

2020年4月28日
株式会社インプレスR&D
<https://nextpublishing.jp/>

一歩リードしたエンジニアになる！
『これでできる！はじめてのOSSフィードバックガイド』発行
技術の泉シリーズ、5月の新刊

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『これでできる！はじめての OSS フィードバックガイド』(著者:結城 洋志)を発行いたします。

最新の知見を発信する『技術の泉シリーズ』は、「技術書典」や「技術書同人誌博覧会」をはじめとした各種即売会や、勉強会・LT 会などで頒布された技術同人誌を底本とした商業書籍を刊行し、技術同人誌の普及と発展に貢献することを目指します。

『これでできる！はじめてのOSSフィードバックガイド』
<https://nextpublishing.jp/isbn/9784844378815>



著者:結城 洋志
小売希望価格:電子書籍版 1800 円(税別)／印刷書籍版 2000 円(税別)
電子書籍版フォーマット:EPUB3／Kindle Format8
印刷書籍版仕様:B5 判／カラー／本文 158 ページ
ISBN:978-4-8443-7881-5
発行:インプレス R&D

<<発行主旨・内容紹介>>

「OSS にフィードバックしてみたい人の背中を押すワークショップ」として日本全国で計 70 回近く開催されている「OSS Gate ワークショップ」。

その中で参加者の方から実際に寄せられた質問への回答を元に、コミュニケーションの取り方や 이슈報告・プルリクエストのマナー、取り込まれやすい変更のコツなど、今まであまり言語化されてこなかった「OSS 開発者・OSS

コントリビューターが日常的にしていること」を幅広く解説します。OSS にフィードバックなんて自分には無理、と思っている方のための1冊です。

(本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。)

OSS へのフィードバックについての認識をまず改める




図1-1: スポットライトが当たるのはごく一部だけ

ただ、働き方や仕事の内容・ライフスタイルなどは様々でも、OSSに関わる人には共通する「すごさ」があると筆者は考えられます。それは、**問題の根本的な解決を図りたいと思っていて、そしてその想いを実現するために行動している**、という点です。

僕たちは根本的な解決を図りたい

OSSを使っている問題に遭遇したとき、「不具合に遭遇した……どうやって回避しよう」と思う人は少なくないでしょう。

でも、ちょっと待ってください。**回避とは本来、問題を根本的には解決できないときの次善の策のほうです**。根本的に解決できるなら、それに越したことはないですね。

真→先に回避策を探りたくなるには、色々な理由があると思います。

- ・未知のことに挑戦するのが恐ろしい。
- ・納期が厳しいのに、失敗したら時間を無駄にしてしまう。
- ・素人が関わることで迷惑をかけてしまう気がする。
- ・プロの仕事に文句を付けるなんて失礼な気がする。
- ・問題の解決はプロジェクトオーナーの責任だと思う。

でも、突き詰めて考えると、これらのための根っこのには、**自分がそこに関わっていいと思っていない**という認識があるのではないのでしょうか。

少なくとも筆者はそうでした。自分はただユーザーで、そのOSSを作っているのはすごい人達だ。ユーザーの自分は使うだけで、直すのはすごい人達の領域だ。ただのユーザーに過ぎない自分には、すごい人達のようなことはできない。無意識のうちにそう考えていました。

ですが、先に述べたとおり、今OSSの開発に関わっている人達も多きは「普通の人間」です。し

かも、OSSは多くのプロプライエタリな製品とは違ってソースコードが手に入ります。さらに、「フィードバックはこちらへどうぞ」と案内して下さいますので、「文句を言ったら失礼かも……」などと付添する必要もありません。**必要な物は揃っていますし、許可だってしてくれている**のですから、自分にも、あなたにも、やってみようという気持ちはあります。

そこで勇気を持って、そのOSS自体を修正して成果を還元したり、情報を提供したり、といった行動を取ること。それが「OSSへのフィードバック」の実態です。**問題を根本的に解決するために地道な行動を取ったら、結果としてOSSへのフィードバックとなっていたという、ただそれだけのことなのです**。

大丈夫、有用なフィードバックを行うための能力は後から付いてきます。必要なのは、最初の一歩を踏み出すという思い切りだけです。

1.2 OSSにフィードバックすると何が嬉しいの？

実利的なメリットもたくさんある

理想論には興味が無い、現実的なものにだけ関心がある、という方もいらっしゃるでしょう。本書は基本的に、「OSSにフィードバックしてみたい」という動機をすでに持っている方を想定して書かれています。しかし、「会社で奨励されているのでとりあえずやってみよう」としているようなケースでは、それほど強い動機がない場合もあります。そういう方のために、「**問題が解決されること**」以外の、OSSにフィードバックすることのメリットを改めて列挙してみます。

