



# Mise en page et mise en texte avec les feuilles de style CSS

Thierry Buquet

## ► To cite this version:

Thierry Buquet. Mise en page et mise en texte avec les feuilles de style CSS. Le médiéviste et l'ordinateur, 2004, 43. halshs-00007964

**HAL Id: halshs-00007964**

**<https://shs.hal.science/halshs-00007964>**

Submitted on 28 Feb 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

[menu](#) [Rubriques](#)

- [Dossiers](#)
- [Varia](#)
- [Numéros récents](#)
  - [42 : Diplomatie](#)
  - [41 : Cognition](#)
  - [40 : Numérisation](#)
  - [39 : Musicologie](#)
- [Archives](#)
- [Rechercher](#)
- [Newsletter](#)
- [Aide](#)
- [Rédaction](#)
- [Liens](#)

# Mise en texte et mise en page avec les feuilles de style CSS

## Auteur

Thierry Buquet  
[buquet@cnrs-orleans.fr](mailto:buquet@cnrs-orleans.fr)  
CNRS-IRHT  
<http://www.irht.cnrs.fr/>

## Citer cet article

Thierry Buquet, « Mise en page et mise en texte avec les feuilles de style CSS », *Le Médiéviste et l'ordinateur*, 43, 2004 [**En ligne**] <http://lemo.irht.cnrs.fr/43/43-13.htm>

## Mots clés

CSS, CSS2, feuilles de style, mise en page, mise en texte, typographie, HTML, XHTML

## Bibliographie

[Consulter la bibliographie de l'article](#)

**Résumé** : Après dix ans d'histoire, les feuilles de style CSS s'imposent aujourd'hui comme un standard incontournable pour la mise en page et la mise en texte sur le web. Malgré un support parfois défectueux par les navigateurs web, leur utilisation apporte de nombreux avantages : séparation contenu / mise en forme, contrôle précis de la typographie se rapprochant des normes habituelles des publications scientifiques, mise en pages personnalisables et mieux adaptées à la consultation écran. L'utilisation impose une meilleure structuration des documents et des articles, pour aboutir à une exploitation des textes sur différents supports : écrans de toutes tailles et impression papier. Pour l'impression, les CSS permettent d'aboutir à un contrôle très fin de la page,

même si les propriétés CSS le permettant, en pleine évolution, ne sont pas encore implémentées parfaitement. L'article se propose de faire le point sur les possibilités et limites actuelles des CSS, en essayant d'encourager le plus possible leur utilisation pour l'édition scientifique, afin d'obtenir une meilleure qualité de présentation sur le web et des documents structurés, modulables, bien hiérarchisés, et facilement imprimables.

## Sommaire

1. [Cadre de l'article : introduction aux CSS et leur utilisation pour l'édition électronique d'articles scientifiques](#)
2. [Définition et histoire des CSS](#)
  1. [Définition](#)
  2. [Historique](#)
  3. [Support des CSS par les navigateurs](#)
3. [Séparer le contenu de sa mise en forme](#)
  1. [L'évolution du HTML](#)
  2. [Avantages de la séparation contenu/forme](#)
  3. [Structurer un document](#)
  4. [Convertir un document Word bien stylé](#)
4. [Un peu de typographie](#)
  1. [Le web et la typographie](#)
  2. [Ce que permettent les CSS](#)
    1. [Tailles des caractères : \(font-size\)](#)
    2. [Police de caractère : \(font-family\)](#)
    3. [Mise en forme des lettres et des mots](#)
    4. [Mise en forme des paragraphes](#)
  3. [Ce qui n'est pas \(pour l'instant\) possible avec les CSS](#)
    1. [Composition avancée et justification](#)
    2. [Typographie et ponctuation](#)
    3. [Tabulations, colonnes](#)
    4. [Polices spécifiques : le cas des langues anciennes](#)
5. [Mettre en page avec les CSS](#)
  1. [Mettre en page pour l'écran](#)
    1. [Les blocs ou ensembles de paragraphes](#)
    2. [De la page au navigateur : différences entre la mise en page d'un article et sa mise à l'écran](#)
  2. [Mettre en page pour l'impression](#)
6. [Conclusion et perspectives](#)
7. [Webographie](#)

## Cadre de l'article : introduction aux CSS et leur utilisation pour l'édition électronique d'articles scientifiques

Le présent article ne se veut pas un tutoriel ou cours en ligne sur les feuilles de styles CSS. Le propos est d'étudier l'apport de cette norme pour éditer des documents scientifiques en ligne, et plus particulièrement des articles. Il s'agit à la fois d'un ensemble de réflexions générales personnelles et de points techniques particuliers qui visent à dresser un état de l'utilisation actuelle des feuilles de style. Ces points concernent principalement les questions de mise en texte et de mise en page. Si le premier point peut sembler évident (réglages typographiques), la mise en page sur le web se trouve fondamentalement modifiée par les CSS, que ce soit à l'écran ou à l'impression des pages HTML. Le propos général de l'article se limitera donc à expliquer ce qui change avec les CSS pour l'édition en ligne.

## Définition et histoire des CSS

## Définition

L'acronyme CSS désigne les Cascading style sheets, c'est-à-dire les feuilles de style en cascade. CSS est un standard du consortium W3C, ouvert, libre et indépendant.<sup>1</sup> Comme pour HTML et d'autres standards, un groupe de travail définit et fait évoluer la norme, dans le respect des règles du W3C. Une feuille de style est un ensemble de règles qui définit la présentation d'un document, d'une façon comparable aux styles des logiciels de traitement de texte et de mise en page (Word, X-Press, In-Design...) Sur le web, une feuille de style précise également un type d'affichage sur un média déterminé (écran, TV, assistant personnel, page imprimée, etc.). Les CSS ont été conçues au départ pour enrichir et normaliser la présentation des pages HTML ; aujourd'hui elles peuvent également être utilisées avec des documents XML. Il existe une autre famille de feuilles de styles liées aux documents XML : les XSL, qui en plus de présenter, permettent de manipuler et transformer les données XML. <sup>2</sup>

## Historique

Le premier document relatif aux feuilles de styles, préfigurant la future norme CSS a déjà dix ans. Il a été rédigé par Håkon Wium Lie, qui fut par la suite à l'origine du groupe de travail CSS au sein du W3C <sup>3</sup>. Mais la question de l'enrichissement de la présentation du HTML se posait depuis le début des années quatre-vingt-dix : les différents navigateurs web de l'époque proposaient leurs propres feuilles de styles personnalisables mais uniquement du côté de l'utilisateur. L'apport de Lie était de proposer aux développeurs web un langage indépendant du navigateur et du système d'exploitation, permettant un contrôle fin sur l'affichage des textes et des éléments de mise en page. L'autre idée neuve était le concept de mise en « cascade », c'est-à-dire la capacité d'un élément particulier d'hériter de son élément « parent » ses caractéristiques de style, et la mise en cascade de plusieurs feuilles de styles successives, permettant un contrôle très fin sur la présentation.

