Annexe à la synthèse concernant l'accès aux données pour la recherche

Tour d'horizon des API des plateformes. Juin 2023

La présente annexe propose un tour d'horizon sommaire, à date, des possibilités offertes, principalement dans la perspective d'études, par les interfaces de programmation d'application (API[1]) mises publiquement à disposition[2] par les principaux médias sociaux utilisés en France.

Après avoir présenté la méthodologie, nous reviendrons sur certains principes généraux régissant le fonctionnement d'une API, pour ensuite présenter une typologie des plateformes selon ce que leurs API permettent en matière de récupération de données. Nous aborderons ensuite certaines questions techniques et pratiques se posant dans la prise en main des API, pour enfin récapituler quelques enseignements en conclusion.

Table des matières

METHODOLOGIE

Ce tour d'horizon a été établi à partir de la seule consultation de la documentation publique d'une douzaine de services numériques, sans expérimentation technique directe. **Cette méthode présente certaines limites : les documentations sont en effet parfois imprécises ou incomplètes**. En particulier, dans de nombreux cas, si ce n'est la plupart, la documentation ne précise pas les quotas temporels que l'API accorde, autrement dit le nombre de requêtes qu'il est possible de soumettre par période de temps donné. Bien

[1] Sigle usuel pour l'anglais « *Application Programming Interface* ».

[2] Bien qu'elles soient limitées dans leur portée et que leur usage soit soumis à de nombreuses conditions restrictives.

souvent, ces quotas sont gérés de manière dynamique, et seule la pratique permet de les mettre à l'épreuve – ce qui peut être gênant si on cherche à se faire une idée des volumes de données qu'il sera possible d'extraire de la plateforme lors de la conception d'un projet de recherche. Par ailleurs, les conditions contractuelles d'utilisation de ces API n'ont pas été analysées en détail pour cette annexe.

FONDAMENTAUX DES API ET DES SERVICES CONCERNES

API : UTILITES ET FONCTIONNEMENT

Une **interface de programmation d'application** permet d'établir une communication entre deux programmes informatiques, s'échangeant des informations selon un format prédéfini et constant. Les API des plateformes en ligne accordent ainsi l'accès à un certain nombre de ressources y étant présentes, de manière structurée. Ceci permet la création de programmes capables d'interagir avec les différentes fonctionnalités des plateformes[3], et peut être utilisé pour explorer les données qui s'y trouvent, y compris en temps réel – ce qui peut intéresser la recherche, la régulation et plus généralement tous les acteurs soucieux de mieux comprendre le fonctionnement et les effets des services numériques.

Pour autant, les API proposées par les plateformes présentent vis-à-vis de cet objectif de nombreuses limites : en effet, la plupart sont avant tout conçues à destination d'acteurs commerciaux, afin de faciliter la gestion de leur présence en ligne. Peu sont spécifiquement conçues pour la recherche : à date, **Crowdtangle**[4], permettant d'accéder à certaines données sur Facebook, Instagram et Reddit, conçu avant tout pour l'écoute du web, et **YouTube**[5], proposent un accès académique à leur API. **Twitter**, qui disposait d'une API dédiée aux chercheurs académiques, l'a suspendue *sine die*[6]. **TikTok**[7] et **Facebook**[8] proposent une API dédiée aux chercheurs, néanmoins celle-ci n'est pour l'heure accessible qu'à un nombre limité d'utilisateurs.

Les API fonctionnent presque toujours avec un **système d'identification et d'autorisations**, souvent complexe : un accès à l'API sera accordé sur une base individuelle, avec un ensemble défini et possiblement variable de permissions, définissant ce à quoi il est possible ou non d'accéder. Un accès à l'API peut venir avec certaines autorisations par défaut, des autorisations supplémentaires pouvant souvent être obtenues sur demande auprès de la plateforme ou auprès des utilisateurs : par exemple, il est possible de solliciter l'autorisation d'un utilisateur pour avoir accès à certaines de ses informations personnelles, ou pour agir en son nom. Un tel système vise surtout, à l'origine, à permettre une offre de services complémentaires – comme des robots, des applications, ou autres services tiers – enrichissant l'expérience de l'utilisateur sur la plateforme ou en interaction avec celle-ci.

[3] Par exemple, elles permettent dans un cas comme celui d'une diffusion Twitch d'afficher à l'écran, en direct, le nom des spectateurs qui effectuent des dons.

