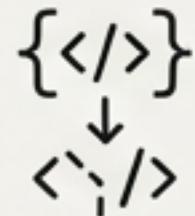


Transform Raw HTML into Actionable Graph Intelligence

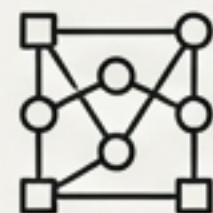
The Html_MGraph Service is a system designed to parse, transform, and visualise any HTML document as a sophisticated multi-graph structure. It moves beyond simple DOM trees to provide a complete, interconnected model of a document's structure, attributes, scripts, and styles.



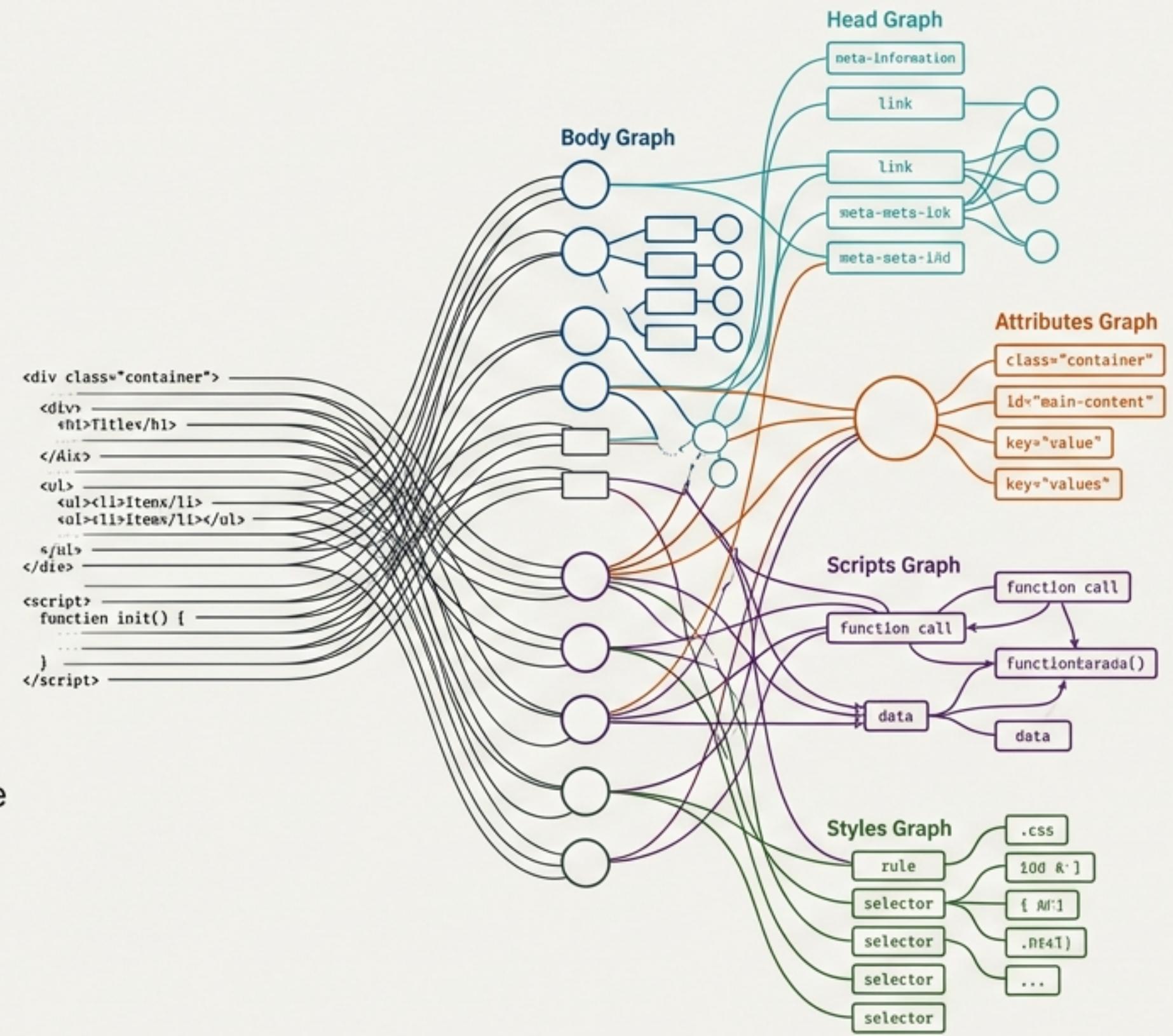
PARSE: Deconstruct any HTML into a structured, multi-graph model.