Dès sa création, le consortium web W3C a soutenu ce projet de feuilles de styles et fin 1996 la première recommandation CSS1 était publiée. En 1997, le groupe de travail CSS au sein du W3C était créé, poursuivant l'évolution de la norme par le CSS 2 en 1998 et par la CSS 3 encore en cours de développement.

## Support des CSS par les navigateurs

La recommandation des feuilles de style resta longtemps sans effet : il fallut attendre 1999 et la sortie d'Internet Explorer 5 de Microsoft pour qu'un navigateur supporte la norme CSS de façon convenable. IE 4 comportait de nombreux bugs, ainsi que les versions 4 à 4.7 de Netscape, qui de son côté a tenté d'imposer sa propre norme de feuille de style, en liaison avec Javascript. À l'époque, le web opère une véritable explosion et le métier de webmaster ou de webdesigner se met en place dans une certaine confusion : les techniques CSS sont alors mal connues et leur mauvais support par les navigateurs de cette période (où rappelons-nous Netscape était fortement majoritaire) les rendent malheureusement inusitées. Les webmasters utilisent toutes sortes de techniques « bricolées » permettant une mise en page élaborée supportée par les navigateurs courants : utilisation de frames (cadres), de tableaux, positionnement d'éléments en utilisant des images invisibles, texte en mode image, etc. sans oublier des développements javascript de plus en plus perfectionnés créant de l'interactivité.<sup>4</sup> Les pages web deviennent de plus en plus lourdes, nécessitant des machines puissantes et / ou une connexion haut-débit. De plus, ces sites combinant images, tableaux, frames, javascripts deviennent de plus en plus difficiles à maintenir et surtout à migrer, sachant qu'un site web professionnel subit en moyenne une refonte complète tous les trois ans.

Aujourd'hui, la situation a changé : Internet Explorer a près de 90 % du marché, et la plupart des navigateurs utilisés proposent un support très correct des CSS (Netscape 6 et 7, Mozilla, Firefox, Opera, etc.). Mais ce support reste imparfait et il demeure de nombreux « bugs » dans tous les navigateurs, et suivant les systèmes d'exploitation.<sup>5</sup> Ces difficultés semblent encore un obstacle à l'utilisation des CSS. La suite de cet article essaiera de fournir des arguments pour cette utilisation, dont les avantages dépassent de loin les difficultés de mise en place. L'autre aspect de cette professionnalisation de plus en plus poussée d'Internet est l'utilisation d'outils de gestion de contenus pour la production et la maintenance des sites : bases de données et CMS (*content*

management system)[6](#). Les feuilles de styles sont ici un moyen pratique de mettre en forme des données « brutes », qu'elles soient issues d'un SGBD ou de fichiers XML.

## Séparer le contenu de sa mise en forme

### L'évolution du HTML

Au début était le HMTL, langage simple de description de page. Sa facilité de mise en œuvre, ainsi que sa légèreté ont assuré le succès du web au début des années 90. Très vite les webmasters ont voulu enrichir ce langage simple pour le rapprocher de la PAO : enrichissement des textes, ajout de graphismes, mise en page élaborée. Alors que les feuilles de style naissent très vite, elles ne sont pas très utilisées comme nous l'avons vu. Le HTML s'est alors enrichi, jusqu'à sa version 4, de nombreuses fonctionnalités permettant d'améliorer la présentation : type de police, taille du texte, couleur de fond et de texte. Alors qu'au début, on avait une page HTML très simpliste mais assez bien structurée (balisage de titres et paragraphes), on est arrivé à des pages très complexes où structure et mise en forme se mélangent :

HTML 1 : `<p>ceci est un paragraphe</p>`

HTML 4 :: `<p align="justify"><b><i><font face="verdana" size="2" color="red">ceci est un paragraphe</i><b></p>`

Mais ces attributs du HTML ne permettaient qu'un contrôle limité sur le texte et sa mise en page, d'où un recours massif à toutes sortes d'astuces évoquées plus haut, notamment l'utilisation des balises de tableaux (TABLE, TR, TD) pour la mise en page, alors qu'elles avaient été prévues au départ pour... créer des tableaux présentant des données tabulaires, à la façon d'Excel !

Aujourd'hui avec le remplacement progressif du HMTL 4 par le XHTML[7](#), le web semble atteindre une certaine maturité, avec pour priorité une meilleure gestion des contenus. XHTML, comme son nom le laisse supposer, tend à se rapprocher d'XML. Comme ce dernier, une plus grande rigueur est demandée pour la syntaxe de son code, et il tend aussi à devenir un puissant outil de structuration de données, en laissant prioritairement la présentation aux feuilles de styles CSS.[8](#)

### Avantages de la séparation contenu/forme

Une page HTML réalisée suivant des normes « dépassées » (HTML 3 ou 4 sans feuilles de style) est une ensemble très hétérogène d'éléments balisant à la fois des contenus (textes, images), leur mise en forme (formatage de texte, marges, espaces, etc.) et des éléments de programmation (javascripts). Le code d'une telle page est une véritable « jungle » foisonnante où il est bien difficile de retrouver ses petits même pour un webmaster expérimenté. Ceci pose évidemment des problèmes de maintenance courante, mais plus encore lors de la reprise et de la refonte d'un site : changement de mise en page de maquettes de couleurs, etc. Malgré l'utilisation de logiciels commerciaux comme Dreamweaver, permettant une gestion « propriétaire » de modèles de pages facilitant le changement de maquette d'un site web, ce type d'opération reste difficile, obligeant souvent à intervenir « manuellement » dans le code. On voit tout de suite l'avantage de la définition des caractéristiques de mise en page et de mise en texte dans une feuille de style séparée : quelques lignes modifiées dans un seul document permet alors de changer radicalement l'apparence d'un site web.[9](#)

