API & Web Service & Rest API

Qu'est-ce qu'une API ?

Les API sont des mécanismes qui permettent à deux composants logiciels de communiquer entre eux à l'aide d'un ensemble de définitions et de protocoles. Par exemple, le système logiciel du bureau météorologique contient les données météorologiques quotidiennes. L'application météo de votre téléphone « s'entretient » avec ce système via des API et affiche des mises à jour météorologiques quotidiennes sur votre téléphone.

Que signifie l'acronyme API ?

Une API, pour sa part, désigne une interface de programmation d'application. Dans le contexte des API, le terme « application » désigne tout logiciel ayant une fonction distincte. L'interface peut être considérée comme un contrat de service entre deux applications. Ce contrat définit la manière dont les deux parties communiquent entre elles à l'aide de demandes et de réponses. La documentation de l'API contient des informations sur la manière dont les développeurs doivent structurer ces demandes et ces réponses.

Comment fonctionnent les API ?

L'architecture des API s'explique généralement en termes de client et de serveur. Ainsi, l'application qui envoie la demande est appelée le client, tandis que celle qui envoie la réponse est appelée le serveur. Dès lors, dans l'exemple de la météo, la base de données météorologique du bureau est le serveur, et l'application mobile est le client.

Les API peuvent fonctionner de quatre manières différentes, selon le moment et le motif de leur création.

API SOAP

Ces API utilisent le protocole simple d'accès aux objets (SOAP). Le client et le serveur échangent des messages via XML. Il s'agit d'une API moins flexible qui était plus fréquemment utilisée par le passé.

API RPC

Ces API sont appelées appels de procédure à distance (RPC). Le client exécute une fonction (ou une procédure) sur le serveur, tandis que le serveur renvoie la sortie au client.

API WebSocket

[L'API Websocket](https://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-websocket-api-overview?pg=wianapi&cta=websocketapi) est un autre développement d'API web moderne qui utilise des objets JSON pour transmettre des données. Une API WebSocket prend en charge la communication bidirectionnelle entre les applications client et le serveur. Le serveur peut envoyer des messages de rappel aux clients connectés, ce qui le rend plus efficace que l'API REST.

API REST

Ce sont les API les plus demandées et les plus flexibles que l'on retrouve aujourd'hui sur le web. Le client adresse des demandes au serveur sous forme de données. Le serveur utilise les données du client pour exécuter des fonctions internes et renvoie les données de sortie au client. Nous allons examiner les API REST plus en détail ci-dessous.

Que sont les API REST ?

REST est l'abréviation de Representational State Transfer. REST définit un ensemble de fonctions comme GET, PUT, DELETE, etc., que les clients peuvent utiliser pour accéder aux données du serveur. Les clients et les serveurs échangent des données au moyen du protocole HTTP.

La principale fonction de l'[API REST](https://docs.aws.amazon.com/apigateway/latest/developerguide/http-api-vs-rest?pg=wianapi&cta=restapi) consiste à être sans état. L'absence d'état signifie que les serveurs ne sauvegardent pas les données du client entre les demandes. Les demandes du client adressées au serveur sont similaires aux URL que vous saisissez dans votre navigateur pour consulter un site web. La réponse du serveur est constituée de données simples, sans le rendu graphique classique d'une page web.

Qu'est-ce qu'une API web ?

Une API web ou une API de service web est une interface de traitement d'applications entre un serveur web et un navigateur web. Tous les services web sont des API, mais toutes les API ne sont pas des services web. L'API REST est un type particulier d'API web qui utilise le style architectural standard expliqué ci-dessus.

Les différents termes relatifs aux API, comme API Java ou API de service, existent parce qu'historiquement, les API ont été créées avant le World Wide Web. Les API web modernes sont des API REST et les termes peuvent être utilisés de manière interchangeable.

Que sont les intégrations d'API ?

Les intégrations d'API sont des composants logiciels qui mettent automatiquement à jour les données entre les clients et les serveurs. Parmi les exemples d'intégration d'API, citons la synchronisation automatique des données vers le cloud à partir de la galerie d'images de votre téléphone, ou la synchronisation automatique de l'heure et de la date sur votre ordinateur portable lorsque vous changez de fuseau horaire. Les entreprises peuvent également les utiliser pour automatiser efficacement de nombreuses fonctions du système.

Quels sont les avantages des API REST ?

