

Интеграция с телефонией

Общие принципы работы интеграции с телефонией

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

Общие принципы работы интеграции с телефонией	4
Способы интеграции с системой обмена сообщениями в Creatio	4
Взаимодействие коннекторов с Creatio	6

Общие принципы работы интеграции с телефонией



Сложный

Creatio предоставляет возможность интеграции с рядом [автоматических телефонных станций](#) (АТС, [Private Branch Exchange](#), PBX) для управления звонками непосредственно из интерфейса системы. Функциональность телефонии представлена в интерфейсе в виде СТИ-панели ([Computer Telephony Integration](#)), а также в виде раздела [*Звонки*]. Стандартные возможности СТИ-панели:

- отображение пользователю входящего звонка с функцией поиска контакта/контрагента по номеру звонящего;
- выполнение звонка из системы в один клик;
- управление звонком в Creatio (ответить, поставить/снять с удержания, завершить, перевести);
- отображение истории звонков для удобного управления связями звонков или возможности перезвонить.

Все звонки, которые были выполнены или приняты с использованием интеграции, сохраняются в разделе [*Звонки*]. В разделе можно посмотреть временные характеристики звонка (дата начала, дата завершения, длительность разговора), а также с чем связан звонок в системе.

Для работы со звонками в приложении необходимо [настроить интеграцию с телефонией](#).

В зависимости от АТС, с которой выполняется интеграция, и особенностей предоставляемого API ([Application Program Interface](#)) используются разные архитектурные механизмы, которые описаны ниже. Также в зависимости от выбранного API может отличаться набор возможностей. Например, функция прослушивания звонков доступна не для всех АТС, а возможность использования Web-телефона доступна только при интеграции с Webitel. Вне зависимости от выбранного механизма интеграции, интерфейс СТИ-панели остается одинаковым для всех пользователей.

Способы интеграции с системой обмена сообщениями в Creatio

Способы интеграции можно разделить на два типа: `first party` и `third party` интеграции.

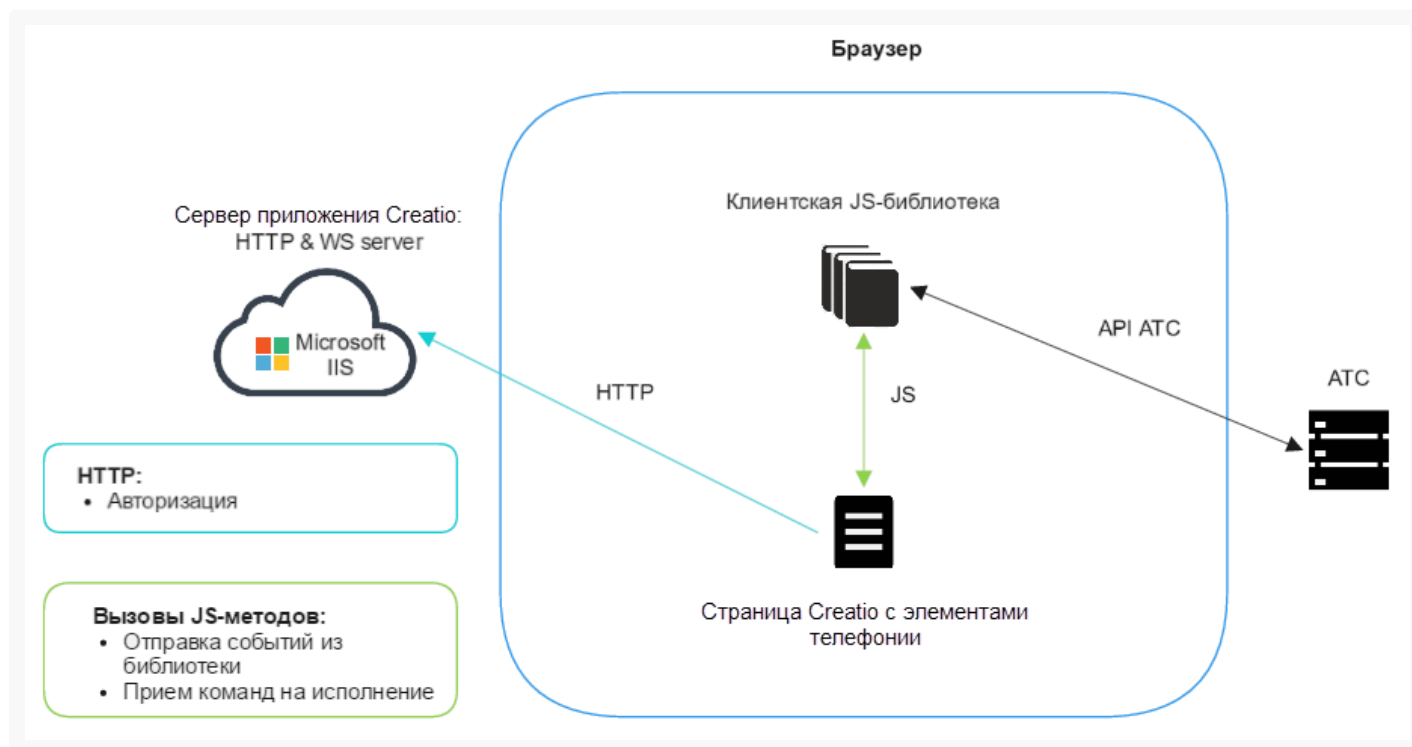
В случае `first party` интеграции для каждого пользователя создается отдельное интеграционное подключение, в рамках которого выполняется обработка событий АТС.

