PMB – CONSTRUCTION DU PORTAIL – CONCEPTS ET EXEMPLES

Préambule

Introduction

Conseils pratiques (1) : travailler sur plusieurs bases

La version de démo du portail
- Plusieurs bases sur le même serveur
- Installer une base PMB
- Organiser les sauvegardes
- Sauvegarder – restaurer la configuration minimum
- Particularités de la démo

Paramétrage du portail
- Activer le module Portail
- Autoriser l'accès à cet onglet
- Paramétrer les URL de base
- Paramétrage de la démo

Les étapes du processus
- Dans quel ordre logique parcourir les étapes ?

Conseils pratiques (2) : phase d'apprentissage et de pré-tests

Journal de bord : liste des changements à faire ultérieurement

Conseils pratiques (3) : phase de construction du portail

Transfert de la vraie base vers la base de travail
- Notices, exemplaires et étagères
- Configuration globale d'une base

Feuilles de style
- Généralités
- Choisir une feuille de style
- Cascading Style Sheets : priorité entre plusieurs CSS

Conseils pratiques (4) : changer de feuille de style

Quelques exemples d'OPAC
- Un OPAC « classique »
- Un OPAC « style portail »

Contenu éditorial
- Configuration préliminaire – champs personnalisés
  - types de contenu pour les rubriques
  - types de contenu pour les articles
  - statuts de publication
- Numéros identifiants (notices, étagères …)
- Rédiger les rubriques et articles – éditeur HTML
- Gérer l'arbre du contenu éditorial
- Outils pour gérer globalement le contenu éditorial
- Sauvegarder – restaurer le contenu éditorial

Pages de l'OPAC et du portail
- Zones – Significations du terme « page »
- Pages de l'OPAC
- Créer les pages du portail
- Paramètres cms et first_page_params
- URL avec variables et navigation
- Dans quel ordre configurer les pages du portail ?

→

Version 1.0 de ce manuel – juillet 2015

http://amcubat.be/docpmb/

PMB est un SIGB : un Système Intégré de Gestion de Bibliothèque.
C'est un logiciel performant, en constante évolution, développé par la société PMB-Services.
Il est distribué sous licence libre et son code-source est ouvert.
C'est dans la même optique de partage que j'ai décidé de mettre à votre disposition un site de documentation.
Prenez le temps d'explorer ces ressources, il y a de nombreuses fonctionnalités de PMB à découvrir …
Anne-Marie Cubat

Ce document est mis à disposition selon les termes de la licence Creative Commons
Attribution – Pas d'utilisation commerciale – Pas de modification

Toute reproduction, même partielle, doit mentionner mon nom : Anne-Marie Cubat
et le lien vers mon site : http://amcubat.be/docpmb

Toutes les captures d'écran de ce manuel sont publiées avec l'accord de PMB Services et sont libres.
Leur réutilisation doit cependant mentionner l'origine : « PMB Services et les divers contributeurs au logiciel PMB »
ainsi que le site de la forge du projet PMB : http://forge.sigb.net/redmine/projects/pmb

PMB – Construction du portail – Concepts et exemples

Modules, cadres et pages
- Cadres de l'OPAC et du portail
- Identifier et repérer les zones et cadres
- Distinguer les modules et les cadres
- Modules disponibles
- Administration des modules
- Analyser quelques exemples de cadres (*)
- Analyser les pages-outils du portail (**)
- Menu basé sur le contenu éditorial
- Analyser la page d'accueil du portail
- Templates Django – Synthèse
- Propriétés CSS des cadres
- Manipuler les cadres
  - Attacher un cadre à une page
  - Insérer un cadre dans une zone
  - Editer un cadre (contenu – CSS)
  - Déplacer un cadre
  - Enregistrer une « version » du portail
  - Actualiser la page
  - Restaurer – supprimer une « version »

Adapter son mode de travail au portail
- Le résumé (contenu éditorial et bibliographique)
- Le logo des rubriques et articles
- Bulletins et notices de bulletins

Exemples supplémentaires (non illustrés dans la démo)
- Variantes de quelques cadres et pages (***)
  - Page du portail qui affiche une notice
  - Page du portail qui affiche un article
- Liste de sites Web
  - Ouvrir un lien externe dans un autre onglet
- Menu créé dans l'interface d'administration

Transfert du portail vers la vraie base
- Préparatifs à effectuer dans la vraie base
- Eléments à transférer depuis la base de travail
- Autres ajustements nécessaires

Conclusion

Bibliographie

Aide-mémoire et tableaux de synthèse
- Liens entre modules, cadres et pages
- Templates Django
- Balises HTML
- Requêtes SQL utiles : liste et commentaires

Table des matières

Variantes de quelques cadres et pages (***)
- Page du portail qui affiche une notice : ajouter un cadre « Notices du même auteur »
- Page du portail qui affiche un article : un autre usage du cadre « Liste de notices – Etagère associée à un article »
- Page du portail qui affiche un article : un autre mode d'affichage du cadre « Notices dans les mêmes catégories qu'un article »

J'ai analysé plusieurs modules et cadres, mais les explications sont réparties à divers endroits.
Voici quelques indications pour faciliter vos recherches d'un exemple précis

Analyser quelques exemples de cadres (*)	**Analyser les pages-outils du portail (**)**
Cadres créés à partir du module Liste d'articles Page d'accueil du portail • Carrousel d'articles d'actualité • Sélection de nouveautés (étagères) • Liste de brèves Page de l'OPAC qui affiche une notice • Articles dans les mêmes catégories qu'une notice **Cadres créés à partir du module Liste de notices** Page d'accueil du portail • Carrousel de notices – Nouveautés Page qui affiche un article et une liste de notices • Notices d'une étagère associée à un article • Notices dans les mêmes catégories qu'un article Page de l'OPAC qui affiche une notice • Notices du même auteur qu'une notice	**Page qui affiche une rubrique** • Affichage d'une rubrique **Page qui affiche un article et une liste de notices** • Affichage d'un article • Cette page comprend aussi 2 autres cadres déjà analysés dans le chapitre consacré au module Liste de notices **Page qui affiche une notice** • Affichage d'une notice **Page qui affiche une liste d'événements (sélection d'une date)** • Affichage d'une liste d'événements (module agenda) + Agenda : calendrier (page d'accueil) **Page qui affiche une liste d'articles (résultat d'une recherche)** • Résultat de recherche : articles (module liste d'articles) + Barre de recherche (catalogue et articles)

PRÉAMBULE

Partir à la découverte du portail de PMB, c'est toute une aventure !

Imaginez … Essayez de retrouver une âme d'enfance … Les fêtes de fin d'année …

Vous saviez quel cadeau vous alliez recevoir : un jeu de construction, des briques de toutes tailles et de toutes les couleurs.

Un jeu de construction spécial, « haut de gamme ». Vous aviez d'ailleurs lu dans la documentation que vous n'étiez pas limité au contenu de départ, et que vous pouviez concevoir vos propres modèles : c'est cet aspect-là qui vous a attiré.

Vous imaginiez déjà les maisons, châteaux et tours que vous pourriez construire.
Vous rêviez aussi d'ériger un grand bâtiment avec un portail majestueux : la bibliothèque !

Le grand jour arrive, et vous déballez enfin le cadeau de PMB Services.
Un cadeau offert par PMB Services ? Oui, c'est distribué sous licence libre et c'est gratuit !

A première vue, la boîte correspond à vos attentes : vous y voyez de belles photos de bibliothèques, on vous a même donné leurs adresses afin que vous puissiez visiter ces sites tous plus beaux et plus attractifs les uns que les autres.

Vous ouvrez la boîte … La surprise est totale …

Il y a certes quelques briques, mais pas beaucoup, et vous ne voyez même pas exactement comment les agencer.

En-dessous des briques, vous découvrez une magnifique imprimante 3D !! En soi, elle est plus performante, c'est vrai, on en fait ce qu'on veut. Hélas, le seul mode d'emploi disponible actuellement est vraiment succinct : Premiers pas avec le portail

Les premiers essais avec cette imprimante 3D ne sont guère concluants.

Certes, vous avez lu « quelque part » qu'il faut parfois modifier des fichiers « 3D – Templates Django » pour piloter cette imprimante et construire le portail de vos rêves, mais l'écart est grand entre la théorie et la pratique …

Vous continuez à explorer le contenu de la boîte, vous y trouvez une version de démo du portail.
La démo est belle, mais si belle et si complexe que vous ne voyez pas par où commencer : comment fabriquer les briques dont vous avez besoin et comment les assembler ?

Au fil des essais et erreurs, la surprise du début risque de se transformer en déception et en découragement …
Ne perdez pas votre enthousiasme initial !

Dans ce tutoriel, j'essaie de vous initier pas à pas aux concepts fondamentaux liés au portail de PMB.
J'espère parvenir à cet objectif, malgré la complexité de la matière.

Je ne suis pas une « spécialiste », je ne prétends pas maîtriser tous les aspects du portail, loin de là.
Toutefois, j'espère que ce manuel vous aidera à tirer le maximum de « l'imprimante 3D » que vous avez reçue ;-)

Bonne exploration du portail et de ses diverses ressources !

Cordialement,
Anne-Marie Cubat

Ce tutoriel a été rédigé sur base d'une formation, organisée par PMB Services à Bruxelles, les 2 et 3 avril 2014.
Sans cette initiation préalable au portail, ce document n'aurait sûrement jamais vu le jour.

La méthodologie que j'ai employée pour rédiger ce manuel est basée sur mon expérience personnelle.
J'avais certes pu bénéficier d'une bonne formation de base, mais il faut pratiquer, tester, travailler et retravailler …
J'avais dès lors besoin d'un modèle de portail bien construit, pour comprendre comment il fonctionne.
La démo du portail m'a servi de référence concrète tout au long de ce processus d'apprentissage.
Je vous conseille de faire de même : on apprend beaucoup de choses en observant attentivement …

Les explications et les captures d'écran sont basées sur diverses versions de PMB (de 4.0.8 à 4.1.5).

Nous tenons à remercier chaleureusement PMB Services pour l'aide apportée lors de la rédaction de ces manuels.

Certes, nous avions suivi une formation de base, un point de départ indispensable, mais vous imaginez sans peine que tous les thèmes développés dans ces tutoriels n'ont pas pu être traités lors de ce stage qui a duré deux jours. Nous avons donc poursuivi nos recherches, afin « d'apprivoiser » le portail autant que possible. Mais nous n'en maîtrisions pas tous les concepts, et il y avait des aspects dont nous n'étions pas sûres : avions-nous vraiment bien compris la portée de tel ou tel paramètre, ou le rôle exact joué par tel ou tel élément ?

Dans cette optique, nous voudrions remercier particulièrement **Camille Bénard**, responsable du service après-vente chez PMB Services. Après la formation initiale, nous avons en effet pu lui poser des questions par mail, et obtenir des réponses pertinentes, claires et précises. Cela nous a permis à plusieurs reprises de compléter ou de corriger certains passages, et d'améliorer ainsi la qualité de nos tutoriels.

Même si ces manuels ne sont pas parfaits, c'est en tout cas grâce à l'aide précieuse de PMB Services qu'ils contiennent désormais moins d'erreurs ou de lacunes que les versions préliminaires !

C'est vraiment cela, l'esprit du libre : partager les connaissances afin que tout le monde puisse en bénéficier ;-)

Rédiger un manuel consacré au portail de PMB, c'est aussi toute une aventure !

Elaborer un tutoriel sur un sujet aussi complexe que le portail est un défi que nous avons décidé de relever.
Nous nous sommes lancées dans cette aventure un peu folle sans réaliser tout à fait où cela nous mènerait, et surtout sans imaginer vraiment le nombre d'heures que nous finirions par y consacrer !

La rédaction même de ces deux documents a été dès le départ une œuvre individuelle, mais cela ne signifie pas qu'il n'y ait pas eu de concertation entre nous, bien au contraire.

Travailler à deux est un avantage indéniable et aussi un plaisir à partager : on peut s'entraider, s'encourager, se plaindre ou râler ensemble parce que cela n'avance pas aussi vite qu'on le voudrait.

Nous avons eu des séances de travail fructueuses, de longs entretiens téléphoniques, et des fous rires mémorables.
C'était vraiment une chouette expérience humaine ;-)

Nos tutoriels ne sont certes pas parfaits. Malgré tous nos efforts, il est probable qu'ils comportent encore des imprécisions ou des erreurs – et nous espérons qu'elles sont peu nombreuses et mineures !

Toutefois, ces manuels sont sûrement meilleurs que si nous avions travaillé séparément.
En effet, nous avons souvent lu nos versions intermédiaires respectives, et cela nous a permis de clarifier certains aspects, de comparer nos points de vue et d'approfondir l'un ou l'autre concept.

Nous espérons que nos différents modes d'approche vous aideront à mieux comprendre le portail.
Même si la perspective n'est pas la même, même si nos documents sont distincts, nous savons que ces deux manuels vont dans la même direction et qu'ils sont complémentaires.

Françoise Goossens
Anne-Marie Cubat

INTRODUCTION

A partir de PMB 4, vous pouvez transformer l'OPAC « traditionnel » en un véritable portail documentaire.
Les deux termes de l'expression « portail documentaire » ont toute leur importance :

- **C'est un « vrai » CMS : le portail gère l'interface graphique et le contenu**
  - Un CMS (Content Management System, système de gestion du contenu) est un logiciel qui permet de construire des sites en gérant d'une part l'interface graphique, et d'autre part le contenu
  - On peut gérer la mise en page : déterminer la taille des zones, créer des modèles de mise en forme grâce aux feuilles de style (CSS : Cascading Style Sheets), choisir les couleurs …
  - On peut définir les divers modules (carrousel d'articles ou de nouveautés, agenda, étagères virtuelles …) : spécifier les dimensions des cadres, leur position à l'écran, leurs attributs (CSS)
  - On peut rédiger des articles, les regrouper dans des rubriques et faciliter la navigation grâce à un menu
  - Le mode d'affichage est dynamique. Exemple : le carrousel de nouveautés liées à une étagère virtuelle n'a pas toujours le même contenu. C'est aussi le cas pour la rubrique Evénements : bien que le contenu de l'article soit statique (ex. infos sur une animation), seuls les articles encore d'actualité sont visibles.
- **Le contenu éditorial fournit une série d'informations, par exemple**
  - description de la bibliothèque et de son fonds documentaire, aide à la recherche
  - présentation de l'institution dont elle dépend éventuellement (école, université, association, organisme …)
  - infos pratiques (localisation, horaire d'ouverture, conditions d'inscription et de prêt …)
  - agenda, calendrier d'activités régulières ou spéciales, animations diverses …
- **Ce portail est « documentaire », c'est (comme tout OPAC) une porte d'accès au contenu bibliographique**
  - on continue à paramétrer librement les options de l'OPAC dans le module Administration
  - recherche simple ou multicritères dans le catalogue (monographies, périodiques, documents numériques)
  - possibilité d'étendre les recherches vers des notices analogues ou vers des sources extérieures
  - possibilité de filtrer davantage, d'affiner le résultat grâce aux facettes, à la recherche multicritères
  - navigation dans les catégories et/ou balade virtuelle à travers les localisations et sections
- **Le portail crée ainsi une « passerelle » entre le contenu éditorial et le contenu bibliographique**
  - On peut relier un article à certains descripteurs du thésaurus, à une étagère virtuelle …
  - Des rubriques comme les « coups de cœur » ou les « nouveautés » contiennent dès lors des articles qui vont associer un contenu éditorial et un contenu bibliographique, afin de mettre tel ou tel aspect en évidence
  - C'est également valable pour les articles de la rubrique « Evénements ». Vous pouvez non seulement annoncer les animations (régulières ou ponctuelles) que vous organisez, mais aussi établir un lien entre ces activités et le catalogue.
- **Dans ce contexte, le mot « article » a dès lors une signification particulière**
  - Ce n'est pas un article au sens bibliographique du terme (dépouillement d'un périodique disponible en rayon ou en ligne sur Internet).
  - Il s'agit d'un texte que vous rédigez, pour fournir des infos sur un sujet précis (le fonds documentaire, les conditions de prêt, un livre, un auteur, une animation, les nouveautés du rayon DVD …), bref un article que vous publiez dans le « journal virtuel de votre bibliothèque ».
  - Vous devenez donc journaliste, voire même rédacteur en chef !

CONSEILS PRATIQUES (1)

Travailler sur plusieurs bases

L'élaboration du portail est un processus vraiment complexe, qui comprend plusieurs aspects : contenu bibliographique, contenu éditorial, pages, cadres et modules, Il faut configurer les zones, gérer les modules, insérer des cadres dans les pages et les lier aux sources de données, rédiger des articles, les lier au catalogue ou à l'agenda, etc.

Il me semble que cette élaboration du portail doit se faire sur une autre base que la « vraie ».
Je ne vois pas, concrètement, comment construire un portail dans une base en production : une base accessible en ligne, une base où on catalogue régulièrement, et dont on consulte l'OPAC de nombreuses fois par jour.

Pendant qu'on essaie de construire le portail (les débuts sont toujours difficiles), l'OPAC n'est guère convivial.

- Le lecteur va se demander ce qui se passe, car vous allez probablement changer la feuille de style, la disposition, donc il perdra ses repères. Parfois, il ne retrouvera plus l'interface habituelle … qui reviendra comme par miracle une heure plus tard ;-)
- Et vous ne pouvez quand même pas afficher régulièrement un message du genre : « Site en maintenance. Excusez-nous » !
- En outre, afficher un portail peu esthétique ou peu performant (au niveau de la recherche) est contre-productif, et risque d'éloigner des lecteurs qui hésiteront peut-être à revenir.
- Dernier point : si les ressources sont limitées (mémoire, vitesse du serveur …), le travail sur le portail risque de ralentir d'autres activités normales de gestion ou de consultation du catalogue.

En effet, même si vous passez au portail, les pages classiques de l'OPAC (recherche, affichage d'une notice …) ne disparaissent pas. Il n'y a pas que le portail à construire, il faut reconfigurer l'OPAC si on change de feuille de style !

D'ailleurs, vous pouvez adapter dès que possible l'OPAC de la base en ligne à cette nouvelle feuille de style, il n'est pas indispensable d'attendre que le portail soit tout à fait prêt pour changer. A vous de choisir le moment qui vous semble opportun.

Les lecteurs verraient un portail inachevé ou non fonctionnel si vous décidiez de travailler exclusivement en ligne.

- La 1e étape est en effet de choisir une feuille de style récente compatible avec le portail, donc d'en tester d'abord plusieurs …
- Ensuite, modifier éventuellement la feuille de style choisie - par exemple l'adapter à la charte graphique de l'institution
- Essayer de créer des modules et de placer des cadres sur les pages du portail – mais ce n'est pas facile, surtout au début

Les essais et erreurs seraient parfois visibles de tous et perturberaient dès lors la consultation du catalogue.

En théorie, on pourrait croire qu'il suffit de modifier certains paramètres comme first_page_params pour passer de l'OPAC traditionnel au portail ou vice versa. C'est vrai en principe …
Mais dans la réalité, les choses ne sont pas aussi simples, il faut aussi modifier d'autres paramètres.

Quelques exemples parmi d'autres (la liste n'est pas exhaustive) :

- Quand on change de feuille de style, on doit aussi modifier en général les valeurs de paramètres comme biblio_main_header, biblio_important_p1, biblio_important_p2, biblio_preamble_p1, biblio_preamble_p2, etc. Ensuite, pour retourner à l'OPAC traditionnel (basé probablement sur une autre feuille de style), il faudra remettre les valeurs d'origine.
- Vous allez probablement remplacer les étagères virtuelles par des articles liés à une étagère. Quand vous êtes en mode « Portail », il faut décocher la case « visible dans l'OPAC » (dans la définition de l'étagère), mais lorsque vous retournez en mode « OPAC classique », il faudra à nouveau cocher ces cases, sinon les étagères seront invisibles !
- Le même genre de problème va se poser pour les infopages actuelles que vous allez aussi remplacer progressivement par des articles. En mode « OPAC traditionnel », il faut que les infopages soient visibles et qu'il y ait un lien quelque part (ex. dans biblio_preamble_2). Pour le mode « Portail », il faut supprimer les liens et/ou reconfigurer les infopages afin de les masquer.
- Etes-vous sûr que les étagères virtuelles affichées actuellement dans votre OPAC seront les mêmes et auront le même contenu le jour où vous les lierez à l'un ou l'autre article de votre futur portail ? Un beau méli-mélo en perspective ! Il faudra 2 séries d'étagères (et de paniers correspondants) : une pendant la durée des travaux, et une autre série pour le « vernissage » du portail.

En résumé, vous devriez jongler entre deux sauvegardes de la configuration (donc entre autres de la table des paramètres) : une basée sur l'OPAC classique et l'autre sur le portail. Sinon, vous risquez de vous emmêler les pinceaux et de déclencher peut-être des catastrophes en ligne !

Le mieux est donc de travailler sur une autre base (installée en ligne ou en local, peu importe)**, mais pas « la vraie ».**
-Cela vous permet d'apprendre comment agencer les cadres et pages. Ensuite, vous créez « l'architecture de base » de votre portail, une structure que vous pourrez restaurer plus tard dans la base en ligne.

Il faudra à un moment que le paramétrage de votre base de test ressemble le plus possible à celui de vraie base en ligne, mais ce n'est pas vital au début, lorsque vous êtes en phase d'apprentissage, ni lorsque vous commencez à élaborer le portail.

Réaliser une copie complète de la base n'est pas obligatoire, c'est d'ailleurs parfois quasi impossible, vu sa taille. Une version réduite convient même mieux : une base paramétrée comme la vôtre, mais avec un nombre limité de notices.

- **L'inconvénient, c'est que vous ne pouvez probablement pas établir dès le départ tous les bons liens (ex. entre articles et notices), puisque les numéros d'id sont différents** (il faut faire des ajustements plus tard en ligne). **Mais vous pouvez établir une concordance pour les catégories et les étagères,** en ciblant bien les tables à transférer vers la base de test.
- **Le 1er avantage de cette méthode, c'est qu'il est plus facile de gérer un stock limité de notices, sélectionnées soigneusement en fonction des articles que vous comptez publier au début**. A quoi bon avoir plusieurs milliers de notices et d'exemplaires ? En soi, c'est tout aussi inutile que restaurer tous vos lecteurs, leurs prêts et les archives de prêt.
- **Autre point positif : vous ne serez pas confronté à des problèmes de sauvegarde**, **inhérents à une base de grande taille. Cela simplifie vraiment le travail**.

Bien sûr, travailler ainsi, tout construire vous-même n'est pas possible si la base est fort complexe (par exemple, plusieurs localisations qui doivent avoir chacune « leur » part du portail), ou si les exigences de l'institution ou les attentes du public cible sont fort grandes … ou bien si vous êtes tout seul et que vous devez vous occuper de tout !

Une petite bibliothèque gérée par un(e) seul(e) bibliothécaire, aidé(e) éventuellement par quelques bénévoles, peut demander – et obtenir ! – de ses lecteurs un peu de compréhension et de patience pendant un certain temps, c'est tout à fait inconcevable dans d'autres contextes, où il faut que tout soit opérationnel immédiatement.

Dans ce cas, il est sûrement préférable de faire appel à des professionnels comme PMB Services, qui créeront un portail sur mesure, réaliseront sans problème la transition d'un type d'interface vers l'autre, et assureront aussi la formation initiale du personnel. Je rappelle que je me situe ici dans l'optique où vous essayez vous-même de transformer votre OPAC en un véritable portail.

Si vous comptez vous lancer vous-même dans l'aventure, je vous conseille de pouvoir accéder en permanence à 3 bases

- **pmb4portail :** c'est la démo du portail, avec les données de test de PMB (en guise de support d'apprentissage et de modèle)
- **pmb4donnees :** c'est votre base de données en production, telle qu'elle existe actuellement en ligne, sans portail
- **pmb4perso :** c'est une base de travail, dans laquelle vous faites des essais et construisez petit à petit votre futur portail

Le contenu de la base de travail (que j'appelle aussi base de test) va certainement évoluer au fil du temps.

- Vous commencerez forcément par une **période de découverte du portail et d'apprentissage**
  - Dans un premier temps, il faut en effet comprendre une série de concepts, et aussi s'entraîner, manipuler, donc alterner théorie et pratique. Pendant cette phase, il n'est pas nécessaire que votre base de test ressemble fort à votre vraie base.
  - **La version de démo du portail convient parfaitement comme support d'apprentissage et pour les tests du début**.

- Plus tard, à un certain moment « indéterminé » car variable (chacun évolue à son rythme), vous vous sentirez prêt à passer des tests préliminaires à une véritable **phase de construction du portail**.
  - A ce moment-là, il faudra travailler sur **une base qui ressemble davantage à la vôtre**, afin de faciliter le transfert ultérieur de certains éléments du portail depuis la base de test vers la vraie base. Mais le nombre de notices peut être réduit, c'est nettement plus pratique pour la gestion. De toute façon, les tables qui gèrent le CMS sont relativement autonomes par rapport au reste de la base, on peut les restaurer ailleurs.

J'ai réparti la suite des « conseils pratiques » en plusieurs chapitres, en me basant sur l'évolution du processus.

Conseils pratiques (2)	pendant la phase d'apprentissage et de pré-tests
Journal de bord	aide-mémoire : liste des changements à faire ultérieurement dans la base en ligne
Conseils pratiques (3)	pendant la phase de construction du portail
Transfert vers la base de travail	éléments à transférer depuis la base en ligne vers la base de travail
Conseils pratiques (4)	exemple concret de changement de feuille de style
Gestion globale du contenu éditorial	ajout de champs personnalisés – requêtes SQL qui donnent une vue d'ensemble
Transfert vers la vraie base	éléments du portail à transférer depuis la base de travail vers la base en ligne

Chacun a sa méthode et travaille à son rythme. J'imagine sans peine que certains d'entre vous aimeront suivre le tutoriel pas à pas, que d'autres auront un parcours moins linéaire, ou bien plus rapide (s'ils maîtrisent déjà quelques notions à propos du portail), et que d'autres encore voudront d'abord voir quelques exemples, afin d'avoir une approche plus concrète.

C'est pourquoi je voulais expliquer dès le début en quoi consistent ces « conseils pratiques », vous saurez ainsi où trouver les infos dont vous aurez besoin à tel ou tel moment. Sachez en tout cas que ces chapitres forment un ensemble.

Quelle version de PMB installer en guise de support pour ce manuel ?

Notes sur la compatibilité entre les versions de PMB

Pour rédiger ce manuel, j'ai employé des versions de PMB entre la 4.0.13 et la 4.1.5
Toutefois, la plupart de mes explications font référence à la version de démo du portail.
C'est un choix délibéré, je ne voulais pas que ce tutoriel soit seulement théorique.
Je me suis donc basée sur la démo, afin que vous puissiez tester vous-même ce que je montre.

PMB 4.0.13

PMB 4.1

Comme vous pouvez le constater en comparant les menus de construction du portail, de nouveaux modules se sont ajoutés lors du passage de PMB 4.0.13 à PMB 4.1.

Il y a certes de nouvelles fonctionnalités dans les scripts Php, **mais** les tables liées au portail n'ont pas changé (ce sont celles dont le nom commence par « cms »).

D'ailleurs, la démo du portail est la même, qu'on installe PMB 4.0.13 ou PMB 4.1.5, et elle illustre seulement les modules disponibles dans la 4.0.13.

Or dans ce manuel, je vous invite à plusieurs reprises à vous « faciliter l'existence ». En effet, vous pouvez restaurer certaines tables de la démo dans votre base de travail, afin de récupérer une partie de la configuration du portail.

Les versions de PMB entre la 4.0.13 et la 4.1.5 sont compatibles entre elles, en ce qui concerne

- la démo du portail que vous pouvez installer (c'est toujours la même)

Les versions de PMB entre la 4.0.13 et 4.1.5 sont compatibles avec votre base

- si votre version de PMB est >= 4.0.13 et <= 4.1.5
- si vous avez choisi lors de l'installation le même encodage de caractères que pour votre base actuelle (ISO 8859-1 ou UTF-8)

Installez toujours PMB avec le même encodage de caractères que votre vraie base.

La démo de la version 4.0.13 convient aussi bien que la démo 4.1.4 ou la 4.1.5

- **pour découvrir – sur base de la démo - comment le portail fonctionne**
- **pour faire - avec la démo - vos premiers tests sur le portail**
- **pour restaurer dans votre base une partie de la configuration du portail**

La version 4.1 s'impose si vous voulez employer tout de suiteles modules supplémentaires.

Prenez la même version que celle employée actuellement en production pour installer une base de travail, et y restaurer la configuration de votre base en ligne.

De toute façon, PMB inclut un système efficace de compatibilité ascendante, donc vous pourrez installer plus tard une version supérieure et lancer le processus de mise à jour : vous ne risquez pas de perdre les fruits de votre travail !

N.B. Ces remarques sur la compatibilité entre les versions correspondent au moment où j'ai fini de rédiger ce manuel.
Je n'ai pas encore eu le temps d'explorer la version 4.2 qui vient de sortir, c'est pourquoi je n'y fais pas référence lorsque j'aborde le thème de la compatibilité. Il est donc possible que je doive modifier certains commentaires plus tard.

LA VERSION DE DÉMO DU PORTAIL

Pour illustrer le portail et les concepts de base qui y sont liés, il me fallait un support, un exemple accessible à tout le monde, c'est pourquoi j'ai choisi une base que vous pouvez facilement télécharger et installer : la version de démo du portail, que vous pourrez ainsi explorer à l'aise. Sinon, les explications fournies ici risquent de rester fort théoriques … Rien ne vaut la pratique pour comprendre comment tout cela fonctionne ! Installez donc la version dé démo !!

N.B. Pour rédiger ce tutoriel, j'ai employé diverses versions de PMB, comprises entre la 4.0.8 et la 4.1.5, et j'ai travaillé sur plusieurs bases, installées sur le même serveur (dans mon cas, en mode local avec EasyPhp 12.1)

Rappel : PMB 4 requiert du Php 5.3 ou 5.4 – Php 5.4 n'est possible qu'à partir de PMB 4.0.8
Lien pour le téléchargement http://forge.sigb.net/redmine/projects/pmb/files
Installation (Windows) http://amcubat.be/docpmb/pmb-installation-windows-easyphp121

Plusieurs bases sur le même serveur

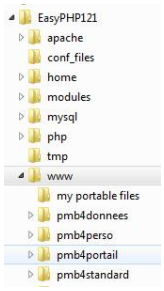
Employer de préférence Firefox - une version récente de Firefox !

Voir Notes sur la compatibilité entre les versions de PMB

La rédaction de ce tutoriel s'est étalée sur plusieurs mois. **Au fil du temps, le répertoire pmb4perso s'est parfois appelé aussi pmbv4perso, et la base bibli4perso a aussi existé sous le nom de bibliv4perso.**

Idem pour les autres répertoires et bases, il y a parfois un « v » en plus ou en moins.

L'existence de 2 noms explique des divergences mineures entre certaines captures d'écran



Commencez par créer plusieurs répertoires dans le dossier www, et veillez à leur donner des noms significatifs. Copiez-collez dans chaque répertoire la version décompressée du zip de PMB.
Rappel : ne faites pas de copie à partir d'un répertoire où PMB aurait déjà été installé, car lors de l'installation, certains fichiers sont renommés, tandis que d'autres sont créés et paramétrés. Il faut démarrer avec un PMB « vierge ».

Voici la liste des bases que j'ai employées dans le cadre de ce tutoriel :

- **pmb4donnees (ou pmbv4donnees)** : **cela symbolise votre « vraie » base, avec vos données**
  - Il n'est pas indispensable de réinstaller ailleurs votre base réelle, vous y avez probablement accès par Internet. L'élaboration du portail peut se faire en bonne partie sur une autre base de taille plus réduite : en effet, on gère plus facilement un stock limité de notices, et les sauvegardes ne posent pas de problème particulier. La phase finale se fera bien sûr sur la vraie base.
  - Pour ce tutoriel, j'ai mis dans pmb4donnees les mêmes données que dans la base de démo du portail, mais j'ai supprimé tous les éléments liés au portail, c'est donc comme si on démarrait à zéro. Il y a déjà un catalogue, mais pas de portail. La feuille de style est genbib.

- **pmb4portail (ou pmbv4portail)**: **la démo du portail, que j'essaie de garder « intacte » ou à peu près**.
  - Elle me sert de modèle, d'exemple, et est paramétrée pour la feuille de style pmb4
  - Les seules modifications que j'y ai apportées sont décrites dans ce chapitre-ci.

- **pmb4perso (ou pmbv4perso)** : la **base de travail, la base de test** (essais personnels du portail)
  - Vous pouvez installer là aussi dans un premier temps la version de démo du portail et faire quelques tests, ainsi la base pmb4portail reste dans son état d'origine. C'est plus pratique pour comparer et comprendre car vous gardez un modèle intact (pmb4perso), mais ce n'est pas indispensable.
  - Lorsque vous aurez mieux compris le fonctionnement du portail, vous devrez de toute façon y installer un « pmb4standard », y restaurer la configuration globale de votre base en production, ainsi que quelques données (en nombre limité !), avant d'entamer la construction de votre portail.
  - Avant d'activer le portail, je vous suggère de sauvegarder l'ensemble des tables liées au portail. Cela vous donnera une copie des tables « portail » quasi vides, avec le strict minimum vital. C'est utile si vos premiers essais ne sont guère concluants : dans ce cas, restaurez ce genre de sauvegarde, et vous pouvez repartir à zéro, avec un portail « vierge ». Voir organiser les sauvegardes

- **pmb4standard (ou pmbv4standard) - avec les données essentielles pour démarrer rapidement**
  - C'est une base sans notices mais déjà bien paramétrée, qui convient par exemple pour faire quelques tests rapides de différentes feuilles de style, avant d'en choisir une que vous pourriez employer.
  - Si vous avez oublié de réaliser une sauvegarde des tables portail « quasi vides » avant d'activer le portail (voir ci-dessus), ou si vous ne la retrouvez plus, vous pouvez installer quelque part un PMB standard, et créer cette sauvegarde, ce « kit de dépannage ».

Bien sûr, vous n'êtes pas obligé d'installer toutes ces bases, mais je trouve utile, surtout au début, de pouvoir ouvrir plusieurs bases dans le navigateur, et de comparer ainsi plus facilement « un modèle à imiter » (pmb4portail) et vos essais personnels. Il suffit en effet de passer d'un onglet à l'autre pour changer de base, cela facilite vraiment la compréhension des concepts fondamentaux.

Astuce au moment de l'installation – si on a plusieurs bases sur le même serveur

SI on a plusieurs bases sur le même serveur, chacune doit avoir un nom différent. Vous allez effectuer des sauvegardes et des restaurations, il faut donc donner des paramètres pour restaurer des tables : nom de la base, nom de l'utilisateur et mot de passe.

Pour la facilité, j'emploie un système mnémotechnique tout simple lorsque je travaille sur mon ordinateur, ainsi je me souviens immédiatement des noms à introduire pour une restauration avec le processus emergency.

En situation réelle de production, le système est plus complexe, et j'évite bien sûr d'employer trois fois le même nom, pour des raisons de sécurité.

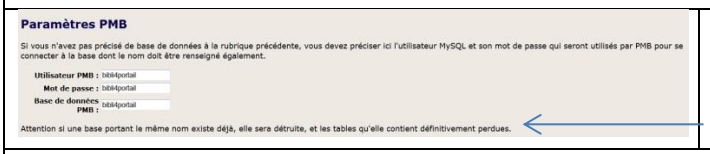
Nom du répertoire d'installation	Nom de la base, de l'utilisateur et mot de passe
pmb4portail (ou pmbv4portail)	bibli4portail (ou bibliv4portail)
pmb4donnees (ou pmbv4donnees)	bibli4donnees (ou bibliv4donnees)
pmb4perso (ou pmbv4perso)	bibli4perso (ou bibliv4perso)
pmb4standard (ou pmb4standard)	bibli4standard (ou bibliv4standard)

Installer une base PMB

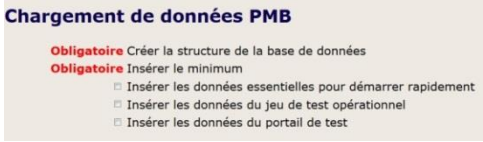
Lorsqu'on lance le processus d'installation, on a le choix entre diverses options.

Puisque vous vous intéressez au portail, je suppose que votre base est déjà accessible en ligne. Mais même si ce n'est pas (encore) le cas, même si la consultation ne se fait pour le moment que via l'Intranet, choisissez des options d'installation compatibles.

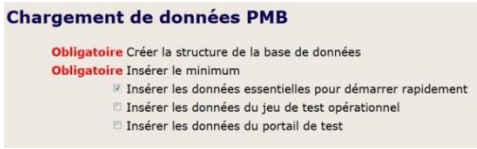
Voir Notes sur la compatibilité entre les versions de PMB



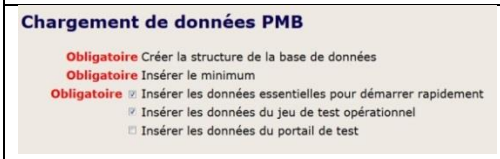
Pour des raisons évidentes de sécurité, il vaut mieux éviter de garder le nom par défaut (bibli), car ce nom est connu de tous. En outre, si on installe plusieurs bases PMB sur le même serveur, il faut leur donner des noms différents, sinon on perd des données.



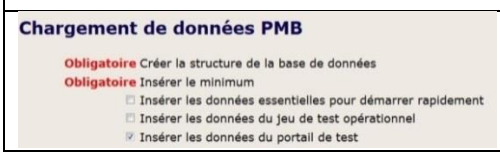
Si vous ne cochez aucune option, vous installez une base quasi vide – c'est plus compliqué à gérer, surtout pour une 1e approche du logiciel, car **rien (ou presque rien) n'est paramétré, donc vous risquez d'oublier de configurer quelque chose d'important, et de créer ainsi des dysfonctionnements !**
Le seul cas où cela pourrait servir, c'est si vous allez ensuite restaurer une base déjà bien paramétrée. **C'est pourquoi je déconseille en général cette option.**



Une base avec les données essentielles pour démarrer rapidement – c'est la meilleure manière d'installer PMB, surtout au début, et même plus tard, **si vous voulez démarrer une base en partant à peu près de zéro**. Au moins, vous êtes sûr que tout ce qui doit être paramétré l'est déjà, et rien ne vous empêche de modifier par la suite les options choisies par défaut. En outre, vous pouvez installer éventuellement un thésaurus et une indexation décimale, plusieurs possibilités vous sont offertes.
A partir de PMB 4, vous disposez aussi de nombreuses requêtes SQL installées dans les 3 onglets concernés (Circulation, Catalogue et Administration). Explorez cet aspect, vous y trouverez sûrement des procédures bien utiles.
N.B. Si vous voulez jeter un coup d'œil du coup du côté des nouveautés (comme les procédures SQL), ne touchez pas à cette base, évitez donc d'y restaurer trop vite une ancienne base déjà configurée, car vous feriez une « marche arrière dans le temps ».



Cliquer sur la 2e option (données du jeu de test) sélectionne automatiquement la 1e (données essentielles). Vous avez donc quelques exemples, mais ils sont moins nombreux, moins diversifiés, ce ne sont d'ailleurs pas les mêmes que dans l'option n° 3. Le thésaurus et l'indexation décimale vous sont imposés, pas de choix possible.



La 3e option installe le portail et des données de test. Tout est préconfiguré, le portail est directement opérationnel. Le thésaurus et l'indexation décimale vous sont imposés. **Même si le portail ne fait pas partie de vos priorités pour le moment, cela vaut la peine d'installer cette démo, car elle met davantage en valeur les nouvelles fonctionnalités de PMB 4.**

Organiser les sauvegardes

Bien entendu, même si vos premiers essais ne sont guère concluants, même si vous avez bouleversé l'agencement des cadres et que l'aspect général est moins séduisant qu'au départ, ce n'est pas grave s'il s'agit de la démo. Dans le pire des cas, il suffirait de réinstaller la base.

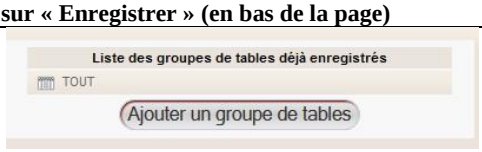
Autant éviter quand même ce genre de désagrément, surtout dès que vous travaillez « en vrai ».

Voici donc quelques précautions à prendre afin de sauvegarder vos plus belles réussites (soyons optimistes) ou de récupérer la version précédente, voire même la version d'origine en cas de catastrophe graphique !

1. Mettre à jour le groupe de tables appelé « tout »

Lorsqu'on installe une nouvelle base, ou lorsqu'on met PMB à jour, le groupe de tables appelé « tout » ne mérite pas vraiment son nom. En effet, toutes les tables ne sont pas cochées, donc si vous lancez cette sauvegarde sans vérifier, vous risquez de vous faire piéger. Vous pourriez croire que tout est sauvegardé, alors que ce n'est pas le cas.

Administration > Sauvegarde > Groupes de tables

Exemple de message d'erreur	Solution – Sélectionner le groupe appelé « tout »
	Cliquer sur « Tout cocher » et ensuite sur « Enregistrer » (en bas de la page)
	Le message d'erreur a disparu, vos sauvegardes sont désormais complètes. Sauvegardez la base dans sa totalité (par défaut, son nom commence par le préfixe full). 

Pensez aussi à faire cette vérification dans vos autres bases !

2. Créer des jeux de sauvegarde supplémentaires

Il vaut mieux réaliser des sauvegardes fréquentes, afin de ne pas perdre les fruits de son travail.

Mais une sauvegarde complète peut générer des fichiers très lourds si la base est volumineuse, donc on hésite parfois. C'est d'ailleurs une des raisons pour lesquelles je vous avais suggéré de travailler avec un nombre limité de notices.

Il est pourtant simple de ne sauver qu'une partie de la base, par exemple celle qui concerne le portail.

Voici les nouveaux groupes de tables et les jeux de sauvegarde que je vous conseille de créer au début.

	Informations sur la sauvegarde				Informations du fichier	
	Nom du fichier	Date	Etat final	Utilisateur	Jeu	Heure
	full_2015_03_02.sav	2015-03-02	La sauvegarde a réussi	Super User	Tout	10:34
	cms_2015_03_02.sav	2015-03-02	La sauvegarde a réussi	Super User	CMS	10:34
	cms_min_2015_03_02.sav	2015-03-02	La sauvegarde a réussi	Super User	CMS - Minimum	10:34
	sql_2015_03_02.sav	2015-03-02	La sauvegarde a réussi	Super User	Requêtes SQL	10:34
	params_2015_03_02.sav	2015-03-02	La sauvegarde a réussi	Super User	Paramètres	10:34

Nom du fichier	Jeu
full_2015_03_02.sav	Tout
cms_2015_03_02.sav	CMS
cms_min_2015_03_02.sav	CMS - Minimum
sql_2015_03_02.sav	Requêtes SQL
params_2015_03_02.sav	Paramètres

Je parlerai plus tard d'un autre jeu de sauvegarde, plus complet que cms_min : toutes les tables liées au contenu éditorial (cms_edit).

Ce jeu cms_edit permet d'enregistrer et de restaurer si nécessaire les rubriques, les articles, leurs liens avec votre base (notices, étagère, descripteurs …) et l'indexation de leur contenu.
Il peut aussi inclure les tables liées au module Portfolio.

Cela peut s'avérer utile si vous voulez repartir à zéro au niveau graphique (feuille de style, agencement des zones et des cadres), mais conserver le contenu éditorial (arborescence des rubriques et articles déjà rédigés).

Voir sauvegarder le contenu éditorial

- **Le jeu de sauvegarde appelé « Tout »** (préfixe « full ») **enregistre l'ensemble de la base, du moins si vous avez veillé à mettre à jour le groupe de tables qui y correspond !!** (voir ci-dessus, point n° 1).

- **Pour enregistrer l'ensemble du portail (articles, rubriques, cadres, modules, propriétés CSS, mise en page …), il suffit de repérer toutes les tables dont le nom commence par cms. On ne peut pas rêver plus simple !**
- **J'ai donc créé un jeu de sauvegarde « CMS »** (préfixe identique)

- **J'ai aussi créé un jeu de sauvegarde « CMS – Minimum »** (préfixe cms_min).
- Il est pratique de regrouper 5 tables dans un jeu de sauvegarde spécifique.
  - cms_editorial_custom, cms_editorial_custom_lists, cms_editorial_publications_status, cms_editorial_types
  - cms_managed_modules
- Vous pourrez ainsi vous épargner du travail, en restaurant dans votre base la configuration éditoriale minimum de la démo, et en l'étoffant ensuite. En outre, cela permet d'employer directement les templates Django inclus dans la démo et les requêtes SQL que j'ai fournies. Voir infos détaillées dans le chapitre suivant.

- **Le portail se configure dans les tables dont le nom commende par cms, mais aussi dans Administration > Outils > Paramètres de l'OPAC / du Portail**
- Pour éviter de devoir sauvegarder l'ensemble de la base à intervalles réguliers, il est parfois nécessaire de **sauvegarder la table des paramètres qui correspond à l'état du portail à tel ou tel moment précis.**
- Un jeu de sauvegarde spécifique, qui contient uniquement cette table, peut donc s'avérer utile pendant la phase d'élaboration du portail (préfixe params).

- **J'ai également prévu une sauvegarde distincte des requêtes SQL des 3 onglets** (Catalogue, Circulation, Administration), **et du classement des procédures** (dans l'onglet Administration). 4 tables sont concernées :
  - caddie_procs, empr_caddie_procs, procs et procs_classements
- *Vous me connaissez, je suppose donc que cela ne vous surprendra pas ;-)*
- Il va de soi que ce dernier jeu de sauvegarde n'est pas indispensable ! (préfixe sql)

3. Sauvegarder seulement la mise en page du portail et les propriétés CSS

Si je mentionne ce sujet ici dans l'introduction, c'est pour attirer votre attention sur le fait qu'il s'agit d'une sauvegarde partielle ! Cliquer sur ce bouton « Enregistrer » n'enregistre qu'un nombre réduit d'éléments.

Enregistrer une « version » du portail ne signifie pas « enregistrer tout le portail » ni « enregistrer tout l'aspect graphique » !

Lorsqu'on construit le portail (onglet Portail, Construire), et qu'on a déplacé ou modifié des éléments, il faut enregistrer les nouvelles propriétés CSS des zones et des cadres afin de préserver les modifications – du moins si on est content de cette mise en page !

Cliquer sur « enregistrer » (dans le volet de droite, en bas).
Cela crée une « version » : une copie de la mise en page.

Ce système de « versions » sauvegarde seulement la mise en page, la disposition des zones et des cadres et leurs propriétés CSS,
- **pas les cadres en tant que tels** (le paramétrage du contenu est enregistré quand on sauvegarde le cadre)
- **ni le contenu éditorial** géré dans l'onglet Administration ou dans l'onglet Portail

Si vous avez entretemps supprimé un cadre, restaurer une « version » antérieure ne va pas le ressusciter !
Ce système de « version » permet uniquement de repositionner les cadres qui existent encore à l'endroit qu'ils occupaient auparavant, en rendant aux zones et aux cadres leurs propriétés CSS antérieures.

Des infos détaillées sur ce sujet seront fournies au moment opportun : Manipuler les cadres

Sauvegarder – restaurer la configuration minimum

La configuration minimum de la base définit les types de contenu des rubriques ou articles et les statuts de publication. Elle contient aussi les templates Django inclus dans la démo.

En théorie, vous pouvez partir de zéro, et tout créer (statuts et types de contenu). Toutefois, si vous avez une version compatible de la démo, vous pouvez facilement récupérer sa configuration éditoriale de base, et l'étoffer ou la modifier ensuite, afin de l'adapter à vos besoins spécifiques. En outre, restaurer ce paramétrage vous permet d'employer directement les templates Django **et les** requêtes SQL **que j'ai mises au point, vous ne devrez pas les adapter. Je vous conseille de le faire, c'est un gain de temps appréciable.**

Voir Notes sur la compatibilité entre les versions de PMB

Créer cette sauvegarde

Configuration minimum du contenu éditorial et templates Django : liste des 5 tables du jeu cms_min		
Nom des tables	**Commentaire**	
Cms_editorial_types	Types de contenu des rubriques et articles	**Attention ! Ces 3 tables forment un tout indissociable. Si on restaure une de ces tables, il faut aussi restaurer les 2 autres !**
Cms_editorial_custom	Champs personnalisés liés aux définitions des types de contenu des rubriques et articles	
Cms_editorial_custom_lists		
Cms_editorial_publication_states	Statuts de publication	
Cms_managed_modules	Templates Django inclus dans la démo	
Ne pas sélectionner cms_editorial_custom_values ! Ce sont des valeurs liées à une base précise, elles ne sont donc pas transférables.		

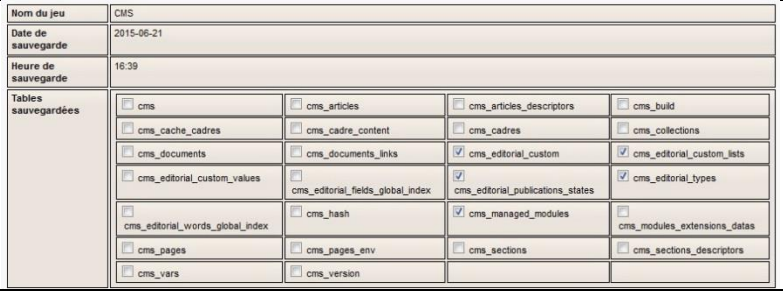
A quoi sert cette sauvegarde partielle, si on a déjà une sauvegarde complète de toutes les tables cms ?

Il ne semble de prime abord pas indispensable de créer un jeu de sauvegarde supplémentaire.

En théorie, vous avez déjà ce qu'il vous faut avec le jeu de sauvegarde cms.

En effet, au début de la phase de restauration, on vous propose de restaurer toutes les tables (elles sont toutes cochées).

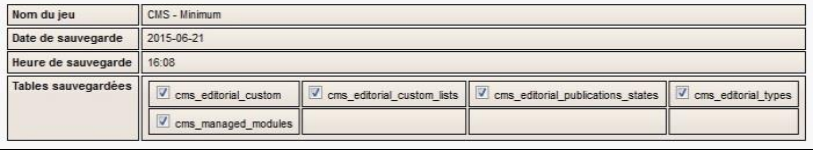
Il suffit donc de décocher les autres tables, afin de ne garder que les 5 que vous pouvez récupérer sans souci.



Le jeu de sauvegarde « cms » comprend 23 tables pour PMB 4.0.13 et 26 tables pour PMB 4.1

Si je vous conseille malgré tout de créer un jeu de sauvegarde minimum avec uniquement ces 5 tables, c'est qu'il y a une raison.
La sauvegarde partielle (cms_min) est plus « sûre » que la sauvegarde complète !

En effet, cela permet d'éviter les erreurs dues à la distraction ou à la fatigue, puisqu'il ne faut plus vérifier quelles tables sont cochées ou non. Sinon, on risque par malheur d'en oublier une, ou d'en restaurer une qui n'aurait pas dû l'être !



Restaurer cette sauvegarde

Vous pouvez restaurer dans votre base la sauvegarde partielle (créée à partir de la démo), grâce au processus emergency.

Rappel : pour lancer le processus emergency, saisir dans le navigateur une adresse de ce type-ci
http://127.0.0.1/pmb4perso/admin/sauvegarde/emergency/emergency.php

Particularités de la démo

1. Tout article (même lié à une actualité événementielle) est éternellement visible dans la démo !

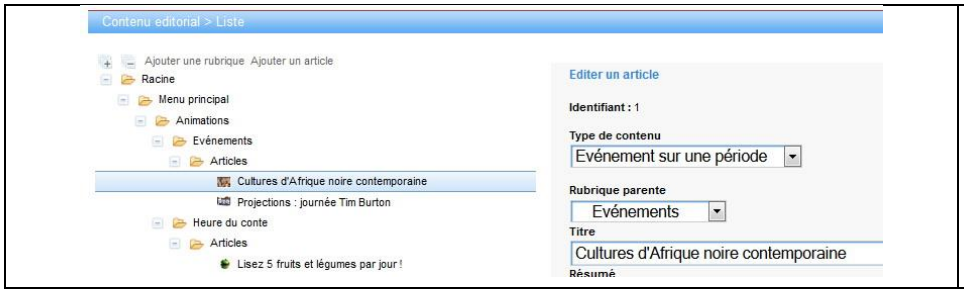
Les dates de publication ne sont pas précisées dans la démo.

Lorsqu'on présente par exemple une animation, deux types de paramètres interviennent au niveau des dates :

- **La date de l'événement lui-même**
  - Soit un jour précis
  - Soit pendant une période (de telle à telle date)
- **Les dates de publication de l'article**
  - La durée pendant laquelle l'article doit être visible

Ces deux types de dates sont distincts. Exemple : on peut parler d'une part d'un événement qui a lieu le 10 octobre, ou bien du 10 au 15 octobre, et décider d'autre part que l'article qui annonce cette animation sera visible dès le 15 septembre, et qu'il disparaitra automatiquement dès le 16 octobre.
Ce système évite de garder en page d'accueil ou ailleurs des infos qui sont visiblement obsolètes ! Parler au futur d'une activité qui s'est déroulée il y a une quinzaine de jours, cela ne fait pas sérieux …

Toutefois, dans le cas de la démo, les dates de publication n'ont pas été spécifiées, afin d'éviter la disparition intempestive de ces articles d'actualité. *En soi, c'est tout à fait logique pour une démo censée résister à l'épreuve du temps ;-)*



Dates de publication

Dates de l'événement

Ce genre de configuration est aussi possible dans un portail « normal », où on ne précise pas toujours les dates de publication, mais c'est pour une autre raison que dans la démo.

> **En effet, il est tout à fait logique de ne pas mentionner les dates de publication s'il s'agit d'un article qui doit être toujours visible car son contenu est permanent**, par exemple la présentation de la bibliothèque ou de l'institution, les infos pratiques, l'aide à la recherche, etc.

Toutes les rubriques et tous les articles de la démo ont le statut « publié », ils sont donc visibles dans l'OPAC.

La visibilité d'un article n'est pas seulement une question de dates de publication, elle dépend aussi de son statut de publication (brouillon, publié et public, publié pour les abonnés …) **et du statut de publication de sa rubrique parente.**
Dans la démo, 2 statuts de publication existent : brouillon et publié. Mais tout a été « publié » ! (tout = rubriques + articles)

Tous les articles de la démo sont accessibles en page d'accueil, soit via le menu, soit via la « liste de brèves ».

Les 2 critères mentionnés jusqu'à présent (dates de publication et statut de publication) ne suffisent pas pour rendre un article visible. **Il faut aussi prévoir un <u>lien vers</u> cet article, « quelque part » dans la configuration du portail.**
Le système le plus simple à mettre en œuvre :

> **en affichant les rubriques parentes dans le menu, on a d'office un lien vers tous les articles qui en font partie.**

Si le paramétrage diffère de celui de la démo, autrement dit si un article n'est pas lié au menu (via sa rubrique parente visible) ou ne fait pas partie d'une liste d'articles affichés dans une page (comme les brèves en page d'accueil), il est quasi introuvable. Si la recherche dans le contenu éditorial a été configurée, elle permet en théorie de trouver l'article, mais vu qu'on ignore a priori qu'il existe, et donc quel mot saisir comme critère de recherche, il faut vraiment espérer un sérieux coup de pouce du hasard !

2. Le carrousel de notices n'est pas visible immédiatement

Lorsqu'on regarde dans la colonne de gauche et qu'on voit la liste des cadres (en page d'accueil), on y trouve 2 carrousels :

- **Un carrousel d'articles d'actualité** (présentés en mode « fondu enchaîné »)
  - Exemple d'article : Des bibliothèques pleines de fantômes
- **Un carrousel de notices** (les couvertures de quelques livres, présentées en défilement vertical)
  - Certaines acquisitions récentes de la bibliothèque

Mais quand on regarde la page d'accueil dans le volet de droite, on ne voit qu'un seul carrousel : celui des articles !

Question existentielle : où se cache donc le carrousel de notices ?

N.B. Ce carrousel n'a pas été créé avec le module Carrousel, mais avec le module Liste de notices.

