In a May 1993 *Word Ways* article, Donald Knuth says that the reason his computer found so many 5x5x5 word cubes is that so many five-letter words can be "tweaked" into others. Each tweak or combination of tweaks can produce a new cube, only slightly different from the old one. All of us who have been finding word squares understand. The large numbers reported by Chris Long in the February 1993 *Word Ways* reflects this. To eliminate the effect of tweaking along the major diagonal (where a tweak always produces a new square), I devised the following procedure. I have been thinking about this for some time in regard to speeding the search for word squares; it can be applied to cubes equally well.

To start, my computer used 4839 common five-letter words to find 178,415 squares in 5.0 minutes. I then used a Carrollian ladder program to create a Boolean catalog. A number, BOO(x), was assigned to each word typically as follows. I illustrate with a subject word, CARES. BARES joins CARES in position 1; since B comes before C in the alphabet, mark BOO(CARES) in position 1. CORES joins CARES in position 2, but since 0 does not come before A, do not mark BOO(CARES). There are several words that mark BOO(CARES) in position 3. CARDS marks position 4, and CARED marks position 5. CARES ends up with a BOO of 10111 (decimal 23), CORES and CURES are similarly scored 31. BARED gets scored 4, being marked only by BAKED, which is scored 0.

In the second round, the computer ran all 4,839 words through a program which when looking for the nth word said if BOO(x) has a 1 in the nth position, do not use word x here. This program found 40,888 squares in only 2.3 minutes. Here are six squares sampled from forty (every thousandth square was printed out).

<table>
<thead>
<tr>
<th>CAMEL D RIP S K V ASS M U F T I  S O A</th>
<th>U N I O N I D E A L</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEA S E S O L ON S T REP INK E D</td>
<td>S N E A K</td>
</tr>
<tr>
<td>EV I L S P I A N O S I E GE T A S T E P O L K A L A N C E</td>
<td>L E A S E S O L ON S T REP I N K E D</td>
</tr>
<tr>
<td>LEA S E S O L ON S T REP I N K E D</td>
<td>S N E A K</td>
</tr>
<tr>
<td>LEA S E S O L ON S T REP I N K E D</td>
<td>S N E A K</td>
</tr>
</tbody>
</table>

And then, for the sake of variety, I required the program to use five new words for each new square (in addition to the BOO test). I only got 225 squares this way; here are the last ones found.
For readers who are not familiar with computer logic, the word Boolean refers to the fact that my programs use Boolean AND/OR logic to mark and test the words.

Having catalogued the five-letter words, I decided to see what happens with cubes. Before proceeding, I reduced my list to 4712 words by eliminating a few that didn’t seem to be common enough. It was not difficult to expand a computer program for squares into one for cubes; the resultant worked fine. Using the BOO logic described above, it found 243 cubes in a little more than three hours. Without the BOO, it might have found a thousand or so, in a much longer time.

Before presenting a few findings, let me describe the architecture of a cube. A 5x5x5 cube contains fifteen different words. When we describe the cube in the following manner, ten of the fifteen words are printed twice (not used twice). Letters along the main diagonal are in boldface. These are the ones to which the BOO logic was applied.

Here are some cubes (asterisks indicate tweakable letters):

* * * *
* * * *
* * * *

abbot banch base octet theta eagle aglow clove hower slope taper owns owns tense arises
balsa amnhis lapsis shite asset morn arena honor shone pedal inne sales tonic erect tests
cable ables blast leave easter blast unto enter stone erion ate one tined venam every redye

dodge dobes does geese esses bruin autdo elder short steel edena solar sems crave steep
ennui neers naval urate isled eclat alive maven stone stem avert into terse enter doors

dide isles clet heave ester svant unto enter stone eridow atomic towel venam essen relax
genade cubes about device erster usamp burno error sone orbit urine totes conic erectrests
heaps eclat alone panda stand drive lims mail tesla orbit naive ester diver alert darts
japan alamo povid averse nodes lilac alone mania occur vowel inert delta digre enter snare
kaput amass posha usher toser tars milt altar scale sword stave haven arena elect rents spanm
legen evate sales ideal tesla vigas agent dance ester leeve erema stemh asmes lease named

These cubes use commoner words than those of Donald Knuth, and they don’t contain his flaw of using a word twice. Based on simple probability arguments, we do not expect a word to be chosen twice but the nature of the system makes it happen often enough. The computer program used here contains specific logic to prevent duplications. This is usually not necessary for squares.