



MindInvader Manuel d'installation

MindInvader est un système marketing offrant des possibilités d'analyse de retour publicitaire, d'optimisation des communications par rotation aléatoire d'alternatives et un programme de partenariat qui repose sur une technologie de tracking efficace.

MindInvader permet de déterminer le retour sur investissement des actions publicitaires sur internet (ou non) même si les conversions se font hors ligne, ou si lorsqu'elles se font en ligne, le client utilise une autre machine au moment de régler que celle qu'il a utilisé lorsqu'il a cliqué sur la publicité.

Le tracking de publicités hors lignes nécessite toutefois l'utilisation de sous domaines ou domaines dédiés à chaque campagne dont les résultats peuvent être intégrés ou dissociés de ceux du site principal.

Le système utilise une interface d'administration <http://app.mindinvader.com> et une librairie client écrite en php à installer sur le site et qui devra être exécutée à chaque visite, à chaque fois qu'une adresse email est collectée, à chaque fois qu'une commande est passée et à chaque fois qu'une transaction est confirmée.

Ce manuel traite de l'installation de cette librairie dans le site client.



1. Préparation

Pour ne pas fausser les données avec le comportement des anciens clients et prospects, il est hautement recommandé de récupérer ceux ci afin de les identifier comme source publicitaire ancienne "old".

Ce type de personne est plus prompt à faire de nouvelles transactions, et peut cliquer sur une nouvelle publicité. Si certaines sources publicitaires ont plus de chances que d'autres d'être vues par ces personnes, elles peuvent alors être mal interprétées comme plus rentables alors que cette rentabilité supérieure n'est liée qu'à la présence de personnes qui sont déjà en relation avec l'entreprise.

En l'absence de source publicitaires identifiée, les transactions sont associées au trafic naturel ! Les transactions des anciens clients peuvent donc être confondues avec le trafic naturel.

Ce point est d'autant plus important que l'activité de l'entreprise est ancienne et que le trafic des anciens clients et prospects peut venir brouiller le nouveau trafic.

Pour y parvenir facilement, le script : **/mindinvader-api-client/tool/synchronisation.php** est un bon exemple générique. Dans la plupart des cas, seules les requêtes de connection à la base client devront être modifiées pour qu'elles récupèrent les bonnes données. La transmission étant gérée par la librairie

Le données à récupérer sont celle décrites dans la partie *3 B - Inscription d'une adresse email dans MindInvader*



2. Tracking des visites

A. Configuration

Dezipper mindinvader-api-client.zip et charger le dossier mindinvader-api-client à la racine de votre site.

Fichier configuration

/mindinvader-api-client/setting.php

```
<?php
define('MVR_TRACK_ID',4);
define('MVR_TRACK_KEY','MR1KLOTF2789OWGB');
define('MVR_COOKIE_NAME','mindinvader');
define('CBM_RETURN_FIELD','cbmc');
$aDoNotTrackVisitPaths = array();

$aNotificationUrlPaths = array();
define('SYSTEMPAY_KEY','656465749874654654');
?>
```

En général, il vous suffit de modifier **MVR_TRACK_ID** et **MVR_TRACK_KEY** avec vos identifiants. pour avoir en terminé avec la configuration de base.

Toutefois quelques options sont disponibles et sont décrites ci-dessous.

MVR_COOKIE_NAME n'a besoin d'être changé que si vous décidez de changer le nom des cookies dans l'interface.

CBM_RETURN_FIELD est à ignorer si CyberMailing Pro n'est pas utilisé. Sinon il doit correspondre au nom du champs renseigné pour MindInvader dans Cybermailing afin que CyberMailing et MindInvader fonctionne de concert.

\$aDoNotTrackVisitPaths est un tableau à une dimension que vous pouvez renseigner dans un souci d'optimisation pour éviter des opérations de tracking inutiles. Cela devrait être le cas pour des scripts ajax, des crontab ou des feedback qui ne représentent pas vraiment des pages visitées par des individus. Vous devriez aussi les renseigner pour les scripts qui utilisent les fonctions sendContact, sendOrder sendTransaction ou sendRefund définies ci-après afin d'éviter un double emploi.

exemple :

```
$aDoNotTrackPaths = array(
    '/ajax/',
    '/cron/daily.php',
    '/feedback/cybermailing.php',
```



```
        '/subscribe.php',  
        '/paysec/ogone.php'  
    );
```

Si l'une des chaînes contenues dans le tableau est trouvée dans le path d'un url visité, le tracking des visites ne s'effectuera pas.

(en reprenant l'exemple ci-dessus : <http://votresite.com/ajax/query.php> ne déclencherait pas le tracking des visites).

Si vous ne renseignez pas ce tableau et que le tracking s'effectue sur ce type de scripts, ces appels seront considérés comme des visites de visiteurs. Les données seront en générales plutôt correctes mais totalement inutiles.

- un appel ajax, trackera le bon visiteur, mais sera compté comme la visite d'une page.
- un appel crontab, ou un feedback envoyé par un autre serveur sera toujours effectué par la même Ip sans possibilité d'utiliser les cookies, cela sera considéré comme le même visiteur si les appels sont rapides mais comme un nouveau visiteur si ils sont plus espacés (la durée précise dépend du paramétrage de la durée de la mémoire des ips dans l'interface)
- un script d'inscription, enregistrera le bon visiteur mais le fera deux fois : une fois au démarrage du script (partie visite) et une fois lorsque l'adresse email aura été transmise à mindinvader par sendContact().

\$aNotificationUrlPaths n'est utile que les pour utilisateurs qui n'utilisent pas déjà leur propre url de feedback pour leur paiement sécurisé et qui ne souhaitent pas en développer un.

Si vous avez déjà votre propre url de feedback vous n'avez pas à vous en préoccuper.