Pour la publication d'articles scientifiques ou l'édition critique de textes, la séparation du contenu et de la mise en forme peut permettre d'adapter la présentation de l'article ainsi que les normes typographiques à celles pratiquées en général dans le site où il sera publié. Comme dans une revue scientifique traditionnelle sur papier, un site web universitaire ou de publication en ligne doit avoir sa propre charte rédactionnelle, ses normes typographiques, sa norme de présentation de la bibliographie. Séparer les éléments de mise en forme du document permet d'affiner ces réglages typographiques et de fixer les normes rédactionnelles dans un document défini, pouvant alors servir de référence.

Enfin, séparer la forme du fond permet aussi de mieux valoriser ce dernier pour son contenu en dégageant ainsi sa structure : car appliquer des styles prédéfinis à un document exige que celui-ci soit bien construit, cohérent et que chaque élément de sa structure soit clairement identifié : sinon les styles ne peuvent être appliqués. Par exemple, la norme habituelle de présentation d'un nom d'auteur est la transformation en petites capitales ; une référence bibliographique complète de fin d'ouvrage est souvent représentée avec un retrait négatif de première ligne. Ces éléments de mise en page et de typographie peuvent être définis dans la feuille de style :

```
.nomauteur {font-variant: small-caps; } définition des petites capitales pour le style  
« nomauteur »
```

```
.titreouvrage {font-style: italic;} définition du style du titre en italique
```

```
.bibliographie {text-indent: -20mm;} la première ligne est décalée de 20mm vers la gauche
```

Pour que ces styles soient opérants le code HTML doit avoir précisé les informations :

```
<p class="bibliographie"> <span class="nomauteur">Lalou</span>, Élisabeth <span  
class="titreouvrage">Titre de l'ouvrage, Ville, éditeur, année (Collection)</p>
```

Le paragraphe s'affichera ainsi :

Lalou Élisabeth, *Les comptes sur tablettes de cire de Jean Sarrazin, chambellan de saint Louis*, Turnhout, 2003 (Monumenta Paleographica Medii Aevi, Séries Gallica, 4).

## Structurer un document

Comme nous venons de l'évoquer, l'utilisation des feuilles de style nécessite au préalable de bien structurer son document : chaque partie doit être clairement identifiée, les éléments sémantiques bien différenciés et marqués comme tels : titres, paragraphes de différents types (courant, citation, bibliographie, notes, annexes...). Ce qui devrait toujours être fait d'ailleurs, mais hélas trop souvent en sciences humaines, les articles sont mal structurés, sans une hiérarchie de titre précise, sans plan. L'utilisation des nouvelles technologies du web impose donc une plus grande rigueur dans la création des documents et plus seulement dans leur mise en forme pour l'impression. La préparation typographique (mise aux normes, uniformisation) des articles scientifiques avant leur publication ne doit pas être seulement d'ordre formel (élégant, bien présenté, uniformisé, respectant des règles, intégré dans une maquette bien définie d'une revue) mais doit déboucher aujourd'hui sur une véritable préparation structurelle des documents. Non seulement ces articles seront plus faciles à mettre en ligne avec les standards actuels d'Internet, mais ils y gagneront en clarté et en cohérence intellectuelle.

L'autre point important concerne la lecture à l'écran, le repérage rapide des informations essentielles et l'indexation par des moteurs de recherche ou des portails : ces questions sont fondamentales pour l'édition en ligne. Un document bien hiérarchisé permettra de fournir un sommaire précis permettant de se repérer dans l'article ou le dossier ; un résumé ou « abstract » permettra d'inciter ou non le lecteur à consulter le document en lui fournissant des informations de base.[10](#)

## Convertir un document Word bien stylé

Les logiciels de traitement de texte (Word et quelques autres plus marginaux[11](#)) sont devenus l'outil privilégié de production de l'écrit scientifique et historique. En attendant dans l'avenir des outils d'éditions orientés XML faciles d'utilisation, il faut s'adapter à l'utilisation majoritaire de Word. Celui-ci permet, s'il est correctement utilisé, la création de documents structurés : titres hiérarchisés, index, table des matières, notes et renvois « dynamiques ». Par l'utilisation des styles Word et d'un document modèle, on peut faciliter la mise en forme du document. L'idée est d'utiliser les styles de mise en forme de Word pour dégager une structure fine du document : chaque style de paragraphe ou de caractère devra porter un nom relatif au contenu qu'il doit mettre en forme. D'une certaine façon on doit chercher à obtenir un document Word quasi « sémantisé » permettant

ensuite de dégager une structure claire, facile à convertir en HTML [12](#). La conversion HTML de ce document permettra aussi de récupérer les styles de Word, affinant encore la structure. L'astuce est de retrouver des noms identiques entre Styles Word et styles CSS comme dans le tableau suivant :

Élément word	Style css	Balisage html
Titre 1 à 4		<h1>, <h2>, <h3>, <h4>
Paragraphe normal		<p class="normal">
Style : bibliographie	.bibliographie	<p class="bibliographie">
Liste à puces		<ul>
Style : nomauteur	.nomauteur	<p class="nomauteur">

## Un peu de typographie

### Le web et la typographie

Pendant les premières années d'Internet, la priorité était d'échanger du texte de la façon la plus simple possible. Au début d'HTML, le texte était transmis de façon brute, sans information relative à la police, à la taille des caractères, etc. Le navigateur s'appuyait sur les polices systèmes des ordinateurs « clients », avec possibilité de paramétrer le navigateur en modifiant la police d'affichage ainsi que sa taille (possibilité qui existe encore aujourd'hui). Petit à petit, avec la demande pressante des premiers webmasters et webdesigners, HTML s'est enrichi de tout un jeu de balises de mise en forme des textes : définition de la police, de sa taille, du gras, de l'italique, alignement et justification des paragraphes. Mais ce jeu de balises assez simple ne permettait d'avoir accès qu'à une très faible partie des règles typographiques habituelles de l'édition papier : par exemple, pas de petites capitales, pas de retraits négatifs, etc. De plus, l'idée longtemps (encore ?) répandue d'une édition « au rabais » et la prise en main du web par des personnes dont les métiers étaient très éloignés de ceux de l'édition (informaticiens, chercheurs) a favorisé un laisser-aller important dans la présentation des textes sur le web. Entre ignorance des règles typographiques élémentaires et outils inadaptés, la « mise en texte » sur Internet est longtemps restée frustrée : ceci a été parfois un argument donné par certains chercheurs refusant de mettre en ligne leur production scientifique dans une publication « au rabais » ne respectant pas les règles les plus élémentaires de l'édition.

