

Développement de logiciel libre

Sébastien Morais
Ingénieur Chercheur au CEA
Contact : sebastien.morais@cea.fr

Références

- <https://www.gnu.org/>
→ système d'exploitation et mouvement du logiciel libre
- <https://producingoss.com>
→ livre sur le développement open source
- <https://openhatch.org>
→ (archive) organisation aidant à contribuer dans l'open source
- <http://teachingopensource.org/>
→ actualité / blog posts sur l'enseignement open source
- <http://open-advice.org/>
→ conseils de développeurs open source
- <http://oss-watch.ac.uk>
→ conseil sur l'utilisation, le développement et les licences open source

V – Communications

- V.1 – Ce que vous écrivez vous défini
- V.2 – Quelques pièges communs (mauvais modèles de communication)
- V.3 – Gérer les personnes difficiles
- V.4 – Gérer la croissance
- V.5 – La publicité

V – Communications

- Note : ce sur quoi on va faire le focus → « comment communiquer »
 - L'importance des compétences en communication est souvent minimisée
 - sur le long terme, cela **peut compter plus que le talent de programmation** d'un individu
 - un **bon programmeur sans compétence de communication** :
 - peut réaliser une chose à la fois
 - aura du mal à convaincre les autres
 - un **mauvais programmeur avec compétence de communication** :
 - aura du mal à réaliser des choses
 - peut coordonner et persuader d'autres de faire plein de choses
- => l'un peut facilement créer une dynamique / apporter de l'élan au projet, l'autre plus difficilement

V – Communications

- Les capacités de communication importantes dans un projet :
 - description des problèmes techniques
 - représente une petite partie des communications d'un projet
 - savoir faire preuve d'empathie avec son auditoire
 - capacité à voir ses messages comme les autres les voient
 - capacité à faire en sorte que les autres voient leurs propres messages avec la même objectivité
 - capacité à constater lorsqu'une méthode de communication n'est plus fonctionnelle

V – Communications

- Les logiciels open sources sont composés d'une audience et de mécanisme de communication très diverses !
 - Il faut donc faire attention à tout un tas de questions :
 - **où doit être exprimée une pensée ?**
 - liste de diffusion ? traqueur de bogues ? commentaire de code ?
 - **quelle niveau de connaissance attendre** lors de votre réponse dans un forum publique ?
 - le lecteur n'est pas uniquement celui qui émet la question !
 - comment savoir quand un **média a atteint sa limite** de capacité et que faire ?
 - comment garder un contact constructif entre les utilisateurs et développeurs ?
- Dans ce chapitre, on va se focaliser sur
 - comment mener vos communications
 - comment gérer les mécanisme de communication

V – Communications

- **V.1 – Ce que vous écrivez vous définit**
- V.2 – Quelques pièges communs (mauvais modèles de communication)
- V.3 – Gérer les personnes difficiles
- V.4 – Gérer la croissance
- V.5 – La publicité

Qu'est ce que savent les gens sur vous ?

Qu'est ce que savent les gens sur vous ?

- Sur internet, ce que les gens savent de vous vient de ce que vous écrivez ou de ce que les autres écrivent sur vous !

=> Il est important de faire attention à son écriture

→ **une personne brillante et charismatique peut passer pour désorganisée**
(voir laxiste)

→ peut se produire si ces mails sont décousus et non structurés

→ **une personne désorganisée n'a pas besoin que tout le monde le sache**

→ peut être couvert par des messages claires et lucides

- Question : comment donner une bonne impression par votre écriture ?

Qu'est ce que savent les gens sur vous ?

>>> Histoire de Jim Blandy:

>>> Contexte : travaux sur GNU Emacs et production de version bêta hebdomadaire

>>>

>>> Communication fréquente avec une personne (Mr X) jamais rencontrée IRL mais

>>> qui faisait du bon travail : rapports de bogues clairs, proposition de correctif

>>> presque toujours juste, ...

>>> Pour reverser le code écrit par quelqu'un d'autre la Free Software Foundation avait

>>> besoin de documents juridiques cédant les droits d'auteur

>>>

>>> => Jim demande à Mr X de remplir des formulaires en lui expliquant ce qu'ils signifient et

>>> demandant une signature de l'employeur de Mr X

>>> Problème : Mr X répond qu'il n'a pas d'employeur

>>> => Jim demande une signature de l'université

>>> Un problème se pose à nouveau

>>> Question : quel est le problème remonté par Mr X ?

Qu'est ce que savent les gens sur vous ?

>>> Histoire de Jim Blandy:

>>> Contexte : travaux sur GNU Emacs et production de version bêta hebdomadaire

>>>

>>> Communication fréquente avec une personne (Mr X) jamais rencontrée IRL mais

>>> qui faisait du bon travail : rapports de bogues clairs, proposition de correctif

>>> presque toujours juste, ...

>>> Pour reverser le code écrit par quelqu'un d'autre la Free Software Foundation avait

>>> besoin de documents juridiques cédant les droits d'auteur

>>>

>>> => Jim demande à Mr X de remplir des formulaires en lui expliquant ce qu'ils signifient et

>>> demandant une signature de l'employeur de Mr X

>>> Problème : Mr X répond qu'il n'a pas d'employeur

>>> => Jim demande une signature de l'université

>>> Un problème se pose à nouveau

>>> Question : quel est le problème remonté par Mr X ?

>>> Problème : Mr X est un enfant de 13 ans qui vit chez ses parents !

Le besoin de structure et de formatage!

- Le besoin de structure et formatage dépend du support de communication

→ dans un **mail** : pas de langage SMS, des phrases complètes, capitalisez le premier mot, utilisez des sauts de paragraphe, ...

→ sur IRC ou un **forum à discussion éphémère** : il est acceptable d'utiliser des **formes abrégées**, des **expressions courantes**, ...

=> Si une communication doit persister :

→ rédaction avec **structure narrative cohérente**

→ besoin de prêter attention à **l'orthographe et à la grammaire**

- 2 raisons principales de prêter attention à ces règles :

→ elles rendent le texte plus lisible

→ elles donnent une bonne impression auprès du lecteur

Le besoin de structure et de formatage!

- Exemple de ma vie professionnelle
- Contexte : MR étant partie en roue libre sur l'utilisation des annotations de type des variables en Python
 - >>> Oui pour tout c'est pas simple, même si comme je le faisais remarquer il y a
 - >>> vite des limites... tout n'est pas possible.
 - >>> J'utilise deux modules un qui crée Toto et l'autre crée Titi. Ces deux classes ont une
 - >>> méthode MaTaille. Je fais une fonction qui va utiliser que la méthode MaTaille de ces deux
 - >>> classes. Faut il et comment faire alors pour dire que Toto et Titi réponde bien au critère
 - >>> de MonTruc qui compris une méthode MaTaille, alors que Toto et Titi peuvent être aussi
 - >>> des implémentations C++, venant d'un autre module dont je n'ai a priori pas la main,
 - >>> etc ?
- Question : quel est votre a priori sur la personne ?

Le besoin de structure et de formatage!

- Exemple de ma vie professionnelle
- Contexte : MR étant partie en roue libre sur l'utilisation des annotations de type des variables en Python
 - >>> Oui pour tout c'est pas simple, même si comme je le faisais remarquer il y a
 - >>> vite des limites... tout n'est pas possible.
 - >>> J'utilise deux modules un qui crée Toto et l'autre crée Titi. Ces deux classes ont une
 - >>> méthode MaTaille. Je fais une fonction qui va utiliser que la méthode MaTaille de ces deux
 - >>> classes. Faut il et comment faire alors pour dire que Toto et Titi réponde bien au critère
 - >>> de MonTruc qui compris une méthode MaTaille, alors que Toto et Titi peuvent être aussi
 - >>> des implémentations C++, venant d'un autre module dont je n'ai a priori pas la main,
 - >>> etc ?
- Question : quel est votre a priori sur la personne ?
 - sans connaître la personne, a priori négatif car manque structure, de relecture, ...
 - pourtant collègue compétent mais qui écrit souvent comme il réfléchit

Le besoin de structure et de formatage!

- Dans le cas des courriers électronique, il existe des conventions :
 - utiliser du texte brut dans le mail et pas de format qui pourrait rendre le texte opaque au lecteur (pas de HTML, RichText, ...)
 - limiter la taille des lignes (par exemple pas plus de 80 colonnes)
 - norme souvent présente dans les messages de commit
 - lors de la citation du courrier de quelqu'un
 - insérez vos réponses là où c'est approprié
 - coupez les parties non utilisées
 - écrivez en haut du courrier les commentaires s'appliquant à l'ensemble du courrier
 - réfléchir sur quoi écrire pour l'objet du mail
 - si c'est une réponse à un fils qui diverge, ne pas hésiter à répondre en **modifiant l'objet du courrier** (préserve l'historique et clarifie que le sujet a dérivé)
 - si vous voulez démarrer un nouveau sujet, faites le avec un nouveau courrier et non en répondant à un courrier existant en changeant le sujet
- Note : certaines de ces conventions peuvent être « évitées » lors d'échanges sur liste privée
 - exemple : utilisation d'abréviations LaTeX

- Question : maintenant que nous avons vu comment ferrer les lecteurs avec du texte formaté, comment faire pour les conserver ?

