Documentation technique du stage de Martin Dazin (29 Mars 2016 au 13 Juillet 2016)

Sommaire

[Projet Plugin de newsletters 3](#_Toc456196179)

[Contexte 3](#_Toc456196180)

[Découpage du plugin au niveau du code 3](#_Toc456196181)

[Premier niveau (coloration en vert) 4](#_Toc456196182)

[Deuxième niveau (coloration en marron) 4](#_Toc456196183)

[Troisième niveau (coloration en bleu clair) 5](#_Toc456196184)

[Découpage du plugin au niveau fonctionnel 8](#_Toc456196185)

[BDD 8](#_Toc456196186)

[préfix\_atm\_nl\_newsletter 9](#_Toc456196187)

[préfix\_atm\_nl\_stat 12](#_Toc456196188)

[préfix\_atm\_nl\_subscriber 13](#_Toc456196189)

[Choix personnels 14](#_Toc456196190)

[Amélioration possibles sur le plugin 14](#_Toc456196191)

[Projet Webmarketing 15](#_Toc456196192)

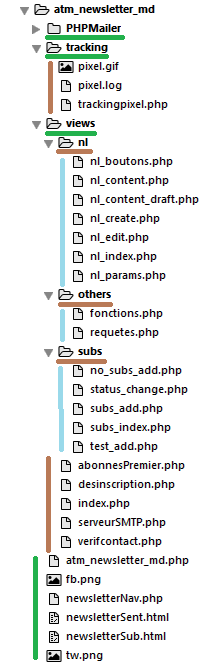
[Mémento Rapide 16](#_Toc456196193)

# Projet Plugin de newsletters

## Contexte

Voir mon rapport de stage (Partie 4).

## Découpage du plugin au niveau du code



*Arborescence du découpage du plugin au niveau du code*

Dans le dossier atm\_newsletter\_md, on peut voir qu'il y a trois niveaux de dossiers. Le premier niveau est coloré en vert, le deuxième en marron et le dernier en bleu clair.

### Premier niveau (coloration en vert)

Au premier niveau on trouve trois dossiers :

* PHPMailer : Est un dossier comportant une classe qui permettra d'envoyer nos différents emails lors de l'envoi de nos nl.
* tracking : Est un dossier qui nous permettra de traquer l'ouverture des emails par les utilisateurs.
* views : Est un dossier qui contient nos vues.

On trouve également des fichiers :

* atm\_newsletter\_md.php : C'est la page php qui nous permet de procéder à l'installation du plugin. Aspects techniques : Script de création de tables dans la BDD, création du fichier d'uploads. Cela a été mis en commentaires parce que je n'en avais l'utilité, mais il y a aussi la création d'un CRON, de plages d'horaires supplémentaires (à retravailler).
* fb.png : Icône du rs Facebook que l'on inclut dans nos nl. Aspects techniques : Appel à l'image avec <img>. <a href> qui pointe vers la page du rs de l'entreprise.
* newsletterNav.php : nl affichée lorsque l'on clique sur "Si ce message ne s'affiche pas correctement, cliquez ici.". Aspects techniques : <a href> de la nl qui pointe vers cette page. On passe des args dans l'URL de redirection vers cette page en $\_GET. Requêtage SQL pour avoir la bonne nl.
* newsletterSent.html : On se sert de ce modèle vide de nl à chaque envoi de nl. Aspects techniques : Alimentation de données dans ce modèle vide en php avec str\_replace().
* NewsletterSub.html : Modèle pré-rempli de mail lorsque qu'une personne doit confirmer son inscription pour recevoir des nl via la technique du double opt-in. Aspects techniques : Pour accéder à cette page il faut tout d'abord remplir un formulaire en double opt-in. Puis on reçoit un mail avec un lien cliquable. Quand on a cliqué sur ce lien on est redirigé vers verifcontact.php. Cette page php prend en $\_GET notre adresse email et va checker avec la BDD s'il y a correspondance. S'il y a correspondance alors le non-confirmé va devenir un sub.
* tw.png : Icône du rs Twitter que l'on inclut dans nos nl. Aspects techniques : Appel à l'image avec <img>. <a href> qui pointe vers la page du rs de l'entreprise.