[4] https://help.crowdtangle.com/en/articles/4302208-crowdtangle-for-academics-and-researchers

[5] https://research.youtube/

[6] https://twitter.com/TwitterDev/status/1641222788911624192

[7] https://www.tiktok.com/transparency/en-us/research-api/

[8] https://developers.facebook.com/docs/fort-pages-api/overview

DES DONNEES PAR NATURE PLUS OU MOINS PUBLIQUES

Si on veut brosser à grands traits les différents types d'API proposées par les plateformes, on doit d'abord prêter attention à la nature variable des différents services : en effet, un service de messagerie privée et une encyclopédie libre ne proposeront, comme on peut s'y attendre, pas le même degré d'accès aux données échangées via le service. On peut ainsi **distinguer les espaces virtuels selon leur caractère plus ou moins publics**, en gardant à l'esprit qu'une même plateforme peut proposer des espaces de différentes natures. Au demeurant et comme nous allons le voir par la suite, dans de nombreux cas, les API ne permettent pas d'accéder à l'ensemble des informations pourtant accessibles à tous publiquement sur la plateforme.

À une extrémité, on a donc des informations accessibles au seul utilisateur – par exemple son carnet d'adresses – et de l'autre des publications accessibles à tous indéfiniment – qui peuvent néanmoins être plus ou moins visibles, et plus ou moins vues. Entre les deux existe tout un gradient de possibilités, de la conversation privée bilatérale aux forums de discussion publics comptant des milliers d'utilisateurs, en passant par des discussions collectives.

Prenons ci-dessous quelques exemples.

Illustration 1 – Captures d'écran de différents espaces virtuels plus ou moins publics[9]

À une extrémité, on trouve les **conversations privées** comme celles présentes sur la messagerie WhatsApp – d'autant moins accessibles que les échanges y sont chiffrés[10]. Celles-ci peuvent néanmoins s'étendre de deux utilisateurs à bien davantage (la plupart

[9] https://blog.whatsapp.com/one-whatsapp-account-now-across-multiple-phones ; https://www.facebook.com/groups/822324281300446/ ; https://discord.com/guild-discovery ; https://twitter.com/elonmusk/status/1659419397327339521 ; https://www.youtube.com/watch?v=uzOXfvVipvs

[10] Ainsi, les messages ne sont pas chiffrés de bout en bout sur Messenger, ils le sont sur WhatsApp, Telegram ou encore Signal.

des services de messagerie proposant des conversations de groupe). Discord par exemple se présente comme une messagerie instantanée, mais sa fonctionnalité fondatrice sont les serveurs collectifs de discussion, qui peuvent aller du groupe confidentiel de quelques amis à la communauté publique très large – comme le serveur dédié à Midjourney, qui compte des millions de membres.

Pour les **conversations collectives**, tout comme pour les groupes collectifs s'approchant davantage de forums, ils peuvent d'une part être répertoriés dans un annuaire public ou partagés uniquement par liens d'invitation personnelle, et d'autre part autoriser la lecture de leur contenu à tout à chacun ou uniquement aux membres ayant rejoint le groupe en ayant été approuvés par les administrateurs[11]. C'est par exemple le cas pour les groupes Facebook, qui peuvent être privés ou publics. Ces facteurs sont donc de nature à influer sur la visibilité des contenus.

Enfin, certains contenus sont accessibles à tous sans restriction, comme la plupart des publications sur Twitter ou sur YouTube – dont les réactions et les commentaires sont également publics[12]. **Pour autant, comme précédemment souligné, le caractère public des contenus ne présume pas leur accessibilité par API**.

TYPOLOGIE DES PLATEFORMES SELON LEURS API

La logique de fonctionnement de chaque API reflète globalement les particularités de chaque plateforme. On peut ainsi distinguer globalement plusieurs familles de plateformes selon ce que leurs API permettent en matière de récupération de données :

- Les plateformes dont les contenus ne sont accessibles par API **que sur autorisation des administrateurs** des espaces, comme WhatsApp[13], Messenger[14], et Discord[15]. Il est en effet possible d'ajouter des robots dans sa conversation de groupe, et celui-ci aura accès à certains des contenus y étant présents[16].

- Les plateformes dont les contenus publics **ne sont pas accessibles** par API, comme Pinterest ou Snapchat. Sur Snapchat[17], les fonctionnalités Map et Spotlight proposent des contenus publics, mais ceux-ci ne sont pas accessibles par API. Les API existantes ne permettent que de gérer son propre contenu. De même pour Pinterest[18], où s'ajoutent néanmoins les tableaux partagés.

