

専門用語の使用法に注目した情報教育の実践

江見圭司

金沢工業大学情報フロンティア学部

〒921-8501 石川県石川郡野々市町扇が丘7-1

電話 076-248-8994 (直通) FAX 076-294-6709

e-mail: emi@infor.kanazawa-it.ac.jp

情報教育では、同じものを指すのに、異なった用語を使うことが多い。オブジェクト指向では、C++、Javaで用語は異なる。画像でもドットとピクセルの2種類の言い方がある。筆者が行ってきた実践の中から、これらの具体例を挙げて議論していきたい。

Practices of Computer Education in Considering How to Use Technical Terms.

Keiji EMI

College of Human and Communications, Kanazawa Institute of Technology

Different technical terms sometimes stand for the same thing in computer education. For instance, in the case of object-oriented programming, different terms are used in C++ and Java. In the case of digital image processing, there are two units: dots or pixels. I will discuss some examples in my practices.

1 はじめに

情報教育において専門用語は複雑な感がある。複雑に感じる原因として3つぐらいに分類できる。

一つめに、頭字語(acronym)や略語(abbreviation)に分類されるものである。たとえば、EUCはEnd-User ComputingとExtended UNIX Codeの2種類があり、文脈で判断しないとどちらのことか勘違いすることがある。

二つめには英語から日本語に訳すときに発生した複数の言い方であったり、不適切な訳語であったりするものである[1]。たとえば、あとでもあげるように「1次」、「線型」、「線形」がすべてlinearの訳語である。こういう場合、誤解を与えないために教員は「リニア」カタカナ語を使うこともあるが、それが却って受講生に難しく感じたりするから大変である。

三つ目には、本来別物であったものがだん

だんと同じ場面で使用されるようになったために、その違いがわかりにくいものである。たとえば、IPマスカレード(IP masquerade)とNAPT(Network Address Port Translation)がそれにあたる。もともと「NAPT」が技術の名称で、「IPマスカレード」はLinuxにおけるNAPTの実装のことだったが、現在では両者が混同されている例が多く見られる。また、ブロードバンドルータのスペック表などではNAPTの意味で「NAT機能搭載」と表記していることもあるのである。

なお、この3分類のどれにも入らない場合もある。ここでは第2節で具体例をあげ、第3節でどのように実践するのか、e-ラーニングなどへどう応用するのかという提案を行い、第4節でまとめることにする。

2 筆者の実践例

本節では筆者が授業中に受講生に対して注

意を促してきた用語について、いくつか紹介する。

(1)オブジェクト指向

オブジェクト指向の用語は大変複雑に感じる学生が多い。図 1. はオブジェクト指向のプログラミングの授業で使用する図である。この図では UML を書いてプログラミングを整理することを教えているのであるが、使う言語によって用語が異なるので、その対応表も合わせて掲載している。オブジェクト指向論ではたいてい、属性と操作という用語を使うが、C++ではデータメンバとメンバ関数、Java ではメンバ変数/フィールドとメソッドを使う。図 1. の最も右の列は空欄にしているが、別のオブジェクト指向言語、例えば C# や Action Script, Visual Basic.net などを解説する必要が生じたときに追加できるようにしている。

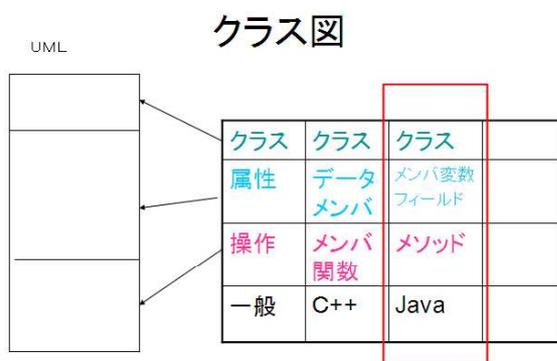


図 1. オブジェクト指向の用語法

(2)データベース

データベースの基本は表（テーブル）があり、行方向はレコード、列方向はフィールドと呼んでいる。C 言語で構造体でデータベースを書こうとすると、レコード 1 行 1 行が構造体のインスタンスとなり、フィールドはメンバとなる。また、最近のバージョンの Microsoft 社のデータベースソフト Access ではフィールドを「項目」と呼んでいる。この