### Deuxième niveau (coloration en marron)

Dans le dossier tracking on trouve des fichiers :

* pixel.gif : Un pixel de 1x1 blanc. Aspects techniques : C'est une image que l'on affiche après avoir exécuté du code dans trackingpixel.php.
* trackingpixel.php : Fichier php permettant d'enregistrer dans pixel.log les ouvertures d'emails. Aspects techniques : Requêtage sql pour récupérer le dernier newsletter\_showed\_id. Fonction php file\_put\_contents() pour écrire des informations à l'intérieur. Utilisation de $\_SERVER['REMOTE\_ADDR'] pour connaître l'IP du sub.
* pixel.log : Un fichier contenant les infos de tracking concernant l'ouverture des emails. Aspects techniques : Ce fichier est lu à chaque fois que l'on est sur la page d'accueil de plugin pour MAJ les stats. Toutes les lignes sont enregistrées en BDD.

Dans le dossier views on a :

* Un sous-dossier nl qui contient le code de traitements des nl.
* Un sous-dossier others qui contient la globalisation des fonctions php et des requêtes sql.
* Un sous-dossier subs qui contient le code de traitements des subs.

On a également des fichiers :

* desinscription.php : Page accessible uniquement lorsque l'on clique sur le lien de désinscription d'une nl. Aspects techniques : Formulaire en $\_POST avec saisie de son adresse mail. Si checking validé en BDD avec la table préfix\_atm\_nl\_subscriber, alors le sub change de statut (C à D).
* index.php : Page d'accueil du plugin. On a deux éléments bien distincts dans cette page. La première est l'affichage des statistiques avec deux graphiques. Le premier graphique est recense toutes les statistiques pour les toutes les nl alors que le second ne référence que les stats pour la dernière nl. Le deuxième élément est un formulaire en double opt-in. Aspects techniques : Pour la gestion des graphiques il y a appel de Google Charts avec du requêtage sql pour les alimenter. On fait donc passer des données php en js (à améliorer : ça n'a pas été fait de manière propre). La gestion du double opt-in se fait par un formulaire en $\_POST. Si l'adresse saisie est correcte alors on envoie un email à cette adresse. Cette adresse est enregistré en BDD dans la table préfix\_atm\_nl\_subscriberen tant que non-confirme (statut N). L'email contient un lien avec un paramètre qui est l'adresse mail de la personne. Si on clique sur ce lien on est redirigé vers une autre page qui nous confirme l'inscription à ce site (à améliorer : encryptage de l'adresse mail, par exemple MD5).On checke que l'adresse existe bien en BDD. Si c'est le cas on change de statut (N à C).
* serveurSMTP.php : Page où l'on rempli les informations pour configurer notre serveur SMTP. Puis quand il est configuré on peut tester si les informations sont correctes en s'envoyant un mail. Aspects techniques : Formulaire en $\_POST et enregistrement des données en BDD dans la table préfix\_options. Quand les informations saisies concernant le serveur sont correctes on affiche le formulaire pour envoyer un email de test à l'adresse de mail renseignée.
* verifcontact.php : Page qui permet de dire après la technique du double opt-in que l'on est bien enregistré dans la BDD. Aspects techniques : Quand cette page est appelée elle prend en paramètre $\_GET l'adresse mail de la personne qui a cliqué sur le lien pour accéder à cette page. Ensuite on checke en BDD dans la table préfix\_atm\_nl\_subscriber. Si c'est bon le sub change de statut (N à C).

### Troisième niveau (coloration en bleu clair)

On a tout d'abord l'ensemble des pages pour le dossier nl :

