

СИСТЕМА ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА
ТЕХНИЧЕСКАЯ АРХИТЕКТУРА

Листов 11

2016

СОДЕРЖАНИЕ

1.	ТЕРМИНЫ, СОКРАЩЕНИЯ И ОПРЕДЕЛЕНИЯ.....	3
2.	ВВЕДЕНИЕ	4
2.2.	Цель	4
2.3.	Краткое описание возможностей СЭДО	4
3.	ТЕХНИЧЕСКАЯ АРХИТЕКТУРА СЭДО.....	5
4.	СЕРВИСЫ МОДУЛЯ «ФЕДЕРАЦИЯ».....	9
4.1.	Обмен данными по документам.....	9
4.2.	Синхронизация справочников	10
4.3.	Диагностика	10

1. ТЕРМИНЫ, СОКРАЩЕНИЯ И ОПРЕДЕЛЕНИЯ

В настоящем документе применены следующие сокращения и термины с соответствующими определениями:

Термин/сокращение	Определения
СЭДО\Система	Система электронного документооборота

2. ВВЕДЕНИЕ

2.2. Цель

Данный документ содержит описание технической архитектуры СЭДО.

2.3. Краткое описание возможностей СЭДО

Система электронного документооборота является многофункциональным программно-техническим комплексом, обеспечивающим автоматизацию управления деловыми процессами в условиях распределенного использования информации.

СЭДО обеспечивает комплексную автоматизацию служб документационного управления в части обработки управленческой документации, а именно - процессов создания, согласования, утверждения, регистрации, хранения и движения управленческих документов, а также контроля исполнения резолюций и поручений.

В настоящее время СЭДО позволяет обрабатывать следующие виды документов:

- организационно-распорядительные документы;
- входящие документы;
- исходящие документы.

В следующих версиях Системы перечень видов обрабатываемых документов может быть расширен.

Модуль «Федерация» обеспечивает распределенное взаимодействие между различными СЭДО, достигая следующих целей:

- единообразие: схема взаимодействия между организациями стандартизирована
- масштабируемость: модуль «Федерация» позволяет пересылать огромные потоки данных
- безопасность: все каналы передачи данных в модуле «Федерация» надежно защищены от внешнего вмешательства
- гибкость: архитектура модуля «Федерация» не привязана к конкретной программной платформе
- надежность: в силу распределенного характера модуля «Федерация» отказ одного из сегментов сети не влияет на функционирование других.

3. ТЕХНИЧЕСКАЯ АРХИТЕКТУРА СЭДО

Архитектура СЭДО представлена на схеме ниже (Рисунок 1).