ように用語が混乱するが、混乱を整理することにより、受講生にすっきりさせることは重要である。

具体的な実践では、2003 年度では筆者は情報工学科で C 言語の授業を行っていた。2004 年度からは情報フロンティア学部メディア情報学科で C 言語を教えることになった。メディア情報学科では 1 つ前に学期に「システムアドミニストレーション I」という授業で関係データベースの基本を終えてから、C 言語を教えるようにした。2004 年度からは構造体の説明をするときに、構造体の用語法とデータベースの用語を比較して授業展開を行ったため、以前に比べると「構造体をなんのために使うのかわからない」という質問件数は減った。

(3)データベースの3層モデル

データベースの 3 層モデルもいろいろな言い方がある。データベースソフトによっても違いがあったり、システム構築者によっても違ったりする。いずれにしても 3 層で解釈することがわかることが重要である（表 1）。

表 1. 三層モデルの呼び方の例

一般	例 1	例 2
外部層	プレゼンテーション層	プレゼンテーション層
アプリケーション層	アプリケーション層	サービス層
内部層	リソース層	データベース層

(4)画像のドット数

画像のドット数単位名もわかりにくい。デジタルカメラなどでは 640 ピクセル× 480 ピクセルといい、その積を 30 万画素などという。一方、ディスプレイなどの解像度を表すときは、640 ドット× 480 ドットという。こ

これらの違いは、前者が画像の入力装置に使用するのに対して、後者は画像の出力装置に対して使う単位であることを理解すれば納得するが、初学者には大変わかりにくい。画像の用語法も解説するときには注意を促している。

(5)ホームページとウェブページ

英語では、ウェブサイト(Web site)があり、そこにあるページ一つ一つをウェブページ(a Web page)といい、ウェブサイトを訪問したときに初めて表示されるページをホームページ(the home page)という。しかるに、日本語ではウェブサイトをホームページと言うことも多く、ウェブページはすべてをホームページといい、ウェブサイトを訪問したときに初めて表示されるページは「トップページ」と言わざる得なくなってしまう。文科省系のサイトはこのあたりの区別は英語式に行っていたいるが、他では英語式の言い方はそれほど一般的ではない。

(6)関数

関数という概念ほど情報教育で頻繁に出て来るのに、わかりにくい言葉はない。筆者は遠山啓が提唱した「暗箱のシェーマ」[2]を用いて説明している(図2. 参照)。

関数は元来「函数」と表記されていた。「函」は「はこ box」の意味である。function という言葉を昔の中国人が「ファンズウ」と聞こえたため、「函数」と書いたようである。このイメージは大変よくできている。

さて、関数は「関係の数」のイメージを持ちやすい。つまりxとyの関係である。確かに間違いではない。写像(mapping)という考え方は関係の数という考え方に近い。しかし、関数は入力xに対して何らかの機能(function)が及んで出力yが出てくると解釈するべきであろう。写像では入力と出力の間の機能が意識されないので筆者は写像という考え方を好まない。

