

# WORDS



A Quarterly Bulletin for Technical Writers & Communicators

Volume 3 | Issue 2 | May 2011

## The beginning

- Numbers have fascinated *Homo sapiens* for millennia. They have bewitched even the most brilliant minds—Pythagoras, for example—into thinking that numbers possess mystical or esoteric powers. And numerology is still alive, seemingly as ineradicable as its stable-mate: astrology. One number that draws more than its fair share of fascination is 7. We have the 7 wonders of the ancient world, secret agent 007, the 7 deadly sins, 7 days in a week, and many more. The number 7 has even found its way into technical writing—by way of an instructional philosophy known as *information mapping*. According to this philosophy, technical writers should always present information to readers in 7 ( $\pm 2$ ) chunks. Many who have been taught this philosophy still swear by it. Continuing our earlier myth-breaking theme, this issue considers the science behind information mapping—and finds it somewhat lacking.
- Conditional tagging has long been a feature of Adobe FrameMaker, the feature that attracted many early exponents of single-sourcing. You can tag a block of text as applicable to one audience or output, and tag another block as applicable to another audience or output. Thus in the one document you can have material applicable to numerous mutually exclusive audiences or outputs. Generating a particular deliverable just means hiding material tagged in certain ways, a process that can yield an endless range of discrete deliverables. Microsoft Word has never offered a similar feature. But, with a little ingenuity, you can do much the same thing. You will need to get your hands dirty with VBA code, but, as Gary Calwell explains in this issue, this is not all that difficult. Indeed, you can use Gary's code as a base for your own single-sourcing projects.
- As software gets more and more expensive, and as the accompanying documentation gets sparser and sparser, many have stepped off the upgrade treadmill. A new philosophy has arisen: what has served us well in the past will do. But there is an alternative: open-source software. In this issue, Lana Brindley discusses its pros and cons.

**Geoffrey Marnell**

Editor [geoffrey@abelard.com.au]

## Contents

Conditional text in Microsoft Word .....	1
The 7 $\pm$ 2 limit: Science or bunk?.....	6
The grass is greener on the open side .....	16
Journals.....	18
Book reviews.....	19
Tips and tricks.....	23
Miscellany .....	24
Mindstretchers .....	25

## Conditional text in MS Word

Now a single document can be used for multiple deliverables

**Gary Calwell**

Have you heard about the new feature in Word that allows you to put tags on text and drawing objects and then turn the tagged content on and off? No? Unfortunately, neither have I.

Although Microsoft doesn't provide a standard conditional text feature, in this article you'll learn how to build your own customised solution.

There are many situations where similar content needs to be provided for different audience groups—each with a different “flavour”. You may need to create an instructor guide and a participant guide for a training course, or perhaps a set of instruction manuals, each one tailored for a different role.

In these situations you could create a duplicate document and then edit the new version. For example, you could write a software manual for call centre operators, save this as a second document and then edit it so that it's appropriate for team leaders.

The problem with this approach is that the duplicated content creates a huge overhead for later maintenance. If you create five versions of your software manual (each for a different role), and some common information changes, you then need to update this in five places—not one.

Some authoring tools, such as Adobe FrameMaker allow you to mark content with a tag. You can then turn different combinations of tagged content on or off to produce different versions of your final output.

Even though there is no out-of-the-box Word feature to do this, fortunately Microsoft provides enough customisation options for you to implement your own single-source solution.

In this article you'll learn:

- the design concepts required to build a conditional-tagging feature
- how to write your very first VBA macro
- how to write and use the conditional-text macros
- implementation tips for real-world designs

## Design concepts

The simplest Word single-source solution can be created using Word's **Hidden Text** feature. Let's suppose you need to create a training participant guide that asks a question and provides the reader with a line to write the answer. You'd also like to create an instructor version that provides the same content but also includes the answer.

The instructor document could look like this:

What is the meaning of life?  
42\_\_\_\_\_


By formatting the answer ("42") as hidden text you can control, using the standard Word options, whether this is displayed on screen or printed. To create the participant version, simply set Word to hide hidden text; to create the instructor version, set Word to display hidden text.

This approach, whilst very useful, could hardly be classified as conditional tagging. To extend the capability further, we need to use Visual Basic for Applications (VBA), a programming environment that is built into Word. We'll start by showing you how to write your very first VBA macro. If you already know how to write VBA macros, skip the next section.

## Your very first VBA macro

These instructions were developed using Word 2007. The macros will work with Word 2010 and earlier versions although the menu commands may be different.

1. Open Word, create a new document and then save it as `SingleSource.docm`. (You will need to choose **Word Macro-Enabled Document (\*.docm)** as the **File Type**. This is necessary if you are to run macros in it.)

2. Ensure your macro security setting will allow macros to run. Some administrators may block this access.  
Click  and select **Word Options > Trust Center > Trust Center Settings > Macro Settings > Disable all macros with notification**. Finally, click **OK** twice to return to your document.
3. Press **ALT + F11** to display the **Microsoft Visual Basic** window.
4. In the top left-hand pane, click on **Project (SingleSource)**. If the pane is not displayed, press **CTRL + R**.
5. Select **Insert > Module**.

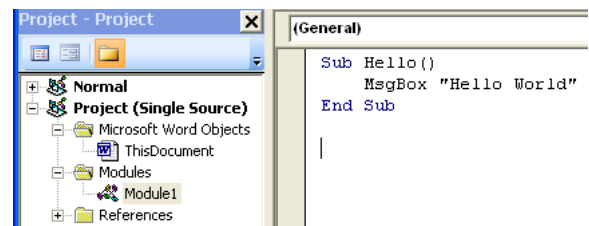
6. Click on **Module1** underneath the **Modules** folder.


There should be a blank window in the right-hand pane. This is where you write the code for your macro.

7. In the code window, type the following:

```
Sub Hello()  
    MsgBox "Hello World"  
End Sub
```

Your screen should now look like this:



8. Press **CTRL + S** to save the file.
9. Click on  in the top right-hand corner to close the **Microsoft Visual Basic** window.  
You are ready to run your new macro.
10. Press **ALT + F8** to display the **Macros** window.
11. Double-click **Hello** in the **Macro name** list. You should see the following message.



If this is the first macro you have written, congratulations! If you need help, search for "Record or run a macro" in Word Help.

12. Click **OK** to close the message.

## Conditional-text macros

This section lists some macros you can paste into your code window to create your conditional tagging feature. For this example, we'll assume that you need to write a series of maths tests for junior and senior students. Instead of authoring in separate documents, you prefer to write all the questions and answers in a single document and then produce different versions to give to the students and teachers.

In your `SingleSource.docm` Word document, type the following five paragraphs:

### Maths Test

What is five times three?

15

What is the square root of 2,304?

48

The first question is for a junior test and the second is for a senior test. Later we want to be able to produce different versions of the test so we will tag each line as follows:

- Line 1: No tag. The heading will be used in all documents.
  - Line 2: Tagged as a Junior Question (JQ)
  - Line 3: Tagged as a Junior Answer (JA)
  - Line 4: Tagged as a Senior Question (SQ)
  - Line 5: Tagged as a Senior Answer (SA)
1. Press **ALT + F11** to display the **Microsoft Visual Basic** window and locate your **Hello** macro.
  2. Leave a blank line underneath the **End Sub** line and paste in the following code. Take care. VBA is not very forgiving of typos.



The code is also available in `SingleSource.txt`, attached to this PDF. You can copy-and-paste the code from that file.

```
Sub AddTagToParagraphs()  
    Dim newTag As String, oldTag As String  
    Dim myPara As Paragraph  
    oldTag = TagPara(Selection.Paragraphs(1), "")  
    newTag = UCase(InputBox("Type tag for selected paragraph(s).", "Single Source", oldTag))  
    For Each myPara In Selection.Paragraphs  
        oldTag = TagPara(myPara, newTag)  
    Next myPara  
    Set myPara = Nothing  
End Sub
```


```
Sub ProcessTags()  
    Dim TagsToShow As String  
    Dim myPara As Paragraph  
    Dim myTag As String  
    Dim i As Integer  
    Dim hidePara As Boolean  
    TagsToShow = UCase(InputBox("Type tags to make visible. Leave blank to show all", "Single Source"))  
    ActiveWindow.View.ShowHiddenText = True  
    ActiveDocument.Range.Font.Hidden = False  
    For Each myPara In ActiveDocument.Paragraphs  
        myTag = TagPara(myPara, "")  
        If myTag <> "" Then  
            hidePara = False  
            For i = 1 To Len(TagsToShow)  
                If InStr(myTag, Mid(TagsToShow, i, 1)) = 0 Then hidePara = True  
            Next i  
            If hidePara Then  
                myPara.Range.Font.Hidden = True  
            End If  
        End If  
    Next myPara  
    ActiveWindow.View.ShowHiddenText = False  
    ActiveWindow.ActivePane.View.ShowAll = False  
    Set myPara = Nothing  
End Sub
```

```
Function TagPara(myPara As Paragraph, myTag As String) As String  
    Dim pos1 As Integer  
    Dim pos2 As Integer  
    Dim myText As String  
    Dim myRange As Range  
    Set myRange = myPara.Range  
    myText = myRange.Text  
    pos1 = InStr(myText, "{")  
    pos2 = InStr(myText, "}")  
    If pos1 > 0 And pos2 > pos1 Then  
        myRange.SetRange Start:=myPara.Range.Start + pos1 - 1, End:=myPara.Range.Start + pos2  
    Else  
        myRange.MoveEnd unit:=wdCharacter, Count:=-1  
    End If  
End Function
```

```

        myRange.Collapse
direction:=wdCollapseEnd
    End If
    If myTag <> "" Then myRange.Text =
    "{" & myTag & "}"
    myText = myRange.Text
    If Len(myText) > 1 Then TagPara =
    Mid(myText, 2, Len(myText) - 2) Else
    TagPara = ""
    myRange.Font.Hidden = True
    Set myRange = Nothing
End Function

```

3. Press CTRL + S to save the file.
4. Click on  in the top right-hand corner to close the **Microsoft Visual Basic** window.
5. Press ALT + F8 to display the **Macros** window.  
You now should now have two new macros listed:

- **AddTagToParagraphs:** This is used to mark one or more paragraphs with a tag (such as "SQ")
- **ProcessTags:** This is used to hide or display tagged content based on criteria you specify.

## Tag the Content

You now need to tag the different content in the document.

1. Position your cursor somewhere in the second line ("What is five times three?").
2. Press ALT + F8 and run the **AddTagsToParagraphs** macro.
3. Enter JQ and click **OK**. This indicates that the second line is a Junior Question.



4. In the same way, add the tag **JA**, **SQ**, and **SA** to the third, fourth and fifth lines respectively.
5. To see what the macro has done, set Word to display hidden text:



> **Word Options** > **Display** > **Hidden Text**

Your document should now look like this:

### Maths Test

What is five times three?{JQ}

15{JA}

What is the square root of 2,304?{SQ}

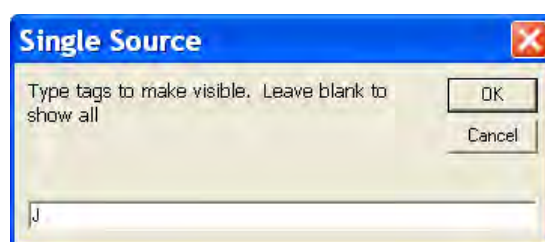
48{SA}

Note that the macro has added the tags between braces at the end of each paragraph. The macro ensures that these tags are always hidden so you never need to see them (unless of course you want to, which you can by manually displaying hidden text).

## Process the Tags

Now that different content has been tagged, you can use the **ProcessTags** macro to display different combinations.

1. Press ALT + F8 and run the macro **ProcessTags**.
2. Enter J in the box and click **OK**.



Your document should now just display the Junior Question and Junior Answer:

### Maths Test

What is five times three?

