This article explains how to use the Wowza Streaming Engine™ Java API and nDVR properties to control playlist requests.

**Note:** nDVR is supported in Wowza Media Server™ 3.0.4 and later. For full functionality, including UTC-based request delegates and MPEG-DASH support, use Wowza Streaming Engine 4.7.7.01 or later.

### How playlists are determined

By default, Wowza nDVR determines which playlist to play when a DVR manifest or playlist is requested via a URL.

**Strobe/OSMF player**

http://[wowza-ip-address]:1935/dvr/myStream/manifest.f4m?DVR

**Apple iOS device**

http://[wowza-ip-address]:1935/dvr/myStream/playlist.m3u8?DVR

**Microsoft Silverlight**

http://[wowza-ip-address]:1935/dvr/myStream/Manifest?DVR

**MPEG-DASH**

http://[wowza-ip-address]:1935/dvr/myStream/manifest.mpd?DVR

The default logic is as follows:

- If the stream is being recorded, it's considered **live**; otherwise, it's considered **recorded**.
- Both **live** and **recorded** use one of the following as a start time:
  - The earliest point in the recording
  - The latest time, minus the DVR window if one is specified
- A **live** stream has no end time.
- A **recorded** stream is played with the end time of the recording.

Due to the nature of live HTTP streaming, **live** streams must cache extra chunks after the currently playing live point. **Recorded** streams don’t have this restriction and include these cached chunks in their playlist.
The playlist/manifest presented to the player differs depending on whether the DVR store is live or recorded. Different player technologies handle these playlists differently as well. For example, live playlists typically start playing at the "live point," while recorded playlists start playing at the earliest time.

Using the playlist request delegate

The `dvrPlaylistRequestDelegate` property is a mechanism to serve up a specific playlist through a Java request. The property can be set in the `Application.xml` file under `/` and should reference a fully-qualified class name that extends the `com.wowza.wms.dvr.DvrBasePlaylistRequestDelegate` class.

When the playlist is requested, the delegate’s `getDvrPlaylistRequest()` method is called to provide the playlist request.

Playlist request

The DVR playlist request delegate generates a playlist request object. The object is passed an application context, the DVR store object (which contains methods for querying the underlying known chunks), and a queryMap of URL parameters that were passed in.

A playlist request has a start time and an optional end time.

A valid start time must be specified. For the start time to be valid, it cannot be before the beginning of the recording, after the end of the recording, or after the specified end time.

If no end time is specified:

- For a live store (one that’s currently recording), the playlist presented to the player is a live playlist.
- For a recorded store (one that isn’t recording), the playlist presented to the player is a recorded playlist, using the end time of the recording.

If an end time is specified for a live DVR store, the playlist is recorded, because live playlists don’t have an end time.

Example playlist delegate

Wowza nDVR includes the playlist request delegate `com.wowza.wms.dvr.impl.DvrStartDurationPlaylistRequestDelegate`. This delegate generates the playlist request based on URL query parameters when specifying the playlist or manifest. For example, you can instruct the playlist to play from minute 1 through minute 6 by specifying a start time of `60000` ms (60 seconds) and a duration of `300000` ms (300 seconds, or 5 minutes).

Strobe/OSMF player

http://[wowza-ip-address]:1935/dvr/myStream/manifest.f4m?
DVR&wowzadvrplayliststart=60000&wowzadvrplaylistduration=300000

Apple iOS device

http://[wowza-ip-address]:1935/dvr/myStream/playlist.m3u8?
DVR\wowzadvrplayliststart=60000&wowzadvrplaylistduration=300000

Microsoft Silverlight

http://[wowza-ip-address]:1935/dvr/myStream/Manifest?
DVR\wowzadvrplayliststart=60000&wowzadvrplaylistduration=300000

MPEG-DASH

http://[wowza-ip-address]:1935/dvr/myStream/manifest.mpd?
DVR\wowzadvrplayliststart=60000&wowzadvrplaylistduration=300000

To use this delegate, add it to the // element of your Application.xml file as follows:

Optionally, include and define the dvrPlaylistDurationQueryParameter and dvrPlaylistStartQueryParameter properties in the same location:

To add debug logging to this delegate, include the dvrPlaylistDebugRequests property in the same location and set it to true:

Using a UTC-based request delegate

Wowza Streaming Engine software also includes the UTC-based playlist request delegate com.wowza.wms.dvr.impl.DvrUtcStartDurationPlaylistRequestDelegate. To use it, add it to
your Application.xml file under // as follows:

Note: By default, the UTC delegate in Wowza Streaming Engine uses GMT time and the UTC format yyyyMMddHHmmss. Although Wowza Media Server used the default format yyyy-MM-dd-HH:mm:ss, we don’t recommend using a format that has dashes, colons, or other special characters. See this list of valid time zone values.

