

CSS2 OU GUTENBERG RETROUVÉ

LAURENT.Kling@epfl.ch, STI



INTRODUCTION

L'éditeur d'une page Web est confronté au double défi de fournir l'information adéquate sous l'aspect le plus agréable pour le lecteur.

Je propose trois réponses à cette problématique:

- une feuille de style adaptée aux contenus existants;
- une taille variable pour l'affichage et l'impression;
- une diminution du temps d'accès aux pages Web.

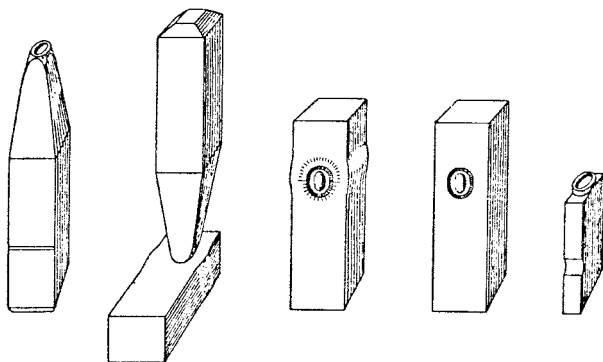
HISTORIQUE

L'impression est souvent comprise comme la possibilité de reproduire en de nombreux exemplaires un objet manufacturé; il est paradoxal que le premier livre ne soit pas reproductible (en dehors de son édition originale).

Une visite au musée de l'imprimerie à Mayence, ville natale de Gutenberg, permet de contempler un des trésors de notre civilisation, la Bible à 42 lignes. La vision d'un de ces exemplaires me renvoie au choc d'un tableau de Piet Mondrian. Dans les deux cas, l'original dépasse largement toutes les tentatives de reproduction de l'œuvre initiale.

Gutenberg est généralement associé à l'invention de l'imprimerie; son apport le plus important me semble être l'invention de la typographie.

Point de départ du processus, le caractère est d'abord ciselé dans un poinçon d'acier. La frappe de celui-ci sur une matrice en cuivre permet d'obtenir une empreinte en creux de la lettre. La fusion d'un alliage de plomb et d'antimoine permet la création du caractère typographique.



Du poinçon au caractère

Les efforts sont démultipliés, un poinçon source de nombreuses matrices, d'une matrice, on coule de nombreux caractères et d'un caractère, on crée de nombreuses compositions, elles-mêmes sources de nombreuses impressions.

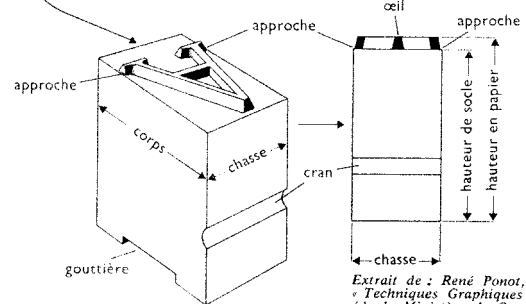
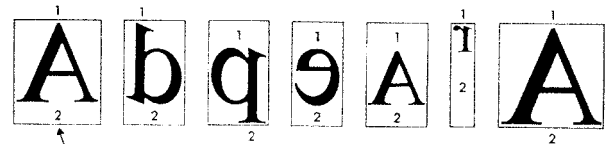
Pour parvenir à son objectif de perfection, et assurer une qualité identique à un manuscrit calligraphié, Gutenberg a multiplié les variantes dans son jeu de caractères. Un alphabet de 26 lettres est transformé en 281 caractères différents.



LES TYPES DE GUTENBERG INVENTORIÉS PAR EMIL RUDER DANS TYPOGRAPHIE, ARTHUR NIGGLI VERLAG, TEUFEN AR, 1967

Ainsi, le caractère de plomb contient en lui-même l'ensemble des informations nécessaires pour une utilisation harmonieuse:

- la largeur du caractère détermine l'approche;
- la hauteur du caractère définit son interligne;
- les approches de paires, ou les ligatures sont définies par des caractères spéciaux.



Extrait de : René Ponot, « Techniques Graphiques », éd. du Ministère de Commerce et de l'Artisanat.

UN CARACTÈRE DANS LE DÉTAIL

Une autre qualité de la méthode est, en juxtaposant des caractères, de parvenir à une composition justifiée avec une qualité graphique très élevée.

