

本サービスにおける著作権および一切の権利はアイティメディア株式会社またはその情報提供者に帰属します。また、本サービスの出力結果を無断で複写・複製・転載・転用・頒布等することは、法律で認められた場合を除き禁じます。

「英語に愛されないエンジニア」のための新行動論(6):

一実践編(パラダイムシフト)——技術英語はプログラミング言語である

<http://eetimes.jp/ee/articles/1208/03/news049.html>

本来、「言語」とは「人間相互理解」の道具であり、「民族」という単位のポピュラーな指標であり、「文化」という荷物を過去から未来へ運ぶリヤカーのような役割もあります。しかし、「技術英語」には、そのいずれの機能もありません。「技術英語」が、言語でなく、「英語」の下位概念ですらないとすれば、それは一体何でしょうか。「技術英語」とは、「図」を構成要素とする「プログラミング言語」です。

2012年08月09日 08時00分 更新

[江端智一, EE Times Japan]

われわれエンジニアは、エンジニアである以上、どのような形であれ、いずれ国外に追い出される……。いかに立ち向かうか?→「[『英語に愛されないエンジニア』のための新行動論](#)」連載一覧

「英語に愛されないエンジニア」である私たちが立ち向かわなければならないものは、「英語そのもの」ではありません。英語という地平線まで広がる大草原の中にある小さな家のようなものです。それは、「技術英語」と呼ばれるものです。

「技術英語」の領域は、ものすごく「小さい」です。どのくらい小さいかというと、主観的には図1に示す程度です。そもそも、技術英語では人間が登場しません。「ハイ、キャシー!どこに行くんだい?」、「あら、トム。今日はマイクとデートよ」という会話だけでも、既に3人が登場しますが、技術英語では、トムとマイクがけんかを始めてキャシーが「私のために争わないで!」などという、あほなセリフが登場する可能性はゼロです。

また、「ああ!左の頬も叩いてちょうだい!!」というようなマゾな教えを説く宗教説法が出てくる余地もなければ、最後の審判の絵の中に暗号を見つけるような話も出てきません。「ブルータス、お前もか」という監視義務を怠った権力者の話もなければ、「生きるべきか、死ぬべきかそれが問題だ」という、勝手に悩んでいやがれ、というような文学も関係ありません。何を考えているか全然分からない隣国の独裁国家の最高権力者の頭の中や、ギリシャのデフォルト(default)を「借金踏み倒し」と置き換える必要もなく、世界のどこでテロが発生して、どのような報復攻撃が開始されたか、という内容の英語を理解する必要は全くありません。

私たちは、真の国際人になることを期待されて、このような全分野全方向型の英語を勉強させられてきたわけですが、正直なところ、エンジニアとして私たちが使う「技術英語」には、あまり関係のないものでした。

