twitch.json

misinformation

Page ID: 96

Source: https://dev.twitch.tv/docs/extensions/reference/

['authentic']

Extensions Reference | Twitch Developers Asset 20 Products Showcase Support Blog Docs Tutorials Log in with Twitch Contents Overview Twitch API EventSub Chat & Chatbots Authentication Organizations Drops Embedding Twitch Extensions Get started Required Technical Background Designing Extensions Building Extensions Using the Twitch API in an Extension Front End Life Cycle Management Monetization A/B Testing Using Google Analytics Increase Feedback Load Testing Extensions Submission Best Practices Guidelines & Policies Reference JavaScript Helper Helper: Extensions Helper: Actions Helper: Configuration Helper: Feature Flags Helper:

Bits Helper: Viewer Client Query Parameters JWT Schema Insights & Analytics Mobile Deep Links PubSub Video Broadcast Twitch CLI Changelog Product Lifecycle Extensions Reference This reference covers the JavaScript helper , client query parameters , and the JWT schema .

For Twitch API endpoints related to Extensions, refer to the Twitch API reference .

JavaScript Helper Namespace See … window.Twitch.ext Helper: Extensions window.Twitch.ext.actions Helper: Actions window.Twitch.ext.configuration Helper: Configuration window.Twitch.ext.features Helper: Feature Flags window.Twitch.ext.bits Helper:

Bits Helper: Extensions The Extensions JavaScript helper uses these strings: version — This encodes the Helper version in 1.1.1 ( semantic versioning ) format. environment — This encodes the environment.

For external users, this is always production .

onAuthorized window.

['authentic']

Twitch.ext.onAuthorized:

function(authCallback: Function)

This callback is fired each time the JWT is refreshed.

authCallback is a function with one argument, an object with these properties: Property Type Description channelId string Channel ID of the page where the extension is iframe embedded.

clientId string Client ID of the extension.

token JWT JWT that should be passed to any EBS call for authentication.

helixToken JWT JWT that can be used for front end API requests.

See Using the Twitch API in an Extension Front End .

userId string Opaque user ID.

For example: window .

Twitch .

ext .

['authentic', 'false']

Valid value: "broadcaster" .

version string The version of configuration with which the segment is stored.

content string The string-encoded configuration.

Helper: Feature Flags isBitsEnabled window.

Twitch.ext.features.isBitsEnabled: Boolean If this flag is true , Bits-in-Extensions features will work in your extension on the current channel.

If this flag is false , disable or hide the Bits in Extensions features in your extension.

This flag will be false if: You did not enable Bits support in the Monetization tab in the Extensions manager.

The broadcaster is not eligible to receive Bits.

The broadcaster disabled Bits in Extensions features for your extension.

isChatEnabled window.

Twitch.ext.features.isChatEnabled: boolean

If this flag is true , you can send a chat message to the current channel using Send Extension Chat Message (subject to the authentication requirements documented for that endpoint).

This flag may be false if: You did not enable the “Chat Capabilities” option for your extension on the Twitch developer site.

The broadcaster disabled chat for your extension.

isSubscriptionStatusAvailable window.

Twitch.ext.features.isSubscriptionStatusAvailable: boolean If this flag is true , your extension has the ability to get the subscription status of identity-linked viewers from both the helper in the window.

Twitch.ext.viewer.subscriptionStatus object and via the Twitch API.

This flag may be false if: You did not enable the “Subscription Status” option for your extension on the Twitch developer site.

The broadcaster did not grant your extension the permission to view their viewer’s subscriber status when they activated the extension, or they revoked that permission from the Manage Permissions page.

onChanged [features] twitch.ext.features.onChanged: void function(callback: Function(changes: []string))

This function enables you to receive real-time updates to changes of the features object.

If this callback is invoked, you should re-check the window.Twitch.ext.features object for a change to any feature flag your extension cares about.

