Beyond meatspace[1] – or, Geeking out[2] in e-English

ANGELA P. CHEATER

The intricate realities of computer-related word formation

AS ELECTRONIC engineers create virtual reality in and through computer software, they are also creating new English words to describe their ongoing technical achievements. While following existing rules of word formation, they are also changing these rules: for example, by making new nouns from prepositions. Their neologisms seem to be driven by humorous responses to electronic requirements of speed, multi-functionality and reversibility. This paper examines the current but on-going impact of their usage on the grammatical structure of the language. In a rapidly-evolving electronic future, their usage is likely to have a major impact on how 'ordinary' Internet users communicate online.

Introduction

In 2004, the Faculty of Engineering at the Chinese University of Hong Kong asked the members of its Writing Across the Curriculum (WAC) project to help improve its students' English language skills. To prepare for the WAC presentations and the ensuing discussions, I went surfing staff web pages to identify particular staff research interests. This exercise alerted me to the potential difficulties in 'teaching' electronic and computing engineers 'English', difficulties that go beyond those of English for mathematics or medicine, because computing is moving so fast. I also discovered an English word new to me – *aboutness* – and later, its opposite – *non-aboutness*.

Aboutness? Sure, it followed normal rules of word construction, and, as I later discovered online, from *Wikipedia*, its origin is traced to nineteenth-century philosophical concerns about consciousness and intentionality – something I had not picked up in my personal reading of Edmund Husserl and Franz Brentano. *Aboutness* also makes sense as a concept to describe the problem of drawing a boundary around queries for search engines, in order to make them more precise.

> Information retrieval (IR) is a reasoning process which is assumed to be driven by determining aboutness (|=) between two information carriers (ie document and query)... Although a number of aboutness properties are commonly discussed in the literature, eg reflexivity, transivitity, symmetry and left (right) monotonicity etc, there is thus far no agreement on a core set of aboutness postulates (Bruza *et al*, online, no date:[3], consulted 2005-01-07)

To a non-hacker, however, 'aboutness' and 'non-aboutness' pack powerful shocks. They use standard rules of noun formation to create nouns from a preposition. Another shock came from Lehman *et al.* (2000:1), who described the issue of 'how' software evolves as the use of

ANGELA P. CHEATER (Ph.D., Natal) teaches both social science and English Language as Professor in the School of Business, Macau Polytechnic Institute, Macau SAR, China. She has undertaken research on three continents into a range of development issues, most recently use of English. She has held chairs in sociology and social anthropology at the University of Waikato, Hamilton, New Zealand, and the University of Zimbabwe. After taking early temporary retirement from academia, she managed research into human rights in Zimbabwe, before serving as co-ordinator for Writing Across the Curriculum, a programme at the Chinese University of Hong Kong. Her publications include 'Social Anthropology: An Alternative Introduction' (Unwin Hyman) and 'The Anthropology of Power' (Routledge), five other books, and some sixty academic papers.

DOI: 10.1017/S0266078406001040

 © 2006 Cambridge University Press

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

'evolution as a verb'. Had they used 'evolving' instead of 'evolution', it would not have generated the same linguistic shock.

In this paper, my concern is not the acronym-loaded English that claims to make computers user-friendly to ordinary people, nor the inventive new vocabulary, sometimes called *techspeak*, which has been developed for this specialty.[4] Instead, from the perspectives of both social anthropology and of language teaching (but *not* linguistics, in which I have no competence), I examine the technical language – what I call *e-English*[5] – of those on the cutting edge of digital development: in webpage design, non-Roman scripts, graphics, music, search engines, and the like. Scrutinising online computing dictionaries and academic publications reveals both the logic and the implications of usage emerging in technical publications, which have been confirmed by listening to the language of colleagues teaching computing courses.

I am not the first to do this. There has already been some internal reflection by L1 hackers themselves, on the ways in which they are changing the English language: *The Jargon File* has for example examined verb doubling, over-generalisation, comparatives, anthropomorphism, and writing style, as well as showing how the demands of punctuation 'modelled on string and character literal syntax in some programming languages' have brought hackers into conflict with traditional American editing conventions.[6] However, on some issues (including verb creation), the evidence I present below conflicts with *The Jargon File's* earlier assertions about grammatical structures.

I am particularly interested here in three aspects of e-English: the rules for creating new words; the elements used to construct them; and the impact of both on grammatical structure. But first, let me speculate about the environmental conditions which generate and nurture e-English.