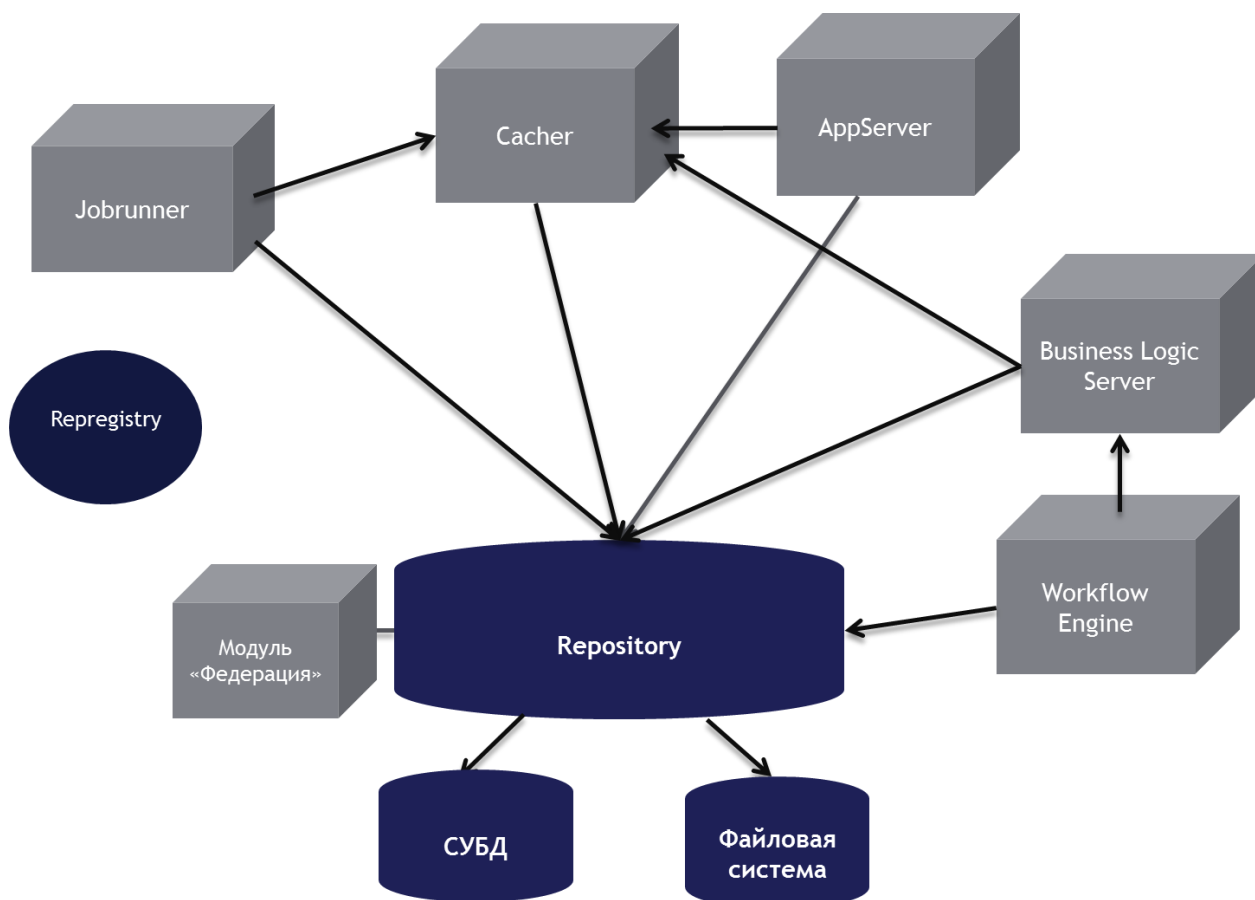


Рисунок 1. Техническая архитектура СЭДО

Описание технических компонент системы приведено ниже (Таблица 1).

Таблица 1. Описание компонентов системы

Компонент	Назначение
СУБД Oracle XE	Система управления реляционными базами данных
Файловая система	Система, предназначенная для хранения контента
Repository	Управляет репозиторием, обеспечивает полный спектр служб управления содержанием, включая контроль доступа к содержимому, контроль версий, workflow, выписку и запись в хранилище, для управления содержанием и процессами в масштабе распределенного предприятия. Обеспечивает возможность хранения, управления и использования всех типов содержания включая HTML и XML, графику, мультимедиа и традиционные документы, созданные при помощи десктоп-приложений
AppServer	Сервер приложений. Предназначен для обработки основной бизнес-логики приложения и обслуживания пользовательского интерфейса

Компонент	Назначение
Cacher	Предназначен для оптимизации Web приложений. В любом приложении встречаются медленные операции (SQL запросы или запросы к внешним API), результаты которых можно сохранить на некоторое время. Сервер кеширования позволяет выполнять меньше таких операций, а большинству пользователей показывать заранее сохраненные данные
Business Logic Server	Компонента, выполняющая простые методы (код СЭДО), автоматические действия и определяющая стадии жизненных циклов документов
Workflow Engine	Компонента, отвечающая за выполнение бизнес-процесса, бизнес-логика которого прописана в Business Logic Server
Registry	Компонента, хранящая адрес репозитория (репозитории, в случае использования распределенного документооборота)
Jobrunner	Включает в себя автоматически запускаемые операции
Модуль «Федерация»	Модуль, предназначенный для настройки распределенного документооборота

Со всеми вышеперечисленными компонентами пользователи взаимодействуют через AppServer посредством Web-приложения (Web-интерфейса).

Web-интерфейс - интерфейс, обеспечивающий доступ к репозиторию и службам управления содержанием из стандартного веб-приложения.

Для функционирования системы необходимо следующее техническое обеспечение:

- сервер системы управления документами (AppServer);
- сервер управления базой данных (СУБД Oracle XE);
- сервер Web-доступа.

Требования к вычислительным мощностям представлены в таблицах ниже:

Таблица 2. Требования к вычислительным мощностям СЭДО. Зона тестирования

Зона тестирования	
Параметр	Значение
Сервер приложений / Сервер содержания / Сервер поиска	
Процессор	4 CPU
Оперативная память	16 GB RAM
Дисковая подсистема	400 GB RAID5
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Прикладное ПО	Glassfish Server 3.1 Open Source Edition Repository
Сервер СУБД	
Процессор	2 CPU
Оперативная память	12 GB RAM
Дисковая подсистема	450 GB RAID5

Зона тестирования	
Параметр	Значение
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Прикладное ПО	Oracle Database XE

