

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ

Пунин Виктор Витальевич

Дипломная работа

Развитие системы многоканальной записи и речевой аналитики

Научный руководитель: Кандидат
физико-математических наук,
Ерёмин А. С.

Санкт-Петербург

2017

Содержание

Содержание	2
Введение	4
Постановка задачи	5
Глава 1. Анализ существующих решений	6
Глава 2. Схемы построения системы многоканальной записи и речевой аналитики	9
2.1. Локальное размещение	9
2.2. Типовой сетевой комплекс	10
2.3. Распределённый сетевой комплекс	12
Глава 3. Автоматические телефонные станции	14
3.1. Цифровые АТС	14
3.2. IP АТС	15
Глава 4. Направления развития	16
Глава 5. Интеграция с CUCM	17
5.1. Постановка задачи	17
5.2. Требования	17
5.3. Исследование возможных решений	17
5.4. Особенности реализации	18
5.5. Выводы	18
Глава 6. Разработка модуля интеграции данных	19
6.1. Постановка задачи	19
6.2. Требования	19
6.3. Общее описание реализуемой технологии	21
6.3.1. Загрузка в автоматическом режиме	21
6.3.2. Загрузка в ручном режиме	22
6.4. Анализ требований	22
6.5. Архитектура решения	24
6.5.1. Хранение состояния обработки файла	25
6.5.2. Организация процесса обработки файла	26

6.5.3. Импорт данных в БД.....	27
6.5.4. Реализация ограничения нагрузки	28
6.5.5. Мониторинг	28
6.5.6. Конфигурирование	30
6.6. Выводы.....	31
Заключение	33
Список используемых источников и литературы.....	34

Введение

Создание контакт-центра является наиболее распространённым, быстрым и эффективным средством повышения качества обслуживания клиентов компании.

Вместе с организацией контакт-центра появляется целый спектр не только новых возможностей, но и проблем, одной из которых является мониторинг качества обслуживания клиентов. Для решения этой задачи используются системы многоканальной записи и речевой аналитики.

В данной работе будут рассмотрены возможности программного обеспечения наиболее распространённых мировых разработчиков инновационных систем в сфере высококачественной записи, обработки и анализа аудио- и видеоинформации, синтеза и распознавания речи, а также, на примере *Smart Logger II* [1] компании ЦРТ [2], будут представлены возможные векторы развития продукта и примеры решений, применённых в рамках данной работы.

Постановка задачи

В рамках данной дипломной работы необходимо:

1. Провести анализ существующих систем многоканальной записи и речевой аналитики (РА).
2. На примере Smart Logger II выявить возможные направления развития существующей системы.
3. Рассмотреть примеры решения задач, связанных с развитием и поддержкой системы.

Глава 1. Анализ существующих решений

На рынке РФ систем многоканальной записи и речевой аналитики на май 2017 года основными игроками являются 3 компании: ЦРТ, NICE [3], VERINT [4].

На основе имеющейся информации на сайтах производителей, а также внутренней документации удалось сравнить системы данных производителей.

Параметр	ЦРТ	NICE	VERINT
Базовые характеристики			
Отечественный разработчик	+	-	-
Разработчик речевых технологий	+	-	-
Автоматическая оценка	+	+	+
Речевая аналитика	+	+	+
Экранная аналитика	+	+	+
Открытый программный интерфейс для интеграции	+	-	-
Консалтинговые услуги в области использования речевых систем для решения задач КЦ на русском языке	+	-	-
3-й уровень технической поддержки на русском языке 24/7 от разработчика системы	+	-	-
Возможность кастомизации от разработчика программного обеспечения	+	-	-
Базовые технологии распознавания речи			
Поиск ключевых слов и выражений с использованием технологии пофонемного распознавания речи	+	+	+
Распознавание спонтанной слитной русской речи с использованием LVCSR	+	+	+
Лингвистическая обработка результатов распознавания и индексация			
Временная разметка распознанного текста	+	+	+
Индексация семантически незначимых лексических единиц	+	+	+
Индексация устойчивых словосочетаний	+	+	+
Приведение частей речи к начальной морфологической форме	+	?	?
Выделение контекста ключевых и частотных слов	+	?	?

Индексирование частей речи	+	-	-
Технологии анализа текстовых документов			
Тематическая классификация документов по ключевым словам	+	+	+
Тематическая классификация с использованием обучающей выборки фонограмм, относящихся к данной теме	+	-	-
Тематическая кластеризация	+	-	-
Статистический анализ текстов	+	-	-
Визуализация результатов анализа			
Представление результатов распознавания в формате «плоского» текста	+	+	+
Представление результатов распознавания в формате диалога, с разбиением по репликам	+	-	-
Контекстная сводка по ключевым и частотным словам	+	-	-
Управляемое семантическое облако (переопределение центроида кластера, управление лексическим составом кластера, на уровне семантически значимых слов, частей речи и др.)	+	-	-
Визуализация динамики изменения тематических кластеров по дням, часам и операторам	+	+	+
Возможности автоматической оценки телефонных переговоров			
Количественно-временной анализ технологических событий в телефонии	+	+	+
Анализ речевой активности дикторов	+	+	+
Анализ ключевых слов и выражений в речи оператора и клиента	+	+	+
Оценка эмоционального состояния абонентов	+	+	+
Возможность создания комплексного шаблона оценки телефонных звонков на основе предыдущих 4 параметров	+	-	-
Автоматическое выявление повторных обращений клиентов	+	-	-
Автоматическое выявление неудовлетворенных клиентов (с использованием комплексного шаблона и семантического анализатора, обученного на обучающей выборке негативных звонков)	+	-	-
Лицензирование			

Лицензирование по каналам записи/рабочим местам	+	+	+
Лицензирование по объему анализируемых данных	+	-	-

Таким образом, можно сделать вывод о том, что возможности решений ЦРТ не уступают мировым лидерам в разработке решений для аналитики взаимодействий с клиентами, терять из внимания потребности которых — непозволительная роскошь. Исходя из необходимости внедрения системы многоканальной записи и речевой аналитики клиентам с различной инфраструктурой, и появляется необходимость в развитии системы.

В следующей главе будут рассмотрены существующие варианты построения системы многоканальной записи и речевой аналитики.

Глава 2. Схемы построения системы многоканальной записи и речевой аналитики

С учётом возможностей клиентов, существует несколько возможных схем построения системы.

2.1. Локальное размещение

При локальном размещении основные аппаратные и программные компоненты системы устанавливаются на одном компьютере, называемом станцией записи (рис. 1).