個人としてOSSに関わることのメリットで、まず一番分かりやすいのは、「**すごい人だ**」と思ってもらえるという点でしょうか。

実際にやったフィードバックの内容はそれほど高度でないとしても、「OSS開発者」「OSS開発に関わった」というだけで「よく分からんが、何だかすごそうだ」と思ってくれる人はまだまだいらっしゃるでしょう。ハッキリだとしても、それが仕事を任せられるきっかけになることもあれば、自分自身にとっても**新しいことに挑戦する自信**になるでしょう。

……というのは半分くらい冗談ですが、「**ちゃんと技術が分かる人**」「**優秀なITエンジニア**」からの信用を得やすいということは実際あります。

実績が見えないのになぜかITイベントでの登壇が多い有名な人を擁護する「IT芸人」なんて言う方がいますが、OSSにフィードバックしたという事実は間違いなく「目に見える実績」ですし、その内容からは後述するような様々な情報を読み取れます。「**この人は口先だけじゃない**」と信用してもらえらる材料を積み重ねる手段として、OSSへのフィードバックは非常に有効です。

口先だけではない**実力**とはどういうものを言うか。たとえば、OSSへのフィードバックを重ねると、**仕事の上で必要になるものも共通の技能**が身に付きます。「修正作業の中でプログラミング・

3 「開発者」は「信頼される」という意味で、「誰でも自由にアクセスできる」と対照して使われる形です。GitHubの用語集「Open Source Management」の「開発者」の項目を参照してください。エンジニアリングの分野では、開発者という用語は、開発者として活躍している人（プロプライエタリソフトウェアの開発者）を指します。

技術的な内容以前のハードルとなる言語の壁についても丁寧に解説

第6章 英語での報告の仕方が分からない

ここまで、フィードバック対象の見つけ方、フィードバックする内容の見つけ方、報告の組み立て方を解説してきました。しかし、それらの技術的な内容以前のハードルとして、**言語の壁**という物もあります。

実際にOSS Gateワークショップでも、フィードバック内容を整理することはできても、それを英語で表現することに手こずっておられる方がかなり多い印象があります。OSSでは英語が共通の言語として使われている場合が非常に多く、このこと自体が日本語を母語とする人のOSSプロジェクトへの参加障壁になっているのは事実です。

ITエンジニア向けに「こういう英語表現を覚えよう」という情報を紹介する記事は時々見かけます。ですが、ビジネス参加者の様子を実際に見ている限りは、必要なのは「実際の現場でよく使われる単語や熟語の慣習」ではなく、「実際の現場で英文を書くときの**考え方の解説**」であるように筆者には感じられました。

そこでこの章では、OSSへのフィードバックを英語でやるときの「考え方」に焦点を当てて解説してみます。対象読者は、英語に苦手意識のある方、中学校の英語の授業で英作文はしたことがあるけれども自分でゼロから何かを説明する英文を書いたことはない方、といったレベル感を想定しています。

6.1 むしろ英語の文を書きすぎない方がいい！

冒頭からちょっとお話を進しますが、OSSへのフィードバックする際には、**英語を自由自在に扱えないといけない、ということはありません**。

前章でも述べた通り、言語での説明が難しいところは図や画像や動画を使う方法もあります。実際に動作するプログラムやゲータファイル、あるいは仮想環境そのものを用意する手もあります。とにかく、そのときの自分に使えるあらゆる手段を柔軟に使って、**伝えたいことをどんな形でもいから伝える**のが一番大切なことです。

これは何度も強調しておきたいのですが、英語で書くことも、文章を書くことも、手段であって目的ではありません。**OSSプロジェクトのイシュートラッカーは「美しい英文を披露して競う発表の場」や「英語スピーチ大会の場」ではないのです**。

筆者自身もOSSを公開してフィードバックを受け立場ですが、その実感から半面なことを言う。筆者が英語が不得意だからということを知り引いても、**流暢な英語で、複雑なことを要点を踏まえないまま長くと説明されても、開発者としては全然嬉しくありません**。それだったら、再現手順の特定をもっと面倒とか、せめて図のひたつで付けるかしてくれたい方がありがたい、と思うことが多々あります。

そう、意外かもしれませんが、**OSSに関わっている人の全員が全員英語が得意とは限りません**。

OSSの開発者は中国人もいれば韓国人もいるし、ロシア人やインド人もいます。「英語が母語である」人はむしろ少数派かもしれませんが、「英語で説明されること」自体が、情報の伝達を妨げる障害にもなり得るのです。