En final, l'encrage de la matrice de caractères, l'imposition de la feuille de papier et la mise sous presse fournissent une épreuve imprimée.

Dans le cas de l'impression au plomb, le chef-d'œuvre a précédé l'utilisation quotidienne. Pour les médias électroniques, et en particulier le Web, le respect des normes de facto issues de l'industrie entraîne un aspect où la qualité graphique et la lisibilité ne sont souvent pas les premiers critères de conception.

CSS2, LE MONDE À L'ENVERS

La feuille de style en cascade (*Cascading Style Sheet*) doit permettre de séparer l'apparence du contenu.

Jusqu'à récemment, l'utilisation des feuilles de style s'apparentait à une course d'obstacles. Pour garantir et préserver des parts de marché, les constructeurs de butineur, navigateur ou browser Web se sont ingénies à inventer des éléments spécifiques. Le résultat: une incompatibilité presque parfaite!

Une solution consistait à générer des feuilles de styles propres à chaque environnement, cette méthode a été utilisée par l'école jusqu'à la fin de l'année dernière.

La plate-forme hégémonique (PC Windows) ayant retrouvé des parts de marché suffisantes, Internet Explorer en version 6 respecte les normes et permet ainsi la création d'une feuille de style unique.

Le point de départ de ce travail est la nouvelle feuille de style réalisée par Natalie Meystre dont un article décrit le travail: [ditwww.epfl.ch/publications-spip/article.php3?id_article=849](http://www.epfl.ch/publications-spip/article.php3?id_article=849).

UN ENVIRONNEMENT DE TRAVAIL

Pour assurer une compatibilité maximale, il est utile de mettre en place un environnement de développement qui va limiter le nombre d'aller-retour entre le codage et ses tests.

Dans mon cas, il se décompose en:

- un logiciel d'édition Web, Golive CS 2 d'une part;
- une batterie de butineurs sur la plate-forme de travail Mac OS X (Safari, Internet Explorer Macintosh 5.2, Firefox et Opéra);
- une batterie similaire sur une machine virtuelle sous Windows 2000 (Internet Explorer 5, IE 5.5, IE 6, Firefox et Opéra);
- et élément précieux, une barre d'outil dans Firefox, WebDeveloper.

Trois éléments demandent des précisions:

GOLIVE CS 2

Je recommande aux usagers de produits concurrents de tester les fonctionnalités suivantes:

- **un éditeur CSS2 graphique.** Enfin, il est possible d'éditer du CSS2 en mode graphique!
- **un moteur de recherche pour les styles.** Cette possibilité est particulièrement intéressante pour la migration de contenu, en particulier pour la migration d'ancien contenu (évitte de devenir un expert en GREP).

PLUSIEURS VERSIONS D'INTERNET EXPLORER

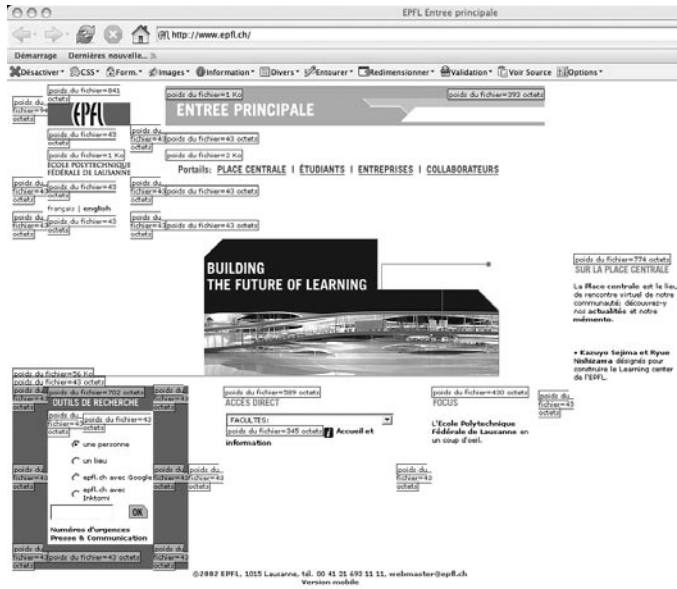
Cette contrainte n'est pas aussi triviale que ça; l'intégration étroite d'Internet Explorer avec son système d'exploitation Windows empêche normalement l'utilisation simultanée de versions différentes d'IE.