授業中には関数という言葉よりは「ファンクション」や「機能」という言葉を多く用いるようにしている。

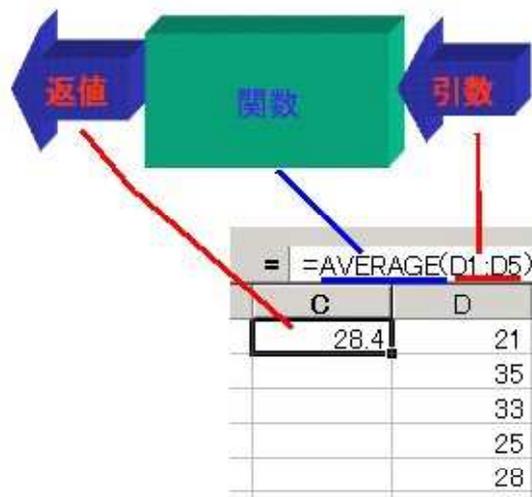


図 2. 関数概念と表計算ソフト EXCEL の対応

また学生の中には、関数を数の一種と誤解しているものもあるようである[3]。確かに数学では入力が数で、出力も数であるのでそれほど問題ではないかもしれない。しかし情報教育という立場では「関数が数の一種である」というとらえ方は大いに問題がある。入力(引数)、出力(返値)、関数の3者が区別をきちんとできることが情報教育では大切であることに異論の余地はないであろう。

(7)1次変換か線形変換か

linear transform の訳であるので「線形変換」が素直である。ところが、linear function は1次関数であるので、「1次変換」と訳したのであろう。しかし、1次関数には2次関数、3次関数と次数があがるが、1次変換に対して2次変換はない。だから、これは線形変換の方が適切であろう。そのうえ、「線形」は人によっては「線型」と書く人もおり、訳し方が人によって異なる典型例である。

(8)有理数と無理数

有理数は英語では rational number, 無理数は irrational number という。rational を辞書で調べると「合理的な」という意味が出てきて, ration(理性)の派生語とされている。だから明治時代の数学者は rational は ration(理)があるので,「有理」, irrational は rationがないので「無理」と訳したと思われる。しかし, rational number の rational は ratio(割合, 比)の派生語であるという説もある。この場合, ratio を分数という意味で解釈すると rational number は分数にできるという意味で「可分数」と訳すとか,「有比数」あるいは「可比数」と訳せばよかったですのでないか。さらに以下のような式

$$f(x) = \frac{x^3 + 5x + 2}{x^4 + x^2 + 4}$$

は「有理関数」と呼ぶこともあるが, これでは何のことかわからない。ここは「分数関数」とよぶべきである。このように, 数学で「有理」といえば「分数」と思えば, ほぼ間違いなく理解することができる。全く, 先人のおもわぬ訳し方で後輩である我々が迷惑をこおむるのである。一方, irrational number は「無理数」ではなく,「無比数」あるいは「不可比数」と訳すべきだった。

帰国子女は数学で使われる日本語が分からなくて苦勞するそうである。これも翻訳語が不適切であることが原因であろう。

(9)べき乗

文科省の学習指導要領では高校までの数学では「累乗」という言葉を使い,「べき乗」は使用しないことで統一されている。これは,「べき」という漢字が常用漢字にないため, 累乗という言葉を使うようにしているのである。しかるに, 専門の数学書籍やプログラミング関連の書籍は「べき乗」が普通であるため,

プログラミングの授業などで「べき乗」が「累乗」のことであることを言う必要がある。

文科省は「べき乗」という言葉を高校以下の教育では追放したが,「降べきの順」や「昇べきの順」という用語からは「べき」を追放できなかった。こういう事情であれば, 高校以下でも「べき乗」にするべきであるようにも思う。現状では情報教育の中では「べき乗」という他はない。

(10)開発環境名と言語名

この問題は深入りしないつもりなので, 簡潔に書いておく。Visual C++という開発環境が Microsoft 社から出ているが, これは開発環境名で C や C++, Win32API (C)や Microsoft Foundation Class (C++)など幅広く扱える。一方, Visual Basic.net というのは開発環境名であり, かつプログラミング言語名でもある。Macromedia 社ではもっと複雑で, 開発環境名は Flash, 開発言語名は Action Script, 作られた製品を動かすテクノロジーも Flash であるが, Director の場合は開発環境名は Director, 開発言語名は Lingo, 作られた製品を動かすテクノロジーは Shockwave といい, 大変混乱する。

最近の学生は表計算ソフトという名前を知らず, EXCEL だけしか知らない学生もいる。だから CSV ファイル(Comma Separated Value)を EXCEL のファイルと思っている学生さえいるので注意が必要である。