TRANSFORM: Apply customisable views to isolate and analyse specific aspects of the document.

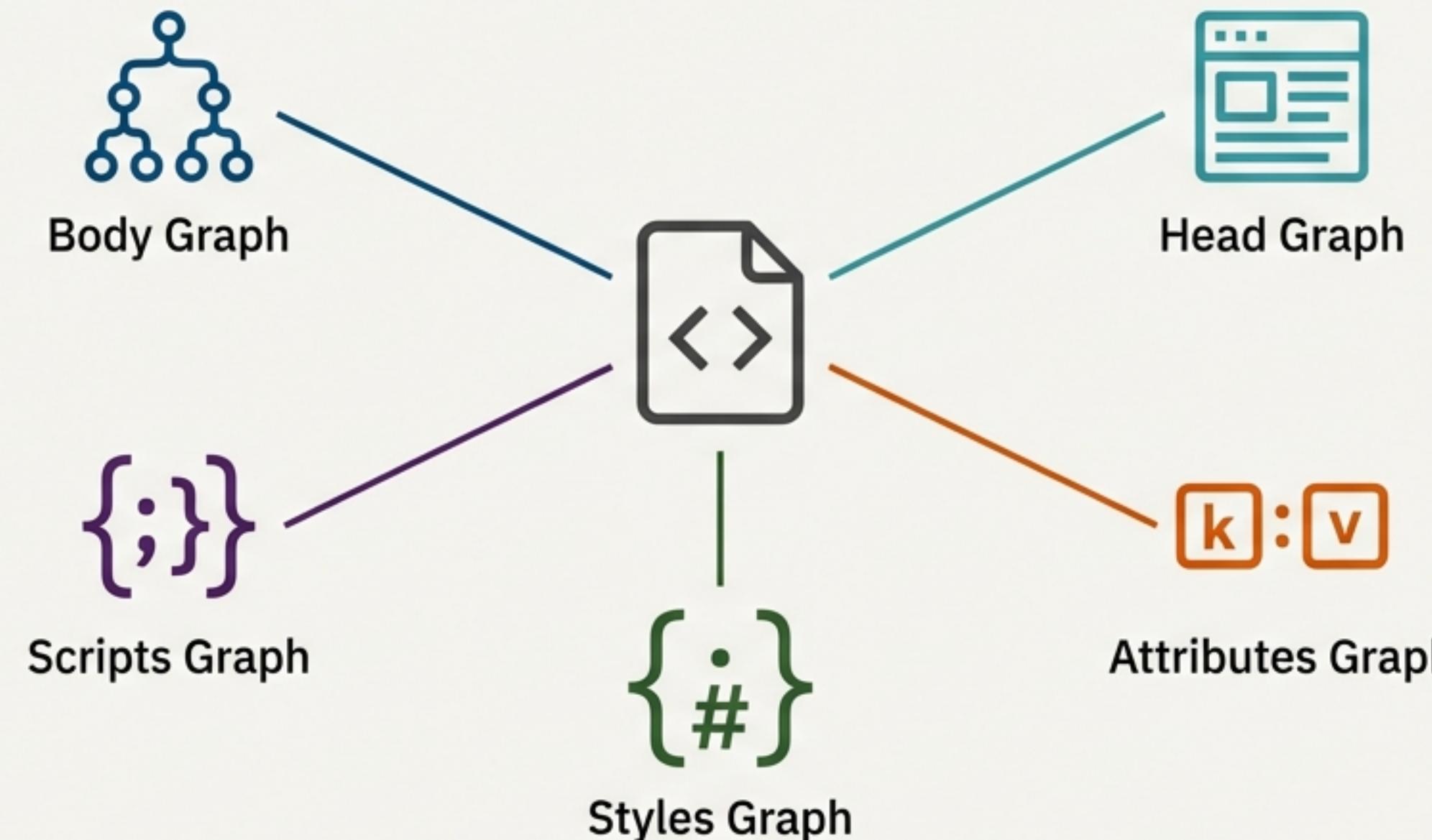


VISUALISE: Render the graph using a suite of powerful engines for static or interactive analysis.



HTML is More Than a Tree: The 5-Subgraph Model

To achieve full-fidelity representation, the service models an HTML document not as a single tree, but as five distinct, interconnected subgraphs. This layered approach allows for granular analysis and precise transformations.



Each subgraph can be analysed individually or combined into a clustered 'Full Document' view for a complete picture.