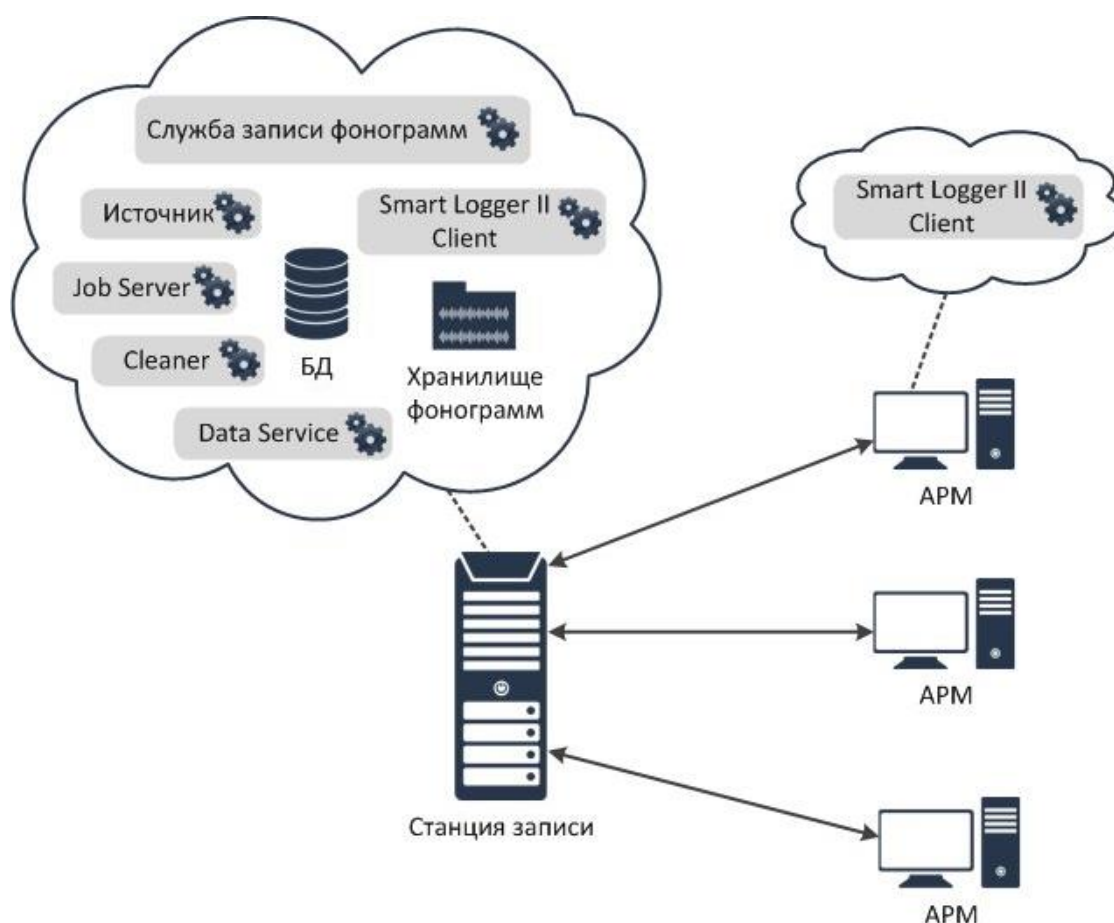


Рис. 1. Схема локального размещения системы

Основным программным компонентом, устанавливаемым на станции записи, является служба записи фонограмм. Служба записи фонограмм управляет работой источника — программного компонента, предназначенного для записи голосового трафика с различных источников

сигнала (аналоговые или цифровые телефонные линии, линии цифрового потока E1, каналы VoIP-телефонии и др.).

На станции записи размещается хранилище фонограмм, содержащее аудиозаписи телефонных разговоров, и база данных с информацией о них. Удаление фонограмм обеспечивается с помощью модуля *Cleaner*. Для работы с базой данных (БД) используется компонент *Job Server*, предназначенный для выполнения по расписанию некоторых служебных процедур, например, оптимизации пространства БД после удаления ненужных фонограмм и т. п.

Клиентская часть ПО (модуль *Smart Logger II Client*) может быть размещена на станции записи или на любом другом компьютере, имеющем доступ по локальной сети к станции записи (автоматизированном рабочем месте, *АРМ*). С помощью приложения *Незабудка II*, входящего в состав модуля *Smart Logger II Client*, пользователь получает доступ к БД и хранилищу фонограмм.

Приложение *Незабудка II* получает доступ к БД через сервис-посредник *Data Service*, а также позволяет осуществлять настройку работы службы записи фонограмм и других параметров системы.

2.2. Типовой сетевой комплекс

При размещении системы в качестве типового сетевого комплекса программные компоненты устанавливаются на нескольких компьютерах, включенных в локальную сеть (рис. 2).

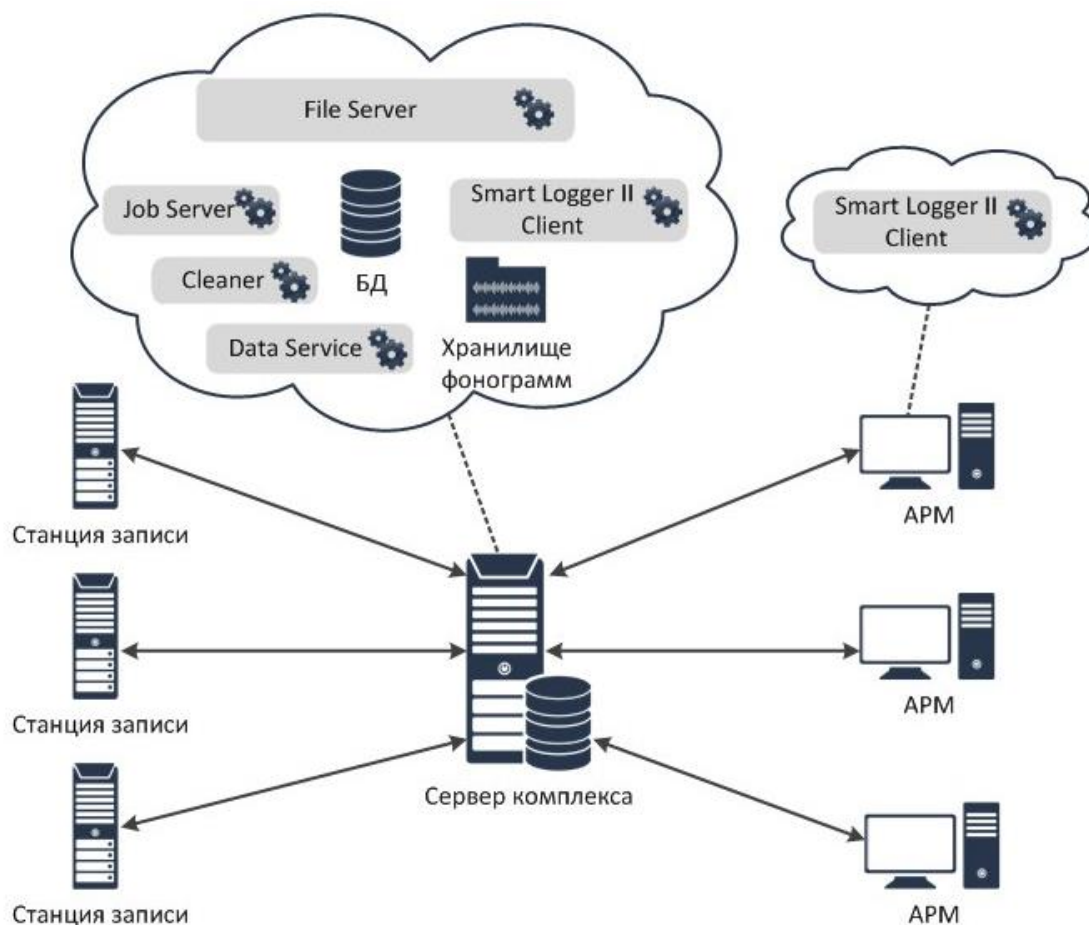


Рис. 2. Схема типового сетевого комплекса

Основной особенностью приведенной схемы является то, что единое хранилище фонограмм и единая база данных системы размещаются на компьютере, называемом сервером комплекса.