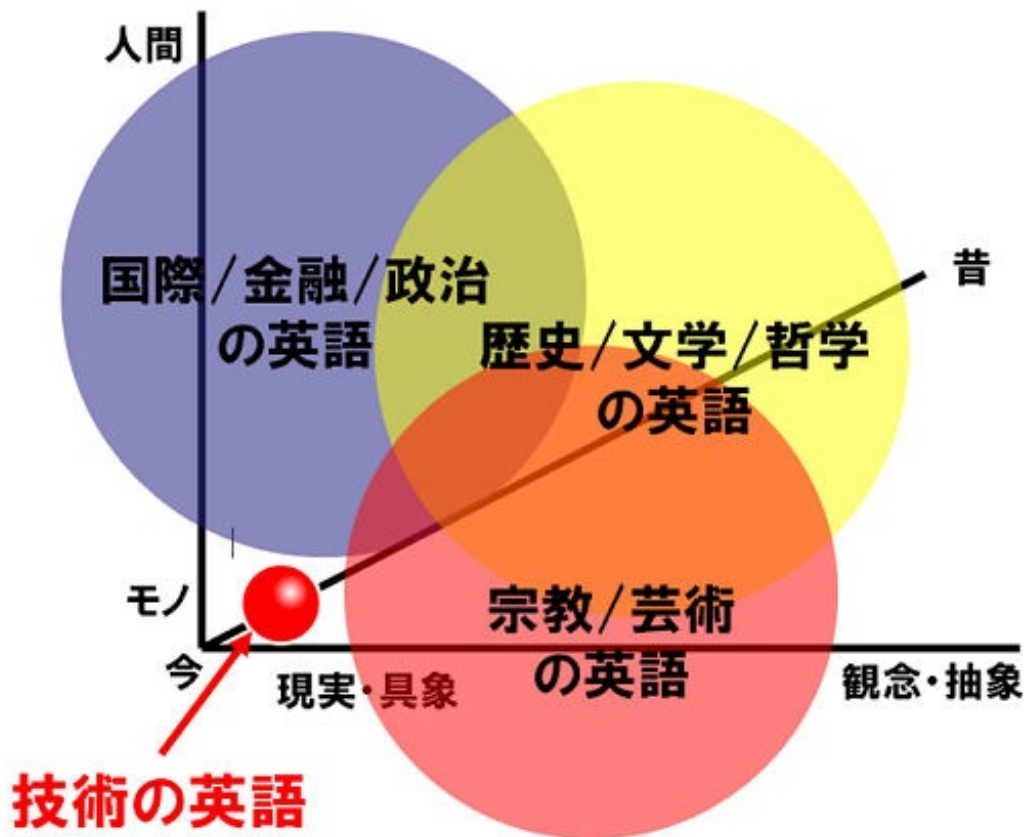


図1 技術英語が対象とする範囲は非常に狭い

「技術英語」の特性 ～ひたすら現実の具象のみが対象に～

さて、もう一度、図1を見て下さい。「技術英語」では、過去の話が必要とせず、ひたすら目の前にある「有体物(以下、物という)」に限定します。そして、その「物」を特定する手段として、観念や抽象表現の入る余地はなく、ひたすら現実の具象のみが対象となります。さらに「物」の中には、非論理的で、行動予測不可能な「人間」は含まれておりません。

英語の話から少々離れますが、「技術」が、人間の日常の挙動からいかに乖離(かいり)しているかを示す、幾つかの例を示してみましょう。

- 「私の青春の思い出は、米国特許庁で公開されたあの『特許明細書』でした」
- 「この『要求仕様書』こそ、人生の座右の書」
- 「教養人必読の『要件定義書』」
- 「評論家が選ぶ:この『マニュアル』がすごい!」
- 「ミシュラン編:3つ星『三相交流モーター設計仕様書』」
- 「週間女性 特集:AKB48のメンバーが隠れて読む『インターネット標準通信プロトコル』の番号を暴く! あのアイドル〇〇が密かに読んでいたのは、なんとRFC959(FTP)だった!」

上記のフレーズは、全く意味を成していません。技術という分野の記述が、いかに世の中の一般的な表現方法と親和性がないか、一目瞭然で分かると思います。

英語も例外ではありません。技術英語の領域は、対象が限定的であり、表現方法にも上限が

あります。データで示したいところなのですが、誌面の都合上、皆さんで調べて頂きたいと思えます。英文の単語を数えてくれるWebサービスやアプリケーションソフトウェアがありますので、そこに、技術文献と時事問題の英文を突っ込んで、そこに登場する英単語(特に動詞)を見比べてみてください。入試の時に必死で覚えた「試験に出る英単語」の丸暗記は一体何だったのかと、本気で泣けてきます。

「技術英語」をプログラミング言語と捉える根拠は？

さて、私は先ほどから「技術英語」という用語を使っていますが、この用語について私には違和感があります。「英語」が「言語」であることは議論の余地がありませんが、私は、英文で記載された特許明細書、論文、設計仕様書の英語を、ただの一度として「言語」として認識したことがないので。

本来「言語」とは、「人間相互理解」の道具であり、「民族」という単位のポピュラーな指標であり、「文化」という荷物を過去から未来へ運ぶリヤカーのような役割もあります。しかし、「技術英語」には、そのいずれの機能もありません。「物」である製品やシステムの仕様、または製造方法や使用方法を達成するための手段のみが、淡々と記載されているだけです。

「技術英語」が「言語」でなく、「英語」の下位概念ですらないとすれば、それは一体何でしょうか。結論から申し上げます。「技術英語」とは、「図」を構成要素とする「プログラミング言語」です。本連載においては、私達「英語に愛されないエンジニア」は、技術英語を以下のように把握するものとします。