The callback is called with an array of feature flags which were updated.

callback is a function with one argument: Argument Type Description changed string

['false']

Otherwise, the promise rejects with an error; this can occur, for instance, if the extension is running in the legacy developer rig or an older version of the mobile app (earlier than V6.4), which does not support Bits in Extensions actions.

Products are returned only if the extension is configured for Bits (see the Extensions Monetization Guide ).

The products that are returned depend on the state of both the extension version (whether it is released) and the product (whether the kit is in development): Is the extension version released?

Which products are returned?

Is the inDevelopment field included in the returned products?

Yes Products whose catalog entries specify inDevelopment = false or undefined

No

No All products for that extension Yes Products have the following fields:

Product Field Type Description cost Cost Cost object.

amount string Number of Bits required for the product.

type string Always the string "bits" .

Reserved for future use.

displayName string Registered display name for the SKU.

inDevelopment boolean or undefined This field is returned only for extension versions that are not in the Released state.

Valid values: true - Indicates that the product's catalog entry specifies it is in development.

If the developer tests the use of Bits, the Bits are not deducted from the user's balance.

false - Indicates that the product's catalog entry specifies it is not in development.

If the developer tests the use of Bits, the Bits are deducted from the user's balance.

undefined - Indicates that the product's catalog entry does not specify anything about whether it is in development.

If the developer tests the use of Bits, the Bits are deducted from the user's balance.

sku string Unique ID for the product.

onTransactionCancelled twitch.ext.bits.onTransactionCancelled: void function(callback:Function)

This function takes a callback that is fired whenever a transaction is cancelled.

['false']

The extension’s mode.

Valid values: "config" , "dashboard" , "viewer" .

platform string The platform on which the Twitch client is running.

Valid values: "mobile" , "web" .

popout boolean Indicates whether the extension is popped out.

If true , the extension is running in its own window; otherwise, false .

state string The release state of the extension.

Valid values: "testing" , "hosted\_test" , "approved" , "released" , "ready\_for\_review" , "in\_review" , "pending\_action" , "uploading" .

JWT Schema Schema Item Type Description channel\_id string Numeric ID of the channel from which your front-end is being served.

exp NumericDate Expiration time for the JWT, expressed as seconds since Jan 1, 1970.

(For a definition of NumericDate, see Terminology in the JWT RFC.)

is\_unlinked boolean true when the token is for a user that previously shared identity; otherwise, false .

opaque\_user\_id string Identifies the session using this JWT.

Any token generated by your EBS should leave this field blank.

When present, the value is interpreted as follows: Values beginning with "U" are stable references to a Twitch account across sessions and channels.

You can use them to provide persistent services to users.

Values beginning with "A" are transient references to an anonymous Twitch session; the user has not logged into the Twitch website.

These values are not stable and should never be associated with persistent data.

['false']

Multiple endpoints require this role.

user\_id string The user’s Twitch user ID.

This is provided only for users who allow your extension to identify them.

There are no guarantees this will be available, as users can revoke an extension’s ability to identify them at any time.

Users who revoke an extension’s ability to identify them are issued a new opaque identifier.

Example JWT Payload { "exp" : 1484242525 , "opaque\_user\_id" : "UG12X345T6J78" , "channel\_id" : "test\_channel" , "role" : "broadcaster" , "is\_unlinked" : "false" , "pubsub\_perms" : { "listen" : [ "broadcast" , "whisper-UG12X345T6J78" ], "send" : [ "broadcast" , "whisper-\*" ] } } Provide feedback for this page Docs Status Support Showcase Blog

What We Do Stream Watch

Develop Advertise twitch.tv Company Jobs Merch Brand TwitchCon Meetups Newsroom News Press Products Bits Subs Turbo Prime Extensions Sings Resources Legal Help Center Security Connect Twitter Facebook Instagram Terms of service Privacy Policy Ad Choices Cookie Policy Partners Affiliates

© 2023 Twitch Interactive, Inc.

×

Let’s start building on Twitch!

Click below to login.