
Read the Docs Template Documentation

Выпуск 1.9

Read the Docs

мар. 03, 2022

1	Назначение документа	1
2	Методология внедрения	3
2.1	Получение ресурсов (боевого и тестового стенда)	3
2.2	Установка и настройка системы	3
2.3	Кастомизация процессов по факту выявленных пожеланий	3
2.4	Обучение сотрудников работе в системе	4
2.5	Опытная эксплуатация	4
2.6	Запуск в промышленную эксплуатацию	4
3	Инструкция по установке Arta Synergy Service	5
3.1	Предварительные требования	5
3.2	Подключение репозитория	5
3.3	Установка приложения	5
4	Инструкция по первичной настройке	7
4.1	В конфигураторе	7
4.2	Разграничение доступа к услугам	8
4.3	В пользовательской подсистеме SYNERGY	8
5	Структура продукта	9
5.1	Как все устроено	10
5.2	Список блокирующих процессов (БП)	11

Назначение документа

Целью создания данного документа является предоставление информации о конфигурации продукта, в частности, предоставление детализированного описания:

- созданных объектов
- произведенных настроек
- предполагаемого процесса работы в системе согласно назначению продукта.

Данный документ также может содержать дополнительные инструкции по возможным сценариям кастомизации существующей функциональности, советы и рекомендации.

2.1 Получение ресурсов (боевого и тестового стенда)

Характеристики серверов:

- Хранилище Jackrabbit
- Поисковые индексы Lucene
- 16 Гб ОЗУ
- Debian GNU/Linux (jessie, wheezy)
- CPU 8-core
- HDD/SSD с 10Гб свободного дискового пространства.

Предупреждение: Настоятельно рекомендуем развернуть тестовый стенд, полностью дублирующий конфигурацию системы боевого сервера, для диагностики и воспроизведения потенциальных проблем, тестирования обновления и изменений конфигурации.

2.2 Установка и настройка системы

Воспользуйтесь инструкциями данного документа для установки ([Инструкция по установке](#)) системы на боевом и тестовом стендах.

2.3 Кастомизация процессов по факту выявленных пожеланий

1. Для выявления пожеланий по изменению стандартной конфигурации продукта, устанавливаемой по-умолчанию, рекомендуется провести демонстрацию всем заинтересованным лицам проекта, в

том числе, конечным пользователем: сотрудникам первой и второй линии.

2. Все пожелания/замечания к продукту должны быть задокументированы и проанализированы на сложность и сроки адаптации продукта под данные требования.
3. В качестве вспомогательного инструмента к определению необходимых настроек в системе для реализации тех или иных пожеланий, рекомендуется ознакомиться со [Структурой продукта](#), а также [Руководством разработчика](#)

Предупреждение: Настоятельно рекомендуем провести полное тестирование всех процессов, после произведения настроек конфигурации.

2.4 Обучение сотрудников работе в системе

1. Все сотрудники организации, которым в рамках проекта предполагается обучение, могут быть объединены в группы согласно осуществляемым ими ролям (диспетчеры, исполнители и т.д.)
2. Рекомендуем адаптировать руководство пользователя, представленное по-умолчанию, согласно произведенным изменениям конфигурации, оргструктуре компании и прочей специфики проекта.
3. В первую очередь, должны быть обучены сотрудники, исполняющие основные роли в процессах, попадающих под опытную эксплуатацию

2.5 Опытная эксплуатация

1. Процесс опытной эксплуатации представляет собой имитацию полноценной работы системы в боевом режиме, но в меньших масштабах. Это может быть прогон основных боевых процессов на выделенном подразделении или на определенной категории документов.
2. Продолжительность процесса опытной эксплуатации зависит от масштабов проекта и количество внедряемых одновременно процессов.
3. Результатом опытной эксплуатации должен являться подтвержденный всеми заинтересованными сторонами факт готовности системы к использованию в промышленном режиме.
4. В случае выявления замечаний в ходе опытной эксплуатации, должны быть произведены соответствующие настройки в системе и измененные процессы должны быть протестированы повторно.

