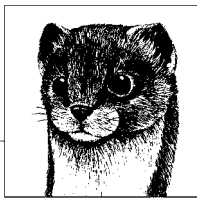

Document Structure

Before marking up your actual content, it is necessary to establish the proper global structure of the (X)HTML document itself. An (X)HTML document is composed of three parts: a *declaration* of the HTML or XHTML version used, a *header* containing information about the document, and the *body* containing the document's content.* This chapter takes a look at each of these components and, in doing so, introduces these elements used for establishing the global structure of the document:

`html`	Root element of an (X)HTML document
`head`	Header
`body`	The body of the document
`title`	Document title
`meta`	Meta data (information about the document)

If you use a professional web authoring tool to create web pages, chances are you're accustomed to the minimal document structural markup inserted for you when you select "New File." This chapter will give you the tools necessary to peek under the hood and decide if the automatically generated declarations accurately represent the mode in which you intend to author.

Minimal Document Structure

This markup sample shows the structure of a minimal XHTML document as specified in the XHTML 1.0 Recommendation. It provides important context to upcoming discussions of global document structure.

* Not all documents have a body. Framed documents are composed of a declaration, header, and a frameset that establishes the number and structure of its frames. Framed documents are discussed in Chapter 14.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

  <head>
    <title>Document Title</title>
  </head>

  <body>
    <p>Content of document...</p>
  </body>

</html>
```

This example begins with an *XML declaration* that identifies the version of XML and the character encoding of the document. XML declarations are encouraged for XHTML documents; however, they are not required when the character encoding is the UTF-8 default as in the above example. Because XML declarations are problematic for current browsers as of this writing, even those that are standards-compliant, they are generally omitted.

Now, take a closer look at the four major components of XHTML (and HTML) documents.

Document type declaration

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

The document type (DOCTYPE) declaration tells the browser which DTD to use to parse the document. This example specifies XHTML Strict. If this example were an HTML document, it would use one of the HTML DTDs. The upcoming "Document Type Declaration" section provides more information on the DTD options and uses for this information.

Root element

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">...
     </html>
```

`html` is the root element for all HTML and XHTML documents. The `html` element and its declarative attributes shown here are discussed in the upcoming section, "The Root Element."

Document header

```
<head>
   <title>Document Title</title>
</head>
```

The `head` element, or header, contains information about the document that is not considered part of the document content. The header must include a descriptive `title` in order to validate. Document headers are covered in more detail later in this chapter.

Document body

```
<body>
    Content of Document...
</body>
```

The `body` element contains all of the content of the document—the part that displays in the browser window or is spoken in a speech browser. The `body` of an (X)HTML document might consist of just a few paragraphs of text, a single image, or a complex combination of text, images, tables, and multimedia objects. What you put on the page is up to you.

Document Type Declaration

To be valid, an (X)HTML document must begin with a document type declaration that identifies which version of HTML or XHTML is used in the document. This is done using a DOCTYPE declaration that names the *document type definition* (DTD) for the document. A DTD is a text document that lists all the elements, attributes, and rules of use for a particular markup language. See Chapter 7 for more information on DTDs.

The inclusion of a document type declaration has always been a requirement of valid HTML documents. With no DOCTYPE declaration, there is no set of rules to validate against. In the years of fast and loose HTML authoring, the DOCTYPE declaration was commonly omitted. However, now that standards compliance is a priority in the web development community, and because there are so many DTDs to choose from, authors are strongly urged to include the DTD declaration and validate their documents. The DOCTYPE declaration (or its omission) also triggers different browser behaviors, as discussed in the upcoming "DOCTYPE Switching" section.

DTD Options

HTML 4.01 and XHTML 1.0 offer three DTD versions:

- Strict
- Transitional
- Frameset

XHTML 1.1 has only one DTD. The DTD documents live on the W3C server at a stable URL.

The `<!DOCTYPE>` (document type) declaration contains two methods for pointing to DTD information: one is a publicly recognized document identifier; the other is a specific URL in case the browsing device does not recognize the public identifier. Descriptions and specific markup for each HTML and XHTML version are listed here.

HTML 4.01 Strict

The Strict DTD omits all deprecated elements and attributes. If you are authoring according to the strict DTD, use this document type definition:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
 "http://www.w3.org/TR/HTML4.01/strict.dtd">
```

HTML 4.01 Transitional

The Transitional DTD includes everything from the Strict DTD, plus all deprecated elements and attributes. If your document includes some deprecated elements or attributes, point to the Transitional DTD using this DOCTYPE declaration:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
 "http://www.w3.org/TR/HTML4.01/loose.dtd">
```

HTML 4.01 Frameset

If your document contains frames—that is, it uses `frameset` instead of `body` for its content—then identify the Frameset DTD. The Frameset DTD is the same as the Transitional version (it includes deprecated yet supported elements and attributes), with the addition of frame-specific elements. The content-containing HTML documents that are displayed within the frames do not need to use the Frameset DTD.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
 "http://www.w3.org/TR/HTML4.01/frameset.dtd">
```

XHTML 1.0 Strict

The same as HTML 4.01 Strict, but reformulated according to the syntax rules of XML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

XHTML 1.0 Transitional

The same as HTML 4.01 Transitional, but reformulated according to the syntax rules of XML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

XHTML 1.0 Frameset

The same as HTML 4.01 Frameset, but reformulated according to the syntax rules of XML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

XHTML 1.1

There is only one DTD for XHTML 1.1. It omits every deprecated element and attribute. It differs from XHTML 1.0 Strict in these ways:

- The `lang` attribute has been replaced with the `xml:lang` attribute.
- The `name` attribute for the `a` and `map` elements has been replaced with `id`.
- A *ruby* collection of elements has been added. The W3C defines ruby as "short runs of text alongside the base text, typically used in East Asian documents to indicate pronunciation or to provide a short annotation."

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
 "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```