A Practical Example: From a Snippet to its Graph

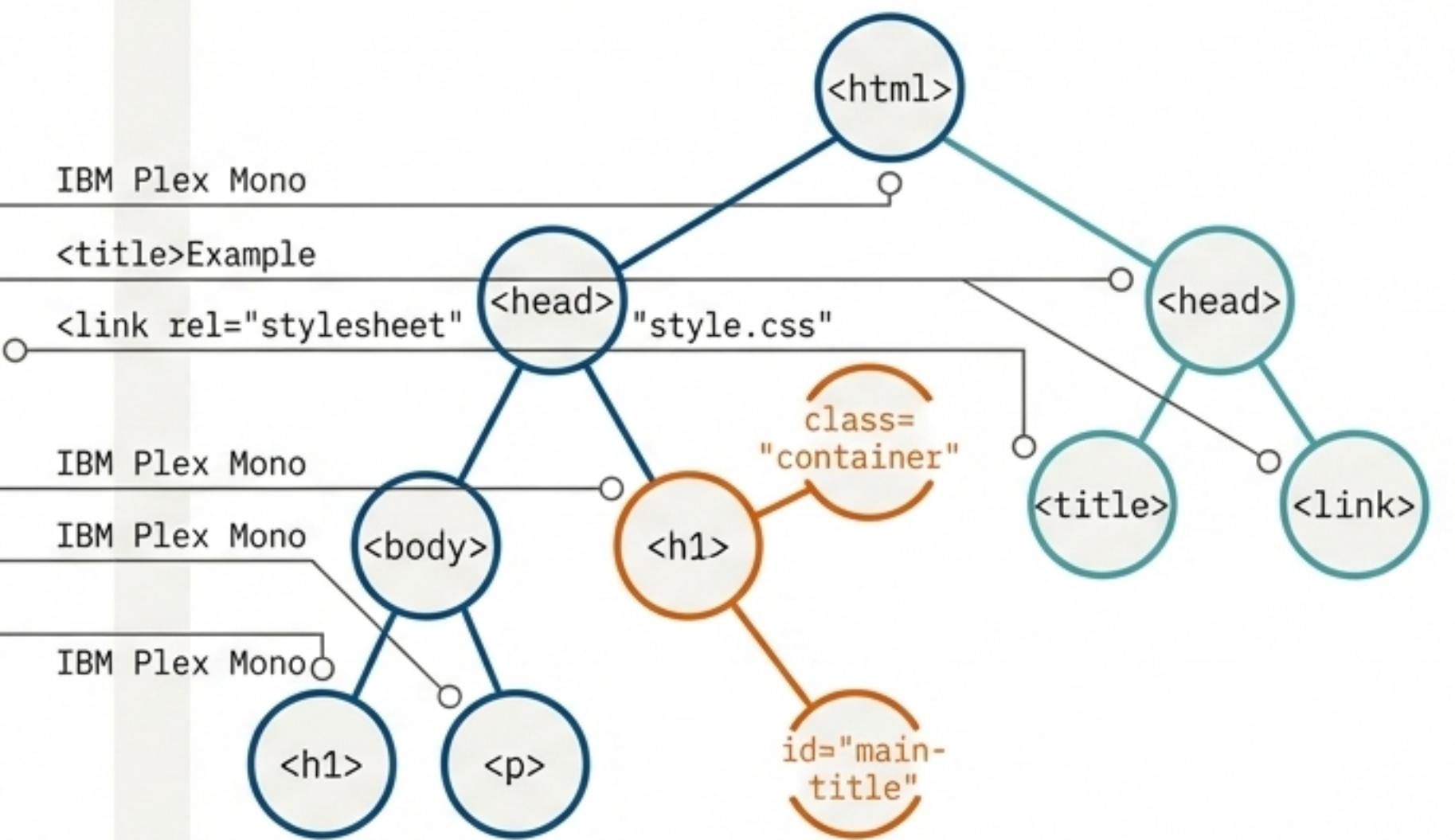
The Source

Inter SemiBold

```
<!DOCTYPE html>
<html>
<head>○
  <title>Example</title>○
  <link rel="stylesheet" href="style.css">○
</head>
<body class="container">○
  <h1 id="main-title">A Title</h1>○
  <p>Some text.</p>○
</body>
</html>
```