15

3. Experiment with different conditional criteria. As you can see in the following table, our small five-line document can now produce ten different outputs depending on our needs.

Criterion	To display
(leave empty)	All text
J	Junior (questions and answers)
S	Senior (questions and answers)
Q	Questions (junior and senior)
A	Answers (junior and senior)
JQ	Junior Questions
JA	Junior Answers
SQ	Senior Questions
SA	Senior Answers
X	Non-tagged text (in our example, just the "Maths Test" heading)

## Implementation tips

These macros have been designed to show you the concepts of a single-source strategy using conditional tagging. Solutions based on these concepts have been used in professional authoring environments and have worked reliably for both small and large documentation teams. However, most implementations will benefit from error checking

and certain other features. Here are some implementation tips to consider.

## Error checking

Robust single-source macros should cater for all the situations likely to be encountered in your documentation. However, the macros demonstrated above will produce an error if you attempt to tag an entire table. Moreover, if your document contains braces, any text between them will be lost. A robust macro would work around these conditions.

## Tagging units

In our example, tags were added to individual paragraphs. There are many other elements that you can use as the basis for a tagging strategy. For example, you could conditionally display:

- everything in a tagged table, row or column
- everything underneath a selected heading (up to the next peer or higher level heading)
- specified combinations of styles
- specified combinations of font colour

## Building blocks (AutoText)

Building blocks are document fragments that you can store for later reuse. Although useful for many reasons, building blocks are particularly helpful when used with conditional tagging. For example, a



**CLARICOM**

**Make your Office Applications Sing!**

Frustrated by repetitive tasks in Office?  
Claricom customisations allow:

- ▶ One-click access to repetitive authoring tasks
- ▶ Automatically transferring information from one application to another
- ▶ Linking Office applications with databases
- ▶ Automated formatting and layout conversions
- ▶ Many more improvements (send us your ideas).

Web: [www.claricom.com.au](http://www.claricom.com.au)  
Email: [gary.calwell@claricom.com.au](mailto:gary.calwell@claricom.com.au)

building block for our maths test document could look like this:

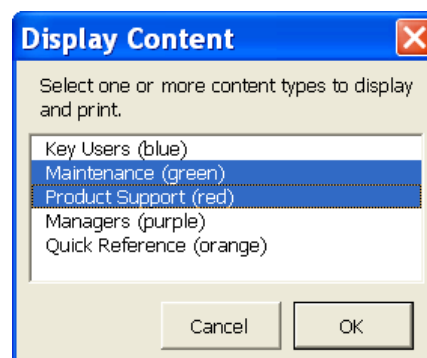
Q1: question

A1: answer

If you tag the first line as "JQ" and the second line as "JA" *before* you create the building block, you won't have to tag this content after insertion. Using this technique you can create a library of tagged building blocks.

## User interface

In the maths test example you ran the macros by pressing ALT + F8 to display the **Macros** window. To make it easier, quicker, and more intuitive for authors to use your new tagging feature you can customise the Word ribbon and dialog boxes as shown in the following examples.



## Drawing layer

You cannot use Hidden Text to control the visibility of floating objects on the drawing layer, such as shapes, pictures or text boxes.

Instead, use VBA to change the Visible property of these objects. To see a demonstration of this:

1. Draw four text boxes in your document and format two of them with a red border. The colour must be pure red (RGB = 255,0,0).
2. Press ALT + F11 to display the **Microsoft Visual Basic** window, paste in the following code, save the file and close the window.

```
Sub RedBoxes()  
    Dim myShape As Shape  
    Dim displayRed As Boolean  
    If MsgBox("Display red boxes?",  
vbYesNo) = vbYes Then displayRed = True
```



```

For Each myShape In
ActiveDocument.Shapes

    If myShape.Line.ForeColor.RGB =
RGB(255, 0, 0) Then myShape.Visible =
displayRed

Next myShape
End Sub

```

3. Press ALT + F8 to and run the **RedBoxes** macro. The following message appears.



4. Click **No** to hide the red boxes. Run the macro again and click on **Yes** to display them again.

\* \* \*

Although Microsoft Word doesn't have a native conditional tagging feature, in this article you've seen how to build your own. Developing a robust, stable

solution does take some time and effort, but there are two key advantages in doing so.

The first advantage is that your solution can be tailored exactly the way you want it. You can choose different tagging units and develop custom menus to hide the complexity from users. In doing so, you can make it very quick for authors to apply consistent tags and then easily display the tagged content.

The second advantage is that as you build your solution, you might start to see the potential to customise Word in many other ways using VBA. If you are like me and hate doing repetitive tasks in Word, VBA can offer many ways to save time whilst also improving the quality and consistency of your output.

Good luck in shaping Word to be just the way you want it—and as Luke Skywalker (almost) said: “may the single source force be with you”.

### **Gary Calwell**

Gary Calwell is the director of Claricom Consulting, a small Melbourne-based consultancy specialising in process improvement, documentation, training and Microsoft Office customisation. He can be contacted at [gary.calwell@claricom.com.au](mailto:gary.calwell@claricom.com.au).

## **The 7 ± 2 limit**

### **Science or bunk?**

#### **Geoffrey Marnell**

There is a view—especially prevalent in the technical writing profession—that information should only be presented to readers in small quanta or chunks. On the face of it, this is a sensible view. A sentence composed of three or more clauses usually breaches the limit of cognitive lodgment. It is just too long for most readers to absorb. Only by atomising a conglomerate of ideas into discrete sentences—that is, by chunking our ideas—do we minimise the effort our readers need to exert to understand what we have written.

We also chunk our writing into paragraphs. A paragraph enables a writer to present an argument, discussion, analysis or whatever in logical and accessible units:

“There are two purposes for paragraphs, the one logical and the other practical. The logical purpose is to signal stages in a narrative or argument. The practical one is to relieve the forbidding gloom of a solid page of text.”<sup>1</sup>

Paragraphing is thus a form of chunking. Without it, readers would be presented with a form of *scriptio continuo*: undemarcated logic rather than

undemarcated words. Such writing would hardly be inviting, nor likely to keep readers engaged. They would have to work out for themselves where each new topic began, causing them to drag anchor as they read. So chunking has been part of writing for a very long time. We chunk ideas into sentences, groups of related ideas into paragraphs, groups of related paragraphs into sections, and so on.

Traditionally, writers could choose the length of a sentence, a paragraph and a section. The amount of material they had about a particular topic, and the form of reasoning being adopted, might place limits on how long a paragraph or section might be. But these were limits imposed by logic and they varied from paragraph to paragraph. No one—or at least no one before the late 1960s—considered that there had to be an upper limit to the size of a paragraph or section no matter how much information the writer had to deliver. We might have deliberately avoided presenting readers with solid pages of text, but we did so without regard to any particular limit on the size of our chunks.

Today there are some who believe that we should limit our chunks to 7 ± 2 units of information, whether we are writing instructional materials, designing billboards or programming computer code. The source of this belief is usually attributed to an analytic literature review published by the American psychologist George Miller in 1956, the

1. N Hudson, *Modern Australian Usage*, OUP, Melbourne, 1993, p. 294

title of which is “The magical number seven, plus or minus two: Some limits on our capacity for processing information”.<sup>2</sup> Many writers were introduced to the notion of a chunking limit through training in what was once a popular writing method: *Information Mapping*. The method is still taught and thus the so-called magical span of  $7 \pm 2$  still holds a spell over many writers. But is this science or flim-flam? Let’s consider the Information Mapping method first before looking at Miller’s often-quoted paper, the paper said to be the foundation on which the method was built.

## Information Mapping

The Information Mapping method was developed by the American political scientist Robert Horn. According to Horn:

“Writers should group information into small, manageable units ... A ‘manageable unit’ of information is one consisting of no more than nine pieces of information ... Rationale: Research suggests that people can best process and remember no more than seven plus or minus two pieces, or units, of information at one time ... Therefore, a general guideline for a ‘manageable unit of information’ is one consisting of  $7 \pm 2$  pieces (also referred to as the chunking limit) ... Writers should create units of information that do not exceed the chunking limit. We should apply this limit at every level of a written document ... By chunking information the writer improves the reader’s comprehension and access and retrieval speed. Since readers can at best retain no more than 5 to 9 pieces of information in short-term memory, they comprehend material that has been ‘chunked’ more quickly and more completely.”<sup>3</sup>

“Therefore, a general guideline for a ‘manageable unit of information’ is one consisting of  $7 \pm 2$  pieces (also referred to as the chunking limit).”

Furthermore:

“... apply the Chunking Principle to:

- sentences
- blocks [that is, a group of  $7 \pm 2$  sentences about a common topic, or a list or a table]
- maps [that is, a group of  $7 \pm 2$  blocks]
- sections [that is, a group of  $7 \pm 2$  maps] and
- chapters [that is, a group of  $7 \pm 2$  sections].”<sup>4</sup>

Also:

“Remember that the Chunking Principle advises  $7 \pm 2$  items in a list.”<sup>5</sup>

2. GA Miller, “The magical number seven, plus or minus two: Some limits on our capacity for processing information”, *The Psychological Review*, 1956, vol. 63, no. 2
3. RE Horn, *Developing Procedures, Policies & Documentation*, Info-Map, Waltham, 1992, p. 3-A-2
4. *ibid.* As we’ll see, Horn did not apply his chunking limit to sentences, despite what he says in this quote. See page 8.

And:

“When your sentence is more than twenty words long, consider dividing it.”<sup>6</sup>

“[A] sentence [should] never be more than 30 words.”<sup>7</sup>

The first question to ask is this: was Horn true to his own method in writing *Developing Procedures, Policies & Documentation*, the book from which these quotes are taken? It would seem not. The work includes numerous sentences of more than 30 words, has lists of more than nine items, and three chapters have more than nine sections or maps. But it would be churlish to put too much emphasis on such inconsistencies. The fact that the author does not follow his own method does not necessarily make the method flawed. To argue so would be to fall foul of an *argumentum ad hominem* fallacy.

## What is a piece of information?

In Information Mapping, writing should be grouped into units of information—blocks, maps, sections, etc.—and a “manageable unit of information is one consisting of no more than nine pieces of information”.<sup>8</sup> But what is a *piece* of information?

Imagine a map with seven blocks and with seven sentences in each block. This would fall within Horn’s  $7 \pm 2$  limit. Such a map would have  $7^2$  (or 49) sentences. So if a unit of information—in this case a map—cannot have more than nine pieces of information in it, then obviously a sentence is not what Horn means by a *piece* of information.

Now if a map can have no more than nine *blocks* and no more than nine *pieces of information*, it would seem that a block and a piece of information are considered the same thing. However, a block is also a “unit of information”<sup>9</sup>, so if a unit of information is composed of pieces of information, a block cannot *exclusively* be a piece of information. Otherwise a block would always have just one element in it: one piece of information.

So it seems that for Horn what constitutes a *piece* of information varies from one unit of information to another. At the level of chapter, a piece of information is a section; at the level of a section, a piece of information is a map; at the level of a map a piece of information is a block and at the level of a block a piece of information is a sentence. But what is a piece of information at the level of a sentence?

5. *ibid.*, p. 10-5
6. *ibid.*, p. 12-3
7. *ibid.*, p. 12-2
8. *ibid.*, pp. 3-B-2, 3-B-5 and 3-A-2
9. *ibid.*, 3-B-1

This is an important consideration. The sentence is the fundamental, indeed necessary, building block in every document. Horn accepts that this is so: “The first and basic *unit of information* is the sentence.”<sup>10</sup> But he also says that “Writers should group information into small, manageable units [and a] manageable unit of information is one consisting of no more than nine pieces of information”.<sup>11</sup> So we would expect Horn to tell us what a piece of information *in a sentence* might look like. If the sentence is the first and basic unit of information, then the entire Information Mapping edifice balances or topples on the answer to this very question. Alas, Horn does not provide an answer. And for all its apparent importance, *sentence* does not even get a mention in the index of Horn’s book. Neither does *piece of information* nor *information, piece of*. We have to work it out for ourselves.