Примечание: Рекомендуем адаптировать руководство пользователя, представленное по-умолчанию, согласно произведенным изменениям конфигурации, оргструктуре компании и прочей специфики проекта.

2.6 Запуск в промышленную эксплуатацию

Инструкция по установке Arta Synergy Service

3.1 Предварительные требования

Важно! Для работы системы требуется:

- установить Arta Synergy 4.1 minsky ([инструкция по установке](#))
- в подсистеме администрирования обновить БД и процессы.

3.2 Подключение репозитория

Установочный пакет находится в репозитории *product-stable*. В файле `/etc/apt/sources.list` необходимо добавить либо раскомментировать следующую строку:

```
deb http://deb.arta.kz/tengri product-stable main contrib non-free
```

3.3 Установка приложения

Для установки приложения необходимо выполнить команду:

```
# apt-get update
# apt-get install arta-synergy-apps-service
```

В ходе установки необходимо выбрать тип установки из предложенного списка

Примечание:

1. Вариант **Clean** полностью удаляет все ранее установленные данные Synergy, устанавливает приложение Synergy Service, автоматически загружает его конфигурацию и загружает дампы базы данных.

2. Вариант `Install/Upgrade Configuration` устанавливает приложение Synergy Service, автоматически загружает его конфигурацию и настраивает необходимые конфигурационные файлы.
 3. Вариант `Manual` устанавливает приложение Synergy Service без автоматической загрузки конфигурации (т.е. только заменяет `war`-файлы).
-

Дальнейший ход установки зависит от выбранного типа установки:

- для варианта `Clean`:
 - указать URL сервера Synergy, на котором производится работа (например: `http://192.168.0.187:8080/Synergy`)
 - ввести пароль пользователя, имеющего root-права mysql, от имени которого будут применены дампы базы данных
 - после завершения установки обязательно нужно выполнить полную переиндексацию форм, файлов и документов в Административном приложении Synergy.
- для вариантов `Install/Upgrade Configuration` и `Manual`:
 - указать URL сервера Synergy, на котором производится работа (например: `http://192.168.0.187:8080/Synergy`):
 - ввести последовательно логин и пароль Системного пользователя:

Внимание: Для предотвращения дальнейшей нечаянной установки нестабильных версий пакетов из репозитория `unstable`, после установки пакета `arta-synergy-apps-service` рекомендуется закомментировать этот репозиторий в файле `/etc/apt/sources.list`.

После установки приложения обязательно выполнить пункты ([Инструкция по первичной настройке](#))

Инструкция по первичной настройке

4.1 В конфигураторе

1. Внести организационную структуру.

Обязательно должны присутствовать сотрудники с ролями:

- Диспетчер
 - Исполнитель
2. В организационной структуре обязательно должна присутствовать должность Клиент с кодом `service_clients`
 4. Добавить пользователей в группы `service_admin_group` и `service_client_group` и/или настроить и разграничить права на реестры по потребности (как минимум, должен быть доступ на просмотр для реестра `service_registry_service`).
 5. Создать формы и реестры на виды услуг в папке `orders` в конфигурации
 6. Для реестров `orders` создать фильтры реестров (например, по статусам заявок), по которым будут автоматически сформированы фильтры в личном кабинете на портале. Рекомендуется использовать условие «содержит текущего пользователя».

Примечание: Не рекомендуется использовать созданные для примере `order1...order6` так как при обновлении ваши изменения на этих формах могут быть потеряны.

7. Создать формы завершения (В дополнительных настройках конфигуратора), ссылаясь на созданные формы: Форма завершения заявки, Форма принятия в работу и Форма подтверждения заявки (формы созданные по-умолчанию расположены в папке `orders`).
8. Проверить корректность указанных форм завершения (для автоматического перехода на следующий этап все формы завершения имеют опцию «не требовать подтверждения») для этапов в маршрутах реестра.
9. Заполнить справочник Режимы рабочего времени и реестр Режимы работы.