Les API REST offrent quatre avantages principaux :

1. Intégration

Les API servent à intégrer de nouvelles applications aux systèmes logiciels existants. Cela permet d'accélérer le développement, étant donné que chaque fonctionnalité ne doit pas être écrite en ne partant de rien. Vous pouvez utiliser les API pour tirer parti du code existant.

2. Innovation

Des secteurs d'activité entiers peuvent changer avec l'arrivée d'une nouvelle application. Aussi, les entreprises doivent réagir rapidement et soutenir le déploiement rapide de services innovants. Pour y parvenir, elles doivent apporter des modifications au niveau de l'API sans avoir à réécrire l'intégralité du code.

3. Expansion

Les API offrent aux entreprises une occasion unique de répondre aux besoins de leurs clients sur différentes plateformes. Par exemple, l'API de cartes permet d'intégrer des informations cartographiques par le biais de sites Web, d'Android, d'iOS, etc. Toute entreprise peut donner un accès similaire à ses bases de données internes en utilisant des API gratuites ou payantes.

4. Facilité de maintenance

L'API fait office de passerelle entre deux systèmes. Chaque système est obligé d'apporter des modifications internes pour que l'API ne soit pas impactée. De cette façon, toute modification future du code par une partie n'a aucune incidence sur l'autre partie.

Quels sont les différents types d'API ?

Les API sont classées à la fois en fonction de leur architecture et de leur champ d'utilisation. Nous avons déjà exploré les principaux types d'architectures d'API, passons donc au champ d'application.

API privées

Elles sont internes à une entreprise et ne servent qu'à connecter des systèmes et des données au sein de l'entreprise.

API publiques

Elles sont ouvertes au public et peuvent être utilisées par tous. Il peut y avoir ou non une autorisation et un coût associés à ces types d'API.

API partenaire

Elles ne sont accessibles qu'aux développeurs externes autorisés afin de faciliter les partenariats interentreprises.

API composites

CElles combinent deux API différentes ou plus pour faire face aux exigences ou aux comportements complexes du système.

Qu'est-ce qu'un point de terminaison d'API et pourquoi est-il important ?

Les points de terminaison d'API sont les points de contact finaux dans le système de communication des API. Il s'agit notamment des URL des serveurs, des services et d'autres emplacements numériques spécifiques à partir desquels les informations sont envoyées et reçues entre les systèmes. Les points de terminaison d'API sont essentiels pour les entreprises pour les deux raisons principales ci-dessous :

1. Sécurité

Les points de terminaison d'API rendent le système vulnérable aux attaques. La surveillance d'API est indispensable pour prévenir les abus.

2. Performances

Les points de terminaison d'API, en particulier ceux à fort trafic, peuvent provoquer des goulets d'étranglement et affecter les performances du système.

Comment sécuriser une API REST ?

Toutes les API doivent être sécurisées par le biais d'une authentification et d'un contrôle appropriés. Les deux principaux moyens de sécuriser les API REST sont les suivants :

1. Les jetons d'authentification

Ils servent à autoriser les utilisateurs à effectuer l'appel d'API. Les jetons d'authentification vérifient que les utilisateurs sont bien ceux qu'ils prétendent être et qu'ils disposent des droits d'accès pour cet appel d'API particulier. Par exemple, lorsque vous vous connectez à votre serveur de messagerie, votre client de messagerie utilise des jetons d'authentification pour sécuriser l'accès.

2. Les clés d'API

Les clés d'API permettent de vérifier le programme ou l'application à l'origine de l'appel d'API. Elles identifient l'application et s'assurent qu'elle dispose des droits d'accès requis pour effectuer l'appel d'API particulier. Les clés d'API ne sont pas aussi sûres que les jetons, mais elles permettent de surveiller les API afin de recueillir des données sur leur utilisation. Vous avez peut-être remarqué une longue chaîne de caractères et de chiffres dans l'URL de votre navigateur lorsque vous consultez différents sites web. Cette chaîne est une clé d'API que le site web utilise pour effectuer des appels d'API internes.

**SOA (Service-Oriented Architecture)** is a type of software architecture, which is based on the integration of applications through services. These services are: compositions, BPM, proxies and even APIs.

REST and SOAP are also a means of implementing services, and they are technologies which on the surface offer the same functionality, but they do behave in the same way.  REST and SOAP follow the same SOA architecture, so both follow the same “principles” with a different approach.