[11] Les modalités d'approbation et le degré de contrôle peuvent être variés.

[12] Lorsqu'ils existent, les commentaires par exemple pouvant souvent être désactivés. Certaines restrictions existent par service : par exemple sur Twitter, il faut être connecté pour pouvoir voir la liste des gens ayant aimé et repartagé une publication.

[13] https://developers.facebook.com/docs/whatsapp

[14] https://developers.facebook.com/docs/messenger-platform

[15] https://discord.com/developers/docs/intro

[16] Selon une granularité d'autorisations plus ou moins fine, particulièrement développée sur Discord.

[17] https://developers.snap.com/

[18] https://developers.pinterest.com/docs/api/v5/

- Les plateformes dont les contenus publics sont **partiellement accessibles** par API, comme Facebook[19] et Instagram[20] – directement ou via Crowdtangle[21] et TikTok[22]. En effet, il est par exemple possible d'accéder aux pages publiques sur Facebook, mais les commentaires sont alors intégralement anonymisés. Les groupes publics ne sont accessibles que via Crowdtangle. À noter que l'API dédiée à la recherche pour les pages Facebook[23] offre davantage de fonctionnalités, mais est pour l'heure limitée à un nombre d'utilisateurs choisis. De plus, davantage d'information sont disponibles sur autorisation des administrateurs d'une page ou d'un groupe, dans la même logique que pour les conversations collectives[24]. Concernant TikTok, l'API de recherche[25] permet de récupérer les vidéos publiques et leurs commentaires anonymisés, néanmoins cette API n'est pour l'heure disponible qu'aux États-Unis.
- Les plateformes dont les contenus publics étaient **globalement accessibles jusqu'à récemment** mais se sont vues imposer de nouvelles restrictions tarifaires, comme Twitter et dans une moindre mesure Reddit. Concernant Twitter[26], il était possible jusqu'il y a peu[27] d'avoir accès à de nombreux contenus – ayant jusqu'à une semaine d'ancienneté pour un accès standard, et sans restriction chronologique pour les chercheurs. Désormais, ne sont accessibles gratuitement que les opérations en écriture, et aucune opération en lecture, y compris pour des chercheurs académiques – il reste néanmoins possible d'utiliser l'API moyennant finance. Concernant Reddit[28], les contenus discussions publiques sont globalement largement accessibles, néanmoins il a récemment été annoncé[29] que les utilisations au-delà d'un certain quota seront facturées.
- Les plateformes dont les contenus publics sont **globalement accessibles**, comme YouTube[30] ou Twitch[31]. Concernant YouTube, il est par exemple possible de récupérer les commentaires, mais pas les fichiers vidéos eux-mêmes. Concernant Twitch, ce sont avant tout les contenus en direct qui sont accessibles.

Il convient toutefois de noter que même lorsque les plateformes permettent d'accéder à certains contenus par API, **le nombre de données récupérables, en une période de temps donnée, est limité**. Par ailleurs, le détail de certaines données est souvent limité,

[19] https://developers.facebook.com/docs/graph-api/

[20] https://developers.facebook.com/docs/instagram-api/

[21] https://help.crowdtangle.com/en/articles/3443476-api-cheat-sheet

[22] https://developers.tiktok.com/doc/overview/

[23] https://developers.facebook.com/docs/fort-pages-api/overview

[24] Avec certaines restrictions : par exemple, pour les groupes, seuls les contenus postérieurs à l'ajout dans le groupe sont disponibles, ou ceux postés il y a moins de trois mois si l'ajout dans le groupe est antérieur à cette durée.

[25] https://www.tiktok.com/transparency/en-us/research-api/

[26] https://developer.twitter.com/en/docs/twitter-api

[27] https://twitter.com/TwitterDev/status/1649191521323995138

[28] https://www.reddit.com/dev/api

[29] https://www.reddit.com/r/reddit/comments/12qwagm/an_update_regarding_reddits_api/

[30] https://developers.google.com/youtube/v3/getting-started

[31] https://dev.twitch.tv/docs/api/

et **cela permet difficilement de retracer l'expérience effective** de la plateforme. Par exemple, les fils d'actualité et les recommandations ne sont presque jamais[32] accessibles avec les API, même si certaines permettent de récupérer les contenus en tendances au moment de la requête, par exemple sur Instagram[33] ou Twitch[34]. Ceci ne permet donc pas d'analyser précisément le fonctionnement des algorithmes de recommandation. On peut également relever que certaines entreprises, comme Meta[35], mettent à disposition, pour des projets de recherche, des jeux de données constitués, sans que ceci ne passe par l'API de leur service.