10. Для реестров `orders` при необходимости настроить блокирующие процессы (пример настроен для `service_registry_order_1`):
- `event.blocking.interpreter.completion_time`: копирует время завершения с ФЗ заявки; с учетом режима обслуживания `service_form_order_timemode` вычисляет затраченное на выполнение время (в часах) в `service_form_order_spenttime`; в `service_form_order_overdue` записывает, просрочено ли решение.
 - `event.blocking.interpreter.set.8wh.timer`: заполняет дату таймера `service_form_order_timerDate` в 8 рабочих часов от времени запуска.
 - `event.blocking.interpreter.closingby.timer`: закрывает работу подтверждения пользователем решения по заявке, если наступает таймер - т.е. пользователь за отведенное время не ответил.
 - `event.blocking.interpreter.email.auth.notification`: отправляет уведомление о смене статуса на емэйл инициатора (установить в нужных местах маршрута);
 - `event.blocking.interpreter.matching`, `event.blocking.interpreter.matching.acceptexec`, `event.blocking.interpreter.matching.approve`, `event.blocking.interpreter.matching.executor`: сопоставление из форм завершения на форму заявки.
 - `event.blocking.interpreter.set.planFinishDateWM`: заполняет плановую дату `service_form_order_planFinishDate = service_form_order_date + service_form_order_duration` (время SLA в часах) с учетом режима обслуживания `service_form_order_timemode` (ссылается на режим работы из реестра *Режимы работы*).

4.2 Разграничение доступа к услугам

Для разграничения доступа к услугам на портале необходимо:

1. Дать нужным группам доступ на создание в реестре соответствующей услуги.
2. Дать нужным группам доступ на просмотр на фильтры соответствующих реестров (при необходимости создать фильтры).

4.3 В пользовательской подсистеме SYNERGY

1. Заполнить реестр **Услуги**, указав коды форм и реестров, созданных в конфигураторе.
2. Заполнить реестр **Группы услуг** (`service_registry_service_group`) и выбрать услуги, которые будут входить в группы. Услуги, которые не будут входить ни в одну группу, не будут отображаться на портале.
3. Заполнить реестры **Типы активов**, **Активы**, **Исполнители**, **Локации**.

Структура продукта

Содержание

- *Структура продукта*
 - *Как все устроено*
 - *Список блокирующих процессов (БП)*
 - * *event.blocking.interpreter.email.auth.notification*
 - * *event.blocking.interpreter.closingby.timer*
 - * *event.blocking.interpreter.completion_time*
 - * *event.blocking.interpreter.matching*
 - * *event.blocking.interpreter.matching.acceptexec*
 - * *event.blocking.interpreter.matching.approve*
 - * *event.blocking.interpreter.matching.executor*
 - * *event.blocking.interpreter.matching.firstapprove*
 - * *event.blocking.interpreter.set.8wh.timer*
 - * *event.blocking.interpreter.set.planFinishDate (участием.)*
 - * *event.blocking.interpreter.status_0, event.blocking.interpreter.status_1,...*
 - * *event.blocking.interpreter.set.planFinishDateWM (ноб.)*

5.1 Как все устроено

Система Synergy Service реализована на базе платформы ARTA SYNERGY, путем создания конфигурации, допускающей гибкую настройку объектов: форм, маршрутов, статусов и пр. . .

