

# Documenter c'est s'enrichir (Libres conseils 18/42)

Chaque jeudi à 21h, rendez-vous sur le framapad de traduction, le travail collaboratif sera ensuite publié ici même.

Traduction Framalang : lamessen, lerouge, Kalupa, Sky, Astalaseven, Alpha, LuD-up, CoudCoud, peupleLa, goofy

## La documentation et moi, avant et après

**Anne Gentle**

*Anne Gentle est une rédactrice technique acharnée et la coordinatrice de la documentation communautaire à Rackspace pour OpenStack, un projet open source d'informatique dans le nuage. Avant de rejoindre OpenStack, Anne travaillait en tant que consultante de publication communautaire, en donnant une direction stratégique aux rédacteurs professionnels qui veulent produire du contenu en ligne sur des wikis contenant des pages et des commentaires générés par les utilisateurs. Sa passion pour les méthodes communautaires de documentation l'a amenée à écrire un livre sur les techniques de publication collaborative pour la documentation technique intitulé Conversation et communauté : la documentation dans le web social. Elle s'occupe aussi bénévolement de la maintenance de la documentation pour les manuels FLOSS qui proposent de la documentation open source pour les projets open source.*

Voilà une prémisse bien étrange : vider mes tripes sur ce que j'aurais voulu savoir de l'*open source* et de la documentation. Plutôt que de vous dire ce que je veux que vous sachiez sur l'*open source* et la documentation, je dois vous dire ce que j'aurais aimé que mon moi antérieur sache. Cette demande suscite un sentiment de nostalgie ou de remords voire cette horrible énigme : « à quoi pouvais-je bien penser ? ».

En ce qui me concerne, avec juste cinq ans de moins, mon moi antérieur était une

trentenaire bien installée dans sa vie professionnelle. D'autres, au contraire, se souviennent qu'ils n'étaient qu'adolescents lors de leurs premières expériences *open source*. Jono Bacon dans son livre *L'art de la Communauté*, raconte comment il s'est tenu devant la porte d'un appartement, le cœur battant, alors qu'il était sur le point de rencontrer quelqu'un à qui il n'avait jamais parlé que sur le réseau, par le biais d'une communauté *open source*. J'ai moi aussi fait cette expérience de la première rencontre physique de gens que j'avais découverts en ligne, mais ma première incursion dans le monde de la documentation *open source* s'est produite lorsque j'ai répondu à une demande d'aide par courriel.

Le courriel provenait d'un ancien collègue qui me demandait de l'aide pour la documentation de l'ordinateur portable XO, le projet fondateur de l'organisation *One Laptop Per Child* (un portable pour chaque enfant). J'ai réfléchi à ce que je pensais être une proposition intéressante, j'en ai parlé à mes amis et à mon époux en me demandant si ce n'était pas une bonne occasion d'expérimenter de nouvelles techniques de documentation et d'essayer une chose que je n'avais jamais faite auparavant : une documentation basée sur un wiki. Depuis cette première expérience, j'ai rejoint *OpenStack*, un projet *open source* sur une solution d'informatique dans le nuage, et je travaille à plein temps sur la documentation et le support communautaires.

Je pense immédiatement aux nombreuses contradictions que j'ai rencontrées tout au long de la route. J'ai découvert que pour chaque observation il existe des champs et contre-champs surprenants. Par exemple, la documentation est absolument indispensable pour l'aide aux utilisateurs, l'éducation, la possibilité de choisir ou bien la documentation promotionnelle qui amène à adopter un logiciel. Pourtant, une communauté *open source* pourra continuer d'avancer malgré le manque de documentation ou de la doc complètement défectueuse. Voici une autre collision apparente de valeurs : la documentation pourrait être une très bonne tâche pour démarrer, un point de départ pour les nouveaux volontaires, pourtant les nouveaux membres de la communauté savent si peu de choses qu'il ne leur est pas possible d'écrire ni même d'éditer de manière efficace. En outre les petits nouveaux ne sont pas bien familiers des différents publics auxquels doit servir la documentation.

