов для поддержания целостной совокупности данных для последующего использования системами мониторинга и информационного анализа (СИА), а также для взаимодействия с внешними информационными системами.

1.2. Функции программы

КСИ реализует следующие функции:

- хранение метаданных ETL-процедур в файловом хранилище;
- хранение метаданных ETL-процедур в реляционной базе данных;
- выполнение ETL-процедур;
- вывод результатов выполнения ETL-процедур в файл и/или базу данных;
- управление и мониторинг состояния сервера трансформаций;
- управление автоматически выполняемыми задачами ETL (запуск/остановка, запуск по расписанию);
- управление ETL-задачами в репозитории метаданных;
- визуальное представление журнала исполнения ETL-задач;
- мониторинг результата выполнения ETL-задачи, вложенных трансформаций, шагов трансформаций;
- конструирование и управление маршрутами сообщений;
- обмен сообщениями между компонентами распределенной вычислительной среды.

1.3. Состав технических и программных средств

КСИ реализован в архитектуре «клиент - сервер». В серверную часть входят:

- сервер(ы) баз данных (сервер БД) – хранение и обработка прикладных и системных данных;
- сервер приложений (сервер КСИ) – реализация программных средств на платформе «Java» и сервере «Tomcat», позволяющего запустить интеграционную платформу.

Минимальные требования к серверу КСИ и серверу БД (только для системных баз данных):

- процессор: «x86-64», от 2х ядер;
- оперативная память: 8 ГБ;
- сетевой интерфейс: 1000 Мб/сек;
- дисковая подсистема: 32 ГБ для ОС, 32 ГБ для данных.

Состав ПО на рабочем месте пользователя:

- операционная система: «Astra Linux SE»;
- браузер: «Firefox» или «Chromium» из состава «Astra Linux SE»;
- редактор сценариев ETL: «Pentaho Data Integration».

Состав ПО на серверах (сервер БД, сервер «LDAP», сервер КСИ):

- операционная система: «Astra Linux SE»;
- сервер КСИ: «Tomcat» и компоненты КСИ на платформе «Java»;
- сервер БД: СУБД «Синергия-БД» / «PostgreSQL».

2. СТРУКТУРА ПРОГРАММЫ

КСИ состоит из следующих базовых функциональных блоков:

- подсистема метаданных, обеспечивающая ведение репозитория;
- подсистема обработки данных, обеспечивающая хранение и доступ к данным;
- подсистема ETL, обеспечивающая исполнение задач трансформации данных;
- подсистема управления СИ, обеспечивающая управление параметрами программы;
- интегрированный сервер трансформаций с компонентом мониторинга состояния сервера;
- подсистема управления сообщениями, обеспечивающая обмен структурированными сообщениями в распределенной вычислительной среде;
- интеграционная шина, обеспечивающая маршрутизацию сообщений.

Схема функциональной структуры КСИ представлена на рис 1.

Структурная схема КСИ

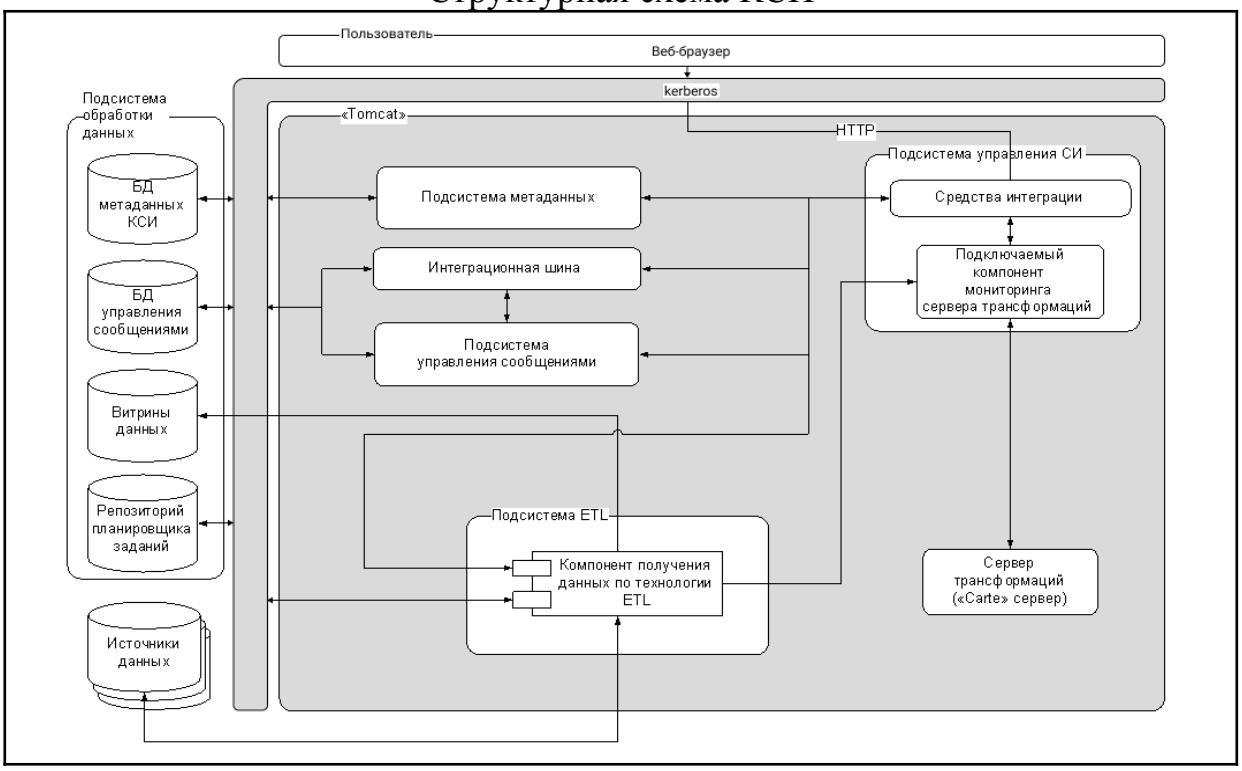

Рисунок 1

2.1.1. Подсистема метаданных

Подсистема обеспечивает:

- хранение метаданных ETL-процедур в файловом хранилище;
- хранение метаданных ETL-процедур в реляционной базе данных.

2.1.2. Подсистема обработки данных

Функциональность подсистемы реализуется СУБД.

Подсистема обеспечивает:

- доступ к хранимым данным;
- хранение программных данных - репозиторий программы;
- хранение прикладных данных - пользовательские БД и витрины данных.

2.1.3. Подсистема «ETL»

Подсистема обеспечивает:

- чтение метаданных ETL-процедур из файлового хранилища, базы данных;
- выполнение ETL-процедур;
- вывод результатов выполнения ETL-процедур в файл и/или базу данных.

2.1.4. Подсистема управления средствами интеграции

Подсистема обеспечивает:

- управление и мониторинг состояния сервера трансформаций;
- управление автоматически выполняемыми задачами ETL (запуск/остановка, запуск по расписанию);
- управление ETL-задачами в репозитории метаданных;
- визуальное представление журнала исполнения ETL-задач;
- мониторинг результата выполнения ETL-задачи, вложенных трансформаций, шагов трансформаций;
- реализацию визуального инструмента для управления серверами в следующих режимах подсистемы: рабочий стол, серверы трансформации, задания трансформации, управление сообщениями, интеграционная шина;

- автоматический запуск и остановку интегрированного сервера при старте и остановке самого приложения;
- автоматический запуск и остановку локального сервера трансформаций при старте и остановке приложения, если в настройках сервера выставлен соответственный признак.

2.1.5. Подсистема управления сообщениями

Подсистема обеспечивает:

- поддержку полной иерархии типов сообщений «Message» («TextMessage», «BytesMessage», «StreamMessage», «MapMessage»);
- кросс-языковой обмен сообщениями между компонентами распределенной вычислительной среды, реализованными на любом языке;
- передачу данных по протоколам «TCP», «UDP», «HTTP», «HTTPS»;
- передачу сообщений по зашифрованным каналам, используя «SSL» протокол;
- возможность коммуникации в синхронном и асинхронном режимах;
- возможность обмена сообщениями по моделям отправитель-получатель и издатель-подписчик;
- передачу сообщений с данными, которые содержат иерархическую структуру топиков в формате «дерева»;
- получение данных подписчиком с нескольких топиков;
- единое пространство хранения сообщений с возможностью использования исторических данных.

2.1.6. Сервер трансформаций

Сервер трансформаций обеспечивает:

- предоставление внешнего программного интерфейса приложения (API) для взаимодействия с подсистемой управления СИ;
- генерацию статуса «Carte» сервера и трансформаций/задач в формате JSON;

- автоматический выбор выходного формата данных (XML, JSON или HTML) в зависимости от соответствующего входного параметра;
- запись только тех данных, которые необходимы для отображения на клиенте.

2.1.7. Интеграционная шина

Интеграционная шина обеспечивает:

- взаимодействие приложений с помощью обмена сообщениями;
- планировщик отправки сообщений в определенное время;
- передачу сообщений в форматах JSON, XML, текст, файл;
- выполнение сложной обработки сообщений при сохранении независимости и гибкости с применением фильтров;
- маршрутизацию сообщений отдельных этапов обработки в зависимости от набора условий;
- переводчик сообщений для использования разных форматов данных;
- подключение по протоколам «RESTful HTTP», «SOAP over HTTP»;
- взаимодействие с подсистемой управления сообщениями в части использования транспортного модуля управления сообщениями для обеспечения интеграционного взаимодействия;
- гарантированную доставку сообщений;
- предупреждения пользователю по сообщениям, которые не могут быть доставлены;
- отдельным приложениям взаимодействовать друг с другом, но разобщенным образом, чтобы приложения можно было легко добавлять или удалять, не затрагивая другие;
- передачу событий из одного приложения в другое;
- отправку запроса на получение ответа от получателя;
- получение обратного адреса, чтобы определить, куда отправить ответ;
- получение идентификатора корреляции для определения, от какого запроса ответ;

