

Structure

2. Structure HTML5

Problématique / Cas utilisateurs

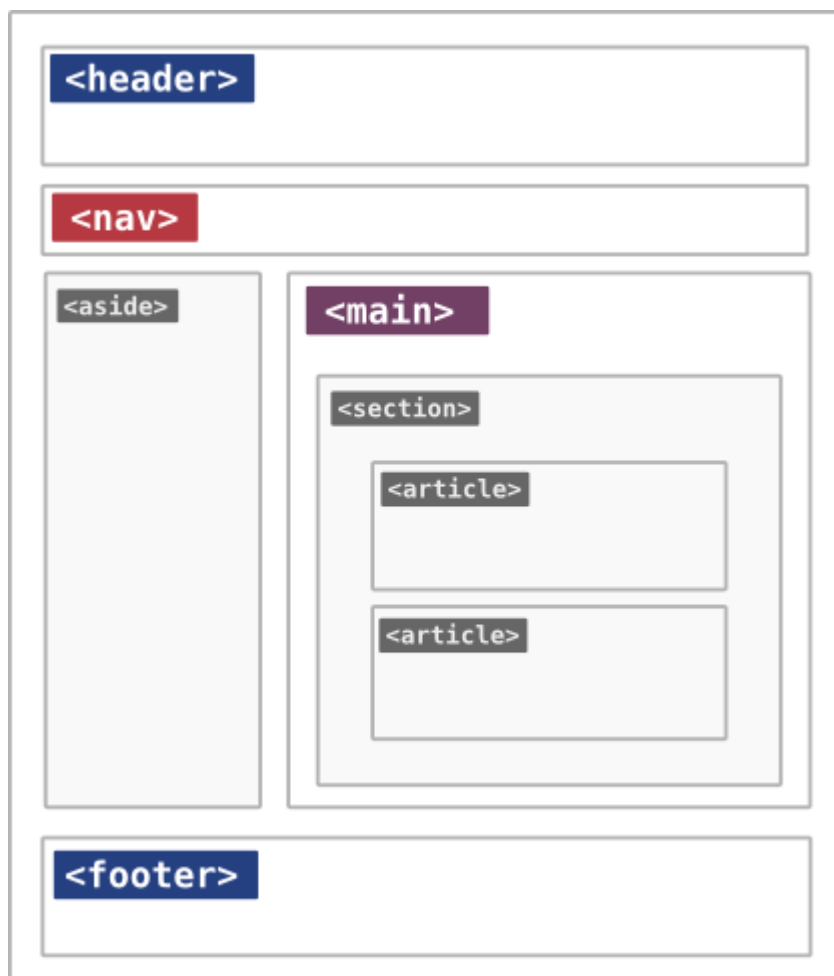
HTML5 propose de nouveaux éléments de structure qui permettent de déterminer les grandes zones du document. Cette structure pourra être utilisée par les technologies d'assistances, en conjonction avec les landmarks par exemple, pour proposer des fonctionnalités de navigation rapide dans le document.

Éléments obligatoires

Pour les documents HTML5, 4 balises sont obligatoires :

- **header** : sur l'en-tête principal du document (généralement le titre) ;
- **nav** : pour les zones de navigation principale, secondaire et dans les collections de pages (pagination par exemple) ;
- **main** : sur l'élément englobant le contenu principal du document. Cette balise est **unique** sur la page, les éléments répétés (navigation, publicité) ne devraient pas être structurés avec ;
- **footer** sur le pied de page principal du document.

Pour conserver l'efficacité et la sémantique de ces éléments, il est préférable de ne pas les imbriquer (par exemple, implémenter la navigation dans la balise **header**).



header

La balise **header** est obligatoire pour l'en-tête du document, mais on peut retrouver plusieurs balises

`header` dans une même page.