Для `third party` интеграций выполняется одно подключение к серверу АТС и в рамках него выполняется обработка событий АТС для всех пользователей интеграции. В случае `third party` интеграций применяется промежуточное звено `Messaging Service` для распределения информационных потоков разных пользователей.

JavaScript-адаптер на стороне клиента

При способе интеграции с JavaScript-адаптером на стороне клиента, работа с АТС происходит

непосредственно из браузера. Взаимодействие с ATC и JavaScript-библиотекой, которая, как правило, поставляется производителем ATC, происходит с помощью API ATC. Библиотека отправляет события и принимает команды на исполнение, используя JavaScript. В контексте данной интеграции страница Creatio взаимодействует с сервером приложения для авторизации, используя протокол HTTP(S).



Данный способ интеграции применим к `first party` API телефонии, например, для коннекторов Webitel, Oktell, Finesse. Для коннекторов Webitel и Oktell в качестве протокола соединения используется [WebSocket](#), а для коннектора Finesse используются [long-polling](#) http-запросы.

Преимуществом метода интеграции `first party` является тот факт, что для него не требуется наличие дополнительных узлов, например, Messaging Service. СТИ-панель с использованием интеграционной библиотеки выполняет подключения напрямую к API сервера телефонии из браузера на ПК пользователя.