- (1) 技術英語とは、2つ以上の技術英単語の組み合わせによって構成される命令文である。
- (2) 技術英単語は、表記文字列で表現される「記号」であり、当該「記号」は「図」に置換可能である。
- (3) 上記(1)、(2)より、技術英語とは、「図」を構成要素とする命令文の集合体からなる、プログラミング言語である。

技術英語というプログラミング言語の「シンタックス」

では、技術英語というプログラミング言語のシンタックスを、図を使って簡単に説明します。図2の「A」、「B」、「C」は全て技術英単語です。「C」は動詞ですが、原則として、過去形も現在完了も過去完了も登場しません。唯一、受動態への活用があります。技術英語というプログラミング言語は、たった3つの単語だけで、1つの「絵」を表現するものと考えてよいです。もう少しかみ砕いて図示すると、以下の図3と図4のような感じになります。図4は、図3をさらに具体的な対象に当てはめたものです。

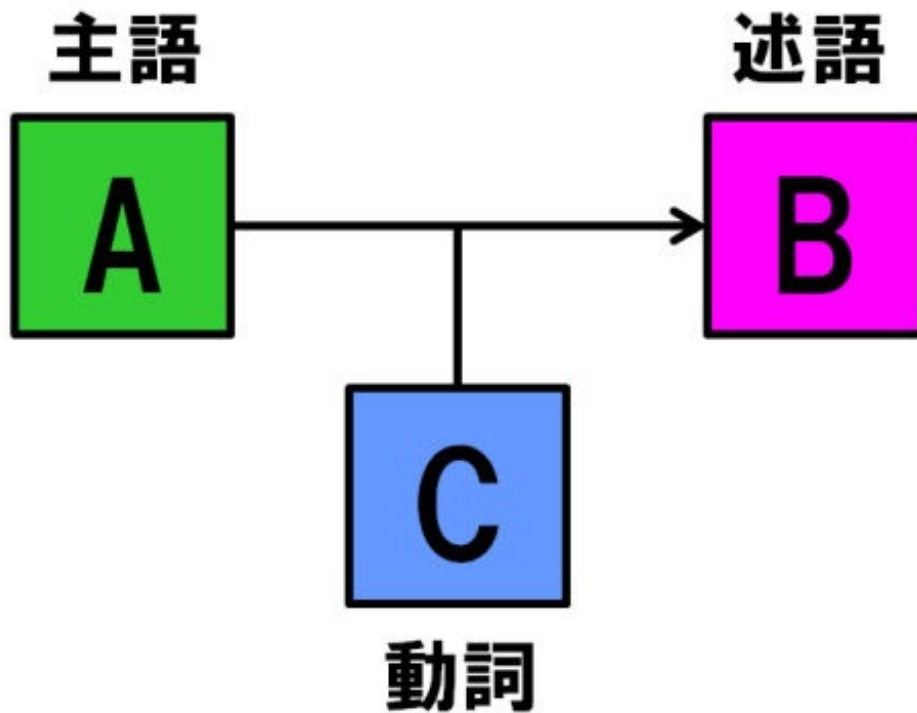


図2 技術英語というプログラミング言語の構成

技術英語という名のプログラミング言語の目的は、リーディングであろうが、ヒアリングであろうが、スピーキングであろうが、基本的にはこの「A」、「B」、「C」の図を、紙の上、ホワイトボードの上、または頭の中で描くこととなります。現実には、このような図をリアルタイムで頭に思い浮かべることは難しいのですが、それについては今後の連載で説明させていただきます。