- настройку фильтров сообщений для отказа получения нежелательных сообщений;
- настройку динамической маршрутизации для избегания зависимости от сохраненных настроек маршрутизатора;
- создание динамических списков получателей для отправки сообщений;
- создание разветвителя для обработки сообщений, если оно содержит несколько элементов;
- создание агрегатора для объединения результатов отдельных, но связанных сообщений;
- преобразование последовательности для получения потока связанных, но непоследовательных сообщений обратно в правильном порядке;
- обработку сообщения, состоящего из нескольких элементов, каждый из которых может требовать различной обработки;
- проверку на загруженность получателя;
- выбор конкретных сообщений за определенный период, с возможностью повторной отправки;
- функциональность, которая позволит отложить отправку сообщения;
- функциональность балансировки нагрузки на несколько конечных потребителей;
- ручное и автоматическое прекращение отправки сообщений, если потребитель не работает;
- удаленный вызов сервиса в распределенной системе, где сервис ищется из реестра сервисов;
- функциональность определения связанных действий на маршруте, которые должны быть либо все успешно завершены, либо не выполнены;
- обработку сообщения в цикле;
- настройку фильтров содержимого для поиска необходимых данных в больших сообщениях;
- обработку сообщений, которые семантически эквивалентны, но приходят в другом формате;

- применение скриптов, которые не влияют на сообщение;
- опрос потребителей (работает, не работает);
- возможность выбора сообщений, которые потребитель хочет получить;
- проверку пропущенных сообщений, пока получатель их не слушает;
- проверку на отправку дублирующих сообщений;
- администрирование модуля межсервисного взаимодействия;
- ведение исторических сообщений;
- отображение журнала обработки сообщений.

3. АУТЕНТИФИКАЦИЯ ПОЛЬЗОВАТЕЛЕЙ

КСИ поддерживает возможность сквозной аутентификации пользователей с использованием сервера «Kerberos».

Ключевая особенность такого способа аутентификации – доступ к данным осуществляется в контексте доменного пользователя согласно правилам СУБД.

Подобная схема обусловлена применением в автоматизированных системах в защищенном исполнении, т. к. предполагает использование доверенных (сертифицированных) средств разграничения доступа ОС и СУБД и их отсутствие в КСИ.

Аутентификация пользователей в системе осуществляется согласно схеме, представленной на рис. 2.

Схема сквозной аутентификации пользователей

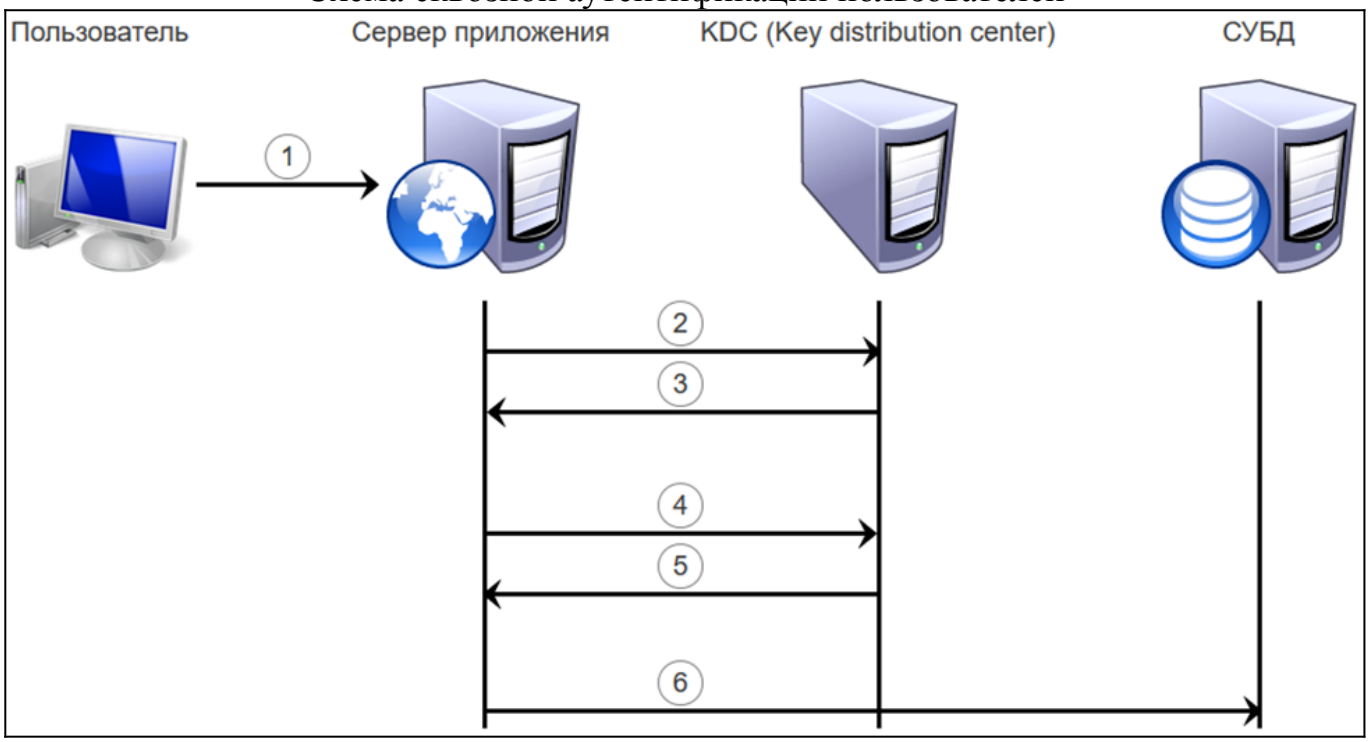

Рисунок 2

Пользователь авторизуется в операционной системе и браузер обращается к серверу приложения, чтобы получить доступ к необходимым ему данным (1). Сервер приложения обращается к «KDC» с запросом первичной аутентификации (2).