* nl\_boutons.php : Cette page est appelée à chaque fois que l'on appuie sur une bouton de la page nl\_edit.php. Il y a quatre choix possibles. Attention, le code du bouton Supprimer est fait dans la page nl\_edit.php. Aspects techniques : En prenant le contenu due $\_POST on arrive à savoir la nl et le bouton qui ont été choisis par l'administrateur :
  + Bouton "Visualiser" : Requêtage sql de la table préfix\_atm\_nl\_newsletter pour afficher la nl en fonction de $\_POST.
  + Bouton "Envoyer en test" : Requêtage sql de la table préfix\_atm\_nl\_newsletter pour choisir la nl à envoyer en fonction de $\_POST à toutes les adresses email où leur statut sera égal à T dans la table préfix\_atm\_nl\_subscriber.
  + Bouton "Envoyer réellement" : Requêtage sql de la table préfix\_atm\_nl\_newsletter pour choisir la nl à envoyer en fonction de $\_POST à toutes les adresses email où leur statut sera égal à C dans la table préfix\_atm\_nl\_subscriber.
  + Bouton "Statistiques" : Requêtage sql de la table préfix\_atm\_nl\_newsletter pour chercher les stats de la nl en fonction de $\_POST. Ensuite on affiche ses stats grâce à Google Charts.
* nl\_content.php : Nl créée avec le tracking l'intérieur. Aspects techniques : Requêtage sql pour rechercher le contenu personnalisé de la nl provenant de la table préfix\_options ainsi que le contenu de la nl dans la table préfix\_posts. Après avoir été cherché tout notre contenu on exécute la fonction php str\_replace() qui va mettre aux bons endroits du modèle de nl vide newsletterSent.php le requêtage sql. Il y a aussi un <a href> vers le pixel espion.
* nl\_content\_draft.php : Nl créée sans le tracking à l'intérieur. Aspects techniques : Pareil que nl\_content.php sauf qu'il n'y a pas le tracking.
* nl\_create.php : Sur cette page on choisit le type de posts que l'on veut avoir dans la nl. Quand on a choisi on crée une nl. Aspects techniques : On récupère via une requête sql dans la table préfix\_post tous les types de posts qui nous intéresse. Ensuite on les affiche dans un <form> qui inclus une <table>. Chaque <tr> correspond à un type de post et contient une checkox. Quand on valide le formulaire de création de nl on regarde quelles checkbox ont été cochées et on affiche quels ont été les articles qui vont être inclus dans la nl. Par défaut on affiche à chaque fois les dix derniers articles pour chaque type de post. Lors de la validation du formulaire on va également créé les nl en BDD dans la table préfix\_atm\_nl\_newsletter. Pour chaque nl créée via ce formulaire on va créer deux nl en BDD : une avec le tracking et une autre sans le tracking.
* nl\_edit.php : Cette page permet visionner toutes les nl que l'on a créé avec leur nom, leur statut, le nombre d'abonnés et leur date . Elles sont rassemblées dans un <table> où il y a cinq boutons : "Visualiser", "Envoyer un test", "Envoyer réellement", "Statistiques" et "Supprimer". Aspects techniques : On requête en SQL pour chercher les différentes informations. Quelque soit leur statut leur nom restera le même. Ensuite, selon leur statut le nombre d'abonnés et leur date risquent de changer. Si la nl a été envoyée aux subs on va afficher sa date d'envoi et non plus sa date de création. Si la nl a été envoyée aus subs on va afficher le nombre d'emails envoyés alors que si elle n'a pas été envoyée on se contentera juste d'indiquer le nombre de subs à ce moment.
* nl\_index.php : Cette page est le squelette vide de la page Newsletters de notre plugin. Aspects techniques : On va afficher une tabs en js pour avoir tous les contenus avec la fonction php include().
* nl\_params.php : Sur cette page, l'administrateur du site sera invité à choisir ses préférences pour personnaliser sa nl. Aspects techniques : Un simple formulaire en $\_POST qui va enregistrer les params en BDD dans la table préfix\_options.

Ensuite dans le dossier others on deux scripts php :

* fonctions.php : Fichier qui contient l'ensemble de nos "fonctions maison" php.
* requetes.php : Fichier qui contient une grande partie de nos requêtes sql. Attention, il y a des requêtes qui ont été faites en dehors de ce fichier parce qu'elles demandaient des args qu'il fallait déterminer auparavant.