### Ce que permettent les CSS

Concernant la mise en forme des textes, les CSS permettent d'avoir un contrôle fin sur la typographie. Les éléments suivants décrits très sommairement [13](#) symbolisent la grande capacité des CSS à mettre en forme du texte, presque aussi finement que dans un logiciel de PAO.

#### Tailles des caractères : (font-size)

- Contrôle précis avec différentes unités (pixels, mm, pt, em – taille de la police en cours sur le système), pourcentage.

#### Police de caractère : (font-family)

Il y a cependant une limitation de taille : alors que pour le papier le choix de fontes est illimité, ici il faut considérer les polices les plus courantes communes à toutes les plates-formes (PC, MAC, Linux, UNIX) pré-installées sur les machines. En pratique, ces polices sont Arial Verdana, Times, Times, New Roman. Il faut savoir que pour des raisons de lisibilité à l'écran on favorise les polices « bâton » ou sans-sérif pour l'écran (ex. Arial ou Verdana), alors que les « sérif » avec empattement (Times) sont plus agréables à la lecture sur papier.

#### Mise en forme des lettres et des mots

Espace entre les mots : `word-spacing`

Espace entre les lettres : `letter-spacing`

Soulignements et barrés : `text-decoration` (`none`, `underline`, `overline`, `line-through`)

Casse : `text-transform` (`uppercase`, `lowercase`, `capitalize`) et pour les petites capitales `font-variant: small-caps`)

Graisse : `font-weight` (différentes valeurs de `bold` à des valeurs numériques)

Style : (`italic`, `normal`, `oblique`)

Couleur : `color`

Alignement vertical : `vertical-align`: `sub`, `sup`, `middle`, `baseline`, etc.

## **Mise en forme des paragraphes**

Marges et retraits : `margin`. Permet de gérer les marges gauche et droite des paragraphes mais aussi et surtout les espaces avant et après les titres et paragraphes. Ceci permet de bien organiser visuellement l'information lors de la mise en texte.

Alignements et justifications : `text-align`: `left`, `right`, `center`, `justify`

Hauteur de ligne : `line-height`. Permet de gérer précisément l'interlignage dans un paragraphe en % ou en valeur d'unité.

Indentations `text-indent` avec sa valeur positive ou négative

## **Ce qui n'est pas (pour l'instant) possible avec les CSS**

Le tableau pourrait paraître idyllique : les CSS permettent un contrôle typographique total, compatible avec l'exigence de l'édition « papier », regroupant la quasi-totalité des fonctionnalités des logiciels de mise en page et de traitement de texte. Nous allons détailler maintenant les problèmes qui restent posés.

## **Composition avancée et justification**

La justification des textes reste très simple : les navigateurs ne disposent pas d'un moteur de composition avancée. Il ne peut pas non plus sur une page web y avoir de césures (coupure de mots pour affiner et équilibrer un texte justifié) car les différences d'affichage du texte selon les plates-formes et les navigateurs rendraient aléatoire cette opération. De plus, le contrôle sur l'impression des documents reste difficile à obtenir, ne pouvant prédire le matériel dont dispose l'internaute. De fait, cette limitation dans la justification des textes est imposée par le média Internet. La justification reste par ailleurs souvent hasardeuse : les polices « bâton » standard utilisés sur Internet (Arial et Verdana) de qualité typographique très moyenne se justifient assez mal, surtout la Verdana qui possède des espaces interlettres et intermots assez élevés, qui permettent sa bonne lisibilité à l'écran. Pour éviter des espaces disgracieux dans une colonne de texte justifiée, nous conseillons de ne pas justifier un bloc de moins de 300 pixels de large.

## **Typographie et ponctuation**

L'utilisation des feuilles de style CSS ne règle pas le problème des règles typographiques liées à la ponctuation : intégration d'espaces insécables avant les ponctuations doubles (guillemets français, point d'interrogation, d'exclamation). Un logiciel comme Word, s'il a pour paramètres des règles typographiques françaises, peut



intégrer ces espaces : ce n'est pas le cas des éditeurs HTML courants et pour l'instant la norme CSS ne permet pas ce type d'insertion typographique conditionnelle.

## Tabulations, colonnes

Un outil très utilisé dans la mise en page est la tabulation, permettant une présentation soignée et bien ordonnée de données tabulaires ou de listes et de séries d'exemples alignés ou de simuler un texte en deux colonnes avec correspondance face à face. Ni le HTML ni par conséquent les CSS ne gèrent les tabulations. donc le placement d'un élément de paragraphe à un endroit fixe dans le bloc ou la page. En général, quand on convertit un fichier Word en HTML, les tabulations sont transformées en une série d'espaces insécables qui ne peuvent garantir un placement et alignement précis des éléments. Certaines astuces permettront de simuler le comportement d'une tabulation en attribuant à l'élément décalé un positionnement fixe dans la page. Mais c'est là une méthode de mise en page avancée, qui nécessite de bien isoler et baliser à l'avance la partie « tabulée » du paragraphe, donc en quelque sorte de mélanger mise en forme et contenu. Il faut donc essayer d'utiliser un découpage cohérent de l'information comme dans le cas suivant :

Nom : **Comestor**

Prénom : **Petrus**

Où on prendra soin de bien baliser les éléments « étiquette » « et contenu » :

```
<p><span class="etiquette">Nom</span>
<span class="nom">Comestor</span></p>
```

```
<p><span class="etiquette">Prénom</span>
<span class="prenom">Petrus</span></p>
```

avec les styles CSS suivants :

```
.etiquette {}
```

```
.prenom {position:absolute; left:100px; font-weight: bold;}
```

```
.nom {position:absolute; left:100px; font-weight: bold; font-variant:small-caps;}
```

