University of Southampton

Hypertext Markup Language

COMP3227 Web Architecture & Hypertext Technologies

Dr Heather Packer – hp3@ecs.soton.ac.uk

What is HTML?

1. The data format for Web pages

2. A markup language for structured documents

3. A platform for Web application development

4. All of the above

Which HTML?

- HTML Tags
- HTML+
- HTML 2.0
- HTML 3.0
- HTML 3.2
- HTML 4
- HTML 4.01

- XHTML
- XHTML 2
- HTML 5
- HTML 5.1
- HTML 5.1 2nd ed.
- HTML 5.2
- HTML 5.3
- …

HTML source

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Document</title>
  </head>
  <body>
    <h1>This is an example HTML document</h1>
    <ol>
      <li>First paragraph</li>
      <li>Second paragraph</li>
    </ol>
    <p>This is the first paragraph. <em>It doesn't really say
      very much.</em> Now is the time for all good men to come
      to the aid of the party.</p>
    <p>This is the second paragraph. The quick brown
      <a href="http://example.org/">fox</a> jumped over the
      lazy dog.</p>
  </body>
</html>
```

HTML markup basics

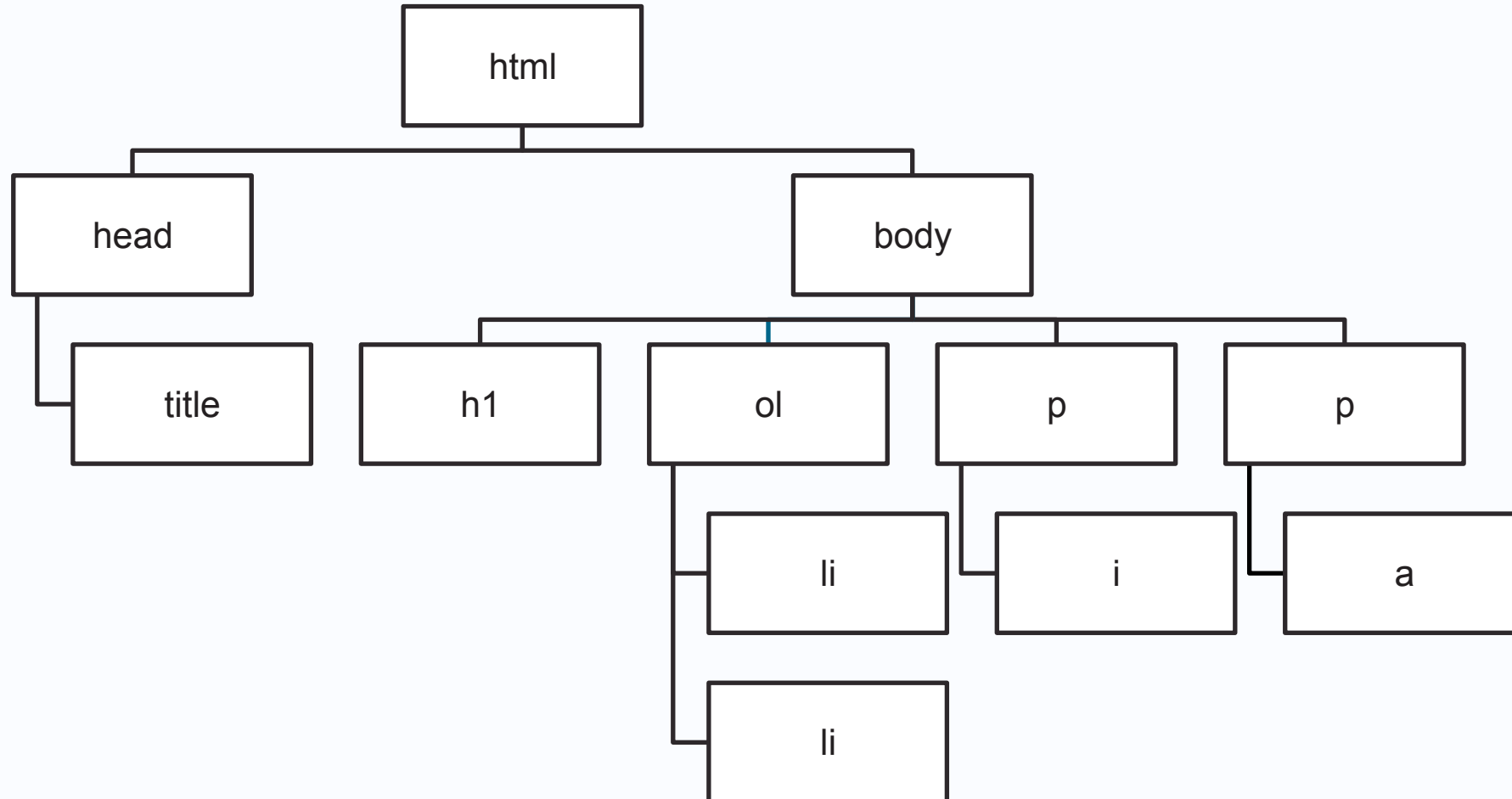Tags that structure a document as a hierarchy of elements

    <p> . . . </p>

Attributes that modify elements

    <a **href="..."**>...

Entity references that represent character sequences

    R**&**D

Element structure

Evolution of HTML (1991-1995)

Initially a language for encoding simple document semantics