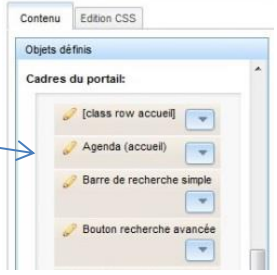
Je vais dès le début vous montrer comment rendre ce carrousel de notices visible !
Il serait quand même dommage de s'en priver … en outre, c'est une occasion d'apprendre comment éditer un cadre ;-)

Deux onglets :

- « **Contenu** » → voir la liste des cadres (de l'OPAC, du portail, hors de la page)
- « **Edition CSS** » → éditer les propriétés CSS d'un cadre préalablement sélectionné

Pour éditer le contenu d'un cadre, cliquez sur l'icône du crayon (à gauche du nom donné au cadre)
Définir les propriétés générales du cadre (nom, page, source de données, template Django ...)

Pour éditer les propriétés CSS d'un cadre, cliquez sur son nom, et ensuite, sur l'onglet CSS.
Définir les attributs CSS du cadre sélectionné (dimensions, positionnement, marges …)
Si vous voulez ensuite retourner à la liste des cadres, cliquez sur l'onglet « **Contenu** »

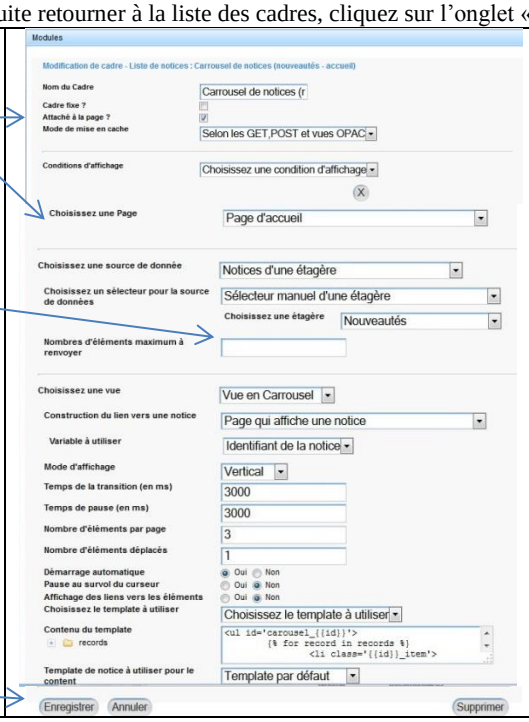


Ce cadre n'est pas « fixe » (fixe = visible partout), il est attaché à une page spécifique, la page d'accueil du portail.

Voici la cause du « mini-problème » : ce champ est vide dans la démo, on n'a pas défini le nombre maximum d'éléments à renvoyer.

N.B. Dans la démo, l'étagère « Nouveautés » est liée à un seul panier, qui comprend 13 notices.
La valeur maximum possible est donc 13.

N'oubliez pas de sauvegarder le contenu (le bouton « enregistrer » est tout en bas)



Eureka !
Dès que vous aurez saisi une valeur logique dans le champ « nombre maximum d'éléments à renvoyer », enregistré le nouveau contenu du module et actualisé la page, vous verrez enfin le carrousel défiler à droite de l'écran !

PARAMÉTRAGE DU PORTAIL

Employer de préférence Firefox - une version récente de Firefox !
Avant d'employer une base prévue pour le portail, il faut absolument passer par les 3 étapes décrites ici : activer le module portail, autoriser l'accès à cet onglet et paramétrer les URL de base. **Dans le cas de la démo, certains de ces paramètres sont déjà corrects, mais pas tous, vérifiez et complétez.** **Dans une base standard, vous devez configurer tous ces paramètres pour créer un portail.**

Activer le module Portail

Active	**Le module Portail n'est pas activé par défaut dans un PMB standard, il faut donc donner à ce paramètre la valeur 1.**
cache_ttl	**Lorsque vous construisez le portail, mettez cache_ttl à 0 pour voir plus vite l'impact des modifications.** **N'oubliez pas de remettre la valeur par défaut (1800) lorsque vous repassez en mode de fonctionnement « normal »** Cette valeur correspond à la durée de vie du cache des cadres du portail (en secondes) – Voir infos sur le cache.
url_base_cms_build	**Valeur à déterminer en fonction de votre installation – voir ci-dessous, configurer les URL de de base**

Configuration dans un PMB standard

Portail

Sous-type	Valeur	
active	0	Module 'Portail' activé. 0 : Non. 1 : Oui.
cache_ttl	1800	durée de vie du cache des cadres du portail (en secondes)
url_base_cms_build		url de construction du CMS de l'OPAC

Configuration adaptée au portail

Sous-type	Valeur	Explication
active	1	Module 'Portail' activé. 0 : Non. 1 : Oui.
cache_ttl	0	durée de vie du cache des cadres du portail (en secondes) Valeur par défaut : 1800
url_base_cms_build	http://127.0.0.1/pmb4perso/opac_css/	url de construction du CMS de l'OPAC

Une règle d'or, valable en toute circonstance pour les paramètres
N'effacez pas trop vite la valeur d'un paramètre, vous devrez peut-être faire marche arrière un jour ou l'autre …

Si le choix se limite à 2 valeurs (par exemple, 0 ou 1), pas de souci ! Si ce n'était pas l'un, c'est l'autre !!
Mais si le choix est plus complexe ? Si vous n'avez aucune chance de deviner la valeur de départ (ex. cache_ttl = 1800) ?

Si vous craignez d'oublier la configuration de départ, faites d'abord un copier-coller de la valeur initiale dans le champ commentaire du paramètre. Ainsi, vous pourrez toujours la retrouver ;-)

Autoriser l'accès à cet onglet

Il faut aussi accorder à un ou à plusieurs utilisateurs, le droit d'accéder à l'onglet Portail.

Administration > Utilisateurs > Modifier les droits

Cocher la case « Portail », enregistrez, le cadenas sera déverrouillé, et l'onglet Portail apparaîtra.

Droits
- Circulation
- Circulation restreinte
- Administration
- Catalogage
- Éditions
- Editions forçage
- Sauvegarde
- Portail
- Autorités
- D.S.I.
- Préférences
- Catégories

Paramétrer les URL de base

Explication préliminaire pour comprendre comment déterminer l'adresse de construction du portail : url_base_cms_build	
Attention ! Le paramètre appelé « url_base » existe en double exemplaire - Ne les confondez pas !	
url_base En gestion (paramètres généraux)	Sa valeur par défaut est http://SERVER/DIRECTORY/ Dans une de mes bases, la valeur est devenue http://127.0.0.1/pmb4perso/
url_base Dans les paramètres de l'OPAC	Sa valeur par défaut est ./ Dans une de mes bases, la valeur est donc devenue http://127.0.0.1/pmb4perso/opac_css/

Comment construire le paramètre url_base_cms_build ? Par concaténation de deux éléments
Vous prenez url_base (de gestion) et vous ajoutez à la fin opac_css/ Ne pas oublier le / final Le résultat de cette concaténation sera donc (dans mon cas) http://127.0.0.1/pmb4perso/opac_css/
Conclusion : la plupart du temps, url_base_cms_build a la même valeur que url_base de l'OPAC. Toutefois, pour certains hébergements, il arrive que l'adresse publique de l'OPAC soit sur un nom de domaine différent que l'adresse de gestion de la base. Dans ce cas, il est nécessaire de créer une « véritable » adresse de construction pour le portail.
Vous devez configurer les 3 URL afin que tout fonctionne correctement : url_base (gestion), url_base (OPAC) et url_base_cms_build

Paramétrage de la démo

Les paramètres cités ci-dessous ont déjà été configurés dans la démo, vous ne devez donc rien y changer.
Au moment de créer un portail dans un PMB standard, pas de précipitation, ne copiez pas ce modèle tout de suite !
Ou bien considérez qu'il s'agit d'une série de tests nécessaires, et soyez prêt à recommencer s'il le faut.

Si vous modifiez certaines valeurs trop vite, si vous changez de feuille de style trop tard, vous risquez de devoir redémarrer à zéro le paramétrage CSS des zones et des cadres – simplement par méconnaissance du sujet. En effet, il y a un certain nombre de notions à maîtriser d'abord : feuilles de style, zones, cadres, pages du portail, configuration du contenu éditorial, etc.
Je donne ici une première approche de ces paramètres, mais j'en reparlerai plus tard en détail, au moment adéquat.

Paramètre default_style

C'est un des paramètres fondamentaux de l'OPAC. Dans la base de démo, il contient la valeur pmb4.

Le choix de la feuille de style pour votre futur portail est évidemment primordial, il détermine l'aspect général et l'harmonie de l'ensemble, la taille des zones, la disposition et les dimensions des cadres, la charte graphique, etc.
Ce choix influence aussi la manière de configurer d'autres paramètres (comme biblio_important_p1, biblio_important_p2, etc).
N'oubliez pas que les pages classiques de l'OPAC (ex. catalogue) continuent à exister, même si vous avez construit un portail.

Infos complémentaires : voir Feuilles de style 3e série de conseils pratiques

Paramètre cms

Valeur par défaut (base « standard »)	cms 0	id du CMS utilisé en OPAC
Valeur (démo du portail)	cms 1	id du CMS utilisé en OPAC

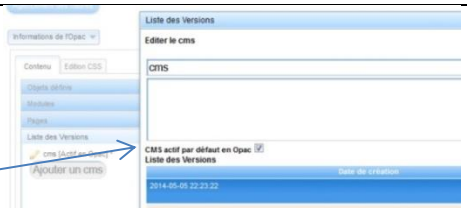
Au départ, dans une base standard, il n'y a pas de CMS. Le simple fait d'activer ce paramètre crée (la première fois qu'on le fait) automatiquement un CMS dont l'id (l'identifiant) est 1, c'est pourquoi il faut donner à ce paramètre la valeur 1.
Pendant que vous construisez le portail, ou à partir du moment où il est finalisé et utilisé en ligne, il faut que le CMS soit actif en permanence, donc il faut que ce paramètre ait la valeur 1.

Modifier la valeur de ce paramètre ne suffit pas, il faut aussi que le cms soit actif dans l'OPAC.

Portail > Construire

Cliquer sur « Liste des versions » et ensuite sur l'icône du crayon pour éditer le cms. La liste des versions apparaît.

La case « CMS actif par défaut en OPAC » est déjà cochée dans la démo.



N.B. Je reparlerai plus tard de la configuration du paramètre cms dans un PMB standard.

Paramètre first_page_params

Sous-type first_page_params

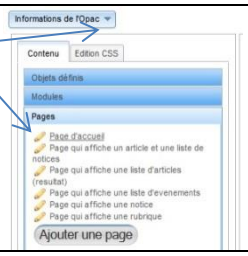
Valeur
```
{
"lvl":"cmspage",
"pageid":"3"
}
```
Valeur 3 dans le cas de démo

Lorsqu'on construit un portail, on peut choisir la page qui deviendra la page d'accueil. On peut garder la page catalogue « classique » de l'OPAC, ou créer une page d'accueil « style portail ».
Dans le 2e cas, il faut d'abord créer soi-même les divers types de pages nécessaires dans le portail.
Chaque page a un identifiant, un numéro d'id. Vu que PMB ne peut pas « deviner » quelle page vous avez choisie comme page d'accueil, ni dans quel ordre vous avez créé les pages, il faut lui donner l'identifiant de la page sélectionnée. Sinon, cela déclenche des problèmes.

Pour connaître le numéro d'id de la page d'accueil ou d'une autre page

Portail > Construire
Cliquer sur « Pages »

Sélectionner la page concernée, et cliquer ensuite sur la petite flèche à droite des mots « Informations de l'OPAC ».
Si c'est une page du portail, vous verrez le numéro d'id de la page à la fin de la liste des paramètres. Dans le cas de la démo, pageid=3 pour la page d'accueil



En situation réelle, vous devrez créer vous-même les pages du portail, donc les numéros identifiants des pages de votre base seront probablement différents de ceux de la démo, et il faudra configurer correctement ce paramètre : voir Configurer first_page_params

CONFIGURER L'ASPECT ÉDITORIAL

Certains paramètres sont déjà configurés dans la démo, sinon on ne peut pas créer des rubriques et des articles :

- **types de contenu pour les rubriques**
- **types de contenu pour les articles**
- **statuts de publication**

Ce paramétrage se fait dans l'onglet Administration > Contenu éditorial.	
Liste des types de contenu des articles, tels qu'ils ont été définis dans la démo, et des champs personnalisés qui y sont liés.	

En guise d'exemple, je vais expliquer un type de contenu prévu pour les articles : un événement à une date précise.

Si vous voulez relier un article à une activité, vous ne pouvez pas vous contenter d'un article au contenu « simple », il vous faut « quelque part » la possibilité de sélectionner cet article uniquement sur base de la date à laquelle cet événement a lieu. La création d'un champ personnalisé de type « date » est dès lors indispensable pour assurer cette connexion entre l'article et le calendrier d'animations de votre bibliothèque.

Formulaire de création ou de modification d'un champ personnalisé - dans ce cas-ci, il est de type « date ».	
Ceci n'est qu'une première approche de ces thèmes (types de contenu et statuts de publication), je les traiterai en détail ultérieurement. Voir Contenu éditorial	

Dans une base standard, vous pouvez partir de zéro, et tout créer (types de contenu et statuts), mais si vous avez une version compatible de la démo, vous pouvez facilement récupérer sa configuration éditoriale, et l'étoffer ensuite.

Voir Notes sur la compatibilité entre les versions de PMB

En outre, cela permet d'employer directement les templates Django inclus dans la démo et les requêtes SQL que j'ai mises au point. Il suffit de restaurer dans votre base de travail le jeu de sauvegarde « CMS – Minimum » (créé dans la démo).

Infos complémentaires : voir Sauvegarder – restaurer la configuration minimum

LES ÉTAPES DU PROCESSUS

ETAPE PRÉLIMINAIRE : DÉCOUVRIR LE PORTAIL GRÂCE À LA DÉMO

On dit souvent qu'une image ou qu'un bon schéma valent mieux qu'un long discours, il va est de même pour le portail.
La théorie expliquée dans un tutoriel ne suffit pas, rien ne vaut la pratique !

Voir Conseils pratiques (2)

Il est de toute façon impossible de créer un portail sans avoir vu et expérimenté des réalisations concrètes !

PREMIÈRE ÉTAPE : CHOISIR UNE FEUILLE DE STYLE – EN CHANGER ÉVENTUELLEMENT

Ne modifiez aucun paramètre lié au portail pour le moment !

Le choix de la feuille de style est primordial. Cela ne sert à rien de « fignoler » le paramétrage du portail, si vous ne disposez pas encore de la feuille de style définitive – au minimum, définitive en ce qui concerne la taille et la position des zones. Un changement ultérieur au niveau de la charte graphique ou de la mise en forme du texte poserait moins de problèmes. Donc, faites si nécessaire des tests de comparaison entre plusieurs feuilles de style avant d'activer le portail.

Si vous changez trop tard de feuille de style, vous devrez recommencer en partie la construction du portail. En effet, il faut configurer les cadres issus des modules. Or si vous avez changé la taille ou la position des zones, vous devrez réajuster un certain nombre de cadres. Bien sûr, on peut reconfigurer leurs propriétés CSS, mais autant s'épargner du travail inutile.

Le **changement de feuille de style** doit se faire dans une base de travail dont la configuration générale de l'OPAC est identique à celle de la base en production, sinon vous aurez des problèmes plus tard, au moment de restaurer le portail dans la base en ligne. Il faut donc au minimum la même table des paramètres, la même feuille de style, les mêmes infopages.

Voir Feuilles de style Conseils pratiques (3) Conseils pratiques (4)

Il est de toute façon impossible de créer un portail « à l'aveuglette », sans avoir une feuille de style de référence !

DEUXIÈME ÉTAPE : PARAMÉTRER ET ÉTOFFER LA BASE DE TRAVAIL

- **restaurer la configuration globale de votre base en ligne, adapter les 2 URL de base** (gestion et OPAC)
- si vous avez changé de feuille de style, vous avez déjà au moins restauré la table des paramètres
- si ce n'est pas encore fait, il y a d'**autres éléments à transférer : indexation décimale, thésaurus, configuration des notices, exemplaires, sections …**
- il faut en effet que le paramétrage global des 2 bases (de travail et en ligne) soit aussi similaire que possible

- **il est inutile de restaurer tout le contenu bibliographique de votre base, cela ne sert à rien**, cela va alourdir inutilement la base de travail et compliquer la gestion
- **il vaut mieux importer un lot réduit mais significatif de notices, sélectionnées en fonction des articles de type bibliographique ou événementiel que vous comptez publier dans un premier temps**
- **importer aussi les données d'exemplaires de ces notices**, pour vérifier l'affichage dans l'OPAC
- répartir ces notices dans quelques **paniers et étagères virtuelles**
- créer **2-3 lecteurs (même fictifs)** – pour tester l'impact des **statuts de publication**
- la période de test aura ainsi plus de sens pour vous, car vous pourrez créer des liens entre le futur contenu éditorial de votre portail et le contenu bibliographique de votre base actuelle

Vous pouvez désormais activer le portail, et configurer certains de ses paramètres

- activer le portail et accorder des droits aux utilisateurs
- configurer le paramètre url_base_cms_build, le paramètre cms et rendre le cms actif dans l'OPAC

Voir Paramétrage du portail Paramètre cms Conseils pratiques (3) Transfert vers la base de travail

Il est de toute façon impossible de créer un portail sans avoir paramétré et étoffé une base de travail similaire à la vraie base !

TROISIÈME ÉTAPE : CRÉER UN CONTENU ÉDITORIAL

Préliminaires : Configurer l'aspect éditorial

- préciser les **types de contenu des rubriques** et créer si nécessaire des **champs personnalisés**
- préciser **les types de contenu des articles** et créer si nécessaire des **champs personnalisés**
- définir les **statuts de publication**
- **restaurer de préférence la** configuration minimum de la base de démo, et l'étoffer ensuite, ce qui évite de devoir tout recréer (on récupère ainsi le paramétrage de certains statuts et types de contenu). En outre, cela permet d'employer directement les templates Django inclus dans la démo et les requêtes SQL que j'ai mises au point.

Il est de toute façon impossible de gérer les rubriques et articles sans avoir défini des statuts de publication et des types de contenu !

Créer les rubriques

- créer l'arborescence des rubriques, distinguer les rubriques qui feront partie du menu principal et les autres, établir d'autres types de distinction en fonction des besoins spécifiques, gérer l'arborescence
- réfléchir à la structure du menu principal : est-elle claire ? les titres sont-ils explicites ?
- il faut que les lecteurs puissent naviguer facilement dans le menu, trouver rapidement où se trouve l'info recherchée

Il est de toute façon impossible de rédiger un article sans avoir une rubrique où le mettre !

Rédiger des articles

- **prévoir des articles de divers types, par exemple**
  - **article « standard » dont le contenu est assez stable, et qui est donc visible en permanence** : présentation de la bibliothèque, de l'institution, aide à la recherche, horaires, conditions d'inscription et de prêt …
  - **article de type bibliographique** : nouveautés, coups de cœur, thème du mois, médiathèque …
  - **article de type événementiel, qui peut avoir un lien direct avec le contenu bibliographique** : une activité qui se déroule à une date précise ou pendant une période (exposition thématique, animation, rencontre avec un auteur …)
- **l'essentiel est de créer, pour les tests, un aperçu représentatif des divers types d'articles que vous seriez susceptible de publier**. **Ils peuvent être courts**, 5 ou 6 lignes significatives suffisent au début.

Il est de toute façon impossible de paramétrer la plupart des cadres issus des modules sans les relier à une rubrique ou à un article !

Certains noms de modules reflètent un lien indéniable vers le contenu éditorial.
Citons pour commencer les **articles, rubriques, listes de rubriques, listes d'articles et carrousels d'articles.**
Le **menu** et le **fil d'Ariane** présupposent l'existence de rubriques bien hiérarchisées et bien organisées.
L'**agenda** n'a de sens que si vous avez des événements (= des articles) à y placer.
Certains modules sont « mixtes » : **recherche dans le catalogue ou dans les articles.**
D'autres modules sont liés uniquement au contenu bibliographique : **notices, liste de notices et carrousel de notices**.
Il y a aussi des **modules autonomes (par rapport à votre base)**, comme le **flux RSS**, qui dépend de sources externes.

Voir Contenu éditorial

QUATRIÈME ÉTAPE : CRÉER LES PAGES DU PORTAIL

- **créer les pages indispensables du portail** (celles présentes dans la démo) et d'autres (pour les tests)
- Attention ! Créer une page ne la rend pas utilisable tout de suite, elle est vide au départ !
- Il faudra donc y insérer des cadres issus des modules (voir étape n° 5).
- **adapter la valeur du paramètre first_page_params** en fonction des pages créées, **si vous voulez remplacer tout de suite la page d'accueil classique de l'OPAC par une page d'accueil « style portail »**
- **aussi longtemps que les pages du portail sont vides, cela ne sert à rien de modifier ce paramètre**

Voir Pages de l'OPAC et du portail Paramètre first_page_params

Il est de toute façon impossible d'exploiter toutes les richesses des modules et du contenu éditorial sans les pages du portail !

CINQUIÈME ÉTAPE : INSÉRER LES MODULES ET CADRES INDISPENSABLES

Au stade actuel, vous ne pouvez hélas rien faire avec les pages du portail !
En effet, elles sont vides, elles ne contiennent aucun cadre issu d'un module.

Il faut rendre les pages du portail opérationnelles, il faut donc y **insérer les modules et cadres indispensables**, afin que ces pages puissent remplir la fonction qui leur a été assignée, par exemple

- page qui affiche une rubrique
- page qui affiche un article lié à une liste de notices
- page qui affiche une liste d'événements …

Voir Modules, cadres et pages et plus particulièrement le chapitre Pages-outils du portail

Il est de toute façon impossible d'établir des liens entre le contenu éditorial et bibliographique si les pages du portail sont vides !

SIXIÈME ÉTAPE : COMPLÉTER L'INTERCONNEXION ENTRE TOUS CES ASPECTS

N.B. Une première interconnexion entre tous ces éléments a déjà été réalisée lors de la 5e étape, au moment où vous avez inséré les modules et cadres indispensable, **mais il faut poursuivre ce travail**.

Vous pouvez désormais faire évoluer votre portail :

- compléter les types de contenu des rubriques et articles, ajouter des statuts de publication, modifier l'arborescence des rubriques, ajouter, supprimer ou déplacer des rubriques ou des articles
- ajouter ou modifier des pages
- ajouter ou modifier certains modules, insérer des cadres sur les pages, les déplacer, créer un menu
- relier les modules au contenu bibliographique et/ou éditorial
- adapter certains templates, afin d'améliorer la présentation …

Vous pourriez même retourner en arrière si nécessaire, et modifier le paramétrage de certains éléments.
Toutefois, c'est dans cet ordre-là que vous aurez parcouru les diverses étapes du processus.

Le seul genre de retour en arrière qui est vraiment déconseillé, c'est un changement trop tardif de feuille de style ou une modification trop tardive de certaines propriétés CSS fondamentales (comme la taille et la position des zones).

Voir Modules, cadres et pages

SEPTIÈME ÉTAPE : TRANSFÉRER LE PORTAIL VERS VOTRE BASE EN LIGNE

C'est la dernière phase du travail : préparer le transfert, restaurer dans votre base en ligne les éléments du portail, et procéder aux derniers ajustements nécessaires.

Il y aura forcément des corrections à effectuer puisque vous avez travaillé sur une copie incomplète de votre base.
Ex. : les liens entre les articles et les notices ne seront pas corrects (numéros identifiants différents).

Mais dans d'autres cas (ex. catégories, étagères), les n° d'id peuvent correspondre si on a travaillé de manière systématique.

Voir Configuration globale d'une base Transfert du portail depuis la base de travail vers la vraie base

L'ordre des chapitres dans ce tutoriel n'est donc pas le fruit du hasard !!

Il reflète l'ordre logique dans lequel il faut effectuer les actions qui transforment un OPAC traditionnel en un portail.

Comme vous le voyez, vous devez d'abord « fabriquer vos briques » avant de pouvoir construire le portail ;-)

CONSEILS PRATIQUES (2)

Phase d'apprentissage et de pré-tests

Le contenu de la base de travail (pmb4perso) va sûrement évoluer au fil du temps.

Vous pouvez y installer d'abord une base identique à la démo du portail. Bien que ce ne soit pas obligatoire, je vous conseille vivement de le faire.

- Bien sûr, cela peut sembler redondant par rapport à pmb4portail (qui contient elle aussi la démo), mais il est parfois pratique de garder la base pmb4portail « intacte », et de faire des essais ailleurs, sans se préoccuper des conséquences.
- Vous pourriez aussi faire ces tests dans pmb4portail : en cas de souci, il suffirait de restaurer une sauvegarde ou de réinstaller la base. Mais si vos essais ne sont guère concluants, vous perdez un atout : votre modèle (pmb4portail) est temporairement inaccessible**.**

En phase d'apprentissage et de pré-tests, il est vraiment plus simple de travailler dans un onglet (pmb4perso) et de consulter un portail de référence (pmb4portail) dans un autre onglet du navigateur.

Cette démo offre un avantage indéniable : vous ne devez pas configurer ni étoffer un PMB standard avant de pouvoir faire les premiers tests, vous disposez tout de suite d'un support complet pour la phase d'apprentissage.

- En effet, vous y trouvez une **base bien paramétrée**, un **éventail réduit mais significatif de notices et d'exemplaires**, un **thésaurus**, une **indexation décimale, ainsi qu'un contenu éditorial (rubriques et articles)**
- Cette démo contient un « modèle de portail » (rubriques et articles, pages, cadres, templates …), c'est donc un exemple que vous pouvez analyser dans un premier temps, afin de comprendre comment il fonctionne. Ensuite, vous pourrez essayer de reproduire certains éléments, et finalement d'innover.

La démo du portail n'est pas qu'une « démo à regarder », c'est aussi une « démo à expérimenter, à modifier ».

- Vous pouvez créer de nouveaux statuts de publication, de nouveaux types de rubriques ou d'articles, rédiger quelques articles, insérer des événements dans le calendrier ...
- Je vous conseille aussi de créer des pages supplémentaires, des pages de test.
- Au départ, ces pages nouvelles seront vides, vous aurez ainsi l'occasion d'apprendre comment y insérer les cadres et modules indispensables, comment établir des liens entre le contenu éditorial et bibliographique ...
- **Ne restez pas passif face à la démo ! Vous vous débrouillerez mieux lorsque vous devrez travailler « en vrai », avec un portail vide au départ, si vous avez testé un certain nombre de choses dans une copie de la démo.**

Il est vrai que cette copie de pmb4portail n'a qu'un temps. La démo n'est pas une base de travail.

- Il faut en effet, à un moment ou l'autre, installer dans pmb4perso un PMB standard avec les données essentielles pour démarrer rapidement, y restaurer la configuration globale de votre base et un éventail de notices.
- *Pas de précipitation inutile ! Travaillez d'abord avec la démo, créez des pages et des cadres, entraînez-vous …*

N.B. Ne prenez pas cette phrase au pied de la lettre : « il faut installer dans pmb4perso un PMB standard » !

- Oui, pour élaborer un portail, il faut partir d'une base qui ressemble globalement à la vôtre, donc un PMB standard dans lequel vous avez restauré certains éléments de votre base en production.
- Mais si vos essais avec la copie de la base de démo (donc dans pmb4perso) ont été particulièrement concluants, pas de masochisme, ne supprimez pas tout de suite vos plus belles réussites éditoriales ou graphiques !
- Conservez cette première ébauche du portail dans pmb4perso, et créez par exemple le dossier pmb4perso2 (ou un autre nom), afin de garder un accès aisé aux modèles que vous avez déjà mis au point ou aux articles déjà rédigés.

JOURNAL DE BORD

Liste des changements à faire ultérieurement

L'idée de tenir un journal de bord peut paraître bizarre de prime abord, mais je pense toutefois que c'est utile.
L'idée d'y consacrer un chapitre spécifique au début de ce tutoriel peut sembler tout aussi saugrenue, et pourtant …

Il y aura des changements à faire ultérieurement, aussi bien dans la base de travail que dans la base en production.

- L'idéal est de démarrer la construction du portail avec une base de travail, dans laquelle vous avez restauré autant que possible la configuration globale de votre base actuelle, donc entre autres la table des paramètres.
- Si vous changez de feuille de style pour passer au portail, vous allez probablement vous rendre compte qu'il faut modifier certains paramètres dans la base de travail (liens de l'OPAC qui changent de fonction, infopages …).
- Or, vous ne pouvez peut-être pas faire tout de suite les mêmes modifications dans la base en ligne, car il se peut que cette dernière doive continuer à fonctionner pour le moment avec l'ancienne feuille de style.
- Donc, la table des paramètres de la base de travail va évoluer au fil des essais successifs et des ajustements nécessaires.
- D'autres changements peuvent intervenir entretemps dans la base en ligne : vous allez peut-être découvrir et activer de nouvelles fonctionnalités de PMB, modifier certains paramètres pour la recherche dans l'OPAC, etc.
- Donc, la table des paramètres va vraisemblablement évoluer aussi dans la vraie base.
- C'est pourquoi il est déconseillé de réaliser, à la fin du processus d'élaboration du portail, un transfert de la table des paramètres depuis la base de test vers la base en ligne. Vous risquez de perdre les changements apportés entretemps à la base en ligne.
- Et il n'y a pas que les paramètres ! Vous allez aussi créer de nouvelles étagères, de nouveaux paniers, etc.

Vu que vous ne pourrez pas toujours faire les mêmes modifications simultanément dans la base de travail et dans celle en ligne, vous courez un risque : celui d'oublier de faire certains changements lorsque vous essayerez d'activer le portail dans la base en production, et d'y créer des dysfonctionnements qui n'existaient pas dans la base de test.

Il y aura forcément un décalage dans le temps entre les premiers essais de portail et son « inauguration officielle », c'est pourquoi je vous conseille de tenir un journal de bord, et de noter les changements à opérer, les points délicats, etc.

N'écrivez pas sur des petits bouts de papier que vous finirez par égarer ou jeter à la poubelle ;-)
Prenez note en traitement de texte, dans le tableur ou la base, et constituez petit à petit la liste des paramètres à modifier, des pièges à éviter.

il y a un zip avec des requêtes SQL spécifiques au portail **que vous pouvez télécharger, et importer dans PMB.**
http://amcubat.be/docpmb/portail-construction-concepts-exemples

Dans ce contexte, il y a une requête bien utile : la liste des paramètres (dossier Utilitaires – Gestion de la base).
Exécutez cette procédure dans l'onglet Editions, et transférez le résultat vers le tableur : vous aurez la liste complète des paramètres, de leurs numéros d'id, de leurs valeurs, des commentaires et des sections dont ils dépendent.
Cela permet des comparaisons aisées entre les tables des paramètres de 2 bases différentes (la même version de PMB !

Exemple de résultat obtenu dans un PMB standard grâce à la requête : contenu de la table des paramètres					
Paramètres - Liste complète : type, nom, valeur, commentaire et section				Tri par type de paramètre, ensuite par nom du paramètre	
N° id	**Type**	**Nom**	**Valeur**	**Commentaire**	**Section**
741	Cms	Active	0	Module 'Portail' activé. 0 : Non. 1 : Oui.	
758	Cms	cache_ttl	1800	durée de vie du cache des cadres du portail (en secondes)	
797	Cms	url_base_cms_build		url de construction du CMS de l'OPAC	
106	Opac	biblio_adr1	ZI de Mont/Loir BP 10023	Adresse 1 de la bibliothèque ou du centre de ressources dans l'opac	b_aff_general
108	Opac	biblio_cp	72500	Code postal dans l'opac	b_aff_general
107	Opac	biblio_town	CHATEAU DU LOIR	Ville dans l'opac	b_aff_general

Il suffit d'insérer une colonne dans le tableur, et vous pouvez noter au fur et à mesure les nouvelles valeurs dans le cas du portail.						
N° id	**Type**	**Nom**	**Ancienne valeur**	**Nouvelle valeur**	**Commentaire**	**Section**
741	cms	Active	0	1	Module 'Portail' activé. 0 : Non. 1 : Oui.	
758	cms	cache_ttl	1800		durée de vie du cache des cadres du portail	
797	cms	url_base_cms_build		http://127.0.0.1/pmb4perso/opac_css/	url de construction du CMS de l'OPAC	
N.B. Je ne complète les cellules de la colonne « Nouvelle valeur » que s'il y a un changement, je repère ainsi plus vite les paramètres à adapter. **Si vous avez bien noté les modifications à faire, cette liste vous aidera à reconfigurer correctement les paramètres de la base en ligne.**						

La liste des changements à faire ultérieurement dans la base en ligne ne concerne pas seulement la table des paramètres. On peut aussi tenir un journal de bord en partie dans la base de travail elle-même, en intégrant des commentaires de gestion dans les articles.
Voir Gérer globalement le contenu éditorial

CONSEILS PRATIQUES (3)

Pendant la construction du portail

- **Installez dans pmb4perso un PMB standard** (avec les données essentielles pour démarrer rapidement).
- **Ne modifiez aucun paramètre lié au portail pour le moment !**
- **Créez les** jeux de sauvegarde **nécessaires, et lancer les sauvegardes.**
- **Une des sauvegardes partielles est importante : CMS, toutes les tables liées au portail.** En effet, si vous devez repartir à zéro, restaurer cette sauvegarde CMS faite au début du processus vous rendra un portail « vierge ».

- **Si vous voulez tester rapidement quelques feuilles de style**, afin de voir s'il y en a une dont vous pourriez vous inspirer**, c'est le bon moment pour le faire, avant d'activer le portail (test dans pmb4perso ou ailleurs).**
- En effet, si certains paramètres fondamentaux du portail sont activés (cms = 1 et cms actif dans l'OPAC) et que vous changez plusieurs fois de feuilles de style, la CSS des zones et des cadres de cette base-là ne ressemblera plus à rien.
- Si ces tests ont eu lieu dans une base où le portail était activé, restaurez la sauvegarde CMS afin de repartir à zéro.
- **Si vous avez personnalisé une feuille de style fournie d'office avec PMB (ex. genbib), ne conservez pas le nom genbib pour désigner le répertoire où elle se trouve.** Appelez ce répertoire par exemple genbib_bis.
- Il est en effet « doublement imprudent » de garder le nom d'origine, et il y a des règles à respecter quand on change le nom d'un répertoire – voir Généralités sur les feuilles de style

- **Si ce n'est déjà fait, inaugurez votre** journal de bord**, il est temps !**
- **Dès que le choix de feuille de style est définitif, vous pouvez restaurer dans pmb4perso certaines données issues de votre base en production.**
- **Restaurez d'abord la** configuration globale de votre base**, surtout la table des paramètres.**
- **Adaptez les 2 URL de base (gestion et OPAC) dans la base de travail :** elles doivent changer, elles sont forcément erronées car vous venez de restaurer ailleurs la table des paramètres de votre base en ligne.

- **Cette série de conseils ne vous concerne que si vous comptez changer de feuille de style.**
- **La configuration des 2 bases doit être au départ identique en ce qui concerne la feuille de style par défaut, les infopages et les liens en page d'accueil, afin que vous déterminiez exactement quelles modifications faire.**
- **Si vous employez en production une feuille de style personnalisée**, pensez à copier-coller dans pmb4perso le répertoire qui correspond à la feuille de style utilisée en ligne (dans le dossier pmb/opac_css/styles/).
- Vous pouvez maintenant **modifier le paramètre default_style et lui donner sa valeur définitive.**
- **N'oubliez pas que les pages classiques de l'OPAC restent visibles, même si vous avez créé un portail.**
- **Je pense surtout à la page catalogue (l'ex-page d'accueil de l'OPAC).**
  - modifiez si nécessaire la valeur de certains paramètres qui contiennent des liens, comme biblio_preamble_p1, biblio_preamble_p2, biblio_important_p1, biblio_important_p2, etc – ces liens ont peut-être changé de fonction
  - vérifiez les liens vers les infopages et envisagez leur avenir – elles vont probablement disparaître plus tard et être remplacées par des articles
  - réfléchissez aussi à l'avenir des étagères virtuelles : liste ou carrousel de notices ? article lié à une liste de notices ?
- **Exemple concret de changement de feuille de style** dans la 4e série de conseils pratiques.
- Si les modifications sont complexes, faites des **sauvegardes intermédiaires de la table des paramètres**.
- **N'oubliez pas de tenir votre** journal de bord **à jour**, **vous devrez plus tard refaire les mêmes modifications dans la base en production**. C'est pourquoi j'ai insisté sur le fait que le processus de changement de feuille de style doit se faire dans une base de travail dont la configuration générale est identique à la vôtre.
- Il ne faut pas toujours attendre que le portail soit tout à fait finalisé pour changer déjà de feuille de style dans la base en ligne. Certains d'entre vous pourront le faire rapidement, et d'autres moins vite. En fait, cela dépend surtout de la nouvelle feuille de style choisie : permet-elle de garder facilement en page d'accueil d'un OPAC classique les liens externes ou vers les infopages ? Supprimer ces liens sans prévoir une alternative peut priver les lecteurs de renseignements utiles.

- Transférez **si nécessaire** d'**autres éléments de la configuration globale** : indexation décimale, thésaurus, paramétrage des notices, exemplaires, localisations, sections …
- N'oubliez pas d'**alimenter la base**. **Importez un lot réduit mais significatif de notices, sélectionnées en fonction des articles de type bibliographique ou événementiel que vous comptez publier dans un premier temps**
- Il faut quelques **données d'exemplaires** pour tester l'affichage dans la page catalogue classique.
- Répartissez ces notices dans quelques **paniers et étagères virtuelles** (les étagères sont importantes dans le portail)
- Créez **2-3 lecteurs (même fictifs)** – pour tester l'impact d'un **statut de publication** (public ou réservé aux abonnés)

- **Vous pouvez désormais activer les paramètres spécifiques du portail, puisque votre feuille de style est définitive** -(c'est donc le bon moment pour tout le monde, que vous ayez changé entretemps de feuille de style ou non).
- Paramétrer le portail **:** activer le portail, accorder des droits aux utilisateurs, configurer url_base_cms_build, configurer le paramètre cms et rendre le cms actif dans l'OPAC.

- Vous devrez définir le contenu éditorial : types de contenu des rubriques et articles, statuts de publication. Pour éviter de démarrer à zéro, vous pouvez restaurer la configuration minimum de la démo, et la compléter ensuite. En outre, cela permet d'employer directement les templates Django de la démo et les requêtes SQL que j'ai mises au point.
- Il faut bien sûr rédiger des rubriques et des articles, un **éventail des divers types d'articles que vous pourriez publier** (événement spécial, exposition, présentation d'un livre ou d'un auteur, présentation de la bibliothèque et de son fonds, infos pratiques, etc). Ces articles peuvent être courts au début, vous êtes en période de test.

- Il faut aussi créer les pages du portail. Attention ! Au départ, la page est vide (aucun cadre), donc inutilisable !
- Adaptez la valeur du paramètre first_page_params en fonction des pages du portail que vous avez créées, du moins si vous voulez remplacer tout de suite la page d'accueil classique de l'OPAC par une page d'accueil de type portail
- Si vos pages sont vides pour le moment (aucun cadre), il est inutile de modifier ce paramètre, vous le ferez plus tard.

- Il faut rendre les pages-outils **du portail opérationnelles**, c'est-à-dire y insérer les modules et cadres indispensables
- Ex. insérer le module « afficher un aticle » sur la « page qui affiche un article », sinon la page ne sert à rien

Il ne vous reste qu'à apprendre « 2 ou 3 petites choses ??? » : l'interconnexion entre tous ces éléments !

- Sélectionner les **modules et les sources de données,** établir des **liens entre contenu éditorial et bibliographique**, **insérer les cadres** issus des modules **dans une page de l'OPAC ou du portail**, **positionner les cadres** dans une zone, les **déplacer** si nécessaire, préciser leurs **propriétés CSS, gérer les templates, créer un menu** …

Il ne reste qu'à …. Il n'y a qu'à … Façon de parler ! Je sais que c'est plus vite dit que fait … Courage ;-)

TRANSFERT DE LA VRAIE BASE VERS LA BASE DE TRAVAIL

Le transfert d'un nombre important de données depuis la base en production vers la base de travail n'est pas obligatoire en toute circonstance. Cela dépend surtout des articles que vous comptez rédiger au début.

Le travail dans une base de test facilite la création de « l'architecture du portail » : créer les pages du portail, y insérer les cadres indispensables et prévoir les interconnexions. **Mais cet aspect-là du travail ne requiert pas une grande ressemblance entre le contenu bibliographique des deux bases.** L'essentiel dans cette optique, c'est qu'il s'agisse de la même feuille de style, et qu'il y ait une similitude au niveau de la configuration générale (la table des paramètres).

Un autre aspect du travail préliminaire dans une base de test, c'est la gestion de l'arbre du contenu éditorial (en d'autres termes, créer l'arborescence des rubriques) et **la rédaction des articles.**

Ces articles peuvent être de type bibliographique : présenter certaines notices (ex. les coups de cœur des bibliothécaires) ou les nouveautés (revisiter le concept des étagères virtuelles). Mais d'autres types d'article existent : annoncer des activités organisées par la bibliothèque, fournir des infos générales (aide à la recherche, conditions d'inscription et de prêt …)

- **Si vous prévoyez d'avoir rapidement de nombreux articles liés au contenu bibliographique de votre base,** il sera plus pratique de transférer certaines données (ex. thésaurus ou étagères) depuis la vraie base vers votre base de test.
- **Mais si ce nombre d'articles bibliographiques est assez réduit,** si vous avez surtout rédigé des articles d'infos générales (afin de remplacer vos anciennes infopages), un transfert relativement complet n'est pas indispensable.

Ne prenez donc pas tous les conseils qui suivent au pied de la lettre.

Ex. Le transfert du thésaurus ne sera guère utile s'il ne faut lier que 4 ou 5 articles aux descripteurs.
On peut créer ce lien entre articles et catégories plus tard, après avoir transféré le portail depuis la base de travail vers la base en ligne, cela ne vous prendra pas beaucoup de temps.

A vous de décider ce que vous allez transférer au début depuis la base en production vers votre base de test.

J'ai expliqué ci-dessous ce qu'il est possible de transférer et comment procéder, je n'ai pas dit que vous deviez le faire !!

Notices, exemplaires et étagères

Notices et exemplaires

- **Créez dans la base source des paniers de notices ou d'exemplaires, afin de les exporter et de les importer ensuite dans la base de travail** : vous récupérerez donc telles quelles les données de votre catalogue d'origine.
- **Vous pouvez aussi compléter l'éventail de notices grâce aux serveurs Z 39.50**.
- **Inutile de vous focaliser sur un catalogage complet jusque dans les moindres détails, allez à l'essentiel**.
- Il ne s'agit de toute façon que d'une copie pour travailler, la notice « bien cataloguée » est dans la base en ligne.
- **Il n'est pas nécessaire d'avoir tout à fait les « bonnes » données pour les exemplaires** : si le numéro d'exemplaire est par exemple fictif, cela ne perturbera en rien les tests. Les infos des exemplaires servent surtout pour tester l'affichage classique du catalogue dans l'OPAC. Il faut donc des cotes de rangement, localisations, sections et statuts cohérents, afin que l'exemplaire soit visible, et que le système fonctionne comme dans votre vraie base.
- Choisissez les notices à créer en fonction des articles que vous comptez rédiger dans un 1er temps, par exemple
  - Si vous parlez d'un livre, essayez d'avoir 3 ou 4 autres notices du même auteur
  - Si vous présentez une série ou une collection, essayez d'avoir 4 ou 5 notices qui en font partie
  - Si vous annoncez une exposition ou un événement, essayez d'avoir 4 ou 5 notices sur le même thème

Etagères virtuelles

- Vous avez probablement 3-4 étagères virtuelles avec les nouveautés dans divers domaines (documentaires, fiction adulte, fiction jeunesse, périodiques …). Il faut bien sûr des paniers et des notices pour alimenter ces étagères, mais pour les tests, il est inutile de disposer d'une liste complète, d'autant plus que les « nouveautés » d'aujourd'hui ne correspondront probablement plus à celles que vous mettrez en évidence le jour de l'inauguration du portail !
- Sachez que les étagères virtuelles ont beaucoup plus d'importance dans le portail que dans un OPAC classique.
- **Si une notice ou un cadre sont liés à une étagère,** cela se fait en général grâce à une requête SQL qui affiche la liste des étagères disponibles, et on choisit la bonne. En d'autres termes, **le lien se base sur le numéro d'id de l'étagère.**
- **Lorsque votre portail sera prêt, vous devrez le transférer depuis la base de test vers la base en ligne. Or, si les numéros identifiants des étagères sont différents dans les deux bases, vous aurez des ajustements à faire plus tard dans la vraie base : il faudra repérer <u>tous</u> les éléments liés à une étagère, et changer le lien.**
- **S'il y a assez peu de notices et de cadres liés à une étagère, ou si vous avez une liste (à peu près) complète des éléments concernés, cela peut se faire rapidement. Si les modifications sont plus nombreuses, agissez préventivement, <u>avant</u> de transférer les données depuis la base en ligne vers celle de test.**
- Ce système assurera un transfert plus « sûr » du portail et vous fera gagner du temps.
  - Avant de sauvegarder la table des étagères (voir ci-dessous, configuration globale d'une base), créez dans la base en ligne des étagères supplémentaires pour le futur portail. Appelez-les par exemple « Portail – Nouveautés Jeunesses », « Portail – Documentaires » et créez même 4-5 étagères « en plus au cas où … » (ex. « Portail – Etagère 1 »)
  - Veillez à ce que personne d'autre que vous ne puisse voir ou supprimer ces étagères (rappel : si vous deviez les recréer, leur numéro identifiant serait différent). Gardez-les en réserve « bien au chaud », et ne les employez pas dans la base en ligne. Aussi longtemps que le portail est en construction, utilisez en production les étagères et paniers habituels.
  - L'avantage de cette méthode, c'est que les numéros d'id des étagères du portail seront ainsi les mêmes dans les 2 bases.
  - Il n'est vraiment pas utile de créer préalablement les paniers qui alimenteront ces étagères. Le transfert des paniers d'une base vers l'autre est tout à fait facultatif : leur nombre risque d'ailleurs d'être fort élevé s'il y a beaucoup d'utilisateurs.
  - Avant de transférer le portail dans l'autre sens, alimentez les étagères de la base en ligne avec de nouveaux paniers qui conviendront pour le futur portail. Même si le contenu de l'étagère est différent, car ce ne sont pas (tout à fait) les mêmes paniers ni les mêmes numéros d'id de notices qu'en test, le numéro d'id de l'étagère sera identique, et c'est là l'essentiel.
- **La création préliminaire des étagères rend donc votre portail plus rapidement opérationnel ultérieurement.**

Configuration globale d'une base

- Il ne me semble pas utile de récupérer toute la base, donc j'ai limité volontairement le nombre de tables à sauvegarder dans la base en ligne, et à restaurer ensuite dans la base de travail.
- Restaurez bien entendu cet ensemble avant de commencer à configurer le portail.
- **Cette liste est une liste standard, il se peut qu'elle contienne quelques tables inutiles pour certains d'entre vous, et que d'autre part, elle ne soit pas assez complète pour d'autres, vu certaines spécificités de votre base.**
- Vous devrez peut-être faire des ajustements, mais au moins, vous avez une liste pour démarrer.

Tables à transférer, afin que la configuration globale de la base de travail ressemble le plus possible à celle de la base en ligne	
Parametres	Paramètres
users, users_groups	Utilisateurs – ce n'est pas indispensable, admin peut suffire pour créer le portail.
notice_statut, origine_notice docs_codestat, docs_location, docs_section, docs_statut, docs_type, docsloc_section, lenders	Configuration des notices : origines, statuts Configuration des exemplaires : codes statistiques, localisations, sections, statuts, types de support, liens entre sections et localisations, propriétaires
notices_custom, notices_custom_lists, expl_custom, expl_custom_lists	Champs personnalisés des notices et exemplaires Seulement le paramétrage – pas le contenu. A vous de voir si c'est utile ou non.
notice_tpl, notice_tplcode	Templates des notices – utile surtout si vous en avez défini pour l'OPAC
z_bib	Configuration des serveurs Z 39.50 – utile surtout si vous avez adapté la liste type
caddie, etagere **Lire d'abord la note sur les étagères !**	Paniers de l'onglet catalogue (facultatifs) et étagères (très utiles) Seulement la liste avec les numéros d'id - pas le contenu
categories, nœuds, thesaurus, voir_aussi indexint, facettes	Thésaurus (4 tables) - Sans les liens entre les notices et les descripteurs Indexation décimale et facettes
Infopages	Infopages
caddie_procs, empr_caddie_procs,, procs, procs_classements,	Toutes les requêtes SQL des onglets Administration, Catalogue et Circulation, ainsi que le classement des procédures de l'onglet Administration
empr_categ, empr_codestat, empr_caddie	Configuration des emprunteurs : catégories, codes statistiques et paniers (vides) Transfert facultatif. En théorie, 2-3 lecteurs, même fictifs, suffisent pour tester.

Attention, je rappelle qu'il il s'agit d'un transfert à réaliser au début, depuis la base en ligne vers la base de travail !

Il serait vraiment dangereux de transférer ces tables dans l'autre sens, après avoir mis au point le portail !

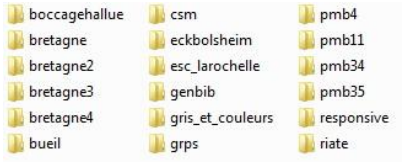
Votre base en production évolue en permanence : on ajoute, modifie ou supprimes des notices, exemplaires, lecteurs, catégories, étagères, etc. Il est aussi fort probable que l'on change entretemps l'un ou l'autre paramètre.

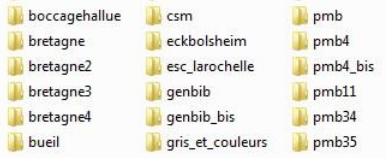
Or la phase d'élaboration du portail va prendre un certain temps, on pourrait même dire un temps certain, donc il serait risqué de restaurer plus tard dans la base en ligne des tables qui ne seraient forcément plus à jour.

Certes, à la fin du processus, il y aura un transfert depuis la base de test vers la base en production, mais ce transfert sera limité aux tables du jeu « cms », donc on ne va restaurer que les tables liées au portail lui-même.

FEUILLES DE STYLE

Généralités

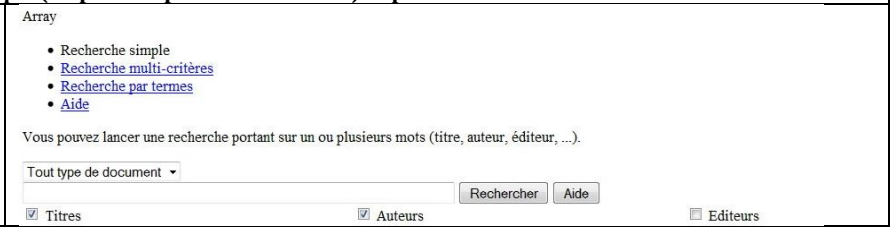
Chemin d'accès : Administration > Outils > Paramètres de l'OPAC : default_style	
Si vous voulez changer de style, il faut connaître la liste des styles possibles ! **Regardez le contenu du dossier pmb/opac_css/styles/** Les noms valides sont donc les noms des sous-dossiers. Attention à l'orthographe, pas de variantes ! **Rappel : saisir simplement ces noms dans la zone supérieure, ne pas ajouter d'espace, ne pas appuyer sur « entrée ». Pour valider le choix, cliquer sur « enregistrer ».**	

S'inspirer d'une feuille de style existante et l'adapter	
Si vous voulez modifier une feuille de style existante, afin de la personnaliser davantage, ne faites pas de modification directement dans le répertoire original. **Commencez par faire une copie du dossier en question, et renommez cette copie.** Saisissez le nouveau nom dans la zone supérieure du paramètre default_style. Vous voyez que j'ai créé les variantes genbib_bis et pmb4_bis N.B. **Des noms comme genbib-bis ou pmb4-bis sont interdits** ! (voir ci-dessous).	

Attention lorsque vous créez ou renommez un répertoire - sinon, pas d'URL valide ! Le choix du nom des variables, des champs personnalisés et des répertoires obéit à des règles précises.
• un nom en <u>un</u> mot (pas d'espace !), qui comporte uniquement des caractères alphanumériques non accentués (donc rien d'autre que des lettres ou des chiffres), et qui doit commencer par au moins une lettre - de préférence, entièrement en minuscules • aucun symbole mathématique, signe de ponctuation ou autre symbole n'est toléré • une seule exception : le trait de soulignement _ (underscore) afin de marquer la séparation en 2 mots (pmb4_bis, genbib_bis …) • ne confondez pas le trait de soulignement et le tiret (identique au symbole mathématique « moins », c'est pourquoi il est interdit)
Attention ! Certains serveurs sont sensibles à la casse : la différence entre majuscules et minuscules. Si le nom donné au répertoire est écrit au départ en minuscules, ne créez pas de variante en majuscules. Pour un serveur de type UNIX, pmb4 ≠ PMB4, donc un répertoire mal orthographié devient introuvable.

Modifier une feuille de style sans créer une copie du dossier est une double imprudence !

- **Si vous réalisez que les changements apportés ne sont finalement pas aussi heureux que vous l'espériez, ou si vous voyez que vous ne terminerez pas dans les délais, vous avez toujours l'original sous la main**. Il suffit de modifier le paramètre default_style, et votre catalogue aura à nouveau un aspect présentable aux yeux des lecteurs ;-)
- **Lors des mises à jour ultérieures de PMB, quand vous décompresserez le zip de PMB au même endroit que précédemment, la nouvelle version de PMB va « écraser » les dossiers et/ou fichiers qui porteraient le même nom.** Donc la version modifiée de votre feuille de style serait remplacée par la version d'origine de la même feuille de style (dans le zip), et vous auriez perdu les fruits de votre labeur, sauf si vous avez pris la précaution d'en faire préalablement une copie « ailleurs » et de copier-coller ensuite ce « clone » au bon endroit. Autant éviter ce genre de désagrément en ne travaillant pas sur le répertoire original.

Et si la feuille de style est introuvable ? **Parce qu'il y a une discordance entre la valeur du paramètre default_style et le nom donné au répertoire (faute de frappe ?), ou parce que le nom donné au répertoire ne respecte pas les règles et ne permet donc pas de construire une URL valide, ou parce que le répertoire n'a pas été recopié (ou pas recopié au bon endroit) - après une réinstallation de la base ailleurs.**	
Genre de message d'erreur qui apparaît **Votre OPAC n'est guère accueillant, c'est le moins que l'on puisse dire !**	

Voir aussi : http://amcubat.be/docpmb/personnaliser-pmb-nom-logo-feuilles-style

Choisir une feuille de style

Il est conseillé de partir d'une feuille de style récente, prévue pour le portail (comme pmb4 ou genbib), et de la reconfigurer si nécessaire. Prenez de préférence une feuille de style qui s'adapte aisément aux diverses tailles d'écran. N'oubliez pas qu'on consulte désormais aussi le catalogue sur tablette ou téléphone portable, en mode portrait ou paysage.

N.B. Pour tester rapidement quelques feuilles de style, pour en trouver une qui « vous inspire », un PMB standard (même vide) suffit amplement. Faites ces essais de feuilles de style avant d'activer les paramètres du portail.

Si vous voulez modifier une feuille de style existante, l'essentiel est de configurer d'abord la taille et la position des zones fixes : introduction (zone supérieure), bandeau de gauche et de droite, page centrale et pied de page.

En effet, il faut positionner les cadres issus des modules. Or si vous rechangez la taille ou la position des zones, vous devrez réajuster un certain nombre de cadres. Bien sûr, vous pouvez retravailler certains aspects de la CSS plus tard, mais il vaut mieux fixer dès le départ les « points de référence » des pages du portail. On s'épargne ainsi beaucoup de travail inutile.
Il faut aussi configurer la **charte graphique**, afin que l'ensemble soit harmonieux, prévoir le **logo**, les **photos** …
Editer une feuille CSS n'est pas évident, surtout si vous le faites pour la première fois, cherchez de l'aide si nécessaire.

Si vous changez de feuille de style, il faut adapter certains paramètres. En effet, les feuilles de style sont différentes, le même paramètre peut avoir des fonctions différentes, d'une feuille de style à l'autre, et d'une base à l'autre.

