CS 200

Lecture 02
Structured Word Processing — Styles

Miscellaneous Notes

Assigned reading

• “The Non–Designer’s Design Book,” by Robin Williams (first half)
• “Adobe on Basic Typography” (https://www.student.cs.uwaterloo.ca/~cs200/Adobe/main.html)

Please read and highlight before this week’s lab

• Assignment 2 (due Monday Jan 21 at 11:59 pm)
• Notes for this lecture

Today and Thursday

• [named] styles
• the Poke pearl
• backups
The Poke Pearl

What happens if you
• double-click here?
• triple-click here?
• option-click here?
• control-click here?
• command-click here?
• etc...

What happens if you
• click and drag on this?
• option-click and drag on this?
• control-click and drag on this?
• command-click and drag on this?
• etc...

Assumptions

You have used a word processor before

You understand and correctly use the following WP tools
• Word Wrap (hard carriage return vs. soft carriage return)
• Cut/Copy and Paste
• Font usage (size, style and typeface)
• Rulers and Margins
• Proper use of tabs
• Headers and footers (page numbering)
• Footnotes
Things to Think About

- What are the data objects in a Word Processor?
- Is there more than one way to do any given task?
- What are deficiencies of the interface?
- What are efficiencies of the interface?

[Named] Styles (1)

The idea

- name a collection of paragraph or character attributes
- you then apply the NAME to text instead of individually setting attributes

Hence changing a document's appearance is

- easier & faster, more accurate (⇒consistency, if used intelligently)
- — you just change the definition of the name; everywhere it's used, text changes

Does appearance matter? → Yes!

- legibility
- understandability
- professionalism
Named Styles (2)

Terminology

- **Attributes**: Helvetica, Palatino, bold, italic, underlined, red, 15 pt leading, etc. are attributes
- **[Named Style]**: A *(named) style* is a (specific) “bundle of attributes”
- We usually specify that it is a “named style” for emphasis/clarity
  - ex. “Heading” could be a named style with attributes “24 pt Myriad Pro Light with 12 pts of Before Paragraph leading & 2 cm of left indent”

Compare this and the preceding slide

Terminology Attributes: Helvetica, Palatino, bold, italic, underlined, red, 15 pt leading, etc. are attributes [Named Style]: A (named) style is a (specific) “bundle of attributes” We usually specify that it is a “named style” for emphasis/clarity ex. “Heading” could be a named style with attributes “24 pt Myriad Pro Light with 12 pts of Before Paragraph leading & 2 cm of left indent”
Indirection

The underlying concept is indirection

• referring to something through something else

Indirection is useful in many contexts, including...

• drawing programs (object properties)
• HTML (tags in web pages; CSS)
• electronic mail (aliases, forwards)
• spreadsheets (named cells/ranges)

Remember the Model pearl?

The text being formatted

The style definitions (built-in & user-defined)

Block

Block_Centered
Font: (Default) Times, 12 pt, English (US), Centered, Line spacing: single, Widow/Orphan control, No bullets or numbering

Default Paragraph Font
The font of the underlying paragraph style +

Emphasis_Slight
Default Paragraph Font + Font: italic

Figure_Caption
Font: (Default) Times, 10 pt, English (US), Indent: Left: 0.39", Right: 0.39", Justified, Line spacing: exactly 12 pt, Space Before: 6 pt, Widow/Orphan control

Footer
Font: (Default) Times, 10 pt, English (US), Left, Line spacing: single, Widow/Orphan control, Tabs: 3.25", Centered + 6.5", Right

Footnote Reference
Default Paragraph Font + Superscript

Footnote Text
Font: (Default) Times, 10 pt, English (US), Justified, Line spacing: exactly 10 pt, Space Before: 4 pt, Widow/Orphan control

Header
Font: (Default) Times, 10 pt, English (US), Left, Line spacing: single, Widow/Orphan control, Tabs: 3.25", Centered + 6.5", Right

Heading 1
Style for Next Paragraph: Block, Font: (Default) Times, 12 pt, Bold, English (US), Centered, Line spacing: single, Space Before: 14 pt, Widow/Orphan control, Keep with next, Level 1

Heading 2
Style for Next Paragraph: Block, Font: (Default) Times, 12 pt, Bold, English (US), Left, Line spacing: single, Space Before: 10 pt, Widow/Orphan control, Keep with next, Level 2

Instruction
Style for Next Paragraph: Instruction_Expl, Font: (Default) Courier, 11 pt, English (US), Indent: Left: 0.2", Left, Line spacing: single, Space Before: 6 pt, Widow/Orphan control, Keep with next
**Naming Named Styles**

Style names should reflect function, not appearance

- “Quotation” not “Indented Paragraph”
- “Emphasis” not “Bold”
- “List Item” not “Bulleted Paragraph”

Style names should be chosen **logically** — **NOT** based on their appearance

so if you change the appearance, you don’t have to change the name, and you avoid confusion

---

**Showing Style Usage in MS Word**

*Word > Preferences... to bring up*

*The text being formatted*

In Draft view, the “style area width” controls the width of the style name column in the doc display; if it’s zero, the style name column vanishes.

