Naming Names

Brian Hayes

Adam's only chore in the Garden of Eden was naming the beasts and birds. The book of Genesis doesn't tell us whether he found this task difficult or burdensome, but today the need to name and number things has become a major nuisance. When you try to choose a name for a new Internet domain or an e-mail account, you're likely to discover that your first choice was taken long ago. One Internet service tells me the name "brian" is unavailable and suggests "brian13311" as an alternative. Perhaps I should think of this appellation in the same category as Louis the 18th or John the 23rd, but being Brian the 13,311th seems a dubious distinction.

The challenge of inventing original names is particularly acute when the name has to fit into a format that allows only a finite number of possibilities. For example, the ticker symbols that identify securities on the New York Stock Exchange can be no more than three characters long, and only the 26 letters of the English alphabet are allowed. The scheme imposes an upper limit of 18,278 symbols. If the day ever comes that 18,279 companies want to be listed on the exchange, the format will have to be expanded. And long before that absolute limit is reached, companies could have a hard time finding a symbol that bears any resemblance to the company name.

It's not just names that are scarce; we're even running out of numbers. A few years ago telephone numbers were in short supply, and so were the numbers that identify computers on the Internet. Those crises have abated, but now attention has turned to the Universal Product Code, the basis of the barcode labels found on virtually everything sold in the United States and Canada. It seems the universe has more products than the UPC has code numbers. For that reason and others, the 12-digit UPC standard is being supplanted by a 13-digit code, with provisions for adding a 14th digit. The "sunrise" date for this transition is January 1, 2005. The old 12-digit codes will continue to be recognized, so you may not notice an immediate change on product labels, but every supermarket and drug store has had to modify its database software to accommodate the extra digits. Some commentators have drawn parallels with the year 2000 rollover, when software had to be patched to deal with four-digit year numbers. That event was a fizzle, anxiously anticipated but with little real disruption on January 1, 2000. This time there has been little advance publicity, so perhaps we should brace for turmoil in the checkout line.

Finishing Adam's Job

Names and numbers were causing trouble long before the Internet age. Biology had a naming crisis in the 17th and 18th centuries. The problem wasn't so much a shortage of names but a surfeit of them: Plants and animals were known by many different names in different places. Then came the great reform of Carolus Linnaeus and his system of Latin binomials, identifying each organism by genus and species. The new scheme revolutionized taxonomy, not because there is any magic in Latin or in two-part names but because Linnaeus and his followers labored to preserve a strict one-to-one mapping between names and organisms. Official codes of nomenclature continue to enforce this rule—one name, one species—although rooting out synonyms and homonyms is a constant struggle.

Linnaeus himself named some 6,000 species, and by now the number of living things in the biological literature is approaching two million. But there could be another 10 million species—or, who knows, even 100 million—yet to be catalogued. Might we run out of names before all the species are described? If we were to insist that every binomial consist of two real Latin words—words known to the Romans—then perhaps there might be trouble ahead. But in practice Linnaean names only have to *look* like Latin, and the only limit on their proliferation is the ingenuity of the biologist. A dictionary of classical Latin will not help you understand the terms *Nerocila* and *Conilera*, which designate two genera of isopods; more helpful is knowing that the biologist who invented the terms was fond of someone named Caroline.

Among all the sciences, the one with the most remarkable system of nomenclature is or-

Brian Hayes is Senior Writer for American Scientist. *Address: 211 Dacian Avenue, Durham, NC 27701; bhayes@amsci.org*

ganic chemistry. Names in most other realms are opaque labels, which identify a concept or object but tell you little about it. For most of us, a Linnaean name such as *Upupa epops* doesn't even reveal whether the organism is animal or vegetable (this one's a bird). In contrast, the full name of an organic compound specifies the structure of the molecule in great detail. "1,1-dichloro-2,2-difluoroethane" is a prescription for drawing a picture of a Freon molecule. The mapping from name to structural diagram is so direct that it can be done by a computer program. The reverse transformation, from diagram to name, is trickier; in other words, it's easier to make the molecule from the name than the name from the molecule.