```
<h1>Things to See in London</h1>
<p>The <i>Millennium Dome</i> is great.</p>
```
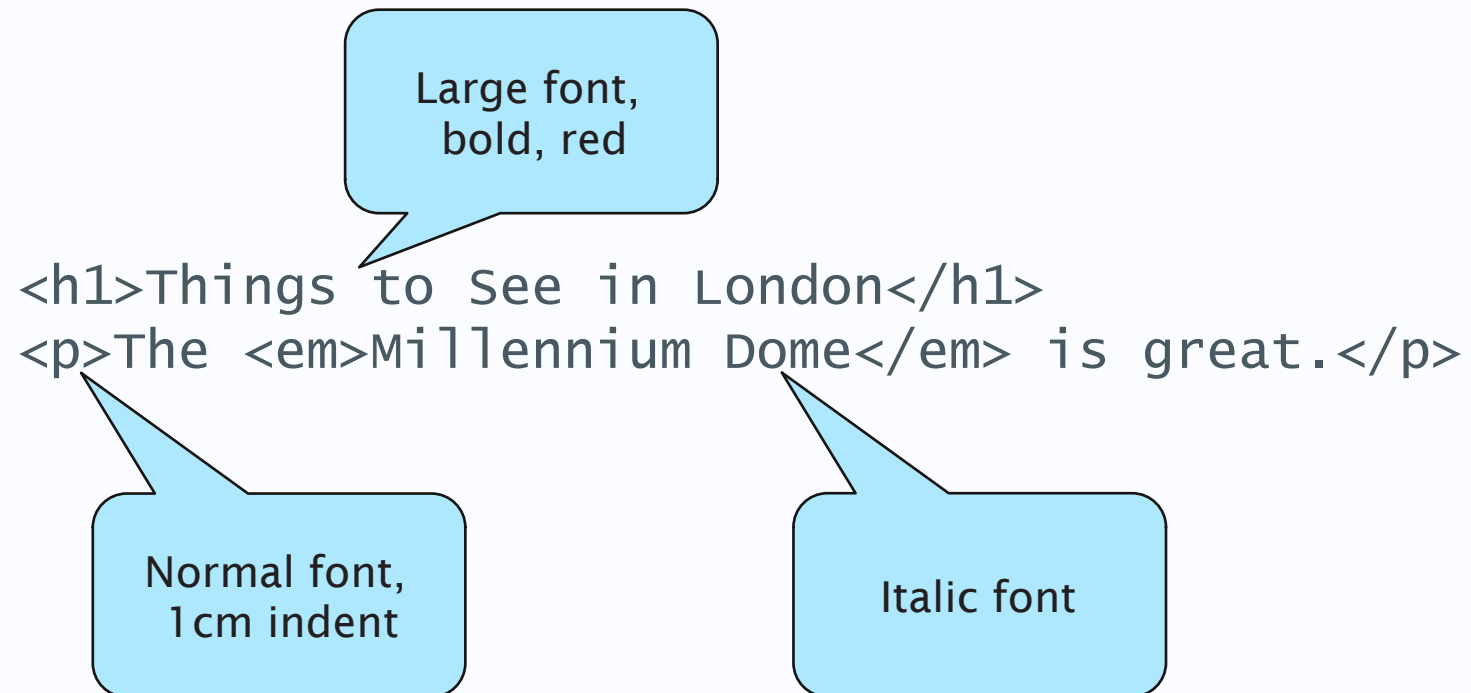
Evolution of HTML (1996-2000)

Subsequently extended to allow more design precision and visual effects