\$aNotificationUrlPaths est un tableau associatif à 2 dimensions contenant les urls relatifs à la racine du site correspondants aux url de feedback qui seront appelés par les systèmes sécurisés. Le fonctionnement est décrit dans 9 - *Prise en charge complète de l'url secret du paiement sécurisé*.

Exemple :

```
$aNotificationUrlPaths = array(  
    'PayPal' => '/modules/paypal/ipn.php',  
    'SystemPay' => '/modules/systempay/feedback.php'  
);
```

Lorsque la librairie est installée par le biais du .htaccess, voir ci-dessous, lorsque le "path"(chemin) de l'url visité vaut l'une des valeurs contenue dans ce tableau, exemple : <http://votresite.com/modules/paypal/ipn.php> la librairie ne trackera pas les visites mais utilisera la classe CMVRTransaction. Celle-ci appellera alors PayPal.php ou SystemPay.php en fonction de l'url utilisé.

Vous pouvez compléter ce tableau avec CequeVousVoulez => /cequevousvoulez/fichier.php si vous avez créé le fichier de la classe correspondante /mindinvader-api-client/classe/CequeVousVoulez.php



SYSTEMPAY_KEY est la clef client de SystemPay et n'est utile que si vous utilisez SystemPay et que vous vous servez de la classe CMVRTransaction pour vérifier la transaction. Voir 9 - *Prise en charge complète de l'url secret du paiement sécurisé.*

B. Installation

La meilleure façon d'installer et d'utiliser la librairie client consiste à ajouter les 2 lignes ci-dessous à la fin du fichier .htaccess situé à la racine du site.

- La première ligne permet d'activer PHP même pour les extensions html et htm.
- La deuxième fait commencer toute exécution de PHP par l'exécution du tracking MindInvader.
- [/PATHSITE] est à remplacer par le chemin absolu qui mène à la racine du site.

Fichier .htaccess

```
AddHandler application/x-httpd-php .php .html .htm
php_value auto_prepend_file "[/PATHSITE]/mindinvader-api-client/init.php"
```

Le tracking des visites fonctionne alors pour toute page php, htm ou html située à la racine du site et en dessous. La provenance originelle des visiteurs est alors récupérée à partir des variables ajoutées aux urls de réception des publicités, puis associée définitivement à un no de cookie. Ainsi qu'à l'ip du visiteur pendant un certain temps (durée paramétrable dans l'interface).

Remarque : Une alternative consiste à faire démarrer chaque page php par le fichier init.php de la librairie client avec l'instruction suivante :

```
<?php
require_once('[/home/web/sitefolder]/mindinvader-api-client/init.php');
?>
```

Attention : Avec cette méthode, le tracking ne sera pas actif si vous oubliez de faire cette inclusion... En revanche, cette méthode dispense de renseigner \$aDoNotTrackVisitPaths.



3 - Tracking à partir de l'email

A. Generalités

La robustesse du tracking est augmentée en transmettant au système de tracking l'adresse email du visiteur dès que celle-ci est connue. La provenance publicitaire du visiteur est alors associée à cette adresse email.

Cette adresse email peut alors être utilisée dans les urls visités par le client pour réactiver régulièrement le cookie de visite, celui ci peut avoir disparu, le visiteur peut avoir changé de navigateur et enfin cela peut permettre d'associer des adresses emails supplémentaires aussi utilisées par ce visiteur.

Par exemple dans une newsletter envoyée à luc@yahoo.fr qui contient un lien vers <http://votresite.com/plusdinfos.php>, vous personnaliserez les liens contenus dans le message pour qu'ils apparaissent à Luc comme étant : <http://votresite.com/plusdinfos.php?email=luckyluc@yahoo.com>

Dans votre système d'envoi, il suffit, en général, d'écrire le lien comme ceci :
<http://votresite.com/plusdinfos.php?email=%Email%> (syntaxe cybermailing)

Du moment que la page plusinfo.php déclenche la librairie client du tracking MindInvader et que l'adresse email a été enregistrée auparavant dans le système de tracking (voir ci-dessous), lorsque Luc clique le lien, quelque soit son ordinateur, son navigateur ou l'état de ses cookies, Luc, et son origine publicitaire seront immédiatement identifiés, ce qui permettra de rétablir le cookie d'origine et de poursuivre le tracking normalement si il navigue sur d'autres pages, se réinscrit, ou même commande avec une autre adresse email !

B. Inscription d'une adresse email dans MindInvader

Sur la page de réception, après que Luc donne son adresse email on utilise la fonction ->sendContact();

Exemple d'enregistrement de contact complet :

```
<?php
$aCustomer = array(
    'gender' => 'M.', // M. Mme ou Mlle
    'firstname' => 'Luc',
    'name' => 'Lucky',
    'email' => 'luckyluc@yahoo.com',
    'company' => 'Sherifs Online Ltd'
);

$oMindInvader->sendContact($aCustomer);
?>
```



Champs obligatoire :

`$aCustomer['email']` : l'adresse email donnée par le client

Exemple d'enregistrement de contact minimal :

```
<?php
$aCustomer = array(
    'email' => 'luckyluc@yahoo.com',
);

$oMindInvader->sendContact($aCustomer);
?>
```

Les informations sont transmises à MindInvader et associées à l'identifiant de visite contenu dans le cookie. L'adresse email fournie peut alors être associée aux informations publicitaires de l'origine. Ce code doit être exécuté sur une page que Luc visite afin que la librairie accède à son cookie MindInvader.

A chaque fois que vous êtes en mesure de connaître une adresse email utilisée par Luc, vous devriez la transmettre à MindInvader avec `$oMindInvader->sendContact($aCustomer)` ;

Fonction	sendContact()
Arguments	Tableau <code>\$aCustomer</code> requis

`$aCustomer` :

Champs	Requis	Type	Remarque
email	x	50 chars max	
gender		4 chars max	défaut vide (illustratif)
firstname		32 chars max	défaut vide (illustratif)
name		32 chars max	défaut vide (illustratif)
company		32 chars max	défaut vide (illustratif)

C. Nouvelle visite d'un client connu

A chaque fois que Luc s'identifie (dans le cas d'un site membre) ou à chaque fois que vous envoyez un email à Luc (ce qui dans la grande majorité des cas est responsable de son retour sur le site), ajoutez :

?email=luckyluc@yahoo.com à l'url sur lequel il est redirigé.



