Модуль SCPL.Core.AgentHelper

SberContact Platform

Exported on 2025-01-30 18:46:24

Table of Contents

1 Архитектура взаимодействия компонентов 5

2 Требования к записи экранов 8

3 Требования к реализации AgentHelper 9

4 Основные характеристики 12

5 Сообщения Web-Socket в сторону AgentServer 13

5.1 agentHelperInfo - редактирование 13

5.2 newRecBatch 14

5.3 resultRestartScreenRecord 16

5.4 resultStartScreenRecord 17

5.5 resultStartStream 19

5.6 resultStopScreenRecord 21

5.7 resultStopStream 23

5.8 streamBlob 24

6 Сообщения Web-Socket в сторону JsSDK 26

6.1 resultAgentOpenConnect 26

7 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 28

7.1 v1 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 28

7.1.1 1. Описание фичи 28

7.1.2 2. Описание процесса/сервиса TO BE 28

7.1.3 3. Описание запросов 30

7.2 v2 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 39

7.2.1 1. Описание фичи 39

7.2.2 2. Описание процесса/сервиса TO BE 39

7.2.3 3. Описание запросов 42

7.3 v3 R2.1.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 54

7.3.1 1. Описание фичи 54

7.3.2 2. Описание процесса/сервиса TO BE 54

7.3.3 3. Описание запросов 58

7.4 v4 R2.8.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 75

7.4.1 1. Описание фичи 76

7.4.2 2. Описание процесса/сервиса TO BE 76

7.4.3 3. Описание запросов 79

7.5 v5 R2.11 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии" 96

7.5.1 1. Описание фичи 97

7.5.2 2. Описание процесса/сервиса TO BE 97

7.5.3 3. Описание запросов 101

8 Фича Core.AgentHelper "Запись экрана оператора" 119

8.1 v1 Фича Core.AgentHelper "Запись экрана оператора" 119

8.1.1 1. Описание фичи 119

8.1.2 2. Описание процесса/сервиса TO BE 119

8.1.3 3. Описание запросов 121

8.2 v2 Фича Core.AgentHelper "Запись экрана оператора" 127

8.2.1 1. Описание фичи 128

8.2.2 2. Описание процесса/сервиса TO BE 128

8.2.3 3. Описание запросов 131

8.3 v3 R 1.11.0 Фича Core.AgentHelper "Запись экрана оператора" 136

8.3.1 1. Описание фичи 136

8.3.2 2. Описание процесса/сервиса TO BE 136

8.3.3 3. Описание запросов 140

8.4 v4 R 1.12.0 Фича Core.AgentHelper "Запись экрана оператора" 146

8.4.1 1. Описание фичи 146

8.4.2 2. Описание процесса/сервиса TO BE 146

8.4.3 3. Описание запросов 150

8.5 v5 R 2.8 Фича Core.AgentHelper "Запись экрана оператора" 158

8.5.1 1. Описание фичи 158

8.5.2 2. Описание процесса/сервиса TO BE 158

8.5.3 3. Описание запросов 162

8.6 v6 R3.2 Фича Core.AgentHelper "Запись экрана оператора" 170

8.6.1 1. Описание фичи 170

8.6.2 2. Описание процесса/сервиса TO BE 170

8.6.3 3. Описание запросов 176

9 Фича Core.AgentHelper "Остановка записи экрана оператора" 187

9.1 v1 Фича Core.AgentHelper "Остановка записи экрана оператора" 187

9.1.1 1. Описание фичи 187

9.1.2 2. Описание процесса/сервиса TO BE 187

9.2 v2 Фича Core.AgentHelper "Остановка записи экрана оператора" 191

9.2.1 1. Описание фичи 191

9.2.2 2. Описание процесса/сервиса TO BE 191

9.3 v3 R1.11.0 Фича Core.AgentHelper "Остановка записи экрана оператора" 195

9.3.1 1. Описание фичи 196

9.3.2 2. Описание процесса/сервиса TO BE 196

9.3.3 3. Описание запросов 198

9.4 v4 R1.12.0 Фича Core.AgentHelper "Остановка записи экрана оператора" 202

9.4.1 1. Описание фичи 203

9.4.2 2. Описание процесса/сервиса TO BE 203

9.4.3 3. Описание запросов 205

9.5 v5 R2.8 Фича Core.AgentHelper "Остановка записи экрана оператора" 208

9.5.1 1. Описание фичи 208

9.5.2 2. Описание процесса/сервиса TO BE 208

9.5.3 3. Описание запросов 211

9.6 v6 R3.2 Фича Core.AgentHelper "Остановка записи экрана оператора" 214

9.6.1 1. Описание фичи 214

9.6.2 2. Описание процесса/сервиса TO BE 214

9.6.3 3. Описание запросов 218

10 Параметры записи экранов, передаваемые из AgentServer в AgentHelper для использования в ffmpeg 222

11 Взаимодействие компонентов для записи экранов при back-to-back подключении 224

12 Архив Core.AgentHelper 225

12.1 архив-resultGetDisplayInfo 225

13 Фича Core.AgentHelper "Запись вне платформы" 228

13.1 R2.12 Фича "Запись экрана оператора вне платформы" 228

13.1.1 1. Описание фичи 228

13.1.2 2. Описание процесса/сервиса TO BE 228

13.1.3 3. Описание запросов 230

13.2 R2.12 Фича "Остановка записи экрана оператора вне платформы" 230

13.2.1 1. Описание фичи 231

13.2.2 2. Описание процесса/сервиса TO BE 231

13.2.3 3. Описание запросов 232

13.3 R2.12 Фича "Подключение агента записи экрана оператора для записи вне платформы" 233

13.3.1 ID фичи в JIRA 233

13.3.2 1. Описание фичи 233

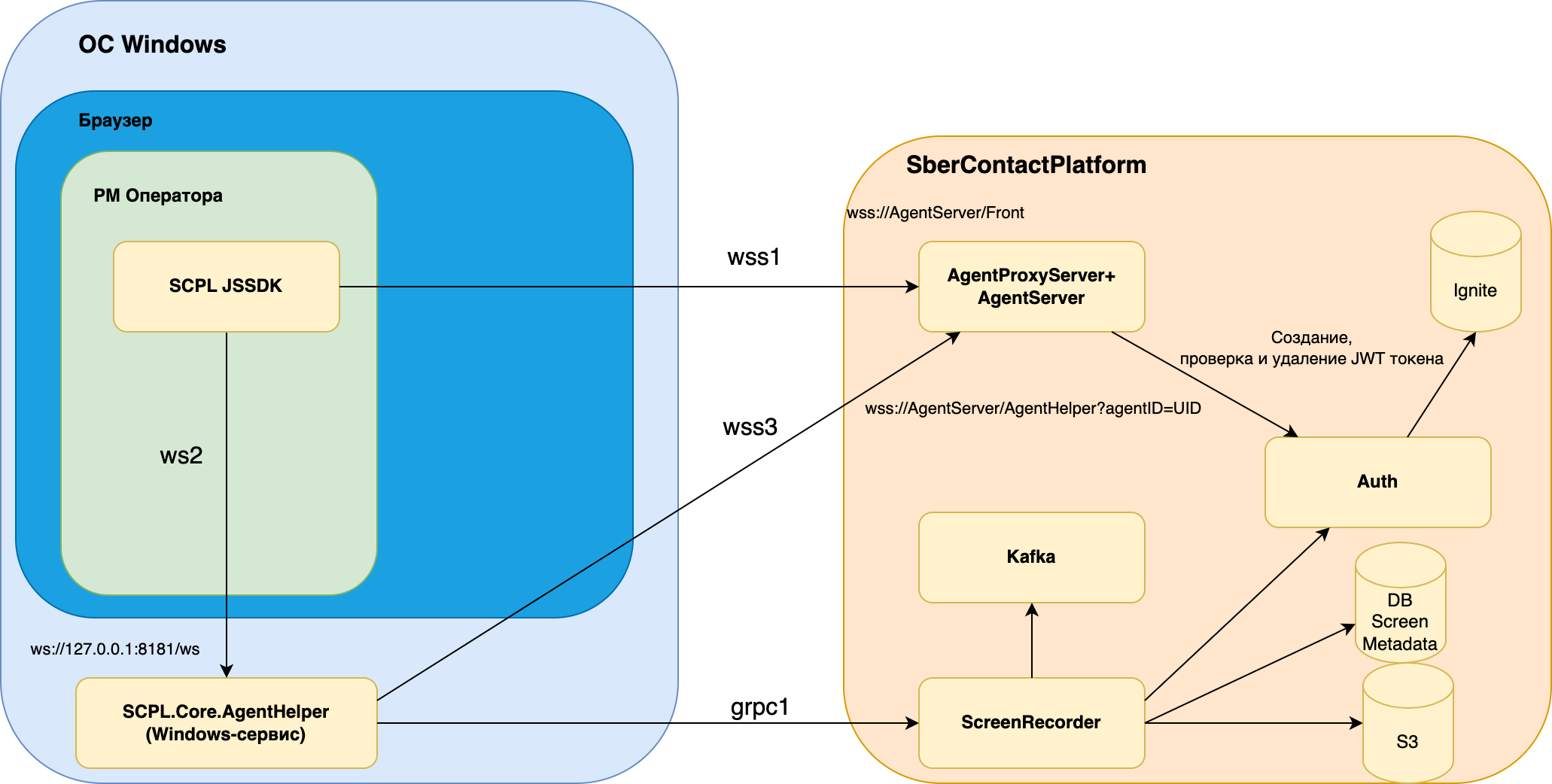
13.3.3 2. Описание процесса/сервиса TO BE 233

13.3.4 3. Примеры запросов 236

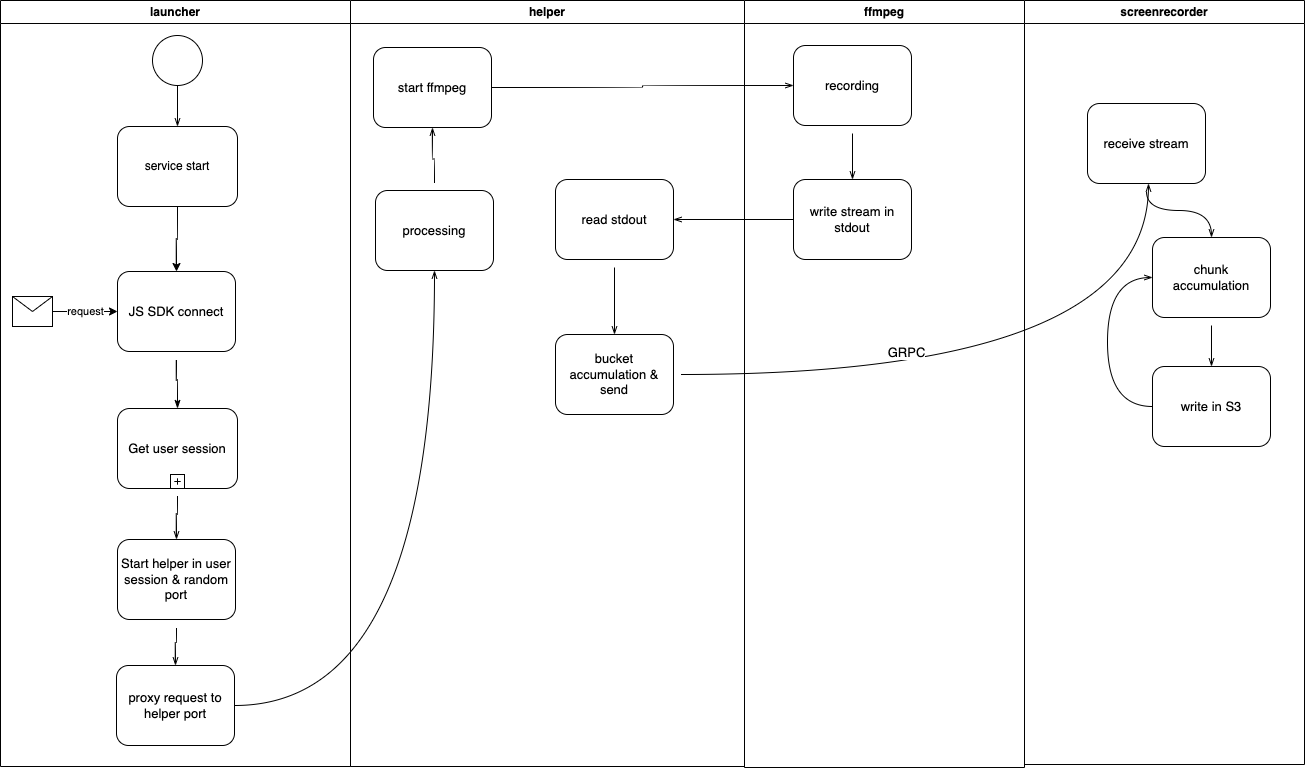
13.3.5 Бизнес-требования 236

* [Архитектура взаимодействия компонентов](#scroll-bookmark-2)
* [Требования к записи экранов](#scroll-bookmark-3)
* [Требования к реализации AgentHelper](#scroll-bookmark-4)
* [Основные характеристики](#scroll-bookmark-5)
* [Сообщения Web-Socket в сторону AgentServer](#scroll-bookmark-6)
* [Сообщения Web-Socket в сторону JsSDK](#scroll-bookmark-7)
* [Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-8)
* [Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-9)
* [Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-10)
* [Параметры записи экранов, передаваемые из AgentServer в AgentHelper для использования в ffmpeg](#scroll-bookmark-11)
* [Взаимодействие компонентов для записи экранов при back-to-back подключении](#scroll-bookmark-12)
* [Архив Core.AgentHelper](#scroll-bookmark-13)
* [Фича Core.AgentHelper "Запись вне платформы"](#scroll-bookmark-14)

# Архитектура взаимодействия компонентов



BPMN



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер | Инициатор | Получатель | Тип  взаим. | Протокол | Назначение |
| wss1 | SCPL JS SDK  Оператора | AgentProxyServer +AgentServer | Асинхронное | wss | Организация сессии пользователей для управления статусами и взаимодействиями, включая управление медиа |
| ws2 | SCPL JS SDK  Оператора | SCPL.Core.AgentHelper | Асинхронное | ws | Определение  запущенного SCPL Helper App на рабочем месте оператора.  Инициализация SCPL Helper App данными для подключения/переподключения к SCPL |
| wss3 | SCPL.Core.AgentHelper | AgentProxyServer +AgentServer | Асинхронное | wss | Подключение к тому-же AgentServer, на котором работает оператор  Выполнение команд по управлению записью экрана/экранов и трансляции экрана супервизору  Передача трафика экрана для супервизора |

|  |  |  |
| --- | --- | --- |
| Модуль | Функциональность |  |
| AgentHelper под Windows как сервис | При |  |
| AgentServer+JS SDK | Определение наличия AgentHelper и параметров  Установка регистрации из браузера на AgentHelper  Получение профиля записи экрана для оператора  Получение JWT-токена  Инициализация ScreenRecorder в части организации регистрации AgentHelper на AgentServer  Прием регистрации AgentHelper  Управление записью экрана оператора через AgentHelper  Формирование идентификаторов сегментов для записи экранов |  |
| ScreenRecorder | Получение трафика записи экрана и метаданных  Получение параметров записи экрана из AgentServer?  Передача данных о записи экрана в отчетность (как)? |  |
|  |  |  |
|  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| регистрация | АГС выписывает JVT токен в AUTh  при поднятии wss1  передается запрос на подключение helper + передача токена  ответ - агента записи нет/ есть - подключился/нет  АГС - SDK: запрос на подкл.xение хелпера (URI для подключения к хелперу, URI для подключения к АГС, JVT token  SDK, userID)+ уровень логирования logLevel (? из профиля записи)  SDK - helper: попытка открыть Ws2  helper-ASG:  попытка открыть WSS3, реги  ASG-Auth: проверка JVT токена на валидность - да/нет  helper-SDK: результат открытия wss3 - поднял/не поднял/не подключился - нет оператора, нет сессии |  |  |  |
|  | при закрытии wss3 не забыть отозвать токен |  |  |  |
|  | сдох агс, переподняли wss1 на другом агс  переподключение со стороны хелпера |  |  |  |
|  | уровень логирования - и в иницииализирующем событии, и в отдельном запросе  при попытке восстановления хелпера продолжается выполняться заданная логика в течение времени восстановления , если не подняли, то процессы останавливаем |  |  |  |
|  | рассмотреть кейсы обрыва связи:   * порвался ws2 * порвался wss3 - retry на стороне хелпера + если неуспешно, то сообщить агс и отправить запрос на подключение заново |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Требования к записи экранов

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| КФ. | 32 | Должна быть доступна функция поиска (по различным критериям) и воспроизведения записей (запись голоса, экрана оператора) голосовых взаимодействий |  |  |  |
| КФ. | 33 | Должна быть доступна функция поиска (по различным критериям) и воспроизведения записей экранов оператора для неголосовых взаимодействий |  |  |  |
| КФ. | 71 | Должна быть возможность записи экрана рабочего места оператора во время обработки голосового взаимодействия, включая постобработку. Запись должна осуществляться и сохраняться в привязке к обработке конкретного взаимодействия оператором |  |  |  |
| ОФ. | 45 | Должна быть возможность записи экрана рабочего места оператора во время обработки неголосового взаимодействия, включая постобработку. Запись должна осуществляться и сохраняться в привязке к обработке конкретного взаимодействия оператором |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Требования к реализации AgentHelper

|  |  |
| --- | --- |
| Требование | Этап реализации |
| Поддержка ОС Windows | 1 |
| Поддержка ОС Linux |  |
| Поддержка ОС MacOS |  |
| Захват экрана под Windows с использованием Desktop Duplication API (DXGI) |  |
| Работа AgentHelper как сервис Windows, демон Linux |  |
| Необходима поддержка записи нескольких мониторов на рабочем месте с централизованной настройкой. |  |
| AgentHelper должен реализовывать WebSocket-сервер для получения управляющих команд из Web-приложения  Сервер должен слушать только на адресе 127.0.0.1 (localhost) |  |
| По соединению WebSocket должна быть возможность   * Передать в AgentHelper   + URI для подключения к AgentProxyServer,   + AgentHelper URI для подключения к ScreenRecorder,   + токен для подключения,   + идентификатор оператора,   + ID-тенанта   + уровень логирования   + ID-сессии * Передавать из AgentHelper в JS SDK   + версию AgentHelper   + ошибки установления регистрации AgentHelper на AgentServer   + события о потере регистрации AgentHelper на AgentServer * Инициировать закрытие WebSocket при вылогинивании оператора * Корректно обрабатывать в AgentHelper разрыв WS-соединения в пределах timeout при при перезагрузке страницы в браузере (без перерегистрации на AgentServer и прерывания записи экранов) * Восстановить WebSocket соединение при обрыве |  |
| AgentHelper должен поддерживать подключение к нему нескольких web-приложений и работать с каждым подключением независимо от других |  |
| AgentHelper должен подключаться к AgentServer по grpc/wss.  Через это подключение должна поддерживаться следующая функциональность:   * Передача данных об AgentHelper:   + Токен для подключения   + Идентификатор оператора   + Идентификатор тенанта   + Версия AgentHelper   + Версия ОС   + Кол-во и параметры дисплеев оператора   + IP-адрес рабочего места   + Возможности AgentHelper * Получение и исполнение команд на запись экрана (старт, останов, пауза, снятие с паузы) * Получение от AgentServer параметры записи экрана (вплоть того, что на каждый вызов): частота кадров, параметры профиля сжатия и др * Получение и исполнение команд на передачу экрана супервизору * Передача трафика оператора для супервизора * Поддержка WebSocket со стороны AgentHelper * Завершение регистрации со стороны AgentServer * Завершение регистрации со стороны AgentHelper |  |
| При взаимодействии с ScreenRecorder по grpc AgentHelper должен:   * передавать метаданные для записи экрана на ScreenRecorder (идентификатор оператора, идентификатор сегмента, timestamp) * передавать стрим для записи экрана на ScreenRecorder. * отдельно передавать данные экранов для разных мониторов (в разных grpc стримах) * передавать отдельные стримы под отдельные сегменты записи экранов * переживать перезагрузку pod * переживать разрыв grpc |  |
| Возможность получения и использования уровня логирования от AgentServer (через JSSDK или через прямую регистрацию) |  |
| AgentHelper должен реализовывать логирование с настраиваемым уровнем и поддержкой ротации |  |
| Конфиг AgentServer должен содержать, как минимум:   * порт для ws сервера * уровень логирования * объем хранимых логов (в днях или чем-то еще) |  |
|  |  |

# Основные характеристики

|  |  |
| --- | --- |
|  | SCPL |
| Поддержка ОС на рабочих местах | Windows  Linux  MacOS |
| Способы захвата экрана под Windows | Desktop Duplication API - DXGI  (GDI как fallback) |
| Частота захвата экрана (FrameRate) | 5 FPS (можно регулировать)  1-30 |
| Кодек захвата записи экрана | H.264 |
| Возможность аппаратного ускорения encoder |  |
| Возможность настройки баланса  качества/компрессии сжатия | Да, отдельно для каждого монитора |
| Инструмент захвата записи экрана |  |
| Способ сжатия | ? |
| Протокол передачи записи экрана на сервер | grpc |
| Возможность шифрования записей экрана | ? FS GW? |
| Просмотр экрана оператора  супервизором в реальном времени | Да  Форк на стороне SCPL Helper и передаем на AG по grpc |
| Контейнер для хранения записей экранов на сервере | не используем |
| Как реализован сервер записи экранов | Windows-сервис  Linux-демон |
| Куда записываются файлы | Стриминг в S3 |
| Где долговременное хранение | S3 |
| Пережимается ли на бэке запись экрана для хранения | Нет |
| Пережимается ли запись экрана для  воспроизведения в рабочем месте супервизора (из записи) | Нет  Добавляем голос |
| Способ воспроизведения записи экрана | HTTP Streaming |

# Сообщения Web-Socket в сторону AgentServer

* [agentHelperInfo - редактирование](#scroll-bookmark-15)
* [newRecBatch](#scroll-bookmark-16)
* [resultRestartScreenRecord](#scroll-bookmark-17)
* [resultStartScreenRecord](#scroll-bookmark-18)
* [resultStartStream](#scroll-bookmark-19)
* [resultStopScreenRecord](#scroll-bookmark-20)
* [resultStopStream](#scroll-bookmark-21)
* [streamBlob](#scroll-bookmark-22)

## agentHelperInfo - редактирование

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *agentHelperInfo*- передача данных о рабочему месте оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | computerName | string | Имя компьютера, в котором запущена платформа, если используется ВАРМ. При подключении с АРМ значение совпадает с realComputerName | [0...1] |
|  | realComputerName | string | Реальное имя устройства, с которого подключен оператор | [0...1] |
|  | realComputerIP | string | Реальный  IP-адрес устройства, с которого подключен оператор | [0...1] |
|  | osLogin | string | Операционная система устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | osDomain | string | Домен устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | versionAgentHelper | string | Версия AgentHelper, если он был обнаружен на рабочем месте. | [0...1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperInfo",  "description": "передача данных о рабочему месте оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "computerName": {  "type": "string"  },  "realComputerName": {  "type": "string"  },  "realComputerIP": {  "type": "string"  },  "osLogin": {  "type": "string"  },  "osDomain": {  "type": "string"  },  "versionAgentHelper": {  "type": "string"  }  }  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "agentHelperInfo",  "version": "v1.0",  "data": {  "computerName": "ccc-fv-452",  "realComputerName": "cac-jo-257",  "realComputerIP": "1.2.3.4",  "osLogin": "0123456789",  "osDomain": "GAMMA.SBER.RU",  "versionAgentHelper": "AgentHelper 1.9.1.14"  }  } |

## newRecBatch

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  newRecBatch  - сообщение содержащее имя файла записи | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат создания файла записи. Возможные значения:   * succes * error | [1] |
|  | fileName | string | Имя файла | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "newRecBatch",  "description": "сообщение содержащее имя файла записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string",  "enum": [  "success",  "error"  ]  },  "fileName": {  "type": "string"  }  },  "required": [  "result",  "fileName"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"newRecBatch",  "version": "1.0",  "data": {  "result": "success",  "fileName": "ABCDEFG"  }  } |

## resultRestartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultRestartScreenRecord* - сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *restartScreenRecord* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultRestartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultRestartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## resultStartStream

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  *resultStartStream* - сообщение о результате исполнения запроса просмотра экрана оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *startStream* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartStream",  "description": "сообщение о результате исполнения запроса просмотра экрана оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartStream",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## resultStopScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## resultStopStream

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  *resultStopStream* - сообщение о результате исполнения запроса остановки просмотра экрана оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *stopStream* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStopStream",  "description": "сообщение о результате исполнения запроса остановки просмотра экрана оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopStream",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## streamBlob

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  *streamBlob* - сообщение содержащее массив байт | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | payload | string | Массив байт | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "streamBlob",  "description": "сообщение содержащее поток информации",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "payload": {  "type": "string"  }  },  "required": [  "payload"  ]  }  },  "required": [  "action",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "streamBlob",  "version": "1.0",  "data": {  "payload": "массив байт"  }  } |

# Сообщения Web-Socket в сторону JsSDK

* [resultAgentOpenConnect](#scroll-bookmark-23)

## resultAgentOpenConnect

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

# Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [v1 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-24)
* [v2 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-25)
* [v3 R2.1.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-26)
* [v4 R2.8.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-27)
* [v5 R2.11 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-28)

## v1 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [1. Описание фичи](#scroll-bookmark-29)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-30)
* [3. Описание запросов](#scroll-bookmark-31)
  + [web-socket сообщения](#scroll-bookmark-32)
    - [getAgentHelperData (JsSDK->AgS)](#scroll-bookmark-33)
    - [resultGetAgentHelperData (AgS→JsSDK)](#scroll-bookmark-34)
    - [agentOpenConnect (JsSDK ->AH)](#scroll-bookmark-35)
    - [resultAgentOpetConnect (AH → JsSDK)](#scroll-bookmark-36)
    - [agentHelperConnect (JsSDK->AgS)](#scroll-bookmark-37)

