

[Docs](#) [Marketing API](#) [Overview](#) [Versioning](#)

On This Page

# Versioning

The current version of the Marketing API is **v23.0**.

Facebook's Platform has a core and extended [versioning](#) model. With Marketing API versioning, all breaking changes will be in a new version. Multiple versions of Marketing APIs or SDKs can exist at the same time with different functionality in each version.

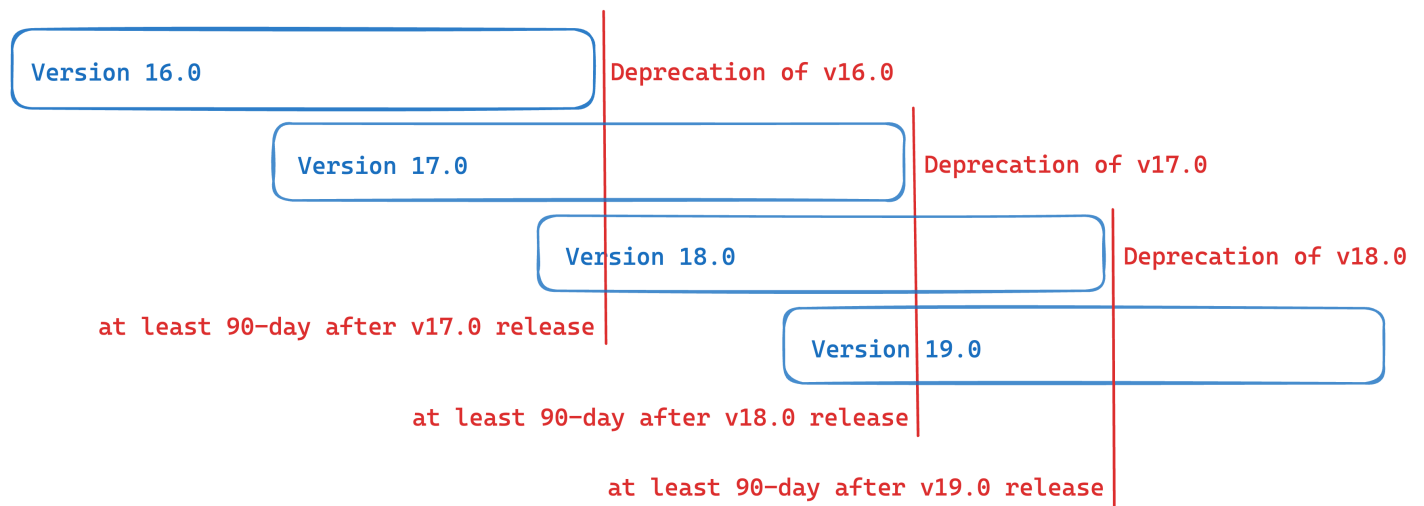
Developers should understand in advance when a Marketing API or SDK will change. While you have a 90-day window to adopt changes, how and when to move to the new version is your choice.

## Version Schedules

When a new version of the Marketing API releases, we continue to support the previous version of the Marketing API for at least 90-days. You have at least a 90-days grace period to move over to the new version. During the 90-days grace period, you can call both the current version and the deprecated version, and you have that 90-days grace period to move to the new version. After the 90-days grace period ends, the deprecated version stops working. Once a version is unavailable, any calls made to that version number may fail or be upgraded to the next available version.

For example, Marketing API v17.0 was released on May 23rd, 2023, and Marketing API v16.0 expired on February 6th, 2024, which provided at least 90 days to move over to the new version.

Here is a sample timeline. Note that we may not always release a new version at the end of the 90-days grace period of the previous version's deprecation. In the example, v16.0 was deprecated some time before v18.0 was released:



For SDKs, a version always remains available as it is a downloadable package, however beyond its end-of-life date, it may rely upon Marketing APIs or methods which no longer work, so you should assume an end-of-life SDK is no longer functional.