Le message de bienvenue sur le site (dans la zone supérieure) ne se trouve pas toujours dans le même paramètre.
Que vont devenir les liens que vous aviez placés auparavant en haut, dans le bandeau de gauche ou en pied de page ?
A cet égard, certains exemples que j'ai montrés sont révélateurs.
Comparez l'exemple n° 1 (OPAC traditionnel avec Genbib) et l'exemple n° 2 (portail avec pmb4)
Vous verrez que des paramètres comme biblio_preamble_1, bilbio_pramble_2 … n'ont pas toujours la même fonction.

En cas de changement de feuille de style, que devient la valeur initiale d'un paramètre ? Pas de règle générale !
Il faudra parfois effacer la valeur d'un paramètre, et parfois la remplacer par une autre, cela dépend de la feuille de style.
En outre, cela peut varier d'une configuration à l'autre. Vous avez peut-être employé certains de ces paramètres afin d'y insérer des liens externes, un menu, ou bien des liens vers des infopages. Une solution valable dans une base peut s'avérer inutile, voire même néfaste dans une autre base. Donc on en arrive vraiment à un « traitement au cas par cas ».

Rappel de la règle d'or : avant de modifier ou d'effacer le contenu d'un paramètre plus complexe (donc pas un simple choix entre 0 et 1), un copier-coller dans le champ commentaire ! Cela permet de retrouver la valeur initiale en cas de besoin.

Exemple concret de changement de feuille de style : voir 4e série de conseils pratiques

Cascading Style Sheets

Rassurez-vous, je ne vais pas vous donner un cours complet sur les feuilles de style, d'autant plus que je serais incapable de le faire !

Toutefois, je vous explique un des concepts de base des CSS : le concept des priorités, et son lien avec le portail de PMB (l'autre concept important est celui de la parenté et de l'héritage).

Le terme « Cascading Style Sheets » signifie « Feuilles de style en cascade ».
N.B. Si vous cherchez de l'info sur le HTML, les CSS, les CMS, etc, sachez que le site d'Alsacréations contient de bons tutoriels.
http://www.alsacreations.com/tutoriels/

Je commence par une comparaison avec le traitement de texte.
Je suppose que vous avez rédigé l'analyse littéraire d'un roman, et que vous avez prévu plusieurs types de style :

- Trois styles pour marquer la hiérarchie entre les chapitres et sous-chapitres (titre1, titre2, titre3)
- Un style pour le corps du texte (vos commentaires sur ce roman)
- Un style pour les interviews avec l'auteur
- Un style pour les citations extraites de ce livre
- Un dernier style pour mettre en exergue une partie d'une citation (quelques mots ou une phrase)

Prenons le cas de la phrase que vous avez mise en exergue dans une citation.
Cette « pauvre phrase » est la victime innocente d'une série d'ordres contradictoires !
En effet, vous lui imposez successivement les styles « corps du texte », « citation » et « extrait d'une citation ».

Que se passe-t-il au stade final ? Qu'advient-il de la phrase extraite de la citation ? Vous le savez déjà …
Elle n'aura pas le style « corps du texte » ni le style « citation », mais le style « extrait d'une citation »

Ceci illustre la règle de priorité : en cas de conflit entre plusieurs feuilles de style, c'est toujours le dernier niveau qui l'emporte, le niveau « le plus proche ».

Dans le cas de PMB en général et du portail en particulier, vous pouvez aussi définir les styles à différents niveaux :

- **La CSS de la feuille de style choisie** (ex. genbib ou pmb4)
- **La CSS définie dans le paramètre default_style_addon**
  - Si votre base est hébergée par exemple sur un serveur académique, il se peut que vous n'ayez pas aisément (ou pas du tout) accès au répertoire d'installation de PMB, et que vous ne puissiez donc pas modifier la feuille de style elle-même (contenue dans le dossier /pmb/opac_css/styles)
  - Dans ce cas, vous pouvez employer le paramètre default_style_addon, et y saisir des modifications de la feuille de style de départ (ex. taille d'une zone, couleurs …), modifications dont PMB tiendra compte au moment d'afficher l'OPAC.
- **La CSS définie directement dans le portail** (dimensions, position d'une zone ou d'un cadre …)

C'est à nouveau le dernier niveau qui l'emporte, donc la CSS définie dans le portail.
C'est pourquoi on parle de « **feuilles de style en cascade** » : elles se succèdent … mais elles ne se ressemblent pas forcément.

Attention ! Les conflits de CSS peuvent générer de l'instabilité dans le portail, surtout s'ils concernent les dimensions et la position des zones (introduction, bandeau de gauche et de droite, page centrale, pied de page) !
Veillez à définir les zones dans la feuille de style-même, et essayez de ne pas changer leurs propriétés principales à d'autres niveaux de CSS.
C'est moins problématique si vous ne modifiez que la mise en forme du texte ou la charte graphique.

CONSEILS PRATIQUES (4)

Changer de feuille de style

Remarques préliminaires

Ce changement se fait dans votre base de travail (pmb4perso), dans laquelle vous avez déjà restauré une copie partielle de votre base réelle - au moins la configuration globale, surtout la table des paramètres (qui contient entre autres le choix de la feuille de style et les liens en page d'accueil de l'OPAC), et les infopages éventuelles.
Adaptez dès le départ au moins 2 paramètres : l'URL de base de gestion, et l'URL de base de l'OPAC.

Il est indispensable de travailler dès le début avec « vos » paramètres : des divergences majeures entre les 2 bases risquent de provoquer des dysfonctionnements ultérieurs au moment de transférer les éléments du portail vers votre base en ligne, car vous avez peut-être oublié de faire l'une ou l'autre modification importante dans la base de test.
L'essentiel est d'avoir au départ **la même configuration de l'OPAC dans les 2 bases**.

Pour illustrer ce changement, cette évolution, j'ai choisi comme point de départ genbib (la feuille de style par défaut d'un PMB standard), **et le but est d'arriver à la feuille de style pmb4** (employée dans la démo du portail, et aussi par de nombreux bibliothécaires, même s'ils n'ont pas activé le portail).
Dans les exemples d'OPAC que j'ai montrés, cela correspond à un passage de l'exemple n° 1 à l'exemple n° 3.

N.B. Dans mon cas, la base pmb4perso contient au départ les mêmes données que la démo, mais aucun élément du portail.
Elle symbolise donc sans problème la situation dans laquelle vous êtes au début du processus.

Situation de départ : un PMB standard - feuille de style genbib

Paramétrage - Pour faciliter la comparaison entre feuilles de style

En page d'accueil, j'ai désactivé l'affichage des catégories (show_categ_browser = 0), afin que tout « tienne » en un écran.
Sinon, on ne voyait pas le pied de page, car la liste des descripteurs du thésaurus Agneaux est assez longue.

Aperçu des liens en page d'accueil

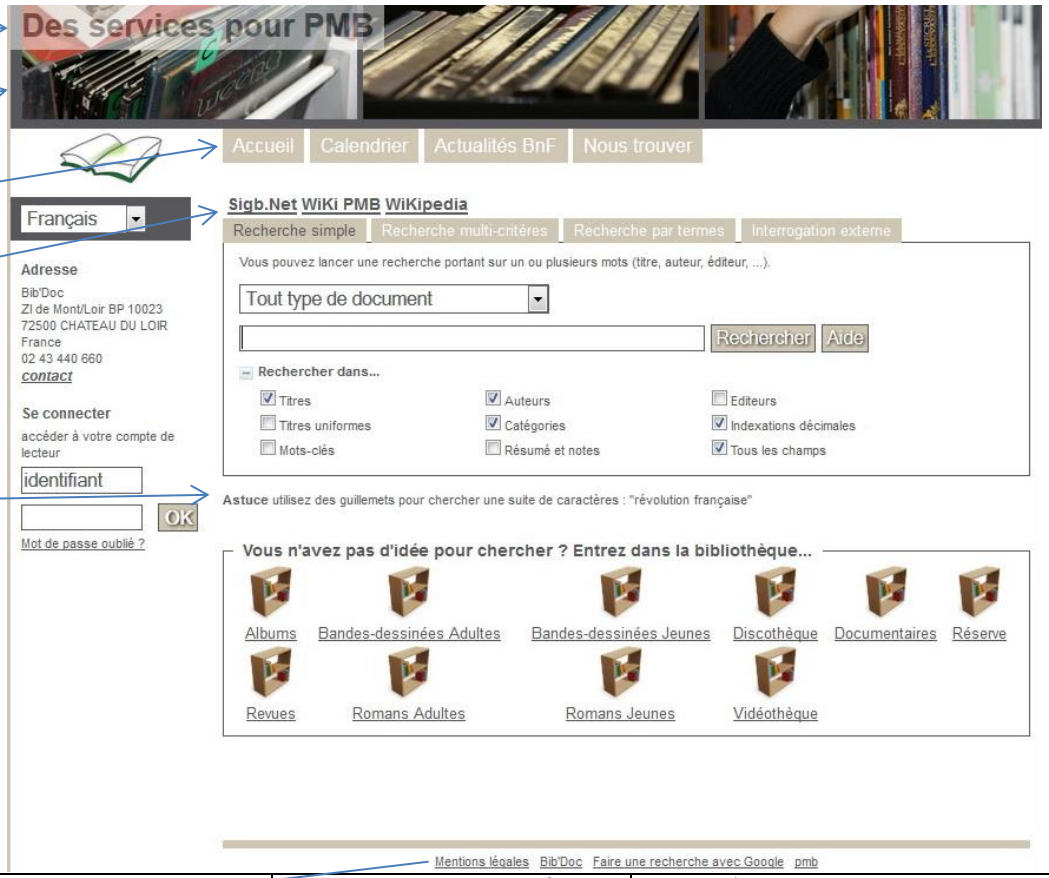
Biblio_main_header
(texte « Des services … »)

Biblio_preamble_p1
(photos)

Biblio_preamble_p2
(liens vers les infopages)

Biblio_important_p1
(liens externes)

Show_infopages_id = 5
Le texte « Astuce : Utilisez … »
se trouve dans l'infopage n° 5



Lien_bas_supplementaire	Biblio_name	Lien_moteur_recherche

Le but est d'obtenir un affichage correct de la page d'accueil classique avec pmb4

En effet, les pages traditionnelles de l'OPAC restent visibles, même si on active le portail.

La nouvelle valeur du paramètre default_style est pmb4	**Solution : effacer le contenu de biblio_preamble_p1 (photos)**
Ce changement n'est sûrement pas une réussite graphique !	**Ouf, il y a déjà une amélioration !**

Rappel de la règle d'or : avant de modifier ou d'effacer le contenu d'un paramètre plus complexe (donc pas un simple choix entre 0 et 1), un copier-coller dans le champ commentaire ! Cela permet de retrouver la valeur initiale en cas de besoin.

Comparaison afin de déterminer les modifications à faire

Version de démo du portail

biblio_preamble_p1
(lien vers le logo)

biblio_preamble_p2
(message de bienvenue)

Un PMB standard que j'essaie d'adapter au style pmb4

Biblio_preamble_p1
Vide (je venais d'effacer son contenu : les photos)

Je préfère créer un lien vers le logo, comme dans la démo

Biblio_preamble_p2
(liens vers infopages)

A garder ou à remplacer par un message de bienvenue suivant le contexte

Recherche par terme

Je préfère la supprimer, elle fait double emploi si on affiche les catégories

Biblio_main_header
(texte « Des services … »)

On peut effacer le contenu, me semble-t-il

Biblio_important_p1
(liens externes)

A garder ou à effacer suivant le contexte.

Show_infopages_id
Le texte « Astuce : … »

A vous de voir si vous gardez l'infopage et le lien vers cette infopage.

Remplacer le contenu de biblio_preamble_p1 afin d'afficher le logo comme dans la démo **Supprimer la recherche par terme** Paramètre de l'OPAC : show_term_search = 0	**Ce que je vais faire en guise d'exemple**	**Effacer le contenu des 2 paramètres : biblio_main_header et show_infopages_id**

Pour biblio_preamble_p2 et biblio_important_p1, je vais expliquer les options possibles en fonction du contexte :
version de la page d'accueil de l'OPAC dans une base sans portail (peu importe la raison)
version de la page catalogue de l'OPAC dans une (future) base avec portail

Que faire pour les paramètres biblio_preamble_p2 (liens vers les infopages **) et biblio_important_p1 (**liens externes) ?

- **Si vous voulez changer de feuille de style en ligne, mais sans activer le portail pour le moment**
  - Soit parce que vous ne pensez pas au portail dans l'immédiat, soit parce que votre portail est en cours d'élaboration ailleurs, et que vous devez garder entretemps un OPAC « classique » en production
  - Dans ce cas, vous allez probablement conserver pour le moment les liens vers les infopages et les liens externes, sinon vous rendriez certaines infos importantes temporairement inaccessibles pour les lecteurs
  - Ceci vous permet en tout cas de changer de feuille de style avant l'inauguration officielle du portail, d'habituer les lecteurs à cette nouvelle interface, et de réaliser éventuellement qu'il y a encore un ou deux petits problèmes graphiques à régler (il vaut mieux s'en rendre compte maintenant que plus tard)

- **Si vous changez de feuille de style dans la base de travail, celle où vous êtes en train de construire le portail**
  - Il est plus que probable que (presque) toutes vos infopages disparaîtront lorsque le portail sera finalisé, donc vous remplacerez assez vite les liens vers ces infopages par un message de bienvenue.
  - N'oubliez quand même pas que des infopages ont existé. Tenez-en compte au moment où vous élaborez l'arborescence des rubriques et où vous rédigez des articles, ces infos doivent se retrouver ailleurs.
  - Même si les liens externes ne se présentent plus sous la forme actuelle, il faudra aussi réfléchir à leur avenir : où les mettre, en ajouter éventuellement, comment les structurer, etc.

A vous de choisir, de toute façon.

C'est pourquoi j'insiste sur le fait de prendre des notes précises dans votre journal de bord. Il n'est pas sûr que certaines modifications se fassent simultanément dans la base de travail et dans la base en ligne.

	Voilà l'impact d'une première série de modifications	
Biblio_preamble_p2 (liens vers infopages) On peut garder ces liens provisoirement. Plus tard, on pourra employer le module « menu » du portail. **Recherche par terme** Je l'ai désactivée.		**J'ai effacé le contenu de ces paramètres, les messages et les liens externes ont donc disparu.** **Biblio_main_header** (texte « Des services … ») **Biblio_important_p1** (liens externes) **Show_infopages_id** Le texte « Astuce : … » .

Suite des modifications : afficher un message de bienvenue au lieu des liens vers les infopages		
Biblio_preamble_p2 (un **bref** message de bienvenue) Si on met seulement le texte (sans <div id=" >), l'affichage est moins beau : le texte n'est pas en diagonale ni en blanc		**Solution** : copier-coller dans votre base la 1e ligne du script html qui est dans le biblio_preamble_p2 de la démo du portail, et remplacer le texte original par celui que vous voulez afficher.
Vu qu'il n'y a qu'une seule ligne à copier, je vous ai épargné la peine de chercher dans la démo du portail ;-)	**Début du script dans la démo du portail** Bienvenue sur le Portail de la Bibliothèque **Début du script dans pmb4perso** Bienvenue dans notre bibliothèque	**Solution valable seulement pour la feuille de style pmb4 !**

Dernière modification : afficher le logo		
Biblio_preamble_p2 (lien vers le logo) **Dans la démo, ce lien est cliquable, et il renvoie vers la page d'accueil de l'OPAC.** **Mini-cours de html (1)**		**Mini-cours de html (2)** **Lien cliquable sur du texte** **Accueil** début de la balise <a> href="./index.php" URL de destination (page d'accueil) Accueil D'où on part, la source Le mot « Accueil » est cliquable Fin de balise
Lien cliquable sur image	****	
Ici, les 2 URL (la destination = page d'accueil et la source = fichier image) sont relatives par rapport au répertoire de départ : pmb/opac_css/ Le script index.php est dans le répertoire pmb/opac_css/ L'image docpmb_url.jpg se trouve dans le répertoire pmb/opac_css/images/site/		
	 href="./index.php" alt=" " 	Début de la balise <a> URL de destination (là où on va = page d'accueil de l'OPAC) URL de la source (d'où on part = ce qui cliquable - ici image) Texte alternatif qui apparaît si l'image n'est pas disponible Fin de balise	a = anchor (une ancre, un lien) href = hyperlink reference img src = image source (source de l'image = URL du fichier) alt = alternate text (texte alternatif)

Finalement, c'était beaucoup plus simple que prévu !

Toutefois, c'est un peu trompeur, la transition de genbib vers pmb4 était particulièrement aisée.

Ce n'est hélas pas toujours le cas ;-)

Rappel … Je sais que je me répète parfois, mais …

Je viens d'illustrer un changement de feuille de style : le passage d'un genbib « standard » à un pmb4 « standard ». Mais toutes les feuilles de style ne sont pas nécessairement des modèles standardisés.

Vous n'allez peut-être pas employer telle quelle une des feuilles de style disponibles au départ dans le dossier pmb/opac_css/styles (ex. genbib ou pmb4). Il se peut que votre ancienne ou votre nouvelle feuille de style soit une variante d'une des feuilles de style fournies d'office avec PMB.

Une variante, car vous avez par exemple remplacé des images ou changé certaines propriétés dans le fichier css lui-même (comme les couleurs), ce n'est donc plus le modèle original.
Dans ce cas, modifier le contenu du paramètre default_style ne suffit pas, il faut aussi transférer d'une base à l'autre le dossier qui contient cette feuille de style modifiée (donc une copie d'une base à l'autre, d'un pmb/opac_css/styles/ à l'autre).

Attention aux noms donnés aux répertoires :

- **Il est « doublement imprudent » de conserver le nom original d'un répertoire qui correspond à une feuille de style que vous auriez modifiée.** Il faut changer le nom du dossier (ex. pmb4_bis au lieu de pmb4), et ne pas prendre n'importe quel nom, il y a des règles strictes à respecter.

Voir Feuilles de style – Généralités

QUELQUES EXEMPLES D'OPAC

Pour réaliser cet aperçu, j'ai employé deux bases que j'ai installées dans des répertoires distincts :

- **pmb4portail** : c'est la démo du portail (PMB 4) - **avec la feuille de style pmb4**
- **pmb4donnees** : c'est un PMB 4 standard qui contient les mêmes données que la démo, mais pas de portail. Cette base symbolise donc votre vraie base de données. C'est un OPAC classique - **avec la feuille de style genbib**

J'en ai profité pour illustrer certains concepts de base :

- les **zones** : introduction, bandeau de gauche, bandeau de droite, page centrale, pied de page
- **les différences entre les pages de l'OPAC « traditionnel » et les pages du portail**
- les **6 types de page du portail qui existent dans la démo** :
  - page d'accueil
  - page qui affiche un article et une liste de notices
  - page qui affiche une liste d'articles (résultat d'une recherche dans les articles)
  - page qui affiche une liste d'événements
  - page qui affiche une notice
  - page qui affiche une rubrique
- **quelques exemples de pages de l'OPAC – visibles même si on a créé un portail**
  - page catalogue « classique » (qui était jusqu'à présent la page d'accueil de l'OPAC)
  - page qui affiche une liste de notices (résultat d'une recherche dans le catalogue)
  - page qui affiche une notice
  - page qui affiche un périodique

Quelle que soit la page concernée (page de l'OPAC ou page du portail), son aspect dépend bien sûr de la configuration de la base (feuille de style et autres paramètres de l'OPAC) et de celle du portail (zones, cadres, CSS, modules employés …). Je ne montre ici que quelques exemples parmi d'autres …

Ne vous inquiétez pas si au stade actuel, vous ne comprenez pas toutes les remarques que je fais dans les pages suivantes à propos des types de contenu des articles ou au sujet des modules et cadres : c'est tout à fait normal !
Mais je vous suggère de revenir à ces exemples plus tard, lorsque vous aurez lu les chapitres Configuration du contenu éditorial **et** Modules, cadres et pages**, et vous verrez que ce sera plus clair ;-)**

Un OPAC « classique »

Exemple n° 1	Page d'accueil d'un OPAC « traditionnel »	Style genbib
Introduction des photos (ou un logo) et un message		Message dans le paramètre biblio_main_header Lien vers les photos (ou un logo) dans biblio_preamble_p1
Bandeau de gauche Un logo en petit format		
Bandeau de gauche Choix de la langue Connexion au compte du lecteur	Français Se connecter accéder à votre compte de lecteur identifiant OK Mot de passe oublié ?	Liens vers les infopages dans biblio_ preamble_p2
Bandeau de gauche Adresse, téléphone, e-mail de contact	Adresse Bib'Doc ZI de Mont/Loir BP 10023 72500 CHATEAU DU LOIR France 02 43 440 660 contact	Liens externes dans biblio_important_p1
Page centrale Possibilité de recherche simple ou de recherche multicritères	Accueil · Calendrier · Actualités BnF · Nous trouver Sigb.Net WiKi PMB WiKipedia Recherche simple · Recherche multi-critères · Interrogation externe Vous pouvez lancer une recherche portant sur un ou plusieurs mots (titre, auteur, éditeur, ...). Tout type de document Rechercher · Aide Rechercher dans... Titres · Auteurs · Editeurs Titres uniformes · Catégories · Indexations décimales Mots-clés · Résumé et notes · Tous les champs Astuce utilisez des guillemets pour chercher une suite de caractères : "révolution française"	N.B. J'ai signalé dans quel paramètre les liens se trouvent, car cela varie d'une feuille de style à l'autre. Il faut donc les vérifier lorsque vous changez de feuille de style.
Page centrale Affichage facultatif du butineur de sections	Vous n'avez pas d'idée pour chercher ? Entrez dans la bibliothèque... Albums · Bandes-dessinées Adultes · Bandes-dessinées Jeunes · Discothèque · Documentaires · Réserve Revues · Romans Adultes · Romans Jeunes · Vidéothèque Mentions légales · Bib'Doc · Faire une recherche avec Google · pmb	**Navigation en pied de page** Autres liens utiles
Page centrale Variante (au lieu des sections) Affichage des catégories	Catégories **Art et culture** Architecture - Art du paysage - Bande dessinée - ... **Institutions et politique** Biographies - Essais politiques - Europe - ... **Jeunesse** Animaux - Art et culture - Corps et santé - ... **Sciences** Astronomie - Chimie - Mathématiques - ...	
Page centrale Autre possibilité Affichage des étagères virtuelles	Etagères virtuelles Nouveautés documentaires Afriques / Raymond Depardon Bébés à croquer / Saxton Freymann Les bibliothèques / Denis Pallier Des bibliothèques pleines de fantômes / Jacques Bonnet Au coeur de l'Afrique / Bernard Nantet Voir la suite...	

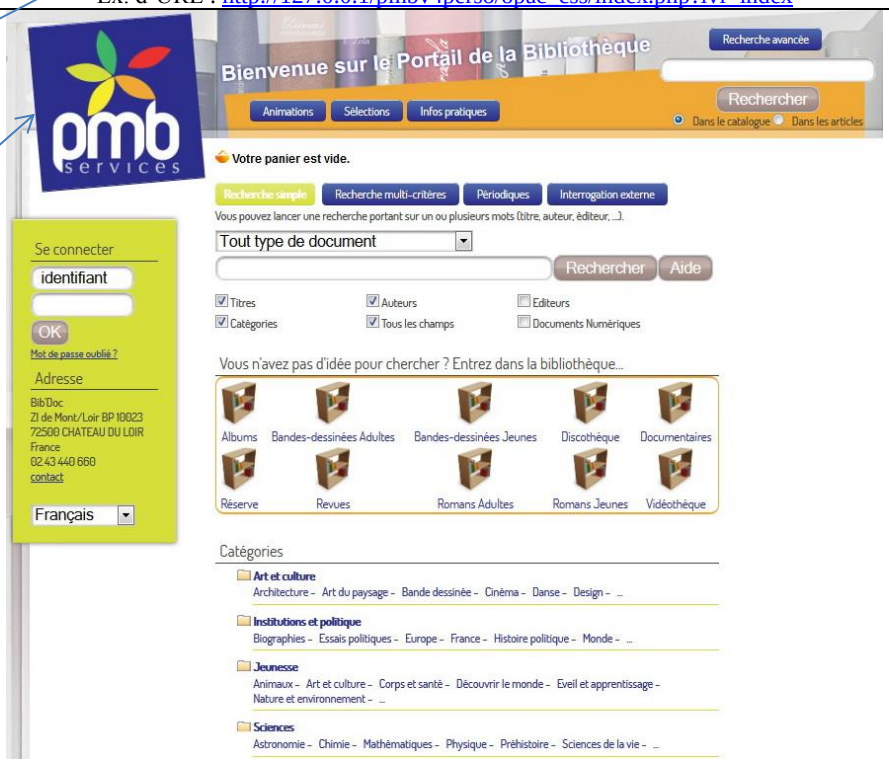
Attention ! Même si vous remplacez la page d'accueil d'un OPAC classique par une page d'accueil de type portail, vous garderez quand même ce genre de page OPAC, visible par exemple lors de la recherche avancée dans le catalogue. Bien sûr, la mise en page évoluera, surtout si vous changez de feuille de style, mais les pages habituelles de l'OPAC sont encore là : catalogue, recherche, liste de notices, affichage d'une notice ou d'un périodique, autorités, etc.
Vous le voyez clairement à la page suivante (l'exemple n° 3 montre la page Catalogue).

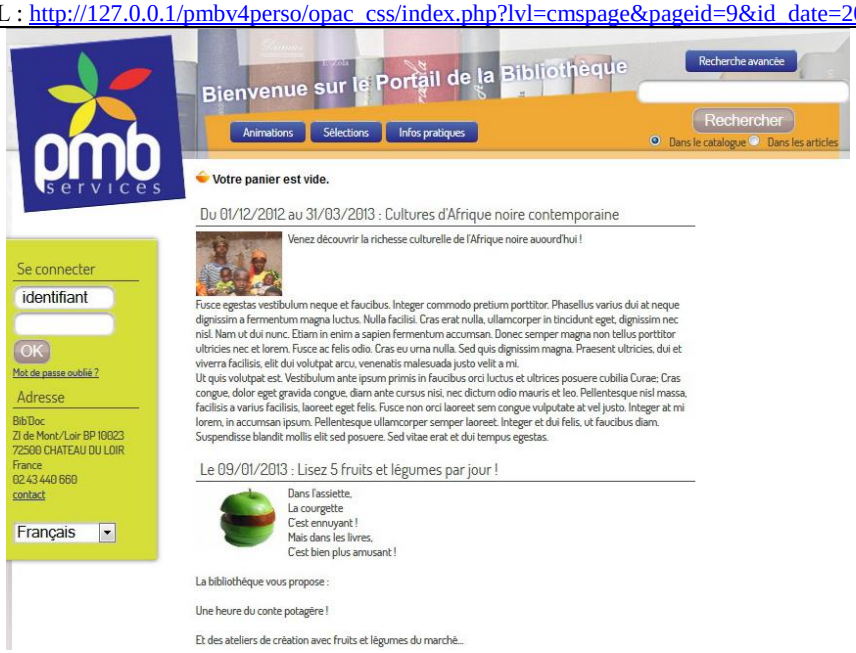
Vous continuerez donc à configurer l'OPAC comme d'habitude (Administration > Outils > Paramètres de l'OPAC), notamment tous les aspects liés à la recherche, les modules à activer, les infos à afficher, les facettes ...

Vous n'êtes pas obligé de remplacer la page d'accueil habituelle par une page du portail (comme dans l'exemple n°2), ou du moins pas tout de suite. Vous pouvez garder temporairement un OPAC classique et y intégrer progressivement quelques éléments du portail, par exemple les nouveautés (liste de notices en carrousel). Par contre, si vous voulez ajouter des rubriques et des articles, il faudra d'abord créer les pages du portail et configurer le contenu éditorial.

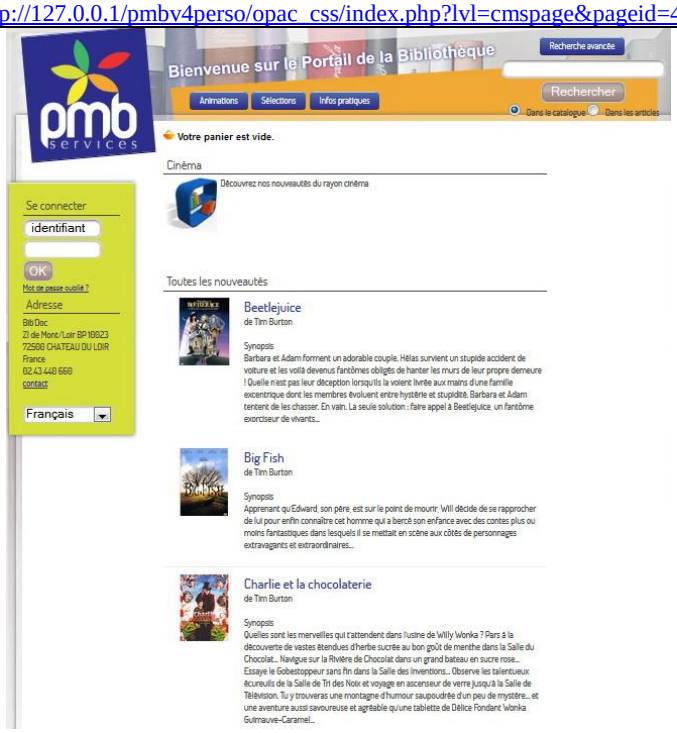
Un OPAC « style portail »

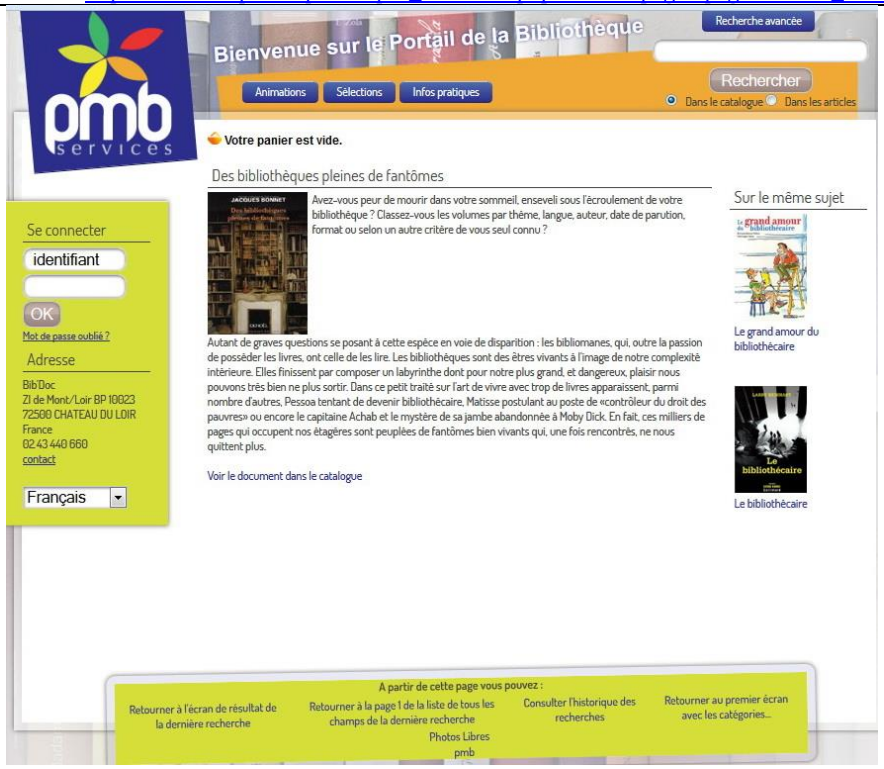
Exemple n° 2	Page d'accueil de « type portail »	Style pmb4
	Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php	
Introduction : logo, message et photo en arrière-plan. La zone supérieure contient aussi un menu, et la recherche (dans le catalogue ou les articles)		**Introduction** Lien vers le logo dans le paramètre biblio_preamble_p1 Message de bienvenue dans biblio_preamble_p2
Bandeau de gauche Connexion au compte du lecteur		**Page centrale** Agenda : calendrier d'activités régulières ou spéciales
Bandeau de gauche Adresse, téléphone, e-mail de contact … Choix de la langue		**Page centrale** Brèves : liste d'articles Infos en 1 ou 2 lignes maximum !
Page centrale Panier du lecteur Actualités : liste d'articles – carrousel en fondu enchaîné		**Bandeau de droite** Carrousel de nouveautés Créé avec le module Liste de notices !
Page centrale Nouveautés : liste d'articles liés à des étagères virtuelles		**Navigation en pied de page** Autres liens utiles

Exemple n° 3	Page classique de l'OPAC – affichée aussi dans le cas du portail	Style pmb4
	Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=index	
Page affichée par ex. - si on clique sur le **lien Retourner au 1er écran avec les catégories** - si on effectue une **recherche dans le catalogue** Cliquer sur le logo pour retourner à la page d'accueil du portail (c'est paramétré ainsi dans la démo).		**Introduction**
		Page centrale
		Panier du lecteur
		Module de recherche
Bandeau de gauche Connexion au compte du lecteur Adresse, téléphone et e-mail de contact Choix de la langue		Butineur de sections Catégories
Pied de page On ne le voit pas dans cette capture d'écran (la liste des catégories est fort longue), mais il n'a pas changé.		**Les modules affichés dépendent des paramètres OPAC**. Ex. activer la recherche simple ou multicritères, cocher des cases par défaut (titre, auteur …), afficher ou non le butineur de sections et la liste des catégories

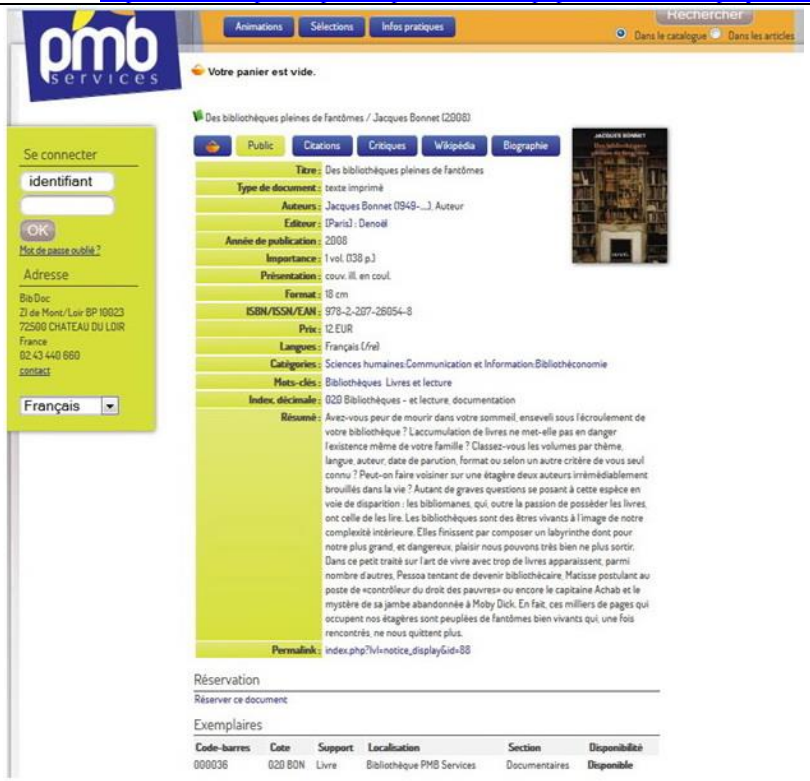
Exemple n° 4	Page qui affiche une liste d'événements	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9&id_date=2013-01-09		
Page du portail **Le critère de sélection est la date choisie dans l'agenda :** 09/01/2013 et on voit 2 articles. Pour d'autres dates, on ne verrait rien, si aucun article de type Evénement n'existe pour l'annoncer. Un **article de type Evénement** annonce une activité prévue (à une date précise ou sur une période déterminée).		Cette page contient un seul cadre, qui affiche une liste d'événements. **Ce cadre a été créé à partir du module Agenda !** **Le module agenda propose en effet 2 vues = 2 modes d'affichage** **calendrier** (ex. dans la page d'accueil) **liste d'événements** (ex. dans cette page)

Exemple n° 5	Page qui affiche une rubrique	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=4		
Page du portail **Le critère de sélection est la rubrique choisie, et non pas la date sélectionnée dans l'agenda.** L'affichage est dynamique, en fonction du contenu des deux rubriques : « Evénements » et « Heure du conte »	 	Les rubriques « Evénements » et « Heure du conte ». n'ont pas de sous-rubrique, on voit la liste des articles. **Cultures d'Afrique … Lisez 5 fruits …** **Ces 2 articles** qui étaient affichés sur base de la date 09/01/2013 (voir ci-dessus) **sont des articles de <u>type</u> Evénement.** **Mais un seul des 2 articles dépend de la <u>rubrique</u> Evénements !**
Page du portail L'affichage du contenu est dynamique, en fonction du contenu de la rubrique sélectionnée : « Animations »		« Animations » est une rubrique parente, on voit les 2 sous-rubriques qui en dépendent : « Evénements » et « Heure du conte ».

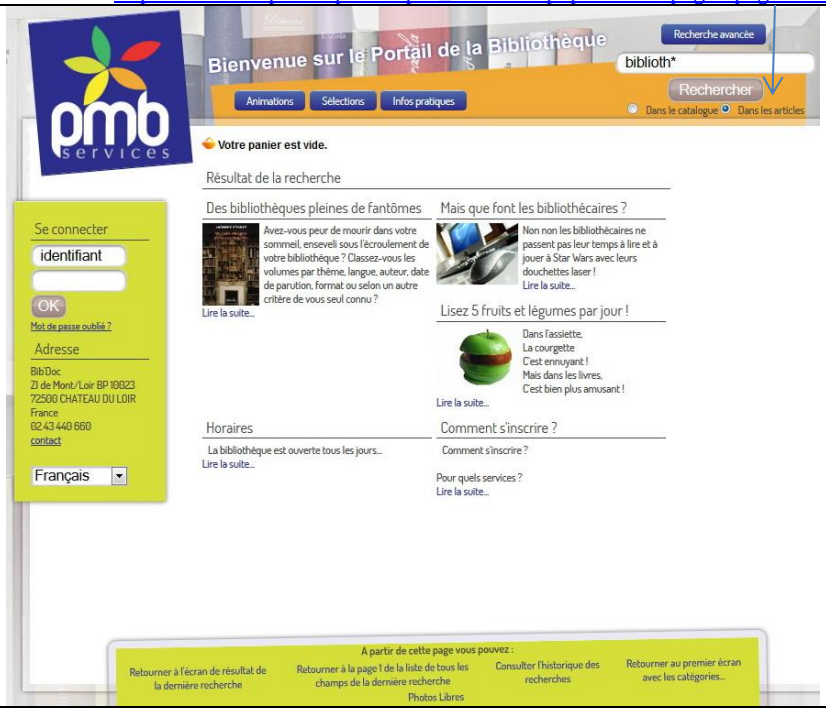
Exemple n° 6	Page qui affiche un article et une liste de notices	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=16		
Page du portail **Critère de sélection l'article** **Article de type « sélection » :** on veut lier l'article à plusieurs notices. Un des cadres de cette page affiche la **liste des notices contenues dans l'étagère liée à cet article.** L'affichage du contenu est dynamique, en fonction de l'article choisi, et des liens éventuels entre cet article et d'autres éléments de la base.		**Ce type de page permet d'enrichir le concept de l'étagère virtuelle.** On peut en effet expliquer dans l'article les spécificités de cette étagère, les liens éventuels avec une animation, etc. Le **template** employé pour cette **liste de notices** montre la vignette, et affiche le titre, l'auteur et le résumé de la notice. Ce template a été créé dans l'onglet Editions > Templates de notices.

Exemple n° 7	Page qui affiche un article et une liste de notices	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=17		
Page du portail **Critère de sélection l'article.** **Une variante de l'exemple n° 6.** **Article de type « coup de cœur » : on veut lier l'article à <u>une seule</u> notice.** Ce type d'article ne prévoit **pas de lien direct vers une étagère**. L'affichage du contenu est dynamique, en fonction de l'article choisi, et des liens éventuels entre cet article et d'autres éléments de la base (voir remarque sur le carrousel à droite).		**Page centrale** Un article « Coup de cœur » : des bibliothèques pleines de fantômes Article et lien vers la notice du document concerné (voir exemple n° 9 : page OPAC qui affiche une notice) **Carrousel** **Autres notices sur le même sujet** Il s'agit de **notices liées aux mêmes catégories que l'article sélectionné** (Bibliothèques …) N.B. Les catégories sont les descripteurs du thésaurus hiérarchisé

Exemple n° 8	Page qui affiche une notice – version « portail »	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=88		
L'affichage du contenu est dynamique, en fonction de la notice sélectionnée. **Dans ce cas, la notice est affichée dans une page du portail.** Dans la démo, on voit ce type de page du portail par ex. si on clique sur une vignette, dans le carrousel à droite.		Afficher une notice dans une page de type portail permet de modifier le mode de présentation de la notice grâce à un **template** adapté. On a employé ici le template par défaut : un nombre réduit de champs de la notice. Si on clique sur le permalink, on voit la notice et les infos de l'exemplaire dans l'OPAC, donc le même écran que l'exemple n° 9.

Exemple n° 9	Page qui affiche une notice – version « OPAC classique »	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=88		
L'affichage du contenu est dynamique, en fonction de la notice sélectionnée. **Dans ce cas-ci, la notice est affichée dans une page classique de l'OPAC.** Les infos disponibles peuvent donc varier, tout dépend du paramétrage de l'OPAC (onglet Administration).		**La page classique de l'OPAC suffit amplement pour afficher une notice.** Le recours à une page de type portail n'est pas obligatoire. Le choix entre l'une ou l'autre de ces options dépend du résultat final qu'on veut obtenir : quelles données afficher et comment ?

Exemple n° 10	Page qui affiche une liste de notices (résultat d'une recherche dans le catalogue)	Style pmb4
	Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=more_results	
Page de l'OPAC **Le résultat dépend bien sûr du critère de sélection : le(s) mot(s) recherché(s) dans le catalogue.** Vous voyez à nouveau que les pages habituelles de l'OPAC sont présentes, même dans un portail. On voit la page de l'OPAC qui affiche le **résultat d'une recherche dans le contenu bibliographique** : une liste de notices.		**A droite** Les infos disponibles peuvent varier, tout dépend du **paramétrage de l'OPAC (onglet Administration).** Ici par exemple, on a activé les **facettes**. Le lecteur peut affiner le résultat de sa recherche, et filtrer suivant divers critères.

Exemple n° 11	Page qui affiche une liste d'articles (résultat d'une recherche dans les articles)	Style pmb4
	Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=11	
Page du portail On voit ici la page du portail qui affiche le **résultat d'une recherche dans le contenu éditorial** : une liste d'articles. **Le résultat dépend** **du critère de sélection :** le(s) mot(s) recherché(s) dans le contenu éditorial **et aussi de ce qu'on a pu trouver !**		Ce cadre a été généré à partir du **module Liste d'articles**. Ce module propose plusieurs **sources de données** : dans ce cas-ci, on a choisi la **recherche dans les articles**. Il suffit de cliquer sur un de ces articles pour en lire le contenu complet.

Exemple n° 12	Page qui affiche un périodique	Style pmb4
Ex. d'URL : http://127.0.0.1/pmbv4perso/opac_css/index.php?search_type_asked=perio_a2z		
L'affichage du contenu est dynamique, en fonction du périodique sélectionné. **On voit cette page classique de l'OPAC si on a activé la navigation de A à Z dans les périodiques.**		Comme vous le voyez, **il n'est pas nécessaire de prévoir une page de type portail pour les périodiques,** la page « standard » de l'OPAC est déjà très esthétique ! La manière de présenter ce genre d'infos peut donc varier, tout dépend du paramétrage de l'OPAC (onglet Administration).

Préparez-vous quand même à un certain « choc psychologique »
au moment où vous quitterez le monde merveilleux de la démo du portail !

Repensez au préambule de mon document, à la comparaison avec l'imprimante 3D …

Lorsque vous commencerez à construire un portail dans une base standard de PMB, vous n'aurez pas tous ces éléments préfabriqués à votre disposition, il faudra par exemple :

- créer les pages du portail
- configurer l'aspect éditorial, créer des rubriques et des articles, réfléchir à l'arborescence
- insérer des modules et cadres dans les pages
- créer ou adapter des templates Django, afin de gérer la mise en forme

Si j'ai choisi de présenter tous ces aspects de la démo, ce n'est pas « pour vous mettre l'eau à la bouche » et déclencher ensuite un sentiment de frustration. Mon but est de vous rendre autonome, de vous permettre de construire vous-même un portail, même s'il est un peu moins beau que celui de la démo.

Or, je ne peux parvenir à cet objectif que si je montre un exemple, et que je vous explique comment il fonctionne.

CONTENU ÉDITORIAL

Configuration préliminaire

Il faut configurer quelques éléments fondamentaux, sinon vous ne pourrez pas gérer les rubriques ni les articles :

- **types de contenu pour les rubriques**
- **types de contenu pour les articles**
- **statuts de publication**

Ce paramétrage se fait dans l'onglet Administration > Contenu éditorial.	

En théorie, vous pouvez partir de zéro, et tout créer (statuts et types de contenu). Toutefois, si vous avez une version compatible de la démo, vous pouvez facilement récupérer sa configuration éditoriale de base, et l'étoffer ou la modifier ensuite, afin de l'adapter à vos besoins spécifiques. En outre, restaurer ce paramétrage vous permet d'employer directement les templates Django **de la démo et les** requêtes SQL **que j'ai mises au point. Je vous conseille de le faire, cela simplifie vraiment le travail.**

Pour savoir comment restaurer cette configuration initiale : Sauvegarder – restaurer la configuration minimum

Voir Notes sur la compatibilité entre les versions de PMB

Pourquoi faut-il créer des types de contenu et des statuts de publication ?
J'ai envie de répondre par l'absurde : s'ils n'existaient pas, il faudrait les inventer car ils sont indispensables ;-)

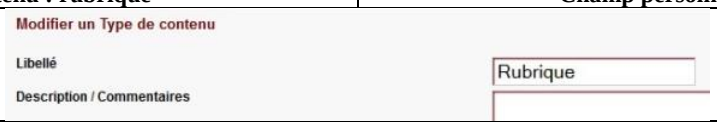
Non, ce n'est pas seulement une boutade, c'est tout à fait vrai. Deux exemples suffisent pour le démontrer :

- **Si ces types de contenu n'existaient pas**, le seul genre d'article possible serait un article « simple », « basique », que vous ne pourriez pas relier à la date d'un événement, ni à une notice ni à une étagère virtuelle. L'existence de champs personnalisés permet de connecter les articles à l'agenda, au contenu bibliographique, aux étagères …

- **S'il n'y avait pas de statuts de publication**, comment PMB pourrait-il établir une différence entre les articles prêts à être publiés en ligne et ceux qui sont encore à l'état de brouillon et doivent donc rester invisibles, ou entre les articles destinés à tous les lecteurs et ceux réservés aux abonnés ?

Parler des types de contenu pour les rubriques et articles est l'occasion d'expliquer (ou de rappeler) quelques aspects importants des champs personnalisés : la différence entre type de saisie et type de données, les 2 sens du mot « texte », et les règles à respecter pour choisir les noms de ces champs. Je vais aussi illustrer certains types de saisie fréquents.

TYPES DE CONTENU POUR LES RUBRIQUES

Dans la démo, un seul type de contenu pour les rubriques : une rubrique simple, qui ne contient aucun champ personnalisé.	
Type de contenu : rubrique	**Champ personnalisé : aucun**
	

TYPES DE CONTENU POUR LES ARTICLES

Dans la démo, on a défini 5 types de contenu pour les articles.

Libellé	Champs personnalisés	Commentaires
Coup de coeur	Identifiant de la notice (*Texte*) Editer la liste des champs	**Coup de cœur** pour un livre, un CD, un DVD … Il faut faire un lien avec l'id d'une notice
Evénement à une date précise	Date de l'événement (*Sélection d'une date*) Editer la liste des champs	**Evénement** qui se déroule **à une date précise** Une seule date suffit.
Evénement sur une période	Date de début (*Sélection d'une date*) Date de fin (*Sélection d'une date*) Editer la liste des champs	**Evénement** qui s'étale **sur une période précise** Il faut une date de début et une date de fin.
Information	Editer la liste des champs	**Information générale – Champ personnalisé inutile** Présentation de la bibliothèque, de l'institution, aide à la recherche, conditions d'inscription et de prêt, horaire …
Sélection	Etagère associée (*Liste de choix à partir d'une requête*) Editer la liste des champs	**Sélection** - Nouveautés Il faut faire un lien avec l'id d'une étagère.

Type de contenu : coup de cœur		Champ personnalisé : identifiant de la notice
Edition du champ identifiant_notice Nom du champ: identifiant_notice Titre à l'affichage: Identifiant de la notice Type de saisie: Texte Options Texte / Liste de choix / Liste de choix à partir d'une requête / Valeurs à partir d'un sélecteur d'autorités / Sélection d'une date / Zone de commentaire / Sélecteur externe / URL / Résolveur de lien / Zone de saisie HTML Annuler / Enregistrer	**Cliquer sur « Options » pour paramétrer le type de saisie.** Texte Taille du champ à l'écran: 10 Nombre maximum de caractères à saisir: 6 Répétable Html autorisé Enregistrer Ne pas oublier d'enregistrer ensuite ! **Voir page suivante comment connaître le numéro identifiant d'une notice.**	**Type de données** Edition du champ identifiant_notice Nom du champ: identifiant_notice Titre à l'affichage: Identifiant de la notice Type de saisie: Texte Options Type de données: Texte Texte / Texte large / Entier / Date / Nombre à virgule recherche dans ce champ 100
Le type de saisie « texte » signifie « caractères saisis au clavier ». **Plusieurs types de données sont donc possibles.**		**Le type de données « texte »** peut contenir tout type de caractère : lettres, chiffres ou divers symboles (mathématiques, de ponctuation)
N.B. Dans la démo, on a choisi « texte » comme type de données, mais vu qu'il s'agit d'un numéro d'id, un entier aurait aussi pu convenir.		

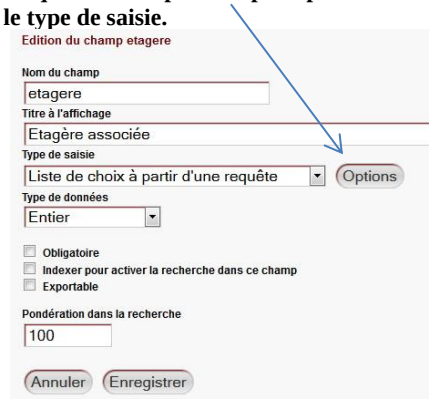
Type de contenu : événement à une date précise	Champ personnalisé : date
Edition du champ date Nom du champ: date Titre à l'affichage: Date de l'événement Type de saisie: Sélection d'une date Options Type de données: Date Obligatoire Indexer pour activer la recherche dans ce champ Exportable Pondération dans la recherche: 100 Cliquer sur « Options » pour paramétrer le type de saisie.	Sélection d'une date Ne pas utiliser la date du jour par défaut Enregistrer **Il faut cocher la case « ne pas employer la date du jour par défaut ».** Ne pas oublier d'enregistrer ensuite ! **Si le champ date ne contient aucune valeur (= le message « aucune date ») au moment de cliquer sur « enregistrer »**, parce que vous venez de créer un nouvel article ou d'effacer la valeur précédente dans un article préexistant, **voici ce que vous verrez au moment où vous éditerez à nouveau cet article** : • **si vous n'aviez pas coché cette case, le champ date aura changé de contenu, il contiendra automatiquement la date du jour**. Si vous êtes distrait, et que vous ne le remarquez pas, la date de l'événement sera fausse ! • **si vous aviez coché cette case, le champ date restera dans l'état où il était antérieurement**. C'est nettement mieux !
Soyons clair. Que signifie exactement l'autre choix possible : ne pas cocher la case « ne pas employer la date du jour par défaut » ? **N'oubliez pas que ces 2 négations s'annulent, donc cela veut dire « employer la date du jour par défaut » !**	
Le type de saisie « date » signifie « sélection grâce au calendrier ». **Un seul type de données est donc possible : une date.**	**Le type de données « date »** ne peut contenir qu'une date « SQL » : année - mois – jour. Ex. 2015-04-01 = 1er avril 2015 *Non, ce n'est pas un poisson d'avril !*

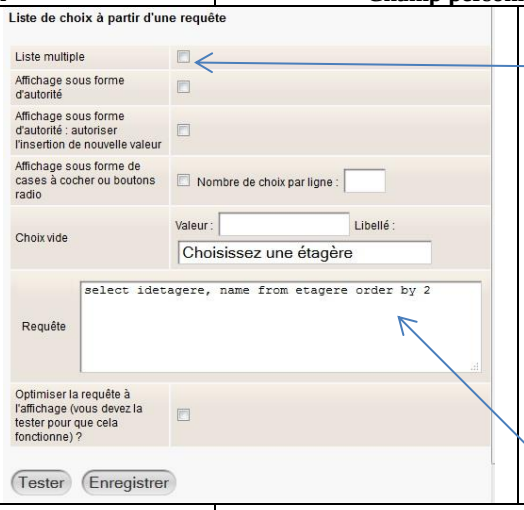
Type de contenu : événement à une période précise	2 champs personnalisés : date_debut et date_fin
Même remarque que pour le champ date : il faut cocher la case « ne pas employer la date du jour par défaut ».	

Type de contenu : information	Champ personnalisé : aucun
Information générale (présentation de la bibliothèque, de l'institution, aide à la recherche, conditions d'inscription et de prêt …) – Champ personnalisé inutile	

Type de contenu : sélection	Champ personnalisé : étagère associée

Cliquer sur « Options » pour paramétrer le type de saisie.





Si on coche la case « **liste multiple** », cela permet de sélectionner plusieurs étagères.

Mais dans ce cas-ci (sélection par exemple de nouveautés), c'est en général peu souhaitable au niveau de l'affichage.
En effet, cela risque souvent de faire beaucoup trop de notices au total (si les paniers qui alimentent ces étagères sont bien remplis).

Cette requête affiche la liste des étagères, triées suivant le nom donné à l'étagère.

Le type de saisie est ici « liste de choix à partir d'une requête ».	**Le type de données est un « entier »** car la requête renvoie le numéro identifiant de l'étagère sélectionnée, donc un nombre entier.

Attention lorsque vous créez un champ personnalisé.
Le choix du nom des variables, des champs personnalisés et des répertoires obéit à des règles précises – sinon, erreur SQL !

- un nom en un mot (pas d'espace !), qui comporte uniquement des caractères alphanumériques non accentués (donc rien d'autre que des lettres ou des chiffres), et qui doit commencer par au moins une lettre - de préférence, entièrement en minuscules
- aucun symbole mathématique, signe de ponctuation ou autre symbole n'est toléré
- une seule exception : le trait de soulignement _ (underscore) afin de marquer la séparation en 2 mots (date_debut, date_fin …)
- ne confondez pas le trait de soulignement et le tiret (identique au symbole mathématique « moins », c'est pourquoi il est interdit)

Les noms des champs personnalisés liés aux types de contenu doivent être uniques dans la base !
Vous ne pouvez pas employer deux fois le même nom, même si un de ces champs concerne les rubriques, et l'autre les articles.
Si vous vous trompez malgré tout, pas de souci, vous verrez un message d'erreur, et l'enregistrement sera impossible.

Comparaison entre 2 types de contenu (pour les articles) et 2 types de saisie (champs personnalisés)

Libellé	Champs personnalisés	Commentaires
Coup de coeur	Identifiant de la notice (*Texte*) Editer la liste des champs	**Coup de cœur** pour un livre, un CD, un DVD … Lien avec l'identifiant d'une notice **Type de saisie : texte (valeur saisie au clavier)**
Sélection	Etagère associée (*Liste de choix à partir d'une requête*) Editer la liste des champs	**Sélection** (nouveautés) Lien avec l'identifiant d'une étagère **Type de saisie : liste de choix à partir d'une requête**

Pourquoi saisir l'id de la notice au clavier, alors qu'on affiche la liste des étagères et qu'on sélectionne ainsi le bon numéro d'id ?
La situation de départ est différente. La « situation de départ », c'est l'état dans lequel vous êtes au moment de rédiger un article.