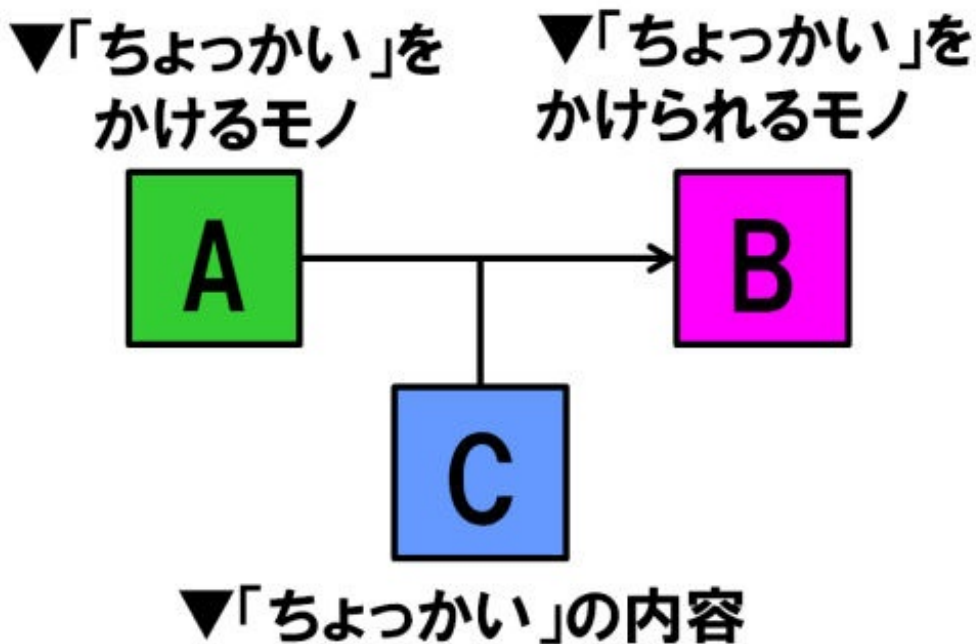
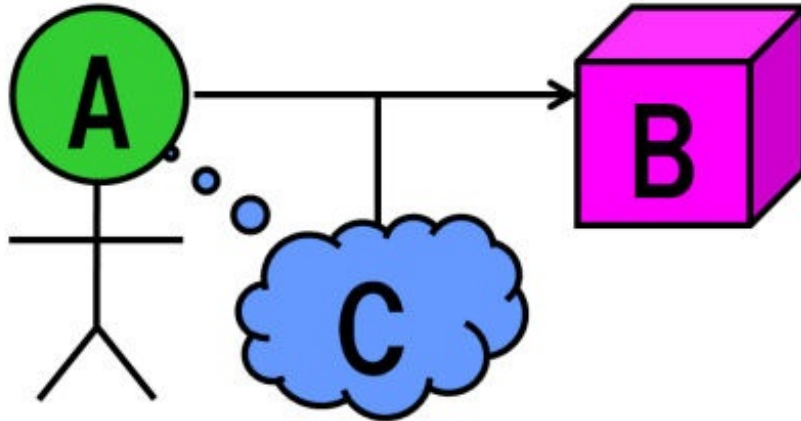


図3 技術英語というプログラミング言語をかみ砕いた構成

(大抵の場合)
私たち(読者)

製品(モノ)
方法(やり方)



“いじる、叩く、こねくり回す”

図4 図3に具体的な事象を当てはめた構成

さて、ここでは皆さんが技術英語に接する機会が最も多いであろう、リーディングについて、もう少し説明したいと思います。リーディングに関しては、さらに図が簡単になります。「A」がなくなって、「B」と「C」だけになり、「C」が受動態になるだけです。これは、技術英語の文章(仕様書、論文、特許明細書)は、主体が明らか(読者)であるからです。

例えば、要求仕様書の中で、「この部分はあなたの『上司』に検討してもらい」、「あの装置はあなたの『パートナー』に作ってもらい」、「この数値データの処理は、『部下』にやらせれば足りる」などと、そんなこと(主体)を、いちいち指定する必要がないからです(図5)。

