

# À 23 ans, cURL se rend utile partout

*Non, ce n'est pas une marque de biscuits apéritifs aux cacahuètes, mais un outil logiciel méconnu qui mérite pourtant de faire partie de notre culture numérique. On révisé ?*

*Bien qu'ignoré de la plupart des internautes, il est d'un usage familier pour celles et ceux qui travaillent en coulisses pour le Web. L'adage : « si un site web n'est pas « curlable », ce n'est pas du Web » fait de curl un outil majeur pour définir ce qu'est le Web aujourd'hui.*

*Son développeur principal suédois Daniel Stenberg retrace ici la genèse de cURL, son continu développement et surtout, ce qui devrait intéresser les moins geeks de nos lecteurs, son incroyable diffusion, à travers sa bibliothèque `libcurl`, au cœur de tant d'objets et appareils familiers. Après des débuts fort modestes, « la quasi-totalité des appareils connectés à Internet exécutent curl » !*

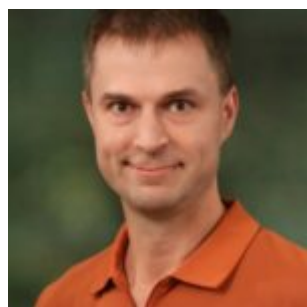
*Si vous n'êtes pas développeur et que les caractéristiques techniques de curl sont pour vous un peu obscures, allez directement à la deuxième partie de l'article pour en comprendre facilement l'importance.*

Article original sur le blog de l'auteur : [curl is 23 years old today](#)

Traduction : Fabrice, Goofy, Julien/Sphinx, jums, NGFChristian, tTh,

## curl fête son 23<sup>e</sup> anniversaire

par **Daniel Stenberg**



La date de naissance officielle de curl, le 20 mars 1998, marque le jour de la toute première archive `tar` qui pouvait installer un outil nommé `curl`. Je l'ai créé et nommé **curl 4.0** car je voulais poursuivre la numérotation de versions que j'avais déjà utilisée pour les noms précédents de l'outil. Ou plutôt, j'ai fait un saut depuis la 3.12, qui était la dernière

version que j'avais utilisée avec le nom précédent : *urlget*.

Bien sûr, curl n'a pas été créé à partir de rien ce jour-là. Son histoire commence une petite année plus tôt : le 11 novembre 1996 a été publié un outil appelé *httpget*, développé par Rafael Sagula. C'est un projet j'avais trouvé et auquel j'avais commencé à contribuer. *httpget* 0.1 était un fichier de code C de moins de 300 lignes. (La plus ancienne version dont j'aie encore le code source est *httpget* 1.3, retrouvée ici)



Comme je l'ai évoqué à de multiples reprises, j'ai commencé à m'attaquer à ce projet parce que je voulais avoir un petit outil pour télécharger des taux de change régulièrement depuis un site internet et les fournir via mon petit *bot*<sup>1</sup> de taux de change sur IRC<sup>2</sup>.

Ces petites décisions prises rapidement à cette époque allaient avoir plus tard un impact significatif et influencer ma vie. curl a depuis toujours été l'un de mes passe-temps principaux. Et bien sûr il s'agit aussi d'un travail à plein temps depuis quelques années désormais.

En ce même jour de novembre 1996 a été distribué pour la première fois le programme *Wget* (1.4.0). Ce projet existait déjà sous un autre nom avant d'être distribué, et je crois me souvenir que je ne le connaissais pas, j'utilisais *httpget* pour mon roboticiel. Peut-être que j'avais vu *Wget* et que je ne l'utilisais pas à cause de sa taille : l'archive tar de *Wget* 1.4.0 faisait 171 ko.

Peu de temps après, je suis devenu le responsable de *httpget* et j'ai étendu ses fonctionnalités. Il a ensuite été renommé *urlget* quand je lui ai ajouté le support de Gopher et de FTP (ce que j'ai fait parce que j'avais trouvé des taux de change sur des serveurs qui utilisaient ces protocoles). Au printemps 1998, j'ai ajouté le support de l'envoi par FTP et comme le nom du programme ne correspondait plus clairement à ses fonctionnalités j'ai dû le renommer une fois de plus<sup>3</sup>.

Nommer les choses est vraiment difficile. Je voulais un nom court dans le style d'Unix. Je n'ai pas planché très longtemps puisque j'ai trouvé un mot marrant assez rapidement. L'outil fonctionne avec des URL, et c'est un client Internet. « c » pour client et URL ont fait que « cURL » semblait assez pertinent et marrant.

Et court. Bien dans de style d'Unix.