The UTC delegate supports the playlist duration, playlist start query, and playlist debug requests query parameters shown in the previous example.

In addition, it provides the following properties to control the format and conversion of the UTC string provided in the URL:

To use this UTC delegate, the URLs look like the following:

Strobe/OSMF player

http://[wowza-ip-address]:1935/dvr/myStream/manifest.f4m?
DVR&wowzadvrplayliststart=20140211083000&wowzadvrplaylistduration=300000

Apple iOS device

http://[wowza-ip-address]:1935/dvr/myStream/playlist.m3u8?
DVR&wowzadvrplayliststart=20140211083000&wowzadvrplaylistduration=300000

Microsoft Silverlight

http://[wowza-ip-address]:1935/dvr/myStream/Manifest?
DVR&wowzadvrplayliststart=20140211083000&wowzadvrplaylistduration=300000

MPEG-DASH

http://[wowza-ip-address]:1935/dvr/myStream/manifest.mpd?
DVR&wowzadvrplayliststart=20140211083000&wowzadvrplaylistduration=300000
Invoking legacy behavior for invalid requests

In Wowza Media Server software, if the playlist start or duration query parameters didn’t make sense (for example, if the start time was after the end of the recording), the playlist delegate returned the entire playlist of a DVR store. In Wowza Streaming Engine software, however, the base class that writes custom playlist delegates (com.wowza.wms.dvr.DvrBasePlaylistRequestDelegate) returns an empty playlist in these cases.

If the old behavior is desired, you can use the playlist request delegate provided below. This delegate has the same properties as DvrStartDurationPlaylistRequestDelegate except that it returns the entire playlist of the DVR store when an invalid request is made.

To use this delegate, add the following property to your Application.xml file under //:

Creating custom playlist delegates

If you want to provide your own playlist delegate, it would look very much like DvrStartDurationPlaylistRequestDelegate. You must provide two public methods and logic to determine your playlist request.

This method is called for a single bitrate playlist request:

This method is called to request an adaptive bitrate playlist:

See the Wowza Streaming Engine User’s Guide for information on how to add your own code.
import com.wowza.wms.application.WMSProperties;
import com.wowza.wms.dvr.*;
import com.wowza.wms.dvr.IDvrConstants.DvrTimeScale;
import com.wowza.wms.httpstreamer.model.IHTTPStreamerApplicationContext;
import com.wowza.wms.logging.WMSLoggerFactory;

public class DvrStartDurationPlaylistRequestDelegate extends DvrBasePlaylistRequestDelegate {
    private static final String CLASSNAME = "DvrStartDurationPlaylistRequestDelegate";
    private static final Class CLASS = DvrStartDurationPlaylistRequestDelegate.class;

    public static final String DVR_QUERYSTR_PLAYLIST_DURATION = "wowzadvrplaylistduration";
    public static final String DVR_QUERYSTR_PLAYLIST_START = "wowzadvrplayliststart";
    public static final String PROPKEY_DVR_PLAYLIST_DURATION_QUERY_PARAMETER = "dvrPlaylistDurationQueryParameter";
    public static final String PROPKEY_DVR_PLAYLIST_START_QUERY_PARAMETER = "dvrPlaylistStartQueryParameter";
    public static final String PROPKEY_DVR_PLAYLIST_LOG_REQUESTS = "dvrPlaylistDebugRequests";

    private boolean doDebug = false;

    public DvrPlaylistRequest getDvrPlaylistRequest(IHTTPStreamerApplicationContext appContext, IDvrStreamStore store, Map queryMap) {
        DvrPlaylistRequest availablePlaylist = getDefaultPlaylistRequest(DvrTimeScale.DVR_TIME, store);
        DvrPlaylistRequest newRequest = createRequestFromQueryParams(appContext, queryMap, availablePlaylist);
        return newRequest;
    }

    public DvrPlaylistRequest getDvrPlaylistRequest(IHTTPStreamerApplicationContext appContext, List stores, Map queryMap) {
        DvrPlaylistRequest availablePlaylist = getDefaultPlaylistRequest(DvrTimeScale.DVR_TIME, stores);
        DvrPlaylistRequest newRequest = createRequestFromQueryParams(appContext, queryMap, availablePlaylist);
        return newRequest;
    }

    private DvrPlaylistRequest createRequestFromQueryParams(IHTTPStreamerApplicationContext appContext, Map queryMap, DvrPlaylistRequest availablePlaylist) {
        DvrPlaylistRequest newRequest = new DvrPlaylistRequest();
        if (availablePlaylist != null) {
            newRequest.setPlaylistEnd(availablePlaylist.getPlaylistEnd());
            newRequest.setPlaylistStart(availablePlaylist.getPlaylistStart());
        }