Attention au contenu

- Question : maintenant que nous avons vu comment ferrer les lecteurs avec du texte formaté, comment faire pour les conserver ?
- 1) Simplifier le plus possible la vie du lecteur
 - beaucoup d'informations dans les projets open source et les lecteurs ne peuvent pas tout savoir
 - => si possible, les messages doivent contenir des informations utiles au lecteur, dans un format pratique pour celui-ci
 - **proposer l'URL** d'un fil de discussion particulier / d'une vidéo / ...
<https://github.com/python-gitlab/python-gitlab/issues/1582>
 - **résumer les conclusions** d'un fil de discussion complexe pour donner du contexte
 - l'effort de simplification est important pour les gros projets où le ratio lecteur / écrivain est élevé (son utilité passe à l'échelle)

Attention au contenu

- 2) Éviter d'exagérer vos commentaires, pas d'hyperbole !
→ course à l'armement classique
- Exemple concret :
 1. un utilisateur rencontre et rapporte l'existence d'un bogue
 2. peu d'attention de la part des développeurs
 3. ajout de commentaires pour décrire le problème comme dramatiquement bloquant pour l'utilisateur, ses collègues, sa famille, son chien, ...
- Note : L'exagération ne se limite pas aux utilisateurs
→ apparaît souvent lors de débats technique où le désaccord est une question de goût plutôt que de justesse !

Attention au contenu

- Exemple perso :

>>> A l'opposé de cette démarche, je conseille de ne pas hésiter à user des ... En

>>> terme de performance, cela n'a pas d'impact dans le code, si tu ne vas pas à

>>> aller jusqu'à utiliser la méthode interdite `isinstance()`

→ pourquoi parler de « méthode interdite » quand celle-ci est proposée par le langage ?

- Autre exemple :

>>> Le faire de cette façon rendrait le code totalement illisible. Ce serait un

>>> cauchemar de maintenance, par rapport à la proposition de M. X...

- Note : il est peu probable de réussir à se débarrasser entièrement de l'exagération

→ pas grave pour le projet car globalement peu dommageable

→ uniquement gênant pour l'auteur car il perd en crédibilité

Attention au contenu

- Autre conseil : penser à relire vos messages
 - important pour ceux de **grandes tailles** ou pour les discussions en ligne où **l'écriture est souvent discontinue**, par exemple :
 - début de rédaction du message
 - plusieurs aller-retour en parallèle pour consulter des ressources, e.g. courriers, pages Web, ...
 - => peut facilement mener à une rupture du sens narratif
- Note : les messages écrits de manière discontinue et non vérifiés sont facilement identifiables à la lecture !
 - => Il est important de se relire
 - plus vos publications sont structurées, plus elles seront lues !

Surveillez votre « ton »

- Avec le temps : probable convergence vers un style concis qui peut être pris pour de la froideur
→ tendance tout à fait normale et style par défaut des développeurs de logiciel libre
- Exemple de réponse reçu par [Karl Fogel] sur une liste de diffusion à propos d'un logiciel de gestion

>>> Can you possibly elaborate a bit more on exactly what problems you ran into, etc?

>>> Also:

>>> - What version of Slash are you using? I couldn't tell from your original message.

>>> - Exactly how did you build the apache/mod_perl source?

>>> - Did you try the Apache 2.0 patch that was posted about on slashcode.com?

>>> Shane

- Question : dans ce message, qu'est ce qui peut être mal interprété ?

Surveillez votre « ton »

- Avec le temps : probable convergence vers un style concis qui peut être pris pour de la froideur
→ tendance tout à fait normale et style par défaut des développeurs de logiciel libre
- Exemple de réponse reçu par [Karl Fogel] sur une liste de diffusion à propos d'un logiciel de gestion

>>> Can you possibly elaborate a bit more on exactly what problems you ran into, etc?

>>> Also:

>>> - What version of Slash are you using? I couldn't tell from your original message.

>>> - Exactly how did you build the apache/mod_perl source?

>>> - Did you try the Apache 2.0 patch that was posted about on slashcode.com?

>>> Shane

- Question : dans ce message, qu'est ce qui peut être mal interprété ?
 - absence de salutation
 - pas de signature autre que le nom de la personne
 - message composée de questions en série et écrites de manière très compacte !
 - la seule phrase déclarative est une critique implicite

Surveillez votre « ton »

- Question : comment prendre le ton utilisé par la personne ?

Surveillez votre « ton »

- Question : comment prendre le ton utilisé par la personne ?
 - un habitué de l'open source pourrait
 - prendre cela comme un signe que la personne était occupée
 - être content que quelqu'un passe du temps sur son problème
 - M. ToutLeMonde n'aurait pas forcément réagi aussi positivement !
- Cas pratique : quelqu'un avec peu de confiance en soit publie en reconnaissant avoir fait une erreur
 - la réponse **peut être compacte** mais **doit être empathique**
 - exemple de bonne réponse** : une brève analyse de la situation et terminant par un petit geste amical comme :
 - donner des conseils sur comment corriger le bogue
 - terminer par un "bonne chance Mr X"
 - lâcher un petit smiley ou un émoticône pour rassurer

Surveillez votre « ton »

- Question : pourquoi se concentrer sur les sentiments du participant et pas uniquement sur ce qu'il dit ?

Surveillez votre « ton »

- Question : pourquoi se concentrer sur les sentiments du participant et pas uniquement sur ce qu'il dit ?
 - **les sentiments affectent la productivité ! ! ! !**
 - un contributeur **mécontent participe moins** et fait des **contributions de moins bonne qualité**

=> Essayez de comprendre ce que ressent une personne en se fondant sur :

- comment la plus part des gens ressentiraient une situation ?
- ce que vous savez sur la personne en fonction de vos interactions passées

- Note : il est important de ne pas attendre des autres qu'ils vous communiquent clairement ce qu'ils ressentent
 - ce n'est pas un comportement standard, même dans la vraie vie
 - **facile** de s'exprimer publiquement sur des **émotions dirigées vers les autres** (gratitude / indignation)
 - **difficile** de s'exprimer publiquement sur des émotions dirigées **vers soi-même** (fierté / insécurité)

=> Surveiller le ton de votre écriture peut avoir une influence surprenante sur le projet

L'impolitesse dans l'open source

- L'open source possède sa propre définition de politesse
 - Question : qu'est ce qui constitue (ou pas) de l'impolitesse dans l'open source ?
- (Réponses aux prochains slides)
 - conventions que tout le monde peut rencontrer sans les connaître
 - non exclusives à l'open source (mathématiques, sciences « dures », ...)

L'impolitesse dans l'open source

- Question : la critique technique (étouffée ou directe) est-elle grossière ?

L'impolitesse dans l'open source

- Question : la critique technique (étouffée ou directe) est-elle grossière ?
 - Non, cela peut même être l'inverse car cela implique que la personne mérite d'être prise au sérieux et qu'on lui consacre du temps
 - Supposons que quelqu'un propose par écrit une manière maladroite de régler un problème. Qu'elle est la réponse la plus impolie ?
 - le fait de proposer une critique de l'approche ?
 - le fait d'ignorer le message ?
- => Plus il est viable d'ignorer un message, plus le fait d'y répondre peut être interpréter comme un « compliment » / forme de respect
- Évidemment, les critiques personnelles et l'impolitesse évidente SONT de l'impolitesse

L'impolitesse dans l'open source

- Question : les questions directes et sans fioritures sont-elles de l'impolitesse ?

L'impolitesse dans l'open source

- Question : les questions directes et sans fioritures sont-elles de l'impolitesse ?

>>> Can you possibly elaborate a bit more on exactly what problems you ran into, etc?

>>> Also:

>>> What version of Slash are you using? I couldn't tell from your original message.

>>> Exactly how did you build the apache/mod_perl source?

>>> Did you try the Apache 2.0 patch that was posted about on slashcode.com?

>>> Shane

→ Non, des questions peuvent sembler froides / rhétoriques elles n'ont pas d'objectif caché

→ Seul objectif : obtenir des informations pertinentes rapidement

L'impolitesse dans l'open source

- Typiquement, les fameuses questions "Avez-vous débranché votre box ?", "Votre ordinateur est-il branché ?", ...
 - réel besoin pour diagnostiquer et écarter rapidement des possibilités

=> A mesure que cette question est posée

- tend à ne **plus être préfixée par des fioritures**
- tend à **converger vers la concision**

- Ce genre de comportement est fréquent dans l'open source
 - l'intention n'est pas d'insulter
 - permet d'écarter les explications évidentes à un problème
- Expérience perso : "l'utilisation d'un calculateur" → temps de calcul OU utilisation d'un code (File system diff)
 - recherche du problème dans le code, dans les dépendances, ...
 - raison éclaircie par la suite avec une question évidente
- Note :
 - Ne pas mal réagir est bien vu / attendu par la communauté open source
 - Il ne faut pas réprimandé ceux qui réagissent mal (collision de culture) et simplement expliquer la raison de cette "froideur"

L'impolitesse dans l'open source

- Question : donc, qu'est ce qui est impoli ? !