Pour un article de type « coup de cœur pour une notice » ou d'autres variantes (coup de cœur pour un auteur, une série …)
Pour le numéro identifiant de la notice (ce serait la même chose pour l'id d'un auteur, d'un éditeur, d'une série, etc)

- A priori, vous savez dès le départ de quel livre, de quel auteur ou de quelle série vous voulez parler.
- Vous pourriez certes créer une requête qui afficherait tous les titres, tous les auteurs ou toutes les séries, mais cela vous obligerait à parcourir une liste fort longue qui contiendrait peut-être des doublons.
- Dans ce cas, être sûr de choisir le numéro identifiant de la « bonne » notice ou du « bon » éditeur est nettement plus compliqué.

Pour un article de type « sélection » (nouveautés – fiction, nouveautés - documentaires …)
Pour le numéro identifiant d'une étagère

- Vous ne savez pas toujours dès le départ dans quelle étagère les notices seront, ni à quel(s) panier(s) l'étagère sera liée.
- En outre, il se peut même que l'étagère ou le panier dont vous avez besoin n'existent pas encore.
- Il est donc plus simple de prévoir un affichage de la liste des étagères disponibles et une sélection manuelle de la bonne étagère.
- Ce système est beaucoup plus flexible.

N.B. Je traite ce sujet de manière plus approfondie dans le chapitre suivant : Numéros identifiants des notices, étagères

STATUTS DE PUBLICATION

Le but des statuts de publication est de préciser, pour les rubriques et les articles,

- **s'ils peuvent être publiés**
  - le cas échéant, on peut éventuellement en restreindre l'accès aux lecteurs authentifiés (les « abonnés », ceux qui ont saisi un login et un mot de passe corrects)
- **s'ils ne peuvent pas, ne peuvent pas encore ou ne peuvent plus être publiés**
  - le cas échéant, on peut préciser les raisons de cette non-diffusion, afin de faciliter la gestion

Rappel - En théorie, vous pouvez partir de zéro, et tout créer (statuts et types de contenu). Toutefois, si vous avez une version compatible de la démo, vous pouvez facilement récupérer sa configuration éditoriale de base, et l'étoffer ou la modifier ensuite, afin de l'adapter à vos besoins spécifiques. En outre, restaurer ce paramétrage vous permet d'employer directement les templates Django **de la démo et les** requêtes SQL **que j'ai mises au point. Je vous conseille de le faire, cela simplifie vraiment le travail.**

Pour savoir comment restaurer cette configuration initiale : Sauvegarder – restaurer la configuration minimum

Voir Notes sur la compatibilité entre les versions de PMB

La démo contient seulement 2 statuts de publication : brouillon et publié, et tous les articles ont le statut « publié ».

Libellé	Visible à l'OPAC	Visible à l'OPAC uniquement pour les abonnés
Brouillon		
Publié	X	

Ajouter un statut

N.B. Pour la clarté, il vaut mieux modifier le libellé du 2e statut de la démo : « Publié » devient « Publié – Pour tous ».

Invisible à l'OPAC

Visible par tous les lecteurs

Voici un exemple un peu plus complexe de statuts, afin de nuancer davantage

Libellé	Visible à l'OPAC	Visible à l'OPAC uniquement pour les abonnés
A valider		
Archives		
Brouillon		
Publié - Abonnés		X
Publié - Pour tous	X	

Vous pouvez créer les 3 statuts supplémentaires : à valider, archives et « publié – abonnés » (et éventuellement d'autres statuts).

3 statuts : rubriques et articles invisibles à l'OPAC

- **A valider** : article en théorie prêt pour la publication, mais qui doit encore être validé par celui qui joue le rôle de rédacteur en chef
  - Bien entendu, cela n'a de sens que si plusieurs personnes rédigent. Si vous êtes seul, vous êtes polyvalent !
- **Archives** : des rubriques ou articles plus anciens, mais qu'on pourrait « recycler » et adapter (ex. activités organisées 2 fois par an)
  - Cela évite de devoir tout recommencer pour certaines animations récurrentes. Pensez quand même à mettre à jour !
  - Cela peut aussi être utile pour certains articles de type bibliographique (présentation d'un auteur, d'un thème).
- **Brouillon** : article ou rubrique encore en chantier
  - Quel que soit l'état d'avancement des travaux …

2 statuts : rubriques et articles visibles à l'OPAC

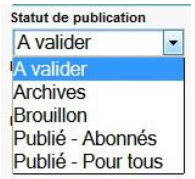
- **Publié – Abonnés**
  - Seuls les lecteurs qui se sont identifiés peuvent les voir
- **Publié – Pour tous**
  - Tout le monde y a accès

N.B. Ordre alphabétique basé sur les codes ASCII
Le statut « A valider » précède les « Archives » à cause de l'espace intermédiaire entre les 2 mots.
Le code ASCII de l'espace est 32, les chiffres démarrent à 48 (code du zéro), les lettres majuscules commencent à 65 (code du « A »), les minuscules à 97 (code du « a »).

Comme vous le constatez, ces statuts sont affichés par ordre alphabétique.
Dans l'onglet Administration > Contenu éditorial, cela ne pose pas vraiment problème.

Toutefois, c'est nettement plus gênant lorsqu'on crée un nouvel article ou une nouvelle rubrique.
Un moment de distraction ou de fatigue, et on se demande pourquoi un article n'apparaît pas ...
En fait, si on oublie de sélectionner un statut avant d'enregistrer, c'est le 1er de la liste qui est choisi par défaut, donc dans ce cas-ci, c'est le statut « à valider », ce qui rend forcément l'article invisible !

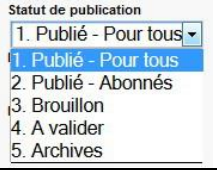
Le problème se corse encore lorsque vous êtes en pleine phase d'élaboration du portail.
En général, on préfère que tous les articles soient visibles par défaut, surtout au début des tests.
Or, le statut « Publié – Visible par tous » est ici le dernier de la liste triée par ordre alphabétique !



La solution est de numéroter les statuts de la liste, en fonction de l'ordre dans lequel vous voulez les voir.

Vous pourriez même prévoir :

- **un ordre d'affichage pour la phase de test, de construction** : mettre en 1er lieu les articles à publier, pour voir tout de suite l'impact de votre travail.
- **un autre ordre d'affichage dès que vous passez en production** : mettre en 1er lieu les articles à ne pas diffuser, afin d'éviter de publier par inadvertance un brouillon ou un article qui n'a pas encore été validé.



Le statut « non publié » d'une rubrique ou d'un article rend forcément invisible la rubrique ou l'article en question.

Si une rubrique a le statut « non publié », on ne la voit dans aucun des écrans où elle est censée apparaître.
Si un article a le statut « non publié », on ne le voit dans aucun des écrans où il est censé apparaître.

Le statut « non publié » d'une rubrique rend invisibles les articles ou sous-rubriques qui en dépendent.

La rubrique « Sélections » contient 2 sous-rubriques : Nouveautés et Coups de cœur

- Si on modifie le statut de la **rubrique « Sélections »**, et qu'elle devient un « brouillon », on ne la verra plus, et les sous-rubriques (Nouveautés et Coups de cœur) et les articles qui en dépendent seront invisibles. L'onglet Sélections disparaît du menu, le carrousel d'articles (en haut à gauche) perd d'un coup plusieurs articles, et la liste de Nouveautés (en bas à gauche) devient vide.
- Si on modifie le statut de la **rubrique « Coups de cœur »**, et qu'elle devient un « brouillon », on ne la verra plus apparaître comme sous-rubrique de « Sélections », et forcément, les articles qui en dépendent seront eux aussi invisibles.

Rubrique Sélections Visible	Rubrique Sélections Invisible	Rubrique « Coups de cœur » invisible Article « Bibliothèques » visible	Rubrique « Coups de cœur » visible Article « Bibliothèques » invisible
		Impossible via le menu principal d'accéder aux Coups de cœur donc impossible de voir la liste des articles qui sont dans cette rubrique. Les fantômes se cachent !	
On voit 3 onglets dans le menu, 4 articles dans le carrousel et les nouveautés	2 onglets dans le menu, moins d'articles dans le carrousel et plus aucune liste de nouveautés	Il y a aussi un impact sur le carrousel Les articles « Coups de cœur » ont disparu, il ne reste que ceux des rubriques Evénements et Heure du conte.	On ne voit plus que 3 articles dans cette rubrique, plus de trace des fantômes ! Ils ont aussi disparu du carrousel.

Le carrousel d'articles est configuré afin d'afficher au maximum 4 articles, choisis dans 3 rubriques : Evénements, Heure du conte et Coups de cœur.
Le nombre total d'articles de ces 3 rubriques est plus élevé, donc augmentez éventuellement ce nombre maximum (4) pour tester l'impact des statuts.

Ce n'est pas parce qu'une rubrique ou un article ont le statut « Publié – Visible par tous » qu'on les voit automatiquement !

Il faut prévoir un **lien vers** un article ou une rubrique afin de les afficher, par exemple

- mettre l'article dans une rubrique parente qui est visible
- mettre l'article dans une liste visible d'articles **ou** un carrousel visible d'articles
- mettre une sous-rubrique dans une rubrique parente visible **et** mettre le parent dans un menu ou dans une liste visible de rubriques

Le meilleur système : regrouper les articles qui doivent être accessibles dans des rubriques visibles qui font partie du menu principal

La rubrique « Coups de cœur » et l'article « Des bibliothèques pleines de fantômes »

Cet article peut être affiché pour plusieurs raisons.

- On le voit dans la rubrique « Coups de cœur », qui est accessible via le menu
- On peut aussi le voir dans le carrousel d'articles présentés en fondu enchaîné (dans le coin supérieur gauche de la page centrale).
- En outre, il peut aussi apparaître suite à une recherche dans le contenu éditorial.

Ce n'est pas parce qu'une rubrique ou un article ont le statut « Publié – Pour les abonnés » qu'on les voit nécessairement !

Avant de sombrer dans le désespoir parce que vous ne comprenez pas pourquoi un article reste obstinément invisible malgré tous vos efforts, **posez-vous à vous-même – ou posez à vos lecteurs - une question existentielle** :

« N'aurais-je pas par hasard / N'auriez-vous pas par hasard oublié de saisir le login et le mot de passe dans l'OPAC ? »

Numéros identifiants

Si vous voulez créer un lien entre d'une part un article, et d'autre part une notice, un auteur, une étagère …, il faut

- **donner le numéro identifiant de l'élément bibliographique concerné**
- **ou créer une requête qui vous permet de choisir un élément dans une liste et d'obtenir ainsi son numéro d'id**

Le choix entre ces deux méthodes dépend de l'objectif visé, et du type d'élément bibliographique concerné.

Connaître dès le départ le numéro d'id d'une notice, d'un auteur, d'une série … vous fera en général gagner du temps.

D'abord un peu de « technique » : comment connaître facilement le numéro identifiant d'une notice ?

En gestion - Sans afficher la notice

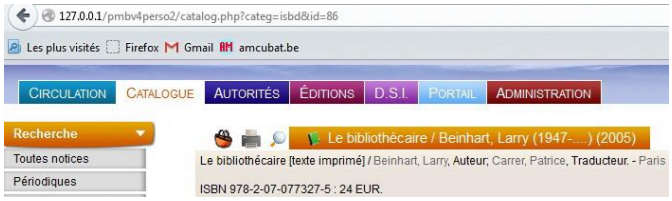
Regarder en bas à gauche
lorsque la souris survole la notice abrégée

Même sans afficher la notice, même sans cliquer, **au simple survol de la souris sur la notice abrégée**, vous verrez apparaître le **lien dans le coin inférieur gauche** de votre écran.
Le dernier paramètre de l'URL indique l'id de la notice (ici 86).

En gestion - En affichant la notice

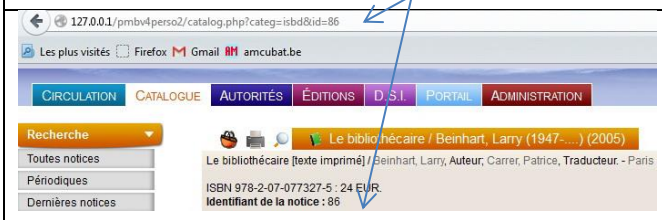
Regarder en haut dans la barre d'adresses
et/ou dans le descriptif de la notice

On voit le numéro identifiant dans la barre d'adresses.



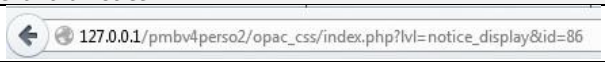
On peut aussi avoir un affichage systématique du n° d'id dans le descriptif de la notice
Administration > Outils > Paramètres généraux
show_notice_id = 1

Et finalement, on peut même combiner les deux !



Dans l'OPAC – En affichant la notice

Comme en gestion, le numéro identifiant est visible dans la barre d'adresses, lorsqu'on affiche la notice dans le catalogue.



En outre, si vous avez activé l'affichage du permalien, ce numéro d'id de la notice est visible lorsque vous dépliez la notice.
Administration > Outils > Paramètres de l'OPAC : permalink = 1

Il n'y a pas que les notices, on peut tout aussi rapidement connaître le numéro identifiant d'un auteur, d'une catégorie, etc.

Le dernier paramètre d'une l'URL composée (= avec variables) donne le numéro d'id du principal élément concerné.
Regardez en haut dans la barre d'adresses, vous verrez divers types de numéros identifiants (collection, éditeur …)
On voit parfois aussi ces numéros dans le coin inférieur gauche de l'écran (au survol de la souris).

Exemples en gestion	**Collection** http://127.0.0.1/pmbv4perso2/autorites.php?categ=collections&sub=collection_form&id=17 **Editeur** http://127.0.0.1/pmbv4perso2/autorites.php?categ=editeurs&sub=editeur_form&id=13
Exemples dans l'OPAC	**Auteur** http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=author_see&id=36 **Catégorie** http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=categ_see&id=1433

Connaître le numéro d'id d'une étagère est également utile, mais en général on procède autrement que pour les notices. On vérifie rarement le n° d'id dans la barre d'adresses, on préfère avoir recours à une requête SQL. Pourquoi ?

D'abord un peu de « technique » : comment connaître facilement le numéro identifiant d'une étagère ?
Le numéro d'id d'une étagère est aussi visible dans la barre d'adresses (à la fin de l'URL) http://127.0.0.1/pmbv4perso2/catalog.php?categ=etagere&sub=edit_etagere&action=edit_etagere&idetagere=3 La plupart du temps, on emploie une requête SQL qui affiche la liste des étagères, triées par nom, et qui permet de récupérer ainsi le n° d'id. SELECT idetagere, name FROM etagere ORDER BY 2

N.B. Ceux qui parcourent ce tutorial de manière linéaire auront l'impression (tout à fait justifiée !) d'avoir déjà lu « quelque chose à ce sujet quelque part ». Désolée si je me répète à intervalle aussi rapproché, mais il me semble nécessaire de reprendre et de compléter ce thème.

Je tiens à donner ici une petite synthèse à propos des numéros identifiants des notices, auteurs, séries, étagères, etc. C'est important avant d'aborder le thème de la rédaction des articles, et de leurs liens avec le contenu bibliographique.

1) Dans certains cas, le lien entre un article et un élément bibliographique est « fixe »

Si vous voulez rédiger un article sur un livre, un auteur, une série …, le lien est « fixe » car il ne changera pas.
Vous avez choisi le livre, le DVD, l'auteur, la série, la collection … que vous vouliez présenter, donc **ce lien correspond à un numéro identifiant et à un seul, qui est entièrement prévisible**.

Certes, vous pourriez employer une requête qui afficherait la liste de tous les livres ou auteurs, de toutes les séries ou collections, mais ce ne serait vraiment pas pratique.

- En premier lieu, cela vous obligerait à parcourir une liste fort longue qui contiendrait probablement des homonymes.
- Des homonymes « normaux » (le même titre peut correspondre à 2 œuvres d'auteurs distincts, ou bien à plusieurs éditions différentes du même livre), et des homonymes « regrettables » (des doublons liés à des erreurs de saisie ou d'import des données).
- En outre, si vous prenez par malheur l'édition la plus ancienne d'un livre, vous n'aurez probablement pas la possibilité d'afficher aisément la vignette de couverture.

Le mieux est donc de saisir le numéro identifiant au clavier, ainsi vous êtes sûr d'avoir choisi le numéro d'id de la « bonne » notice, du « bon » auteur, de la « bonne » série, etc.
Mais le numéro d'id d'une notice ou d'un auteur dans la base de test n'est pas le même que dans votre vraie base!
Donc il faudra le changer plus tard, après avoir transféré le portail depuis la base de travail vers la base en ligne.
Conseils pratiques à ce propos : voir Gestion globale du contenu éditorial

2) Dans d'autres cas, le lien entre un article et un élément bibliographique est « variable »

Chaque étagère virtuelle du groupe « nouveautés » va probablement être remplacée par un article lié à une étagère de notices. Dans cet article, vous décrirez les spécificités de l'étagère, le point commun entre ces notices. Le même type d'article (lié à une étagère) est aussi envisageable pour présenter certaines activités, comme une exposition thématique.

Le lien est variable : tout est susceptible de changer, aussi bien le lien vers l'étagère que le panier qui l'alimentera et son contenu. Fixer le numéro d'id de l'étagère dès le départ (comme pour les notices) ne serait d'ailleurs pas toujours possible, certains éléments sont parfois imprévisibles au début.

- Au moment où vous rédigez un article « standard » qui présente les nouveautés du rayon jeunesse, vous ne pensez pas à une série de notices en particulier. On peut espérer que votre budget vous permettra quand même d'envisager quelques acquisitions de temps en temps, donc le contenu de l'étagère « Nouveautés – Jeunesse » va évoluer (→ changement dans les paniers).
- Au moment où vous rédigez un article qui présente une exposition thématique, vous pensez probablement à une série de notices en particulier, mais cette liste n'est peut-être pas encore complète (ou bien, elle est trop longue !). D'ailleurs, rien ne garantit que l'étagère et le panier dont vous avez besoin existent déjà, vous ne les avez peut-être pas encore créés.

Certes, vous pourriez saisir le numéro d'id de l'étagère au clavier, mais ce ne serait vraiment pas pratique.

- Il faudrait chercher ailleurs (onglet Catalogue) pour voir la liste des étagères, ce qui est en soi une perte de temps.

Il vaut mieux employer une requête qui affiche la liste des étagères disponibles et permet une sélection manuelle.

- Heureusement, cette liste ne risque pas d'être aussi longue que celle des notices, des auteurs ou des éditeurs.
- Les risques d'homonymie sont en théorie (quasi) nuls … à moins que vous n'ayez mal géré les noms donnés aux étagères !

Ce système plus flexible convient mieux pour les étagères, et il rend le fonctionnement du portail plus dynamique.

Rédiger les rubriques et articles

GÉNÉRALITÉS

Les articles ne sont pas « isolés », il y a toute une arborescence qui comprend :

- **des rubriques parentes et des sous-rubriques**
- **des articles**

Il est impossible de créer et d'enregistrer un article si vous n'avez pas prévu au moins une rubrique pour le mettre. C'est pourquoi la création des rubriques précède forcément la rédaction des articles. La première rubrique que vous créez a d'office pour parent la « racine » de l'arbre du contenu éditorial.

Cette arborescence n'est pas fixée une fois pour toutes : vous pouvez plus tard **déplacer** facilement des rubriques et/ou des articles (en changeant leur rubrique parente ou par un système tout simple de glisser-déposer), **ajouter** des rubriques ou des articles, ou bien en **supprimer**. La suppression d'une (sous-)rubrique n'est bien sûr possible que si celle-ci est vide.

Soyez logique, cohérent et n'exigez pas trop de vous-même !

- **l'arborescence et les titres des rubriques doivent être conçus pour faciliter la navigation dans le menu du portail, afin que l'utilisateur puisse trouver rapidement l'information recherchée**
- **l'arborescence et les titres des rubriques doivent également être conçus pour faciliter le travail de celui ou celle qui va gérer globalement le portail ! Pensez aussi à vous, inutile de vous compliquer l'existence …**

Le menu de la version de démo du portail

L'arborescence des rubriques, sous-rubriques et articles : l'arbre du contenu éditorial

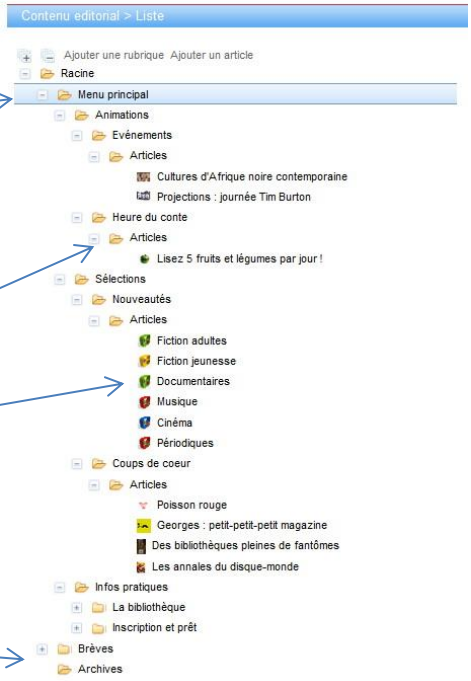
Les titres affichés dans le menu (visible en haut de la page) sont des sous-rubriques qui dépendent de la rubrique « menu principal » : animations, sélections et infos pratiques.

J'ai déplié presque toutes les rubriques du menu principal, afin que vous sachiez où se trouve tel ou tel article.

Dès qu'on ajoute le premier article dans une rubrique, un **dossier « Articles »** se crée automatiquement.

Si vous avez prévu un **logo pour un article ou une rubrique**, on le voit aussi.
Ici, les logos sont des étagères. S'il n'y a pas de logo, des **icônes** appropriées symbolisent les rubriques et articles.

Je n'ai pas déplié les « Brèves », mais le signe + (à gauche du dossier) indique que cette rubrique a un contenu.
Par contre, la rubrique « Archives » est vide pour le moment.



La répartition en rubriques est aussi une question de logique, de cohérence.

Les 3 rubriques principales ont chacune un objectif spécifique :

- animations, activités organisées par la bibliothèque
- sélection, mise en évidence des nouveautés, des coups de cœur
- informations générales sur la bibliothèque ou infos pratiques

Le nombre de rubriques principales visibles en haut dans le menu sera forcément limité, vu la largeur disponible à l'écran, il faut donc bien structurer et hiérarchiser l'ensemble.

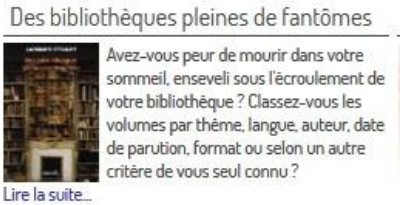
Si vous comptez **réserver certaines rubriques ou certains articles aux abonnés** (les lecteurs qui se sont identifiés dans l'OPAC), il est souvent préférable de prévoir une rubrique principale spécifique. Cela dépend aussi du nombre d'articles concernés. S'il y en a peu, autant les répartir aux endroits les plus adéquats.

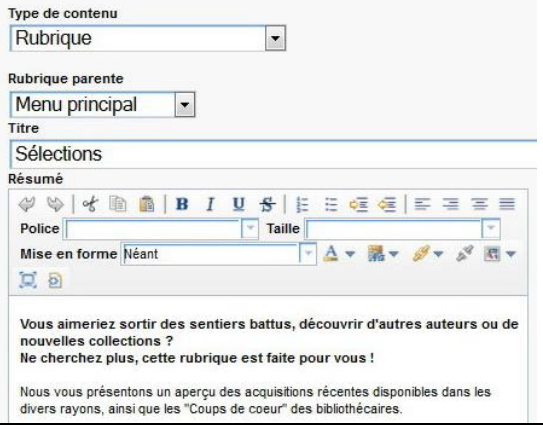
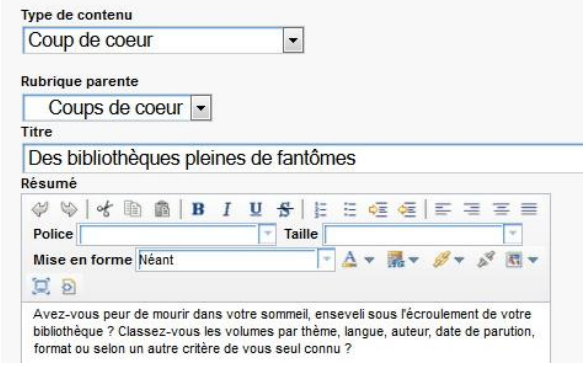
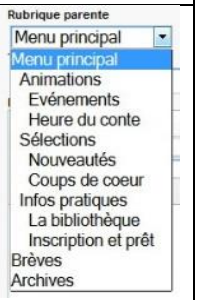
Dans la démo, on a créé un menu (en haut de la page) en se basant sur une rubrique parente (le « menu principal »), et en calquant l'ordre d'affichage des rubriques et des articles sur leur place dans l'arbre du contenu éditorial. il y a d'autres manières de procéder pour créer un menu, mais cette méthode est sûrement la plus simple à mettre en œuvre. Pas de complications inutiles au début …

AJOUTER UNE RUBRIQUE OU UN ARTICLE

N.B. Vu qu'il y a peu de différences entre la création des rubriques et celle des articles, et que leurs champs respectifs sont à peu près identiques, je vais traiter les deux aspects ensemble, et signaler au passage quelques divergences.

Situation initiale	
	Rappel : vous devez avoir défini d'abord des types de contenu (pour les rubriques et articles) **et des statuts de publication, sinon la création des rubriques et articles est impossible.** Voir configuration préliminaire Au départ, il n'y a aucune rubrique ni aucun article à la racine de l'arbre éditorial. **Etape indispensable : créer au moins une rubrique, afin de pouvoir créer ensuite un article.**

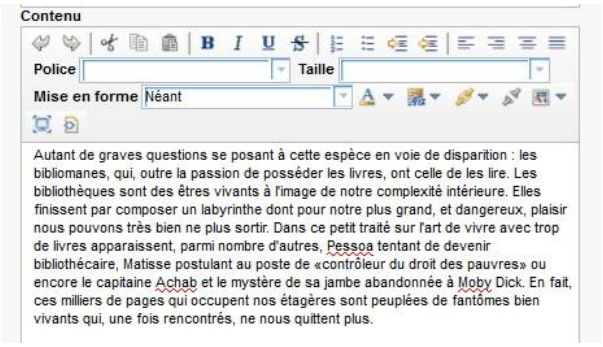
Les deux exemples qui servent de support pour ces explications : une rubrique et un article	
	La **rubrique « Sélections »** comprend 2 sous-rubriques : « Nouveautés » et « Coups de cœur » **L'article « Des bibliothèques pleines de fantômes »** se trouve dans la sous-rubrique « Coups de cœur »
 N.B. Ce texte d'introduction à la rubrique « Sélections » ne figure pas dans la démo, je l'ai ajouté.	 N.B. **Ceci n'est pas la totalité de l'article**, c'est en réalité le **« chapeau » de l'article** : le 1er paragraphe destiné à accrocher le lecteur, et à lui donner **envie de cliquer pour « lire la suite ».**

Type de contenu, rubrique parente, titre et résumé	
Champs des rubriques	**Champs des articles**
	 Rappel : un article de type « Coups de cœur » a un champ qui contient un numéro d'id de notice, puisque son but est de présenter un livre, un DVD …
Rubrique parente et résumé	
Rubrique parente **Une rubrique peut devenir une rubrique parente, située à la racine de l'arbre.** Donc le choix « à la racine » est proposé. **Résumé** C'est un petit texte d'introduction, qui décrit les spécificités de la rubrique. **Aucune suite n'est prévue pour ce résumé de rubrique.**	**Rubrique parente** **Un article ne peut pas se situer à la racine,** il **doit avoir une rubrique parente.** Donc le choix « à la racine » n'est pas proposé. **Résumé** C'est aussi un petit texte d'introduction, mais **le chapeau de l'article incite le lecteur à cliquer pour lire la suite : l'article lui-même.**
	
N'oubliez pas de préciser le type de contenu des rubriques ou articles que vous allez créer ! **Pour l'équilibre visuel et la clarté des menus et listes, veillez à ce que les titres des rubriques soient courts et significatifs.**	

Champ « contenu » d'un article - Ce champ n'existe pas pour les rubriques

N.B. Le genre d'article peut varier en fonction du type de contenu, dans ce cas-ci le but est de présenter une notice.
Ce n'est donc qu'un exemple de contenu parmi d'autres, d'autres articles présentent des événements, des sélections, etc.

Seuls les articles ont un champ « contenu », c'est le texte de l'article lui-même.



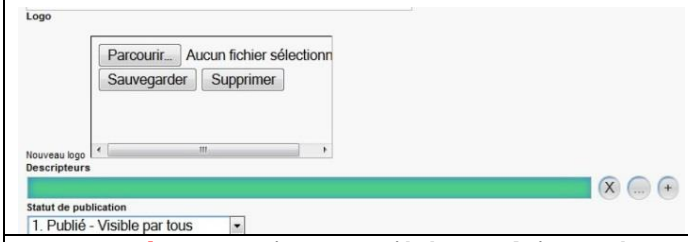
Lorsqu'on clique sur le lien « Lire la suite », on voit la totalité de l'article : le résumé (chapeau), le contenu et le logo éventuel.

Le lien (« voir le document dans le catalogue ») vous mène vers la page OPAC qui affiche une notice.

N.B. Je n'ai pas affiché le résumé (il est trop long !)

Suite de la page classique de l'OPAC, qui affiche une notice.

Logo, liens vers les catégories du thésaurus, statuts de publication

Champs des rubriques	Champs des articles
	

Le logo est une image associée à une rubrique ou à un article. Il en symbolise le contenu mais n'est pas obligatoire.
Employez des logos de dimensions et de proportions similaires, surtout si vous voulez en afficher plusieurs côte à côte, c'est plus esthétique.
N.B. Pour les logos des étagères virtuelles, vous trouverez plusieurs modèles d'étagères dans le dossier **pmb/opac_css/images**

Evitez des logos trop grands, pour l'équilibre graphique, et aussi pour réduire le « poids » et le temps de chargement des images. Redimensionnez éventuellement les logos avant de les relier au contenu éditorial, car ils sont **enregistrés dans la base elle-même, en même temps que les articles et rubriques**. De toute façon, cela ne sert à rien de stocker dans la base un logo de grandes dimensions et en haute définition, sa taille sera quand même réduite à l'affichage dans certains écrans. Inutile donc d'alourdir votre base sans raison valable.

On peut relier un article ou une rubrique à une ou plusieurs **catégories du thésaurus**.
N.B. C'est sur base des descripteurs que PMB sélectionne les notices affichées sous le titre « Sur le même sujet »

Je rappelle quelques points importants à propos des statuts de publication - Voir Configuration préliminaire

Ce n'est pas parce qu'une rubrique ou un article ont le statut « Publié – Visible par tous » qu'on les voit nécessairement !
L'affichage d'une rubrique ou d'un article n'est pas automatique, il faut le paramétrer correctement !
En d'autres termes, il faut « quelque part » un lien URL **vers** cette rubrique ou cet article, sinon ils sont inaccessibles.

Le statut « non publié » d'une rubrique rend invisibles les articles et sous-rubriques qui en dépendent !

Dates de publication, champs personnalisés éventuels

Champs des rubriques	Champs des articles
Pas de dates de publication dans la démo (voir ci-dessous)	Pas de dates de publication dans la démo (voir ci-dessous)

Ne confondez pas les dates d'un événement et les dates de publication d'un article ou d'une rubrique.
Les dates de publication indiquent la durée pendant laquelle un élément (rubrique ou article) doit rester visible.
Exemple : on peut parler d'une part d'un événement qui a lieu le 15 octobre, ou bien du 10 au 15 octobre, et décider d'autre part que l'article qui annonce cette animation sera visible dès le 15 septembre, et qu'il disparaitra automatiquement dès le 16 octobre.

Ce système évite de garder en page d'accueil ou ailleurs des infos qui sont visiblement obsolètes !
Parler au futur d'une activité qui s'est déroulée il y a une quinzaine de jours, cela ne fait pas sérieux …

On ne précise pas toujours les dates de publication, et c'est tout à fait logique dans certains cas.
En effet, certaines rubriques ou certains articles doivent toujours être visibles, car leur contenu est permanent, par exemple la présentation de la bibliothèque ou de l'institution, l'aide à la recherche, les infos pratiques, etc.
Donc on ne saisit dans ce cas aucune valeur pour les dates de publication, pour éviter la disparition intempestive de ce type d'infos.
N.B. C'est d'ailleurs pour cette raison que les rubriques et articles de la démo n'ont pas de dates de publication : il faut que la demo puisse résister à l'épreuve du temps ! Voir Particularités de la démo

Rappel : la visibilité d'une rubrique ou d'un article ne dépend pas que des dates de publication, mais aussi
- **de son statut de publication** (brouillon, à valider, publié et visible par tous, publié mais réservé aux abonnés …)
- **du statut de publication de sa rubrique parente éventuelle** (un article a toujours une rubrique parente ; une rubrique, pas nécessairement)
- **du bon paramétrage du portail** : l'affichage des rubriques et articles n'est pas automatique, il faut le configurer !

Champs personnalisés

La rubrique « Sélections » n'a pas de champ personnalisé. **Mais ce n'est pas une règle générale.** Dans la démo, il n'y a qu'un seul type de rubrique, une rubrique « simple », mais on aurait pu créer un autre modèle de rubrique liée à un champ personnalisé.	**L'article « Des bibliothèques pleines de fantômes » a un champ personnalisé** : l'identifiant de la notice. **C'est logique pour un article de type « Coups de cœur ».** **Mais ce n'est pas une règle générale : un article de type « Information » n'en a pas.** En outre, on aurait pu avoir un autre genre de champ personnalisé, ex. la date d'un événement, ou l'identifiant d'une étagère virtuelle, voire même deux champs : la date de début et la date de fin d'une période.

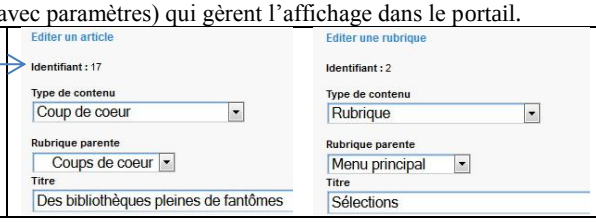
La présence ou non de champs personnalisés et la définition de ces champs dépendent du type de contenu choisi pour la rubrique ou l'article, et aussi du type de saisie : ex. texte (= saisie au clavier), date (= calendrier), liste de choix à partir d'une requête.

Il est très facile de connaître les numéros identifiants des rubriques et des articles, on les voit en mode édition.
Ces numéros d'id servent à construire les URL composées (donc avec paramètres) qui gèrent l'affichage dans le portail.

Deux exemples d'URL avec variables

http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=17
pageid=4 dans la démo, c'est la page qui affiche un article
id_article = 17 dans la démo, c'est l'article « Des bibliothèques … fantômes »

http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=2
pageid=6 dans la démo, c'est la page qui affiche une rubrique
id_rubrique = 2 dans la démo, c'est la rubrique « Sélections »

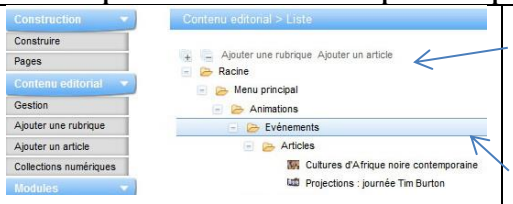


Il y a 2 manières d'ajouter une rubrique ou un article : l'une est parfois un peu plus rapide que l'autre.

Si on clique sur l'option « ajouter » dans la colonne de gauche, **aucune rubrique parente n'est sélectionnée par défaut**.

Peu importe la position de la ligne surlignée en bleu (dans le volet central). Il faut donc choisir absolument un parent.

C'est pourquoi l'autre manière de faire est parfois un peu plus rapide que celle-ci. Du moins si la ligne surlignée en bleu est déjà « à un bon endroit ».



N.B. Il n'est pas indispensable que le titre d'une rubrique soit surligné en bleu. Si un des articles avait été présélectionné (« Cultures d'Afrique noire » ou « Projections »), la rubrique parente par défaut aurait également été « Evénements ».

Si on clique sur l'option « ajouter » juste au-dessus de la racine, **une rubrique parente est sélectionnée par défaut**.

Le choix de cette rubrique parente dépend de la position de la ligne surlignée en bleu (dans le volet central).
Ici, elle vise la rubrique « Evénements ».

Si le choix par défaut ne correspond pas à ce qu'on voulait, on peut bien sûr sélectionner une autre rubrique parente.

EDITEUR HTML

Dans le portail, vous disposez d'un éditeur HTML simplifié pour mettre le texte en forme : dijit/Editor.

En cas de doute sur la signification d'une icône, sachez qu'une **infobulle** apparaît au survol de la souris, et vous indique sa fonction.

Risque de dégâts collatéraux ! Ne faites jamais de copier-coller direct depuis le traitement de texte !

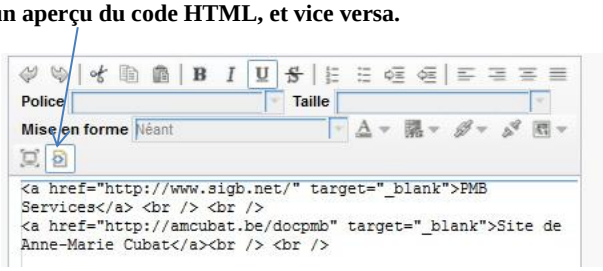
C'est vraiment une très mauvaise idée, car Word ou OpenOffice insèrent des codes de mise en forme qui peuvent désorganiser l'affichage – ou pire encore – perturber le fonctionnement normal de PMB. Il faut d'abord « nettoyer » ce texte.

Comment nettoyer rapidement un texte Word ou OpenOffice avant de le copier-coller dans l'éditeur html du portail ou ailleurs ?
J'ouvre tout simplement le bloc-notes ou Notepad ++ (le bloc-notes suffit amplement)

- un premier copier-coller du texte depuis Word ou OpenOffice vers le bloc-notes
- le bloc-notes nettoie automatiquement ce texte, il supprime « dès l'arrivée » les codes de mise en forme
- je refais un copier-coller du même texte, mais cette fois-ci à partir du bloc-notes
- je peux ensuite coller le résultat obtenu ailleurs sans souci - dans ce cas, dans l'éditeur html du portail
- je suis sûre que tous les codes de mise en forme ont disparu

Il existe aussi une autre manière facile de nettoyer le code dans l'éditeur même du portail.
J'ai découvert cette astuce grâce à un support de formation que Camille Bénard nous a fourni.
Une des icônes permet de basculer d'un texte mis en forme au code-source en HTML, et vice-versa.
C'était la seule fonction de cette icône que je connaissais, j'ignorais qu'elle avait aussi des « pouvoirs nettoyants » !

Comme dans tout éditeur de ce genre, une des icônes permet de passer d'un mode d'affichage à l'autre : d'un texte mis en forme à un aperçu du code HTML, et vice versa.



Explications complémentaires de Camille Bénard

« Si vous souhaitez coller du texte provenant d'un éditeur de traitement de texte (ex : MS Word, OpenOffice, etc.) ou d'un autre éditeur HTML, cliquez sur cette icône avant de coller votre texte. Cela vous permettra de récupérer uniquement le texte brut sans mise en forme, afin d'éviter des problèmes d'affichage liés au code provenant d'un autre programme. Ensuite, cliquez à nouveau sur l'icône afin de mettre en forme votre texte si nécessaire. »

Bénard, Camille, Portail PMB. Introduction aux concepts du portail et à l'utilisation du contenu éditorial - support non finalisé, 2014, p. 8

Que faire si le texte-source provient d'un autre éditeur HTML ?

Cette question n'est pas que rhétorique, vous y serez probablement confronté concrètement **le jour où vous voudrez intégrer dans le portail vos ex-infopages ou des pages HTML créées autrement.**

En général, le passage d'un éditeur HTML à l'autre pose moins de problème, car même si les codes de mise en forme sont transférés, il s'agit de balises HTML qui risquent moins de perturber le bon déroulement des choses que le traitement de texte.

Quelle option choisir : récupérer le texte brut ou le texte déjà formaté par les balises HTML ?

- **La plupart du temps, il vaut mieux récupérer le texte brut**. En effet, vous risquez de reproduire telle quelle la mise en page de vos ex-infopages, or rien ne garantit que leurs propriétés CSS (polices, tailles et couleurs des caractères) soient vraiment compatibles visuellement avec la feuille de style du portail. **Vous risquez de rompre l'équilibre graphique de l'ensemble**.

Attention toutefois !

- **Vous aurez du travail de reconstitution à faire ensuite, si la page présente une liste assez longue de ressources externes (liens vers d'autres sites).** En effet, les balises ne contiennent pas seulement des codes de mise en forme du texte.
- **Les liens URL sont aussi enregistrés dans des balises HTML !** Exemple : PMB Services
- **Si vous n'avez récupéré que le texte brut, les adresses ont disparu, et vous devez recréer tous ces liens URL un par un …**

Si vous voulez copier d'un éditeur html à l'autre en gardant les balises mais sans risquer de déclencher des problèmes

- Dans les deux éditeurs, cliquez sur l'icône qui permet d'afficher le texte brut (avec les balises visibles)
- Copiez-collez à partir de l'autre éditeur vers celui du portail. En théorie, la copie d'un texte html brut se fait sans souci.
- Dans l'éditeur du portail, cliquez à nouveau sur la même icône, vous voyez désormais le texte mis en forme et avec ses URL
- Ensuite, supprimez ou modifiez les balises de mise en forme (pour les raisons esthétiques expliquées ci-dessus).

Pour éviter toute surprise désagréable : des liens URL qui auraient disparu par mégarde

- **Ne supprimez pas trop tôt les infopages qui contiennent des liens vers d'autres sites ! Faites-en une copie externe à PMB.**
- Dans TinyMCE, cliquez sur l'icône qui permet d'afficher le texte brut (avec les balises, donc avec les URL des sites), copiez-collez ce texte brut ailleurs (bloc-notes ou Notepad ++) et sauvegardez ce fichier séparément.

Insérer ou supprimer un lien URL

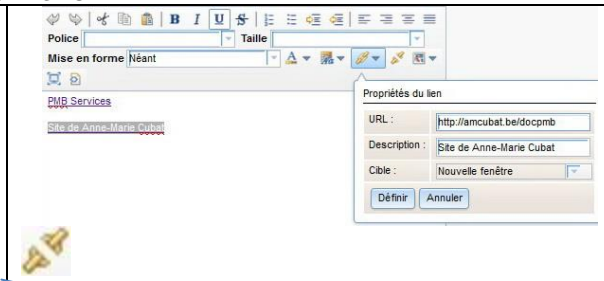
Sélectionner un ou plusieurs mots - Cliquer sur l'**icône « Lien »**

Saisir **l'URL de destination** et une brève description
Choisir la cible :

- *Fenêtre actuelle* pour un lien vers une page de votre portail
- *Nouvelle fenêtre* pour un lien vers un autre site (le lien s'ouvrira ainsi dans un autre onglet du navigateur)

Cliquer sur « Définir » pour créer et enregistrer ce lien.

Pour supprimer un lien URL : positionner le curseur quelque part dans le texte sélectionné, et cliquer sur l'icône à droite de « Lien »



Insérer une image

Positionner le curseur à l'endroit adéquat – Cliquer sur **l'icône « Image »**

Saisir l'URL de l'image et une brève description
(pour la balise « alt », le texte alternatif)
Cliquer sur « Définir » pour créer et enregistrer cette image.

Vous pouvez ensuite la redimensionner à la souris.

N.B. Ceci permet uniquement d'afficher des images à partir de leur URL, donc elles doivent déjà être en ligne « quelque part ».
Vous pourriez créer un répertoire spécifique, par exemple pmb/opac_css/images/portail/ ou bien les stocker ailleurs.
Rappel : pour les logos, le système est différent, ils sont enregistrés dans la base elle-même.

Comment rendre cette image cliquable ?
Cet éditeur ne permet pas de le faire directement via une icône spécifique, mais il suffit de cliquer sur l'icône « aperçu du code HTML » pour créer manuellement un lien à partir de l'image. L'aide-mémoire HTML vous rappelle quelles balises insérer.

Conseil final et mise en garde

Vous disposez d'un éditeur HTML pour mettre en forme le texte, mais n'en abusez pas !

Vous pouvez employer cet éditeur sans problème pour modifier un peu les choses (indenter le texte, ajouter des puces ou des listes numérotées, insérer des images ou des liens URL …), **mais n'en faites pas trop.**

N'oubliez pas que le portail est un CMS : c'est la feuille de style qui doit gérer l'aspect visuel de l'ensemble.

Si vous changez trop de propriétés CSS dans plusieurs articles (des polices différentes, ou des polices de tailles ou de couleurs différentes), vous pourriez vraiment rompre l'harmonie globale de votre site.

En effet, il se peut par exemple que le chapeau d'un de ces articles « modifiés » soit affiché juste à côté du chapeau d'un article dont la mise en forme respecte la feuille de style choisie, ce qui risque de créer un contraste peu esthétique.

Dans la même optique, veillez à ne pas réutiliser telle quelle la mise en forme d'une ancienne infopage, elle n'est peut-être pas vraiment compatible avec votre feuille de style actuelle. Importez de préférence le texte brut des ex-infopages, sans les balises html.

N'importez le texte formaté avec les balises html que si vos anciennes pages contiennent beaucoup de liens URL vers d'autres sites. Sinon, vous devriez reconstituer tous ces liens URL.

Gérer l'arbre du contenu éditorial

NAVIGUER DANS L'ARBRE DU CONTENU ÉDITORIAL

Comment afficher l'arbre du contenu éditorial et y naviguer ? C'est tout simple ;-)

Dès que vous accédez à l'onglet Portail, vous lancez automatiquement le sous-menu **« Gestion »**. Vous pouvez donc naviguer immédiatement dans l'arbre du contenu éditorial.

Si vous avez entretemps ajouté des pages, inséré ou déplacé des cadres, modifié des propriétés CSS …, il suffit de cliquer sur « Gestion », et vous avez à nouveau une vue d'ensemble sur l'arborescence.

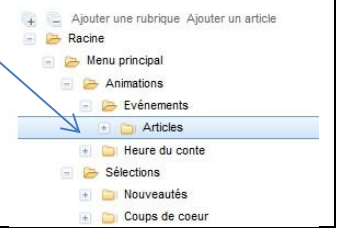
C'est l'interface classique des arborescences, les + et les – permettent de déplier ou replier les rubriques et les dossiers « Articles ».

Les dossiers « Articles »

Dès qu'on ajoute le premier article dans une rubrique, un dossier « Articles » se crée automatiquement.
Les dossiers « Articles » ne sont pas des « rubriques » à proprement parler, mais plutôt une manière astucieuse d'afficher l'arbre.

Astuce : on peut connaître le contenu d'un dossier « Articles » sans devoir le déplier !

Même si un dossier « Articles » est replié, vous pouvez voir la **liste des articles dans le volet de droite**, en cliquant sur le titre « Articles ». **Vous avez donc à la fois une vue d'ensemble sur l'arborescence (puisque les dossiers « Articles » sont repliés) et un aperçu du contenu de la rubrique.**



MODIFIER – SUPPRIMER UNE RUBRIQUE OU UN ARTICLE

Il suffit de cliquer sur le titre de la rubrique ou de l'article concerné, et vous entrez en mode Edition.
C'est exactement **la même interface** qu'auparavant, lorsque vous avez ajouté une rubrique ou un article.
Si vous avez modifié le contenu de l'un ou l'autre champ, n'oubliez pas d'**enregistrer** ensuite la version corrigée.

La **suppression** d'une rubrique (ou sous-rubrique) n'est bien sûr possible que si celle-ci est vide. Si elle contenait par exemple des articles ou des sous-rubriques, il faudrait d'abord déplacer ou supprimer ces articles ou sous-rubriques.

DÉPLACER UNE RUBRIQUE OU UN ARTICLE

Il y a deux manières de déplacer une rubrique ou un article.

1) Editer la rubrique ou l'article en question, changer la rubrique parente, et enregistrer la modification.
2) Afficher l'arbre du contenu éditorial, et déplacer les rubriques et articles grâce au « glisser-déposer » (drag and drop).

La 2e méthode est bien sûr plus rapide, puisque vous ne devez pas éditer chaque élément séparément.

Le principe de base du « glisser-déposer » est simple : N.B. On peut aussi déplacer plusieurs rubriques ou articles en même temps.

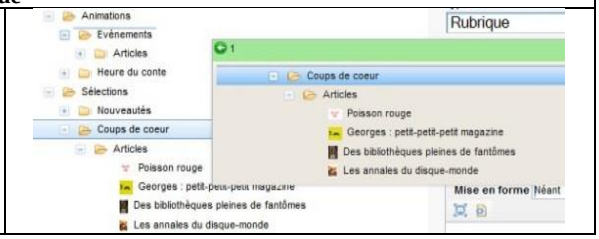
- positionner le curseur sur la rubrique ou l'article à déplacer
- cliquer et maintenir le bouton gauche de la souris enfoncé
- déplacer l'élément concerné vers sa nouvelle position
- si la nouvelle position est correcte, relâcher le bouton gauche de la souris

Attention ! Si vous pensez être parti dans une mauvaise direction, ne relâchez pas le bouton gauche de la souris, retournez d'abord à un endroit où le signal est rouge.

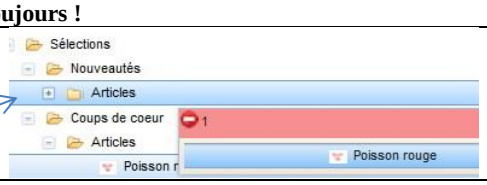
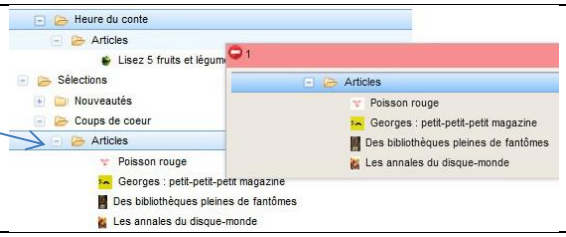
En effet, dès que vous relâcherez le bouton de la souris, la rubrique ou l'article changeront définitivement de place. L'enregistrement est immédiat, du moins si vous êtes dans une zone où le signal est vert !
Dans ce cas, vous devriez déplacer à nouveau la rubrique ou l'article, il n'y a pas de CTRL-Z qui vous ramènerait en arrière.

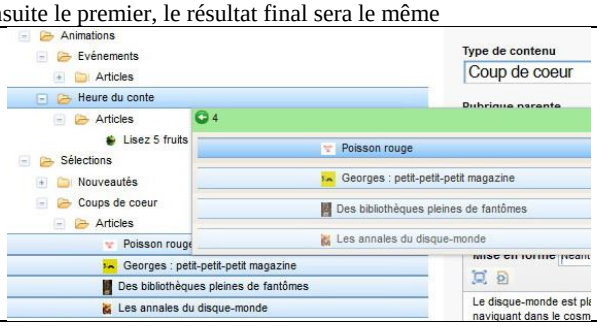
Je sais que cela a l'air tout à fait nébuleux pour le moment, surtout mes signaux verts ou rouges, mais les exemples des pages suivantes vous aideront à comprendre …

Quelques exemples pour clarifier le « code de la route » des déplacements dans l'arbre du contenu éditorial.

Déplacer un article vers une autre rubrique (ou dans la même rubrique)	
Exemple : « balader » le « pauvre poisson rouge » qui s'ennuie dans son bocal, et lui faire découvrir de nouveaux horizons …	
Le « poisson rouge » peut passer de la rubrique Coups de cœur à celle des Nouveautés. **Le signal est vert, car le point de destination est une rubrique.**	
Je n'ai pas pu préciser la place exacte que le « poisson rouge » devait occuper dans la liste des Nouveautés, puisque le dossier « Articles » était replié. J'ai seulement pu indiquer une rubrique de destination : Nouveautés. Conséquence : le « poisson rouge » a changé de rubrique. Hélas, il occupe la place par défaut, il est le dernier de la liste !	
Si je veux amener le « poisson rouge » dans la rubrique des Nouveautés, et spécifier où il doit aller, je dois d'abord déplier le dossier « Articles » avant de déplacer mon poisson. **Le signal est vert, car le point de destination est situé à l'intérieur d'un dossier « Articles », et le trait bleu indique l'endroit précis où l'article sera inséré** (il sera le premier de la liste).	
Tout va bien désormais : le « poisson rouge » occupe la place d'honneur dans la rubrique Nouveautés !	
Autre exemple de déplacement : le « poisson rouge » reste dans la même rubrique, mais il change de place, il devient le 2e article de la liste. **Le signal est vert, et le trait bleu indique l'endroit précis où l'article sera inséré** (entre Georges et les Bibliothèques).	
Déplacer une rubrique	
Je peux déplacer une rubrique complète, il suffit de positionner le curseur sur le titre de la rubrique concernée. **Une rubrique qu'on veut déplacer entraîne dans son sillage les articles et les sous-rubriques qui en dépendent.**	

Les dossiers « Articles » ne sont <u>pas</u> des « rubriques » ni des « articles ».
La différence est importante, car elle explique pourquoi certains déplacements sont autorisés, et d'autres pas.

Déplacer un article ? Oui, mais pas toujours !	
Le « poisson rouge » ne peut pas passer des Coups de cœur aux Nouveautés. **Le signal est rouge, car le point de destination n'est pas une rubrique ni l'intérieur d'un dossier « Articles », mais le titre « Articles ».**	
Déplacer un dossier « Articles » ? Non !	
Il est impossible de déplacer un dossier « Articles ». Dès que vous avez sélectionné un dossier « Articles » comme élément à déplacer, **le signal reste rouge tout le temps, peu importe la destination.** Par contre, la rubrique « Coups de cœur » peut changer de place (voir dernier exemple de l'encadré précédent)	

Déplacer plusieurs articles en même temps	
Sélectionner plusieurs articles consécutifs • Positionner le curseur sur le 1er article à déplacer et cliquer • Appuyer sur MAJ et maintenir la touche enfoncée • Positionner le curseur sur le dernier article et cliquer • Tous les articles intermédiaires sont aussi sélectionnés N.B. Pour varier, vous pouvez sélectionner d'abord le dernier article, et ensuite le premier, le résultat final sera le même	**Sélectionner plusieurs articles non consécutifs** • Appuyer sur CTRL et maintenir la touche enfoncée • Positionner le curseur sur un article et cliquer pour le sélectionner - répéter l'opération • En cas d'erreur, positionner le curseur sur l'article concerné et cliquer à nouveau pour le désélectionner • Certains articles sont ainsi sélectionnés
	
Attention ! Dans l'exemple de gauche, j'ai déplacé tous les articles de la rubrique Coups de cœur, mais je n'ai pas sélectionné la rubrique Coups de cœur elle-même. Cette rubrique ne changera donc pas de place pour le moment, mais elle deviendra forcément vide.	
On peut aussi déplacer plusieurs rubriques en même temps, mais la manœuvre est plus délicate. Une erreur est vite arrivée, et les conséquences éventuelles sont d'une autre ampleur ...	

Rappel : dès que vous relâchez le bouton gauche de la souris, l'enregistrement est instantané, donc la rubrique ou l'article changent définitivement de place, du moins si le signal est vert ! Il n'y a pas de CTRL-Z qui puisse vous ramener en arrière.

En cas d'erreur ou de doute sur le positionnement, déplacez d'abord le curseur dans une zone où le signal est rouge avant de relâcher le bouton gauche de la souris.
Ceci évite l'enregistrement de la nouvelle position de la rubrique ou de l'article.

Et si vous avez bien suivi mes explications, vous devez savoir désormais où positionner le curseur pour que le signal devienne rouge ;-)
Avez-vous réussi ce petit test sur le code de la route des déplacements ?

Vous pourriez vous demander pourquoi j'ai passé autant de temps à expliquer les subtilités des déplacements dans l'arbre du contenu éditorial, et pourquoi il y a autant de captures d'écran.

Un menu doit être clair et structuré pour le lecteur, il faut qu'il trouve rapidement l'info recherchée.
Un menu – même complexe - doit aussi être relativement facile à gérer pour le responsable du site.