```
<h1><center>Things to See in London</center></h1>
<p><font size="3" text="#FF0000">The <i>Millennium Dome</i> is
<blink>great</blink>.</font></p>
```

Evolution of HTML (2000-)

Separation of semantics from presentation using stylesheets

Large font, bold, red

```
<h1>Things to See in London</h1>
<p>The <em>Millennium Dome</em> is great.</p>
```

Normal font, 1cm indent

Italic font

Further reading

Raggett, D. (1998) *Raggett on HTML 4*. Boston, MA: Addison-Wesley. Chapter 2.

https://www.w3.org/People/Raggett/book4/ch02.html

HTML5

WHATWG and W3C

World Wide Web Consortium (W3C)

- 1994

- Businesses, nonprofit organizations, universities, governmental entities, and individuals

- Formed to standardise the technologies used on the web

- 2002-2006 XHTML 2.0

- Adopted HTML5 2007

- Recommendation HTML5 2014

Web Hypertext Application Technology Working Group

- 2004

- Apple Inc., Mozilla Foundation and Opera Software, leading Web browser vendors

- Formed in response to the slow development of World Wide Web Consortium (W3C) Web standards

- Against the use of XML-based technologies

  – A platform for dynamic web applications

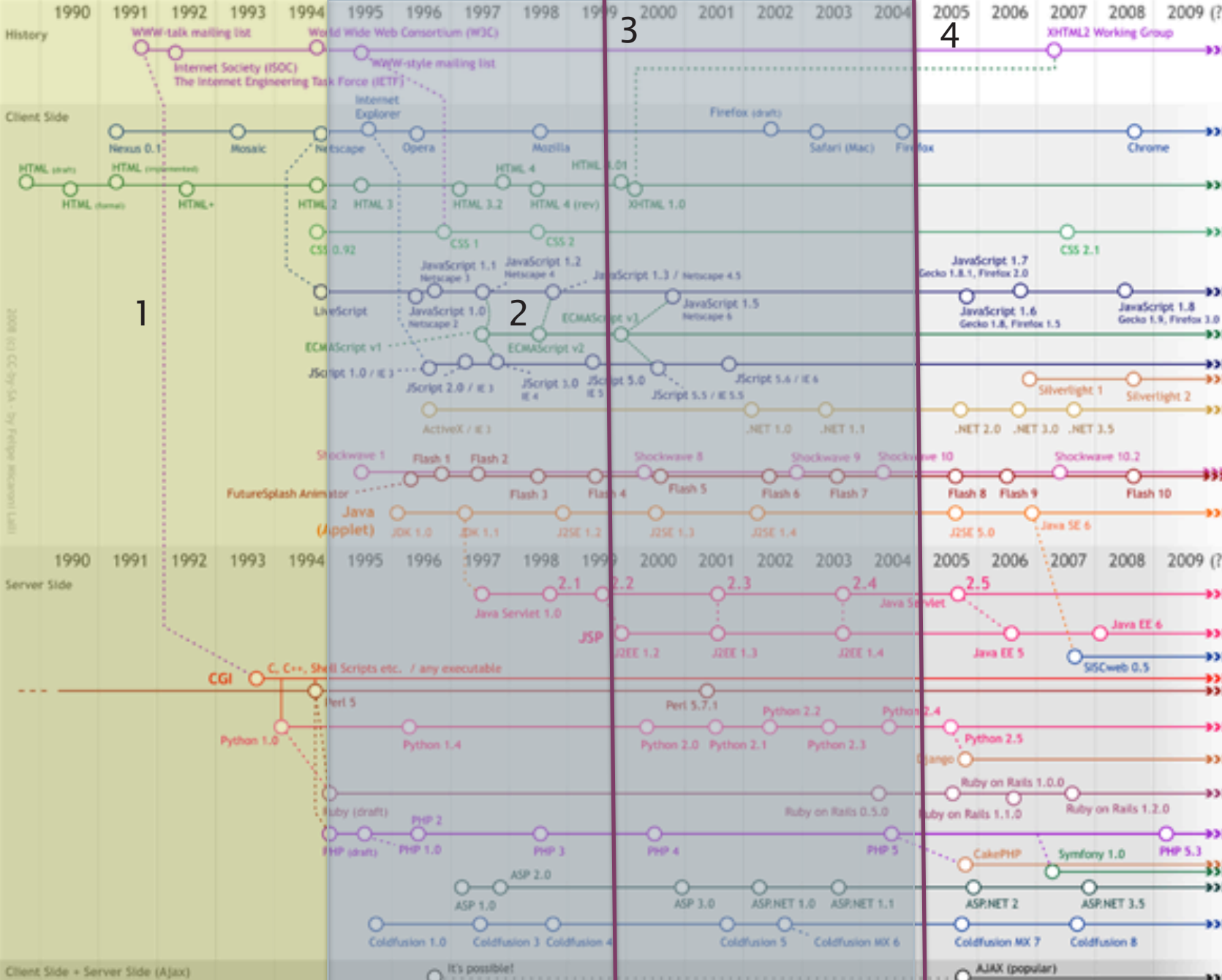 – Too document-centric, and not suitable for forums or shops

Why HTML5?

Usage of HTML in the wild had changed

- – Greater emphasis on JavaScript-based web applications

Web Technologies

1. Document centric
2. Move towards Web Application
3. AJAX and XMLHTTPRequest
4. Formation of WHATWG

Early JavaScript based Web Applications

XMLHttpRequest (XHR)

- API

- Transfer data between a web browser and a web server

- Can continually modify a loaded web page

AJAX (Asynchronous JavaScript and XML)

- Set of tools

- Send and retrieve data from a server asynchronously

- Modify data without reloading the page

- Used HTML, XMLHttpRequest, JavaScript, DOM, and more.

Why HTML5?

Usage of HTML in the wild had changed

- Greater emphasis on JavaScript-based web applications
  (AJAX, XMLHttpRequest, etc)

