

Интеграция с телефонией

Интеграция с Oktell

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.

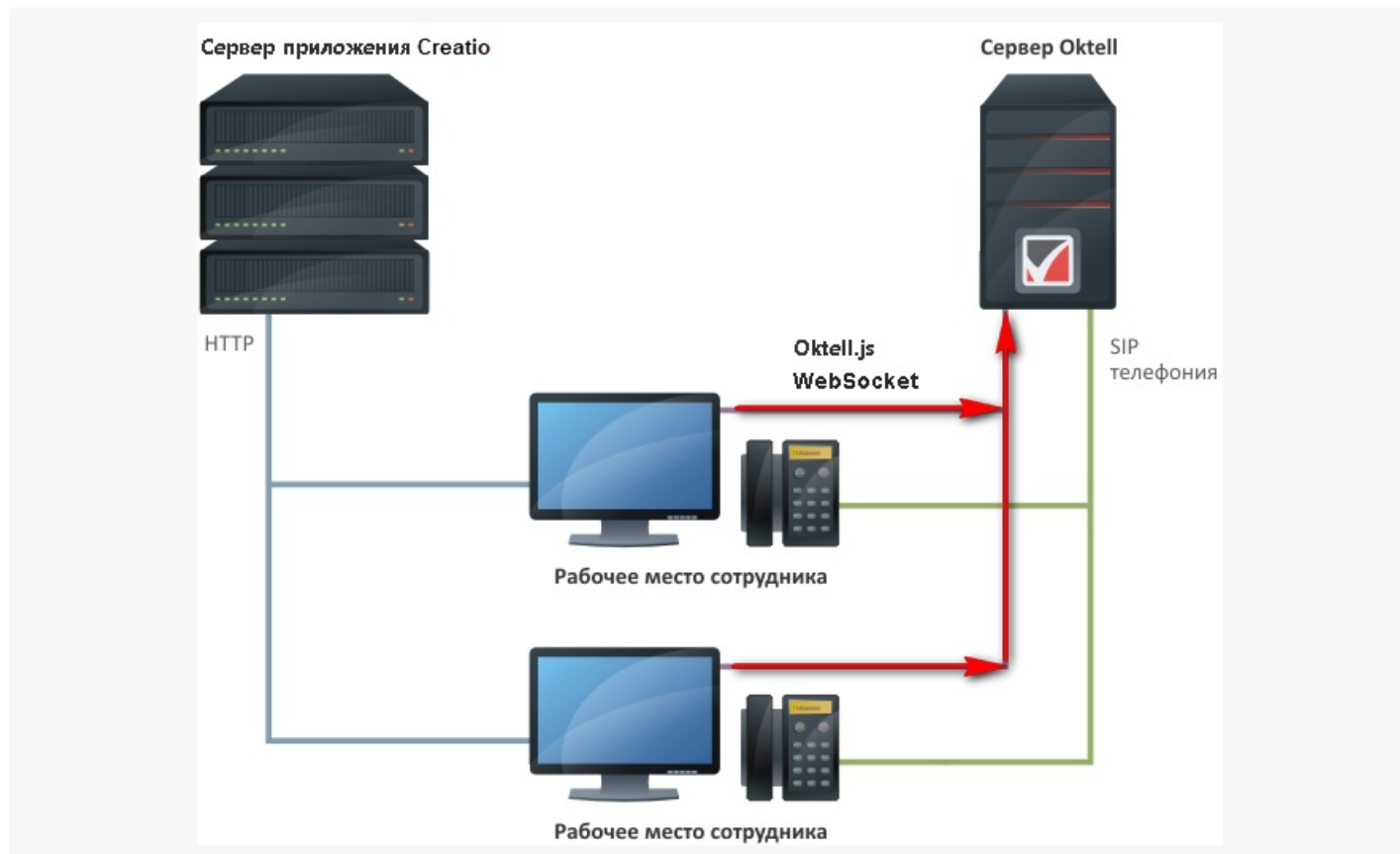
Содержание

Интеграция с Oktell	4
Oktell.js	4

Интеграция с Oktell



Интеграция Oktell с Creatio реализована на клиентском уровне с помощью библиотеки `oktell.js`. Исходный код библиотеки `oktell.js` находится в конфигурационной схеме `oktellModule` пакета `CTIBase`. Сервер Oktell взаимодействует с телефонами и с конечными клиентами (браузерами). При таком способе интеграции в Creatio не требуется наличие собственного WebSocket-сервера. Каждый клиент подключается по WebSocket-протоколу непосредственно к серверу Oktell. Сервер приложения Creatio занимается формированием страниц и предоставлением данных из базы данных приложения. Непосредственная взаимосвязь между серверами Creatio и Oktell отсутствует, доступ не требуется, клиенты самостоятельно обрабатывают и объединяют данные двух систем. По такому принципу реализованы Web-клиент Oktell и плагин `oktell.js`, доступный для встраивания в другие проекты.