Shaping electronic communication

Some might argue that the basic language of computing is algorithms. Friedrich, for example, averred as a 'fact' that physicists (and by implication other scientists) 'use calculus (or some other mathematics) as their basic language, the language of conceptualisation' (1986:159). However, like many, he appears to want to eat his cake, given his claim (1986:24) that 'poetic' language permeates both ordinary and technical languages, particularly under conditions of humour and strong emotion.

Denis Howe has explained, from a 'Commonwealth Hackish' perspective (but his colleagues who use 'American Hackish' agree[7]) that hackers share a distinctive style of intellectual humour marked by the following characteristics:

1. Fascination with form-vs-content jokes, paradoxes, and humour having to do with confusion of metalevels
2. Elaborate deadpan parodies of large intellectual constructs, such as specifications, documents, language descriptions and even entire scientific theories (such as 'quantum bogodynamics').
3. Jokes that involve screwily precise reasoning from bizarre, ludicrous, or just grossly counter-intuitive premises.
4. Fascination with puns and wordplay.
5. A fondness for apparently mindless humour with subversive currents of intelligence in it – for example, old Warner Brothers and Rocky & Bullwinkle cartoons, the Marx brothers, the early B-52s, and Monty Python's Flying Circus. Humour that combines this trait with elements of high camp and slapstick is especially favoured.[8]
6. References to the symbol-object antinomies and associated ideas in Zen Buddhism and (less often) Taoism.

What (other than humour) have hackers used to construct their new words and grammatical forms? Following Bolinger (1961) and Friedrich (1986), Wang (1989) has suggested that, in any language, speakers reach for what e-English calls 'shelfware' – readily-available 'prefabricated' words, phrases, clichés, proverbs, idioms, metaphors, similes, epigrams, slogans and the like – to create their own unique speech. Friedrich (1986:23) has emphasised the use of prefabs by 'hurried people with practical jobs to get done,' *inter alia* in medicine, advertising and politics, but was also concerned with the imaginative, creative and poetic use of prefabs. From the evidence examined below, e-use of prefabs seems to be creatively practical, but more earthy than poetic. Prefabs arise from the ever-greater demands of speed and efficiency which directly affect e-English as part of the computing environment.

Hacker prefabs are based on their own spe-

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

cific definition of a lexeme. e-Usage of lexemes[9] is perhaps modeled on English prefixes and suffixes, but also incorporates acronyms, punctuation and other symbols. Firstly, their definition and usage suggest that 'e-prefabs' come in smaller units than Wang (1989) identified. One e-English dictionary defines a *lexeme* as:

> a minimal lexical unit of a language. Lexical analysis converts strings in a language into a list of lexemes. For a programming language these *word-like pieces* would include keywords, identifiers, literals and punctuation. The lexemes are then passed to the parser for syntactic analysis.[10] (my emphasis)

Hacker usage is closer to the definition of a lexeme by *Wordnet* as a 'minimal' unit than to the *American Heritage Dictionary*'s 'fundamental' unit in the lexicon of a language. It is also in line with an unnamed *Wikipedia* author's note:

> [W]hen people read, they do not in reality try to work out the sequence of sounds composing each word, but instead either recognize words as a whole, or as a sequence of small number of semantically significant units (eg 'morphology' might be read as 'morph'+ 'ology', rather than as a sequence of a larger number of phonemes).[11]

Secondly, e-English seems to prefer multifunctional verbs, nouns, and adjectives, built up from such lexemes, to single-function nouns, verbs, and adjectives. This preference, like noun strings in e-English, may reflect mother-tongue influences on L2 English speakers from the Far East who are at the cutting edge of many technical developments in computing.

Inter-intelligibility among L2 users (probably already the majority of English-speakers worldwide) is also important. 'English as a whole is already heading ... towards pure-positional grammar like Chinese; hackers are simply a bit ahead of the curve'[12] – possibly because many of the cutting-edge innovations come from East Asia.

Perhaps the single most critical driver is that the world of *digital*[13] computing is built on reciprocally-defined binary opposites. Every concept has its antithesis, every state its negation, every action its reversal. Yet English verbs are generally asymmetrical,[14] the main sets of antonyms being paired prepositions. It is possible, therefore, that, where antonyms do not yet exist, e-English will develop binary opposites for all of its key words. Whether these necessary oppositions will systematically use one among several possibilities, is a question for future research. Here I merely begin to explore what has already started and is still in the process of happening.