В комплексе может функционировать несколько станций записи, на каждой из которых имеется локальная БД и локальное хранилище записываемых фонограмм. Работа с данными на станции записи осуществляется так же, как при локальном размещении.

На сервере комплекса устанавливается модуль *File Server*, который обеспечивает копирование фонограмм и соответствующей информации из БД (этикеток фонограмм) со станций записи на сервер комплекса. Остальные программные компоненты, устанавливаемые на сервере комплекса, работают аналогично соответствующим программным компонентам, устанавливаемым на станции записи.

Клиентская часть ПО (модуль *Smart Logger II Client*) может быть размещена на сервере комплекса или на любом другом компьютере, имеющем доступ по локальной сети к серверу комплекса (*APM*). С помощью приложения *Незабудка II*, входящего в состав модуля *Smart Logger II Client*, пользователь получает доступ к данным, хранящимся на сервере комплекса.

Приложение *Незабудка II* получает доступ к БД через сервис-посредник *Data Service*.

Доступ модуля *Smart Logger II Client* к хранилищу фонограмм может осуществляться по протоколу HTTPS или путём предоставления общего доступа к каталогу фонограмм через ЛВС.

2.3. Распределённый сетевой комплекс

Вариант размещения системы в качестве распределённого сетевого комплекса подразумевает использование нескольких серверов, для которых существует единое централизованное хранилище (рис. 3).

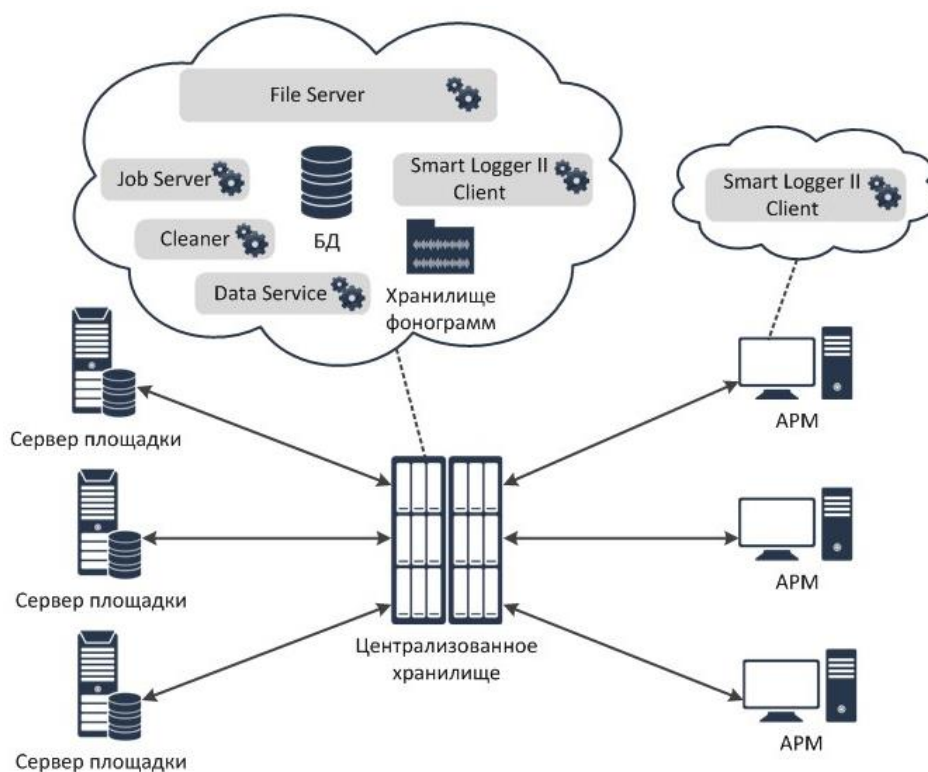


Рис. 3. Схема распределенного сетевого комплекса

Комплекс обслуживает несколько площадок, на каждой из которых имеется сервер с едиными для площадки фонограммами и базой данных. На сервере площадки размещаются те же компоненты системы, которые устанавливаются на сервере типового сетевого комплекса.

На компьютере, выполняющем функции централизованного хранилища, устанавливается модуль *File Server*, который обеспечивает копирование фонограмм и соответствующей информации из БД (этикеток фонограмм) с серверов площадок в централизованное хранилище. Остальные программные компоненты, устанавливаемые на компьютере, выполняющем функции централизованного хранилища, работают аналогично соответствующим программным компонентам, устанавливаемым на сервере типового сетевого комплекса.

Клиентская часть ПО (модуль *Smart Logger II Client*) может быть размещена на компьютере, выполняющем функции централизованного хранилища, или на любом другом компьютере, имеющем доступ по локальной сети к централизованному хранилищу (*APM*). С помощью приложения *Незабудка II*, входящего в состав модуля *Smart Logger II Client*, пользователь получает доступ к данным централизованного хранилища.

Приложение *Незабудка II* получает доступ к БД через сервис-посредник *Data Service*.

Доступ модуля *Smart Logger II Client* к хранилищу фонограмм может осуществляться по протоколу HTTPS или путём предоставления общего доступа к каталогу фонограмм через ЛВС.

Распределённый сетевой комплекс устанавливается сотрудниками ООО «ЦРТ» на площадке заказчика.

Глава 3. Автоматические телефонные станции

Автоматические телефонные станции (АТС) принято [5] разделять на несколько типов:

- Машинные,
- Декадно-шаговые,
- Координатные,
- Квазиэлектронные,
- Электронные аналоговые,
- Электронные цифровые,
- IP АТС.

Подробнее о каждом типе рассказано в [6]. Согласно статистике обращения клиентов, в настоящее время наиболее распространены последние два типа, количество аналоговых АТС мало, а иные — сохранены лишь в музеях и государственных корпорациях.

3.1. Цифровые АТС.

В случае цифровой АТС на машине дополнительно устанавливается PCI, или PCI-Express плата [7]. Если же нет возможности выделить машину под данные нужды, заказчику предлагается воспользоваться устройством Smart Logger Box [8].

Плата считывает с телефонной линии поток данных в цифровом виде и передает его на анализ специальным службам. Службы извлекают из потока данных звуковую и сигнальную информацию в соответствии с протоколом работы цифровой АТС.