A piece of information at the level of a sentence cannot be the same as a character or a word, for Horn allows up to 30 words per sentence.<sup>12</sup> A thirty-word sentence would then have at least thirty pieces of information in it, well above the specified limit of nine. Anyway, words like *the*, *a*, *an* and the like

couldn’t possibly be pieces of information. Perhaps no word on its own could be a piece of information.

After words, the next level of granularity in a sentence is a *phrase*. A phrase is a string of words that, although potentially meaningful when combined with other words, does not have a subject (a thing singled out for discussion) or a predicate (something said about whatever is singled out for discussion). Put another way, “a phrase is a group of words that act together as a unit within a sentence [but which] can’t stand on [its] own and make a sensible message”.<sup>13</sup> If a phrase has no subject nor can stand on its own to make a sensible message, then obviously it cannot be a piece of information.

The next level of granularity is the *clause* (that is, a string of words with a subject and a predicate). In fact, the clause is as far as we can go. After a clause, we have a sentence, and it would make no sense to say that the *fundamental* piece of information in a sentence is the sentence itself while at the same time allowing that a sentence, being a unit of information, can have up to *nine* pieces of information.

And now it should be clear why Horn did not extend his  $7 \pm 2$  limit to the pieces of information that make up a sentence. The prospect of reading a nine-clause sentence would repulse most readers. Even a five-clause sentence would be indigestible to many readers, even those familiar with the topic. (A sentence composed of five independent clauses is equivalent to *five separate sentences* glued together with conjunctions and punctuation.) And this, no doubt, is why Horn opted instead for a word limit, not a pieces-of-information limit, on sentences: “an average of 20 words and never more than 30”.<sup>14</sup>

The sentence is not the only unit of information that Horn excludes from his  $7 \pm 2$  limit. If a chunk as large as a chapter can be a unit of information, there seems no reason why a book or even a large report should not also be considered a unit of information. (It is just a collection of chapters as a chapter is a collection of sections.) But Horn nowhere limits the number of chapters in a book or report.

## Is there science behind Information Mapping?

Horn repeatedly states that his method is based on research. Here are just two of the many mentions:

“Research suggests that people can best process and remember no more than seven plus or minus two pieces, or units, of information at one time.”<sup>15</sup>

“The ... method implements research-based findings on how individuals process and understand information most efficiently.”<sup>16</sup>

### *Effective Onscreen Editing: new tools for an old profession*

Editors are increasingly being asked to edit on the screen using a word processor, but most are finding it challenging to transfer their skills to editing with a word processor. *Effective Onscreen Editing* teaches the basics you need to learn to make the transition, plus proven tips and tricks to maximise your productivity and effectiveness. The book describes general principles valid for any software, then illustrates the principles using Microsoft Word to make them more concrete.

Available as a printed book from Lulu.com and also available as an eBook optimised for onscreen reading.

Learn more at the book’s Web page:  
<http://www.geoff-hart.com/books/eoe/onscreen-book.htm>

 Diskeuasis  
Publishing

10. *ibid.*, p. 12-3. Emphasis added.

11. *ibid.*, p. 3-A-2

12. *ibid.*, p. 12-2

13. P Peters ed., *The Macquarie Student Writers Guide*, Jacaranda, Milton, 1989, p. 339.

14. RE Horn, *op. cit.*, p. 12-2

15. *ibid.*, p. 3-A-2



Despite being repeated a number of times in the book, no references to research to back up any particular claim are provided in footnotes or endnotes. There is a substantial “core bibliography” which, we are told, provides “complete citations for the research base on which the Information Mapping methodology was built”<sup>17</sup> and yet this bibliography includes many references of dubious relevance to Information Mapping. As one reads through it, a sense of quantity out-doing quality quickly intrudes. For example, there are references to generalist books such as *Statistical Methods* and *The Structure of Science*. How could such books be *specifically* relevant to Information Mapping? Including such general texts is about as silly as, say, Einstein, tendering a textbook on calculus in support of his general theory of relativity.

One also finds references in Horn’s bibliography that seem hardly related to scholarly research of any kind, such as Whitlock’s 1972 interview with Horn. Can a mere interview be classified as core research? There is also a reference to a 1945 article published in *Atlantic Monthly*, more a magazine than a research journal. Further, there are no immediately recognisable references to back up any specific Information Mapping principle, such as the chunking rule.

A fifteen-page bibliography might look impressive and suggest that much scientific rigor has gone into the recommendations in the book. But if the bibliography provides no help to the interested reader who wants to check how a recommendation is supported, it fails its very purpose. An author who makes a claim that is said to be based on research and yet provides no reference to that research is akin to the scientist who reports on some research but fails to explain to fellow scientists how the experiment was done and how it can be replicated. To expect the interested reader to read every item in a 15-page bibliography to verify that the accompanying text is soundly based on research is utterly unreasonable. Further, an “unpublished draft [of a] manuscript for [a] 1967 course at Harvard” along with three “unpublished proposal[s] to the] US Air Force” are listed in *The Information Mapping Method: 30 Years of Research*, a 1999 publication of Information Mapping, Inc., the company Horn established to promote the Information Mapping method.<sup>18</sup> Can unpublished course notes and commercial proposals really constitute research? Quantity over quality again?

## The overlooked Miller

The American psychologist George Miller is widely credited with pointing out the potential significance

of  $7 \pm 2$  as a limit to various aspects of human cognition. In research conducted in the early 1990s, Miller’s seminal paper — “The magical number seven, plus or minus two”, referred to earlier — shared with two others the honour of being the most cited papers in “24 well-known introductory psychology texts [that] fully covered the field of psychology”.<sup>19</sup> Horn tells us that he began developing Information Mapping in 1967, with the first version of his book on the subject published in 1976.<sup>20</sup> Given the excitement generated by Miller’s paper, it seems odd that Horn does not include the paper in his own bibliography. Let’s repeat that: Miller set the  $7 \pm 2$  debate in train, his paper is much discussed and cited (and still is), twenty years later Horn expounds on the importance of presenting writing in chunks of  $7 \pm 2$  and tells us that this is research-based — and yet neglects to mention Miller’s paper. That is nothing less than odd. Just as odd is the fact that when Miller’s and Horn’s bibliographies are compared, not a single reference can be found in one that is also in the other.

The failure to acknowledge Miller, and the sources Miller quotes, becomes even more curious when we read, on the Information Mapping website, that:

“The chunking limit is a guideline, based on George A. Miller’s 1956 research”.<sup>21</sup>

Can we assume that this is Horn’s view too? Well according to Horn’s own page on the Stanford University website, he has been chairman of Information Mapping, Inc. since 1987.<sup>22</sup> This is the company he established twenty years earlier to promote the Information Mapping method. The website of Information Mapping, Inc. is <http://www.infomap.com>, the very site from which the quote above was taken. Perhaps it is not stretching it too far to think that the chairman of a company would endorse the claims publicly made about the chief product of that company — especially when the product is the creation of the chairman. Thus it is not unreasonable to think that Horn did, and still does, think that George Miller’s research supports the Information Mapping method. But does it?

19. DW Gorenflo & JV McConnell, “The most frequently cited journal articles and authors in introductory psychology textbooks”, *Teaching of Psychology*, vol. 18., no. 1, 1991, p. 8f. Miller’s paper was cited in 22 of the 24 texts surveyed.

20. op. cit., p. i

21. See [http://www.infomap.com/index.cfm/themethod/Mapping\\_FAQs](http://www.infomap.com/index.cfm/themethod/Mapping_FAQs) (viewed 23 January 2011). Emphasis added. One might expect that a bibliography that includes a general textbook on statistics would include the very paper on which Information Mapping is based.

22. See <http://www.stanford.edu/~rhorn/a/site/HornCV.html>. Viewed on 13 January 2011.

16. *ibid.*, p. 2-3

17. *ibid.*, p. B-1

18. Available from <http://www.infomap.com>. Viewed 21 January 2011.

## Miller's magic number

For a start, let's make one thing clear. Miller did not do the research described in his justly famous 1956 paper about the limits of human cognition. He simply reported the results that other experimenters had published and tried to make sense of them (including the apparent fact that  $7 \pm 2$  kept cropping up in various, unrelated studies as a limit to cognition, something he suggested, in the final paragraph of his paper, might be nothing more than "a pernicious Pythagorean coincidence"<sup>23</sup>). So to say that Information Mapping is "based on George A. Miller's 1956 research" is a little misleading. If it is based on Miller's 1956 paper—as opposed to any of Miller's own research—then it must be based on the research of those who Miller quoted. But, as noted in the last section, not one of the researchers Miller quoted is mentioned in Horn's bibliography. Let's put this down to sloppy citing on Horn's part and assume that he meant to say that he based Information Mapping on the research that prompted Miller to think that  $7 \pm 2$  was somehow significant in cognitive psychology.

Miller's paper covers three distinct topics, all in some way related to memory. Since the Information Mapping website quoted above does not mention which topic or topics in Miller's paper form the basis of the Information Mapping method, let's look at each one.



commercial  
translation  
centre

For Effective  
Translation/Localisation  
Of Technical Documentation  
In 80 Languages

Australia Wide Service Since 1988  
Call 1800 655 224 (Australia Toll Free)  
Email: mail@ctc.com.au  
Web: www.ctc4.com

commercial translation centre

23. op. cit., p. 96

## The span of absolute judgment

Miller first considers a number of experiments in which subjects are asked to make absolute judgments about such stimuli as frequencies, loudness, saltiness, the size of rectangles, and the like. The objective was to determine how well humans can distinguish between differing levels of intensity of particular stimuli. All these experiments follow a similar pattern: subjects are exposed to a random sequence of varying stimuli after each stimuli had been given an identifying number. The subjects were then asked to repeat the numbers in the same order as the stimuli were randomly given. For example, a frequency of 100 Hz might be assigned the number 2, 6 000 Hz assigned the number 9, 8 000 Hz assigned the number 3, and so on. When exposed to tones at frequencies, say, of 6 000 Hz, 8 000 Hz and 100 Hz in that order, subjects were tested to see if they correctly responded 2, 9 and 3 respectively. The number of frequencies (or whatever stimuli is being used) is gradually increased until the number of correct responses drops to zero.

In summarising an experiment conducted by Irwin Pollack in 1952 using tones of varying frequency as stimuli, Miller writes:

"When only two or three tones were used the listeners never confused them. With four different tones confusions were quite rare, but with five or more tones confusions were frequent. With fourteen different tones listeners made many mistakes ... The result means that we cannot pick more than six different pitches that the listener will never confuse. Or, stated slightly differently, no matter how many alternative tones we ask him to judge, the best we can expect him to do is assign them to about six different categories."<sup>24</sup>

This is a misrepresentation of Pollack's result. Pollack concluded that subjects could assign tones to only *five* categories:

"... an informational transfer of approximately 2.3 bits is the maximum obtained. This is equivalent to perfect identification among only about 5 tones."<sup>25</sup>

Even so, the number of tones Miller quotes is 6, not 7—his magic number. Where, then, did Miller get his magic number? He got it by looking at a number of similar experiments—not just Pollack's—and

24. *ibid.*, p. 83–4

25. I Pollack, "The information of elementary auditory displays", *Journal of the Acoustical Society of America*, 1952, vol. 24., no. 6., p. 748. Pollack's 2.3 bits of information is equivalent to  $2^{2.3}$  correctly identifiable tones, which equates to 4.925. In general, the number of categories correctly judged (which Miller calls *channel capacity* and *span of absolute judgment*) is equal to  $2^b$ , where  $b$  is the number of bits of information presented by the stimuli as a whole. Thus the number of bits associated with a channel capacity of, say,  $c$  is  $\log_2 c$ . For our purpose, we are only interested in channel capacity, not bits of information.

taking the *average* span of immediate judgment (what he also calls the *channel capacity*, or the number of correctly identifiable *categories*, in Pollack's language):

"... the channel capacities measured ranged from 1.6 bits for curvature to 3.9 bits for positions in an interval. Although there is no question that the differences among the variables are real and meaningful, the more impressive fact to me is their considerable similarity. If I take the best estimates I can get of the channel capacities for all the stimulus variables I have mentioned, the *mean* is 2.6 bits and the standard deviation is only 0.6 bit. In terms of distinguishable alternatives, this mean corresponds to about 6.5 categories, one standard deviation includes from 4 to 10 categories, and the total range is from 3 to 15 categories."<sup>26</sup>

Note that the channel capacities across a number of like experiments ranged from 1.6 bits (or 3 categories, when judging the curvature of lines) to 3.9 bits (or 15 categories when judging positions along a linear interval). Miller took what he thought was the mean value—2.6 bits of information—equated that to 6.5 categories, and then rounded it up to 7.