Rules for creating new words and the elements used to construct them

1 The basic example

FOLDOC (1996-08-26) notes that the verb *hack* originally meant 'cut roughly', and a *hacker* made rough-hewn furniture with an axe. In e-English, *hacking* means 'to interact with a computer in a playful and exploratory rather than goal-directed way' and 'diffs' *hack on* from *hack up*.[15] The noun *hacker* has developed many variations from its base e-meaning as one 'who enjoys exploring the details of programmable systems and how to stretch their capabilities'. Adjectival forms include *hack, hacker, hackerly*, and *hackish* (which, when capitalised, is the language of hackers). Derivative nouns include *hackishness*, *hac(k)titude* and *hackerdom*. As yet there appear to be no abverbial forms, though *hackingly* is imaginable.

The illegal versions of the above forms are based on the verb stem *crack*, including *cracker, cracking, the cracker underground* and *cracker slang*; and the insult version (when applied to others, not oneself) is *geek*, leading on to *Alpha Geek (the boss geek*, or *geek's geek)*, *geek out*, and *clustergeeking* (doing homework at a cluster of computers). There appears to be a generation gap between self-confessed (older) hackers and those they refer to as *crackers*, *phone phreaks*, and *warez d00dz* [dudes purveying illegal wares, as indicated by the *-z* ending which is always applied to illegal behaviour and/or software clones] – mostly teenagers running IBM-PC clones from their bedrooms,[16] and it is quite possible that my evidence is specific to the older generation which has used its power to record its views in accessible places.[17]

2 Analogies, metaphors, similes, and spoonerisms

Analogies and metaphors have always been an important initial mechanism for extending meanings in new fields, so perhaps chronologically it all started with the 'machine learning researchers ...[who] preferred anthropomorphic terminology and allusions to the biologi-

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

cal sciences to rigorous mathematical proofs' (Berry & Linoff 2000:16). *Dogcow*, *mouse*, *sheepdip*, and *Trojan horse* spring immediately to mind, followed by *sex changer/gender mender/homosexual adaptor*, *vaxectomy*, and the famous poster hanging inside many toilet doors, which maps computing terms onto their sanitation engineering equivalents: the toilet pan is the central processing unit (CPU), the toilet seat the ineterface, the lid the floppy disk, the water cistern a piece of peripheral hardware, the toilet paper the software, etc. *The Jargon File* avers that 'hackers who anthropomorphize are expressing ... a mechanistic view of human behaviour' when they say 'You can't run those two cards on the same bus; they fight over Interrupt 9', or 'The protocol handler got confused...'[18]

'Parallel' hackish humour also explains the origins *inter alia* of: *depeditate* ('to cut the feet off' the lower part of a display on screen); *digirati* or, alternatively, *digerati* (computing experts, the 'digital literati'); *feeping creaturism*, (a spoonerism, whereby more and more new features creep into updated and more expensive versions of software); *friode* ('fried' or electrocuted diodes in a computer, killed by an electrical surge); *lexiphage* (when your screen suddenly goes blank, your words have obviously been eaten by something inside the computer); *lossage* (information loss resulting from software 'bugs' or malfunctioning); *screwage* (information loss resulting from deliberately designed misfeatures in software); *manularity* (*manual granularity* as a measure of automation – or doing a simple computing task by hand rather than building another computing tool to do it); *nastygram* (an electronic message couched in disapproving language); and *phishing* (a form of computer fraud that steals identity details).

However, the following items, among many others, are 'semi-different': *brouters* (network devices that bridge some packets of information and *route* other packets), *disusered* (being denied entry as a user), *notwork* (see below: Nouns (e) Reversing states), *rwho* (a specific Berkeley Unix command to *report who* is logged into a particular segment of the network), *uncompression* (the expansion of compressed data), *vhosts* (virtual hosts which, despite being single computers, appear to be multiple websites because they simultaneously have different Internet Protocol (IP) addresses and therefore undertake different hosting functions), and *xmosaic* (the Mosaic programme specifically for the X Window System). So let us look at what is happening to different categories of words. In so doing, I deliberately use e-English lexemes and creative techniques in the base tense of the scientific simple present.

Impacts on grammatical structure: parts of speech

1 Verb-nouns

The lexical indistinguishability of present participle and gerund (e.g., code-cracking) is perhaps our starting-point, but verb stems are now also becoming nouns. *Quantify* is the name of a performance analysis tool, in the same way that one Unix program, itself called *Make*, allows FOLDOC the following sentence (with lower-case *m*): 'If make is run with no arguments, it looks for a makefile...'.[19]). *The Jargon File* appears to give precedence to verbing nouns (making verbs from nouns) over nouning verbs (making nouns from verbs),[20] but I think it more likely that in e-contexts action precedes state.