В качестве основных протоколов выступают:

Протокол	Всего каналов	Каналов со звуком/данными	Каналов с сигнализацией	Пропускная способность
----------	---------------	---------------------------	-------------------------	------------------------

ISDN PRI E1	32	30	2	2048 Кбит/с
ISDN PRI T1	24	23	1	1553 Кбит/с
ISDN BRI	3	2	1	128 Кбит/с

Вышеперечисленные протоколы являются открытыми, документация по ним находится в свободном доступе. Так же производители разрабатывают и используют в своих АТС проприетарные протоколы, в качестве основы которого нередко выступает ISDN BRI. Среди таких производителей можно отметить *Alcatel*, *Siemens*, *Samsung* и другие. Однако в последние годы объем предложений цифровой телефонии сильно уступает IP и другим решениям.

3.2. IP АТС

В подобных ТС передача данных осуществляется с помощью IP-протокола, для каждой из них производителем, как правило, предусмотрено API. Основными производителями являются *Avaya*, *Genesis*, *Cisco*.

В одной из следующих глав будет рассмотрен пример подключения к *Cisco UCM* [9], представляющую собой операционную систему для IP телефонии на базе *Red Hat*.

Глава 4. Направления развития

На основании анализа возможностей конкурентов и беседы с руководителем отдела разработки многоканальных систем ЦРТ, были сформулированы основные направления развития системы:

1. Повышение производительности системы записи (увеличение максимального количества записываемых телефонных номеров).
2. Поддержка новых технологий записи и хранения (SIP-REC, SIP, использование ленточных хранилищ).
3. Поддержка новых поставщиков АТС.
4. Разработка и поддержка протоколов цифровых АТС.
5. Оптимизация пользовательского интерфейса, повышение удобства использования.
6. Интеграция с внешними системами управления взаимоотношениями с клиентами (CRM) [10] и корпоративными хранилищами данных (КХД) [11].

О нескольких работах по развитию существующей системы и пойдет речь в дальнейших главах.

Глава 5. Интеграция с CUCM

5.1. Постановка задачи

В рамках данной работы необходимо исследовать возможности СТИ-интеграции с интерфейсом Cisco Jtapi, а также создать модуль, позволяющий произвести интеграцию. Полученный модуль будет использован в качестве основы для плагина к существующему модулю записи.

5.2. Требования

1. Разрабатываемый модуль должен быть реализован на .Net, несмотря на то, что Jtapi, как следует из названия, реализован на Java.
2. Нагрузка на машину – минимальна.
3. Отсутствие недокументированных ошибок и падений.

5.3. Исследование возможных решений

Для решения поставленной задачи можно выделить несколько способов:

1. Разработать службу на java, которая будет подключаться к Cisco Jtapi. Далее из .Net подключаться к этой службе по RabbitMQ/WEB/REST/Socket и получать данные о событиях. В качестве быстрой и легковесной сериализации можно использовать JSON, Protocol Buffers.
 - + Надежность, проверенная временем.
 - Поддержка двух продуктов вместо одного.
2. Использование ikvm [12], утилиты, с помощью которой можно перевести .jar сборку в .dll и использовать все классы java из этой сборки как родные для C#. Фактически, можно взять Jtapi, собрать из неё пакет, а затем использовать его напрямую из клиентской части.
 - + Использование кода от Oracle, проверенного тестами и временем.
 - + Всего один продукт.

- Открытые вопросы со сборкой мусора.

После совещания с ведущим архитектором системы *Smart Logger*, было принято решение об использовании второго способа.

5.4. Особенности реализации

Для подключения к CUCM необходимо указать адрес, порт, логин и пароль. При создании устройства, за которым будет вестись наблюдение, можно назначить ему обозревателя, который обязан имплементировать интерфейс *CallControlCallObserver*. Лишь после этого события от АТС начнут приходить в клиентскую часть модуля.

Любое событие содержит в себе *Id*, который показывает (согласно [13]) какое именно событие произошло. Для высокопроизводительной системы это возможно и хорошее решение, но для первичного анализа полноты получаемой информации, необходимой для интеграции, это является совершенно неоптимальным. Благодаря рефлексии, предусмотренной *.Net Framework*, удалось без особых сложностей получить всю информации по приходящим событиям.

5.5. Выводы

Решение данной задачи послужило началом для разработки плагина к существующему модулю записи, что позволит в будущем расширить список поддерживаемых АТС ещё одним производителем — *Cisco*.

Глава 6. Разработка модуля интеграции данных

6.1. Постановка задачи

Необходимо создать механизм для загрузки бизнес-данных, сформированных по итогам диалогов сотрудников различных отделов, в систему речевой аналитики.

6.2. Требования

1. Импорт данных из файла формата «.csv».
2. Автоматическая загрузка данных в систему РА.
3. Возможность ручной загрузки данных для корректировки данных или для передачи данных, которые не смогли сохраниться ранее.
4. Мониторинг каталога на наличие новых готовых файлов для импорта данных. Мониторинг как файлов, содержащих данные для автоматического периодического импорта данных, так и для ручного импорта.
5. Сервис должен работать с каталогами как на локальной машине установки сервиса, так и с сетевыми каталогами по протоколу FTP. Дополнительно реализовать работу через сетевую папку.
6. Архивирование исходных файлов импорта данных в определенное хранилище. После импорта файлы с данными должны переноситься в отдельный каталог.
7. При архивировании файлы должны сжиматься в архив zip. Каждый файл в отдельный архив, название архива повторяет название файла.
8. Функция архивирования должна включаться/выключаться через конфигурацию работы сервиса. При выключенной функции файлы с данными должны просто удаляться после обработки.
9. Удаление исходных файлов импорта после обработки файла из каталога импорта.
10. Сохранение отчета по импорту в виде отдельного файла в определенном хранилище отчетов.

10.1. Файл отчета должен содержать информацию по результатам импорта данных для каждой фонограммы и по каждой строке данных.

10.2. Функция сохранения отчета должна включаться/выключаться через конфигурацию работы сервиса.

11. Конфигурация модуля. Реализация UI-конфигуратора для настройки:

11.1. Периода опроса каталога на появление новых файлов импорта;

11.2. Каталогов импорта, архивирования, сохранения журналов импорта;

11.3. Учетных записей для доступа к каталогам (при необходимости). По умолчанию доступ от учетной записи, под которой запущен сервис;

11.4. Доступа к БД системы РА;

11.5. Конфигурации числа импортируемых полей.

12. Сервис должен автоматически очищать каталоги хранения журналов импорта и каталоги хранения архивных файлов импорта данных от соответствующих файлов.

12.1. Должна быть возможность задать срок хранения файлов отдельно для каждого типа файлов.

13. Сервис должен поддерживать работу с большими файлами данных для импорта, размеров в несколько сотен тысяч строк. Сервис должен импортировать до 1 000 000 строк (набор данных для 1 фонограммы) в сутки.

14. При работе с большими файлами сервис не должен создавать излишней нагрузки на компьютер, должно быть реализовано ограничение потребляемых ресурсов, в том числе нагрузки на диск.

15. Поддержка на ОС Windows 7 и новее, а также MSSQL Server 2008 R2 и новее.

16. Все операции должны журналироваться в журналах событий соответствующего уровня (прикладной, ОС, СУБД).

