

Les APIs et leurs enjeux business

Table of Contents

1. Définition de l'API	1
2. L'origine des API	2
a. EDI: Échange de données informatisé	2
b. L'émergence des "web APIs"	2
c. Les avantages d'une web API par rapport à un EDI	3
d. API REST?	3
3. Les conséquences commerciales des API	4
a. Les APIs ont ouvert le logiciel au monde	4
b. Les API ont accéléré l'innovation logicielle	4
c. Les API ont ouvert les données	5
4. Votre entreprise doit-elle ouvrir une API pour partager ses données?	6
5. L'écosystème des API	7
a. Une multitude d'API	7
b. API: un monde professionnel à part entière	7
Des vidéos simples sur le sujet	8
Pour aller plus loin	8

last modified: 2023-05-15



1. Définition de l'API

API: acronyme de **Application Programming Interface**. Une API est le moyen de rendre les logiciels "faciles à brancher et à partager" avec d'autres programmes. Une API est simplement un groupe de règles (que l'on peut aussi appeler une convention, ou un accord ...) que les programmeurs suivent lorsqu'ils écrivent la partie de leur code qui est en charge de communiquer avec d'autres logiciels. Ces règles sont ensuite publiées (sur une page Web par exemple), afin que toute personne ayant besoin de se connecter au programme puisse apprendre quelles règles suivre.

Une API est-elle simplement un moyen d'écrire du code pour s'interfacer avec d'autres programmes? Oui. Pourquoi les API suscitent-elles tant d'enthousiasme alors? Avoir des

conventions sur la façon d'écrire un logiciel pour que quelqu'un puisse le brancher à son propre logiciel est une chose. [APIs comme un aspect de la conception de logiciels](#) est un sujet classique en informatique, mais nous ne sommes pas concernés par cela ici. Les API que nous allons discuter concernent la communication entre des ordinateurs distants, dans un contexte professionnel. Pour mieux comprendre leur pertinence commerciale, il est utile de rappeler un bref historique des API:

2. L'origine des API

Les entreprises qui ont besoin d'échanger des données n'ont rien de nouveau. Fabricants, détaillants, banques, ... ils ont besoin d'échanger des informations à intervalles réguliers. Envoi de factures, réception de reçus pour la marchandise et de nombreux autres documents administratifs générés dans le cours des affaires. Ces reçus, factures ... peuvent être imprimés et envoyés par la poste (cette solution existe toujours bien sûr). Avec l'évolution de l'informatique dans les années 1970 et 1980, un nouveau système d'échange de données et de documents est apparu: l'échange d'informations par ordinateur : l'[Échange de données informatisé \(EDI\)](#).

a. EDI: Échange de données informatisé

EDI n'est pas un échange de pièces jointes dans les emails ou via un transfert de fichiers sur un site web, car les emails et les sites n'existaient pas à l'époque! (Les courriels et le Web ont été adoptés par les entreprises à la fin des années 1990). Au lieu de cela, l'échange de données via EDI consistait à utiliser des outils électroniques de transmission par l'équivalent d'une ligne téléphonique (comme le fax mais encore plus compliqué). Ce système d'EDI était complexe et lourd pour les raisons suivantes :

- chaque industrie a son propre protocole d'échange de données (un protocole pour la logistique, un pour les paiements, un pour telle ou telle chaîne de détaillants, etc.)
- vous avez besoin d'un périphérique ou d'un logiciel dédié pour chaque protocole EDI, et ceux-ci ne sont pas fournis gratuitement
- Les protocoles EDI peuvent varier d'un pays à l'autre
- Les protocoles EDI sont contrôlés par des associations industrielles qui n'adoptent pas l'innovation rapidement
- Les protocoles EDI ont créé des «systèmes fermés»: une entreprise A peut se connecter à la société B via un EDI uniquement si les deux ont un accord préalable pour utiliser cet EDI.