In Draft view, *View > Reveal Formatting* to enable the (yellow) Formatting popup
Hierarchical Styles

The idea

- style B is “everything style A is, except for . . .”
- style C is “everything style B is, except for . . .”
- etc.

- **Style A**
  - Helvetica, 24 pt, Bold

- **Style B**
  - Helvetica, 22 pt, Bold

- **Style D**
  - Times, 22 pt, Bold

- **Style C**
  - Helvetica, 24 pt, Italics

- **Style E**
  - Helvetica, 26 pt, Italics
Hierarchical Styles cont’d

When you change an attribute of A’s definition

- styles based on A change, too, unless that attribute has been explicitly set for the derived style

Hierarchical Styles Relativity

There are choices in how to implement some aspects of this; eg, if I change B’s font size, is the new value

- absolute (MS Word), or relative to the size of A?
- does changing A’s size later automatically change B's?
- by a fixed number of points, or by a percentage?

Ditto with paragraph indents

Etc.
Hierarchical Styles — Comments

Comments

• this is a model of named styles, though with parameters
• there’s no point to this unless parent and child styles:
  • share some attributes
  • but not others
• you might have a forest instead of a tree
  (our previous example had no hierarchy at all — a forest of flat trees, so to speak)

In the “Files for Styles Lectures” under Week 2: Styles on Learn, compare

• User Manual Extract (Tree)
• User Manual Extract (Forest)
A Paragraph Style Sheet for UME.doc Using Multiple Trees

Built-in styles
(The names are built-in, however the

<table>
<thead>
<tr>
<th>Default Paragraph Font</th>
<th>The font of the underlying paragraph style +</th>
</tr>
</thead>
<tbody>
<tr>
<td>Footer</td>
<td>Normal = Font:10 pt, Right: 0.25&quot;, Tabs: 3.25&quot;, Centered + 6.5&quot;, Right</td>
</tr>
<tr>
<td>Footnote Reference</td>
<td>Default Paragraph Font + Superscript</td>
</tr>
<tr>
<td>Footnote Text</td>
<td>Normal = Font:10 pt, Justified, Line spacing: exactly 10 pt, Space Before: 4 pt</td>
</tr>
<tr>
<td>Header</td>
<td>Normal = Tabs: 3&quot;, Centered + 6&quot;, Right</td>
</tr>
<tr>
<td>Heading 1</td>
<td>Style for Next Paragraph: Block</td>
</tr>
<tr>
<td></td>
<td>Heading_Abstract + Font:Bold, Centered, Space Before: 14 pt, Keep with next, Level 1</td>
</tr>
<tr>
<td>Heading 2</td>
<td>Style for Next Paragraph: Block</td>
</tr>
<tr>
<td></td>
<td>Heading_Abstract + Font:Bold, Space Before: 7 pt, Keep with next, Level 2</td>
</tr>
<tr>
<td>No List</td>
<td>No List +</td>
</tr>
<tr>
<td>Normal</td>
<td>Font (Default) Times New Roman, 12 pt, English (US), Left, Line spacing: single, Space Before: 3 pt, Widow/Orphan control</td>
</tr>
<tr>
<td>Page Number</td>
<td>Default Paragraph Font +</td>
</tr>
<tr>
<td>Table Normal</td>
<td>Font (Default) Times New Roman, 10 pt, Left, Line spacing: single, Widow/Orphan control</td>
</tr>
</tbody>
</table>

User-defined styles

| Block                   | Body_Abstract + Justified, Space Before: 6 pt |
| Block_Centered          | Block + Centered, Numbered |
| Body_Abstract           | Normal + |
| Emphasis_Slight         | Default Paragraph Font + Font:italic |
| Figure_Caption          | Body_Abstract = Font:10 pt, Indent: Left: 0.39", Right: 0.39", Justified, Line spacing: exactly 12 pt, Space Before: 6 pt |
| Heading_Abstract        | Normal + |
| Instruction             | Style for Next Paragraph: Instruction_Expl |
|                        | Body_Abstract + Font:Courier, 11 pt, Indent: Left: 0.2", Space Before: 6 pt, Keep with next |
| Instruction_Expl        | Body_Abstract = Indent: Left: 0.44", Numbered |
| Instruction_Item        | Style for Next Paragraph: Instruction_Expl |
|                        | Body_Abstract + Font:10 pt, Indent: Left: 0.63", Hanging: 0.19", Line spacing: exactly 12 pt |
| Item                    | Body_Abstract + Indent: Left: 0.19", Hanging: 0.19", Space Before: 6 pt |
| Menuitem_Expl           | Body_Abstract + Indent: Left: 0.44" |
| Operand                 | Body_Abstract = Indent: Left: 0.25" |
| Operand_Expl            | Body_Abstract = Indent: Left: 0.56", Space Before: 6 pt |

Demo 2!