La partie correspondant au nom sera décalée de 100 px par rapport au bord de la page web ou de l'élément positionné qui le contient (ici le paragraphe)

[\[voir page de test\]](#)

La gestion de bloc de texte en deux colonnes n'est pas encore possible : elle doit être implémentée dans la future version CSS 3<sup>14</sup> avec continuité du texte d'une colonne à l'autre. Tous les éléments de la mise en page multi-colonnes de la PAO seront utilisables : gouttières, bordures, espaces. Un titre courant pourra également s'étendre sur 2 ou 3 colonnes ; un bloc (exemple d'une illustration) pourra venir en habillage entre deux colonnes. On peut se demander si ces possibilités, qui rapprochent le web de la mise en page « physique » d'un magazine ou d'un livre, sont bien adaptées à l'écran, que l'œil ne parcourt pas de la même façon qu'une page ou qu'une affiche. Il peut paraître illusoire de vouloir retrouver des habitudes de lecture « papier » sur un autre média. Par contre, cette utilisation des colonnes permettra d'obtenir des mises en page pour l'impression très élaborées qui n'auront presque plus rien à envier au PDF distribué sur le web, simulant la « photocopie » numérique d'une mise en page papier.

## Polices spécifiques : le cas des langues anciennes

Une feuille de style permet de déclarer différentes polices (ou famille de polices). Nous avons vu précédemment que les polices communes à toutes les plates-formes sont peu nombreuses. La précision d'une police particulière

et rare risque de ne pas donner de bons résultats. Au-delà des aspects esthétiques, cette question s'avère cruciale pour la mise en ligne d'articles d'historiens médiévistes ou antiquisants. Le plus souvent, ceux-ci ont recours à des polices très spécifiques pour donner à lire des langues anciennes : grec ancien polytonique, hébreu, arabe, copte, éthiopien, langues nordiques, etc. Ces polices rares ne sont utilisées que par des spécialistes, et souvent chaque chercheur utilise une police différente qu'il est souvent obligé de fournir à la revue avec son article afin d'obtenir le résultat espéré à l'impression. Préciser ce type de polices dans la feuille de style est possible, mais rien ne garantit qu'elle sera trouvée sur la machine de l'utilisateur. Souvent ces polices ont des jeux de caractères différents : un texte codé en Graeca ne rendra pas la même chose en Supergreek, par exemple. L'astuce qui consiste à donner une liste longue de polices proches, en espérant que l'une d'elles pourra se trouver sur le système client, reste très aléatoire.

```
.textegrec {font-family: Supergreek, Sgreek, greek}
```

On peut aussi utiliser des polices UNICODE dans la feuille de style, à condition que la page HTML liée soit déclarée dans ce jeu de caractère [15](#) :

```
.textegrec { font-family: "Arial Unicode MS", "Galatia SIL", "Palatino Linotype", "Galilee Unicode Gk", "Athena Unicode", "Gentium", "Porson", "Cardo", "Caslon", "Monospace", "jGaramond", "MgOldTimes UC Pol", "Alphabetum", "Vusillus Old Face", "Code2000", "Titus Cyberbit Basic"; }
```

Le codage des caractères sera identique dans chaque police. La limite de ce système est dans le type de police UNICODE installé chez le client : Arial Unicode MS n'est pas systématiquement installé sur les machines, les autres polices listées ci-dessus sont assez rares. De plus, l'UNICODE n'est pas utilisé systématiquement chez les professionnels et ne propose **pas** le support de toutes les langues anciennes. Dans l'avenir, CSS 3 permettra de spécifier le téléchargement d'une police précise (ou pour alléger le transfert des données le jeu de caractères utilisé dans la page), l'inconvénient d'une telle méthode demeure le poids total d'une police allant de 100 Ko jusqu'à quelques Mo. Des solutions de polices dynamiques : [Portable Font Ressources (PFR), embedded fonts ou WebFonts] existent déjà mais sont encore mal supportées par les navigateurs, nécessitent souvent une technologie de serveurs particulière, l'utilisation d'un produit commercial, voire l'ajout d'un plug-in ou Active-X dans le navigateur et sont de ce fait très peu utilisées.

## Mettre en page avec les CSS

### Mettre en page pour l'écran

#### Les blocs ou ensembles de paragraphes

Le HTML depuis sa version 4 permet de regrouper des ensembles d'éléments (paragraphes, images, tableaux) dans des blocs (balise <div>). Ces blocs sont parfois appelés de façon impropre calques ou layers, de façon à rappeler l'univers de la PAO et de ses logiciels. L'un des atouts de ces balises, si elles sont utilisées avec parcimonie et cohérence est de structurer un document par de grands ensembles de façon sémantique. [16](#) Par exemple un article pourra être structuré en quatre grandes parties :

1. Entête de l'article : sommaire, résumés, mots clés, présentation de l'auteur
2. Le corps de l'article (titres, paragraphes, tableaux, images)
3. Les notes de bas de page
4. Les annexes (bibliographie, documents annexes, glossaires, etc.)

La balise DIV, qui peut recevoir des attributs de nom et de style, a été prévue pour être formatée par les CSS. On obtient à partir de l'exemple précédent la structure suivante :

```

<div id="article" class="compterendu">
<div id="entetearticle"> </div>
<div id="corpsarticle"> </div>
<div id="notes"> </div>
<div id="annexes"> </div>
</div>

```

Ici l'attribut ID précise le nom unique de l'élément, class désigne le style particulier. La première balise DIV regroupe l'article entier, il est de style « compte rendu ». Chaque élément DIV suivant va regrouper l'ensemble des autres informations : le DIV des notes va regrouper l'ensemble de chaque paragraphe de note, l'entête regrouper un paragraphe de résumé, un paragraphe de mots clés, etc.

```

<div id="article" class="compterendu">
<div id="entetearticle">
    <h1>Titre de l'article</h1>
    <p id="auteur"><p>
<p id="resume"><p>
    <p id="motscles"><p>
</div>
<div id="corpsarticle">
    <h2>Titre de niveau 2</h2>
<h3>Titre de niveau 3</h3>
<p>Paragraphe</p>
<p>paragraphe </p>
<blockquote>Bloc de citation</blockquote>
</div>
<div id="notes">
    <p id="note1"></p>
    <p id="note2"></p>
</div>
<div id="annexes">
    <p class="annexe"></p>
    <p class="annexe"></p>
    <table>Tableau</table>
    <div id="biliographie">
        <p class="biblio"></p>
        <p class="biblio"></p>
    </div>
</div>

```

</div>

</div>

La feuille de style CSS permettra de formater cet ensemble d'éléments hiérarchisés, sans avoir à modifier la structure de la page HTML qui demeure avant tout une organisation intellectuelle de l'information.