Exhausting the supply of names for organic compounds is not something we need to worry about: By the very nature of the notational system, there is a name for every molecule. On the other hand, the names can get so long and intricate that only a computer can parse them.

Namespace

Although difficulties with names are nothing new, the nature of name-giving changed with the introduction of computer technology. There is greater emphasis now on making names uniform and unique. Second, many names and identifying numbers must conform to a rigid format, with a specified number of letters or digits drawn from a fixed alphabet.

Place names—and abbreviations for them—offer a good example of how names have changed. In the old days, a letter from overseas addressed to the "U.S." or the "U.S.A." or even the "EE.UU." would stand a chance of being delivered, but e-mail for the corresponding geographic domain must have the exact designation "US"; no variation is tolerated (except that upper case and lower case are not distinguished). The list of acceptable country codes for Internet addresses is maintained by the Internet Assigned Number Authority (IANA). Each code consists of exactly two characters, drawn from an alphabet of 26 letters. Thus the number of available codes—the total namespace—is 26 × 26, or 676. The current IANA list has 247 entries, so the filling factor—the fraction of the space that's occupied—is 0.365. That leaves room for growth if a few more nations decide to deconstruct themselves the way Yugoslavia and the Soviet Union did. But not every nation can get its first choice code.

Consider the case of the Åland Islands, which, according to the Web site www.aland.fi, "form an autonomous, demilitarized and unilingually Swedish province of Finland." The islands are sufficiently autonomous to have persuaded IANA to issue them a country code of their own—but *which* code? Perhaps the first choice would have been AL, but Albania already had that one. Or maybe AI, if Anguilla hadn't claimed it. Why isn't Anguilla AN? Because that's the code for the Netherlands Antilles (which might have been NA if it weren't for Namibia). The preemption of AN also leads to less-than-obvious assignments for Andorra, Angola, Antigua and even Antarctica. In the end, the Ålanders have wound up with the code AX (although, as the address www.aland.fi indicates, not everyone uses it).

There is more to say about the difficulty of finding an unused name as a namespace fills up. But first some more examples of finite namespaces:

Stock market ticker symbols. Ticker symbols began as telegraphers' informal shorthand, but today they are registered with the various exchanges. The New York Stock Exchange and the American Stock Exchange share a namespace; no symbol is allowed to have a different meaning in the two markets. Ignoring certain minutiae, the symbols consist of one, two or three letters; thus the size of the namespace is $26^3 + 26^2 + 26 = 18{,}278$. The listing I consulted (at www.commerce-database.com) had 3,926 active symbols, for a filling factor of about 0.22. Stocks traded on the NASDAQ market use four-letter symbols. There are fewer of these stocks (about 3,400) and a much larger namespace (456,976), so it should be considerably easier to find a symbol for a new company there. (The most notable recent addition is Google, which chose the symbol GOOG.)

Telephone numbers. Telephone numbers in North America have 10 decimal digits (including the area code), which suggests that the capacity of the namespace should be 10 billion numbers. Under the rules prevailing through the 1980s, however, fewer than a tenth of those combinations were valid telephone numbers. The format of a phone number in those days was expressed as NZX-NNN-XXXX, where N represents the digits 2–9, Z the digits 0–1 and X any digit in