        WMSProperties dvrProperties = getDvrProperties(appContext);
        String playStartQueryParameter = dvrProperties.getPropertyStr(PROPKEY_DVR_PLAYLIST_START_QUERY_PARAMETER);
        newRequest.setPlaylistStart(playStartQueryParameter);
        return newRequest;
    }
}

}
String playStartQueryParameter = dvrProperties.getPropertyStr(PROPKEY_DVR_PLAYLIST_START_QUERY_PARAMETER, DVR_QUERYSTR_PLAYLIST_START);
String playDurationQueryParameter = dvrProperties.getPropertyStr(PROPKEY_DVR_PLAYLIST_DURATION_QUERY_PARAMETER, DVR_QUERYSTR_PLAYLIST_DURATION);
this.doDebug = dvrProperties.getPropertyBoolean(PROPKEY_DVR_PLAYLIST_LOG_REQUESTS, doDebug);
String playStartStr = queryMap.get(playStartQueryParameter);
String playDurationStr = queryMap.get(playDurationQueryParameter);
if (doDebug) {
    WMSLoggerFactory.getLogger(CLASS).info(String.format("%s : Request: %s:%s %s:%s ", CLASSNAME, playStartQueryParameter, playStartStr, playDurationQueryParameter, playDurationStr));
    WMSLoggerFactory.getLogger(CLASS).info(String.format("%s : Available Playlist: %s ", CLASSNAME, availablePlaylist));
}
if (availablePlaylist == null) {
    if (doDebug) {
        WMSLoggerFactory.getLogger(CLASS).warn(String.format("%s : availablePlaylist is null. ", CLASSNAME));
    }
    return newRequest;
}
if (playStartStr != null) {
    try {
        long playStart = Long.parseLong(playStartStr);
        if (playStart < availablePlaylist.getPlaylistStart()) {
            if (doDebug) {
                WMSLoggerFactory.getLogger(CLASS).warn(String.format("%s : requestedStart:%d < availableStart:%d.  Using availableStart. ", CLASSNAME, playStart, availablePlaylist.getPlaylistStart()));
            }
        } else if (availablePlaylist.hasSpecifiedEnd() && playStart > availablePlaylist.getPlaylistEnd()) {
            if (doDebug) {
                WMSLoggerFactory.getLogger(CLASS).warn(String.format("%s : requestedStart:%d > availableEnd:%d. ", CLASSNAME, playStart, availablePlaylist.getPlaylistEnd()));
            }
        } else {
            newRequest.setPlaylistStart(playStart);
        }
    } catch(Exception e) {
    }
}
if (playDurationStr != null) {
}
Example: Query DVR store times

The following example code queries the DVR store in the playlist delegate to determine available times. Similar code may be useful when creating your own playlist request delegate.

The `IDvrTimeMap` maps between DRV manifest time, packet time, and UTC time when the time is reset:
Example: Create a playlist request based on UTC time

The following example code creates a playlist request based on UTC time.
String UTC_FORMAT = "yyyy-MM-dd-HH:mm:ss";
DateFormat formatter = new SimpleDateFormat(UTC_FORMAT);

public DvrPlaylistRequest getDvrPlaylistRequest(IHTTPStreamerApplicationContext appContext, IDvrStreamStore store, Map queryMap) {
// . . . This could come from URL param or some other manner
String startStr = "2012-02-14-11:30:00";
// This is entire playlist request in UTC
DvrPlaylistRequest fullPlaylistRequest = getDefaultLivePlaylistRequest(DvrTimeScale.UTC_TIME, store);
// Convert start String to UTC
Date date = null;
if (!StringUtils.isEmpty(startStr)) {
    try {
        date = (Date)formatter.parse(startStr);
    } catch (ParseException e) {
        date = null;
// e.printStackTrace();
    }
}
// System.out.printf("'%s' --> date:%s", startStr, date);
// If the date specified is less than the initial date we have to play, its not valid
if (date != null && date.before(new Date(fullPlaylistRequest.getPlaylistStart()))) {
    System.out.println("Requested start time before actual recording.");
    date = new Date(fullPlaylistRequest.getPlaylistStart());
}
DvrPlaylistRequest req;
if (date != null) {
    req = new DvrPlaylistRequest(DvrTimeScale.UTC_TIME);
    req.setPlaylistStart(date.getTime());
} else {
    // Use default
    req = super.getDvrPlaylistRequest(appContext, store, queryMap);
}
return req;
}