Voici la méthode la plus simple pour créer un menu dans le portail :

- **calquer la structure du menu sur une partie de l'arborescence du contenu éditorial**
- **calquer l'ordre d'affichage des articles sur l'ordre dans lequel ils sont classés dans les rubriques**

Toutefois, élaborer un portail est un processus évolutif.

- **Même si vous avez un projet de départ assez précis, il est fort probable que la structure évolue au fil du temps**, et que vous décidiez de modifier un peu la hiérarchie et l'ordre de présentation, car vous avez découvert entretemps de nouvelles fonctionnalités du portail, ou créé de nouvelles rubriques.

- **A ce moment-là, vous devrez réorganiser votre menu, et par conséquent, déplacer des rubriques, sous-rubriques et articles dans l'arbre du contenu éditorial.**

C'est pourquoi j'ai voulu expliquer en détail comment modifier l'arborescence …

Gérer globalement le contenu éditorial

Au départ, un portail ne comprend que peu de rubriques et d'articles, et sa gestion ne pose pas trop de problème. Mais au fil du temps, le nombre d'articles augmentera, et il deviendra probablement plus difficile de s'y retrouver.

C'est particulièrement vrai s'il s'agit d'un portail multi-sites où chaque département ajoute des articles, parfois sans savoir exactement ce que les autres font. **Cela peut aussi être vrai dans une base mono-site, s'il y a plusieurs rédacteurs**. **Il faut donc qu'une personne joue le rôle de responsable général, afin de coordonner le tout**, d'éviter les redites (ou pire, les contradictions), et de veiller aussi à ce que les articles obsolètes soient mis à jour ou supprimés régulièrement.

Voici deux types d'outils pour vous aider dans cette tâche :

- **des requêtes SQL** qui vous aideront à garder une vue d'ensemble sur le contenu éditorial
- **la création de champs personnalisés supplémentaires**, qui seront visibles uniquement en gestion
  - ajouter dans les articles un commentaire de gestion, la date de la dernière mise à jour ou le nom du rédacteur
  - il ne me semble guère utile d'ajouter ce type d'info pour les rubriques, mais rien ne vous en empêche …

REQUÊTES SQL – LISTES DIVERSES : ARTICLES, RUBRIQUES, PAGES, TYPES DE CONTENU, ETC

Voici 2 exemples. N.B. La liste détaillée des requêtes disponibles en téléchargement se trouve à la fin de ce document.

Tableau de bord : vue d'ensemble des articles, classés par rubrique – N.B. Liste abrégée, je ne montre que 9 articles sur 21 !

Rappel : l'actualité est éternelle dans une démo, les articles n'ont donc pas de dates de parution et tous ont le statut « publié ».
J'ai donc dû inventer des dates de publication, pour montrer quelques exemples dans les colonne F et G du tableur.
J'ai aussi dû changer certains statuts de publication, pour avoir un peu de diversité dans les exemples de la colonne E.

Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
24/03/2015	Menu principal > Animations > Evénements	Cultures d'Afrique noire contemporaine	Evénement sur une période	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
24/03/2015	Menu principal > Animations > Evénements	Projections : journée Tim Burton	Evénement à une date précise	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
24/03/2015	Menu principal > Animations > Heure du conte	Lisez 5 fruits et légumes par jour !	Evénement à une date précise	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
24/03/2015	Menu principal > Infos pratiques > Inscription et prêt	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
24/03/2015	Menu principal > Infos pratiques > La bibliothèque	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
24/03/2015	Menu principal > Infos pratiques > La bibliothèque	Mais que font les bibliothécaires ?	Information	4. A valider			25/10/2012	7
24/03/2015	Menu principal > Sélections > Coups de coeur	Georges : petit-petit-petit magazine	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
24/03/2015	Menu principal > Sélections > Coups de coeur	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
24/03/2015	Menu principal > Sélections > Coups de coeur	Des bibliothèques pleines de fantômes	Coup de coeur	1. Publié - Pour tous			26/10/2012	17

Au départ, les données sont triées sur base de l'arbre du contenu éditorial, donc regroupées par rubrique.
Transférez le résultat vers le tableur, et supprimez d'abord la 1e ligne (celle qui contient le nom de la procédure) **avant** de trier.

Dans un tableur, il est très facile de trier les données dans l'ordre qui vous convient, par exemple par statut de publication.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
3	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
4	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
5	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
6	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
7	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
8	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
9	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7
10	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légumes	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2

On peut aussi **trier les données par type de contenu**.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
3	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
4	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
5	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
6	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légumes	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
7	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
8	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
9	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
10	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7

Le tri par type de contenu permet de vérifier si les dates de parution sont cohérentes. On voit ici des anomalies pour les Coups de cœur. En théorie, ce type de contenu a une durée de vie limitée, or les dates de début et/ou de fin de publication manquent pour 2 articles.
La situation est normale pour les Evénements (dates mentionnées) et les Informations (il n'y a aucune date car la publication est permanente)

On peut aussi **trier les données par numéro identifiant (du plus grand au plus petit)**, afin de **voir d'abord les articles les plus récents**.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
3	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
4	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
5	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
6	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
7	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7
8	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
9	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légumes	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
10	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1

Tableau de bord : vue d'ensemble des rubriques

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique parente	Titre de la rubrique	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	27/03/2015	Racine	Menu principal	Rubrique	1. Publié - Pour tous			26/10/2012	12
3	27/03/2015	Racine	Série de brèves	Rubrique	1. Publié - Pour tous			26/10/2012	13
4	27/03/2015	Racine	Archives	Rubrique	1. Publié - Pour tous			17/11/2012	14
5	27/03/2015	Racine > Menu principal	Animations	Rubrique	1. Publié - Pour tous			25/10/2012	1
6	27/03/2015	Racine > Menu principal	Sélections	Rubrique	1. Publié - Pour tous			25/10/2012	2
7	27/03/2015	Racine > Menu principal	Infos pratiques	Rubrique	1. Publié - Pour tous			25/10/2012	3
8	27/03/2015	Racine > Menu principal > Animatior	Evénements	Rubrique	1. Publié - Pour tous			25/10/2012	4
9	27/03/2015	Racine > Menu principal > Animatior	Heure du conte	Rubrique	1. Publié - Pour tous			25/10/2012	5
10	27/03/2015	Racine > Menu principal > Infos prati	La bibliothèque	Rubrique	1. Publié - Pour tous			25/10/2012	9
11	27/03/2015	Racine > Menu principal > Infos prati	Inscription et prêt	Rubrique	1. Publié - Pour tous			25/10/2012	10
12	27/03/2015	Racine > Menu principal > Sélection	Nouveautés	Rubrique	1. Publié - Pour tous			25/10/2012	7
13	27/03/2015	Racine > Menu principal > Sélection	Coups de coeur	Rubrique	1. Publié - Pour tous			25/10/2012	8

AJOUTER DES COMMENTAIRES DE GESTION

Un commentaire de gestion des articles peut être un aide-mémoire pour les bibliothécaires et/ou un outil de communication entre eux. Quelques exemples de commentaires possibles :

- article à retravailler – trop long, des redites
- article presque prêt – il manque l'adresse exacte de la salle et un plan d'accès
- article à garder dans les archives – on peut le réutiliser l'an prochain après une légère adaptation

En outre, ces commentaires de gestion peuvent aussi s'avérer fort utiles pendant la phase de construction du portail, comme **aide-mémoire des modifications à faire plus tard lors du transfert vers la vraie base**.

- Si un article a le type de contenu « Coups de cœur », il a un champ personnalisé pour l'identifiant de la notice.
- Toutefois, la base de travail ne contient qu'un nombre réduit de notices, donc les numéros d'id des notices n'y sont pas les mêmes que dans la base en ligne, et il faudra par conséquent modifier ces numéros plus tard, après le transfert du portail.
- Le plus simple est de noter, dans le champ « commentaires de gestion », le numéro d'id de notice qui **sera** nécessaire plus tard. Vous pouvez le trouver sans problème, vous avez accès à votre vraie base, donc il apparaît quelque part dans une barre d'adresses.
- Pour savoir comment repérer un numéro d'id, voir Numéros identifiants

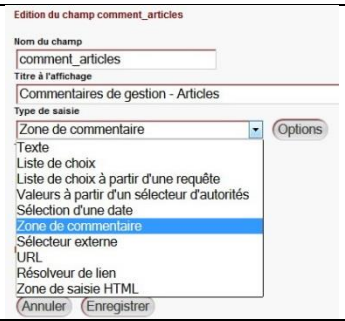
Pour éviter le travail fastidieux de créer ce champ un nombre x de fois (par exemple pour chaque type d'article)**, je vous conseille de recourir à un système prévu dans PMB : les champs génériques.**

Les champs génériques sont des champs communs soit à toutes les rubriques, soit à tous les articles.
Attention ! Il ne peut pas y avoir d'homonymes parmi les champs personnalisés, donc les noms doivent être tous différents.
Rappel - Le choix du nom des variables, des champs personnalisés et des répertoires obéit à des règles précises - sinon, erreur SQL !

Si vous voulez utiliser les commentaires de gestion partout, il faut donc créer 2 champs génériques : un pour les rubriques, et un autre pour les articles.
La méthode est la même : cliquer sur « Editer la liste des champs génériques »

Rappel : ce champ me semble fort utile pour les articles mais tout à fait superflu pour les rubriques.
En outre, ce n'est parce que ce champ existe qu'il doit absolument contenir un commentaire pour chaque article, il peut être vide !

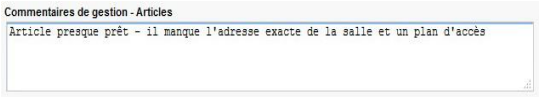
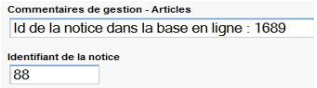
Champ personnalisé générique : commentaires de gestion des articles - Choix du type de saisie : 2 possibilités



Le type « texte » convient parfois moins bien, car on doit préciser une largeur du champ à l'écran, qui ne dépasse en général pas 120 caractères. Or, un commentaire de gestion risque parfois d'être plus long (255 caractères), ce qui oblige à scroller, sinon on ne voit pas la fin du message.

Le **type de saisie « zone de commentaire » me semble ici plus pratique**, car le commentaire s'affiche sur 3 lignes de 85 caractères (= 255 au total) et on peut tout lire d'emblée, il ne faut pas scroller.

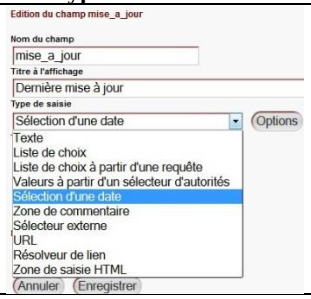
Le choix vous revient, de toute façon. Une petite aide visuelle pour vous guider : comparaison entre 2 types de saisie

Exemples : 2 types de saisie 2 genres de contenu	Zone de commentaire	Texte
	 Commentaire général pour un article de type « événement »	 Aide-mémoire pour faciliter le transfert ultérieur vers la vraie base

AJOUTER LA DATE DE LA DERNIÈRE MISE À JOUR

Un autre exemple de champ personnalisé : la table des articles contient un champ qui enregistre leur date de création, mais au stade actuel, **rien n'est prévu pour la date de la dernière mise à jour. Il faut donc ajouter un champ générique.**

N.B. Un champ générique est commun à tous les articles ou à toutes les rubriques (voir commentaires de gestion)

Type de saisie : date	Champ personnalisé générique des articles : date de la dernière mise à jour	
		**???** **Cocher ou ne pas cocher la case « ne pas employer la date du jour par défaut », telle est la question, dirait Shakespeare ….**
	Cliquer sur « Options » pour paramétrer le type de saisie. **Le type de saisie « date »** signifie « sélection grâce au calendrier ». **Un seul type de données est donc possible : une date.**	**Le type de données « date »** ne peut contenir qu'une date « SQL » : année - mois – jour. Ex. 2015-04-01 = 1er avril 2015

Clarifions d'abord le débat ! Que signifie le choix : ne pas cocher la case « ne pas employer la date du jour par défaut » ?
N'oubliez pas que ces 2 négations s'annulent, donc cela veut dire « employer la date du jour par défaut » !

Cas n° 1 : le champ « date de dernière mise à jour » est vide au moment de cliquer sur « enregistrer »

Si le champ « mise à jour » ne contient aucune valeur (= le message « aucune date ») au moment de cliquer sur « enregistrer », parce que vous venez de créer un nouvel article ou d'effacer la valeur précédente dans un article préexistant, **voici ce qui se passera au moment où vous éditerez à nouveau cet article** :

- **Si vous n'aviez pas coché cette case, ce champ changera de contenu au moment de l'enregistrement dans la base, il contiendra automatiquement la date du jour où vous avez enregistré l'article pour la dernière fois**.
  Donc la date de mise à jour correspondra au départ à la date de création de l'article, et plus tard à la date de la dernière édition.
- **Si vous aviez coché cette case, ce champ date restera dans l'état où il était antérieurement, sans contenu.**
  Donc il risque peut-être de rester vide un certain temps ...

Cas n° 2 : le champ « date de dernière mise à jour » a un contenu réel au moment de cliquer sur « enregistrer »

Dès que le champ « dernière mise à jour » a un contenu réel (autre que « aucune date ») au moment de cliquer sur « enregistrer », il ne va plus se mettre à jour automatiquement dès qu'on éditera l'article, même si la case n'avait pas été cochée au départ (lorsqu'on avait configuré le contenu éditorial).

- Si l'utilisateur a bénéficié d'une facilité lors de la rédaction d'un nouvel article (= le champ « mise à jour » contient d'office une valeur correcte, la date de création de l'article), il risque de croire que cela se répétera plus tard, lorsqu'il devra éditer cet article.
- Or, ce n'est pas vrai. Si on édite un article dont le champ « dernière mise à jour » contient déjà une valeur, la date qui s'y trouve ne changera pas, sauf intervention manuelle de l'utilisateur – peu importe que la case ait été cochée ou non.
- Puisque le bibliothécaire devra quand même changer la date lui-même dès qu'il éditera un article, autant ne pas lui donner « de mauvaises habitudes » au départ, en lui laissant croire que « la mise à jour du champ date de mise à jour » est automatique !
- En outre, on édite beaucoup plus souvent qu'on ne crée, donc les chances statistiques d'avoir déjà un contenu (même erroné) dans ce champ sont plus élevées.
- Dernier argument : si vous employez les requêtes SQL que je propose pour avoir une vision d'ensemble du contenu éditorial, un champ « dernière mise à jour » qui est vide va davantage attirer votre attention qu'un champ qui contient une valeur identique à la date de création de l'article. Ce sera pour vous l'occasion de rappeler à tout le monde l'importance de ce champ.

Pour toutes ces raisons, je vous suggère donc de cocher la case « ne pas employer la date du jour par défaut ».

Le champ « dernière mise à jour » restera vide au départ, et même peut-être pendant un certain temps, malgré les rééditions éventuelles de l'article par des bibliothécaires distraits.
Mais au moins, le champ « dernière mise à jour » ne contiendra pas une valeur qui risquerait vite de devenir erronée et de passer inaperçue : la date de création de l'article.

Cela simplifie aussi les consignes à donner aux autres rédacteurs :

- Lorsque vous créez un article, inutile de compléter le champ « date de mise à jour », la date de création suffit.
- Dès que vous éditez un article, il faut saisir une valeur dans ce champ, en ouvrant le petit calendrier intégré :
  - y mettre pour la première fois une date, ou remplacer la date qui s'y trouvait par une autre

AJOUTER LE NOM DU RÉDACTEUR

Si plusieurs personnes collaborent au contenu éditorial, il est également souhaitable de prévoir un champ personnalisé, afin d'enregistrer le nom des rédacteurs des divers articles (ce champ me semble lui aussi superflu pour les rubriques).

Le type de saisie « texte » ne convient pas, même si c'est celui auquel certains pourraient penser au départ.

- En effet, si on laisse les rédacteurs libres d'introduire leur nom au clavier, rien ne garantit que d'une fois à l'autre, ils saisissent leur nom sous la même forme : le prénom, le prénom abrégé, parfois le surnom, parfois le prénom et le nom de famille, etc.
- Toutes ces variantes empêcheraient de sélectionner les articles sur base du rédacteur, il faut normaliser les choses.

Deux types de saisie sont dès lors envisageables pour ce champ générique. A vous de choisir une des deux solutions !

N.B. Un champ générique est commun à tous les articles ou à toutes les rubriques (voir commentaires de gestion)

Première hypothèse : une liste de choix préétablie

Type de saisie : liste de choix	Champ personnalisé générique : nom du rédacteur	Importance du champ « choix vide »
Cliquer sur « Options » pour paramétrer le type de saisie.		Champ complété / Champ vide
		Complétez le champ « choix vide » ! **Si vous avez complété ce champ**, vous voyez les 2 noms et une possibilité de choix vide : « sélectionner un nom ». **Si vous n'avez pas complété ce champ**, il faut choisir un des 2 noms. Si vous ne faites rien, PMB prendra par défaut le 1er nom de la liste, et XX travaillera plus que XY !
		L'ordre d'affichage est imposé : Ladies first !
Le type de saisie « liste de choix » se base sur une liste préétablie. Chaque choix possible (ici, le nom du rédacteur) correspond à une valeur numérique (1, 2 …), ce qui va déterminer le type de données.	**Le type de données est un « entier »** car le type correspond à la valeur choisie : des nombres (1, 2). Mais on aurait aussi pu choisir le type « texte » pour enregistrer ces nombres. Si on avait choisi des lettres pour symboliser les valeurs (ex. les initiales des rédacteurs), le type de données aurait été « texte ».	

Deuxième hypothèse : une liste de choix à partir d'une requête (liste des utilisateurs)

Type de saisie : liste de choix à partir d'une requête	Champ personnalisé générique : nom du rédacteur	Importance du champ « choix vide »
		Champ complété / Champ vide
		Complétez le champ « choix vide » : voir remarque dans l'encadré précédent Si vous ne complétez pas ce champ, admin devra travailler davantage que les autres !
		Ordre d'affichage : d'abord admin La requête spécifie que les logins seront affichés dans l'ordre alphabétique. Order by 2 = tri sur le 2e champ cité (tri sur username, le login de l'utilisateur)
Le type de saisie « liste de choix à partir d'une requête » se base sur le contenu de la table users, donc la liste de tous les utilisateurs.	**Le type de données est un « entier »** car la requête renvoie le numéro identifiant de chaque utilisateur, donc un entier.	

Dans des circonstances « normales », on choisit <u>un</u> de ces 2 types de saisie pour les champs personnalisés :
liste de choix basée sur une liste préétablie **<u>ou</u>** liste de choix à partir d'une requête.
Pour faire « cohabiter » ces 2 types de rédacteur dans la même base
j'ai forcément dû donner un nom SQL différent au 2e champ que j'ai créé : redacteur_user
Pour clarifier mes explications, et éviter tout risque de confusion entre ces 2 types de saisie
j'ai adapté le titre à l'affichage : Rédacteur (user) et j'ai engagé d'autres rédacteurs choisis dans l'équipe de PMB Services !

Pourquoi faut-il dans ce cas-ci compléter la case « choix vide » au moment de créer un champ personnalisé ?

Si le champ « choix vide » est vide, un choix vide est impossible !

Ce type de problème se pose ici quelle que soit la méthode employée : liste préétablie ou liste à partir d'une requête.
Si vous n'avez rien saisi dans le champ « choix vide » (dans la définition du champ personnalisé), un choix vide (dans l'article) est impossible. Vous devrez malgré tout choisir un nom avant de sauvegarder l'article, même si aucun nom de la liste ne correspond au vôtre.
Si vous tentez d'esquiver ce choix, un nom par défaut est de toute façon imposé au moment d'enregistrer l'article : le 1er nom de la liste !
La même chose se passe si un rédacteur est distrait, et oublie de sélectionner son nom : c'est le 1er nom de la liste qui est enregistré.
Cela crée l'impression erronée que XX ou admin ont beaucoup plus travaillé que les autres ! (voir exemples de la page précédente).
L'existence d'un choix vide est nécessaire, non seulement pour éviter de fausser les données, mais aussi pour avoir la possibilité d'effacer le contenu d'un champ personnalisé via l'interface de PMB. Si le choix vide n'existe pas, vous ne pouvez supprimer la valeur d'un champ personnalisé que par l'interface de PhpMyAdmin ou par une requête SQL.

Quelques pistes de réflexion : liste de choix préétablie ou liste de choix à partir d'une requête ?

Mise à jour manuelle ou automatique ?

Liste de choix préétablie
Il faut penser à mettre cette liste à jour **manuellement**, si des rédacteurs s'ajoutent.
Vérifiez bien sûr si les noms saisis dans la liste correspondent réellement à des utilisateurs qui ont accès à l'onglet Portail !

Liste de choix à partir d'une requête
Cette liste se met à jour **automatiquement**, il ne faut pas intervenir, vu qu'elle se base sur le contenu actuel de la table users.
La liste risque d'être fort longue car elle comporte les noms de tous les bibliothécaires, qu'ils aient ou non accès à l'onglet Portail.

Que se passe-t-il si un bibliothécaire change de département ou quitte l'institution ?

Liste de choix préétablie
Vous pouvez postposer le moment de mettre la liste à jour, et conserver donc le nom de l'ex-utilisateur pendant quelque temps.
Cela vous permet de garder temporairement une trace des activités des rédacteurs qui sont partis ou ont changé de fonction.

Liste de choix à partir d'une requête
En général, on supprime rapidement un ex-utilisateur de la table users, afin qu'il n'ait plus accès à la base.
Il est dès lors impossible de retrouver le nom du rédacteur initial de certains articles, puisque son numéro d'id n'existe plus.
PMB est une base de données relationnelle, donc tout se résume toujours à des questions de numéros identifiants !

Ordre d'affichage des champs génériques

Comment changer l'ordre d'affichage des champs génériques ?

L'ordre dans lequel on crée les champs personnalisés génériques est parfois une question de hasard, et ce n'est pas toujours l'ordre le plus fonctionnel, dès qu'on doit saisir des données dans les articles. Il est possible de le modifier, et le processus est fort simple.

Au départ, les champs sont affichés dans l'ordre de leur création (aussi bien dans cet écran que dans l'article lui-même).
Pour modifier l'ordre d'affichage, cliquer sur les + et les – afin de déplacer les champs un par un vers le haut (-) ou vers le bas (+).

	Nom du champ	Titre à l'affichage
- +	comment_articles	Commentaires de gestion - Articles
- +	redacteur	Rédacteur
- +	mise_a_jour	Dernière mise à jour

Requêtes SQL – sélection d'articles selon les types de contenu standards

Si vous avez restauré le jeu de sauvegarde cms_min (configuration du contenu éditorial), vous avez des types de contenu standards : leurs numéros d'id sont identiques à ceux de la démo, et il en va de même pour les 5 premiers champs personnalisés qui sont associés à ces types de contenu (voir liste page suivante). Dans ce cas, ces requêtes sont immédiatement prêtes à l'emploi !
Sinon, il faudra bien sûr adapter les numéros identifiants, sur base des infos fournies par la 1e requête de la page suivante.

Ces requêtes sélectionnent les articles selon le type de contenu, et permettent des vérifications. Quelques exemples.

Coups de cœur

Id de la notice ?
Annales ? Ou
La 8e couleur ?

	A	B	C	D	E	F	G	H	I	J	K
1	Edité le	Rubrique	Titre de l'article	Type de conten	Id	Titre abrégé de la notice	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Sélec	Des bibliothèques pl	Coup de coeur	88	Des bibliothèques pleines	1. Publié - Pour tous			26/10/2012	17
3	12/04/2015	Menu principal > Sélec	Georges : petit-petit-	Coup de coeur	129	Georges	1. Publié - Pour tous			26/10/2012	12
4	12/04/2015	Menu principal > Sélec	Les annales du disque	Coup de coeur	104	La huitième couleur	1. Publié - Pour tous			26/10/2012	18

Après vérification, on voit que le n° d'id de la notice est correct : « La 8e couleur » est un des titres de la série « Les Annales ».

Sélection
Etagère associée à l'article

	A	B	C	D	E	F	G	H	I	J	K
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Id	Etagère	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Sélé	Cinéma	Sélection	8	Nouveautés DVD	1. Publié - Pour			26/10/2012	16
3	12/04/2015	Menu principal > Sélé	Documentaires	Sélection	6	Nouveautés documentaires	1. Publié - Pour			26/10/2012	14

Vérification
Article sans type de contenu

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Tests divers	Test n° 4		1. Publié - Pour tous			3/04/2015	30

Ces requêtes servent aussi à la fin du processus, quand on prépare la mise en ligne du portail, par exemple pour vérifier les n° d'id des notices ou des étagères (ces numéros ne sont probablement pas tous les mêmes dans la base de test et dans la base en production).

REQUÊTES SQL - SÉLECTION D'ARTICLES SELON UN CHAMP PERSONNALISÉ SUPPLÉMENTAIRE

Les procédures de cette série-ci sont « spécifiques », elles sont liées à des champs personnalisés supplémentaires dont les numéros d'id peuvent différer d'une base à l'autre (commentaire de gestion, date de la dernière mise à jour, nom du rédacteur).

Toutefois, si vous avez défini les champs personnalisés comme je l'ai fait (même type de saisie et même type de contenu), et si vous leur avez donné le même nom SQL (ex. nouvel_onglet), il n'y a aucun changement à faire ;-)

Outils - Liste des champs personnalisés – enregistrés dans la table cms_editorial_custom – triés par numéro d'id

	A	B	C	D	E	F	G	H	I	J	K
1	Type de contenu	Id	Titre (affichage)	Champ (nom SQL)	Type de saisie	Type de données	Valeur	Contenu	Obligatoire?	Indexation?	Exportable?
2	Sélection	1	Etagère associée	etagere	Liste à partir d'une requête	Entier			Non	Non	Non
3	Événement sur une période	2	Date de début	date_debut	Sélection de date	Date			Non	Non	Non
4	Événement sur une période	3	Date de fin	date_fin	Sélection de date	Date			Non	Non	Non
5	Coup de coeur	4	Identifiant de la notice	identifiant_notice	Texte (saisie au clavier)	Texte court			Non	Non	Non
6	Événement à une date précise	5	Date de l'événement	date	Sélection de date	Date			Non	Non	Non
7	CP pour Rubrique	8	Commentaires de gestion - Rubriques	comment_rubriques	Texte (saisie au clavier)	Texte court			Non	Non	Non
8	CP pour Article	9	Commentaires de gestion - Articles	comment_articles	Zone de commentaire	Texte court			Non	Non	Non
9	CP pour Article	10	Rédacteur	redacteur	Liste de choix préétablie	Entier	1	XX	Non	Non	Non
10	CP pour Article	10	Rédacteur	redacteur	Liste de choix préétablie	Entier	2	XY	Non	Non	Non
11	CP pour Article	11	Rédacteur (user)	redacteur_user	Liste à partir d'une requête	Entier			Non	Non	Non
12	CP pour Article	12	Dernière mise à jour	mise_a_jour	Sélection de date	Date			Non	Non	Non

N.B. Ce sont les valeurs dans ma base de travail ; selon toute probabilité, certaines seront différentes dans votre base.
Toutefois, si vous avez restauré le jeu de sauvegarde cms_min, les 5 premiers champs sont identiques (ce sont les valeurs de la démo).

Pour les types de contenu, la signification de 2 libellés (« CP pour … ») pourrait intriguer de prime abord.

CP = champ personnalisé

CP pour Article = champ générique pour article
→ commun à tous les articles
CP pour Rubrique = champ générique pour rubrique
→ commun à toutes les rubriques

L'explication se trouve dans la table qui définit ces types de contenu : cms_editorial_types

	A	B	C
1	Id	Type de contenu	Type (nom SQL)
2	1	Rubrique	section
3	2	Sélection	article
4	3	Événement sur une période	article
5	4	Information	article
6	5	Coup de coeur	article
7	6	Événement à une date précise	article
8	7	CP pour Article	article_generic
9	8	CP pour Rubrique	section_generic

Lorsqu'on base un type de contenu sur une liste de choix (c'est le cas ici pour le rédacteur), la liste des valeurs et contenus possibles est enregistrée dans une autre table : cms_editorial_custom_lists

Cela correspond à la liste des rédacteurs possibles.
Voir lignes 9 et 10 du tableur.

cms_editorial_custom_champ	cms_editorial_custom_list_value	cms_editorial_custom_list_lib	ordre
10	1	XX	0
10	2	XY	1

Sélection : liste partielle des articles – seulement s'il y a une date de mise à jour

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Mis à jour le	Créé le	Id
2	04/04/2015	Menu principal > Anima	Projections : journée Tin	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/03/2015	25/10/2012	11
3	04/04/2015	Menu principal > Infos	Mais que font les bibliotl	Information	4. A valider			26/03/2015	25/10/2012	7

Sélection : liste partielle des articles – seulement si le rédacteur est connu (liste préétablie de choix)

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Rédacteur	Créé le	Id
2	29/03/2015	Menu principal > Animations > Evér	Cultures d'Afrique noire c	Evénement sur une pério	1. Publié - Pour tous	13/04/2015	30/05/2015	XX	25/10/2012	1
3	29/03/2015	Menu principal > Animations > Evér	Projections : journée Tim	Evénement à une date pi	2. Publié - Abonnés	11/05/2015	24/05/2015	XY	25/10/2012	11
4	29/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légumes	Evénement à une date pi	5. Archives	23/03/2014	23/03/2014	XX	25/10/2012	2

Sélection : liste partielle des articles – seulement si le rédacteur est connu (liste à partir d'une requête – contenu de la tables users)

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Rédacteur	Créé le	Id
2	04/04/2015	Menu principal > Sélections > Cou	Des bibliothèques pleines	Coup de coeur	1. Publié - Pour tous			Eric	26/10/2012	17
3	04/04/2015	Menu principal > Sélections > Cou	Les annales du disque-m	Coup de coeur	1. Publié - Pour tous			Florent	26/10/2012	18
4	04/04/2015	Menu principal > Sélections > Cou	Georges : petit-petit-petit	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	admin	26/10/2012	12
5	04/04/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		Camille	25/10/2012	4

Sélection : liste partielle des articles – seulement s'il y a des commentaires de gestion

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Commentaire de gestion	Créé le	Id
2	04/04/2015	Menu principal > Infos	Mais que font les bibliothé	Information	4. A valider			Il manque 2 photos pour illustrer l'article	25/10/2012	7
3	04/04/2015	Menu principal > Séle	Des bibliothèques pleines	Coup de coeur	1. Publié - Pour tous			Id de la notice dans la base en ligne : 1689	26/10/2012	17

Ceci n'est qu'un aperçu de quelques procédures.

J'ai donné la liste complète des requêtes SQL, des explications sur la manière de les adapter si nécessaire, et le lien pour télécharger le zip à partir de mon site.

Sauvegarder – restaurer le contenu éditorial

Comme le nom l'indique clairement, c'est tout le contenu éditorial qui est sauvegardé dans le jeu « cms_edit » :

- la configuration éditoriale minimum (types de contenu pour les rubriques et articles, et statuts de publication)
- le contenu des rubriques et articles
- les liens entre d'une part les articles ou rubriques et d'autre part les catégories
- les tables d'indexation, indispensables pour la recherche dans le contenu éditorial
- on peut y ajouter les collections de documents numériques (présentées grâce au module Portfolio)

C'est un ensemble dont tous les éléments sont interconnectés, et qui correspond à une base précise - vu les liens entre le contenu éditorial et le contenu bibliographique (notices, catégories du thésaurus …). Il comprend 11 ou 14 tables.

Jeu cms_edit - Liste des 11 tables de base du contenu éditorial (PMB 4.0.13 – PMB 4.1)	
Nom des tables	**Commentaire**
Cms_articles	Contenu des articles
Cms_articles_descriptors	Liens entre articles et catégories
Cms_editorial_custom	Définition des champs personnalisés liés aux types de contenu (rubriques ou articles)
Cms_editorial_custom_lists	
Cms_editorial_custom_values	Valeurs de ces champs personnalisés dans la base
Cms_editorial_fields_global_index	Indexation pour la recherche dans les rubriques et articles
Cms_editorial_publication_states	Statuts de publication
Cms_editorial_types	Types de contenu (rubriques et articles)
Cms_editorial_words_global_index	Indexation pour la recherche dans les rubriques et articles
Cms_sections	Contenu des rubriques
Cms_section_descriptors	Liens entre rubriques et catégories

PMB 4.1 contient des modules qui n'existaient pas dans la version 4.0, entre autres le Portfolio, qui permet de créer et de présenter des collections de documents numériques. 3 tables se sont ajoutées au jeu cms_edit, c'est pourquoi le nombre total de tables liées au portail a augmenté lors du passage à la version 4.1 de PMB (c'est désormais 26 au lieu de 23).

Jeu cms_edit - Liste des 3 tables supplémentaires (PMB 4.1)	
Nom des tables	**Commentaire**
Cms_collections	Collections de documents numériques
Cms_documents	Documents numériques
Cms_documents_links	Liens entre les documents numériques et les articles ou rubriques

Si vous faites des sauvegardes régulières de tout le contenu du portail, il n'est pas indispensable de créer ce jeu de sauvegarde spécifique. Mais il peut s'avérer vital dans certains cas de connaître la liste des tables qui en font partie !

En effet, si vous avez déjà bien avancé au niveau du contenu éditorial, mais que vous constatez que vous êtes parti sur une mauvaise piste au niveau graphique (feuille de style, agencement des modules et cadres), il est vraiment utile et agréable de pouvoir repartir à zéro pour l'aspect visuel, mais sans perdre le contenu des rubriques et des articles déjà rédigés, ni leurs liens avec votre catalogue, ni les liens avec les documents numériques intégrés dans le Portfolio.

Le portail de PMB est un « vrai » CMS qui permet de gérer séparément l'aspect visuel et le contenu, c'est pourquoi je voulais absolument vous donner la liste complète des tables liées au contenu éditorial et aux documents numériques.

Vous voudrez ou devrez peut-être les restaurer un jour, on ne sait jamais ;-)

Vous progressez !

Votre imprimante 3D vient de vous fournir une belle série de briques éditoriales.
Il ne reste qu'un seul modèle de briques à fabriquer : les cadres issus des modules !

PAGES DE L'OPAC ET DU PORTAIL

Zones et pages

Le portail est divisé en cinq zones : introduction, bandeau (à gauche), bandeau2 (à droite), page et pied de page
C'est le cas pour toutes les pages qui sont affichées, et on retrouve ces zones dans la définition de toutes les feuilles de style

Les 5 zones sont « fixes » graphiquement, dans la mesure où elles ne changent ni de taille ni de position ni de couleur quelle que soit la page affichée, elles existent donc partout.

La plupart du temps, 3 zones sont aussi « fixes » au niveau du contenu : introduction, bandeau et pied de page.
C'est plus esthétique, cela donne plus de cohésion et de « stabilité » à l'ensemble (ce n'est qu'une partie de l'écran qui est rafraîchie lorsqu'on appelle une autre URL).

Bandeau2 a en général un contenu variable (ex. facettes ou carrousel), **voire même inexistant dans certaines pages.**
Il n'existe d'ailleurs pas dans toutes les feuilles de style.

Choisissez bien la largeur disponible pour la page centrale.
N'oubliez pas que votre catalogue sera aussi consulté sur tablette ou téléphone portable (mode portrait ou paysage). C'est pourquoi il vaut parfois mieux se passer du bandeau de droite, car sa présence réduit forcément la largeur de la page centrale !

La page centrale est la seule zone dont le contenu varie en fonction de l'activité du lecteur.
C'est une URL différente qui est appelée à chaque fois (URL d'une page du portail ou URL d'une page de l'OPAC), suivant ce que le lecteur fait ou demande : recherche dans le catalogue ou dans les articles, consultation d'une notice, réservation d'un document, lecture d'un article …
Une exception : en général, on positionne le panier du lecteur tout en haut, c'est le seul cadre fixe de cette page.

Introduction		
Bandeau (le bandeau de gauche	Page (ou page centrale)	Bandeau2 (le bandeau de droite)
Pied de page		

Les 5 zones sont « fixes » graphiquement dans la mesure où, à partir du moment où vous les avez définies, elles ne changeront ni de taille ni de position ni de couleur, quelle que soit la page affichée (OPAC classique ou portail).
Mais ces valeurs ne sont pas figées une fois pour toutes. Vous pouvez les redéfinir, en adaptant la feuille de style choisie au départ, et/ou en modifiant des paramètres CSS. Mais ne le faites pas trop tard, sous peine de devoir recommencer beaucoup de choses !

La zone « main » (la partie principale)

Lorsqu'on consulte la liste des zones de l'OPAC (onglet Portail), on voit apparaître une zone supplémentaire, la zone intitulée « main ». Elle existe pour des raisons de CSS, de mise en page, et elle regroupe plusieurs zones.

Configuration dans le cas de la feuille de style pmb4
(j'ai modifié un peu le style pmb4 afin de vous expliquer)

La zone « main » est celle qui est encadrée en bleu.
La zone « page » est celle qui est encadrée en rouge.

Ceci n'est <u>pas</u> une vérité générale.
La configuration de « main » est complexe et varie d'une feuille de style à l'autre.

Ex. dans genbib, « main » est « flottant à droite » (alors qu'il est défini de manière absolue dans pmb4), et il comprend d'autres zones.

« Main » sert à regrouper plusieurs zones et existe dans toutes les feuilles de style, mais ce ne sont pas toujours les mêmes zones, cela varie d'une feuille de style à l'autre.
Lorsque vous allez construire votre portail, vous devrez placer des cadres dans des zones : ces cadres sont issus d'un module (ex. agenda, carrousel, rubrique, liste de notices ou d'articles …). Vous pouvez positionner des cadres dans les 5 zones citées au départ : introduction, bandeau, bandeau2, page et pied de page, mais pas directement dans la zone intitulée « main ».

SIGNIFICATIONS DU TERME « PAGE »

Dans le contexte du portail, le terme « page » a plusieurs significations.

La page est l'unité élémentaire du portail de PMB : c'est un ensemble cohérent affiché à l'écran, qui est constitué de plusieurs zones, comme la partie supérieure, la partie centrale, le bandeau de gauche, etc.

- La mise en page est la même dans toutes les pages de type OPAC ou portail (taille et position des zones)
- Le contenu de certaines zones est en général fixe (introduction, bandeau de gauche et pied de page), ce qui donne une cohérence et une stabilité à l'ensemble, même quand on appelle une autre URL
- En général, ce n'est que dans la partie centrale que l'affichage change (et parfois aussi dans le bandeau de droite – du moins s'il existe)

Le portail de PMB affiche deux types de pages :
- **les pages « traditionnelles » de l'OPAC** (catalogue, recherche, affichage de notices, de périodiques …)
- **les pages « de type portail »** (page d'accueil, page qui affiche une rubrique, un article et une liste de notices …)

Mais la page est aussi une des 5 zones définies dans la feuille de style !

Lorsqu'on élabore une feuille de style pour PMB, on divise en effet la page en 5 zones :
- l'introduction (la partie supérieure, qui contient en général un logo, un menu, un message de bienvenue …)
- le bandeau de gauche (connexion au compte du lecteur, choix de la langue, adresse de la bibliothèque …)
- la « page », c'est-à-dire la partie centrale, celle dont le contenu varie en fonction de l'activité du lecteur
- le bandeau de droite (il n'existe pas dans toutes les feuilles de style – on peut y mettre un carrousel ou les facettes)
- le pied de page (utilisé pour la navigation)

Donc, la page (unité élémentaire du portail) comprend plusieurs zones (définies dans la feuille de style), et l'une de ces zones s'appelle aussi page !!

Si le contexte peut prêter à confusion, j'emploie l'expression « page centrale » pour désigner la page en tant que zone.

Pages de l'OPAC

Même si on a créé un portail, toutes les pages « classiques » de l'OPAC subsistent, vous avez pu le constater dans certaines captures d'écran des pages précédentes :

- exemple n° 3 : page catalogue (c'était la page d'accueil classique de l'OPAC) – recherche, catégories, butineur de sections …
- exemple n° 8 : page qui affiche une liste de notices (résultat d'une recherche dans le catalogue)
- exemple n° 9 : page qui affiche une notice
- exemple n° 12 : page qui affiche un périodique

On pourrait aussi citer d'autres exemples : les pages de recherche (simple, multicritères, prédéfinie ou externe), les pages de résultats, les pages des autorités, les pages du compte du lecteur, etc.

Je tiens à rappeler quelques points importants :
- **Vous n'êtes pas obligé de remplacer la page d'accueil habituelle de l'OPAC par une page d'accueil de type portail, ou du moins pas tout de suite**, vous pouvez le faire plus tard.
- N'oubliez pas de configurer correctement le paramètre first_page_params dès que vous faites ce changement (page d'accueil).
- **Vous pouvez intégrer progressivement dans votre OPAC quelques éléments du portail, comme la liste de notices, le carrousel de nouveautés ou le flux RSS**, **même si vous n'avez pas encore créé ou employé les pages du portail**. En effet, ces modules peuvent être « accrochés » à une page de l'OPAC ou à une page du portail.
- N.B. Il vaut mieux employer le module « liste de notices » plutôt que le carrousel : une liste de notices peut être affichée sous forme de liste ou sous forme de carrousel, et les possibilités de paramétrage de la liste sont plus étendues (grâce aux templates).
- Par contre, **si vous voulez ajouter des éléments typiques du portail, par exemple des rubriques et des articles**, il faut d'abord **créer les pages du portail et configurer le contenu éditorial**.

Créer les pages du portail

Comment créer les pages du portail

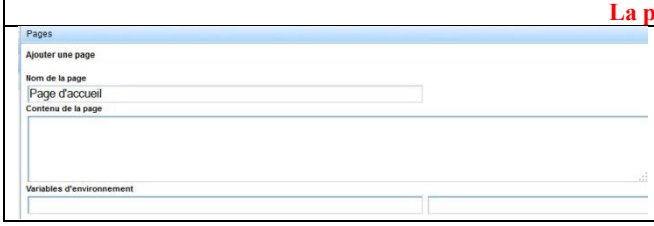
Dans votre base de travail, vous devez créer vous-même les divers types de pages du portail qui sont utiles au départ.
Vous trouverez la liste de ces 6 pages et leurs caractéristiques dans la démo du portail.

Le mieux est de commencer par la page d'accueil, qui aura ainsi l'identifiant n° 1.

Pour ajouter une page

Si nécessaire, reparcourez les exemples de pages du portail : j'ai précisé le type de page illustré dans chaque capture d'écran.

La page d'accueil



La page d'accueil n'a apparemment aucune caractéristique particulière, si ce n'est qu'elle s'appelle « page d'accueil ».
Vous imaginez sans peine que ce n'est pas uniquement le fait de lui donner ce nom qui suffira !

Il faudra en outre configurer le paramètre first_page_params, qui contient le numéro identifiant de cette page.

Les 4 pages suivantes ont un point commun : elles se basent sur ce qu'on appelle une « variable d'environnement »

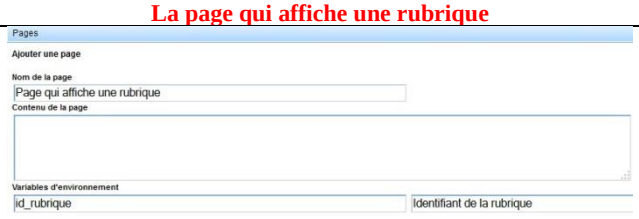
Ces variables d'environnement sont employées pour construire des URL, on ajoute un (des) paramètre(s) après le « ? » dans l'URL. C'est ainsi que se construisent les liens de navigation entre d'une part les pages et d'autre part les rubriques, articles, notices, etc.
Ex. id_rubrique dans la démo du portail http://127.0.0.1/pmbv4portail/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=4
La page n° 6 est celle qui affiche une rubrique (dans la démo), et « on donne l'ordre » d'afficher la rubrique dont le n° d'id est 4.

Le choix du nom des variables, des champs personnalisés et des répertoires obéit à des règles précises - sinon, erreur SQL !

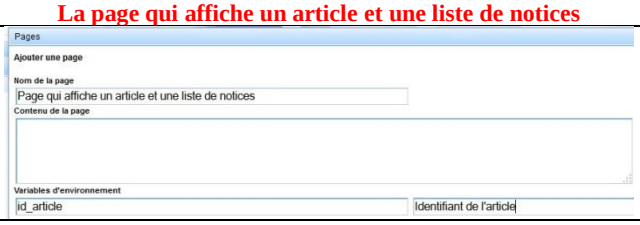
- un nom en un mot (pas d'espace !), qui comporte uniquement des caractères alphanumériques non accentués (donc rien d'autre que des lettres ou des chiffres), et qui doit commencer par au moins une lettre – de préférence, entièrement en minuscules
- aucun symbole mathématique, signe de ponctuation ou autre symbole n'est toléré
- une seule exception : le trait de soulignement _ (underscore) afin de marquer la séparation en 2 mots (id_article, id_date …)
- ne confondez pas le trait de soulignement et le tiret (identique au symbole mathématique « moins », c'est pourquoi il est interdit)

Par contre, la formulation du libellé (ex. page qui affiche une rubrique) est libre, c'est le message qui sera affiché à l'écran.

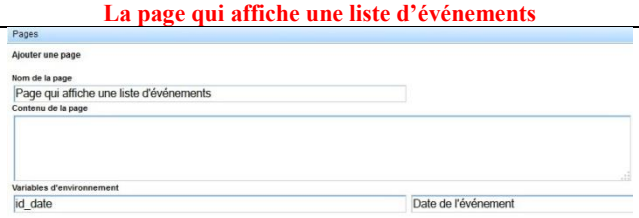
La page qui affiche une rubrique



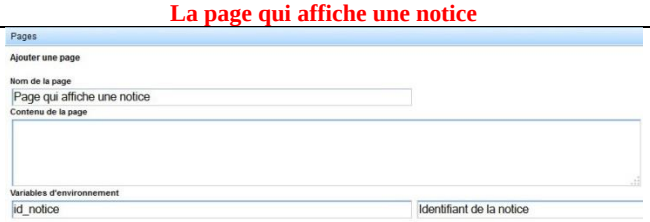
La page qui affiche un article et une liste de notices



La page qui affiche une liste d'événements

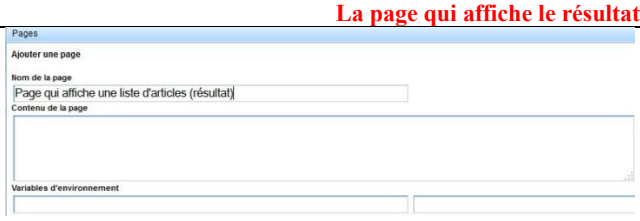


La page qui affiche une notice



La dernière page prédéfinie ne fait référence à aucune variable globale d'environnement.

La page qui affiche le résultat d'une recherche dans les articles



C'est logique, car elle affiche une liste d'articles, sélectionnés après une recherche dans le contenu éditorial.

On ne peut pas deviner a priori ce qui sera recherché, il n'est donc pas possible de définir une variable, tout dépend de ce que le lecteur va saisir au clavier comme critère de recherche et de ce qu'il peut trouver dans la base.

Si vous préférez voir la liste des pages dans le même ordre que dans la démo, créez les pages dans cet ordre-ci.
Même si vous changez l'ordre, essayez quand même de créer la page d'accueil en 1er lieu, ainsi son n° d'id sera 1, c'est plus simple à mémoriser.

N° id	Nom de la page	Variable	N° id	Nom de la page	Variable
1	Page d'accueil		4	Page qui affiche une liste d'événements	id_date
2	Page qui affiche un article et une liste de notices	id_article	5	Page qui affiche une notice	id_notice
3	Page qui affiche une liste d'articles (résultat)		6	Page qui affiche une rubrique	id_rubrique

Les 6 pages de la démo suffisent au départ pour les besoins courants, mais vous pouvez en créer d'autres !

Comment obtenir la liste des pages et de leurs numéros identifiants, ainsi que des variables d'environnement ?

Il suffit d'exécuter ces requêtes dans l'onglet Editions. N.B. Voir Requêtes SQL utiles : liste des procédures et zip à télécharger.

	A	B
1	Id	Nom de la page
2	3	Page d'accueil
3	4	Page qui affiche un article et une liste de notices
4	6	Page qui affiche une rubrique
5	7	Page qui affiche une notice
6	9	Page qui affiche une liste d'événements
7	11	Page qui affiche une liste d'articles (recherche)

	A	B	C	D	E
1	Id (page)	Nom de la page	Id (variable)	Variable globale	Libellé
2	4	Page qui affiche un article et une liste de notices	6	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	4	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	7	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	10	id_date	Identifiant de date

N.B. Ce sont les valeurs dans la démo du portail.

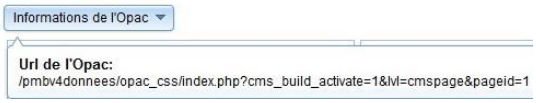
Comment connaître rapidement et avec certitude le numéro identifiant d'une page du portail ?
Il y a deux manières de sélectionner la page qui vous intéresse.

Naviguer dans le volet de droite afin d'afficher la page concernée
Ex. page d'accueil

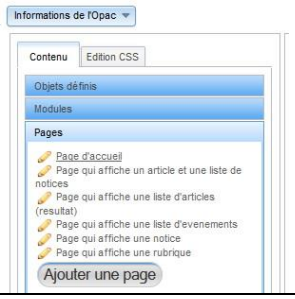
Cliquer ensuite sur la petite flèche à droite des mots « Informations de l'OPAC »

Vous verrez le numéro d'id de la page à la fin de la liste des paramètres.

Dans cet exemple, pageid=1 pour la page d'accueil.

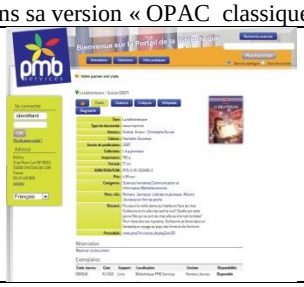
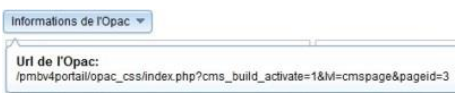
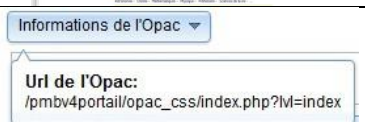


Sélectionner la page concernée dans la liste des pages



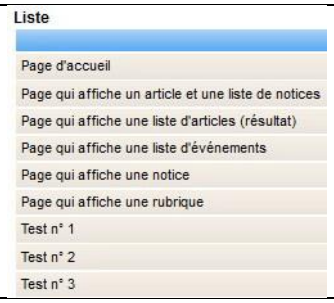
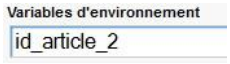
Comment distinguer aisément les pages du portail et les pages de l'OPAC ?

Naviguer dans le volet de droite afin d'afficher la page à propos de laquelle on a des doutes.
Cliquer ensuite sur la petite flèche, à droite des mots « Informations de l'OPAC », et on voit l'URL.

Page d'accueil dans sa version « portail »	Page d'accueil dans sa version « OPAC classique »	Page qui affiche une notice dans sa version « OPAC classique »
		
		
Page du portail : dans l'URL, vous voyez l'id du cms actif (« cms_build_activate=1») et l'id de la page (« pageid =… »)	**Page de l'OPAC : l'adresse est construite autrement L'URL ne comporte aucune référence au cms actif ni à pageid**	

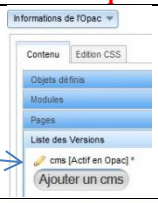
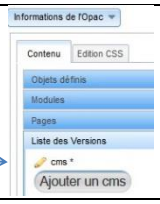
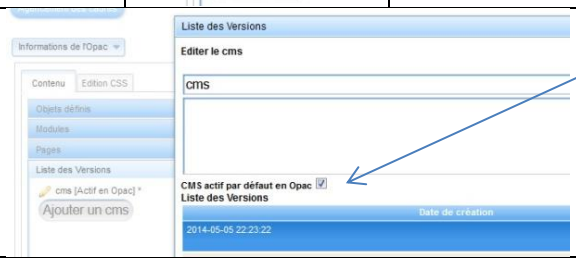
A ce stade-ci, on a seulement créé 6 types de pages du portail, mais elles ne sont pas encore directement exploitables.
Si on veut les employer, il faudra ajouter dans les pages-outils des cadres issus des modules : afficher une rubrique, un article, une liste d'articles (résultat d'une recherche) …

PAGES SUPPLÉMENTAIRES

	**Les 6 pages de la démo suffisent normalement pour gérer un portail. Mais vous pouvez créer d'autres pages dans la démo, c'est d'ailleurs une excellente manière de vous entraîner !** Vous pouvez créer 3 pages comme dans l'exemple ci-contre, cela permet déjà de faire des tests. Essayez d'y insérer des cadres issus des modules, cela vous aidera à comprendre le processus et à acquérir une certaine dextérité dans le positionnement des cadres. A partir du moment où une de vos pages de test est plus élaborée, vous pouvez même en faire une page d'accueil provisoire, en mettant son numéro identifiant dans le paramètre first_page_params (voir ci-dessous).
Nouveau nom 	**N.B. Ne pas employer le même nom de variable d'environnement pour des pages différentes**. Au départ, une des variables d'environnement s'appelait id_article (pour la page qui affiche un article). Utilisez un autre nom pour la nouvelle page, par exemple id_article_2 ou id_article_bis. Si nécessaire, faites de même pour d'autres pages de test.

Paramètres cms et first_page_params

PARAMÈTRE CMS

Avant de vous lancer dans la construction du portail, vérifiez l'état du paramètre cms.		
Pendant que vous construisez le portail sur une base de test, ou à partir du moment où il est finalisé et utilisé en ligne, il faut que le CMS soit actif en permanence, donc il faut que le paramètre cms ait la valeur 1 (Administration > Outils > Paramètres de l'OPAC)		
Valeur par défaut (base standard)	cms 0	id du CMS utilisé en OPAC
Valeur par défaut (base avec un portail)	cms 1	id du CMS utilisé en OPAC
N.B. Le bouton « ajouter un CMS » (qu'on voit dans la capture d'écran ci-dessous) est prévu pour des développements ultérieurs. On envisage en effet de pouvoir créer plusieurs cms, afin de les associer à des vues différentes, mais ce n'est pas encore opérationnel. C'est pourquoi le paramètre cms doit garder pour le moment la valeur 1 (c'est l'id par défaut au stade actuel).		
Il faut aussi que le cms soit actif dans l'OPAC.		
Portail > Construire **Dans ce cas-ci, le cms est déjà actif en OPAC, le message est clair.**		**L'absence du message « cms actif en OPAC » indique qu'il faut agir !** 
Cliquer sur l'icône du crayon pour éditer le cms. La liste des versions apparaît.		**Cocher la case « CMS actif par défaut en OPAC »** **N'oubliez pas d'enregistrer !** (le bouton « enregistrer » est tout en bas, à la fin de la liste des versions sauvegardées)

PARAMÈTRE FIRST_PAGE_PARAMS

Le paramètre first_page_params sert à désigner une page d'accueil différente de celle par défaut, donc autre que la page d'accueil classique de l'OPAC (appelée aussi la page catalogue).

Bien que ce paramètre soit fréquemment utilisé avec le portail, il n'est pas lié uniquement au portail.

- **Il n'est pas indispensable de configurer ce paramètre pour mettre en œuvre certains modules du portail**, car vous pouvez conserver la page d'accueil classique et y ajouter des cadres (liste de notices en carrousel, flux RSS)
- **Vous pouvez employer ce paramètre dans un OPAC classique, même sans portail**

A quoi sert ce paramètre dans le cas du portail ?		
Lorsqu'on construit un portail, on remplace en général la page d'accueil classique par une page de « type portail ». **Mais vous savez déjà que la page d'accueil qui a été créée (dans le portail) n'a aucune caractéristique spéciale, et que le nom qu'on lui a donné ne suffit pas à la transformer en une page d'accueil !** **Vu que PMB ne peut pas « deviner » quelle page vous avez choisie comme page d'accueil du portail, ni dans quel ordre vous avez créé les divers types de pages, il faut lui donner le numéro identifiant de la page sélectionnée (son numéro d'id).**		
Structure Json récapitulant les paramètres à initialiser pour la page d'accueil : Exemple : { "lvl":"cmspage", "pageid":2 }		Le commentaire de ce paramètre contient un petit script à recopier dans le champ valeur.
Contenu du champ « valeur » Dans un PMB standard	first_page_params	Au départ, ce champ est vide.
Contenu du champ « valeur » Dans la démo du portail	first_page_params { "lvl":"cmspage", "pageid":"3" }	On a donné à pageid la valeur 3, car dans la démo, la page d'accueil du portail a le numéro identifiant 3.

