



Webinar Papis du Web :
Un site rapide pour mieux convertir

comprendre les problématiques de Performance Web
et optimiser pour mieux transformer



Les Papis du Web vous proposent d'organiser des Séminaires (+20 thématiques) qui vous permettront de prendre du recul, de bénéficier d'une expérience consolidée sur plus de 20 ans et de recueillir les bonnes pratiques issues de projets menés avec succès.

- Depuis de nombreuses années, nos experts « Papis du Web » analysent les tendances du marché.



Et bien d'autres...

- Les séminaires que nous proposons sont animés par les meilleurs spécialistes du digital.
- **Etat de l'art** et **analyse du marché**, les séminaires Clever Age vous donneront les clés pour votre stratégie digitale.

Web TV Les Papis du Web

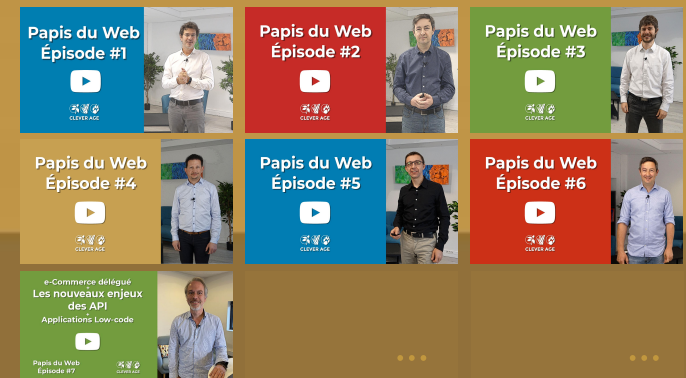
L'émission de Clever Age sur le digital

La chaîne Youtube de Clever Age

- Les Papis du Web #1 - [Omnicanalité](#)
- Les Papis du Web #2 - [Les Progressive Web Apps](#)
- Les Papis du Web #3 - [Architecture Serverless](#)
- Les Papis du Web #4 - [Product Information Management](#)
- Les Papis du Web #5 - [Customer Data Platform](#)
- Les Papis du Web #6 - [Web Performance](#)
- Les Papis du Web #7 - [API Gateway](#)

You Tube

Papis
du
Web





Boris, à toi de jouer



**Boris
Schapira**



w e  s p e e d

Comprendre
Collecter & Automatiser
Savoir-faire, faire savoir
Maintenir

X ms \Rightarrow X %

X ms \Rightarrow X %

X ms \Rightarrow X %



wpostats.com

Un retour d'XP
est une histoire
pas une promesse.

sites lents
=
mauvaise UX
(impacts variés)