«KDC» – служба, работающая на защищенном сервере. «KDC» хранит базу данных с информацией об учетных записях всех клиентов сети. Вместе с

информацией о каждом абоненте в базе данных «KDC» хранится криптографический ключ, известный только этому абоненту и службе «KDC». Этот ключ служит для связи клиента с центром дистрибуции ключей.

После успешного подтверждения его подлинности, «KDC» выдает первичное удостоверение пользователя для доступа к сетевым ресурсам – «Ticket Granting Ticket» («TGT») (3). Затем, пользователь, предъявляя «TGT» (4), получает от «KDC» удостоверение для доступа к конкретному сетевому ресурсу – «Ticket Granting Service» («TGS») (5).

После получения «TGS», пользователь обращается с ним к СУБД (6) и, после взаимной проверки подлинности, получает доступ к запрашиваемым данным согласно матрице доступа.

В рассматриваемом случае считается, что все пользователи домена допущены к визуальному интерфейсу КСИ, имеют равный доступ к объектам виртуальной файловой системы КСИ («видят» все сценарии интеграции и т. п.).

Если существует необходимость ограничить список пользователей, допущенных к КСИ, предлагается использовать защищенный веб-сервер «Apache2», входящий в состав «Astra Linux SE».

В этом случае при обращении к КСИ клиентский браузер передает веб-серверу «Apache2» учетные данные «Kerberos».

Если значения параметров не заданы – учетные данные «Kerberos» не передаются веб-серверу и происходит отказ в доступе.

Веб-сервер, обращаясь к «LDAP» службе каталогов, проверяет действительность представленных данных. В случае успешного прохождения проверки пользователю выводится веб-страница КСИ.

Доступ субъектов к интерфейсу КСИ ограничен разрешительным списком, задаваемым в конфигурационном файле веб-сервера.

Взаимодействие веб-сервера и КСИ осуществляется по протоколу «AJP» (по определенному порту), который обеспечивает проксирование запросов к серверу КСИ и передачу запрашиваемой информации.

Для того, чтобы ограничить возможность доступа пользователей / программ непосредственно к серверу КСИ и использовать только веб-сервер «Apache2» следует:

- средствами сетевого экранирования ОС «Astra Linux» закрыть порт «AJP» для всех «ip» адресов кроме веб-сервера командой «iptables»;
- отключить любые возможности доступа к КСИ (порты доступа 8080 и т. п.) за исключением «AJP».

4. ПОДГОТОВКА И УСТАНОВКА ПРОГРАММЫ

Установку КСИ рекомендуется выполнять в следующей последовательности:

- подготовка и конфигурирование серверов;
- сборка из исходных кодов (в случае варианта поставки в исходных кодах);
- установка и настройка КСИ и программного окружения (ОС, СУБД, виртуальная машина «Java»);
- установка и настройка веб-сервера «Apache2» (опционально).

Дистрибутив, тексты программ (в случае варианта поставки в исходных кодах) и вспомогательные файлы, требуемые для сборки и установки КСИ, поставляются на машинном носителе (инсталляционный пакет).

Каталоги инсталляционного пакета:

- «build_tools» – инструменты сборки программных компонентов КСИ;
- «config» – конфигурационные файлы, содержащие параметры сборки и установки КСИ;
- «demo» – демонстрационные сценарии взаимодействия с внешними системами и маршруты сообщений;
- «dependencies» – вспомогательные файлы для сборки программных компонентов КСИ;
- «etl_helper» – исходные коды справочной подсистемы редактора сценариев ETL;
- «ip_helper» – исходные коды справочной подсистемы КСИ;
- «jdk» – дистрибутив «Java Development Kit» («JDK»);
- «pdi» – дистрибутив «Pentaho Data Integration»;
- «resources» – дополнительные ресурсы;
- «samples» – пример демонстрационной базы данных;
- «scripts» – вспомогательные программы сборки и установки;
- «service» – скрипт создания службы запуска КСИ;
- «sql» – скрипт создания служебной базы данных КСИ;
- «src» – тексты программных компонентов КСИ;

- «tomcat» – сервер приложений «Tomcat».

Файлы инсталляционного пакета:

- «build.sh» – скрипт сборки компонентов КСИ из исходных кодов;
- «install.sh» – скрипт установки КСИ.

4.1. Подготовка и конфигурирование серверов

КСИ предполагает наличие следующих (виртуальных) серверов:

- сервер с исходными кодами и/или дистрибутивом системы (далее – «`ipsrc`») и дистрибутивом операционной системы, доступный по протоколу «FTP»;
- сервер службы каталогов «LDAP» (далее – «`ipldap`») – обеспечивает возможность входа пользователей в КСИ без ввода «логина»/пароля с использованием технологии единого входа в сеть («SSO»);
- сервер(ы) СУБД (далее – «`ipdb`») – сервер для хранения системных данных и сервер для хранения прикладных данных (могут быть объединены);
- сервер КСИ (далее – «`ip`») – сервер, обеспечивающий функционирование комплекса средств интеграции.

Перед установкой КСИ необходимо выполнить следующие действия:

- присвоить всем серверам статические «ip» адреса;
- присвоить всем серверам соответствующие имена (в файлах «`/etc/hosts`», «`/etc/hostname`»);
- обеспечить возможность доступа ко всем серверам по протоколу «SSH»;
- обеспечить размещение «FTP» репозитория ОС «Astra Linux SE» на сервере «`ipsrc`», на который будут ссылаться другие серверы при установке системы.

