

# **Программа «Система управления взаимодействием с оборудованием и внешними ресурсами “Forward SP”»**

Правообладатель ООО «Форвард-Телеком»

Forward SP (Service Provisioning) – является промежуточной платформой между информационными системами, в частности автоматизированными системами расчетов, системами управления, системами подписки средств массовой информации (СМИ), облачных сервисов, таких как SaaS, и управляемыми ими ресурсами (например, коммутационным оборудованием, хостингом, подписками, виртуальными машинами и т.д.). Система поддерживает распределенные и составные команды и обеспечивает обратную связь, повторение запросов, получение статуса исполнения и первичную обработку полученных данных. Основное назначение системы - организация эффективного взаимодействия между информационной системой и ресурсами при изменении состояния услуг, уровней, вариантов или параметров доступа клиентов.

Использование системы будет максимально полезно для операторов связи, СМИ, SaaS и т.д., которые хотят получить быструю и надежную систему предоставления клиентам управляемых услуг. Система поддерживает кластерное масштабирование и может использоваться в распределенных сетях крупных федеральных и региональных СМИ, операторов связи и других информационных систем.

Контактная информация: [support@fw-t.ru](mailto:support@fw-t.ru)

# 1 Распространение программного обеспечения

Дистрибутив комплекта программ распространяется на физических носителях (USB-flash накопителе) или по ссылке на облачное хранилище.

Дистрибутив Forward SP является частью OSS (все актуальные дистрибутивы могут располагаться вместе с OSS).

Дистрибутив состоит из набора компонентов, дополнительных модулей расширения функционала, распространяемых библиотек.

Дистрибутив формируется для каждого заказчика с учетом технологической экосистемы и необходимых модулей расширения функционала

Подробная документация поставляется после адаптации системы в процессе внедрения с учетом особенностей кастомизации.

## 2 Содержание дистрибутива программного обеспечения

Перечень устанавливаемых дополнительных компонентов осуществляется в соответствии со спецификацией поставки. Дистрибутивы в комплекте поставки содержат компоненты SP:

1. Java-модуль;
2. Java библиотеки;
3. дополнительные модули (плагины).

Таким образом, дистрибутивы модуля представляют собой архивы вида forward-oss-platform-\*.zip и обеспечивают в дальнейшем корректную установку Forward SP.

Быстрая разархивация: `ls|xargs -n 1 unzip`

## 3 Порядок установки программного обеспечения

Ввиду вариативности используемого технологического стека на стороне заказчика и как следствие, – невозможности создания единой инструкции по установке – установку, первичную настройку программного обеспечения комплекса выполняют специалисты Forward.

Оборудование (сервер) должен удовлетворять требованиям в соответствии с приложением к договору (требования к аппаратной части сервера зависят от предполагаемой нагрузки сервера).

Программное обеспечение системы непрерывно развивается и с каждой версией дополняется новым функционалом. Обновление всех или отдельных программных компонентов системы, а также добавление дополнительных модулей расширения функционала производится специалистами Forward.

Специалистами Forward проводится групповое обучение специалистов Заказчика по разворачиванию и настройкам системы в целях обеспечения её дальнейшей корректной эксплуатации.

### 3.1 Подготовка к установке

1. Перед установкой Forward SP необходимо убедиться, что на целевом сервере установлена JDK актуальной версии ([cdn.azul.com/zulu/](https://cdn.azul.com/zulu/)):  
Актуальная версия на начало 2023 года – azul zulu 17
2. Для дальнейшей работы обязательно завести отдельного пользователя, под которым будут устанавливаться, а в последствии и запускаться все приложения Forward SP. Например, пользователь: oss  
При том, если это необходимо, на той же машине допускается развернуть тестовый контур. Для этого следует завести отдельного тестового пользователя. Например, тестовый пользователь: oss-test

### 3.2 Установка Forward SP

Порядок инсталляции Forward SP представляет собой последовательное выполнение действий по:

1. Установке Java-модуля SP
2. Установке Java-библиотек
3. Настройке Java-модуля SP
4. Установке плагинов (опционально)

### 3.3 Порядок установки и настройки модулей SP

#### 3.3.1 Установка Java-модуля

Java-модуль SP первоначально копируется, а затем архив распаковывается в каталог вида /usr/local/oss6/sp или любой другой, предназначенный для установки

модулей (при определении пути установки необходимо соблюдать наследование прав для пользователя).

