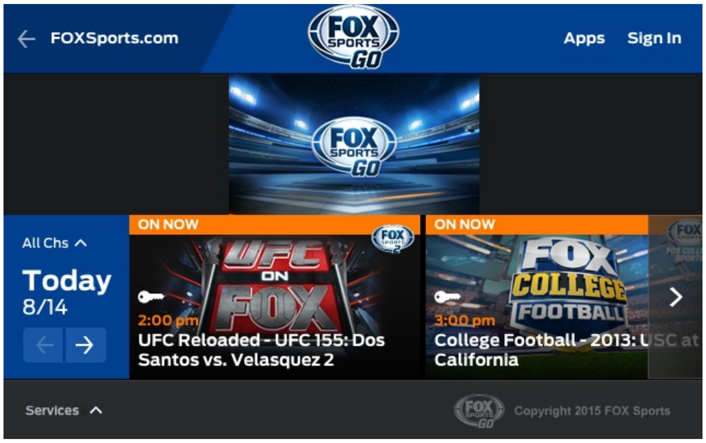
foxsportsgo

foxsports.com/foxsportsgo


FOXSports.com
FOX SPORTS GO
Apps
Sign In
All Chs
Today
8/14
ON NOW
2:00 pm
UFC Reloaded - UFC 155: Dos Santos vs. Velasquez 2
ON NOW
3:00 pm
College Football - 2013: USC at California
Services
Copyright 2015 FOX Sports

FOX Sports Go (FSGO) is a video broadcasting portal. FSGO features live video feeds (eg, World Cup), "bonus feeds" and Cable subscriber access to Fox Sports network television programming.

FSGO applications exist for Web, Android, iOS, Kindle Fire, Windows PC and Windows Phone.

This Document, credit to **Mark Dolar**, **David Feldstein**, **Reggie Christian**, **Nesim Sisa**, **Ron Rodrigo**, **Chris DePauw**

---

foxsportsgo video

Video programs are provided through two channel types. *Linear channel* types feature programs correlating to a *linear television* sports network. *Virtual channel* types feature category-filtered programming for things like World Cup Bonus Feeds, U.S. Open or Big East.

Note that licensing prevents distribution of some programs on FSGO.

- FOX Broadcasting Company
  - 182 stations available as channels *ex, KTTV*
- FOX Sports 1
- FOX Sports 2
- FOX Regional Sports Networks
  - 23 networks available as channels *ex, FOX Sports Arizona*
- FOX College Sports
- FOX Deportes
- FOX Sports Go Extras *ex, World Cup*
- Virtual (eg, USGA, U.S. Open, Big East, etc.)

FSGO obtains a given program's video stream metadata from *thePlatform* as JSON. The metadata for each program includes a path to a .smil file. The .smil file includes more metadata and a path to a master.m3u8 playlist file. The playlist includes paths to .ts transport stream files (mpeg2-container) that are decoded and rendered as video using the HLS H264 protocol.

Per Apple's HLS documentation, the player requests *master.m3u8* and *prog.m3u8* files.

> *The master index file is downloaded only once, but for live broadcasts the alternate index files are reloaded periodically. The first alternate listed in the master index file is the first stream used—after that, the client chooses among the alternates by available bandwidth.*

example

- *JSON metadata from thePlatform*, origin, json
- *meta.smil*, origin, .smil, .txt
- *master.m3u8*, origin, .m3u8, .txt
- *prog.m3u8*, origin, .m3u8, .txt
- *FOX affiliated and owned station list*, origin, .txt
- *FOX affiliated and owned regional network list*, origin, .txt

foxsportsgo ads

A lesson in
CO W-OLOGY
g to a group of animals called
s means they have four stomach
not four stomachs) that each play
in the digestion of food. The four
mach are called, in order, rumen,
omasum, and the abomasum.
MOO-VE ME
The ins & outs of being
a California dairy cow.

Ads are targeted to specific programs using program metadata requested from *thePlatform* (example). Data used for targeting includes things like eventId, leagueId, leagueNames, teamNames, sportName. Analytics, likewise, target programs using the same metadata.

Ad servers

- **DFP** *(DoubleClick for Publishers, Google)*, serves national and house ads
- **MMP** *(Media Monetization Platform, Anvato)*, serves local ads

Ad rolls

- **Prerolls** render before a video and are obtained directly by the client through VAST/xml communication with an ad server.
- **Midrolls** render mid video and are obtained indirectly through a technique called "*m3u8 substitution*" -a session-manager, called an "*ad-stitcher*", communicates with the ad server (VAST/xml) on behalf of the client.

