

# QToneADService



ClosedCaption 의 SPA(SetPenAttributes) 명령 중  
text\_tag 가 15 (text not to be displayed) 일 때 들어오는 평문 데이터를 큐톤광고 정보로서 전달한다.

## Class QToneADService

oipfObjectFactory.createQToneADManager()를 통해 객체를 생성한다.

### Methods of QToneADService

	<b>void startMonitoring()</b>
Description	큐톤광고 정보를 가져오기 위한 모니터링을 시작한다. 큐톤광고 정보가 감지되면 등록된 리스너를 통해서 전달한다.

	<b>void stopMonitoring()</b>
Description	큐톤광고 정보를 가져오기 위한 모니터링을 중지한다.

## usage

```
function qtoneADInfoChanged(str) {  
    // use 'str' as a QTone AD Information.  
}  
var qtADManager = oipfObjectFactory.createQToneADManager();  
qtADManager.onQToneADInfoChanged = qtoneADInfoChanged;  
qtADManager.startMonitoring();  
qtADManager.stopMonitoring();
```

# DQToneADService



ANSI/SCTE 35 2019a spec 을 구현하여 새로 개발된 digital qtone 서비스를 위한 web app interface 이다.

## Class DQToneADService

oipfObjectFactory.createDQToneADManager() 를 통해 객체를 생성한다.

Methods of DQToneADService

	<b>void startMonitoring()</b>
Description	digital 쿠폰광고 정보를 가져오기 위한 모니터링을 시작한다. digital 쿠폰광고 정보가 감지되면 등록된 리스너를 통해서 JSON object 형태로 전달한다.

아래는 전달 받는 JSON Object 의 예시 (예시에서 데이터 값은 의미가 없다.)

```

{
  "SpliceInfo": {
    "table_id": 1,
    "section_syntax_indicator": 2,
    "private_indicator": 3,
    "reserved": 4,
    "section_length": 5,
    "protocol_version": 6,
    "encrypted_packet": 7,
    "encryption_algorithm": 8,
    "pts_adjustment": 9,
    "cw_index": 10,
    "tier": 11,
    "splice_command_length": 12,
    "splice_command_type": 13
  },
  "SpliceInsert": {
    "splice_event_id": 1,
    "splice_event_cancel_indicator": 2,
    "reserved": 3,
    "out_of_network_indicator": 4,
    "program_splice_flag": 5,
    "duration_flag": 6,
    "splice_immediate_flag": 7,
    "reserved_inner": 8,
    "splice_time": {
      "time_specified_flag": 9,
      "reserved": 10,
      "pts_time": 11,
      "pts_time_kst": 1604480400000
    },
    "break_duration": {
      "auto_return": 19,
      "reserved": 20,
      "duration": 21
    },
    "unique_program_id": 22,
    "avail_num": 23,
    "avails_expected": 24
  }
}

```

callback 으로 전달 받는 JSON Object 는 RFC 8259 spec 을 만족한다.

#1

November 4th 2020, 5:36:17 pm



VALID (RFC 8259)

## Formatted JSON Data

```
{
  "SpliceInfo": {
    "table_id": 1,
    "section_syntax_indicator": 2,
    "private_indicator": 3,
    "reserved": 4,
    "section_length": 5,
    "protocol_version": 6,
    "encrypted_packet": 7,
    "encryption_algorithm": 8,
    "pts_adjustment": 9,
    "cw_index": 10,
    "tier": 11,
    "splice_command_length": 12,
    "splice_command_type": 13
  }
}
```



	<b>void stopMonitoring()</b>
Description	digital 쿠폰광고 정보를 가져오기 위한 모니터링을 중지한다.

## Class SpliceVODEventListener

SpliceVODEvent 를 전달 받기 위한 listener.

### Constants of SpliceVODEventListener

name	value	description	state of SpliceVODPlayer
EVENT_OPEN_SUCCESS	0	광고 VOD setup 완료	STATE_OPEN
EVENT_OPEN_FAIL	1	광고 VOD setup fail	STATE_STOP
EVENT_NO_ADVERT	2	광고 없음	STATE_STOP
EVENT_PLAY_ERROR	3	재생 에러 (에러코드는 여기서 정의하지 않음)	STATE_STOP
EVENT_CHANNEL_CHANGE	4	광고 VOD 재생 중 채널이 변경됨	STATE_STOP
EVENT_END_OF_ADVERT	5	광고 VOD 재생이 완료됨	STATE_STOP
EVENT_ADVERT_TIME_END	6	광고 VOD 재생이 완료되지 않았으나 endTime 이 지나서 자동 종료됨	STATE_STOP
EVENT_ADVERT_START	7	광고 VOD 재생 시작됨	STATE_START
EVENT_CLOSE	8	광고 VOD player close 됨	STATE_CLOSE

### Methods of SpliceVODEventListener

	<b>void notifySpliceVODEvent(Number event, Number errorCode)</b>
Description	광고 VOD 재생 상태 변화를 알려 준다.
Argument	<ul style="list-style-type: none"> <li>event : EVENT_xxx (e.g. EVENT_OPEN_SUCCESS)</li> </ul>

Argument	<ul style="list-style-type: none"> <li>• errorCode : EVENT_PLAY_ERROR 에 대한 error code.</li> </ul>
----------	---

	<b>void contentChanged(Number eventCode)</b>
Description	재생중인 광고 VOD 의 contents 가 변경되었음을 알려 준다.
Argument	<ul style="list-style-type: none"> <li>• eventCode 변경된 사항은 Castis event code 로 판단할 수 있으며, Castis event code는 M/W(AltCaptor) 에서 전혀 가공하지 않고, 그대로 MediaContentChangedEvent 를 통해 Application 으로 전달된다.</li> </ul>

## Class SpliceVODPlayer

digital qtone 의 광고 재생을 위한 광고 전용 VOD player.

oipObjectFactory.createSpliceVODPlayer() 를 통해 객체를 생성한다.