Cette limitation est fâcheuse en regard des différences majeures de rendu entre la version actuelle, IE 6 et ses prédécesseurs (IE 5 & 5.5). Heureusement, ce

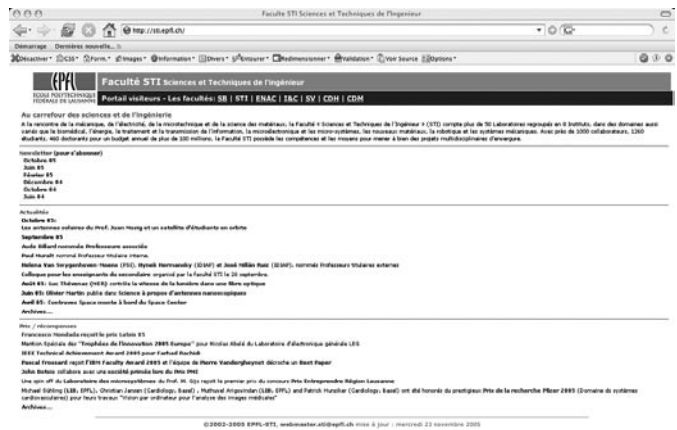
problème peut être résolu: <http://labs.insert-title.com/labs/article795.aspx>.

FIREFOX & WEBDEVELOPER

Si *Firefox* connaît un succès croissant, c'est probablement que son implémentation est particulièrement réussie. Son principal avantage dans ce travail est la possibilité de bénéficier d'un grand nombre d'extensions propres à faciliter le travail. *Chris Pederick* a certainement réalisé l'extension la plus utile pour un éditeur de page Web, *WebDeveloper*: <http://chrispederick.com/work/WebDeveloper/>.



IMAGES > AFFICHER LE POIDS DES IMAGES



CSS > AFFICHER LES CSS PAR TYPE DE MÉDIA > PRINT



ENTOURER > ENTOURER LES CELLULES DU TABLEAU

La palette de fonctions étant particulièrement complète, les éléments précédents méritent votre attention.

Pour la validation, WebDeveloper offre l'analyse dans un nouvel onglet du navigateur. Le dernier élément de la barre d'outils n'est pas le moins intéressant, car il affiche le mode de compatibilité de la page!

DANS LA TOURMENTE

Après quelques essais infructueux et des résultats incohérents, un recours à des sources d'informations fiables est indispensable:

- <http://www.alistapart.com/>
C'est le site de référence, la quasi-totalité de la méthodologie utilisée est issue de cette source. À lire absolument.
- <http://pompage.net/>
C'est la traduction en français d'articles de référence.

UNE FEUILLE DE STYLE ADAPTÉE AUX ANCIENS CONTENUS

C'est le point de départ de ce travail, la nouvelle feuille de style doit être compatible avec les pages statiques déjà existantes.

TRAVAILLER EN MODE STRICT

Les butineurs peuvent interpréter le contenu selon deux approches:

- **Le mode de compatibilité.** Dans ce mode, le comportement de l'affichage du navigateur suit le mode de fonctionnement historique. Cela signifie que les astuces (ou *hacks*) sont nécessaires pour obtenir un résultat satisfaisant. En pratique, ce mode ne respecte pas la norme CSS2. Pour éviter une réécriture massive des sites existants, il est activé par défaut.
- **Le mode strict de respect des standards.** C'est le mode qui correspond au standard de CSS2, il est activé uniquement par la présence d'un en-tête conforme aux spécifications du Web, pour un fichier en HTML transitoire il doit être:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

Cet en-tête ouvre la porte d'un travail cohérent, son absence entraîne des comportements erratiques des différents butineurs.

Par exemple, une erreur mineure, comme un numéro de version erroné suffit pour rendre incompatible un en-tête:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

Le bon code doit être:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

Une page dans le mode compatible et strict offre des aspects différents:

Dans le mode compatible, des incohérences de rendu apparaissent:

- taille de caractères plus grands dans la 2^e partie
- styles des liens.



COMPATIBLE: <http://sti.epfl.ch/css/table-css2-compat/fixe2.html>

Dans le mode strict, l'ensemble de ces incohérences disparaît. L'affichage avec FireFox et WebDeveloper du mode de rendu est particulièrement utile.