ASPECTS TECHNIQUES ET PRATIQUES

Sous un aspect davantage technique et pratique, les différentes API partagent certains grands traits communs, néanmoins elles sont chacune singulière dans leur architecture sous-jacente et dans leur présentation documentaire – hétérogénéité qui peut entraver leur prise en main.

DES ASPECTS COMMUNS MAIS DES ARCHITECTURES DIFFERENTES

La plupart des API considérées emploient les **mêmes protocoles techniques** de communication : presque systématiquement[36], la communication est établie via le protocole de transfert hypertexte (HTTP[37]), et les données sont échangées dans la notation d'objet JavaScript (JSON[38]). Dans de nombreux cas, des bibliothèques de fonctions[39] sont disponibles dans plusieurs langages de programmation pour rendre la manipulation de l'API plus aisée – ces bibliothèques peuvent être fournies par la plateforme elle-même[40] ou développées par des tiers[41], ce qui au demeurant ne change rien dans l'usage qu'on peut

[32] Pour Twitter, il est possible d'obtenir le fil d'actualité sous sa forme chronologique : https://developer.twitter.com/en/docs/twitter-api/tweets/timelines/introduction

[33] Il est possible de récupérer les médias les plus populaires sur un mot-clic (*hashtag*) donné. https://developers.facebook.com/docs/instagram-api/reference/ig-hashtag/top-media

[34] Il est possible de récupérer les jeux et catégories de diffusions les plus populaires. https://dev.twitch.tv/docs/api/reference/#get-top-games

[35] https://fort.fb.com/researcher-datasets

[36] Par exemple, on relève une exception pour accéder aux tchats sur une diffusion Twitch : le protocole utilisé est alors celui de la discussion relayée par internet (IRC, sigle de « *Internet Relay Chat* »). https://dev.twitch.tv/docs/irc/

[37] Sigle usuel pour l'anglais « *HyperText Transmission Protocol* ».

[38] Sigle usuel pour l'anglais « *JavaScript Object Notation* ». Le JavaScript est un langage de programmation.

[39] Aussi appelé trousse de développement logiciel ou devkit (SDK, sigle de « *Software Development Kit* »).

[40] Par exemple, l'API de YouTube est rendue accessible via des bibliothèques en Java, en JavaScript, en .NET, en Objective-C, en Python, en version bêta pour PHP, Dart, ainsi qu'en version alpha pour Go, Node.js et Ruby. https://developers.google.com/youtube/v3/libraries

[41] Par exemple, l'API Discord est rendue accessible en Node.js par la communauté de développeurs de Discord.js, et en Python par la communauté Discord.py. https://discord.js.org/ ; https://discordpy.readthedocs.io/

en faire. On peut relever que les API sont aussi parfois disponibles sur des services tiers de gestion d'API, comme Postman, sur lequel on trouve par exemple l'API de WhatsApp[42].

Néanmoins, **passé ces fondamentaux communs, chaque plateforme dispose de sa propre architecture**, ce qui se reflète dans ses API. En effet, les objets, les schémas d'objets, et l'organisation des permissions, sont propres à chaque plateforme.

Pour illustrer brièvement cette idée, prenons un exemple simple : comparons la manière de récupérer le nom de l'auteur d'une vidéo sur quatre plateformes différentes. Soit « MON_ID » l'identifiant de la vidéo qu'on cible, et « RESULTAT » l'objet renvoyé dans la réponse de l'API. La requête est similaire à un hyperlien usuel[43], et ce que nous appelons chemin est le parcours en cascade qu'il faut effectuer sur les propriétés de l'objet renvoyé[44] pour trouver le nom de l'auteur.