4. Nouvelle Commande

Techniquement, cette étape pourrait être omise, mais en pratique, elle procure un gain de performance important et son implémentation est donc fortement conseillée.

Lors d'une commande, le client fournit une adresse email. Cette adresse est soit une adresse connue par MindInvader (section 3B) auquel cas l'origine publicitaire y est déjà associée, soit une adresse inconnue (ce qui arrive assez souvent).

C'est justement pour ce dernier cas que cette étape est importante car cette nouvelle adresse va pouvoir être associée au cookie présent et donc à l'origine publicitaire associée à ce cookie. Si le cookie a été régulièrement réactivé, grâce à une connection site membre ou des communications par email (section 2 C), l'origine publicitaire a de très très fortes chances d'avoir été correctement conservée par le cookie et la nouvelle adresse pourra alors y être associée.

Par ailleurs, à cette étape, vous générez en général un no de commande que vous allez pouvoir associer aux informations publicitaires et qui sera facilement identifiable lors de la phase suivante.

Lors d'une commande, vous devez donc transmettre les informations du client et les informations relatives à la commande notamment un no de commande et le montant à régler. Tout cela est réalisé par la fonction ->sendOrder() qui doit être utilisée dans un script visité par le client (script de génération de la commande ou page de récapitulation de la commande).

Exemple de commande complet :

```
<?php

$aCaisse = Array(
    'customOrderID' => 645674,
    'amount'=> 596.3,
    'currency' => 'EUR',
);

$aCustomer = Array(
    'email' => 'martin.dupont@gmail.com',
    'gender' => 'Mlle', // M. Mme ou Mlle
    'firstname' => 'Martine',
    'name' => 'Dupont',
    'company' => 'MD Services SARL',
);

$aPanier = Array(
    1 => array(
        'articleID' => 7895,
        'label' => 'Service no 1',
```




```
'ean13' => '761234567890',
'reference' => 'H24-1234',
'supplier_reference' => 'S1-80PX-UFZ',
'prix_ht_unit' => 350,
'qty' => 1
),
2 => array(
    'articleID' => 45,
    'label' => 'Boites en plastique 100x30x30',
    'ean13' => '782590512486',
    'reference' => 'H30-12',
    'supplier_reference' => 'V1-60Q-UFR',
    'prix_ht_unit' => 5,
    'qty' => 15,
),
);
```

```
$oMindInvader->sendOrder($aCaisse, $aCustomer, $aPanier);
```

Les champs suivants sont obligatoires:

- `$aCaisse['customOrderID']` : votre no de commande
- `$aCaisse['amount']` : le montant TTC de la commande
- `$aCustomer['email']` ; l'adresse email donnée par le client

Tous les autres champs sont optionnels et sont décrits dans la section suivante.

`$aPanier` peut être omis intégralement et être transmis (ou non) à la phase suivante (transactions)

Outre le fait de mieux décrire la commande, le panier transmis à MindInvader permet de peupler votre base de produits qui peut vous être utile pour déterminer votre marge opérationnelle. Transmettre le panier à cet étape permet de remplir plus rapidement votre base produit que si vous ne le faite qu'au moment des transactions.

Exemple de commande minimal :

```
<?php

$aCaisse = Array(
    'customOrderID' => 645674,
    'amount'=> 596.3,
);

$aCustomer = Array(
    'email' => martin.dupont@gmail.com,
);
```



```
$oMindInvader->sendOrder($aCaisse, $aCustomer);
```

Fonction	sendOrder()
Arguments	Tableaux \$aCaisse (requis), \$aCustomer (requis), \$aPanier(option)

\$aCaisse :

Champs	Requis	Type	Remarque
customOrderID	x	14 chars max	votre No de commande
amount	x	decimal 9999999.99 max	montant TTC que le client doit verser pour que la commande soit réglée.
currency		3 chars majuscule	défaut définit dans l'interface, valeurs 'EUR', 'USD', 'CAD'... liste complète sur

\$aCustomer :

Champs	Requis	Type	Remarque
email	x	50 chars max	
gender		4 chars max	défaut vide (illustratif)
firstname		32 chars max	défaut vide (illustratif)
name		32 chars max	défaut vide (illustratif)
company		32 chars max	défaut vide (illustratif)

\$aPanier (optionnel) : Tableau associatif à 2 dimensions

```
$aPanier = Array(  
    0 => $aArticle_0,  
    1 => $aArticle_1  
);
```

\$aArticle :

Champs	Requis	Type	Remarque
articleID	x	10 chars max	
label	x	255 chars max	
qty		entier	défaut 0, max 999 999 (utilisation possible pour calcul de la



			marge)
prix_ht_unit		decimal 9999999.99 max	défaut 0, prix de vente unitaire du produit (utilisation possible pour calcul de la marge)
ean13		13 chars max	défaut vide (illustratif)
reference		32 chars max	défaut vide (illustratif)
suplier_reference		32 chars max	défaut vide (illustratif)



5. - Tracking des transactions

Enfin vous recevez de l'argent.. mais comment le savez vous ? de qui provient il ? pour quelle raison ? quelle est votre marge opérationnelle ? et comment se décompose t'elle selon vos différentes sources publicitaires ? Voici tout l'objectif du système de tracking.

Le tracking des transactions nécessite de disposer d'un script (ou plusieurs) appelé(s) à chaque commande validée, que ce soit par votre administration lors d'encaissement de chèques ou plus fréquemment par votre système de paiement sécurisé.