Ces derniers temps on entend dire un peu partout : « les développeurs devraient écrire la doc de développement » parce qu'ils connaissent bien ce public et par conséquent cela lui serait aussi utile qu'à eux-mêmes. D'après mon expérience, un

regard frais et neuf est toujours le bienvenu sur un projet et certaines personnes ont la capacité d'écrire et de partager avec d'autres ce regard frais et empathique. Mais vous n'allez sûrement pas vous mettre à créer une culture « réservée aux novices » autour de la doc parce qu'il est important que des membres essentiels de la communauté technique apportent leur contribution aux efforts de documentation, et qu'ils encouragent aussi les autres à y participer.

Une partie du vilain petit secret sur la documentation des projets *open source* est qu'il n'existe qu'une frontière pour le moins floue entre leur documentation officielle et leur documentation officieuse. Si seulement j'avais su que les efforts de documentation devraient être sans cesse renouvelés et que de nouveaux sites web pourraient apparaître là où il n'y en avait pas... Une documentation extensive n'est pas le moyen le plus efficace pour s'initier à des projets ou des logiciels mais un parcours sinueux dans les méandres de la documentation sur le Web peut s'avérer plus instructif pour ceux qui veulent lire entre les lignes et avoir ainsi une idée de ce qui se passe dans la communauté grâce à la documentation. Avoir beaucoup de forks(1) et des publics variés peut indiquer que le produit est complexe et qu'il est très suivi. Cela peut aussi signifier que la communauté n'a mis en place aucune pratique quant à la documentation de référence ou que les efforts désorganisés sont la norme.

À mes débuts, j'aurais aimé avoir la capacité de ressentir la « température conviviale » d'une communauté en ligne. Quand vous entrez dans un restaurant rempli de tables nappées de blanc, de couples qui dînent et de conversations feutrées, l'information visuelle et auditive que vous recevez détermine l'ambiance et vous donne quelques indices sur ce que vous vous apprêtez à vivre lors de votre repas. Vous pouvez tout à fait transposer ce concept de température conviviale à une communauté en ligne.

Une communauté *open source* vous donne quelques indices dans ses listes de discussion, par exemple. Une page de présentation de la liste qui commence par de nombreuses règles et conventions sur la manière de poster indique une gouvernance très stricte. Une liste de discussion qui contient de nombreuses publications mettant l'accent sur le fait qu'il « n'y a pas de question bête » est plus accueillante pour de nouveaux rédacteurs de documentation.

J'aurais aussi aimé connaître un moyen de faire non seulement de l'audit de contenu, c'est-à-dire lister le contenu à disposition pour le projet *open source*,

mais aussi de l'audit de communauté, donc lister les membres influents de la communauté open source, qu'il s'agisse ou non de contributeurs.

Pour terminer, une observation à propos de l'*open source* et de la documentation que j'ai pu vérifier avec plaisir, c'est l'idée que la rédaction de la documentation peut s'effectuer via des « sprints » — grâce à des brusques dépenses d'énergie avec un public ciblé et un but précis, pour aboutir à un ensemble documentaire de référence.

J'ai été très contente d'entendre lors d'une conférence à SXSW Interactive que les sprints sont tout à fait acceptables pour la collaboration en ligne, qu'il faut s'attendre à des fluctuations du niveau d'énergie de chacun et que c'est OK. La documentation des logiciels est souvent faite à l'arrache dans les moments d'accalmie d'un cycle de release(2) et ça ne pose aucun problème dans la documentation *open source* qui est basée sur la communauté. Vous pouvez adopter une approche stratégique et coordonnée et continuer tout de même de proposer des événements de grande intensité autour de la documentation. Ce sont des moments exaltants dans l'*open source* et mon ancien moi les a pleinement ressentis. C'est une bonne chose que vous continuiez à passer de votre moi antérieur à votre moi actuel avec un paquet de conseils en poche.

(1) Un *fork* est un nouveau logiciel créé à partir du code source d'un logiciel existant.

(2) La *release* est la publication d'un logiciel.