

Microsystems - THE document experts
3025 Highland Parkway, Suite 450
Downers Grove, IL 60515
Phone: 630-598-1100
Fax 630-598-9520
<http://www.microsystems.com/>

To Style or Not to Style

The Question You Should Be Asking



Table of Contents

What <i>Word</i> Styles Are.....	3
What Paragraph Styles Consist Of.....	3
Word's Built-In Styles	3
Should We Use Styles?	4
<i>Word</i> is Different Than <i>WordPerfect</i>	5
Is Consistency Important?	5
Styles and Automation	6
Okay – You've Convinced Me – Now How Do We Deploy Styles?	6
Why Not Just Use Word's Built-In Styles?	7
Styles and Templates.....	7
The Challenge: Getting Users to Employ Styles	7
Step One: The Classroom	7
Step Two: The Desktop	9
Step Three: The Converted Documents	9
The Payoff	9

What *Word* Styles Are

Microsoft Word utilizes a formatting feature called *Styles* that essentially collect any combination of formatting attributes and names them, so that users can apply these attributes easily and consistently throughout their documents.

There are two kinds of styles in Word: *Paragraph* styles and *Character* styles. This article will focus exclusively on paragraph styles.

What Paragraph Styles Consist Of

Paragraph styles contain virtually any kind of formatting attributes that can be applied to an entire paragraph. These include:

- Paragraph indents
- Line spacing
- Keep with next
- Alignment
- Font and font attributes
- Outline numbers
- Paragraph spacing
- Borders
- Tab stops

Word's Built-In Styles

Microsoft Word contains a number of built-in, predefined styles. Whether or not you choose to create custom, firm-defined styles, these built-in styles are present in your document and your users will need to understand how they work in order to successfully edit and format their documents. Some of these styles are:

Normal

This is the *default* style in your document, upon which *most other styles are based*. If you start typing in a blank document based on your *normal.dot* template, this is the paragraph style that will be applied.

If you wish to make a *global change* throughout your entire document (e.g., change your default *font*), this is the style

that you want to modify.

Header	<p>When you create or edit a header in Word, the <i>Header</i> style is generally applied. Because this is based on <i>Normal</i> style, if you change the font in Normal, it will change here, too.</p> <p>Modify this style if you wish to: set <i>custom tab stops</i> in your header, change the <i>point size</i> for your font (or change the font entirely), or <i>right-</i> or <i>center-</i>align your text.</p>
Footer	<p>Similar to the “header” style, this is based on Normal but often contains additional attributes.</p> <p>Modify this style to change tab stops, alignment, font point size, etc.</p>
Footnote Text	<p>Also based on Normal, this can be defined to apply: a smaller point size, a first-line or hanging indent, different line spacing, and <i>space after</i> (which will determine how multiple footnotes on the same page are separated from each other).</p> <p><i>WordPerfect</i> had a “Footnote Options” dialog box; <i>Word</i> has a <i>Footnote Text</i> <u>style</u>. It is important to realize that this is where you will determine how your footnotes are formatted.</p>
Footnote Reference	<p>This is the style that determines how your footnote reference appears (e.g., superscripted, point size, etc.).</p>

There are many other built-in styles in Word (*Body Text*, *Block Text*, *Heading 1-9*), but most of those will be applied only if the user *chooses* them.

Should We Use Styles?

As the above chart illustrates, you *will* be using Word styles to some degree or another whether you consciously decide to or not. The question is how will you use them and to what extent?

Many *Word* users migrating from a *WordPerfect* environment (where styles were generally not applied) made the (not unreasonable) assumption that one word processor was pretty much like another. Advanced users quickly grasped that *Word* handled certain features very differently (like linking a lot of formatting to the *paragraph* instead of the entire document, e.g. line spacing and tabs), but once they grasped these differences, they were confident that they could continue to work in *Word* more or less the way they had in *WordPerfect*. This meant, for the most part, *ignoring* styles altogether and using the *Tab* and *Enter* keys to create the kinds of spacing they desired.

Word is Different Than *WordPerfect*

This may seem obvious to many, but it bears constant repeating. Why? Because to be successful in *Word* requires that one *think* like *Word*.

Each Paragraph Mark Contains At Least 30 Commands. Since *Word* stores formatting choices in the paragraph mark, each one contains a whole lot of information, whether or not you intentionally apply it. This includes, alignment, indents, line spacing, widow and orphan control, font and point sizes, tab stops and text flow. If you have 500 paragraphs in your documents, 350 of which are formatted the same way, every one of those paragraph contains every one of those commands.