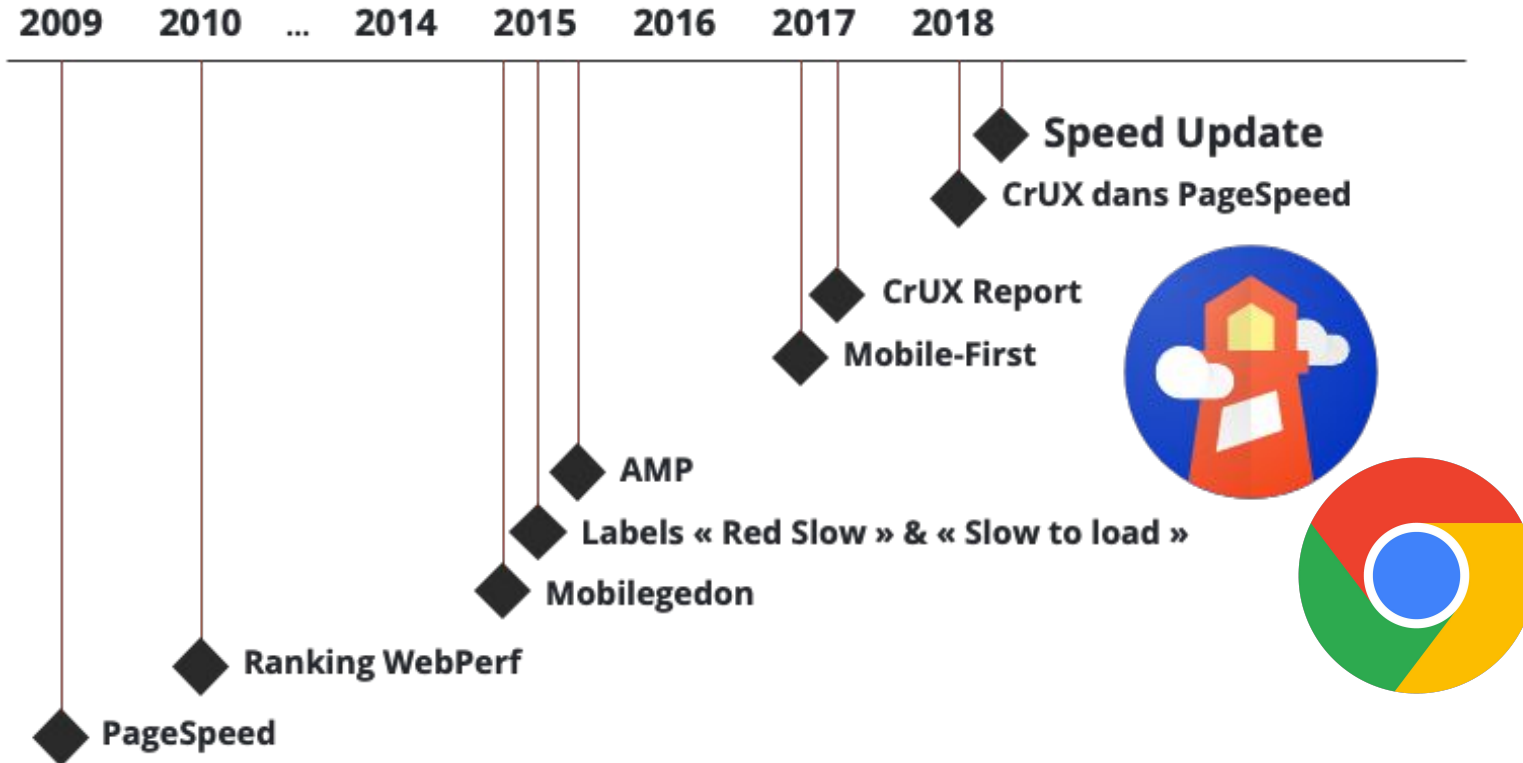
Une demande en augmentation

- De plus en plus d'expériences numérique
- Multiplication des canaux de diffusion (RS)
- Évolutions technologiques :
 - Infrastructure : 5G, fibre...
 - Matériel client : batteries, capteurs...
 - Protocoles et formats : HTTP/2, WebP...
- Plateformes capturantes (AMP)
- **Mais**, effet rebond qui nécessite une éco conception

Alors la technique s'améliore

- Mises-en-page adaptatives
- Formats et tailles d'image suivant les besoins
- Optimisation dans la diffusion des contenus
 - Content Delivery Network ;
 - Proxy d'optimisation d'images ;
 - Proxy d'optimisation totale...
- Nouvelles architectures :
 - JamStack, Single-Page Apps
 - microservices
 - serverless
- Navigation sur une « page » \Rightarrow Parcours Utilisateur

Google & la WebPerf



Impacts des gains de performance

- ➤ Coût d'exploitation
 - Réseau*
 - CPU client-side
- UX++ → Engagement & Conversion
- Augmentation des gains publicitaires
- Intégration des problématiques de Qualité Web à la culture de l'entreprise

* à l'exception des CDN



Éco-conception ?


- Globalement, la Performance Web amène à un Web plus léger, moins coûteux pour les postes clients (moins d'usure, moins de consommation réseau)
- **Cependant** : toutes les pratiques de Performance Web ne sont pas issues d'une éco-conception

Pour aller plus loin :

- “Comment lier performance et écologie ?” de Romuald Priol (vidéo)
- Les travaux et collectifs de Frédéric Bordage

D'où viennent les mesures ?