On the face of it, this looks like sloppy arithmetic. If the mean is 2.6 bits of information, it is much closer to 6 categories than 6.5:  $2^{2.6} = 6.06$ . To get a channel capacity of 6.5, the average number of bits would have to be 2.7 (as  $\log_2 6.5 = 2.7$ ). To get Miller's magical number 7, the average number of bits would have to be 2.8 (that is,  $\log_2 7$ ). In Miller's defence, however, he was looking for a *limit* or *ceiling* on absolute judgment and this limit obviously has to be a whole number. In that case, rounding up rather than down makes sense.

Let's ignore Miller's arithmetic and concentrate instead on the wide range of channel capacities observed for various stimuli. Miller was surprised that the range wasn't greater, but if channel capacity—that is, the span of absolute judgment—is relevant to writing, there is a world of difference between advising writers to limit, say, the number sentences in a block to 3 rather than 15. If Horn relied on Miller's consideration of the observed range of channel capacities, then he needs to explain why judging meaning (or whatever) is more akin to judging the frequencies of tones (where the channel capacity is 5, given Pollack's research) than judging positions along a linear interval (where the channel capacity is 15). To opt without reason for 7 is hardly scientific, given that Miller did not report any experiments on the absolute span of judgment when written material was the stimuli. Indeed, for all that Miller tells us, that span could fall outside the observed 3–15 range.

Anyway, the relevance of any span of absolute judgment to effective written communication is

---

26. *ibid.*, p. 86. Emphasis added.

tenuous. Readers do not have to make the sort of comparative assessments of stimuli Miller considered in order to understand what they are reading. That is, understanding a piece of written material simply does not involve recalling the relative intensities of some semantic stimulus or other. There is, in other words, no parallel between naming tonal frequencies according to a provided legend and unravelling the meaning of a map, block or sentence. And, as we'll see a little later, Miller said that himself.

## The span of attention

The next part of Miller's paper—entitled *Subitizing*—has to do with "the discrimination of number".<sup>27</sup> The experiments that Miller reports have long precedents. In the nineteenth century, the Irish mathematician William Hamilton and the English economist Stanley Jevons independently showed that if subjects were briefly shown marbles (or stones) in a box, they could easily remember the number they had seen up to about seven, after which accurate recall fell away. Miller quotes more recent experiments that give the same result, in this case, dots flashed on a screen for 0.2 seconds at a time. An accurate count drops away dramatically after seven dots.

Could this be what Horn was basing Information Mapping on? It seems unlikely. No writer writes with the expectation that readers will only see what they have written for 0.2 seconds at a time (or two minutes, for that matter).

## Span of immediate memory

The last part of Miller's paper is taken up with what he calls the *span of immediate memory* (also known as the capacity of our short-term memory). This may be what Horn based his method on given his repeated appeal to the limit of our short-term memory:

"Since readers can at best retain no more than 5 to 9 pieces of information in *short-term memory*, they comprehend material that has been 'chunked' more quickly and more completely."<sup>28</sup>

In the two experiments that Miller cites, subjects were given several stimuli in succession and then asked to immediately recall them. The types of stimuli included binary digits, decimal digits, letters of the alphabet, letters plus decimal digits and monosyllabic words. The results:

"With binary items the span [of immediate memory] is about nine [and it] drops to about five with monosyllabic English words."<sup>29</sup>

---

27. *ibid.*, p. 90

28. RE Horn, *op.cit.* p.3-A-2. Also on page 12-2. Emphasis added.

29. Miller, *op. cit.*, p. 92

A span of 5 to 9 looks suspiciously like Horn's magic range of  $7 \pm 2$ . But just how relevant to comprehension is a measure of our ability to *immediately* recall a list of digits, words or whatever? Take one of Horn's maps, for example. A map can have  $7 \pm 2$  blocks and each block can have  $7 \pm 2$  sentences (or a list of  $7 \pm 2$  items). Now consider the simplest map possible while keeping within the  $7 \pm 2$  range: a map with five blocks each with a list of five items. Now try this experiment: on each of five cards write a list with five items in it. If you like, make it easier for yourself to remember the items by including only like items in each list (say the names of birds in one list, cities in another and so on). As you complete a list, turn the card over so that you cannot see what you have just written. When you have finished writing all five lists, try to recall, without looking at the cards, every item in all five lists in the order in which you wrote them. What are the chances of you correctly recalling all items? About the same as getting a money-back guarantee from a palm reader. What about recalling all the items in any order?<sup>30</sup> Only marginally better. You will probably recall the first few and last few items—what psychologists call the *primacy* and *recency* effects—but struggle with those in the middle. But if Miller's recall research is relevant to comprehension at the level Horn envisages, you should be able to correctly recall all 25 items, since they were in just five chunks (that is, five blocks). But if you can't correctly recall the items, you can't possibly have comprehended the material.

It might be retorted that a test of memory is not a test of comprehension, and that is true—in some cases. I might be able to recall the sentence "The work done was five ergs" but have no comprehension of what it means if I haven't studied physics. But in the case where you have listed the words to be recalled yourself, it is difficult to see what comprehension could be other than recall. Ability to recall at a later date, perhaps? Well that's hardly likely to be better than your ability to recall immediately after listing the items. Anyway, we are now straying far from Miller's paper, which we are told is the foundation for the Information Mapping method. Miller never examined delayed recall. Nor did he examine comprehension.

We poorly remember five lists of five items (and even shorter material) because, by definition, short-term memory is short:

"Forgetting over intervals measured in seconds was found."<sup>31</sup>

"... 30 seconds is ample time for forgetting to occur".<sup>32</sup>

This makes Horn's view that the limited capacity of our short-term memory should compel writers to apply the  $7 \pm 2$  chunking rule at the *molecular* level—

30. It's not clear from Miller's paper whether the tests discussed were of serial recall (recall in the order given) or free recall (recall in any order).

at the level of blocks, maps and sections—decidedly odd. By the time I've read even one seven-chunk block in, say, a map, 30 seconds is likely to have passed. If there are seven blocks in the map, many minutes may have passed by the time I have read them all. That is, as Baddeley says, "ample time for forgetting to occur". Further, a map could have over 50 items in it: 7 blocks each with a list of 7 items, plus all the headings. Given Miller's limit, my short-term memory will have been flushed out many times by the time I reached the end of that map. So what is the relevance of limiting a map to seven blocks? If  $7 \pm 2$  applies only to *short-term* memory—and that is all that Miller said—why must it apply to chunks of material that could never be accommodated in a single frame of short-term memory?

The capacity of our short-term memory might well be relevant to our ability to take in and comprehend material at the *atomic* level of a text—the clause or sentence—but at the *molecular* level—the level of blocks, maps, sections and chapters—its relevance is doubtful. If this is what Horn is claiming, then the onus is on him to advance some supporting research.

## Flawed reading assumptions

Even if Miller had been right in thinking that the span of immediate memory is limited to  $7 \pm 2$ , the way this span was established—by testing *immediate* recall—and the span itself are quite irrelevant to the way people read. Suppose I want to know how to use time-shift on a personal video recorder and consult the accompanying user guide for instructions. Suppose, too, that I encounter a seven-step procedure (indeed, the number of steps is irrelevant for this argument). Am I meant to read each step and then recall them all before I can successfully complete the procedure? Of course not. Further, in a recall experiment such as those Miller considered, the subject cannot ask for the items to be repeated; in the real world a reader can go back and read an earlier step in a procedure, if they really needed to. The relevance of Miller's work is looking shakier and shakier.

Further, hierarchical chunking—from block to chapter—is unlikely to help most readers of many types of documents. Consider the types of documents Horn was primarily concerned with: procedures and policies. Most readers don't read whole chapters in user manuals and sets of work

31. LR Peterson & MJ Peterson, "Short-term retention of individual verbal items", *Journal of Experimental Psychology*, 1959, vol. 88, no. 3, p. 198. The experiments reported in this paper suggest that after about four seconds, correct recall drops to about 50%, and to zero after 18 seconds. See figure 3 on page 195 of the paper.

32. A Baddeley, *Working Memory*, Oxford University Press, Oxford, 1987, p. 10.



instructions, the types of documents where procedures and policies are prevalent. They dip into such documents when they want to learn (or be reminded of) how to do something in particular. They want to activate time-shift on their personal video recorder and the steps are not obvious. They might then scan the contents pages or index, or electronically search, for the topic of interest and then read *just that topic*. Perhaps in the effective life of a personal video recorder, the owner might consult the user guide a dozen times but never read it through in its entirety. Much to the chagrin of technical writers, a user guide is primarily consulted as a last resort: when the product it describes does not work as the user expected it to, or when the information needed cannot be got by asking someone else. And when it is, only a small part of it is consulted at any one time.

In Information Mapping, there is a special type of map for procedures and work instructions: the procedure map.<sup>33</sup> If most readers of instructional materials only dip into the materials to learn how to do a particular task, on each reading they are likely to be reading only one, perhaps two, maps. In which case they will not need any special guidance that might be given by the  $7 \pm 2$  structure of a *section*. (Recall that a section is a group of  $7 \pm 2$  maps.) Nor will they need any special guidance that might be given by the  $7 \pm 2$  structure of a *chapter*. In other words, readers of procedures and work instructions are more interested in the trees than the wood. Indeed, most won't even see the wood.

Let's put this another way: will I *understand*, say, a procedure more quickly or more thoroughly if the chapter of which it is only a small part has seven rather than, say, ten sections? Unlikely. The *molecular* structure of the chapter will simply be unnoticed (and how can something I don't notice influence my degree of comprehension). Will I *find* that procedure more quickly with a seven-section structure than a ten-section structure? Hardly, given that I, like most readers, will go to the index or table of contents for help in finding a procedure. (And even if there were no index or table of contents, molecular chunking is unlikely to improve the speed with which topics are located. Indeed it may *impede* that speed, given that complex topics that should logically be kept together—and which readers would expect to find together—might have been split across chapters, and for no other reason than to avoid breaching Horn's chunking limit.)

All this puts paid to Horn's claim that:

"By chunking information the writer improves the reader's comprehension and access and retrieval speed."<sup>34</sup>

The *molecular* structure of a document has no bearing at all on my ability to understand any *atomic* part of

that document; nor does it necessarily improve how quickly I can find a particular part of that document.