Puis on a le dossier subs avec les fichiers suivants :

* no\_subs\_add.php : Cette page permet d'uploader un fichier .csv pour ajouter des personnes en blacklistés ou de remplir un formulaire pour ajouter manuellement un blacklisté. Aspects techniques : Le fichier va être uploadé dans le répertoire suivant : /nom\_du\_projet/wp-content/uploads/atm\_newsletter\_md. Ensuite on va lire le fichier avec les fonctions php fopen() et fgetcsv(). On va ajouter en BDD les adresses mails en tant que sub (statut B) qui sont correctes (càd les adresses qui sont correctes syntaxiquement et qui n'ont pas été encore ajoutées à la BDD) dans la table préfix\_atm\_nl\_subscriber. On peut faire la même chose juste pour une adresse. Dans ce cas-là il faudra la saisir manuellement.
* statut\_change.php : On peut changer le statut des personnes. Aspects techniques : Dans un formulaire on affiche toutes les personnes dans une <table>. Chaque <tr> correspond à une personne, son statut respectif et contient une checkox. Pour le moment toutes les personnes cochées passeront forcément en blacklistés (càd ? -> B) (à améliorer : gestion plus fine des changement des statuts des personnes).
* subs\_add.php : Cette page permet d'uploader un fichier .csv pour ajouter des personnes en confirmés ou de remplir un formulaire pour ajouter manuellement un confirmé. Aspects techniques : Le fichier va être uploadé dans le répertoire suivant : /nom\_du\_projet/wp-content/uploads/atm\_newsletter\_md. Ensuite on va lire le fichier avec les fonctions php fopen() et fgetcsv(). On va ajouter en BDD les adresses mails en tant que sub (statut C) qui sont correctes (càd les adresses qui sont correctes syntaxiquement et qui n'ont pas été encore ajoutées à la BDD) dans la table préfix\_atm\_nl\_subscriber. On peut faire la même chose juste pour une adresse. Dans ce cas-là il faudra la saisir manuellement.
* subs\_index.php : Cette page est le squelette vide de la page Ajout & Gestion des de notre plugin. Aspects techniques : On va afficher une tabs en js pour avoir tous les contenus avec la fonction php include().
* test\_add.php : Cette page permet de remplir un formulaire pour ajouter manuellement un testeur. Aspects techniques : On va ajouter en BDD les adresses mails en tant que testeur (statut T) qui sont correctes (càd les adresses qui sont correctes syntaxiquement et qui n'ont pas été encore ajoutées à la BDD) dans la table préfix\_atm\_nl\_subscriber.



*Affichage du chemin pour accéder au dossier d'uploads*

Le logo de l'entreprise ainsi que les fichiers csv seront contenus dans ce dossier.

## Découpage du plugin au niveau fonctionnel

Voir mon rapport de stage (Partie 4).

## BDD

Préfix de la BDD : test.

3 tables créées par le plugin où des requêtes SQL se font :

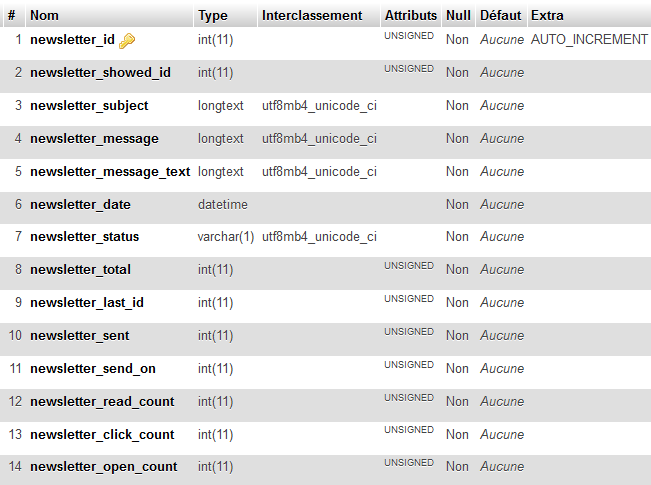
* préfix\_atm\_nl\_newsletter : Gestion des nl.
* préfix\_atm\_nl\_stat : Gestion des stats.
* préfix\_atm\_nl\_subscriber : Gestion des personnes.

Utilisation de 2 autres tables déjà créées lors de l'installation WP :

* préfix\_options : Gestion des params uniques tels que les infos du serveur STMP, les préférences de l'utilisateur pour ses nl (`option\_name LIKE '%atm\_nl%'`).
* préfix\_posts : Gestion des posts par WP. Requêtage sql sur cette table pour alimenter les nl.

