

Mode Shortcut

Contenu :

[Principe](#)
[Intégration](#)
[Feuille de style](#)
[Pages associées](#)

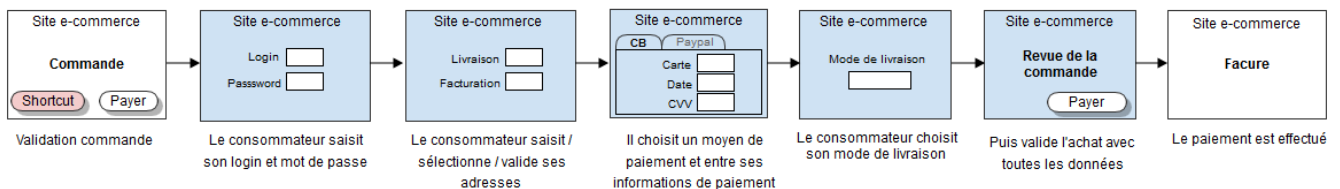
Principe

Avec ce mode d'intégration, les commerçants peuvent intégrer le bouton Payer en amont du parcours d'achat, directement sur le catalogue ou le panier au lieu de mettre un bouton sur la dernière étape de paiement.

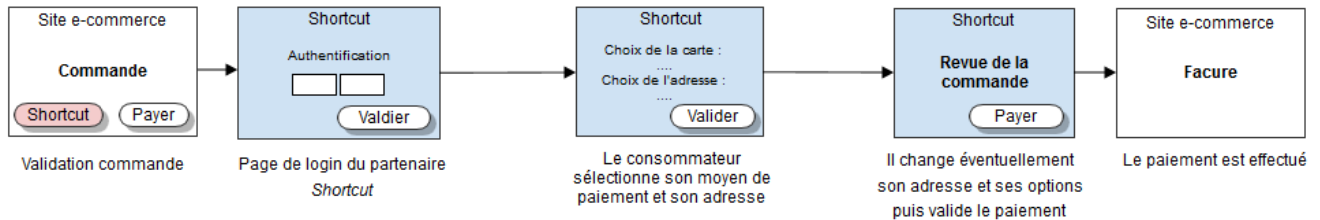
Ainsi, le consommateur évite les étapes de login/création de compte, choix de l'adresse de livraison, choix et saisie des informations carte de paiement.

Payline récupère les données du partenaire *Shortcut* et les met à disposition du commerçant.

Le parcours standard, sans *Shortcut* :



Le parcours avec *Shortcut* :



Les moyens de paiement disponibles sont : [Visa Check Out](#), [Paypal Express Checkout Shortcut](#), [ApplePay](#) [AmazonPay](#) et bientôt Google Pay et Samsung Pay.

Intégration

Pour intégrer ce mode de paiement, vous devez vérifier si le contrat est bien enrôlé chez l'acquéreur et configuré.

Ensuite vous devez intégrer le mode widget sur la page du panier.

1. Initialisation du paiement : depuis votre SI vous devez utiliser le service [doWebPayment](#) de l'API Web Payment pour initialiser un paiement sur les pages de paiement Payline. Cet appel ne doit comporter que des moyens de paiement compatibles *Shortcut*. Vous devez récupérer le jeton de session nommé token dans la réponse du service comme une intégration Widget classique.

2. Sur la page du panier : ajout d'un script [widget-min.js](#), d'un css [widget-min.css](#) et d'une balise <DIV> avec le paramètre [data-template="shortcut"](#).

Avec cette balise Payline affiche le/les bouton(s) *Shortcut* du/des moyen(s) de paiement avec le widget.

Code XML

```
<script src="https://homologation-payment.payline.com/scripts/widget-min.js" charset="utf-8"></script>
<link rel="stylesheet" href="https://homologation-payment.payline.com/styles/widget-min.css" charset="utf-8">
```

Rajouter la div PaylineWidget en template shortcut avec le jeton de session (data-token) obtenu dans la réponse à la demande d'initialisation d'un paiement avec le service [doWebPayment](#).

Cette balise doit comporter la référence à la fonction [callBack](#) de gestion des états du widget, pour plus d'informations, voir la documentation [API Callback](#).

Code XML

```
<div id="PaylineWidget" data-token="xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
      data-template="shortcut"
      data-event-didshowstate="widgetStateCallback"
      data-embeddedredirectionallowed="true">
```

3. Sur la page de paiement, à l'affichage d'un contexte [PAYMENT_TRANSITIONAL_SHORTCUT](#) via le callback [data-event-didshowstate](#) du Widget, il sera possible de récupérer les données acheteur (nom, prénom, adresse mail, ...) ainsi qu'à minima, son adresse de livraison.

Code Javascript

```
<script>
function widgetStateCallback(state) {
    if (state.state === "PAYMENT_TRANSITIONAL_SHORTCUT") {
        <!-- on cache le widget -->
        Payline.Api.hide();
        <!-- redirection vers une page gérant l'édition d'adresse de livraison, le choix du mode
livraison ...-->
        window.location.href="./paiementshortcut.php?token=" + Payline.Api.getToken();
    }
}
</script>
```

4. Les données acheteur : le SI Commerçant récupère l'adresse de livraison et des données acheteur via la fonction API [Payline.Api.getBuyerShortcut\(\)](#).