Styles Store Those Commands In One Place. If all of those 350 paragraphs are styled the same way (e.g., Body Text), then the information pertaining to their formatting is stored *in one place* – the *style definition*. This cuts way down on the system overhead needed to edit, print and render to the document to the screen.

Fighting Word or Making Peace With It. If you choose to use *Word* more or less the way you used *WordPerfect*, you're inviting frustration. If, however, you and your users learn how *Word* "thinks," all of your tasks will be a lot easier.

Is Consistency Important?

Consistency in formatting makes a document easier to edit, easier to read, and more professional in appearance. It also makes the document easier to *troubleshoot* when problems arise (you can start by looking at your *style definitions*, instead of visiting each paragraph to see where the offending code might be).

Styles, by their very nature, encourage and foster consistency not only within individual documents, but across your entire document population.

Consistency also makes editing easier when it comes to making *global changes*. If you decide that all of your section headings should be underlined and centered (instead of left-aligned and bolded), you change the *style definition* and you've changed every heading in your document.

This kind of consistency is particularly useful in *law firms*, where documents have a notoriously long life-span (how much do you want to bet that the document that one of your secretaries is working *this very minute* was copied from another document, which was copied from another document, which was probably first created in *Wang*, *WordStar*, *MultiMate* or even *Mass-11?*).

The Next Conversion

Yes, even as you're fighting your way through *this* conversion, it's not too soon to be thinking about the next conversion. If we've learned anything

over the last few years, it's that the rate of change will only accelerate in the foreseeable future. The *media* through which we will be creating, viewing and manipulating work product will be constantly evolving. Some "conversions" on the horizon: .pdf, html and xml.

Styles are important in this regard because they promote *consistency* and because they minimize the number of locations where formatting information is stored. This makes your documents infinitely more *portable* from one medium to another (if the document needs to be tweaked in the new environment, one only has to visit the *style definitions*, not every paragraph in the document).

Your text documents are a form of electronically managed data, in much the same way as your accounting and billing information. Firms take great care in the way those data are entered. You need to take the same amount of care in the way your text documents are *constructed* – not simply in the way they *print*.

Styles and Automation

Another advantage of using styles in a law firm is that any *automation* that you employ can easily link up with the styles that you've created so you get consistent and successful results. This is much harder to do when automation can't detect a *consistent pattern* in your documents.

The use of styles also allows you to take better advantage of Word's built-in features, such as **Outline View** and **Document Map** (a way of reviewing your documents by *heading* levels).

Okay – You've Convinced Me – Now How Do We Deploy Styles?

Law firms around the country have employed a number of different methodologies for creating, naming and deploying styles. Some take advantage of *Word's* built-in styles (like *Body Text* and *Block Text*), but re-define them for their unique environments. Many firms, however, construct firm-defined styles and employ a *naming convention* that begins with the firm's *initials*, followed by a *descriptive-but-abbreviated* name for the style, followed by a *keyboard alias* for the style which allows the end user to apply the style simply by typing a couple of letters.

Example: The firm wants a "body text" style (*body text* being defined as a left-aligned paragraph with a first-line indent, without any numbering). They want to signify that it's a *firm-defined* style, and they want a keyboard alias for it. They might name the style "*lfBod,b*".

The "*lf*" stands for *law firm* (or the firm's initials), the "Bod" is an abbreviation of "Body Text" and the comma followed by "b" creates a *keyboard alias* that the user can type in the **Style Box** (on the **Formatting** toolbar) in order to quickly access and apply the style.

Putting the firm's initials at the beginning of the style name ensures that all of the firm's styles *will reside together* in Word's style lists.

Why Not Just Use Word's Built-In Styles?

You certainly can, and some firms have done this. However, your end user will have no way of knowing that these styles *are defined the way the firm wants them to be*. Also, including the firm's initials in the style name allows for firm-wide *automation* in deploying style changes or new styles later on. Through *Visual Basic for Applications* you can populate the users' templates with these styles and update them as necessary later on. As long as the naming convention is the same, you can add or update *only those styles that are defined by the firm*.

Styles and Templates

One easy way to get users comfortable with styles and to make sure that they apply them is to incorporate them into whatever templates you create. You can do this in a couple of ways:

Make Sure That Templates' Built-In Text Is Styled. If your template contains captions, headings, dates, etc., style those paragraphs appropriately. This sets a good example for the end-user.