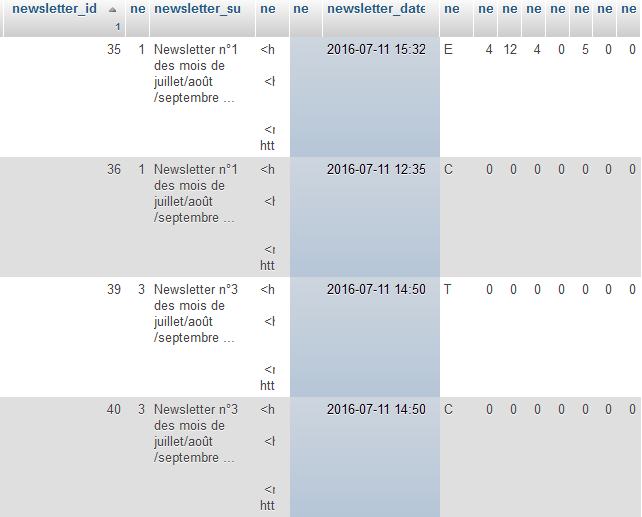
Par convention un sub est l'abréviation de subscriber. Un subscriber est un abonné en français. Ce sont les subs qui nous intéressent le plus d'où ce nom de table préfix\_atm\_nl\_subscriber même si elle contient plus que les subs.

### préfix\_atm\_nl\_newsletter



*Structure de la table préfix\_atm\_nl\_newsletter*

* newsletter\_id : L'id.
* newsletter\_showed\_id : L'id de la nl montré à l'utilisateur.
* newsletter\_subject : Le titre de la nl.
* newsletter\_message : Tout le contenu de la nl est dans ce champs.
* newsletter\_message\_text : Le message qui envoie vers un lien pour lire la nl dans leur navigateur pour les subs qui ne peuvent pas lire le HTML (pas fait).
* newsletter\_date : Date à laquelle la nl a été créée. Peut être amené à changer.
* newsletter\_status : Statut de la nl. Peut être amené à changer.
* newsletter\_total : Le nombre d'emails total envoyé de cette nl.
* newsletter\_last\_id : Le dernier id de sub à qui la nl a été envoyée.
* newsletter\_sent : Le nombre d'emails de cette nl envoyée à un instant T.
* newsletter\_send\_on : Le nombre d'emails de cette nl qui reste à envoyer à un instant T.
* newsletter\_read\_count : Le nombre total d'emails ouverts par les subs pour cette nl.
* newsletter\_click\_count : Le nombre total de clics qui ont été faits par les subs sur cette nl (pas fait).
* newsletter\_open\_count : Le nombre total de clics qui ont été faits par les subs générant des ouvertures de pages sur notre site Internet (pas fait).



*Exemple de données insérées dans la table* *préfix\_atm\_nl\_newsletter (le nom des colonnes a été rétréci pour pouvoir afficher toutes les donnés en une seule capture d'écran mais elles sont dans le même ordre que la précédente capture d'écran)*

newsletter\_id et newsletter\_showed\_id peuvent avoir des id différents pour une même nl. En effet pour chaque création de nl je crée deux nl en BDD : une sans les infos de tracking et une seconde avec les infos de tracking. Sur la capture d'écran j'ai deux nl. Une qui a le n°3 et une autre qui a le n°1. Les newsletter\_id 35 et 36 correspondent à la nl n°1 et les newsletter\_id 39 et 40 correspondent à la nl n°3 .

Il existe plusieurs statuts pour une nl :

* C : Créée. La nl a été créée mais n'a pas été envoyée.
* E : Envoyée. La nl a été envoyée à tous les subs à moment donné.
* T : Tracking. Cette nl contient le pixel espion qui nous ramènera les informations de tracking.

Pour chaque nl qui contiendra les infos de tracking aura le statut T lors de sa création puis le statut E lorsqu'elle aura été envoyée. Pour les nl qui ne contiendront pas de tracking leur statut sera toujours C.

Lorsqu'une nl est envoyée ses infos en BDD se mettent à jour (seulement la nl ayant le statut de départ T est modifiée) :

* newsletter\_date MAJ : On passe de la date de sa création à la date où tous les emails de cette nl ont été envoyés. Sur la capture d'écran on voit pour la nl qui a le newsletter\_showed\_id n°1 (donc les newsletter\_id n°35 et 36) une date de création le 11 juillet 2016 à 12h35 et une date d'envoi le même jour à 15h32. La date d'envoi sera forcément ultérieure à celle de création.
* newsletter\_date : Passe de T à E.