16.1. Журналирование должно осуществляться на русском и английском языках (должны быть подготовлены шаблоны записей о событии)

17. Должен быть реализован мониторинг разрабатываемого компонента или разрабатываемой части функционала.

6.3. Общее описание реализуемой технологии

В корпоративном хранилище данных настраивается процесс ежедневного построения витрин по требуемым данным, из которых затем по расписанию формируются два файла выгрузок в формате «.csv».

Загрузка файлов из корпоративного хранилища данных в БД возможна в двух режимах: автоматическом и ручном.

6.3.1. Загрузка в автоматическом режиме

Автоматический режим является штатным режимом работы сервиса и предполагает следующие шаги.

1. Система КХД:

1.1. по расписанию формирует файлы, расширение которых не регламентировано,

1.2. по завершению формирования файла меняет расширение файла на «.csv»,

1.3. выгружает файлы КХД в FTP-каталог импорта системы РА.

2. Система РА:

2.1. периодически сканирует FTP -каталог импорта на предмет появления новых файлов,

2.2. загружает файлы КХД в БД *Smart Logger* и производит их обработку,

2.3. формирует файлы отчетов о загрузке в БД *Smart Logger* и помещает его в FTP-каталог с отчетами,

- 2.4. помещает файлы КХД в архив,
- 2.5. удаляет исходные файлы КХД из FTP-каталога импорта.

6.3.2. Загрузка в ручном режиме

Ситуация возможна в случае, если существует проблема в загрузке на стороне РА и тогда после ее устранения службой поддержки системы РА, загрузка ведется по следующему алгоритму:

1. файл КХД из архивного каталога в ручном режиме переносится в FTP-каталог импорта системы РА;
2. система РА:
 - 2.1. периодически сканирует FTP-каталог импорта на предмет появления новых файлов,
 - 2.2. загружает файлы КХД в БД *Smart Logger* и производит их обработку,
 - 2.3. формирует файл отчета о загрузке в БД *Smart Logger* и помещает его в FTP-каталог с отчетами, который будет отслеживаться системой мониторинга Банка,
 - 2.4. помещает файл КХД в архив,
 - 2.5. удаляет исходный файл КХД из FTP-каталога импорта.

Сроки хранения в каталоге FTP файлов КХД и отчетов о загрузке для всех режимов загрузок в БД – 30 дней.

6.4. Анализ требований

Исходя из требований заказчика можно сделать вывод о том, что необходимо разработать:

1. отказоустойчивый Windows-сервис,
2. способный загружать файлы из:
 - 2.1. локального хранилища,
 - 2.2. сетевой папки,

- 2.3. FTP-сервера,
3. в локальное хранилище,
4. производящий валидацию содержимого файла,
5. импортирующий содержимое полученных файлов в БД (MSSQL),
6. на основе произведенных действий, строящий отчеты,
7. выгружающий исходный файл и отчет в папку:
 - 7.1. локального хранилища,
 - 7.2. сетевой папки,
 - 7.3. FTP-сервера,
8. журналирующий свои действия,
9. отправляющий в мониторинг сведения о своём состоянии,
10. конфигурируемый в режиме настоящего времени,
11. практически не нагружающий:
 - 11.1. ПЗУ,
 - 11.2. ЦП,
 - 11.3. Сеть.
12. Кроме того, инсталляция сервиса должна быть под силу любому человеку, который хоть раз в жизни устанавливал программное обеспечение, а также в автоматическом режиме.

6.5. Архитектура решения

Язык	C#
Версия .NET Framework	4.0
Взаимодействие с базой	Singleton
Взаимодействие с мониторингом	Singleton
Взаимодействие с логгером	Singleton, log4net
Взаимодействие с системой загрузки-выгрузки файлов	Abstract factory
Обработка файлов	Template method
Инсталлятор	Wix
Тесты	Microsoft Unit Test Framework

Основной задачей сервиса является работа с файлами. Логически этот процесс можно разделить на несколько этапов:

1. сканирование каталога импорта,
2. загрузка новых файлов в локальное хранилище,
3. валидация содержимого полученных файлов, согласно схеме таблицы, в которую их содержимое будет загружено,
4. перекодировка содержимого (MSSQL не поддерживает utf-8, необходима utf-16),
5. импорт данных в БД,
6. ожидание окончания сопоставления данных с фонограммами,
7. выгрузка отчета о сопоставлении данных с фонограммами,
8. архивирование исходного файла,
9. архивирование отчета,
10. выгрузка архивированного исходного файла,
11. выгрузка архивированного отчета,
12. удаление локальных файлов,
13. удаление файла из каталога импорта,
14. удаление файлов из каталогов выгрузки, согласно времени их жизни.

Как видно, каждый этап зависит от предыдущего, кроме седьмого этапа, который зависит только от БД (именно она осуществляет сопоставление фонограмм и данных). Необходимо продумать как можно более универсальное решение, которое наиболее дешево в сопровождении.

6.5.1. Хранение состояния обработки файла

Существует лишь две объективных причины, из-за которых сервис перестанет работать: отключат питание компьютера, или пользователь выключит сервис. Чтобы предотвратить обработку файла с самого начала, а вместо этого, продолжить обработку именно с того этапа, на котором она была остановлена, необходимо предусмотреть такой параметр как «состояние файла»: файл в состоянии «валидирован» не нужно ещё раз загружать из директории импорта, ведь он уже загружен и проведена проверка на соответствие его содержимого схеме таблицы в БД. Существует несколько возможных вариантов:

Хранить локально в json или xml список обработанных файлов.

- + Простота доступа и поддержки изменений в коде.
- Дополнительные затраты на поддержание целостности данных.

Хранить в БД.

- + Низкие затраты на поддержание целостности данных.
- + Хорошее сочетание с хранением импортированных данных: всегда имеется информация о том, из какого файла и когда пришли данные.
- + Удобство в построении отчетов.
- Потеря связи с БД – блокирует работу сервиса, впрочем, это относится и к первому способу.

6.5.2. Организация процесса обработки файла

Учитывая строгий регламент обработки файла, можно сделать вывод о том, что необходимо предусмотреть абстракцию, согласно которой, объекты всех классов, имплементирующих её (будем называть их *рабочими*), будут уметь обрабатывать тот, или иной этап работы алгоритма. Таким образом появляется идея о следующем алгоритме обработки файлов. Его краткая схема представлена ниже (рис. 4).