### 3.3.2 Установка Java-библиотек

Останавливаем процесс `./sp.sh`

Распаковываем содержимое предоставленного архива в каталог вида `/usr/local/oss6/sp`, после чего однократно запускаем `./sp-db-sync-once.sh`

### 3.3.3 Настройка Java-модуля SP

Скрипт `service-control.sh` оставляем без изменения.

Скрипт `service-env.sh` содержит конфигурацию, которую нужно изменить в соответствии с актуальными данными.

В каталоге вида `/usr/local/oss6/sp/classes/com/forward/cfg/` содержится файл `sp.properties`, который так же хранит в себе настройку процесса.

### 3.3.4 Опциональные действия

#### 3.3.4.1 Ручное управление процессами SP

Возможны ситуации, когда необходимо вручную управлять процессами SP.

Запуск Java-модуля SP выполняется из директории вида `/usr/local/oss6/sp/bin` командой:

```
./sp.sh start
```

Остановка выполняется командой:

```
./sp.sh stop
```

#### 3.3.4.2 Установка плагинов

Для использования плагинов необходимо (без изменения существующих настроек `sh` и `properties`) обновить сборки всех процессов и плагинов или добавить в текущую установку новый.

В случае выполнения установки плагина вручную:

1. Останавливаем SP (`./sp.sh stop`)
2. Архив с плагином распаковываем в папку действующего SP
3. Разово запускаем `./sp-db-sync-once.sh`
4. Запускаем SP (`./sp.sh start`)

Если указано отдельно, необходимо установить модули из состава дистрибутива `oss`, например:

- `sp-plugin-rest-*.zip`
- `oss-platform-web-*.zip`

Дополнительно, каждый плагин содержит конфигурацию, настройка которой может потребоваться.

Указания по установке и настройке плагинов передаются отдельно.

#### **3.3.4.3 Обновление java библиотек**

При замене библиотеки необходимо удалить старую версию из каталога вида /usr/local/oss6/sp, после чего остановить процесс ./sp.sh и однократно запустить ./sp-db-sync-once.sh

#### **3.3.4.4 Автозапуск SP при включении сервера**

Реализация позволяет автоматически, при старте сервера, запускать расположенный на нём программный комплекс OSS.

При автозапуске скрипт проверяет соединение с БД, после чего запускает процесс SP. Если соединение с БД отсутствует, то скрипт автозапуска выполняет повторную проверку через 10 минут ожидания, после чего, если коннекта по-прежнему нет, запускает процесс SP в автономном режиме.

#### **3.3.4.5 Замена модуля SP**

Для замены модуля, нужно остановить сервис SP командой ./sp.sh stop

Удалить старую версию, скопировать новую и запустить процесс командой ./sp.sh start

## 4 Эксплуатация системы

### 4.1 Общие сведения

SP может быть развернут как на физических серверах, так и в облачной среде с предоставлением сетевого доступа к необходимым элементам инфраструктуры Заказчика – оператора связи различного масштаба. В период пуско-наладки специалистам Forward требуется предоставление доступа на сервер системы.

Специалистами Forward может быть проведено групповое обучение специалистов Заказчика по администрированию и эксплуатации системы с предоставлением необходимых инструкций в условиях реализации конкретного проекта.

Администрирование Forward SP производится эксплуатирующей организацией с привлечением, при необходимости, специалистов технической поддержки и разработчиков Forward.

Контроль работоспособности системы в целом и корректности процессов осуществляется за счет использования функциональности мониторинга SP, формирующегося из набора метрик JMX и инструментов аудита ошибок (парсинга логов). Используются метрики типа Gauge/Timer/Meter, которые отражают признак работы процессов, показывают, сколько время тратится на каждую из операций и количество операций.

Система Forward SP обеспечивает:

1. взаимодействие Решения и коммутационного оборудования посредством использования SP при изменении состояния услуг или параметров доступа абонента;
2. приём на вход управляющих ресурсами команд;
3. отправка управляющих ресурсами команд во внешнюю систему;
4. получение ответов от внешней системы;
5. контроль результатов выполнения команд на оборудовании;
6. протоколирование действий пользователей по отправке команд;
7. использование стандартизированных модулей (плагинов) управления оборудованием.