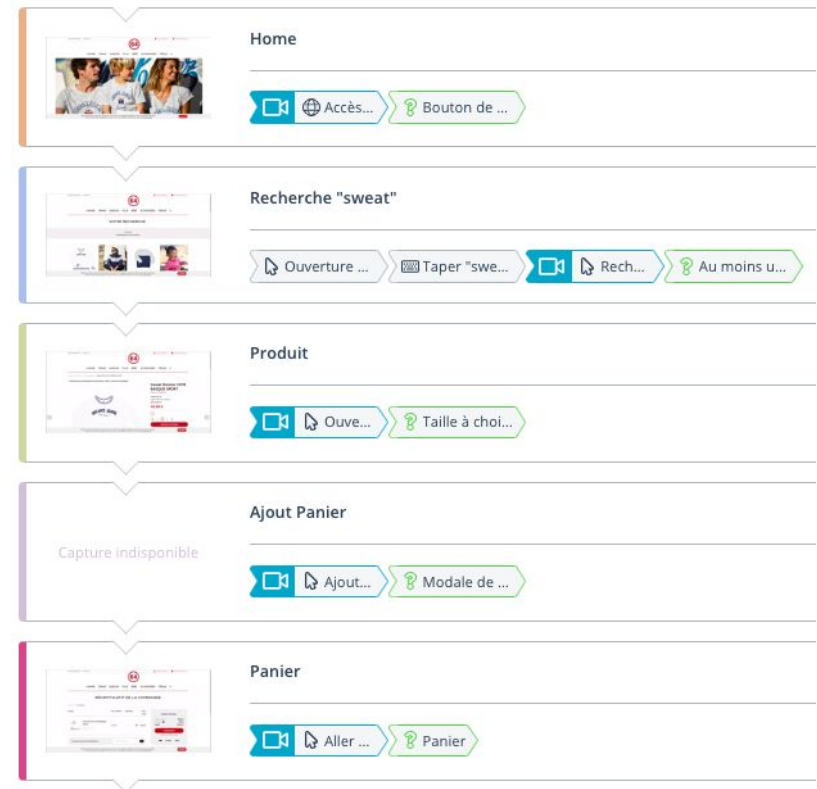
Surveillance synthétique

- Contextes limités mais maîtrisés
 - Suivi dans le temps
 - Validation d'hypothèses
 - Pas d'installation (permet la mesure de concurrents)
- Stabilité permettant la levée d'alertes
 - Tests reproductibles
 - mais... contextes limités à ceux définis
-  **dareboost** mais pas uniquement

Parcours Utilisateurs

L'expérience utilisateur n'est pas composée de chargement de pages mais d'(inter)actions, avec une notion d'étapes, et de cache navigateur.

Surveiller un parcours permet de comprendre les frustrations et les abandons.



RUM : Real User Monitoring

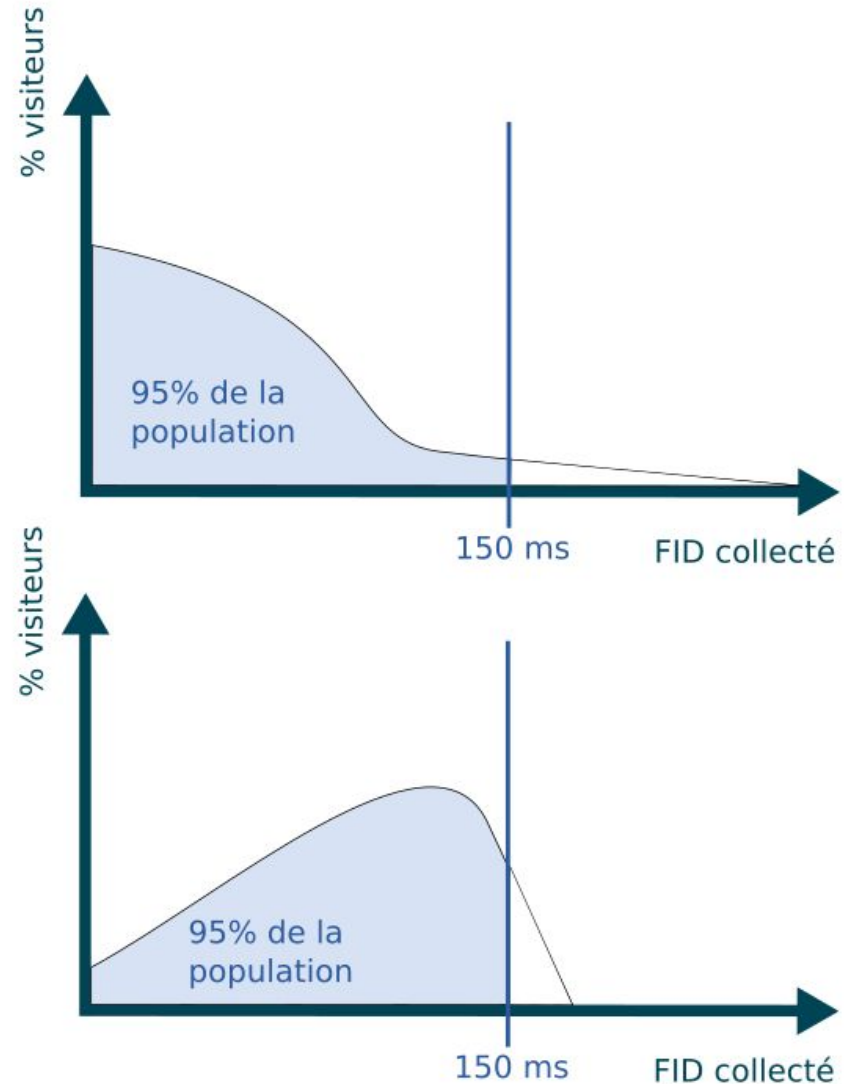
- Mesure de la **performance réelle**
- JavaScript implémenté sur le site client
- Grande quantité d'information collectée, agrégée, réduite, permettant l'exploitation.
⇒ services télémétrie similaire aux *Analytics*
- Coûts d'implémentation (dév.) et d'exploitation++
- Difficile de déterminer des pistes d'optimisation
- Impossible de tester avant la Production