- Concerns about widespread use of Adobe Flash for interactivity
  (openness, reliability, security, performance)

Thoughts on Flash – April 2010

- "Thoughts on Flash" is an open letter published by Steve Jobs

- The letter criticizes Adobe Systems' Flash platform

  – Rapid energy consumption, computer crashes, poor performance on mobile devices, abysmal security, and lack of touch support

  – Flash is a closed system

  – Flash is no longer necessary to watch video or consume any kind of web content

- The letter drew accusations of falsehood and an ulterior motive.

- Retrospectively many publications agreed with Jobs

Flash is no longer necessary…

Type of content	HTML 1.2	HTML 4.01	HTML5	Purpose
Heading	Yes	Yes	Yes	Organise page content by adding headings and subheadings
Paragraph	Yes	Yes	Yes	Identify paragraphs of text
Address	Yes	Yes	Yes	Identify a block of text containing contact information
Anchor	Yes	Yes	Yes	Link to other web content
List	Yes	Yes	Yes	Organise items into a list
Image	Yes	Yes	Yes	Embed a photograph or drawing into a web page
Table	No	Yes	Yes	Organise data into rows and columns
Style	No	Yes	Yes	Add CSS to control how objects are presented
Script	No	Yes	Yes	Add JS to make pages respond to user behaviors
Audio	No	No	Yes	Add audio to a web page with a single tag
Video	No	No	Yes	Add video to a web page with a single tag
Canvas	No	No	Yes	Add an invisible drawing pad to a web page, using JS

University of Southampton

Why HTML5?

Usage of HTML in the wild had changed

- – Greater emphasis on JavaScript-based web applications
  (AJAX, `XMLHttpRequest`, etc)

- – Concerns about widespread use of Adobe Flash for interactivity
  (openness, reliability, security, performance)

- – Concerns about consistency of handling invalid markup
  (browser "quirks modes")

Modes

- Before W3C standardisation pages were typically written for:
  - Netscape Navigator
  - Microsoft Internet Explorer
- Quirks mode is a technique used by some web browsers to maintaining backward compatibility