Pour pouvoir suivre en temps réel l'envoi des emails d'une nl aux subs il y a des colonnes dans la BDD qui dont destinées à cela.

* newsletter\_total : Commence de 0 au nombre de subs inscrits lors de l'envoi de la nl.
* newsletter\_last\_id : Se met à jour à chaque envoi d'email en fonction de l'id du sub auquel la nl est en train d'être envoyée. Plus les envois d'emails avanceront, plus la valeur de newsletter\_last\_id sera grande. A la fin de l'envoi de la nl, newsletter\_last\_id sera égal au dernier id du sub inscrit dans la BDD au moment T.
* newsletter\_sent : Commence à partir 0 pour finir à la valeur newsletter\_total. A chaque envoi d'emails il y a incrémentation de newsletter\_sent.
* newsletter\_send\_on : Commence à la valeur de nl\_total pour finir à 0.A chaque envoi d'emails il y a décrémentation de newsletter\_send\_on.

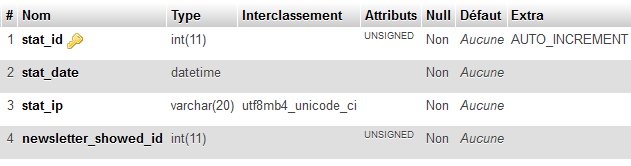
Tout compte fait pendant l'envoi et après l'envoi, la valeur numérique de newsletter\_total sera toujours égale à celle de newsletter\_sent plus celle de newsletter\_send\_on.

On peut également effectuer des statistiques sur les nl grâce à des colonnes dans a BDD.

* newsletter\_read\_count : La technique du pixel espion nous permet de stocker les logs d'ouvertures des emails pour une nl.
* newsletter\_click\_count : (pas fait).
* newsletter\_open\_count : (pas fait).

### préfix\_atm\_nl\_stat

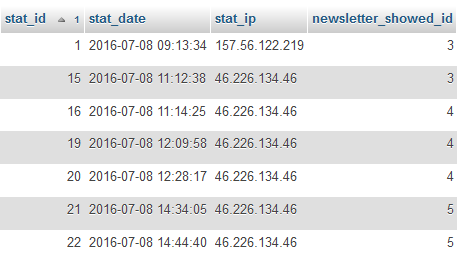
Cette table est une table de transition entre le fichier pixel.log et la table préfix\_atm\_nl\_newsletter.



*Structure de la table préfix\_atm\_nl\_stat*

* stat\_id : L'id.
* stat\_date : La date où l'email a été lu par le sub.
* stat\_ip : L'IP du sub qui a lu l'email.
* newsletter\_showed\_id : Le newsletter\_showed\_id nl qui est tout compte fait le mail lu par le sub. Devrait être clé étrangère avec les UPDATE et les DELETE ON CASCADE (pas fait).

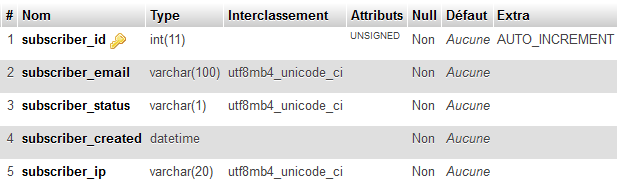
Avec la technique du pixel espion on stocke les informations des subs qui lisent les nl dans un fichier nommé pixel.log. Ce fichier est lu et alimente la table préfix\_atm\_nl\_stat. Après l'alimentation de cette table, la table préfix\_atm\_nl\_newsletter et plus particulièrement les colonnes (newsletter\_read\_count, newsletter\_click\_count et newsletter\_open\_count) sont MAJ.



*Exemple de données insérées dans la table* *préfix\_atm\_nl\_stat*

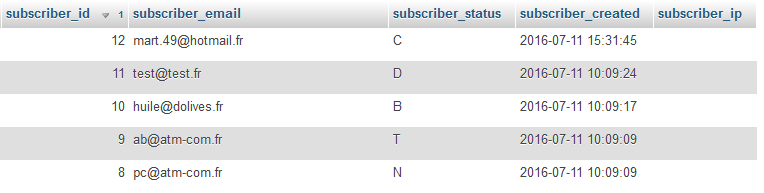
Sur cette capture d'écran on peut voir quelques lectures de nl à des périodes différentes de la journée. Comme tous mes tests se déroulent chez ATM l'IP reste la même. Pour avoir une IP différente (stat\_id égal à 1) il a fallu lire l'email depuis un terminal mobile en utilisant sa propre connexion Internet (pas celle d'ATM).