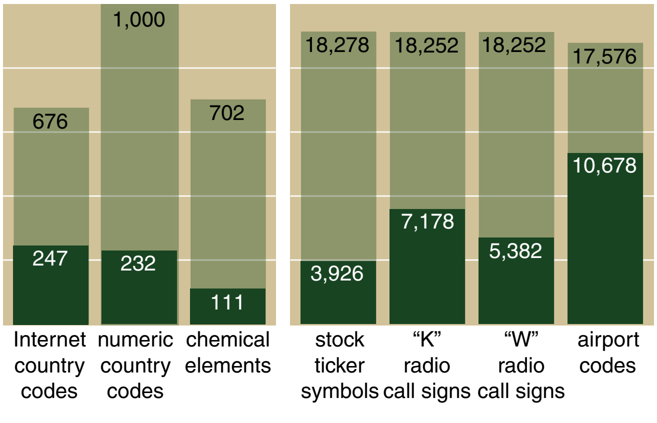

Figure 1. Constraints on the size and format of a name, numeric label or abbreviation create a finite "namespace," with room for only a fixed number of combinations. The bar graphs show the total capacity of a few namespaces and the level to which they are currently filled. (Left and right portions of the graph have different scales.) From left to right the namespaces are two-letter country codes for Internet domains, three-digit numeric country codes assigned by the United Nations, the symbols of elements in the Periodic Table, ticker symbols of stocks on the New York and American exchanges, call signs of American radio stations beginning with "K" and with "W" and finally the three-letter codes assigned to airports.

the full range 0–9. That works out to about 819 million numbers. Even that quantity should be plenty; there are roughly 300 million telephones in use in the United States. Nevertheless, during the early 1990s the supply of numbers within many area codes came close to exhaustion. Although the crisis was often blamed on the proliferation of modems, fax machines and cellular telephones, the real culprit was an inefficient scheme of allocation: If a telephone company had even one subscriber within a region, the company was assigned a block of 10,000 numbers. The main remedy was allocating numbers in smaller blocks, but along the way the grammatical rules defining a telephone number were relaxed, and the namespace expanded. Any combination of digits of the form NXX-NXX-XXXX is now a valid phone number, allowing some 6.4 billion possibilities. With careful conservation, the supply is expected to last until sometime in the 2030s.

Product codes. As in the telephone system, the shortage of Universal Product Codes is partly a matter of allocation policy. Although a UPC number has 12 digits (implying a maximum capacity of a trillion items), the first digit is a category code that in practice is almost always 0, and the final digit is a checksum used for detecting errors. Of the remaining 10 digits, 5 identify the manufacturer and 5 the individual product. Because of this fixed structure, every manufacturer automatically gets a block of 100,000 item numbers, even though most companies need far fewer. The new 13-digit standard coming into force on the first day of 2005 not only expands the total namespace by a factor of 10 but also allows a more flexible division of resources. In particular, some companies will be given a longer manufacturer code and fewer item codes.

The new product-code standard isn't really new. The United States and Canada are merely acceding to another standard, called the European Article Number, that is already in use almost everywhere else in the world. (How quaint that the scheme known only in part of North America is the one labeled "Universal.") After the merger, the entire suite of product codes will be renamed the Global Trade Item Number. Most of the bar-code scanning devices at checkout counters have long been able to read the 13-digit EAN format, but in many cases the database in the back office could not handle the extra digit. While making the necessary conversions, retailers have been urged to allow space for a 14-digit version of the GTIN. In 2007 publishers and libraries will get their turn to renumber their world as the International Standard Book Number is expanded to 13 digits and brought under the GTIN umbrella.

Social Security numbers. With nine-digit decimal numbers, there should be a billion possibilities. The Social Security Administration has excluded only a few of them ("No SSNs with an area number of '666' have been or will be assigned"), so that the actual size of the namespace appears to be 987,921,198. Some 415 million numbers have been issued since in 1936, for a filling factor of about 0.4. The supply of numbers may well outlast the supply of funds to pay benefits.

Other countries have quite different systems for allocating numbers analogous to the U.S. Social Security number. In particular, the Italian *codice fiscale* is not an arbitrary number assigned to a person but rather a string of alphanumeric symbols calculated from personal data such as name