2 Transitivising verbs

The vast majority of verbs in e-English are transitive. In addition, the transitive e-use of three normally intransitive verbs – *evolve, migrate, mutate* – gives them added e-functionality. Transitivisation sheds both prepositions (sometimes collapsing them into verb lexemes) and indirect objects and starts a knock-on series of further lexeme evolution (e.g., creating adverbial forms from either or both of the participles). *Boundedly* has finished this process; *multiplexing* and *multithreadinged* have yet to develop as adverbs.

3 Prepositionising verbs/collapsing phrasal verbs

Phrasal verbs cause L2 users problems, as both verbs and adverbial particles. The simplest way to deal with them is to collapse the particle into the verb as a single new word: *backpropagate*, *cutover*, *downlink*, *overtrain*, *signon*, *signoff*, *timeout*, *traceback*, *upload*, *workaround*. Such collapses have the useful secondary effect of creating flexible verb-nouns, much as *to feed back* produces *feedback*. Such multi-skilled verb-nouns, especially when evolved further

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

into, for example, 'feedbackward' and its binary opposite 'feedforward', can then be used both adjectivally and adverbially.

4 Finalizing actions

To *finalize* is defined by the *Oxford Advanced Learner's Dictionary* (2000:474) as 'to complete the last part of' some process. So when Roxio software asks whether the person copying selected files onto a Read/Write CD wishes to 'finalize selection', an English-speaker follows the reasoning. However, Roxio then gives the further choice 'finalize CD'. Choosing this option then makes the process of writing to that particular Read/Write CD irreversible, adding a new meaning to the repertoire of *finalize*.[21]

5 Reversing actions

To reverse the meaning of a verb (or its derivative noun), the prefixes *de-*, *dis-* and *un-* are available. Since the meaning of all three is the same, they may be regarded by L2 users in particular as interchangeable: What, after all, is the difference between *decompressing* and *uncompressing* a file? However, the demands of speed and efficiency have created generic reversors which appear in most user-friendly software as *cancel*, *don't + verb*, or *undo + verb*. Appendix 1 lists common e-verbs with as well as without antonyms, to all of which these generic reversors can be applied.

6 Verbising adjectives 'discriminatively'

The adjective *active* clearly derives from the infinitive *to act*, but its opposite, *passive*, does not come from the verb *to pass*. From the adjective *active* comes the reworked verb *to activate* and the noun *activator*. The opposite adjectival form is also used to add the necessary reciprocal verb (*to passivate*, plus its participles and a new derivative noun *passivator*) to the e-lexicon. Traditional adjectives are a major source of new verbs and (especially *activator*-type) nouns.

7 Actives and passives

Scientific convention favours the use of the passive voice. However, a quick-and-dirty verb count of two short, multiply-authored papers of approximately the same length (one by L1 British, the other by L2, predominantly Chinese speakers) shows some interesting results. Only in the use of uncomplicated active and passive forms is the writing style similar. But for the more complicated uses of active and passive infinitives and bare participles, L1 speakers preferred the past participle and passive infinitive, whereas L2 speakers went for the active infinitive and bare present participle. The L2 speakers also used modal auxiliary verbs more often than, and adverbs less than half as frequently as, their L1 colleagues. See Table.

8 Adjectives

a Rule-bound neologisms

Adding *-y* or *-less* or *-free* to a noun is the usual e-method of constructing adjectives, but the words so created may raise eyebrows, as with *contactless*, *cuspy*, *deadlockfree*, *lossless*, *lossy*, *mismatchfree*.

b Reversals/binary opposites

What is the opposite of *deadlocked* (as an adjective)? Both *deadlockfree* and *livelocked* are available.[21] Which will prevail? That is a political question, concerning who in the e-English community has the power to define and dictionarise neologisms. The generic reversor is *non* + adjective, usually in hyphenated form: *non-maskable*, *non-monotonic*, *non-polynomial*.

c Collapses

Adjectival speed and efficiency require the combination of formerly separate descriptives: *mechantronic*, for example, instead of *mechanical and electronic*. Noun prefixes (*bit*, *core*, *fibre*, *net*, *time*, *web*) may be collapsed with or without hyphenation, depending on personal and cultural inclination. Greek or Latin prefixes indicating number or quantity are not usually hyphenated (e.g., *gigabyte*, *terabyte*, *nanotechnology*).

d Jokers

The words *pseudomorphic, pseudospectral, pseudo-exhaustive, pseudo-inverse, pseudo-random* all describe things which are not actually what they appear to be.

e Willing and able?