Even if readers did notice the molecular structure of a document, how can the number of items in any particular structure or sub-structure interfere with one's ability to understand material at the atomic level? Suppose that two people of otherwise equal intelligence, education and experience are asked to read and comprehend a paragraph (the same paragraph in both cases). Let's call them A and B. In A's case, the paragraph to be read is embedded in a group of seven paragraphs. In B's case, the paragraph to be read is embedded in a group of 10 paragraphs. A and B are both asked to count the number of paragraphs presented to them before reading, and answering questions about, just the specified paragraph. Can we conclude that A—with fewer paragraphs in the material presented—will better comprehend the paragraph? In other words, will A score higher than B on the comprehension questions? It would be surprising if any evidence could be found to support that view. Certainly Miller didn't do such an experiment, and thus the claim that Information Mapping is based on Miller's research is looking even slimmer still. Indeed, the *prima facie* silliness of the idea that the super-structure of a document affects comprehension of a sub-structure puts the onus on information mappers to advance some plausible evidence in support of the claim that the limited capacity of our *short-term* memory helps us comprehend anything other than atomic information (such as a clause or sentence).

## Is Miller relevant but memory not?

Horn's main point is that "since readers can at best retain no more than 5 to 9 pieces of information in short-term memory [writers are compelled to apply a chunking limit of  $7 \pm 2$ ] at every level of a written document" (see page 7). Short-term memory, it seems, is important. It is *the* determining factor. We also noted that the website of the Information Mapping company of which Horn is the chairman states that "the chunking limit is a guideline, based on George A. Miller's 1956 research" (see page 9). As we've just seen, short-term memory is the very stuff of Miller's paper, especially its role in judgment, attention and recall. What, then, are we to make of the rest of the reference to Miller on the Information Mapping website:

"The chunking limit is a guideline, based on George A. Miller's 1956 research, for creating information that people have to memorize. Documents do not have to be 'memorized', but maintaining these chunking limits aids in a reader's ability to process information."<sup>35</sup>

This is puzzling. If readers don't need to memorise material they read—which is true—then why must that material be limited to chunks that do

33. RE Horn, op. cit., p. 4-B-1

34. op.cit., p. 3-A-2

## HyperTrain dot Com

HyperTrain dot Com specialises in:

- Online Help
- HTML
- JavaScript
- CSS
- XML
- XSLT and
- DITA.

We also offer a full range of tech pubs services including group or individual training, project jumpstart, technical writing, and end-to-end project development.

Toll free (US): 888-722-0700  
International: +1-760-214-0698  
Web: [www.hypertrain.com](http://www.hypertrain.com)  
Email: [dgash@hypertrain.com](mailto:dgash@hypertrain.com)

not exceed the capacity of short-term memory? And why say that the chunking limit is based on Miller's research when Miller's research says nothing about what might aid a reader's ability to process information *other than information that has to be immediately recalled*. If we are not discussing information that needs to be immediately recalled—which we can't be if memorisation is not an issue—then we can't be basing our research on Miller. So where is the research that shows that even though memorisation is not required of readers, writers must "at every level of a written document" chunk their material in line with the constraints imposed by the capacity of short-term memory? Claiming Miller as an authority is looking a touch like fabrication.

But the very same web page on which we are told that "Documents do not have to be memorized" also tells us that:

"Chunking... involves making the information digestible either for memorization or comprehension."<sup>35</sup>

So perhaps all that the first quote meant was that we don't have to memorise *whole* documents in order to understand their contents. That's too obvious to warrant discussion. But what about parts of a document? It's worth repeating here that Miller was concerned solely with *immediate* recall. Does anyone

ever attempt to—or even need to—memorise a chapter, section, or even a map for *immediate* recall? Of course not. Does anyone even attempt to memorise a chapter, section, or map for *later* recall? Again, of course not. I might prefer to remember the main points in, say, a map so that I don't have to resort to the user guide in the future. But if my attempts at memorisation are not tied to the limits of my short-term memory—if I have time, that is, to rehearse and to mnemonically code what I want to remember—then Miller's magic number, pertaining as it does to *immediate recall*, is utterly irrelevant. And if the concern is to minimise the effort involved in rehearsing and mnemonically coding what I want to remember, then perhaps we should be applying the *minimum* chunk limit possible before recall errors occur: three (as noted in the next section). Miller was, you may recall, talking about the *limits of perfect recall*.

A parallel should reveal the absurdity of Horn's position, if such revelation is still needed. A listener does not need to memorise a melody, section or movement of a piece of music in order to appreciate it. (Otherwise we would never be enthralled by a piece of music we had never heard before.) Now let us suppose, for the sake of argument, that humans can only accurately hum 7 notes of any melody they hear for the first time when asked to immediately recall it. Does it then follow that composers should restrict to seven the number of notes in a melody, the number of melodies in a section, the number of sections in a movement, and so on? Again: of course not.

## Research after Miller

In a 98-page article published in 2000, American psychologist Nelson Cowan reviewed the data then available on the limits of short-term memory. He noted that researchers post-Miller had found that short-term memory is limited to between three and five chunks. In summarising the masses of data he reviewed, Cowan concluded that there is a:

"single, central capacity limit averaging about four chunks ..."<sup>37</sup>

He also noted that:

"[Miller's magic] number was meant more as a rough estimate and a rhetorical device than as a real capacity limit."<sup>38</sup>

Perhaps the Information Mapping fraternity might like to update their research instead of relying on Miller's survey of a handful of experiments conducted more than 50 years ago. If the span of short-term memory is only  $4 \pm 1$  and if, as Horn claims, the span of short-term memory sets the chunking limit, then Information Mapping needs to

35. See [http://www.infomap.com/index.cfm/themethod/Mapping\\_FAQs](http://www.infomap.com/index.cfm/themethod/Mapping_FAQs) (viewed 23 January 2011).

36. *ibid.*

37. N Cowan, "The magical number 4 in short-term memory: A reconsideration of mental storage capacity", *Behavioral and Brain Sciences*, vol. 24, 2000, p. 87

38. *ibid.*

be radically updated to bring it into line with current knowledge in cognitive psychology.

However, Cowan's reported limit is a limit on the recall of *unrelated* items (as was the limit reported by Miller). But when psychologists look at strings of *related* words—such as words in a sentence—the span of immediate memory is significantly greater:

"Immediate memory for sentential material is typically substantially greater than span for unrelated words ... Baddeley *et al.* ... found spans of around five for unrelated words [in line with what Cowan reported] and 15 [words] for sentences."<sup>39</sup>

So if the span of short-term memory determines the chunking limit—as Horn contends—and if the span of short-term memory *for the sort of material that writers primarily present to readers* (namely sentences) is 15, perhaps the chunking limit should be raised to 15. Either way, Information Mapping has been left behind by research more recent than Miller's. A limit of  $7 \pm 2$  is yesterday's guesstimate. Today it is  $4 \pm 1$  for unrelated items and 15 words for sentences, the very entities that Horn claimed are the "first and basic unit of information" (see page 8).<sup>40</sup>

Then again, short-term memory, as we've seen, has little bearing on the optimum way to structure a document. It might be relevant to the construction of a sentence (or even a short paragraph). But it is irrelevant to the molecular structure of a document. The molecular structure is either unnoticed by readers or, if it is noticed, does not influence our ability to comprehend material at the atomic level.

## What George Miller might say about Information Mapping?

Well-known author and freelance editor Mark Halpern wrote to George Miller in the mid-1990s when confronted with a workplace edict to limit the items in a list and the steps in a procedure to  $7 \pm 2$ . Halpern knew that Miller's name was associated with research on the limits of cognitive processing and that Miller had publicly complained about the unfounded conclusions some had drawn from his research. Miller replied to Halpern detailing one of those unfounded conclusions. In the 1970s, some local authorities had passed by-laws restricting the number of items that could be displayed on a billboard to  $7 \pm 2$ , using Miller's research to justify the laws. (It turned out that a group of landscape architects, funded by the big motel chains, had

lobbied the authorities to introduce the law.) In his reply to Halpern, Miller said:

"... the point was that 7 was a limit for the discrimination of unidimensional stimuli (pitches, loudness, brightness, etc.) and also a limit for immediate recall, *neither of which has anything to do with a person's capacity to comprehend printed text.*"<sup>41</sup>

Perhaps the information mappers might wish to reconsider citing Miller's research as the basis for their chunking rule.

## To sum up

- Information Mapping is not based on Miller's research nor on any research that Miller quoted. To claim that it ignores, or misunderstands, what Miller actually wrote.
- Even so, Miller's research has been superseded by more recent studies.
- Information Mapping is at odds with the way people read texts. The molecular structure of a text is rarely if ever noticed by readers (whether seeking spur-of-the-moment information or in the exceptionally rare case where a reader reads the text all the way through).
- Even if readers did notice the molecular structure of a text, no plausible evidence has been adduced to support the claim that the limited capacity of our *short-term* memory restricts our ability to comprehend anything other than atomic information (such as a clause or sentence). At the molecular level of chapters, sections, maps and blocks, the capacity of our short-term memory appears entirely irrelevant.

To conclude: a document-wide chunking limit of  $7 \pm 2$  is not science. It is pure bunk.

## Geoffrey Marnell

41. The entire thread between Halpern and Miller can be read at <http://members.shaw.ca/philip.sharman/miller.txt>. Emphasis added. Viewed 14 January 2011. In an email to me on 8 February 2011, Halpern confirmed the accuracy of his reported exchange with Miller. An email to Professor Miller at his last-known Princeton University address bounced.

39. A Baddeley, *Working Memory, Thought, and Action*, Oxford University Press, Oxford, 2007, p. 143

40. Note that Horn's omission of sentences from his  $7 \pm 2$  chunking rule (see page 8) is not relevant here. We are only considering the raw capacity of short-term memory, which Horn uses to justify the chunking of blocks, maps, sections and chapters.

## The Words team

Artwork: Christine Weaver

Copy-editors: Claire Mahoney & Ross Mortimer

Editor: Geoffrey Marnell

Contact: [words@abelard.com.au](mailto:words@abelard.com.au)

© Individual contributors or Abelard Consulting Pty Ltd, 2011  
[unless otherwise noted]



# The grass is greener on the open side

Lana Brindley

Freedom gets mentioned a lot when discussing open-source software. But what does *freedom* mean? Like many English words, the meaning of *freedom* depends on the context. This is because the word has a number of distinct ancestors in Latin, the most notable of which are *gratis* and *liber*.

*Gratis* means free of charge, or without cost; *liber* means the state of being free, or of having liberty. Both meanings of *free* can be applied to software. Let's see how.

We all know that the best beer is the beer you don't have to pay for. That is, it's beer that is *gratis*, or free of charge. We can refer to software as being "free as in beer" when we mean that it doesn't cost any money to use. This is the meaning of *free* when we are discussing *freeware*. Freeware is free as in beer, but it has its catch: you still need to read and agree to the end-user license agreement in order to use it. This usually means that you won't be allowed to change the way the software works, or create any add-ons or extras, such as documentation or translations. In many cases, freeware can only be installed on personal rather than business networks, and quite often it cannot be shared.

When we use *free* in the sense of *liber*, we mean *freedom* (as in freedom of speech). When we talk about freedom with regards to open-source software, this is the freedom we mean. It's the freedom to see the nuts and bolts of the software you're using, the freedom to make changes and share them with your friends, the freedom to take the code and use it in your own project, and the freedom to suggest and submit changes to the code itself, or the stuff that wraps around the program (such as the documentation).

## Have you got a licence for that thing?

Before we move on there's one other term I'd like to straighten out: *pirates*. My entire network at home was set up using free software: that's free as in beer. It didn't cost me a cent. However, I'm not a pirate (and that's not just because I don't have a wooden leg and a parrot). Every piece of software I use in my home network is open-source and was obtained perfectly legally. This is because the free-as-in-freedom and the free-as-in-beer is written into the license agreement for the software I use.

