

# Передача JSON данных подписчикам потока

- [Описание](#)
- [Отправка сообщения](#)
- [Прием сообщения](#)

## Описание

В сборке [2.6.36](#) добавлена возможность отправить с публикующего клиента сообщение с данными в формате JSON всем подписчикам публикуемого потока, а также получить такое сообщение на играющем клиенте.

## Отправка сообщения

Для отправки сообщения используется метод `WCSSStream.sendData()`. Данные должны представлять собой сериализованный JSON объект

[code](#)

```
@IBAction func sendDataPressed(_ sender: Any) {
    let dataByte = dataToSendTextView.text.data(using: .utf8)
    do {
        let data = try JSONSerialization.jsonObject(with: dataByte!, options: []) as! [String:Any]
        publishStream?.sendData(data)
        dataStatus.text = "Data sent";
        dataStatus.textColor = .green;
    } catch {
        dataStatus.text = "JSON is not valid";
        dataStatus.textColor = .red;
    }
}
```

## Прием сообщения

Для приема сообщения необходимо в обработчике `onStreamEvent` проверить: если тип полученного сообщения `fpwcsStreamEventTypeData`, то сообщение содержит сериализованные JSON данные

[code](#)

```
playStream?.onStreamEvent({streamEvent in
    if (streamEvent!.type == FPWCSEApi2Model.streamEventType(toString: .fpwcsStreamEventTypeData)) {
        let currentDateTime = Date()
        let formatter = DateFormatter()
        formatter.timeStyle = .short
        formatter.dateStyle = .none
        do {
            let jsonData = try JSONSerialization.data(withJSONObject: streamEvent!.payload as Any,
options: .prettyPrinted)
            let data = String(data: jsonData, encoding: .utf8) ?? "Fatal data"
            self.dataReceivedTextView.text = self.dataReceivedTextView.text! + formatter.string
(from: currentDateTime) + " - " + data + "\n"
        } catch {
        }
    }
})
```

Таким образом может быть принято сообщение, отправленное из [Web SDK](#) или по [REST API](#)