En savoir plus

RUM : Real User Monitoring

Attention aux données agrégées, elles ne reflètent pas nécessairement la diversité des expériences ressenties.

En RUM, on observe les distributions.



Chrome UX Report

- Données RUM collectées par le **navigateur Chrome** lui-même, et agrégée par Google.
- Remontée dans la Search Console et dans PageSpeed Insights
([en savoir plus sur la Search Console](#))
- Utile pour avoir une vague idée des performance, mais uniquement sur des sites à fort trafic
- Sinon, trop de biais (matériels, pays...)
- Données disponibles dans [Google Cloud BigQuery](#)

Comprendre
Collecter & Automatiser
Savoir-faire, faire savoir
Maintenir

Ok mais on
mesure quoi ?



Temps de chargement

Dans un monde idéal...

... nous voudrions savoir quand un·e utilisateur·ice :

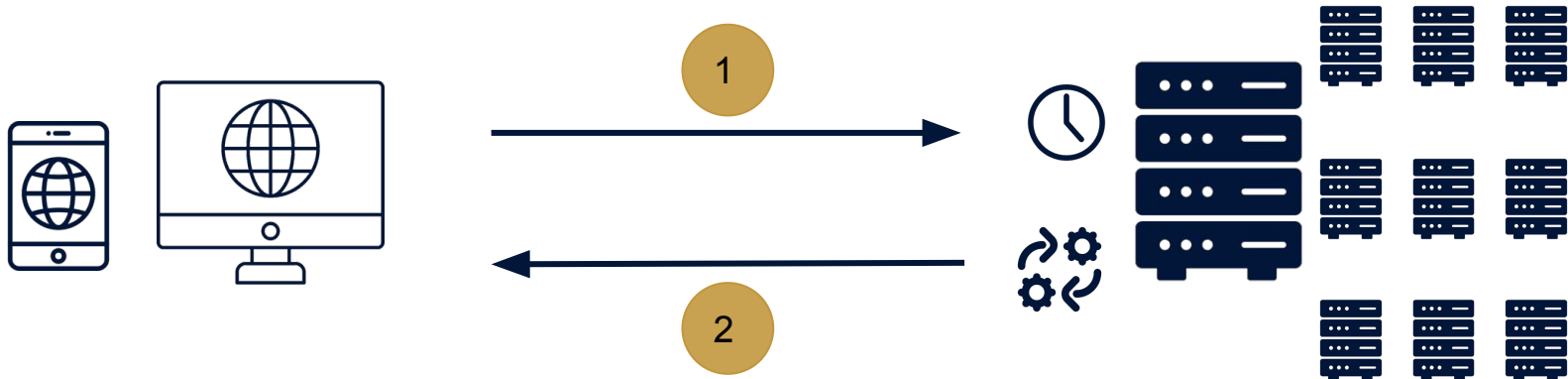
1. a la confirmation que le chargement **a commencé**
2. a l'impression, **en regardant la page**, qu'il peut interagir
3. **peut réellement interagir** avec la page
4. peut interagir **sans délai et sans ralentissements**

Ce qui peut se décomposer en plusieurs questions simples...

Est-ce que le
chargement a
commencé ?

Les « temps serveur »

- côté serveur, avec une éventuelle instrumentation
 - Blackfire.io, New Relic, Dynatrace...
- côté client : **Time To First Byte (TTFB)**



Est-ce que ça
s'affiche ?