Выполнение инструкций настоящего руководства необходимо осуществлять под учетной записью пользователя (далее – «user»), имеющей возможность выполнять команды от имени суперпользователя («sudo»). Такая учетная запись должна быть определена на всех серверах.

После конфигурирования серверов необходимо скопировать файлы инсталляционного пакета на сервер «ipsrc» в домашний каталог пользователя «user» в каталог «install».

4.1.1. Разрешение подключения к серверу по протоколу «SSH»

Для обеспечения доступа к серверу под управлением ОС «Astra Linux SE» необходимо:

- проверить наличие пакета командой:

```
sudo apt list ssh
```

- в случае, если пакет отсутствует, установить его командой:

```
sudo apt install ssh
```

- для разрешения доступа по «SSH» выполнить команды:

```
sudo systemctl enable ssh
sudo systemctl start ssh
```

Указанные команды выполняются для всех серверов, участвующих в установке системы.

4.1.2. Создание локального репозитория

Для создания в ОС «Astra Linux SE» репозитория из «ISO» образов установочных дисков необходимо на сервере «ipsrc»:

- создать каталог для размещения репозитория:

```
sudo mkdir -p /srv/repo/smolensk/main
```

- «примонтировать» образ установочного диска (если на компьютере нет каталога «/media/cdrom» – то создать каталог «/media/cdrom»):

```
[ -d /media/cdrom ] || sudo mkdir /media/cdrom
sudo mount /путь_к_ISO-образу /media/cdrom
```

- скопировать файлы из образа в каталог репозитория:

```
sudo cp -a /media/cdrom/* /srv/repo/smolensk/main
```

- установить «FTP» сервер:

```
sudo apt install vsftpd
```

- «отмонтировать» «ISO» образ диска:

```
sudo umount /media/cdrom
```

- в конфигурационный файл «/etc/vsftpd.conf» внести следующие данные:

```
listen=YES
listen_ipv6=NO
#Анонимный доступ разрешен
anonymous_enable=YES
local_enable=NO
anon_root=/srv/repo
no_anon_password=YES
hide_ids=YES
```

- перезапустить сервис «FTP»:

```
sudo systemctl restart vsftpd
```

- настроить источники пакетов (файл «/etc/apt/sources.list»):

```
deb ftp://localhost/smolensk/main smolensk main contrib non-free
```

Ссылка на созданный «FTP» репозиторий для других серверов настраивается автоматически в процессе установки КСИ (согласно последнему пункту инструкции – через файл «/etc/apt/sources.list»).

4.2. Сборка из исходных кодов

Сборка из исходных кодов осуществляется на сервере «ipsrc» при запуске сценария:

```
cd $HOME/install
./build.sh
```

В процессе исполнения сценария сборки КСИ на сервере «ipsrc» осуществляется:

- установка пакетов «rsync», «dos2unix», «sshpass», «zip», «xrdp»;
- распаковка архивов с исходными кодами (в каталог «$HOME/ip/», причем имя «ip» задается в конфигурационном файле «build.cfg» параметром «appName»);
- распаковка архивов с зависимостями (в каталог «$HOME»);
- для текстов программ (в каталоге «$HOME/ip/») выполняется преобразование кодировки файлов (командой «dos2unix»).

Кроме этого, осуществляется установка следующего программного обеспечения:

- «Java Development Kit» («JDK») версии 1.8.0_282-b08 и 11.0.1;
- «Apache Maven» версии 3.6.3 или выше;
- «Gradle» версии 6.6.1 или выше;
- «Node.js» версии 14.16.0 или выше (включая «npm» версии 6.14.11 или выше).

После выполнения подготовительных процедур осуществляется сборка программных компонентов КСИ.

Дистрибутивы, полученные в результате сборки, помещаются в каталоги «`$HOME/ip/dist/app`» и «`$HOME/ip/dist/appSSO`».

В состав дистрибутива также включаются:

- «Java Development Kit» – среда функционирования КСИ (каталог «`$HOME/ip/dist/jdk`»);
- «Pentaho Data Integration» – редактор сценариев ETL в русской и английской локализации (каталог «`$HOME/ip/dist/pdi`»);
- демонстрационные сценарии интеграции и маршруты сообщений (каталог «`$HOME/ip/dist/demo`»);
- дополнительные ресурсы (каталог «`$HOME/ip/dist/resources`»).

Далее приведены команды и порядок сборки компонентов:

- «pentaho-kettle»:

```
mvn clean install -DskipTests
```

- «ip»:

```
gradle clean build --offline
```

4.3. Установка программы

Для запуска программы установки КСИ на сервере «`ipsrc`» следует выполнить сценарий:

```
cd $HOME/install
./install.sh
```

4.3.1. Описание конфигурации

Перед началом установки КСИ на сервере «ipsrc» следует определить значения основных параметров, задаваемых в файле «$HOME/install/config/install.cfg».

4.3.1.1. Параметры конфигурации сервера с дистрибутивом системы

В отношении сервера «ipsrc» выполняются настройки:

- «srcIP» – «IP» адрес сервера;
- «srcAdminUsr» – имя учетной записи пользователя, имеющего возможность выполнять команды от имени суперпользователя («sudo»);
- «srcAdminPwd» – пароль для этой учетной записи (если значение не определено – потребуется ввод с клавиатуры во время установки).