Possibly the most FAQ in computing begin *Can it…?/Is it able to…?* Neo-adjectives answering these simple questions end in *-able* or *-ible*, and generate new *-ability* nouns. *Compute* and

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

A table comparing verb forms in two short technical computing papers by L1 and L2 speakers of English

Verb form	L1 n	L1 %	L2 n	L2 %
Active	62	517	57	50
(is/are – bare)	(14)	(11.7)	(18)	(15.8)
Passive	22	18.3	23	20.2
Infinitives: active	13	10.8	20	17.5
Infinitives: passive	7	5.8	2	1.8
Present participle – bare	2	1.7	11	9.6
Past participle – bare	14	11.7	1	0.9
Total verbs	120	100	114	100
Modal auxiliaries: non-tense	(9)	(7.5)	(14)	(12.3)
Total number of adverbs	16	Nil	7	Nil

evolve have already completed this process: *computable, evolvable*. Among currently-evolving adjectives are: *migratable*, *mutatable*, *parametrizable*, *representable*, and *stabilisable/stabilizable*. Of these, my research indicates that only *migratable* has so far evolved *migratability* as a derivative noun, and (as of Jan 05) I have only *heard* this noun used, not read it anywhere.

9 Nouns

a Active nouns

Everything is potentially active and mobile in the virtual world, so nouns accustomed to sit statically, doing little except describe states, must become 'activatable'. English customarily uses either of two methods to achieve this activation. For e-English, the main method gives nouns (usually via a derivative adjectival form) potential, or abilities. By 1959, mathematicians had already named the concept *axiomatisability*. Capabilities subsequently potentiated many other nouns. *Wikipedia*[23] identifies the following, in the field of software engineering: *administrability, composability, configurability, degradability, demonstrability, deployability, distributability, extensibility, evolvability, interoperability, maintainability, reusability, scalability, schedulability, supportability, survivability* and *tailorability*. As yet, however, I have nowhere encountered *paramet(e)risability* – but it is undoubtedly in the pipeline. *Dissipativity* is already here, masquerading as a state noun presumably because that form sounds better than *dissipatability*, although the *-ity* form is less common than the *-ability* form.

The longer-established method of activation is a two-stage process, first creating a new verb (ending in *-ise/-ize*[24]), then using a single derivative noun-form (*-isation/-ization*) for both process and end-result. From *parameter* there therefore evolves the verb *parameterise*, then *parameterisation*, defined by *Wikipedia* as the 'action of defining any set of factors whose values determine the characteristics of behavior of a program'. And a *quantum* is no longer simply an infinitesimally small lump of material or data, whether recorded in the act of moving or not; it now actively undergoes *quantisation*.

b Noun/verb names

If recognized verbs are used as nouns, equally what may first appear to be nouns are also verbs: 'all nouns can be verbed'.[25] Many e-neologisms have formally recognised dual noun + verb functions (e.g., *codesign*, *diff*, *disuser*, *gritch*, *sheepdip*, *skew*, *traceroute*), and many more that are still emerging in the literature are not yet formally dictionarised (e.g., *backhoe*, *miscompare*, *traceback*, and *turbocode*). A question, for others with a greater penchant for quantification than myself, is whether in e-English new dual-function noun-verbs outnumber new single-function nouns and/or verbs.

c Noun collapses

New single nouns may be created by collapsing an adjective into an existing noun in one step

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

(*cryptanalysis*, *whitespace*), or two (*multipleaccess* → *multiaccess*, multipleobject → *multiobject*), or by collapsing two nouns into one (*bitwidth*, *ciphertext*, *dogcow*, *namespace*, *timestamp*, *webgraphics*, *zoneplate*).

d Noun strings

'Strings' are central to computing, so long noun strings come as no surprise, as with *face recognition committee machine*. Many successful noun strings are later collapsed into acronyms for reasons of speed and efficiency: *FRCM*. An intriguing question, since many of these noun strings emanate from L2 English speakers in East Asia, concerns possible L1 influence on these constructions – an issue for further research.

e Reversing states

Nominal and adjectival reversals normally use *a-*, *in-*, *non-* or *un-* to reverse verbs as in the language at large. *Notwork*, however, reveals how e-humour can harness the tersest possible description of a non-working network that is 'down'.