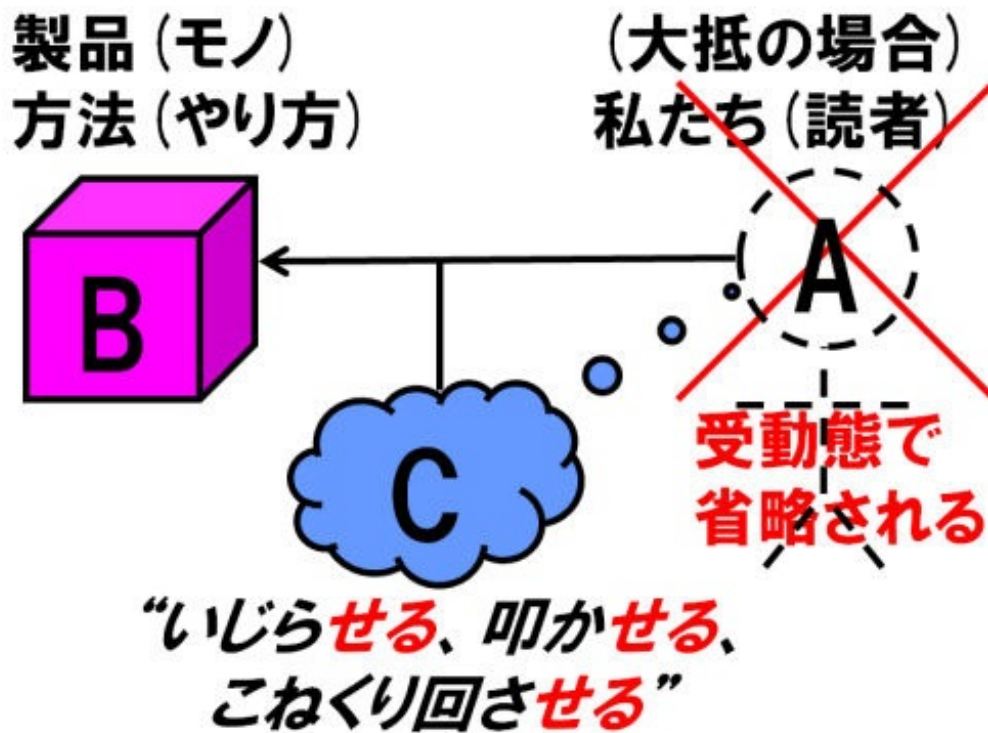


図5 リーディングの場合の「A(主語)」、「B(述語)」、「C(動詞)」の関係

つまり、どんなに長い文章であっても、2つの技術英単語を発見すれば足够了。「技術英語」という名のプログラミング言語は、表現力に乏しく、拡張性に欠けるという大欠点があるのですが、それは「英語に愛されないエンジニア」にとっては、神様の贈り物というくらいの恩恵なのです。では、具体例を示します(図6)。図6に引用しているフレーズは、机の上に置いてあった、あるシステムの国際標準化仕様書の適当なページから持ってきました。

The **results** from the phase shall be **documented**, along with any assumption and justification made during the phase.

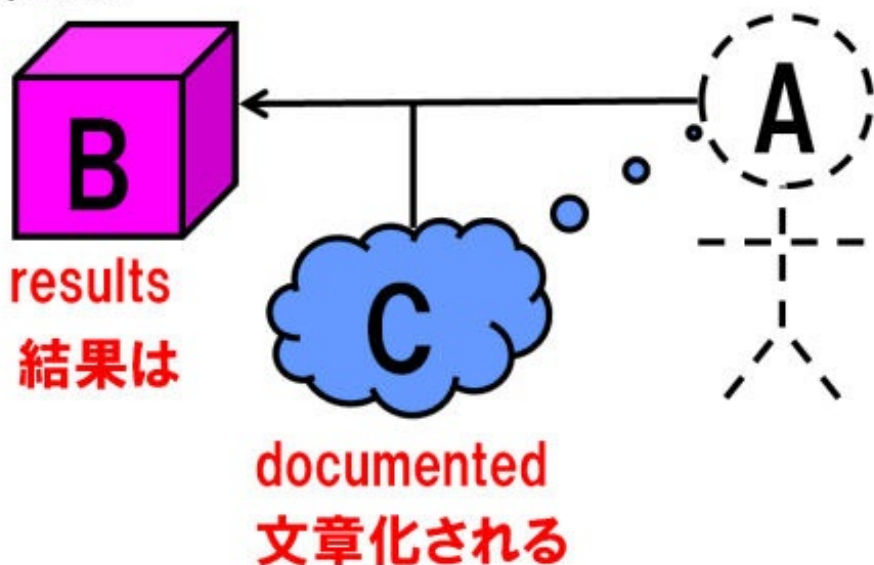


図6 図5の構成に具体的な例文を当てはめた例

この文章の概要が、「結果が文章化される」という内容であることは、簡単に理解できるはず——なのですが、ちょっとここだけは、断言することができないのです。膨大な英単語の中からこの2つの「B」と「C」を簡単に見つけられる人は、そもそも「英語に愛されている人」です。このような「B」と「C」を見つける能力がないからこそ、私たちは「英語に愛されないエンジニア」なのです。しかし、ここで、「たくさんの文章を読めば、分かるようになります」などという文脈で語ったら、私は読者の皆さんに殺されても文句は言えない、とまで思っています。