Configurer le paramètre first_page_params	
Sous-type first_page_params Valeur { "lvl":"cmspage", "pageid":1 } En théorie, pour vous, ce sera 1	**Selon toute vraisemblance, les numéros identifiants chez vous seront différents de ceux de la démo.** **Pour votre future base, la valeur sera probablement 1 - du moins si la page que vous avez créée en 1er lieu est la page d'accueil** (et si vous ne l'avez pas supprimée enretemps et recréée ensuite). Vous copiez-collez dans le champ valeur le petit script que vous trouvez dans le champ commentaire. Il donne comme exemple la valeur 2, donc vous devrez probablement remplacer le 2 par le chiffre 1.
Si first_page_params n'est pas configuré Si le champ valeur du paramètre est vide	on voit la page d'accueil par défaut, donc la page d'accueil de l'OPAC
Si first_page_params est bien configuré Le n° d'id est valide, il correspond à une des pages	on voit la page d'accueil du portail - du moins si c'est le « bon numéro » ! sinon, on verrait une autre page, celle dont l'id est mentionné
Si first_page_params est mal configuré Le n° d'id n'est pas valide, il ne correspond à aucune page	la partie centrale de la page est vide mais on voit l'introduction, le bandeau et le pied de page
Si la page ne contient pas encore de cadre **ou si elle ne contient pas de cadre bien paramétré**	la partie centrale de la page est vide mais on voit l'introduction, le bandeau et le pied de page
Si vous venez à peine de créer les pages du portail, et qu'elles sont encore vides, il est trop tôt pour configurer first_page_params ! **En effet, aussi longtemps qu'une page ne contient pas le cadre requis (celui qui correspond à sa fonction), elle ne sert à rien !** **Donc, les premiers cadres à insérer correspondent aux types de pages de portail qu'on a créées** (page qui affiche une rubrique, un article …), **et la page d'accueil sera vraisemblablement la dernière dont on s'occupera, puisqu'elle fait appel aux autres pages.**	

URL avec variables et navigation

AUTRES USAGES DU PARAMÈTRE FIRST_PAGE_PARAMS

N'importe quelle page du portail ou de l'OPAC peut être désignée comme page d'accueil à l'aide du paramètre first_page_params. Pour cela il faut décomposer les variables de l'URL de la page choisie. Voici quelques exemples.		
Les symboles « ? » et « & » permettent de « fractionner » l'URL ? = début des variables & = séparateur entre 2 variables. Voici la syntaxe à respecter s'il y a plusieurs variables **http://** **début de l'URL** **?** **paramètre1** **&** **paramètre2** **&** **paramètre3**		

Une page du portail sans variable
Ex. la page d'accueil du portail
	http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=3
	{
	"lvl":"cmspage",
	"pageid":"3"
	}	

Une page du portail avec une variable d'environnement qui a été définie Exemple n° 1 page qui affiche une rubrique (pageid = 6) rubrique sélectionnée (id_rubrique = 3) Exemple n° 2 page qui affiche un article (pageid = 4) article sélectionné (id_ärticle = 9)	http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=3 http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=9	

	{
	"lvl":"cmspage",
	"pageid":"6"
	"id_rubrique":"3"
	}
		{
		"lvl":"cmspage",
		"pageid":"4"
		"id_article":"9"
		}

Une page de l'OPAC
Ex. une infopage dont l'id est 2
	http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=infopages&pagesid=2
	{
	"lvl":"infopages",
	"pagesid":"2"
	}	

Une page de l'OPAC
Ex. une recherche prédéfinie
	http://127.0.0.1/pmb4perso/opac_css/index.php?search_type_asked=extended_search&onglet_persopac=1
	{
	"search_type_asked":"extended_search",
	"onglet_persopac":"1"
	}	

« POINTER » DEPUIS L'EXTÉRIEUR VERS UNE PAGE SPÉCIFIQUE DE VOTRE SITE

Maîtriser ces URL avec variables peut également vous servir dans d'autres contextes.

La plupart des institutions importantes ont 2 sites : un pour l'institution elle-même et un autre pour la bibliothèque.

A partir du moment où la bibliothèque dispose elle aussi d'un site avec portail intégré, il est fort possible que certains événements importants liés à l'activité de la bibliothèque soient annoncés sur les deux sites.

Ex. une rencontre avec un auteur, une exposition thématique dans les locaux de la bibliothèque

Dans votre portail, vous aurez donc vraisemblablement un article lié à une date précise (la conférence de l'auteur), ou à une période (la durée de l'expo).

Pour un article spécifique, visible dans la page du portail qui affiche un article, l'URL dans la barre d'adresses ressemblera à ceci : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=9
Vous voyez qu'il y a plusieurs paramètres après le « ? » dans l'adresse (ils sont séparés par le symbole « & »)

C'est le moment de mettre vos collègues au travail ! Les collègues qui s'occupent du site de l'institution … Demandez-leur de faire un lien, mais pas n'importe lequel, **un lien vers l'article où vous présentez l'auteur ou le thème de l'expo**. Il suffit de transmettre le lien mentionné ci-dessus, et les internautes aboutiront directement à l'article concerné.

Attention toutefois à la coordination !

Si vos collègues laissent trop longtemps leur article en ligne, il se peut que le vôtre ait entretemps disparu, car **vous** aurez bien sûr paramétré correctement ses dates de publication ! Cela va de soi …
Donc le lien ne fonctionnera plus, et on verra le message d'erreur 404 : page non trouvée.

En conclusion, on peut dire que c'est un bon système, qui permet de « pointer depuis l'extérieur » vers une page précise de votre catalogue (portail ou OPAC, peu importe), page qui est liée à un contenu spécifique.
Mais il ne faut l'employer que si vous êtes sûr de la coopération des collègues qui travaillent sur l'autre site (ou si c'est vous qui êtes en charge des deux !), et de la régularité des mises à jour de l'autre côté. Sinon, l'effet risque d'être contre-productif.

Dans quel ordre configurer les pages du portail ?

A force de se focaliser sur la page d'accueil, on en oublierait presque que le portail contient d'autres pages, et qu'aucune de ces pages n'est utilisable pour le moment, car elles sont toutes vides, on vient à peine de les créer !

En outre, je sais que je vais décevoir certaines personnes qui piaffent d'impatience.

La page d'accueil est probablement la dernière où vous mettrez des cadres issus des modules !!

Il faut d'abord configurer les autres pages du portail, car jusqu'à présent elles ne peuvent pas remplir la fonction qui leur a été assignée. Le nom qu'on leur a donné ne leur confère pas de « pouvoirs particuliers », l'existence éventuelle d'une variable d'environnement n'y change rien, **il faut y insérer des cadres, qui afficheront tel ou tel contenu (rubrique, article, notice, liste d'événements, résultat d'une recherche)**. **Il faut que ces pages deviennent de véritables** pages-outils.

Essayer de mettre trop tôt un lien vers le contenu éditorial en page d'accueil **sera en général voué à l'échec, sauf** une liste d'articles, mais il faut alors **une liste de « brèves »** comme dans la démo - sinon, c'est une catastrophe graphique.

Les autres modules de type éditorial qui sont visibles en page d'accueil de la démo (liste ou carrousel d'articles, agenda …) **font appel à une « page-outil » du portail**, car ce que le lecteur sélectionne (un article, une rubrique, une date dans l'agenda …) sera affiché dans une autre page du portail que celle d'accueil, il faut donc que cette autre page soit réellement opérationnelle.

Exemple concret : on a créé la page-outil qui affiche un article avec une variable d'environnement qui s'appelle id_article, mais si cette page ne contient pas effectivement un cadre issu du module « afficher un article », elle ne servira à rien !

Toutefois, vous pouvez insérer rapidement en page d'accueil du portail (ou de l'OPAC) un module bibliographique (ex. carrousel de nouveautés), **ou un module autonome par rapport à votre base** (ex. flux RSS), **puisque ces modules ne font pas référence au contenu éditorial, et ne font donc pas appel aux pages-outils du portail**.

Repensez à l'analogie avec l'imprimante 3D (dans le préambule) : il faut d'abord fabriquer les briques ;-)

Sans les pages-outils, pas de portail opérationnel !

MODULES, CADRES ET PAGES

Cadres de l'OPAC et du portail

Un cadre est un élément visuel de forme rectangulaire, un « petit ensemble fondamental ». Il se caractérise par

- **les conditions d'affichage** : visible dans toutes les pages ou dans une page spécifique ? page(s) de l'OPAC ou du portail ?
- **les propriétés CSS** : positionnement ? dans quelle zone ? où exactement ? dimensions ?
- **le contenu et la source des données** : quels types d'informations (notices, articles …) ? d'où viennent ces données ?

LES CADRES DE L'OPAC SONT LES « CADRES DE BASE » DE PMB, ILS EXISTENT D'OFFICE

- **Ils sont visibles même si on n'a pas activé le portail, et visibles aussi après avoir activé le portail** !
- **Ils sont visibles dans les pages de l'OPAC <u>et</u> dans les pages du portail**

Certains cadres de l'OPAC sont « fixes », c'est-à-dire présents dans toutes les pages (OPAC ou portail, peu importe)

- **On les place en général dans les zones « fixes », c'est-à-dire celles dont le contenu ne change pas**, **comme l'introduction, le bandeau de gauche et le pied de page.** En effet, ces cadres sont en général assez petits, donc ces 3 zones conviennent a priori.
- **Il y a toutefois une exception notable : le panier du lecteur** est « fixe » (présent partout) mais on le place la plupart du temps en haut de la page centrale, qui est une zone dont le contenu est variable.

Introduction
Message de bienvenue

Page centrale
Panier du lecteur

Bandeau de gauche
Bloc de connexion
Adresse
Accueil (choix de la langue)

Pied de page
Barre de navigation
Liens externes

D'autres cadres de l'OPAC sont « variables », ils sont liés à un type spécifique de page.

- **On préfère placer les cadres variables dans la page centrale, la zone dont le contenu varie en fonction de l'activité du lecteur** (consultation du catalogue, recherche, affichage d'une notice ou d'une autorité, accès au compte du lecteur …)

Page centrale

Onglets de recherche (simple, multicritères, dans les périodiques)

Types possibles de recherche simple

Butineur de sections

OPAC - Page d'accueil

Liste des catégories

OPAC – Résultats d'une recherche

Liste des résultats Facettes

On configure toujours les cadres de l'OPAC dans l'onglet Administration, peu importe que vous ayez activé le portail ou non.

- Administration > Outils > Paramètres de l'OPAC
- Administration > OPAC > Infopages – Navigation - Recherche prédéfinie – Facettes …

Ce que le portail apporte en plus, par rapport au paramétrage « classique » dans l'onglet Administration, c'est la possibilité de modifier l'aspect visuel des cadres de l'OPAC

- Déplacer les cadres de l'OPAC d'une zone à l'autre
- Déplacer les cadres de l'OPAC dans la même zone, afin de changer l'ordre d'affichage
  - Exemple : déplacer le cadre « accueil » (choix de la langue)
- Editer les propriétés CSS des cadres de l'OPAC : modifier la position, les dimensions, les marges …

LES CADRES DU PORTAIL NE PRÉEXISTENT PAS DANS PMB, IL FAUT LES CRÉER À PARTIR DES MODULES

- **Ils ne sont visibles <u>que si</u> on a activé le portail** !
- **Ils sont visibles dans les pages du portail <u>et</u> dans les pages de l'OPAC**

Certains cadres du portail sont « fixes », c'est-à-dire présents dans toutes les pages (OPAC ou portail, peu importe)

- **Il vaut mieux mettre les cadres fixes dans les zones « fixes », c'est-à-dire celles dont le contenu ne change pas,** comme l'introduction, le bandeau de gauche et le pied de page. Mais concrètement, les 2 dernières zones conviennent moins bien.
- **Les cadres fixes du portail (ex. le menu ou le bloc de recherche) sont souvent trop grands que pour être placés de manière ergonomique dans le bandeau de gauche ou le pied de page, c'est pourquoi on les place en général dans l'introduction.**

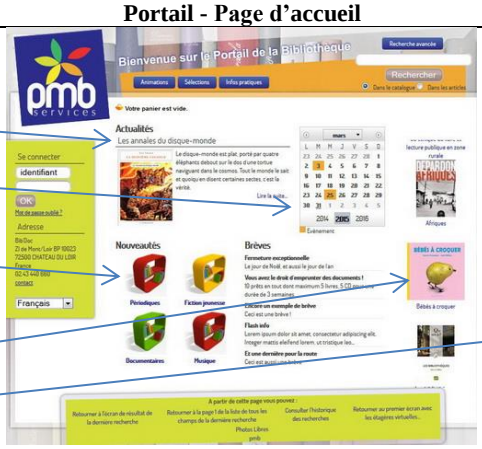
Introduction
Menu
Bloc de recherche

N.B. Si le menu était dans le bandeau de gauche, il serait difficile de s'y retrouver !

Explications
Voir page suivante

D'autres cadres du portail sont « variables », ils sont liés à un type spécifique de page.

- **On préfère placer les cadres variables dans la page centrale, la zone dont le contenu varie en fonction de l'activité du lecteur** (contenu d'une rubrique, affichage d'un article, liste ou carrousel d'articles, carrousel de notices, agenda …)

	Portail - Page d'accueil	Portail – Rubrique « Nouveautés »
Page centrale Carrousel d'articles Agenda Liste d'articles Carrousel de notices Contenu d'une rubrique		

On crée et on configure toujours les cadres du portail dans l'onglet Portail.

Quand on crée un cadre à partir d'un module, il faut en définir le contenu, la source de données, la (les) page(s) de destination, les propriétés CSS, les liens avec le contenu bibliographique ou éditorial, la vue (ex. vue générique, template Django, liste ou carrousel)

Dans quelle zone placer les cadres de l'OPAC et du portail ?

- **En théorie, tous les cadres de l'OPAC et du portail peuvent être placés dans n'importe quelle zone, et dans n'importe quel type de page (OPAC ou portail).**
- **Mais dans la pratique, il vaut mieux privilégier des zones ou des pages spécifiques pour certains types de cadres, pour des raisons d'ergonomie et d'équilibre graphique.**

- **Il vaut mieux mettre les cadres fixes (= présents dans toutes les pages) dans les zones fixes, c'est-à-dire celles dont le contenu ne change pas, comme l'introduction, le bandeau de gauche et le pied de page.**
- Les 3 zones citées ci-dessus conviennent en général bien pour les cadres de l'OPAC (qui sont assez petits).
- Certains cadres fixes du portail (menu ou bloc de recherche) sont souvent trop grands que pour être placés de manière ergonomique dans le bandeau de gauche ou le pied de page, c'est pourquoi on les place en général dans l'introduction.
- Voir ci-dessous un exemple pour le menu : le déplacer dans le bandeau de gauche ne convient pas pour certaines feuilles de style !

- **Il y a toutefois une exception : le panier du lecteur** est fixe (présent partout) mais on le positionne habituellement en haut de la page centrale, qui est une zone dont le contenu est variable.

- **Il vaut mieux mettre les cadres variables (= présents dans certains types de pages) dans une zone variable.**
- **On préfère les placer dans la page centrale, la zone dont le contenu varie en fonction de l'activité du lecteur** (consultation d'une notice, recherche dans les articles ou dans le catalogue, lecture d'un article, accès au compte du lecteur ...)

- **Ce système contribue à la stabilité graphique à l'ensemble, puisque ce n'est qu'une partie de l'écran (la page centrale) qui est rafraîchie lorsqu'on appelle une autre URL**, et que les autres zones (introduction, bandeau de gauche et pied de page) ne changent ni de place ni d'aspect visuel ni de contenu. On n'a pas l'impression d'un bouleversement fondamental dès que le lecteur passe à une autre activité, et change dont d'URL.

- **On peut aussi placer un cadre (fixe ou variable) dans le bandeau de droite, mais il faut y réfléchir.**
- Certes, le bandeau de droite semble convenir parfaitement pour un carrousel de nouveautés ou pour les facettes, mais cette zone n'existe pas dans toutes les feuilles de style, et pose en outre un problème qu'on ne perçoit peut-être pas de prime abord.

- **Si on prévoit un bandeau de droite relativement large, on perd forcément de la place pour la partie centrale de la page**
  - **pas uniquement dans la page où on voit par exemple un cadre variable (comme un carrousel ou les facettes)**
  - **mais dans <u>toutes</u> les pages de l'OPAC ou du portail, même s'il n'y a aucun cadre dans le bandeau de droite**
- **Or l'écran n'est pas extensible !** Ce rétrécissement de la page centrale sera perceptible partout, et obligera par exemple le lecteur à scroller davantage pour parcourir une notice relativement détaillée.

- C'est pourquoi certains préfèrent laisser à la page centrale presque toute la largeur disponible (hormis le bandeau de gauche), et placer un carrousel dans la partie droite de la page centrale, et non pas dans le bandeau de droite.
- **A vous de peser le pour et le contre.**

Peut-on déplacer le menu ? Le mettre dans le bandeau de gauche plutôt que dans l'introduction ?

Voici le menu tel qu'il apparaît dans la zone introduction.
La rubrique Animations contient 2 sous-rubriques : Evénements et Heure du conte.

Le template qui gère ce menu est conçu dans la perspective où les rubriques sont affichées côte à côte, et les sous-rubriques en-dessous. Ceci permet d'avoir éventuellement d'autres sous-niveaux, qui viendraient à droite des titres Evènements et Heure du conte.
En effet, il faut de l'espace pour déployer en partie l'arbre du contenu éditorial.

D'un point de vue ergonomique, ce n'est pas une bonne idée de mettre un tel menu dans le bandeau de gauche, surtout s'il y a plusieurs rubriques et sous-rubriques.
Les sous-rubriques vont en effet masquer les titres des autres rubriques !
Bien entendu, cette remarque n'a pas une portée générale : elle concerne une feuille de style comme pmb4, dont le bandeau de gauche est assez étroit.

N.B. On peut aussi créer un menu avec le module Liste de rubriques.
En effet, on peut aussi obtenir un affichage ergonomique de l'arborescence en adaptant la feuille de style et le template lié à ce cadre (via des balises HTML).
Toutefois, il est nettement plus facile de créer un menu avec le module Menu !

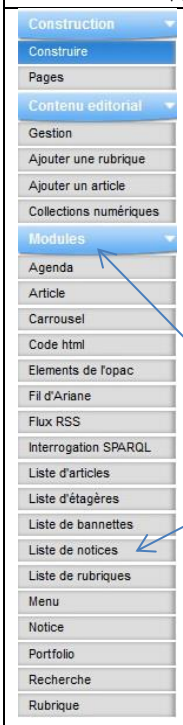
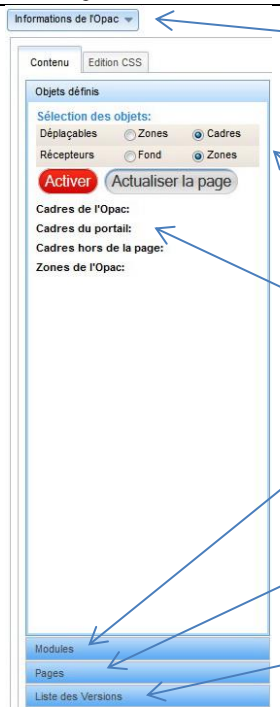
La zone introduction convient en général mieux pour des cadres fixes du portail, comme le menu ou la recherche.

Peut-on placer des cadres du portail dans les pages de l'OPAC ?
En théorie, on peut insérer les cadres du portail dans les pages du portail ou dans celles de l'OPAC.

- **Mais concrètement, on n'a guère de place disponible dans l'OPAC, si ce n'est dans la page d'accueil.** En effet, les autres pages de l'OPAC (recherche, recherche, affichage d'une liste de notices ou de périodiques …) sont déjà tellement remplies qu'on ne voit pas comment y ajouter un cadre supplémentaire (ex. un article complet) sans nuire à la lisibilité de l'ensemble.
- **En outre, même la page d'accueil de l'OPAC risque vite d'être en surcapacité** si elle contient déjà des liens vers les infopages, un onglet déplié de recherche, une liste assez longue de catégories et 3 étagères virtuelles.
- **Ce que vous pouvez en général ajouter sans trop de problème dans la page d'accueil de l'OPAC, c'est un carrousel de nouveautés ou un flux RSS**. Mais attention, n'en faites pas trop, cela deviendrait vite un fouillis.
- **Dès que vous voulez créer et étoffer le contenu éditorial, et le relier au contenu bibliographique, il faut envisager le passage au portail.** Afficher des articles complets ou des rubriques dans une page de l'OPAC est rarement une réussite ergonomique.
- **« Passer au portail » n'implique pas de remplacer tout de suite la page d'accueil classique de l'OPAC par une page portail, mais il y a un minimum requis pour exploiter valablement le contenu éditorial et le relier au contenu bibliographique** :
  - créer et configurer au moins les **2 « pages-outils » essentielles** du portail : la **page qui affiche une rubrique**, et la **page qui affiche un article** (accompagné éventuellement d'une liste de notices)
  - paramétrer un **menu basé sur l'arborescence de certaines rubriques**, afin de pouvoir naviguer dans le contenu éditorial, et d'accéder ainsi aisément à tous les articles publiés.
- **Vous pouvez dès lors ajouter certains cadres dans la page qui affiche une notice.** Deux modules conviennent tout à fait dans cette optique : **les articles dans la même catégorie que la notice, et les articles du même auteur que la notice**.
- **Au point où vous en êtes, pourquoi vous arrêter en si bon chemin ? Continuez !**
  - Si vous voulez un agenda ou un bloc de recherche dans les articles, la liste des pages nécessaires s'allonge encore : la page qui affiche une liste d'événements, et celle qui affiche le résultat d'une recherche dans le contenu éditorial.
  - Terminez en beauté : créez et fignolez la page d'accueil du portail ;-)

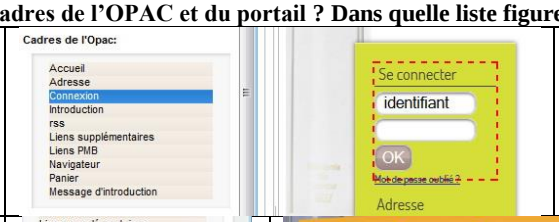
Identifier et repérer les zones et cadres

Comment accéder au menu qui permet de créer et d'éditer des cadres ?
Cliquer d'abord sur « Construire », cela ouvre un volet à droite

Menu latéral Version PMB 4.1		**Le menu des cadres se déplie automatiquement**	**Menu « replié » des cadres** J'ai replié les listes de cadres pour donner une vue d'ensemble	
	**Construire** **Créer ou éditer des pages** (= créer une page, changer son nom ou ajouter une variable d'environnement). Rien d'autre ! **Gérer le contenu éditorial et les collections numériques** **Administration des modules** Il s'agit seulement de **configurer les modules** !. **Cliquer ici sur le nom d'un module ne permet pas d'insérer dans une page un cadre lié au module choisi !**			**Informations de l'OPAC** URL de la page visible dans le volet de droite **Liste des objets définis** **Sélection des objets** **Activer le droit de déplacer des cadres** (et parfois aussi des zones) Listes des cadres et zones qu'on peut identifier et repérer. **Cadres de l'OPAC** **Cadres du portail** **Cadres hors de la page** **Zones de l'OPAC** **Modules** : **sélectionner un module et insérer ensuite un cadre dans une page** **Pages : sélectionner la bonne page du portail, celle dans laquelle on va insérer un cadre** **Liste des versions** : pour restaurer ou supprimer une version antérieure, ou pour activer le CMS dans l'OPAC

Comment repérer et identifier les cadres de l'OPAC et du portail ? Dans quelle liste figurent-ils ?

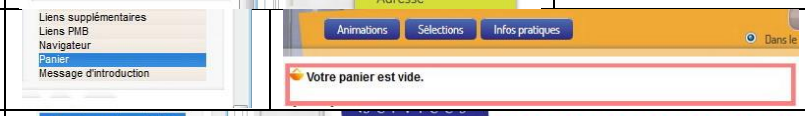
Si vous cliquez sur le libellé d'un cadre de l'OPAC, vous le repérez facilement dans le volet de droite, car ce cadre est entouré d'un **rectangle rouge**.



Liste des cadres et zones

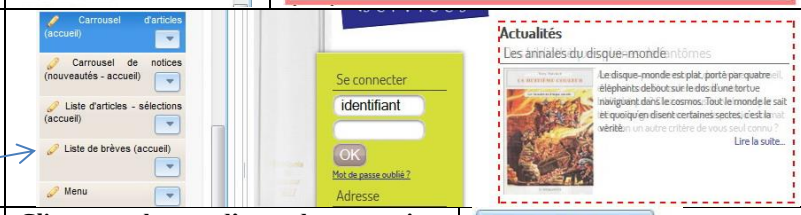
Cadres de l'Opac:
Cadres du portail:
Cadres hors de la page:
Zones de l'Opac:

N.B. Depuis la version 4.1.4 de PMB, les pointillés rouges ont été remplacés par un trait continu de la même couleur.



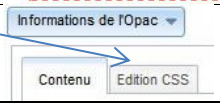
Pour voir (ou revoir) la liste des cadres, cliquez sur l'onglet Contenu.

Si vous cliquez sur le libellé d'un cadre du portail, vous le repérez facilement dans le volet de droite, car ce cadre est entouré d'un **rectangle rouge**.



Cliquez sur l'icône du crayon pour éditer le contenu d'un cadre (nom du cadre, dans quellle page, source de données, page de destination, template Django …)

Cliquez sur le nom d'un cadre et ensuite sur l'onglet Edition CSS pour vérifier ou modifier ses propriétés CSS



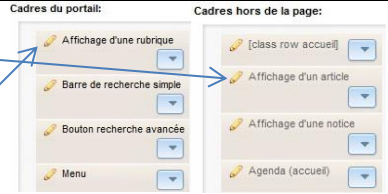
Attention ! Un cadre est parfois « un cadre du portail « dans une page, mais « un cadre hors de la page » dans une autre page !
Comparez par exemple les cadres qui affichent un article ou une rubrique dans ces deux types de pages.

Dès que vous naviguez dans le volet de droite, les infos visibles sous chaque titre (cadres de l'OPAC, du portail, cadres hors de la page) changent automatiquement.

En effet, la liste des cadres varie en fonction de l'URL de la page affichée à droite.
Essayez et vous verrez ;-)

Page qui affiche un article

Page qui affiche une rubrique



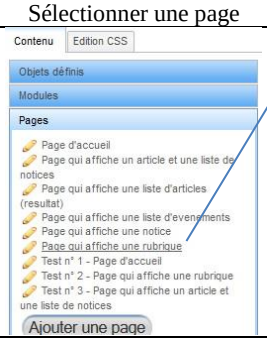
Les cadres du portail aboutissent « hors de la page » dans plusieurs cas :

- Cela peut être tout à fait **normal**, s'ils sont liés à une autre page que celle visible dans le volet de droite (voir exemple ci-dessus)
- Cela peut être lié à une **erreur de paramétrage**, si on ajoute un cadre et que ses conditions d'affichage ne correspondent pas au type de page sélectionnée, ou si on modifie les conditions d'affichage d'un cadre préexistant et que cela déclenche un problème
- Cela peut aussi révéler la **maladresse des débutants** … si on n'a pas bien « accroché » le cadre à une zone ;-)

N.B. Je reparlerai plus tard de ces cadres « hors de la page », au moment de vous expliquer comment manipuler les cadres.

Cette méthode de repérage permet aussi de voir des cadres présents sur les pages, même si ceux-ci ne sont pas directement visibles à l'écran.
Voir ce qui n'est pas directement visible à l'écran ? Oui, c'est possible, car il y a des zones qui semblent vides, mais qui ne le sont pas !

Sélectionner une page

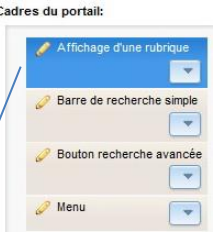


On voit dans le volet de droite la page qui affiche une rubrique

La zone « page centrale » a l'air aussi vide que le panier du lecteur.

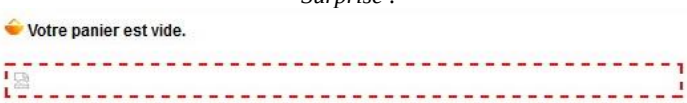
Liste des objets définis

Cliquer sur le nom du cadre



La zone « page centrale » de la page qui affiche une rubrique n'est en réalité pas aussi vide qu'on l'aurait cru, elle contient un cadre intitulé « affichage d'une rubrique ».
N.B. Les 3 autres cadres du portail mentionnés dans la liste sont situés dans la zone « introduction »

Surprise !



Distinguer les modules et les cadres

Liste de quelques modules et cadres du portail visibles dans la page d'accueil – dans les zones introduction et page centrale

En noir (police standard) : le nom du cadre | ***En rouge et en italiques : le nom du module***

Menu
Menu

Carrousel d'articles (accueil)
Liste d'articles

Liste d'articles - sélections (accueil)
Liste d'articles

Liste de brèves (accueil)
Liste d'articles

Ces exemples me serviront de fil confucteur pour expliquer quelques concepts fondamentaux liés aux modules et cadres : source de données, construction des liens, templates Django …

Barre de recherche simple
Recherche

Agenda (accueil)
Agenda

Carrousel de notices (nouveautés - accueil)
Liste de notices

N.B. Vous remarquez que ce carrousel n'a pas été créé avec le module « carrousel », mais avec le module « liste de notices ».

Comment savoir avec certitude quel module est à l'origine de tel ou tel ou cadre ?
Cliquer sur l'icône du crayon et éditer le cadre concerné.

Cadres du portail

En premier lieu, le nom du module qui a servi à créer le cadre | **Ensuite, le nom donné au cadre**

Surprenant de prime abord
Une liste de notices qui génère un carrousel

Modules
Modification de cadre - Liste de notices : Carrousel de notices (nouveautés - accueil)

Trois fois le même module (liste d'articles), mais des concrétisations différentes dans ces 3 cadres présents en page d'accueil.

Le nom donné au cadre contient des indications plus précises, afin de repérer tout de suite de quel cadre il s'agit, et dans quelle page il se trouve. On peut en effet avoir une liste d'articles ailleurs que dans la page d'accueil.

Modules
Modification de cadre - Liste d'articles : Carrousel d'articles (accueil)

Modules
Modification de cadre - Liste d'articles : Liste d'articles - sélections (accueil)

Modules
Modification de cadre - Liste d'articles : Liste de brèves (accueil)

Un module est un « élément de programmation », un outil qui permet de connecter divers aspects, et de définir ainsi le cadre concret qui sera créé au bout du processus.

- **La source de données : comment on va remplir le cadre**
  - Sources internes (contenu bibliographique ou éditorial) ou externes (flux RSS)
- **La navigation : comment on va accéder au cadre**
  - Par le menu principal, via un carrousel d'articles ou de notices, grâce au module recherche …
- **La vue : comment on va afficher les données dans le cadre**
  - quels champs afficher ? quelle mise en forme employer ?
  - la réponse aux 2 questions précédentes peut varier, on peut spécifier des conditions d'affichage dans le template Django
  - on emploie des langages spécifiques pour définir la mise en forme : HTML, CSS, templates Django, templates de notices

A partir du même module, on peut créer plusieurs cadres différents, selon le type de contenu, la valeur d'une variable globale d'environnement ou la valeur d'un champ personnalisé, selon le mode d'affichage (ex. liste ou carrousel) …

Il est donc essentiel de donner un nom adéquat au cadre, un nom plus précis que celui du module, car le même module peut être employé plusieurs fois dans la même page ou dans des pages différentes. Il faut éviter tout risque de confusion.

Un cadre est un élément visuel concret, de forme rectangulaire, qui a été défini à partir d'un module.

- **Il est fixe** (présent dans toutes les pages) **ou variable** (lié à un type spécifique de page)
- **Il peut être présent sur les pages de l'OPAC ou du portail**
- **Il doit être placé dans une zone précise et adéquate, et avoir des conditions d'affichage et des propriétés CSS cohérentes, sinon on ne le voit pas.** Il risque de disparaître dans les cadres hors de la page en cas d'erreur de paramétrage.

Modules disponibles

J'ai regroupé ces modules en diverses catégories, afin de donner une vue d'ensemble.
Les chapitres Modules, cadres et pages et Exemples supplémentaires contiennent des explications détaillées sur de nombreux modules.

PMB 4.0.13

- Construction
  - Construire
  - Pages
- Contenu editorial
  - Gestion
  - Ajouter une rubrique
  - Ajouter un article
- Modules
  - Agenda
  - Article
  - Liste d'articles
  - Fil d'Ariane
  - Carrousel
  - Code html
  - Menu
  - Notice
  - Liste de notices
  - Flux RSS
  - Recherche
  - Rubrique
  - Liste de rubriques
  - Liste d'étagères

PMB 4.1

- Construction
  - Construire
  - Pages
- Contenu editorial
  - Gestion
  - Ajouter une rubrique
  - Ajouter un article
  - Collections numériques
- Modules
  - Agenda
  - Article
  - Carrousel
  - Code html
  - Elements de l'opac
  - Fil d'Ariane
  - Flux RSS
  - Interrogation SPARQL
  - Liste d'articles
  - Liste d'étagères
  - Liste de bannettes
  - Liste de notices
  - Liste de rubriques
  - Menu
  - Notice
  - Portfolio
  - Recherche
  - Rubrique

Certains modules sont fondamentaux, car ils génèrent le cadre principal d'une des pages-outils du portail ; ils sont en général liés à une variable d'environnement

- Module rubrique → **variable id_rubrique** → page qui affiche **une rubrique**
- Module article → **variable id_article** → page qui affiche **un article**
- Module notice → **variable id_notice** → page qui affiche **une notice**
- Module agenda → **variable id_date** → page qui affiche **une liste d'événements**
- Module liste d'articles → **pas de variable** → page qui affiche le **résultat d'une recherche** dans le contenu éditorial : **une liste d'articles**

Listes présentées de manière « classique » ou en carrousel

- liste d'articles
- liste de notices
- liste de rubriques

} au choix : affichage sous forme de liste ou de carrousel (défilement horizontal, vertical, en fondu enchaîné)

- liste de bannettes
- liste d'étagères

} un seul mode d'affichage : la liste

- carrousel de notices → un seul mode d'affichage : le carrousel

Aide à la navigation

- menu créé sur base du contenu éditorial ou dans l'interface d'administration
- fil d'Ariane (fournir des liens en arrière sur les pages déjà parcourues)

Aide à la recherche interne ou externe

- recherche interne (dans le contenu bibliographique et/ou éditorial)
- recherche externe : interrogation SPARQL (accès au Web sémantique)

Insertion directe

- agenda (calendrier des activités organisées par la bibliothèque)
- code HTML (ajouter un cadre qui contient du code HTML)
- éléments de l'OPAC (ex. infopages, navigateur de périodiques)

Insertion d'éléments externes à la base

- flux RSS
- portfolio (gérer une collection de documents numériques non catalogués)

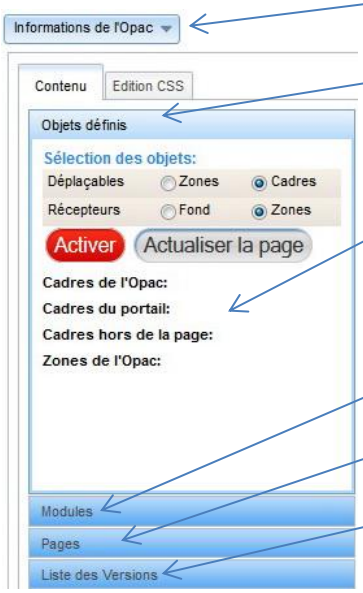
Interface de l'onglet Portail > Un clic sur « Construire » ouvre un volet à droite

Administration des modules

Un clic sur le nom d'un module (dans cette colonne) ne permet pas d'insérer dans une page un cadre lié au module choisi !!

Mais un clic sur le nom d'un module (dans cette colonne) donne accès à l'interface d'administration : on peut y configurer un module ou voir les vues qui y sont associées (templates Django).

N.B. 5 modules doivent être d'abord configurés, sinon ils ne sont pas utilisables :
Agenda
Interrogation SPARQL
Menu créé dans l'interface d'administration
Portfolio
Recherche



Informations de l'OPAC
URL de la page visible dans le volet de droite

Liste des objets définis

~~**Sélection des objets**~~
Activer le droit de déplacer des cadres (et parfois aussi des zones)

Listes des cadres et zones
Cadres de l'OPAC
Cadres du portail
Cadres hors de la page
Zones de l'OPAC

Modules : **sélectionner un module afin d'insérer ensuite un cadre dans une page**

Pages : sélectionner la bonne page du portail, celle dans laquelle on va insérer un cadre

Liste des versions :
pour restaurer ou supprimer une version antérieure ou pour activer le CMS dans l'OPAC

Administration des modules

Interface d'administration des modules

PMB 4.1

Si on clique sur le nom d'un module, on voit 0, 1 ou 2 boutons : Général et Vues.

Général Vues

Général permet de configurer le module - Vues donne accès à la gestion des templates

N.B. Rien pour les modules Code HTML et Eléments de l'OPAC : ni Général ni Vues.

Général

Le bouton Général n'est pas souvent affiché, car on peut employer la plupart des modules immédiatement.
Toutefois, il est indispensable de passer d'abord par l'interface d'administration pour 5 modules.
En effet, ils doivent être configurés avant usage.

- **Agenda**
  - Définir les calendriers
- **Interrogation SPARQL (recherche externe)**
  - Définir les sources de données
- **Menu : seulement s'il s'agit du menu créé dans l'interface d'administration**
  - Comme le nom l'indique clairement !
  - Inutile pour un menu basé sur le contenu éditorial (rubriques) : on peut l'employer tout de suite
- **Portfolio**
  - Pour ce module, il n'y a pas de bouton « Général », la configuration se fait ailleurs
  - Paramétrer un répertoire d'upload (onglet Administration) et définir les Collections numériques (onglet Portail)
- **Recherche interne**
  - Définir les types de recherche (contenu éditorial et bibliographique) et les pages de destination

N.B. J'explique comment paramétrer l'agenda, le menu créé dans l'interface d'administration et la recherche interne lorsque j'analyse ces modules en détail.

Vues

Presque tous les modules proposent de gérer les vues et les templates liés à ces vues.
Pas de « Vues » pour le code HTML, les éléments de l'OPAC et la recherche interne (articles - catalogue)

Les templates Django sont des scripts qui permettent de sélectionner les données à afficher, de personnaliser la mise en forme, la présentation de ces données. Ces templates gèrent aussi la navigation d'une page à l'autre (création d'URL et d'éléments cliquables), sur base de la réponse à la question « Construction du lien vers ? ».

Pour faciliter le travail de conception du portail, PMB a prévu une possibilité de sauvegarder ces templates Django et d'en faire des espèces de « modèles » à réutiliser, à améliorer.
N.B. Le chapitre Templates Django – Synthèse explique de manière détaillée comment gérer ces sauvegardes.

Général	Vues
Exemple : module Agenda **Définition des calendriers**	**Exemple : module Liste d'articles** **Liste des templates Django avec lien**

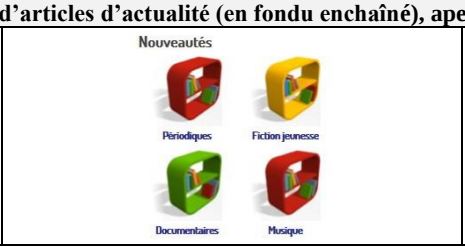
Analyser quelques exemples de cadres

Les formulaires de création des cadres sont tous bâtis sur le même modèle, et on pose les questions de base dans le même ordre.
Nom du cadre ? fixe ou non ? où ? page de l'OPAC ou du portail ? conditions d'affichage ? source de données ? combien d'éléments ? à trier dans quel ordre ? quel filtre ? quelle vue ? où aller ensuite ? quelle variable employer ? quel template ?
Bien entendu, les questions varient en fonction du type de cadre et du module qui l'a généré : liste d'articles ? liste de notices ? …
Il est donc important de se familiariser le plus vite possible avec ce mode de présentation, et avec ce vocabulaire.

J'ai réparti les cadres en plusieurs groupes « logiques », par exemple sur base du module qui a servi à les créer.
Ce mode de présentation permet de mieux comprendre « l'architecture du portail », l'interconnexion entre les éléments.

Cadres qui affichent une liste d'articles

Le module « liste d'articles » a généré des cadres de fonction et d'aspect différents. Voici quelques exemples.
Page d'accueil du portail : carrousel d'articles d'actualité (en fondu enchaîné), aperçu des nouveautés et liste de brèves.



Liste de brèves

Formulaire pour la liste de Brèves (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Dans cet exemple, il est attaché à une page.
Mode de mise en cache ?
Selon les GET, POST et vues OPAC
Conditions d'affichage ?
Aucune dans ce cas-ci

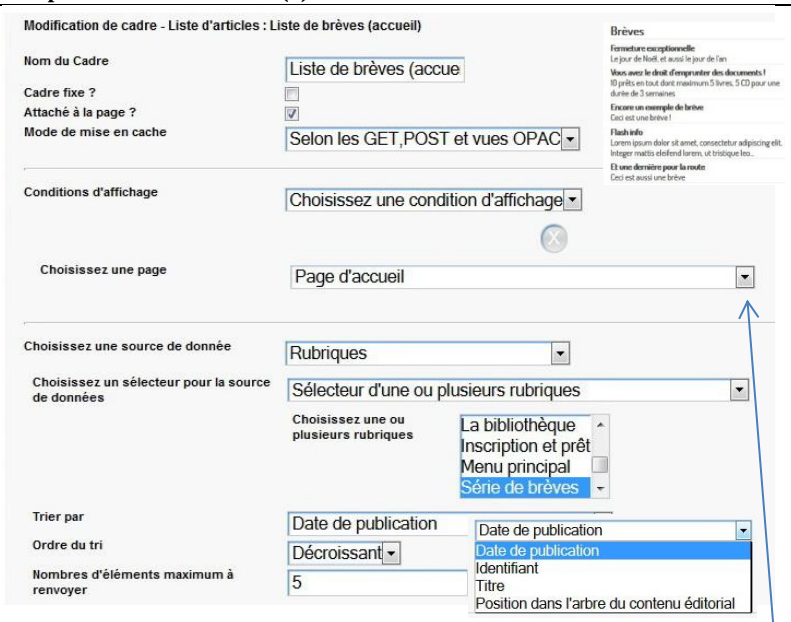
Dans quelle page afficher le cadre ?
Page d'accueil du portail

Source de données ? Sélecteur ?
Les articles d'une ou de plusieurs rubriques

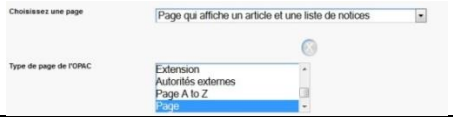
Critère et ordre de tri ?
Par date de publication - décroissant

Nombre maximum d'éléments à renvoyer ?
= nombre maximum d'articles à afficher

Surprenant mais vrai !
Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection (ex. appartenance à une rubrique).
Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.



Mode édition du cadre : comment choisir la bonne page ou en changer ?
Si on est en mode édition du cadre, on ne voit plus qu'**une seule option** : soit la liste des pages de l'OPAC soit la liste des pages du portail.
Ici, le choix se limite aux pages du portail, car c'était le type de page qui avait été retenu au départ, au moment de créer ce cadre. Si vous voulez placer ce cadre ailleurs, il faut donc sélectionner une autre page du portail dans la liste déroulante.



Mode création du cadre : comment choisir la bonne page ou en changer ?
Si on est en mode création du cadre, on voit les **deux options** : la liste des pages du portail ou la liste des pages de l'OPAC.
Le mieux est de naviguer dans le volet de droite, afin d'afficher d'abord la bonne page, elle sera ainsi présélectionnée automatiquement dans le formulaire.

Formulaire pour la liste de Brèves (2)

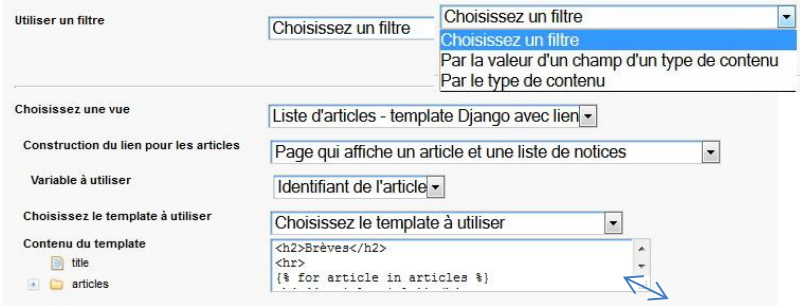
Filtre
Aucun filtre sélectionné
N.B. Dans tous les cadres liés aux listes d'articles, ce sont les mêmes filtres qui sont proposés.

Choisir une vue = comment afficher la liste ?
Liste d'articles – template Django avec lien
Autre possibilité : vue en carrousel

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?
Un template spécial a été défini pour les brèves.
Il ne prévoit aucun lien URL possible vers la page qui affiche un article, rien n'est cliquable.



Au départ, le cadre du template est « minuscule » !!!
Si vous amenez le curseur de la souris dans le coin inférieur droit, vous verrez une flèche bidirectionnelle, qui vous indique qu'il est possible d'agrandir ce cadre, afin de faciliter la lecture.

```
<h2>Brèves</h2>
<hr>
{% for article in articles %}
<h4>{{article.title}}</h4>
<blockquote>{{article.resume}}</blockquote>
<hr>
{% endfor %}
```

C'est vraiment mieux ainsi !

Construction du lien pour les articles = où aller ensuite ? vers quelle page ? comment ? grâce à quelle variable ? Page qui affiche un article et une liste de notices

En soi, la question est surprenante pour la liste de Brèves, car ce cadre ne mène à aucune autre page, rien n'est prévu dans le template Django pour assurer un lien éventuel. Mais c'est une question standard qui est posée pour tous les cadres qui affichent une liste d'articles.

La réponse peut elle aussi surprendre au début : pourquoi la page qui affiche un article et une liste de notices ?
Dans la démo du portail, il y a une seule page qui est prévue pour afficher un article, mais c'est une page multifonctions !
Cette page est **conçue pour afficher tous les types de contenu d'articles de la démo** :
Informations, Coup de cœur, Sélection, Evénement à une date précise, Evénement à une période précise

Il faut bien sûr transmettre une variable (un paramètre, si vous préférez) : le numéro identifiant de l'article.
Sinon, on irait sur la bonne page, celle qui affiche un article, mais PMB ne saurait pas quoi faire : quel article afficher ?

Template Django de la liste de Brèves

Résultat à l'écran		
	Brèves	On affiche le titre "Brèves"
	<hr>	Une 1e ligne horizontale
	{% for article in articles %}	Début de la boucle : **{% for article in articles %}**
	{{article.title}}	Permet d'afficher les articles prévus on a donc une "boucle" qui va du premier au dernier
	<blockquote>{{article.resume}}</blockquote>	On affiche le champ titre de l'article : {{article.title}} Ensuite le résumé (chapeau) de l'article : {{article.resume}}
	<hr>	Une autre ligne horizontale
	{% endfor %}	Fin de la boucle : **{% endfor %}**

Syntaxe des templates Django	**Les instructions entre {% %}** **Les champs entre {{ }}**	Exemple : la boucle de répétition {% for article in articles %} Exemple : le titre de l'article {{article.title}}
Explications des balises HTML	**Les balises <h2> et <h4>** reflètent la hiérarchie titre / sous-titre et déterminent la taille des caractères **La balise <hr>** affiche une ligne horizontale **La balise <blockquote>** détermine un mode d'affichage (bloc de citation)	

Les Brèves s'affichent en page d'accueil, et c'est tout. On n'affiche que 2 champs : le titre et le chapeau de l'article.
Ce template ne permet pas de créer un lien vers quoi que ce soit, car rien n'est cliquable, la balise <a href=" …> brille par son absence !

Félicitations ! Vous venez de comprendre le sens de votre premier template Django ;-)

Sélection de nouveautés

Formulaire pour les Nouveautés - symbolisées par des étagères (1)

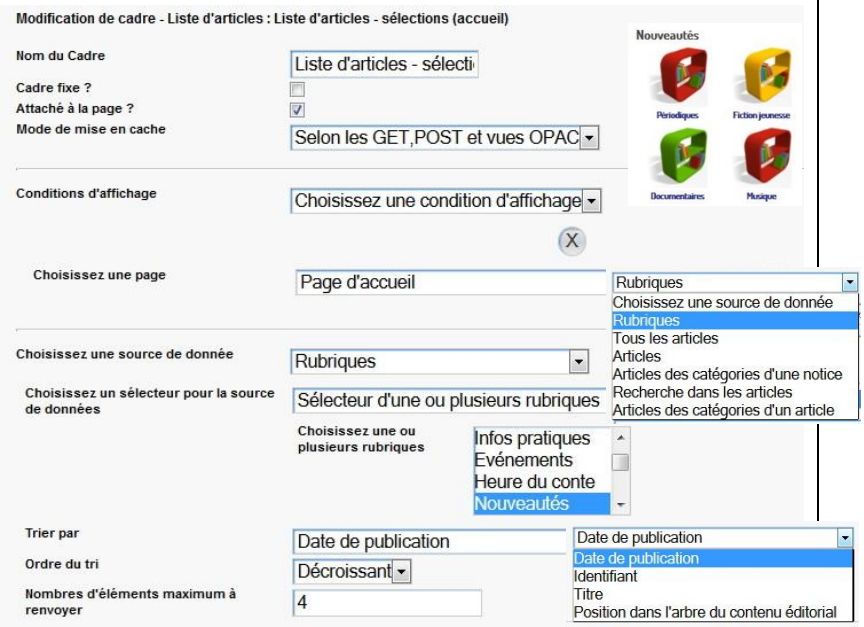
Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Attaché à la page.
Mode de mise en cache ?
Selon les GET, POST et vues OPAC
Conditions d'affichage ?
Aucune dans ce cas-ci
Dans quelle page afficher le cadre ?
Page d'accueil du portail

Source de données ? Sélecteur ?
Les articles d'une ou de plusieurs rubriques
Critère et ordre de tri ?
Par date de publication - décroissant

Nombre maximum d'éléments à renvoyer ?
= nombre maximum d'articles à afficher

Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection
(ex. appartenance à une rubrique).

Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.



N.B. La liste des sélecteurs proposés varie forcément en fonction de la source de données choisie au départ.

Les logos indiquent clairement que les articles qui y sont associés remplacent les étagères virtuelles de nouveautés.
Soyez donc cohérent ! Il faut que les articles choisis soient du type sélection, afin d'avoir un lien vers une étagère !

Formulaire pour les Nouveautés - symbolisées par des étagères (2)

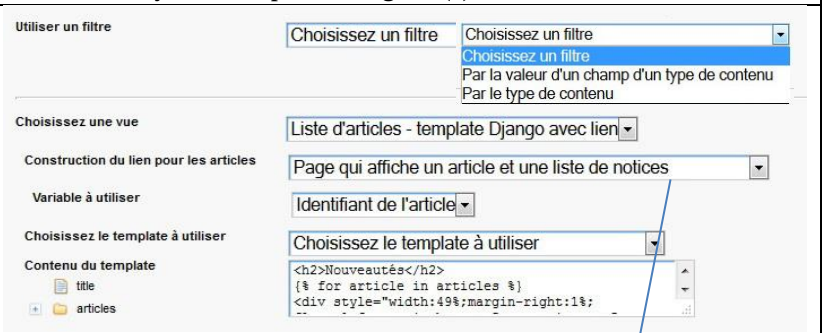
Filtre
Aucun filtre sélectionné

Choisir une vue = comment afficher la liste ?
Liste d'articles – template Django avec lien
Autre possibilité : vue en carrousel

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?
Le template affiche les logos et les titres des articles, et il gère aussi la navigation.



Clic sur le logo ou le titre d'une étagère (en page d'accueil) → on change de page.

Le même article, donc le même numéro d'id, affiché avec des mises en page différentes, dans 2 pages (page d'accueil – page qui affiche un article)

Puisque cet article est de type Sélection, il a un champ personnalisé qui contient le numéro d'id d'une étagère.
Donc, on peut voir (sur la page qui affiche un article) non seulement cet article, mais aussi les **notices liées à cette étagère**.

« Construction du lien vers … ? » : où aller ensuite ?
Un cadre issu du module « Liste d'articles » mène toujours vers la page qui affiche un article et une liste de notices.
Je rappelle que c'est une page multifonctions, conçue pour afficher tous les types de contenu d'articles de la démo.

Template Django des Nouveautés - symbolisées par des étagères		
Résultat à l'écran **Ces logos suggèrent clairement l'idée qu'on va revisiter le concept d'étagère virtuelle**. Ces étagères sont ici les logos des articles de type Sélection	Nouveautés {% for article in articles %} {{article.title}} {% endfor %} **Ce template s'appelle « Liste d'articles - template Django avec lien », ce n'est pas un hasard !** Il prévoit la possibilité de cliquer sur le titre ou le logo pour aller vers la page qui affiche un article.	On affiche le titre "Nouveautés" Début de la boucle : **{% for article in articles %}** Permet d'afficher les articles prévus on a donc une "boucle" qui va du premier au dernier Début de la balise <div style=" … qui gère la mise en forme Début de la balise <a href=" … (lien URL) D'abord l'URL de destination, le lien Ensuite, la « source » = d'où on part = où peut-on cliquer ? Sur le logo de l'article : {{article.logo.large}} Sur le titre de l'article : {{article.title}} Tout ce qui est situé entre le début et la fin de la balise sera donc cliquable Fin de la balise Fin de la balise Fin de la boucle : **{% endfor %}**
Syntaxe des templates Django	**Les instructions entre {% %}** **Les champs entre {{ }}**	Exemple : la boucle de répétition {% for article in articles %} Exemple : le titre de l'article {{article.title}}
Explications des balises HTML	**Les balises <h2> et <h4>** reflètent la hiérarchie titre / sous-titres et déterminent la taille des caractères **Les balises <div style="** … > et ** ** marquent le début et la fin d'un « bloc de texte » à mettre en forme. L'affichage du texte situé entre ces 2 balises peut être personnalisé : marges, alignement, police de caractères … **La balise <img>** signifie image. **<img … src=' …'>** signifie la source de l'image = l'URL de l'image.	

Risque de dégâts collatéraux ! Ne faites jamais de copier-coller d'un template Django depuis le traitement de texte !
C'est vraiment une très mauvaise idée, car Word ou OpenOffice insèrent des codes de mise en forme qui peuvent perturber le fonctionnement normal de ces templates, et aussi de PMB. En outre, vous risquez d'avoir des sauts de ligne au mauvais endroit.
Si vous voulez récupérer le contenu d'un template Django, faites un copier-coller à partir de la démo du portail.