実際の所、前章で述べた

- ・再現手順 (steps to reproduce)
- ・実際に得られた結果 (actual result)
- ・期待される結果 (expected result)

という**基本的3要素を簡潔書きにするだけで、報告としての体裁も充分に整っています**。

具体的な再現手順を書くときも、GUIの操作は「Click XXX (XXXをクリックする)」「Input XXX (XXXを入力する)」といった簡潔な表現で充分ですし、コマンドライン操作なら、実行したコマンドや出力結果をそのままコピーペーストするだけいいです。あとは最初か最後一言「Is this intentional? (この結果は意図的なものですか?)」とでも添えれば十分でしょう。誤解を懸念すれば、**OSSへの報告は、英語の文章以外の形で具体的な説明が多いほどよく、英語の文章での抽象的な説明は少ないほどよい**、というのが筆者の持論です。

「英語での報告の仕方」に困ったら、冷静になって思い出してください。皆さんがしたいことは、「英語の文章を書くこと」ではなく、「**不具合を伝えること**」や「**要望を伝えること**」だったはずですよ。画面から一度目を離して深呼吸すれば、英語以外の何か別の手段を思い浮かべられるかもしれません。

とはいえ、英語の文章での説明が必要な場面も現実にはたくさんあります。なので、ここからは、どうしても英語を使うことを避けられないとなったときのために、ゼロからすらすら英語の文章を書けるようになるためのコツをお伝えしていこうと思います。

6.2 「日本語の文章を英語に変換する」のではなく、「同じことを説明する英語の文章を作る」と考えよう

英作文に不慣れな人は、まず「日本語の文章」を考えて、それを「対応する内容の英語の文章」に変換するものだと考えがちなのではないでしょうか。

いきなり断言してしまうのですが、**それは無理なので、潔く諦めてしましましょう**。

外出先から帰宅した人が在宅していた人に言う「ただいま」や、それに応える「おかえりなさい」は、**英語に翻訳しようがない**とよく言われます。そもそも欧米には「帰宅した時に挨拶を交わす」文化が無いので、そういう言葉どころか概念が無いのが根本です。この例をひとつとってみても、「日本語で書いた文章をきちん英語に変換する」のは本来的には不可能だということが分かるのではないのでしょうか。

ではどうするかといえば、「**伝えたい内容、対象、そのものの方に立ち戻って、それを表現する英語の文章を考える**、ということになります」。

たとえば、ここに「日本のペンがある」とします。その状態を英語で表すには、どんな文章が考えられるでしょうか？

第13章 バグ報告やパッチ提供以外の形での OSS 開発への参加

ここまで、イシュー報告とプルリクエストに焦点を当てて、OSS へのフィードバックの仕方を色々解説してきました。
ですが、OSS へは他にもさまざまな関わり方があります。本章では周辺情報として、それらも併せて簡単に紹介しておきます。

13.1 翻訳への参加

未訳を見つけたらフィードバックのチャンス

OSS を使っていて、UI がまるっきり英語だったり、あるいは、ほとんどの部分は日本語化されているのに一部の項目だけ英語だったり、という状況に遭遇したら、翻訳に参加するチャンスです。

図 13.1: 一部のメニュー項目が英語のままの様子 (GIMP)



GUI を持つ OSS の多くは英語を基本の表示言語にしていて、それ以外の各言語への翻訳（ローカライズ）は、その言語を母語とする人による協力頼みとなっています。善意の協力者の人が翻訳してくれても、その人が何らかのなんらかの事情で活動できなくなってしまうと、追加された機能や変更された機能の部分が未訳のまま取り残されてしまう、ということはよくあります。
小規模な OSS プロジェクトでは、翻訳はフィードバックの種類の中でもハードルが低い方だと言えます。ロジックに対する変更のプルリクエストでは、プロジェクトの方針に則っているかのレビューがあるのに対し、翻訳リソースに対するプルリクエストは、大抵は作者がその翻訳の内容を理解できないので、ほぼノーチェックの素通しとなることが多いからです。

翻訳用システムを使った翻訳

ただ、プロジェクトによっては、複数人でひとつの言語のリソースを分担して翻訳するために、「翻訳用システム」を使っている場合があります。

たとえば、ownCloud やその派生の Nextcloud が使っている Transiflex というサービスでは、既訳・未訳部分の管理や、訳語の統一のための辞書機能、翻訳結果の相互レビューなど、各人が作業に集中しつつ翻訳の品質を高めるための機能を提供しています。