Oktell.js

`oktell.js` — javascript-библиотека для встраивания функциональности управления звонками в CRM-систему. `oktell.js` использует протокол Oktell WebSocket для соединения с сервером Oktell. Преимущество этого протокола заключается в создании постоянного асинхронного соединения с сервером, которое позволяет без задержек получать события с сервера Oktell и выполнять определенные команды. Поскольку протокол Oktell WebSocket достаточно сложен для реализации,

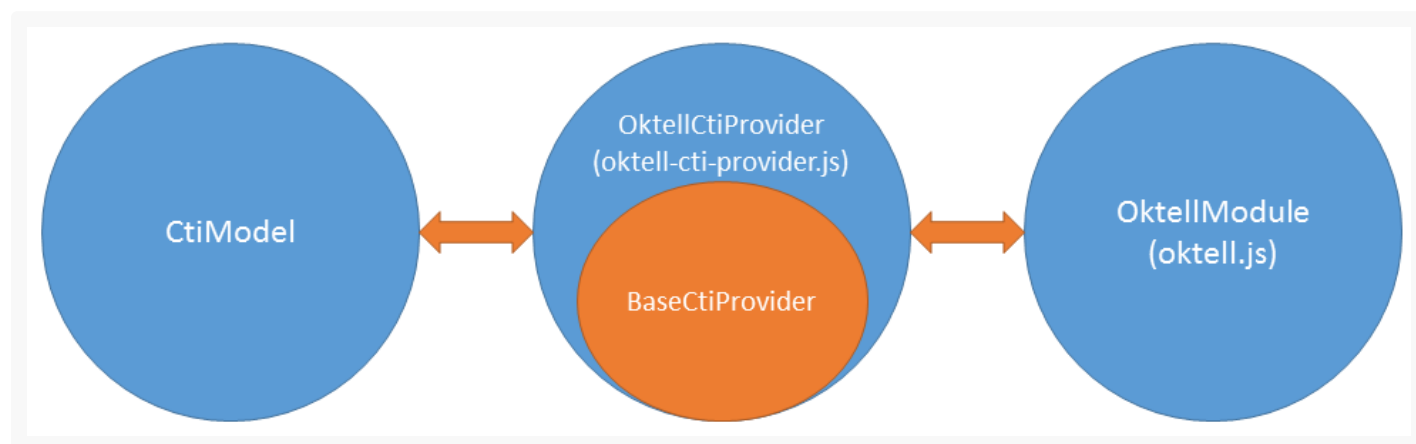
библиотека `oktell.js` оборачивает внутри себя методы WebSocket-протокола, предоставляя простую функциональность для управления.

Передача голоса между абонентами

Голос при разговоре между операторами `oktell` Creatio передается по протоколу [Session Initiation Protocol](#) (SIP). Для этого требуется использование либо [IP-телефона](#), либо установленного на компьютер оператора [софтфона](#).

