

On naming

It is impossible to do mathematics without introducing names. The main issues seem to be what things to name and how to name them. An important distinction seems to be whether a name occurs primarily in formulae to be manipulated or is to be embedded in sentences. We shall explore the latter context first.

Names in a linguistic context

One should be cautious when introducing a "meaningless" identifier that is identical to a word in the language of the surrounding prose. Standard examples are the Dutch "U" -- U is een onsamenvangende graaf-- and the English "a" -- in order that a might be removed-- and "I" -- because I may be empty-- . On paper this problem can be alleviated by typographical means: the identifier can be made to stand out by the use of italics or an additional pair of surrounding spaces, but these distinctions are hard to render in the spoken word.

Great caution is called for when giving a normal word a specific technical meaning. Naming a set "checked" or "preferred", a boolean "found", a logical connector "defines", or a type "chosen" all but precludes the use of these normal words in their traditional meaning.

But even if an author explicitly states that some common words will only be used in his technical sense and faithfully sticks to that discipline, the choice of

such "meaningless" identifiers requires great care, because their normal connotations can still invite confusion. For instance in a recent manuscript we encountered wires that could be "open" or "closed", but on closer scrutiny, wires could be in a third state, which had been left anonymous but could have been called "ajar". This naming would have been very faithful in the sense that transitions between "open" and "closed" only occurred via the state "ajar". Yet, the nomenclature open/closed was unfortunate, because too often the two are considered each other's negation. In situations like these we often benefited from the use of colours, e.g. red and blue for just exclusive states, yellow as the name of some property, and white, grey, and black when we wished to express a linear order. We also remember a graph in which the vertices were red or blue, thereby introducing red, blue, and violet edges. So much for the confusion that can be caused by the multiplicity of meaning of common words and by the author's inability to control which connotations they evoke.

A totally different concern is what we might call grammatical flexibility, i.e. does the term to be introduced have or admit the derivatives needed. To give an example of the problems one might run into, consider the adjectives simple and complex. Connected to the first we have the noun "simplicity", the verb "to simplify", and the noun "simplification"; for the other one we have the noun "complexity", the verb "to complicate", but we lack the analogous noun "complification", as the noun complication won't do: it refers to the result and not the act of complicating. Confronted with existing terminology, one may find oneself forced to coin the

the missing term, e.g. "to truthify" in the meaning of "to effectuate the transition from false to true" in analogy with "to falsify"; In choosing a term, the need for such coinages should be taken into account. (For instance, the fast general acceptance of the technical term "stack" could well be due to the fact that it is linguistically more comfortable than the older "push-down list".) So much for grammatical flexibility.

We would like to draw attention to a linguistic irregularity. In principle, adjectives and prefixes are constraining: a red dress is a dress, a wash-tub is a tub. Some prefixes, however, are generalizing and hence imply a hidden negation: a half-truth is not a truth. Natural languages can apparently live with this irregularity, but we cannot recommend it for the coinage of mathematical terms as they are specifically intended to be used in a logical context. So much for the semi-group that isn't a group and the weakly increasing sequence that isn't increasing.

Another fuzziness to be aware of is the natural use of "not", "and", and "or". The problem with "not" is that we cannot indicate its scope, vide "not retired or disabled" and "not uniformly convergent". The problem with "or" is that it is mostly exclusive, vide "Should I marry Jane or Ann?" and "Is it a boy or a girl?", but sometimes inclusive, vide "retired or disabled". The trouble with "and" is that it is sometimes conjunctive, vide "retired and disabled", and sometimes enumerative, vide "men and women". By the time we encounter "retired and disabled

persons qualify" we have had it. In Texas the law says "Stop while schoolbus is loading and unloading.", in Missouri it says "while loading or unloading.". Evidently natural languages are not well adapted to cope with these constructs. The terminology we are talking about being intended to be used in verbal reasoning, it pays to construct it in such a way that such constructs can be avoided. We have the trichotomy "less than", "equal to", and "greater than". It really helps to introduce new names for their negations, viz. "at least", "different from", and "at most" respectively. In this way one avoids "not greater than" and "less than or equal to" -- compare the latter one with "up to and including"! --. Similarly, "different from" enables us to avoid "not equal to", and the clumsy "less or greater than".