10 Adverbs

e-Adverbs are generally constructed from nouns or adjectives, as follows: *bitwise*, *pairwise*, *piecewise*; *boundedly*, *statically*, *linearily*, *superlinearly*, *hardwarily*, *softwarily*. The form *hackerly*, however, follows *gentlemanly* in being adjectival. In e-English, adverbs appear to be both used and constructed much less frequently than verbs, nouns, and adjectives, perhaps because *-ise* verbs are becoming binary and therefore in less need of (graduated) qualification.

11 Articles

A student recently sought my advice on which article she should use in front of *servlet*, in the context of a network within which definite information is *randomly* packetised and transferred indeterminately among numerous computers, servers, and servlets in the network before arriving at its end-destination. What advice could I give – *L'article est mort, vive l'article?* Or perhaps the hitherto rarest article (*any*) may anticipate greater popularity in the future?

12 Greek influences

a Size prefixes

Systeme International defines standard metric prefixes for scientific measurements, from $\log 10^{-8}$ to $\log 10^{8}$. With two Danish exceptions (*atto-* and *femto-*), these are Greek,[26] as are many other widely-used prefixes (e.g., *hyper-*, *hypo-*, *meta-*, *pseudo-*) as well as many whole words (*anatomy, hierarchy*, *physiology*). Latin also contributes prefixes, mainly relational (e.g., *bi-*, *inter-*, *pre-*, *post-*, *semi-*, *sub-*, *super-*) and of course the promiscuous *multi-* which couples with nouns, verbs and adjectives alike: *multibus*, *multicarrier*, *multicast*, *multihomed*, *multihop*, *multilayer*, *multimodal*, *multipath*, *multiplatform*, *multiplexing*, *multiplexor*, *multirate*, *multistage*, *multistar*, *multiuse*, *multiuser*.

b 'greeking'

> To display text as abstract dots and lines in order to give a preview of layout without actually being legible. This is faster than drawing the characters correctly which may require scaling or other transformations. Greeking is particularly useful when displaying a reduced image of a document where the text would be too small to be legible on the display anyway.

13 Multiform neologisms created around core words

These are mainly nouns, but can also be adjectives and noun-verbs. Some use technical terms as their foundation, as with *band* (*bandlimited*, *bandpass*, *bandwidth*, *baseband*, *broadband*, *in-band*, *narrowband*, *out-of-band*, *stopband*, *wideband*) and *bit* (*bitwidth*, *bitmap*, *bitwise*).

Others borrow common words for new technical applications, such as *net* (*cheapernet*, *ethernet*, *extranet*, *fleenet*, *footnet*, *gripenet*, *intranet*, *honeynet*, *papernet*, *piconet*, *scatternet*, *shufflenet*, *sneakernet*, *subnet*, *thicknet*, *thinnet*, *toasternet*, *tubnet*) or *web* (*webcam*, *webcast*, *webhead*, *weblint*, *webmaster*, *webmistress*, *webmonkey*, *webpage*); or extend existing meanings, as for example of *code* (*bytecode*, *codebook*, *codesign*, *codewalker*, *pseudocode*, *turbocode*).

A third category is created from verbs, nouns and adjectives given an appropriate suffix, such as *-ware* (*abandonware*, *adware*, *annoyware*, *bloatware*, *brochureware*, *callware*, *careware, charityware, courseware, crippleware,*

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

Appendix 1: e-Verbs (in mid-position), with synonyms (preceding) and binary opposites/antonyms (where appropriate), many formed by adding generic reversors

Synonym	VERB	Antonym (binary opposite)
admit	accept	reject
	activate	deactivate
	allow	disallow
	approve	disapprove
	arrange	NIL
	assemble	disassemble
	automate	de-automate
	batch	unbatch
jump, split	branch ('goto')	merge
stop temporarily, halt	break	resume
	bug	debug, misbug
	cache	flush
	churn	
	clean	
	clear	restore
quit	close	open
	collect	despatch
	command	
	compare	miscompare
	compress	decompress, uncompress
	compute	
	configure	unconfigure
	connect	disconnect
	consolidate	
build	construct	deconstruct
	contain	
	convert	unconvert
	copy	
	customise	standardise
	cut	paste
	define	undefine
	delay	
	delete	undelete, restore
	delimit	undelimit
	delineate	
	design	
	designate	undesignate
	despatch	collect
	detect	undetect
	divert	
	downlink	uplink
	draw	undraw
	edit	
	empty	fill
	enter	exit, quit
	evolve	
timeout	expire	extend
	feature	misfeature
	file	
	filter	
	find	lose
	flood	
	form	unform
	format	unformat
	formulate	
	frag(ment)	defrag(ment)
	function	
	group	ungroup
	hack	
	hide	show, view
	import	export
	inform	
	initialise	
	insert	withdraw
	install	uninstall
	jitter	
merge	join	depart, leave
open	launch	quit
	limit	