The Initial Graph Representation

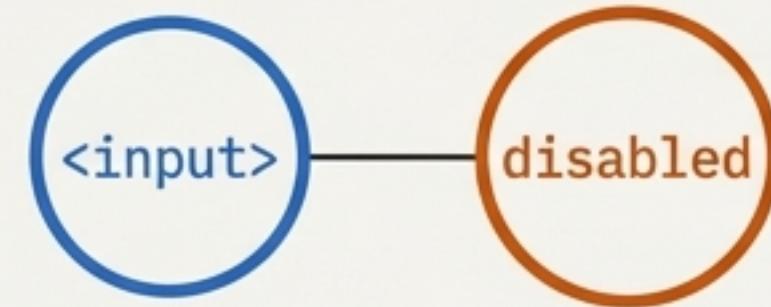
Inter SemiBold



The 3-Node Model: Engineering Efficient and Accurate Attributes

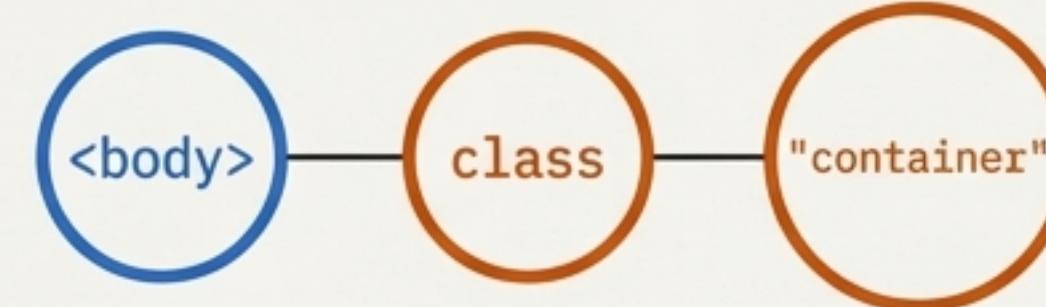
Standard key-value pairs are insufficient for HTML. Our system uses a **sophisticated 3-node model** to handle all attribute types with precision and storage efficiency.

Scenario 1: Boolean Attributes



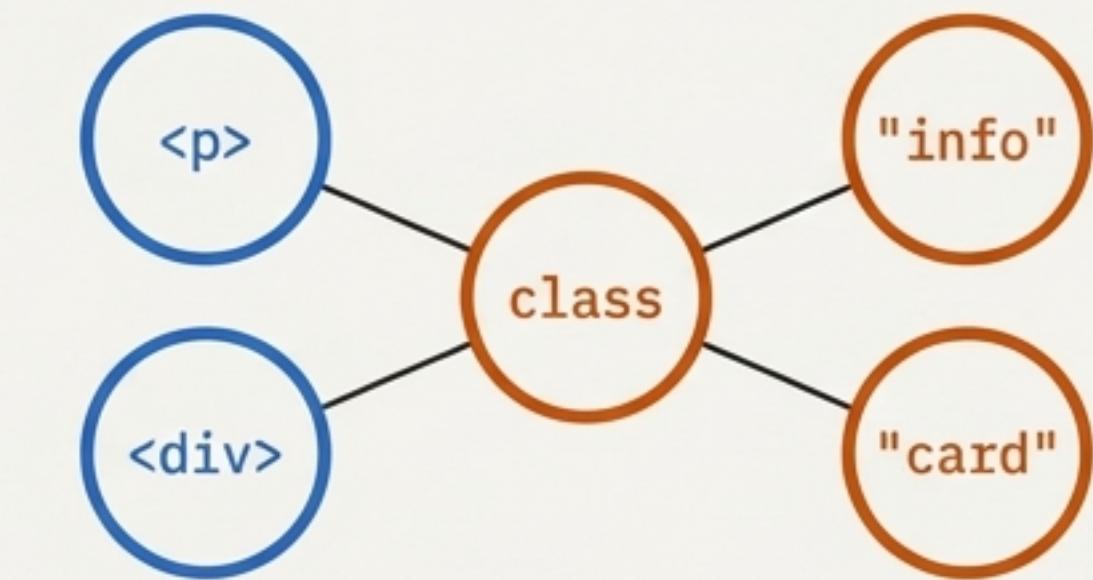
Handles valueless attributes like `required` or `disabled`.

Scenario 2: Valued Attributes



Represents standard name-value pairs.