Voici très rapidement présentés les différents moyens de mettre en page ces blocs de structure :

Habillages de blocs : possibilité de faire « flotter » un paragraphe, une image ou un bloc dans un autre.

float : left, right

Taille de blocs : width et height, avec des valeurs en pourcentage ou en unité de valeur

Couleurs et image et fonds : background-color et background-image, avec pour cette dernière propriété différents attributs permettant de positionner l'image

Bordures: border, border-style, border-color, border-width.

Marges internes et externes : margin et padding

## **De la page au navigateur : différences entre la mise en page d'un article et sa mise à l'écran**

La mise en page d'articles ou de monographies historiques et philologiques répond à des contraintes et habitudes bien définies : notes en bas de pages le plus souvent, références bibliographiques détaillées dans les notes, mise en page sobre et rigoureuse. Nous allons détailler dans cette partie ce qui différencie la mise en page traditionnelle papier d'une mise en ligne d'un article, les contraintes de celle-ci mais aussi les nouveautés.

**La notion de page :** Elle disparaît à l'écran, au profit du défilement à l'écran d'un document entier. La mise en page doit s'imaginer différemment, en considérant l'ensemble de l'article et de sa structure, pour la positionner à l'écran, en terme d'ergonomie (confort visuel, hiérarchie) et de visibilité (trouver rapidement l'information sur l'écran). Plutôt que de mise en page, on devrait parler de mise à l'écran. Le terme anglais de webdesign reste trop vague et recouvre des métiers différents (graphiste, infographiste, webmaster, etc.). Il ne faut pas oublier qu'un article ou un document doit toujours être contextualisé dans le dossier, la rubrique et le site auquel il appartient (menus de navigations, chemins, etc.) La mise en page ou mise à l'écran d'un article doit toujours prévoir de s'insérer dans un ensemble plus général : celui de la maquette du site. Avec les feuilles de style, il est facile de rendre indépendant la structure du document de son affichage : selon les usages, les matériels (ebooks, PDA, écran d'ordinateur, page imprimée comme nous le verrons plus loin). C'est là un grand avantage par rapport au monde de l'édition papier où un document est préparé spécifiquement en fonction d'un format prédéfini de page.

**Les notes de bas de page :** on considère le plus souvent un article comme une page web, un unique document HTML. Les notes sont le plus souvent renvoyées en fin d'article. La notion de note de bas de page disparaît. Pour faciliter les renvois entre texte et appareil critique, l'usage a imposé les liens hypertexte entre appel de note et paragraphe de note, liens fonctionnant dans les deux sens, pour naviguer facilement entre des éléments visuellement éloignés. Les CSS permettent de transformer ces notes en fin de document en notes marginales en définissant le positionnement du bloc de notes (ensemble des paragraphes de notes) en bloc latéral. Le format d'un écran d'ordinateur « à l'italienne » autorise plus naturellement ce type de positionnement, contrairement à une page de livre de format vertical. Une autre possibilité est d'afficher dynamiquement chaque note dans la marge ou au-dessus du texte à la demande. L'inconvénient de cette méthode est d'insérer un Javascript dans le corps du texte au niveau de l'appel de note, ce qui contredit le principe de séparation de la forme et du contenu. Ceci pourra être résolu dans les prochaines versions de CSS en intégrant des éléments Javascript dans la feuille de style.

## Mettre en page pour l'impression

L'un des grands intérêts des CSS pour l'édition est la possibilité des mises en page différentes pour l'écran et pour le papier à partir d'un seul document HTML. Le principe en est simple : il faut attacher deux feuilles de styles à la page HTML, avec des propriétés « média » différentes :

```
<link rel="stylesheet" type="text/css" href="feuille_de_style_pour_ecran.css" media="screen" />
```

```
<link rel="stylesheet" type="text/css" href="feuille_de_style_impression.css" media="print" />
```

« screen » indique la feuille de style écran, « print » celle pour l'impression. Il existe également d'autres types de médias : vocal, braille, ordinateur de poche, TV, vidéo-projecteur, téléphone portable, etc.

L'utilité des feuilles de style d'impression est cruciale pour l'édition scientifique en ligne : les articles sont souvent longs, riches d'informations et complexes, se prêtant assez mal à la lecture prolongée à l'écran. L'usage est souvent d'imprimer l'article pour le lire sur papier, l'annoter, surligner les passages importants, etc. Il faut donc soigner la feuille de style d'impression pour fournir un document agréable à lire sur le papier.

Par rapport à la feuille de style pour l'écran, l'idée est d'imprimer uniquement les informations de la page relatives à l'article : titres, paragraphes, images, annexes, en supprimant les éléments de navigation du site, inutiles à la lecture papier. On se sert de la propriété CSS `display: none;` que l'on va attribuer aux éléments de page que l'on veut voir disparaître à l'impression. Sans détailler trop en avant toutes les techniques CSS pour l'impression [17](#), voici quelques éléments de base incontournables :