## Making Versioned Requests

All Marketing API endpoints are available through a versioned path. Pre-pend the version to the start of the request path. For example:

```
curl -G \  
-d "access_token=<ACCESS_TOKEN>" \  
"https://graph.facebook.com/v23.0/me/adaccounts"
```

This works for all versions, in this general form:

```
https://graph.facebook.com/v{n}/{request-path}
```

where `n` is the version needed. See a full list of available versions in our [Changelog](#). All of our [Marketing API Reference](#) provide per-version information.

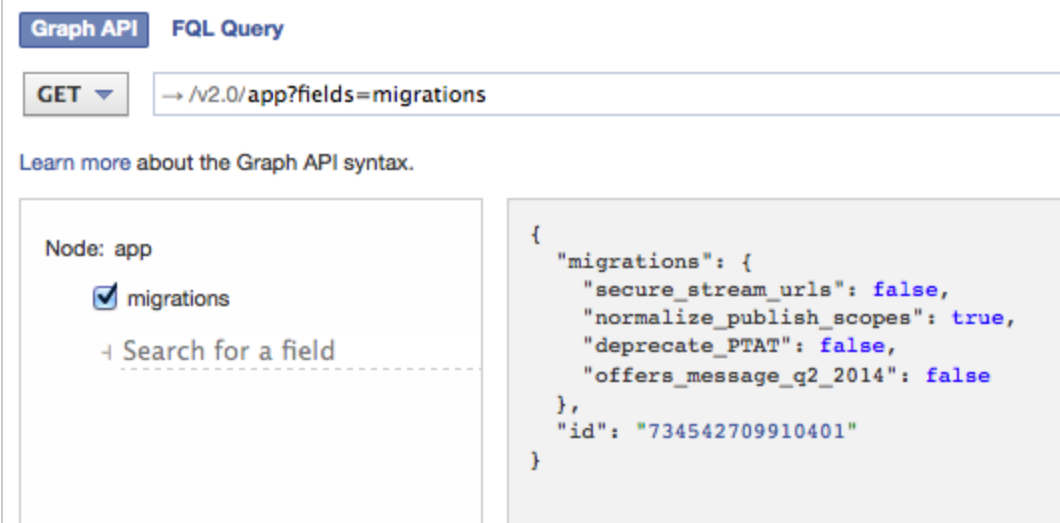
## Migrations

Migrations are only for special scenarios where changes need to be made that cannot go into versioning. Typically this is if the underlying data model has changed. Migrations apply across all versions.

Migrations that are currently still in progress are listed on our [migrations page](#). Migrations have at least a 90-day window during which you must migrate your app. Once a window begins, the post-migration behavior will become the default for new apps. Then, when the migration window is completed, the pre-migration behavior will no longer be available at all.

## Manage Migrations via Graph API

Migrations can be managed via the [migrations](#) field in the `/app` node:



The screenshot shows the Facebook Graph API Explorer interface. At the top, there are tabs for "Graph API" (selected) and "FQL Query". Below the tabs, there is a "GET" method selector and a URL input field containing `→ /v2.0/app?fields=migrations`. A link "Learn more about the Graph API syntax." is visible. The interface is split into two main sections. The left section, titled "Node: app", shows a tree view where the "migrations" field is selected with a checkbox. Below this is a search bar labeled "Search for a field". The right section displays the JSON response for the query:

```
{
  "migrations": {
    "secure_stream_urls": false,
    "normalize_publish_scopes": true,
    "deprecate_PTAT": false,
    "offers_message_q2_2014": false
  },
  "id": "734542709910401"
}
```

You can [make an update call on the edge](#) to activate and deactivate migrations.

## Manage Migrations via App Dashboard

You can activate and deactivate available migrations in the [App Dashboard](#) under **Settings > Migrations**. Please note, that the list of migrations may not be the same as in the image below, as the available migrations are different for different apps, at different time. And if you see a migration [Use Graph API v2.0 by default](#), it is for Graph API only, not Marketing API.