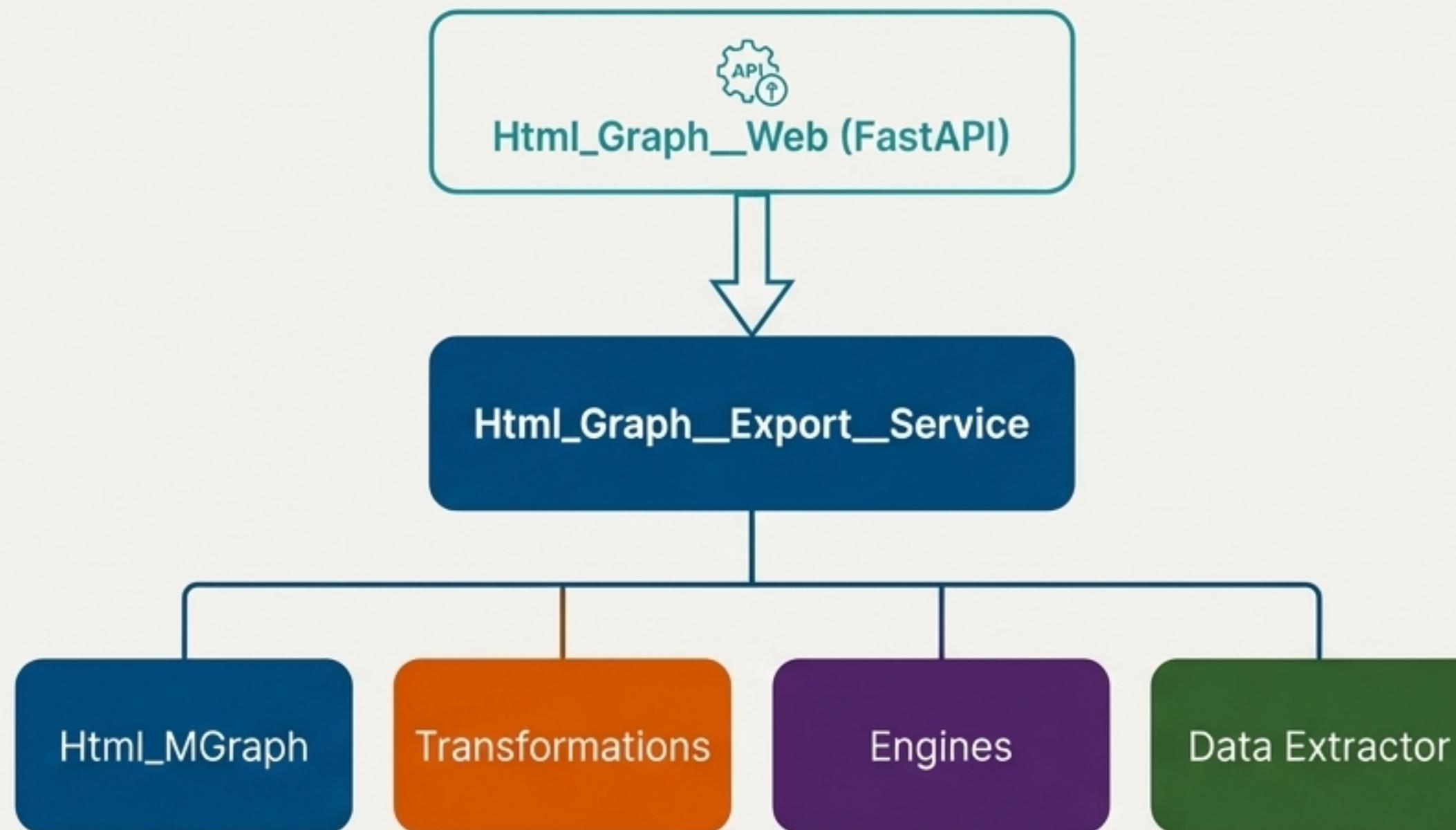
Scenario 3: Node Reuse



Attribute names and common values are reused, minimising graph size.

A Modular Architecture for Ultimate Flexibility

The service is built on a series of distinct components, each with a clear responsibility. This modular design, exposed via a REST API, ensures the system is both powerful and extensible.



Html_Graph_Web

FastAPI server exposing REST endpoints.

Html_Graph_Export_Service

Orchestrates the transformation pipeline.

Html_MGraph

The core multi-graph representation of HTML.

Transformations

Customise which graph or view is rendered.