STRICT: <http://sti.epfl.ch/css/table-css2-strict/fixe2.html>

QUELLE EST LA BONNE UNITÉ DE MESURE ?

La mesure de la grandeur du texte en CSS2 ressemble à la très grande variété de mesures qui prévalait en France avant la révolution. À cette époque, on pouvait mesurer la longueur en:

- Toise:
 - du latin *tensa*, *étendue*. Elle était d'environ 6 pieds, soit 1,949 m. De 1812 à 1840, la toise métrique était de 2 m.

LIEUE DE PARIS:

- du gaulois *leuca*, *distance entre deux pierres*. Jusqu'en 1674, elle valait 1 666 toises. De 1674 à 1737, elle représente 2 000 toises, soit 3'898 km. En 1737, elle mesure 2 400 toises pour les tarifs du transport de grains, 2 000 pour les Ponts et Chaussées, et 2 200 pour les Postes.

D'après [Thierry Sabot \(http://www.histoire-genealogie.com/IMG/_article_PDF/article_396.pdf\)](http://www.histoire-genealogie.com/IMG/_article_PDF/article_396.pdf)

Si l'action normative de la révolution permet l'émergence du système MKS (mètre, kilo, seconde) et malgré les mesures existantes en typographie (comme le point), les architectes du Web ont réalisé un compendium baroque.

EN PRATIQUE, QUELQUES COMPORTEMENTS OBSERVÉS

Les tailles relatives

Hauteur ou largeur de M, **em**

Hauteur de X, **ex**

Pixel, **px**

- Px, pixel: à bannir ! Ce choix représente une fausse solution, car si un pixel est facilement défini, sa grandeur réelle est relative. En effet, la résolution n'est pas définie, elle peut varier de 72 dpi (dot par inch) pour les anciens moniteurs Macintosh à plus de 2400 dpi pour une impression. En plus, il n'est pas possible d'agrandir le texte avec le navigateur.
- Em, largeur ou hauteur du M: bien, mais... Dans mon article précédent, **Le maillon faible ou les feuilles de styles appliquées**, http://ditwww.epfl.ch/publications-spi/article.php?id_article=226, je recommandais le choix de cette unité pour permettre une modification de la grandeur de l'affichage dans le navigateur. Son comportement peut prêter à confusion dans un tableau.

Les tailles absolues

pouces: **in.**, centimètre: **cm**, millimètre: **mm**, points: **pt**, picas: **pc**.

- Si l'idée des tailles absolues peut sembler la solution, elle échoue sur le même problème que les tailles relatives, la résolution n'est pas définie. Un bémol, cette constatation est vraie pour les navigateurs, par contre pour les imprimantes, la résolution est généralement définie et le résultat est cohérent. Le point peut être utilisé pour une feuille de style d'impression.

Les proportions en pourcentage

pourcentage: %

- Une bonne solution si le point de départ est dans une taille absolue.

Les tailles par mot-clé

xx-small, **x-small**, **small**, **medium**, **large**, **x-large**, **xx-large**.

- Dans ce cas, la police ne possède pas de taille précise connue de l'utilisateur. Cela peut paraître une hérésie, mais en pratique c'est une bonne solution, car le problème des tableaux imbriqués ne se pose pas. La relation entre les différentes tailles est un facteur d'échelle de 1.5. Il est aisé de modifier cette taille fixe par un pourcentage pour obtenir la variation de grandeur souhaitée.

LES PROBLÈMES DES STYLES IMBRIQUÉS

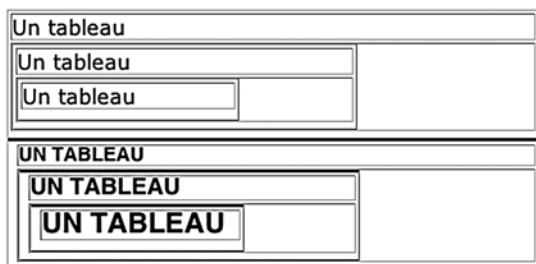


TABLEAU EM

L'utilisation d'une unité absolue évite ce problème:

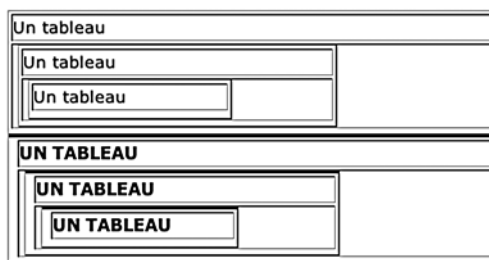


TABLEAU X-SMALL

L'incorporation d'un style imbriqué peut entraîner un agrandissement (ou une réduction) de la taille obtenue avec une unité relative.