Les « visuels »

- **Start Render / First Paint / First Contentful Paint :**

la page procède à son premier rendu

- problème, il n'est pas toujours visible ([en savoir plus](#))

Rapide : <1s

Moyen : <3s

Lent : >=3s

- **Visually Complete** : la partie visible de la page (*viewport*) est entièrement rendue

Problème : *ces indicateurs seuls ne permettent pas de savoir ce qui se passe.*

Le Largest Contentful Paint (LCP)

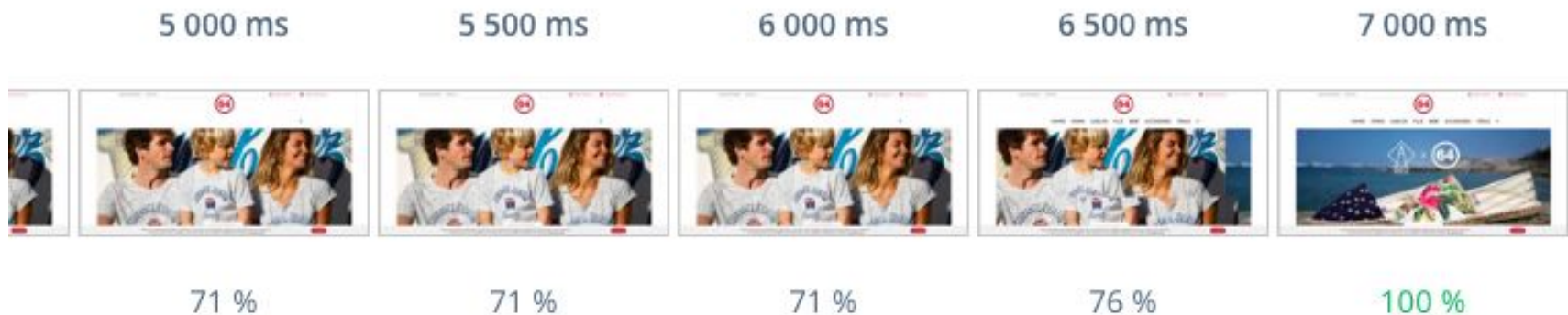
- Correspond au moment où le contenu principal de la page s'est probablement chargé
- Surveillance :
 - les ``, les `image` dans des `<svg>` ;
 - les `<video>` (y compris leur image de poster) ;
 - les éléments ayant un background CSS (`url()`) ;
 - les éléments contenant du texte.

Problème : *on ne sait toujours pas comment le chargement se déroule avant et après.*

Est-ce que
l'affichage est
progressif ?

Le Speed Index (en savoir plus)

- Mesure la **progressivité** de l'affichage
 - part de l'état final
 - à rebours, mesure un % de complétion frame par frame
 - calcule la progressivité
- Attention, perturbé par tout ce qui change l'état final :
 - Les pop-ins (promotions, *EU Cookie Law*)
 - Les carrousels, modales, vidéos...



Est-ce que
l'affichage est
progressif et
fiable ?

Le Cumulative Layout Shift (CLS)

Que se passe-t-il quand un élément se déplace **pendant** le chargement ? **De la frustration.**



Le Cumulative Layout Shift (CLS)

- Basé sur la *Layout Instability API* ([W3C draft](#)), qui permet de détecter le déplacement d'éléments du DOM en raison du chargement asynchrone du contenu.
- Permet de détecter des mauvaises pratiques dans le continuité du chargement, d'en mesurer **la fiabilité**.

Est-ce qu'à un
moment, c'est
parfaitement
utilisable ?

Time To Interactive (en savoir plus)

- Mesure les conditions d'une « **interaction qualitativement garanties** »
- Celui de Google se base sur des fenêtres d'évaluation de 5 s
 - + les Long Tasks JavaScript (50 ms)
 - + le trafic réseau
 - + d'autres événements

Problème : *on ne sait toujours pas comment le chargement se déroule avant et après.*

Total Blocking Time

- Mesure la sommes des temps de blocages
 - toutes les tâches CPU qui dépassent 50 ms
 - TBT = somme des dépassements
entre le FCP (début du rendu)
et le TTI (plus de ralentissements)
- Bon indicateur de mesure des ralentissements
potentiellement ressentis

Les interactions retardées

- À quelle fréquence l'interaction d'une personne a-t-elle été retardée de plus de 50 ms ? La première *interaction retardée* est le **First Input Delay**.

Pour Google (Chrome UX Report) :

- **Rapide** : <100ms
- **Moyen** : <300ms
- **Lent** : >=300ms

Les indicateurs comportementaux

- **Rage Clicks** : clics répétés et rapides sur une zone du *viewport*
 - informe sur l'agacement des utilisateurs
 - entre 1,25 et 1,5 fois le temps nécessaire à l'affichage*
- **Mouse Movements / Cursor Trashing** : enregistrement des mouvements rapides de souris sur l'interface
 - informe sur l'efficacité de l'interface

En savoir plus : ["User Experience & Performance: Metrics that Matter", par Philip Tellis \(Boomerang\)](#)

Ok, je suis perdu.