Engines

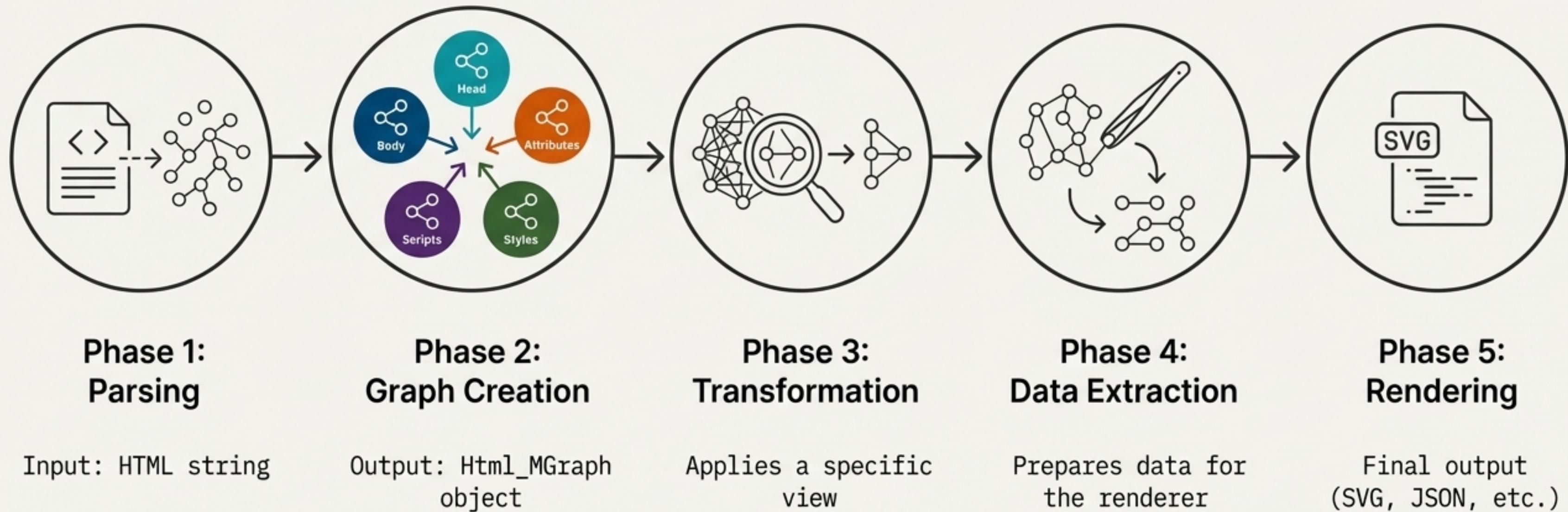
Convert the MGraph to final visual output formats.

Data Extractor

Extracts nodes and edges for visualisation.

The 5-Phase Pipeline: A Controlled Journey from HTML to Graph

Every request follows a structured 5-phase pipeline. This provides fine-grained control and allows for customisation at each stage of the process, a feature accessible through the Transformation Playground UI for debugging.



Transformations Define Your Perspective

Transformations are pluggable views that select and arrange subgraphs to answer specific questions about an HTML document. The system includes a set of standard transformations, and new custom views can be easily created.



Full Document

All subgraphs clustered into a complete view.



Default

The core Body DOM structure.



Attributes View

A dedicated view of all tags and their attributes.



Head View

The document's head section structure.



Scripts View

All script elements and their content.