L'idée générale consiste à informer MindInvader que tel client (identifié par son adresse email ou son No de commande) a réalisé un paiement d'un montant X. En précisant des frais relatifs à la transaction, ou en ayant préalablement paramétré l'interface à cet effet, MindInvader calculera une marge opérationnelle que vous pourrez alors comparer à chaque dépense publicitaire afin d'en déterminer le retour sur investissement.

En précisant l'objet du paiement (panier), vous retracerez plus facilement ce que Mindinvader fait, vous pourrez vous servir de MindInvader pour suivre vos commandes et ce dernier peuplera une base de produits qui pourra vous servir pour calculer (ou recalculer après coup) les frais relatifs aux transactions.

Pour tout ce qui touche aux frais ci-dessous, retenez le fait que vous pouvez, à tout moment, modifier la définition des frais pour recalculer la marge opérationnelle correspondant à une période donnée.

Vous pouvez donc démarrer sans frais (marge = CA), ou au moins utiliser ceux que vous connaissez, puis affiner vos résultats en prenant en compte des frais précis une fois que vous les aurez déterminés ou si ceux ci viennent à changer.

Exemple de Transaction complet :

```
<?php

$aCustomer = Array(
    'email' => 'martin.dupont@gmail.com',
    'gender' => 'Mlle', // M. Mme ou Mlle
    'firstname' => 'Martine',
    'name' => 'Dupont',
    'company' => 'MD Services SARL',
);

$caisse = Array(
    'customOrderID' => 'vente123456',
    'validationID' => 'FDHGDF5754DSGDS8',
    'mode' => 'paypal',
    'amount'=> 74.15,
    'currency' => 'EUR',
    'taxes'=> 12.15,
    'frais_port' => 8,
```



```
'frais_emb' => 4,
'frais_adm' => 3,
'frais_banc' => 2.5);

$panier = Array(
    0 => Array (
        'label' => 'Savon de marseille rose',
        'articleID' => '45785',
        'ean13' => '762224567850',
        'reference' => 'S12-1230',
        'supplier_reference' => 'MS33/44',
        'frais_port_unit' => 0,
        'frais_emb_unit' => 0,
        'frais_adm_unit' => 0,
        'frais_banc_unit' => 0,
        'frais_prod_unit' => 2,
        'prix_ht_unit' => 8,
        'qty' => '2'),
    1 => Array (
        'label' => 'Serviette de bain coton 100x180 bleue',
        'articleID' => '9578',
        'ean13' => '1456424567850',
        'reference' => 'SB5-1234',
        'supplier_reference' => 'ser12',
        'frais_port' => 0,
        'frais_emb_unit' => 0,
        'frais_adm_unit' => 0,
        'frais_banc_unit' => 0,
        'frais_prod_unit' => 4,
        'prix_ht_unit' => 18,
        'qty' => '1'),
    2 => Array (
        'label' => 'miroir style ancien 80x30',
        'articleID' => '16785',
        'ean13' => '2864843767850',
        'reference' => 'MS2-1235',
        'supplier_reference' => 'VB47.235',
        'frais_port_unit' => 0,
        'frais_emb_unit' => 0,
        'frais_adm_unit' => 0,
        'frais_banc_unit' => 0,
        'frais_prod_unit' => 15,
        'prix_ht_unit' => 28,
        'qty' => '1')
    );

$omMindInvader->sendTransaction($aCaisse, $aCustomer, $aPanier);

?>
```



Les champs requis sont :

- \$aCaisse['amount'] un nombre décimal, montant versé par le client (ex: 55.7)
- \$aCustomer['email'] (*) // adresse email du client, celle qui est déjà connue si possible.
- \$aCaisse['customOrderID'] (*) // votre no de commande.

(*) Ces champs ne sont pas toujours requis mais s'ils sont disponibles, par simplicité transmettez les !

La logique complète est celle-ci :

Si sendOrder() a été utilisé en phase 3, \$aCustomer['email'] n'est pas obligatoire. \$aCaisse['customOrderID'] n'est obligatoire que si il a été transmis par sendOrder()

Toutes les autres données sont optionnelles.

Recommandation :

Vous gagnez à transmettre un maximum de données et à ne pas vous limiter au minimum. Le minimum n'existe que pour les cas où il n'est pas possible de faire faire autrement.

Certaines données optionnelles n'ont qu'une fonction illustrative pour rendre plus lisible la base de donnée de MindInvader, c'est le cas de "gender", "firstname", "name", "company" du tableau \$aCustomer et des références du panier "ean13", "reference" et "suplier_reference". Les autres données, en revanche, ont une fonction propre lorsqu'elles sont présentes.

Les données optionnelles (qui peuvent avoir une fonction) sont :

- \$aCaisse['mode'] peut servir à regrouper les données par mode de paiement et est utilisé pour créer un no de commande si vous n'en transmettez pas.
- \$aCaisse['currency'] s'il n'est pas transmis, le montant sera interprété comme exprimé dans la devise définie par défaut dans l'interface. Sinon si la devise indiquée est différente de la devise par défaut, les montants transmis seront convertis dans la devise par défaut au dernier taux de change connu (actualisé toutes les heures).
- \$aCaisse['validationID'] est le no de validation qui garantit que la transaction ne sera bien comptabilisée qu'une seule fois, même si l'information est transmise une autre fois. Si vous ne transmettez pas celui du paiement sécurisé, la librairie créera un identifiant arbitraire.

Calcul de la marge opérationnelle :

Le calcul du ROI passe par la détermination d'une marge opérationnelle qui doit être calculée à chaque transaction.

Si vous vendez un service 2500 TTC mais qu'il vous en coute 1200 en salaire pour le réaliser votre marge n'est que de $2500 - \text{Taxes (500)} - \text{Salaire (1200)} = 800 \text{ €}$

A supposer que vous n'ayez pas d'autres frais (ce qui est peu probable), s'il vous en coute 600 € de publicité pour faire une vente, vous avez un ROI publicitaire de seulement 33% et non de 216% comme le suggèrera



un calcul naïf réalisé uniquement à partir du montant ttc !

