Les sociétés [Izardev](http://www.izardev.fr/%2A) et [BlueLogic](http://www.bluelogic.fr/) m’ont accordé ‘gracieusement’ du temps pour que je leur explique notre problématique sur les performances de notre site : voici leurs conclusions sur la méthode à apporter pour améliorer les performances de bilatu :

* Point 1 : Izardev déconseillé l’Ajax mais BlueLogic le préconise fortement (comme moi-même d’ailleurs) : c’est plus rapide et surtout cela permet, avec les call-backs, d’attendre que l’annuaire soit chargé pour charger la carte. C’est dommage que j’ai supprimé les appels Ajax depuis mes discussions avec Izardev.
* Point 2 : Le chargement d’1 Mo sur la page d’accueil que vous avez constaté provient de cache manquant sur WordPress ; en standard WP n’en propose pas : nous avons fait un essai concluant cet après-midi avec BlueLogic et le cache [W3 total Cache](http://wordpress.org/plugins/w3-total-cache/)
* Point 3 : Le chargement séquentiel des news, web, carte et bp est à surtout à cause des news : les news seront chargée en asynchrone, c’est-à-dire qu’on n’attendra pas que leur chargement se termine avant de lancer le web, carte et annuaire.
* Point 4 : Le plugin exec-Php que j’utilisais jusqu’à présent pour nos pages php, est à éviter d’après Izardev et BlueLogic: il peut ralentir WP et contient des failles de sécurité. J’avais déjà effectué ces changements et transformé mon code en widget WP.

J’ai donc proposé à ***BlueLogic*** (plus compétent qu’Izardev) de nous faire une proposition pour réaliser la prestation suivante :

* améliorer les performances de notre site d’un point de vue des outils à utiliser et des méthodes de programmation à en mettre en œuvre suivant les points cités ci-dessous. Je ne lui ai pas demandé de chiffrer une refonte graphique du site (menu, centrage, …).

Nous aurons sa propale lundi matin. Je pense que ce sera de quelques jours et financé par ETICOOP, mais cela pourra me dégager du temps sur le déploiement de la nouvelle infra bilatu. Il leur faudra donc un accés au site. De plus je vais lister ce qu’il reste à faire (cf planning page suivante) sur bilatu et on validera cela DEFINITIVEMENT lundi matin avec franck (beaucoup d’infos sont déjà sur le bugtracking : Le critère des track à prendre en compte est : Priorité = Immédiate)

**Apache :**

* Paramétrer : rajouter les modules **mod-expires** et **mod-headers** pour améliorer les performances de chargement du serveur.

**mod\_headers :** permet de manipuler et contrôler les en-têtes de requêtes et réponses HTTP. C’est lui qui va permettre ou non la mise en cache par le navigateur.

**mod\_expires :** permet d’indiquer au navigateur la validité des fichiers téléchargés. Lors du premier chargement, le navigateur télécharge tous les éléments de la page et les met en cache. Dans les chargements suivants, le navigateur va contrôler la validité des fichiers présents dans son cache et téléchargera uniquement que les fichiers qui ne sont plus valides.

**Wordpress :**

* Rajouter le plugin de gestion de cache : [W3 total Cache](http://wordpress.org/plugins/w3-total-cache/)  : quelques paramètres
* css et jss cache : charge en 1 seule fois les fichiers css et js
* page cache : à activer
* minify cache : si on active les menus hodei (menu droite ‘fixe’ et menu de gauche ‘solutions numériques’) sont désorganisés

si on l’active mettre mode = Auto

* databases cache : permet de mettre en cache les données de base de WP : efficace pour le plugin WP business directory
* debug cache : ne pas l'activer car une erreur d'écriture se produit : il faut que WP est un accès W sur **wp-content** et plus particulièrement sur **htaccess** pour y rajouter ses règles
* HungryFeed :
	+ mettre le cache à 3600 sec (déjà le cas)
	+ charger dans un div fixe (par exemple 300 pixels) le résultat du flux : la zone se chargera au fur et à mesure sans perturber la recherche web : cela permet d'éviter à l'utilisateur de cliquer sur un lien 'qui bouge'.
	+ Tester les flux avant de les charger
* OSM (carte)

Pour optimiser le chargement des données il faut (c’est déjà prévu mais non réalisé) un fichier lat/lon par utilisateur ou sessionid ; en effet chaque utilisateur ne consulte pas les mêmes données : ce chargement de fichier (un peu long) peut être remplacé par un ensemble de variables de session stockant les différents points lat/lon trouvés.

**Programmation de nos pages :**

* La méthode ajax est la meilleure : elle remplacera le chargement séquentiel des div (fichier bilatu.php) et leur includes php. De plus elle permet de lancer la carte après l'annuaire avec un **jquery.load** et une fonction de callback. On modifiera donc le chargement de la page d’accueil et on pourra (comme c’était le cas avant avec ajax), relancer une recherche.
1. Optimiser le code de nos pages PHP chargé par WP : installer sur le serveur php [APC](http://fr.php.net/apc)
2. Analyse des performances du code :
	* [NewRelic](http://www.newrelic.com/?utm_source=GOOG&utm_medium=adwords&utm_content=rpm&utm_campaign=RPM&utm_term=NewRelic&mpc=PS-GOOG-RPM-EN-0-Europe-Brand-NewRelic&gclid=CKuLvNbmt7wCFUnjwgodKU8ATg*) (il existe peut-être un plugin WP) est un module apache qui permet d'analyser la vitesse d'exécution de toutes les pages du site : pages WP ou nos propres pages.
	* Mesurer le temps d’exécution des scripts PHP : [technique ici](http://www.fobec.com/tuto/1028/mesurer-temps-execution-script-php.html)

**Planning :**