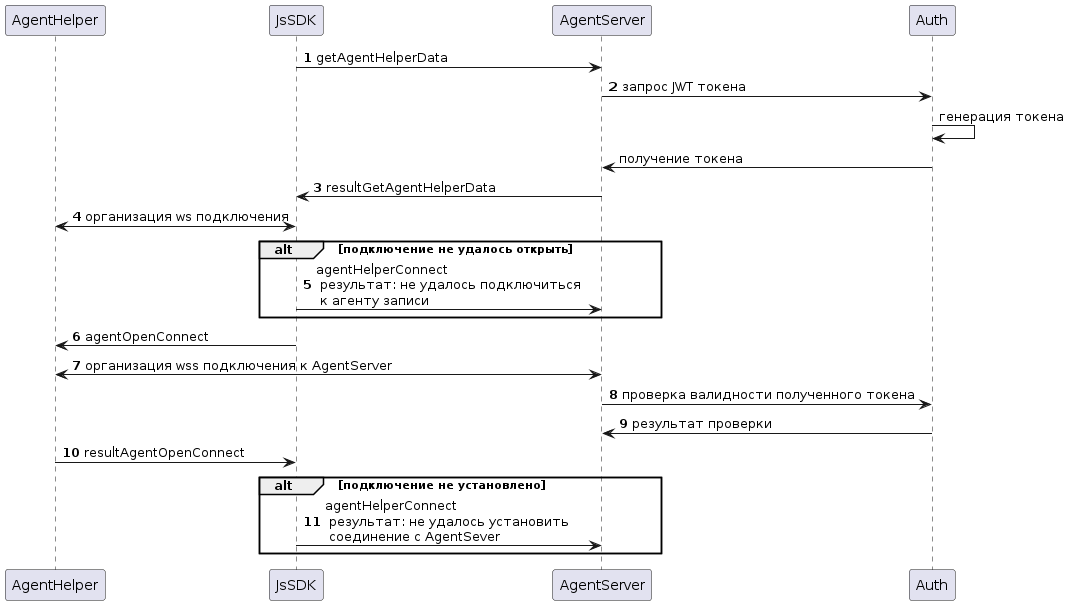
**ID фичи в JIRA**

[Epic SCPL-3494](https://jira.sberbank.ru/browse/SCPL-3494?src=confmacro) - Запись экранов **done**

### 1. Описание фичи

При старте операторской сессии необходимо инициировать процесс подключения агента записи экрана оператора для получения команд управления логикой записи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | JsSDK инициирует обращение к AgentHelper за информацией по подключению к агенту записи экранов | JsSDK направляет запрос [*getAgentHelperData*](#scroll-bookmark-38) |
| 2 | AgentServer обращается в Auth сервис за генерацией JWT-токена для подключения AgentHelper | Auth сервис генерит JWT-токен и передает его AgentServer'у |
| 3 | AgentServer отправляет в сторону JsSDK запрос на подключение агента записи | AgentServer передает [*resultGetAgentHelperData*](#scroll-bookmark-39) с параметрами подключения к AgentHelper, в котором передается   * URI AgentHelper, по которому должен JsSDK должен запрашивать подключение * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен |
| 4 | JsSDK инициирует обращение к AgentHelper по полученному в сообщении URI | Осуществляется попытка открыть незащищенное web-socket подключение   * если подключение неуспешное, то осуществляется переход на шаг 5 * если подключение успешное, то осуществляется переход на шаг 6 |
| 5 | При неуспешной попытке подключения к AgentHelper JsSDK передает результат исполнения запроса в AgentServer | AgentServer получает ответ с неуспешным результатом обращения в AgentHelper [*agentHelperConnect*](#scroll-bookmark-40) |
| 6 | После установления ws соединения JsSDK направляет сообщение с информацией о подключении к AgentServer | JsSDK передает ws сообщение [*agentOpenConnect*](#scroll-bookmark-41) , в котором передает   * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен   из полученного от AgentServer сообщения (см шаг 3) |
| 7 | AgentHelper пытается установить защищенное web-socket  подключение  к AgentServer  по полученному URI | AgentHelper инициирует подключение к AgentServer по указанному URI и передает полученный JWT токен |
| 8 | AgentServer проверяет валидность полученного от AgentHelper токена |  |
| 9 | Если токен валидный, то между AgentHelper и AgentServer устанавливается защищенное web-socket подключение. AgentServer сохраняет параметры AgentHelper, который к нему подключился в связке с текущей операторской сессией  Если токен невалидный, то подключение к AgentServer не установлено |  |
| 10 | JsSDK передает AgentServer результат подключения AgentHelper к AgentServer | AgentHelper передает в JsSDK сообщение  [*resultAgentOpenConnect*](#scroll-bookmark-42)  result = success в случае успешного подключения  result = error в случае неуспешного подключения |
| 11 | JsSDK передает в AgentServer сообщение [*agentHelperConnect*](#scroll-bookmark-40) о неуспешной попытке подключения AgentHelper к AgentServer | AgentServer   * Проверяет заданные правила по повторным попыткам подключения |

### 3. Описание запросов

#### web-socket сообщения

##### getAgentHelperData (JsSDK->AgS)

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *getAgentHelperData*- Запрос на передачу данных для подключения к агенту записи экранов пользователя | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getAgentHelperData",  "description": "Запрос на передачу данных для подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getAgentHelperData",  "version": "1.0",  "transaction": "string"  } |

##### resultGetAgentHelperData (AgS→JsSDK)

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultGetAgenHelperData*- Передача данных для инициализации подключения к агенту записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction на запрос *getAgentHelperData* | [0...1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса. Заполняется только для result = error | [0...1] |
|  | agentHelperURI | string | Адрес для подключения к AgentHelper  Обязательно для result = success | [0...1] |
|  | agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов  Обязательно для result = success | [0...1] |
|  | screenServer | string | Адрес screenServer для отправки потока  Обязательно для result = success | [0...1] |
|  | JWTToken | string | Аутентификационный токен  Обязательно для result = success | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetAgenHelperData",  "description": "Передача данных для инициализации подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  },  "agentHelperURI": {  "type": "string"  },  "agentServerURI": {  "type": "string"  },  "screenServer": {  "type": "string"  },  "JWTToken": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultGetAgenHelperData",  "version": "1.0",  "transaction": "string",  "data":{  "result":"success",  "agentHelperURI":"XXX",  "agentServerURI":"XXX",  "screenServer":"screenServer",  "JWTToken":"XXX"  }  } |

##### agentOpenConnect (JsSDK ->AH)

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *agentOpenConnect*- сообщение для подключения агента записи экранов к AgentServer | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0...1] |
| data | object | Тело сообщения | [1] |
| agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов | [1] |
| JWTToken | string | Аутентификационный токен | [1] |
| userID | UUID | Идентификатор пользователя | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentOpenConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "agentServerURI": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "userID": {  "type": "string"  }  },  "required": [  "agentServerURI",  "JWTToken",  "userID"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "agentServerURI":"XXX",  "JWTToken":"WWW",  "userID": "UUID"  }  } |

##### resultAgentOpetConnect (AH → JsSDK)

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### agentHelperConnect (JsSDK->AgS)

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *agentHelperConnect*- результат подключения агента записи | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |
| result | string | Результат исполнения запроса   * success * error | [1] |
| resultDesc | string | Причина неуспешного подключения. Заполняется только для result = error  Возможные значения:  "AgentHelper not found" - не удалось подключиться к агенту записи  "connection failedf" - агент записи не смог установить подключение к AgentServer  "invalid token" - аутентификационный токен невалиден | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "result":"error",  "resultDesc":"invalid token"  }  } |

## v2 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [1. Описание фичи](#scroll-bookmark-43)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-44)
* [3. Описание запросов](#scroll-bookmark-45)
  + [web-socket сообщения](#scroll-bookmark-46)
    - [getAgentHelperData (JsSDK->AgS)](#scroll-bookmark-47)
    - [resultGetAgentHelperData (AgS→JsSDK)](#scroll-bookmark-48)
    - [agentOpenConnect (JsSDK ->AH)](#scroll-bookmark-49)
    - [resultAgentOpenConnect (AH → JsSDK)](#scroll-bookmark-50)
    - [agentHelperConnect (JsSDK->AgS)](#scroll-bookmark-51)
    - [agentHelperConnectionState (AgS → JsSDK)](#scroll-bookmark-52)

**ID фичи в JIRA**

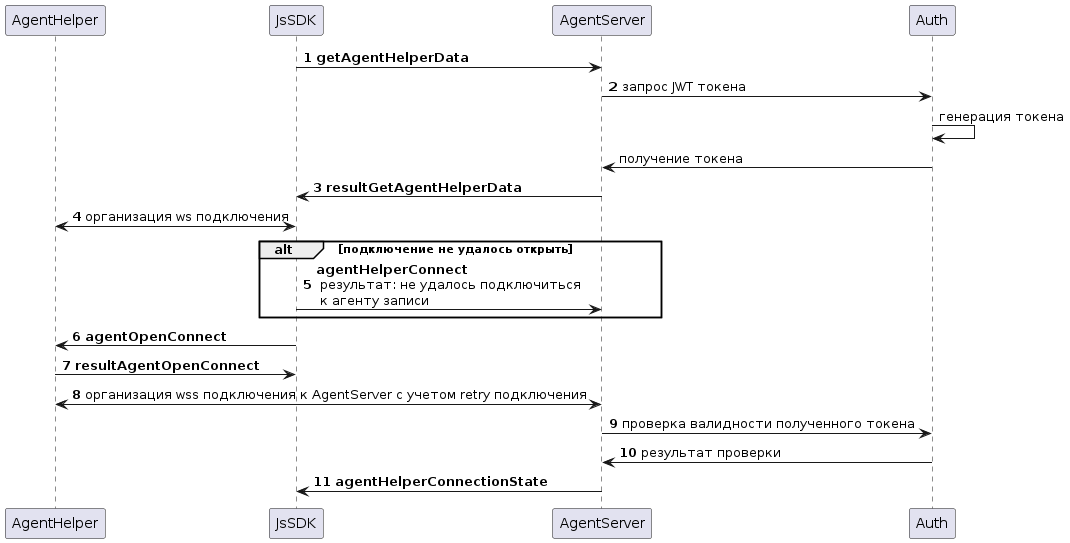
[Epic SCPL-3494](https://jira.sberbank.ru/browse/SCPL-3494?src=confmacro) - Запись экранов **done**

[Epic SCPL-6585](https://jira.sberbank.ru/browse/SCPL-6585?src=confmacro) - Q3 Развитие агента записи экранов **done**

### 1. Описание фичи

При старте операторской сессии необходимо инициировать процесс подключения агента записи экрана оператора для получения команд управления логикой записи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | JsSDK инициирует обращение к AgentHelper за информацией по подключению к агенту записи экранов | JsSDK направляет запрос [*getAgentHelperData*](#scroll-bookmark-53) |
| 2 | AgentServer обращается в Auth сервис за генерацией JWT-токена для подключения AgentHelper | Auth сервис генерит JWT-токен и передает его AgentServer'у |
| 3 | AgentServer отправляет в сторону JsSDK запрос на подключение агента записи | AgentServer передает [*resultGetAgentHelperData*](#scroll-bookmark-54) с параметрами подключения к AgentHelper, в котором передается   * URI AgentHelper, по которому должен JsSDK должен запрашивать подключение * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен |
| 4 | JsSDK инициирует обращение к AgentHelper по полученному в сообщении URI | Осуществляется попытка открыть незащищенное web-socket подключение   * если подключение неуспешное, то осуществляется переход на шаг 5 * если подключение успешное, то осуществляется переход на шаг 6   В сторону рабочего места отдается уведомление о статусе открытия ws подключения к AgentHelper |
| 5 | При неуспешной попытке подключения к AgentHelper JsSDK передает результат исполнения запроса в AgentServer | AgentServer получает ответ с неуспешным результатом обращения в AgentHelper [*agentHelperConnect*](#scroll-bookmark-55) |
| 6 | После установления ws соединения JsSDK направляет сообщение с информацией о подключении к AgentServer | JsSDK передает ws сообщение [*agentOpenConnect*](#scroll-bookmark-56) , в котором передает   * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен   из полученного от AgentServer сообщения (см шаг 3) |
| 7 | JsSDK передает AgentServer результат обработки запроса на подключение AgentHelper к AgentServer | AgentHelper передает в JsSDK сообщение  [*resultAgentOpenConnect*](#scroll-bookmark-57)   * result = success - AgentHelper принял запрос на организацию подключения к AgentServer. Осуществляется переход на шаг 8 * result = error - AgentHelper не смог принять в работу запрос на организацию подключения к AgentServer. Процесс завершен |
| 8 | AgentHelper пытается установить защищенное web-socket  подключение  к AgentServer  по полученному URI | AgentHelper инициирует подключение к AgentServer по указанному URI и передает полученный JWT токен |
| 9 | AgentServer проверяет валидность полученного от AgentHelper токена | * Если токен валидный, то осуществляется переход на шаг 10 * Если токен невалидный , то AgentHelper приостанавливает попытки реконнекта и осуществляется переход на шаг 11 |
| 10 | AgentHelper пытается установить устанавливается защищенное web-socket подключение с AgentServer | * Если соединение установлено, то AgentServer сохраняет параметры AgentHelper, который к нему подключился в связке с текущей операторской сессией. Осуществляется переход на шаг 11 * Если соединение не установлено, то осуществляется повторная попытка через заданное время. Период повторных попыток подключения получен AgentHelper'ом в сообщении [*getAgentHelperData*](#scroll-bookmark-53)на шаге 1. |
| 11 | AgentServer передает в JsSDK информацию об актуальном состоянии подключение агента записи экраном | AgentServer передает в JsSDKсообщение [***agentHelperConnectionState***](#scroll-bookmark-58)   * connectionState = active - AgentHelper подключен * connectionState = inactive - AgentHelper не подключен   + при деактивации подключения AgentServer в errorDesc передает ошибку  (по возможности реализовать передачу ошибки из-за разрыва wed-socket подключения (**connectionLost**) и из-за невалидного токена (**tokenInvalid**)   (предупреждение) AgentServer уведомляет JsSDK каждый раз при изменении состояния подключения |

### 3. Описание запросов

#### web-socket сообщения

##### getAgentHelperData (JsSDK->AgS)

getAgentHelperData

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *getAgentHelperData*- Запрос на передачу данных для подключения к агенту записи экранов пользователя | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getAgentHelperData",  "description": "Запрос на передачу данных для подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getAgentHelperData",  "version": "1.0",  "transaction": "string"  } |

##### resultGetAgentHelperData (AgS→JsSDK)

resultGetAgentHelperData

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultGetAgentHelperData*- Передача данных для инициализации подключения к агенту записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction на запрос *getAgentHelperData* | [0...1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса. Заполняется только для result = error | [0...1] |
|  | agentHelperURI | string | Адрес для подключения к AgentHelper  Обязательно для result = success | [0...1] |
|  | agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов  Обязательно для result = success | [0...1] |
|  | screenServer | string | Адрес screenServer для отправки потока  Обязательно для result = success | [0...1] |
|  | JWTToken | string | Аутентификационный токен  Обязательно для result = success | [0...1] |
|  | retryDelay | string | Период повторения попытки установления подключения к AgentServer  Обязательно для result = success | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetAgentHelperData",  "description": "Передача данных для инициализации подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  },  "agentHelperURI": {  "type": "string"  },  "agentServerURI": {  "type": "string"  },  "screenServer": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "retryDelay":{  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultGetAgentHelperData",  "version": "1.0",  "transaction": "string",  "data":{  "result":"success",  "agentHelperURI":"XXX",  "agentServerURI":"XXX",  "screenServer":"screenServer",  "JWTToken":"XXX",  "retryDelay": "1"  }  } |

##### agentOpenConnect (JsSDK ->AH)

agentOpenConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *agentOpenConnect*- сообщение для подключения агента записи экранов к AgentServer | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0...1] |
| data | object | Тело сообщения | [1] |
| agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов | [1] |
| JWTToken | string | Аутентификационный токен | [1] |
| userID | UUID | Идентификатор пользователя | [1] |
| retryDelay | string | Период повторения попытки установления подключения к AgentServer | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentOpenConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "agentServerURI": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "userID": {  "type": "string"  },  "retryDelay": {  "type": "string"  }  },  "required": [  "agentServerURI",  "JWTToken",  "userID",  "retryDelay"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "agentServerURI":"XXX",  "JWTToken":"WWW",  "userID": "UUID",  "retryDelay":"1"  }  } |

##### resultAgentOpenConnect (AH → JsSDK)

resultAgentOpenConnect

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### agentHelperConnect (JsSDK->AgS)

agentHelperConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *agentHelperConnect*- результат подключения агента записи | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |
| result | string | Результат исполнения запроса   * success * error | [1] |
| resultDesc | string | Причина неуспешного подключения. Заполняется только для result = error  Возможные значения:  "AgentHelper not found" - не удалось подключиться к агенту записи  "connection failedf" - агент записи не смог установить подключение к AgentServer  "invalid token" - аутентификационный токен невалиден | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "result":"error",  "resultDesc":"invalid token"  }  } |

##### agentHelperConnectionState (AgS → JsSDK)

agentHelperConnectionState

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  agentHelperConnectionState - состояние подключения агента записи экрана в рамках сессии текущего оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string |  | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | userID | string | Идентификатор пользователя | [1] |
|  | connectionState | string | Текущее состояние подключения к агенту записи экранов  Возможные значения:   * active * inactive | [1] |
|  | resultDesc | string | описание ошибки  только при connectionState=inactive | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnectionState",  "description": "состояние подключения агента записи экрана в рамках сессии текущего оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "userID": {  "type": "string"  },  "connectionState": {  "type": "string",  "enum":[  "active",  "inactive"  ]  },  "resultDesc": {  "type": "string"  }  },  "required": [  "userID",  "connectionState"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnectionState",  "version": "1.0",  "transaction": "string",  "data": {  "userID":"UUID",  "connectionState":"active"  }  } |

## v3 R2.1.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [1. Описание фичи](#scroll-bookmark-60)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-61)
* [3. Описание запросов](#scroll-bookmark-62)
  + [web-socket сообщения](#scroll-bookmark-63)
    - [getAgentHelperData (JsSDK->AgS)](#scroll-bookmark-64)
    - [resultGetAgentHelperData (AgS→JsSDK)](#scroll-bookmark-65)
    - [agentOpenConnect (JsSDK ->AH)](#scroll-bookmark-66)
    - [resultAgentOpenConnect (AH → JsSDK)](#scroll-bookmark-67)
    - [agentHelperConnect (JsSDK->AgS)](#scroll-bookmark-68)
    - [agentHelperConnectionState (AgS → JsSDK)](#scroll-bookmark-69)
    - [agentHelperInfo (AH → AgS)](#scroll-bookmark-70)
  + [REST-запросы](#scroll-bookmark-71)
    - [/private/v1.0/users/{id}](#scroll-bookmark-72)

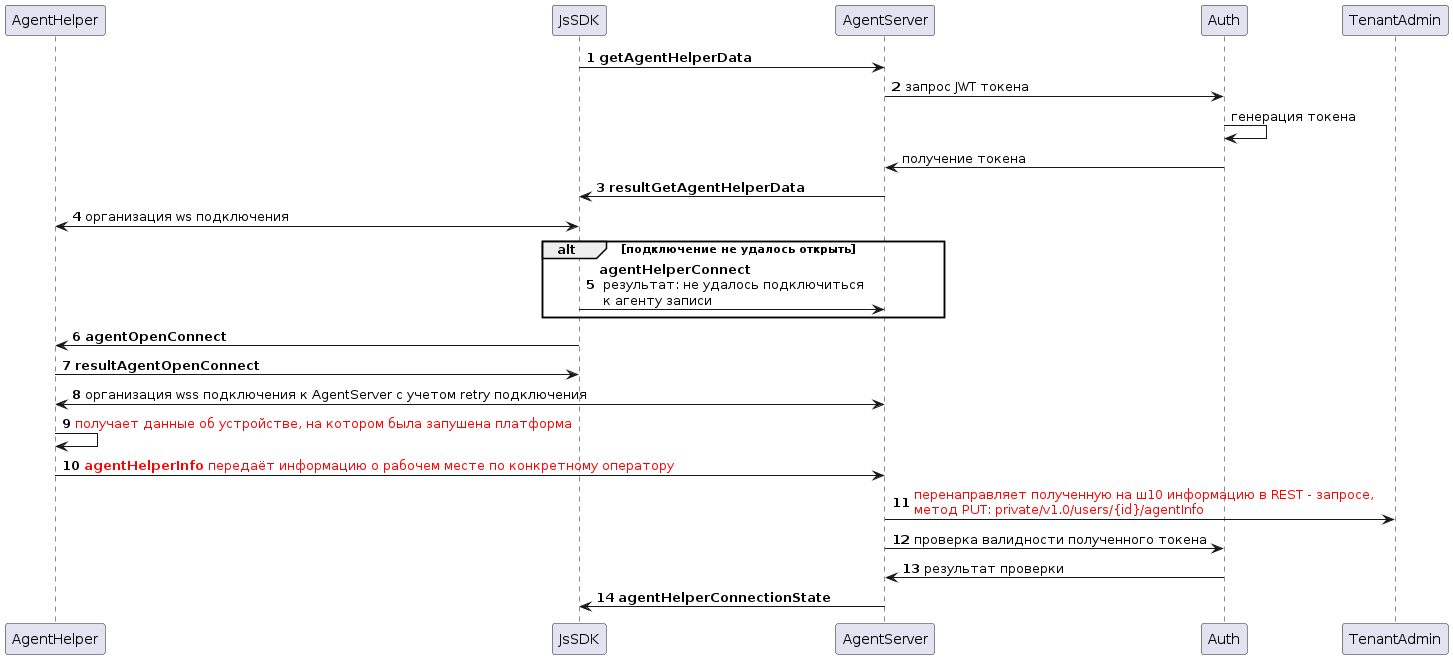
**ID фичи в JIRA**

[Epic SCPL-10511](https://jira.sberbank.ru/browse/SCPL-10511?src=confmacro) - Передача параметров сессии оператора в админку/в отчетность/в логи **done**

### 1. Описание фичи

При старте операторской сессии необходимо инициировать процесс подключения агента записи экрана оператора для получения команд управления логикой записи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | JsSDK инициирует обращение к AgentHelper за информацией по подключению к агенту записи экранов | JsSDK направляет запрос [*getAgentHelperData*](#scroll-bookmark-73) |
| 2 | AgentServer обращается в Auth сервис за генерацией JWT-токена для подключения AgentHelper | Auth сервис генерирует JWT-токен и передает его AgentServer'у |
| 3 | AgentServer отправляет в сторону JsSDK запрос на подключение агента записи | AgentServer передает [*resultGetAgentHelperData*](#scroll-bookmark-74) с параметрами подключения к AgentHelper, в котором передается   * URI AgentHelper, по которому должен JsSDK должен запрашивать подключение * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен |
| 4 | JsSDK инициирует обращение к AgentHelper по полученному в сообщении URI | Осуществляется попытка открыть незащищенное web-socket подключение   * если подключение неуспешное, то осуществляется переход на шаг 5 * если подключение успешное, то осуществляется переход на шаг 6   В сторону рабочего места отдается уведомление о статусе открытия ws подключения к AgentHelper |
| 5 | При неуспешной попытке подключения к AgentHelper JsSDK передает результат исполнения запроса в AgentServer | AgentServer получает ответ с неуспешным результатом обращения в AgentHelper [*agentHelperConnect*](#scroll-bookmark-75) |
| 6 | После установления ws соединения JsSDK направляет сообщение с информацией о подключении к AgentServer | JsSDK передает ws сообщение [*agentOpenConnect*](#scroll-bookmark-76) , в котором передает   * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен   из полученного от AgentServer сообщения (см шаг 3) |
| 7 | JsSDK передает AgentServer результат обработки запроса на подключение AgentHelper к AgentServer | AgentHelper передает в JsSDK сообщение  [*resultAgentOpenConnect*](#scroll-bookmark-77)   * result = success - AgentHelper принял запрос на организацию подключения к AgentServer. Осуществляется переход на шаг 8 * result = error - AgentHelper не смог принять в работу запрос на организацию подключения к AgentServer. Процесс завершен |
| 8 | AgentHelper пытается установить защищенное web-socket  подключение  к AgentServer  по полученному URI | AgentHelper инициирует подключение к AgentServer по указанному URI и передает полученный JWT токен |
| 9 | После установленного на шаге 8 защищенного web-socket подключение к AgentServer, AgentHelper получает данные о компьютере, на котором была запушена платформа | Список получаемых атрибутов:   * *realComputerName* - реальное имя устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $CLIENTNAME имя ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientName * *realComputerIP* - реальный  IP-адрес устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $SESSIONNAME IP-адрес ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientAddress * *computerName* - имя устройства, на котором запущена платформа SCPL (если используется ВАРМ, то имя компьютера ВАРМ). При подключении с АРМ значение совпадает с realComputerName. Полученное по средствам func Hostname * *osLogin* - логин в операционную система устройства, с которого подключен оператор. Значение полученное в переменной окружения $USERNAME или по средствам winAPI * *osDomain* - домен устройства, с которого подключен оператор (по средствам winAPI). * *versionAgentHelper* - версия AgentHelper, если он был обнаружен на рабочем месте. |
| 10 | AgentHelper передаёт в сообщении [***agentHelperInfo***](#scroll-bookmark-78)информацию о рабочем месте оператора | AgentServer получает данные и сохраняет из у себя локально |
| 11 | AgentServer перенаправляет полученную на ш10 информацию в REST - запросе, метод PUT: [/private/v1.0/users/{id}/agentInfo](#scroll-bookmark-79). В случае если ответ !=200 AgentServer делает ещё одну попытку запроса. |  |
| 12 | AgentServer проверяет валидность полученного от AgentHelper токена | * Если токен валидный, то осуществляется переход на шаг 10 * Если токен невалидный , то AgentHelper приостанавливает попытки реконнекта и осуществляется переход на шаг 11 |
| 13 | AgentHelper пытается установить устанавливается защищенное web-socket подключение с AgentServer | * Если соединение установлено, то AgentServer сохраняет параметры AgentHelper, который к нему подключился в связке с текущей операторской сессией. Осуществляется переход на шаг 11 * Если соединение не установлено, то осуществляется повторная попытка через заданное время. Период повторных попыток подключения получен AgentHelper'ом в сообщении [*getAgentHelperData*](#scroll-bookmark-73)на шаге 1. |
| 14 | AgentServer передает в JsSDK информацию об актуальном состоянии подключение агента записи экраном | AgentServer передает в JsSDKсообщение [***agentHelperConnectionState***](#scroll-bookmark-80)   * connectionState = active - AgentHelper подключен * connectionState = inactive - AgentHelper не подключен   + при деактивации подключения AgentServer в errorDesc передает ошибку  (по возможности реализовать передачу ошибки из-за разрыва wed-socket подключения (**connectionLost**) и из-за невалидного токена (**tokenInvalid**)   (предупреждение) AgentServer уведомляет JsSDK каждый раз при изменении состояния подключения |