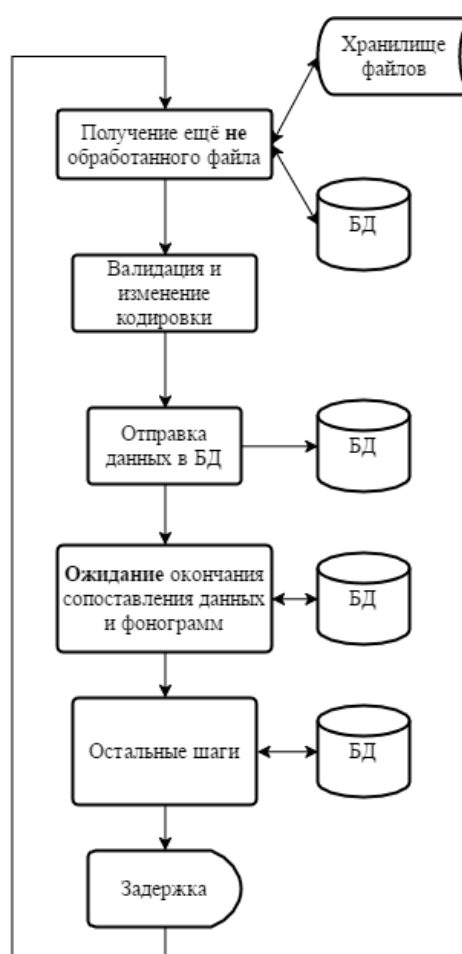


Рис. 4. Обработка файлов

В рамках данного алгоритма необходимо из необработанных файлов, ожидающих в папке с входящими данными, выбрать один (дольше остальных находящихся), и произвести с ним все необходимые действия. Затем, выбрать следующий файл. Основная проблема данного подхода заключается в том, что время ожидания окончания сопоставления данных и фонограмм не регламентировано, что грозит простоем рабочей машины. Для решения

данной проблемы пришлось изменить подход к обработке файлов следующим образом: теперь каждый *рабочий* принимает в качестве входных данных не файл, а состояние. Все файлы, находящиеся в этом состоянии, будут являться для него рабочим набором данных: каждый из них он обработает и изменит его состояние. Принципиальная схема работы представлена ниже (рис. 5.)

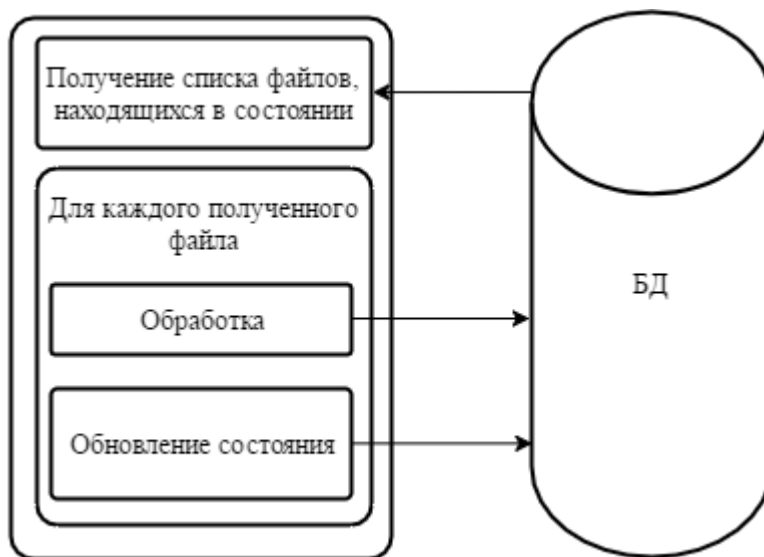


Рис. 5. Измененный режим работы рабочего

Управление подобными *рабочими* необходимо передать диспетчеру, который отловит все ошибки. Именно через него будет настраиваться очередь рабочих. Примененный подход можно считать модификацией паттерна «Chain of Responsibility» [14]. Кроме того, он полностью соответствует принципам SOLID разработки [15]. Таким образом, отсутствие файлов в состоянии «Сопоставлен» никаким образом не приостановит работы сервиса.

6.5.3. Импорт данных в БД

Осуществлять импорт данных в БД можно несколькими способами: *построчно, пакетами* и, наиболее интересным, *Bulk Insert* [16]. Для выбора наилучшего способа была проведена серия тестов. Рабочая машина включала в себя процессор Intel Intel® Core™ i5-7400, 8 Гб оперативной памяти, жесткий диск 6400 об/мин. Объем данных составлял 1 млн строк, в каждой по 30 полей,

разделенных точкой с запятой, состоящих из GUID [17]. Поля тестовой таблицы имели тип nvarchar(255).

Тест	Время, в среднем
Построчный импорт	5 минут 10 секунд
Импорт пакетами по 1000 строк	4 минуты 8 секунд
Импорт пакетами по 100 строк	1 минута 13 секунд
Импорт пакетами по 10 строк	35 секунд
Импорт пакетами по 5 строк	51 секунда
Bulk Insert	1 секунда

Лидирующую позицию занял последний метод. Он и был выбран в качестве основного. Согласно документации по Bulk Insert, скорость его обусловлена [18] использованием API нативного клиентского драйвера SQL.

6.5.4. Реализация ограничения нагрузки

Во все операции чтения, передачи, обработки данных была добавлена задержка в 1 мс, которую можно выключить в конфигураторе сервиса. Иного способа ограничения потребления ресурсов найти не удалось.

6.5.5. Мониторинг

Одним из плюсов разработки коммерческого программного продукта является написанные, протестированные, документированные и качественно выполняющие свою работу библиотеки, продукты и приложения. К таким относится приложение для мониторинга работы сервисов на рабочих станциях. Именно с ним и будет взаимодействовать будущий продукт. Для мониторинга были выбраны такие параметры как:

- доступность подключения к БД;
- доступность локальной папки;
- доступность папки с входящими файлами;
- доступность папки с архивом файлов;
- доступность папки с архивом отчетов.

Также, помимо стандартных счетчиков: загрузка ЦП, ОЗУ, ПЗУ, были введены следующие:

- количество обработанных файлов за сутки;
- количество обработанных строк за сутки;
- количество строк, не прошедших валидацию за сутки.

Введение мониторинга позволяет своевременно отследить проблемы, поэтому его присутствие в любом корпоративном сервисе необходимо. На изображениях ниже (рис. 6, рис. 7) представлено окно мониторинга.