L'unité choisie est la taille par mot-clé **x-small**, son utilisation entraîne une taille à l'écran similaire à celle de l'école.

RÉUTILISER LES NOMS DES STYLES ANTÉRIEURS ET LIMITER LES ASTUCES

Le principal problème dans cette réutilisation est qu'il faut se rattacher au nom plutôt qu'au contenu. Cette réécriture doit impérativement prendre en compte le mode de rendu strict, qui modifie profondément l'écriture de la feuille de style.

Par exemple dans la feuille de style utilisé par Jahia du 10 novembre 2005:

```
body { background-color: white; margin: 15px 10px 10px; padding: 0; height: 100% }
table { font-size: 1em; margin-left: 4px; font-family: Verdana, Helvetica, Arial, sans-serif }
.layout{ font-size: 0.6em; font-family: Verdana, Helvetica, Arial, sans-serif }
td, pre{ font-family: Verdana, Helvetica, Arial, sans-serif }
```

on remarque à l'analyse de ce code:

- la redéfinition de la police de caractère dans de nombreux Tags (layout, td, pre..);
- pas de définition de la taille par défaut dans le corps de la page (body);
- une taille plus grande pour les tableaux (1em au lieu de 0.6em)...

Si le néophyte en perd son latin, l'expert aura remarqué que ces apparentes incohérences par rapport aux caractéristiques CSS2 sont en fait des astuces pour que le comportement soit harmonieux dans les navigateurs en mode compatible.

UNE FEUILLE DE STYLE RESPECTANT CSS2

Si l'en-tête est correct, d'où l'importance essentielle de ce paramètre, votre navigateur actuel interprète correctement CSS2. Comme indiqué dans l'article de Natalie, la notion d'héritage est essentielle pour comprendre le fonctionnement de style CSS2. Il ne faut pas hésiter à subdiviser son style en sous-éléments et ne pas répéter une propriété déjà décrite. En mode strict, les héritages sont respectés. En épurant le code superflu, on obtient:

```
body { font-size: small; font-family: verdana, Helvetica, Arial, Geneva, Swiss, SunSans-Regular, sans-serif; margin-top: 15px }
#layout{ line-height: 140%; }
img { border-style: none; border-width: 0; }
table{ font-size: 100%; margin-left: 4px}
th { font-weight: bold }
```

Quelques explications:

- une définition unique de la police du caractère, dans le corps de la page Web (body);
- une définition unique de l'interligne (pour la partie principale de la composition);
- la suppression de l'encadrement des images qui possèdent un lien (pour éviter de le définir pour chaque image);
- pour la table, une astuce de compatibilité pour les anciens navigateurs est présente;
- une mise en évidence pour les titres de tableau.

La limitation des astuces est nécessaire. Il est quasi impossible à un éditeur de site Web de comprendre une feuille de style si elle contient de nombreuses astuces non documentées. Les astuces doivent uniquement assurer une compatibilité avec des versions antérieures de navigateur. Deux astuces sont présentes dans la feuille de style en production:

UNE TAILLE COHÉRENTE DE X-SMALL

Une implémentation erronée de la taille des caractères dans les versions antérieures d'Internet Explorer (IE 5 ou IE 5.5) entraîne une apparence plus grande que celle qui est attendue. Internet Explorer 6 se comporte correctement, comme l'ensemble des autres navigateurs!

Le responsable du développement du moteur de rendu d'Internet Explorer pour Macintosh, Tantek Çelik, a écrit une astuce qui permet de contourner le problème: <http://www.tantek.com/CSS/Examples/boxmodelhack.html>.