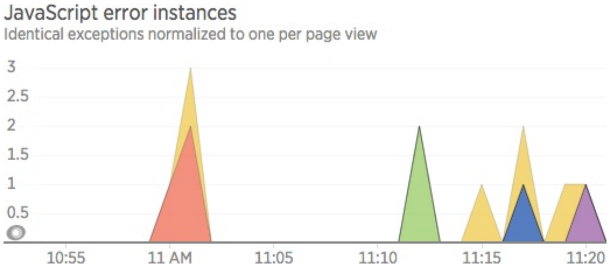
The *ad-stitcher* tracks the playout stream to replace m3u8 paths featuring slate and linear TV ads with m3u8 paths from the ad servers. It uses SCTE ("*scutty*") tags embedded in the video stream to anticipate breaks, their length and the amount of time allocated for national or local ads. It makes VAST feed requests on behalf of the client to DFP and MMP, parses the required data and sends it to the client. That data includes m3u8 paths and analytics beacon urls to be used to confirm ad playback at first quartile, midpoint etc.

The ad-stitcher maintains a session id for each session and logs VAST activity against that session-id.

example

- *preroll VAST request*, txt
- *midroll VAST request*, **pending**

foxsportsgo analytics



FSGO obtains video stream metadata from *thePlatform* as json. The metadata for each program is used to generate analytics data. Metadata includes things like eventId, leagueId, leagueNames, teamNames, sportName.

1. **Real-User Monitoring (New Relic)**

   A script in the head of the document used to collect browser request activity and scripting errors.

2. **Adobe Analytics / Omniture**

   A page-level analytics software composed of other analytics software with names like “localytics” and “analytics digital”. It associates the client with page content -for example, "sports", "video" and "LA Rams". This “first party” data, is available for reporting in a report suite.

   Cookies ***s_cc*** and ***s_sq*** are created and used by Adobe Analytics

3. **Adobe Audience Manager, *AAM***

   A common type of analytics softwre called a DMP (*Data Managment Platform*). AAM joins first-party data with third-party sources for improved targeting. Using AAM would allow an advertiser to target clients who are women, age 25 to 35, interested in the NBA. Applications using AAM will collect different types of data at different times.

   Integration with FSGO is incomplete, but the first goal is to join AAM-yielded data with DFP preroll ad requests.

4. **Heartbeat (Adobe)** *part of omniture*

   A video analytics service being integrated with Nielsen by Adobe. Its measurements are compatible with those from other systems that follow Nielsen's standards. It is used in the player (not the application itself) to measure activity at the session level -tracking one video title per session. In contrast to the Adobe Analytics, Heartbeat is both event-driven *and* time-driven.

   Event-specifc requests are sent to Heartbeat when video state is changed (eg, stop, play, ad-playback). Events are customizable and may have meta-data attached, for example, when an ad plays the associated ad-id is sent to Heartbeat as metadata. In addition, Heartbeat pings are sent at 10-second intervals to measure time between recorded events. Sessions that do not ping for 2 minutes are expired.

   The Anvato player wraps video event calls with corresponding Heartbeat event methods.

5. **Tealium IQ** *tealium*, *utag.js*

   A common type of analytics software called a TMS (*Tag Management Platform*). Tealium IQ consolidates the data for multiple analytics services behind one interface. It is controlled through the use of custom 'Tags', behaviour-directing patterns interpreted by Tealium. Foxsports uses hundreds of custom Tags that drive tracking-pixels, beacons and other request activity.

   Mobile FSGO applications removed Tealium due to performance considerations.

example

- *JSON metadata from thePlatform*, origin, json
- *Heartbeat event request*, txt
- *Heartbeat ping request*, txt

foxsportsgo auth services overview

Authentication and authorization involve multiple services.

```
client
├── adobe-pass
└── anvato
    ├── dapz
    └── tkx
        └── tpi
            ├── maxmind
            ├── mpx
            |   └── userdata
            └── uzt
```

Permissions are negotiated for each user and media combination (called “*the entitlement process*”). NFL media choices, for example, are served based on the location of a user's device (*geozip*). NBA media choices are served based on the location of a user's registered service (*homezip*). When a client accesses regional (RSN) or local (FBC) programming, the zip code is decrypted and used to return programming for which the user in that zip code (with that subscription package) is entitled to.

Production AdobePass signin is preferred for testing as it is more stable and production accounts may consume staging mpx feeds. The 'Darwin' MVPD exists for testing purposes.