J'ai toujours voulu que curl soit un citoyen qui s'inscrive dans la tradition Unix d'utiliser des *pipes*<sup>4</sup>, la sortie standard, etc. Je voulais que curl fonctionne comme la commande *cat*, mais avec des URL. curl envoie donc par défaut le contenu de l'URL sur la sortie standard dans le terminal. Tout comme le fait *cat*. Il nous permettra donc de « voir » le contenu de l'URL. La lettre C se prononce si (Ndt : ou see, « voir » en anglais) donc « see URL » fonctionne aussi. Mon esprit blagueur n'en demande pas plus, mais je dis toujours « kurl ».



Le logo original, créé en 1998 par Henrik Hellerstedt

J'ai créé le paquet curl 4.0 et l'ai rendu public ce vendredi-là. Il comprenait alors 2 200 lignes de code. Dans la distribution de curl 4.8 que j'ai faite quelques mois plus tard, le fichier THANKS mentionnait 7 contributeurs[Note] Daniel Stenberg, Rafael Sagula, Sampo Kellomaki, Linas Vepstas, Bjorn Reese, Johan Anderson, Kjell Ericson[/note]. Ça nous a pris presque sept ans pour atteindre une centaine de contributeurs. Aujourd'hui, ce fichier liste plus de 2 300 noms et nous ajoutons quelques centaines de noms chaque année. Ce n'est pas un projet solo !

## Il ne s'est rien passé de spécial

curl n'a pas eu un succès massif. Quelques personnes l'ont découvert et deux semaines après cette première distribution, j'ai envoyé la 4.1 avec quelques corrections de bugs et une tradition de plusieurs décennies a commencé : continuer à publier des mises à jour avec des corrections de bug. « Distribuer tôt et souvent » est un mantra auquel nous nous sommes conformés.

Plus tard en 1998, après plus de quinze distributions, notre page web présentait cette excellente déclaration :

## Mailing List

Due to the massive popularity, We have started a mailing list about this tool. (Version 4.8.4 alone got downloaded more than 300 times from this site.)

Capture d'écran : *Liste de diffusion. Pour cause de grande popularité, nous avons lancé une liste de diffusion pour parler de cet outil. (La version 4.8.4 a été téléchargée à elle seule plus de 300 fois depuis ce site.)*

## Trois cents téléchargements !

Je n'ai jamais eu envie de conquérir le monde ou nourri de visions idylliques pour ce projet et cet outil. Je voulais juste que les transferts sur Internet soient corrects, rapides et fiables, et c'est ce sur quoi j'ai travaillé.

Afin de mieux procurer au monde entier de quoi réaliser de bons transferts sur Internet, nous avons lancé la bibliothèque libcurl, publiée pour la première fois à l'été 2000. Cela a permis au projet de se développer sous un autre angle <sup>5</sup>. Avec les années, libcurl est devenue de facto **une API de transfert Internet**.

Aujourd'hui, alors qu'il fête son 23e anniversaire, c'est toujours principalement sous cet angle que je vois le cœur de mon travail sur curl et ce que je dois y faire. Je crois que si j'ai atteint un certain succès avec curl avec le temps, c'est principalement grâce à une qualité spécifique. En un mot :



## Persistence

Nous tenons bon. Nous surmontons les problèmes et continuons à perfectionner le produit. Nous sommes dans une course de fond. Il m'a fallu deux ans (si l'on compte à partir des précurseurs) pour atteindre 300 téléchargements. Il en a fallu dix ou presque pour qu'il soit largement disponible et utilisé.

En 2008, le site web pour curl a servi 100 Go de données par mois. Ce mois-ci, c'est 15 600 Go (soit fait 156 fois plus en 156 mois) ! Mais bien entendu, la plupart des utilisatrices et utilisateurs ne téléchargent rien depuis notre site, c'est leur distribution ou leur système d'exploitation qui leur met à disposition curl.

curl a été adopté au sein de Red Hat Linux à la fin de l'année 1998, c'est devenu un paquet Debian en mai 1999 et faisait partie de Mac OS X 10.1 en août 2001. Aujourd'hui, il est distribué par défaut dans Windows 10 et dans les appareils iOS et Android. Sans oublier les consoles de jeux comme la Nintendo Switch, la Xbox et la PS5 de Sony.

Ce qui est amusant, c'est que libcurl est utilisé par les deux principaux systèmes d'exploitation mobiles mais que ces derniers n'exposent pas d'API pour ses fonctionnalités. Aussi, de nombreuses applications, y compris celles qui sont largement diffusées, incluent-elles leur propre version de libcurl : YouTube,

Skype, Instagram, Spotify, Google Photos, Netflix, etc. Cela signifie que la plupart des personnes qui utilisent un smartphone ont curl installé plusieurs fois sur leur téléphone.