ですが、あんまり心配しなくて良いのです。95%以上の確率で、「B」は文章の最初に出てきた名詞で、「C」は最初に出会った動詞で間違っていないです。技術英語は、レトリック（倒置法やら強調法やら）を使うことが忌避（きひ）されています。なぜなら、事象を客観的に記述できなくなるからです。

日本語の例文でイメージしてみましょう。

例1

誤:「その回転数の変化を検知する、まさにその装置こそが、このデバイスである」

正:「このデバイスは、回転数の変化を検知する装置である」

例2

誤:「この仕様書を具現化し、未来のスマートグリッド社会を創成させることで永遠の繁栄が約束されるのである」

正:「この仕様書は、スマートグリッドを実現する手段の1つを記載するものである」

このルールで理解できない文章があれば、それはあなたではなく、文章を作成したヤツが悪いのです。技術英語は、プログラミング言語だと申し上げました。読みにくい技術英語とは、バグがたんまり仕込まれたプログラムコードであり、そのようなプログラムコードを書くエンジニアの文章など、読むに値しません。どうせ、内容も的外れなものでしょう。

技術英語における「前置詞」と「助動詞」とは!?

「B」と「C」が決まれば、あとは、前置詞という接着剤でくっついた形容句や、動詞の回りにまわりついている助動詞を読み取れば足ります。ちょっと面倒なのは、関係代名詞で入れ子になっているフレーズですが、それでも、私の知る限り、図7に示す以外のパターンが出てくることは、まずありません。もし出てきたら諦めても構いません。

前置詞というのは、要するに「B」という技術英単語に、アクセサリーとか眼鏡とか衣服のような付随物を「くっつける」道具です。ということは、前置詞の後ろは読まなくても、文意に大きく影響を与えません。関係代名詞の後の文脈も、思い切ってバッサリと読み飛ばしても良いでしょう。大切な内容であれば、著者は、独立したセンテンスとすべきだったのです。関係代名詞なんぞ