### 3. Описание запросов

#### web-socket сообщения

##### getAgentHelperData (JsSDK->AgS)

getAgentHelperData

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *getAgentHelperData*- Запрос на передачу данных для подключения к агенту записи экранов пользователя | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getAgentHelperData",  "description": "Запрос на передачу данных для подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getAgentHelperData",  "version": "1.0",  "transaction": "string"  } |

##### resultGetAgentHelperData (AgS→JsSDK)

resultGetAgentHelperData

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultGetAgentHelperData*- Передача данных для инициализации подключения к агенту записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction на запрос *getAgentHelperData* | [0...1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса. Заполняется только для result = error | [0...1] |
|  | agentHelperURI | string | Адрес для подключения к AgentHelper  Обязательно для result = success | [0...1] |
|  | agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов  Обязательно для result = success | [0...1] |
|  | screenServer | string | Адрес screenServer для отправки потока  Обязательно для result = success | [0...1] |
|  | JWTToken | string | Аутентификационный токен  Обязательно для result = success | [0...1] |
|  | retryDelay | string | Период повторения попытки установления подключения к AgentServer  Обязательно для result = success | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetAgentHelperData",  "description": "Передача данных для инициализации подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  },  "agentHelperURI": {  "type": "string"  },  "agentServerURI": {  "type": "string"  },  "screenServer": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "retryDelay":{  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultGetAgentHelperData",  "version": "1.0",  "transaction": "string",  "data":{  "result":"success",  "agentHelperURI":"XXX",  "agentServerURI":"XXX",  "screenServer":"screenServer",  "JWTToken":"XXX",  "retryDelay": "1"  }  } |

##### agentOpenConnect (JsSDK ->AH)

agentOpenConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *agentOpenConnect*- сообщение для подключения агента записи экранов к AgentServer | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0...1] |
| data | object | Тело сообщения | [1] |
| agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов | [1] |
| JWTToken | string | Аутентификационный токен | [1] |
| userID | UUID | Идентификатор пользователя | [1] |
| retryDelay | string | Период повторения попытки установления подключения к AgentServer | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentOpenConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "agentServerURI": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "userID": {  "type": "string"  },  "retryDelay": {  "type": "string"  }  },  "required": [  "agentServerURI",  "JWTToken",  "userID",  "retryDelay"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "agentServerURI":"XXX",  "JWTToken":"WWW",  "userID": "UUID",  "retryDelay":"1"  }  } |

##### resultAgentOpenConnect (AH → JsSDK)

resultAgentOpenConnect

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### agentHelperConnect (JsSDK->AgS)

agentHelperConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *agentHelperConnect*- результат подключения агента записи | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |
| result | string | Результат исполнения запроса   * success * error | [1] |
| resultDesc | string | Причина неуспешного подключения. Заполняется только для result = error  Возможные значения:  "AgentHelper not found" - не удалось подключиться к агенту записи  "connection failedf" - агент записи не смог установить подключение к AgentServer  "invalid token" - аутентификационный токен невалиден | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "result":"error",  "resultDesc":"invalid token"  }  } |

##### agentHelperConnectionState (AgS → JsSDK)

agentHelperConnectionState

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  agentHelperConnectionState - состояние подключения агента записи экрана в рамках сессии текущего оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string |  | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | userID | string | Идентификатор пользователя | [1] |
|  | connectionState | string | Текущее состояние подключения к агенту записи экранов  Возможные значения:   * active * inactive | [1] |
|  | resultDesc | string | описание ошибки  только при connectionState=inactive | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnectionState",  "description": "состояние подключения агента записи экрана в рамках сессии текущего оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "userID": {  "type": "string"  },  "connectionState": {  "type": "string",  "enum":[  "active",  "inactive"  ]  },  "resultDesc": {  "type": "string"  }  },  "required": [  "userID",  "connectionState"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnectionState",  "version": "1.0",  "transaction": "string",  "data": {  "userID":"UUID",  "connectionState":"active"  }  } |

##### agentHelperInfo (AH → AgS)

agentHelperInfo

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *agentHelperInfo*- передача данных о рабочему месте оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | computerName | string | Имя компьютера, в котором запущена платформа, если используется ВАРМ. При подключении с АРМ значение совпадает с realComputerName | [0...1] |
|  | realComputerName | string | Реальное имя устройства, с которого подключен оператор | [0...1] |
|  | realComputerIP | string | Реальный  IP-адрес устройства, с которого подключен оператор | [0...1] |
|  | osLogin | string | Операционная система устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | osDomain | string | Домен устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | versionAgentHelper | string | Версия AgentHelper, если он был обнаружен на рабочем месте. | [0...1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperInfo",  "description": "передача данных о рабочему месте оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "computerName": {  "type": "string"  },  "realComputerName": {  "type": "string"  },  "realComputerIP": {  "type": "string"  },  "osLogin": {  "type": "string"  },  "osDomain": {  "type": "string"  },  "versionAgentHelper": {  "type": "string"  }  }  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "agentHelperInfo",  "version": "v1.0",  "data": {  "computerName": "ccc-fv-452",  "realComputerName": "cac-jo-257",  "realComputerIP": "1.2.3.4",  "osLogin": "0123456789",  "osDomain": "GAMMA.SBER.RU",  "versionAgentHelper": "AgentHelper 1.9.1.14"  }  } |

#### REST-запросы

##### /private/v1.0/users/{id}

/private/v1.0/users/{id}

|  |  |  |  |
| --- | --- | --- | --- |
| № версии | Задача в JIRA | Описание изменений | Реализовано |
| v1.0 | [Task SCPL-5090](https://jira.sberbank.ru/browse/SCPL-5090?src=confmacro) - [Space] SA: получение из отчётности/AgS информации для отображении карточке оператора **closed** | Эндпоинты для передачи информации по пользователю из AgS | **not ready** |
| v1.1 | [Story SCPL-13550](https://jira.sberbank.ru/browse/SCPL-13550?src=confmacro) - [Space] Конфигурирование IP телефонов в Universe **done** | Добавлен атрибут нового ТА | **not ready** |
| v1.2 | [Story SCPL-13373](https://jira.sberbank.ru/browse/SCPL-13373?src=confmacro) - [Space] 2.11. Скорректировать логику отображения времени логаута пользователя **done** | Скорректировать логику отображения времени логаута пользователя | **not ready** |

Описание API "Запрос на обновление информации о пользователе из AgS v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из AgS |
| URL | /private/v1.0/agents/{id}~~/agentInfo~~ |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание | Комментарий |
| loggedIn | int | 0...1 | Время логина оператора  Сохранить в tc/tenantusers.logged\_in | Отправить 0 - значение не пришло, не заполнено |
| loggedOff | int | 0...1 | Время логаута оператора  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logged\_off = NULL  Иначе Сохранить в tc/tenantusers.logged\_off |
| logoffReason | string | 0...1 | Причина отключения  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logoff\_reason= NULL  Иначе Сохранить в tc/tenantusers.logoff\_reason |  |
| computerIP | string | 0...1 | IP-адрес компьютера клиента  Сохранить в tc/tenantusers.comp\_ip |  |
| versionOS | string | 0...1 | Имя и версия операционной системы оператора  Сохранить в tc/tenantusers.os\_version |  |
| browser | string | 0...1 | Имя и версия браузера оператора  Сохранить в tc/tenantusers.browser\_version |  |
| timezone | string | 0...1 | Часовой пояс оператора  Сохранить в tc/tenantusers.timezone |  |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.comp\_name |  |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.os\_login |  |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.os\_domain |  |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |  |
| realComputerIP | string | 0...1 | Адрес тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |  |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.agent\_helper\_version |  |
| phoneSetNumber | string | 0...1 | Номер телефонного аппарата  Сохранить в tc/tenantusers.phone\_set\_number |  |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

Описание API "Запрос на обновление информации о пользователе из Agent Helper v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из Agent Helper |
| URL | /private/v1.0/users/{id}/agentHelperInfo |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.computer\_name |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.OS\_login |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.OS\_domain |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |
| realComputerIP | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.AgentHelper\_version |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

## v4 R2.8.0 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [1. Описание фичи](#scroll-bookmark-82)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-83)
* [3. Описание запросов](#scroll-bookmark-84)
  + [web-socket сообщения](#scroll-bookmark-85)
    - [getAgentHelperData (JsSDK->AgS)](#scroll-bookmark-86)
    - [resultGetAgentHelperData (AgS→JsSDK)](#scroll-bookmark-87)
    - [agentOpenConnect (JsSDK ->AH)](#scroll-bookmark-88)
    - [resultAgentOpenConnect (AH → JsSDK)](#scroll-bookmark-89)
    - [agentHelperConnect (JsSDK->AgS)](#scroll-bookmark-90)
    - [agentHelperConnectionState (AgS → JsSDK)](#scroll-bookmark-91)
    - [agentHelperInfo (AH → AgS)](#scroll-bookmark-92)
  + [REST-запросы](#scroll-bookmark-93)
    - [/private/v1.0/users/{id}](#scroll-bookmark-94)

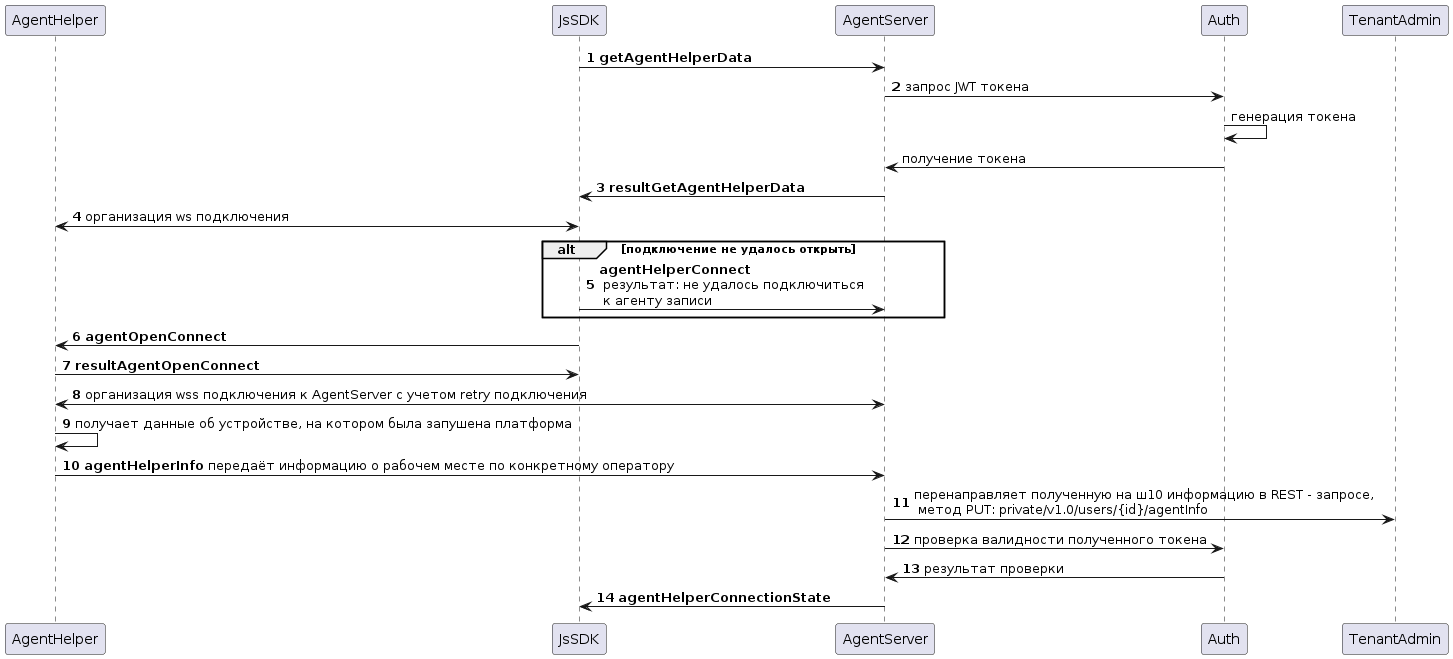
**ID фичи в JIRA**

[Epic SCPL-10511](https://jira.sberbank.ru/browse/SCPL-10511?src=confmacro) - Передача параметров сессии оператора в админку/в отчетность/в логи **done**

### 1. Описание фичи

При старте операторской сессии необходимо инициировать процесс подключения агента записи экрана оператора для получения команд управления логикой записи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | JsSDK инициирует обращение к AgentHelper за информацией по подключению к агенту записи экранов | JsSDK направляет запрос [*getAgentHelperData*](#scroll-bookmark-95) |
| 2 | AgentServer обращается в Auth сервис за генерацией JWT-токена для подключения AgentHelper | Auth сервис генерирует JWT-токен и передает его AgentServer'у |
| 3 | AgentServer отправляет в сторону JsSDK запрос на подключение агента записи | AgentServer передает [*resultGetAgentHelperData*](#scroll-bookmark-96) с параметрами подключения к AgentHelper, в котором передается   * URI AgentHelper, по которому должен JsSDK должен запрашивать подключение * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен |
| 4 | JsSDK инициирует обращение к AgentHelper по полученному в сообщении URI | Осуществляется попытка открыть незащищенное web-socket подключение   * если подключение неуспешное, то осуществляется переход на шаг 5 * если подключение успешное, то осуществляется переход на шаг 6   В сторону рабочего места отдается уведомление о статусе открытия ws подключения к AgentHelper |
| 5 | При неуспешной попытке подключения к AgentHelper JsSDK передает результат исполнения запроса в AgentServer | AgentServer получает ответ с неуспешным результатом обращения в AgentHelper [*agentHelperConnect*](#scroll-bookmark-97) |
| 6 | После установления ws соединения JsSDK направляет сообщение с информацией о подключении к AgentServer | JsSDK передает ws сообщение [*agentOpenConnect*](#scroll-bookmark-98) , в котором передает   * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен   из полученного от AgentServer сообщения (см шаг 3) |
| 7 | JsSDK передает AgentServer результат обработки запроса на подключение AgentHelper к AgentServer | AgentHelper передает в JsSDK сообщение  [*resultAgentOpenConnect*](#scroll-bookmark-99)   * result = success - AgentHelper принял запрос на организацию подключения к AgentServer. Осуществляется переход на шаг 8 * result = error - AgentHelper не смог принять в работу запрос на организацию подключения к AgentServer. Процесс завершен |
| 8 | AgentHelper пытается установить защищенное web-socket  подключение  к AgentServer  по полученному URI | AgentHelper инициирует подключение к AgentServer по указанному URI и передает полученный JWT токен |
| 9 | После установленного на шаге 8 защищенного web-socket подключение к AgentServer, AgentHelper получает данные о компьютере, на котором была запушена платформа | Список получаемых атрибутов:   * *realComputerName* - реальное имя устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $CLIENTNAME имя ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientName * *realComputerIP* - реальный  IP-адрес устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $SESSIONNAME IP-адрес ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientAddress * *computerName* - имя устройства, на котором запущена платформа SCPL (если используется ВАРМ, то имя компьютера ВАРМ). При подключении с АРМ значение совпадает с realComputerName. Полученное по средствам func Hostname * *osLogin* - логин в операционную система устройства, с которого подключен оператор. Значение полученное в переменной окружения $USERNAME или по средствам winAPI * *osDomain* - домен устройства, с которого подключен оператор (по средствам winAPI). * *versionAgentHelper* - версия AgentHelper, если он был обнаружен на рабочем месте. |
| 10 | AgentHelper передаёт в сообщении [***agentHelperInfo***](#scroll-bookmark-100)информацию о рабочем месте оператора | AgentServer получает данные и сохраняет из у себя локально |
| 11 | AgentServer перенаправляет полученную на ш10 информацию в REST - запросе, метод PUT: [/private/v1.0/users/{id}/agentInfo](#scroll-bookmark-101). В случае если ответ !=200 AgentServer делает ещё одну попытку запроса. |  |
| 12 | AgentServer проверяет валидность полученного от AgentHelper токена | * Если токен валидный, то осуществляется переход на шаг 10 * Если токен невалидный , то AgentHelper приостанавливает попытки реконнекта и осуществляется переход на шаг 11 |
| 13 | AgentHelper пытается установить устанавливается защищенное web-socket подключение с AgentServer | * Если соединение установлено, то AgentServer сохраняет параметры AgentHelper, который к нему подключился в связке с текущей операторской сессией. Осуществляется переход на шаг 11 * Если соединение не установлено, то осуществляется повторная попытка через заданное время. Период повторных попыток подключения получен AgentHelper'ом в сообщении [*getAgentHelperData*](#scroll-bookmark-95)на шаге 1. |
| 14 | AgentServer передает в JsSDK информацию об актуальном состоянии подключение агента записи экраном | AgentServer передает в JsSDKсообщение [***agentHelperConnectionState***](#scroll-bookmark-102)   * connectionState = active - AgentHelper подключен * connectionState = inactive - AgentHelper не подключен   + при деактивации подключения AgentServer в errorDesc передает ошибку  (по возможности реализовать передачу ошибки из-за разрыва wed-socket подключения (**connectionLost**) и из-за невалидного токена (**tokenInvalid**)   (предупреждение) AgentServer уведомляет JsSDK каждый раз при изменении состояния подключения |

### 3. Описание запросов

#### web-socket сообщения

##### getAgentHelperData (JsSDK->AgS)

getAgentHelperData

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *getAgentHelperData*- Запрос на передачу данных для подключения к агенту записи экранов пользователя | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getAgentHelperData",  "description": "Запрос на передачу данных для подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getAgentHelperData",  "version": "1.0",  "transaction": "string"  } |

##### resultGetAgentHelperData (AgS→JsSDK)

resultGetAgentHelperData

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultGetAgentHelperData*- Передача данных для инициализации подключения к агенту записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction на запрос *getAgentHelperData* | [0...1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса. Заполняется только для result = error | [0...1] |
|  | agentHelperURI | string | Адрес для подключения к AgentHelper  Обязательно для result = success | [0...1] |
|  | agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов  Обязательно для result = success | [0...1] |
|  | screenServer | string | Адрес screenServer для отправки потока  Обязательно для result = success | [0...1] |
|  | JWTToken | string | Аутентификационный токен  Обязательно для result = success | [0...1] |
|  | retryDelay | string | Период повторения попытки установления подключения к AgentServer  Обязательно для result = success | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetAgentHelperData",  "description": "Передача данных для инициализации подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  },  "agentHelperURI": {  "type": "string"  },  "agentServerURI": {  "type": "string"  },  "screenServer": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "retryDelay":{  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultGetAgentHelperData",  "version": "1.0",  "transaction": "string",  "data":{  "result":"success",  "agentHelperURI":"XXX",  "agentServerURI":"XXX",  "screenServer":"screenServer",  "JWTToken":"XXX",  "retryDelay": "1"  }  } |

##### agentOpenConnect (JsSDK ->AH)

agentOpenConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *agentOpenConnect*- сообщение для подключения агента записи экранов к AgentServer | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0...1] |
| data | object | Тело сообщения | [1] |
| agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов | [1] |
| JWTToken | string | Аутентификационный токен | [1] |
| userID | UUID | Идентификатор пользователя | [1] |
| retryDelay | string | Период повторения попытки установления подключения к AgentServer | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentOpenConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "agentServerURI": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "userID": {  "type": "string"  },  "retryDelay": {  "type": "string"  }  },  "required": [  "agentServerURI",  "JWTToken",  "userID",  "retryDelay"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "agentServerURI":"XXX",  "JWTToken":"WWW",  "userID": "UUID",  "retryDelay":"1"  }  } |

##### resultAgentOpenConnect (AH → JsSDK)

resultAgentOpenConnect

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### agentHelperConnect (JsSDK->AgS)

agentHelperConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *agentHelperConnect*- результат подключения агента записи | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |
| result | string | Результат исполнения запроса   * success * error | [1] |
| resultDesc | string | Причина неуспешного подключения. Заполняется только для result = error  Возможные значения:  "AgentHelper not found" - не удалось подключиться к агенту записи  "connection failedf" - агент записи не смог установить подключение к AgentServer  "invalid token" - аутентификационный токен невалиден | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "result":"error",  "resultDesc":"invalid token"  }  } |

##### agentHelperConnectionState (AgS → JsSDK)

agentHelperConnectionState

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  agentHelperConnectionState - состояние подключения агента записи экрана в рамках сессии текущего оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string |  | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | userID | string | Идентификатор пользователя | [1] |
|  | connectionState | string | Текущее состояние подключения к агенту записи экранов  Возможные значения:   * active * inactive | [1] |
|  | sessionID | string | Идентификатор сессии AgentHelper | [0..1] |
|  | helperVersion | string | текущая версия агента записи экранов, использующаяся на рабочем месте пользователя | [0..1] |
|  | resultDesc | string | описание ошибки  только при connectionState=inactive | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnectionState",  "description": "состояние подключения агента записи экрана в рамках сессии текущего оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "userID": {  "type": "string"  },  "sessionID": {  "type": "string"  },          "connectionState": {  "type": "string",  "enum":[  "active",  "inactive"  ]  },  "helperVersion": {  "type": "string"  },         "resultDesc": {  "type": "string"  }  },  "required": [  "userID",  "connectionState"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnectionState",  "version": "1.0",  "transaction": "string",  "data": {  "userID":"UUID",  "sessionID":"UUID",  "connectionState":"active",  "helperVesion": "2.7.0"  }  } |

##### agentHelperInfo (AH → AgS)

agentHelperInfo

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *agentHelperInfo*- передача данных о рабочему месте оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | computerName | string | Имя компьютера, в котором запущена платформа, если используется ВАРМ. При подключении с АРМ значение совпадает с realComputerName | [0...1] |
|  | realComputerName | string | Реальное имя устройства, с которого подключен оператор | [0...1] |
|  | realComputerIP | string | Реальный  IP-адрес устройства, с которого подключен оператор | [0...1] |
|  | osLogin | string | Операционная система устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | osDomain | string | Домен устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | versionAgentHelper | string | Версия AgentHelper, если он был обнаружен на рабочем месте. | [0...1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperInfo",  "description": "передача данных о рабочему месте оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "computerName": {  "type": "string"  },  "realComputerName": {  "type": "string"  },  "realComputerIP": {  "type": "string"  },  "osLogin": {  "type": "string"  },  "osDomain": {  "type": "string"  },  "versionAgentHelper": {  "type": "string"  }  }  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "agentHelperInfo",  "version": "v1.0",  "data": {  "computerName": "ccc-fv-452",  "realComputerName": "cac-jo-257",  "realComputerIP": "1.2.3.4",  "osLogin": "0123456789",  "osDomain": "GAMMA.SBER.RU",  "versionAgentHelper": "AgentHelper 1.9.1.14"  }  } |

#### REST-запросы

##### /private/v1.0/users/{id}

/private/v1.0/users/{id}

|  |  |  |  |
| --- | --- | --- | --- |
| № версии | Задача в JIRA | Описание изменений | Реализовано |
| v1.0 | [Task SCPL-5090](https://jira.sberbank.ru/browse/SCPL-5090?src=confmacro) - [Space] SA: получение из отчётности/AgS информации для отображении карточке оператора **closed** | Эндпоинты для передачи информации по пользователю из AgS | **not ready** |
| v1.1 | [Story SCPL-13550](https://jira.sberbank.ru/browse/SCPL-13550?src=confmacro) - [Space] Конфигурирование IP телефонов в Universe **done** | Добавлен атрибут нового ТА | **not ready** |
| v1.2 | [Story SCPL-13373](https://jira.sberbank.ru/browse/SCPL-13373?src=confmacro) - [Space] 2.11. Скорректировать логику отображения времени логаута пользователя **done** | Скорректировать логику отображения времени логаута пользователя | **not ready** |

Описание API "Запрос на обновление информации о пользователе из AgS v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из AgS |
| URL | /private/v1.0/agents/{id}~~/agentInfo~~ |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание | Комментарий |
| loggedIn | int | 0...1 | Время логина оператора  Сохранить в tc/tenantusers.logged\_in | Отправить 0 - значение не пришло, не заполнено |
| loggedOff | int | 0...1 | Время логаута оператора  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logged\_off = NULL  Иначе Сохранить в tc/tenantusers.logged\_off |
| logoffReason | string | 0...1 | Причина отключения  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logoff\_reason= NULL  Иначе Сохранить в tc/tenantusers.logoff\_reason |  |
| computerIP | string | 0...1 | IP-адрес компьютера клиента  Сохранить в tc/tenantusers.comp\_ip |  |
| versionOS | string | 0...1 | Имя и версия операционной системы оператора  Сохранить в tc/tenantusers.os\_version |  |
| browser | string | 0...1 | Имя и версия браузера оператора  Сохранить в tc/tenantusers.browser\_version |  |
| timezone | string | 0...1 | Часовой пояс оператора  Сохранить в tc/tenantusers.timezone |  |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.comp\_name |  |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.os\_login |  |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.os\_domain |  |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |  |
| realComputerIP | string | 0...1 | Адрес тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |  |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.agent\_helper\_version |  |
| phoneSetNumber | string | 0...1 | Номер телефонного аппарата  Сохранить в tc/tenantusers.phone\_set\_number |  |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

Описание API "Запрос на обновление информации о пользователе из Agent Helper v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из Agent Helper |
| URL | /private/v1.0/users/{id}/agentHelperInfo |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.computer\_name |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.OS\_login |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.OS\_domain |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |
| realComputerIP | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.AgentHelper\_version |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

## v5 R2.11 Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"

* [1. Описание фичи](#scroll-bookmark-104)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-105)
* [3. Описание запросов](#scroll-bookmark-106)
  + [web-socket сообщения](#scroll-bookmark-107)
    - [getAgentHelperData (JsSDK->AgS)](#scroll-bookmark-108)
    - [resultGetAgentHelperData (AgS→JsSDK)](#scroll-bookmark-109)
    - [agentOpenConnect (JsSDK ->AH)](#scroll-bookmark-110)
    - [resultAgentOpenConnect (AH → JsSDK)](#scroll-bookmark-111)
    - [agentHelperConnect (JsSDK->AgS)](#scroll-bookmark-112)
    - [agentHelperConnectionState (AgS → JsSDK)](#scroll-bookmark-113)
    - [agentHelperInfo (AH → AgS)](#scroll-bookmark-114)
  + [REST-запросы](#scroll-bookmark-115)
    - [/private/v1.0/users/{id}](#scroll-bookmark-116)