Par ailleurs, libcurl est utilisé par certains des jeux vidéos les plus populaires : GTA V, Fortnite, PUBG mobile, Red Dead Redemption 2, etc.

libcurl alimente les fonctionnalités de lecteurs média ou de box comme Roku, Apple TV utilisés par peut-être un demi milliard de téléviseurs.

curl et libcurl sont pratiquement intégrés dans chaque serveur Internet, c'est le moteur de transfert par défaut de PHP qui est lui-même utilisé par environ 80% des deux milliards de sites web.

Maintenant que les voitures sont connectées à Internet, libcurl est utilisé dans presque chaque voiture moderne pour le transfert de données depuis et vers les véhicules.

À cela s'ajoutent les lecteurs média, les appareils culinaires ou médicaux, les imprimantes, les montres intelligentes et les nombreux objets « intelligents » connectés à Internet. Autrement dit, **la quasi-totalité des appareils connectés à Internet exécutent curl.**

Je suis convaincu que je n'exagère pas quand je déclare que curl existe sur plus de dix milliards d'installations sur la planète.



**10,000,000,000  
installations**

## **Seuls et solides**

À plusieurs reprises lors de ces années, j'ai essayé de voir si curl pouvait rejoindre une organisation cadre mais aucune n'a accepté et, en fin de compte, je pense que c'est mieux ainsi. Nous sommes complètement seuls et indépendants de toute organisation ou entreprise. Nous faisons comme nous l'entendons et nous ne suivons aucune règle édictée par d'autres. Ces dernières années, les dons et les soutiens financiers se sont vraiment accélérés et nous sommes désormais en situation de pouvoir confortablement payer des récompenses pour la chasse aux bugs de sécurité par exemple.

Le support commercial que nous proposons, wolfSSL et moi-même, a je crois renforcé curl : cela me laisse encore plus de temps pour travailler sur curl, et permet à davantage d'entreprises d'utiliser curl sereinement, ce qui en fin de compte le rend meilleur pour tous.

Ces **300** lignes de code fin 1996 sont devenues **172 000** lignes en mars 2021.

## **L'avenir**

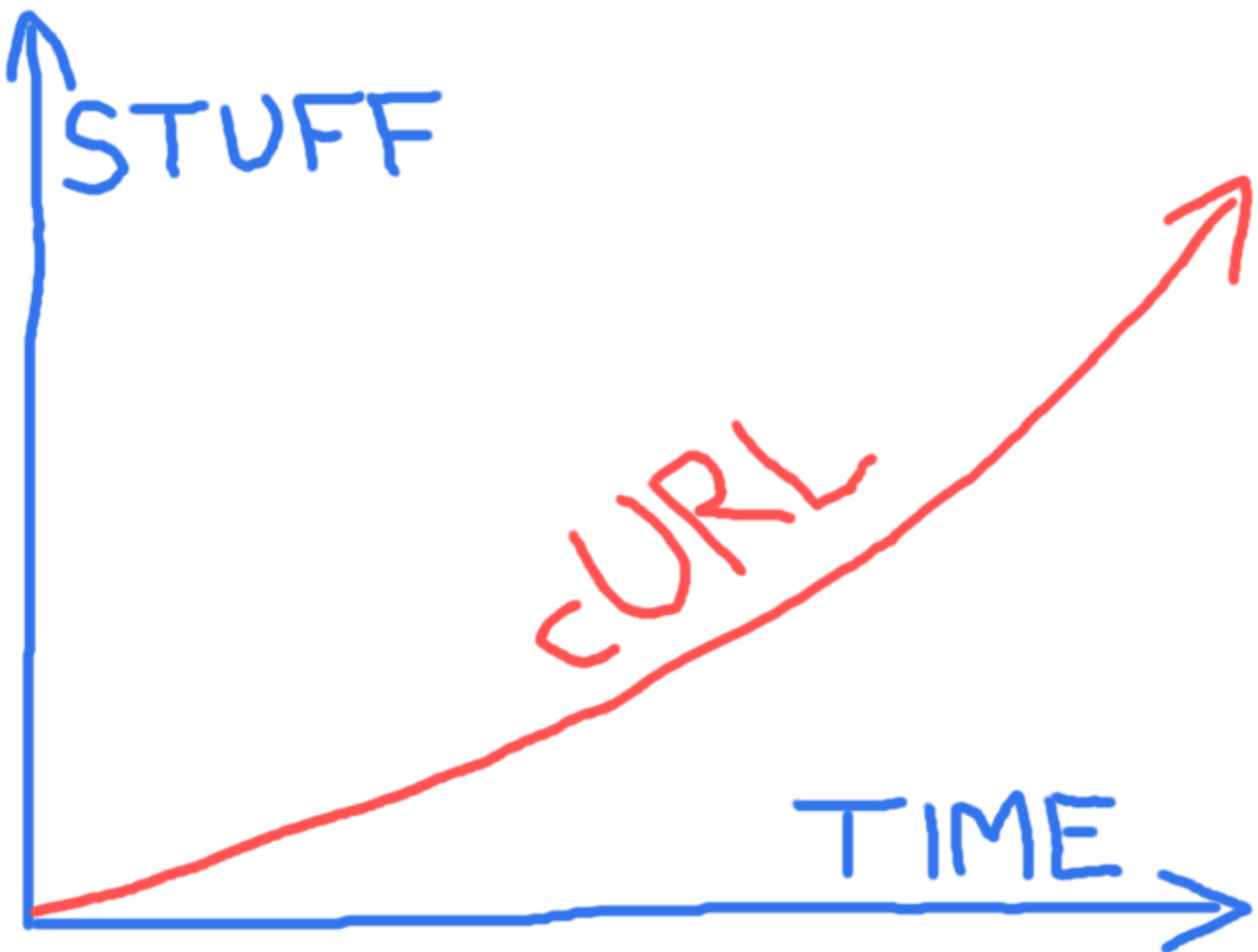
Notre travail le plus important est de ne pas « faire chavirer le navire ». Fournir la meilleure et la plus solide bibliothèque de transfert par Internet, sur le plus de