In short we suggest that when a concept is named it might be a good thing to give its complement an equally "positive" name. There are two reasons for this; firstly, sentences with too many negations quickly become utterly confusing, vide "I'm not sure that I don't dislike it if you are not present."; secondly, the negation as in "equal" versus "unequal" creates an asymmetry in the dichotomy that is often undesirable: it suggests that the positively named term is the more fundamental one. (Before you object by pointing out that equality is more fundamental than difference because equality is transitive while difference is not, please realize that "at most" is as transitive as its negation "greater than".)

To drive home the message we shall analyse the emergence of a well-known but unfortunate turn of

phrase. Corresponding to "greater than" and "less than" we have the "increasing sequence" and the "decreasing sequence" respectively. Let us now admit equality as well. In the absence of new terminology, "greater than" is then weakened to "not less than" or to "greater than or equal to". For a sequence with $X_i \leq X_{(i+1)}$ the corresponding characterizations a "not decreasing sequence" or an "increasing or constant sequence" are equally unacceptable. Mathematical parlance programmed around it by calling it a monotonically not increasing sequence. The whole problem is solved by introducing the adjectives "ascending" and "descending".

A final confession to end this section with. In a recent discussion of a permutation algorithm our elements began "unmarked" and ended up "marked". We have now several reasons for regretting this terminology: it destroys the symmetry, it is very operational, and is on the verge of introducing marks as mathematical objects.

Names in a formal context

In a formal context there should be no confusion as to what are names and what are more complicated expressions. Are P' , A^* , \bar{q} , \hat{r} , and \tilde{Q} identifiers or are they the result of applying special functions, denoted by $'$, $*$, $\bar{}$, $\hat{}$, and $\tilde{}$ to P , A , q , r , and Q respectively? Is P'' a name or is it equal to $(P')'$? Is subscription a mechanism for name construction, e.g. $\Delta A_1 A_2 A_3$, or is it a special notation for function application? The latter is the

only tenable interpretation if, after the introduction of the infinite sequence x_0, x_1, \dots , we encounter a reference to x_i . The difference seems slight, but notice that in the case of A_1, A_2 , and A_3 the name A is in the same scope available for other purposes, whereas in the case of x_0, x_1, \dots the name x is already in use as function identifier.

Aside. Note that such special function notations can create typographical and syntactic problems: the operator $\hat{}$ could have to be applied to an expression or, in general, to a multi-character argument, and what is the value of $\hat{\hat{\hat{}}}$ if those operators do not commute? Unbridled introduction of such special function notations is therefore not recommended. (End of Aside.)

In the following we assume a relatively modest syntax for names introduced by the author, say not much more complicated than a sequence of letters and digits starting with a letter. We do so because in our experience little is gained by the introduction of two-dimensional names in fancy scripts, e.g. \mathcal{T}_a^H . But even with such a strict syntax our freedom in choosing is almost unlimited and it is the writer's task to exploit that freedom wisely.

We take the position that formulae's main purpose is to be manipulated and that, therefore, our choice of names should be guided by the ease of manipulation. The first consequence is that a short name is better than a long one. Trivial and obvious as the consequence may be, in this day and age it needs to be stated explicitly as it runs counter to the propaganda with

which higher-level programming languages have been sold to the programming community, the idea being that the choice of sufficiently descriptive names would make programs "self-explanatory". This, however, was a fallacy: many a programming error can be traced down to the use of identifiers that were deemed to be sufficiently descriptive. In the case of a totally meaningless identifier it is obvious both to the writer and to the reader that all relevant properties must be stated explicitly.

A second consequence is that one should choose one's dummies carefully. Particularly in the case of a closely knit theory it can pay: in the case of substitution it is preferable to have the variables substituted for named differently from what is substituted, a difference which removes a possible source of confusion and error; often substitution can be avoided altogether by choosing the same names, e.g. naming an arbitrary solution of an equation by the same identifier as has been used for the unknown of the equation. Note that both difference and equality of names can be exploited.