You're probably familiar with the end-user licence agreement (EULA). That's what you have to agree to when you install closed-source software. It's usually a long chunk of text, written in legalese, and we all ignore it and click **I agree** to continue. Open-source software doesn't have a EULA, but it does

have a licence. The licence works in more or less the same way as a EULA, except instead of saying "You may not sell, license or distribute copies of the software" it says something like "you can use this software free of charge, as long as you keep it that way". In other words, if I wanted to pay for it, or I wanted to sell it to my friends, I would be breaking the agreement, in the same way that giving away copies of Microsoft Office for free would be breaking the EULA.

There are lots of open-source licences, but they all work in much the same way, with only minor differences between them. The main licence is the *GNU General Public License*, which is referred to as the *GPL*. The main restriction imposed by the GPL is that whatever you do with the code covered by the licence, it needs to include a copy of the GPL with it. And that's really as scary as it gets.

## Decision time

Suppose that you're in the market for a new bit of software. Often your purchase decision will come down to features, and if one option has the features you need and the other one doesn't, then it makes sense to buy the one that has those features. Provided you agree to the terms of the licence or the EULA, and you pay whatever is requested, then there's no issue. But what about when the features are equal, and the difference comes down to licences and cost? Most big-name software will cost you money in some form or another for the full version. After you've paid your money the product is yours though, right? Wrong. The software company can decide to change the software whenever it wants to. You would have all seen this happen on your Windows machines. You agreed to a EULA when you installed it, but Windows gets new updates every other day. Got any idea what's in those updates? No, and nor do most people. We need to trust that the big software companies are going to do right by us, and in most cases that's reasonable to do. They're big companies with millions of users, and if they tried anything nasty, we'd probably know about it. So accepting automatic updates is a risk most of us are perfectly willing to take.

So you've paid your money, you've got your software, and you've been happily working away with it for a while. Then someone sends you a file that you can't open because it was created with a newer version of the software. All of a sudden you realise that your old version doesn't have the features that the new version has got.

So what do you do? You have to upgrade, which incurs more cost. Once again, you click through the EULA, agree to it, pay your money, and you're off. You're probably very familiar with this process.



Now, what happens if you don't like something about the software, or if you discover a bug in it? For the most part, you probably shrug it off. There's not much you can do about it. What about the documentation? We've all come across laughably bad documentation. What happens when the documentation for your piece of software doesn't describe things properly? Or doesn't include information that you need? Have you ever thought "Gee, I could write that so much better". If you're multi-lingual, have you ever wished that a company would provide documentation in a different language? Have you ever wanted to create a guide that covers a situation you use every day, and you think others would find it useful too? You can't do any of that with closed-source software, because the EULA specifically forbids it. The only way you could make those changes would be to get a job at Adobe or Microsoft. And while we'd all love to land a position like that, unfortunately for most of us it's unlikely.

So let's look at an alternative. Most open-source software will cost you nothing. It's free as in beer and free as in freedom. You can go to the website, pick the version you want, and you're off. If you need help, you can check out the embedded help, or the official documentation on the site, just like with any other software. And if you don't like either of those options, there are many other places to get help: wikis, forums, chat channels, blogs and websites. You could also go to an open-source manual website, like [flossmanuals.net](http://flossmanuals.net), and find out if someone else has written a full guide. And what about problems or bugs? The first thing to do is to search the web. It's possible that someone else has come across the issue and has already found a solution. If not, then get in contact with the developers—all open-source projects have a number of ways to do this—and let them know about it. They'll probably ask you for some more details about the problem so that they can get it fixed, and then they'll go right ahead and fix it for you.

The other fun part is if you think you have something to add. If you're a programmer and you'd like to write a new feature, or fix a bug, you can do that. Likewise, if you're a writer and you want to improve the documentation, or you want to do a translation, then you can do that. In those cases, you will usually be welcomed with open arms and given everything you need to get started. And that's because open-source software is created by a community, and anyone who wants to be a part of that community and work to improve the software they're using will always be welcome. Of course, you don't have to contribute to a project if you don't want to. You can just download the software and use it, just as you would with any other software.

## Arguments, naturally

Of course there are arguments both for and against open- and closed-source software. Most of them, on both sides, have some merit. One of the main arguments against open-source software is that closed-source software is usually more stable. This is because the company producing the software will have a room full of developers who are paid to write features and to fix bugs. In some cases it's true. The stability of Microsoft Word is no doubt better than the stability of software created by two guys in a garage for three friends to use. But this is not a fair comparison. A fairer approach would be to compare the stability of Microsoft Word with the stability of Open Office, an open-source project supported by Oracle, IBM and others that has been around for over 10 years. Neither Word nor Open Office is likely to go away any time soon.

The other side of the coin is the little software development groups, those without the backing of industry giants like Oracle and IBM. Anyone can produce software and sell it, and if those little development groups go bust, you can end up with an unsupported product. But that's the case whether their software is open- or closed-source. The difference is that with open-source, the code is available to anyone who wants to look at it, so there's at least a chance that someone at some point will pick up the code and have another bash at it. That's never going to happen with closed-source software, simply because the licence won't allow it.

## I'm still not sold

The good news is that you don't need to be totally sold on either open- or closed-source software. Open- and closed-source software can co-exist on the same system. And because open-source software is free to use and easy to get, you can freely download it to your current computer and see if you like it.

## It's not just about the freedom

Open-source advocates bang on a lot about freedom. Freedom is a driving force behind the open-source movement. But I think there's more to it than just freedom. There is also *community*. Whenever you get group of people together with a common goal, they can achieve just about anything. When the goal of that group is freedom, then I think that the world can really only become a better place because of it.

### Lana Brindley

This article was originally presented by Lana Brindley as a web seminar for the Society of Technical Communication. Lana has been writing technical documentation for about five years, but has been an open-source advocate for much longer.

# Journals

## Journal of Technical Writing and Communication

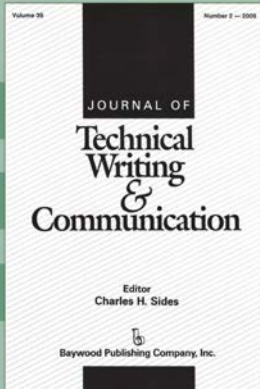
### Contents of the current issue

Click the heading below to read the editorial, abstracts and reviews, or to purchase an article or subscribe.

#### Volume 41, Issue 2, 2011

- Moving International Technical Communication Forward: A World Englishes Approach, Michael Jarvis Kwadzo Bokor, Ph.D.
- Outlining Purposes, Stating the Nature of the Present Research, and Listing Research Questions or Hypotheses in Academic Papers, Wasima Shehzad
- Verbal Abuse in the Army of the Cumberland: William Rosecrans' Acid Tongue as a Major Factor in the Union Defeat at the Battle of Chickamauga, Max Loges
- Meeting Students Where They Are: Advancing a Theory and Practice of Archives in the Classroom, Christina Saidy, Mark Hannah, Tom Sura
- A Bibliography of Works Published in the History of Professional Communication from 1994-2009: Part 1, Michael G. Moran and Elizabeth Tebeaux

award-winning, authoritative international voice, publishing the latest research by recognized scholars from around the globe



Every article ever published in the JTWC is now available in clear, concise, comprehensive PDF format.

**New Online-Only Delivery Options**


- Enhanced subscription (all articles from volume 1 through current volume)
- Current volume subscription
- Individual articles on pay-per-view basis

**Click here for all the details and ordering information.**

- Read all article abstracts free
- Download a complimentary sample issue


**BAYWOOD PUBLISHING COMPANY, INC.**  
<http://baywood.com>

### Journal




quarterly  
52 full colour pages  
£45 a year (£10 off for ASTC members)  
contribute: [journal.editor@istc.org.uk](mailto:journal.editor@istc.org.uk)  
archive (ISTC members only): [www.istc.org.uk/Members\\_Area/communicator\\_archive.htm](http://www.istc.org.uk/Members_Area/communicator_archive.htm)  
**subscribe: [istc@istc.org.uk](mailto:istc@istc.org.uk)**  
**advertise: [felicity@tou-can.co.uk](mailto:felicity@tou-can.co.uk)**

### Broaden your horizons with subscriptions to



### periodicals

### Institute of Scientific and Technical Communicators Newsletter



monthly  
PDF (website or e-mail)  
freely available at no charge  
contribute: [newsletter.editor@istc.org.uk](mailto:newsletter.editor@istc.org.uk)  
archive: [www.istc.org.uk/Publications/Newsletter/newsletter\\_archive.htm](http://www.istc.org.uk/Publications/Newsletter/newsletter_archive.htm)

## Book reviews

■ *Txtng: The gr8 db8*, David Crystal, cartoons by Ed McLachlan, Oxford University Press, Oxford, 2008, 239 pages, ISBN 978-0-19-954490-5.

Reviewed by Jim Rountree

A great debate? Texting? I'm sure I'm not the only one who might take that with a grain of salt. Before reading this book I was certainly aware of complaints that texting was undermining the language—how children in England, for example, are apparently unable to spell in any other way than “txt”, even in exams. But I wasn't too worried. I've heard the more general claims of this type—poor education standards are perhaps most often blamed—but been convinced by the Pinkers of this world that most of what people see as the decay of the language is simply linguistic change, not degradation.

So, in fact, I wasn't convinced that there was a serious issue to address about texting—not one that might require a whole book to investigate, at least. On the other hand, having read *How Language Works* by Crystal and appreciating his straightforward, clear style (not to mention his authority—author of the third edition of the *Cambridge Encyclopaedia of Language* for example, amongst much else) I wondered if I was missing something. In the end, at least with respect to the main arguments of the objectors (I am being generous calling them arguments), I don't think I was. But that's not to say I regret having read the book.

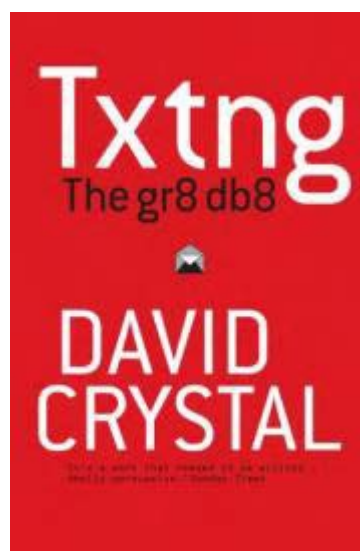
Crystal, I should say, makes clear at the outset his view that texting is “...not a bad thing” [p10]. So for me the worry was that he had set himself up against something of a straw man. As it turns out I suspect the “gr8 db8” of the title shows the influence of the publishers, more than anything else. Crystal uses quotes from the detractors to introduce sections (the strength of the opinions voiced certainly adds emotional spice), but what follows, while it inevitably successfully rebuts the detractors' points, stands on its own merits. Even if there isn't the development of argument and counter-argument that you would expect in a debate, it is still interesting to have a close look at what really occurs in texting, and what it might signify.

For a start, *Txtng* presents a good selection of real data, from real research, that puts texting into context. We learn that fewer than 20% of texts use any abbreviations at all, for example. Which means that straight away we need to redefine the detractors' target—it is not texting *per se* that they disapprove of, but the small proportion of it that uses abbreviations, initialisms, pictograms, logograms and alternative spellings. Following others, I will call these devices collectively “txt”, or “txtng”, or “txtsms”, as appropriate (just to get in the spirit of things). Crystal gives thorough descriptions of each of the txtng tricks of the trade.

No one can deny that texting is a significant new medium, and its uptake has been quite incredible (if not entirely unique, in our ever digitising world), so it's good to learn how far it has penetrated, and something about what messages actually get sent. I skimmed some of the facts and figures, I admit, but they do not weigh the book down too much if, like me, you have a limit for the number of numbers you can take. But quite apart from the edifying accumulation of facts, there was a lot about the book that I really enjoyed. It's made explicit in the chapter “Why do they do it?” Two reasons: it's quick (who'd have thought?), and ... it's fun. At least some of what the txtrs think is fun, I like too. But I'll come back to this—first a little about the first reason people txt: to be quick.