Le jeu idéal pour un suivi régulier

- Sur la plupart des pages :
 - Un indicateur de ce qui se passe côté serveur : le **Time To First Byte** (Premier Octet)
 - Un indicateur visuel : le **Speed Index** est un bon point d'entrée.
- Sur les pages interactives : un indicateur d'interactivité (**Time To Interactive**, **Total Blocking Time**, **First Input Delay** si RUM disponible)
- Enfin, des **métriques personnalisées** (posées en JavaScript)

Comprendre
Collecter & Automatiser
Savoir-faire, faire savoir
Maintenir

Premières analyses “one-shot”

Ici sur Dareboost, mais aussi Lighthouse ou Pagespeed Insights

Analyse de qualité et performance web

<https://www.albertine-swim.com/fr/>

Rapport privé généré le 3 juil. 2019 18:34:40 | Relancer

Télécharger le rapport



Pas mal !
Mais peut mieux faire.

3

Problèmes

4

Améliorations

85

Succès



Voir les priorités

VISITEUR SIMULÉ : Chrome Paris 8.0/1.5Mbps (Latence : 50 ms)

Modifier

Mémoriser

Requêtes

37

HTML CSS Scripts Images Autres

Poids

1.26 Mo

Premier octet

1.08 sec

Début affichage

1.57 sec

Fin chargement

3.30 sec

Timeline / Waterfall

Détail des étapes

Le chargement en images

Erreurs navigateur 0 OK

Support d'HTTP/2 : 89%

Speed Index: 1835

Technologies :



Bonnes pratiques et conseils : Toutes les catégories

Ce que vous pouvez améliorer

Volume de données

0/100

2 images sont redimensionnées côté navigateur

Une page récupérant une image plus grande que celle réellement affichée charge inutilement un volume de données trop important.

Sécurité

0/100

Il manque une politique de sécurité sur la provenance de vos ressources

Protégez votre site Web contre les attaques de type XSS (Cross-Site Scripting) en mettant en place une politique restrictive de sécurité du contenu.

Optimisation du rendu

60/100

Différez l'utilisation du code JavaScript

Lorsque le navigateur web rencontre du code JavaScript en interprétant le code source d'une page web, cela peut ralentir considérablement l'affichage de la page, surtout s'il est nécessaire de télécharger un script externe.

Sécurité

0/100

17 cookies ne sont pas sécurisés

Un cookie transmis au navigateur via le protocole HTTP doit pouvoir transiter uniquement sur une connexion sécurisée (sauf cas particulier).

Référencement (SEO)

0/100

Cette page n'indique pas de fil d'Ariane aux moteurs de recherche

Vous pouvez mettre en valeur vos pages sur les moteurs de recherche en indiquant un fil d'Ariane via des données structurées.

Premières analyses “one-shot”

Analyse de qualité et performance web

<https://www.albertine-swim.com/fr/>

Rapport privé généré le 3 juil. 2019 18:34:40 | [Relancer](#)

Télécharger le rapport



Pas mal !
Mais peut mieux faire.

3

Problèmes

4

Améliorations

85

Succès



Voir les priorités

VISITEUR SIMULÉ : Chrome Paris 8.0/1.5Mbps (Latence : 50 ms)

Modifier

Mémoriser

Requêtes

37

Poids

1.26 Mo

HTML CSS Scripts Images Autres

Timeline / Waterfall

Premier octet

1.08 sec

Détail des étapes

Début affichage

1.57 sec

Le chargement en images

Fin chargement

3.30 sec

Erreurs navigateur 0 OK

Support d'HTTP/2 : 89%

Speed Index: 1835

Technologies :

Indicateurs
(nombres, poids,
jalons
temporels)

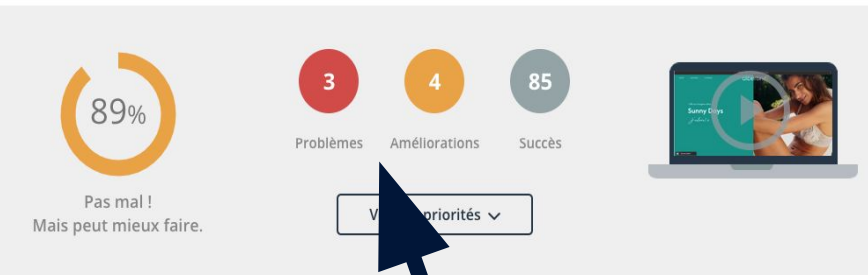
Premières analyses “one-shot”