L'impolitesse dans l'open source

- Question : donc, qu'est ce qui est impoli ? !
 - Ne pas fournir une critique détaillée peut être une sorte d'insulte
 - pas d'efforts revient à dire que le contributeur et le sujet ne valent pas votre temps
 - la critique non détaillée est acceptable si vous vous exprimez sur la lacune de votre message ("Je pense que ... mais je n'ai pas eu le temps de le rechercher, désolé")
 - Note : l'absence de réaction à une contribution, e.g. modification du code, nouvelle issue, ... est acceptable car tout le monde supposera que vous n'aviez pas le temps
 - pas acceptable si vous aviez promis par avance une réaction détaillée
- => Mieux vaut montrer que votre temps est précieux en étant concis qu'en étant paresseux !
- D'autres formes d'impolitesse existent mais sont généralement « standard », i.e. non spécifiques au développement de logiciels libres
 - cf chapitre sur le démarrage d'un projet

Identité

- **Contrairement à la vraie vie, personne ne voit votre visage en ligne**
 - votre nom de compte / mail est votre visage en ligne !
 - conseil sur le choix de ce nom : permutation liée à votre nom réel (ex: SMorais)
 - avoir un pseudonyme fantaisiste ne change pas qui vous êtes et n'impressionne personne
 - effet inverse car cela donne l'impression de superficialité / insécurité
- **Lors du besoin d'anonymat, utilisez un faux vrai nom !**
- **Pour garder la face en ligne vous pouvez aussi faire attention à :**
 1. **l'utilisation abusive de votre « titre »** (e.g. « docteur », « professeur », « directeur », ...)
 - dans l'open source, perçu comme signe d'insécurité / d'exclusion
 - acceptable dans une signature de mail mais pas comme outil de persuasion

Identité

2. l'utilisation d'une signature réduite est de bon goût

→ évitez les déclarations juridiques (surtout si incompatibles avec l'open source)

- Exemple :

>>> IMPORTANT NOTICE

>>>

>>> If you have received this e-mail in error or wish to read our e-mail disclaimer statement and monitoring policy, please

>>> refer to the statement below or contact the sender.

>>>

>>> This communication and any attachments contain information which is confidential and may also be privileged. It is for

>>> the exclusive use of the intended recipient(s). If you are not the intended recipient(s) please note that any form of

>>> disclosure, distribution, copying or use of this communication or the information in it or in any attachments is strictly

>>> prohibited and may be unlawful.

>>> If you have received this communication in error, please return it with the title "received in error" to blablabla@truc.com

>>> then delete the email and destroy any copies of it.

>>> When addressed to our clients, any opinions or advice contained in this e-mail and any attachments are subject to the

>>> terms and conditions expressed in the governing blablabla client engagement letter.

Identité

- Question : dans le contexte de la participation à un projet open source, qu'est ce qui ne va pas dans la signature précédente ?

Identité

- Question : dans le contexte de la participation à un projet open source, qu'est ce qui ne va pas dans la signature précédente ?
 - incapacité à configurer des outils de communication
 - pas / peu de support de l'entreprise pour des contributions open source
- Note : la personne devrait demander une modification de la signature auprès de son entreprise
 - dans le cas d'un refus, elle devrait utiliser son mail perso
- Note : signature qui peut ne pas déranger dans des échanges privés ponctuels

V – Communications

- V.1 – Ce que vous écrivez vous définit
- **V.2 – Quelques pièges communs (mauvais modèles de communication)**
- V.3 – Gérer les personnes difficiles
- V.4 – Gérer la croissance
- V.5 – La publicité

Évitez de publier sans raison ...

- Un travers fréquent est de croire qu'il est nécessaire de répondre à tout !
 - problématique car le nombre de fils de discussion devient vite trop important
 - certains commentaires n'attendent pas de réponse !
 - => Ne répondez pas systématiquement !
- Question : à votre avis, quels sont les 3 types de messages les plus courant dans les forums de développement ?

Évitez de publier sans raison ...

- Un travers fréquent est de croire qu'il est nécessaire de répondre à tout !
 - problématique car le nombre de fils de discussion devient vite trop important
 - certains commentaires n'attendent pas de réponse !
 - => Ne répondez pas systématiquement !
- Question : à votre avis, quels sont les 3 types de messages les plus courant dans les forums de développement ?
 - message proposant quelque chose de non trivial
<https://github.com/numpy/numpy/issues/19735>
 - message d'approbation / d'opposition au commentaire d'un tiers
<https://github.com/python-gitlab/python-gitlab/issues/1529> + 2nd message
 - message récapitulatif
<https://github.com/docker/compose/issues/3656> + ctrl F "sum up"
- Ces messages ne nécessite pas particulièrement de réponse !

Évitez de publier sans raison ...

- Question : quand doit-on apporter une réponse ?

Évitez de publier sans raison ...

- Question : quand doit-on apporter une réponse ?
 - lorsque la **réponse** est **motivée par un objectif précis** !
 - qu'est ce que vous souhaitez réaliser ?
 - est-ce que cela ne sera pas accomplis sans votre réponse ?
- Question : quelles seraient des **bonnes raisons** pour ajouter sa voix à un fil de discussion ?

Évitez de publier sans raison ...

- Question : quand doit-on apporter une réponse ?
 - lorsque la **réponse** est **motivée par un objectif précis** !
 - qu'est ce que vous souhaitez réaliser ?
 - est-ce que cela ne sera pas accomplis sans votre réponse ?
- Question : quelles seraient des bonnes raisons pour ajouter sa voix à un fil de discussion ?
 - vous voyez un **défaut dans une proposition** et pensez être le seul à le voir
 - vous voyez un **problème de communication** entre les participants et souhaitez le corriger avec un post de clarification
 - (optionnel) remercier ou affirmer son soutien
 - **acceptable car ne nécessite pas d'effort mental**

=> Dans l'ensemble essayez de faire en sorte que les gens souhaitent que vous postiez plus plutôt que moins !

Fils de discussion productifs / non productifs

- Sur une liste de diffusion chargée, il y a deux impératifs:
 - **déterminer ce à quoi faire attention / ignorer**
 - se comporter de manière à **éviter de faire du « bruit »**

=> Objectif : maximiser le ratio signal / bruit et chercher à ce que les autres en fassent autant

- Question : Quelles sont les caractéristiques d'un fil improductif ?

Fils de discussion productifs / non productifs

- Sur une liste de diffusion chargée, il y a deux impératifs:

- **déterminer ce à quoi faire attention / ignorer**
- se comporter de manière à **éviter de faire du « bruit »**

=> Objectif : maximiser le ratio signal / bruit et chercher à ce que les autres en fassent autant

- Question : Quelles sont les caractéristiques d'un fil improductif ?

- **répéter des arguments** déjà avancés comme si personne ne les avait entendus
- augmentation de l'implication et de l'**exagération** alors que les enjeux sont de plus en plus petits
- majorité de commentaires de **personnes qui contribuent peu / pas** et, en parallèle, **silence des autres**
- beaucoup d'idées discutées mais **pas de propositions claires** de faites

Fils de discussion productifs / non productifs

- Cependant, un fil de discussion non productif != sujet important
=> il est important de pouvoir guider un fil de discussion vers l'utilité !
- Question : comment faire cela ?
 - >>> Cette discussion ne mène nulle part. Peut-on s'il vous plaît laisser
 - >>> tomber ce sujet jusqu'à ce que quelqu'un ait un correctif pour
 - >>> mettre en œuvre l'une de ces propositions ? Il n'y a aucune raison
 - >>> de continuer à dire les mêmes choses. Le code parle plus fort que
 - >>> les mots, les amis.
- Question : est-ce que ce message est viable pour rendre la discussion utile pour le projet ?

Fils de discussion productifs / non productifs

- Réponse : non !
- Pour guider un fil de discussion vers l'utilité, faut-il :
 - avertir les gens d'arrêter de perdre leur temps ? (non !)
 - demander de ne pas publier sans avoir quelque chose de constructif à dire ? (non !)
 - l'un ou l'autre est juste offensant !
 - suggérer des conditions pour progresser, (oui !)
 - par exemple : donner un itinéraire / chemin vers le résultat que vous souhaitez !
- Attention : il est important de ne pas dicter la conduite à suivre sous peine que cela ait l'effet inverse
 - => attention au ton utilisé !