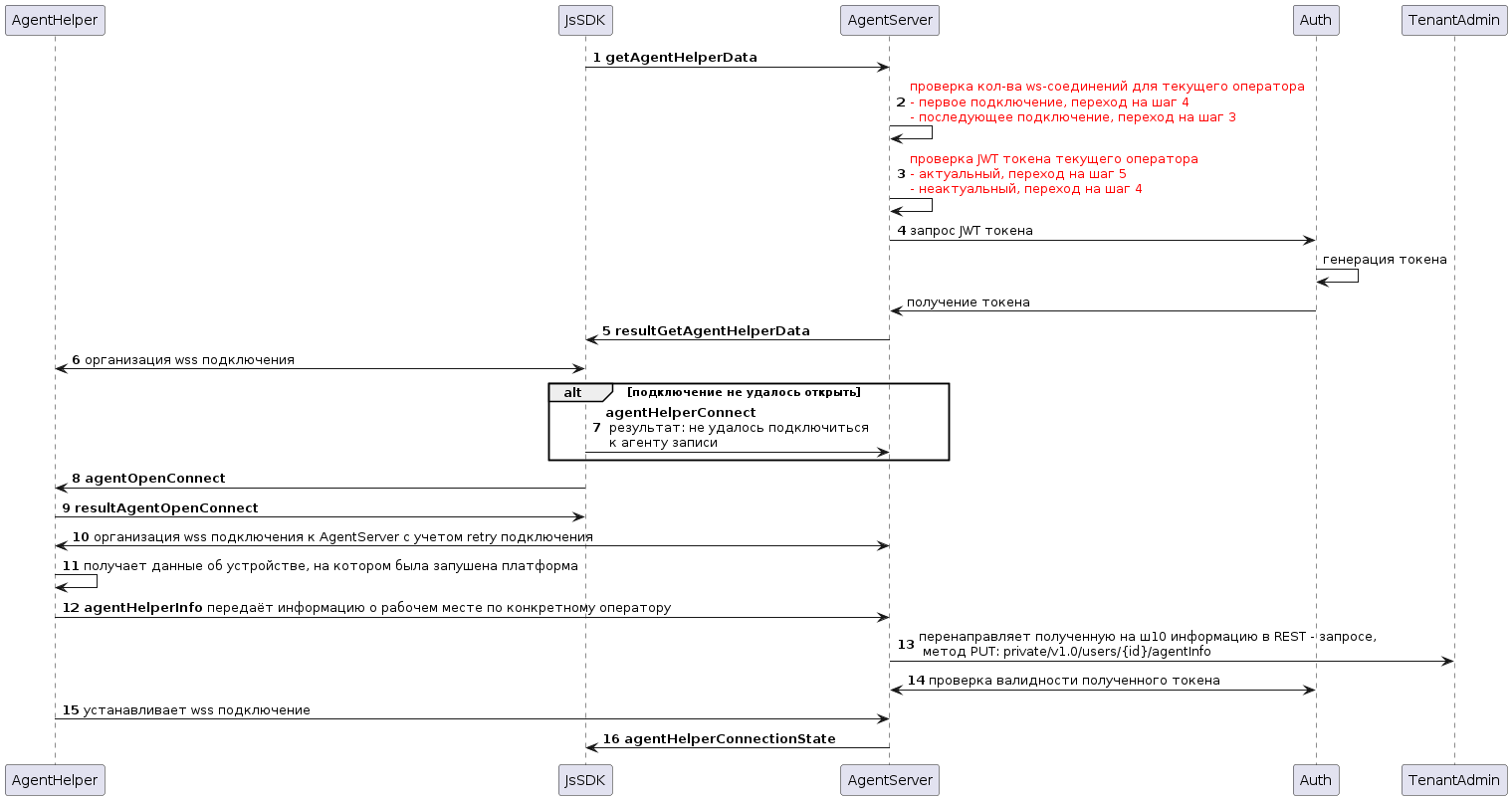
**ID фичи в JIRA**

[Epic SCPL-15271](https://jira.sberbank.ru/browse/SCPL-15271?src=confmacro) - Работа с JSSDK из нескольких РМ **done**

### 1. Описание фичи

При старте операторской сессии необходимо инициировать процесс подключения агента записи экрана оператора для получения команд управления логикой записи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | JsSDK инициирует обращение к AgentHelper за информацией по подключению к агенту записи экранов | JsSDK направляет запрос [*getAgentHelperData*](#scroll-bookmark-117) |
| 2 | AgentServer проверяет количество ws-соединений для оператора | * первое подключение, переход на **шаг 4** * последующее подключение, переход на **шаг 3** |
| 3 | AgentServer проверяет актуальность токена | * актуальный, переход на **шаг 5** * неактуальный, переход на **шаг 4** |
| 4 | AgentServer обращается в Auth сервис за генерацией JWT-токена для подключения AgentHelper. Получив ответ от Auth-сервис, AgentServer сохраняет полученный токен. | Auth-сервис генерирует JWT-токен и передает его AgentServer'у |
| 5 | AgentServer отправляет в сторону JsSDK запрос на подключение агента записи | AgentServer передает [*resultGetAgentHelperData*](#scroll-bookmark-118) с параметрами подключения к AgentHelper, в котором передается   * URI AgentHelper, по которому должен JsSDK должен запрашивать подключение * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен |
| 6 | JsSDK инициирует обращение к AgentHelper по полученному в сообщении URI   * Для организации нового подключения или переподключения для текущего wss-соедиения использовать broadcast. * Для подключения к другому SDK при разрыве текущего wss-соединения использовать random. | Осуществляется попытка открыть незащищенное web-socket подключение   * если подключение неуспешное, то осуществляется переход на **шаг 7** * если подключение успешное, то осуществляется переход на **шаг 8**   В сторону рабочего места отдается уведомление о статусе открытия ws подключения к AgentHelper |
| 7 | При неуспешной попытке подключения к AgentHelper JsSDK передает результат исполнения запроса в AgentServer | AgentServer получает ответ с неуспешным результатом обращения в AgentHelper [*agentHelperConnect*](#scroll-bookmark-119) |
| 8 | После установления ws соединения JsSDK направляет сообщение с информацией о подключении к AgentServer | JsSDK передает ws сообщение [*agentOpenConnect*](#scroll-bookmark-120) , в котором передает   * URI AgentServer, по которому должен обращаться AgentHelper для установки соединения * JWT-токен   из полученного от AgentServer сообщения (см шаг 5) |
| 9 | JsSDK передает AgentServer результат обработки запроса на подключение AgentHelper к AgentServer | AgentHelper передает в JsSDK сообщение  [*resultAgentOpenConnect*](#scroll-bookmark-121)   * result = success - AgentHelper принял запрос на организацию подключения к AgentServer. Осуществляется переход на шаг 10 * result = error - AgentHelper не смог принять в работу запрос на организацию подключения к AgentServer. Процесс завершен |
| 10 | AgentHelper пытается установить защищенное web-socket  подключение  к AgentServer  по полученному URI | AgentHelper инициирует подключение к AgentServer по указанному URI и передает полученный JWT токен |
| 11 | После установленного на шаге 10 защищенного web-socket подключение к AgentServer, AgentHelper получает данные о компьютере, на котором была запушена платформа | Список получаемых атрибутов:   * *realComputerName* - реальное имя устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $CLIENTNAME имя ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientName * *realComputerIP* - реальный  IP-адрес устройства, с которого было физическое подключение/отключение оператора к платформе SCPL. Значение полученное в переменной окружения $SESSIONNAME IP-адрес ПК/тонкого клиента или по средствам winAPI, при его отсутствии или не возможности получения через winAPI, брать значение из реестра. Необходимо учесть, что для разных сессий путь может отличаться (после предоставления пути заказчиком, путь задаётся в config-файле в зависимости от инстанса):   + для ЦКР банка и экосистемы: HKEY\_LOCAL\_MACHINE\SOFTWARE\Citrix\ICA\Session\1\Connection\ClientAddress * *computerName* - имя устройства, на котором запущена платформа SCPL (если используется ВАРМ, то имя компьютера ВАРМ). При подключении с АРМ значение совпадает с realComputerName. Полученное по средствам func Hostname * *osLogin* - логин в операционную система устройства, с которого подключен оператор. Значение полученное в переменной окружения $USERNAME или по средствам winAPI * *osDomain* - домен устройства, с которого подключен оператор (по средствам winAPI). * *versionAgentHelper* - версия AgentHelper, если он был обнаружен на рабочем месте. |
| 12 | AgentHelper передаёт в сообщении [***agentHelperInfo***](#scroll-bookmark-122)информацию о рабочем месте оператора | AgentServer получает данные и сохраняет из у себя локально |
| 13 | AgentServer перенаправляет полученную на ш11 информацию в REST - запросе, метод PUT: [/private/v1.0/users/{id}/agentInfo](#scroll-bookmark-123). В случае если ответ !=200 AgentServer делает ещё одну попытку запроса. |  |
| 14 | AgentServer проверяет валидность полученного от AgentHelper токена | * Если токен валидный, то осуществляется переход на шаг 15 * Если токен невалидный , то AgentHelper приостанавливает попытки реконнекта и осуществляется переход на шаг 16 |
| 15 | AgentHelper пытается установить устанавливается защищенное web-socket подключение с AgentServer | * Если соединение установлено, то AgentServer сохраняет параметры AgentHelper, который к нему подключился в связке с текущей операторской сессией. Осуществляется переход на шаг 16 * Если соединение не установлено, то осуществляется повторная попытка через заданное время. Период повторных попыток подключения получен AgentHelper'ом в сообщении [*getAgentHelperData*](#scroll-bookmark-117)на шаге 1. |
| 16 | AgentServer передает в JsSDK информацию об актуальном состоянии подключение агента записи экраном | AgentServer передает в JsSDKсообщение [***agentHelperConnectionState***](#scroll-bookmark-124)   * connectionState = active - AgentHelper подключен * connectionState = inactive - AgentHelper не подключен   + при деактивации подключения AgentServer в errorDesc передает ошибку  (по возможности реализовать передачу ошибки из-за разрыва wed-socket подключения (**connectionLost**) и из-за невалидного токена (**tokenInvalid**)   (предупреждение) AgentServer уведомляет JsSDK каждый раз при изменении состояния подключения |

### 3. Описание запросов

#### web-socket сообщения

##### getAgentHelperData (JsSDK->AgS)

getAgentHelperData

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *getAgentHelperData*- Запрос на передачу данных для подключения к агенту записи экранов пользователя | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getAgentHelperData",  "description": "Запрос на передачу данных для подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getAgentHelperData",  "version": "1.0",  "transaction": "string"  } |

##### resultGetAgentHelperData (AgS→JsSDK)

resultGetAgentHelperData

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultGetAgentHelperData*- Передача данных для инициализации подключения к агенту записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction на запрос *getAgentHelperData* | [0...1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса. Заполняется только для result = error | [0...1] |
|  | agentHelperURI | string | Адрес для подключения к AgentHelper  Обязательно для result = success | [0...1] |
|  | agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов  Обязательно для result = success | [0...1] |
|  | screenServer | string | Адрес screenServer для отправки потока  Обязательно для result = success | [0...1] |
|  | JWTToken | string | Аутентификационный токен  Обязательно для result = success | [0...1] |
|  | retryDelay | string | Период повторения попытки установления подключения к AgentServer  Обязательно для result = success | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetAgentHelperData",  "description": "Передача данных для инициализации подключения к агенту записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  },  "agentHelperURI": {  "type": "string"  },  "agentServerURI": {  "type": "string"  },  "screenServer": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "retryDelay":{  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultGetAgentHelperData",  "version": "1.0",  "transaction": "string",  "data":{  "result":"success",  "agentHelperURI":"XXX",  "agentServerURI":"XXX",  "screenServer":"screenServer",  "JWTToken":"XXX",  "retryDelay": "1"  }  } |

##### agentOpenConnect (JsSDK ->AH)

agentOpenConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *agentOpenConnect*- сообщение для подключения агента записи экранов к AgentServer | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0...1] |
| data | object | Тело сообщения | [1] |
| agentServerURI | string | Адрес AgentServer, к которому должен запрашивать подключение агент записи экранов | [1] |
| JWTToken | string | Аутентификационный токен | [1] |
| userID | UUID | Идентификатор пользователя | [1] |
| retryDelay | string | Период повторения попытки установления подключения к AgentServer | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentOpenConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "agentServerURI": {  "type": "string"  },  "JWTToken": {  "type": "string"  },  "userID": {  "type": "string"  },  "retryDelay": {  "type": "string"  }  },  "required": [  "agentServerURI",  "JWTToken",  "userID",  "retryDelay"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "agentServerURI":"XXX",  "JWTToken":"WWW",  "userID": "UUID",  "retryDelay":"1"  }  } |

##### resultAgentOpenConnect (AH → JsSDK)

resultAgentOpenConnect

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultAgentOpenConnect - результат обработки запроса на подключение агента записи экранов к AgentServer | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события agentOpenConnect | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения запроса  Возможные значения:   * success - запрос принят * error - запрос не принят | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultAgentOpenConnect",  "description": "результат обработки запроса на подключение агента записи экранов к AgentServer",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultAgentOpenConnect",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### agentHelperConnect (JsSDK->AgS)

agentHelperConnect

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentServer  *agentHelperConnect*- результат подключения агента записи | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string | Ответный transaction на запрос данных | [0...1] |
| data | object | Тело сообщения | [1] |
| result | string | Результат исполнения запроса   * success * error | [1] |
| resultDesc | string | Причина неуспешного подключения. Заполняется только для result = error  Возможные значения:  "AgentHelper not found" - не удалось подключиться к агенту записи  "connection failedf" - агент записи не смог установить подключение к AgentServer  "invalid token" - аутентификационный токен невалиден | [0...1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnect",  "description": "результат подключения агента записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnect",  "version": "1.0",  "transaction": "string",  "data":{  "result":"error",  "resultDesc":"invalid token"  }  } |

##### agentHelperConnectionState (AgS → JsSDK)

agentHelperConnectionState

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  agentHelperConnectionState - состояние подключения агента записи экрана в рамках сессии текущего оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string |  | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | userID | string | Идентификатор пользователя | [1] |
|  | connectionState | string | Текущее состояние подключения к агенту записи экранов  Возможные значения:   * active * inactive | [1] |
|  | sessionID | string | Идентификатор сессии AgentHelper | [0..1] |
|  | helperVersion | string | текущая версия агента записи экранов, использующаяся на рабочем месте пользователя | [0..1] |
|  | resultDesc | string | описание ошибки  только при connectionState=inactive | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperConnectionState",  "description": "состояние подключения агента записи экрана в рамках сессии текущего оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "userID": {  "type": "string"  },  "sessionID": {  "type": "string"  },          "connectionState": {  "type": "string",  "enum":[  "active",  "inactive"  ]  },  "helperVersion": {  "type": "string"  },         "resultDesc": {  "type": "string"  }  },  "required": [  "userID",  "connectionState"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"agentHelperConnectionState",  "version": "1.0",  "transaction": "string",  "data": {  "userID":"UUID",  "sessionID":"UUID",  "connectionState":"active",  "helperVesion": "2.7.0"  }  } |

##### agentHelperInfo (AH → AgS)

agentHelperInfo

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *agentHelperInfo*- передача данных о рабочему месте оператора | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | computerName | string | Имя компьютера, в котором запущена платформа, если используется ВАРМ. При подключении с АРМ значение совпадает с realComputerName | [0...1] |
|  | realComputerName | string | Реальное имя устройства, с которого подключен оператор | [0...1] |
|  | realComputerIP | string | Реальный  IP-адрес устройства, с которого подключен оператор | [0...1] |
|  | osLogin | string | Операционная система устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | osDomain | string | Домен устройства, с которого подключен оператор (вопрос) | [0...1] |
|  | versionAgentHelper | string | Версия AgentHelper, если он был обнаружен на рабочем месте. | [0...1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "agentHelperInfo",  "description": "передача данных о рабочему месте оператора",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "computerName": {  "type": "string"  },  "realComputerName": {  "type": "string"  },  "realComputerIP": {  "type": "string"  },  "osLogin": {  "type": "string"  },  "osDomain": {  "type": "string"  },  "versionAgentHelper": {  "type": "string"  }  }  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "agentHelperInfo",  "version": "v1.0",  "data": {  "computerName": "ccc-fv-452",  "realComputerName": "cac-jo-257",  "realComputerIP": "1.2.3.4",  "osLogin": "0123456789",  "osDomain": "GAMMA.SBER.RU",  "versionAgentHelper": "AgentHelper 1.9.1.14"  }  } |

#### REST-запросы

##### /private/v1.0/users/{id}

/private/v1.0/users/{id}

|  |  |  |  |
| --- | --- | --- | --- |
| № версии | Задача в JIRA | Описание изменений | Реализовано |
| v1.0 | [Task SCPL-5090](https://jira.sberbank.ru/browse/SCPL-5090?src=confmacro) - [Space] SA: получение из отчётности/AgS информации для отображении карточке оператора **closed** | Эндпоинты для передачи информации по пользователю из AgS | **not ready** |
| v1.1 | [Story SCPL-13550](https://jira.sberbank.ru/browse/SCPL-13550?src=confmacro) - [Space] Конфигурирование IP телефонов в Universe **done** | Добавлен атрибут нового ТА | **not ready** |
| v1.2 | [Story SCPL-13373](https://jira.sberbank.ru/browse/SCPL-13373?src=confmacro) - [Space] 2.11. Скорректировать логику отображения времени логаута пользователя **done** | Скорректировать логику отображения времени логаута пользователя | **not ready** |

Описание API "Запрос на обновление информации о пользователе из AgS v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из AgS |
| URL | /private/v1.0/agents/{id}~~/agentInfo~~ |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание | Комментарий |
| loggedIn | int | 0...1 | Время логина оператора  Сохранить в tc/tenantusers.logged\_in | Отправить 0 - значение не пришло, не заполнено |
| loggedOff | int | 0...1 | Время логаута оператора  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logged\_off = NULL  Иначе Сохранить в tc/tenantusers.logged\_off |
| logoffReason | string | 0...1 | Причина отключения  Если нет переданного атрибута loggedOff, то необходимо записывать tc/tenant\_users.logoff\_reason= NULL  Иначе Сохранить в tc/tenantusers.logoff\_reason |  |
| computerIP | string | 0...1 | IP-адрес компьютера клиента  Сохранить в tc/tenantusers.comp\_ip |  |
| versionOS | string | 0...1 | Имя и версия операционной системы оператора  Сохранить в tc/tenantusers.os\_version |  |
| browser | string | 0...1 | Имя и версия браузера оператора  Сохранить в tc/tenantusers.browser\_version |  |
| timezone | string | 0...1 | Часовой пояс оператора  Сохранить в tc/tenantusers.timezone |  |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.comp\_name |  |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.os\_login |  |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.os\_domain |  |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |  |
| realComputerIP | string | 0...1 | Адрес тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |  |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.agent\_helper\_version |  |
| phoneSetNumber | string | 0...1 | Номер телефонного аппарата  Сохранить в tc/tenantusers.phone\_set\_number |  |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

Описание API "Запрос на обновление информации о пользователе из Agent Helper v1.0"

|  |  |
| --- | --- |
| Наименование | Обновление информации о пользователе из Agent Helper |
| URL | /private/v1.0/users/{id}/agentHelperInfo |
| Формат | REST |
| Режим | Синхронный |
| Метод | PUT |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |
| --- | --- | --- | --- |
| Атрибут | Тип | Кратность | Описание |
| computerName | string | 0...1 | Имя компьютера  Сохранить в tc/tenantusers.computer\_name |
| osLogin | string | 0...1 | Системный логин в операционной системе  Сохранить в tc/tenantusers.OS\_login |
| osDomain | string | 0...1 | Домен пользователя  Сохранить в tc/tenantusers.OS\_domain |
| realComputerName | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_name |
| realComputerIP | string | 0...1 | Имя тонкого клиента  Сохранить в tc/tenantusers.real\_comp\_ip |
| versionAgentHelper | string | 0...1 | Версия AgentHelper  Сохранить в tc/tenantusers.AgentHelper\_version |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| id | Идентификатор обновленного оператора | string | 1 |

**Коды ответов:**

|  |  |
| --- | --- |
| Ответ | Код ответа |
| Запрос успешно обработан | 200 |
| Объект не найден | 404 |
| Изменение пользователя невозможно, так как он не является оператором | 400 |

# Фича Core.AgentHelper "Запись экрана оператора"

* [v1 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-126)
* [v2 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-127)
* [v3 R 1.11.0 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-128)
* [v4 R 1.12.0 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-129)
* [v5 R 2.8 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-130)
* [v6 R3.2 Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-131)

## v1 Фича Core.AgentHelper "Запись экрана оператора"

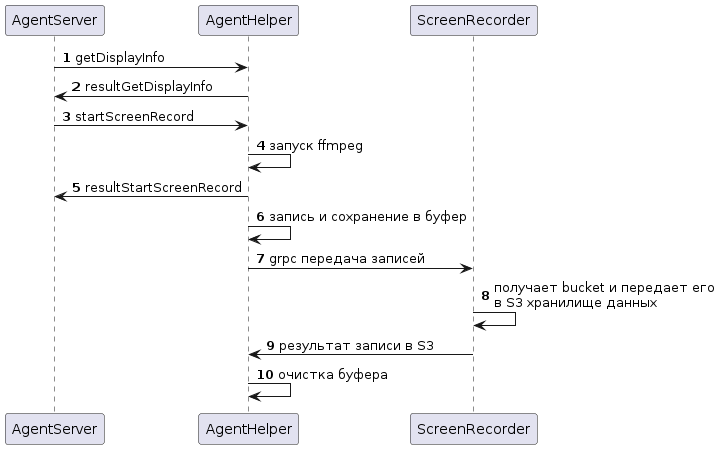
* [1. Описание фичи](#scroll-bookmark-132)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-133)
* [3. Описание запросов](#scroll-bookmark-134)
  + [web-socket сообщения](#scroll-bookmark-135)
    - [getDisplayInfo](#scroll-bookmark-136)
    - [resultGetDisplayInfo](#scroll-bookmark-137)
    - [startScreenRecord](#scroll-bookmark-138)
    - [resultStartScreenRecord](#scroll-bookmark-139)

**ID фичи в JIRA**

[Epic SCPL-3494](https://jira.sberbank.ru/browse/SCPL-3494?src=confmacro) - Запись экранов **done**

### 1. Описание фичи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer отправляет команду [***getDisplayInfo***](#scroll-bookmark-140) на получение данных о количестве подключенных мониторов и их разрешении экрана в РМ оператора | AgentHelper формирует ответ |
| 2 | AgentHelper отправляет сформированный ответ [***resultGetDisplayInfo***](#scroll-bookmark-141) с информацией об экранах мониторов в сторону AgentServer | AgentServer получает информацию об экранах оператора и формирует набор команд для начала записи экрана. |
| 3 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-142) от AgentServer |
| 4 | AgentHelper запускает ffmpeg | Приложение запущено |
| 5 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-143) |
| 6 | AgentHelper осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps | Запускается непрерывный процесс записи данных |
| 7 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 8 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 9 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 7 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 10 |
| 10 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 7 для записи нового bucket |  |

### 3. Описание запросов

#### web-socket сообщения

##### getDisplayInfo

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *getDisplayInfo*- команда на получение данных о подключенных экранах | [1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | string |  | [0..1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "getDisplayInfo",  "description": "команда на получение данных о подключенных экранах",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  }  },  "required": [  "action",  "version",  "transaction"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"getDisplayInfo",  "version":"1.0",  "transaction": "UUID"  } |

##### resultGetDisplayInfo

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.JsSDK  resultGetDisplayInfo - сообщение с информацией об экранах мониторов | [1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | | string | Ответный transaction при обработке события getDisplayInfo | [0..1] |
| data | | | object | Тело сообщения | [1] |
|  | result | | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | displayData | | Array |  | [1] |
|  |  | monitorID | int | Идентификатор монитора | [1] |
|  |  | width | int | Ширина разрешения экрана | [1] |
|  |  | height | int | Высота разрешения экрана | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetDisplayInfo",  "description": "сообщение с информацией об экранах мониторов",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "displayData": {  "type": "array",  "items": [  {  "type": "object",  "properties": {  "monitorID": {  "type": "integer"  },  "width": {  "type": "integer"  },  "height": {  "type": "integer"  }  },  "required": [  "monitorID",  "width",  "height"  ]  }  ]  }  },  "required": [  "result",  "displayData"  ]  }  },  "required": [  "action",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "resultGetDisplayInfo",  "transaction": "UUID"  "data": {  "result": "success",  "displayData": [  {  "monitorID": 115,  "width": 800,  "height": 600  }  ]  }  } |

##### startScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | string |  | [0..1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения |  |
|  | fps | int |  | [1] |
|  | interactionID | string | идентификатор взаимодействия | [0..1] |
|  | workitemID | string | идентификатор  workitem | [0..1] |
|  | userID | string | идентификатор пользователя | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "fps": {  "type": "int"  },  "interactionID": {  "type": "int"  },  "workitemID": {  "type": "int"  },  "userID": {  "type": "int"  }  },  "required": [  "fps",  "userID"  ]  }  },  "required": [  "action",  "transaction",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction":"string",  "version":"1.0",  "data":  {  "fps":"30",  "interactionID":"string",  "workitemID":"string",  "userID":"userID"  }  } |

##### resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## v2 Фича Core.AgentHelper "Запись экрана оператора"

* [1. Описание фичи](#scroll-bookmark-144)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-145)
* [3. Описание запросов](#scroll-bookmark-146)
  + [web-socket сообщения](#scroll-bookmark-147)
    - [startScreenRecord](#scroll-bookmark-148)
    - [resultStartScreenRecord](#scroll-bookmark-149)

**ID фичи в JIRA**

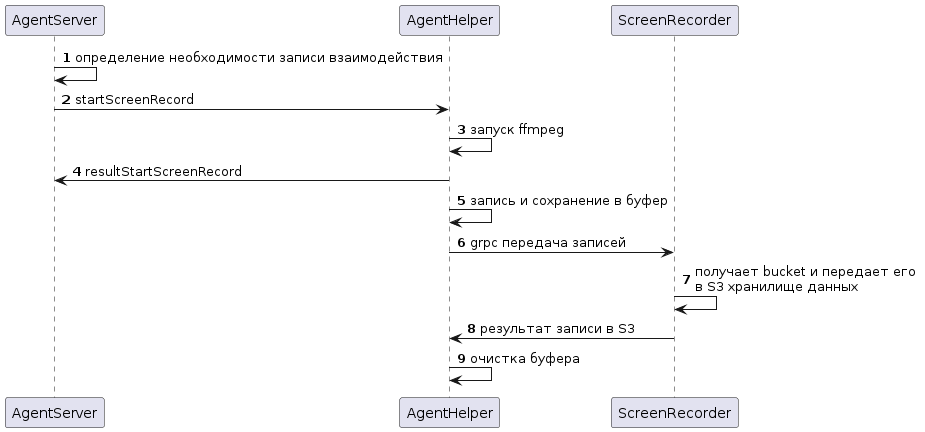
[Epic SCPL-3494](https://jira.sberbank.ru/browse/SCPL-3494?src=confmacro) - Запись экранов **done**

