



Nesta Página

# Solicitações em lote

Envie uma única solicitação HTTP com múltiplas chamadas da Graph API do Facebook. As operações independentes são processadas em paralelo, enquanto as operações dependentes são processadas em sequência. Quando todas as operações forem concluídas, uma resposta consolidada será enviada de volta a você e a conexão HTTP será encerrada.

O ordenamento das respostas corresponde à ordem das operações na solicitação. Processe as respostas de acordo com a ordem para determinar quais operações foram bem-sucedidas e quais deverão ser repetidas em uma operação subsequente.

## Limitações

- Há um limite de 50 solicitações por lote. Cada chamada dentro do lote será contada separadamente para fins de cálculo dos limites de chamada de API e de recurso. Por exemplo, um lote com 10 chamadas de API será contabilizado como 10 chamadas, e cada chamada dentro do lote contribuirá com os limites de recurso de CPU da mesma maneira. Para ver mais informações, consulte nosso [Guia de limitação de volume](#).
- As solicitações em lote não podem incluir vários [conjuntos de anúncios](#) na mesma [campanha](#). Saiba mais sobre [como fazer solicitações da API de Marketing em lote](#).

## Solicitação em lote

Uma solicitação em lote usa um objeto JSON contendo uma matriz das solicitações. Ela retorna uma matriz de respostas HTTP lógicas representadas como matrizes JSON. Cada resposta tem um código de status, uma matriz de cabeçalhos opcionais e um corpo opcional (que é uma string codificada em JSON).

Para fazer uma solicitação em lote, envie uma solicitação **POST** a um ponto de extremidade em que o parâmetro **batch** é seu objeto JSON.

```
POST /ENDPOINT?batch=[JSON-OBJECT]
```



Neste exemplo, obteremos informações sobre duas Páginas gerenciadas pelo nosso aplicativo.

*Formatado para leitura*

```
curl -i -X POST 'https://graph.facebook.com/me?batch=
[
  {
    "method":"GET",
    "relative_url":"PAGE-A-ID"
  },
  {
    "method":"GET",
    "relative_url":"PAGE-B-ID"
  }
]
&include_headers=false           // Included to remove header information
&access_token=ACCESS-TOKEN'
```

Depois que todas as operações forem concluídas, uma resposta será enviada com o resultado de cada uma. Como os cabeçalhos retornados podem ser muito maiores do que a resposta real da API, convém removê-los para manter a eficiência. Para incluir as informações do cabeçalho, remova o parâmetro `include_headers` ou o defina como `true`.

### Exemplo de resposta

O campo "body" contém um objeto JSON codificado com string:

```
[
  {
    "code": 200,
    "body": "{
      \"name\": \"Page A Name\",
      \"id\": \"PAGE-A-ID\"
    }"
  },
  {
    "code": 200,
    "body": "{
      \"name\": \"Page B Name\",
      \"id\": \"PAGE-B-ID\"
    }"
  }
]
```



## Solicitações em lote complexas

É possível combinar operações que normalmente usariam métodos HTTP diferentes em uma única solicitação em lote. Embora as operações **GET** e **DELETE** possam ter apenas um campo **relative\_url** e um campo **method**, as operações **POST** e **PUT** podem conter um campo **body** opcional. O corpo deve ser formatado como uma string HTTP POST bruta semelhante a uma string de consulta de URL.

### Exemplo de solicitação

Em uma única operação, o exemplo a seguir faz uma publicação em uma Página que gerenciamos e na qual temos permissões de publicação e, em seguida, no feed da Página:

```
curl "https://graph.facebook.com/PAGE-ID?batch=[
  {
    "method":"POST",
    "relative_url":"PAGE-ID/feed",
    "body":"message=Test status update"
  },
  {
    "method":"GET",
    "relative_url":"PAGE-ID/feed"
  }
]&access_token=ACCESS-TOKEN"
```

O resultado dessa chamada seria o seguinte:

```
[
  { "code": 200,
    "headers": [
      { "name": "Content-Type",
        "value": "text/javascript; charset=UTF-8" }
    ],
    "body": "{\\"id\\":\\"...\\"}"
  },
  { "code": 200,
    "headers": [
      { "name": "Content-Type",
        "value": "text/javascript; charset=UTF-8" }
```



```

    }
  ],
  "body": "{\\"data\\": [{...}]}"
}
]