Fils de discussion productifs / non productifs

- Exemple de message viable pour rendre la discussion utile :

>>> Plusieurs propositions ont été avancées dans ce fil, mais aucune n'a été
>>> détaillée dans tous les détails, du moins pas assez pour un vote positif ou
>>> négatif.

>>> Pourtant, nous ne disons rien de nouveau et ne faisons que répéter ce qui a
>>> déjà été dit. La meilleure chose à ce stade serait probablement que les
>>> prochains commentaires contiennent soit une spécification complète du
>>> comportement proposé, soit un correctif. Ensuite, nous aurions au moins une
>>> action précise à prendre, i.e. obtention d'un consensus sur la spécification ou
>>> appliquer et tester le correctif.

Fils de discussion productifs / non productifs

- Question : quelles sont les différences entre la seconde approche et la première ?

Fils de discussion productifs / non productifs

- Question : quelles sont les différences entre la seconde approche et la première ?
 - **ne trace pas une ligne entre vous et les autres**
 - **n'accuse pas** les autres de faire tourner en rond la discussion
 - décrit de manière **impartiale** pourquoi le fil de discussion n'avance pas (pas péjoratif et pas de jugements)
 - **propose un plan d'action positif** pour ramener la conversation à un niveau plus constructif

Fils de discussion productifs / non productifs

- Note : parfois, vous voudrez simplement faire disparaître le fil de discussion
- Il n'existe pas de moyen infaillible de fermer un fil de discussion
 - Vous pouvez très bien demander au gens de proposer des progrès visible ou d'arrêter de publier TANT QUE cela est fait avec diplomatie
- Question : comment le faire sans en avoir l'air ?

Fils de discussion productifs / non productifs

- **Note :** parfois, vous voudrez simplement faire disparaître le fil de discussion
- **Il n'existe pas de moyen infaillible de fermer un fil de discussion**
 - Vous pouvez très bien demander au gens de proposer des progrès visible ou d'arrêter de publier TANT QUE cela est fait avec diplomatie
- **Question : comment le faire sans en avoir l'air ?**
 - à ma connaissance, peu d'approche viable car situation très délicate
 - une approche possible serait de proposer des mesures que personnes ne va réellement prendre
- **Note :** il est peu probable qu'il n'y ait pas de post après le votre
 - mails qui se croisent
 - personne voulant avoir le dernier mot
 - => laissez simplement le fil de discussion s'épuiser
- **Note :** il est important de ne pas fermer les fils de discussions trop tôt car cela mettrait fin au processus créatif

Plus le sujet est simple, plus la discussion tend à s'éterniser

- Question : à votre avis, pourquoi est-ce le cas ?

Plus le sujet est simple, plus la discussion tend à s'éterniser

- Question : à votre avis, pourquoi est-ce le cas ?
 - la probabilité de **faire de l'esbroufe augmente** à mesure que la **difficulté technique diminue** !
 - si le sujet technique est compliqué, seuls les développeurs expérimentés peuvent suivre
- ⇒ Le consensus est plus difficile à atteindre sur des questions techniques simples où il est facile de se faire une opinion
- C'est aussi le cas pour des sujets « mous » comme l'organisation, le financement, ...
 - pas de qualification nécessaire pour participer au débat
 - pas de moyen clair de décider si une décision est bonne ou non
 - faire durer la prise de décision est une tactique efficace

Plus le sujet est simple, plus la discussion tend à s'éterniser

- Le principe selon lequel la quantité de discussion est inversement proportionnelle à la complexité du sujet est connu sous le nom de Loi de futilité de Parkinson
- Définition Wikipedia :
 - >>> La loi de futilité de Parkinson est une loi empirique selon
 - >>> laquelle les organisations donnent une importance
 - >>> disproportionnée à des questions futiles
- Cette idée est illustré par Parkinson avec un garage à vélos et une centre atomique

Question : à votre avis, en quoi la création de ces deux objets permet à Parkinson d'illustrer son idée ?

Plus le sujet est simple, plus la discussion tend à s'éterniser

- Parkinson explique d'une part comment:
 1. entrer dans le conseil d'administration
 2. obtenir l'approbation pour la construction d'une centrale atomique de plusieurs millions / milliards de dollars
- D'autre part il explique que construire l'abri à vélos sera un sujet de discussions sans fin
- Explication : cela vient du fait qu'un sujet est vaste et compliqué, et que les gens ne peuvent pas le comprendre directement
 - n'essaye pas de comprendre et suppose que quelqu'un a vérifié les détails
 - ce n'est pas le cas du garage à vélo
 - => Quelqu'un saisi l'occasion pour montrer qu'il fait attention / qu'il est là / qu'il fait bien son travail / ...
- Question : comment éviter que ce phénomène se produise ?

Plus le sujet est simple, plus la discussion tend à s'éterniser

- Parkinson explique d'une part comment:
 1. entrer dans le conseil d'administration
 2. obtenir l'approbation pour la construction d'une centrale atomique de plusieurs millions / milliards de dollars
- D'autre part il explique que construire l'abri à vélos sera un sujet de discussions sans fin
- Explication : cela vient du fait qu'un sujet est vaste et compliqué, et que les gens ne peuvent pas le comprendre directement
 - n'essaye pas de comprendre et suppose que quelqu'un a vérifié les détails
 - ce n'est pas le cas du garage à vélo
 - => Quelqu'un saisi l'occasion pour montrer qu'il fait attention / qu'il est là / qu'il fait bien son travail / ...
- Question : comment éviter que ce phénomène se produise ?
 - difficile à faire car il est presque impossible de persuader tout le monde d'éviter ce principe
 - lorsqu'il se produit faites-le remarquer !
 - **objectif** : diffuser une prise de conscience de ce phénomène pour limiter sa fréquence d'apparition

Évitez les guerres saintes

- Question : qu'est ce qu'une guerre sainte ?

Évitez les guerres saintes

- Question : qu'est ce qu'une guerre sainte ?
 - différend souvent lié à une question mineur
 - exemple : est-ce que les pizzas à l'ananas sont vraiment des pizzas ?!
 - pas résolue par des arguments
 - suscite les passions et tend à faire argumenter pour faire remporter son camp
- Question : voyez-vous pourquoi les guerres saintes diffèrent du principe de Parkinson ?

Évitez les guerres saintes

- **Question : qu'est ce qu'une guerre sainte ?**

- différend souvent liée à une question mineur

- exemple : est-ce que les pizzas à l'ananas sont vraiment des pizzas ?!

- pas résolue par des arguments

- suscite les passions et tend à faire argumenter pour faire remporter son camp

- **Question : voyez vous pourquoi les guerres saintes diffèrent du principe de Parkinson ?**

- principe de Parkinson : les gens expriment leurs opinions sans en être convaincus et peuvent donner des opinions contradictoire

- guerre sainte : les gens sont convaincu de leur opinion et comprendre celui des autres est une faiblesse

- **Question : comment sont perçues les guerres saintes ?**

Évitez les guerres saintes

- **Question : qu'est ce qu'une guerre sainte ?**
 - différend souvent liée à une question mineur
 - exemple : est-ce que les pizzas à l'ananas sont vraiment des pizzas ?!
 - pas résolue par des arguments
 - suscite les passions et tend à faire argumenter pour faire remporter son camp
- **Question : voyez vous pourquoi les guerres saintes diffèrent du principe de Parkinson ?**
 - principe de Parkinson : les gens expriment leur opinion sans en être convaincus et peuvent donner des opinions contradictoire
 - guerre sainte : les gens sont convaincu de leur opinion et comprendre celui des autres est une faiblesse
- **Question : comment sont perçues les guerres saintes ?**
 - Participants : l'autre côté est obtus et ne pense pas clairement mais pourraient être « sauvé » s'il est suffisamment intimidé
 - Extérieur : les deux côtés ne vont pas changer d'avis
- **Note : il peut exister un bon « côté dans une guerre sainte mais il n'existe pas d'algorithme pour prouver que l'un ou l'autre a raison**
 - dans mon cas, c'était toujours mon côté le « bon »

Évitez les guerres saintes

- Question : comment faire pour résoudre une guerre sainte ?

Évitez les guerres saintes

- Question : comment faire pour résoudre une guerre sainte ?

Proposition : dire que la discussion a consommé beaucoup de temps et d'énergie alors que cela n'en vaut pas la peine, et proposer de laisser tomber

- Question : que pensez-vous de cette approche ?

Évitez les guerres saintes

- Question : comment faire pour résoudre une guerre sainte ?

Proposition : dire que la discussion a consommé beaucoup de temps et d'énergie alors que cela n'en vaut pas la peine, et proposer de laisser tomber

- Question : que pensez-vous de cette approche ?