As Crystal points out, phone interfaces are poorly designed for texting. Most phones have a numeric keypad, so you frequently need several presses to get particular letters, and more if you're capitalising. And if a letter is commonly used it gets no special treatment—it might still be four presses away. So for anyone, but especially a ham-fingered texter like me, frustration rises, and any way of completing a message more quickly is happily taken. Well, not always happily. A friend told me she hated text abbreviations. “So you don't use them?”. “I do, but I hate it”. Even the defenders of a pure English (or Japanese, or French, or Spanish *etc.*—they all text) can be driven to txt by the practicality of fewer button presses.

So, if we're going to cut corners when we text, how do we do it? Crystal observes that one of the complaints about texting is that outsiders can't understand it—“another language” some go so far as to say. But of course, if your intention is to get a message across to someone, you can deviate from standard language only to the relatively small extent that you can still be sure the receiver will understand it. So when people use non-standard English it's most often in ways that aren't too hard to figure out. This isn't to say that people, especially young, won't





develop in-group txtsms that others can't follow, but that is hardly unique to texting. Internet text dictionaries have thousands of initialisms and other texting tricks that you and I could never figure out, but we are not in the in-groups that developed them (and developed them in good part to keep the likes of us out) so this is nothing to worry about. And, in fact, it seems likely that many of these terms must have fleeting lives even within their groups ... how much usage is AGKWE (and God knows what else) going to get before it's boring?

So, essentially, what Crystal tells us is that if people want to shorten their messages, because it is quicker and easier, they make up their own abbreviations. He has seen *tonight*, for example, as *tnight*, *tonyt*, *tonite*, *tonit*, *2nt*, *2night*, *2nyt* and *2nite*. And of course, abbreviating is nothing we haven't done before, in notes to ourselves, or telegrams, or, indeed, in usages that are fully incorporated into standard English (e.g. RIP etc.).

But then comes the fun factor. It seems that txtrs, having made the first step to shorten their texts in whatever ways occur, or appeal, to them, take it as licence to go further. Crystal argues that our general proclivity to play with language comes in, and far beyond simply trying to get out something understandable with the fewest possible presses, people play with spellings and introduce other devices just for the fun of it. This drive can even lead in some cases to alternative spellings that take more presses than the real word.

Crystal has got to be right about this. We love playing with language—in light-hearted fun (funny accents, puns), pastimes (crossword puzzles, Scrabble) and more sober amusements (poetry and novels)—and it seems obvious that txtng falls somewhere amongst this (mostly at the more light-hearted end, I'm sure). I think, for example, of a recent text from my daughter, responding to a text from my wife telling her what we were up to on holiday in New Zealand. *kewl*, she txted. She was in Australia texting us in New Zealand, so she said *cool* with an Australian accent.

I quite like the phenomenon of “textonyms” that Crystal mentions. On phones with predictive text, the same key presses can specify different words. 2665 gives *cool* as well as *book*. My daughter could have sent us *book*. But of course it's an in-joke and we wouldn't have got it—we would now. *Request* and *pervert* is another pair of textonyms.

And then there are poems. Crystal presents quite a few, starting with the winners from a text-messaging poetry competition run by the *Guardian* in 2001. One of the poems doesn't use txt at all, but it fits within the 160-character limit of a text message. Its

compression of images reminds me of haiku, and of course, at 17 syllables, all haiku would fall well within the limit. Other poems use txt devices quite liberally, although (remembering the principle of receiver understanding) not as many, on analysis, as you think from first impression. The meanings of the txtsms were not always obvious, and I always read Crystal's gloss to decode them before thinking of trying to work them out for myself, and then regretted it. Just as in standard poetry where we read and reread to find meaning, or read and reread a riddle in order to solve it, it can be fun, working in a space somewhere between these two examples, to have to think a little to work out a txtsm.

Don't get me wrong—I didn't like all the txt-plays Crystal presents (as you may not have been amused by my examples), but the message is that we really don't have to take this all that seriously. If the kiddies can't spell, well, they can't spell, but texting is as likely to be helping them as hindering (there is evidence for this, Crystal notes).

In the end, what for me lingers after reading *Txtng* is an appreciation of the strength of our attachment to language. The outrage (moral outrage, even) that we

feel when we believe some hallowed linguistic tenet has been undermined (in spite of my accepting Crystal's position I have my own language bug-

bears, believe me), or the pleasure we take in playing with language, both show a feeling for language that goes far beyond what you would expect from something used simply to transmit the meanings that we need to, to get on in life.

So, having brought up morality, I think it apt to finish with Norman Silver's *txt commndmnts* ([www.txtcafe.com](http://www.txtcafe.com)), quoted by Crystal:

1. u shall luv ur mobil fone with all ur **hart**
2. u & ur fone shall neva b apart
3. u shall nt lust aftr ur neibrs fone nor thiev
4. u shall b preparad @ all times 2 tXt & 2 recv
5. u shall use LOL & othr acronyms in conversatns
6. u shall be zappy with ur ast\*r\*sk&s & exc!matns!!
7. u shall abbrevi8 & rite words like theyr sed
8. u shall nt speak 2 sum1 face2face if u cn msg em insted
9. u shall nt shout with capitls **XEPT IN DIRE EMERNCY +**
10. u shall nt consult a nenglish dictnry

**Jim Rountree**

Jim Rountree is a Melbourne-based technical writer

---

Why do people txt? Two reasons: it's quick and it's fun.

---



- *Culture, Communication and Cyberspace: Rethinking Technical Communication for International Online Environments*, edited by Kirk St. Amant & Filipp Sapienza, Baywood Publishing Company, New York, 2010, 243 pages. ISBN: 978-0-89503-397-7.

Reviewed by Jonathan Burns

This book is a collection of ten essays on cross-cultural communication in cyberspace. They explore the differences in symbolism, semantics and etiquette that affect such communication, and describe the experience of writers of diverse cultures working collaboratively on online systems. Cross-cultural nuances in producing educational web content, advertising, public health campaigns and business correspondence are also discussed.

These studies arise in an academic context and accordingly the writing style is factual, well-referenced and plain. This is certainly not a guidebook to being likeable, accurate or persuasive on the net, nor is there much specific information on localisation resources or utility websites. The collection is a sample of studies conducted between 2006 and 2008, and makes no claim to be comprehensive.

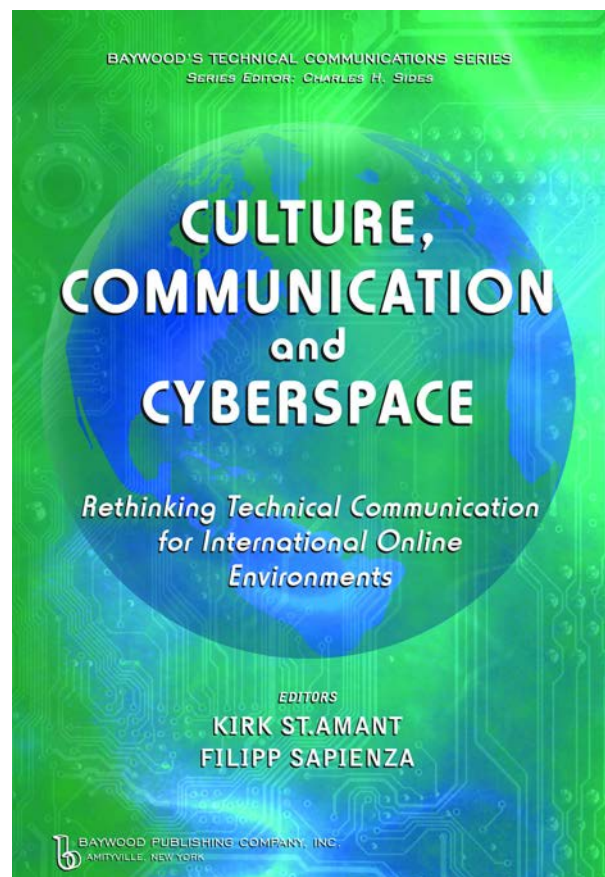
The first section explores properties and dimensions of cultures and language which may help in creating a taxonomy and analysis of communication.

- **R. Peter Hunsinger** argues for an expanded concept of localisation to cover not just the translation of content but the assimilation of the reader's culture. He argues it is a mistake only to decorate content with local images and idioms; rather we should provide a *locality*, an environment furnished with local resources that the reader would naturally use. He classifies such resources into *scapes*: *ethnoscapes* connecting people of common background or way of life; *technoscapes* making communications media available; *finanscapes* affording modes of commerce; *mediascapes* presenting and indexing content; and *ideoscapes* demarcating legality and authority. This model of locality is shown in a simple form, in the web pages of Philip Morris International as they appear in France, Korea and elsewhere.
- **Clinton R. Lanier** describes how a reader could customise Web interfaces by choosing a language and resources from a database of localisation data so that appropriate pages are automatically generated server-side. An example is Yahoo International, which provides a choice of localised portals to news, finance, RSS feeds and so on. Lanier is frank about the limits of this technique in this era (a) when information providers seem to have little grasp of how users will respond to the different rhetoric of different cultures, (b) when there is little feedback on

users' experience and (c) where interface design principles are poorly understood.

- **Matthew McCool** argues that since our adaptive subconscious mode of attention handles polysemous language with more versatility than our analytic attention, information retrieval by search engines should be backed up by free-form semantic networks rather than strict-form structures. (This appears similar in intention to the push for the *Semantic Web* where data is supplemented with structured metadata accessible to search engines and hopefully artificial intelligence. The danger is inserting automation between the provider and the user when personal feedback is critical.)
- **Martine Courant Rife** reviews writings in international intellectual property law. She argues that identity, membership, originality and property become different phenomena when information is widely and cheaply shared, and that the law must adapt to remain applicable. As ownership or membership cease to be supported by physical location or evidence, legal definitions may tend to get their authority from the rhetoric with which they are stated. She certainly has an interesting variety of cases to quote, such as a legal case restricting commercial use of the name *catfish* to one traditional species.

The second section of the book deals with the varying forms and conventions of discourse between cultures.



- **Daniel D. Ding** shows a traditional Confucian method at work in the Chinese Foreign Ministry's visa application web page in naming all objects systematically. This contrasts with many Western presentations along the lines of "Your goals and how to achieve them", and the Aristotelian model of hierarchies and oppositions. The use of standard names to confirm a reassuring common context is a point well made, and should warn us against enthusiastically coining neologisms. At the same time, Ding points out the Chinese Foreign Ministry's website can be frustrating to the westerner hoping for a how-to guide.
- **Carol M. Barnum** explores the widely-cited Hofstede measures of cultural values (especially in relation to the distinction between individualism and collectivism). She shows how Indian and Chinese business correspondence changes according to the Hofstede measures. Several examples of actual correspondence are discussed, and it is alarming to see in how many ways one can get things slightly but tellingly wrong (or how snobbish one's education can make one).
- **Boyd Davis et al.** report on their detailed inspection of the language of Chinese and American students in online letters of introduction and essays. They ask "what variety of English one might expect and how one might expect it to be taught: from a first-language, second-language, foreign-language, additional language, or specific-purposes perspective". They suggest that an appropriate corpus of standard examples of writing be used as a standard for this kind of research.

The third section of the book deals with collaborative learning efforts supported by software and communications.

- **Audrey Bennett, Ron Eglash and Mukkai Krinshnamoorthy** describe a "virtual design studio" shared with people in Kenya for an HIV prevention campaign. This was a typical graphical design exercise but also involving the Learnlinc collaborative system with VOIP, shared interactive whiteboard and shared editing tools.
- **Judith B. Strother** explains the theory and strategy behind the design of a blended web and classroom course in aviation English for pilot trainees by the Florida Institute of Technology. Respect for the students' personal qualities is notable, and so is the use of cultural measures (Hofstede again) so as not to impose a course format but to distinguish suitable options. In the end, online learning seems ancillary to good classroom teaching of the traditional kind.