Comment rendre une image cliquable en HTML ?	
<a href="… : URL de destination	**<img src="… : source, d'où on part : on clique sur l'image**
	
N.B. Pour les balises alt=" " et title=" ", employez des guillemets, mais pas l'apostrophe. **alt=" " : texte alternatif à afficher si l'image n'est pas disponible title=" " : texte à afficher au survol de la souris sur l'image** **L'apostrophe posera problème dès que le texte (dans le alt ou le title) contient des apostrophe.** **Modifiez donc quelques templates Django si nécessaire.** En effet, certains d'entre eux créent le contenu des balises alt ou title avec le titre d'une notice, or ces titres peuvent contenir des apostrophes, donc ils seraient tronqués dès la 1e apostrophe. ***L'Avare* de Molière se réduirait ainsi à la seule lettre "L" !!** Exemples corrects : **alt="{{record.title}}" ou title="{{record.title}}"**	
Comment rendre un texte cliquable en HTML ?	
<a href="… : URL de destination	**Texte entre <a> et : source, d'où on part : on clique sur le texte**
Site de Anne-Marie Cubat	
Comment rendre le texte et l'image cliquables en une seule balise HTML ? On combine le tout !	
Site de PMB	
C'est d'ailleurs ce qui se passe dans ce template Django et aussi dans d'autres : une seule balise <a href=" pour le texte et l'image	
 {{article.title}} **La variable {{article.link}} désigne l'URL de destination, son contenu est créé sur base de vos réponses.**	

Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page qui affiche un article – page dont l'id est 4** est dans la démo
- PMB sait qu'il faut **afficher certains articles** de certaines rubriques, et il connaît le **numéro identifiant de chaque article**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque article**, par exemple http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=14
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{article.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que le logo et le titre de l'article soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

Carrousel d'articles d'actualité

Formulaire pour le carrousel d'articles d'actualité (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Attaché à la page.
Mode de mise en cache ?
Selon les GET, POST et vues OPAC
Conditions d'affichage ?
Aucune dans ce cas-ci

Dans quelle page afficher le cadre ?
Page d'accueil du portail

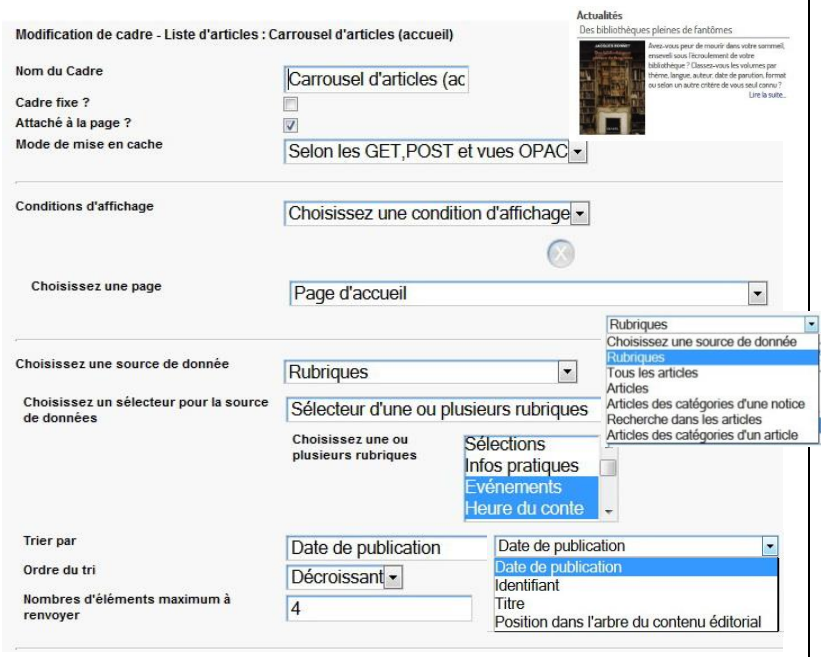
Source de données ? Sélecteur ?
Les articles d'une ou de plusieurs rubriques

Critère et ordre de tri ?
Par date de publication - décroissant

Nombre maximum d'éléments à renvoyer ?
= nombre maximum d'articles à afficher

Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection (ex. appartenance à une rubrique).

Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.



Dans ce cadre, les articles d'actualité sont extraits de **3 rubriques** : **Evénements, Heure du conte et Coups de cœur.** C'est donc l'actualité au sens large :

- Les animations (ex. Cultures d'Afrique contemporaine, Tim Burton)
- Les nouveautés « Coups de cœur » (ex. Bibliothèques, Georges)

Formulaire pour le carrousel d'articles d'actualité (2)

Filtre
Aucun filtre sélectionné

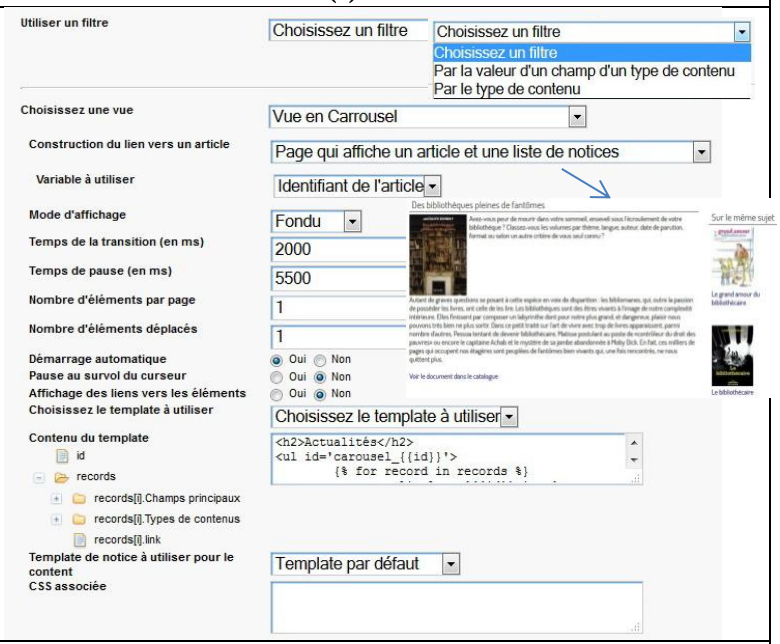
Choisir une vue = comment afficher la liste ?
Vue en carrousel
Autre possibilité : Liste d'articles – template Django + lien

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Mode d'affichage du carrousel ? Fondu enchaîné
Autres possibilités : défilement horizontal ou vertical
Définir les autres paramètres du carrousel

N.B. Pour la clarté, j'ai voulu montrer la 2e partie du formulaire dans son ensemble, mais je traite les questions sur les templates dans les encadrés suivants.



« Construction du lien vers … ? » : où aller ensuite ?
Un cadre issu du module « Liste d'articles » mène toujours vers la page qui affiche un article et une liste de notices.
Je rappelle que c'est une page multifonctions, conçue pour afficher tous les types de contenu d'articles de la démo.

Formulaire pour le carrousel d'articles d'actualité (3)

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Le template affiche les logos, les titres et les résumés des articles.

Le template gère aussi la navigation, car deux éléments sont cliquables (voir code HTML) :

- Le logo de l'article
- Le texte « Lire la suite »

Choisissez le template à utiliser

```
<h2>Actualités</h2>
<ul id='carousel_{{id}}'>
        {% for record in records %}
                <li class='{{id}}_item'
style="height:200px">
<h3>{{record.title}}</h3>
<a href='{{record.link}}' alt='{{record.title}}'
title="{{record.title}}"><img style="float:left;margin-
right:5px;max-height:200px;max-width:200px;"
src='{{record.logo.large}}' /></a>
{{record.resume}}
<a style="text-align:right;" href="
{{record.link}}">Lire la suite...</a>
                </li>
        {% endfor %}
</ul>
```

Template Django du carrousel d'articles d'actualité

Résultat à l'écran

Soit actualité bibliographique :
Coups de cœur

Soit actualité événementielle :
Animations

```
<h2>Actualités</h2>

<ul id='carousel_{{id}}'>

{% for record in records %}

  <li class='{{id}}_item' style="height:200px">

 <h3>{{record.title}}</h3>

  <a href='{{record.link}}' alt='{{record.title}}'
title="{{record.title}}"><img style="float:left;margin-
right:5px;max-height:200px;max-width:200px;"
src='{{record.logo.large}}' /></a>

{{record.resume}}

<a style="text-align:right;" href="{{record.link}}">Lire
la suite...</a>

  </li>

{% endfor %}

</ul>
```

N.B. Remplacez les apostrophes par des guillemets dans la balise alt='{{record.title}}'
Cette balise devient alt="{{record.title}}"

On affiche le titre "Actualités"

Début de la liste non numérotée du carrousel (

Début de la boucle : **{% for record in records %}**
Permet d'afficher les enregistrements (= articles) du carrousel

Début de la définition d'un élément de la liste (<li>)

Titre de l'enregistrement : {{record.title}}

Début de la 1e balise <a href=" … (lien URL)
URL de destination
On clique sur le logo : {{record.logo.large}}
Fin de la 1e balise

Résumé de l'enregistrement : {{record.resume}}

Début de la 2e balise <a href=" … (lien URL)
URL de destination
On clique sur le texte « Lire la suite »
Fin de la 2e balise

Fin de la définition de cet élément (

Fin de la boucle : **{% endfor %}**

Fin de la liste non numérotée (

Un carrousel (de notices, d'articles ou de rubriques) est toujours bâti sur le modèle de la boucle {% for record in records %}

Syntaxe des templates Django	**Les instructions entre {% %}** **Les champs entre {{ }}**	Exemple : la boucle de répétition {% for record in records %} Exemple : le titre de l'enregistrement {{record.title}}
Explications des balises HTML Explication pour la balise alt **alt="{{record.title}}"** Voir aide-mémoire - HTML	**Les balises et ** marquent le début et la fin d'une liste non ordonnée = non numérotée (ul = unordered list) Une liste non ordonnée peut prendre la forme d'une liste à puces, mais cela peut aussi signifier tout simplement une liste d'éléments non numérotés. C'est le cas ici, les articles se succèdent dans le carrousel en fondu enchaîné. **Les balises et ** marquent le début et la fin de chaque élément de la liste (li = list) **La balise <img>** signifie image. **<img … src=' …'>** signifie la source de l'image = l'URL de l'image.	

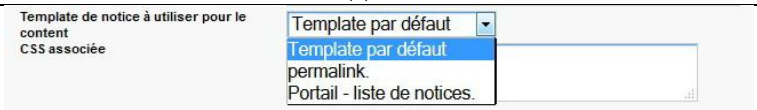
Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page qui affiche un article – page dont l'id est 4** est dans la démo
- PMB sait qu'il faut **afficher certains articles** de certaines rubriques, et il connaît le **numéro identifiant de chaque article**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque article**, par exemple
  http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=17
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{record.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que le logo et le texte « Lire la suite » soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

Formulaire pour le carrousel d'articles d'actualité (4)

Template de notice à utiliser pour le contenu ?
Template par défaut - Deux autres choix possibles
CSS associée ?
Aucun contenu



Quel template de notice employer ? N.B. Cette question concerne l'affichage des notices dans la page courante.
Un template de notice créé dans Editions > Templates de notices sert à personnaliser l'affichage du contenu du cadre.
Dans la démo, vous pouvez employer le template par défaut, ou « Portail – liste de notices » (visible dans l'onglet Editions).

Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}
- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :
- Si le template Django contient la **boucle {% for record in records %} ou bien** s'il faut **afficher une seule notice.**

Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :
- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template de notice employer ? » est sans objet, car il s'agit d'articles et non pas de notices !

Les carrousels (d'articles ou de notices) proposent une CSS associée.
On peut par exemple changer la couleur de fond, ou prévoir une bordure.
Evitez toutefois les fantaisies qui risqueraient de rompre l'harmonie visuelle de l'ensemble ! N'oubliez pas les niveaux de priorité de la CSS : la CSS que vous définiriez dans ce cadre-ci l'emporterait sur la CSS de la feuille de style (c'est toujours le dernier niveau qui l'emporte). Voir Cascading Style Sheets

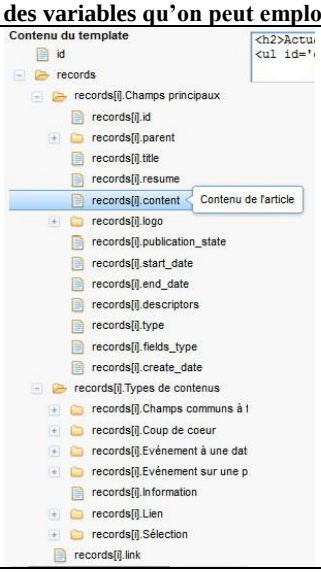
Comment connaître les noms des variables qu'on peut employer dans les templates Django ?

Voilà ce qu'on voit au départ

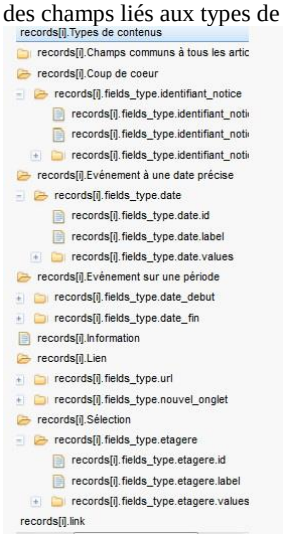
En cliquant sur les + et les -, on peut déplier ou replier cette arborescence.

C'est ainsi qu'on découvre la « longue » liste des champs qui existent, et donc des variables utilisables dans ce template.

Cerise sur le gâteau : au survol de la souris, vous avez même droit à une petite explication complémentaire, sous la forme d'une infobulle ;-)

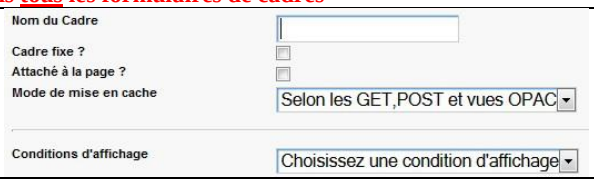


Aperçu des champs liés aux types de contenu



Les 4 premières questions sont identiques dans tous les formulaires de cadres

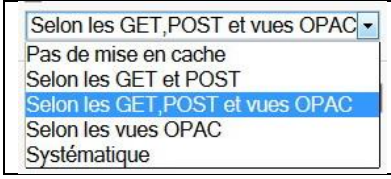
Nom précis donné au cadre ?
Fixe (présent dans toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?



Les 2 premières questions ne requièrent que peu d'explications.
- Puisqu'un module peut générer des cadres différents, **il faut donner au cadre un nom précis**, afin d'éviter toute confusion. Ce nom doit indiquer de quelle manière les données sont affichées et, si nécessaire, dans quelle page le cadre se trouve.
- **Un cadre est**
  - **fixe, c'est-à-dire présent dans toutes les pages** (ex. menu, barre de recherche, connexion au compte du lecteur)
  - **ou attaché à une page précise : une page du portail ou de l'OPAC**

Par contre, répondre de manière adéquate aux 2 dernières questions n'est pas évident !

Je voudrais remercier particulièrement Camille Bénard, qui a expliqué clairement les différences entre les options proposées.
Presque tous les commentaires de cette page et de la suivante sont « de sa plume », ou plutôt, « de son clavier » ;-)
Quant aux autres explications, je les ai rédigées sur base des infos fournies par Camille.

Mode de mise en cache (Camille Bénard)	
	**La mise en cache signifie que l'on garde le cadre en mémoire de façon à ne pas le charger une nouvelle fois lors de son prochain affichage.** Cependant certains cadres nécessitent d'être rechargés selon certaines conditions. C'est pourquoi il y a différents modes de cache :

- **Pas de mise en cache** : le cadre est rechargé pour chaque nouvelle page affichée, à recommander pour les cadres dont le contenu peut être mis à jour TRÈS fréquemment (par exemple pour le module flux RSS).

- **Selon les GET** : le cadre est rechargé lorsque l'on passe une nouvelle information (une variable) dans la page via l'URL.
  Exemple de GET : on a une "page qui affiche une rubrique" qui contient un cadre "rubrique" unique, dont le contenu affiché dépend de la rubrique sur laquelle on a cliqué. Donc dans une même page et un même cadre, on peut afficher des contenus différents, en fonction de la valeur d'une variable passée dans l'URL (ici l'identifiant de la rubrique), qui correspond à un GET. Il faut donc que le cadre "rubrique" soit rechargé selon les GET.

- **Selon les POST** : le cadre est rechargé lorsque l'on passe une nouvelle information dans la page via un formulaire.
  Exemple de POST : on a une "page qui affiche une liste d'articles (résultat d'une recherche)" qui contient un cadre "liste d'articles" unique, dont le contenu dépend des mots saisis dans le formulaire de recherche, ce qui correspond à un POST. Il faut donc que le cadre "liste d'articles" soit rechargé selon les POST.

- **Selon les vues OPAC** : le cadre est rechargé lorsque l'on change de vue OPAC.

- **Systématique** : le cadre est chargé lors du 1er affichage, ensuite il n'est plus chargé à nouveau au cours de la navigation (jusqu'à la fin de la durée de vie du cache définie dans le paramètre 'cache_ttl'). A recommander pour les cadres dont le contenu ne bouge pas, quelle que soit la page (les éléments fixes).

Le paramètre cache_ttl **Jusqu'à présent, je n'avais donné que ces infos succinctes à propos de la configuration de ce paramètre**	
cache_ttl	Lorsque vous construisez le portail, mettez cache_ttl à 0 pour voir plus vite l'impact des modifications. **N'oubliez pas de remettre la valeur par défaut (1800) lorsque vous repassez en mode de fonctionnement « normal »** N.B. Cette valeur correspond à la durée de vie du cache des cadres du portail (en secondes)

En phase de construction du portail, on donne à cache_ttl la valeur 0, ainsi les changements sont visibles immédiatement.

En situation réelle de production, il arrive souvent que la valeur de cache_ttl soit nettement plus élevée que 1800, surtout s'il s'agit d'une base assez volumineuse et que beaucoup de personnes consultent le portail en même temps.
Sinon, recharger de nombreuses fois les mêmes pages ralentirait nettement le « trafic ».

La table cms_cache_cadres
C'est dans cette table que certains cadres sont mis en cache, afin de ne pas devoir être rechargés lors du prochain affichage. Le nom donné à la table reflète particulièrement bien sa fonction.

C'est une table qui se remplit et se vide automatiquement :
- Elle se remplit en fonction des « GET, POST et Vues OPAC » (voir ci-dessus)
- Elle se vide plus ou moins vite en fonction de la durée de vie du cache = la valeur du paramètre cache_ttl

Si vous avez effectué des changements relativement importants dans le portail – par exemple
- Si vous avez ajouté ou modifié le contenu d'un article lié à un événement qui aura lieu bientôt
- Si vous avez ajouté ou modifié la date d'un événement dans l'agenda

Il est utile que les lecteurs soient prévenus le plus tôt possible.
Mais si la durée de vie de votre cache est d'une heure ou deux, les lecteurs ne verront pas tout de suite l'impact de ces changements.

Dans ce genre de circonstances, il vaut mieux vider le cache « manuellement ».
Une des requêtes SQL permet de vider le contenu de la table cms_cache_cadres (voir liste des utilitaires du portail).

Conditions d'affichage – Variables globales (Camille Bénard)	
Choisissez une condition d'affichage Choisissez une condition d'affichage Présence d'une variable globale Personnes authentifiées Page Type de page OPAC Valeur d'une variable globale Vues OPAC	**Le cadre ne sera affiché que si une condition spécifique est remplie.** **4 conditions sont aisées à comprendre** • Personnes authentifiées : lecteur qui a saisi un login et un mot de passe corrects • Page (il s'agit d'une des pages du portail) • Type de page OPAC • Vues OPAC **Mais les 2 conditions liées à une variable globale méritent quelques explications complémentaires.**

Les variables globales désignent des éléments qui composent l'URL. Ces conditions d'affichage sont utiles si l'on souhaite ajouter un cadre sur une page que l'on ne retrouve pas dans les conditions "page" et "type de page OPAC".

Si on prend l'exemple de l'URL d'un onglet de recherche prédéfinie :
http://.../pmb/ opac_css/index.php?search_type_asked=extended_search&onglet_persopac=3

Les 2 variables qui composent cette URL sont 'search_type_asked' et 'onglet_persopac'.
Elles ont respectivement pour valeur 'extended_search' et '3'.

Si on souhaite ajouter un cadre sur la page de cet onglet de recherche prédéfinie en particulier, on peut utiliser la condition "**valeur d'une variable globale**" ainsi : *Nom de la variable globale à récupérer = onglet_persopac* *Valeur à tester = 3*

Si on souhaite ajouter un même cadre sur les pages de tous les onglets de recherche prédéfinie, on peut utiliser la condition "**présence d'une variable globale**" ainsi : *Nom de la variable globale à récupérer = onglet_persopac*

Les 4 premières questions sont identiques dans tous les formulaires de cadres, et on propose le même éventail de réponses.	
Nom précis donné au cadre ? Fixe (présent dans toutes les pages) ou attaché à une page ? Mode de mise en cache ? Conditions d'affichage ?	Nom du Cadre Cadre fixe ? Attaché à la page ? Mode de mise en cache: Selon les GET,POST et vues OPAC Conditions d'affichage: Choisissez une condition d'affichage

Les 2 premières questions ne posaient pas de problème majeur.
Je viens de fournir des réponses détaillées aux 2 questions suivantes
Mode de mise en cache ? Conditions d'affichage ?

C'est pourquoi je présenterai désormais ces 4 questions de manière abrégée, il me semble inutile de tout répéter à chaque fois.

Cadres qui affichent une liste de notices

Le module « liste de notices » a généré des cadres de fonction et d'aspect différents. Voici quelques exemples.
Page d'accueil du portail : carrousel de notices - nouveautés en général (à droite de l'écran)
Page qui affiche un article : étagère associée à un article - nouveautés d'un rayon (ex. fiction jeunesse)
Page qui affiche un article : carrousel « sur le même sujet » - notices dans les mêmes catégories qu'un article

Carrousel de notices – les nouveautés en général

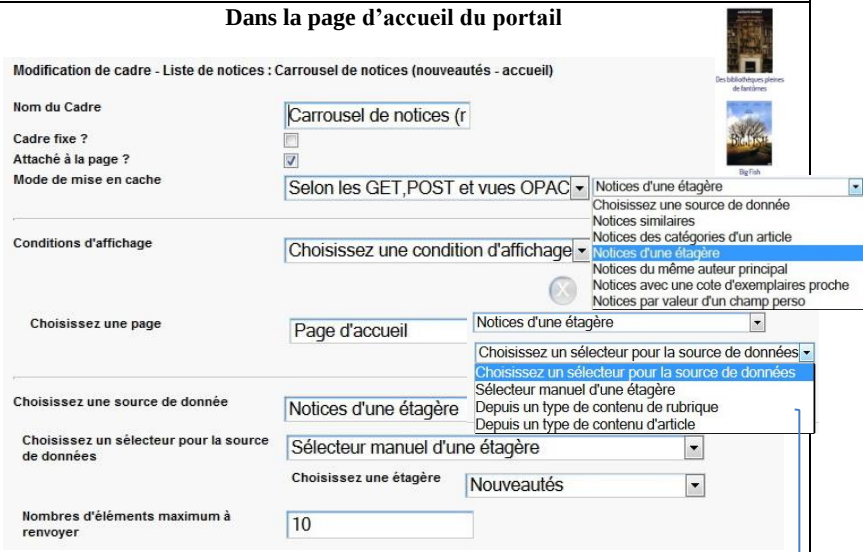
Formulaire pour le carrousel de nouveautés (1) – Créé à partir du module « Liste de notices »

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page d'accueil du portail
Source de données ? 6 possibilités
Ici, les notices d'une étagère
Sélecteur ?
Sélecteur manuel d'une étagère
Nombre maximum d'éléments à renvoyer ?
= nombre maximum de notices à afficher

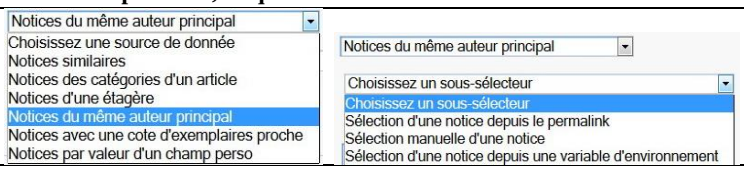
Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.
Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection.

Dans la page d'accueil du portail



Liste de notices – Ce module propose 6 sources de données possibles, ce qui influence ensuite le choix du sélecteur

Ex. Source de données = même auteur
Ces 3 possibilités de sélecteur sont proposées non seulement pour les notices du même auteur principal, mais aussi pour les notices similaires et pour celles dont la cote d'exemplaire est proche.



N.B. Notices similaires – Les champs concernés sont : titre propre, éditeur, indexation décimale, catégories et mots-clés libres.

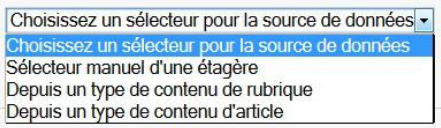
Comment créer un carrousel de notices ? Avec le module Carrousel ou avec le module Liste de notices ?

N.B. Ces deux modules permettent de créer les 3 types de carrousel : horizontal, vertical ou en fondu enchaîné.

Dans le cas du module Liste de notices, on peut créer des listes ou des carrousels.
En outre, on choisit d'abord une source de données (parmi 6), et ensuite un sélecteur.
→ On peut diversifier le mode d'affichage (liste ou carrousel) et la source de données.

Dans le cas du module Carrousel, on peut seulement créer un carrousel.
On ne choisit pas la source de données, c'est d'office « Notices d'une étagère ».
→ On ne peut pas diversifier le mode d'affichage (carrousel) ni la source de données.

Carrousel - une seule source de données
Notices d'une étagère



Conclusion : on peut certes créer un carrousel de notices avec le module Carrousel, mais l'éventail de choix est plus réduit.

Formulaire pour le carrousel de nouveautés (2) – Créé à partir du module « Liste de notices »

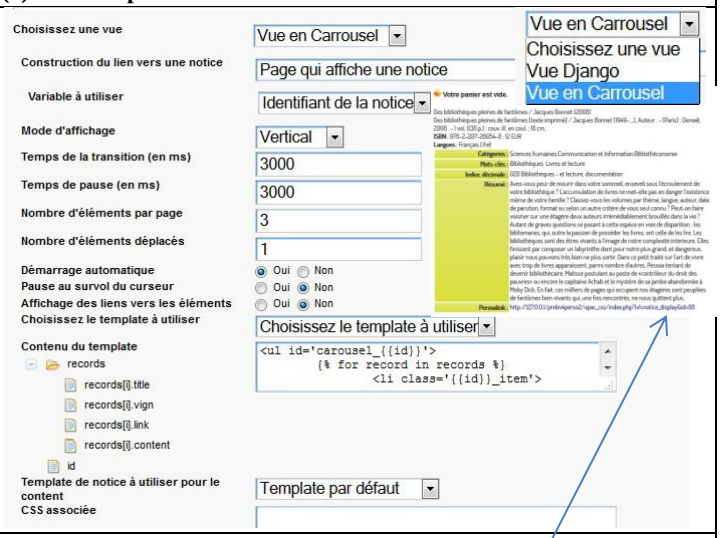
Choisir une vue = comment afficher la liste ?
Vue en carrousel
Autre possibilité : vue Django (liste)

Construction du lien vers une notice
= où aller ensuite ? vers quelle page ?
Page du portail qui affiche une notice

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la notice est ici la seule option.

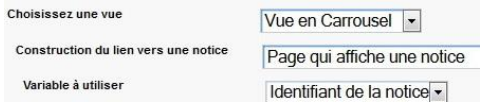
Mode d'affichage du carrousel ? Fondu enchaîné
Autres possibilités : défilement horizontal ou vertical
Définir les autres paramètres du carrousel

N.B. Pour la clarté, j'ai voulu montrer la 2e partie du formulaire dans son ensemble, mais je traite les questions sur les templates dans l'encadré suivant.



Dans ce type de module (liste de notices), la réponse à la question « Où aller ensuite ? » peut varier :
on peut aller vers la page OPAC « classique » qui affiche une notice, ou bien vers la page du portail qui affiche une notice.
Toutefois, dans la démo, on choisit toujours (dans le cas d'un carrousel) d'aller vers la page du portail qui affiche une notice.

Pour aller vers la page du portail qui affiche une notice,
on choisit la page adéquate dans la liste déroulante



Le lien vers la page du portail se fait par une URL avec variables

Pour aller directement vers la page de l'OPAC qui affiche une notice,
on ne choisit aucune des pages du portail qui sont proposées

Le lien vers l'OPAC se fait par le permalink de la notice

La page du portail qui affiche une notice propose (à la dernière ligne) un lien permanent (permalink) pour aller vers l'OPAC, donc on peut ensuite voir les infos habituelles sur la notice, sur les exemplaires et leur disponibilité. Mais la route est plus longue !
N.B. Si le permalink ne s'affiche pas, vérifiez le paramétrage de l'OPAC : le paramètre permalink doit avoir la valeur 1.

Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page du portail qui affiche une notice – page dont l'id est 7** est dans la démo
- PMB sait qu'il faut **afficher la notice correspondante,** et il connaît le **numéro identifiant de chaque notice**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque notice**, par exemple
  http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=88
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{record.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que la vignette et le titre de la notice soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

Dans la démo, on a choisi d'afficher la notice dans une page du portail, mais on aurait aussi pu l'afficher dans une page de l'OPAC, et l'URL de destination aurait correspondu au permalink : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=88

Formulaire pour le carrousel de nouveautés (3) – Créé à partir du module « Liste de notices »

Template Django à utiliser ? quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Le template affiche la vignette prévue et le titre de la notice.

Le template gère aussi la navigation, car ces deux éléments sont cliquables (voir code HTML) : la vignette du livre, CD, DVD … et le titre de cette notice

Choisissez le template à utiliser

```
<ul id='carousel_{{id}}'>
        {% for record in records %}
                <li class='{{id}}_item'>
                        <a href='{{record.link}}'
alt='{{record.title}}' title='{{record.title}}'
style="text-align:center;">
<img style="max-width:120px;max-height:120px;
text-align:center;" src='{{record.vign}}'/><br />
{{record.title}}
<br />
                        </a>
                </li>
        {% endfor %}
</ul>
```

Formulaire pour le carrousel de nouveautés (4) – Créé à partir du module « Liste de notices »

Template de notice à utiliser pour le contenu ?	CSS associée ?	
Template par défaut - Deux autres choix possibles	Aucun contenu	Template de notice à utiliser pour le content CSS associée Template par défaut Template par défaut permalink. Portail - liste de notices.

Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}

- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :

- Si le template Django contient la **boucle {% for record in records %}** **ou bien** s'il faut **afficher une seule notice.**

Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :

- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template employer ? » est sans objet, car on n'a pas employé la variable {{record.content}}

Template Django du carrousel de nouveautés

Résultat à l'écran

Des bibliothèques pleines de fantômes

Big Fish

```
<ul id='carousel_{{id}}'>

{% for record in records %}

<li class='{{id}}_item'>

  <a href='{{record.link}}' alt='{{record.title}}'
title='{{record.title}}' style="text-align:center;">

<img style="max-width:120px;max-height:120px;text-
align:center;" src='{{record.vign}}'/><br />
  {{record.title}}  <br />
 </a>

</li>

{% endfor %}
</ul>
```

N.B. Remplacez les apostrophes par des guillemets
alt="{{record.title}}" title="{{record.title}}"

- Début de la liste non numérotée du carrousel (
  - Début de la boucle : **{% for record in records %}**
    Permet d'afficher les vignettes des notices du carrousel
    - Début de la définition d'un élément de la liste (<li>)
      - Début de la balise <a href=" … (lien URL)
        URL de destination
        On clique sur la vignette : {{record.vign}}
        Ou on clique sur le titre de la notice : {{record.title}}
      - Fin de la balise
    - Fin de la définition de cet élément (
  - Fin de la boucle : **{% endfor %}**
- Fin de la liste non numérotée (

Un carrousel (de notices, d'articles ou de rubriques) est toujours bâti sur le modèle de la boucle {% for record in records %}

Syntaxe des templates Django	**Les instructions entre {% %}** **Les champs entre {{ }}**	Exemple : la boucle de répétition {% for record in records %} Exemple : le titre de l'enregistrement {{record.title}}
Explications des balises HTML Explication - balises alt et title Voir aide-mémoire - HTML	**Les balises et ** marquent le début et la fin d'une liste non ordonnée = non numérotée (ul = unordered list) Une liste non ordonnée peut prendre la forme d'une liste à puces, mais cela peut aussi signifier tout simplement une liste d'éléments non numérotés. C'est le cas ici, les notices se succèdent dans le carrousel vertical. **Les balises et ** marquent le début et la fin de chaque élément de la liste (li = list) **La balise <img>** signifie image. **<img … src=' …'>** signifie la source de l'image = l'URL de l'image.	

Petite adaptation du template Django du carrousel de nouveautés – La fonction limitstring

Template d'origine

N° 402 – août 2012 – Compostelle, Shikoku, Lalibela… Les chemins de la sagesse

Template modifié

N° 402 – août 2012 – Compostelle, Shikoku,…

```
<ul id='carousel_{{id}}'>

{% for record in records %}

<li class='{{id}}_item'>

 <a href='{{record.link}}' alt="{{record.title}}"
 title="{{record.title}}" style="text-
align:center;">
<img style="max-width:120px;max-
height:120px;text-align:center;"
src='{{record.vign}}'/><br />
  {{record.title | limitstring,45,"..." }} <br />
 </a>

</li>

{% endfor %}

</ul>
```

PMB a ajouté une fonction aux templates Django : limitstring
Elle permet de limiter la longueur d'une chaîne de caractères.

Dans le template d'origine {{record.title}}
Avec la fonction limitstring **{{record.title | limitstring,45,"..." }}**

Attention à la syntaxe ! Le tout doit être entre {{ }}
Il faut une barre verticale | entre la variable et limitstring

Cela signifie que la longueur du texte à afficher (le titre de la notice, record.title) **sera limitée à 45 caractères, et que les caractères manquants seront remplacés par le texte entre guillemets** (les 3 petits points). **Le découpage se fait à l'aveugle**, et le résultat final est plus ou moins heureux suivant le cas = suivant le titre à afficher.
Si vous avez une série de titres assez longs dans un carrousel de notices, cette fonction est fort utile. Bien entendu, son rôle n'est pas limité aux carrousels, elle peut aussi servir en d'autres circonstances.

REMARQUES GÉNÉRALES À PROPOS DES CARROUSELS

Ne prenez pas « au pied de la lettre » les valeurs maximales liées au nombre d'éléments, n'oubliez pas que tout dépend en premier lieu de la taille de la zone où vous affichez le carrousel, et des dimensions que vous lui avez données !
Il se peut par exemple que dans votre cas, le nombre idéal d'éléments à afficher par page soit inférieur aux valeurs suggérées ci-dessous, car votre carrousel est de taille assez réduite.

Pour des raisons d'ergonomie et de lisibilité, il est vivement conseillé de

- Mettre un seul carrousel par page – définir ses dimensions en fonction du nombre d'éléments à afficher en une fois (par page)
- Limiter le nombre d'éléments qui défilent horizontalement ou verticalement (10-15 maximum)
- Définir des temps de transition et de pause entre 2 et 10 secondes (donc entre 2000 et 10000 ms)
- Limiter le nombre d'éléments à afficher en une fois (par page)
  - dans un carrousel horizontal (max. 4-5) ou vertical (max. 3-4)
  - dans un carrousel en fondu enchaîné : un seul !!!
- Limiter le nombre d'éléments à déplacer en fonction du nombre d'éléments affichés et des temps de transition! (ex. 1 ou 2)

Les carrousels (d'articles ou de notices) proposent une CSS associée.
On peut par exemple changer la couleur de fond, ou prévoir une bordure. Evitez toutefois les fantaisies qui risqueraient de rompre l'harmonie visuelle de l'ensemble ! N'oubliez pas les niveaux de priorité de la CSS : la CSS que vous définiriez dans ce cadre l'emporterait sur la CSS de la feuille de style (c'est toujours le dernier niveau qui l'emporte). Voir Cascading Style Sheets

REMARQUES GÉNÉRALES À PROPOS DES VIGNETTES

Pré-requis : il faut que l'extension Php curl ait été activée dans la configuration de Php (le fichier php.ini).

En cas de doute, si vous travaillez en mode local, vérifiez la configuration du serveur (par exemple EasyPhp, Wamp ou Xampp).
Sinon, contactez le service informatique de votre institution ou votre hébergeur.

Comment savoir si une extension Php est activée ou non ?

Cherchez (à la fin du fichier php.ini) les lignes qui commencent par extension = …, et regardez le début de la ligne.
S'il y a un point-virgule au début, c'est une ligne de commentaire qui ne sera pas exécutée, donc l'extension ne sera pas activée.
S'il n'y a pas de point-virgule au début, c'est une ligne de commande qui sera exécutée, donc l'extension sera activée.
Dans le cas de extension=php_curl.dll, vous devrez donc supprimer le point-virgule qui se trouverait éventuellement en début de ligne.

Où chercher ces vignettes qui représentent les couvertures des livres ? 3 options possibles

- A partir d'un répertoire spécifique, le site de Amazon
- A partir d'un répertoire spécifique propre à votre institution, un répertoire que vous avez créé et rendu accessible sur Internet
- Quelque part sur Internet, « là où elles se trouvent actuellement ». Dans ce cas, il faut compléter le champ « URL de la vignette » lorsqu'on catalogue la notice. Ce champ se trouve tout à la fin de la liste, dans le bloc « Informations de gestion ».

N.B. On peut combiner les options 1 et 3, ou bien les options 2 et 3. Mais les options 1 et 2 s'excluent mutuellement !
Options 1 ou 2 : l'URL de la vignette est créée de manière dynamique, à partir de l'ISBN de la notice (sans les tirets).
Option 3 : l'URL de la vignette est « statique », il faut l'enregistrer préalablement dans un des champs de la notice.

Il faut paramétrer PMB à deux niveaux : en mode gestion et pour l'OPAC.

A chaque fois, il faut préciser s'il faut afficher les vignettes (oui ou non), et dans quel répertoire il faut éventuellement aller les chercher (sur base de l'ISBN) : sur le site de Amazon ou dans votre répertoire spécifique.

Infos complémentaires : http://amcubat.be/docpmb/afficher-vignettes-couvertures

Chemins d'accès :	Administration > Outils > Paramètres généraux		Administration > Outils > Paramètres de l'OPAC
	Paramètre	**Valeurs**	**Signification**
Paramètres généraux	book_pics_show	0 ou 1	0 : ne pas afficher les vignettes 1 : afficher les vignettes
Option 1 : Amazon	book_pics_url	http://images-eu.amazon.com/images/P/!!isbn!!.08.MZZZZZZZ.jpg	
Option 2 : votre répertoire spécifique	book_pics_url	http://mon_site/pmb/opac_css/images/vignettes/!!isbn!!.jpg http://127.0.0.1/pmb/opac_css/images/vignettes/!!isbn!!.jpg	Ce sont bien sûr des exemples
Paramètres de l'OPAC	show_book_pics	0 ou 1	0 : ne pas afficher les vignettes 1 : afficher les vignettes
Option 1 : Amazon	book_pics_url	http://images-eu.amazon.com/images/P/!!isbn!!.08.MZZZZZZZ.jpg	
Option 2 : votre répertoire spécifique	book_pics_url	http://mon_site/pmb/opac_css/images/vignettes/!!isbn!!.jpg http:// 127.0.0.1/pmb/opac_css/images/vignettes/!!isbn!!.jpg	Ce sont bien sûr des exemples
Ce tableau montre clairement pourquoi les options 1 et 2 s'excluent mutuellement : le paramètre book_pics_url ne peut avoir qu'un seul contenu !			

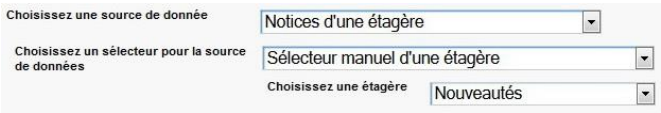
NOTICES D'UNE ÉTAGÈRE ASSOCIÉE À UN ARTICLE : NOUVEAUTÉS D'UN RAYON (EX. FICTION JEUNESSE)

Attention au sens du titre ! La nuance est importante.

- J'ai écrit « Nouveautés **d'un** rayon (exemple fiction jeunesse) »
- Je n'ai pas écrit « Nouveautés **du** rayon fiction jeunesse »

Je commence par des captures d'écran pour situer le contexte, et clarifier ainsi les explications qui suivront.

Rappel visuel : comment accède-t-on à ce cadre ? (Nouveautés, par exemple fiction jeunesse) On clique sur une des étagères de la **page d'accueil**. Cela mène à la **page qui affiche un article.** N.B. Les logos suggèrent clairement l'idée qu'on va revisiter le concept d'étagère virtuelle.		**Attention ! Ces logos d'étagères ne sont pas de simples images, ce sont des logos d'articles !** Le template (en page d'accueil) prévoit d'afficher seulement le logo et le titre de l'article. (Si nécessaire, relisez l'analyse du cadre Nouveautés dans la série des Cadres issus du module Liste d'articles)
Deux exemples pour la page qui affiche un article : les nouveautés du rayon cinéma - les nouveautés du rayon fiction jeunesse		
	Le même article, (donc le même numéro d'id) **affiché avec des mises en page différentes, dans 2 pages :** la page d'accueil du portail, et la page qui affiche un article et une liste de notices.	
Cette page qui affiche un article comprend trois cadres, dont deux sont visibles dans la capture d'écran ci-dessus : • **Un cadre issu du module Affichage d'un article :** un bref article de type sélection (ex. Cinéma ou Fiction jeunesse) • **Le cadre « Nouveautés du rayon … » - Etagère associée à un article** : une liste de notices – cadre que j'analyse maintenant		

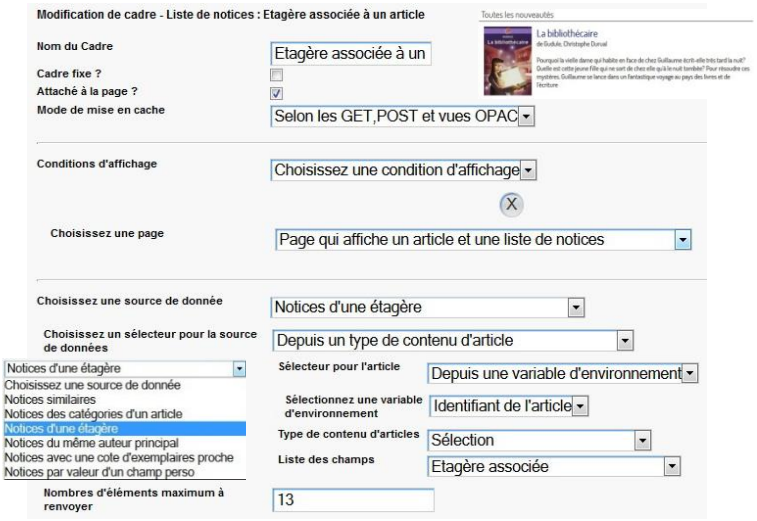
Comparaison entre deux cadres issus du module « Liste de notices »	
Dans les deux cas, la source de données est la même : les notices d'une étagère, mais le choix du sélecteur est différent. Pourquoi ?	
Page d'accueil Carrousel de nouveautés	**Page qui affiche un article et une liste de notices Nouveautés d'un rayon (ex. fiction jeunesse)**
Ce carrousel présente des nouveautés présélectionnées par le bibliothécaire, et probablement choisies dans divers rayons. Il est donc logique de lier l'ensemble du carrousel au contenu d'une étagère précise, alimentée par un panier spécifique. **Le contenu de ce carrousel est entièrement prévisible dès le début, au moment où on crée le cadre.** **En effet, la liste des notices à afficher dépend uniquement des choix effectués auparavant par le bibliothécaire : quelle étagère sélectionner pour le carrousel ? Quel panier va alimenter cette étagère ?**  **Ce genre de cadre ne doit pas être polyvalent, le lecteur ne peut de toute façon pas interagir.**	Cette liste présente des nouveautés présélectionnées par le bibliothécaire, et choisies dans un seul rayon (ex. jeunesse, cinéma). Il est donc logique de lier chaque article de type sélection au contenu d'une étagère précise, alimentée par un panier spécifique. Mais un autre aspect intervient. En fonction du choix effectué par le lecteur en page d'accueil, il faut afficher dans cette page l'article et la liste de nouveautés qui correspondent à l'étagère sur laquelle le lecteur a cliqué. **Le contenu de cette liste n'est pas entièrement prévisible dès le début, au moment où on crée le cadre.** **En effet, la liste des notices à afficher dépend :** • **des choix effectués auparavant par le bibliothécaire : quelle étagère associer à chaque article ?** • **mais aussi du choix effectué au moment même par le lecteur : sur quelle étagère cliquer en page d'accueil ?** **Ce genre de cadre doit être polyvalent, il doit s'adapter aux décisions prises par le lecteur**.
Carrousel : on présente les nouveautés de l'étagère que le bibliothécaire a préalablement sélectionnée et alimentée. Liste de nouveautés : on ne présente pas les nouveautés du rayon jeunesse, mais les nouveautés d'un rayon choisi par le lecteur.	

Formulaire pour les nouveautés d'un rayon – par exemple fiction jeunesse (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche un article et une liste de notices
Source de données ? 6 possibilités
Notices d'une étagère
Sélecteur ? (voir explications dans l'encadré suivant)
Depuis un type de contenu d'article.
Nombre maximum d'éléments à renvoyer ?
= nombre maximum de notices à afficher

Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.
Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection.



Formulaire pour les nouveautés d'un rayon – par exemple fiction jeunesse (2)

Choix du sélecteur pour la source de données et autres choix à effectuer

Cette série de choix est liée à l'objectif qu'on s'est fixé : si on clique sur une des étagères en page d'accueil, il faut voir (dans la page qui affiche un article) l'article qui correspond à l'étagère sélectionnée, et en-dessous, la liste des notices de cette étagère.

Un petit rappel : pages qui ont une variable d'environnement - types de contenu

Certaines pages sont liées à une variable d'environnement, entre autres la page qui affiche un article et une liste de notices.
Voir Créer les pages du portail

	A	B	C	D	E
1	Id (page)	Nom de la page	Id (variable)	Variable globale	Libellé
2	4	Page qui affiche un article et une liste de notices	6	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	4	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	7	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	10	id_date	Identifiant de date

Parmi les types de contenu définis pour les articles, un seul convient dans ce cas : sélection. (voir Configuration du contenu éditorial)
En effet, un article de type sélection a un champ personnalisé, qui permet d'associer l'article à l'id d'une étagère.

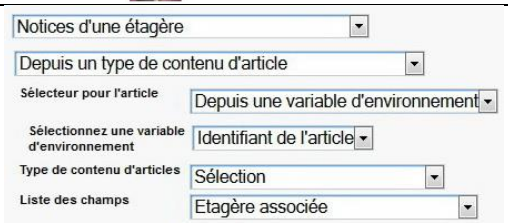
	A	B	C	D	E	F
1	Type de contenu	Id	Titre (affichage)	Champ (nom SQL)	Type de saisie	Type de données
2	Sélection	1	Etagère associée	etagere	Liste à partir d'une requête	Entier
3	Evénement sur une période	2	Date de début	date_debut	Sélection de date	Date
4	Evénement sur une période	3	Date de fin	date_fin	Sélection de date	Date
5	Coup de coeur	4	Identifiant de la notice	identifiant_notice	Texte (saisie au clavier)	Texte court
6	Evénement à une date précise	5	Date de l'événement	date	Sélection de date	Date

Le lecteur clique sur un logo d'étagère en page d'accueil.

N'oubliez qu'il s'agit en fait du logo d'un article, et que c'est le même article qu'on voit dans la page d'accueil et dans la page qui affiche un article et une liste de notices. Le numéro d'id de l'article est donc connu dès le départ.

Puisqu'on va vers la page qui affiche un article, on peut transmettre un paramètre via la variable d'environnement : l'identifiant de l'article (id_article), afin que PMB sache quel article afficher dans cette page.

En outre, puisque cet article est théoriquement de type sélection (sauf erreur du bibliothécaire !), on va disposer d'un 2e paramètre : cet article contient en effet (dans ses propres champs personnalisés) le numéro identifiant de l'étagère que vous avez sélectionnée au moment de rédiger l'article.



Tout ceci permet d'afficher le « bon » article, et la liste des notices contenues dans l'étagère choisie par le lecteur !

Cet enchaînement de choix et de variables rend le système polyvalent : on peut présenter les nouveautés d'un rayon, quel qu'il soit.
Tout se résume en fait à une question de paramétrage correct !

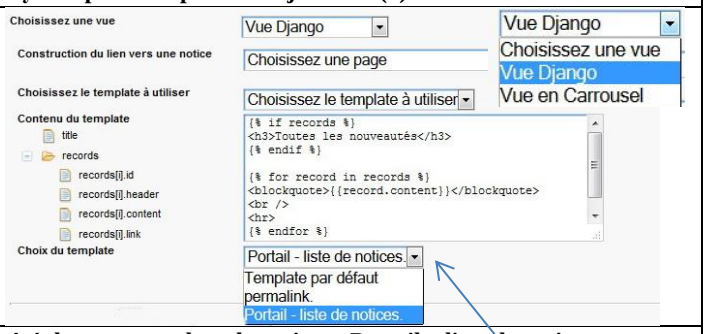
Ceci permet d'enrichir le concept de l'étagère virtuelle.
Elle ne se limite plus nécessairement à une simple liste de notices, elle inclut aussi la possibilité d'expliquer, dans le texte de l'article, les spécificités de l'étagère, les points communs entre les notices : thème, auteur, lien avec une activité …

Formulaire pour les nouveautés d'un rayon – par exemple fiction jeunesse (3)

Choisir une vue = comment afficher la liste ?
Vue Django - Autre possibilité : Vue en carrousel

Construction du lien vers une notice
= où aller ensuite ? vers quelle page ?
Apparemment, nulle part. Mais ce n'est pas tout à fait vrai …
En tout cas, aucune page du portail n'est sélectionnée, ce qui ne laisse qu'une seule issue : aller vers la page de l'OPAC qui affiche une notice !

Du coup, la série suivante de questions n'est pas posée.
Que faire dans cette page ? Comment le signaler → Quelle variable ?

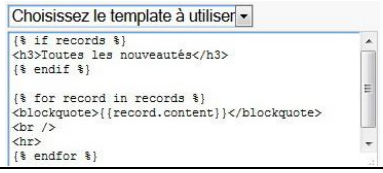


Le choix de la page de destination et des moyens d'y accéder est géré dans un template de notices : Portail – liste de notices.

Formulaire pour les nouveautés d'un rayon – par exemple fiction jeunesse (4)

Template Django à utiliser ? Quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Le template affiche éventuellement le titre « Toutes les nouveautés ».
Ensuite, on a une boucle (du premier au dernier enregistrement) qui affiche le **contenu de la notice : record.content.**



Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}
- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :
- Si le template Django contient la **boucle {% for record in records %}** **ou bien** s'il faut **afficher une seule notice.**

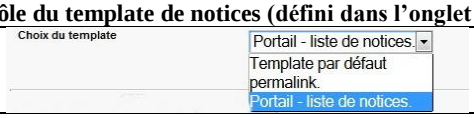
Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :
- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template de notice employer ? » a un sens, car toutes les conditions sont vérifiées.

Ce template Django semble plutôt « vide », en tout cas, il ne répond pas à certaines questions fondamentales :
- il ne précise pas **quels champs** afficher (record.content désigne tout le contenu de la notice)
- il ne signale pas **comment mettre en forme les données** (si ce n'est le fait de tracer une ligne horizontale)
- il ne prévoit pas de **liens pour aller ensuite « ailleurs »**, vers la page qui affiche une notice (version portail ou OPAC)

Il faut donc une possibilité de répondre à ces questions vitales, ce sera le rôle du template de notices (défini dans l'onglet Editions).

Template de notice à utiliser pour le contenu ?
Portail – Liste de notices
Deux autres choix possibles



Template Django des nouveautés d'un rayon – par exemple fiction jeunesse

Résultat à l'écran

```
{% if records %}
  <h3>Toutes les nouveautés</h3>
{% endif %}

{% for record in records %}
  <blockquote>{{record.content}}</blockquote>
  <br />
  <hr>
{% endfor %}
```

Début de la condition **{% if records %}**
S'il y a des notices à afficher
On affiche le titre « Toutes les nouveautés »
Fin de la condition **{% endif %}**

Début de la boucle : **{% for record in records %}**
Afficher le contenu de la notice {{record.content}}
Saut de ligne
On trace une ligne horizontale
Fin de la boucle : **{% endfor %}**

Une liste de notices (vue Django ou carrousel) est toujours bâtie sur le modèle de la boucle {% for record in records %}

Syntaxe des templates Django	**Les instructions entre {% %}** la boucle de répétition {% for record in records %} la condition {% if records %} {% endif %} **Les champs entre {{ }}** le contenu de la notice {{record.content}}	**La condition {% if records %} est nécessaire ici** **Elle signifie : s'il y a des enregistrements = notices** Inutile d'afficher le titre « Toutes les nouveautés » s'il n'y a pas d'étagère associée à l'article ou si elle est vide !
Explications des balises HTML	**La balise ** provoque un saut de ligne (br = break) **La balise <hr>** trace une ligne horizontale (hr = horizontal rule) **La balise <blockquote>** détermine un mode d'affichage (citation)	**{{record.content}} = tout le contenu de la notice !** C'est trop ! Via le template de notices, il faut réduire le nombre de champs, et préciser comment les afficher.

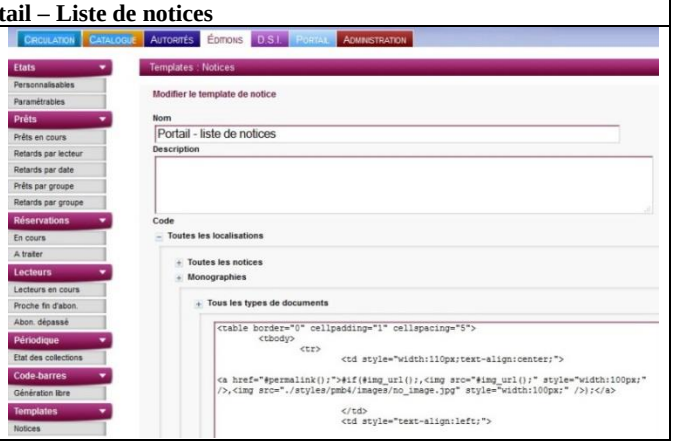
Template de notices : Portail – Liste de notices

Ce genre de template se crée dans l'onglet Editions
Editions > Templates > Notices
Il permet de définir un format d'affichage pour les notices.

Chaque template contient des fonctions spécifiques, qui font référence aux champs de la base de données, du code HTML pour la mise en forme et parfois quelques mots à afficher.

Ce template comprend en fait 4 parties : monographies, articles de périodiques, périodiques (notices-mères) et notices de bulletins.

Voir le guide en ligne de PMB
http://doc.sigb.net/pmb/co/edit_tpl_notice.html
http://doc.sigb.net/pmb/co/edit_tpl_notice_fonctions.html



Portail – Liste de notices – Le template des monographies gère cette présentation succincte de la notice

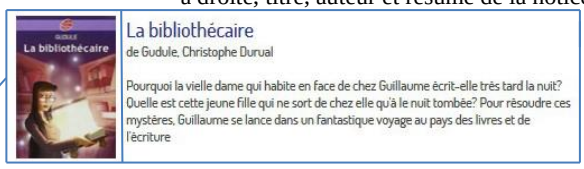
Explication préliminaire pour les balises HTML liées aux tableaux
Les balises <table> et </table> marquent le début et la fin d'un tableau.
Les balises <tbody> et </tbody> marquent le début et la fin de la définition du tableau.

Les balises <tr> et </tr> marquent le début et la fin de chaque ligne
Les balises <td> et </td> marquent le début et la fin de chaque cellule.

<table>
<tr> Début de l'unique ligne de ce tableau
<td> 1e cellule : la vignette </td>
<td> 2e cellule : titre, auteur et résumé de la notice </td>
</tr> Fin de la ligne
</table>

On ne le croirait pas à première vue, mais cette notice est affichée dans un tableau dont les bordures sont invisibles !
Je trace ces bordures pour clarifier les choses ;-)

Ce tableau (défini dans le template) comprend
une seule ligne : l'ensemble des infos sur la notice
deux cellules : à gauche, la vignette
à droite, titre, auteur et résumé de la notice



Où va-t-on ensuite si on clique sur un lien : sur la vignette ou sur le titre ?
La balise est <a href="#permalink();"> L'URL de destination est donc la page de l'OPAC qui affiche une notice

```
<table border="0" cellpadding="1" cellspacing="5">
  <tbody>
      <tr>
            <td style="width:110px;text-align:center;">

<a href="#permalink();">#if(#img_url();,<img src="#img_url();"
style="width:100px;" />,<img src="./styles/pmb4/images/no_image.jpg"
style="width:100px;" />);</a>

            </td>

            <td style="text-align:left;">

<a href="#permalink();"><font
size=3>#title();#a_empty(#complement_title();, : );</font></a>

#str_replace( , ,#a_empty(#authors_by_type_with_tpl(1,3,\, ,\,
,0,0,70\,71\,72,{rejete} {name});,<br />de ););

#a_empty(#resume();,<br /><br />);
            </td>
      </tr>

  </tbody>

</table>
```

Créer un tableau dont les bordures sont invisibles (border="0")

Lien cliquable sur une image = sur la vignette <a href="
URL de destination = permalink de la notice dans l'OPAC
Si cette vignette existe #if(#img_url();
on affiche la vignette du livre <img src="#img_url();
Sinon = s'il n'y a pas de vignette
On affiche une vignette standard (pas d'image pour ce titre)
<img src="./styles/pmb4/images/no_image.jpg">
Fin du lien cliquable

Lien cliquable sur un texte = sur le titre <a href="
URL de destination = permalink de la notice dans l'OPAC
On affiche le titre
S'il y a un sous-titre (complément de titre)
On affiche le signe de ponctuation : et le sous-titre
Fin du lien cliquable (

S'il y a un ou plusieurs auteurs
On affiche le mot « de » et le(s) nom(s) et prénom(s)

S'il y a un résumé
On l'affiche

N.B. Explication de la fonction « auteurs » à la page suivante

Pour employer ce template de notices dans votre futur portail, il faut le créer dans la base de travail et dans la base en ligne !
Ce template comprend 4 parties : monographies, articles de périodiques, périodiques et notices de bulletins. Copiez tout ;-)
Pas de copier-coller à partir du traitement de texte, cela crée des problèmes. Copier-coller dans l'onglet Editions !!!

