

# CHAPITRE 23 — Le catalogue produit dynamique en profondeur

Le catalogue produit dynamique transforme votre table `llx_product` en pages web générées automatiquement. Ce chapitre détaille l'ensemble du fonctionnement côté développeur : architecture, classes, configuration et points d'extension.

## Architecture en couches

Couche	Rôle
<b>StudioProductCatalog</b>	Lit les produits Dolibarr (filtres, tris, multilingue) et expose un format normalisé.
<b>StudioSolutionWrapper</b>	Génère et supprime les wrappers Apache <code>solution-&lt;ref&gt;.php</code> en fonction des produits publiés.
<b>Le gabarit solution-detail (page&lt;X&gt;.tpl.php)</b>	Le gabarit unique qui affiche un produit. Une seule page Dolibarr Website pour N produits.
<b>Trigger PRODUCT et CATEGORY</b>	Régénère automatiquement les wrappers à chaque modification de produit ou de catégorie.
<b>Tâche planifiée horaire</b>	Filet de sécurité : relance la génération si un trigger a été manqué.

## Le modèle de données produit

Chaque produit Dolibarr utilisé dans le catalogue exploite plusieurs tables :

Table	Données
<code>llx_product</code>	Champs natifs : ref, label, description, prix, tosell.

<code>llx_product_lang</code>	Traductions natives (label, description par langue).
<code>llx_product_extrafields</code>	Champs personnalisés canoniques : tagline, hero_image, badge, cta_label, cta_url, deployment, compatibility, support, languages, features (JSON), pricing_tiers (JSON), infrasstudio_published.
<code>llx_infrasstudio_product_translation</code>	Surcharges par langue des champs personnalisés traduisibles.
<code>llx_categorie_product</code>	Liens entre produits et catégories (utilisés pour la cartographie d'univers).

## Provisionner les champs personnalisés

Les onze champs personnalisés nécessaires ne sont pas livrés en standard avec Dolibarr. Pour les créer en une commande, utilisez le script de provisionnement générique :

```
php htdocs/custom/infrasstudio/scripts/preset_default.php \
    htdocs/custom/infrasstudio/presets/keaticweb.json
```

Le fichier JSON livre les définitions par défaut (voir Chapitre 28 pour le format). Vous pouvez créer votre propre preset si vous avez besoin d'autres champs personnalisés.

## Utiliser StudioProductCatalog

La classe expose plusieurs méthodes pour interroger les produits :

```
dol_include_once('/infrasstudio/class/studioproductcatalog.class.php');
$catalog = new StudioProductCatalog($db);

// Tous les produits publiés, dans la langue courante
$products = $catalog->fetchPublishedProducts(array(), $iso2);

// Filtres
$products = $catalog->fetchPublishedProducts(array(
    'univers' => 'supply-chain',
    'type'    => 'saas',
), $iso2);
```

```
// Un produit par référence
$product = $catalog->fetchProductBySlug('supplyflow-pro', $iso2);

// Univers utilisés (pour le filtre du catalogue)
$univers = $catalog->fetchUsedUnivers();
```

## Format normalisé d'un produit

```
array(
  'id'          => 5,
  'ref'         => 'supplyflow-pro',
  'label'       => 'SupplyFlow Pro',           // résolu selon la langue
  'description' => '<p>Description...</p>',
  'price'       => 290.00,
  'badge'       => 'Nouveau',
  'hero_image'  => '/medias/...',
  'cta_label'   => 'Demander une démo',
  'cta_url'     => '/contact',
  'tagline'     => 'Optimisez votre supply chain...',
  'deployment' => 'Cloud SaaS',
  'compatibility'=> 'SAP, Oracle, Sage',
  'support'     => '24/7',
  'languages'   => 'fr,en,de',
  'features'    => array('Prévision IA', 'Multi-entrepôts'),
  'pricing_tiers'=> array(...),
  'univers'     => 'supply-chain',
  'type'        => 'saas',
)
```

## Cartographie catégorie vers univers

Les produits sont rattachés à des catégories Dolibarr. Le module associe chaque catégorie à un univers (concept éditorial : Supply Chain, Health, Legal, etc.).