→ Elle est maladroite :

- le **temps et l'énergie déjà dépensés** ne seront jamais récupérés
→ si quelqu'un pense qu'un peu plus de discussion terminera le débat, elle continuera à discuter
- proposer **l'abandon revient souvent à faire remporter une des parties** (celle du statu quo) et il se peut que tout le monde soit d'accord sur le besoin de prendre une décision

- Bonne proposition : faire en sorte que cette guerre ne se produise pas !

- souvent anticipable car certains sujets sont propices à cela, e.g. les langages de programmation, les licences, ...
- certaines guerres saintes sont propres au projet et ont tendance à ressurgir

- Autre proposition : exprimer sa sympathie et sa compréhension pour les points soulevés par l'autre camp

Évitez les guerres saintes

- Retour sur M. X :

>>> Pour Any, je m'attendais à cette réponse de ta part, @M. Y, et je te l'accorde que c'est

>>> « anti-annotation ». ;)

>>> Dans le cas d'Optional, je pourrais comprendre ce que tu dis si... on ne fixait pas de valeur par

>>> défaut à Optional. Ok, on aurait un paramètre obligatoire ou un paramètre obligatoire ayant une

>>> valeur par défaut. Ça me va. Mais alors ensuite, un paramètre optionnel, vraiment optionnel n'existe

>>> que si l'utilisateur le définit... sinon, elle n'a pas de valeur par défaut puisqu'elle n'a pas

>>> d'existenceblablabla...

- Question : pourquoi cette tentative d'expression de compréhension de l'autre camp ne fonctionne pas ?

Évitez les guerres saintes

- Retour sur Mr X :

>>> Pour Any, je m'attendais à cette réponse de ta part, @Mr Y, et je te l'accorde que c'est

>>> « anti-annotation ». ;)

>>> Dans le cas d'Optional, je pourrais comprendre ce que tu dis si... on ne fixait pas de valeur par

>>> défaut à Optional. Ok, on aurait un paramètre obligatoire ou un paramètre obligatoire ayant une

>>> valeur par défaut. Ça me va. Mais alors ensuite, un paramètre optionnel, vraiment optionnel n'existe

>>> que si l'utilisateur le définit... sinon, elle n'a pas de valeur par défaut puisqu'elle n'a pas

>>> d'existenceblablabla...

- Question : pourquoi cette tentative d'expression de compréhension de l'autre camp ne fonctionne pas ?

- manque d'expression d'une certaine incertitude par Mr X

- il aurait dû :

- montrer qu'il comprenait les arguments que l'autre côté avance
- expliquer qu'il trouve les arguments au moins raisonnables, si ce n'est convaincants !

=> Cela commençait bien et aurait pu passer pour un geste qui laisse place à un geste réciproque pour converger vers une situation qui s'améliore, i.e. limiter les dommages collatéraux au moral du projet

Évitez les guerres saintes

- Dernières propositions pour éviter une guerre sainte :
 - **Abandonnez publiquement si c'est dans votre intérêt !**
 - préciser que c'est parce que la guerre sainte n'en vaut pas la peine
 - n'exprimez pas d'amertume / dernier coup d'adieu
 - l'abandon n'est efficace que s'il est fait élégamment
 - **Défendez votre idée sans dire ouvertement qu'elle est mieux qu'une autre**
 - Exemple** : le choix du langage de programmation dans un projet
 - défendre votre langage en prétextant que c'est celui dans lequel vous êtes à l'aise pour coder rapidement / efficacement
 - ne pas le défendre en disant qu'il est mieux que les autres langages envisagés

V – Communications

- V.1 – Ce que vous écrivez vous définit
- V.2 – Quelques pièges communs (mauvais modèles de communication)
- **V.3 – Gérer les personnes difficiles**
- V.4 – Gérer la croissance
- V.5 – La publicité

Personne difficile

- Le terme « difficile » ne signifie pas « grossier » ici
 - nous avons vu comment gérer les gens grossiers (cf chapitre 2)
- Ici, « difficile » désigne les personnes qui manipulent ou abusent des processus du projet
 - cela finit par coûter de l'énergie / du temps aux autres
 - cela n'apporte rien au projet
- Objectif des personnes difficile : se donner plus d'influence qu'elle ne devrait en avoir
 - plus insidieux que la grossièreté : comportement et dommages invisibles à l'observateur occasionnel
- Question : pouvez-vous me donner des exemples de ce genre de comportement ?

Personne difficile

- Exemple : supposons une discussion autour de la prise de décision sur une question ouverte.
→ **Question** : que peut faire une personne difficile et pourquoi ?

Personne difficile

- Exemple : supposons une discussion autour de la prise de décision sur une question ouverte.
 - **Question** : que peut faire une personne difficile et pourquoi ?
 - [quoi] continuer de prétendre que la question n'est pas résolue et proposer de nouvelles solutions / de nouveaux points de vue sur d'ancienne solution
 - [pourquoi] un consensus ou un vote semble imminent et la résultat attendu ne lui convient pas
- Exemple : supposons un débat qui ne converge pas vers un consensus mais où le groupe essaye de clarifier la situation, i.e. exprimer les points de désaccord et écrire un résumé qui servira de référence
 - **Question** : que peut faire une personne difficile et pourquoi ?

Personne difficile

- Exemple : supposons une discussion autour de la prise de décision sur une question ouverte.
 - **Question** : que peut faire une personne difficile et pourquoi ?
 - [quoi] continuer de prétendre que la question n'est pas résolue et proposer de nouvelles solutions / de nouveaux points de vue sur d'ancienne solution
 - [pourquoi] un consensus ou un vote semble imminent et la résultat attendu ne lui convient pas
- Exemple : supposons un débat qui ne converge pas vers un consensus mais où le groupe essaye de clarifier la situation, i.e. exprimer les points de désaccord et écrire un résumé qui servira de référence
 - **Question** : que peut faire une personne difficile et pourquoi ?
 - [quoi] retarder le résumé en :
 - compliquant la question sur ce qui devrait s'y trouver ou non
 - s'opposant à des suggestion raisonnables
 - introduisant de nouveaux éléments
 - [pourquoi] le résumé peut conduire a un résultat que ne plaît pas

Comment gérer les personnes difficiles

- Question : à votre avis, pourquoi ces personnes manipulent-elles cyniquement le système ?

Comment gérer les personnes difficiles

- Question : à votre avis, pourquoi ces personnes manipulent-elles cyniquement le système ?
 - sentiment d'être exclu des interactions et des décisions de groupe
 - impression de ne pas être pris au sérieux
 - impression de complot contre soit

=> sorte de paranoïa où la personne peut penser être la seule à se battre pour le projet
- Neutraliser une personne difficile demande du travail
 - besoin de preuves solides lorsque quelqu'un ne le remarque pas
 - besoin d'un certain doigté distribuer les preuves de manière réfléchie
 - cela peut avoir des effets secondes néfastes pour le projet si ce n'est pas bien fait

=> Si le mal est bénin, tolérez le pendant un certain temps

Comment gérer les personnes difficiles

- Si cela devient trop dommageable pour le projet:
 - **construire un dossier solide** (avec références aux archives publiques)
 - **communiquer en privé** pour savoir ce que les autres ont remarqués
 - objectif : recueillir des commentaires non filtrés avant de focaliser l'attention sur le fauteur de troubles
 - si vous n'êtes pas le seul à voir la personne difficile : **passer à l'action !**
 - ne pas accuser directement la personne de ce qui lui est reproché
 - risque de se cacher derrière une "agression injuste"
 - paraître raisonnable et soucieux du bien-être du projet
 - objectif : réformer le comportement de la personne ou la faire quitter le projet
- Note : il est important d'être précautionneux dans ce genre de cas !
 - Si vous n'avez pas de quoi appuyer vos propos, vous pourriez passer pour le « mauvais » de l'histoire

Cas d'étude

- Cas d'étude rencontré et décrit par Karl Fogel
- Historique et contexte :
 - projet SVN
 - listes de diffusions ouvertes au public
 - personne qui n'était pas impoli et cherchait sincèrement à être utile
- Problème : la personne ne savait pas quand envoyer / ne pas envoyer des courriers électronique
=> trop de « bruit » à cause de ces courriers trop fréquents et de ces questions sur divers sujets
- Tentative de solution : réponse invitant gentiment à faire des recherches plus poussées avant de poster
→ malheureusement c'est un échec
- La solution trouvée est un exemple parfait de la façon de construire un dossier solide sur des données quantitatives neutres (cf prochain slide)
- Question : vous, qu'auriez vous fait de plus que la première tentative ?

Cas d'étude

- **Solution apportée par un des développeurs du projet :**
 - fouilles dans les archives (courriers électroniques)
 - envoi de mails en privé à quelques développeurs
- **Le troisième nom sur la liste est notre contrevenant qui n'avait jamais fourni de code ou de documentation**

>>> In the last 25 days, the top 6 posters to the svn [dev|users] list have been :

>>> 294 kfogel | 236 "C. Michael Pilato" | 220 "J. Random" | 176 Branko Čibej | 130 Philip Martin | 126 Ben Collins-Sussman

>>> I would say that five of these people are contributing to Subversion hitting 1.0 in the near future.