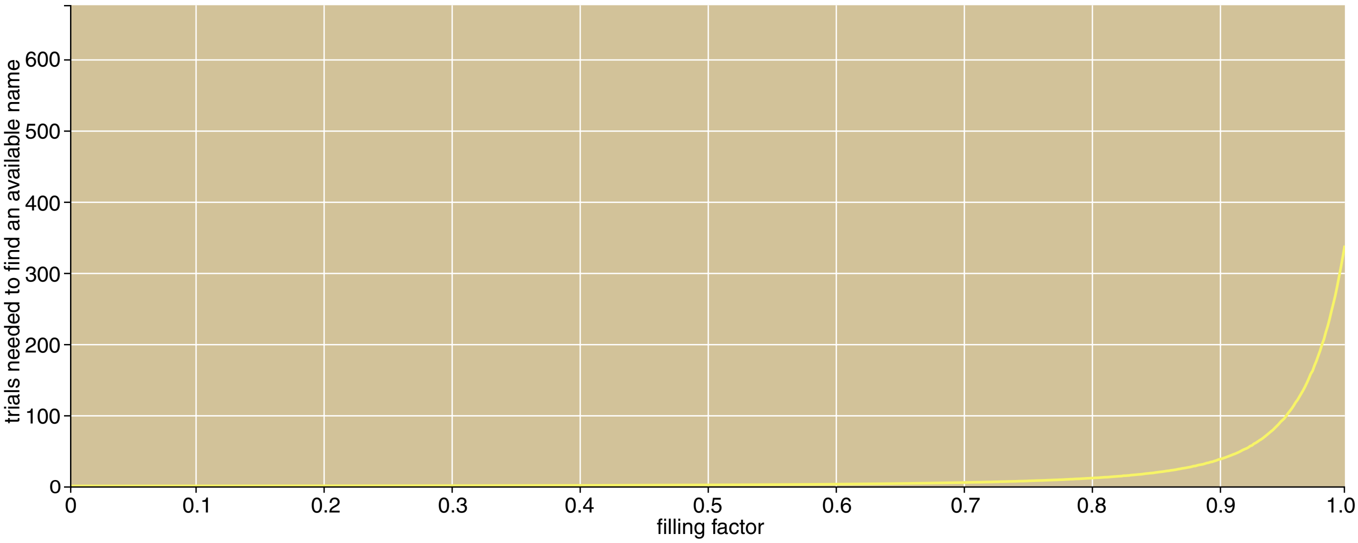

Figure 2. Simulation of the filling of a namespace suggests that finding a unique name becomes impractically difficult when the space is much more than half full. Starting with an empty namespace of 676 slots, the simulation adds names one at a time. First a name is generated at random; if the corresponding slot is empty, it is marked as filled. If the slot is already filled, the program tries the next slot in alphabetic sequence and continues in this way (wrapping around to the beginning of the space if necessary) until coming to an empty slot. For each name added, a dot is marked on the graph at the horizontal position corresponding to the filling factor at that moment and at the vertical position indicating the number of slots checked. The spray of green dots superimposes 40 repetitions of the entire process of filling the namespace; the yellow line averages the results of 100,000 trials.

and date and place of birth. This scheme eliminates all concerns over running out of numbers, but it has another potential hazard: If the algorithm for calculating the *codici* is not chosen very carefully, two individuals may wind up with the same number.

Radio station call signs. Broadcast radio stations in the United States have call signs of either three or four letters, but the first letter is always either K or W. These rules create a namespace with room for 36,504 entries. I was surprised to discover how densely filled this space is. Combining the AM and FM bands (many stations broadcast on both), there are 12,560 call signs currently registered with the Federal Communications Commission, a filling factor of more than one-third.

Airport codes. When you check a bag at the airport, the luggage tag is marked with a three-letter code that indicates where, if all goes well, you'll eventually retrieve your belongings. The codes are administered by the International Air Transport Association (IATA). There's a code for every airport that has airline service, not to mention a few bus and train stations. Surprisingly, the IATA codes are the most densely packed of all the naming schemes I have encountered. Out of 17,576 possible codes, 10,678 are taken, a filling factor of 0.6. This may be why some of the codes are less than obvious (YYC for Calgary?), although many such minor mysteries have historical explanations. Chicago's O'Hare airport is ORD because it was once called Orchard Field.

Making Hash of a Name

Suppose you've just built a new airport or radio station or founded a sovereign nation, and you want to register an identifying code with the appropriate agency. What is the likelihood that your first choice will be available? Or your second or third choice? How do these probabilities change as the namespace fills up?