4.3.1.2. Параметры конфигурации сервера приложения

В отношении сервера «ip»:

- «appIP» – «IP» адрес сервера;
- «appAdminUsr» – имя учетной записи пользователя, имеющего возможность выполнять команды от имени суперпользователя («sudo»);
- «appAdminPwd» – пароль для этой учетной записи (если значение не определено – потребуется ввод с клавиатуры во время установки);
- «dirApp» – каталог, куда будет установлена система;
- «appUsr» – пользователь, от имени которого запускается система (владелец каталога «dirApp»);
- «appUsrPwd» – пароль этого пользователя (если значение не определено – потребуется ввод с клавиатуры во время установки);
- «appUsrPwdForceChange» – флаг (принимает значение «1» или «0»), определяющий необходимость / отсутствие необходимости изменения пароля пользователя «appUsr», если пользователь был создан ранее;

- «installDemo» – флаг (принимает значение «1 или «0»), определяющий необходимость / отсутствие необходимости установки демонстрационного примера.

4.3.1.3. Параметры конфигурации сервера службы каталогов

В отношении сервера «ipldap» выполняются настройки:

- «ldapIP» – «IP» адрес сервера;
- «ldapAdminUsr» – имя учетной записи пользователя, имеющего возможность выполнять команды от имени суперпользователя («sudo»);
- «ldapAdminPwd» – пароль для этой учетной записи (если значение не определено – потребуется ввод с клавиатуры во время установки);
- «domainName» – имя устанавливаемого домена;
- «domainAdminUsr» – администратор домена;
- «domainAdminPwd» – пароль администратора домена (если значение не определено – потребуется ввод с клавиатуры во время установки);
- «testUser» – имя доменного пользователя для тестирования входа в систему;
- «testUserPwd» – пароль этого пользователя (если значение не определено – потребуется ввод с клавиатуры во время установки).

4.3.1.4. Параметры конфигурации сервера баз данных

В отношении сервера «ipdb» выполняются настройки:

- «dbIP» – «IP» адрес сервера;
- «dbAdminUsr» – имя учетной записи пользователя, имеющего возможность выполнять команды от имени суперпользователя («sudo»);
- «dbAdminPwd» – пароль для этой учетной записи (если значение не определено – потребуется ввод с клавиатуры во время установки).

4.3.2. Описание процедуры установки

В процессе исполнения сценария установки КСИ осуществляются следующие действия:

- конфигурация репозитория и установка пакетов «rsync», «dos2unix», «sshpass», «zip», «xrdp» на серверах «ip», «ipldap», «ipdb»;
- установка СУБД на сервере «ipdb» и ее конфигурация для доступа пользователей с использованием механизмов сквозной аутентификации (тип аутентификации – «GSS»);
- копирование дистрибутива на сервер «ip», распаковка архивов дистрибутива в каталог, определенный конфигурационным параметром «dirApp»;
- установка и конфигурация контроллера домена на сервере «ipldap» и клиентов домена на серверах «ip», «ipdb»;
- установка и настройка КСИ и его программных компонентов;
- конфигурация КСИ для доступа пользователей с использованием механизмов сквозной аутентификации;
- установка демонстрационного примера (определяется конфигурационным параметром «installDemo»);
- создание «тестового» доменного пользователя;
- создание службы запуска КСИ;
- запуск КСИ.

4.4. Совместное применение с веб-сервером

Веб-сервер «Apache2» может выступать в роли обратного прокси-сервера и использоваться для организации единой точки доступа к серверу КСИ.

Для обеспечения доступа к КСИ через веб-сервер «Apache2» необходимо установить сам веб-сервер и дополнительные модули к нему, выполнив команды:

```
apt-get install apache2
apt-get install libapache2-mod-proxy-html libapache2-mod-auth-
kerb
a2enmod proxy
```

```
a2enmod proxy_http
```

Для аутентификации пользователей посредством «Kerberos» необходимо:

- отключить модуль аутентификации через «PAM»:

```
a2dismod auth_pam
```

- активировать модуль «auth_kerb»:

```
a2enmod auth_kerb
```

- в директории «/etc/apache2/sites-available/» создать конфигурационный файл виртуальных хостов «host.conf»:

```
<VirtualHost *:80>
  ServerName localhost
  ErrorLog /var/log/apache2/ajp.error.log
  CustomLog /var/log/apache2/ajp.log combined

  DocumentRoot /var/www/
  <Directory />
    AddDefaultCharset Off
    Order deny,allow
    Allow from all
    AuthType Kerberos
    KrbServiceName host/apache-srv.vniief.local@VNIIEF.LOCAL
    Krb5Keytab /etc/web.keytab
    KrbMethodNegotiate on
    KrbMethodK5Passwd off
    KrbSaveCredentials on
    KrbLocalUserMapping on
    require valid-user
  </Directory>

  <Proxy "http://apache-srv.vniief.loc/ip/*">
    AddDefaultCharset Off
    Order deny,allow
    Allow from all

    AuthType Kerberos
    KrbServiceName host/apache-srv.vniief.loc@VNIIEF.LOC
    Krb5Keytab /etc/web.keytab
    KrbMethodNegotiate on
    KrbMethodK5Passwd off
    KrbSaveCredentials on
```

```
    KrbLocalUserMapping on

    require valid-user
  </Proxy>

  <Proxy "ajp://ip-srv.vniief.loc:8009/ip/*">
    AddDefaultCharset Off
    Order deny,allow
    Allow from all

    AuthType Kerberos
    KrbServiceName host/apache-srv.vniief.loc@VNIIEF.LOC
    Krb5Keytab /etc/web.keytab
    KrbMethodNegotiate on
    KrbMethodK5Passwd off
    KrbSaveCredentials on
    KrbLocalUserMapping on

    require valid-user

  </Proxy>

  ProxyPass /pentaho ajp://ip-srv.vniief.loc:8009/ip
  ProxyPassReverse /ip ajp://ip-srv.vniief.loc:8009/ip

</VirtualHost>
```

