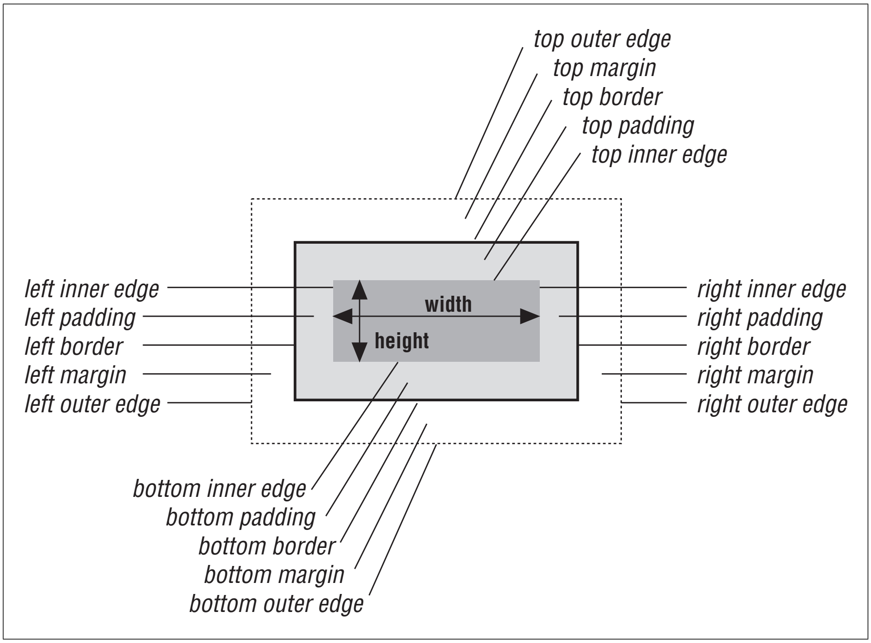

Figure 19-1. The box model

applies the `width` property to the entire width of the element box, from outer edge to outer edge. When margins, borders, or padding are also applied, this results in potentially large discrepancies between how the element should be sized and how it will appear in IE/Windows.

Inline Boxes

The element box is not the only implied box in the CSS visual formatting model. Every character and element in a line of text also generates a box on the fly. These *inline boxes* are used by the user agent (the browser) behind the scenes to calculate the height of each line in the flow of text and the space around elements. Line boxes and inline boxes are not elements, they are merely a device of the visual layout model. Therefore, you cannot use a selector to target line or inline boxes and apply styles to them (you can apply styles to inline elements, of course).

Having a familiarity with the various inline boxes at work behind the scenes is useful for predicting and controlling line height as well as for specifying the vertical alignment of inline elements. They also come into play when specifying box properties, particularly to inline elements. Figure 19-2 highlights the various inline boxes for a line of text.

The four invisible boxes that the user agent keeps track of when formatting each line of text include:

Em box
In CSS, the this is a square unit that is equal to the `font-size` of the element. Its relation to the actual characters in the font is dependent on the typeface design, but in general, it encloses the ascenders and descenders of the font.

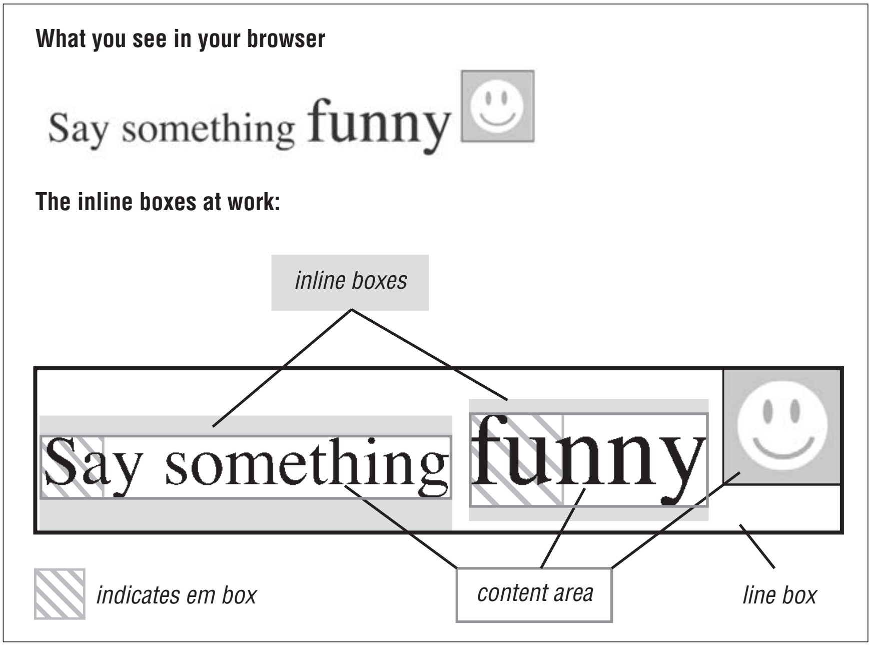

Figure 19-2. Inline boxes

Content area
: Every element in a line has a content area box that corresponds to the content area in the box model (Figure 19-1). For text elements (also called *non-replaced elements* because their content appears in the source), the height of the content area is determined by the element's `font-size`. For anonymous text (text not specifically contained within an inline element), the `font-size` is inherited from the parent element. For images (and other replaced elements), the content area is the width and height of the image in pixels.

Inline box
: The height of the inline box is calculated as the total of the element's content area plus the *leading* added above and below it (see Figure 18-12 in Chapter 18). Leading is the difference between the element's `font-size` and `line-height` values. It may be a negative value, which means that lines may overlap. For images (and replaced elements), the inline box is the height of the image in pixels, plus the height of any added borders and margins on the `img` element.

Line box
: The line box is drawn around the top of the highest inline box and the bottom of the lowest inline box. It represents the total required vertical space for the line and all its elements. The `vertical-align` values `top` and `bottom` are relative to the top and bottom edges of the line box.

In the sections that follow, we'll see how the line box (the total height of a line) is affected (or not affected) by the addition of margins, borders, and padding on inline elements.