同様のサービスは Crowdin、GitLocalize、BabelZilla などいくつかあります。リポジトリ管理における GitHub ほどには「デファクトスタンダード」と言える一人勝ちのプレイヤーは決まっていな様子で、それぞれサービスごとの使い勝手の差異も大きいので、本書では、特定のサービスの具体的な使い方は詳述しないことにします。

翻訳作業への協力の仕方は、その OSS プロジェクトへの参加方法の案内の中に「翻訳 (localize) に参加したい場合はこちらから」のように記載されていたり、あるいは、コミュニティ紹介の中で「開発者コミュニティ (Developers Community)」とは別に「翻訳コミュニティ (Translation Community)」として紹介されていたりします。案内が見つからない場合、とりあえずリポジトリに普通にプルリクエストの形で言語リソースの修正を提案した上で、「翻訳に協力する際のルールはありますか? (Is there any guideline for translation?)」のように質問すれば、適切な情報に誘導してもらえるでしょう。

機械翻訳の結果は使わない方が安全

第6章「英語での報告の仕方が分からない」でも同様のことを述べましたが、元の英語リソースの内容を Google 翻訳などの機械翻訳にかけた結果を、そのまま翻訳作業の成果としてフィードバックするのは避けましょう。

理由のひとつは、訳文の権利の問題です。

機械翻訳のサービスやツールによっては、結果の訳文の二次利用や商用利用を規約で禁止している場合があります。OSS は商用利用を禁止しないため、そのような禁止事項がある訳文を OSS にフィードバックしてしまうと、その OSS プロジェクト全体が「翻訳サービスやツールの規約に違反した」ことになってしまいます。実際に、Ubuntu コミュニティなどで、Web 翻訳の結果を OSS に不意に入れた人がいたために、その人が過去関わった全ての翻訳済みリソースについて権利面での妥当性を再確認せざるを得なくなった、という事例もあります。

また、もうひとつの理由として、機械翻訳は文脈を汲み取ってはいくれないという問題もあります。OSS の UI に登場する文は断片的なフレーズが多く、そのまま機械翻訳にかけた結果を埋め込むと、UI 上で意味をなさないおかしな箇所が発生しやすいです。たとえば、Firefox には「Reload Tab」というコマンドがありますが、Google 翻訳で単に英和翻訳すると「再読み込みタブ」と訳されるため、そのまま UI 上に反映すると非常に違和感がある状態になってしまいます (実際の日本語版では「タブを再読み込み」と訳されています)。

1: 「Tracking Protection」という機能も「トラッキング保護」と訳すが「行動履歴の記録からの保護」と訳すか、というように、訳し手の判断が翻訳によっておきかえられていること。

<<目次>>

- 第1章 本当に「誰でもできる」の?
- 第2章 OSS を探そう
- 第3章 何をフィードバックしたらいいのかわからない
- 第4章 どこにフィードバックしたらいいのかわからない
- 第5章 どう報告すればいいのかわからない
- 第6章 英語での報告の仕方がわからない
- 第7章 それでもためらってしまうあなたへ
- 第8章 プルリクエストしてみたい!
- 第9章 他の人のフィードバックから学ぼう
- 第10章 要望が通らない!
- 第11章 プルリクエストをマージしてもらえない!
- 第12章 仕事が忙しくて OSS 活動に時間を割けない!
- 第13章 バグ報告やパッチ提供以外の形での OSS 開発への参加
- 第14章 ライセンスについてより正確に知る

<<著者紹介>>

結城 洋志

1982 年生まれ、大阪出身。自作イラスト発表の場として Web サイトを運営する中で、Web 標準技術への傾倒から Mozilla に関心をもち、「Tree Style Tab」など Mozilla Firefox 用のアドオンを多数開発するようになる。現在は株式会社クリアコードにて、主に Firefox・Thunderbird の法人向け技術サポートに従事。また、日経 Linux 誌(日経 BP)にてシェルコマンド操作の解説漫画「シス管系女子」を連載中。

<<販売ストア>>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple Books、紀伊國屋書店 Kinopyy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしだい開始されます。

※ 全国の一般書店からもご注文いただけます。

【インプレス R&D】 <https://nextpublishing.jp/>

株式会社インプレスR&D(本社:東京都千代田区、代表取締役社長:井芹昌信)は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知の流通を目指しています。

【インプレスグループ】 <https://www.impressholdings.com/>

株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「旅・鉄道」「学術・理工学」を主要テーマに専門性の高いメディア&サービスおよびソリューション事業を展開しています。さらに、コンテンツビジネスのプラットフォーム開発・運営も手がけています。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

TEL 03-6837-4820

電子メール: np-info@impress.co.jp