Взаимодействие компонентов

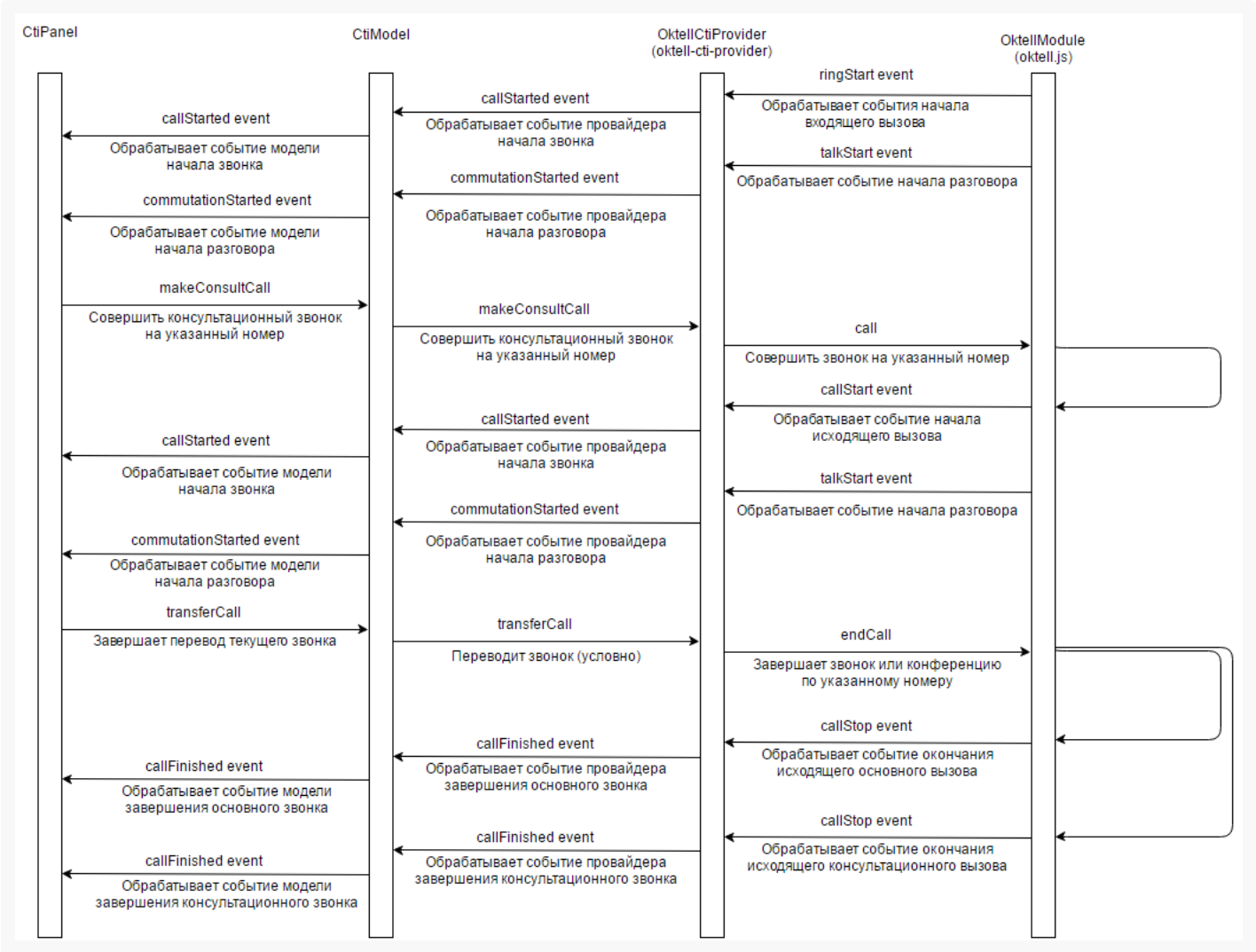
Взаимодействие с библиотекой `oktell.js` осуществляется с помощью класса `OktellCtiProvider`, который является связующим звеном между `CtiModel` и `OktellModule`, в котором находится код `oktell.js`. Класс [OktellCtiProvider](#) реализует интерфейс класса [BaseCtiProvider](#).



Примеры взаимодействия `CtiModel`, `OktellCtiProvider` и `OktellModule`:

Исходящий звонок оператора абоненту с установкой звонка на удержание абонентом, снятием с удержания абонентом и завершением звонка оператором

© 2022 Terrasoft. Все права защищены.



Список поддерживаемых событий класса библиотеки `oktell.js` перечислен в таблице.

Список поддерживаемых событий класса библиотеки `oktell.js`

Событие	Описание
connect	Событие успешного соединения с сервером
connectError	Событие ошибки соединения с сервером в методе connect. Коды ошибок такие же, как для callback-функции метода connect
disconnect	Событие закрытия соединения с сервером. В callback-функцию передается объект с описанием причины разрыва соединения
statusChange	Событие изменения состояния агента. В callback-функцию передается два строковых параметра — новое и прошлое состояние
ringStart	Событие начала входящего вызова
ringStop	Событие завершения входящего вызова
backRingStart	Событие начала обратного вызова
backRingStop	Событие завершения обратного вызова
callStart	Событие начала исходящего вызова
callStop	Событие смены UUID звонка
talkStart	Событие начала разговора
talkStop	Событие окончания разговора
holdAbonentLeave	Событие выхода абонента из удержания. В callback-функцию передается объект abonent с информацией по абоненту
holdAbonentEnter	Событие входа абонента в удержание. В callback-функцию передается объект abonent с информацией по абоненту
holdStateChange	Событие изменения состояния удержания. В callback-функцию передается информация по удержанию
stateChange	Событие изменения состояния линии
abonentsChange	Событие изменения списка текущих абонентов
flashstatechanged	Низкоуровневое событие изменения состояния удержания
userstatechanged	Низкоуровневое событие изменения состояния пользователя