Analyse de qualité et performance web

<https://www.albertine-swim.com/fr/>

Rapport privé généré le 3 juil. 2019 18:34:40 | [Relancer](#)

Télécharger le rapport



Conseils
(bonnes pratiques)

Bonnes pratiques et conseils : [Toutes les catégories](#)

Ce que vous pouvez améliorer

Volume de données

0/100

2 images sont redimensionnées côté navigateur

Une page récupérant une image plus grande que celle réellement affichée charge inutilement un volume de données trop important.



Sécurité

0/100

Il manque une politique de sécurité sur la provenance de vos ressources

Protégez votre site Web contre les attaques de type XSS (Cross-Site Scripting) en mettant en place une politique restrictive de sécurité du contenu.



Optimisation du rendu

60/100

Différez l'utilisation du code JavaScript

Lorsque le navigateur web rencontre du code JavaScript en interprétant le code source d'une page web, cela peut ralentir considérablement l'affichage de la page, surtout s'il est nécessaire de télécharger un script externe.



Sécurité

0/100

17 cookies ne sont pas sécurisés

Un cookie transmis au navigateur via le protocole HTTPS doit pouvoir transiter uniquement sur une connexion sécurisée (sauf cas particulier).



Référencement (SEO)

0/100

Cette page n'indique pas de fil d'Ariane aux moteurs de recherche

Vous pouvez mettre en valeur vos pages sur les moteurs de recherche en indiquant un fil d'Ariane via des données structurées.



Corpus de “Bonnes pratiques”

- Consensuelles
- Réalistes
- Vérifiables (automatiquement)
- Dépendantes des technologies employées

Technologies :         

Monter en compétence

! 2 images sont redimensionnées côté navigateur

Une page récupérant une image plus grande que celle réellement affichée charge inutilement un volume de données trop important.

Redimensionnement d'images

Le redimensionnement d'une image côté navigateur pour en réduire la taille d'affichage n'est pas recommandé.

Fournissez directement vos images aux dimensions d'affichage utilisées sur votre site. Vous évitez ainsi l'envoi de données inutiles sur le réseau, ce qui diminue le temps de chargement de la page.

Responsive design ? Images Retina ?

Le responsive design ou encore l'utilisation d'images retina ne justifient pas un redimensionnement d'images. Même dans ces cas, différentes méthodes existent afin de délivrer vos images aux bonnes tailles. Nous vous conseillons de lire les ressources suivantes :

- [Introduction sur les images responsive](#)
- [Picturefill, pour supporter l'élément <picture> dès maintenant \(EN\)](#)
- [RICG, groupe de travail sur les images responsive \(EN\)](#)

Ne redimensionnez pas les images suivantes :

- [www.albertine-swim.com/modu\[...\]ngerie.jpg](http://www.albertine-swim.com/modu[...]ngerie.jpg) (taille d'affichage : 676x688)
- [www.albertine-swim.com/modu\[...\]tropic.jpg](http://www.albertine-swim.com/modu[...]tropic.jpg) (taille d'affichage : 340x510)

Ne pas tomber dans le piège

- Certaines optimisations sont simples à comprendre et difficiles à implémenter. Pour d'autres, c'est l'inverse.
- Mon astuce :
 - Est-ce une configuration ?
⇒ **Probablement complexe (découpable en petites tâches).**
 - Est-ce une fonctionnalité à part entière ?
⇒ **Probablement compliqué (nécessite des décisions)**



Les grands thèmes

- **Optimisation de la diffusion** : configuration serveur(s), protocole(s), CDN(s)
- **Optimisation du rendu** : ressources critiques, ressources d'amélioration
- **Réduction des délai d'interaction** : organiser le code de manière à solliciter le moins possible le CPU

Quelques indémodables

- Un code **HTML, CSS, et JS** taillés avec soin, valide et épuré
 - CSS Purging ; JS tree-shaking... automatisez !
- Des domaines spécialisés (pages, assets) des web serveurs adaptés, et du cache (Varnish, CDN...)
- Des polices de caractères optimisées (subset) et non-bloquantes pour le rendu (utilisez `font-display`)

Transmettre des données

- Réduire le nombre de requêtes BDD
- Du *Caching* partout où c'est possible
 - Dans la BDD
 - Dans vos services
 - Devant vos frontaux
 - Au niveau du CDN
 - Côté client, et même dans les *service workers (offline)*

mais attention à l'invalidation du cache !