1. **adobe-pass** *authentication*
   adobe.com/products/adobe-pass.html

   > *Adobe Pass provides user/device authentication and authorization across multiple content providers. The user must have a valid cable TV or satellite TV subscription.*

   Adobe-pass provides authenticated clients with long-lived “*authn*” (authentication) and “*authz*” (authorization) tokens. Adobe uses the tokens for authorized resource access with supported MVPDs (MVPDs integrate with an Adobe-provided “*Access Enabler*” component to handle SAML SSO exchanges with adobe-pass)

   authz tokens are used to request short-lived “*shortauth*” tokens that are in turn used to request stream data.

   Standard adobe-pass implementations are also used by ESPN and ATT U-verse.

   Note that web clients embed an ip address in signatures applied to authz tokens.

2. **anvato** *authorization*

   The anvato service returns zip-code related data collected from other servers.

3. **tkx**, *Token exchange*
   tkx-prod.fs.anvato.net
   `anvato -> Tkx`

   middleware layer which wraps thePlatform Identity adapter (tpi) with session management

4. **tpi**, *thePlatform Identity*
   tpi.fs.anvato.net
   `anvato -> Tkx -> tPI`

   adapter layer that aggregates data from Fox UZT, MaxMind and MPX

5. **uzt** *User Zip-code Table*
   fstve.com/uzt
   `anvato -> Tkx -> tPI -> UZT`

   Serves a JSON-formatted list of channels available for a given zipcode and mvpdid

6. **maxmind**
   maxmind.com
   `anvato -> Tkx -> tPI -> MaxMind`

   Serves a geoip for a given ip address, used by the application as geozip

7. **dapz**
   dapz.fs.anvato.net/dapz
   `anvato -> dapz`

   Unzips the Adobe Pass encrypted subscriber zip codes and returns plaintext used as metadata with the MpX identity adapter in the user object authenticated workflow

example

- *AdobePass Overview*, origin
- *AdobeAccess Overview (AccessEnabler)*, pdf
- *Adobe Access, auth test page*, origin
- *Web Panel, MVPD List Managment* (VPN-required), stg, prod
- *uzt documentation*, origin
- *uzt test page, UZT req/res* (fs-playa-lab required), origin
- *uzt sample response, uzt mvpd:ATT,mvpdid:17384,zip:90094*, json
- *uzt sample request*, txt

foxsportsgo authn post-signin requests

A post-signin authentication request cycle is summarized below. An actual request sequence is archived and referenced here.

1. **Obtain Basic Configuration**, *anvato* md

   - GET anvato-api-config.s3.amazonaws.com/anvacks/fs2go...1f2f9a7.json
   - Request a static JSON file defining general configuration for things like analytics, authorization, feed locations and video settings.

2. **Obtain tkx Server Time**, *anvato* md

   - POST tkx-prod.fs.anvato.net/rest/v2/server_time?anvack=fs2go...b49cf1f2f9a7
   - Request a timestamp from tkx, *probably* used with "replay protection".

3. **Obtain List of Supported MVPDs**, *adobe access enabler* md

   - GET sp.auth.adobe.com/adobe-services/config/fs2go?noflash=tr...440111353849
   - Request a list of MVPD configurations passed to Adobe Access and Adobe Pass

4. **Obtain Authn Token and Session Data**, *adobe access enabler* md

   - POST sp.auth.adobe.com/adobe-services/session
   - If authentication succeeds, returns xml-formatted authn w/ user metadata. Else, create default "*user_object*" and "*tptoken*" (called "***user_object 1***") step 9

5. **Obtain Userid and Zip (encrypted)**, *adobe access enabler* md

   - POST sp.auth.adobe.com/adobe-services/usermetadata
   - Uses authn token to obtain an encrypted user zip-code.

   Note, this is an adobe-encrypted value, the plaintext of which may be an MVPD-encrypted cipher. For example Comcast zip-codes are pre-encrypted before adobe-encryption. Official documentation.

   Note also the web application requests this multiple times, likely in error as only one response data is used thereafter

6. **Obtain Decrypted Zip**, *anvato* md

   - POST dapz.fs.anvato.net/dapz
   - The encrypted zip is used to obtain a decrypted zip

7. **Signout, thePlatform**, *anvato* md

   - GET euid.theplatform.com/idm/web/Authentication/sig...kZAucECkIHBS
   - Pre-existing "*user_object*" and "*tptoken*" are cleared from session management at tpi. *user_object* and *tptoken* are found in cookies named 'user_object' and 'token' and under .www.foxsports.com.

8. **Obtain tptoken**, *anvato* md

   - POST tpi.fs.anvato.net/tpa2?anvack=fs2go_fs2go_go_w...b49cf1f2f9a7
   - User data is posted to tpi in exchange for a "*tptoken*" -used to obtain session data from tpi.