- Three modes used by the layout engines in web browsers:
  - **quirks mode** - emulates nonstandard behavior in Navigator and Internet Explorer
  - **almost standards mode** - are only a very small number of quirks implemented.
  - **full standards mode** - behavior described by the HTML and CSS specifications

Why HTML5?

Usage of HTML in the wild had changed

- Greater emphasis on JavaScript-based web applications
  (AJAX, `XMLHttpRequest`, etc)

- Concerns about widespread use of Adobe Flash for interactivity
  (openness, reliability, security, performance)

- Concerns about consistency of handling invalid markup
  (browser "quirks modes")

- Overuse of generic semantics-light markup
  (div and span for everything)

HTML5 Design Principles

Compatibility

- – Support existing content (HTML 4 and earlier, broken markup)
- – Degrade gracefully (work well in older browsers)

Utility

- – Separation of concerns (content versus presentation)
- – Priority of constituencies (users > authors > implementers)

Interoperability

- – Well-defined behaviour
- – Graceful error handling

Universal access

- – Work on all platforms
- – Work for all users

HTML5 Design Principles in Practice

- Define what should happen when a browser gets bad markup (cannot assume well-formed or valid HTML)

- Define APIs that improve programmatic access to document and browser (HTML5 as Flash-killer)

- Define markup that better captures document semantics (better alternatives to span and div)
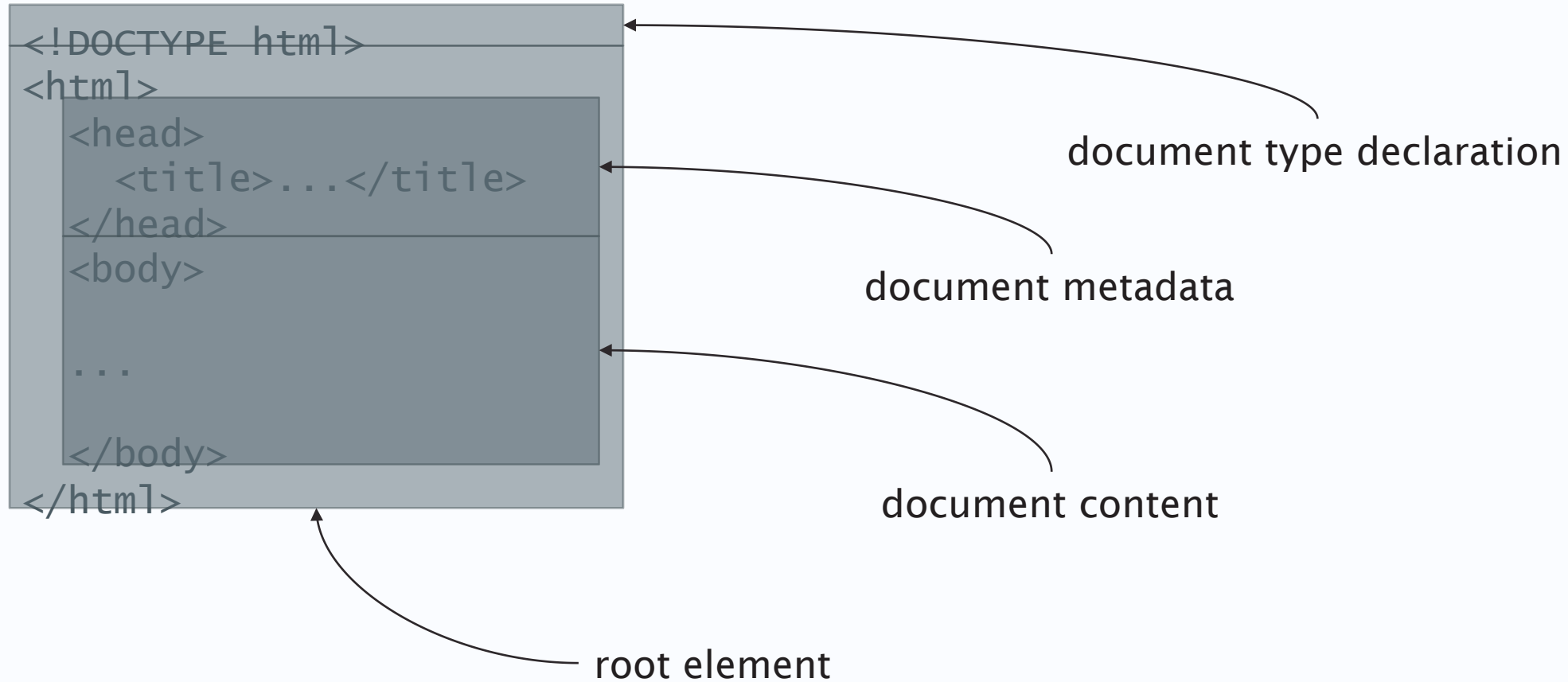
Notes on HTML5 parsing

HTML 4.01 (and earlier) and XHTML define a document grammar

- Declarative definition
- Describes the structure, but not how the structure should be processed
- Behaviour with ill-formed or invalid documents is undefined

HTML 5 also defines a document grammar

- Procedural definition
- Describes the process by which a compliant implementation should parse an HTML5 document (i.e. state machine)
- Defines behaviour with ill-formed or invalid documents

HTML5 top-level structure