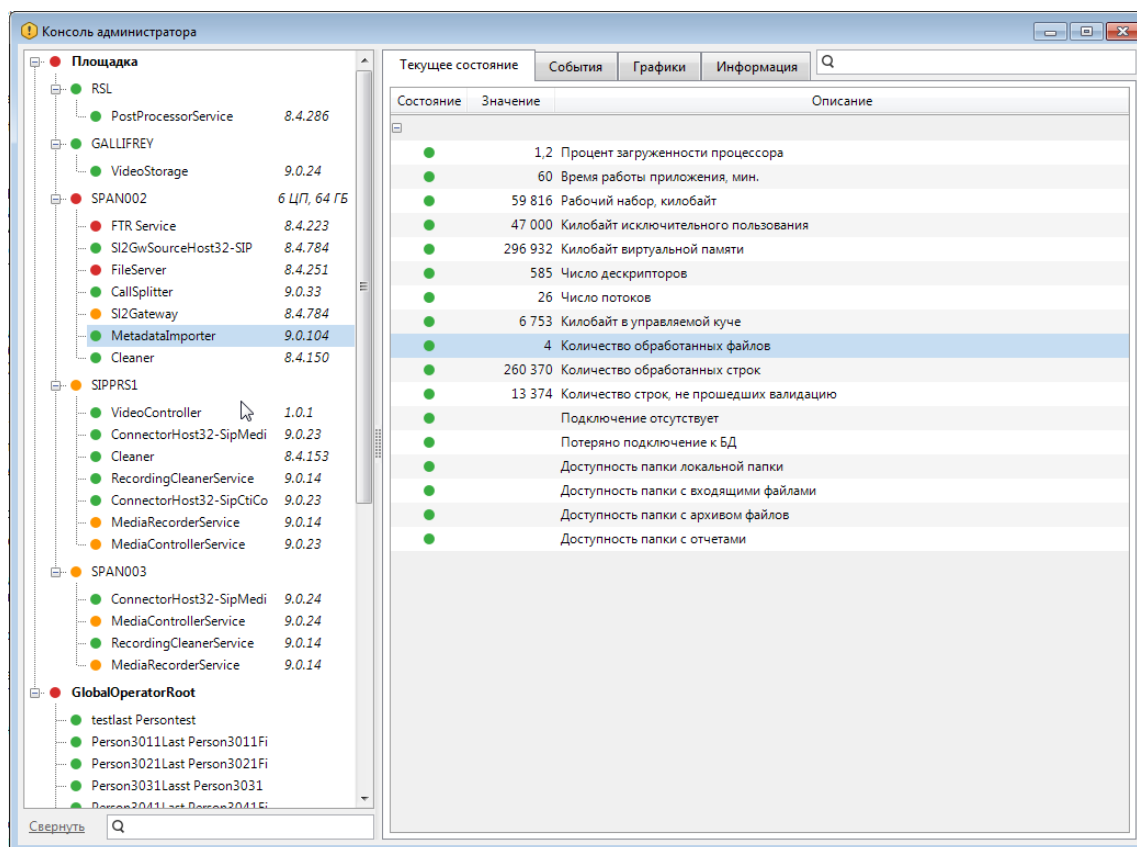


Рис. 6. Основное окно мониторинга

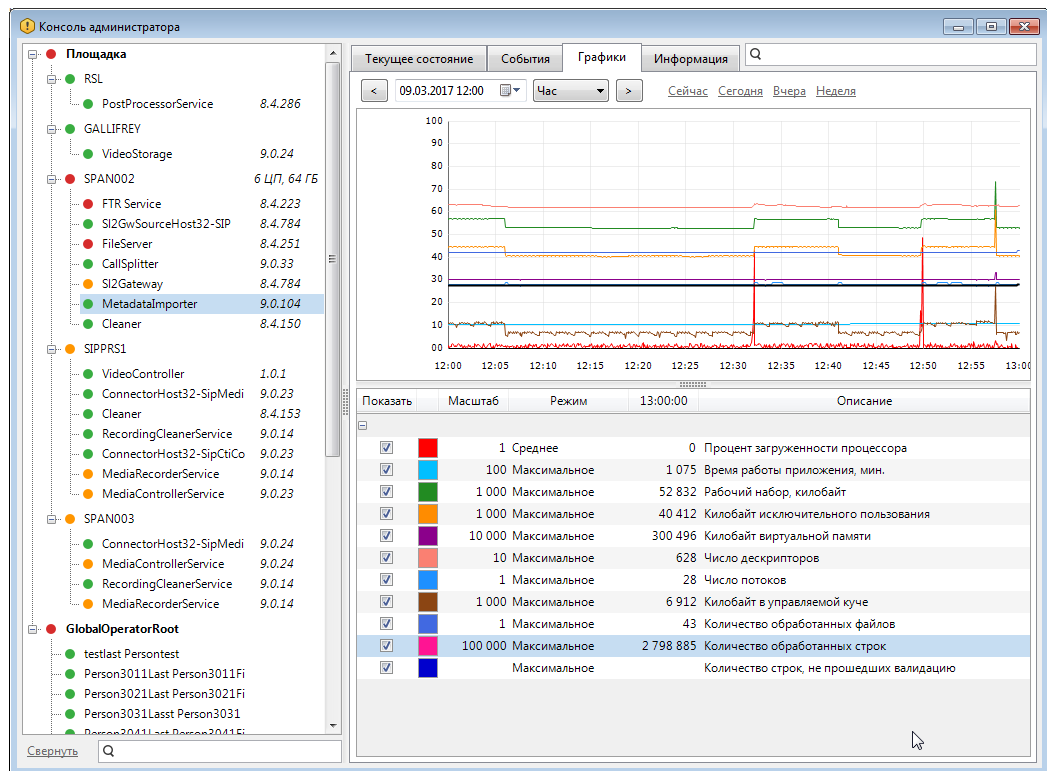


Рис. 7. Графики основных показателей

6.5.6. Конфигурирование

Для изменения конфигурации сервиса в режиме реального времени был также использован уже разработанный продукт. Внешний вид конфигуратора представлен на рисунке ниже.