[Epic SCPL-5835](https://jira.sberbank.ru/browse/SCPL-5835?src=confmacro) - Правила записи экранов **done**

### 1. Описание фичи

Запись экранов оператора во время обработки взаимодействия должна осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатывается взаимодействие

### 2. Описание процесса/сервиса TO BE



Предварительно на AgentServer сохранена информация о настройках правил записи по сервисам. Процесс получения этих данных описан по [ссылке](https://confluence.sberbank.ru/x/_wtOmQI)

|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | При получении входящего взаимодействия или при инициализации исходящего взаимодействия AgentServer определяет необходимость записи экранов оператора | AgentServer получил сообщение agentConnection (перевел оператора в Ringing, создал workitem в статусе Ringing) ) от модуля WorkflowEngine  ИЛИ  AgentServer начал осуществление исходящего звонка (перевел оператора в Dialing, создал workitem в статусе Ringing):   * AgentServer определяет, надо ли записывать текущий звонок. Для этого анализирует параметр **recordingPercentage** по сервису текущего звонка - выбирает рандомное число от 1 до 100   + если выбранное число попадает в диапазон от 1 до заданного процента, то звонок записывается   + если нет, то не записывается - на этом процесс завершается * Если звонок необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingStage**:   + для recordingStage =  ringingProgress запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние Ringing+InProgress)   + для recordingStage =  progressWrapup запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Completed (записываются состояние InProgress+Wrapup)   + для recordingStage =  progress запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Wrapup или Completed (записывается только состояние InProgress)   + для recordingStage =  ringingProgressWrapup запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Completed (записываются состояние Ringing+InProgress+Wrapup)   Переход на шаг 2 осуществляется при переводе workitem в состояние, которое требует записи в соответствие с полученными правилами по сервису. |
|  | ~~AgentServer отправляет команду~~[***~~getDisplayInfo~~***](#scroll-bookmark-150) ~~на получение данных о количестве подключенных мониторов и их разрешении экрана в РМ оператора~~ | ~~AgentHelper формирует ответ~~ |
|  | ~~AgentHelper отправляет сформированный ответ~~ [***~~resultGetDisplayInfo~~***](#scroll-bookmark-151)~~с информацией об экранах мониторов в сторону AgentServer~~ | ~~AgentServer получает информацию об экранах оператора и формирует набор команд для начала записи экрана.~~ |
| 2 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-152) от AgentServer |
| 3 | AgentHelper запускает ffmpeg | Приложение запущено |
| 4 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-153) |
| 5 | AgentHelper осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps | Запускается непрерывный процесс записи данных |
| 6 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 7 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 8 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 5 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 8 |
| 9 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 7 для записи нового bucket |  |

### 3. Описание запросов

#### web-socket сообщения

##### startScreenRecord

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | | string |  | [0..1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | | object | Тело сообщения |  |
|  | interactionID | | string | идентификатор взаимодействия | [0..1] |
|  | workitemID | | string | идентификатор  workitem | [0..1] |
|  | userID | | string | идентификатор пользователя | [1] |
|  | ttl | | int | срок хранения записи в днях | [1] |
|  | startRecord | | string | время старта записи | [1] |
|  | recordParam | |  |  |  |
|  |  | fps | int | частота кадров в секунду | [1] |
|  |  | h264Preset | string | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | [1] |
|  |  | h264CFR | int | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | [1] |
|  |  | h264GOP | int | Частота ключевых кадров (H.264 GOP) | [1] |
|  |  | h264GBF | int | Максимальное количество последовательных B-кадров | [1] |
|  |  | h264ZeroLatency | boolean | Оптимизация сжатия для стриминга (H.264 ZeroLatency) | [1] |
|  |  | h264Threads | int | Количество потоков сжатия | [1] |
|  |  | FfmpegLogLevel | string | Уровень логирования ffmpeg | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactionID": {  "type": "int"  },  "workitemID": {  "type": "int"  },  "userID": {  "type": "int"  },  "recordParam": {  "type": "object",  "properties":{  "fps": {  "type": "int"  },  "h264Preset": {  "type": "string"  },  "h264CFR": {  "type": "int"  },  "h264GOP": {  "type": "int"  },  "h264GBF": {  "type": "int"  },  "h264ZeroLatency": {  "type": "boolean"  },  "h264Threads": {  "type": "int"  },  "FfmpegLogLevel": {  "type": "string"  }  },  "required": [  "fps",  "h264Preset",  "h264CFR",  "h264GOP",  "h264GBF",  "h264ZeroLatency",  "h264Threads",  "FfmpegLogLevel"  ]  }  },  "required": [  "fps",  "userID",  "recordParam"  ]  }  },  "required": [  "action",  "transaction",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction":"string",  "version":"1.0",  "data":  {  "interactionID":"string",  "workitemID":"string",  "userID":"userID",  "startRecord": "timePointer (time.Now().UTC())",  "ttl": "30",  "recordParam":  {  "fps": "5",  "h264Preset": "veryfast",  "H264CRF": "30",  "H264GOP": "250",  "H264BF": "2",  "H264ZeroLatency": " ",  "H264Threads": "0",  "FfmpegLoglevel": "error"  }  }  } |

##### resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## v3 R 1.11.0 Фича Core.AgentHelper "Запись экрана оператора"

* [1. Описание фичи](#scroll-bookmark-154)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-155)
* [3. Описание запросов](#scroll-bookmark-156)
  + [web-socket сообщения](#scroll-bookmark-157)
    - [startScreenRecord](#scroll-bookmark-158)
    - [resultStartScreenRecord](#scroll-bookmark-159)
    - [newRecBatch](#scroll-bookmark-160)
  + [Protobuf](#scroll-bookmark-161)
    - [RecCoreEvent](#scroll-bookmark-162)

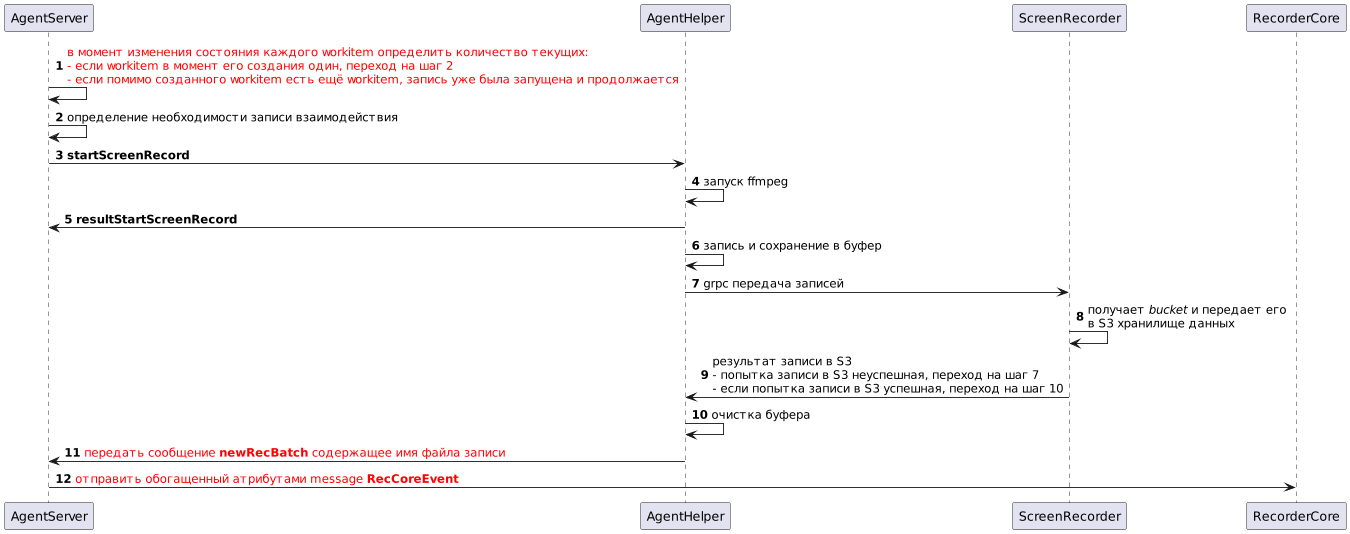
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Запись экранов оператора во время обработки взаимодействий должна осуществляться с учетом правил записи, заданных на сервисах, по которым обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



Предварительно на AgentServer сохранена информация о настройках правил записи по сервисам. Процесс получения этих данных описан по [ссылке](https://confluence.sberbank.ru/x/_wtOmQI).

|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | При получении входящего взаимодействия AgentServer получил сообщение agentConnection (перевел оператора в Ringing, создал workitem в статусе Ringing) ) от модуля WorkflowEngine или при инициализации исходящего взаимодействия AgentServer начал осуществление исходящего звонка (перевел оператора в Dialing, создал workitem в статусе Ringing), AgentServer  определяет количество текущих workitem. | Возможные варианты:   * если workitem в момент его создания один, переход на шаг 2 * если помимо созданного workitem есть ещё workitem, запись уже была запущена и продолжается. |
| 2 | AgentServer определяет необходимость записи экранов оператора согласно состояния workitem в соответствие с полученными правилами по сервису. | Возможные варианты:   * AgentServer определяет, надо ли записывать текущий звонок. Для этого анализирует параметр **recordingPercentage** по сервису текущего звонка - выбирает случайное число от 1 до 100   + если выбранное число попадает в диапазон от 1 до заданного процента, то звонок записывается   + если нет, то не записывается - на этом процесс завершается * Если звонок необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingStage**:   + для recordingStage =  ringingProgress запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние Ringing+InProgress)   + для recordingStage =  progressWrapup запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Completed (записываются состояние InProgress+Wrapup)   + для recordingStage =  progress запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Wrapup или Completed (записывается только состояние InProgress)   + для recordingStage =  ringingProgressWrapup запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Completed (записываются состояние Ringing+InProgress+Wrapup)   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует записи в соответствие с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-163) от AgentServer |
| 4 | AgentHelper запускает ffmpeg | Приложение запущено |
| 5 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-164) |
| 6 | AgentHelper осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps | Запускается непрерывный процесс записи данных |
| 7 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 8 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 9 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 7 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 10 |
| 10 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 6 для записи нового bucket |  |
| 11 | AgentHelper от момента начала записи, каждые 10 минут или при разрыве/восстановлении соединения, формирует и отправляет в сторону AgentServer сообщение [***newRecBatch***](#scroll-bookmark-165) с именем файла и результатом его записи, в котором содержится 10 минутный (при разрыве/восстановлении файл может содержать меньшее количество времени записи) отрезок записи экрана. |  |
| 12 | AgentServer обогащает полученное имя файла связанными с ним атрибутами и отправляет обогащенный protobuf [***RecCoreEvent***](#scroll-bookmark-166) в сторону RecorderCore |  |

### 3. Описание запросов

#### web-socket сообщения

##### startScreenRecord

startScreenRecord

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | | string |  | [0..1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | | object | Тело сообщения |  |
|  | interactionID | | string | идентификатор взаимодействия | [0..1] |
|  | workitemID | | string | идентификатор  workitem | [0..1] |
|  | userID | | string | идентификатор пользователя | [1] |
|  | ttl | | int | срок хранения записи в днях | [1] |
|  | startRecord | | string | время старта записи | [1] |
|  | recordParam | |  |  |  |
|  |  | fps | int | частота кадров в секунду | [1] |
|  |  | h264Preset | string | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | [1] |
|  |  | h264CFR | int | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | [1] |
|  |  | h264GOP | int | Частота ключевых кадров (H.264 GOP) | [1] |
|  |  | h264GBF | int | Максимальное количество последовательных B-кадров | [1] |
|  |  | h264ZeroLatency | boolean | Оптимизация сжатия для стриминга (H.264 ZeroLatency) | [1] |
|  |  | h264Threads | int | Количество потоков сжатия | [1] |
|  |  | FfmpegLogLevel | string | Уровень логирования ffmpeg | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactionID": {  "type": "int"  },  "workitemID": {  "type": "int"  },  "userID": {  "type": "int"  },  "recordParam": {  "type": "object",  "properties":{  "fps": {  "type": "int"  },  "h264Preset": {  "type": "string"  },  "h264CFR": {  "type": "int"  },  "h264GOP": {  "type": "int"  },  "h264GBF": {  "type": "int"  },  "h264ZeroLatency": {  "type": "boolean"  },  "h264Threads": {  "type": "int"  },  "FfmpegLogLevel": {  "type": "string"  }  },  "required": [  "fps",  "h264Preset",  "h264CFR",  "h264GOP",  "h264GBF",  "h264ZeroLatency",  "h264Threads",  "FfmpegLogLevel"  ]  }  },  "required": [  "fps",  "userID",  "recordParam"  ]  }  },  "required": [  "action",  "transaction",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction":"string",  "version":"1.0",  "data":  {  "interactionID":"string",  "workitemID":"string",  "userID":"userID",  "startRecord": "timePointer (time.Now().UTC())",  "ttl": "30",  "recordParam":  {  "fps": "5",  "h264Preset": "veryfast",  "H264CRF": "30",  "H264GOP": "250",  "H264BF": "2",  "H264ZeroLatency": " ",  "H264Threads": "0",  "FfmpegLoglevel": "error"  }  }  } |

##### resultStartScreenRecord

resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### newRecBatch

newRecBatch

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  newRecBatch  - сообщение содержащее имя файла записи | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат создания файла записи. Возможные значения:   * succes * error | [1] |
|  | fileName | string | Имя файла | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "newRecBatch",  "description": "сообщение содержащее имя файла записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string",  "enum": [  "success",  "error"  ]  },  "fileName": {  "type": "string"  }  },  "required": [  "result",  "fileName"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"newRecBatch",  "version": "1.0",  "data": {  "result": "success",  "fileName": "ABCDEFG"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v4 R 1.12.0 Фича Core.AgentHelper "Запись экрана оператора"

* [1. Описание фичи](#scroll-bookmark-167)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-168)
* [3. Описание запросов](#scroll-bookmark-169)
  + [web-socket сообщения](#scroll-bookmark-170)
    - [startScreenRecord](#scroll-bookmark-171)
    - [resultStartScreenRecord](#scroll-bookmark-172)
    - [restartScreenRecord](#scroll-bookmark-173)
    - [resultRestartScreenRecord](#scroll-bookmark-174)
  + [Protobuf](#scroll-bookmark-175)
    - [RecCoreEvent](#scroll-bookmark-176)

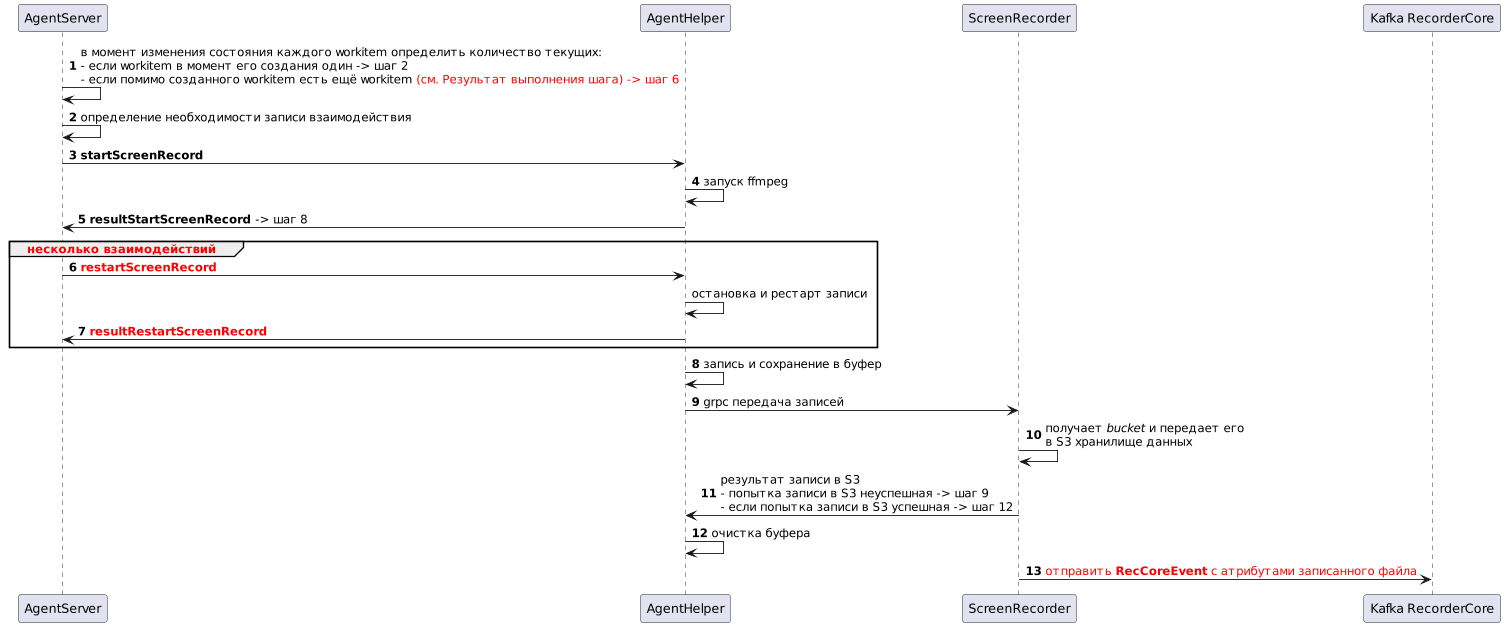
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Запись экранов оператора во время обработки взаимодействий должна осуществляться с учетом правил записи, заданных на сервисах, по которым обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



Предварительно на AgentServer сохранена информация о настройках правил записи по сервисам. Процесс получения этих данных описан по [ссылке](https://confluence.sberbank.ru/x/_wtOmQI).

|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | При получении входящего взаимодействия AgentServer получил сообщение agentConnection (перевел оператора в Ringing, создал workitem в статусе Ringing) от модуля WorkflowEngine или при инициализации исходящего взаимодействия AgentServer начал осуществление исходящего звонка (перевел оператора в Dialing, создал workitem в статусе Ringing), AgentServer  определяет количество текущих workitem. | Возможные варианты:   * если workitem в момент его создания один, переход на шаг 2 * если помимо созданного workitem есть ещё workitem, переход на шаг 6 |
| 2 | AgentServer определяет необходимость записи экранов оператора согласно состояния workitem в соответствие с полученными правилами по сервису. | Возможные варианты:   * AgentServer определяет, надо ли записывать текущий звонок. Для этого анализирует параметр **recordingPercentage** по сервису текущего звонка - выбирает случайное число от 1 до 100   + если выбранное число попадает в диапазон от 1 до заданного процента, то звонок записывается   + если нет, то не записывается - на этом процесс завершается * Если звонок необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingStage**:   + для recordingStage =  ringingProgress запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние Ringing+InProgress)   + для recordingStage =  progressWrapup запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Completed (записываются состояние InProgress+Wrapup)   + для recordingStage =  progress запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Wrapup или Completed (записывается только состояние InProgress)   + для recordingStage =  ringingProgressWrapup запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Completed (записываются состояние Ringing+InProgress+Wrapup)   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует записи в соответствие с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-177) от AgentServer |
| 4 | AgentHelper запускает ffmpeg | Приложение запущено |
| 5 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-178), переход на шаг 8 |
| 6 | AgentServer передает в сторону AgentHelper команду для рестарта записи экрана оператора с учётом добавленного workitem указанного в массиве workitems | AgentHelper принимает команду [***restartScreenRecord***](#scroll-bookmark-179) от AgentServer:   * останавливает запись текущего отрезка записи экрана (процесс описан в фиче ["Остановка записи экрана оператора"](#scroll-bookmark-180), шаг 6), * делает рестарт записи с учётом изменённого количества workitem и запускает процесс записи, описанный в шаге 8. |
| 7 | AgentHelper передает в AgentServer результат исполнения запроса о рестарте записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultRestartScreenRecord***](#scroll-bookmark-181) |
| 8 | AgentHelper от момента начала записи осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" (параметр *batchRecDuration* задаётся в Админке) минут или при разрыве/восстановлении соединения (при разрыве/восстановлении файл может содержать меньшее количество времени записи) | Запускается непрерывный процесс записи данных |
| 9 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 10 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 11 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 9 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 12 |
| 12 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 8 для записи нового bucket |  |
| 13 | ScreenRecorder формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-182), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### startScreenRecord

startScreenRecord

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | | string |  | [0..1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | | object | Тело сообщения |  |
|  | interactions | | array | массив идентификаторов взаимодействия | [0..1] |
|  | workitems | | array | массив идентификаторов workitem | [0..1] |
|  | userID | | string | идентификатор пользователя | [1] |
|  | ttl | | integer | срок хранения записи в днях | [1] |
|  | startRecord | | string | время старта записи | [1] |
|  | batchRecDuration | | integer | продолжительность записи пакета по времени в минутах | [1] |
|  | recordParam | |  |  |  |
|  |  | fps | integer | частота кадров в секунду | [1] |
|  |  | h264Preset | string | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | [1] |
|  |  | h264CFR | integer | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | [1] |
|  |  | h264GOP | integer | Частота ключевых кадров (H.264 GOP) | [1] |
|  |  | h264GBF | integer | Максимальное количество последовательных B-кадров | [1] |
|  |  | h264ZeroLatency | boolean | Оптимизация сжатия для стриминга (H.264 ZeroLatency) | [1] |
|  |  | h264Threads | integer | Количество потоков сжатия | [1] |
|  |  | FfmpegLogLevel | string | Уровень логирования ffmpeg | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactions": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "workitems": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "userID": {  "type": "string"  },  "startRecord": {  "type": "string"  },  "ttl": {  "type": "integer"  },  "batchRecDuration": {  "type": "integer"  },          "recordParam": {  "type": "object",  "properties": {  "fps": {  "type": "integer"  },  "h264Preset": {  "type": "string"  },  "H264CRF": {  "type": "integer"  },  "H264GOP": {  "type": "integer"  },  "H264BF": {  "type": "integer"  },  "H264ZeroLatency": {  "type": "boolean"  },  "H264Threads": {  "type": "integer"  },  "FfmpegLoglevel": {  "type": "string"  }  },  "required": [  "fps",  "h264Preset",  "H264CRF",  "H264GOP",  "H264BF",  "H264ZeroLatency",  "H264Threads",  "FfmpegLoglevel"  ]  }  },  "required": [  "userID",  "startRecord",  "ttl",  "recordParam"  ]  }  },  "required": [  "action"  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction": "UUID1",  "version":"1.0",  "data":  {  "interactionID": ["UUID2", "UUID3"],  "workitemID": ["UUID4", "UUID5"],  "userID":"userID",  "startRecord": "timePointer (time.Now().UTC())",  "ttl": 30,  "batchRecDuration": 15  "recordParam":  {  "fps": 5,  "h264Preset": "veryfast",  "H264CRF": 30,  "H264GOP": 250,  "H264BF": 2,  "H264ZeroLatency": **true**,  "H264Threads": 0,  "FfmpegLoglevel": "error"  }  }  } |

##### resultStartScreenRecord

resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### restartScreenRecord

restartScreenRecord

* [R1.11 restartScreenRecord](https://confluence.sberbank.ru/display/SCPLATFORM/R1.11+restartScreenRecord)
* [R3.1 restartScreenRecord](https://confluence.sberbank.ru/display/SCPLATFORM/R3.1+restartScreenRecord)

##### resultRestartScreenRecord

resultRestartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultRestartScreenRecord* - сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *restartScreenRecord* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultRestartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultRestartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v5 R 2.8 Фича Core.AgentHelper "Запись экрана оператора"

* [1. Описание фичи](#scroll-bookmark-183)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-184)
* [3. Описание запросов](#scroll-bookmark-185)
  + [web-socket сообщения](#scroll-bookmark-186)
    - [startScreenRecord](#scroll-bookmark-187)
    - [resultStartScreenRecord](#scroll-bookmark-188)
    - [restartScreenRecord](#scroll-bookmark-189)
    - [resultRestartScreenRecord](#scroll-bookmark-190)
  + [Protobuf](#scroll-bookmark-191)
    - [RecCoreEvent](#scroll-bookmark-192)

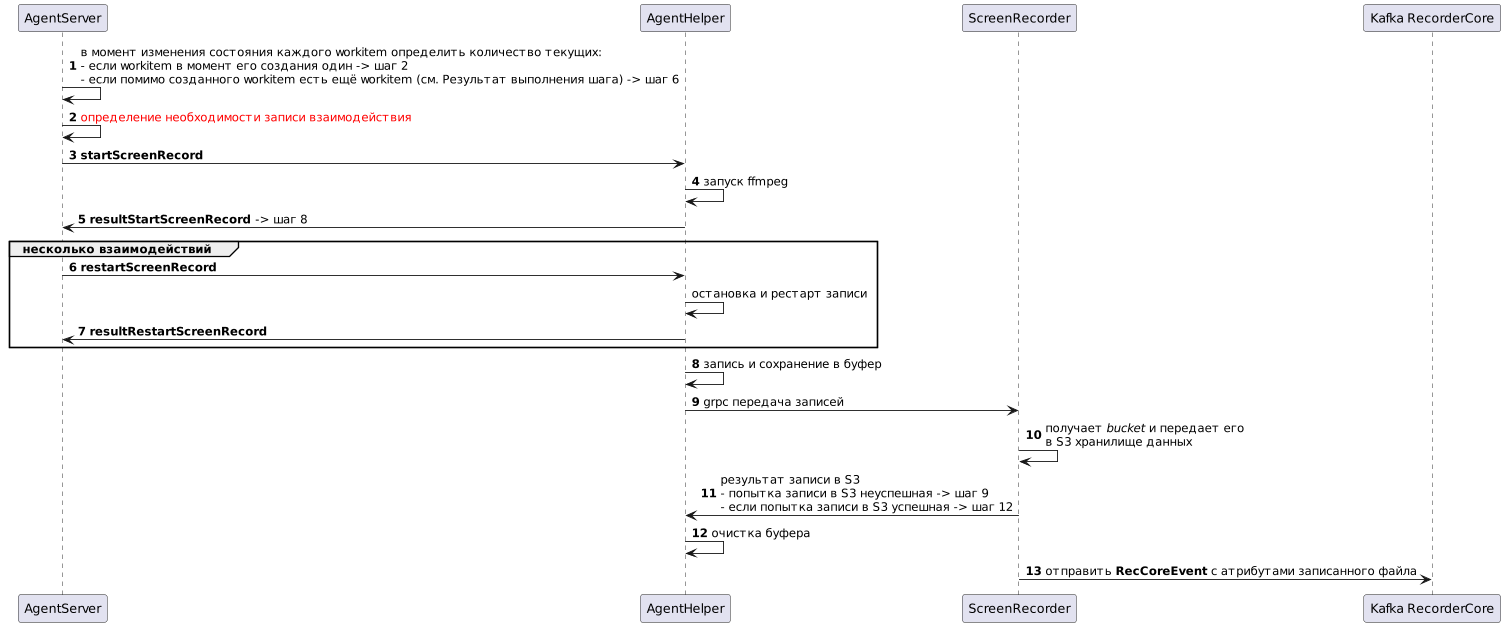
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Запись экранов оператора во время обработки взаимодействий должна осуществляться с учетом правил записи, заданных на сервисах, по которым обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