Pour résumer : les EDI sont fragmentés, compliqués à mettre en œuvre, lents à évoluer, chers, et limitent la communication à un «club» de partenaires qui ont accepté de l'utiliser. Les [EDIs existent toujours](#), en particulier dans les grandes industries B2B comme le transport, mais elles ont perdu en popularité dans l'économie en général parce que ... les API sont arrivées.

b. L'émergence des "web APIs"

À la fin des années 1990 et au début des années 2000, Internet et le World Wide Web se sont considérablement développés. De plus en plus de serveurs dans différentes parties du monde ont besoin d'échanger des données entre eux, ce qui nécessite d'utiliser des interfaces plus pratiques

que les EDI. Il devenait de plus en plus commode de définir des conventions simples et universelles que tout le monde pouvait apprendre et suivre pour standardiser ces échanges, gratuitement et facilement. C'est ce que font les **web API**. Elles sont aussi souvent appelées:

- **API** pour faire court
- **web services**
- **REST API** (voir ci-dessous).

Une **web API** étend la logique des API que nous avons vues au début de ce document, à la communication logicielle **via le web**. Pour rappel, une API est une convention suivie lors de l'écriture d'un logiciel, rendant ce logiciel disponible pour d'autres logiciels.



Exemple: l'API de Microsoft PowerPoint permet l'importation de tableaux Excel dans des documents pptx, car l'API de Powerpoint se branche sur l'API d'Excel. Dans cet exemple, Excel et Powerpoint sont censés être installés sur le même ordinateur bien sûr!

Une web API est une API qui permet à deux logiciels de communiquer via Internet. **Ils n'ont pas besoin d'être installés sur le même ordinateur.**

c. Les avantages d'une web API par rapport à un EDI

- Contrairement à un **EDI**, une API web supprime toute contrainte technique spécifique à l'industrie. Les API Web sont simplement une convention suivant les standards du web pour envoyer et recevoir des données sur Internet, sans dire quoi que ce soit sur le contenu des données.
- Les données envoyées et reçues peuvent être des factures, des pages Web, des horaires de train, de l'audio, de la vidéo ... peu importe.
- Contrairement à un EDI, une entreprise qui crée une API web peut choisir de laisser un accès **ouvert** à son API : n'importe quel ordinateur capable de se brancher à Internet peut alors se brancher à ce que cette entreprise rend disponible via cette API (rappelez-vous que les EDI ont besoin que les deux parties aient un accord préétabli).
- Un client potentiel intéressé par l'API Web d'une entreprise peut s'y "brancher" en quelques lignes de code, au lieu d'attendre des semaines ou des mois avant la signature d'un contrat et la configuration de l'EDI.



Dire que les API sont ouvertes ne signifie pas une absence de sécurité: la communication via les API peut facilement être identifiée et cryptée, selon les besoins.

d. API REST?

Deux conventions d'API web populaires ont émergé dans les années 1990 et ont rivalisé en terme d'adoption :

- [SOAP: Simple Object Access Protocol](#)
- [REST: Representational State Transfer](#)

Les **API REST** sont finalement devenues les plus répandues, car elles utilisent les mêmes principes simples que les pages Web utilisent pour être transférées sur Internet (le protocole "http" que vous voyez dans les adresses de pages web). C'est pourquoi les API sont souvent appelées [API REST, présentées dans cette vidéo pédagogique](#). En 2000-2010, il est devenu de plus en plus facile et naturel d'adopter la convention REST pour mettre son logiciel et ses données à la disposition d'un autre ordinateur via Internet. Cette évolution simple pour faciliter l'interopérabilité a eu **des effets immenses** :

3. Les conséquences commerciales des API

a. Les APIs ont ouvert le logiciel au monde

Une API transforme un logiciel fermé en quelque chose qui peut être branché sur n'importe quel autre ordinateur ou objet, à condition qu'il soit connecté à Internet. Par exemple, les API ont été un facteur clé de succès pour [SalesForce](#) au début des années 2000. SalesForce, créé en 1999, a réalisé un chiffre d'affaires de 8,39 milliards de dollars en 2017 :