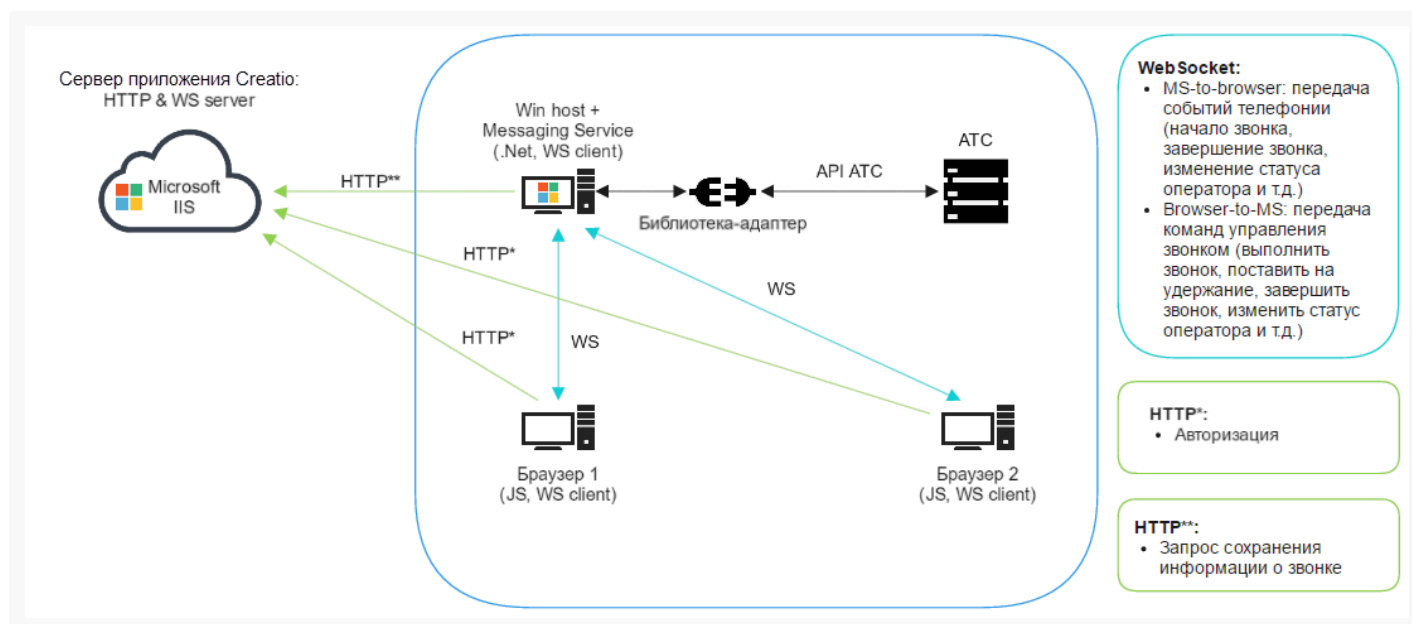
В случае с входящим звонком, сервер телефонии передает событие начала нового звонка и его параметры через WebSocket в библиотеку клиентской интеграции. Библиотека при получении команды нового звонка генерирует событие `RingStarted`, которое передается на страницу приложения.

В случае с исходящим звонком, клиентская часть генерирует команду начала звонка, которая по WebSocket передается на сервер телефонии.

Terrasoft Messaging Service на серверной стороне

При способе интеграции с Terrasoft Messaging Service (TMS) на серверной стороне все события телефонии проходят через данный сервис, который взаимодействует с ATC посредством библиотеки производителя ATC. Библиотека взаимодействует с ATC посредством API. Также TMS взаимодействует с сервером приложения Creatio для запроса сохранения информации о звонке в базе данных, используя HTTP(S). С клиентским приложением взаимодействие (передача событий и прием команд на исполнение) происходит посредством WebSocket. Как и в случае с интеграцией с JavaScript-адаптером на стороне

клиента, страница Creatio взаимодействует с сервером приложения для авторизации, используя протокол HTTP(S).



Данный способ интеграции применим к `third party` API телефонии (TAPI, TSAPI, New Infinity protocol, WebSocket Oktell). Для данного типа интеграции необходим `Messaging Service` — windows проху-служба, которая работает с библиотекой-адаптером ATC. `Messaging Service` является универсальным хостером библиотек интеграции с ATC, таких, как Asterisk, Avaya, Callway, Ctios, Infinity, Infra, Tapi. `Messaging Service` при получении клиентских подключений автоматически подключит используемую Creatio библиотеку и инициирует подключение к ATC. Фактически, `Messaging Service` является функциональной "оберткой" для тех коннекторов телефонии, которые не поддерживают клиентскую интеграцию, для того чтобы взаимодействовать с функциональностью телефонии в браузере (генерация и обработка событий, передача данных). Компьютер пользователя производит два вида коммуникации:

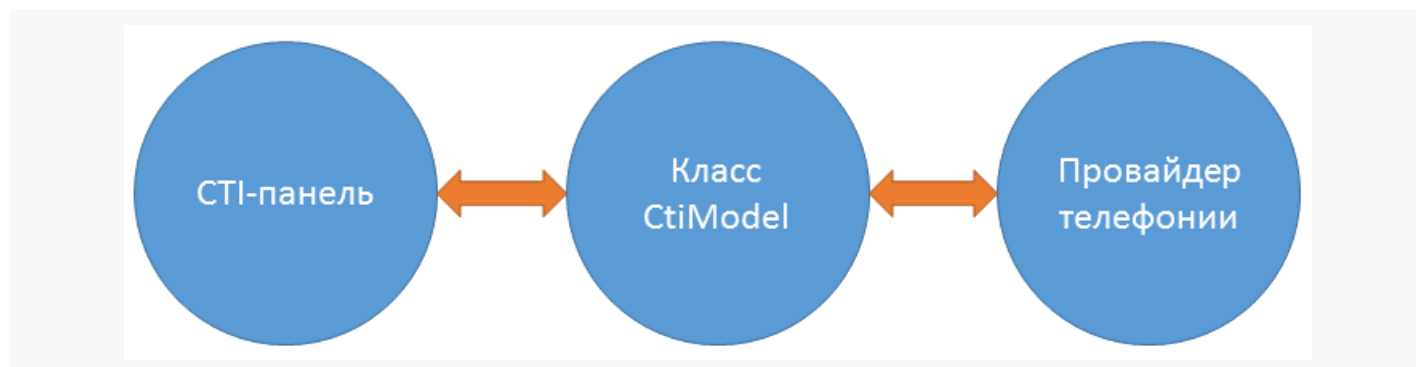
- по протоколу HTTP с сервером приложения Creatio для авторизации и с хостом, на котором установлен `Messaging Service`;
- по WebSocket для непосредственной работы с функциональностью телефонии.

В случае с входящим звонком, ATC передает событие начала нового звонка и его параметры через библиотеку-адаптер на хост с `Messaging Service`. `Messaging Service` при получении команды нового звонка генерирует событие `RingStarted`, которое передается на клиент.

В случае с исходящим звонком, клиентская часть генерирует команду начала звонка, которая по WebSocket передается в `Messaging Service`, который, в свою очередь, генерирует событие исходящего звонка для ATC.

Взаимодействие коннекторов с Creatio

Все коннекторы взаимодействуют с конфигурацией через класс `CtiModel`. Он получает события от коннектора и обрабатывает их.



Список поддерживаемых событий класса перечислен в таблице.

Список поддерживаемых событий класса CtiModel

Событие	Описание
initialized	Срабатывает при завершении инициализации провайдера.
disconnected	Срабатывает при отключении провайдера.
callStarted	Срабатывает при начале нового вызова.
callFinished	Срабатывает после завершения звонка.
commutationStarted	Срабатывает после установки соединения звонка.
callBusy	Срабатывает при переводе звонка в состояние "занят" (только TAPI).
hold	Срабатывает после постановки звонка на удержание.
unhold	Срабатывает после снятия звонка с удержания.
error	Срабатывает при возникновении ошибки.
lineStateChanged	Срабатывает после изменения набора доступных операций линии либо звонка.
agentStateChanged	Срабатывает после изменения состояния агента.
activeCallSnapshot	Срабатывает при актуализации списка активных звонков.
callSaved	Срабатывает после создания или обновления звонка в базе данных.
rawMessage	Обобщенное событие в провайдере. Срабатывает при любом событии провайдера.
currentCallChanged	Срабатывает при изменении основного звонка. Например, завершается основной звонок при консультации.

callCentreStateChanged	Срабатывает при входе или выходе оператора из режима Call-центр.
callInfoChanged	Срабатывает при изменении данных звонка по идентификатору в БД.
dtmfEntered	Срабатывает, если в линию были отправлены сигналы Dtmf.
webRtcStarted	Срабатывает при старте webRtc сессии.
webRtcVideoStarted	Срабатывает при старте видеопотока webRtc сессии.
webRtcDestroyed	Срабатывает при завершении webRtc сессии.