Предварительно на AgentServer сохранена информация о настройках правил записи по сервисам. Процесс получения этих данных описан по [ссылке](https://confluence.sberbank.ru/x/_wtOmQI).

|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | При получении входящего взаимодействия AgentServer получил сообщение agentConnection (перевел оператора в Ringing, создал workitem в статусе Ringing) от модуля WorkflowEngine или при инициализации исходящего взаимодействия AgentServer начал осуществление исходящего звонка (перевел оператора в Dialing, создал workitem в статусе Ringing), AgentServer  определяет количество текущих workitem. | Возможные варианты:   * если workitem в момент его создания один, переход на шаг 2 * если помимо созданного workitem есть ещё workitem, переход на шаг 6 |
| 2 | AgentServer определяет необходимость записи экранов оператора согласно состояния workitem в соответствие с полученными правилами по сервису. | Возможные варианты:   * AgentServer определяет, надо ли записывать текущий звонок. Для этого анализирует параметр **recordingPercentage** по сервису текущего звонка - выбирает случайное число от 1 до 100   + если выбранное число попадает в диапазон от 1 до заданного процента, то звонок записывается   + если нет, то не записывается - на этом процесс завершается * Если звонок необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingStage**:   + для recordingStage =  ringingProgress запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние Ringing+InProgress)   + для recordingStage =  **preparing**RingingProgress запись начинается от перехода workitem в состояние **Preparing** и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние **Preparing** + Ringing+InProgress)   + для recordingStage =  progressWrapup запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Completed (записываются состояние InProgress+Wrapup)   + для recordingStage =  progress запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Wrapup или Completed (записывается только состояние InProgress)   + для recordingStage =  ringingProgressWrapup запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Completed (записываются состояние Ringing+InProgress+Wrapup)   + для recordingStage =  **preparing**RingingProgressWrapup запись начинается от перехода workitem в состояние **Preparing** и заканчивается при переходе в состояние Completed (записываются состояние **Preparing**+Ringing+InProgress+Wrapup)   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует записи в соответствие с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-193) от AgentServer |
| 4 | AgentHelper запускает ffmpeg | Приложение запущено |
| 5 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-194), переход на шаг 8 |
| 6 | AgentServer передает в сторону AgentHelper команду для рестарта записи экрана оператора с учётом добавленного workitem указанного в массиве workitems | AgentHelper принимает команду [***restartScreenRecord***](#scroll-bookmark-195) от AgentServer:   * останавливает запись текущего отрезка записи экрана (процесс описан в фиче ["Остановка записи экрана оператора"](#scroll-bookmark-180), шаг 6), * делает рестарт записи с учётом изменённого количества workitem и запускает процесс записи, описанный в шаге 8. |
| 7 | AgentHelper передает в AgentServer результат исполнения запроса о рестарте записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultRestartScreenRecord***](#scroll-bookmark-196) |
| 8 | AgentHelper от момента начала записи осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" (параметр *batchRecDuration* задаётся в Админке) минут или при разрыве/восстановлении соединения (при разрыве/восстановлении файл может содержать меньшее количество времени записи) | Запускается непрерывный процесс записи данных |
| 9 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 10 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 11 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 9 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 12 |
| 12 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 8 для записи нового bucket |  |
| 13 | ScreenRecorder формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-197), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### startScreenRecord

startScreenRecord

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | | string |  | [0..1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | | object | Тело сообщения |  |
|  | interactions | | array | массив идентификаторов взаимодействия | [0..1] |
|  | workitems | | array | массив идентификаторов workitem | [0..1] |
|  | userID | | string | идентификатор пользователя | [1] |
|  | ttl | | integer | срок хранения записи в днях | [1] |
|  | startRecord | | string | время старта записи | [1] |
|  | batchRecDuration | | integer | продолжительность записи пакета по времени в минутах | [1] |
|  | recordParam | |  |  |  |
|  |  | fps | integer | частота кадров в секунду | [1] |
|  |  | h264Preset | string | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | [1] |
|  |  | h264CFR | integer | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | [1] |
|  |  | h264GOP | integer | Частота ключевых кадров (H.264 GOP) | [1] |
|  |  | h264GBF | integer | Максимальное количество последовательных B-кадров | [1] |
|  |  | h264ZeroLatency | boolean | Оптимизация сжатия для стриминга (H.264 ZeroLatency) | [1] |
|  |  | h264Threads | integer | Количество потоков сжатия | [1] |
|  |  | FfmpegLogLevel | string | Уровень логирования ffmpeg | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactions": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "workitems": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "userID": {  "type": "string"  },  "startRecord": {  "type": "string"  },  "ttl": {  "type": "integer"  },  "batchRecDuration": {  "type": "integer"  },          "recordParam": {  "type": "object",  "properties": {  "fps": {  "type": "integer"  },  "h264Preset": {  "type": "string"  },  "H264CRF": {  "type": "integer"  },  "H264GOP": {  "type": "integer"  },  "H264BF": {  "type": "integer"  },  "H264ZeroLatency": {  "type": "boolean"  },  "H264Threads": {  "type": "integer"  },  "FfmpegLoglevel": {  "type": "string"  }  },  "required": [  "fps",  "h264Preset",  "H264CRF",  "H264GOP",  "H264BF",  "H264ZeroLatency",  "H264Threads",  "FfmpegLoglevel"  ]  }  },  "required": [  "userID",  "startRecord",  "ttl",  "recordParam"  ]  }  },  "required": [  "action"  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction": "UUID1",  "version":"1.0",  "data":  {  "interactionID": ["UUID2", "UUID3"],  "workitemID": ["UUID4", "UUID5"],  "userID":"userID",  "startRecord": "timePointer (time.Now().UTC())",  "ttl": 30,  "batchRecDuration": 15  "recordParam":  {  "fps": 5,  "h264Preset": "veryfast",  "H264CRF": 30,  "H264GOP": 250,  "H264BF": 2,  "H264ZeroLatency": **true**,  "H264Threads": 0,  "FfmpegLoglevel": "error"  }  }  } |

##### resultStartScreenRecord

resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### restartScreenRecord

restartScreenRecord

* [R1.11 restartScreenRecord](https://confluence.sberbank.ru/display/SCPLATFORM/R1.11+restartScreenRecord)
* [R3.1 restartScreenRecord](https://confluence.sberbank.ru/display/SCPLATFORM/R3.1+restartScreenRecord)

##### resultRestartScreenRecord

resultRestartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultRestartScreenRecord* - сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *restartScreenRecord* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultRestartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultRestartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v6 R3.2 Фича Core.AgentHelper "Запись экрана оператора"

* [1. Описание фичи](#scroll-bookmark-198)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-199)
* [3. Описание запросов](#scroll-bookmark-200)
  + [web-socket сообщения](#scroll-bookmark-201)
    - [startScreenRecord](#scroll-bookmark-202)
    - [resultStartScreenRecord](#scroll-bookmark-203)
    - [restartScreenRecord](#scroll-bookmark-204)
    - [resultRestartScreenRecord](#scroll-bookmark-205)
  + [Protobuf](#scroll-bookmark-206)
    - [RecCoreEvent](#scroll-bookmark-207)

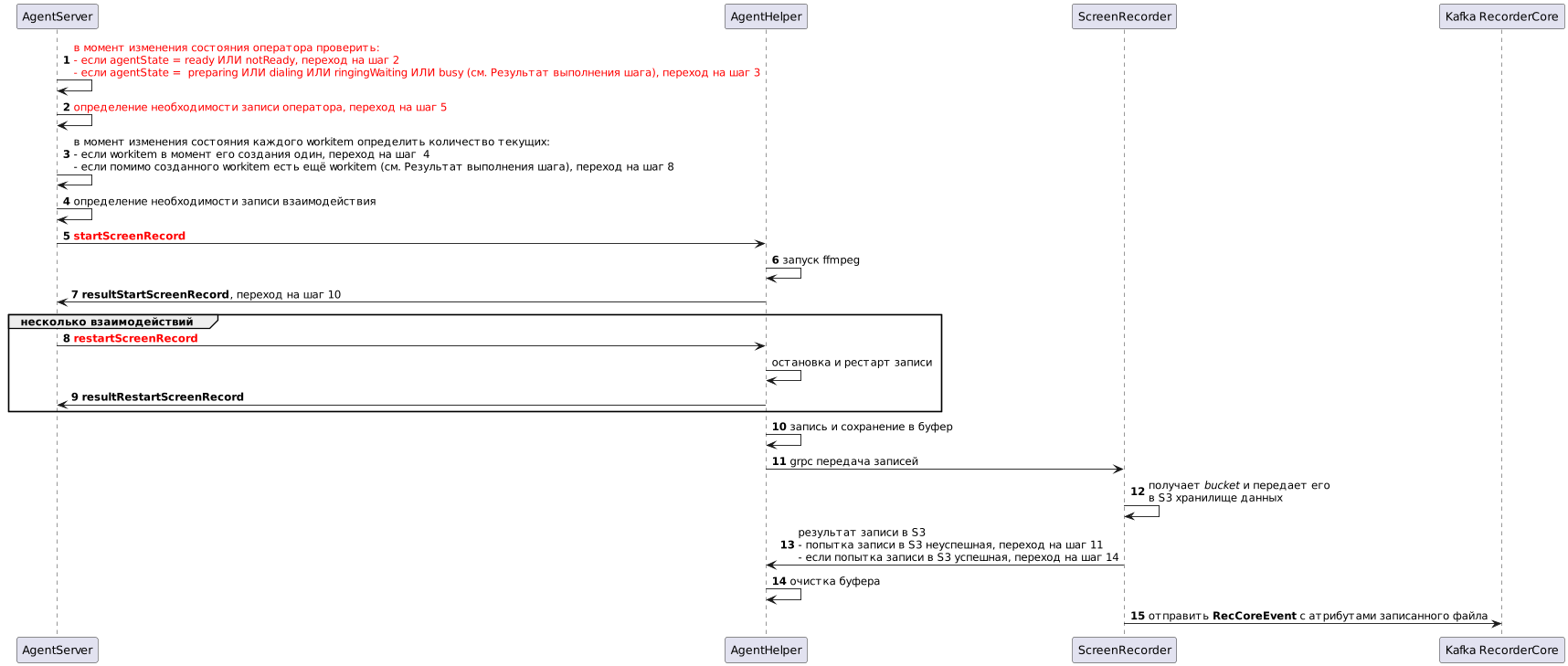
**ID фичи в JIRA**

[Epic SCPL-15272](https://jira.sberbank.ru/browse/SCPL-15272?src=confmacro) - Запись экрана на основании статуса оператора **in progress**

### 1. Описание фичи

Запись экранов оператора должна осуществляться, как согласно правилам записи его статуса, так и  во время обработки взаимодействий должна осуществляться с учетом правил записи, заданных на сервисах, по которым обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



Предварительно на AgentServer сохранена информация о настройках правил записи по сервисам. Процесс получения этих данных описан по [ссылке](https://confluence.sberbank.ru/x/_wtOmQI).

|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | * При изменении статуса оператора AgentServer получил сообщений newAgentState (перевёл оператора в состояние preparing ИЛИ dialing ИЛИ ringingWaiting ИЛИ busy), переход на **шаг 3** * при changeAgentState (перевел оператора Ready ИЛИ notReady), переход на **шаг 2** |  |
| 2 | AgentServer определяет необходимость записи экранов оператора согласно его состояния в соответствие с полученными правилами по записи оператора.   * сохранить признак *recordType = agent* | Возможные варианты:   * Если статус необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingScreenRule**:   + для ***agentStatus = Ready*** запись начинается от перехода оператора в состояние Ready и заканчивается при переходе в любое другое состояние   + для ***agentStatus = notReady*** запись начинается от перехода оператора в состояние ***notReady*** + ***agentReasonCodes*** и заканчивается при переходе в любое другое состояние   + для ***agentStatus = readyAndNotReady*** запись начинается от перехода оператора в состояние ***ready*** ИЛИ ***notReady*** + ***agentReasonCodes*** и заканчивается при переходе в любое другое состояние   В случае если  ***agentStatus = notReady ИЛИ readyAndNotReady*** в массиве **agentReasonCodes** учитывать значения для определения нужной для записи причины перерыва выбор осуществляется с учётом атрибута *kind*:   * + для *kind = system*, брать значение атрибута *code*   + для *kind = user*, брать значение атрибута *id*   Переход на **шаг 5** осуществляется при переводе оператора в состояния, которое требует записи в соответствие с полученными правилами записи оператора. |
| 3 | * При получении входящего взаимодействия AgentServer получил сообщение agentConnection (перевел оператора в Ringing, создал workitem в статусе Ringing) от модуля WorkflowEngine * при инициализации исходящего взаимодействия AgentServer начал осуществление исходящего звонка (перевел оператора в Dialing, создал workitem в статусе Ringing) * при инициализации начала подготовки к исходящему вызову, AgentServer получил сообщение startPreaptring   AgentServer определяет количество текущих workitem | Возможные варианты:   * если workitem в момент его создания один, переход на шаг 4 * если помимо созданного workitem есть ещё workitem, переход на шаг 8 |
| 4 | AgentServer определяет необходимость записи экранов оператора согласно состояния workitem в соответствие с полученными правилами по сервису.   * сохранить признак *recordType = workitem* | Возможные варианты:   * AgentServer определяет, надо ли записывать текущий звонок. Для этого анализирует параметр **recordingPercentage** по сервису текущего звонка - выбирает случайное число от 1 до 100   + если выбранное число попадает в диапазон от 1 до заданного процента, то звонок записывается   + если нет, то не записывается - на этом процесс завершается * Если звонок необходимо записывать, то начало записи определяется в соответствии со значением параметра **recordingStage**:   + для recordingStage =  ringingProgress запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние Ringing+InProgress)   + для recordingStage =  **preparing**RingingProgress запись начинается от перехода workitem в состояние **Preparing** и заканчивается при переходе в состояние Wrapup или Completed (записываются состояние **Preparing** + Ringing+InProgress)   + для recordingStage =  progressWrapup запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Completed (записываются состояние InProgress+Wrapup)   + для recordingStage =  progress запись начинается от перехода workitem в состояние InProgress и заканчивается при переходе в состояние Wrapup или Completed (записывается только состояние InProgress)   + для recordingStage =  ringingProgressWrapup запись начинается от перехода workitem в состояние Ringing и заканчивается при переходе в состояние Completed (записываются состояние Ringing+InProgress+Wrapup)   + для recordingStage =  **preparing**RingingProgressWrapup запись начинается от перехода workitem в состояние **Preparing** и заканчивается при переходе в состояние Completed (записываются состояние **Preparing**+Ringing+InProgress+Wrapup)   Переход на **шаг 5** осуществляется при переводе workitem в состояние, которое требует записи в соответствие с полученными правилами по сервису. |
| 5 | AgentServer передает в сторону AgentHelper команду для начала записи экрана оператора | AgentHelper принимает команду [***startScreenRecord***](#scroll-bookmark-208) от AgentServer |
| 6 | AgentHelper запускает ffmpeg | Приложение запущено |
| 7 | AgentHelper передает в AgentServer результат исполнения запроса на начало записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultStartScreenRecord***](#scroll-bookmark-209), переход на шаг 10 |
| 8 | AgentServer передает в сторону AgentHelper команду для рестарта записи экрана оператора с учётом добавленного workitem указанного в массиве workitems | AgentHelper принимает команду [***restartScreenRecord***](#scroll-bookmark-210) от AgentServer:   * останавливает запись текущего отрезка записи экрана (процесс описан в фиче ["Остановка записи экрана оператора"](#scroll-bookmark-180), шаг 6), * делает рестарт записи с учётом изменённого количества workitem и запускает процесс записи, описанный в шаге 8. |
| 9 | AgentHelper передает в AgentServer результат исполнения запроса о рестарте записи экрана | AgentServer получает от AgentHelper результат исполнения команды [***resultRestartScreenRecord***](#scroll-bookmark-211) |
| 10 | AgentHelper от момента начала записи осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" (параметр *batchRecDuration* задаётся в Админке) минут или при разрыве/восстановлении соединения (при разрыве/восстановлении файл может содержать меньшее количество времени записи) | Запускается непрерывный процесс записи данных |
| 11 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder  по протоколу grpc |  |
| 12 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 13 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 11 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 14 |
| 14 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 8 для записи нового bucket |  |
| 15 | ScreenRecorder формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-212), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### startScreenRecord

startScreenRecord

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.AgentHelper  *startScreenRecord*- команда начать запись экранов пользователя | [1] |
| transaction | | | string |  | [0..1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | | object | Тело сообщения |  |
|  | interactions | | array | массив идентификаторов взаимодействия  (предупреждение)обязательное для *recordType = workitem* | [0..1] |
|  | workitems | | array | массив идентификаторов workitem  (предупреждение)обязательное для *recordType = workitem* | [0..1] |
|  | userID | | string | идентификатор пользователя | [1] |
|  | ttl | | integer | срок хранения записи в днях | [1] |
|  | startRecord | | string | время старта записи | [1] |
|  | batchRecDuration | | integer | продолжительность записи пакета по времени в минутах | [1] |
|  | agentStateID | | string | идентификатор записи  (предупреждение)обязательное для *recordType = agent* | [0..1] |
|  | recordParam | |  |  |  |
|  |  | fps | integer | частота кадров в секунду | [1] |
|  |  | h264Preset | string | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | [1] |
|  |  | h264CFR | integer | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | [1] |
|  |  | h264GOP | integer | Частота ключевых кадров (H.264 GOP) | [1] |
|  |  | h264GBF | integer | Максимальное количество последовательных B-кадров | [1] |
|  |  | h264ZeroLatency | boolean | Оптимизация сжатия для стриминга (H.264 ZeroLatency) | [1] |
|  |  | h264Threads | integer | Количество потоков сжатия | [1] |
|  |  | FfmpegLogLevel | string | Уровень логирования ffmpeg | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "startScreenRecord",  "description": "команда начать запись экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactions": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "workitems": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "userID": {  "type": "string"  },  "startRecord": {  "type": "string"  },  "ttl": {  "type": "integer"  },  "batchRecDuration": {  "type": "integer"  },  "agentStateID": {  "type": "string"  },        "recordParam": {  "type": "object",  "properties": {  "fps": {  "type": "integer"  },  "h264Preset": {  "type": "string"  },  "H264CRF": {  "type": "integer"  },  "H264GOP": {  "type": "integer"  },  "H264BF": {  "type": "integer"  },  "H264ZeroLatency": {  "type": "boolean"  },  "H264Threads": {  "type": "integer"  },  "FfmpegLoglevel": {  "type": "string"  }  },  "required": [  "fps",  "h264Preset",  "H264CRF",  "H264GOP",  "H264BF",  "H264ZeroLatency",  "H264Threads",  "FfmpegLoglevel"  ]  }  },  "required": [  "userID",  "startRecord",  "ttl",  "recordParam"  ]  }  },  "required": [  "action"  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"startScreenRecord",  "transaction": "UUID1",  "version":"1.0",  "data":  {  "interactionID": ["UUID2", "UUID3"],  "workitemID": ["UUID4", "UUID5"],  "userID":"userID",  "startRecord": "timePointer (time.Now().UTC())",  "ttl": 30,  "batchRecDuration": 15,  "agentStateID": "UUID6",  "recordParam":  {  "fps": 5,  "h264Preset": "veryfast",  "H264CRF": 30,  "H264GOP": 250,  "H264BF": 2,  "H264ZeroLatency": **true**,  "H264Threads": 0,  "FfmpegLoglevel": "error"  }  }  } |

##### resultStartScreenRecord

resultStartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStartScreenRecord - сообщение о результате исполнения запроса на начало записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события startSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### restartScreenRecord

restartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentHelper  *restartScreenRecord* - команда начать запись экранов пользователя с обновлённым набором workitem | [1] |
| transaction | | string |  | [0..1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения |  |
|  | interactions | array | массив идентификаторов взаимодействия  (предупреждение)обязательное для *recordType = workitem* | [0..1] |
|  | workitems | array | массив идентификаторов workitem  (предупреждение)обязательное для *recordType = workitem* | [0..1] |
|  | userID | string | идентификатор пользователя | [1] |
|  | agentStateID | string | идентификатор записи  (предупреждение)обязательное для *recordType = agent* | [0..1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "restartScreenRecord",  "description": "команда начать запись экранов пользователя с обновлённым набором workitem",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "interactionID": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "workitemID": {  "type": "array",  "items": [  {  "type": "string"  }  ]  },  "userID": {  "type": "string"  },  "agentStateID": {  "type": "string"  }      },  "required": [  "userID"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"restartScreenRecord",  "transaction": "UUID1",  "version":"1.0",  "data":  {  "interactionID": ["UUID2", "UUID3"],  "workitemID": ["UUID4", "UUID5"],  "userID": "UUID5",  "agentStateID": "UUID6"  }  } |

##### resultRestartScreenRecord

resultRestartScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  *resultRestartScreenRecord* - сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события *restartScreenRecord* | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultRestartScreenRecord",  "description": "сообщение о результате исполнения запроса на начало записи экранов пользователя с учётом добавленного workitm",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultRestartScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

# Фича Core.AgentHelper "Остановка записи экрана оператора"

* [v1 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-213)
* [v2 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-214)
* [v3 R1.11.0 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-215)
* [v4 R1.12.0 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-180)
* [v5 R2.8 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-216)
* [v6 R3.2 Фича Core.AgentHelper "Остановка записи экрана оператора"](#scroll-bookmark-217)

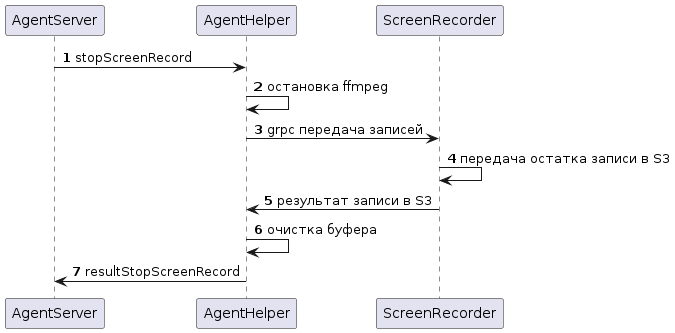
## v1 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-218)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-219)
  + [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-220)
  + [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-221)

**ID фичи в JIRA**

### 1. Описание фичи

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-222)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 2 | AgentHelper останавливает ffmpeg | Приложение запущено остановлено |
| 3 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 4 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 5 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 3 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 6 |
| 6 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 7 для записи нового bucket |  |
| 7 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-223) исполнения запроса об остановке записи экрана |  |

**3. Описание запросов**

#### stopScreenRecord (AgentServer → AgentHelper)

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

#### resultStopScreenRecord (AgentHelper → AgentServer)

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## v2 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-224)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-225)
  + [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-226)
  + [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-227)

**ID фичи в JIRA**

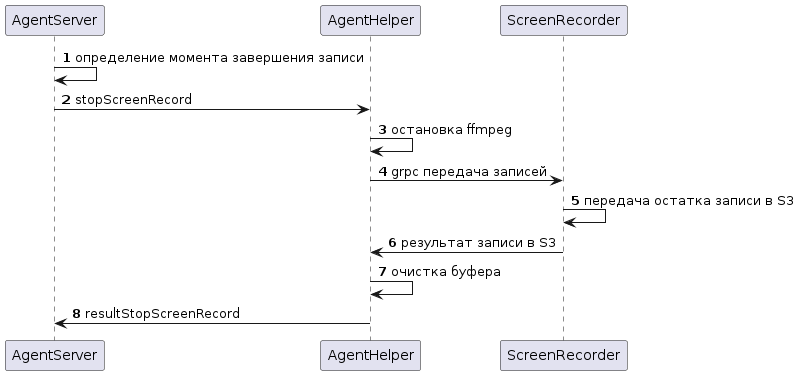
[Epic SCPL-3494](https://jira.sberbank.ru/browse/SCPL-3494?src=confmacro) - Запись экранов **done**

[Epic SCPL-5835](https://jira.sberbank.ru/browse/SCPL-5835?src=confmacro) - Правила записи экранов **done**

### 1. Описание фичи

Завершение записи экранов оператора во время обработки взаимодействия должно осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатывается взаимодействие

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer определяет момент завершения записи экранов оператора | Состояние workitem, при переходе в которое должна останавливаться запись взаимодействия определяется в соответствии с настроенными на сервисе правилами записи (значение параметра **recordingStag)**:   * для recordingStage =  ringingProgress заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  progressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  progress запись заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  ringingProgressWrapup запись заканчивается при переходе в состояние Completed   Переход на шаг 2 осуществляется при переводе workitem в состояние, которое требует остановки записи в соответствии с полученными правилами по сервису. |
| 2 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-228)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 3 | AgentHelper останавливает ffmpeg | Приложение запущено остановлено |
| 4 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 5 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 6 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 3 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на шаг 6 |
| 7 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на шаг 7 для записи нового bucket |  |
| 8 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-229) исполнения запроса об остановке записи экрана |  |

**3. Описание запросов**

#### stopScreenRecord (AgentServer → AgentHelper)

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

#### resultStopScreenRecord (AgentHelper → AgentServer)

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

## v3 R1.11.0 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-230)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-231)
* [3. Описание запросов](#scroll-bookmark-232)
  + [web-socket сообщения](#scroll-bookmark-233)
    - [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-234)
    - [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-235)
    - [newRecBatch (AgentHelper → AgentServer)](#scroll-bookmark-236)
  + [Protobuf](#scroll-bookmark-237)
    - [RecCoreEvent (AgentServer → RecordingCore)](#scroll-bookmark-238)