- SalesForce a développé un CRM en tant que SaaS où les fonctionnalités du CRM étaient **exposées en tant qu'API** (ce qui signifie que ces fonctionnalités pouvaient être connectées à des applications externes via le protocole REST).
- SalesForce a créé un PaaS pour héberger des applications pouvant être connectées au CRM SalesForce via les API développées par SalesForce. Cette plate-forme est appelée [Force.com](#) et les développeurs externes peuvent y mettre leurs applications, à condition qu'elles soient compatibles avec les APIs SalesForce. Salesforce prend une commission sur les ventes réalisées par ces applications tierces hébergées sur Force.com, mais plus important encore, la plate-forme crée un **écosystème** d'applications et de développeurs autour des produits Salesforce, ce qui rend difficile pour une entreprise cliente de passer à un produit différent.

b. Les API ont accéléré l'innovation logicielle

Grâce à l'API, il est désormais plus facile d'ajouter des blocs logiciels et de créer de nouvelles applications, même si ces blocs logiciels proviennent de différents pays ou industries. À titre d'exemple extrême : la police australienne de Victoria a déployé un projet de reconnaissance des véhicules volés grâce à la reconnaissance vidéo des plaques d'immatriculation des voitures circulant dans la rue (les véhicules volés se voient immédiatement reconnaître leurs plaques d'immatriculation). C'est un projet de 86 000 000 \$. Un individu a répliqué ce [projet avec seulement 57 lignes de code et une webcam](#). Comment? Simplement parce qu'il a pu utiliser un logiciel existant pour la reconnaissance de plaques d'immatriculation, disponible en tant qu'API, au lieu de le re-développer par lui-même. Un autre exemple significatif est l'application imaginée par [Pieter Levels](#): un service mondial de livraison de bagages, porte à porte, conçu sans une ligne de code:

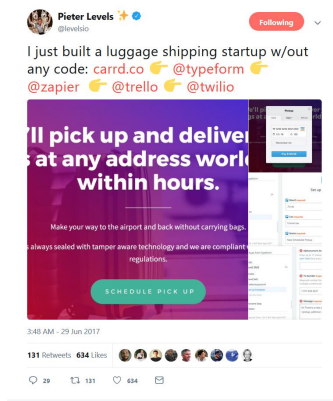


Figure 1. Un service de livraison de bagages porte à porte - conçu sans une ligne de code

Ce service est conçu en organisant plusieurs sous-services, qui se coordonnent en communiquant via leurs APIs.

Comment la communication entre services est-elle possible, en pratique ? Qui "orchestre" ces services? Pieter Levels utilise [Zappier](#), un service dont le rôle est de faire communiquer ces APIs entre elles.

Au-delà de ces exemples frappants, les leçons à tirer sont:

- de plus en plus de services sont disponibles via API. On ne réinvente pas la roue, il suffit de s'en servir.
- coordonner plusieurs APIs permet de créer des services entièrement nouveaux (pas simplement: "gérer mes emails par API")
- des services comme [Zappier](#) permettent la coordination / communication entre APIs, mais cela favorise également l'automatisation.

c. Les API ont ouvert les données

Les entreprises et les organisations publiques possèdent de nombreuses bases de données d'un grand intérêt commercial. L'utilisation de ces ensembles de données peut être gratuite (quand l'utilisateur est développeur un projet à but non lucratif par exemple) ou monétisée si l'utilisateur est une entreprise. Sans APIs, les ensembles de données peuvent être rendus disponibles publiquement sous forme de docs (par exemple, tableurs Excel) à télécharger mais ce n'est pas pratique (essayez de télécharger quelque chose comme [all_train_schedules_2000_to_2017.xls!](#)). Prenons l'exemple d'une entreprise de transport comme la SNCF française qui trouve intéressant de publier les noms des gares, les horaires des trains, les informations en temps réel sur le trafic ferroviaire, etc. car elle pourrait être utilisée par d'autres entreprises pour construire de nouveaux services: comment faire?

- Les données sont sur un serveur de la SNCF
- La SNCF ajoute [une API et sa documentation](#), mettant les données à la disposition des développeurs capables de [se connecter aux API](#), ce qui est une compétence de base dans le [développement de logiciels](#).