9. **Obtain UserObject** md

   - GET euid.theplatform.com/idm/web/Self/getSelf?schem...oRAWgED6oJCW
   - tptoken is used to obtain "***user_object 2***", a non-default user object composed of data from multiple services such as `uzt`, `dapz`, `madmax`...

10. **Channel Preauthorization**, *adobe access enabler* md

    - POST sp.auth.adobe.com/adobe-services/preauthorize
    - authn token is sent with a list of resource_ids (fs2, spso) to adobe. A new list is returned with `true` or `false` values for each item.

11. **Signout, thePlatform**, *anvato* md

    - GET euid.theplatform.com/idm/web/Authentication/si...oRAWgED6oJCW
    - Pre-existing "*user_object*" and "*tptoken*" (***user_object 2***) are cleared from session management at tpi.

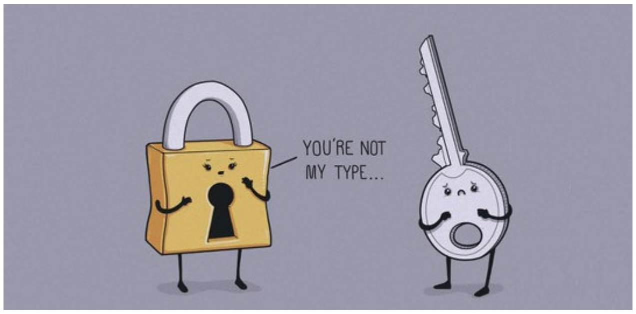
12. **Obtain new tptoken**, *anvato* md

    - POST tpi.fs.anvato.net/tpa2?anvack=fs2go_fs2go_go_w...b49cf1f2f9a7
    - User data is posted to tpi in exchange for a "*tptoken*" -used to obtain session data from tpi.

13. **Obtain UserObject** *anvato* md

    - GET euid.theplatform.com/idm/web/Self/getSelf?schem...sQD6QIB6QEBW
    - tptoken is used to obtain "***user_object 3***", a non-default user object composed of data from ***user_object 2*** with additional properties "authorizedResources", "afterPreflight".

foxsportsgo authz media requests



A user-selected authorization media request cycle is summarized below. An actual request sequence is archived and referenced here.

1. **Obtain User Media Options**, *anvato* md
   - GET anvato-api-config.s3.amazonaws.com/anvacks/fs2go...1f2f9a7.json
     GET feed.theplatform.com/f/fKc3BC/dSIgD7iSdSrr?range=1-40...|tudor}
     GET feed.theplatform.com/f/fKc3BC/dSIgD7iSdSrr?range=41-80...|tudor}
     GET feed.theplatform.com/f/fKc3BC/dSIgD7iSdSrr?range=81-120...|tudor}
     GET feed.theplatform.com/f/fKc3BC/dSIgD7iSdSrr?range=121-160...|tudor}
     ...
     GET feed.theplatform.com/f/fKc3BC/dSIgD7iSdSrr?range=841-880...|tudor}
   - Metadata is requested for a user's authorized resource_ids -metadata for all media items up to one week in the future, chronologically sorted. Each media is potentially available to the user, but must pass the “*the entitlement process*” before it is fully accessible.
2. **Authorize**, *accessenablerhelper* md
   - POST sp.auth.adobe.com/adobe-services/authorize
   - obtain a long-lived authz token (8-24 hours) associated with resource_id and redirect_uri.
3. **ShortAuthorize**, *accessenablerhelper*
   - POST sp.auth.adobe.com/adobe-services/shortAuthorize
   - Obtain a short lived token for media access (~5min?). The token is passed to the platform which uses an adobe pass media token verifier library.
4. **dapz**
   - POST dapz.fs.anvato.net/dapz
   - obtain decrypted zip code
5. **smil file**
   - GET link.theplatform.com/s/fKc3BC/8Ucgnx1Y9kCe/meta.smil?...&format=smil
   - Obtains the smil file used to play the selected video
6. **Obtain Ads/Analytics**
   - POST tkx-prod.fs.anvato.net/rest/v2/mcp/video/fs2?anvack=fs2...gfhf8Q%3D%3D
   - User zip-code and video metadata are passed to tkx to obtain ads/analytics data applied to the video