```
<!DOCTYPE html>
<html>
  <head>
    <title>...</title>
  </head>
  <body>

    ...

  </body>
</html>
```

document type declaration

document metadata

document content

root element

Document-level metadata

Document title   `<title>`

Base URI   `<base>`

Document-level links   `<link>`

Presentation information   `<style>`

Other metadata   `<meta>`

Flow content

Things that behave like paragraphs…

- Paragraphs   `<p>`

- Lists   `<ol> <ul> <li> <dl> <dt> <dd>`

- Figures   `<figure> <figcaption>`

- Quotations   `<blockquote> <pre>`

- Miscellaneous   `<hr> <main> <div>`

- Headings   `<h1> <h2> <h3> <h4> <h5> <h6>`

Flow example

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Document</title>
  </head>
  <body>
    <h1>This is an example HTML document</h1>
    <ol>
      <li>First paragraph</li>
      <li>Second paragraph</li>
    </ol>
    <p>This is the first paragraph. <em>It doesn't really say
      very much.</em> Now is the time for all good men to come
      to the aid of the party.</p>
    <p>This is the second paragraph. The quick brown
      <a href="http://example.org/">fox</a> jumped over the
      lazy dog.</p>
  </body>
</html>
```

a top-level heading

an ordered list

the items in the list

a paragraph

another paragraph

Phrasing content

Things that behave like the contents of paragraphs...

30+ elements for structuring text within paragraphs

   – Links   <a>

   – Emphasis   <em> <strong> <i> <b>

   – Abbreviations   <abbr> <dfn>

   – Quotes and citations   <q> <cite>

   – Dates, times   <time>

   – Bidirectional text   <bdi>

   – Ruby annotations (pronunciation guides for East Asian typography – furigana, etc)

Phrasing example

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Document</title>
  </head>
  <body>
    <h1>This is an example HTML document</h1>
    <ol>
      <li>First paragraph</li>
      <li>Second paragraph</li>
    </ol>
    <p>This is the first paragraph. <em>It doesn't really say
    very much.</em> Now is the time for all good men to come
    to the aid of the party.</p>
    <p>This is the second paragraph. The quick brown
    <a href="http://example.org/">fox</a> jumped over the
    lazy dog.</p>
  </body>
</html>
```

some emphasised text

a link

Sectioning content

Elements that can be used to group together paragraph-like things:

- – Main body, headers and footers

- – Articles

- – Sections

- – Asides

- – Navigation

`<main>`

Represents the dominant content of the body of a document

- Should be no more than one `<main>` element in a document

```
<header><nav><a href="/">Home</a></nav></header>
<main>
  <h1>The history of the Web</h1>
  <h2>Introduction</h2>
  <blockquote>
    <p>Where shall I begin, please your Majesty?</p>
    <p>Begin at the beginning, the King said gravely, and go on
      till you come to the end: then stop.</p>
    <cite>Alice's Adventures in Wonderland; Lewis Carroll</cite>
  </blockquote>
...
</main>
```

<header>

Represents introductory content for its nearest ancestor, typically contains a group of introductory or navigational aids.

– When the nearest ancestor is the body element, then it applies to the whole page

```
<header>
  <h1>Scalable Vector Graphics (SVG) 1.2</h1>
  <p>W3C Working Draft 27 October 2004</p>
  <ul>
    <li><a href="http://www.w3.org/TR/2004/WD-SVG12-20041027/">This version</a></li>
    <li><a href="http://www.w3.org/TR/2004/WD-SVG12-20040510/">Previous version</a></li>
  </ul>
  <p>Editor: <a href="mailto:dean@w3.org">Dean Jackson, W3C</a></p>
</header>
```

\<footer\>

Represents a footer for its nearest ancestor, and typically contains information about its section (who wrote it, links to related documents, copyright data, etc)

– May contain entire sections: appendices, indexes, colophons