- Les entrepreneurs et les programmeurs en général pourront accéder aux données via l'API et les utiliser, en créant de [nouveaux services basés sur ces informations sur les trains](#).

L'**Open data** désigne ce mouvement pour rendre les jeux de données accessibles à un large public, et les API web ont été un ingrédient technologique clé dans ce mouvement.

4. Votre entreprise doit-elle ouvrir une API pour partager ses données?

Une entreprise peut créer ses propres APIs pour "projeter" ses services plus loin et plus fort que par la simple interface "page web". Comment faire? Ici il faut brainstormer, avec deux repères à garder en tête. Ces deux repères ou "balises" sont exemplifiées par ce qu'a vécu Pieter Levels, un entrepreneur hollandais qui se spécialise dans la création de plateformes web au service des personnes qui travaillent à distance ("remote workers"). Il a créé:

- [Nomadlist](#), qui présente les villes du monde et en quoi elles peuvent être attractives pour y vivre et travailler
- [RemoteOK](#), qui est un job board pour annonces exclusivement de jobs à distance.

Pieter Levels a mis en place des APIs pour que des utilisateurs puissent accéder ("consommer") ces deux services, **avec des résultats bien différents**:

- L'accès de NomadList par API, sans contrôle, a abouti à se faire siphonner ses datas par des concurrents qui s'en sont servi pour faire des copycats. (relisez les tweets annonçant [l'ouverture de l'API](#) et [sa fermeture](#)).
- L'accès aux annonces de RemoteOK par une API permet à des plateformes tierces d'intégrer le catalogue d'annonces de RemoteOK aux leurs : c'est du référencement gratuit et à grande échelle, qui accroît la probabilité qu'une annonce trouve un candidat (relisez le tweet qui annonce [l'ouverture de l'API](#)).

La capture d'écran ci-dessous résume le contraste entre les deux situations : bien que dans les deux cas il s'agisse d'ouvrir ses données via une API, le résultat est opposé :



Akshay Kadam(A2K)

@deadcoder0904 · Maker of all things JavaScript 🤖

Nice job Pieter. 2 Questions -

1. How is it gonna help you ?

2. Are you gonna be closing it if someone builds replica of Remote OK just like they did it with NomadList ?

▲ UPVOTE (14) ↩ REPLY MESSAGE SHARE · 7 MONTHS AGO ...



Pieter Levels

MAKER

@levelsio · Serial maker 🏠 Nomad List + et al

@deadcoder0904 Great questions! No. Remote OK has had a public RSS feed for years now. Now I've built this full API firehose in JSON.

The goal here is increasing the amounts of applicants a remote job gets so that the company can recruit from a bigger pool of hiring talent. By increasing distribution (through this API), hopefully that goal is reached.

So opposite of the Nomad List API: the more people use this, the more valuable a Remote OK job post will be for companies that are hiring remote workers.

▲ UPVOTE (22) ↩ REPLY MESSAGE SHARE · 7 MONTHS AGO ...

Figure 2. Pieter Levels explique ses raisons pour ouvrir ou fermer une API

Il faut donc une appréciation fine des usages que l'ouverture d'une API permettra de stimuler. Un des effets les plus vertueux est celui d'une **"caisse de résonance"** : en réutilisant notre contenu pour leurs propres objectifs, les utilisateurs de l'API vont sans le vouloir faire la promotion de notre contenu, ce qui en renforce l'efficacité.

5. L'écosystème des API

a. Une multitude d'API

Pour découvrir de nouvelles API, ou pour faciliter la découverte de vos API, allez sur <https://apilist.fun/>, <https://apislist.com/> ou <https://publicapis.io>. En effectuant une recherche sur ces sites, vous trouverez des APIs fournissant des services commerciaux, ou des APIs d'un genre amusant ou absurde.