foxsportsgo jargon

1. **MVPD** *Multichannel Video Programming Distributor*
    - https://en.wikipedia.org/wiki/Multichannel_video_programming_distributor
    - short-hand for distributors like AT&T, Verizon and Comcast
2. **DFP** *DoubleClick For Publishers*
    - https://en.wikipedia.org/wiki/DoubleClick_for_Publishers
    - ad-stitching and delivery service for national and house ads
3. **VOD** *Video on Demand*
    - https://en.wikipedia.org/wiki/Video_on_demand
    - video seen in *viewable* time at viewer demand (vs *linear* broadcast time)
4. **VAST** *Video Ad Serving Template*
    - https://en.wikipedia.org/wiki/Video_Ad_Serving_Template
    - IAB-specified standard for video server and client communication using XML
5. **IAB** *Interactive Advertising Bureau*
    - https://en.wikipedia.org/wiki/Internet_Advertising_Bureau
    - powerful trade association for online and mobile advertising
6. **MPX** *Media Publishing Solution (formerly MPS)*
    - https://en.wikipedia.org/wiki/ThePlatform#Products
    - video management and publishing system
7. **MCP** *Media Content Platform*
    - https://mcp.anvato.com
    - a metadata managment platform from Anvato, similar to thePlatform's *mpx*. *information still pending -mcp is used for ad detection and replacement*
8. **HLS** *HTTP Live Streaming*
    - https://en.wikipedia.org/wiki/HTTP_Live_Streaming
    - (H264) adaptive streaming communications protocol by Apple
9. **MMP** *Media Monetization Platform*
    - http://fs2go.mmp.anvato.com/
    - ad-stitching and delivery service for local ads
10. **SMIL** *Synchronized Multimedia Integration Language*
    - https://en.wikipedia.org/wiki/Synchronized_Multimedia_Integration_Language
    - (XML) markup language to describe multimedia presentations
11. **authZ** / **authN**
    - https://wikidocs.adobe.com/wiki/pages/viewpage.action?pageId=78774427
    - terms for SAML long-lived authentication (authN) and authorization (authZ) tokens
12. **UZT** *User Zip-code Table*
    - fstve.com/uzt
    - lookup table used by thePlatform to obtain the list of channels (JSON) available to a user's (encrypted) zipcode and MVPD subscription.
13. **tPI** *thePlatform, Inc.*
    - tpi.fs.anvato.net
    - a service layer used with thePlatform that maintains sessions used for quickly obtaining user/zip-code related media choices

foxsportsgo environments

Trunk is continuously deployed to *dev* environments and tagged branches are intermittently deployed to the others.

	environments
dev	foxsportsgo.dev.data.foxsports.com
dev2	foxsportsgo.dev2.data.foxsports.com
qa	foxsportsgo.qa.data.foxsports.com
qa2	foxsportsgo.qa2.data.foxsports.com
stg	foxsportsgo.staging-data.foxsports.com
stg2	foxsportsgov2.staging-data.foxsports.com
prod	foxsportsgo.data.foxsports.com (origin)
prod	foxsports.com/foxsportsgo

Configurations differ. Staging mpx feeds may serve simulated content for testing. A basic list of configurations is below (detailed list is here):

configuration	dev	qa	stg	prod
global config	qa	qa	stg	prod
thumbnails	qa	qa	stg	prod
tealium	dev	qa	qa	prod
playlist, *mpx feed*	stg	stg	stg	prod
anvato accesskey	stg	stg	stg	prod
anvato accesscontrol	stg	stg	stg	prod
anvato plugin heartbeat	dev	dev	dev	dev
anvato plugin dfp preroll suffix	stg	stg	stg	prod

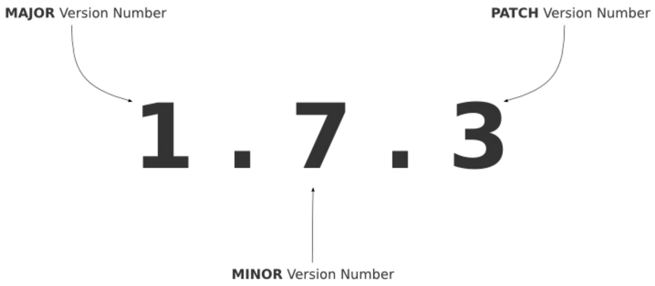
example

- *configurations list*, json

foxsportsgo repositories

	repositories
fsgowebv2	fbc.svn.cloudforge.com/fsgowebv2
fsgo_player	fbc.svn.cloudforge.com/fsgo_player

foxsportsgo versioning



Naming follows the semver pattern, *MAJOR.MINOR.PATCH*.

- *ex, tag name*
  - `fsgo-2.1.1`
- *ex, branch name (created from tag fsgo-2.1.0)*
  - `branch_from_fsgo-2.1.0`

---

changelog

05.01.2015

- published first release

08.06.2015

- added version.js file, DV

08.17.2015

- added gulp minification, DV