Le précédent tag **body** se transforme en:

```
/*hack ie5*/
body { font-family: verdana, Helvetica,
Arial, Geneva, Swiss, SunSans-Regular, sans-
serif; margin-top: 15px; background-color:
#fff;font-size: xx-small; /* for IE5/Win */
voice-family: "\"\"\"";
voice-family: inherit;
font-size: x-small; /* for compliant browsers
*/
}
html>body {/* be nice to Opera */
font-size: x-small;
}
```

L'apparence redevient cohérente entre les versions d'Internet Explorer, mais la lisibilité du code CSS n'est pas améliorée.

La deuxième astuce corrige un bug présent dans les anciens navigateurs par la répétition de la taille de la police héritée dans les tableaux:

```
table { font-size: 100%; margin-left: 4px}
```

Offrir une lisibilité adaptée

Il existe pour les documents imprimés des critères objectifs de lisibilité. Dans son article dans l'encyclopédie de la chose imprimée, édition Retz 1999, François Richaudeau décrit le processus de la lecture, et n'évoque pas de différence notable de lisibilité entre un corps 9 et un corps 12.

La seule différence qu'il relève est que les caractères avec serif (ou empâtement) sont considérés plus lisibles pour les peuples latins. Les caractères sans serif (comme ceux qui sont choisis par l'école) appartiennent à la tradition gothique et sont plus lisibles pour les peuples nordiques.

Une page imprimée possède une résolution élevée (généralement 2400 dpi) avec un procédé de lecture réfléchit (la lumière est réfléchiée par le papier). La qualité du support et la qualité de l'encre déterminent également le confort de l'utilisateur. À chaque support, la taille employée est en rapport avec l'encre et le papier, par exemple l'édition de la Pléiade imprimée sur un papier bible.

Le paradoxe du Web veut que le navigateur adapte le contenu au contenant. La diversité des navigateurs est pratiquement sans limites, la résolution varie fortement entre chaque média électronique utilisé (d'un téléphone portable à un écran de 30 pouces). De ces contraintes, il devient illusoire d'utiliser une taille précise, mais l'éditeur doit réaliser une gamme d'enrichissement typographique:

- utilisation de différentes grandeurs
- utilisation du gras.

UNE TAILLE CHOISIE PAR L'USAGER

La possibilité de modifier la taille de l'affichage dans le navigateur est une aide précieuse pour assurer une lisibilité adaptée aux différents usagers.

Une autre possibilité est d'offrir une solution propre au site que l'on observe. Paul Sowden est le créateur de cette technique: <http://www.alistapart.com/articles/alternatel>. Utilisée par le site de Wired: <http://www.wired.com/>.

UN INTERLIGNE PLUS GRAND

François Richaudeau observe qu'un interlignage aéré offre une plus grande lisibilité.

Par défaut, un navigateur Web utilise un interlignage équivalent à 120 % de la taille du caractère. Pour améliorer la visibilité, l'interlignage a été porté à 140 % et 145 % pour les tailles plus grandes de caractères.

EN PRATIQUE



TAILLE MINIMUM

Trois étapes sont nécessaires pour implémenter cette fonctionnalité:

1. s'assurer que la page Web s'affiche en mode strict;
2. créer les feuilles de style pour les différentes tailles;
3. ajouter le code nécessaire pour que le navigateur puisse tenir compte de ces options alternatives d'affichage.

OPTIMISER LA VITESSE

Offrir le confort le plus élevé pour l'internaute doit être un principe sous-jacent à la réalisation d'une page Web.

Le premier confort est que l'attente de l'information soit réduite au minimum.

Une solution serait de bannir les images et d'offrir un site Web utilisant uniquement du texte. Si cet objectif ne peut être atteint pour les sites de l'école, en particulier par l'incorporation obligatoire du logotype de l'EPFL, il est possible de minimiser le poids des composants à transmettre. L'outil *WebDeveloper* utilise le site *WebSiteOptimisation* pour calculer les rapports de vitesse. <http://www.Websiteoptimization.com/services/analyzel>.

Pour la page Web de la Faculté STI, le résultat est mitigé, en effet la taille globale de 48'102 bytes entraîne un délai de 10 secondes avec un modem 56K.

WebSiteOptimisation n'est pas exempt d'erreurs:

- ne pas tenir compte de la compression des feuilles de style (CSS) et Javascript (JS)
- oublier la feuille de style *dynastyle.php* (entrée et place)
- oublier l'image de garde de la Faculté ENAC.