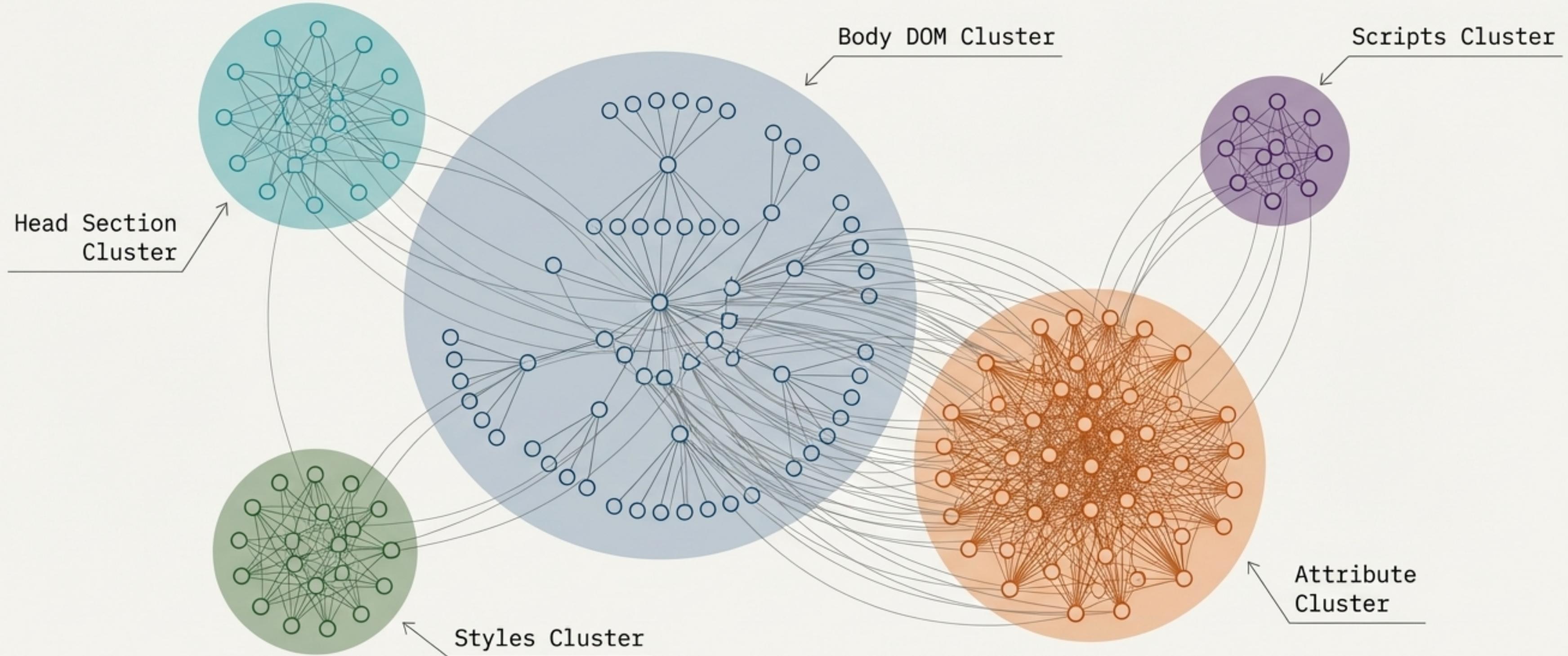


Styles View

Style elements and inline CSS.

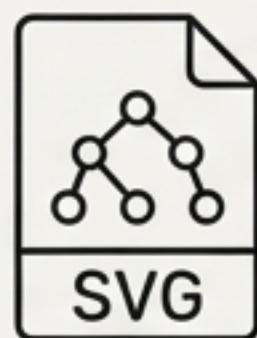
Spotlight: The ‘Full Document’ Transformation

The Full Document transformation is the most powerful view, combining all five subgraphs into a single, clustered visualisation. This provides an unparalleled, holistic overview of the entire HTML document’s composition.



Multiple Engines for Multiple Outputs

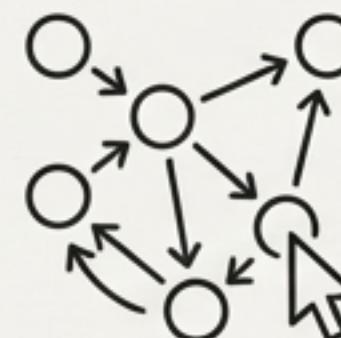
Once a graph is transformed, one of six pluggable rendering engines converts it into a final output format. Each engine is highly configurable and can be customised via engine-specific callbacks.



DOT (Graphviz)