Apply a Style to the Insertion Point. When a user accesses your template, and after any automation is launched, the paragraph where they will begin typing should *already be styled*. This lets the user know how that paragraph should be formatted, and sets up the formatting for certain paragraphs that follow.

The Challenge: Getting Users to Employ Styles

Using styles in your documents can make a lot of things easier: applying multiple levels of formatting, effecting global changes, achieving document- and firm-wide consistency, and troubleshooting problem documents.

Often, however, using styles can seem like a lot of extra work. Instead of hitting the keystrokes your users are familiar with (**Tab**, **Enter**, etc.), they need to access the appropriate style name in order to apply formatting. There are several ways of making this process easier.

Step One: The Classroom

Users need to *understand* what styles are, what they can do, how they can *help* their productivity, and how intrinsic they are to *Word's* logic before they will embrace them. Their initial training is your opportunity for imparting this message.

Demonstration. A *demonstration* of how styles work is usually the most effective method, followed by some kinds of *hands-on* practice where they get to use styles and perform some complex editing tasks. One thing that works particularly well: show them a “section heading” paragraph (non-numbered) created *without* styles.

Walk them through the various steps for: *centering* the paragraph, turning on *bold* or *underline*, accessing the Format | Paragraph dialog box to apply “Keep with next” and “12 points after”. Then, type your next paragraph.

As soon as you hit “Enter”, you’ll have to go *undo* all of that formatting so that you can type a body text paragraph (turn off the underline, the centering, the keep with next, the 12 points after).

Now show them the same thing using a “Section Heading” style that includes all of those attributes *plus* a “Style for Next Paragraph” command that gives you a “Body Text” style as soon as you hit enter. You’ve now performed the same task using styles in about 2 steps instead of 8!

Shortcuts. Demonstrate and let students practice different methods for applying styles. Those can include:

Using the **Style Box** on the *Formatting* toolbar

- Click on the down-arrow on the Style Box
- Scroll until you find the *style* name you wish to apply
- Click on it

Using the **Format** menu

- Click on **Format** menu
- Choose **Style**
- Scroll until you find the *style* name you wish to apply
- Click on the appropriate style name
- Click on **Apply** or press **Enter**

*(You can also type the 1st letter of the style name, and this will move your cursor to that group of styles – you can then use your **arrow** key to find the one that you want.)*

Use **Keyboard Aliases**

- Press **Ctrl+Shift+S**
- **Type** the appropriate keyboard alias (e.g., “b” for *stdBod*, or “sl” for *stdSecL*)
- Press **Enter**

*This selects the name of the currently applied style in the **Style Box** on the toolbar
You can also type the first few letters of a style name, then use your up or down arrow keys to scroll through all of the style names.
This applies the style corresponding to the alias you have typed.*

Note: **Beware:** If you type a keyboard alias that *does not currently exist*, you will be creating a **new style**, which will be defined based upon the formatting that already exists in your paragraph.

You can tell that this has happened because: the *style name* in the Style Box consists *only* of the letters you typed (there is no “*stdStyle,S*” visible) and the formatting on your paragraph *will not change*.

You can fix this just by applying another style.

Step Two: The Desktop

The second place where you can ease this transition is in the *Word* environment you create. You might create a **keyboard shortcut** for the Format | Style dialog box so that a user can:

- Press the **keyboard shortcut**
- Type the **letter** that begins the name of all of your firm styles
- Use your **down-arrow key** to move to the style you wish to apply
- Press **Enter** to apply the style.

Other kinds of desktop automation might involve a special *Styles* menu or keyboard shortcuts assigned to *specific styles* (this option, however, can get complicated: there are a limited number of key combinations available, and users may have difficulty remembering too many of them).

Step Three: The Converted Documents

If you use a conversion utility that applies firm-defined styles *during conversion*, your users will go return from training and find documents that are already formatted in a consistent manner. This makes the job of getting users to embrace and employ styles infinitely easier.

The Payoff

Changing your users document creation and editing habits is definitely a huge undertaking. But when they see how much easier documents are to work with when they’ve been consistently styled, and when you see how much easier documents are to automate and troubleshoot, the payoff becomes clear.

About Microsystems:

Microsystems is a leader in providing complete, “first draft to final delivery” document lifecycle solutions to the life sciences and legal markets and other document-intensive businesses. The company provides more than 650 client firms and companies with an innovative combination of software products, consulting services and training designed to improve the total document creation, quality control and delivery processes. Microsystems is headquartered near Chicago in Downers Grove, IL.

For additional information please call 630.598.1100 or visit www.microsystems.com