RÉDUIRE LA TAILLE TRANSMISE AVEC `mod_gzip`

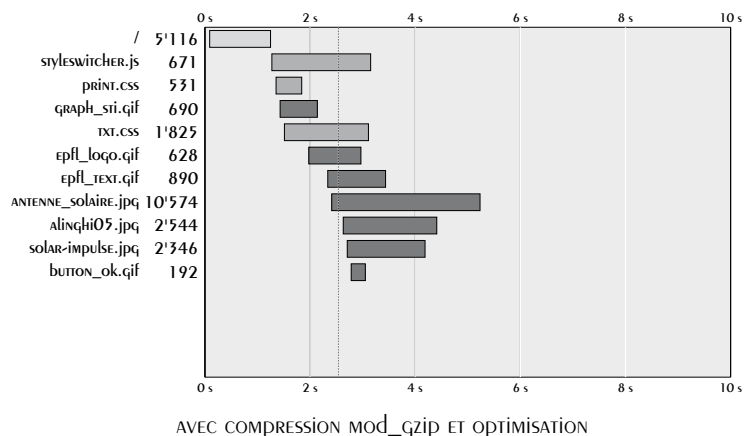
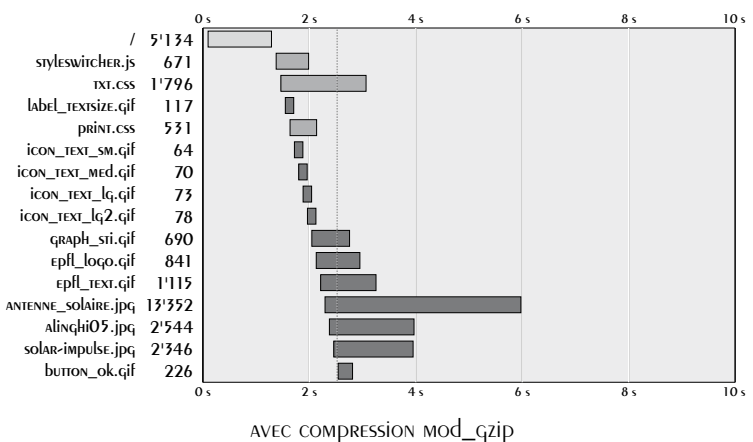
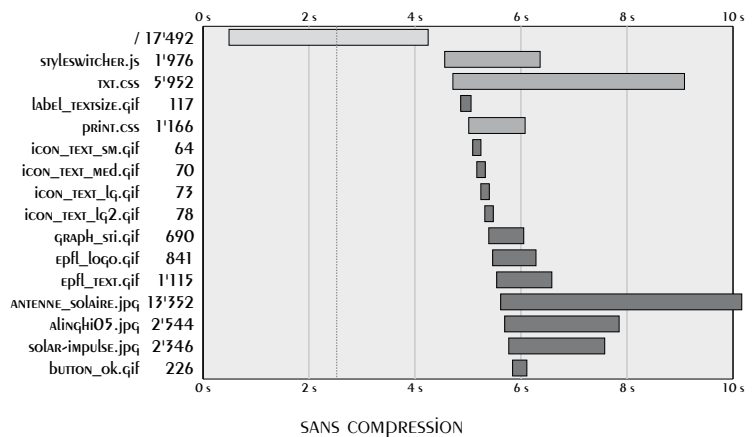
À partir du protocole HTTP 1.1, le navigateur peut signaler au serveur Web sa compatibilité avec un transfert comprimé GZIP, le RFC est disponible depuis mai 1996! <http://www.ietf.org/rfc/rfc1952.txt>.

En pratique, Michael Schröpl a écrit une extension au serveur Apache en automne 2000 qui permet de réaliser ce travail de compression à la volée, `mod_gzip`: http://www.schröpl.net/projekte/mod_gzip/index.htm.

L'intérêt de ce travail est flagrant, en particulier avec la taille grandissante des fichiers HTML et de ses coreligionnaires (CSS, JS). Si la compilation et l'installation du module ne posent pas de problème, la configuration adéquate dans Apache peut poser problème, en particulier pour la compression des fichiers CSS et JS.