If we can make the assumption that preferences for codes are distributed randomly throughout the namespace, then the question is easily answered. The probability that your first choice is already taken is just the filling factor of the namespace. The probability that both your first choice and your second choice are taken is the square of the filling factor, and so on. For example, if the namespace is two-thirds filled, then in two-thirds of the cases a randomly chosen code will already be present; four-ninths of the time, two randomly generated codes will both be taken.

Searching at random for an unused name is related to the process known in computer science as hashing. The idea of hashing is to store data items for quick retrieval by scattering them seemingly at random throughout a table in computer memory. The arrangement isn't *truly* random; each item's position is set by a deterministic "hash function." Sometimes the hash function sends two data items to the same location; the collision must be resolved by putting one of the items elsewhere. This is analogous to requesting your favored name or code and finding that someone else has already claimed it.

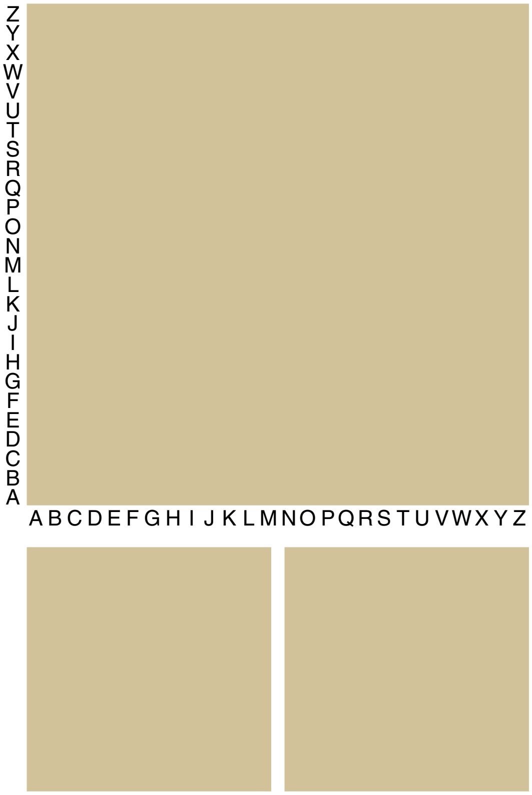

Figure 3. Names are distributed nonrandomly in real namespaces. The large tableau at the top has a colored dot for each two-letter country code in the list maintained by the Internet Assigned Number Authority. The table is read from bottom to top and left to right; thus AZ (Azerbaijan) is at the lower right and ZA (South Africa) at the upper left. Color is determined by the product of the number of occupied cells in each dot's column and row, so that brighter colors call attention to letters that are particularly popular. Of the two smaller tableaux below, the one at left is based on another set of two-letter abbreviations for countries, published as a Federal Information Processing Standard and used by the Postal Service; the structure of clusters is similar even though many of the individual codes are different. For comparison, the small tableau at right has random entries.

The resemblance between name search and hashing is worth noting because the performance of various hashing algorithms has been carefully analyzed and documented. Much depends on the strategy for resolving collisions, or, in the context of name search, the policy for choosing an alternative when a desired name is not available. Figure 2 reports the results of a simulation of a name search equivalent to one of the simplest hashing methods. The rule here is to generate a first-choice name at random; if that choice is taken, try the next name in alphabetical order and continue until an opening is found. Naturally, the number of collisions increases as the namespace fills up, but the increase is not linear; the shape of the curve is concave upward. Thus at any filling factor below about one-half, there is a reasonable chance you will get one of your first few choices. At higher filling factors, the average number of attempts before you find an available name rises steeply.

But there is a flaw in this analysis: The assumption that preferences for names are random is obviously bogus. People prefer names that appear to mean something or that have some trait that distinguishes them from random strings of symbols. In the stock market, the rare one-letter ticker symbols carry much prestige; radio call signs that spell a pronounceable word (WARM, KOOL) are in demand. It would be difficult to codify or quantify these biases, but as a simple way of estimating their effect I tried looking at the first-order statistics of the code words in various data sets. The first-order statistics are simply the letter frequencies at each position within a word. (Higher-order statistics take into account correlations between the letters.)