```
<footer>
  <nav>
    <ul>
      <li><a href="/credits.html">Credits</a></li>
      <li><a href="/tos.html">Terms of Service</a></li>
      <li><a href="/index.html">Blog Index</a></li>
    </ul>
  </nav>
  <p>Copyright © 2009 Gordon Freeman</p>
</footer>
```

<article>

Represents a complete, or self-contained, composition in a document, page, application, or site and that is, in principle, independently distributable or reusable

- – A forum post, newspaper article, a blog entry, comment, etc

```
<article>
  <header>
    <h1>The Very First Rule of Life</h1>
    <p><time>3 days ago</time></p>
  </header>
  <p>If there's a microphone anywhere near you, assume it's hot and
    sending whatever you're saying to the world. Seriously.</p>
  <footer><a href="?comments=1>Show comments</a></footer>
</article>
```

<section>

Represents a generic section of a document or application, a thematic grouping of content

- Part of something else (unlike an article, which is a thing in its own right)

```
<article>
  <header><h2>Apples</h2></header>
  <p>The apple is the pomaceous fruit of the apple tree.</p>
  <section><h3>Red Delicious</h3><p>These bright red apples are the
    most common found in many supermarkets.</p></section>
  <section><h3>Granny Smith</h3><p>These juicy, green apples make a
    great filling for apple pies.</p></section>
</article>
```

\<aside\>

Represents a section of a page that consists of content that is tangentially related to the content around the aside element, and which could be considered separate from that content

– Pull quotes, sidebars, footnotes, endnotes, etc

```
<aside>
  <h1>Switzerland</h1>
  <p>Switzerland, a land-locked country in the middle of geographic
    Europe, has not joined the geopolitical European Union, though
    it is a signatory to a number of European treaties.</p>
</aside>
```

<nav>

Represents a navigation block that groups links to other pages or to parts of the current page whose role is purely navigation