Таблица 3. Требования к вычислительным мощностям СЭДО. Зона эксплуатации

Зона эксплуатации	
Параметр	Значение*
Сервер приложений	
Процессор	8 CPU
Оперативная память	32 GB RAM
Дисковая подсистема	200 GB RAID5
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Прикладное ПО	Glassfish Server 3.1 Open Source Edition
Сервер содержания	
Процессор	8 CPU
Оперативная память	36 GB RAM
Дисковая подсистема	3000 GB RAID5(6), медленные диски SATA
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Прикладное ПО	Repository
Сервер поиска	
Процессор	6 CPU
Оперативная память	20 GB RAM
Дисковая подсистема	500 GB RAID10, быстрые диски SAS
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Сервер СУБД (желательно выделенный сервер)	
Процессор	2 x Intel 6-Core CPU
Оперативная память	48 GB RAM
Дисковая подсистема	500 GB RAID10, быстрые диски SAS
Операционная система	RedHat Linux 6.3 x64 (Oracle Linux)
Прикладное ПО	Oracle Database XE

* - Должна быть возможность увеличения вычислительных мощностей

Операционную систему рекомендуется ставить в режиме full install, выбрав все доступные в дистрибутиве пакеты.

Рекомендуется создать следующие разделы при установке ОС Linux.

Сервер приложений:

/	12 GB
swap	8 GB
/tmp	8 GB
/var	4 GB
/home	6 GB
/boot	500 MB
/u01	160 GB (все оставшееся место)

Сервер содержания:

/	12 GB
swap	16 GB
/tmp	16 GB
/var	4 GB
/home	6 GB
/boot	500 MB
/u01	1450 GB (все оставшееся место)

Сервер поиска

/	12 GB
swap	16 GB
/tmp	12 GB
/var	4 GB
/home	6 GB
/boot	500 MB
/u01	450 GB (все оставшееся место)

Сервер СУБД:

/	12 GB
swap	24 GB
/tmp	4 GB
/var	4 GB
/home	6 GB
/boot	500 MB
/u01	450 GB (все оставшееся место)

Разделы /u01 необходимо организовывать под управлением LVM.

4. СЕРВИСЫ МОДУЛЯ «ФЕДЕРАЦИЯ»

4.1. Обмен данными по документам dsjb_FMessageOutSender

Переносит сообщения из папки /Office Utils/FederationOutMessages в брокер очередей (а содержимое в FTPS)

Принцип работы исходящей очереди:

Джоба берет все события в очереди, группирует их в цепочки, в рамках каждой цепочки событий сами события обрабатываются последовательно. Цепочки обрабатываются параллельно в разных потоках.

При старте джобы поднимается пул потоков, и каждый поток хватается цепочку и рассылает ее. Если размер пула потоков меньше размера цепочек в очереди - соответственно часть цепочек будет висеть до тех пор, пока не освободится один из потоков.

Неограниченно потоки увеличивать нельзя, есть ограничение сверху по возможностям сервера, базы данных, количеству запущенных процессов curl в операционной системе.

Есть возможность вмешаться в процесс выбора цепочки потоком - у всех событий есть атрибут - приоритет (dsi_priority).

Цепочки, у которых приоритет событий выше - берутся в обработку первыми. У всех приоритет по умолчанию - 0.

Задержки:

1. Если возникнет ошибка при рассылке события в цепочке, последующие события в цепочке не смогут разослаться. Ошибки делятся как правило на 2 типа:
 - o Ошибки транспорта - при невалидном состоянии серверов принимающей стороны (ActiveMQ или FTPS);
 - o Ошибки формирования исходящего пакета для отложенных событий - ряд событий являются лишь маркерами для очереди, которые говорят, что нужно сформировать другое событие в целевую площадку. Такие события:
 - ✓ FFinishWorkflowTaskMessage_v2 (порождает FFinishWorkflowTaskMessage);
 - ✓ FSendDocumentOnPolicyStateMessage_v3 (порождает FSendDocumentOnPolicyStateMessage);
 - ✓ FSendDocumentOnPolicyStateMessage_v4 (порождает FSendDocumentOnPolicyStateMessage_v2).

При формировании этих сообщений часто возникают ошибки формирования федеративного сообщения из-за неконсистентности справочника оргструктуры.