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

Appendix 1 continued

Synonym	VERB	Antonym (binary opposite)	Synonym	VERB	Antonym (binary opposite)
	list	unlist, delist		script	
	load	unload		search	
	locate			seek	
	login	logout		select	deselect
	loop	unloop		serialise	deserialise
	merge	branch		serve	
	migrate			set	unset
	model			shift	
	monitor			signal	
	move	freeze, halt, stop	emulate	simulate	
	mutate			skew	straighten
	operate			sort	unsort
	plug	unplug		spam	
	print			store	
	program(me)	unprogram(me)		stream	unstream
	protect	unprotect		string	unstring
	pull	push		struct	destruct
isolate	quarantine			synthesise	analyse
	queue	unqueue		tabulate	
	receive	send, transmit		test	
	recognise	unrecognise		timeout	
	reference	dereference		traceback	traceforward
	refresh			track	
	register	unregister		transfer	
	repair			transform	untransform
	repeat			transpose	untranspose
	replace			use	
	restore			validate	invalidate
	route			wait	
	run			wire	unwire
	save			zoom in	zoom out

crudware, firmware, freeware, fritterware, guiltware, hardware, hyperware, malware, meatware, microware, middleware, nagware, netware, postcardware, shareware, shelfware, shovelware, slackware, software, spyware, suicideware, vapo(u)rware, wetware).

Some inconclusive questions: teaching implications

I have tried to demonstrate here that, in conceptualizing and naming their new activities and products, computing experts are evolving, rather than merely mutating, the rules of Eng-

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

lish grammar in systematic ways. When teaching the course 'English for Information Technology', I have been asked grammatical questions about e-applications to which there are, at best, only speculative answers. From the puzzlement on my students' faces, I have the uneasy feeling that they are probably learning to compartmentalize what they learn in English classes from what they learn in computing courses.

To me, one thing is clear. e-English is *not* a perhaps-amusing 'dialect'. The internet is the future of virtually all forms of communication, written and verbal. If it renders the pre-electronic rules of English redundant, non-hackers will be able to do little about that, except (willingly or unwillingly) learn a new language. ■

Notes

1 'The physical world (as opposed to virtual reality), where you might spend facetime with the carbon community' (FOLDOC, 1999-01-15).
2 'To temporarily enter techno-nerd mode while in a non-hackish context, for example at parties held near computer equipment. Especially used when you need to do or say something highly technical and don't have time to explain: "Pardon me while I geek out for a moment".' (FOLDOC, 1995-01-18) An alternative subtitle would be 'Greeking e-English': see below.
3 For example: 'Currying is necessary if full laziness is to be applied to functional sub-expressions' (FOLDOC, *curried function* 2002-07-24). *The Jargon File* Version 4.4.7 (*comparatives*) asserts that 'hackers have even more words for equipment failures than Yiddish has for obnoxious people'.
4 Because *Inglish* is already the name of an early programming language (see FOLDOC).
5 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/writing-style.html
6 FOLDOC, hacker humour (1995-12-18); *The Jargon File* Version 4.4.7
7 These examples would probably not be cited by younger computing enthusiasts, whom *The Jargon File* Version 4.4.7 identifies as having 'their own characteristic jargon, heavily influenced by skateboard lingo and underground-rock slang'. (*The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/crackers.html)
8 Appropriately, lexeme itself is also a 'collapse': lex(icon) + eme.
9 FOLDOC, lexeme (1996-04-06).
10 <http://en.*wikipedia*.org/wiki/Spelling_reform>
11 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/crackers.html
12 When artificial intelligence finally succeeds in designing analog computers of comparable speed and capacity to those of the digital revolution, this demand may change.
13 Appendix 1 shows this problem for the most frequently used verbs in computing.
14 A free translation of this last clause is: and differentiates continue from make a mess.
15 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/writing-style.html
16 'The Blogging Glossary editors are God and God moves in mysterious ways. If you have a suggestion, amendment, correction or new entry, preferably attributed, that you think belongs in the glossary, e-mail it to us and we might publish it. Or not.' http://www.samizdata.net/blog/glossary.html
17 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/anthropomorphization.html
18 FOLDOC, makefile (1995-01-05).
19 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/overgeneralization.html
20 Re the acceptability of 'finalize', the online edition of *The American Heritage Dictionary of the English Language* (4th Edition) has an interesting Usage Note:

> Once considered objectionable because of its association with the language of bureaucracy, *finalize* is steadily gaining acceptance. In the late 1960s, 90 percent of the Usage Panel found the example *finalize plans for a class reunion* unacceptable; in the late 1980s, 71 percent disapproved. By 1997, only 28 percent of the Usage Panel found it unacceptable in the sentence 'We will send you more information once we finalize plans for the reunion.' Although substitutes for *finalize* can be found among *complete*, *conclude*, *make final*, and *put into final form*, none of these is an exact synonym. This may be why resistance to *finalize* is eroding.

21 Similarly for nouns, both *copyleft* and *copywrong* have been suggested as opposites for *copyright*, and both *polygoose* and *mongeese* as opposites for *mongoose*.
22 As at 2005-01-11.
23 In citing *-ise/-ize* as among 'the unimaginative verb-making techniques characteristic of marketroids, bean-counters, and the Pentagon', *The Jargon File* appears to have overlooked its interim function in the longer verb-evolving process! http://catb.org/~esr/jargon/html/overgeneralization.html
24 *The Jargon File* Version 4.4.7: http://catb.org/~esr/jargon/html/overgeneralization.html
25 FOLDOC, prefix (2003-05-06). Greek size prefixes are also being incorporated into Japanese "otaku jargon", according to Mark Schreiber (2005).
26 FOLDOC, greeking →greek (v) (1999-06-27).

References

Berry, M. J. A., & G. S. Linoff. 2000. *Mastering data mining*. New York: John Wiley.

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press

Bruza, P., Song Dawei, & Wong Kam-Fai. No date. 'Fundamental Properties of Aboutness': .pdf file, downloaded 17 Jan 05.
Fujita, S. No date. 'Reflections on "Aboutness".' TREC-9 Evaluation Experiments at JustSystem: .pdf file, downloaded 17 Jan 05.
Ghosh, A. K. 1998. *E-commerce Security: Weak links, best defences.* New York: John Wiley.
Hornby, A. S. 2000. *The Oxford Advanced Learner's Dictionary of Current English*, 6th edn, ed. Sally Wehmeier, phonetics ed. Michael Ashby. Oxford: University Press.
Howe, Denis. 2005. *Free Online Computing Dictionary (*FOLDOC*).* Host: Imperial College, London.
Lehman, M. M., J. F. Ramil, & G. Kahen. 2000. 'Evolution as a noun and evolution as a verb.' Paper presented to the Software and Organisation Co-Evolution Workshop, Imperial College, London, Jul 00.
Schreiber, M. 2005. 'Getting ready to gab with a geek.' *The Japan Times*, 2 Oct 05, p. 19.
Wang, W. S. Y. 1989. Language prefabs and habitual thought.' Forum Lecture to the TESOL Summer Institute, San Francisco State University. In *Other Studies* pp. 397–412.
Also:
The American Heritage Dictionary of the English Language, Fourth Edition. 2000. online. New York: Houghton Mifflin.
*The Jargon File (*Version 4.4.7*),* online.
WordNet 2.0. 2003. Princeton University (online)

Online references

<http://catb.org/~esr/jargon/html>
<http://en.wikipedia.org/wiki/Spelling_reform>
<http://www.samizdata.net/blog/glossary.html>
<http://wombat.doc.ic.ac.uk/foldoc>
<www.webopedia.com/Networks/>
<www.webopedia.com/Networks/Network_Protocols/>
<www.webopedia.com/Networks/Network_Typologies/>
<www.webopedia.com/Networks/Routing/>
<www.webopedia.com/Networks/Security/>

POST & MAIL

Tolkien: The video game?

From John Algeo

Bravo to Ross Smith for his assessment in *ET*83 of *The Lord of the Rings* films. To appreciate Smith's essential insight, one need not agree with every detail in his spot-on observation that these cinematographic new clothes are really naked stuff. *The Lord of the Rings* (the book) is philosophical, poetic, meditative, archetypal, and psychologically perceptive. The film looks like something made by a bunch of young men who grew up playing video games. That doesn't make it a bad film, if you like video games; but it does make it a betrayal of Tolkien's book, which is of a fundamentally different kind. The critics who heaped unstinting praise on the film's conglomeration of special effects and battle scenes need to find an occupation more suited to their perceptions and taste.

https://doi.org/10.1017/S0266078406001040 Published online by Cambridge University Press