Parce que vos marges peuvent varier d'une vente à l'autre, MindInvader rend possible la comptabilité de divers frais afin de pouvoir déterminer la marge de chaque transaction avec précision et flexibilité.

Vous pouvez aussi bien ignorer les frais, considérer un frais fixe ou un pourcentage fixe à chaque transaction, ou bien déterminer les frais pour chaque produit. L'ensemble pouvant être fait à la volée (chaque transmission de transaction) ou prédéterminé par l'interface ou une combinaison des deux. Par ailleurs les règles déterminées dans l'interface peuvent être modifiées à posteriori et la marge de vos transactions passées recalculée.

La formule suivante est utilisée :

$$Marge = Montant_{TTC} - Taxes - Frais_{transaction}$$

avec

$$Frais_{transaction} = Frais_{banc} + Frais_{port} + Frais_{emb} + Frais_{prod} + Frais_{adm}$$

et les articles i du panier étant en quantité respective Qty_i :

$$Frais_{port} = Frais_{port}_{caisse} + \sum_i^{panier} ([Frais_{port} unit]_i * Qty_i)$$

Frais de transport nécessaires pour livrer l'objet de la transaction

$$Frais_{emb} = Frais_{emballage}_{caisse} + \sum_i^{panier} ([Frais_{emballage} unit]_i * Qty_i)$$

Frais d'emballages relatifs à l'objet de la transaction

$$Frais_{adm} = Frais_{administratif}_{caisse} + \sum_i^{panier} ([Frais_{administratif} unit]_i * Qty_i)$$

Frais administratifs, en général des frais globaux qui se prêtent mal à une évaluation par transaction. Toutefois vous pourrez déterminer régulièrement la moyenne et ces frais pour chaque transaction et les appliquer ici. Ou les recalculer par le biais de l'interface.

$$Frais_{banc} = Frais_{bancaire}_{caisse} + \sum_i^{panier} ([Frais_{bancaire} unit]_i * Qty_i)$$

Frais bancaires relatifs à la transaction (en général un pourcentage du montant ttc pour les systèmes de paiement sécurisés par internet). La composante unitaire par produit du panier devrait en général être nulle ou ignorée.

$$Frais_{prod} = Frais_{production}_{caisse} + \sum_i^{panier} ([Frais_{production} unit]_i * Qty_i)$$

Frais de production, cela peut être le prix grossiste dans les cas d'achat/vente ou la masse salariale dans le cas de vente services.

Les éléments de cette formule "montant_ttc" et "taxes" sont transmis à chaque transaction.

Les éléments composant les frais de la transaction sont soit transmis à chaque transaction soit calculés à partir de coefficients et/ou valeurs fixes renseignés dans l'interface. Soit les deux.



L'interface permet donc aussi de déterminer à l'avance chacune des 5 composantes des frais.

Sale **Refund**

Transaction costs for sales

Displaying 1-5 of 5 results.

Shipping Cost = 0.00 EUR + (0.00 % / 100 * Tax Inclusive Transaction amount)

Fixed Cost per transaction	<input type="text" value="0.00"/>	EUR
Cost Percentage of transaction amount	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost is not given by conversion tracking call ▼	
<input type="button" value="Submit"/>		

Packaging Cost = 0.00 EUR + (0.00 % / 100 * Tax Inclusive Transaction amount)

Fixed Cost per transaction	<input type="text" value="0.00"/>	EUR
Cost Percentage of transaction amount	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost from conversion call is zero ▼ Always use this formula Use this formula if cost is not given by conversion tracking call Use this formula if cost from conversion call is zero Use this formula if cost from conversion call is missing or zero Never use this formula	

Admin Cost = 0.00 EUR + (0.00 % / 100 * Tax Inclusive Transaction amount)

Fixed Cost per transaction	<input type="text" value="0.00"/>	EUR
Cost Percentage of transaction amount	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost is not given by conversion tracking call ▼	
<input type="button" value="Submit"/>		

Banking Cost = 0.00 EUR + (0.00 % / 100 * Tax Inclusive Transaction amount)

Fixed Cost per transaction	<input type="text" value="0.00"/>	EUR
----------------------------	-----------------------------------	-----



De manière similaire, chaque produit renseigné dans l'interface peut avoir son jeu de frais. Les produits peuvent être entrés manuellement, ou seront récupérés au fur et à mesure des commandes lorsqu'elles transmettent un contenu de panier. Le pourcentage est ici appliqué au prix de vente unitaire HT du produit précisé dans le panier.

Exemple pour un produit X

Sale Costs	Refund Costs	Product Reference
------------	--------------	-------------------

The cost components related to each sale of a single item of this product

Product Sale Cost = Product Quantity * (Shipping Cost + Packaging Cost + Administrative Cost + Banking Cost)

Shipping Cost = 0.00 + (0.00 / 100 * Tax Free Unit Product Price) EUR

Fixed Cost	<input type="text" value="0.00"/>	EUR
Percentage Cost	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost is not given by conversion tracking call ▼	
<input type="button" value="Submit"/>		

Packaging Cost = 0.00 + (0.00 / 100 * Tax Free Unit Product Price) EUR

Fixed Cost	<input type="text" value="0.00"/>	EUR
Percentage Cost	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost is not given by conversion tracking call ▼	
Always use this formula		
Use this formula if cost is not given by conversion tracking call		
Use this formula if cost from conversion call is zero		
Use this formula if cost from conversion call is missing or zero		
Never use this formula		

Admin Cost = 0.00 + (0.00 / 100 * Tax Free Unit Product Price) EUR

Fixed Cost	<input type="text" value="0.00"/>	EUR
Percentage Cost	<input type="text" value="0.00"/>	%
Rule	Use this formula if cost is not given by conversion tracking call ▼	
<input type="button" value="Submit"/>		

Pour chaque type de frais on a donc la formule possible :



$$Frais_{type} = \oplus [Frais\ Type_{caisse}]_{script} \oplus [Frais\ Type_{caisse}]_{interface} \oplus [Frais\ Type_{panier}]_{script} \oplus [Frais\ Type_{panier}]_{interface}$$

Le symbole \oplus signifie que les éléments sont ajoutés ou non en fonction de la logique choisie dans l'interface.