2. При корректной обработке сообщения, поток данных делится на 2 типа:
 - o метаданные - пересылаются в ActiveMQ;
 - o содержимое - пересылается в FTPS целевой площадке. Учитывая объем связанных документов, поток контента получается большой. Поэтому содержимое передается при первой рассылке, а при последующих только в случае внесения изменений.

dsjb_FMessageConsumer

Переносит федеративное сообщение из брокера очередей в папку /Office Utils/FederationInMessages целевого репозитория

dsjb_FMessageIncomingProcessor

Джоба-обработчик входящих сообщений. Забирает сообщения из папки /Office Utils/FederationInMessages. Группирует их на основании атрибута `dss_key` (идентификатор группы, включающий в себя идентификатора документа и целевой репозиторий (название)). Каждая группа сообщений обрабатывается в отдельном потоке.

4.2. Синхронизация справочников

dsjb_FastDictSync

Эта джоба формирует из справочников ОШС своей площадки федеративное сообщение и рассылает во все известные репозитории (которые прописаны в файле `docbases.properties`)

4.3. Диагностика

dsjb_SendTestMessage

Джоба формирует тестовое сообщение в репозитории, переданные в аргументе *docbase*.

Если аргумент не передан - сообщение формируется во все репозитории, прописанные в файле `docbases.properties`

Схема взаимодействия приведена ниже (Рисунок 2).

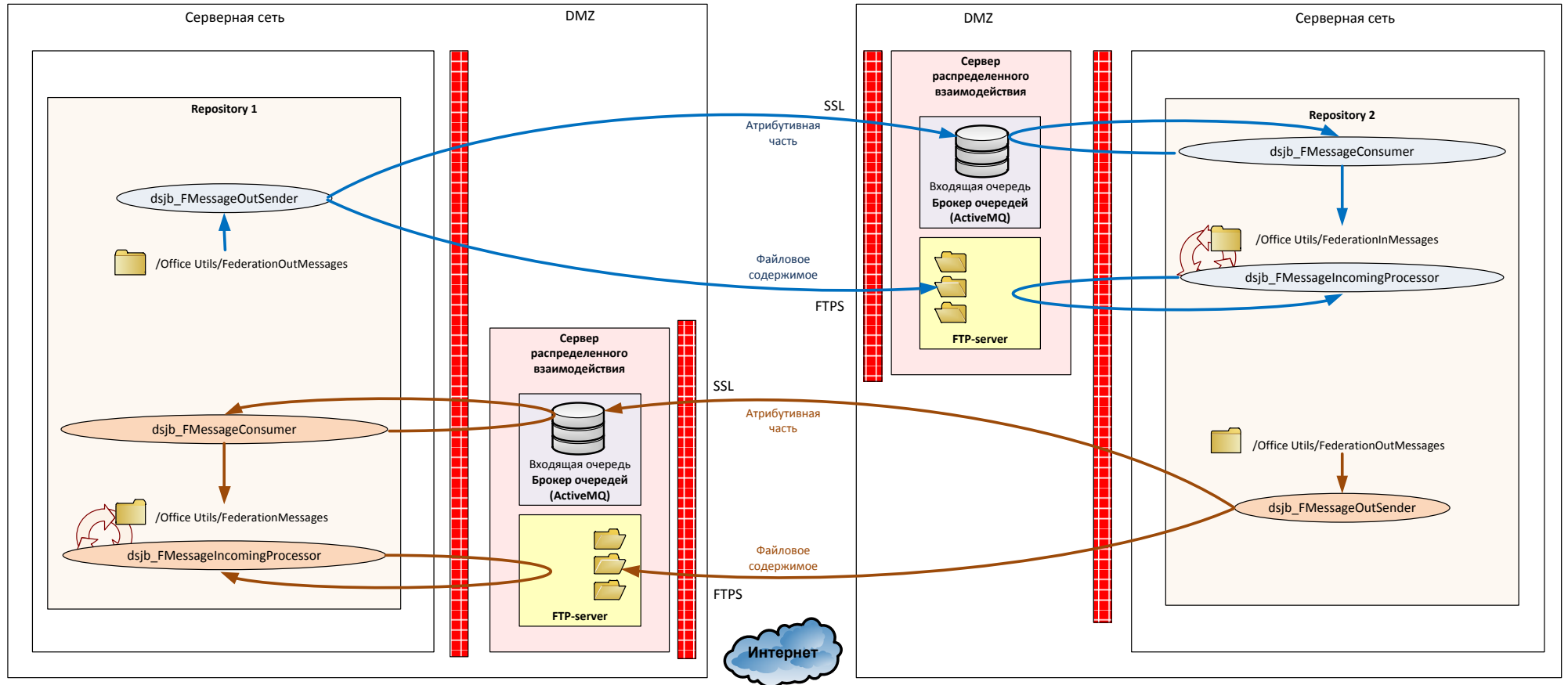


Рисунок 2. Схема обмена данных