Example iOS App ▾

Dashboard
Settings
Status & Review
App Details
Roles
Open Graph
Alerts
Localize
Payments
Test Apps
Insights

Basic | Advanced | Migrations

OFF

Stream post URL security

Links in stream posts must go to app's Canvas or Website URL.

ON

Normalize all publish scopes to 'publish\_actions'

All permissions that allow publishing on behalf of the user will be normalized to a single permission: publish\_actions.

OFF

Deprecate PTAT (People Talking About This)

Some PTAT metrics (page\_story\_adds\_unique, page\_storytellers, page\_story\_adds\_by\_story\_type\_unique, page\_story\_adds\_by\_locale\_unique) will be removed from the insights API

OFF

Offers Message field will be required after 7/2/2014

Offers Message field will be required after 7/2/2014

Discard

Save Changes

## Temporary Client-side Activation of Migrations

Instead of activating the migration in your App Dashboard or via the Marketing API, it's possible to add a special flag to your Marketing API calls that sets the migration. The flag is called `migrations_override` and requires you to specify a JSON blob that describes what migrations you want to turn on or off. For example, if you were making a raw call you could pass:

```
http://graph.facebook.com/path?
  migrations_override={"migration1":true, "migration2":false}
```

Using this, you can call the new Marketing API through client updates instead of having to get all callers to update to calling the new Marketing API at the same time. It's also very useful for debugging.

The names for these migrations are found on the [/app node mentioned above](#).

## Version auto-upgrade

Given the rapid rotation of Marketing API versions approximately every four months, we are streamlining the upgrade process. Starting May 2024, we will enable the auto-version upgrade feature for Marketing

API endpoints that are not affected between versions. This means between a version to be deprecated and the next available version, if an endpoint is not affected, the platform will upgrade the call to the next available version, rather than directly failing the request. This change is designed to ensure a smoother, more efficient API experience.

For example, on May 14, 2024, v17.0 will be deprecated. According to the [changelog of v18.0](#), the following endpoints will be affected:

- `POST /act_{ad-account-id}/reachfrequencypredictions`
- `GET /act_{ad-account-id}/reachestimate`
- `GET /act_{ad-account-id}/delivery_estimate`
- `POST /act_{ad-account-id}/adsets`
- `POST /{adset-id}`
- `POST /act_{ad-account-id}/saved_audiences`
- `POST /{saved-audience-id}`
- `POST /act_{ad-account-id}/credit_cards`

If your app is calling `POST /{adset-id}` with v17.0 after it was deprecated on May 14, 2024, this API request will fail as the auto-upgrade is not applied to endpoints affected by next available version (v18.0).

If your app is calling `GET /{ad-account-id}/insights` with v17.0 after deprecation, the platform will upgrade your call to the next available version (v18.0).

**Note:** If your app is already making calls with versions higher than v17.0, nothing should have changed on the version deprecation date.

To check endpoints affected at each version, please refer to the [Marketing API Changelog](#).

## FAQ

### Version Schedules

---

What if I don't specify a version for the Marketing API?

---

Can I make calls to versions older than the current version?

---

How is this different from Platform API versioning?

## Making Versioned Requests

---

How is this different than migrations?

## Version auto-upgrade

---

Does the upgrade only apply to the version to be deprecated and the next available version?

---

Does this mean developers don't need to do anything during version deprecation?

---

How can I find out which endpoints will not be auto-upgraded?

---

How can I opt-out of this behavior?

---

Can I check if any specific API call has been auto-upgraded?

## Marketing API

### Overview

#### Versioning

#### Rate Limiting

#### Data Processing Options

#### Get Started

#### Ad Creative

#### Bidding

#### Ad Rules Engine

#### Audiences

#### Insights API

#### Brand Safety and Suitability

#### Best Practices

#### Troubleshooting

#### API Reference

#### Changelog