De cette façon, tous les cas de figures, même ceux qui nécessitent une certaine complexité pour l'élaboration des frais sont possibles.

Si le mode de calcul proposé par l'interface vous convient, vous pouvez donc vous dispenser de transmettre ces frais à chaque transaction. Si certains frais sont plus faciles à transmettre lors de chaque transaction (comme par exemple les frais bancaires s'ils sont transmis par le système de paiement sécurisé) vous pouvez choisir de les transmettre ces frais à chaque transaction, tandis que d'autres seront déterminés par des paramètres fixes de l'interface.

De plus, il est possible de revenir à tout moment sur le calcul de marge en modifiant les paramètres de l'interface et en redemandant un calcul. Il existe cependant une limitation : ce "re-calcul" peut ignorer les frais précédemment transmis ou non mais ces derniers ne peuvent pas être modifiés.

► Understanding Formula Components

► Define Transaction Costs

► Define Product Costs

▼ Recalculate Margins on past transactions

09/11/2012 11/12/2013 ReCalculate

☒ Ignore costs (= only considers costs as defined in the interface)

November 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Copyright © 201
Tous droits réservés

Résumé :

Après avoir peuplé \$aCaisse, \$aCustomer et \$aPanier



Vous appelez directement :

```
$oMindInvader->sendTransaction($aCaisse, $aCustomer, $aPanier);
```

ou

```
$oMindInvader->sendTransaction($aCaisse, $aCustomer);
```

ou

```
$oMindInvader->sendTransaction($aCaisse);
```

Cette dernière version minimaliste est valable uniquement si \$aCaisse contient customerOrderId et que vous avez transmis cette valeur avec sendOrder() en phase 3

D'une manière générale nous recommandons d'utiliser systématiquement :

```
$oMindInvader->sendTransaction($aCaisse, $aCustomer, $aPanier);
```

Fonction	sendTransaction()
Arguments	Tableaux \$aCaisse (requis), \$aCustomer (requis*), \$aPanier (option) \$aCustomer n'est pas requis si vous avez déjà transmis une adresse email et un No de commande avec sendOrder()

Description de \$aCaisse : Tableau associatif

Champs	Requis	Type	Remarque
customOrderID	x	14 chars max	votre No de commande
amount	x	decimal 9999999.99 max	maximum 9999999.99, montant TTC versé par le client
currency		3 chars majuscule	défaut défini dans l'interface, EUR, USD, CAD.. liste complète sur
taxes		decimal 9999999.99 max	défaut 0, maximum 9999999.99, montant des taxes que vous devez reverser
validationID		40 chars max	défaut reconstruit, identifiant de la transaction qui permet de garantir l'unicité du traitement



mode		10 chars max	défaut 'unknown', 'identifiant du système de paiement sécurisé, exemple paypal
frais_banc		decimal 9999999.99 max	Frais bancaires reliés à la transaction (utilisation possible pour le calcul de la marge)
frais_port		decimal 9999999.99 max	Frais de port reliés à la transaction (utilisation possible pour le calcul de la marge)
frais_emb		decimal 9999999.99 max	Frais de d'emballage reliés à la transaction (utilisation possible pour le calcul de la marge)
frais_prod		decimal 9999999.99 max	Frais de production reliés à la transaction (utilisation possible pour le calcul de la marge)
frais_adm		decimal 9999999.99 max	Frais de administratifs reliés à la transaction (utilisation possible pour le calcul de la marge)

Description de \$aCustomer : : Tableau associatif

Champs	Requis	Type	Remarque
email	x	50 chars max	
gender		4 chars max	défaut vide (illustratif)
firstname		32 chars max	défaut vide (illustratif)
name		32 chars max	défaut vide (illustratif)
company		32 chars max	défaut vide (illustratif)

Description de \$aPanier (optionnel) : : Tableau associatif à 2 dimensions

```
$aPanier = Array(  
    0 => $aArticle_0,  
    1 => $aArticle_1  
);
```

Description de \$aArticle_X : Tabeau associatif

Champs	Requis	Type	Remarque
articleID	x	10 chars max	
label	x	255 chars max	
qty		entier	defaut 0, max 999 999
prix_ht_unit		decimal(9,2)	défaut 0, max 9999999.99



ean13		13 chars max	défaut vide (illustratif)
reference		32 chars max	défaut vide (illustratif)
suplier_reference		32 chars max	défaut vide (illustratif)
frais_banc_unit		decimal 9999999.99 max	Frais bancaires unitaires attribués à ce type d'article (utilisation possible pour le calcul de la marge)
frais_port_unit		decimal 9999999.99 max	Frais de port unitaires attribués à ce type d'article (utilisation possible pour le calcul de la marge)
frais_emb_unit		decimal 9999999.99 max	Frais de d'emballage unitaires attribués à ce type d'article (utilisation possible pour le calcul de la marge)
frais_prod_uni		decimal 9999999.99 max	Frais de unitaires attribués à ce type d'article (utilisation possible pour le calcul de la marge)
frais_adm_unit		decimal 9999999.99 max	Frais de administratifs unitaires attribués à ce type d'article (utilisation possible pour le calcul de la marge)

Exemples minimalises :

Transmission uniquement à la confirmation de la transaction :

```
<?
$aCaisse = Array('customOrderID' => '1234','amount'=> 74.15);
$oCustomer = Array('email' => 'joel@gmail.com',);
$oMindInvader->sendTransaction($aCaisse, $aCustomer, $aPanier);
```