### Constants of SpliceVODPlayer

name	value	description	comment
STATE_OPENED	0	광고 VOD setup 완료	
STATE_STARTED	1	광고 VOD 재생 중	
STATE_STOPED	2	광고 VOD stop 됨	광고 영상을 모두 재생했거나 에러가 발생하여 재생 중이 아닌 상태
STATE_CLOSED	3	광고 VOD player close 됨	광고 VOD player가 모든 resource를 해제하고 사용중이지 않은 상태

### Methods of SpliceVODPlayer

	<b>void scheduleAdvert(Number startTime, String advertUrl[,Number duration])</b>
Description	광고영상은 지정된 duration 만큼 재생 후 종료되고 라이브 채널 영상으로 돌아간다.
Argument	<ul style="list-style-type: none"> <li>• startTime : 광고 시작시간 (ms)</li> </ul>
Argument	<ul style="list-style-type: none"> <li>• advertUrl : 광고 영상 URL</li> </ul>
Argument	<ul style="list-style-type: none"> <li>• duration : 광고 영상 재생 시간 (ms) - 생략 가능</li> </ul>

	<b>void rescheduleAdvertDuration(Number duration)</b>
Description	현재 스케줄된 광고의 duration 을 변경한다. 스케줄링된 광고가 없으면 아무 동작도 하지 않는다. 광고 영상은 scheduleAdvert()를 통해 설정된 startTime + duration 만큼 재생하고 종료한다.
Argument	<ul style="list-style-type: none"> <li>• duration : 광고 영상 재생 시간 (ms)</li> </ul>

	<b>void cancelAdvert()</b>
Description	광고 영상을 즉시 중지하고 라이브 방송으로 복귀하고 광고 영상 스케줄을 제거한다. 광고가 영상이 시작하지 않은 상태이면 스케줄만 삭제된다.
Argument	<ul style="list-style-type: none"> <li>• 없음</li> </ul>

	Number getState()
Description	현재 상태를 얻어온다.
Argument	<ul style="list-style-type: none"> <li>없음</li> </ul>
Return	STATE_xxx (e.g. STATE_START)

	String getCurrentAdvertFileName()
Description	현재 재생 중인 광고의 파일명을 리턴한다.
Argument	<ul style="list-style-type: none"> <li>없음</li> </ul>
Return	<ul style="list-style-type: none"> <li>재생중인 상황일 때 : 광고 파일명</li> <li>재생중이 아닐 때 : 빈 문자열 ("")</li> </ul>

	String getAdsInfo()
Description	현재 재생 중인 광고 동영상의 정보를 리턴한다.
Argument	<ul style="list-style-type: none"> <li>없음</li> </ul>
Return	리턴하는 String 의 형태와 리턴 시점 등은 firmware 변경 없이도 VoD 서버 측에서 변경이 가능한 사항이므로 여기에 명시하지 않으며, Castis에 문의하도록 한다. <ul style="list-style-type: none"> <li>재생중인 상황일 때 : 동영상 정보</li> <li>재생중이 아닐 때 : 빈 문자열 ("")</li> </ul>

	void addSpliceVOEventListner(SpliceVOEventListner listener)
Description	SpliceVOEventListner 를 등록한다. 이미 추가된 listener 를 다시 추가하는 경우 해당 listener는 무시되고 아무 동작도 하지 않는다.
Argument	<ul style="list-style-type: none"> <li>listener : SpliceVOEventListner 구현체</li> </ul>

	void removeSpliceVOEventListner(SpliceVOEventListner listener)
Description	등록된 SpliceVOEventListner 를 제거한다. 추가되지 않은 listener 를 제거하려고 하는 경우 해당 listener는 무시되고 아무 동작도 하지 않는다.
Argument	<ul style="list-style-type: none"> <li>listener : addSpliceVOEventListner() 으로 등록했던 구현체</li> </ul>

## usage

```

function qtoneADInfoChanged(jsonObject) {
    // 'jsonObject' is a json object that is including QTone AD Information (SpliceInfo, SpliceInsert).
    console.log("info = " + info + "");
    console.log("JSON.stringify(info) = " + JSON.stringify(info) + "");
    ....
}

var qtADManager = oipfObjectFactory.createDQToneADManager();
qtADManager.onQToneADInfoChanged = qtoneADInfoChanged;
qtADManager.startMonitoring();
qtADManager.stopMonitoring();

dqtadp = oipfObjectFactory.createSpliceVODPlayer();

mycallback = {
    "notifySpliceVODEvent" : function(event, errorCode) {
        console.log("event = " + event + ", errorCode = " + state + "");
        ....
    };

    "contentChanged" : function(eventCode)
    { console.log("eventCode = " + eventCode + "");
        ....
    };
};

dqtadp.addSpliceVODEventListener(mycallback);
dqtadp.removeSpliceVODEventListener(mycallback);

dqtadp.scheduleAdvert(startTime, advertUrl, duration);
dqtadp.rescheduleAdvertDuration(duration);
dqtadp.cancelAdvert();

var playerState = dqtadp.getState();
var adFileName = dqtadp.getCurrentAdvertFileName();
var adInfo = dqtadp.getAdsInfo();

```