>>>

>>> I would also say that one of these people is consistently drawing time and energy from the other 5, not to mention

>>> the list as a whole, thus (albeit unintentionally) slowing the development of Subversion. I did not do a threaded

>>> analysis, but vgrepping my Subversion mail spool tells me that every mail from this person is responded to at least

>>> once by at least 2 of the other 5 people on the above list.

>>>

>>> I think some sort of radical intervention is necessary here, even if we do scare the aforementioned person away.

>>> Niceties and kindness have already proven to have no effect.

>>>

>>> dev@subversion is a mailing list to facilitate development of a version control system, not a group therapy

>>> session.

>>>

>>> -Fitz, attempting to wade through three days of svn mail that he let pile up

Cas d'étude

- Remarques :
 - cas **classique** d'**abus des procédures** d'un projet
 - **pas d'actions évidente** (exemple : obstruction à un vote)
 - tire parti de la procédure d'auto-modération de la liste de diffusion
 - i.e. autorisation de publier n'importe quand et sur n'importe que sujet

=> **pas de règle violée sur laquelle s'appuyer ...**
- La solution apportée :
 - rassemble des preuves factuelles accablantes
 - envoie un "dossier" par mail pour avoir du soutien pour toute action qui suivrait
- Au final J. Random a été contacté et le problème lui a été exposé :
 - demande d'arrêter de poster sur les listes de diffusions
 - pas de compréhension de la situation par l'intéressé
 - fin des posts intempestifs et liste de diffusion de nouveau utilisable

V – Communications

- V.1 – Ce que vous écrivez vous définit
- V.2 – Quelques pièges communs (mauvais modèles de communication)
- V.3 – Gérer les personnes difficiles
- **V.4 – Gérer la croissance**
- V.5 – La publicité

Croissance d'un projet

- Question : qu'implique la croissance d'un projet sur sa communauté ?
- Question : et pour la communication ?

Croissance d'un projet

- **Question : qu'implique la croissance d'un projet sur sa communauté ?**
 - le nombre de personne cherchant des informations augmente drastiquement
 - le nombre de personnes fournissant des informations augmente plus lentement
- **Question : et pour la communication ?**
 - besoin d'évolution sur la façon dont le projet gère les communications !
- **Exemple de besoin d'adaptation : liste de diffusion pour les utilisateurs**
 - objectif de cette liste : lieu d'échange question / réponse ou des tiers peuvent absorber des connaissances via les échanges
 - problème : comment faire lorsque la liste comporte plus de quelques milliers d'utilisateur et centaines de messages par jour ?
 - le nombre de message dépasse la capacité de traitement d'un lecteur
 - **illustration** : une liste de diffusion pour les utilisateurs de **windows 10** qui est installé sur plus d'un milliard d'appareils ...
- **Ce problème lié à la croissance de la communauté s'applique aussi à IRC, aux forums de discussions en ligne ... tout système où un groupe entend des questions d'individus**
 - => Le modèle open source classique de support massivement parallélisé ne s'adapte pas aux grosses charges

Stratégies pour gérer la croissance

- L'inertie n'est pas une solution

- lorsque le point de rupture est atteint, il n'y a pas d'explosion

- les gens se désabonnent des listes / IRC

- pourquoi partent-ils ?

- trop de « bruit » pour être « écouté »

=> l'activité du forum va sembler rester gérable (compensation avec les départs)

- l'utilisation de modèle de communication non évolutif va tendre à faire diminuer la qualité des questions et des réponses

Question : pourquoi ?

Stratégies pour gérer la croissance

- **L'inertie n'est pas une solution**

- lorsque le point de rupture est atteint, il n'y a pas d'explosion
 - les gens se désabonnent des listes / IRC
- pourquoi partent-ils ?
- trop de « bruit » pour être « écouté »

=> l'activité du forum va sembler rester gérable (compensation avec les départs)

- l'utilisation de modèle de communication non évolutif va tendre à faire diminuer la qualité des questions et des réponses

Question : pourquoi ?

- les personnes expérimentées vont chercher ailleurs des réponses à leurs questions !
- les personnes inexpérimentées continuent de poster

- **Stratégies par ajustement :**

- 1) identifier les parties qui ne connaissent pas une croissance illimitée, contrairement au reste du forum, et séparez les du reste !
 - ne pas laisser le bon être entraîné par le mauvais !
- 2) proposer de nombreuses sources d'informations qui soient bien organisées, à jours et facile à trouver !

Première stratégie par ajustement

- La première stratégie n'est pas difficile et se fait souvent naturellement
- Par exemple :
 - au démarrage d'un projet on a souvent une liste de diffusion générale
 - discussion sur les problèmes de code, les nouvelles fonctionnalités, ...
 - après un certain temps, la liste a évolué en plusieurs sous-listes de sujets distincts

=> **identification de famille de sujets pour diviser la liste de diffusion en sous-listes séparées**
- Note : si le partitionnement est bien fait, les fils de discussions traversent rarement les frontières des familles de sujets
- Cette stratégie de séparation est décomposée en deux étapes :
 1. **créer une nouvelle liste / canal IRC / ...**
 2. **rappeler** aux contributeurs **d'utiliser la nouvelle liste** de manière appropriée
 - > l'étape 2. peut durer longtemps !
 - > lorsqu'un courrier est envoyé à la mauvaise destination, répondre de manière visible en indiquant la bonne destination (encourage les autres)

Seconde stratégie par ajustement : utilisation des archives de liste de diffusion

- La seconde stratégie est un processus continu
 - dure toute la vie du projet
 - implique de nombreux participants
- Question : les archives peuvent-elles être utilisées dans ce contexte ? Oui
 - les communications sont presque toutes archivées
 - une archive est enregistrée à une adresse particulière
 - => **utilisez les archives visiblement et autant que possible**
- Question : si vous avez la réponse, pourquoi citer l'archive ?

Seconde stratégie par ajustement : utilisation des archives de liste de diffusion

- La seconde stratégie est un processus continu
 - dure toute la vie du projet
 - implique de nombreux participants
- Question : les archives peuvent-elles être utilisées dans ce contexte ? Oui
 - les communications sont presque toutes archivées
 - une archive est enregistrée à une adresse particulière
 - => utilisez les archives visiblement et autant que possible**
- Question : si vous avez la réponse, pourquoi citer l'archive ?
 - **signale aux gens que les archives existent**
 - montre que la recherche dans les archives peut apporter des réponses
 - renforce la normal sociale de **non duplication de l'information**
 - => bonne pratique pour le projet**

Seconde stratégie par ajustement : utilisation des archives de liste de diffusion

- Question : pourquoi éviter de dupliquer l'information ?
 - facilite le fait de retrouver l'information si besoin (mémoire des mots clefs)
 - améliore le classement de l'information dans les moteurs de recherche Internet
 - Note : la duplication peut avoir du sens dans le cas où la réponse dans les archives n'est pas suffisante
 - Les archives sont un enregistrement de toutes les décisions prises dans le projet et de la manière dont elles ont été prises !
 - permettent de démarrer une discussion récurrente avec un résumé de l'état actuel
 - permettent d'anticiper les objections
 - permettent de préparer des réfutations
 - permettent éventuellement d'avoir de nouvelles idées
- => Étant donné ces avantages, l'utilisation des archives est normale ET attendu !
- Cela permet aux participants de voir par eux-mêmes que :
 - les précédentes communication n'ont menées nulle part
 - vous avez fait un travail préparatoire et que vous avez probablement quelque chose de nouveau à dire

Seconde stratégie par ajustement : généralisation de l'utilisation des archives

- Conseils précédents s'appliquent à d'autres informations du projet
 - la seule contrainte : informations à des adresses stables et facilement trouvables
 - Illustration avec la FAQ. Comment les gens utilisent-ils une FAQ ?
 1. Recherchent par mots / phrases spécifiques
 - la FAQ doit être disponible en format textuel
 - => pas de format PostScript, de document scanné, ...
 2. Flânent aléatoirement pour absorber des informations (sans cible en tête)
 - => pages conçus pour avoir une navigation facile et avoir une table des matières
 3. S'attendent à pouvoir accéder à la FAQ lors de recherches depuis des moteurs de recherches tels que Google, ...
 - => utilisation de pages HTML
 4. Réfèrent d'autres personnes sur des éléments spécifiques de la FAQ
 - > chaque entrée doit avoir une balise référençable
 5. Veulent pouvoir ajouter des nouveaux éléments à la FAQ (peu fréquent)
 - > fichiers sources disponibles et faciles à éditer
- Note** : le ratio lecture / écriture est très élevé

Seconde stratégie par ajustement : généralisation de l'utilisation des archives

- Le formatage de la FAQ n'est qu'un exemple sur la manière de rendre une source d'informations présentable
 - Les propriétés à retrouver sont :
 - **recherche directe**
 - disponibilité pour les principaux **moteurs de recherche** Internet
 - **capacité de navigation**
 - **stabilité référentielle**
 - (potentiellement) possibilité de modification
- => L'idée se généralise à d'autres pages Web, e.g. la documentation, le gestionnaire de bogues, ...
- Pour les sources d'informations différentes des fils de discussions, les logiciels associés risquent de ne pas assurer nativement les propriétés à retrouver
- => efforts supplémentaires à fournir de la part du responsable
- essayez de répartir cette charge de maintenance sur les participants !