Avant le paiement, ces informations pourront être affichées pour proposer à l'utilisateur de les modifier / de choisir un mode de livraison. L'utilisateur peut éventuellement être amené à renseigner une donnée (ex. CVV pour VisaCheckout) puis il valide le paiement en utilisant le bouton Payer de Payline

Après le paiement, les données acheteur (object buyer) pourront également être récupérées lors du retour du paiement en utilisant le service [getWebPaymentDetails](#).

Clé	Information	Toujours présent ?	pour Amazon Pay
firstName	Le prénom de l'acheteur	OUI	NON
lastName	Le nom de l'acheteur	OUI	
email	L'adresse email de l'acheteur	OUI	
street1	L'adresse de livraison de l'acheteur	OUI	
street2	Le complément de l'adresse de livraison de l'acheteur	OUI	
cityName	La ville de l'adresse de livraison de l'acheteur	OUI	
zipCode	Le code postal de l'adresse de livraison de l'acheteur	OUI	
country	Le pays de l'adresse de livraison de l'acheteur	OUI	

billingStreet1	L'adresse de facturation de l'acheteur	NON	
billingStreet2	Le complément de l'adresse de facturation de l'acheteur	NON	
billingCityName	La ville de l'adresse de facturation de l'acheteur	NON	
billingZipCode	Le code postal de l'adresse de facturation de l'acheteur	NON	
billingCountry	Le pays de l'adresse de facturation de l'acheteur	NON	

DIV et SCRIPT à placer dans dans la page "paiementsshortcut" vers laquelle on a redirigé l'utilisateur précédemment :

HTML

```
<div id="paylineToReplace" class="pl-top">
  <h2 class="BdCn">> Modifier mes adresses</h2>
  <span>Prénom : #firstName#</span>
  <span>Nom : #lastName#</span>
  <span>Mail : #email#</span>
  <div class="pl-left">
    <h3>Adresse de facturation</h3>
    <div>
      <span>Adresse : #billingStreet1#</span>
      <span>Code postal : #billingZipCode#</span>
      <span>Ville : #billingCityName#</span>
      <span>Pays : #billingCountry#</span>
      <a href="#">Editer</a>
    </div>
  </div>
  <div class="pl-right">
    <h3>Adresse de livraison</h3>
    <div>
      <span>Adresse : #street1#</span>
      <span>Code postal : #zipCode#</span>
      <span>Ville : #cityName#</span>
      <span>Pays : #country#</span>
      <a href="#">Editer</a>
    </div>
  </div>
</div>

<script>
  // Fonction qui parse toutes les chaînes contenues entre des caractères #
  function parse(str) {
    return str.match(/#([^\#]*)#/g);
  }

  // Fonction remplaçant toutes les chaînes
  function replace(blockId) {
    // Récupération du tableau de toutes les chaînes à remplacer de la div "main"
    var block = Payline.jQuery('#' + blockId);
    var stringArrayToReplace = parse(block.text());
    // Pour chaque élément du tableau on remplace avec les données fournies par le Widget
    var result = block.html();
    var buyerSC = Payline.Api.getBuyerShortCut();
    for (i = 0; i < stringArrayToReplace.length; i++) {
      var hashedKey = stringArrayToReplace[i];
      var key = hashedKey.substring(1, hashedKey.length-1);
      result = result.replace(hashedKey, buyerSC[key]);
    }
    block.html(result);
    block.show();
  }

  function widgetStateCallback(state) {
    if (state.state === "PAYMENT_TRANSITIONAL_SHORTCUT") {
      replace("paylineToReplace");
    }
  }
</script>
```

Une fois les informations acheteur récupérées, le commerçant peut finaliser la demande de paiement shortcut en appelant la fonction API [Payline.Api.finalizeShortcut\(\)](#) ou déclencher son appel via un clic sur un bouton. Le paiement sera alors finalisé par les serveurs Payline et un résultat sera renvoyé à la page comme dans une intégration Widget classique (affichage du ticket de paiement ou redirection automatique vers la returnUrl fournie dans l'appel au doWebPayment, selon la configuration du point de vente).

5. Après le paiement : Après le paiement, les données acheteur (object buyer) pourront également être récupérées lors du retour du paiement en utilisant le service [getWebPaymentDetails](#).

Feuille de style

Vous pouvez personnaliser la page de la même manière que le [mode widget](#).

Pages associées

- [Mode in-Shop](#)
- [Mode Lightbox](#)
- [Mode Shortcut](#)
- [Paielement Page Web](#)
- [PW - Intégration Widget](#)
- [PW - L'API JavaScript](#)
- [PW - Personnalisation du widget : Feuille de style](#)
- [PW - Personnalisation du widget : Fonction CallBack](#)
- [PW - Widget customization: Style sheet](#)