### Cartographie par défaut

```
// Les six univers livrés par défaut
'supply-chain' => array(...catégories supply chain),
'health'       => array(...catégories santé),
'legal'        => array(...catégories juridique),
'digital-humanities' => array(...catégories sciences humaines),
```

```
'transversal'      => array(...catégories outils transversaux),
'fsm'              => array(...catégories field service)
```

## Surcharge par JSON

Pour personnaliser, définissez la constante :

```
// Via dolibarr_set_const ou via la page de configuration admin
$conf->global->INFRASTUDIO_PRODUCT_UNIVERS_MAP = json_encode(array(
    'mon-univers' => array(12, 13, 14), // identifiants de catégories
    'autre'       => array(15, 16),
));
```

# StudioSolutionWrapper

Cette classe gère la génération des wrappers Apache à partir des produits publiés.

## Génération manuelle

```
dol_include_once('/infrasstudio/class/studiosolutionwrapper.class.php');
require_once DOL_DOCUMENT_ROOT . '/website/class/website.class.php';

$website = new Website($db);
$website->fetch(0, 'monsite');

$wrapper = new StudioSolutionWrapper($db);
$stats = $wrapper->rebuildAll($website, '/var/www/monsite');
print_r($stats);
// array('created' => 3, 'updated' => 2, 'deleted' => 1, 'skipped' => 5)
```

## Anatomie d'un wrapper généré

```
<?php
// /var/www/monsite/solution-supplyflow-pro.php
// Generated by StudioSolutionWrapper – do not edit manually.
$solution_ref = 'supplyflow-pro';
$infrasstudio_current_product_ref = 'supplyflow-pro';
global $dolibarr_main_data_root, $conf;
if (empty($dolibarr_main_data_root)) {
    $res = include './page42.tpl.php'; // page solution-detail
} else {
```

```

$res = include $dolibarr_main_data_root
    . ($conf->entity > 1 ? '/' . $conf->entity : '')
    . '/website/monsite/page42.tpl.php';
}
if ($res === false) { http_response_code(500); print 'Failed to make include'; }

```

**Sécurité anti-collision** — Le module ne touche qu'aux wrappers qu'il a lui-même créés (présence du marqueur « StudioSolutionWrapper » dans l'en-tête du fichier). Les pages Dolibarr standards portant un slug commençant par `solution-` sont préservées.

## Configuration du générateur de wrappers

Constante	Rôle
<code>INFASSTUDIO_WEBSITE_KEY</code>	Référence du site cible.
<code>INFASSTUDIO_PUBLIC_DOCROOT</code>	Docroot Apache absolu (par exemple <code>/var/www/monsite</code> ).
<code>INFASSTUDIO_SITE_&lt;id&gt;_WRAPPER_PREFIX</code>	Préfixe des wrappers (par défaut <code>solution-</code> ). Exemples : <code>produit-</code> , <code>service-</code> .
<code>INFASSTUDIO_SITE_&lt;id&gt;_WRAPPER_TEMPLATE_PAGEURL</code>	Slug du gabarit solution-detail (par défaut <code>solution-detail</code> ).

## Le gabarit solution-detail

Il s'agit d'une page Dolibarr Website classique (avec son slug et son tpl.php), mais utilisée par tous les wrappers. Elle reçoit la variable globale `$solution_ref` qui identifie le produit à afficher.

### Squelette type

```

<?php
// page42.tpl.php (slug='solution-detail', type_container='page')
require_once __DIR__ . '/master.inc.php';
require_once DOL_DOCUMENT_ROOT . '/core/lib/website.lib.php';