Система Forward SP состоит из следующих функциональных блоков:

1. Блок получения и подтверждения обработанных команд управления (интеграционный слой взаимодействия с Forward Billing или любой другой системой);
2. Блок исполнения команд (ядро).

Команды исполняются на экземплярах оборудования. Каждый экземпляр оборудования привязан к типу оборудования, а для типа оборудования, в свою

очередь, указывается управляющий модуль. В зависимости от способа управления, блок выполнения команд вызывает разные модули управления, реализующие протокол управления оборудованием. Для типа оборудования задается его имя, внешний идентификатор и способ управления (исполнитель или модуль управления), а все настройки хранятся непосредственно в экземплярах оборудования (IP-адрес оборудования, порт, логин и пароль, чтобы туда попасть и т.д.).

Forward SP предоставляет API для подключения внешнего решения, которое обеспечивает работу оператора в веб-интерфейсе. Основными возможностями оператора, использующего инструменты GUI при интеграции веб-интерфейса с предоставляемым API являются:

- управление командами на оборудование;
- повторная отправка команд;
- редактирование команд и отправка модифицированной команды на повторное выполнение в случае ошибки при её первом исполнении;
- просмотр списка выполненных и ожидающих команд с возможностью выставления условий фильтрации;
- просмотр команд, требующих ручного выполнения и обеспечение возможности подтверждения её выполнения в ручном режиме;
- ведение каталога вендоров оборудования;
- управление типами и экземплярами оборудования.

## 4.2 Блок получения и подтверждения команд

Блок получения и подтверждения команд, сформированных с помощью подсистемы выгрузки команд во внешние системы, забирает сформированные команды, обрабатывает их и записывает ответ.

Реализуется отдельно для каждой системы, с которой предусмотрена интеграция.

### 4.2.1 Формат команд

Каждая команда представляет собой управляющую последовательность, которая сформирована в представление, понятное конечному оборудованию. По умолчанию в базовой поставке для этого используется простой JSON формат команды (полное описание формата — <https://www.json.org/json-ru.html>)

В случае ответа может быть сформирована ответная управляющая последовательность (зависит от модуля исполнения). Разбор ответной последовательности лежит вне сферы ответственности SP.

Для работы с внешними системами в качестве плагинов SP поставляется поддержка следующих форматов команд:

- JAX-WS 2.1 (Webservices) — см. <http://jax-ws.java.net/>



- Apache Thrift — см. <http://incubator.apache.org/thrift/>
- Apache Etch — <https://cwiki.apache.org/ETCH/home.html>

#### 4.2.2 Источники команд

Архитектура системы обеспечивает возможность интеграции SP в любое существующее решение и систему и обеспечивает поддержку различных источников команд.

Источниками команд могут являться (но не ограничиваясь):

- Используемая СУБД;
- АСР «ФОРВАРД 5.0»;
- Веб-сервисы (JAX-WS 2.1, Apache Thrift и Etch).

Используемый универсальный интерфейс получения команд с помощью веб-сервисов позволяет немедленно выполнить запрошенную команду на указанном оборудовании.

Для этого применяемая процедура `execute` получает данные по следующим входным параметрам:

1. `hardwareExtId` – код оборудования (или его внешний номер), в которое передается запрос;
2. `Login` – уникальный идентификатор абонента/сущности, с которой производится работа;
3. `Params` – список параметров, которые будут переданы в модуль обработки, полностью зависит от используемого модуля;
4. `externalId` (опционально) – внешний код команды, возвращается в ответе на команду,

и передаёт соответствующие данные по таким выходным параметрам как:

1. `code` – один из возможных вариантов: 3 – команда выполнена успешно 10 – внутренняя ошибка в системе 11 – ошибка при проверке параметров команды 12 – ошибка при выполнении команды 13 – вызов не был произведен за отведенное ему время 14 – ошибка при проверке параметров железа/софта;
2. `uniqueId` – внутренний уникальный идентификатор команды;
3. `externalId` – внешний идентификатор команды, если он был получен;
4. `exceptionStackTrace` - трассировка исключения;
5. `executedAt` – дата реального выполнения команды;
6. `message` – сообщение. Зависит от используемого модуля.