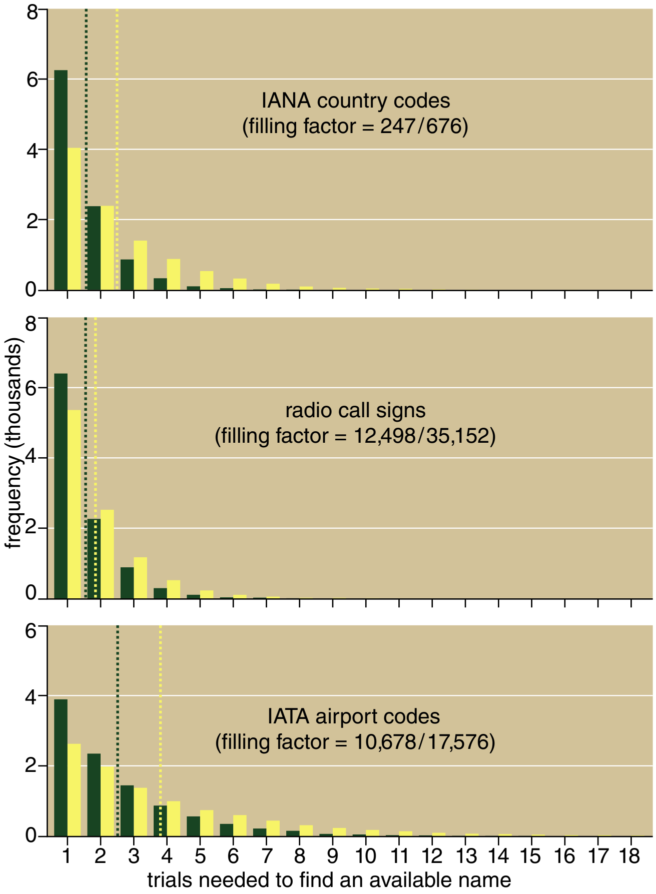

Figure 4. Statistical bias within a namespace makes the search for an unclaimed name even harder. Each of these graphs records the results of 10,000 independent attempts to add a single new name to an existing namespace. The height of the bars indicates the number of times the attempt succeeded on the first try, the second, the third and so on. Green bars are for names generated completely at random, yellow bars for names with the same first-order statistics as the names already in the data set. The dotted green and yellow lines give the average number of attempts needed to find an open slot. In the case of airport codes, for example, it took about 2.5 trials on average to find an unused random code but 3.9 trials with codes that reflect the biased letter frequencies.

My experiments compared the success of two players—one who chooses names utterly at random and another whose random choices are biased to match the statistics of the names already in the data set. In other words, the latter player tends to favor names that are like those already present. Not surprisingly, the random player has an easier time finding an available name. The magnitude of the effect can be quite large. In the case of IANA country codes, random choices succeed after an average of 1.6 probes, but finding a name with letter frequencies similar to the existing population takes 2.5 trials on average. For IATA airport codes, the statistical bias raises the average number of attempts from 2.5 to 3.9. These results suggest that some namespaces may become impractically full much sooner than would be expected from an analysis based on hashing algorithms.

The experiment itself has a curious bias. Using an existing data set to infer people's preferences neglects the fact that many of the code words may not have been anyone's first choice; they may have been selected merely because the real first choice was already taken. Furthermore, the statistical bias varies with the filling factor. If there are only a few names in the data set, the letter frequencies will be strongly biased. Indeed, some letters may not appear at all, and so the algorithm used in the experiment would assign them a probability of zero. At the opposite end of the spectrum, variations in letter frequencies inevitably diminish as the namespace fills up. Once almost all the code words are taken, all letters must have nearly the same frequency.

Horse Sense

As namespaces get larger, analyses based on random character strings become less illuminating. A case in point is the naming of thoroughbred horses. Under rules enforced by the Jockey Club, a horse's name can have from 2 to 18 characters, drawn from an alphabet consisting of the usual 26 letters plus the space, the period and the apostrophe. This is an enormous namespace, with room for more than 2×10^{26} entries. At any one time there are about 450,000 names assigned to active or recently retired horses. Most of these names will eventually become available for reuse, and so the pool of active names stays at roughly constant size. (Only the names of very famous steeds are permanently withdrawn; there will never be another Kelso or Secretariat.)