- **Les couleurs** : autant les couleurs servent à structurer l'information à l'écran (au-delà bien sûr des effets de mode et d'esthétique) autant il faut les éviter à l'impression, surtout par économie (les cartouches d'encre sont coûteuses : inutile de faire supporter financièrement à l'utilisateur final les facéties chromatiques du webdesigner) mais aussi parce qu'elles n'apportent rien à la lecture. On pourra tout de même garder 2 ou 3 couleurs pour les titres de niveau 1 et 2). Toutes les couleurs d'arrière-plan, (ainsi que les images de fond qui seront supprimées) de bordures doivent être converties en blanc le plus souvent possible pour les mêmes raisons.
- **Les polices** : Il faut convertir les tailles de polices, fixées le plus souvent en pixels ou en *em* pour l'écran en taille de point (10 pt, 12 pt) qui demeure l'unité de référence pour l'impression. On utilisera des polices sérif plus élégantes et plus agréables à lire sur papier : Times, Times New Roman, Garamond, etc.
- **Les marges et espaces** : Convertir les unités en millimètres, et trouver un compromis entre utilisation maximale de la page (économiser le papier) et mise en page aérée agréable à l'œil.
- **Impression des URL** : Parfois, pour plus de lisibilité, par économie de place, les adresses URL des liens hypertextes ne sont pas affichées de façon complète à l'écran (lien du type : « en savoir plus sur le site de la BnF » ou « article de M. Dupont dans ce même numéro ») Un utilisateur imprime souvent des documents pour les archiver et sera content de retrouver plus tard des adresses web complètes dans l'article. Une propriété CSS permet d'imprimer les URL  
`a:after {content: " (" attr(href) ") "};`  
Ici l'adresse complète sera imprimée après (after) le lien défini dans la balise de lien a. Attention : cette fonctionnalité n'est pas encore prise en compte dans Internet Explorer (qui couvre aujourd'hui 95 % du marché...)

**Gestion de la page** : dans certaines techniques avancées de mise en page, il est possible de gérer les paragraphes en fonction des fins et débuts de pages (lignes veuves et orphelines, paragraphes solidaires, saut de pages conditionnels, etc.) Bien évidemment, cela n'a pas de sens dans une configuration écran. Mais dans le cadre d'une feuille de style dédiée à l'impression des documents cela a toute son importance. Malheureusement les propriétés CSS gérant la page [18](#) à l'impression ne sont pas encore respectées par les navigateurs, même les plus récents. L'une des rares propriétés supportées est le saut de page conditionnel, ici avant un titre de niveau 2 :

Les exemples et principes rapidement présentés ici n'excluent pas une sophistication plus grande, pour rapprocher la mise en page « web – papier » d'une belle mise en page de revue savante ou de magazine. Là encore, les futures versions de CSS permettront beaucoup d'enrichissements typographiques, l'insertion de polices plus élégantes, une gestion de page avancée. Pour conclure cette partie, on peut même conjecturer, que plus encore pour l'affichage et la mise à l'écran, les CSS révolutionneront l'impression des documents en ligne, fournissant à terme une alternative efficace au format PDF qui reproduit fidèlement, comme une photocopie numérique, une mise en page élaborée prévue pour le papier (polices intégrées, positionnement des éléments, etc.), dans un respect total cette fois des standards du web, indépendamment d'une technologie « propriétaire » qui nécessite l'utilisation d'un plug-in ou greffon dans le navigateur web.<sup>19</sup>

## Conclusion et perspectives

L'utilisation des feuilles de style se généralise actuellement sur le web, que ce soit pour mettre en forme des pages web statiques (séparation de la forme et du contenu de pages HTML) ou pour des sites dynamiques (bases de données, logiciels de gestion de contenu). Nous avons vu tout l'apport des CSS pour l'édition en ligne d'articles historiques et scientifiques, mais qui nécessitent une meilleure structuration à la base des documents à publier. En ce sens, les CSS deviennent le complément idéal des futures publications électroniques structurées, interopérables, considérées comme des ressources scientifiques partageables et distribuées. Dans l'avenir, la mise en forme des articles (typographie, mise en page, normes rédactionnelles de présentation) ne sera plus qu'un élément de la chaîne documentaire et éditoriale de la production de l'écrit scientifique.

L'utilisation des CSS pour un meilleur contrôle de l'impression des articles sur le web pourra fournir une véritable alternative au format PDF, trop souvent utilisé aujourd'hui par facilité ou pour fournir une « photocopie » en ligne d'un document déjà édité pour le papier. Le PDF est d'ailleurs massivement utilisé pour la rétro numérisation de livres anciens ou de revues, comme sur le site Gallica <sup>20</sup> de la BnF ou sur le site PERSEE <sup>21</sup>. Notre propos a été plutôt de situer l'enjeu de l'utilisation des CSS pour les articles mis en ligne au format HTML, dans une perspective d'avenir où de nombreuses revues n'auront plus qu'une existence électronique, comme aujourd'hui le *Médiéviste et l'ordinateur*. Dans ce cadre, les CSS proposent une alternative d'avenir au format PDF, même si un contrôle aussi poussé que le PDF sur l'impression ne sera sans doute jamais possible car le format d'Adobe se comporte en fait comme un véritable pilote d'impression. Mais l'utilisation des CSS dès aujourd'hui est une garantie pour l'avenir : grâce à la séparation forme contenu, il sera toujours possible d'enrichir les styles typographiques et de mettre en page avec les toutes nouvelles propriétés qui ne manqueront pas d'être ajoutés aux futures versions de normes de feuilles de style.

Un autre avantage des CSS, qui mériterait d'être développé dans la consultation d'article à l'écran, est leur possibilité de spécifier des qualités d'affichages à des éléments : afficher ou pas un menu, un sommaire, à la demande du lecteur <sup>22</sup>, alors que le code HTML de base est une simple liste à puces. De même pour les notes. D'une façon plus générale, l'utilisation de feuilles de style alternatives peut permettre à l'utilisateur de personnaliser sa lecture à l'écran, selon ses habitudes ou ses capacités visuelles.