Output: DOT language →
SVG

Best For: Static diagrams,
documentation



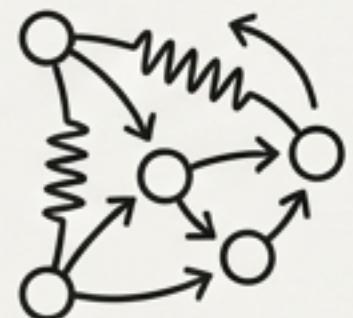
D3

Output: JSON for D3.js
Best For: Interactive
web visualisations



Cytoscape

Output: JSON for
Cytoscape.js
Best For: Complex
network analysis



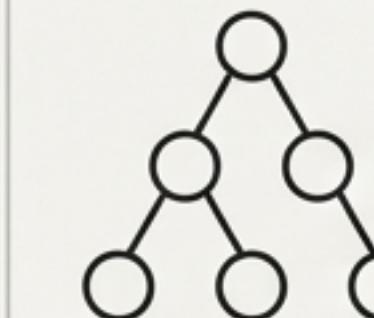
VisJs

Output: JSON for vis.js
Best For: Dynamic,
physics-based layouts



Mermaid

Output: Mermaid syntax
Best For: Markdown-
embeddable diagrams

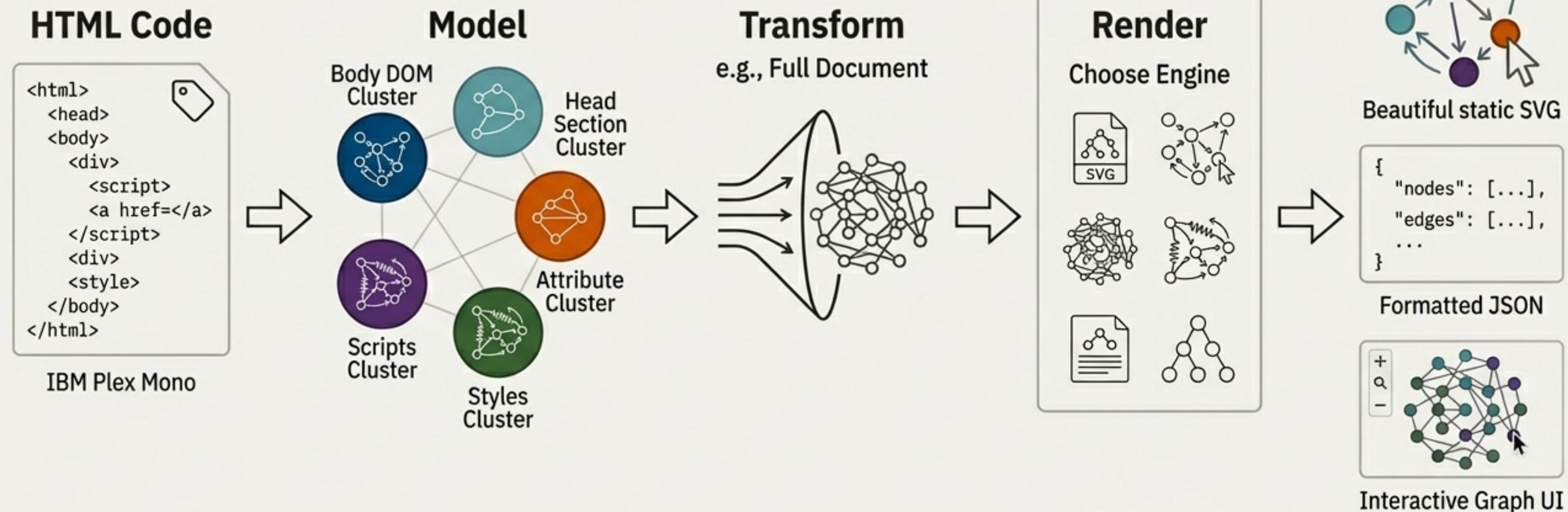


Tree

Output: JSON tree
structure
Best For: Hierarchical
views

The Complete Journey: From Code to Clarity

This unified process—from parsing, through multi-graph modelling, transformation, and rendering—provides a robust and repeatable way to gain insight into any HTML document.



Practical Applications and Use Cases

The ability to model and visualise HTML as a graph unlocks powerful new capabilities across development, security, and analysis workflows.



Web Scraping Analysis

Visualise DOM structure of scraped pages to identify layout patterns.



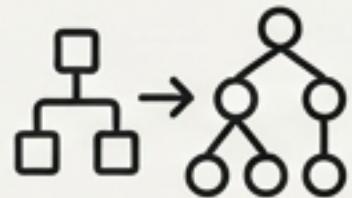
HTML Diff Visualisation

Compare two HTML documents by visualising differences in their graph representations.



Accessibility Auditing

Analyse attribute patterns to find missing ARIA attributes.



DOM Manipulation Planning

Model and visualise before/after states when planning complex DOM transformations.



Security Analysis

Isolate and inspect script and style injection points within a document.



Educational Tool

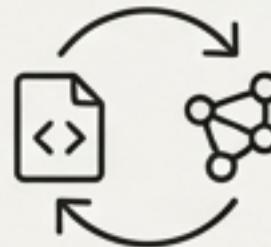
Teach HTML structure and relationships by showing a live visual representation.

Hallmarks of the Html_MGraph Service

The system's power is derived from a core set of design principles and features that ensure accuracy, efficiency, and extensibility.



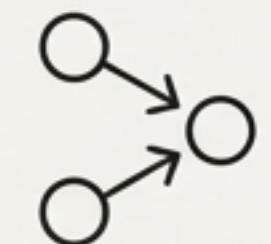
Multi-Graph Model: 5 interconnected graphs for complete HTML representation.



Round-Trip Fidelity: $\text{HTML} \rightarrow \text{Graph} \rightarrow \text{HTML}$ conversion preserves all structure perfectly.



Boolean Attribute Support: Properly handles attributes like `required` and `disabled`.



Efficient Node Reuse: Efficient storage with shared name/value nodes for attributes.



Pluggable Transformations: A flexible system for creating custom views and analyses.



Multiple Renderers: A suite of 6 engines (DOT, D3, Cytoscape, VisJs, Mermaid, Tree).



Clustered Views: Visually group related subgraphs with coloured clusters.



REST API: Full programmatic access and integration.

A System for Deep Structural Insight

The Html_MGraph Service provides a complete, extensible, and high-fidelity system for treating HTML not as a static document, but as a rich, queryable, and visualisable data structure.

It is a tool designed for developers, researchers, and architects who need to go beyond the surface of the web.