Thirdly, we often introduce names in groups with some internal structure, e.g. in pairs, in cycles, in a hierarchy, or reflecting a combinatorial state of affairs (e.g. vertices and edges of a triangle). We strongly suggest to choose names that reflect such structure as fully as possible. We don't need to elaborate on the available techniques such as primes, correspondence between small and capital letters, alphabetic order, alphabetic nearness, multi-character names, but we do urge the writer to pay attention. (Two recent examples. Re-

cently we saw in an argument the identifiers Xs , Ys , f_x , f_y replaced by P , Q , p , and q , and the "simplification" was striking. The new terminology significantly reduced the demands on the hand-eye coordination required for the argument's development; it also made the argument much more pleasant to read. The other recent example was a theorem about sets A , B , C , and D . Formulation and proof strongly suggested that the number four was essential, though the theorem was a special case of any number of cyclically arranged sets. In the author's formulation, the cyclic order was $ABDC$. We would like to suggest that the unfortunate terminology -- and further notation, to be quite honest -- could have played a rôle in the author's failure to discover the generalisation.)

Besides the question how to name there is the question what to name, and here lies a second reason why we urge the author to choose his names as carefully as possible. The choice usually being difficult, the author thus increases the probability of discovering that the name is not needed after all. Two things should be avoided: superfluous names and names for the wrong objects. Not claiming an exhaustive treatment of the topic, we confine ourselves to the most frequent symptoms of inadequate naming. An arbitrary identifier used only once is superfluous, e.g. "Every integer N greater than 1 can be factored into a product of primes in only one way." -- Courant and Robbins in "What Is Mathematics?" --. Though a secretary could weed it out, this sin is remarkably frequent. (In passing we note that "greater than 0" would have been preferable, the empty product of primes

being unique as well.) Superfluity can also be noticed in a statement like "In triangle ABC the bisectors of the angles A, B, and C respectively are concurrent." instead of "In a triangle the angular bisectors are concurrent." Mild symptoms are displayed when the text gets enumerative, with lots of "respectively". A more serious symptom is usually the occurrence of "without loss of generality we can choose" or a similar clause, for then the chances are high that an underlying symmetry has been destroyed by an overspecific nomenclature. We called the latter symptom "more serious" because we have more in mind than just a cosmetic reformulation of the arguments: it is often indicative of the possibility of designing an essentially superior argument -- eg. replacing a combinatorial case analysis by a counting argument--.

So much for avoidable names. We now turn to the question whether the right things are named. If not, the general symptom is clumsiness of the development. We shall mention some common examples first.

- Defining the natural numbers as the integers from 1 onwards (with in its wake the failure to recognize 1 as the empty product, etc.). Avoidable case analyses and clumsy formulation are the standard by-products.
- The superfluous argument, or naming a function instead of an expression. A function's characteristic property is that it can be applied to different arguments, and if that flexibility is not needed and a function name, Π say, always occurs in the combination " $\Pi(\sigma)$ ", it could be argued that a name for that combination, P say, would have been more appropriate.

The phenomenon occurs often, and if we count subscription, etc., as notation for function application it occurs very often. Sometimes it is defended on the ground that in a few isolated instances the function is applied to other arguments, but a notation for substitution could cater for these instances, eg. P_{σ}^{σ} , $P(\sigma/\sigma)$, or $(\lambda\sigma.P)(\sigma')$ for $\bar{\Pi}(\sigma')$ -- we have also seen $\sigma := \sigma'; P$ with the rule that the low-priority semi-colon is right-associative -- . Sometimes the need to express universal quantification (over σ) is felt as a compelling reason for adopting the convention. People feel comfortable with $(\forall\sigma :: \bar{\Pi}(\sigma))$ but less so with $(\forall\sigma :: P)$, understandably wondering whether the dummy σ is arbitrary or not. In contexts where universal quantification over σ is the order of the day, great economy of expression can be achieved by denoting universal quantification over σ by a reserved bracket pair, eg. $[P]$ instead of $(\forall\sigma :: \bar{\Pi}(\sigma))$.