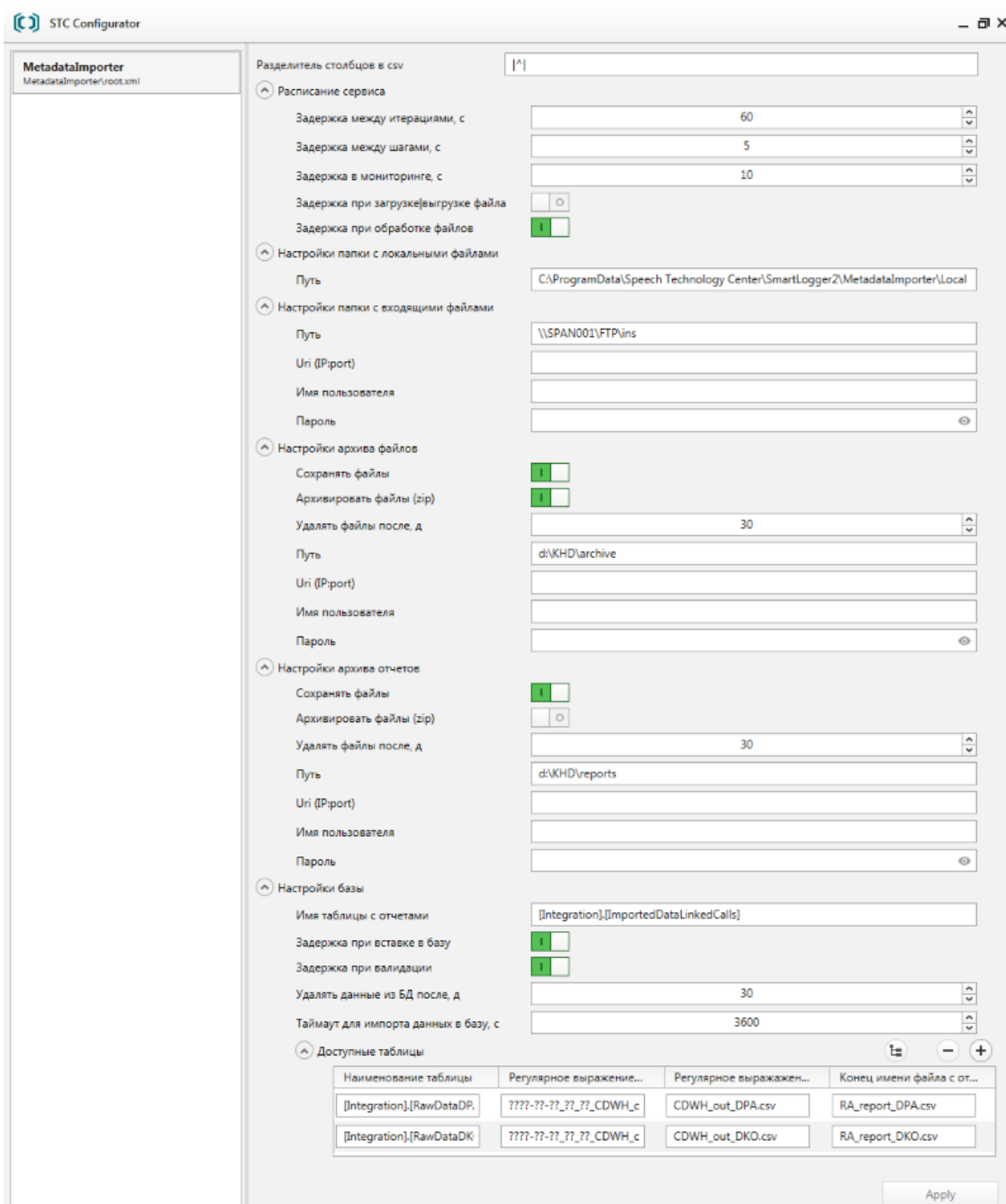


Рис. 8. Окно конфигуратора сервиса

6.6. Выводы

В рамках данной работы в системе *Smart Logger* появилось новое решение по интеграции с CRM-системами и корпоративными хранилищами данных. Благодаря гибкости настройки, интеграцию можно реализовать с любыми системами заказчика, поддерживающими экспорт в формат табличных данных («.csv»).

Импорт данных о вызовах в *Smart Logger* позволяет проводить речевую аналитику и оценку качества обслуживания с учётом дополнительных характеристик взаимодействия с клиентом, например, стажа и возраста клиента, номера заявки, факта продажи услуги и т. д.

Заключение

В данной работе были рассмотрены способы взаимодействия с современными АТС, существующие программные комплексы ведущих мировых производителей в области систем многоканальной записи и речевой аналитики, возможные направления развития имеющейся системы, а также раскрыты некоторые аспекты его программной реализации.

Каждая возникающая проблема была пристально изучена, а её решение теоретически и практически обосновано.

Полученные в результате данной работы приложения продемонстрировали стабильность своей работы на площадках заказчиков.

Список используемых источников и литературы

1. ЦРТ, «Smart Logger II - система автоматического контроля качества работы операторов и мониторинга удовлетворенности клиентов | ЦРТ» [Электронный ресурс]. URL: <http://www.speechpro.ru/product/sistemy-upravleniya-kachestvom-i-avtomatizatsii/smart-logger-2>. (Дата обращения: 20.05.2017).
2. ЦРТ, «О ЦРТ | Центр Речевых Технологий» , [Электронный ресурс]. URL: <http://www.speechpro.ru/about/>. (Дата обращения: 20.05.2017).
3. Nice, «Cross-Channel Recording,» [Электронный ресурс]. URL: <http://www.nice.com/engage/cross-channel-interaction-recording> . (Дата обращения: 20.05.2017).
4. Verint Systems, «Full-Time and Compliance Recording | Verint Systems» [Электронный ресурс]. URL: <http://www.verint.com/solutions/customer-engagement-optimization/enterprise-workforce-optimization/products/recording/> . (Дата обращения: 20.05.2017).
5. «Автоматическая телефонная станция» [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Автоматическая_телефонная_станция . (Дата обращения: 24.05.2017).
6. «Автоматические телефонные станции и классификация АТС» [Электронный ресурс]. URL: <http://vse-lekcii.ru/zheleznodorozhnyj-transport/ats/avtomaticheskie-telephonnye-stancii> . (Дата обращения: 25.05.2017).
7. ЦРТ, «Smart Logger II: Multi-channel call recording and monitoring system» [Электронный ресурс]. URL: <http://speechpro.com/product/voice-recording/smartlogger2#tab4> . (Дата обращения: 24.05.2017).
8. ЦРТ, «Многоканальный автономный сервер записи Smart Logger BOX - спецификации» [Электронный ресурс]. URL: <http://www.speechpro.ru/product/sistemy-zapisi-telefonnykh-razgovorov/smart-logger-box/specification>. (Дата обращения: 24.05.2017).

9. Cisco Systems, «Cisco Unified Communications Manager (CallManager) - Cisco» [Электронный ресурс]. URL: <http://www.cisco.com/c/en/us/products/unified-communications/unified-communications-manager-callmanager/index.html>. (Дата обращения: 25.05.2017).
- 10.«Система управления взаимоотношениями с клиентами — Википедия» [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Система_управления_взаимоотношениями_с_клиентами. (Дата обращения: 25.05.2017).
- 11.«Что такое корпоративное хранилище данных (Data Warehouse) и кому его продавать. Модель базы данных. Витрина данных. OLAP.» [Электронный ресурс]. URL: http://www.prj-exp.ru/dwh/what_is_dwh.php. (Дата обращения: 25.05.2017).
- 12.Frijters J., «IKVM.NET Home Page» [Электронный ресурс]. URL: <https://www.ikvm.net/>. (Дата обращения: 25.05.2017).
- 13.Cisco Systems, «CTI Server Message Reference Guide (Protocol Version 15) for Cisco Unified Contact Center Enterprise» 2010. [Электронный ресурс]. URL: http://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/cust_contact/contact_center/ctios/cti_server_reference/programming/guide/CTI_Svr_Msg_Ref_Gde_Prot_15.pdf. (Дата обращения: 25.05.2017).
- 14.Костюков. В., «Паттерн проектирования «Цепочка обязанностей» / «Chain of Responsibility» / Хабрахабр» [Электронный ресурс]. URL: <https://habrahabr.ru/post/113995/>. (Дата обращения: 25.05.2017).
- 15.«От STUPID кода к SOLID коду» [Электронный ресурс]. URL: <https://habrahabr.ru/post/273843/>. (Дата обращения: 25.05.2017).
- 16.Microsoft, «BULK INSERT (Transact-SQL)» [Электронный ресурс]. URL: <https://msdn.microsoft.com/ru-ru/library/ms188365.aspx>. (Дата обращения: 25.05.2017).

17. Microsoft, «Структура Guid (System)» [Электронный ресурс]. URL: [https://msdn.microsoft.com/ru-ru/library/system.guid\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.guid(v=vs.110).aspx) . (Дата обращения: 25.05.2017).
18. Microsoft, «Import and Export Bulk Data by Using the bcp Utility (SQL Server) | Microsoft Docs» [Электронный ресурс]. URL: <https://docs.microsoft.com/en-us/sql/relational-databases/import-export/import-and-export-bulk-data-by-using-the-bcp-utility-sql-server>. (Дата обращения: 25.05.2017).