```

```

require_once DOL_DOCUMENT_ROOT . '/core/website.inc.php';

dol_include_once('/infrastudio/class/studioproductcatalog.class.php');
$catalog = new StudioProductCatalog($db);
$iso2 = infrastudio_current_lang();
$product = $catalog->fetchProductBySlug($solution_ref, $iso2);
if (!$product) {
    http_response_code(404);
    print '<h1>Produit introuvable</h1>';
    exit;
}

ob_start();
try {
    ?>
    <html lang="<?php echo $iso2; ?>">
    <head>
        <title><?php echo dol_escape_htmltag($product['label']); ?></title>
    </head>
    <body>
        <section class="hero">
            <h1><?php echo dol_escape_htmltag($product['label']); ?></h1>
            <p><?php echo dol_escape_htmltag($product['tagline']); ?></p>
            
        </section>
        <section class="features">
            <ul>
                <?php foreach ($product['features'] as $feature): ?>
                    <li><?php echo dol_escape_htmltag($feature); ?></li>
                <?php endforeach; ?>
            </ul>
        </section>
    </body>
</html>
<?php
} catch (Exception $e) { print $e->getMessage(); }
include dol_buildpath('/infrastudio/core/tpl/website_output.tpl.php', 0);

```

**Mélanger slots et données produit** — Vous pouvez combiner les deux : les zones éditoriales (par exemple un titre marketing au-dessus de la grille) en slots, et les zones produit (label, prix,

fonctionnalités) lues via `StudioProductCatalog`. Les deux cohabitent sans conflit.

## Le trigger PRODUCT et CATEGORY

Le module installe un trigger qui écoute :

- `PRODUCT_CREATE`
- `PRODUCT_MODIFY`
- `PRODUCT_DELETE`
- `PRODUCT_PRICE_MODIFY`
- `CATEGORY_LINK` (uniquement si l'objet lié est un produit)
- `CATEGORY_UNLINK`

À chaque événement, le trigger appelle `StudioSolutionWrapper::rebuildAll()` sur le site configuré.

**Note** — Si `INFRASSTUDIO_WEBSITE_KEY` ou `INFRASSTUDIO_PUBLIC_DOCROOT` ne sont pas configurées, le trigger se termine silencieusement. Ni bruit, ni erreur. Cette discrétion convient aux instances Dolibarr qui n'utilisent pas le catalogue dynamique.

## La tâche planifiée horaire (filet de sécurité)

Une tâche planifiée s'exécute toutes les heures et appelle `StudioSolutionWrapper::rebuildAllConfigured()`. Elle lit les constantes et relance la régénération.

Cette tâche apporte les bénéfices suivants :

- Rattrapage si un trigger a été manqué (import en masse, modification SQL directe, etc.).
- Synchronisation après une migration de serveur.
- Garantie de cohérence sans intervention manuelle.

## Reconstruction manuelle en ligne de commande

```
php htdocs/custom/infrasstudio/scripts/rebuild_solution_wrappers.php \  
  <ref-site-ou-id> <docroot-public> [entity]  
  
# Exemple  
php htdocs/custom/infrasstudio/scripts/rebuild_solution_wrappers.php \  
  monsite /var/www/monsite 2
```

## Récapitulatif

## Vous savez désormais :

- Comprendre l'architecture du catalogue (StudioProductCatalog, StudioSolutionWrapper, gabarit, trigger, tâche planifiée).
- Provisionner les champs personnalisés produit avec le preset générique.
- Interroger les produits via StudioProductCatalog.
- Cartographier les catégories Dolibarr vers vos univers via INFRASSTUDIO\_PRODUCT\_UNIVERS\_MAP .
- Comprendre comment StudioSolutionWrapper::rebuildAll() génère les wrappers Apache.
- Configurer le préfixe et le slug du gabarit par site.
- Écrire le gabarit solution-detail qui sert tous les produits.
- Comprendre le rôle du trigger automatique et de la tâche planifiée de sécurité.
- Lancer une reconstruction manuelle en ligne de commande.

**Fin de la Partie IV** — Vous maîtrisez désormais l'intégration côté développeur : préparer un site, annoter avec des slots, comprendre la grammaire, brancher des données Dolibarr, gérer le multilingue, créer vos gabarits et exploiter le catalogue dynamique. Vous êtes en mesure de livrer un site clé en main.

La Partie V aborde l'administration et la maintenance. Si vous êtes administrateur Dolibarr, c'est votre prochaine destination.

---

Revision #2

Created 2026-05-07 11:41:35 UTC by Lucky Ranasolonirina

Updated 2026-05-08 07:49:45 UTC by Lucky Ranasolonirina