



Javascript and the DOM

Web Development
Supplemental

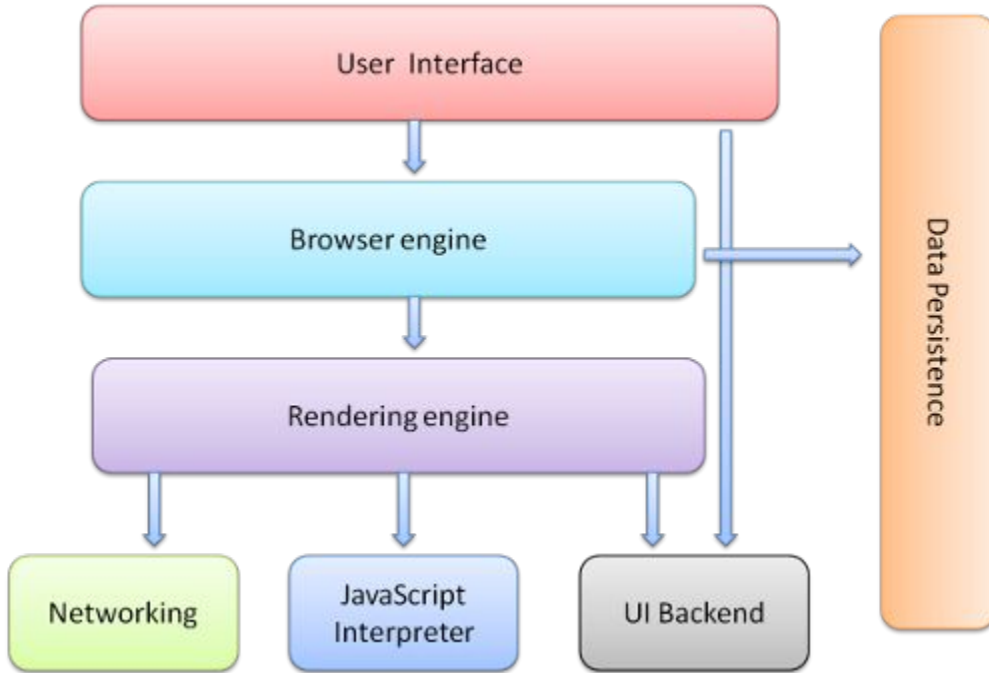




NOTE:

The part of the browser that handles Javascript is separate from the part of the browser that renders the display

Browser layers



The rendering engine is separate from the Javascript interpreter.



We need a way for the Javascript to listen for user interaction and to change the display based on that interaction

DOM - Document Object Model

The DOM is a bridge between the Rendering engine and the Javascript engine

The DOM provides to the Javascript Interpreter

1. Hooks for listening to user events (`document.addEventListener`)
 2. A predictable tree structure of elements and node that can be traversed in parent, child and sibling relationships (e.g., `document.body.children`)
 3. A means to query the tree to access an element or group of elements (`document.querySelector` and `document.querySelectorAll`)
 4. A means to change an element's content, style and attributes (`myEl.textContent`, `myEl.style`, `myEl.setAttribute`)
 5. A means to create new elements (`document.createElement`) and add them to the DOM (`document.appendChild`)
-

In Practice

index.html

```
<html>
  <head>...</head>
  <body>

    <h1 id="myHeader">My Header</h1>

    <script src="app.js"></script>

  </body>
</html>
```

app.js

```
// this establishes a connection between the HTML display and the Javascript
//
// headerEl is a js variable that stores a reference to the myHeader element
var headerEl = document.querySelector("#myHeader");
```