Images

- adaptées à la surface de rendu et aux qualités de l'écran (`srcset`, `picture`)
- optimisées : le bon format en fonction du contenu, optimisé suivant le navigateur
(GIF animés => `video`)
- chargées uniquement si présentes dans le *viewport* (lazy loading natif ou sur-mesure)

Et niveau JavaScript ?

Se reposer, autant que possible, sur HTML et CSS, voire n'utiliser JS que pour de l'amélioration progressive.

- Réduire le JS au maximum (faire en CSS tout ce qui est faisable)
- Différer le JS restant ou l'injecter à la demande
- Bien choisir ses librairies en fonction de ses besoins.

React, Vue, Svelte, Angular... est-ce utile ?

Comment faire le tri ?

- Garder l'historique (git, SVN, noeuds statiques)
- Itérer, tester, communiquer
 - Audits synthétiques : comparaison de versions A / B
 - Des tests fonctionnels, pour éviter les régressions
 - Anticiper la désactivation des optimisations
 - Avoir des outils permettant de communiquer transversalement sur les résultats
- Pas le temps ? Des “raccourcis” existent
 - Optimisation des images : Cloudinary, ImageKit, TwicPics
 - WebPerf As A Service : Fasterize

Comprendre
Collecter & Automatiser
Savoir-faire, faire savoir
Maintenir et dépasser

Une fois en Production

- On contrôle ses budgets de performance



Réviser la chaîne de Production

- La WebPerf doit être, comme l'accessibilité ou le référencement, **une valeur de premier ordre**
- Fonctionnellement, beaucoup de concepts ne verraient pas le jour si c'était le cas : identifier les problèmes au plus tôt en instaurant **une culture** de la web performance
- Adoptez une vision concurrentielle pour **éviter la surqualité**



Merci !



testez !



regardez !

Nos prochains webinars *Papis du Web*

..... events.clever-age.com/webinars/#programmation

Mercredi 8 avril : Docker : Pour qui ? Pour quoi ? Dans quels contextes ? par Jean-Yves Camier (Clever Age)

Jeudi 9 avril : Qualité Web par Elie Sloïm (Opquast)

Vendredi 10 avril : PWA (Progressive Web App) par Nicolas Hoizey (Clever Age)

Définition, différentes stratégies de mise en œuvre et principaux composants et outils nécessaires ou facilitant les projets

Mercredi 15 avril : Kubernetes par Jean-Yves Camier (Clever Age)

Jeudi 16 avril : Cartographier les Parcours Clients pour améliorer sa Performance Clients par par Antoine Coubray (CustUp)

Etapes. Méthodes. Outils.

Vendredi 17 avril : MDM par David Chemla, Be Clever Partner

Du rêve à la réalité

Mardi 21 avril : Stratégie de contenus par Nicolas Pellichero (Clever Age)

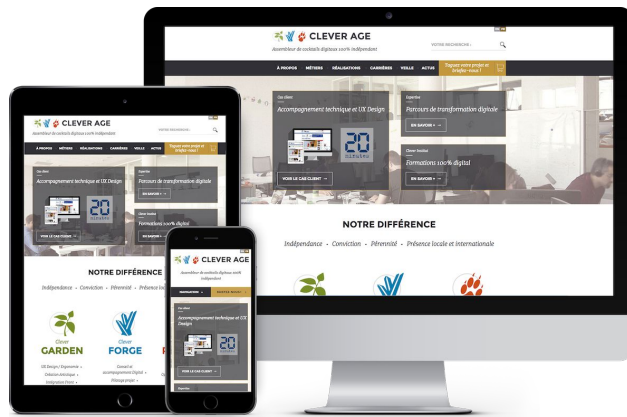
Prendre la parole pour transformer

Mercredi 22 avril : L'Intelligent Data Hub: une nouvelle ère pour le MDM par Julien Peltier (Semarchy)

Merci, gardons le contact !

Nos actualités, nos recrutements, nos petits-déjeuners et des exemples de réalisations pour nos clients...

Livres blancs, fiches produits, blog, études...



Siège social | Clever Age

34 rue de Saint-Pétersbourg

75008 Paris - FRANCE

commercial@clever-age.com

+33 1 53 34 66 10

www.clever-age.com



linkedin.com/company/clever-age



twitter.com/cleverage