- Naming a subset versus naming the characteristic predicate. Given the predicate `prime` on natural numbers, we can define the set `PRIME` by

$$\text{PRIME} = \{n \mid \text{prime}.n\} \quad ;$$

conversely, given the set `PRIME`, we can define on the natural numbers the predicate `prime` by

$$(0) \quad \text{prime}.n \equiv n \in \text{PRIME} \quad .$$

In view of this one-to-one correspondence, we need only one of the two. We have observed that the majority of mathematical texts opt for `PRIME`. In view of the similarity of the left- and the right-hand sides of (0) the choice seems almost irrelevant. But it isn't, because the logical connectives

have been developed better than the set-theoretical operators. The latter \cup , \cap , \div , and \setminus -- suffer from the constraint that each element of the result is an element of at least one of the operands or \subseteq and $=$ -- suffer from the anomaly that the result is a boolean value. By way of illustration, we recently saw a proof that for some predicates a , b , c , and d the predicate

$$a \equiv b \equiv c \equiv d$$

was everywhere true; the author, however, had used sets and had formulated his demonstrandum as

$$A \div B = C \div D .$$

In short, by his decision to name the sets instead of the predicates, the author had obscured most of the symmetry (with the result that he failed to discover straightforward generalizations of his theorem). Also, do we like to need to know

$$A \cap (B \setminus C) = (A \cap B) \setminus C ,$$

whereas

$$(A \setminus B) \cap C = A \setminus (B \cap C)$$

is invalid? In general, we found the application of the logical connectives to the predicates much more convenient.

Note. The preponderance of named sets over named predicates is the result of an educational bias, which is also regrettable for other reasons: Venn-diagrams do invite case analysis. (End of Note.)

- In the derivation of an algorithm computing the

coordinates (X, Y) of a pixel on the screen, the subexpressions $(X - X_0)$ and $(Y - Y_0)$ occurred all over the place; calling the pixel coordinates $(X_0 + x, Y_0 + y)$ would have circumvented this. This was a very simple example; anyone with experience in analytical geometry knows how much difference the choice of a coordinate system can make. Repetition of lengthy or similar subexpressions is the usual indication of an unfortunate choice.

The gain can be more radical. The algebraic identity

$$(a-b) \cdot (b-c) + (b-h) \cdot (c-a) + (c-h) \cdot (a-b) = 0,$$

which for reasons of symmetry is very easy to verify, is the shortest proof (analytical or not) we know of the fact that the altitudes of a triangle are concurrent: if two of the addenda equal 0, so does the third! The above proof, however, presupposes the conceptual availability of vectors, their differences, and their scalar products plus the rules of the corresponding calculus.

Inadequate naming tends to manifest itself pretty clearly. We are the first to admit that in general it is less clear how to improve the situation: as the above example shows, the concept to be named might need to be invented. We do hold the belief, however, that in the process of such invention considerations of notational convenience can be of considerable heuristic value. (We do not know how Cayley and Sylvester invented the matrix calculus, but we can think of a scenario that starts with them getting tired of writing over and over again

$$\begin{array}{rcl}
 A_{11} \cdot x_1 + A_{12} \cdot x_2 + \dots + A_{1n} \cdot x_n & = & y_1 \\
 A_{21} \cdot x_1 + A_{22} \cdot x_2 + \dots + A_{2n} \cdot x_n & = & y_2 \\
 & \vdots & \\
 A_{n1} \cdot x_1 + A_{n2} \cdot x_2 + \dots + A_{nn} \cdot x_n & = & y_n
 \end{array}$$

Sylvester losing his patience and Cayley tactfully suggesting "What about $A \cdot x = y$?" .)

(End of •'s.)

* * *

A final remark, be it on the borderline of naming and notation. We all know how, early in the last century, British mathematics was nearly killed by its adherence to Newton's fluxions and how a coup of the Analytical Society in Cambridge was needed for the conversion to Leibniz's method. In retrospect it is hard to see why several generations of mathematicians could stick so tenaciously to the Newtonian version of the calculus. British insularity has been offered as an explanation, so has an ill-directed patriotism, but these are merely sociological explanations. The whole affair must have had a technical side as well. It has taken so long that we must conclude that at least from today's vantage point there was only a dim awareness of what a well-designed formalism can do for us. Letting the symbols do the work while manipulating uninterpreted formulae represents a conception of doing mathematics that must have been almost entirely foreign to those generations: had ease of manipulation been a recognized issue, the tragedy would probably have been

prevented.

Austin, 13 May 1986

Prof. dr. Edsger W. Dijkstra
Dept. of Computer Sciences
The University of Texas at Austin
Austin, TX 78712 - 1188
United States of America

A.J.M. van Gasteren
BP Venture Research Fellow
Dept. of Mathematics and Computing Science
Eindhoven University of Technology
Postbox 513
5600 MB EINDHOVEN
The Netherlands