plateformes possible.

Mais pour rester attractifs, nous devons vivre avec notre temps, c'est à dire de nous adapter aux nouveaux protocoles et aux nouvelles habitudes au fur et à mesure qu'ils apparaissent. Prendre en charge de nouvelles versions de protocoles, permettre de meilleures manières de faire les choses et, au fur et à mesure, abandonner les mauvaises choses de manière responsable pour ne pas nuire aux personnes qui utilisent le logiciel.

À court terme je pense que nous voulons travailler à nous assurer du fonctionnement d'HTTP/3, faire de Hyper<sup>6</sup> une brique de très bonne qualité et voir ce que devient la brique rustls.

Après 23 ans, nous n'avons toujours pas de vision idéale ni de feuille de route pour nous guider. Nous allons là où Internet et nos utilisatrices nous mènent. Vers l'avant, toujours plus haut !



La feuille de route de curl ☐



## 23 ans de curl en 23 nombres

Pendant les jours qui ont précédé cet anniversaire, j'ai tweeté 23 « statistiques de curl » en utilisant le hashtag #curl23. Ci-dessous ces vingt-trois nombres et faits.

- 2 200 lignes de code en mars 1998 sont devenues 170 000 lignes en 2021 alors que curl fête ses 23 ans
- 14 bibliothèques TLS différentes sont prises en charge par curl alors qu'il fête ses 23 ans
- 2 348 contributrices et contributeurs ont participé au développement de curl alors qu'il fête ses 23 ans
- 197 versions de curl ont été publiées alors que curl fête ses 23 ans
- 6 787 correctifs de bug ont été enregistrés alors que curl fête ses 23 ans
- 10 000 000 000 d'installations dans le monde font que curl est l'un des jeunes hommes de 23 ans des plus distribués au monde
- 871 personnes ont contribué au code de curl pour en faire un projet vieux de 23 ans
- 935 000 000 récupérations pour l'image docker officielle de curl (83 récupérations par seconde) alors que curl fête ses 23 ans
- 22 marques de voitures, au moins, exécutent curl dans leurs véhicules alors que curl fête ses 23 ans
- 100 tâches d'intégration continue sont exécutées pour chaque commit et pull request sur le projet curl alors qu'il fête ses 23 ans
- 15 000 heures de son temps libre ont été passés sur le projet curl par Daniel alors qu'il fête ses 23 ans
- 2 systèmes d'exploitation mobile les plus utilisés intègrent et utilisent curl dans leur architecture alors que curl fête ses 23 ans
- 86 systèmes d'exploitation différents font tourner curl alors qu'il fête ses 23 ans
- 250 000 000 de télévisions utilisent curl alors qu'il fête ses 23 ans
- 26 protocoles de transport sont pris en charge alors que curl fête ses 23 ans
- 36 bibliothèques tierces différentes peuvent être compilées afin d'être utilisées par curl alors qu'il fête ses 23 ans
- 22 architectures différentes de processeurs exécutent curl alors qu'il fête ses 23 ans
- 4 400 dollars américains ont été payés au total pour la découverte de

bugs alors que curl fête ses 23 ans

- 240 options de ligne de commandes existent lorsque curl fête ses 23 ans
- 15 600 Go de données sont téléchargées chaque mois depuis le site web de curl alors que curl fête ses 23 ans
- 60 passerelles vers d'autres langages existent pour permettre aux programmeurs de transférer des données facilement en utilisant n'importe quel langage alors que curl fête ses 23 ans
- 1 327 449 mots composent l'ensemble des RFC pertinentes qu'il faut lire pour les opérations effectuées par curl alors que curl fête ses 23 ans
- 1 fondateur et développeur principal est resté dans les parages du projet alors que curl fête ses 23 ans

---

Crédits images AnnaER