7. Output – выходные параметры, сформированные модулем выполнения (могут быть не заполнены). Специфичны для каждого модуля обработки

#### 4.2.3 Настройка команд SP

Настройка команд SP, отправляемых на оборудование, выполняется за счет:

- Настройки услуг и шаблонов команд в биллинге;
- Изменения настроек SP в конфигурационных файлах (содержатся в директории вида: /usr/local/oss6/classes/com/forward/cfg).

Примечание: для применения выполненных изменений в настройках необходим перезапуск модуля SP вручную.

#### 4.3 Блок исполнения команд

Блок исполнения команд сформирован набором модулей управления.

Каждый модуль управления полностью независим от всех остальных подсистем и самостоятельно работает с известным ему оборудованием, а также имеет уникальный код, по которому происходит привязка к команде (через экземпляр оборудования и его тип).

Все необходимые модули управления разнообразным оборудованием и ПО разрабатываются по мере необходимости и/или поставляются и включаются в блок исполнения команд в зависимости от условий проекта.

Так, модуль “sh” (вызов Shell скриптов и программ), Код 2, из состава базовой поставки передаёт оборудованию в переменных окружения (либо командной строке) все параметры, которые ему пришли в управляющей команде.

Обязательные параметры экземпляра оборудования – путь к файлу или название скрипта, который выполняется. Переменные окружения при вызове команды заполняются всегда. Встроенные переменные (могут быть подставлены в командную строку и переданы в переменную окружения):

- логин пользователя;
- IP-адрес экземпляра оборудования;
- порт;
- название оборудования;
- путь;
- логин;
- пароль;

Дополнительные переменные имеют те же названия, что поступают к ним из команды с установленным префиксом.

## 4.4 Модули SP

Модули (плагины) SP обеспечивают включение нового оборудования в список поддерживаемого без изменения функциональности SP и оформляются в виде JAR-библиотек.

Для каждого плагина на основании документации на оборудование разрабатываются отдельные структуры для совершения обмена данных с оборудованием. На основе этих структур плагины генерируются отдельные последовательности данных, отправляемые в оборудование и также получаемые от оборудования последовательности, которые зависят от типа оборудования.

Базовая поставка SP может быть сформирована и дополнена включением перечня готовых плагинов для:

1. Синхронизации данных между двумя БД;
2. Вызова веб-сервисов MD Gateway;
3. Отправки APDU на SIM-карту абонента;
4. Принудительной отправки настроек APN;
5. Интеграции с Bridgewater AAA посредством XML API;
6. Интеграции с CONAX в части формирования команд на оборудование и дальнейшей выгрузки в файлы (в директорию оборудования);
7. Интеграции с веб-сервисами Dr. Web в части создания абонента, смены параметров абонента и удаления абонента;
8. Интеграции с веб-сервисом MNO (SOAP-интерфейс), в том числе в части обработки асинхронных методов;
9. Взаимодействия с провайдерами, предоставляющих услуги IP-телевидения и использующих NetUP IPTV Comple;
10. Взаимодействия с провайдерами, использующих STALKER IPTV;
11. Управления системой подписки Abel Quintic Back Office System;
12. Управления системой подписки Conax Contengo Web Services;
13. Управления приставками через API COMMPASS;
14. Управления LanBilling с помощью MySQL;
15. Управления DHCP сервером по OMAPI;
16. Отправки SMS через http шлюз MNO;
17. Отправки SMS по протоколу SMPP;
18. Отправки команд на HLR (Ericsson, Huawei);

19. Отправки команд в СУБД;

20. Отправки команд на платформы Protei;

Кроме того, Forward SP предусматривает возможность использования таких модулей расширения, как:

1. Интеграционная шина расширений – для приема сообщений из разных источников, отправки сообщения для его обработки в несколько модулей и результатов выполнения вызывающей системе, выполнения в один момент времени только одной транзакции с участием нужного оборудования;
2. Модуль расширения оборудования – обеспечивающий возможность выполнения опроса текущего состояния оборудования для того, чтобы любую команду, отправленную в сторону SP, можно было отменить;
3. Модуль расширения выполнения – реализующий расширение команды на основе заданных шаблонов и условий.