YouTube[45]	Requête : `https://youtube.googleapis.com/youtube/v3/videos?id=MON_ID` Chemin : `RESULTAT.items[0].snippet.channelTitle`
Twitch[46]	Requête : `https://api.twitch.tv/helix/videos?id=MON_ID` Chemin : `RESULTAT.data[0].user_name`
Dailymotion[47]	Requête : `https://api.dailymotion.com/video/MON_ID` Chemin : `RESULTAT.owner.fullname`
Facebook[48]	Requête : `https://graph.facebook.com/v17.0/MON_ID` Chemin : `RESULTAT.from.name`

Tableau 1 – Requête à formuler et chemin à parcourir pour obtenir le nom de l'auteur d'une vidéo

On voit ainsi qu'à la fois la structure des requêtes et celles des résultats varie entre chaque plateforme. Cela demande donc, pour trouver les informations que l'on cherche, de développer une expertise particulière sur chacune des plateformes que l'on souhaite investiguer.

[42] https://www.postman.com/meta/workspace/whatsapp-business-platform/

[43] En vérité, la requête doit comprendre de nombreux autres paramètres. Ce qui est retenu ici est disons le cœur de la requête.

[44] En programmation informatique, un objet peut posséder différentes propriétés. Une propriété peut être elle-même un objet (ou une collection de plusieurs objets). On pourra donc de nouveau interroger les propriétés de l'objet enfant de l'objet de départ. Chaque relation parent-enfant est séparée par un point, de gauche à droite. Le [0] signifie que l'on sélectionne le premier objet d'une collection de plusieurs objets. Ainsi « RESULTAT.data[0].user_name » signifie que l'on cherche la propriété « user_name » du premier objet de la collection « data » de l'objet « RESULTAT ».

[45] https://developers.google.com/youtube/v3/docs/videos/list#response

[46] https://dev.twitch.tv/docs/api/reference/#get-videos

[47] https://developers.dailymotion.com/api/#video-fields

[48] https://web.archive.org/web/20211019021055/https://developers.facebook.com/docs/graph-api/reference/video

PRENDRE EN MAIN LES API : POINTS D'ENTREE ET DOCUMENTATION

Ce phénomène d'hétérogénéité est renforcé par la différence en matière de périmètres d'accès et de permissions. Une enquête sur une plateforme demande en effet de **trouver des points d'entrées adéquats** à partir desquels explorer la plateforme dans un périmètre autorisé – et ce y compris pour les contenus publics, dans la mesure où il est impossible de récupérer l'intégralité des données des plateformes pour ensuite les étudier à souhait. À cet égard, **les plateformes proposent des points d'entrée divers, qui sont parfois insatisfaisants**, les paramètres de recherche accessibles par API pouvant notamment être moindres que ceux disponibles directement sur le service lui-même. Par exemple, l'API de Facebook permet de trouver des pages par mot-clé présent dans le titre[49], mais pas de trouver des publications, postées sur une page, en fonction du texte qu'elles contiennent – ceci n'est possible que dans la version de l'API dédiée à la recherche[50] ou via CrowdTangle[51]. L'API de Twitter de son côté propose entre autres un échantillon aléatoire de publications postées en direct[52] – fonctionnalité qui peut s'avérer utile et qui n'est pas proposée par les autres API.

Une fois qu'un point d'entrée a été trouvé, il est possible de passer d'objets en objets en explorant les liens qui existent entre eux, en suivant une logique illustrée dans le schéma suivant – on suppose alors une plateforme basique composé de groupes et d'utilisateurs, chacun doté de propriétés et de collections d'objets. La logique est la suivante : lorsqu'on récupère le groupe A, on est en capacité de récupérer ses membres, à savoir les utilisateurs 1 et 2, et ainsi de suite de proche en proche.