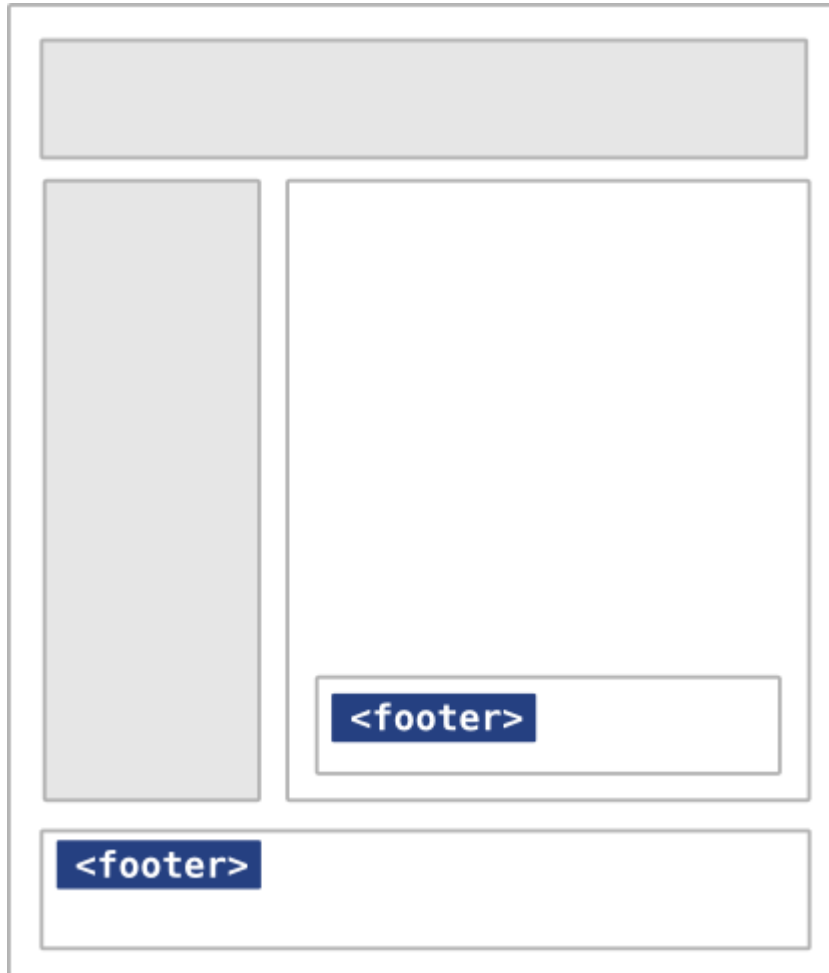
Une balise `header` peut contenir des titres de section, par exemple, de la navigation.



`footer`

La balise `footer` est obligatoire pour le pied de page du document, mais on peut retrouver plusieurs balises `footer` dans une même page.

Une balise `footer` peut être utilisée comme pied d'un article par exemple, pour indiquer des informations relatives au contenu (date, auteur ...).



Éléments optionnels

HTML5 propose d'autres éléments de structure : `section` , `article` et `aside` .

Ces éléments ne sont pas requis pour l'accessibilité, mais il est intéressant de les utiliser, ils constituent, avec `body` et `nav` , le plan du document (*outline*).

section et article

Une `section` est une partie d'un document ayant une valeur sémantique (e.g un regroupement thématique de contenus).

Au regard de la spécification, une section doit toujours être titrée soit via un titre `hx` ou un attribut `aria-label`

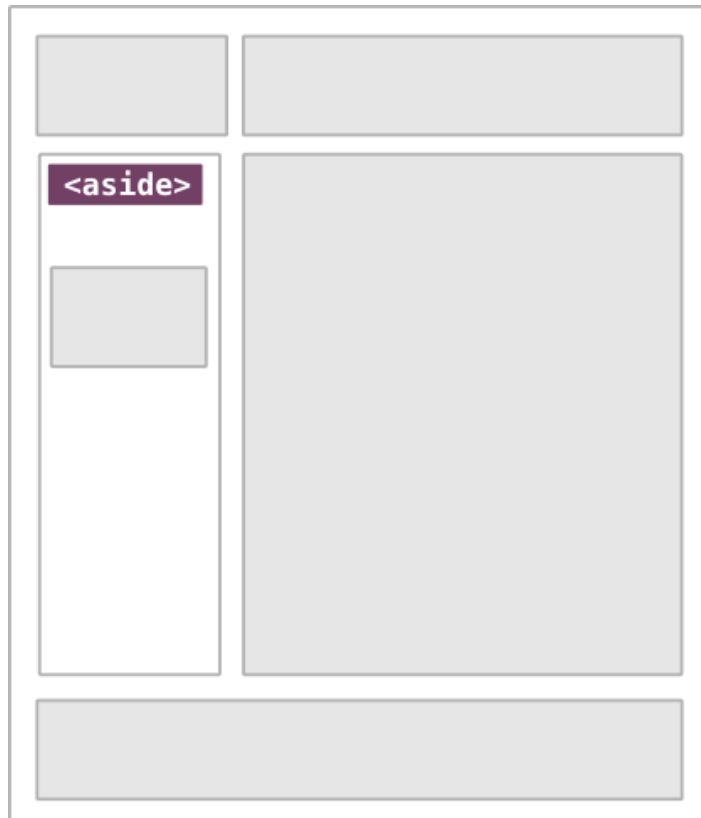
L'élément `section` peut servir de racine à une nouvelle hiérarchie de titres.