В системе Synergy Service реализованы следующие глобальные процессы и их подпроцессы:

- Портал самообслуживания
- Диспетчеризация заявок
- Исполнение заявок
- Управление каталогами услуг, активов, исполнителей

В конфигураторе приложение SynergyService состоит из следующих папок:

- **assets - Активы и Локации**
 - Справочник видов активов
 - Формы, реестры Активов
 - Формы, реестры Локаций
- **clients - информация по клиентам**
 - Форма и реестр карточки клиента
- **completion_forms - формы завершения**
 - Формы завершения заявки, форма принятия в работу и подтверждения завершения заявки
- **interpreter_scripts - скрипты для автоматической смены статусов и работы с формами завершения**
- **orders - формы и реестры заявок**
 - Папки на каждую услугу, содержащие форму и реестр (в конфигурации по-умолчанию - их 6)
 - Счетчик и шаблон номера
 - Справочник статусов исполнения заявок
- **performers**
 - Форма и реестр карточек исполнителей
 - Форма и реестр навыков исполнителей
 - Форма и реестр полей исполнителей
- **services - Услуги**
 - Форма и реестр услуг
- **Группы:**
 - SynergyService - группа для объединения всех реестров
 - service_admin_group - группа пользователей с полным доступом
 - service_client_group - группа пользователей на портал (клиенты)

В конструкторе приложение SynergyService состоит из следующих страниц:

- **auth_page - Авторизация**

- `main_page` - Главная
- `my_orders` - Мои заявки
- `registration_page` - Регистрация

5.2 Список блокирующих процессов (БП)

5.2.1 `event.blocking.interpreter.email.auth.notification`

Блокпроцесс отправляет уведомление автору заявки. По умолчанию в уведомлении указаны: - номер; - дата и время регистрации; - услуга; - статус; - плановое время завершения.

5.2.2 `event.blocking.interpreter.closingby.timer`

Данный БП закрывает заявку, если за отведенное время (например, 8 рабочих часов) инициатор не подтвердил завершение или не отправил эту заявку на доработку.

5.2.3 `event.blocking.interpreter.completion_time`

БП вычисляет (в зависимости от режима работы `service_form_order_timemode`) затраченное исполнителем время и была ли просрочена заявка.

5.2.4 `event.blocking.interpreter.matching`

БП осуществляет сопоставление данных из формы завершения `work_completion_form_orders` (ФЗ исполнителем заявки) на основную форму заявки.

5.2.5 `event.blocking.interpreter.matching.acceptexec`

БП осуществляет сопоставление данных из формы завершения `work_form_completion_accept` (ФЗ принятия в работу) на основную форму заявки.

5.2.6 `event.blocking.interpreter.matching.approve`

БП осуществляет сопоставление данных из формы завершения `work_completion_form_approve` (ФЗ подтверждения результата решения заявки) на основную форму заявки при завершении.

5.2.7 `event.blocking.interpreter.matching.executor`

БП осуществляет сопоставление данных из формы завершения `work_completion_form_executor_choose` (ФЗ выбора исполнителя заявки) на основную форму заявки.

5.2.8 `event.blocking.interpreter.matching.firstapprove`

БП осуществляет сопоставление данных из формы завершения `work_completion_form_approve` (ФЗ подтверждения результата решения заявки) на основную форму заявки при согласовании.

5.2.9 event.blocking.interpreter.set.8wh.timer

БП устанавливает таймер (на определенное количество рабочих минут, по умолчанию на 480). Этот таймер нужен для запуска БП event.blocking.interpreter.closingby.timer

5.2.10 event.blocking.interpreter.set.planFinishDate (устаревш.)

БП заполняет плановое время решения по заявке в зависимости от выбранного (одного из двух - календаря рабочего времени в конфигураторе и круглосуточного) режима работы (service_form_order_timemode) - .

5.2.11 event.blocking.interpreter.status_0, event.blocking.interpreter.status_1,...

Эти блокпроцессы меняют статус заявки после определенного этапа маршрута.

5.2.12 event.blocking.interpreter.set.planFinishDateWM (нов.)

БП заполняет плановое время решения по заявке в зависимости от выбранного справочного режима работы (service_form_order_timemode). Значение справочника соответствует записи реестра *Режимы работы*, где хранятся данные по рабочему времени для будних дней (пн-пт), для выходных (сб-вс) и для праздничных дней.