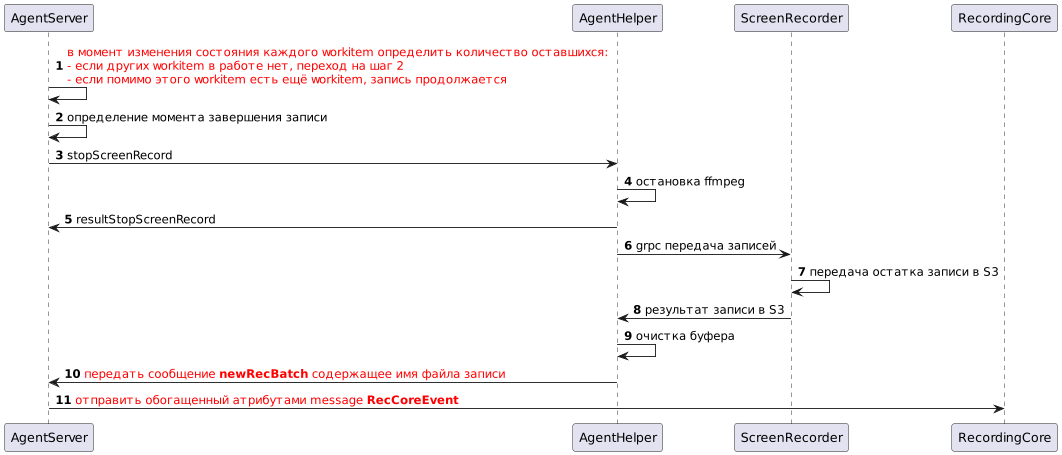
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Завершение записи экранов оператора во время обработки взаимодействий должно осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer определяет количество оставшихся workitem после изменения состояния каждого workitem | Возможные варианты:   * если в работе один текущий workitem , переход на шаг 2 * если помимо этого есть ещё workitem, запись продолжается |
| 2 | AgentServer определяет момент завершения записи экранов оператора | Состояние workitem, при переходе в которое должна останавливаться запись взаимодействия определяется в соответствии с настроенными на сервисе правилами записи (значение параметра **recordingStag)**:   * для recordingStage =  ringingProgress заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  progressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  progress запись заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  ringingProgressWrapup запись заканчивается при переходе в состояние Completed   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует остановки записи в соответствии с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-239)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 4 | AgentHelper останавливает ffmpeg | Приложение остановлено |
| 5 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-240) исполнения запроса об остановке записи экрана |  |
| 6 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 7 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 8 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 6 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit), осуществляется переход на шаг 9 |
| 9 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, запись останавливается |  |
| 10 | AgentHelper от момента начала записи, каждые 10 минут или при разрыве/восстановлении соединения, формирует и отправляет в сторону AgentServer сообщение [***newRecBatch***](#scroll-bookmark-241) с именем файла и результатом его записи, в котором содержится 10 минутный (при разрыве/восстановлении файл может содержать меньшее количество времени записи) отрезок записи экрана. |  |
| 11 | AgentServer обогащает полученное имя файла связанными с ним атрибутами и отправляет обогащенный protobuf [***RecCoreEvent***](#scroll-bookmark-242) в сторону RecorderCore |  |

### 3. Описание запросов

#### web-socket сообщения

##### stopScreenRecord (AgentServer → AgentHelper)

stopScreenRecord

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

##### resultStopScreenRecord (AgentHelper → AgentServer)

resultStopScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

##### newRecBatch (AgentHelper → AgentServer)

newRecBatch

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.AgentServer  newRecBatch  - сообщение содержащее имя файла записи | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат создания файла записи. Возможные значения:   * succes * error | [1] |
|  | fileName | string | Имя файла | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "newRecBatch",  "description": "сообщение содержащее имя файла записи",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string",  "enum": [  "success",  "error"  ]  },  "fileName": {  "type": "string"  }  },  "required": [  "result",  "fileName"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"newRecBatch",  "version": "1.0",  "data": {  "result": "success",  "fileName": "ABCDEFG"  }  } |

#### Protobuf

##### RecCoreEvent (AgentServer → RecordingCore)

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v4 R1.12.0 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-243)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-244)
* [3. Описание запросов](#scroll-bookmark-245)
  + [web-socket сообщения](#scroll-bookmark-246)
    - [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-247)
    - [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-248)
  + [Protobuf](#scroll-bookmark-249)
    - [RecCoreEvent](#scroll-bookmark-250)

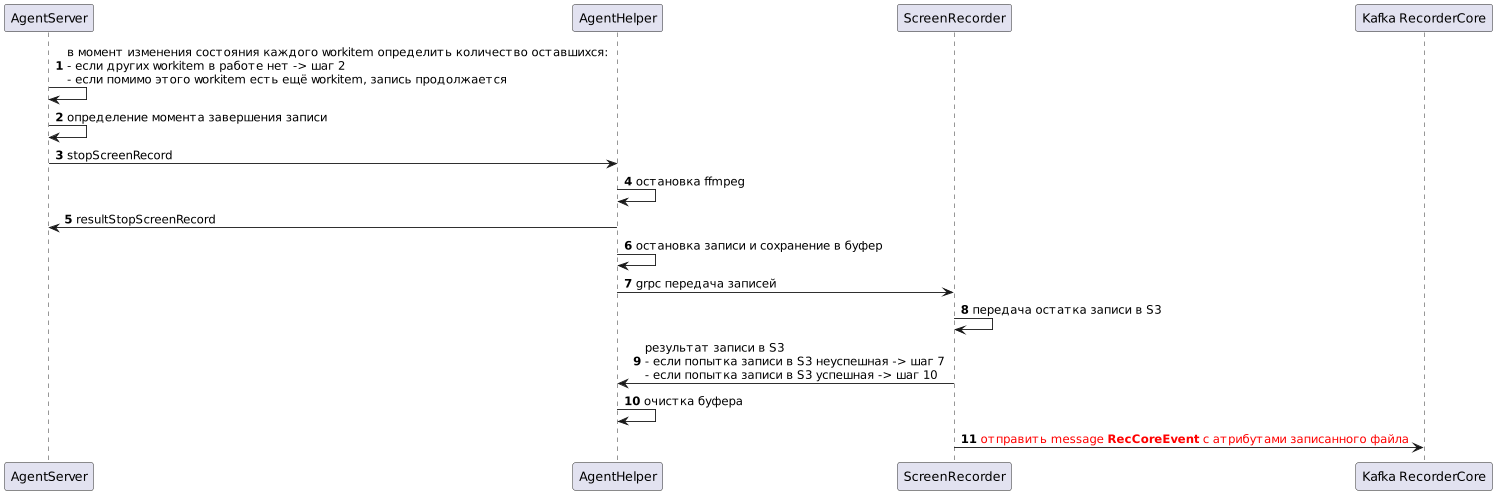
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Завершение записи экранов оператора во время обработки взаимодействий должно осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer определяет количество оставшихся workitem после изменения состояния каждого workitem | Возможные варианты:   * если в работе один текущий workitem , переход на шаг 2 * если помимо этого есть ещё workitem, запись продолжается |
| 2 | AgentServer определяет момент завершения записи экранов оператора | Состояние workitem, при переходе в которое должна останавливаться запись взаимодействия определяется в соответствии с настроенными на сервисе правилами записи (значение параметра **recordingStag)**:   * для recordingStage =  ringingProgress заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  progressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  progress запись заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  ringingProgressWrapup запись заканчивается при переходе в состояние Completed   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует остановки записи в соответствии с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-251)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 4 | AgentHelper останавливает ffmpeg | Приложение остановлено |
| 5 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-252) исполнения запроса об остановке записи экрана |  |
| 6 | AgentHelper осуществляет остановку записи текущего файла и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" минут или при остановке записи раньше заданного интервала *batchRecDuration* (параметр задаётся в Админке)файл может содержать меньшее количество времени записи |  |
| 7 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 8 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 9 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 7 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit), осуществляется переход на шаг 10 |
| 10 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, запись останавливается |  |
| 11 | ScreenServer формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-253), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### stopScreenRecord (AgentServer → AgentHelper)

stopScreenRecord

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

##### resultStopScreenRecord (AgentHelper → AgentServer)

resultStopScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v5 R2.8 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-254)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-255)
* [3. Описание запросов](#scroll-bookmark-256)
  + [web-socket сообщения](#scroll-bookmark-257)
    - [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-258)
    - [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-259)
  + [Protobuf](#scroll-bookmark-260)
    - [RecCoreEvent](#scroll-bookmark-261)

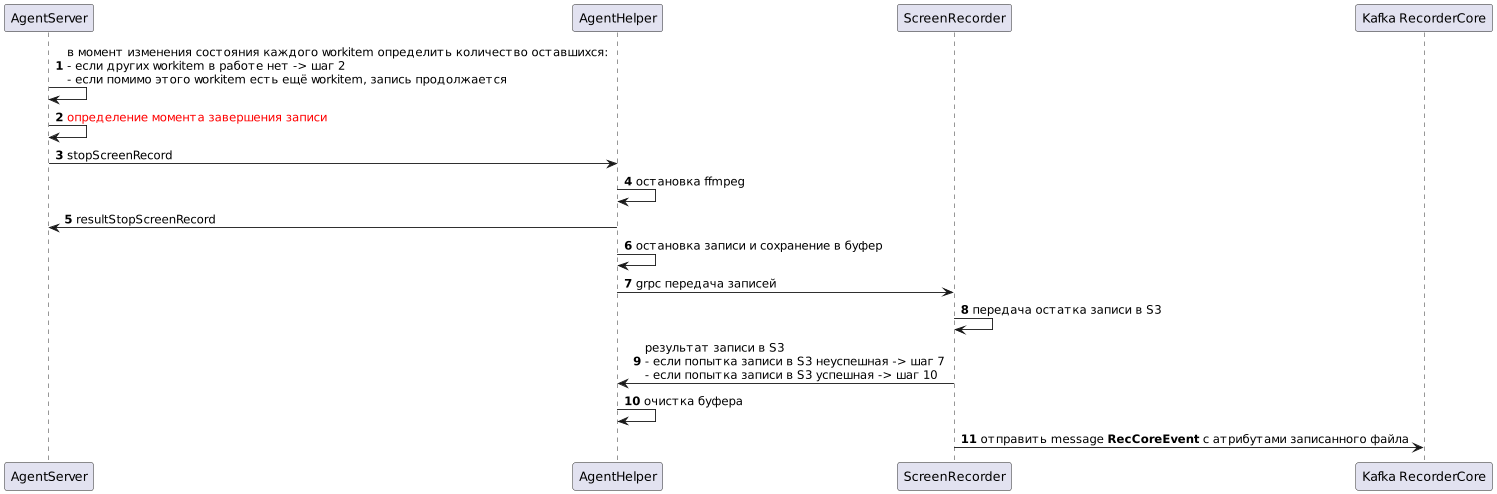
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Завершение записи экранов оператора во время обработки взаимодействий должно осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | AgentServer определяет количество оставшихся workitem после изменения состояния каждого workitem | Возможные варианты:   * если в работе один текущий workitem , переход на шаг 2 * если помимо этого есть ещё workitem, запись продолжается |
| 2 | AgentServer определяет момент завершения записи экранов оператора | Состояние workitem, при переходе в которое должна останавливаться запись взаимодействия определяется в соответствии с настроенными на сервисе правилами записи (значение параметра **recordingStag)**:   * для recordingStage =  ringingProgress заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  **preparing**RingingProgress  заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  progressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  progress запись заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  ringingProgressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  **preparing**RingingProgressWrapup запись заканчивается при переходе в состояние Completed   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует остановки записи в соответствии с полученными правилами по сервису. |
| 3 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-262)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 4 | AgentHelper останавливает ffmpeg | Приложение остановлено |
| 5 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-263) исполнения запроса об остановке записи экрана |  |
| 6 | AgentHelper осуществляет остановку записи текущего файла и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" минут или при остановке записи раньше заданного интервала *batchRecDuration* (параметр задаётся в Админке)файл может содержать меньшее количество времени записи |  |
| 7 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 8 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 9 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 7 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit), осуществляется переход на шаг 10 |
| 10 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, запись останавливается |  |
| 11 | ScreenServer формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-264), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### stopScreenRecord (AgentServer → AgentHelper)

stopScreenRecord

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

##### resultStopScreenRecord (AgentHelper → AgentServer)

resultStopScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## v6 R3.2 Фича Core.AgentHelper "Остановка записи экрана оператора"

* [1. Описание фичи](#scroll-bookmark-265)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-266)
* [3. Описание запросов](#scroll-bookmark-267)
  + [web-socket сообщения](#scroll-bookmark-268)
    - [stopScreenRecord (AgentServer → AgentHelper)](#scroll-bookmark-269)
    - [resultStopScreenRecord (AgentHelper → AgentServer)](#scroll-bookmark-270)
  + [Protobuf](#scroll-bookmark-271)
    - [RecCoreEvent](#scroll-bookmark-272)

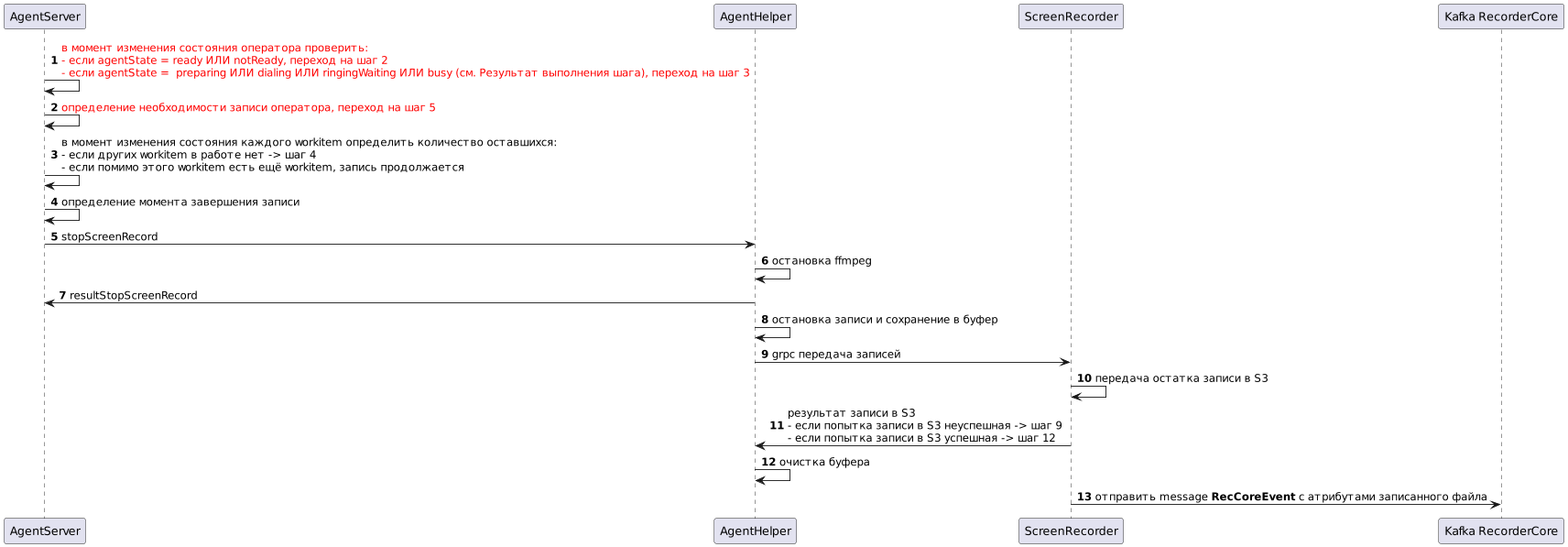
**ID фичи в JIRA**

[Epic SCPL-15272](https://jira.sberbank.ru/browse/SCPL-15272?src=confmacro) - Запись экрана на основании статуса оператора **in progress**

### 1. Описание фичи

Завершение записи экранов оператора во время обработки взаимодействий должно осуществляться с учетом правил записи, заданных на сервисе, по которому обрабатываются взаимодействия

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | * При изменении статуса оператора AgentServer получил сообщений newAgentState (перевёл оператора в состояние preparing ИЛИ dialing ИЛИ ringingWaiting ИЛИ busy), переход на **шаг 3** * при changeAgentState (перевел оператора Ready ИЛИ notReady), переход на **шаг 2**   AgentServer определяет количество текущих workitem. |  |
| 2 | AgentServer определяет необходимость остановки записи экранов оператора согласно его состояния в соответствие с полученными правилами записи оператора. | Возможные варианты:   * Если статус необходимо перестать записывать, то остановка записи определяется в соответствии со значением параметра **recordingScreenRule**:   + для ***agentStatus = Ready*** запись заканчивается, при переходе в любое другое состояние   + для ***agentStatus = notReady*** запись заканчивается, при переходе оператора в в любое другое состояние отличное от ***notReady*** + ***agentReasonCodes***   + для ***agentStatus = readyAndNotReady*** запись заканчивается, при переходе оператора в в любое другое состояние отличное от ***ready*** ИЛИ ***notReady*** + ***agentReasonCodes***   В случае если  ***agentStatus = notReady ИЛИ readyAndNotReady*** в массиве **agentReasonCodes** учитывать значения для определения нужной для записи причины перерыва выбор осуществляется с учётом атрибута *kind*:   * + для *kind = system*, брать значение атрибута *code*   + для *kind = user*, брать значение атрибута *id*   Переход на **шаг 5** осуществляется при переводе оператора в состояние, отличное от состояний полученных в правиле записи оператора |
| 3 | AgentServer определяет количество оставшихся workitem после изменения состояния каждого workitem | Возможные варианты:   * если в работе один текущий workitem , переход на **шаг 4** * если помимо этого есть ещё workitem, запись продолжается |
| 4 | AgentServer определяет момент завершения записи экранов оператора | Состояние workitem, при переходе в которое должна останавливаться запись взаимодействия определяется в соответствии с настроенными на сервисе правилами записи (значение параметра **recordingStag)**:   * для recordingStage =  ringingProgress заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  **preparing**RingingProgress  заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  progressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  progress запись заканчивается при переходе в состояние Wrapup или Completed * для recordingStage =  ringingProgressWrapup запись заканчивается при переходе в состояние Completed * для recordingStage =  **preparing**RingingProgressWrapup запись заканчивается при переходе в состояние Completed   Переход на шаг 3 осуществляется при переводе workitem в состояние, которое требует остановки записи в соответствии с полученными правилами по сервису. |
| 5 | AgentServer передает в сторону AgentHelper команду [***stopScreenRecord***](#scroll-bookmark-273)для остановки записи экрана оператора | AgentHelper принимает команду от AgentServer |
| 6 | AgentHelper останавливает ffmpeg | Приложение остановлено |
| 7 | AgentHelper передает в AgentServer результат [***resultStopScreenRecord***](#scroll-bookmark-274) исполнения запроса об остановке записи экрана |  |
| 8 | AgentHelper осуществляет остановку записи текущего файла и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" минут или при остановке записи раньше заданного интервала *batchRecDuration* (параметр задаётся в Админке)файл может содержать меньшее количество времени записи |  |
| 9 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 10 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 11 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на шаг 9 * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit), осуществляется переход на шаг 12 |
| 12 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, запись останавливается |  |
| 13 | ScreenServer формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-275), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### stopScreenRecord (AgentServer → AgentHelper)

stopScreenRecord

Параметры сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Элемент | Тип | Описание | Кратность |
| action | string | Название сообщения в сторону Core.AgentHelper  *stopScreenRecord*- команда остановки записи экранов пользователя | [1] |
| transaction | string |  | [0..1] |
| version | string | Версия API. По умолчанию 1.0 | [1] |

Json-схема сообщения

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "stopScreenRecord",  "description": "команда остановки записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "version": {  "type": "string"  }  },  "required": [  "action",  "version"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"stopScreenRecord",  "transaction":"string",  "version":"1.0"  } |

##### resultStopScreenRecord (AgentHelper → AgentServer)

resultStopScreenRecord

Параметры сообщения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Элемент | | Тип | Описание | Кратность |
| action | | string | Название сообщения в сторону Core.JsSDK  resultStopScreenRecord - сообщение о результате исполнения запроса на остановку записи экранов пользователя | [1] |
| version | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | string | Ответный transaction при обработке события stopSceenRecord | [0..1] |
| data | | object | Тело сообщения | [1] |
|  | result | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | resultDesc | string | Причина ошибки исполнения запроса  Заполняется только для result = error | [0..1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultStartScreenRecord",  "description": "сообщение о результате исполнения запроса на остановку записи экранов пользователя",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "version": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "resultDesc": {  "type": "string"  }  },  "required": [  "result"  ]  }  },  "required": [  "action",  "version",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action":"resultStopScreenRecord",  "version": "1.0",  "transaction": "string",  "data": {  "result":"error",  "resultDesc":"xxx"  }  } |

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

# Параметры записи экранов, передаваемые из AgentServer в AgentHelper для использования в ffmpeg

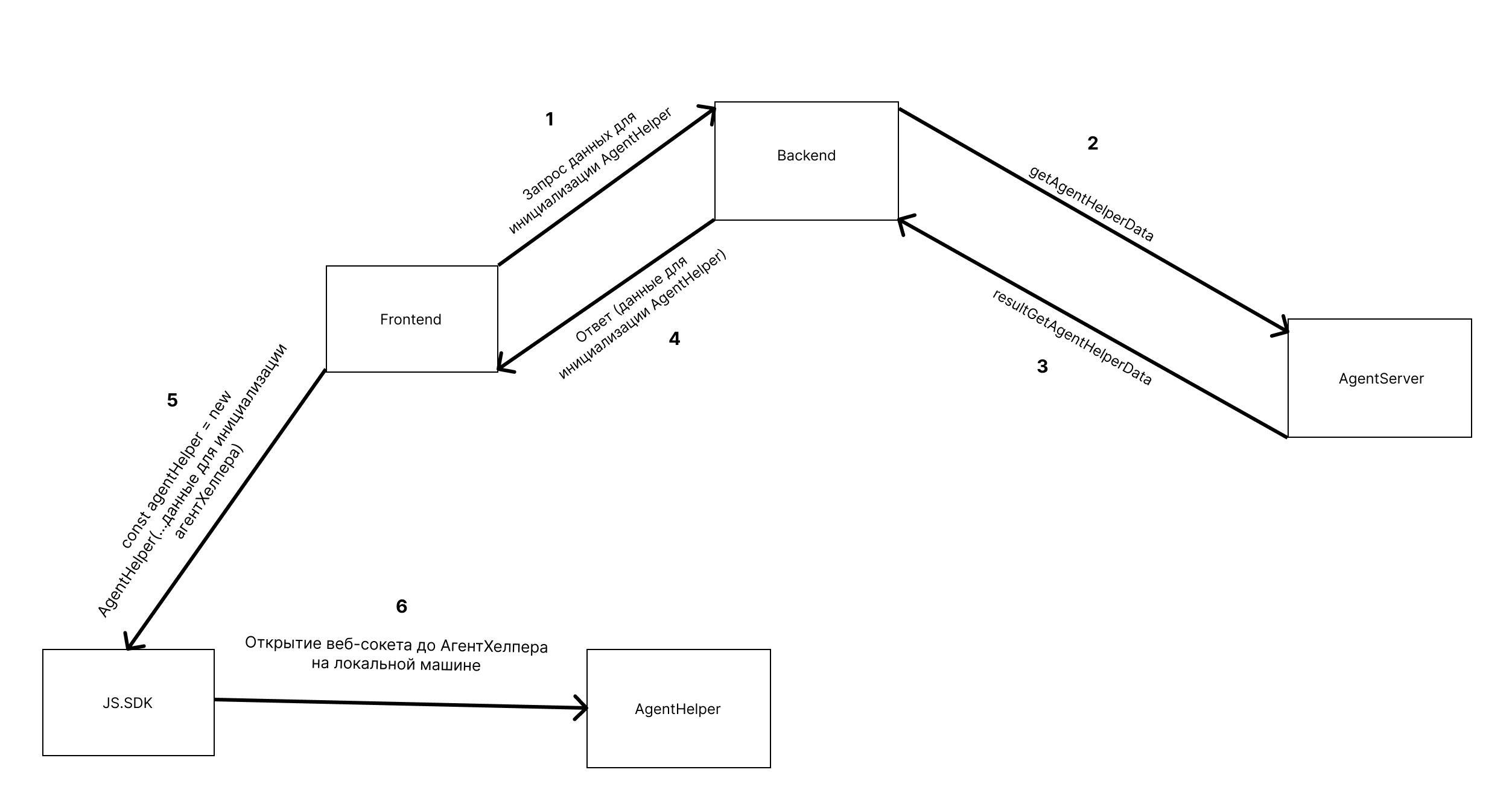
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Настраиваем в Space | Название параметра в Space | Название значений в Space | Названия полей в БД и/или API | Значения в БД/API | Значение  поумолчанию | Пример задания параметра  в опциях ffmpeg |
| Нет | Способ захвата экрана в Windows | Desktop Duplication  GDI Grab | windowsGrabMode | ddaGrab (win)  gdiGrab (win)  avfoundation (macOS) | gdiGrab | -f gdigrab |
| Да | Частота кадров | Целое число от 1-10 | screenFramerate | Целое число от 1-10 | 5 | -framerate 5 |
| Да | Баланс между скоростью сжатия и степенью сжатия (H.264 Preset) | ultrafast  superfast  veryfast  faster  fast  medium  slow | h264Preset | ultrafast  superfast  veryfast  faster  fast  medium | veryfast | -preset veryfast |
| Да | Баланс между качеством и степенью сжатия (H.264 Constant Rate Factor) | Целое число от 0-63 | h264CRF | Целое число от 0-63 | 30 | -crf 23 |
| Да | Частота ключевых кадров (H.264 GOP) | Целое число от 1-500 | h264GOP | Целое число от 1-500 | 250 | -g 10 |
| Да | Максимальное количество последовательных B-кадров | Авто  0  1  2 | h264BF | Целое число от -1,0,1,2  Авто - если -1 | 2 | -bf 2 |
| Да | Оптимизация  сжатия для стриминга (H.264 ZeroLatency) | Да  Нет | h264ZeroLatency | yes/no | No | -tune zerolatency |
| Да | Кол-во потоков сжатия | Целое число от 0-24 | h264Threads | Целое число от 0-24 | 0 | -threads |
|  | Уровень логирования ffmpeg | quiet  panic  fatal  error  warning  info  verbose  debug | ffmpegLoglevel | quiet  panic  fatal  error  warning  info  verbose  debug | error | -loglevel panic |