- активировать виртуальный хост:

```
a2ensite /etc/apache2/sites-available/host.conf
```

- перезапустить сервер «apache2»:

```
service apache2 restart
```

Теперь сервер КСИ может быть доступен по адресу:

```
http://apache-srv.vniief.loc/ip
```

Все подчеркнутые значения зависят от инфраструктуры предприятия.

Обратите внимание на выделенный цветом параметр «`require`». Он определяет доступ пользователей:

- значение «`valid-user`» определяет, что все аутентифицированные на контроллере домена пользователи имеют доступ;

- для ограничения списка допущенных пользователей следует задать значение параметра в виде списка их имен, разделенных символом пробела («require user-01 user-02» и т. п.).

Веб-сервер для аутентификации пользователей использует «Kerberos». Указанный в «host.conf» файл ключей «/etc/web.keytab» необходимо скопировать с сервера «ip», расположенного по аналогичному пути.

Взаимодействие веб-сервера и КСИ осуществляется по протоколу «AJP» (по определенному порту), который обеспечивает проксирование запросов к серверу КСИ и передачу запрашиваемой информации.

Настройка протокола «AJP» для КСИ осуществляется в файле «./tomcat/conf/server.xml», например:

```
<Connector URIEncoding="UTF-8" port="8009" protocol="AJP/1.3"
redirectPort="8443"/>
```

Для того, чтобы ограничить возможность доступа пользователей / программ непосредственно к серверу КСИ и использовать только веб-сервер «Apache2» следует:

- средствами сетевого экранирования ОС «Astra Linux SE» закрыть порт «AJP» для всех, кроме веб-сервера командой «iptables»:

```
iptables -A INPUT -p tcp ! -s apache-srv.vniief.loc --dport 8009
-j DROP
```

- добавить исполнение этого правила при загрузке ОС – в файле «/etc/network/if-up.d» создать файл «iptables»:

```
#!/bin/sh
iptables -A INPUT -p tcp ! -s apache-srv.vniief.loc --dport 8009
-j DROP
```

- добавить файлу атрибут «исполняемый»:

```
chmode +x /etc/network/if-up.d/iptables
```

- в файле «/tomcat/conf/server.xml» удалить строки, определяющие порты доступа («8080» и т. п.) за исключением «AJP»:

```
<Connector URIEncoding="UTF-8" port="8009" protocol="AJP/1.3"
redirectPort="8443"/>
```

4.5. Запуск и остановка программы

Для запуска КСИ на сервере «ip» в директории «/etc/systemd/system» расположен файл «ip.service».

Структура файла представляется в виде шаблона:

```
[Unit]
Description=APP_NAME
After=network.target

[Service]
Type=forking
ExecStart=APP_PATH/START_CMD_FILE
ExecStop=APP_PATH/STOP_CMD_FILE
User=SERVICE_USER
WorkingDirectory=APP_PATH

[Install]
WantedBy=multi-user.target
```

В приведенном шаблоне:

- «APP_NAME» – имя приложения (например, «ip»);
- «APP_PATH» – путь к каталогу с командами запуска и остановки приложения (например, «/opt/ip/tomcat/bin»);
- «START_CMD_FILE», «STOP_CMD_FILE» – команды запуска и остановки приложения («startup.sh» и «shutdown.sh» соответственно);
- «SERVICE_USER» – имя пользователя (владельца каталога «APP_PATH»), от которого выполняется запуск команд «START_CMD_FILE» и «STOP_CMD_FILE» (например, «ip»).

Для регистрации сервиса запуска КСИ выполняются команды («$FILE» - полный путь до файла описания сервиса, в рассматриваемом случае - «/etc/systemd/system/ip.service»):

```
chown root:root "$FILE"
chmod 0777 "$FILE"
chmod a+r "$FILE"
systemctl daemon-reload
```

После регистрации сервиса запуск КСИ осуществляется командой:

```
sudo systemctl start ip
```

Остановка КСИ осуществляется командой:

```
sudo systemctl stop ip
```

В случае изменения настроек конфигурационных файлов КСИ требуется его перезапуск. При этом перед запуском рекомендуется удалить содержимое каталогов:

- «./tomcat/logs»;
- «./tomcat/temp»;
- «./tomcat/work».

В процессе работы приложения ведется журнал, отражающий результаты основных системных событий.

Файлы журнала КСИ расположены в каталоге «./tomcat/logs».

Сообщения журнала представлены следующими основными типами:

- «INFO» – информационные сообщения;
- «WARNING» – предупреждение;
- «ERROR» – сообщения об ошибке.