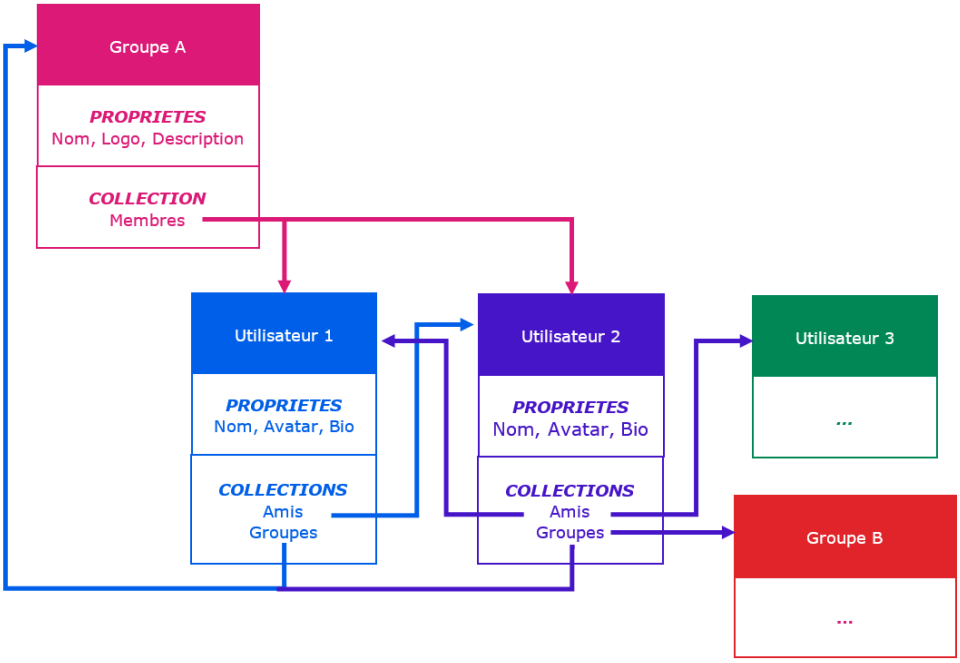

Illustration 2 – Schéma explicitant la logique d'exploration à mettre en œuvre pour parcourir des données via une API

[49] https://developers.facebook.com/docs/pages/searching

[50] https://developers.facebook.com/docs/fort-pages-api/

[51] https://github.com/CrowdTangle/API/wiki/Search

[52] https://developer.twitter.com/en/docs/twitter-api/tweets/volume-streams/introduction

En conséquence de ces disparités, mener une enquête sur une plateforme demande de se familiariser avec son fonctionnement et sa structure, rendant plus fastidieux les études comparatives. Au demeurant, cette diversité reflète le fait que les plateformes ne reposent pas sur un standard commun qui permettrait facilement de les rendre interopérables.

Le caractère fastidieux de cette hétérogénéité est par ailleurs renforcé par le fait que chaque **documentation se présente elle aussi dans une mise en page propre, et selon son propre formalisme**. Si les documentations consultées sont généralement rigoureuses et bien organisées, avec de nombreux hyperliens permettant d'y naviguer, plusieurs présentent des fioritures parfois mineures pouvant heurter la navigation, de même que, de temps à autre, des imprécisions si ce n'est des contradictions. Les documentations proposées par Meta notamment renvoient l'impression d'une moindre rigueur que les autres, alors même que les services traités sont d'une grande complexité. À leur lecture, on relève entre autres choses un certain nombre de pages brisées ne chargeant pas[53] ou des incohérences[54].

CONCLUSIONS : UN POTENTIEL EXISTANT MAIS ENCORE A DEVELOPPER

Pour conclure, on peut retenir six enseignements principaux de ce tour d'horizon.

Les API des plateformes sont aujourd'hui limitées dans ce qu'elles autorisent :

- Elles sont avant tout conçues pour des usages commerciaux, plutôt que pour permettre la collecte de données.
- Peu sont dédiées spécifiquement aux projets de recherche, et même pour celles-ci, des limitations et des restrictions importantes existent.
- Elles permettent d'accéder à certaines données, mais ne permettent pas d'éprouver directement le fonctionnement des algorithmes eux-mêmes.

Elles sont mouvantes et peu homogènes, augmentant ainsi l'investissement nécessaire pour les prendre en main :

- Chaque API repose sur l'architecture, singulière, de sa plateforme.
- Les documentations sont hétérogènes, et parfois imprécises.
- Les API sont évolutives, aussi bien dans leur architecture et leur documentation que dans leurs conditions d'utilisation.

[53] Les exemples suivants sont particulièrement gênants, touchant à des objets essentiels de la plateforme : https:/developers.facebook.com/docs/graph-api/reference/photo/ ; https:/developers.facebook.com/docs/graph-api/reference/video ; https:/developers.facebook.com/docs/graph-api/reference/page-post

[54] Par exemple, la page de l'objet user/videos ne fait pas référence à celle de l'autorisation user_videos, ce qui laisse à penser qu'elle n'est pas obligatoire (car les autres pages d'objet mentionnent les autorisations nécessaires pour accéder à l'objet), alors que la page de l'autorisation indique bien que l'autorisation user_videos est nécessaire pour accéder à user/videos. https://developers.facebook.com/docs/graph-api/reference/user/videos/ ; https://developers.facebook.com/docs/permissions/reference/user_videos