Примеры ffmpeg для windows

ffmpeg -loglevel panic -f gdigrab -framerate 5 -i desktop  -c:v libx264 -preset veryfast -g 250 -crf 30 -bf 2 -threads 0 -pix\_fmt yuv420p -vf "pad=ceil(iw/2)\*2:ceil(ih/2)\*2"

ffmpeg -loglevel panic -f gdigrab -framerate 5 -i desktop  -c:v libx264 -preset veryfast -g 250 -crf 30 -bf 2 -threads 0 -pix\_fmt yuv420p -tune zerolatency -vf "pad=ceil(iw/2)\*2:ceil(ih/2)\*2"

# Взаимодействие компонентов для записи экранов при back-to-back подключении



# Архив Core.AgentHelper

## архив-resultGetDisplayInfo

Параметры сообщения

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Элемент | | | Тип | Описание | Кратность |
| action | | | string | Название сообщения в сторону Core.JsSDK  resultGetDisplayInfo - сообщение с информацией об экранах мониторов | [1] |
| version | | | string | Версия API. По умолчанию 1.0 | [1] |
| transaction | | | string | Ответный transaction при обработке события getDisplayInfo | [0..1] |
| data | | | object | Тело сообщения | [1] |
|  | result | | string | Результат выполнения операции изменения  Возможные значения:   * success * error | [1] |
|  | displayData | | Array |  | [1] |
|  |  | monitorID | int | Идентификатор монитора | [1] |
|  |  | width | int | Ширина разрешения экрана | [1] |
|  |  | height | int | Высота разрешения экрана | [1] |

JSON-схема

|  |
| --- |
| {  "$schema": "http://json-schema.org/draft-04/schema#",  "title": "resultGetDisplayInfo",  "description": "сообщение с информацией об экранах мониторов",  "type": "object",  "properties": {  "action": {  "type": "string"  },  "transaction": {  "type": "string"  },  "data": {  "type": "object",  "properties": {  "result": {  "type": "string"  },  "displayData": {  "type": "array",  "items": [  {  "type": "object",  "properties": {  "monitorID": {  "type": "integer"  },  "width": {  "type": "integer"  },  "height": {  "type": "integer"  }  },  "required": [  "monitorID",  "width",  "height"  ]  }  ]  }  },  "required": [  "result",  "displayData"  ]  }  },  "required": [  "action",  "transaction",  "data"  ]  } |

Пример сообщения

|  |
| --- |
| {  "action": "resultGetDisplayInfo",  "transaction": "UUID"  "data": {  "result": "success",  "displayData": [  {  "monitorID": 115,  "width": 800,  "height": 600  }  ]  }  } |

# Фича Core.AgentHelper "Запись вне платформы"

* [R2.12 Фича "Запись экрана оператора вне платформы"](#scroll-bookmark-277)
* [R2.12 Фича "Остановка записи экрана оператора вне платформы"](#scroll-bookmark-278)
* [R2.12 Фича "Подключение агента записи экрана оператора для записи вне платформы"](#scroll-bookmark-279)

## R2.12 Фича "Запись экрана оператора вне платформы"

* [1. Описание фичи](#scroll-bookmark-280)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-281)
* [3. Описание запросов](#scroll-bookmark-282)
  + [web-socket сообщения](#scroll-bookmark-283)
    - [Protobuf](#scroll-bookmark-284)
    - [RecCoreEvent](#scroll-bookmark-285)

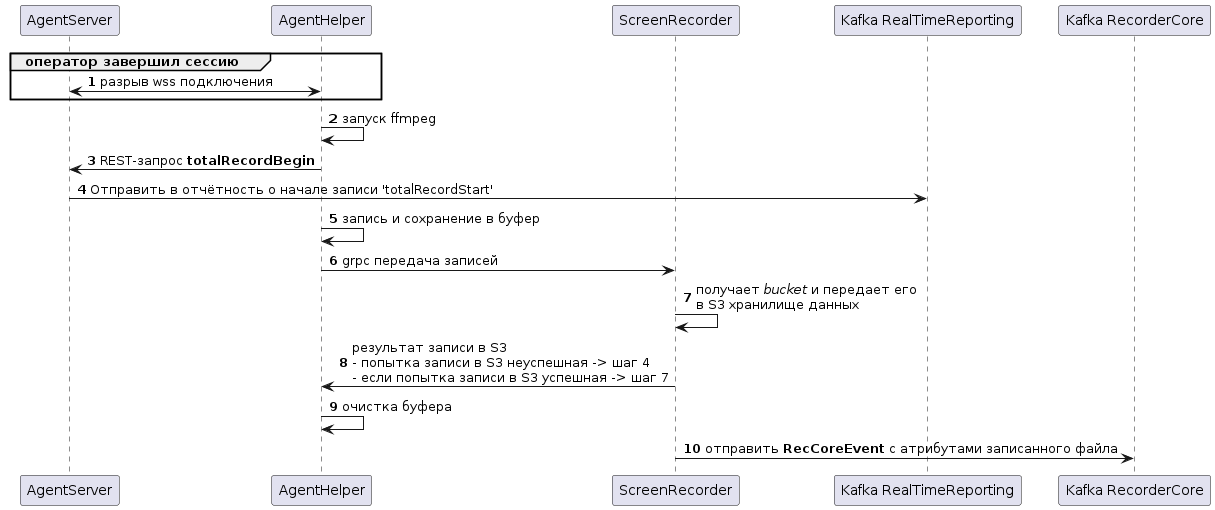
**ID фичи в JIRA**

[Epic SCPL-8923](https://jira.sberbank.ru/browse/SCPL-8923?src=confmacro) - Запись экранов оператора с учетом нескольких взаимодействий **done**

### 1. Описание фичи

Запись экранов оператора во время нахождения оператора вне платформы (от момента залогина оператора в операционной системе до залогина его в платформу SCPL)

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | Предварительно был осуществлён процесс: [R2.11 Фича "Завершение операторской сессии"](https://confluence.sberbank.ru/pages/viewpage.action?pageId=15966044568) или [R2.9 Фича Core.AgentServer "Аварийное завершение операторской](https://confluence.sberbank.ru/pages/viewpage.action?pageId=15150584025). | wss подключение разорвано И оператор в платформе SCPL не найден, переход на шаг 2 |
| 2 | AgentHelper стартует ffmpeg по причинам:   * 1. указанной на ш 1   2. по техническим причинам в процессе работы самого ffmpeg | Приложение запущено |
| 3 | AgentHelper отправляет в сторону AgentServer REST-запрос ***totalRecordBegin*** о начале записи |  |
| 4 | AgentServer отправляет в сторону RealTimeReporting событие '[*totalRecordStart*](https://confluence.sberbank.ru/x/QxxnuwM)*'* о начале записи |  |
| 5 | AgentHelper от момента начала записи осуществляет запись и сохранение логических сущностей для организации хранения объектов (bucket), каждые "n" (по умолчанию 15 мин, задаётся в config-файле) минут или при разрыве/восстановлении соединения (при разрыве/восстановлении файл может содержать меньшее количество времени записи) | Запускается непрерывный процесс записи данных |
| 6 | AgentHelper осуществляет потоковую передачу bucket в ScreenRecorder по протоколу grpc |  |
| 7 | ScreenRecorder на своей стороны получает данные в виде bucket и передает его в S3 хранилище данных |  |
| 8 | ScreenRecorder передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на **шаг 6** * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit) для передачи следующего bucket||chunk, осуществляется переход на **шаг 10** |
| 9 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, осуществляется переход на **шаг 5** для записи нового bucket |  |
| 10 | ScreenRecorder формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-286), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### web-socket сообщения

##### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## R2.12 Фича "Остановка записи экрана оператора вне платформы"

* [1. Описание фичи](#scroll-bookmark-287)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-288)
* [3. Описание запросов](#scroll-bookmark-289)
  + [Protobuf](#scroll-bookmark-290)
    - [RecCoreEvent](#scroll-bookmark-291)

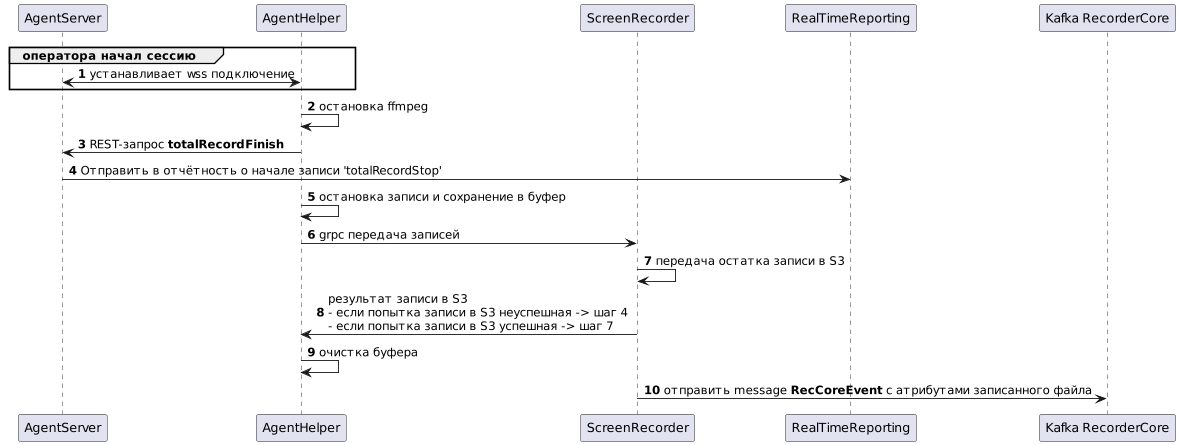
**ID фичи в JIRA**

[Epic SCPL-15272](https://jira.sberbank.ru/browse/SCPL-15272?src=confmacro) - Запись экрана на основании статуса оператора **in progress**

### 1. Описание фичи

Завершение записи экранов оператора во время нахождения оператора вне платформы (от момента залогина оператора в операционной системе до залогина его в платформу SCPL)

### 2. Описание процесса/сервиса TO BE



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | Предварительно был успешно осуществлён процесс подключения AgenHelper к РМ оператора. Процесс описан в [Фича Core.AgentHelper "Подключение агента записи экрана оператора при старте операторской сессии"](#scroll-bookmark-28), ш 15 | Уcтановление wss-соединений |
| 2 | AgentHelper останавливает ffmpeg | Приложение остановлено |
| 3 | REST-запрос ***totalRecordFinish***об окончании записи |  |
| 4 | AgentHelper отправляет в сторону RealTimeReporting событие '[*totalRecordStop*](https://confluence.sberbank.ru/x/QxxnuwM)*'* о начале записи |  |
| 5 | AgentHelper осуществляет остановку записи текущего файла и сохранение логических сущностей для организации хранения объектов (bucket) с заданным fps, каждые "n" минут или при остановке записи раньше заданного интервала *batchRecDuration* (параметр задаётся в Админке)файл может содержать меньшее количество времени записи |  |
| 6 | AgentHelper доотправляет bucket||chunk и осуществляет завершение их передачи в ScreenRecorder  по протоколу grpc |  |
| 7 | ScreenRecorder на своей стороны дополучает данные в виде bucket||chunk и передает его в S3 хранилище данных |  |
| 8 | ScreenServer передаёт в AgentHelper результат записи в S3 | * если попытка записи в S3 неуспешная, ScreenRecorder передаёт в сторону AgentHelper результат Rollback для повторной передачи не записанного bucket (предусмотреть кол-во повторных попыток записи), осуществляется переход на **шаг 6** * если попытка записи в S3 успешная, ScreenRecorder передаёт в сторону AgentHelper результат Commit (при одновременной записи нескольких backet, нужно предусмотреть запись их в один chunk и передачу группового commit), осуществляется переход на **шаг 10** |
| 9 | В AgentHelper удаляет bucket, по которому пришел успешный результат записи в S3, запись останавливается |  |
| 10 | ScreenServer формирует и отправляет в сторону Kafka RecorderCore сообщение [***RecCoreEvent***](#scroll-bookmark-292), в котором содержится отрезок записи экрана. |  |

### 3. Описание запросов

#### Protobuf

##### RecCoreEvent

RecCoreEvent

|  |
| --- |
| syntax = "proto3";    **import** "google/protobuf/timestamp.proto";    option go\_package = "proto/events";    message RecCoreEvent {  google.protobuf.Timestamp timestamp = 1;  string user\_id = 2;  repeated string interactions\_id = 3;  repeated string workitems\_id = 4;  string message\_id = 5;  google.protobuf.Timestamp start\_date\_time = 6;  google.protobuf.Timestamp finish\_date\_time = 7;  string file\_name = 8;  int32 ttl = 9; // время жизни файла в S3 (в днях)  int64 pause\_duration = 10;  string rec\_rule\_id = 11;  google.protobuf.Timestamp msg\_time = 12; // текущее время отправки сообщения  string tenant\_id = 13;  } |

## R2.12 Фича "Подключение агента записи экрана оператора для записи вне платформы"

* [ID фичи в JIRA](#scroll-bookmark-293)
* [1. Описание фичи](#scroll-bookmark-294)
* [2. Описание процесса/сервиса TO BE](#scroll-bookmark-295)
* [3. Примеры запросов](#scroll-bookmark-296)
  + [REST-запросы](#scroll-bookmark-297)
    - [findByExternalID](#scroll-bookmark-298)
* [Бизнес-требования](#scroll-bookmark-299)
  + [1. Описание метода получения пользователя](#scroll-bookmark-300)
    - [totalRecord](#scroll-bookmark-301)
    - [totalRecordCfg](#scroll-bookmark-302)

### ID фичи в JIRA

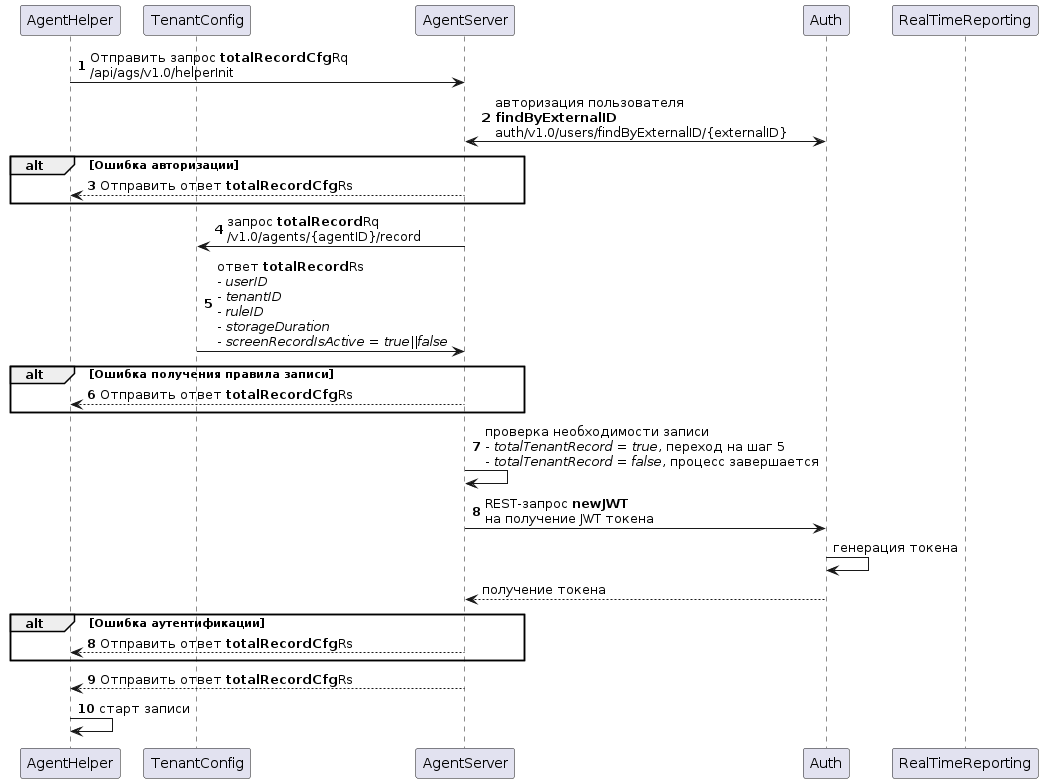
[Epic SCPL-17026](https://jira.sberbank.ru/browse/SCPL-17026?src=confmacro) - Запись экрана оператора вне платформы **done**

### 1. Описание фичи

При залогине оператора в операционной системе необходимо инициировать процесс подключения агента записи экрана оператора для осуществления его записи вне платформы

### 2. Описание процесса/сервиса TO BE

Предварительно на АРМ/ВАРМ была установлена программа AgentHelper по средствам получения дистрибутива (процесс описан в [Фича Core.ScreenRecorder "Получение дистрибутива AgentHelper"](https://confluence.sberbank.ru/pages/viewpage.action?pageId=11344555070)) или установки через SberUserSoft



|  |  |  |
| --- | --- | --- |
| № | Описание шага | Результат выполнения шага |
| 1 | В момент логина оператора в операционную систему АРМ/ВАРМ, launcher стартует под системой как служба и начинает отслеживать новую активную сессию (на Windows раз в "n" секунд (по умолчанию 5 сек); на Linux событийно). AgentHelper отправляет REST-запрос [***totalRecordCfg***](#scroll-bookmark-303)***Rq*** с сторону AgentServer на подключение | AgentServer получает запрос и обрабатывает его |
| 2 | AgentServer обращается в Auth-сервис для авторизации пользователя и получения *agentID* в платформе по средствам запроса [***findByExternalID***](#scroll-bookmark-304) | Auth-сервис получает его, обрабатывает и отправляет ответ   * если ошибка авторизации, переход на **шаг 3** * если авторизация успешная, переход на **шаг 4** |
| 3 | **Ошибка авторизации:** В случае ошибки авторизации AgentServer отправляет ответ [***totalRecordCfg***](#scroll-bookmark-303)***Rs*** |  |
| 4 | AgentServer отправляет в TC запрос [***totalRecord***](#scroll-bookmark-305)***Rq*** получения правила записи его анализа и принятия решения о начале записи | TC получает запрос обрабатывает его и формирует ответ |
| 5 | TC отправляет ответ [***totalRecord***](#scroll-bookmark-279)***Rs*** в сторону AgentServer.  AgentServer получает правила записи анализирует его принимает решения о начале записи   * *userID* * *tenantID* * *divisionID* * *storageDuration* * *screenRecordIsActive = true||false* | Возможные варианты:   * если ошибка получения правила записи, переход на **шаг 5** * если в полученном правиле записи:   + атрибут *screenRecordIsActive* отсутствует, переход на **шаг 6**   + значение *screenRecordIsActive* отсутствует, переход на **шаг 6**   + значение *screenRecordIsActive* = *false*, **процесс завершается** * если получение правила записи успешное:   + значение *screenRecordIsActive* = *true*, переход на **шаг 7** |
| 6 | **Ошибка получения правила записи:** В случае ошибки получения получения правила записи оператора AgentServer отправляет ответ [***totalRecordCfg***](#scroll-bookmark-303)***Rs*** |  |
| 7 | AgentServer проверка необходимость глобальной записи на тенанте. Получение информации о тенанте описано в[Фича "Загрузка данных тенантов из модуля Администрирования платформы"](https://confluence.sberbank.ru/pages/viewpage.action?pageId=9409076370) | Возможные значения:   * *totalTenantRecord = true*, переход на **шаг 7** * *totalTenantRecord = false*, **процесс завершается** |
| 8 | AgentServer обращается в Auth сервис REST-запросом ***newJWT*** за генерацией и получением JWT-токена для подключения AgentHelper | Auth сервис генерирует JWT-токен и передает его AgentServer'у   * аутентификация не пройдена, переход на **шаг** **8** * аутентификация не пройдена, переход на **шаг** **9** |
| 9 | **Ошибка аутентификации:** В случае если аутентификация не пройдена AgentServer отправляет ответ [***totalRecordCfg***](#scroll-bookmark-303)***Rs*** |  |
| 10 | В случае если аутентификация пройдена AgentServer отправляет ответ [***totalRecordCfg***](#scroll-bookmark-303)***Rs*** с информацией об успешном подключении AgentHelper и возможности начала записи. |  |
| 12 | AgentHelper начинает запись. Процесс описан в [Фича Core.AgentHelper "Запись экрана оператора"](#scroll-bookmark-277) с шага 2 |  |

### 3. Примеры запросов

#### REST-запросы

##### findByExternalID

findByExternalID

|  |  |
| --- | --- |
| Задача в JIRA | Описание изменений |
| [Story SCPL-15430](https://jira.sberbank.ru/browse/SCPL-15430?src=confmacro) - [Space] доработки Auth при блокировке пользователей **done** | Доработана логика передачи пользователя (добавлена проверка на статус - userState) |

### Бизнес-требования

**User Story**

Я, как модуль, хочу иметь возможность получать информацию по пользователю

|  |  |  |  |
| --- | --- | --- | --- |
| Задача | Номер задачи | Сервис | Дорабатываемые компоненты (бэк, фронт, БД) |
|  |  |  | * **Бэк:** API авторизации, API cоздания прав доступа * **БД:** таблица с перечнем прав доступа в разрезе actions и entities |
| Описание истории  (Кто? Что? Зачем?) | Я, как модуль, хочу иметь возможность получать информацию по пользователю | | |
| **Аналитик** | [Жуков Владимир Валерьевич](https://confluence.sberbank.ru/display/~22260901) | | |

#### 1. Описание метода получения пользователя

Описание API "Получение пользователя"

|  |  |
| --- | --- |
| Наименование | Получение пользователя |
| Системное наименование | findByExternalID |
| URL | auth/v1.0/users/findByExternalID/{externalID} |
| Формат | REST |
| Режим | Синхронный |
| Метод | GET |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | string | Уникальный идентификатор запроса |

**Параметры запроса:**

-

|  |
| --- |
| 1. Логика по SberPDI  Для некоторых инсталляций предусмотрена логика поиска пользователей при аутентификации не по external\_id, а по sberpdi. Для конфигурации этой настройки используются два атрибута:   * enableSberPDI - выполняет проверку по sberpdi * checkFirstBySberPDI - определяет очередность проверки (если checkFirstBySberPDI = false, сначала выполняется поиск по external\_id, если пользователь не найден, то происходит поиск по sberpdi. Если checkFirstBySberPDI = true, то сначала идет поиск пользователя по sberpdi, а потом по external\_id   2. Возвращать информацию только по пользователю с userState = 0 (активный пользователь).  Если пользователь неактивен (userState = 1), в ответе вернуть 404 |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | string | Уникальный идентификатор ответа |

**Параметры ответа:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Атрибут | | Тип | Кратность | Описание | Комментарий |
| status | | boolean | 1 | Статус |  |
| **data** | | **object** | **1** | **Ответ на запрос** |  |
|  | userID | string | 1 | Идентификатор пользователя  Получение из таблицы **USERS.id** |  |

**Коды ответов:**

|  |  |  |
| --- | --- | --- |
| Код ответа | Код состояния (code) | Описание кода ошибки (text) |
| OK | 200 | - (запрос успешно обработан) |
| Unauthorized | 401 | Не авторизован запрос |
| Not found | 404 | Не найдена запись/пользователь неактивен |
| Error | 500 | Неизвестная ошибка |
| Bad Request | 400 | Некорректный запрос |
| Forbidden | 403 | Ошибка доступа (отсутствуют необходимые права доступа) |
| Service Unavailable | 503 | Сервер не готов обработать данный запрос |

##### totalRecord

totalRecord

|  |  |  |  |
| --- | --- | --- | --- |
| № версии | Задача в JIRA | Описание изменений | Реализовано |
| v.1.1.1 | [Story SCPL-17030](https://jira.sberbank.ru/browse/SCPL-17030?src=confmacro) - [Space] 2.11. Конфигурирование записи экрана оператора вне платформы **done** | Конфигурирование записи экрана оператора вне платформы | **not ready** |

Описание API "Запрос на получение записи оператора"

|  |  |
| --- | --- |
| Наименование | Получение правила записи тенанта для оператора |
| URL | private/v1.0/agents/{agentID}/totalRecord |
| Формат | REST |
| Режим | Синхронный |
| Метод | GET |

**Заголовок запроса:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры запроса:**

|  |  |  |  |
| --- | --- | --- | --- |
| Сиснейм параметра | Описание | Тип атрибута | Кратность |
| agentID | Идентификатор пользователя | string | 1 |

**Заголовок ответа:**

|  |  |  |
| --- | --- | --- |
| Наименование | Тип | Описание |
| x-request-id | Строка 32 | Уникальный идентификатор запроса |

**Параметры ответа:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Атрибут | | Тип | | Кратность | | Описание | | Комментарий | |
| status | | string | | 1 | | Результат выполнения запроса:   * success - http-код = 200 * fail - http-код = 4\*\* * error - http-код = 5\*\* | |  | |
| errorText | | string | | 0...1 | | Описание кода ошибки | |  | |
| **data** | | **object** | |  | |  | |  | |
|  | userID | string | | 1 | | Идентификатор пользователя тенанта  **TENANT\_USERS.user\_uuid** | |  | |
|  | tenantID | string | | 1 | | Идентификатор тенанта  **TENANT\_USERS.tenant\_id** (где **TENANT\_USERS.current\_tenant** = true) | |  | |
|  | divisionID | string | | 1 | | Подразделение пользователя  **TENANT\_USERS.division\_id** | |  | |
|  | screenRecordIsActive | boolean | | 1 | | Запись экрана вне платформы  **TENANT\_USERS.screen\_record\_is\_active** | |  | |
|  | storageDuration | int | | 1 | | Время хранения (в днях)  **Получить значение атрибута SCREEN\_RECORD\_STORAGE\_DURATION\_DAYS из конфига tc**  **создать переменную в конфиге, значение по умолчанию для стендов = 92 дня** | |  | |

##### totalRecordCfg

totalRecordCfg

Подключение оператора к взаимодействию

Описание endpoint'a запроса на завершение операторской голосовой сессии для указанного взаимодействия

|  |  |
| --- | --- |
| Наименование | **totalRecordCfg** |
| Назначение | Команда агенту на подключение к РМ оператора для записи экрана вне-платформы |
| Формат | REST-api |
| Тип взаимодействия | синхронный |
| Метод | GET |
| URI | [https://[host]:[port]/api/ags/v1.0/totalRecordCfg](#scroll-bookmark-279)  [host]:[port] для каждого стенда уточняются перед эксплуатацией и должны быть вынесены в параметры подключения вызывающей стороны |
| Коммуникационный шаблон | Запрос-ответ |
| Потребитель | SCPL.Core.AgentHelper |

Общие правила вызова REST-сервисов описаны [по ссылке](https://confluence.sberbank.ru/x/I5qUrQE)

Состав данных ответа AgentServer

При получении запроса в ответ от AgentServer к AgentHelper отправляется пустой HTTP-ответ с кодом 200 (или с соответствующим кодом ошибки)

|  |  |
| --- | --- |
| код | описание |
| 400 | Bad Request |
| 403 | Forbidden |
| 500 | Internal Server Error |