```

O exemplo a seguir cria um anúncio para uma campanha e, em seguida, obtém os detalhes do objeto recém-criado. Observe a **URLEncoding** do parâmetro "body":

```

curl \
-F 'access_token=...' \
-F 'batch=[
  {
    "method":"POST",
    "name":"create-ad",
    "relative_url":"11077200629332/ads",
    "body":"ads=%5B%7B%22name%22%3A%22test_ad%22%2C%22billing_entity_id%22%3A1
  },
  {
    "method":"GET",
    "relative_url":"?ids={result=create-ad:$.data.*.id}"
  }
]' \
https://graph.facebook.com

```

O exemplo a seguir adiciona várias páginas a um Gerenciador de Negócios:

```

curl \
-F 'access_token=<ACCESS_TOKEN>' \
-F 'batch=[
  {
    "method":"POST",
    "name":"test1",
    "relative_url":"<BUSINESS_ID>/owned_pages",
    "body":"page_id=<PAGE_ID_1>"
  },
  {
    "method":"POST",
    "name":"test2",
    "relative_url":"<BUSINESS_ID>/owned_pages",
    "body":"page_id=<PAGE_ID_2>"
  },
  {
    "method":"POST",

```



```
},  
]' \\  
"https://graph.facebook.com/v12.0"
```

Em que:

- `<ACCESS_TOKEN>` é um token de acesso com a permissão `business_management`.
- `<BUSINESS_ID>` é o ID do Gerenciador de Negócios ao qual as páginas devem ser obtidas.
- `<PAGE_ID_n>` são as identificações das Páginas que serão obtidas.

## Erros

É possível que uma das operações solicitadas resulte em erro. Isso pode ocorrer por você não ter permissão para executar a operação solicitada, por exemplo. A resposta será semelhante à padrão da Graph API, porém encapsulada na sintaxe de resposta em lote:

```
[  
  { "code": 403,  
    "headers": [  
      { "name": "WWW-Authenticate", "value": "OAuth...",  
        { "name": "Content-Type", "value": "text/javascript; charset=UTF-8" } ],  
      "body": "{ \"error\": { \"type\": \"OAuthException\", ... } }"  
    }  
  ]
```

As outras solicitações no lote deverão ser concluídas e retornadas normalmente com o código de status `200`.

## Tempos-limite

Lotes grandes ou complexos poderão atingir o tempo-limite caso a operação demore demais para concluir todas as solicitações. Nesse caso, o resultado será um lote concluído parcialmente. Nos lotes concluídos parcialmente, as solicitações bem-sucedidas retornarão a resposta-padrão com o código de status `200`. As solicitações não concluídas retornarão a resposta `null`. É possível refazer todas as solicitações malsucedidas.



A API em lote é compatível com JSONP, assim como o restante da Graph API. A função de retorno de chamada JSONP é especificada usando o parâmetro post de formulário ou de string de consulta `callback`.

## Como usar diversos tokens de acesso

As solicitações individuais de uma solicitação em lote única podem especificar os próprios tokens de acesso como uma string de consulta ou um parâmetro post de formulário. Nesse caso, o token de acesso de nível superior é considerado um token de fallback que será usado caso uma solicitação individual não especifique um token de acesso de forma explícita.

Isso poderá ser útil se você quiser consultar a API usando diversos tokens de acesso de usuário ou de Página ou caso algumas das chamadas precisem ser feitas com um token de acesso do aplicativo.

Você precisará incluir um token de acesso como parâmetro de nível superior mesmo quando todas as solicitações individuais tiverem os próprios tokens.

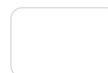
## Carregar dados binários

É possível carregar diversos itens binários como parte de uma chamada em lote. Para isso, você precisa adicionar todos os itens binários como anexos multipart/mime à solicitação. Além disso, cada operação precisa referenciar os itens binários por meio da propriedade `attached_files`. A propriedade `attached_files` pode receber uma lista de nomes de anexos separados por vírgula no próprio valor.

O exemplo a seguir mostra como carregar duas fotos em uma única chamada em lote:

```
curl
  -F 'access_token=...' \
  -F 'batch=[{"method":"POST","relative_url":"me/photos","body":{"message=My
  -F 'file1=@cat.gif' \
  -F 'file2=@dog.jpg' \
  https://graph.facebook.com
```

-->

[Meus apps](#)[Ações necessárias](#)[Mais](#)[Docu](#)

## Graph API

[Visão geral](#)[Status, Support, and Tools](#)[Introdução](#)[Solicitações em lote](#)[Solicitações de depuração](#)[Solução de erros](#)[Field Expansion](#)[Secure Requests](#)[Carregar um arquivo ou vídeo](#)[Registro de alterações](#)[Referência de recursos](#)[Permissions Reference](#)[Reference](#)