3 用語解説はどうおこなうのか

さて、筆者の教育実践で取り上げてきた専門用語を実際にはどんな風に扱えばよいのであろうか。用語解説は受講生に内容理解を正確にするためには大変重要であるが、あくまでも授業の流れの中では本線ではない。支線である。支線であるからには再び本線にもどる必要がある。

(1)黒板を使う場合

筆者の大学では黒板は前に3面あり、真ん中が最も大きな面積を占めており、左右に小さな面がある。筆者は板書するときには必ず、メインの解説は真ん中の黒板を使い、左右の小さな面で用語解説などを行うようにしている。そうすれば、用語解説をしているときは支線で、真ん中の黒板で解説を始めると本線に戻ることが一目瞭然でわかるのである。これを映像配信のe-ラーニングで行うときの画面設計を筆者の研究室で行った。それが図3である。この写真では、ビデオ撮りを4つのカメラで行うようなシステムで想定している。実際にはたかが大学の講義ぐらいで4つのカメラも利用した配信は現状では無理であるが、臨場感のあるe-ラーニングシステムを目指すのであれば、こういう提案もよいと考えている。



図 3. e-ラーニング用画面設計。下段真ん中が用語解説画面で、黒板の端のスペースを利用している。

(2)書籍またはウェブの場合

筆者の著書[4]ではこのような用語解説は本文中ではあまり行わず、ページの左右の脚注スペースに掲載するようにした。脚注スペースに図版も掲載する形式にしたため、図版によって本文が中断されることもない。また、脚注に用語解説を載せることによって、本文中では用語解説によって本文の流れを中断されることなく読むことが可能となる。用語解説の脚注が左右ではなく下段にあると、視線を上下する必要があるが、左右に配置することによって視線を上下に移動する必要がない。

これをウェブ用教材でどう実現するかはまだ検討の余地がある。本文のすぐ横に用語解説をつけるのか、用語をクリックすれば用語の解説が出てくるのかで結論が出ていない。

4 まとめ

専門用語の使い方をここでは議論してきた。徹底して整理してきたわけではないので、まだまだまともは悪いが、問題提起とe-ラーニングでどうするかという方向性は示した。

名前がさす実体の意味(シニフィアン, *significant*)が同じなのに、名前自体(シニフィエ, *signifié*)が異なることを中心に議論してきた[5]。特に翻訳語が多い数学や情報の分野では同じものを別の日本語に訳したり、カタカナ語に直したりとバリエーションが多すぎて学習者には苦勞の種になる。

ことばは支障がなければ簡潔な表現に向かう。そのため、情報分野では頭字語や省略語が多い。そのためコンテキストによって元の言葉が何かを考える必要があるものも多い。「ことばの意味は、コンテキスト(文脈あるいは前後関係)によって変わる」ということが、数学者や多くのコンピュータ科学関係者には理解されていないことが、情報教育としては問題であろう。しかし、頭字語や省略語はここではそれほど議論しなかった。

何を説明しようとしているかで、ことばの使い方は変わってくるのであり、厳密な表現が必要な専門家が、それを必要としない一般人の世界にその用法を強要するのは、望ましくないであろう。教育では、受講生は一般人から専門家へ移行させる過程であるから、ある程度の厳密な表現身につけやすい配慮した講義が必要であろう。

脚注

[1] 柳父 章「翻訳語成立事情 岩波新書黄版 189」(岩波書店, 1982年) この本では日本語の漢字語彙が明治時代にどのように成立していったが詳しく述べられている。とにかく、西洋のものを輸入するときの苦しみがわかる

[2] 遠山啓「数学入門(上), (下)」(岩波新書, それぞれ1959年, 1960年)

[3] 「高等学校の数学I 指導資料」(三省堂, 1982年)

[4] 江見圭司・江見善一「ベクトル・行列がビジュアルにわかる線形代数と幾何」(共立出版, 2004年)

[5] フェルディナン・ド・ソシュール, 小林英夫(訳)「一般言語学講義」(岩波書店, 1986年) シニフィアンとシニフィエのことは構造主義の解説本の方がわかりやすいが、オリジナルにはこの本である。