```
<nav>
  <ul>
    <li><a href="/">Home</a></li>
    <li><a href="/events">Events</a></li>
    <li><a href="/contact">Contact us</a></li>
  </ul>
</nav>
```
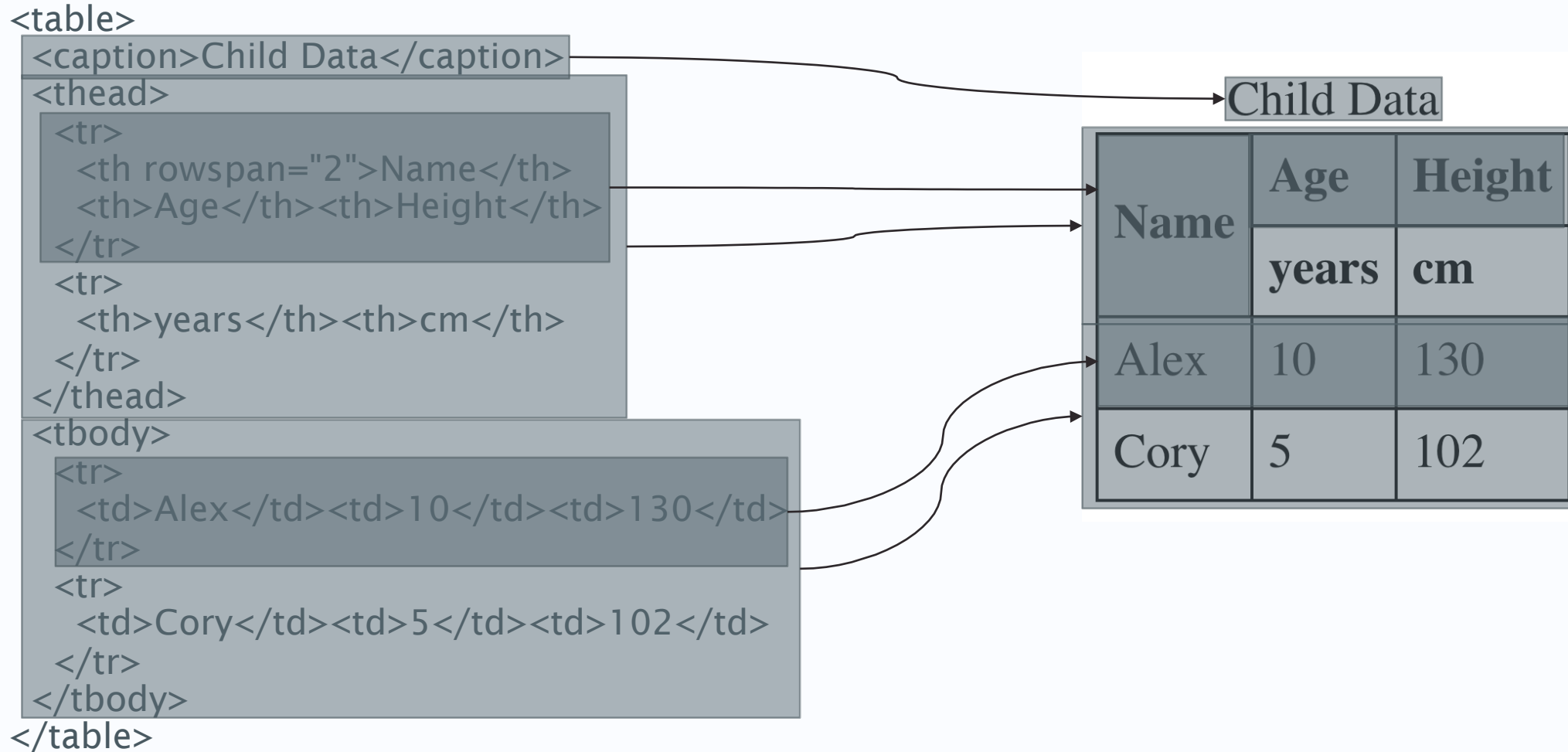
Tables

Expressive data model for tables

- Table header, body, footer   `<thead> <tbody> <tfoot>`

- Row-by-row data   `<tr> <th> <td>`

- Grouping of columns   `<colgroup>`

Table example

```
<table>
 <caption>Child Data</caption>
 <thead>
  <tr>
   <th rowspan="2">Name</th>
   <th>Age</th><th>Height</th>
  </tr>
  <tr>
   <th>years</th><th>cm</th>
  </tr>
 </thead>
 <tbody>
  <tr>
   <td>Alex</td><td>10</td><td>130</td>
  </tr>
  <tr>
   <td>Cory</td><td>5</td><td>102</td>
  </tr>
 </tbody>
</table>
```

Child Data

Name	Age	Height
	years	cm
Alex	10	130
Cory	5	102

Forms

Structured data entry widgets

- Text fields

- Buttons (including checkboxes and radio buttons)

- Date/time pickers

- Colour pickers

- File upload

- Sliders

- Single/multiple selections

- Progress indicators and meters

Embedded content

Images   `<img> <map> <area>`

Multimedia   `<video> <audio> <picture> <track> <source>`

Nested documents   `<iframe>`

Miscellaneous   `<object> <embed>`

HTML5 vs HTTP content negotiation

Most HTTP content negotiation is on format (i.e. media type)

Server-driven content negotiation has drawbacks

    – Difficult for servers to choose an appropriate format without knowing the user's intent

    – Clients often don't realistically advertise their preferences (i.e. `Accept: */*`)

Client-driven content negotiation also has a significant drawback

    – An extra HTTP round trip

HTML5 provides an alternative via the <source> element:

```
<video width="400" height="300">
  <source src="/media/intro.webm" type="video/webm">
  <source src="/media/intro.mp4" type="video/mp4">
  This browser does not support the HTML5 video element.
</video>
```

fallback content

Further Reading

HTML5.2 Spec

https://www.w3.org/TR/html52/ (not for the faint-hearted)

WHATWG HTML Living Standard

https://html.spec.whatwg.org/ (over 1200 pages!)

W3Schools HTML5 Tutorial

https://www.w3schools.com/html/ (accessible tutorials)

Mozilla Developer Network

https://developer.mozilla.org/en-US/docs/Web/HTML (comprehensive reference resource)

Next:
Web Formats

University of
Southampton