### préfix\_atm\_nl\_subscriber



*Structure de la table préfix\_atm\_nl\_subscriber*

* subscriber\_id : L'id de la personne.
* subscriber\_email : Son email.
* subscriber\_status : Le statut de la personne. Peut être amené à changer.
* subscriber\_created : La date où la personne a été ajoutée dans cette table.
* subscriber\_ip : L'IP de la personne (pas fait). Quelle IP prendre ? Celle de son ordinateur fixe ? Celle de sa tablette, de son téléphone portable ? Comment trouver l'IP d'une personne juste avec l'envoi d'un email ? Si l'email est transféré à quelqu'un et que cette personne non sub le lit comment fait-on ?



*Exemple de données insérées dans la table* *préfix\_atm\_nl\_subscriber*

Sur cette capture d'écran on peut voir quelques personnes ajoutées à la BDD. Chaque personne a été ajoutée à des moments différents de la journée du 11 juillet 2016 (soit par fichier .csv ou soit manuellement). On peut également remarquer qu'elles ont ses statuts différents :

* B : Blacklisté. Une personne blacklistée ne pourra plus recevoir d'emails provenant du plugin.
* C : Confirmé. Une personne confirmée recevra tous les prochains envois d'emails de nl.
* D : Désinscrit/Désabonné. C'est une personne qui s'est désinscrite via le lien "Désinscription" en bas à droite de la nl. Par conséquent elle ne recevra plus de nl à l'avenir.
* N : Non-confirmé. C'est une personne qui souhaite recevoir nos nl mais qui n'a pas confirmé son inscription via la technique du double opt-in.
* T : Testeur. Personne vers laquelle on pourra envoyer nos tests de nl.

Le statut d'une personne est amené à changer en fonction de ses actions qu'il fera (la plupart du temps) et des actions de l'administrateur du plugin (très rare). Les actions de l'administrateur sont expliquées dans la fonctionnalité "Changement de statut".

## Choix personnels

* Pas de CRON pour l'envoi de nl.
* Deux nl créées à chaque fois. Ce n'est pas optimisé et donc cela ne répond pas à une dynamisation de passage de variables avec la fonction php str\_replace().

## Amélioration possibles sur le plugin

On pourra trouver dans le fichier atm\_newsletter\_md.php une autre table qui est en commentaires : préfix\_atm\_nl\_sent. Cette table permet d'avoir une gestion plus précise des envois de nl aux personnes.

# Projet Webmarketing

# Mémento Rapide

Statuts des nl :

* C : Créée. Nl créée mais non envoyée.
* E : Envoyée. Nl envoyée à tous les subs à instant T.
* T : Tracking. Nl contenant le pixel espion qui nous ramènera les informations de tracking.

Statuts des personnes :

* B : Blacklisté. Une personne blacklistée ne recevra aucune nl.
* C : Confirmé. Une personne confirmée tous les envois d'emails de nl.
* D : Désinscrit/Désabonné. Personne qui s'est désinscrite des nl via le lien contenu dans les nl. Ne recevra plus de nl.
* N : Non-confirmé. Personne qui souhaite recevoir nos nl mais qui n'a pas confirmé son inscription via le double opt-in.
* T : Testeur. Personne vers laquelle on pourra envoyer nos tests de nl.

Déroulement de la technique du double opt-in :

* Formulaire où l'on saisi son adresse mail.
* Envoi d'un mail à cette adresse.
* Dans le mail il faut cliquer sur un lien pour confirmer que l'on veut recevoir nos emails.
* Quand on a cliqué sur le lien du mail en question, une page s'affiche avec notre confirmation à l'abonnement.

Abréviations :

* Arguments -> args.
* C'est-à-dire -> càd.
* Formulaire/s -> form.
* Réseau social -> rs.
* Subscriber -> sub.