を使って手を抜いた著者が悪いのであって、私たちの責任ではありません。このような「割り切り」で、リーディングの時間は半分以下になるはずですよ。

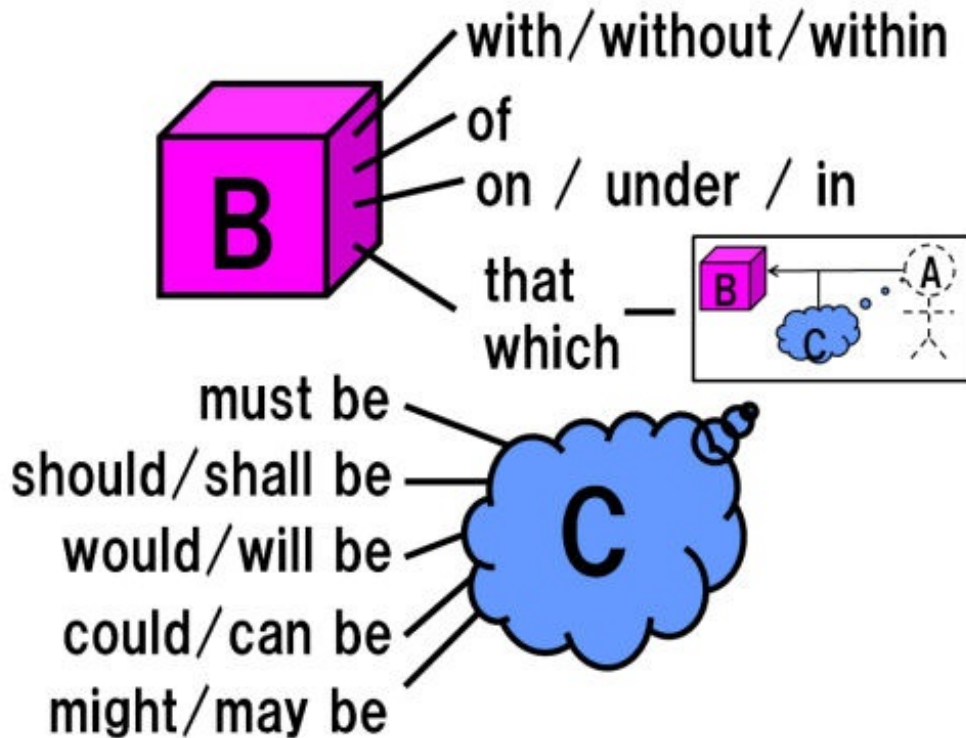


図7 述語と前置詞、動詞と助動詞の関係

一方、助動詞ですが、皆さんには信じられないかもしれませんが、技術英語として使われている限り、助動詞は全て同じ意味として使われます。基本は「なんとかしてね、お願い♥」です。ただ、「must」は、あなたに「銃を突きつけて」お願いしているのに対して、「may」は、「空を自由に飛びたいな〜♪」と歌っているドラえものののび太がするようなお願いになっています。お願いの必死さが「must」→「shall」→「will」→「can」→「may」の順番で弱くなっていると考えれば十分です。日本語に翻訳しなければならない場合は、あまり気にしなくてもよいでしょう。

「技術英語」は最終的に一枚の絵を描けば足りる

ではまとめます。(1) 技術英語は、図面を記述するためのプログラミング言語であり、(2) そのプログラムコードをコンパイル(翻訳)するあなたは、3つの技術英単語(リーディングの場合は2つの技術英単語)を発見することがメインのタスクで、(3) 前置詞や関係代名詞以下のセンテンスは、面倒ならスキップしてもよく、(4) 助動詞は、単なる「お願いの気持ち」として理解して、(5) 最終的に一枚の絵を描けば足りる、ということを理解していただけたでしょうか。

英語翻訳エンジンが、技術英語の分野で比較的使いものになっていることは、技術英語がプログラミング言語であることの一つの証拠になると思います。しかし、プログラミング言語である以上、それは人間が読んでいて楽しいものではありません。プログラムリストを見ながら笑っている人がいたら、それは不気味な光景です。私の知る限り、過去に一人だけいましたが、その後、風の便りで「心の病が原因で離職された」と聞いております。

それはともあれ、楽しくないなら、手を抜く必要があります。これについては、「文献調査編」で説明します。念のため繰り返しますが、今回の「技術英語=プログラミング言語」は、技術分野だけに適用可能な概念です。間違っても、この考え方を、時事問題や文学の分野に適用しないようお願いします。「まぜるな!危険!」と表記された漂白剤よりも、危険なことが起こるかもしれません。

今回は、このプログラミング言語を翻訳する側である、「私たち自身の改造方法」についてお話しします。

本連載は、毎月1回公開予定です。[アイティメディアID](#)の登録会員の皆さまは、下記のリンクから、公開時にメールでお知らせする「連載アラート」に登録できます。



Profile

江端智一(えばた ともいち) [@Tomoichi_Ebata](#)

日本の大手総合電機メーカーの主任研究員。1991年に入社。「サンマとサバ」を2種類のセンサーだけで判別するという電子レンジの食品自動判別アルゴリズムの発明を皮切りに、エンジン制御からネットワーク監視、無線ネットワーク、屋内GPS、鉄道システムまで幅広い分野の研究開発に携わる。

意外な視点から繰り出される特許発明には定評が高く、特許権に関して強いこだわりを持つ。特に熾烈(しれつ)を極めた海外特許庁との戦いにおいて、審査官を交代させるまで戦い抜いて特許査定を奪取した話は、今なお伝説として「本人」が語り継いでいる。共同研究のために赴任した米国での2年間の生活では、会話の1割の単語だけを拾って残りの9割を推測し、相手の言っている内容を理解しないで会話を強行するという希少な能力を獲得し、凱旋帰国。

私生活においては、辛辣(しんらつ)な切り口で語られるエッセイをWebサイト「[江端さんのホームページ](#)」で発表し続け、カルト的なファンから圧倒的な支持を得ている。また週末には、LANを敷設するために自宅の庭に穴を掘り、侵入検知センサーを設置し、24時間体制のホームセキュリティシステムを構築することを趣味としている。このシステムは現在も拡張を続けており、その完成形態は「本人」も知らない。

本連載の内容は、個人の意見および見解であり、所属する組織を代表したものではありません。

関連リンク

[筆者の個人Webサイト「江端さんのホームページ」](#)

Copyright© 2016 ITmedia, Inc. All Rights Reserved.