## Webographie

Eric Costello, Glish, <http://www.glish.com/css/>

Openweb, <http://openweb.eu.org/>

W3C : <http://www.w3c.org/style/css/>

MaxDesign : <http://css.maxdesign.com.au/>

Douglas Bowman, Stop Design, <http://www.stopdesign.com/>

Css Zen Garden : <http://www.csszengarden.com/tr/francais/>

Eric Meyer : <http://www.meyerweb.com/eric/css/>

Alsacreation: <http://www.alsacreations.com/articles/>

CSS Vault; <http://cssvault.com/>

SelfHTML, <http://fr.selfhtml.org/css/index.htm>

Liens sur les CSS : <http://www.dezwozhere.com/links.html>

Support selon les navigateurs des différentes propriétés css : <http://www.csscreator.com/attributes/>

 [retour haut](#) [retour haut de page](#)

## Notes

1 <http://www.w3.org/Style/CSS/>

2 Sur la différence entre CSS et XSL : <http://www.w3.org/Style/> et <http://www.w3.org/Style/CSS-vs-XSL>

3 Voir la page personnelle de H. K. Lie : <http://www.w3.org/People/howcome/> et son premier article sur les feuilles de style en cascade : « Cascading style sheet. A proposal », 1994, *site web du W3C*, [En ligne] <http://www.w3.org/People/howcome/p/cascade.html>

4 Certains ont pu parler à ce propos de « balkanisation du web » : cf. Tristan Nitot, « La fin de la balkanisation du web », *Openweb*, 2003 [En ligne] <http://openweb.eu.org/humeurs/balkanisation/> À propos des frames (cadres) voir aussi Denis Boudreau, « Pour en finir avec les cadres », *Openweb*, 2003 [En ligne] [http://openweb.eu.org/articles/finir\\_cadres/](http://openweb.eu.org/articles/finir_cadres/) et Mathieu Pillard « Les problèmes de la mise en page par tableaux », *Openweb*, 2003 [En ligne] [http://openweb.eu.org/articles/problemes\\_tableaux/](http://openweb.eu.org/articles/problemes_tableaux/)

5 Le principal conseil reste de tester toute mise en page élaborée sur les principaux navigateurs et sur différentes plates-formes (Windows, MAC, Linux) afin de trouver le meilleur compromis : une mise en page élaborée ne s'affichera pas exactement de la même façon partout...

6 Informations sur les CMS disponibles en Open Source : <http://www.opensourcecms.com/> ; liste commentée de logiciels libres de CMS : <http://www.framasoft.net/rubrique168.html>. Il existe un grand nombre de CMS avec des fonctions différentes : édition, blog, portail, forum, tutoriel, galerie d'images, etc.

7 <http://www.w3.org/TR/xhtml1/>

8 Laurent Denis, « Le XHTML », *Openweb*, 2003 [En ligne] <http://openweb.eu.org/xhtml/>

9 L'exemple le plus saisissant de cette souplesse de modification de la mise en page d'un site nous est donné par CSS Zen Garden (<http://www.csszengarden.com>). Ce site propose aux webdesigners volontaires de modifier l'apparence de sa page d'accueil en ne changeant que la feuille de style. Des centaines de versions différentes sont consultables en ligne permettant de se rendre compte visuellement de la souplesse et de la puissance des CSS (et de l'imagination des participants également...)

10 Pour tous ces aspects, consulter dans ce même dossier l'article de Gautier Poupeau « L'édition électronique change tout et rien », notamment sa partie II « Permettre l'exploitation de l'information » <http://lemo.irht.cnrs.fr/43/43-03.htm#titreDyn96>

11 Signalons ici un logiciel libre comme OpenOffice (<http://www.openoffice.org>) qui propose les mêmes fonctionnalités que le logiciel commercial de Microsoft.

12 Certains sites universitaires proposent à leurs étudiants utiliser un modèle de document Word : Cid-Cédille (Cellule d'édition et de diffusion des initiatives en ligne sur l'École et sur l'extérieur) service de l'ENS – Lettres et Sciences humaines, propose sur son site deux modèles de documents pour aider à mettre en ligne un mémoire universitaire : thèse.dot et cid-cedille.dot (avec sa feuille de style css) : [http://cid-cedille.ens-lsh.fr/rubrique.php3?id\\_rubrique=16](http://cid-cedille.ens-lsh.fr/rubrique.php3?id_rubrique=16). Voir aussi les thèses électroniques de Lyon II (plate-forme SDX) : <http://demeter.univ-lyon2.fr:8080/sdx/theses/fds.shtm>. Enfin le logiciel d'édition en ligne LODEL (<http://www.lodel.org>) utilisé par Revues.org (<http://www.revues.org>) s'appuie sur un modèle de document Word <http://sourceforge.net/projects/lodel/>

13 Nous renverrons aux différents sites décrivant les CSS pour une description plus précise des propriétés CSS. Cet article ne se veut pas une initiation ni un cours.

14 <http://www.w3.org/TR/2001/WD-css3-multicol-20010118/>

15 <http://www.weblettres.net/languesanc/index.php?page=grec1>

16 <http://www.w3.org/TR/html4/struct/global.html#h-7.5.4>

17 Quelques articles de référence en ligne : Laurent Denis, « Une feuille de style pour l'impression », *Openweb*, 2003 [En ligne] [http://www.openweb.eu.org/articles/css\\_impression/](http://www.openweb.eu.org/articles/css_impression/) ; Eric Meyer, « CSS Design : Going to Print », *Alistapart*, 2002 [En ligne] <http://alistapart.com/articles/goingtoprint/>, et sa traduction en français : Samuel Lachtman, « Faites bonne impression avec les css », *Pompage.net* [En ligne] <http://pompage.net/pompe/impression/>

18 <http://fr.selfhtml.org/css/proprietes/printlayouts.htm>

19 Le format PDF a un double inconvénient : le document produit est en général plus volumineux que son équivalent HTML, donc plus long à télécharger ; de plus le greffon Acrobat Reader, dans ses versions récentes est parfois long à s'ouvrir, cet affichage dépendant énormément de la puissance du PC ou du MAC du lecteur.

20 <http://gallica.bnf.fr/>

21 <http://persee.univ-lyon2.fr> qui regroupe 7 revues de références en sciences de l'homme (Annales, BEC, L'Homme, Revue économique, Revue de l'art, Revue française de sciences politiques, Matériaux pour l'histoire de notre temps). Les numéros ont été numérisés en mode image, mais les données textuelles sont extraites par reconnaissance de caractères pour permettre une recherche en texte intégral. [Voir l'article d'E. Lalou dans ce même numéro.](#)

22 Comme par exemple sur le site Wikipedia, où les sommaires des articles s'affichent ou se masquent par click : <http://fr.wikipedia.org/wiki/CSS>

© CNRS - IRHT 2004 - [Contacts](#) - [Infos légales](#) - [Rédaction](#)