V – Communications

- V.1 – Ce que vous écrivez vous définit
- V.2 – Quelques pièges communs (mauvais modèles de communication)
- V.3 – Gérer les personnes difficiles
- V.4 – Gérer la croissance
- **V.5 – La publicité**

La publicité dans l'open source

- Question : à votre avis, pourquoi est-il difficile de gérer la publicité dans l'open source ?

La publicité dans l'open source

- **Question : à votre avis, pourquoi est-il difficile de gérer la publicité dans l'open source ?**
 - le public cible n'est pas toujours celui qui reçoit l'information (les messages sont accessibles à tout le monde)
 - => le projet n'a pas un contrôle total sur l'impression que le monde obtient
- **Exemple de situation dans laquelle cela peut se produire :**
 - un article sur Hacker News <https://news.ycombinator.com/>
 - un article Slash Dot <https://slashdot.org/>

=> attire l'attention de millions de personnes sur un article du projet (que personne n'attendait à voir en dehors du projet)
- **Note : ce genre d'événement se produit rarement en pratique**
 - normalement, les annonces les plus médiatisées sont celle que vous souhaitez faire connaître (si vous vous débrouillez bien)

=> Il est important d'avoir les bons mécanisme pour susciter l'intérêt du monde extérieur

Annnonce des sorties et événements majeurs

- Les annonces majeures doivent être réalisées simultanément sur vos canaux de distribution principaux !
- Question : Quels sont-ils ?

Annonce des sorties et événements majeurs

- Les annonces majeures doivent être réalisées simultanément sur vos canaux de distribution principaux !
- Question : Quels sont-ils ?
- 1. La page d'accueil de votre projet ! (ex : <https://git-scm.com/>)
 - probablement l'endroit le plus vu parmi tous ceux de votre projet
 - pour une annonce importante, ajoutez un texte de présentation bref qui renvoie à un texte plus détaillé
- 2. La page/zone "Actualités" ou "Communiqués" sur le site web de votre projet
 - annonce unique rédigée en détail
 - objectif : fournir un objet sur lequel d'autres sites peuvent pointer
 - => besoin d'une certaine structuration, par exemple:
 - une page Web par communiqué
 - une entrée de blog distinct
 - tout autre format qui permettant d'être référencé et garantissant la distinction avec les autres communiqués du même domaine

Annonce des sorties et événements majeurs

- 3. Les comptes Twitter pertinents / channels d'information / flux RSS (https://twitter.com/github?ref_src=twsrc%5Egoogle%7Ctwcamp%5Eserp%7Ctwgr%5Eauthor)
- 4. (potentiellement) sur les forums (ex : <https://www.developpez.net/forums/> , <https://forums.codeguru.com/>)
- 5. La liste de diffusion d'annonce de votre projet
 - convention de nommage : `announce@nomdomaineprojet.org`
 - généralement la liste de diffusion la plus suivie de votre projet
 - raison : se limite aux annonces importantes du projet
 - monté de version du logiciel
 - découverte d'une faille de sécurité
 - changement dans la direction du projet, ...

Annnonce d'événements moins important

- Question : quels canaux de communication utiliser / ne pas utiliser ?

Annnonce d'événements moins important

- Question : quels canaux de communication utiliser / ne pas utiliser ?
- N'utilisez aucun ou presque aucun des canaux de distribution cités précédemment
 - même s'il n'utilise pas ces canaux, l'événement sera quand même remarqué par le monde extérieur en proportion de son importance

Exemple : le fait de fixer la date de la prochaine version

→ digne d'intérêt mais moins important que la version elle-même

=> envoyez un mail aux listes de diffusion quotidienne (!= de la liste d'annonce) et mettez à jour la chronologie du projet / page Web d'état, pas plus

- Note : ne pas spécialement chercher à médiatiser un événement ne l'empêchera pas de l'être
 - apparition dans des discussions avec des personnes intéressées par le projet
 - le bouche à oreille donne une très large diffusion
 - => il est intéressant d'en tirer parti
 - => il est important de construire les annonces de manières à faciliter la transmission informelle précise !

Annnonce d'événements moins important

- Exemple d'annonce facile à transmettre :

>>> Juste une mise à jour : nous prévoyons de publier la version 1.0 de Elo à la
>>> mi-octobre 2021. Vous pouvez toujours consulter <http://www.elo.org/status.html>
>>> pour les mises à jour. La principale nouveauté sera l'intégration du versionnement
>>> des workflows et le partage de ceux-ci.
>>>

>>> Les autres nouvelles fonctionnalités incluent : ... Il y aura également diverses
>>> corrections de bugs, notamment : ...

→ le premier paragraphe est court et donne les deux informations les plus importantes (date et nouveauté majeure) ainsi qu'une URL pour visiter d'autres nouvelles

=> il est facile que ce paragraphe traverse l'écran de quelqu'un d'autre (point très positif)

→ le second paragraphe peut être "perdu" sans affecter l'essentiel du contenu

→ dans le meilleur des cas les gens donneront un lien vers l'intégralité du courrier (c'est rare)

Annnonce des failles de sécurité

- Question : à votre avis, comment doit-on gérer l'annonce d'une faille de sécurité ?

Annnonce des failles de sécurité

- Question : à votre avis, comment doit-on gérer l'annonce d'une faille de sécurité ?
 - La gestion des vulnérabilité de sécurité est différentes des autres type de bogue
 - dans l'open source, faire les choses de manière ouverte et transparente est normalement un credo religieux
 - la gestion standard d'un bogue est visible pour tous et consiste en :
 1. l'arrivée du rapport initial
 2. la discussion associée au bogue
 3. le correctif éventuel
 - la gestion d'un bogue de sécurité est différente car il peut compromettre les données des utilisateurs / l'ordinateur des utilisateurs
- => discuter ouvertement sur ce bogue n'est pas possible !**
- cela reviendrait à annoncer l'existence du bogue au monde, y compris les parties malveillantes
 - même le simple fait de valider un correctif annonce l'existence d'un bogue et annonce aux parties malveillantes quelles peuvent attaquer le code pré-correctif

Annnonce des failles de sécurité

- La gestion de ce problème est à peu près standard dans les projets open source et consiste à suivre les directives :
 1. **ne pas parler du bogue publiquement** jusqu'à la création d'un correctif, puis distribution du correctif en parallèle de l'annonce du bogue
 - moyen de distribution judicieux : empaquetage en tant que version ou validation dans le référentiel public du projet
 2. **trouver le correctif le plus vite possible !**
 - urgent lorsque le signalement du bogue est fait par un non contributeur
 - une personne en dehors du projet est capable d'exploiter la vulnérabilité !
- Voyons comment ces principes sont réalisés en pratiques

La réception du rapport de la faille de sécurité

- Question : quel mécanisme de soumission mettre en place ?

La réception du rapport de la faille de sécurité

- Question : quel mécanisme de soumission mettre en place ?
- Les canaux classiques de rapport de bogue ne peuvent pas être utilisés (car public)
=> besoin d'une liste de diffusion / formulaire de contact séparé
- Ce forum doit:
 - **ne pas avoir d'archives publiquement lisibles**
 - **avoir un système d'abonnement contrôlé**
 - développeurs de longue date et de confiance (par exemple droit de commit depuis N année(s))
 - personnes qui ont la confiance des développeurs
 - => des personnes fiables (bonne intention & capacité à ne pas divulger d'informations)
- Idéalement, cette passerelle de rapport ne doit pas être modérée ou protégée contre le spam
 - risque de filtrage d'un rapport / retard car un modérateur n'était pas disponible
- Le mécanisme de soumission peut être décrit dans un fichier du code source, e.g. SECURITY
<https://github.com/spack/spack/blob/develop/SECURITY.md>
- Le mécanisme de soumission doit être sécurisé
 - utilisation d'une page <https://...> (protégée par TLS)
 - recours à une adresse email : avoir une clé publique bien annoncée pour envoyer des courriers cryptés à cette adresse

Développement du correctif

- Question : quelle est la première tâche à réaliser après la réception du rapport ?