With just 450,000 of 2×10^{26} slots occupied, the filling factor of this namespace might as well be zero. Generating strings of characters at random, you would have to try 10^{21} of them before you would have much chance of stumbling on a name in use. And yet real-world experience gives a very different impression. Of all the names submitted by horse breeders, the fraction rejected is not 1 in 10^{21} but close to 1 in 4. According to a spokesperson for the Jockey Club, the most common reason for rejection is that the proposed name is too close to an existing one. In this context names can clash even if they are not spelled identically—mere phonetic similarity is enough to bar a name. But even allowing for this broader criterion of uniqueness, the thoroughbred namespace is not nearly as empty as it would seem from a naive counting of character strings. A fair estimate of the true filling factor would probably have to be based not on the combinatorics of random letters but on combinations of words or some other higher linguistic unit.

The same is surely true for Internet domain names. Each component of a domain name—each part between dots—can have up to 63 characters, and the acceptable characters include both letters and numbers as well as the hyphen. The size of the namespace is nearly 10^{100}; we won't use them all up anytime soon. But meaningful, pithy, clever domain names—that's another matter.

Even outside the confines of finite namespaces, the sheer onomastic challenge of modern life sometimes gets to be a burden. Where's Adam when we need him? Years ago, I could save a clipping from the newspaper without any need to name it. Now, for every document I create or choose to keep, I must enact a little ceremony of naming: I dub thee "FILE-037.TXT." The workload has gotten serious enough that consultants make a living out of nothing more than dreaming up names. (One firm named itself A Hundred Monkees—well named!)

When my daughter was a voluble three-year-old, she would greet passersby with the enthusiastic salute: "Hi! My name is named Amy. What is your name named?" A dizzying recursion yawns before us. Once we start naming names, and then the names of names of names, where do we ever stop?

Bibliography

The Airline Codes Web Site. http://www.airlinecodes.co.uk/

Book Industry Study Group. 2004. The evolution in product identification: Sunrise 2005 and the ISBN-13. http://www.bisg.org/docs/The_Evolution_in_Product_ID.pdf

Federal Communications Commission. Undated. Index of Media Bureau CDBS public database files. http://www.fcc.gov/mb/databases/cdbs

Garfield, Eugene. 1961. *An Algorithm for Translating Chemical Names to Molecular Formulas*. Doctoral dissertation, University of Pennsylvania. http://www.garfield.library.upenn.edu/essays/v7p441y1984.pdf

Jeffrey, Charles. 1973. *Biological Nomenclature*. New York: Crane, Russak & Co.

The Jockey Club. 2003. *The American Stud Book: Principal Rules and Requirements*. Lexington, Ky.: The Jockey Club. http://www.jockeyclub.com/pdfs/RULES_2003_PRINT.pdf

Knuth, Donald E. 1973. *The Art of Computer Programming. Vol. 3: Sorting and Searching*. Section 6.4, Hashing. Reading, Mass.: Addison-Wesley.

McNamee, Joe. 2003. Why do we care about names and numbers? http://www.circleid.com/article/336_0_1_0/

Mockpetris, P. 1987. Domain names: implementation and specification. Network Working Group Request for Comments 1035. http://www.ietf.org/rfc/rfc1035.txt

NeuStar, Inc. 2003. *North American Numbering Plan Administration Annual Report, January 1–December 31, 2003*. http://www.nanpa.com/reports/2003_NANPA_Annual_Report.pdf

Savory, Theodore. 1962. *Naming the Living World: An Introduction to the Principles of Biological Nomenclature*. London: The English Universities Press.

Uniform Code Council, Inc. Undated. 2005 Sunrise: Executive summary. http://www.uc-council.org/ean_ucc_system/stnds_and_tech/2005_sunrise.html