`article` est généralement comprise dans un élément `section` , elle est une partie indépendante d'un document pouvant être consultée, extraite ou réutilisée pour elle-même.



aside

La balise `aside` sert à indiquer des regroupements de contenus indirectement reliés au contenu principal et pouvant être consultés de manière indépendante (zones de contenus secondaires, publicités, groupement de liens de navigation secondaire...).



Plan du document : *outline*

Le plan du document est déduit des éléments sectionnants du document :

- **body** (racine de section) ;
- les éléments sectionnants **nav article section aside** ;
- les sections implicites créées à partir des titres **hx** .

Une navigation dans le site ou dans une rubrique, un sommaire, un contenu adjacent au contenu principal, le contenu principal ou des contenus secondaires comme un fil d'actualités Twitter sont autant d'exemples de contenus sectionnés.

Le rang des titres est déterminé par le nombre du nom de l'élément, où **>h1<** > a le rang le plus élevé et **>h6<** a le rang le plus bas. Le classement relatif entre les titres (dépendance, hiérarchie) ne compte qu'à l'intérieur d'une même section.

C'est la structure des sections qui détermine le plan, et non pas le rang des titres de sections.

Par exemple, considérons la structure suivante :

```
<section>
<h1>À la une aujourd'hui
</h1>
<section>
<h2>Le dernier arrive </
</h2>
</section>
</section>
<section>
<h3>Acheter à moindre co
</h3>
</section>
```

Cela conduit, avec l'algorithme de l'*outline* à restituer la structure suivante :

```
1. À la une aujourd'hui
1.1 Le dernier arrive bi
2. Acheter à moindre coû
```

On constate que le rang de l'élément titre n'est pas important dans la détermination du schéma du document. “Tout rang peut être utilisé comme en-tête d'une section explicitement définie, bien que cette pratique ne soit pas recommandée.” ([l'algorithme outline HTML5 sur MDN](#))

Le RGAA demande à ce que le plan du document doit rester cohérent, toutefois, devant la complexité de la détermination d'un tel plan, conjugué aux implémentations malheureuses des développeurs et aux interprétations différentes par les navigateurs, il [explique ce cas dans une note technique](#) :

Lorsqu'elle est utilisée, l'arborescence du document peut donc être différente de l'arborescence du contenu représentée par l'ensemble des titres `<hx>` de la page, même si les deux structures restent similaires.

Cette arborescence doit donc être représentative de la structure du document et être cohérente avec la structuration du contenu générée par l'utilisation des balises `<hx>`. La structuration du contenu générée par les balises `<hx>` pouvant être, théoriquement, déduite de l'arborescence du document, la spécification HTML5 recommande d'utiliser uniquement des titres `<h1>`. Cet usage est proscrit et le critère 9.1 impose d'utiliser une hiérarchie de titres (`<hx>`) cohérente.

Si l'arborescence du document (à la condition qu'elle soit cohérente) peut permettre de proposer à l'utilisateur des fonctionnalités d'exploration et de navigation, sur certaines technologies d'assistance, elle influe sur la hiérarchie de titres générée par l'utilisation des balises `<hx>` en modifiant le niveau des titres restitués.

Pour accompagner la prise en charge progressive de l'arborescence du document et compte tenu du fait que le référentiel exige de disposer, en tout état de cause, d'une structure de contenu (balises `<hx>`) robuste et cohérente, **il est acceptable de considérer le test 9.2.2 comme non applicable lorsqu'il n'est pas possible de s'assurer que l'arborescence du document est parfaitement cohérente.**

Dans ce cas, la non-conformité au test devrait être relevée sous la forme d'une simple alerte.

Le [validateur du W3C](#) vous permet

d'afficher le plan du document ainsi que l'*outline* qui découle (cochez la case *outline* avant de valider le formulaire)/



L'exemple ici montre la structure des titres de la page access42.net et l'arborescence qui en découle. Ces informations sont disponibles tout en bas de la page des résultats du validateur.

L'outil en ligne [HTML 5 Outliner](#) vous permet également de dégager l'outline du document et d'en évaluer la pertinence.

- [L'outline dans la spécification HTML](#) ;
- [Consulter l'article de Mozilla sur l'algorithme HTML5](#) ;
- [Description de l'outline sur HTML5 Doctor](#) ;
- [Note technique du RGAA concernant l'arborescence \(outline\) du document](#)

Exemples

Conforme

Le site du [Conseil Général du Pas-de-Calais](#) possède une structure HTML5 robuste.

Non conforme

La structure du site de la [ville de Marseille](#) est défaillante : pas de

[nav](#) et mauvaise utilisation de [section](#) et [article](#) .

Ressources

Références

RGAA

- [9.2 \[A\]](#)

Critères de succès WCAG

- [1.3.1](#)

Techniques et échecs WCAG

- [G115](#)
- [ARIA11](#)