Transmission à la nouvelle commande et la confirmation de la transaction

1 à la commande sur une page visitée :

```
$aCaisse = Array('customOrderID' => '1234','amount'=> 74.15);
$oCustomer = Array('email' => 'joel@gmail.com',);
$oMindInvader->sendOrder($aCaisse, $aCustomer);
```

2 à la transaction :

```
$aCaisse = Array('customOrderID' => '1234');
$oMindInvader->sendTransaction($aCaisse);
```

ou encore avec un panier :



```
$oMindInvader->sendTransaction($aCaisse,$aPanier);
```

Exemples classiques :

Pour une boutique :

```
$caisse = Array(
    'customOrderID' => '156AZ5654',
    'currency' => 'EUR',
    'amount' => 74.15,
    'taxes' => 12.15,
    'frais_emb' => 4,
    'frais_adm' => 3,
    'frais_banc' => 2.5);

$customer = Array('email' => 'joel@gmail.com');

$panier = Array(
    0 => Array (
        'label' => 'Savon de marseille rose';
        'articleID' => '45785';
        'frais_port_unit' => 3;
        'frais_prod_unit' => 2,
        'prix_ht_unit' => 8,
        'qty' => '2'),
    1 => Array (
        'label' => 'Serviette de bain coton 100x180 orange';
        'articleID' => '9578';
        'frais_port' => 3;
        'frais_prod_unit' => 4,
        'prix_ht_unit' => 18,
        'qty' => '1'),
    2 => Array (
        'label' => 'porte-serviette style ancien 80x30';
        'articleID' => '16785';
        'frais_port_unit' => 8;
        'frais_prod_unit' => 15,
        'prix_ht_unit' => 28,
        'qty' => '1')
);

$oMindInvader->sendTransaction($caisse,$customer,$panier);
```



Pour une vente de produit numérique :

```
$caisse = Array(
    'customOrderID' => '546752'
    'currency' => 'EUR',
    'amount'=> 74.15,
    'taxes'=> 12.15,
    'frais_adm'  => 3,
    'frais_banc' => 2.5
);

$customer = Array('email' => 'joel@gmail.com');

$panier = Array(
    0 => Array (
        'label' => 'Comment produire du savon soi même';
        'articleID' => '5';
        'frais_prod_unit' => 0,
        'prix_ht_unit' => 62)
    );

$soMindInvader->sendTransaction($caisse,$customer,$panier);
```

Pour une vente de service :

```
$caisse = Array(
    'currency' => 'EUR',
    'amount'=> 1574,
    'taxes'=> 257.95,
    'frais_adm'  => 15,
    'frais_banc' => 2.5);

$customer = Array('email' => 'joel@gmail.com');

$panier = Array(
    0 => Array (
        'label' => 'Etude etanchéité';
        'articleID' => 'E_INST';
        'frais_prod_unit' => 850,
        'prix_ht_unit' => 1316.05,
        'qty' => '2')
    );

$soMindInvader->sendTransaction($caisse,$customer,$panier);
```



6 Cas des Remboursements

Les remboursements sont traités comme des transaction en utilisant la fonction `sendRefund()`

La logique des champs requis est la même que pour le tableau `$aCaisse` de `sendTransaction()`.

En général, il suffit de transmettre le no de commande concernant le remboursement et le montant.

- Le montant est positif, c'est celui que vous reversez au client
- Le montant des taxes est positif, c'est le montant de taxes que vous pourrez retirer des taxes que vous avez précédemment collectées.
- Les frais sont soit positifs soit négatifs. Ils sont négatifs si par exemple le client vous a renvoyé le produit et que vous pouvez le revendre tel quel, dans ce cas le `Frais_production_unit` pourra être négatif. En revanche, si vous devez prendre à votre charge les frais bancaires, ils seront positifs tout comme vos frais emballage si vous devez refaire un conditionnement, ou une vérification du produit avant de pouvoir le remettre en vente.

Le calcul des frais fonctionne de la même manière que pour la transaction normale, vous pouvez utiliser l'interface pour définir des coefficients qui sont uniquement dédiés au cas des remboursements et/ou transmettre les valeurs avec les tableau `$aCaisse` et `$aPanier`.

Exemple simple :

```
$aCaisse = array(
    'customOrderID' => 'CM6724',
    'amount' => '150'
    'currency' => 'USD'
);
$oMindInvader->sendRefund($aCaisse);
```

Exemple de remboursement d'un ebook :

```
$aCaisse = Array(
    'customOrderID' => '546752'
    'currency' => 'EUR',
    'amount'=> 74.15,
    'taxes'=> 12.15,
    'frais_banc' => 0
);

$aPanier = Array(
    0 => Array (
```




```
'label' => 'Comment produire du savon soi même';  
'articleID' => '5';  
'frais_prod_unit' => -62,  
);  
  
$oMindInvader->sendRefund($aCaisse, $aPanier);
```