- **Sipai Klein and Sharon Trujillo Lalla** discuss how similar cultural measures might inform learning management systems. After a rather abstract discussion, a case study is discussed; but this mostly seems to be just a sensible exercise in using discussion forums and other software tools pragmatically.

Literature reviews are a prerequisite to research. They make a map of the subject, a chronology and a contest of working theories. The researcher picks those items from the literature which support or challenge the specific questions under study, so that the reader not only sees what the researcher is doing, but how and why.

There are roughly fifty pages of literature reviews in this book, not counting references. Some of the discussion directs us to influential studies in the sociology of communications. Some refers to relevant history, such as Tynes' story of how the Leonenet mailing list maintained the institutions, and the national identity, of Sierra Leone when the country was temporarily without a government.

But much of the discussion consists of diffuse, tentative and abstract statements proceeding from broad generalities about the challenges of transnational communication, through to thematic statements defining the topic, and finally to the author's own work, with no variation of tone (as much as to say "Now we come to where my work makes a difference"). Instead of illuminating the issues at hand, this background discussion submerges them in truisms.

Moreover, the essays are more than two years old. While this book was being prepared there must have been hundreds of projects undertaken and essays written covering similar ground. One can't but feel that even though the authors' works are appropriate and mostly well-founded, they seem fated to be overtaken by events.

I would not take this book as an introduction to communications studies, but I would scavenge the references for key authors to look up elsewhere. Nor would I take it as a summary, because no summary is possible at this time: cross-cultural communications is too diverse a field, computer media are changing too quickly and their adoption is in an explosive phase. But if I became involved in a cross-cultural media project, I would like to have some sketch of a communications philosophy at hand, and hence I would very likely consult this book for applicable insights and reminders.

---

**Jonathan Burns**

Jonathan Burns is an Melbourne-based technical writer.

# Tips and Tricks

## Outlook templates

If you often send similar emails, you probably spend a good deal of time searching for a previous version to use as a template so as to not waste time writing an almost identical version of it. Perhaps the time it takes to search for a previous version is longer than it takes to rewrite the email from scratch (especially if the email you are looking for is in an archive).

One way to avoid such time-wasting is to turn a potentially multi-use email that you are about to send into a template. Then when you need to send an identical or similar email, you simply open the template, make whatever changes are necessary and send it on its way. Here is how it is done in Microsoft Outlook 2003 and 2007.

1. Write the email, but don't send it just yet.
2. Select **File > Save As**.
3. Give the template you are about to create a name, choose **Outlook Template (\*.oft)** from the **Save as type** drop-down list and click **Save**.
4. Send the email.

When you want to send an identical or similar email:

1. Open the **New** drop-down list and select **Choose Form**.
2. Select **User Templates in File System** from the **Look In** drop-down list.
3. Double-click on the name of the template you want to use.

The email you originally saved as a template appears.

4. Make whatever changes are needed and click **Send**.

## Manipulating objects in a PDF File

In the previous issue of *Words* we explained some of the changes you can make to a PDF file using the **TouchUp Text Tool** feature in the full version of Adobe Acrobat. You can also manipulate objects with the **TouchUp Object Tool** feature. You can, for example, delete, move or scale an object. You can also delete, move or scale a block of text.

If you have tried to do this, you may have had problems selecting just the object you want to manipulate. You may have found that when you clicked on an object, other objects in the vicinity were also selected (objects that you didn't want to manipulate). But there is a trick you can use:



To select just one object, hold down the **CTRL** key and then click on the object.

## To manipulate an object

1. Select **Tools > Advanced Editing > TouchUp Object Tool**.
2. Select the object you want to manipulate (holding down the **CTRL** key if necessary).

You can select several objects by holding down both the **SHIFT** and **CTRL** keys and clicking on them.

3. Manipulate the object as you please. For example, you can:
  - drag it to a new position (or use the arrow keys)
  - delete it (by pressing **DELETE** or cutting it)
  - resize it (by dragging the appropriate handle)

You can also copy an object and then paste it elsewhere in the PDF. (Alas, you cannot paste it into another type of document, but you can paste it into another PDF.)

## Transposing characters in Adobe FrameMaker

A common error in typing is to mistakenly transpose adjacent characters in a word. For example, instead of typing *computer* we type *compuetr*. A neat feature in Adobe FrameMaker lets you quickly transpose the mistyped characters:

1. Place your cursor between the two mistyped characters.
2. Press **CTRL + F9**.



## MACQUARIE DICTIONARY ONLINE

[www.macquariedictionary.com.au](http://www.macquariedictionary.com.au)

Subscribe to the complete Macquarie Dictionary online, updated annually with new words and definitions.

Also available online is the full Macquarie Thesaurus - that perfect word is just a click away.

**Try it out now for FREE!**

Macquarie Online is offering free extended trial access. Simply contact Macquarie Online to set up your 3 months free access. Quote code: *3mfTrialAC*

Macquarie Online Support  
phone: 1800 645 349  
email: [support@macquarieonline.com.au](mailto:support@macquarieonline.com.au)



*Australia's national dictionary*



## Miscellany

### Word of the Year: *googleganger* ... but keep an eye out for red-light-running *mamils*

The Macquarie Dictionary Word of the Year Committee has announced the noun *googleganger* as the 2010 Word of the Year. Its definition: a person with the same name as oneself whose online references are mixed with one's own among search results for one's name. It's a portmanteau word built from GOOGLE + [DOPPEL]GANGER. (A *doppelganger* is a ghostlike double of a person, an apparition of considerable likeness felt to haunt them, a word that continues to find a place in English dictionaries from the popularity of Franz Schubert's *Schwanengesang*.)

The judging panel comprised Dr Michael Spence (Vice-Chancellor of the University of Sydney), Professor Stephen Garton (Pro-Vice-Chancellor of the University of Sydney), Les Murray (renowned Australian poet), Evan Hannah (Group Editorial General Manager, News Ltd) and Susan Butler (publisher of the Macquarie Dictionary). The People's Choice Award was *shockumentary*: a documentary film or television show featuring footage of accidents or violence.

Other additions to the lexicon given an honourable mention by the panel include *wet signature* (a handwritten signature on paper (as opposed to a digital signature)); *gym bunny* (a young woman who frequently works out at a gym because of a desire to perfect her body shape and muscle tone, especially one seen as being obsessive in her training and as flaunting her supposed resulting attractiveness); *cube farm* (an office space that is divided into small cubicles, each with its own desk); *koala ears* (patches of pubic hair protruding from the leg openings of a swimming costume or underwear); *ego-surf* (to search the internet for instances of one's own name, as in mentions in text, links to one's blog, etc.) and *mamil*, a man who takes up cycling as a pursuit during middle age, especially one who makes this a lifestyle statement by acquiring the accoutrements of a professional cyclist: the most modern and powerful form of bicycle, lycra clothing, etc. It is an acronym formed from Middle-Aged Man In Lycra.

One wonders if it is only *mamils* who encounter *googlegangers* while *ego-surfing*.

See [www.macquariedictionary.com.au](http://www.macquariedictionary.com.au) for more new words that have made it into the Macquarie Dictionary.

### Dictionaries ain't dictionaries

And while on the subject of dictionaries: two years ago you could get an single-seat annual subscription to the Macquarie Dictionary Online for \$19.95. The price doubled last year. But that's nothing compared to Oxford English Dictionary Online. A mere £246

(about AUD \$400) a year is all you'll need. (Take 20% off if you once studied in the city of dreaming spires—or is that perspiring dreams?) The Macquarie boasts 140 000 references and over 210 000 definitions; the OED boasts 600 000 words and 3 million quotations. What a boon to logophiles—or at least those with long pockets—given that the last edition of the OED rolled off the presses 22 years ago, long before the web, and the democratisation of publishing, gave English a good kick-along.

### Despising e

In 1939, Ernest Wright wrote a 50 000 word novel in which the letter *e* does not, by design, appear. Not unexpectedly, the novel—called *Gadsby*—failed to attract a publisher. We know of the book thanks only to the wonderful service provided by vanity publishers.

But books are so old hat. If you want to check out the best in *modern* vanity publishing, place a bucket beside your desk, google “worst blogs of 2010” and check out what the illiterati are smudging cyberspace with.

### Dumb, dumber, dumbest: the innumeracy of Microsoft Word

We have become so inured to the bugs in Microsoft Word that most of us now let them pass without comment. For example, no one any longer pays attention to that erstwhile annoying onscreen accusation “Fragment (consider revising)” since, more often than not, the grammar of the accused sentence is flawless.

But one longstanding bug that should worry technical writers—or at least those who write materials that will be translated—is Word's inability to correctly count words. Try this yourself: open a blank document, create a three-bullet list with three words, four words and three words respectively (so that you have ten words against three bullets). Now run the word-count tool. What should you get? 10, of course. What do you get? 13. Yes, silly old Word counts the bullets as words. It also counts the numbers in a numbered list as words. And open hyphens. And full stops on their own.

Students facing the burden of writing an essay of, say, 2000 or 3000 words should take note. It might be easier than you thought.

Translation companies typically charge a set rate per 100 words (about AU\$35). If you have just written a 300-page user guide, you would have written close to 100 000 words. In such a document you might well have keyed 2 000 bullets or numbers, characters that don't need to be translated. That's \$700 per language straight into the translator's pocket if they rely on Word's word count, as just about all of them do. Money for old rope.



# Mindstretchers

Geoffrey Marnell

## Weighing up the trickery involved

Using a common beam balance and any number of identical steel balls, how is it possible for there to be more balls on one side of the balance and yet the balance to be tilted to the *other* side? Try to think of three distinct answers.

## Last puzzle

The puzzle in the last issue of *Words* went like this:

After much pestering by her students for the results of a recent test (where the maximum score was 50), mathematics teacher Gwendolyn Prime decided to relent just a little by telling her impatient class that:

- the product of the best mark and the lowest mark is 1012
- six times the average of the best mark and the lowest mark equals the sum of all the marks together
- half the sum of the second best and third best marks equals twice the difference between the best and third best marks and
- all marks—no two of which are the same—are whole numbers.

“Armed with this information”, she told her charges, “you should be able to work out the marks I have awarded”. Can you?

## Solution

In ascending order, the marks awarded were 22, 23, 24, 25, 30, 34 and 46.

From clues **a** and **d** and given that the maximum score was 50, it follows that the lowest and best marks are either 22 and 46, or 23 and 44. Let  $A$  = the best mark,  $B$  = the second best mark and  $C$  = the third best mark. It follows from clue **c** that  $\frac{1}{2}(B + C) = 2(A - C)$ . Hence given a minimum value for  $C$  of 22 and a corresponding value for  $A$  of 46, the only possible values for  $C$  and  $B$  are {28, 44}, {29, 39} and {30, 34}. Alternatively, given a minimum value for  $C$  of 23 and a corresponding value for  $A$  of 44,  $C$  and  $B$  could only be {23, 41}, {24, 36} or {25, 31}.

Clue **b** gives, now, either 204 or 201 as the sum of all the marks awarded in the test. Hence the sum of the marks still to be allocated is:

- If  $A = 46$  (and the lowest mark is 22): 64, 68 or 72
- If  $A = 44$  (and the lowest mark is 23): 70, 74 or 78

with all such marks being between, but not equal to,  $C$  and the lowest mark. A little trial and error will show that only in the third of the six possibilities above can you find unique whole numbers meeting all the necessary criteria—namely 23, 24 and 25—being numbers enclosed between the lowest mark of 22 and the third highest mark of 30. This gives the next best of 34 and a top mark of 46.

This space has been intentionally left blank

[As if you would have thought otherwise]