

Соответствие между клиентскими вызовами и REST-методами

В таблице перечислены методы, их типы, соответствующие REST-методы и события, которые связаны с этими REST-методами. В поле **Web SDK** показано, как вызов методов и обработка событий выглядит на стороне клиента.

Description	Web SDK	REST hook	Events
connect			
Установить соединение с WCS	<code>Flashphoner.createSession()</code>	connect	ConnectionStatusEvent, RegistrationStatusEvent
Прямые вызовы - Стриминг			
Проиграть поток	<code>stream.play()</code>	playStream	StreamStatusEvent
Поставить на паузу воспроизведение потока	<code>stream.pause()</code>	pauseStream	StreamStatusEvent
Остановить воспроизведение потока	<code>stream.stop()</code>	stopStream	StreamStatusEvent
Опубликовать поток	<code>stream.publish()</code>	publishStream	StreamStatusEvent
Сохранить снимок потока	<code>stream.snapshot()</code>	snapshot	StreamStatusEvent
Завершить публикацию потока	<code>stream.stop()</code>	unPublishStream	StreamStatusEvent
Прямые вызовы - Звонки			
Сделать исходящий звонок	<code>call.call()</code>	call	CallStatusEvent
Отправить тональный сигнал (DTMF)	<code>call.sendDTMF()</code>	sendDtmf	\-
Ответить на входящий звонок	<code>call.answer()</code>	answer	CallStatusEvent

Description	Web SDK	REST hook	Events
Поставить звонок на удержание	<code>call.hold()</code>	hold	CallStatusEvent
Снять звонок с удержания	<code>call.unhold()</code>	unhold	CallStatusEvent
Перевести звонок на другой номер	<code>call.transfer()</code>	transfer	TransferStatusEvent
Завершить звонок	<code>call.hangup()</code>	hangup	CallStatusEvent
Входящие вызовы - Звонки			
Принять входящий звонок	<code>session.on(SESSION_STATUS.INCOMING_CALL, handler)</code>	OnCallEvent, CallStatusEvent	
Прямые и входящие вызовы - Данные			
Отправить данные	<code>session.sendData()</code>	OnDataEvent, DataStatusEvent	
Принять данные	<code>session.on(SESSION_STATUS.APPLICATION_DATA, handler)</code>	OnDataEvent, DataStatusEvent	
Прямые вызовы - Диагностика			
Отправить отчет об ошибке для сохранения на сервере	<code>session.submitBugReport()</code>	submitBugReport	BugReportStatusEvent
Сохранить логи клиента на сервере	<code>session.pushLogs()</code>	pushLogs	\-
Включить отладочные логи для стриминга	<code>session.startDebug()</code>	sessionDebug	SessionDebugStatusEvent
Выключить отладочные логи для стриминга	<code>session.stopDebug()</code>	sessionDebug	SessionDebugStatusEvent
События websocket сессии			
Статус соединения	<code>session.on(SESSION_STATUS..., handler)</code>	ConnectionStatusEvent	ConnectionStatusEvent
Статус регистрации на SIP сервере	<code>session.on(SESSION_STATUS..., handler)</code>	RegistrationStatusEvent	RegistrationStatusEvent

Description	Web SDK	REST hook	Events
Статус отладки сессии	<code>session.on(SESSION_STATUS.DEBUG, handler)</code>	SessionDebugStatusEvent	SessionDebugStatusEvent
События медиапотока			
Статус потока	<code>stream.on(STREAM_STATUS..., handler)</code>	StreamStatusEvent	StreamStatusEvent
События звонка			
Статус звонка	<code>call.on(CALL_STATUS..., handler)</code>	CallStatusEvent	CallStatusEvent
Другие события			
Событие, отправляемое сервером на бэкенд во время публикации / воспроизведения потока	Not used by the client	StreamKeepAliveEvent	StreamKeepAliveEvent