Character Styles

Should character attributes be part of a paragraph style definition?

Are character styles hierarchical?
- yes in MS Word
- no in FrameMaker
- yes in Nisus Writer Express & Nisus Writer Pro
- yes in Adobe InDesign

Are character styles used to specify the (default) character attributes of paragraph styles?
- no in MS Word
- optionally in Nisus Writer Express & Pro

Incidentally, if you own a Mac, Nisus Writer Pro is a very nice $39 US (ed) word processor (www.nisus.com).
Tables-of-Contents (TOCs)

What’s our model of a TOC? (example on the next slide)

Steps to create a table-of-contents

• identify paragraph styles from which to build the TOC
  • ex. Heading 1, Heading 2, Heading 3, etc, in Word
• specify a TOC style for each TOC level
  • TOC 1, TOC 2, TOC 3, etc, in Word
• build the TOC (most word processors do this for you)
  • copy paragraphs with specified paragraph style tags into the TOC
  • apply the corresponding TOC style to each
    • *Heading 1 to TOC 1, Heading 2 to TOC 2, Heading 3 to TOC 3, etc.*
  • append a tab character & page number to each
• modify the TOC styles to get the desired layout
• rebuild the TOC whenever appropriate

Note the application of two distinct styles to each piece of text

*Demo 3!*
CS 200 Winter 2019

---

**Menu Commands in the Machine Emulator (GEM):**

GEM initially displays two windows.

- The "cpu window" displays the contents of the CPU registers. On Unix and Windows machines, it contains a menu bar from which the emulator can be controlled. (On Macintoshes, these menus are appended to the menu bar at the top of the screen.)
- The "memory window" shows a selectable display of the contents of memory.

**File Menu (ROM35 Menu on the Macintosh):**

- **Load**: Use this command to load a relocatable object module or memory. (Each "new file" is created by the assembler and when it translates an assembly program into machine language.)
- **Reload**: Use this command to reload the contents of the last previous relocatable object module loaded. This provides a convenient, minimum-effort way to re-execute a program.

**Quit**: Terminate execution of the emulator.

**Execute Menu:**

- **Step**: Execute a single instruction and stop. Clicking on the cpu window’s "step button" is equivalent to selecting this menu item.
  
  When you execute an instruction, certain mnemonic labels are displayed in the CPU and memory windows to help you keep track of what’s happening:
  - The Guru (G) labels the "previous" and "next" instructions. The previous instruction in the instruction set is current. The current instruction is the instruction that will be executed next, that is, the instruction whose address is in the program counter. These labels are printed shall when you have just executed a jump instruction.
  - The Guru (L) labels any register or memory word that was selected as a source operand for the instruction just executed.
  - The data (L) labels the register or memory word, if any, into which a result value was stored by the instruction just executed.
  - The Guru (W) labels a register of the instruction just executed referenced as operand indirectly through a.

- **Run**: Execute program instructions continuously, one right after the other. Clicking on the cpu window’s "Run button" is equivalent to selecting this menu item.

---

**Table of Contents**

<table>
<thead>
<tr>
<th>Menu Commands in the Machine Emulator (GEM)</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Menu (ROM Menu on the Macintosh)</td>
<td>2</td>
</tr>
<tr>
<td>Execute Menu</td>
<td>3</td>
</tr>
<tr>
<td>Display Menu</td>
<td>4</td>
</tr>
<tr>
<td>GEM’s Architecture</td>
<td>5</td>
</tr>
<tr>
<td>Program Counter</td>
<td>6</td>
</tr>
<tr>
<td>Instruction Register</td>
<td>7</td>
</tr>
<tr>
<td>Memory</td>
<td>8</td>
</tr>
<tr>
<td>CPU Registers</td>
<td>9</td>
</tr>
<tr>
<td>Condition Code Register</td>
<td>10</td>
</tr>
<tr>
<td>The Instruction Set and Assembly Syntax</td>
<td>11</td>
</tr>
<tr>
<td>Data Movement</td>
<td>12</td>
</tr>
<tr>
<td>Arithmetic / Logical Instructions</td>
<td>13</td>
</tr>
<tr>
<td>Flow-of-Control</td>
<td>14</td>
</tr>
<tr>
<td>Pseudo-Instructions</td>
<td>15</td>
</tr>
</tbody>
</table>