7 Tester votre installation

La librairie que vous venez d'installer dispose d'une fonctionnalité de reporting qui vous permet d'analyser les communications entre votre site et mindinvader. Pour l'utiliser, il suffit d'ajouter ?MVR_dbug=XXXXXXX à n'importe quel url pour lequel le tracking est actif en remplaçant XXXXXX par votre clef client (MVR_CLIENT_KEY)

```
Query sent to MindInvader =>
http://localhost/web/mindinvader.com/track/?
type=visit|MVR_clid==7|call_type==server|MVR_key==X3CI23IY6WU6EM90|URI==http://localhost/web/test-
mindinvader.com/?
MVR_dbug=X3CI23IY6WU6EM90&trackv=1|IP==127.0.0.1|MVR_trackid==1+511614bfd7a8741a66d8345d670e0766|2+d85d90
<= Json Response from MindInvader
stdClass::__set_state(array( 'trackinfo_display' => '
Connection base client OK - base no :60
Pas de email dans le lien !
Pas de hv dans le lien !
Le cookie du site est present, valeur : 1+511614bfd7a8741a66d8345d670e0766 donc VID+hash =
1+511614bfd7a8741a66d8345d670e0766 Confirmation : VID trouvée dans la table visitor : VID = 1
on a des infos contact pour ce VID prenons les dernières
- Tentative de récupération des infos dans la table contact à partir de CID
OK - Informations trouvées dans la base contact
email :
nom : Lucky
CID : 2
VID : 1

RUN :php /home/lionel/web/mindinvader.com/track/4stats.php 60 1 \'ses\' \'server\' \'2013-11-
05+15%3A54%3A02\'
RESULT : array ( 0 => \'\' , 1 => \'We are LOCAL\' , )
FIN Tracking :67.5 ms', 'tracking_status' => 1, 'tracking_method' => 'cookie site', 'access_id' => '60',
'cookie_domain' => array ( 0 => 'localhost', ), 'VID' => '1', 'vid2' => '0', 'GLOBVID' => '0', 'ad' =>
'0', 'source' => '2', 'campagne' => '1', 'groupe' => '1', 'annonce' => '1', 'motclef' => '1', 'search'
=> '1', 'typevis' => 'ses', 'cookie_days' => '1095', 'email' => 'luckyluc@yahoo.com', 'name' => 'Lucky',
'CID' => '2', 'optin' => 1, 'hv' => 'c4ca4238a0b923820dcc509a6f75849b', 'cookie_value' =>
'1+511614bfd7a8741a66d8345d670e0766|2+d85d980fe923a359a335b49348cdd20f', ))

Sending tracking cookie
Name: mindinvader | Domain: localhost | Value:
1+511614bfd7a8741a66d8345d670e0766|2+d85d980fe923a359a335b49348cdd20f | Validity: 1095 days | Folder:
/web/test-mindinvader.com => Success

MindInvader Logic Reporting :

Connection base client OK - base no :60
Pas de email dans le lien !
Pas de hv dans le lien !
Le cookie du site est present, valeur : 1+511614bfd7a8741a66d8345d670e0766 donc VID+hash =
1+511614bfd7a8741a66d8345d670e0766
Confirmation : VID trouvée dans la table visitor : VID = 1
on a des infos contact pour ce VID prenons les dernières
- Tentative de récupération des infos dans la table contact à partir de CID
OK - Informations trouvées dans la base contact
```



8. Transmission des adresses postales

Mindinvader peut stocker plusieurs adresses postales de vos contacts. Cela n'a pas d'influence sur le tracking mais peut vous être utile dans le cadre d'un suivi de commande où pour utiliser le CRM que nous envisageons de développer.

La transmission d'une adresse peut se faire de manière asynchrone (ne nécessite pas la présence du contact) et s'utilise de la façon suivante :

```
$aAddress = Array(
    0 => Array(
        'alias' => 'Bureau',
        'address' => '12 rue des petits champs',
        'city' => 'Toulon',
        'postcode' => '83000',
        'id_state' => '',
        'country' => 'FR',
        'company' => 'e-Genèse',
        'lastname' => 'Palazzi',
        'firstname' => 'Lionel',
        'phone' => '0494506285',
        'phone_mobile' => '0625854265',
    ),
    1 => Array(
        'alias' => 'Livraison',
        'address' => '12 rue des petits champs',
        'city' => 'Toulon',
        'postcode' => '83000',
        'id_state' => '',
        'country' => 'FR',
        'company' => 'e-Genèse',
        'lastname' => 'Palazzi',
        'firstname' => 'Lionel',
        'phone' => '0494506285',
        'phone_mobile' => '0625854265',
    )
);

$oMindInvader->sendContactAddress($aCustomer,$aAddress);
```

on y retrouve le tableau \$aCustomer utilisé lors des phases 2, 3 et 4 et un tableau \$aAddress contenant les adresses postales et le no de téléphone.

Description de \$aAddress (optionnel) : Tableau associatif à 2 dimensions

```
$aAddress = Array(
    0 => $aAddress_0,
    1 => $aAddress_1
```



);

Description de \$aAddress_X : Tableau associatif

Champs	Requis	Type	Remarque
alias	x	32 chars max	permet de donner un nom à l'adresse pour la distinguer entre plusieurs (maison, bureau, etc..)
company		32 chars max	défaut vide (illustratif)
lastname		32 chars max	défaut vide (illustratif)
firstname		32 chars max	défaut vide (illustratif)
address		128 chars max	défaut vide (illustratif)
postcode		12 chars max	défaut vide (illustratif)
city		64 chars max	défaut vide (illustratif)
id_state		10 chars max	défaut vide (illustratif)
country		64 chars max	défaut vide (illustratif)
phone		16 chars max	défaut vide (illustratif)
phone_mobile		16 chars max	défaut vide (illustratif)
active		0 ou 1	défaut 1
deleted		0 ou 1	défaut 0, pour supprimer l'affichage de l'adresse.



9. Prise en charge complète de la communication avec le paiement sécurisé.

Si vous utilisez PayPal ou SystemPay et que vous n'utilisez pas déjà l'appel secret de ces systèmes de paiement, vous pouvez obtenir une bonne partie des informations requises par MindInvader très rapidement.

Pour cela, il suffit de renseigner `$aNotificationUrlPaths` dans le fichier `settings.php` avec les urls d'appel secret des paiements sécurisés pour que la transmission des principaux champs de transaction soit prise en charge par la librairie à chaque fois que ces url sont appelés.

Toutefois, le résultat obtenu ne sera proportionnel qu'à la pertinence des informations disponibles lors de l'appel de votre url par le paiement sécurisé. Par exemple les frais ne seront pas connus tout comme le contenu du panier (dans le cas de `systemPay`).

Pour obtenir plus de précision, vous devrez certainement développer la classe `CMVRTransaction` pour récupérer dans votre bases de données les informations pertinentes afin d'obtenir un comportement similaire à la méthode directe (décrite en phase 4).