Développement du correctif

- Question : quelle est la première tâche à réaliser après la réception du rapport ?
 - évaluation de la gravité / urgence du problème
- Pour cela il est important de répondre aux questions:
 - quelle est **la gravité de la situation** par rapport à la faille ?
 - permet-elle de prendre le contrôle de l'ordinateur de votre utilisateur ?
 - permet-elle "juste" de récupérer des informations "simples" comme la taille de certains fichiers ?
 - est-ce que la faille est-elle **facilement exploitable** ?
 - est-ce que la faille peut être scénarisée, i.e. reproduite facilement ?
 - nécessite-telle des connaissances circonstanciées / déduction / chance ?
 - **qui** vous a signalé le problème ?
 - la réponse ne change pas le problème mais une estimation de sa diffusion
 - si le rapport vient d'un développeur, vous pouvez respirer un peu plus
 - si le rapport vient d'un anonyme, i.e. random12@hackerz.net, il faut vite agir car aucune garanti sur la propagation de la vulnérabilité

Développement du correctif

=> Répondre à ces questions va vous permettre d'avoir une fourchette d'urgence du besoin d'avoir un correctif

→ pas urgente si le bogue ne compromet pas très gravement la sécurité

→ urgente / très urgente autrement

→ le fait que cela soit rapporté par un développeur n'empêche pas que la faille soit déjà connue et exploité par des tiers

- L'exemple du rapport par random12@hackerz.net peut réellement se produire

→ personne qui aura l'impression de vous avoir joué un tour

→ il est important de jouer le jeu, i.e. répondre en remerciant, prévenir d'une date de correctif, tenir le rapporteur au courant, ...

=> traitez ces rapporteurs correctement et espérez qu'ils en fassent autant

Développement du correctif

- Le rapporteur peut aussi être un professionnel de la sécurité, i.e. auditeur de bogue
 - donnera souvent un délai pour corriger la vulnérabilité avant de la rendre publique (ce que peut aussi demander random12@hackerz.net)
- Question : pourquoi imposer un délai de correction ?

Développement du correctif

- Le rapporteur peut aussi être un professionnel de la sécurité, i.e. auditeur de bogue
 - donnera souvent un délai pour corriger la vulnérabilité avant de la rendre publique (ce que peut aussi demander random12@hackerz.net)
- Question : pourquoi imposer un délai de correction ?
 - seul moyen fiable d'amener une organisation à résoudre rapidement un problème de sécurité
 - le délai peut être négociable mais c'est le rapporteur qui détient toutes les cartes
 - => la date limite n'est pas "impolie", c'est une tradition séculaire
- Notes :
 - Le niveau d'urgence vous donnera un indicateur sur la réalisation du correctif (élégance vs rapidité)
 - Ne validez pas le correctif dans un référentiel public avant la date prévue (risque d'être vu et exploité par des tiers)

Notions relatives à la vulnérabilité

- Identifiant CVE ("Common Vulnerabilities and Exposures") : item dans la liste publique de failles de sécurité informatique (<https://cve.mitre.org/>)
 - nom standardisé et unique, e.g. CVE-2021-0160, pour référencer une faille
 - centralise des informations relatives à la faille:
 - ne contient pas d'information détaillée du bogue et de comment s'en prémunir
 - contient un résumé et des références à des ressources externes
 - les identifiant CVE sont sous silence jusqu'à l'annonce de la publication
 - <https://cve.mitre.org/cgi-bin/cvename.cgi?name=2021-0042> (sous silence)
 - <https://cve.mitre.org/cgi-bin/cvename.cgi?name=2021-0012> (publiée)
- Question : nous avons vu qu'il est important de mesurer la gravité d'une faille mais comment évaluer la gravité associée ?

Notions relatives à la vulnérabilité

- Identifiant CVE ("Common Vulnerabilities and Exposures") : item dans la liste publique de failles de sécurité informatique (<https://cve.mitre.org/>)
 - nom standardisé et unique, e.g. CVE-2021-0160, pour référencer une faille
 - centralise des informations relatives à la faille:
 - ne contient pas d'information détaillée du bogue et de comment s'en prémunir
 - contient un résumé et des références à des ressources externes
 - les identifiant CVE sont sous silence jusqu'à l'annonce de la publication
 - <https://cve.mitre.org/cgi-bin/cvename.cgi?name=2021-0042> (sous silence)
 - <https://cve.mitre.org/cgi-bin/cvename.cgi?name=2021-0012> (publiée)
- Question : nous avons vu qu'il est important de mesurer la gravité d'une faille mais comment évaluer la gravité associée ?
 - difficile de l'évaluer avec précision
 - => il ne faut pas improviser un langage pour essayer de l'exprimer !
- Le système de notation CVSS ("Common Vulnerability Scoring System") <https://nvd.nist.gov/cvss.cfm>
 - norme utilisée pour évaluer la gravité d'une vulnérabilité

Pré-notifier avec de rendre publique

- Question : comment distribuer le correctif ?

Pré-notifier avec de rendre publique

- Question : comment distribuer le correctif ?
 - approches possibles : appliquez le correctif dans votre référentiel, annoncez le correctif au monde, ...
 - => force les utilisateurs à se mettre à jour immédiatement ou ils risquent d'être piratés
- Pré-notification pour certains utilisateurs importants
 - controversées car privilège des utilisateurs par rapport à d'autres
 - Question : qu'en pensez vous ?

Pré-notifier avec de rendre publique

- Question : comment distribuer le correctif ?
 - approches possibles : appliquez le correctif dans votre référentiel, annoncez le correctif au monde, ...
 - => force les utilisateurs à se mettre à jour immédiatement ou ils risquent d'être piratés
- Pré-notification pour certains utilisateurs importants
 - controversées car privilège des utilisateurs par rapport à d'autres
 - Question : qu'en pensez vous ?
 - perso : bon choix dans certaines circonstances, surtout lorsque des services en lignes connus utilisent votre logiciel
 - doit être réalisée en contactant en privé les administrateurs avant la date de publication, en outre le destinataire doit :
 - travailler sur un service important où une faille serait un problème sérieux
 - être connu comme quelqu'un qui ne parlera pas du problème avant la publication
 - disposer d'un moyen de communication sécurisé
- Note : Si vous envoyez un courrier électronique pour pré-notifier, CRYPTÉZ-LE !

Exemple de pré-notification

- Voici un exemple de courrier de pré-notification (sur 2 slides) :

>>> De : Votre nom ici

>>> À : admin@service-important.com

>>> Répondre à : Votre nom ici (pas l'adresse de la liste de sécurité)

>>> Objet : Notification importante confidentielle.

>>>

>>> Cet e-mail est une pré-notification confidentielle d'une alerte de sécurité dans le logiciel mon-super-projet.

>>>

>>> Veuillez ne transmettre aucune partie de ce courrier à qui que ce soit.

>>> L'annonce publique n'est que le 12 décembre, et nous aimerions maintenir l'information sous silence jusque-là.

>>>

>>> Vous recevez ce courrier parce que (nous pensons) que vous utilisez mon-super-projet et je voudrais qu'il soit

>>> corrigé avant que cette faille de sécurité ne soit rendue publique.

>>>

>>> Références

>>>

>>> CVE-2020-0012 : Débordement de mon-super-projet dans les requêtes

Exemple de pré-notification

>>> Vulnérabilité

>>>

>>> Le logiciel peut être amené à exécuter des commandes arbitraires si les paramètres régionaux du serveur sont mal

>>> configurés et que le client envoie une requête mal formée.

>>>

>>> Gravité

>>>

>>> Note de base CVSSv3 : 9,0

>>> Vecteur de base CVSSv3 : AV:N/AC:L/Au:N/C:C/I:C/A:C

>>>

>>> Solutions de contournement

>>>

>>> Définir l'option « traitement du langage naturel » sur « désactivé » dans mon-super-projet.conf ferme cette vulnérabilité.

>>>

>>> Pièce

>>>

>>> Le correctif ci-dessous s'applique à mon-super-projet 1.0.1, 1.0.2

>>>

>>> La nouvelle version publique 1.0.3 sera publiée juste avant le 12 décembre, afin qu'elle soit disponible en même temps que

>>> cette vulnérabilité est rendue publique. Vous pouvez patcher maintenant, ou attendez simplement la sortie publique. Le seul la

>>> différence entre la version 1.0.2 et 1.0.3 sera ce patch.

Distribution publique du correctif

- Faire une annonce unique et complète :
 - décrire le problème
 - donner le **numéro CVE** associé (s'il y en a un)
 - décrire comment contourner et résoudre le problème
- « Réparer » le problème signifie usuellement une mise à jour du logiciel, mais cela peut aussi signifier l'application d'un correctif
 - si distribution par montée de version, celle-ci doit différer de la version précédent par exactement le correctif de sécurité, rien d'autre !
 - rassure les administrateurs conservateurs qui pourraient avoir peur des effets de bord
- L'annonce du correctif doit être réalisée comme l'annonce d'une nouvelle version, i.e. courrier aux listes de diffusions concernées, communiqué sur le site principal, ...
 - il est important de mesurer son ton lors de l'annonce du correctif et de la faille, e.g. évitez de faire trop d'agitation si la faille est mineure