**The Trap Instruction**

<table>
<thead>
<tr>
<th>Trap #</th>
<th>Description</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trap #1</td>
<td>Reading an integer</td>
<td>11</td>
</tr>
<tr>
<td>Trap #2</td>
<td>Write an integer to the display and to the display, base 10</td>
<td>11</td>
</tr>
<tr>
<td>Trap #3</td>
<td>Write an integer to the display and to the display, base 10</td>
<td>11</td>
</tr>
<tr>
<td>Trap #4</td>
<td>Write an integer to the display and to the display, in some radix</td>
<td>11</td>
</tr>
<tr>
<td>Trap #5</td>
<td>Write an integer to the display, in some radix</td>
<td>12</td>
</tr>
<tr>
<td>Trap #6</td>
<td>Read a string</td>
<td>12</td>
</tr>
<tr>
<td>Trap #7</td>
<td>Write a string to the display and to the listing</td>
<td>12</td>
</tr>
<tr>
<td>Trap #8</td>
<td>Write a string to the display and to the listing</td>
<td>12</td>
</tr>
<tr>
<td>Comments</td>
<td></td>
<td>13</td>
</tr>
</tbody>
</table>
Conditional Text

Display or not display
Print or not print

• in MS Word these are separate attributes
• is that a good idea?

Look up “Hidden Text” in Word’s help index

Word only implements one class of conditional

• but it makes sense to have more
• you can simulate having more than one class of conditional text using named styles

The Format > Style… Dialog in Word

Format → Style…

Interaction techniques

A preview checkbox would be better. & everywhere—not just here.
The Format > Style... > Modify Style sub-dialog

Recall the Model Pearl (interaction)

- Nested dialog boxes
  + radio buttons, check boxes, etc.
- Which is better? Should you have both?

Use this dropdown menu to change more attributes.

The Format > Style... > Organizer sub-dialog
Word’s Format > Paragraph dialog box

Not all paragraph attributes are available in this dialog
ex. bullets, borders...
although they belong here

Widows and Orphans

Widow
stranded line at top of page

Orphan
stranded line at bottom of page
Word’s Format > Font... (ie. Character) dialog box

Style Templates

Store style definitions in a separate “template” or “style sheet” file

Use templates to keep styles consistent across multiple documents

- each such document is linked to the template
- when the template is changed, the appearance of every linked document
  - changes automatically? (optional in MS Word)
  - or do you have to request an update? (Keynote)

Another data model...
Style Templates in MS Word

Every document is based on a template (“Normal” by default)
  • apparently in the “~ / Documents / Microsoft User Data / “ folder.
    (instead of “~ / Library / Application Support / Microsoft… sigh)
  • whose styles are copied to the document

To use a different base template
  • use the “Attach” button in the Tools → Templates and Add-ins…
    dialog box

To have Word reload the template’s styles every time you
open the document
  • check “Automatically Update Document Styles”
    in the Tools → Templates and Add-ins… dialog box

To cause a style change applied in a document to update
its template definition
  • Use the “Add to template” button in the Modify Styles
    dialog box, or change it directly in the template

“Invisible” (aka “non-printing”) characters

See Word’s Preferences > View dialog

Or, pres: " from Word’s Home menu
Things That May Confuse You in MS Word

Character attributes in paragraph styles
- are a matter of convenience
- when done, one set of character attributes is a property of the paragraph as a whole

The Style Column
- is visible only in Outline or Draft View, not Web Layout or Print Layout

Some paragraph attributes
- are in “auxiliary” dialog boxes instead of the paragraph dialog box even though they are paragraph attributes
- ex. borders, bullets, numbering, shading

Paragraph attributes
- are “stored in” the ¶ at the end of each paragraph
- are automatically copied to a new paragraph if you press RETURN
- you can copy/paste this character to transfer its attributes to another paragraph

The Case For Styles

It's easier / quicker to change
- the appearance of an existing document
- determine the appearance of a conforming document

They make it easier to achieve consistent appearance
- within a document
- using templates, across documents

You can switch media much more easily
The Case Against

It takes longer to get started

Application Interface and Design

In well-designed applications you can do everything via
  • menu items
  • dialog boxes opened by a menu item
  • a toolbar opened from a menu item
    — Why?

Often there are other ways of doing things
  • typically faster but more obscure
    — Why?

Are there other ways to
  • **Define** named styles in Word?
  • **Apply** named styles In Word?

Explore the Styles pull-down in the Formatting toolbar
  • (The Poke pearl)
Styles & Style Templates (aka “Style Sheets”) Elsewhere

These ideas are applicable anywhere you have objects with attributes, though the term may not be used

- graphics applications
- printing
- web pages

So look for them!

Next Week

Pixel Graphics

Have a digital photo ready for lab next week.