5. ПРОВЕРКА ПРОГРАММЫ

Для работы с КСИ следует в адресной строке браузера ввести адрес вида:

<протокол>://<адрес КСИ>:<порт>/ip

Адрес веб-приложения определяется на этапе развертывания и передается пользователям КСИ средствами, определенными в регламентах взаимодействия служб предприятия.

В случае ввода корректной адресной информации в окне браузера отобразится интерфейс пользователя КСИ (рис. 3).

Главная страница КСИ

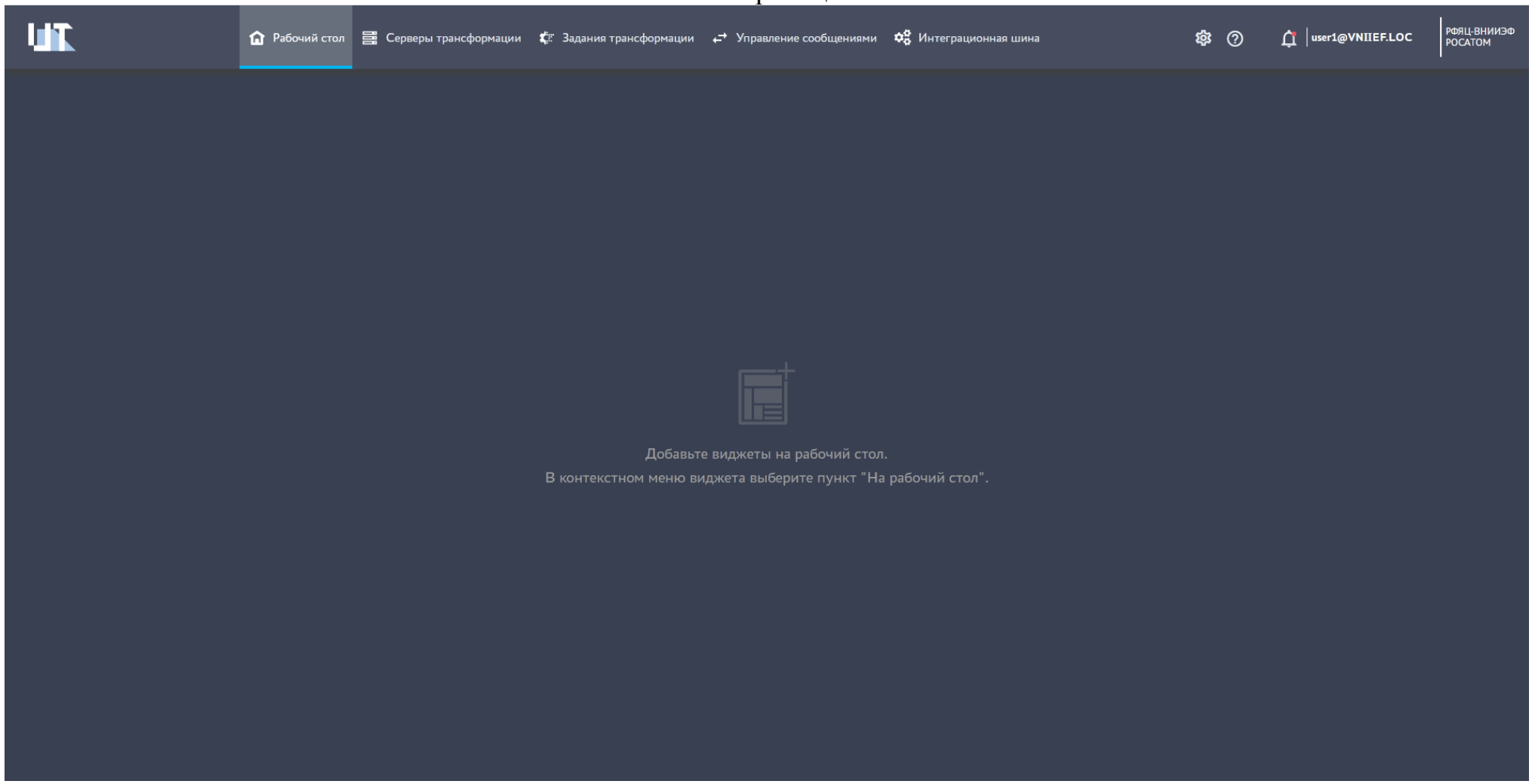

Рисунок 3

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

API	–	(англ. application programming interface) - программный интерфейс приложения
ETL	–	(от англ. Extract, Transform, Load — дословно «извлечение, преобразование, загрузка») - процесс управления хранилищами данных
HTML	–	(от англ. HyperText Markup Language — «язык гипертекстовой разметки») - стандартизированный язык разметки веб-страниц
JSON	–	(англ. JavaScript Object Notation) — текстовый формат обмена данными, основанный на JavaScript
XML	–	(eXtensible Markup Language) - расширяемый язык разметки
БД	–	база данных
КСИ	–	программный модуль «Комплекс средств интеграции»
ОС	–	операционная система
ПО	–	программное обеспечение
СИ	–	средства интеграции
СИА	–	программный модуль «Система информационного анализа»
СУБД	–	система управления базами данных
СУБД «Синергия-БД»	–	программный модуль «Система управления базами данных «Синергия-БД»

Лист регистрации изменений								
Изм.	*Номера листов (страниц)*				*Всего листов (страниц) в докум.*	*№ документа*	*Подп.*	*Дата*
	измененных	*замененных*	*новых*	*аннулированных*				