Pourtant, de nombreuses API publiques ne sont pas listées sur ces sites. Dans ce cas, une recherche google du type "info dont j'ai besoin + API" est aussi un bon moyen de savoir si l'API que vous recherchez existe. [Intéressé-e par les observations d'oiseaux? Il y a une API pour ça.](#)

b. API: un monde professionnel à part entière

Les API sont devenues essentielles à l'économie. En conséquence, un grand nombre de services associés aux API ont été développés pour répondre à tous les besoins des entreprises qui les utilisent :

- comment créer une API
- comment gérer la documentation d'un grand nombre d'API
- comment connecter une grande variété d'API
- comment contrôler et auditer la sécurité des API
- comment monétiser les API ...

Mehdi Medjaoui et APIscene publient un [panorama mis à jour et interactif](#) des entreprises actives dans l'écosystème des APIs:

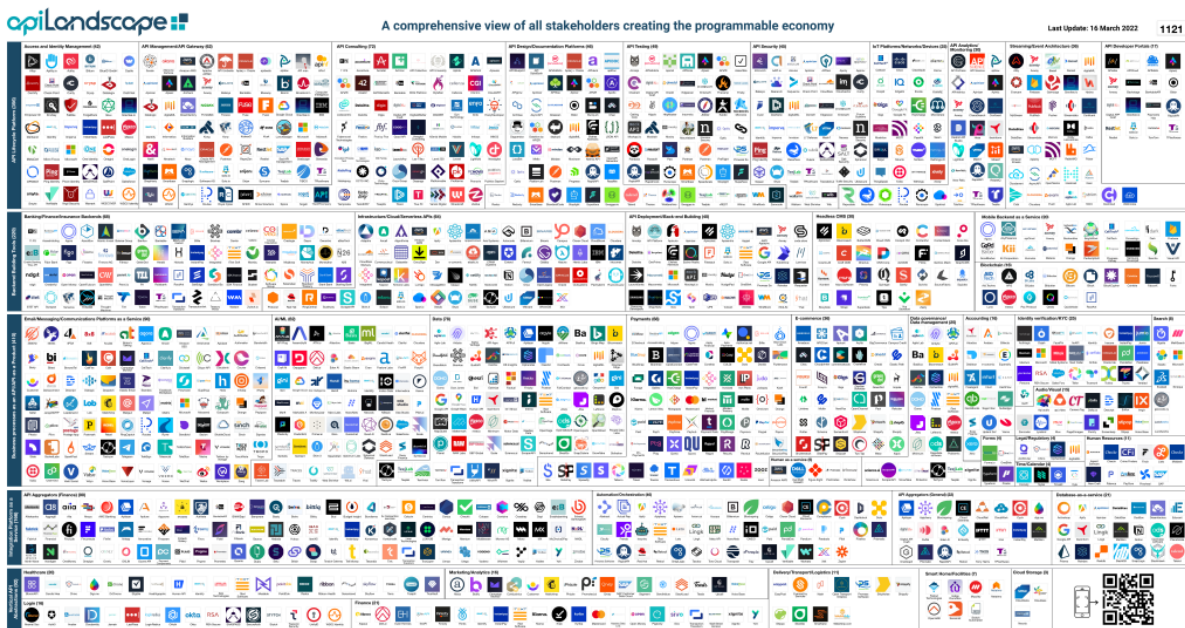


Figure 3. L'écosystème API en 2023 par Mehdi Medjaoui

(ou téléchargez la version haute résolution [de ce poster ici](#))

Des vidéos simples sur le sujet

- Qu'est-ce qu'une API? <https://youtu.be/yKlDe-dXNA4>
- L'API management expliqué à mon boss : https://youtu.be/vUI_y43ao2M

Pour aller plus loin

- [Au-delà de la technologie : penser l'API comme un produit](#) (□ 10 min de lecture).
- [C'est quoi une API ? Une explication \(compréhensible\) pour les utilisateurs métier](#) (□ 10 min de lecture)
- [ma carte memo sur les web APIs en pdf.](#)



Retrouvez le site complet : [ici](#).

Clement Levallois

Découvrez mes autres cours et projets : <https://www.clementlevallois.net>

Ou contactez-moi via Twitter: [@seinecle](#)