En effet, WebSiteOptimisation, tient compte dans son rapport uniquement de la compression du code HTML. Pour les autres fichiers, les tailles sont affichées sans compression. Une entreprise offre un outil intéressant pour l'analyse détaillée de la performance de site Web www.siteconfidence.com. J'ai intensivement utilisé la période gratuite de test d'une semaine pour réaliser l'optimisation de *sti.epfl.ch*.

L'affichage graphique de l'estimation de la durée de transfert avec un modem de 56k est particulièrement utile. Dans ce cas, la durée de téléchargement est de 10.1 secondes avec une taille de 48'102 octets.



L'intérêt de la compression est facilement démontrable, la durée de téléchargement est passée à 6.2 secondes pour 29'648 octets.

POUR ALLER ENCORE PLUS LOIN

On pourrait se contenter de ce résultat, mais l'analyse du graphique fait apparaître le coût prohibitif engendré par les images, même de petite taille. Sur WebSiteOptimisation, le délai de l'obtention d'un fichier est fixé à 0.2 seconde. Les mesures suivantes d'optimisation ont été prises:

- **Limiter le nombre de fichiers:** la suppression des images GIF pour la sélection de la taille du texte économise 5 fichiers. Ces images sont remplacées par du texte mis en forme par la feuille de style.
- **Limiter le nombre des couleurs dans les fichiers GIF:** la limitation du nombre de couleurs utilisées par un fichier GIF permet de limiter la taille transmise.

- **Favoriser l'élément central de l'image:** cette astuce utilise le principe même de la compression JPEG. Comme le réglage de la profondeur de champs en photographie, elle consiste à rendre flou le fond d'une image pour favoriser le premier plan. Cette manipulation offre le double avantage d'une taille réduite du fichier JPEG et la mise en valeur du sujet principal. Le site Web **WebsiteOptimization** contient l'explication détaillée. <http://www.Websiteoptimization.com/speed/tweak/blur/>
Avec ces mesures, on obtient un gain modeste, mais appréciable:

La page finale possède une durée de téléchargement de 5.5 secondes pour une taille de 26'007 octets.

IL EST UTILE DE COMPARER LES RÉSULTATS

Le point de comparaison ultime me paraît être Google.

On est particulièrement surpris de la différence faible qui existe pour la page principale de Google, 13'130 octets, et celle obtenue après une recherche sur le mot clé STI, 13'316 octets. Ce tour de force est possible par l'absence d'objets externes autres que les 4 images GIF. L'inclusion de la feuille de style dans le fichier HTML évite le retard issu de l'appel de l'objet.

J'ai comparé le site STI avec trois autres sites de l'EPFL, celui de l'ENAC et deux sites statiques, l'entrée dans l'EPFL et sa place centrale. Cette comparaison peut être mise en relation avec un travail similaire réalisé en 2000: http://ditwww.epfl.ch/publications-spip/article.php3?id_article=671.

On constate:

- Google est imbattable, il a remplacé Yahoo;
- la compression avec mod_gzip du site STI permet une réduction spectaculaire des fichiers textes (HTML, CSS, JS). On passe d'un total de 25'420 à 8'915 octets, soit une réduction de 65%;
- La proportion importante d'images par rapport au total dans la page d'entrée de l'école (78 %, 66'441) et celle de la place centrale (73 %, 82'276).
L'ensemble de ce travail est disponible dans un fichier comprimé: <http://sti.epfl.ch/i-base/sti-css2-base.tgz>.

En conclusion, le confort pour l'utilisateur de choisir une vision adaptée est appréciable. L'impact de cette modification est négligeable sur la grandeur d'une page Web. L'emploi de la compression Gzip permet d'offrir une interactivité plus élevée pour les usagers en réduisant la bande passante consommée par le client et par le serveur. La réduction de la taille totale des images reste un élément important dans l'optimisation de la vitesse d'un site.

*Comme cadeau pour les fêtes, je vous recommande la lecture du cycle de **L'Aube de la Nuit** de Peter F Hamilton qui le long de ces 3'000 pages peut vous captiver pour une partie de l'hiver. Pour faciliter la manipulation, ce cycle est décomposé en 6 volumes parus chez Robert Laffont (Rupture dans le Réel - Émergence, Rupture dans le Réel - Expansion, L'Alchimiste du Neutronium - Consolidation, L'Alchimiste du Neutronium - Conflit, Le Dieu Nu - Résistance, Le Dieu Nu - Révélation).*