Portail – Liste de notices – Le template des monographies

Paramètres de la fonction qui affiche le nom des auteurs : #authors_by_type_with_tpl(1,3,\, ,\, ,0,1,70\,71\,72,{rejete}{name});

Cette fonction retourne les auteurs de la notice en fonction de leur type, selon une mise en forme à définir

#authors_by_type_with_tpl(param0,param1,param2,param3,param4,param5, param6,param7);

// $param[0] = 0=principal seul, 1=principal + autres, 2=tous
// $param[1] = nombre maximum d'auteurs à afficher
// $param[2] = séparateur entre auteurs
// $param[3] = séparateur entre auteur principal / autres auteurs / auteurs secondaires
// $param[4] = afficher la fonction : 0=non, 1=toujours
// $param[5] = afficher "et al." s'il y a plus d'auteurs que le maximum prévu dans le paramètre 1
// $param[6] = 70=physique, 71=collectivités, 72=congrès
// $param[7] = template pour l'affichage des auteurs, à définir à partir des éléments suivants :
{name} = nom ou élément d'entrée
{rejete} = prénom ou élément rejeté
{function} = fonction de l'auteur dans la notice
{author_comment} = commentaire

Pourquoi y a-t-il une barre oblique inversée \ avant certaines virgules ?
Regardez par ex. les paramètres 2 et 3 : pour le séparateur entre auteurs, et celui entre auteur principal / autre / secondaire

Vu que le séparateur entre les paramètres dans la fonction est une virgule, si on veut que le séparateur entre les noms de plusieurs auteurs soit lui aussi une virgule, il faut éviter à PMB de s'emmêler les pinceaux !

Dans ce cas, en Php - MySQL, si on emploie un signe qui sert normalement de séparateur de données dans une autre fonction, on doit le précéder d'un caractère appelé "caractère d'échappement". Le caractère "échappe" en quelque sorte à sa fonction habituelle (qui est celle de séparateur), il prend une autre fonction. En général, il récupère souvent sa fonction de départ : on veut afficher une « vraie » virgule !

Par convention, c'est la barre oblique inversée, \ (le backslash) qui sert de caractère d'échappement en Php - MySQL.

Si on analyse les paramètres 0, 1 et 2 de cette fonction : #authors_by_type_with_tpl (1,3,\, ,

- une simple virgule est le séparateur entre les paramètres ex. 1,3,
- \, (barre oblique + virgule = pas d'espace entre les 2) indique qu'on veut une « vraie » virgule (\ est le caractère d'échappement)
- \, , ex. paramètre 2 et séparateur entre les paramètres 2 et 3 : barre oblique + virgule espace virgule
  la 1e virgule (après \) sera une « vraie » virgule et elle sera suivie d'un espace (ex. après le nom d'un auteur)
  la 2e virgule est le séparateur entre les paramètres

Nouveautés d'un rayon – par exemple fiction jeunesse

Peut-on afficher la notice autrement ? Oui, si vous avez un autre template ! De toute façon, il y a déjà celui par défaut ...

On a choisi le template : Portail – Liste de notices

La bibliothécaire
de Gudule, Christophe Durual

Pourquoi la vielle dame qui habite en face de chez Guillaume écrit-elle très tard la nuit? Quelle est cette jeune fille qui ne sort de chez elle qu'à le nuit tombée? Pour résoudre ces mystères, Guillaume se lance dans un fantastique voyage au pays des livres et de l'écriture

Voici ce qu'on aurait vu avec le template par défaut

La bibliothécaire [texte imprimé] / Gudule, Auteur ; Christophe Durual . - [S.l.] : Hachette Jeunesse, 2007 . - 192 p. ; 17 cm. - (L.d.p.jeunesse) .
ISBN : 978-2-01-322406-2 : 4,90 eur

Catégories :	Sciences humaines:Communication et Information:Bibliothéconomie
Mots-clés :	Romans Jeunesse Littérature jeunesse Albums Jeunesse en format poche
Résumé :	Pourquoi la vielle dame qui habite en face de chez Guillaume écrit-elle très tard la nuit? Quelle est cette jeune fille qui ne sort de chez elle qu'à le nuit tombée? Pour résoudre ces mystères, Guillaume se lance dans un fantastique voyage au pays des livres et de l'écriture
Permalink :	http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=notice_display&id=80

Pour bien comprendre cette URL

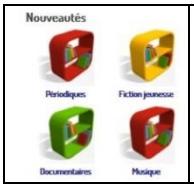
L'URL de cette image <img src="./styles/pmb4/images/no_image.jpg"> **est relative par rapport à pmb/opac_css/**
Autrement dit, son **adresse absolue** est par exemple http://127.0.0.1/opac_css/styles/pmb4/images/no_image.jpg

Peut-on employer une autre image pour symboliser l'absence de vignette ? Oui.

Dans ce contexte-ci (nouveautés d'un rayon), c'est simple, il suffit changer l'URL dans le template de notices.

Vous pouvez prendre une autre image en ligne « quelque part », il suffit d'indiquer son URL dans la balise <img src=" ">.
D'ailleurs, il y a déjà une image homonyme employée par défaut dans l'OPAC (dans le dossier pmb/opac_css/images/) : no_image.jpg
Dans le template de notices, l'URL relative de cette autre image deviendrait donc <img src="./images/no_image.jpg">

Dans d'autres contextes, si le mode d'affichage de la notice est géré par le template Django, il faut adapter ce template.
Il faut y ajouter une condition : si la vignette manque, on redirige le lien vers une autre image que celle prévue par défaut.
Cette solution sera expliquée dans l'analyse du carrousel suivant : notices dans les mêmes catégories qu'un article.



Je rappelle que ce template (Portail – Liste de notices) comprend en réalité 4 parties.
Jusqu'à présent, j'ai repris et illustré dans ce chapitre le template des **monographies** (rayon Fiction – Jeunesse).
Si vous cliquez sur l'étagère des périodiques, vous verrez comment on gère l'affichage d'autres types de documents :

- Un **périodique** (la notice-mère)
- Un **article de périodique**
- Une **notice de bulletin**

Exemples concrets : périodiques, articles et notices de bulletins	
Affichage géré par le template « Portail – Liste de de notices »	**Affichage « classique » dans l'OPAC**
Partie « Périodique » A l'écran, on voit seulement le titre du périodique et son logo. En effet, le champ résumé de la notice-mère du périodique est vide.	
Partie « Article de périodique » Par contre, le champ résumé a été complété pour cet article, ainsi que pour la notice de bulletin ci-dessous. Mais il n'y a pas de logo. Voir Adapter son mode de travail au portail Le champ « résumé »	
Partie « Notice de bulletin » **Remarque valable pour un OPAC classique et pour le portail. Un bulletin « ordinaire » n'est pas une notice, donc on ne peut pas le mettre dans un panier de notices pour le lier à une étagère. Il faut d'abord transformer le bulletin en « notice de bulletin ».** Voir Bulletins et notices de bulletins.	

Template Portail – Liste de notices (suite)

Template des périodiques

```
<table border="0" cellpadding="1" cellspacing="5">
  <tbody>
        <tr>
                <td style="width:110px;text-align:center;">
<a href="#permalink();">#if(#img_url();,<img src="#img_url();" style="width:100px;" />,<img src="./styles/pmb4/images/no_image.jpg"
style="width:100px;" />);</a>
                </td>

                <td style="text-align:left;">
<a href="#permalink();"><font size=3>#title();#a_empty(#complement_title();, : );</font></a>
#a_empty(#resume();,<br /><br />);
                </td>
        </tr>
  </tbody>
</table>
```

Template des articles

```
<table border="0" cellpadding="1" cellspacing="5">
  <tbody>
        <tr>
                <td style="width:110px;text-align:center;">
<a href="#permalink();">#if(#img_url();,<img src="#img_url();" style="width:100px;" />,<img src="./styles/pmb4/images/no_image.jpg"
style="width:100px;" />);</a>
                </td>

                <td style="text-align:left;">
<a href="#permalink();"><font size=3>#title();#a_empty(#complement_title();, : );</font></a>
#str_replace(  , ,#a_empty(#authors_by_type_with_tpl(1,3,\, ,\, ,0,0,70\,71\,72,{rejete} {name});,<br />de ););
<br />
<i>In #nom_revue();#a_empty(#numero_bulletin();,\, ); (#if(#extract_path((.*) [(],#date_bulletin(););,#extract_path((.*)
[(],#date_bulletin(););,#date_bulletin(););)</i>
#a_empty(#resume();,<br /><br />);
                </td>
        </tr>
  </tbody>
</table>
```

Template des notices de bulletins

```
<table border="0" cellpadding="1" cellspacing="5">
  <tbody>
        <tr>
                <td style="width:110px;text-align:center;">
<a href="#permalink();">#if(#img_url();,<img src="#img_url();" style="width:100px;" />,<img src="./styles/pmb4/images/no_image.jpg"
style="width:100px;" />);</a>
                </td>

                <td style="text-align:left;">
<a href="#permalink();"><font size=3>#extract_path((.*) [(],#title(););#a_empty(#complement_title();, : );</font></a>
#str_replace(  , ,#a_empty(#authors_by_type_with_tpl(1,3,\, ,\, ,0,0,70\,71\,72,{rejete} {name});,<br />de ););
<br />
<i>Bulletin de #nom_revue();</i>
#a_empty(#resume();,<br /><br />);
                </td>
        </tr>
  </tbody>
</table>
```

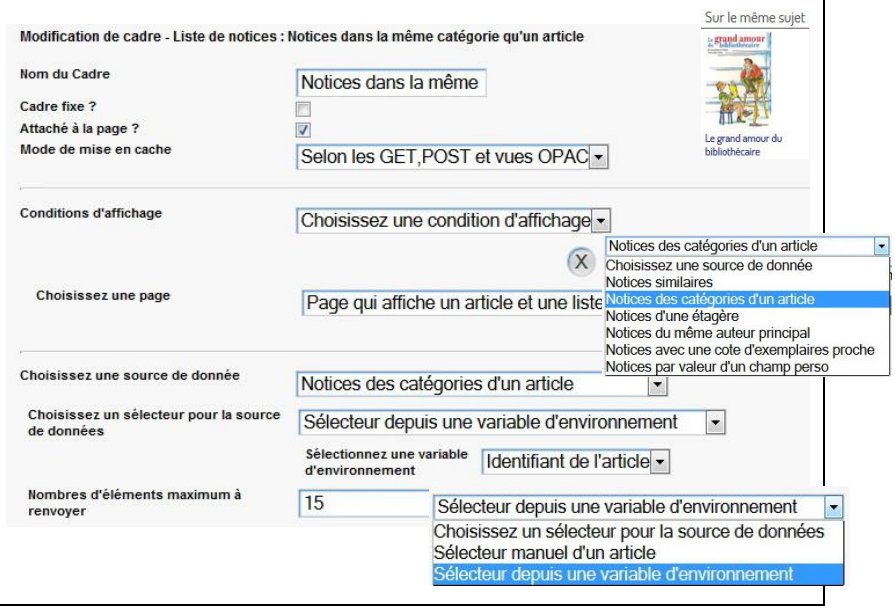
CARROUSEL « SUR LE MÊME SUJET » : NOTICES DANS LES MÊMES CATÉGORIES QU'UN ARTICLE

Formulaire pour le carrousel de notices dans les mêmes catégories qu'un article (1) – Module « Liste de notices »

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

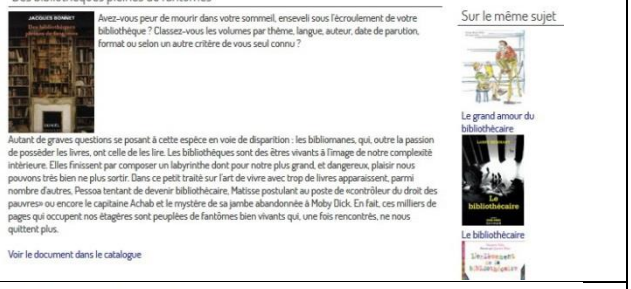
Dans quelle page afficher le cadre ?
Page qui affiche un article et une liste de notices
Source de données ? 6 possibilités
Notices dans les mêmes catégories qu'un article
Sélecteur ?
Depuis une variable d'environnement
Quelle variable d'environnement ?
L'identifiant de l'article est ici la seule option
Nombre maximum d'éléments à renvoyer ?
= nombre maximum de notices à afficher

Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.
Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection.



Choix du sélecteur pour la source de données

Rappel visuel : quand voit-on ce carrousel de notices dans les mêmes catégories qu'un article ?



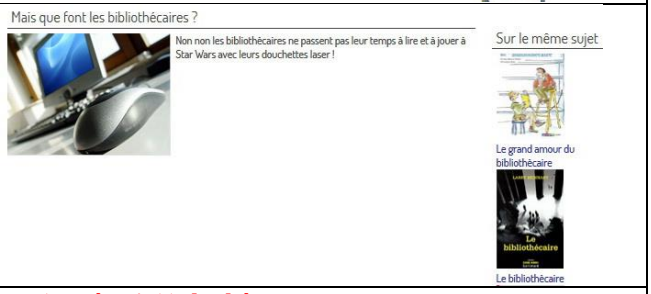
On est bien entendu dans la page qui affiche un article.

Ce carrousel n'a visiblement aucun lien avec le type de contenu de l'article. Ces 3 exemples illustrent la diversité à ce niveau-là :

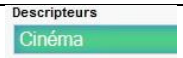
- Evénement (Tim Burton)
- Coup de cœur (Bibliothèques pleines de fantômes)
- Information (Mais que font les bibliothécaires ?)

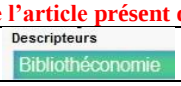
En outre, c'est le même carrousel pour 2 de ces articles.

Quel est donc le lien entre ces articles et ces notices ?



Tous ces articles sont liés à une (ou plusieurs) catégorie(s) du thésaurus.
Les notices affichées dans un carrousel sont liées aux même(s) descripteur(s) que l'article présent dans la page.





Les étagères « classiques » de l'OPAC et certaines étagères du portail sont « statiques » en ce sens que leur contenu est préétabli.
Dans ce cas-ci, il s'agit d'une « étagère dynamique », dont le contenu se crée au moment-même, en fonction des données présentes dans la base (données = notices + catégories + articles).
On a employé le module « Liste de notices », donc on peut afficher ces notices dans une vue Django (une liste) ou dans un carrousel.
N.B. Il est impossible d'obtenir ce genre de résultat avec le module Carrousel.

Le sélecteur pour la source de données est une variable d'environnement : l'identifiant de l'article. Ce numéro d'id suffit, puisqu'il permet de connaître les catégories associées à cet article, et ensuite de sélectionner les notices adéquates.

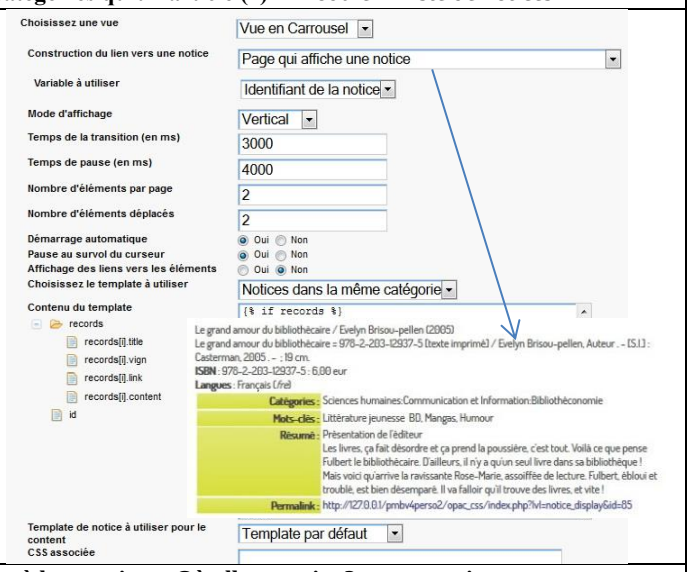
Formulaire pour le carrousel de notices dans les mêmes catégories qu'un article (2) – Module « Liste de notices »

Choisir une vue = comment afficher la liste ?
Vue en carrousel - Autre possibilité : vue Django (liste)

Construction du lien vers une notice
= où aller ensuite ? vers quelle page ?
Page du portail qui affiche une notice

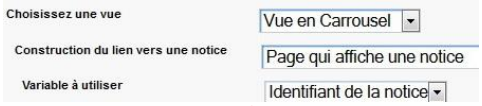
Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la notice est ici la seule option.
.
Mode d'affichage du carrousel ?
Défilement vertical
Autres possibilités : défilement horizontal ou fondu enchaîné
Définir les autres paramètres du carrousel

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?
La vignette prévue et le titre de la notice sont cliquables.



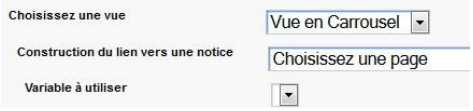
Dans ce type de module (liste de notices), la réponse à la question « Où aller ensuite ? » peut varier :
on peut aller vers la page OPAC « classique » qui affiche une notice, ou bien vers la page du portail qui affiche une notice.
Toutefois, dans la démo, on choisit toujours (dans le cas d'un carrousel) d'aller vers la page du portail qui affiche une notice.

Pour aller vers la page du portail qui affiche une notice,
on choisit la page adéquate dans la liste déroulante



Le lien vers la page du portail se fait par une URL avec variables

Pour aller directement vers la page de l'OPAC qui affiche une notice,
on ne choisit aucune des pages du portail qui sont proposées



Le lien vers l'OPAC se fait par le permalink de la notice

La page du portail qui affiche une notice propose (à la dernière ligne) un lien permanent (permalink) pour aller vers l'OPAC, donc on peut ensuite voir les infos habituelles sur la notice, sur les exemplaires et leur disponibilité. Mais la route est plus longue !
N.B. Si le permalink ne s'affiche pas, vérifiez le paramétrage de l'OPAC : le paramètre permalink doit avoir la valeur 1.

Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page du portail qui affiche une notice – page dont l'id est 7** est dans la démo
- PMB sait qu'il faut **afficher la notice correspondante,** et il connaît le **numéro identifiant de chaque notice**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque notice**, par exemple
  http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=87
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{record.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que la vignette et le titre de la notice soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

Dans la démo, on a choisi d'afficher la notice dans une page du portail, mais on aurait aussi pu l'afficher dans une page de l'OPAC, et l'URL de destination aurait correspondu au permalink : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=87

Formulaire pour le carrousel de notices dans les mêmes catégories qu'un article (3) – Module « Liste de notices »

Template de notice à utiliser pour le contenu ?
Template par défaut - Deux autres choix possibles

CSS associée ?
Aucun contenu

Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}

- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :

- Si le template Django contient la **boucle {% for record in records %}** **ou bien** s'il faut **afficher une seule notice.**

Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :

- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template employer ? » est sans objet, car on n'a pas employé la variable {{record.content}}

Template Django du carrousel de notices dans les mêmes catégories qu'un article

Résultat à l'écran

Parties du code que je vais adapter dans l'encadré suivant.

```
{% if records %}
    <h3>Sur le même sujet</h3>
{% endif %}

<ul id='carousel_{{id}}'>

{% for record in records %}

<li class='{{id}}_item'>

  <a href='{{record.link}}' alt='{{record.title}}'
title='{{record.title}}' >

<img style="max-width:120px;max-height:120px;text-
align:center;" src='{{record.vign}}'/>

<br />  {{record.title}}  <br />
 </a>

</li>
{% endfor %}
</ul>
```

N.B. Remplacez les apostrophes par des guillemets
alt="{{record.title}}" title="{{record.title}}"

Début de la condition **{% if records %}**
S'il y a des notices à afficher
On affiche le titre « Sur le même sujet »
Fin de la condition **{% endif %}**

Début de la liste non numérotée du carrousel (

Début de la boucle : **{% for record in records %}**
Permet d'afficher les vignettes des notices du carrousel

Début de la définition d'un élément de la liste (<li>)

Début de la balise <a href=" … (lien URL)
URL de destination

On clique sur la vignette :
Ou on clique sur le titre de la notice : {{record.title}}

Fin de la balise

Fin de la définition de cet élément (

Fin de la boucle : **{% endfor %}**

Fin de la liste non numérotée (

Un carrousel (de notices, d'articles ou de rubriques) est toujours bâti sur le modèle de la boucle {% for record in records %}

Syntaxe des templates Django	**Les instructions entre {% %}** la boucle de répétition {% for record in records %} la condition {% if records %} {% endif %} **Les champs entre {{ }}** URL de la notice {{record.link}}	**La condition {% if records %} est nécessaire ici** **Elle signifie : s'il y a des enregistrements = notices** Inutile d'afficher le titre « Sur le même sujet » s'il n'y a pas de notices à afficher dans cette étagère dynamique !
Balises HTML	Toutes les balises ont déjà été expliquées précédemment. Si nécessaire, voir aide-mémoire - HTML	

Petite adaptation du template Django du carrousel de notices dans les mêmes catégories qu'un article
Objectif : afficher une autre image que celle employée par défaut pour symboliser l'absence de vignette

Notice avec vignette
Template d'origine

Notice sans vignette
Template d'origine

Notice sans vignette
Template modifié

```
{% if records %}
<h3>Sur le même sujet</h3>
{% endif %}

<ul id='carousel_{{id}}'>

{% for record in records %}
<li class='{{id}}_item'>

<a href='{{record.link}}' alt="{{record.title}}"
title="{{record.title}}">

{% if
record.vign!="http://127.0.0.1/pmbv4perso2/opac_css/i
mages/vide.png" %}
   <img style="max-width:120px;max-
height:120px;text-align:center;" src='{{record.vign}}'
/>

{% else %}

<img style="max-width:120px;max-height:120px;text-
align:center;"
src='http://127.0.0.1/pmbv4perso2/opac_css/images/no_
image_question.jpg' />
{% endif %}

<br />  {{record.title | limitstring,45,"..." }}  <br />
</a>

</li>

{% endfor %}

</ul>
```

Début de la condition **{% if records %}**
S'il y a des notices à afficher
On affiche le titre « Sur le même sujet »
Fin de la condition **{% endif %}**

Début de la liste non numérotée du carrousel (

Début de la boucle : **{% for record in records %}**
Permet d'afficher les vignettes des notices du carrousel

Début de la définition d'un élément de la liste (<li>)

Début de la balise <a href=" … (lien URL)
URL de destination

Début de la condition
Si la vignette à afficher est différente d'une vignette vide
{% if record.vign!="http://... /vide.png" %}
On affiche la vignette du livre et on peut cliquer

Sinon **{% else %}**
On affiche une autre vignette et on peut cliquer

Fin de la condition **{% endif %}**
Ou on peut cliquer sur le titre de la notice : {{record.title}}

Fin de la balise

Fin de la définition de cet élément (

Fin de la boucle : **{% endfor %}**

Fin de la liste non numérotée (

Template Django : la condition {% if %} …. {% else %} … {% endif if %}
Objectif : en cas d'absence de vignette, afficher une image alternative, autre que celles prévues par défaut dans PMB

N.B. Je suppose que le paramétrage des vignettes est correct. Relire si nécessaire les explications que j'ai déjà fournies

Comment accéder à ces vignettes ? Voici les trois options possibles :
- A partir d'un **répertoire spécifique, le site de Amazon**
- A partir d'un **répertoire spécifique propre à votre institution**
- **Sur base du contenu du champ « URL de la vignette »**

Images que PMB affiche par défaut en cas d'absence de vignette

Dans les deux premiers cas, l'URL de la vignette est construite de manière dynamique, à partir de l'ISBN (sans les tirets).
Le paramètre book_pics_url contient dès lors une valeur comme http://images-eu.amazon.com/images/P/!!isbn!!.08.MZZZZZZZ.jpg
Dans le 3e cas, cela dépend de ce que vous avez catalogué : le champ URL de la vignette est-il complété ?

Dans les 2 premiers cas, la vignette s'affiche automatiquement si la notice contient un ISBN correct !
La vignette s'affiche … du moins, si une vignette est effectivement disponible !
- **L'ISBN ne concerne que les monographies**, donc la création dynamique d'une URL de vignette n'est pas possible pour les périodiques, les livres plus anciens ou certaines parutions à tirage limité (thèses, mémoires, travaux de fin d'étude, brochures …)
- En outre, **ce système vous rend dépendant d'un site**, par exemple Amazon : si un livre, un CD, un DVD n'est pas (plus) en vente chez eux, vous ne verrez pas de vignette, même si votre ISBN est correct.

Dans le cas des périodiques (notices-mères, notices de bulletins ou articles de périodiques), ou de monographies sans ISBN, vous pouvez seulement employer l'option n° 3 : compléter le champ URL de la vignette.

Comment PMB gère-t-il l'absence de vignette si l'URL a été construite de manière dynamique à partir de l'ISBN ?
PMB « transite » d'abord par l'affichage d'une image vide : « vide.png », ensuite il affiche une image par défaut.
C'est pourquoi le script expliqué ci-dessous vérifie si la vignette à afficher est différente de l'image « vide.png »

Remplacer une des images par défaut de PMB par une image de votre choix ne sera donc possible que
- s'il s'agit d'une **monographie** (il faut un ISBN)
- **si l'URL** est construite **de manière dynamique, sur base de l'ISBN** (options 1 ou 2)

Dans le cas n° 3, vous continuerez à voir les images par défaut prévues par PMB, car l'URL de la vignette à afficher est enregistrée dans un champ de la notice, et même si ce champ est vide ou contient une valeur erronée, on ne « transite » pas par l'image « vide.png ».

Monographies avec ISBN

Options n° 1 ou n° 2 : URL dynamique
Si l'ISBN dans votre base est correct et correspond à une vignette stockée en ligne, on voit la vignette.
Si l'ISBN dans votre base est erroné ou manquant, ou bien s'il n'y a pas de vignette en ligne pour cet ISBN, on voit votre image alternative.

Option n° 3 : champ URL de la vignette
Si le champ est complété correctement, et qu'il y a une vignette disponible en ligne, on voit la vignette.
Sinon, on voit l'image par défaut de PMB.

Périodiques ou monographies sans ISBN

J'espère que ces explications complémentaires vous ont permis de comprendre pourquoi on a ajouté une condition dans le template Django, et pourquoi on a absolument voulu vérifier si une image était différente ou non de « vide.png »

```
<a href='{{record.link}}' alt="{{record.title}}" title="{{record.title}}">

{% if record.vign!="http://127.0.0.1/pmbv4perso2/opac_css/images/vide.png" %}
   <img style="max-width:120px;max-height:120px;text-align:center;" src='{{record.vign}}' />

{% else %}

<img style="max-width:120px;max-height:120px;text-align:center;" src='http://127.0.0.1/pmbv4perso2/opac_css/images/no_image_question.jpg' />
{% endif %}

<br />   {{record.title | limitstring,45,"..." }}    <br />

</a>
```

Début de la balise <a href=" … (lien URL)
URL de destination

Début de la condition
Si la vignette à afficher est différente d'une image vide
{% if record.vign!="http://... /vide.png" %}
On affiche la vignette du livre et on peut cliquer

Sinon **{% else %}**
On affiche une autre image et on peut cliquer

Fin de la condition **{% endif %}**
Ou on peut cliquer sur le titre de la notice : {{record.title}}

Fin de la balise

{% if %} Si la condition est vérifiée
Instruction n° 1
{% else %} Sinon
Instruction n° 2
{% endif %} Fin de la condition

Par convention

!= signifie "différent de"

L'image de remplacement que j'ai prévue s'appelle no_image_question.jpg.

Elle est enregistrée dans pmb/opac_css/images

Bravo !! Vous venez de parcourir (et je l'espère, de comprendre) toutes les instructions qui existent dans les templates Django ;-)
Vous voyez bien que ce n'était pas un obstacle insurmontable : les boucles, les conditions et la fonction limitstring …

Cadres du portail attachés à une page de l'OPAC

Liste d'articles dans les mêmes catégories qu'une notice - Liste de notices du même auteur qu'une notice
Dans la démo, ces cadres sont attachés à une page de l'OPAC, mais on aurait aussi pu sélectionner une page du portail.

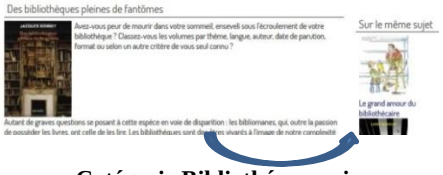
Pourquoi ces exemples supplémentaires ?

- Je voulais montrer que la démo contient aussi des exemples de cadres liés à une page de l'OPAC, et **rappeler ainsi qu'on peut insérer un cadre dans une page du portail ou de l'OPAC** (même si un choix convient parfois mieux que l'autre).
- En outre, **ces cadres complètent les 2 séries précédentes d'exemples** : le 1er cadre affiche une **liste d'articles**, et le 2e affiche une **liste de notices**.

Vu leur principe de base, ces cadres s'insèrent idéalement dans une page (portail ou OPAC) où on affiche une notice :

- Les **Articles dans les mêmes catégories qu'une notice** et les **Notices du même auteur qu'une notice** font référence au même point de départ : une notice. Leur libellé le montre d'ailleurs clairement.

Il y a un lien entre le dernier cadre analysé (notices dans les mêmes catégories qu'un article**) et le prochain que je vais présenter.**
Les modules qui ont généré ces cadres ont des fonctions similaires – dans un sens ou dans l'autre …

Liste de notices dans les mêmes catégories qu'un article	Liste d'articles dans les mêmes catégories qu'une notice
Dans la page du portail qui affiche un article	**Dans la page de l'OPAC qui affiche une notice**
	
Catégorie Bibliothéconomie	**Catégorie Cinéma**

Articles dans les mêmes catégories qu'une notice

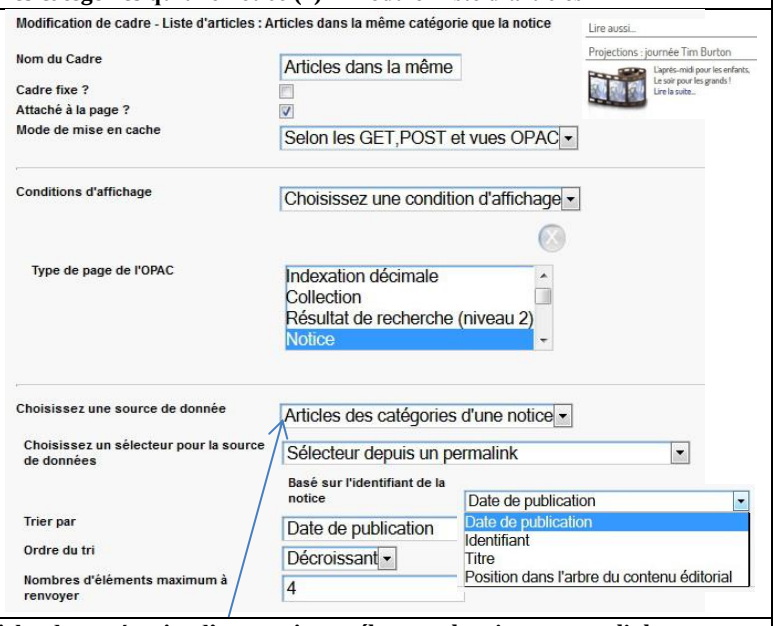
Formulaire pour les articles dans les mêmes catégories qu'une notice (1) – Module Liste d'articles

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page de l'OPAC qui affiche une notice
Source de données ?
Articles des catégories d'une notice
Sélecteur ?
Sélecteur d'une notice depuis un permalink, c'est donc basé sur l'identifiant de la notice affichée au départ dans cette page de l'OPAC
Critère et ordre de tri ?
Par date de publication - décroissant

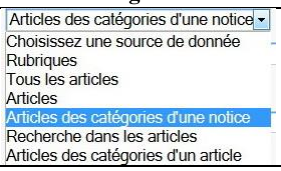
Nombre maximum d'éléments à renvoyer ?
= nombre maximum d'articles à afficher

Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection.

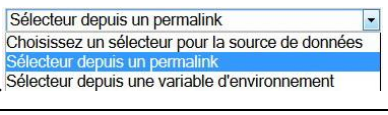


Source de données et choix du sélecteur : les articles des catégories d'une notice - sélecteur depuis un permalink

Dans les 3 exemples de listes d'articles vus jusqu'à présent, le choix était toujours « Rubriques », et ensuite, « sélecteur manuel d'une ou de plusieurs rubriques ».
On change ici de source de données et de sélecteur.



Choix du sélecteur



Il s'agit ici d'une liste dynamique, créée au moment même, en fonction des catégories communes à la notice et aux articles.
La liste des articles à afficher peut changer de jour en jour, en fonction de l'évolution du contenu éditorial et bibliographique.

Formulaire pour les articles dans les mêmes catégories qu'une notice (2) – Module Liste d'articles

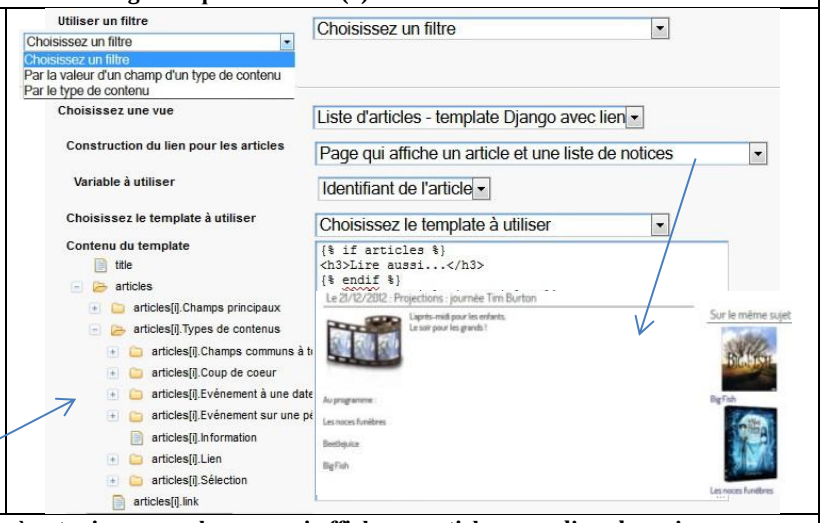
Filtre
Aucun filtre sélectionné

Choisir une vue = comment afficher la liste ?
Liste d'articles – template Django avec lien
Autre possibilité : Vue en carrousel

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django
Rappel : si vous dépliez les dossiers, vous pouvez voir la liste des variables utilisables dans le template.



Un cadre issu du module « Liste d'articles » mène toujours vers la page qui affiche un article et une liste de notices.
Je rappelle que c'est une page multifonctions, conçue pour afficher tous les types de contenu d'articles de la démo.

Template Django des articles dans les mêmes catégories qu'une notice

Résultat à l'écran

```
{% if articles %}
<h3>Lire aussi...</h3>
{% endif %}

{% for article in articles %}

<div style="width:49%;margin-right:1%;float:left;margin-bottom:5px;margin-top:5px;">

<h3>{{article.title}}</h3>

<a href='{{article.link}}' alt="{{article.title}}" title="{{article.title}}"><img style="float:left;margin-right:5px;" src='{{article.logo.vign}}' />
</a>

<p>{{article.resume}}</p>

<a style="text-align:right;" href="{{article.link}}">Lire la suite...
</a>

</div>

{% endfor %}
```

Ce template s'appelle « Liste d'articles - template Django avec lien », ce n'est pas un hasard !

Il prévoit la possibilité de cliquer sur le logo de l'article ou sur le texte « Lire la suite » pour aller vers la page qui affiche un article.

Début de la condition **{% if articles %}**
S'il y a des articles à afficher
On affiche le titre « Lire aussi … »
Fin de la condition **{% endif %}**

Début de la boucle : **{% for article in articles %}**
une "boucle" qui va du premier au dernier article

Début de la balise <div style=" … qui gère la mise en forme

On affiche le titre de l'article

Début de la 1e balise <a href=" … (lien URL)

D'abord l'URL de destination, le lien
Ensuite, la « source » = d'où on part = où peut-on cliquer ?
Sur le logo de l'article :

Fin de la 1e balise

On affiche le résumé = le chapeau de l'article

Début de la 2e balise <a href=" … (lien URL)

D'abord l'URL de destination, le lien
Ensuite, la « source » = d'où on part = où peut-on cliquer ?
Sur le texte « Lire la suite … »

Fin de la 2e balise

Fin de la balise

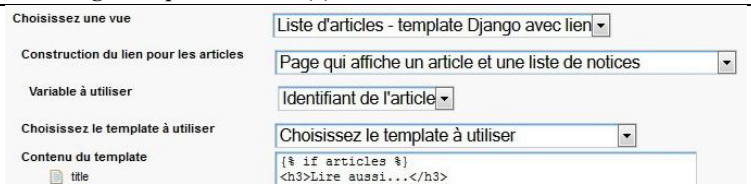
Fin de la boucle : **{% endfor %}**

Syntaxe des templates Django	**Les instructions entre {% %}** la boucle de répétition {% for article in articles %} la condition {% if articles %} {% endif %} **Les champs entre {{ }}** URL de l'article {{article.link}}	**La condition {% if articles %} est nécessaire ici** **Elle signifie : s'il y a des articles liés aux mêmes catégories que la notice affichée au départ** Inutile d'afficher le titre « Lire aussi » s'il n'y a pas d'articles à afficher dans cette liste dynamique !
Explications des balises HTML	**La balise <h3>** détermine la taille des caractères **Les balises <div style="** … > et ** ** marquent le début et la fin d'un « bloc de texte » à mettre en forme. L'affichage du texte situé entre ces 2 balises peut être personnalisé : marges, alignement, police de caractères … **La balise <img>** signifie image. **<img … src=' …'>** signifie la source de l'image = l'URL de l'image.	

Formulaire pour les articles dans les mêmes catégories qu'une notice (3) – Module Liste d'articles

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.



On va vers la page qui affiche un article et une liste de notices.
Je rappelle que c'est une page multifonctions, conçue pour afficher tous les types de contenu d'articles de la démo.

Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page qui affiche un article – page dont l'id est 4** est dans la démo
- PMB sait qu'il faut **afficher un ou plusieurs articles**, et il connaît le **numéro identifiant de chaque article**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque article**, par exemple
  http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=11
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{article.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que le logo et le texte « Lire la suite » soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

NOTICES DU MÊME AUTEUR QU'UNE NOTICE

Carrousel de notices du même auteur qu'une notice - Module « liste de notices »

Dans la page de l'OPAC qui affiche une notice

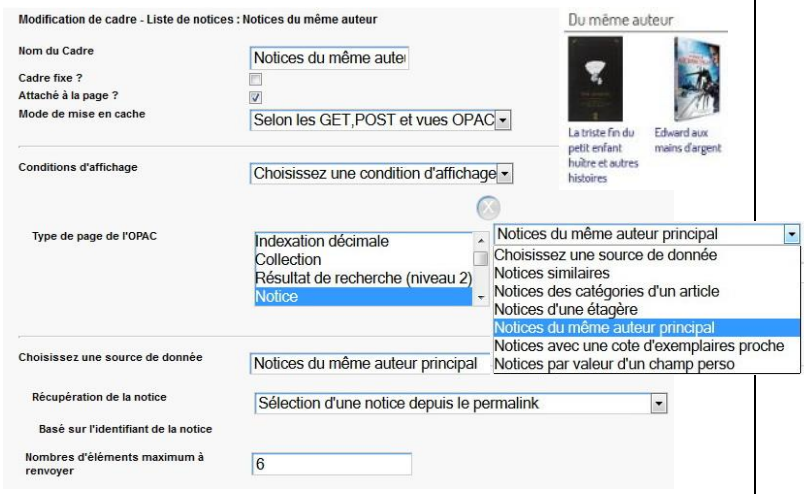
Auteur : Tim Burton

Formulaire pour le carrousel de notices du même auteur qu'une notice (1) – Module « Liste de notices »

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page de l'OPAC qui affiche une notice
Source de données ? 6 possibilités
Notices du même auteur principal
Sélecteur ?
Sélecteur d'une notice depuis un permalink, c'est donc basé sur l'identifiant de la notice affichée au départ dans cette page de l'OPAC
Nombre maximum d'éléments à renvoyer ?
= nombre maximum de notices à afficher

Dans le cas d'une liste ou d'un carrousel de notices, la valeur zéro a un sens « normal » : cela veut dire qu'on n'affiche aucune notice.



Il s'agit d'un carrousel dynamique, créé au moment même, en fonction des auteurs et notices disponibles dans la base.
La liste des notices à afficher peut changer de jour en jour, en fonction de l'évolution du contenu bibliographique.

Formulaire pour le carrousel de notices du même auteur qu'une notice (2) – Module « Liste de notices »

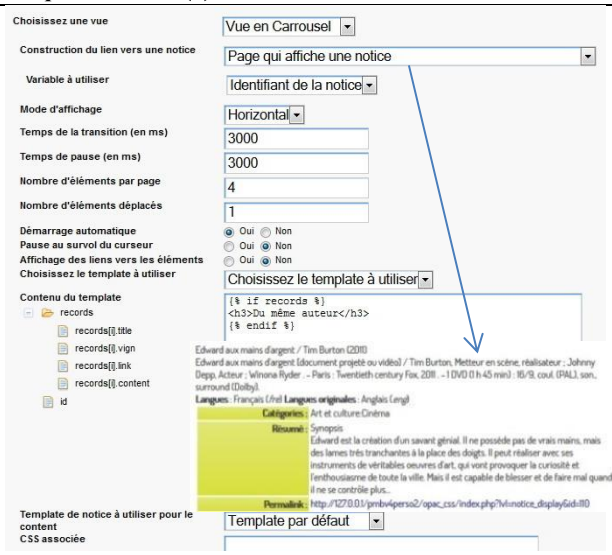
Choisir une vue = comment afficher la liste ?
Vue en carrousel - Autre possibilité : vue Django (liste)

Construction du lien vers une notice
= où aller ensuite ? vers quelle page ?
Page du portail qui affiche une notice

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la notice est ici la seule option.
.

Mode d'affichage du carrousel ?
Défilement horizontal (ou vertical ou en fondu enchaîné
Définir les autres paramètres du carrousel

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?
Le template affiche la vignette prévue et le titre de la notice, et il gère aussi la navigation, car ces 2 éléments sont cliquables.



Dans ce type de module (liste de notices), la réponse à la question « Où aller ensuite ? » peut varier :
on peut aller vers la page OPAC « classique » qui affiche une notice, ou bien vers la page du portail qui affiche une notice.
Toutefois, dans la démo, on choisit toujours (dans le cas d'un carrousel) d'aller vers la page du portail qui affiche une notice.

Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}

- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :

- Si le template Django contient la **boucle {% for record in records %}** **<u>ou bien</u>** s'il faut **afficher une seule notice.**

Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :

- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template employer ? » est sans objet, car on n'a pas employé la variable {{record.content}}

Template Django du carrousel de notices du même auteur qu'une notice

Résultat à l'écran

.

```
{% if records %}
    <h3>Du même auteur</h3>
{% endif %}

<ul id='carousel_{{id}}'>

{% for record in records %}

<li class='{{id}}_item'>

<a href='{{record.link}}' alt="{{record.title}}"
title="{{record.title}}">
<img style="max-width:120px;max-height:95px;text-
align:center;" src='{{record.vign}}'/>

<br />{{record.title}}<br />

</a>

</li>

{% endfor %}

</ul>
```

Début de la condition **{% if records %}**
S'il y a des notices à afficher
On affiche le titre « Du même auteur »
Fin de la condition **{% endif %}**

Début de la liste non numérotée du carrousel (

Début de la boucle : **{% for record in records %}**
Permet d'afficher les vignettes des notices du carrousel

Début de la définition d'un élément de la liste (<li>)

Début de la balise <a href=" … (lien URL)
URL de destination

On clique sur la vignette :
<u>Ou</u> on clique sur le titre de la notice : {{record.title}}

Fin de la balise

Fin de la définition de cet élément (

Fin de la boucle : **{% endfor %}**

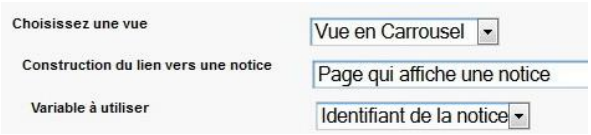
Fin de la liste non numérotée (

Syntaxe des templates Django Balises HTML voir aide-mémoire HTML	**Les instructions entre {% %}** la boucle de répétition {% for record in records %} la condition {% if records %} {% endif %} **Les champs entre {{ }}** URL de la notice {{record.link}}	**La condition {% if records %} est nécessaire ici** **Elle signifie : s'il y a des enregistrements = notices** Inutile d'afficher le titre « Du même auteur » s'il n'y a pas de notices à afficher dans ce carrousel dynamique !

Formulaire pour le carrousel de notices du même auteur qu'une notice (3) – Module « Liste de notices »

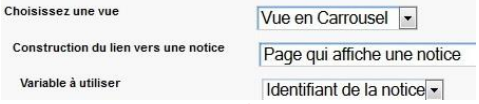
Construction du lien vers une notice
= où aller ensuite ? vers quelle page ?
Page qui affiche une notice

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la notice est ici la seule option.



Dans ce type de module (liste de notices), la réponse à la question « Où aller ensuite ? » peut varier :
on peut aller vers la page OPAC « classique » qui affiche une notice, ou bien vers la page du portail qui affiche une notice.
Toutefois, dans la démo, on choisit toujours (dans le cas d'un carrousel) d'aller vers la page du portail qui affiche une notice.

Pour aller vers la page du portail qui affiche une notice,
on choisit la page adéquate dans la liste déroulante



Le lien vers la page du portail se fait par une URL avec variables

Pour aller directement vers la page de l'OPAC qui affiche une notice,
on ne choisit aucune des pages du portail qui sont proposées

Le lien vers l'OPAC se fait par le permalink de la notice

La page du portail qui affiche une notice propose (à la dernière ligne) un lien permanent (permalink) pour aller vers l'OPAC, donc on peut ensuite voir les infos habituelles sur la notice, sur les exemplaires et leur disponibilité. Mais la route est plus longue !
N.B. Si le permalink ne s'affiche pas, vérifiez le paramétrage de l'OPAC : le paramètre permalink doit avoir la valeur 1.

Que se passe-t-il au moment où vous répondez à ces questions ?
Construction du lien ? Où aller ? vers quelle page ? Que faire dans cette page ? Comment le signaler ? Quelle variable transmettre ?

- PMB sait qu'il faut **aller ensuite vers la page du portail qui affiche une notice – page dont l'id est 7** est dans la démo
- PMB sait qu'il faut **afficher la notice correspondante,** et il connaît le **numéro identifiant de chaque notice**
- **PMB peut ainsi construire une URL avec variables, une page de destination pour chaque notice**, par exemple
  http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=110
- **Cette URL qui a été construite de manière dynamique est enregistrée dans la variable {{record.link}}** (a link = un lien)
- **Mais pour avoir un lien fonctionnel, il faut en outre qu'on puisse « cliquer sur quelque chose » pour accéder à la page de destination : c'est le rôle du template** qui prévoit (dans ce cas-ci) que la vignette et le titre de la notice soient cliquables.

N.B. Si nécessaire, relire le chapitre URL avec variables et navigation

Dans la démo, on a choisi d'afficher la notice dans une page du portail, mais on aurait aussi pu l'afficher dans une page de l'OPAC, et l'URL de destination aurait correspondu au permalink : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=110

Conseils complémentaires – Listes et carrousels de notices créés de manière dynamique

Si on crée un carrousel manuellement à partir d'une étagère, le résultat final est prévisible, car on connaît dès le début

- le nombre maximum d'éléments à renvoyer – c'est le contenu du panier lié à cette étagère
- le nombre d'éléments par page (visibles en une fois) – on l'a fixé soi-même
- le nombre d'éléments à déplacer – on l'a fixé soi-même

N.B. **Veillez si possible à ce que le nombre maximum d'éléments à renvoyer soit un multiple du nombre d'éléments par page, cela rend le mouvement plus fluide** (s'il y a 3 vignettes par page, il vaut mieux un carrousel de 12 éléments plutôt que 11 ou 13). Dans le même ordre d'idées, tenez compte du nombre de vignettes par page lorsque vous déterminez le nombre d'éléments à déplacer.

Lorsqu'on crée ce genre de carrousel de manière dynamique, il faut faire particulièrement attention au paramétrage, surtout s'il s'agit d'un carrousel comme les « autres notices du même auteur », car le résultat final est aléatoire.

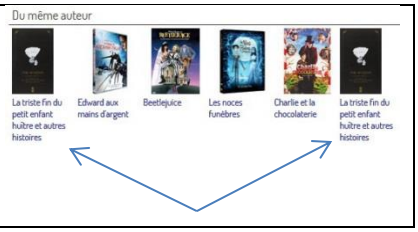
Si vous mettez ce genre de cadre dans la page de l'OPAC qui affiche une notice, on verra ce cadre pour toutes les notices de l'OPAC ! (sauf si l'auteur de la notice de départ n'a qu'une seule œuvre dans le catalogue)

Il faut en tout cas éviter de fixer un nombre trop élevé d'éléments à renvoyer, et privilégier l'idée de placer ce cadre « notices du même auteur » dans une page du portail. Ainsi, il ne s'affichera que dans certains cas prévisibles.

Des problèmes peuvent se poser pour un carrousel de notices si le nombre de notices du même auteur est peu élevé …

En effet, dans l'état actuel du paramétrage, si vous avez par exemple demandé d'afficher 6 vignettes, et qu'il n'y a que 6 œuvres de l'auteur dans la base, on ne peut créer que 5 vignettes différentes de celle de la notice de départ. Dans ce cas, PMB affiche 2 fois la même vignette !

N.B. Ce genre de problème (afficher plusieurs fois la même notice donc la même vignette) se pose pour les carrousels de notices, mais pas pour les listes de notices.
Ce problème ne concerne pas seulement les notices du même auteur (ce n'est qu'un exemple), on peut aussi le rencontrer pour les notices des catégories d'un article, les notices similaires …



Et aussi si le **nombre de notices du même auteur est fort élevé …**

Les notices affichées ne seront peut-être pas les plus intéressantes, ni les plus récentes, ni celles liées à une vignette …

Listes et carrousels de notices créés de manière dynamique ou manuelle ?

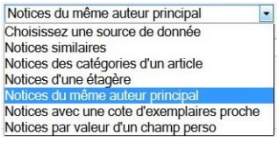
Avant de faire un choix définitif pour un modèle ou l'autre, il faut vraiment tester dans la base réelle.

Certes, il est plus rapide de créer de manière dynamique une liste ou un carrousel de notices : **notices du même auteur, dans les mêmes catégories qu'un article, dont la cote d'exemplaire est proche …** Les modèles ne manquent pas.
Les **notices similaires à une notice de départ** sont sélectionnées sur base des champs suivants : titre propre, éditeur, indexation décimale, catégories et mots-clés libres.

Le mode de création est dynamique : à partir du moment où vous avez choisi un modèle, la liste des notices à afficher change automatiquement en fonction de l'évolution du contenu bibliographique de la base, et parfois aussi du contenu éditorial (ex. les notices des catégories d'un article).

Créer une liste ou un carrousel de manière manuelle (à partir des notices d'une étagère) prend un peu plus de temps, car il faut créer une étagère spécifique, un panier qui lui est consacré, et remplir le panier avec des notices adéquates.

C'est un choix fixe : il faut éventuellement éditer le cadre plus tard pour changer d'étagère, et/ou modifier le contenu du (des) panier(s) qui alimente(nt) cette étagère, voire même supprimer carrément l'étagère si elle est devenue inutile.



La création manuelle offre un avantage indiscutable : celui de pouvoir <u>vraiment</u> sélectionner les notices à afficher.

Ex. Je reprends le cas des « notices du même auteur », des « notices similaires » ou des « notices liées aux mêmes catégories qu'un article ».
Si « vous laissez faire les choses sans intervenir », vos lecteurs risquent parfois de voir

- des notices fort anciennes ou peu pertinentes – si le choix se fait dans une base bien étoffée
- souvent les mêmes notices - si le choix se fait dans une base de taille réduite

Si la pertinence des résultats est à vos yeux le critère essentiel pour certains carrousels ou listes spécifiques, choisissez vous-même les notices à afficher, via le sélecteur manuel d'une étagère.

Sinon, s'il s'agit simplement d'offrir quelques suggestions de lecture complémentaire, optez sans souci pour la création dynamique de l'étagère.

De toute façon, vous pouvez avoir plusieurs carrousels ou listes de notices dans le portail, donc vous pouvez employer des modèles différents, et diversifier les critères de sélection des données.

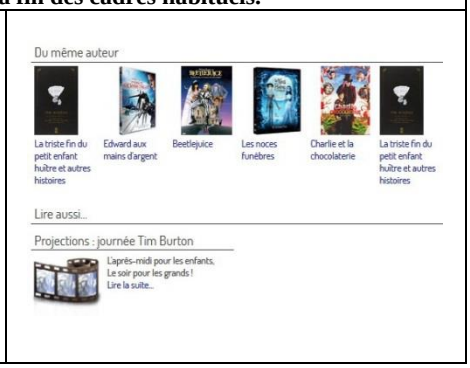
Page de l'OPAC qui affiche une notice : gestion ergonomique de l'ensemble des données

Cette page de l'OPAC est en général assez longue, car elle contient déjà beaucoup d'infos : les données de la notice et celles des exemplaires, et aussi des suggestions de lecture (Dans le même rayon et Peut-être aimerez-vous).

Que se passe-t-il si on y ajoute les deux derniers cadres que je viens d'analyser ?

Cadre généré par le module « liste d'articles » Articles dans les mêmes catégories qu'une notice	Cadre généré par le module « liste de notices » Notices du même auteur qu'une notice

Ces cadres sont visibles dans la page OPAC qui affiche une notice, à la fin des cadres habituels.



La page de l'OPAC qui affiche une notice est devenue fort longue, malgré un résumé qui est ici très bref !

Si vous affichez déjà les « notices dans le même rayon » et les suggestions « Peut-être aimerez-vous », ajouter sur la même page 2 cadres supplémentaires obligerait le lecteur à scroller beaucoup avant de les atteindre, surtout si la page centrale est assez étroite et le résumé fort long.
Il n'est pas sûr que les lecteurs découvrent rapidement les « notices du même auteur » et les « articles à lire » !

Il convient dès lors de se poser quelques questions, surtout à propos du cadre « Notices du même auteur ».
Avoir à sa dispostion ce genre de module est indéniablement un atout, mais dans quelle page placer ce cadre ?

Si vous l'insérez dans la page de l'OPAC qui affiche une notice, vous risquez de créer une impression de répétition – regardez les vignettes ci-dessus. En effet, il y a déjà 2 cadres de l'OPAC qui proposent des notices : « Dans le même rayon » et « Peut-être aimerez-vous ». Ajouter un cadre du portail comme « Du même auteur » peut encore augmenter le nombre de notices et de vignettes identiques.
Or le paramétrage est global : **si vous avez décidé d'afficher ces cadres, on verra toujours ceux de l'OPAC, et parfois aussi celui du portail** (si la base contient au moins 2 notices du même auteur). Et il serait dommage de devoir renoncer en toute circonstance aux cadres de l'OPAC qui proposent des suggestions de lecture (Dans le même rayon et Peut-être aimerez-vous).

Relativisons quand même : la démo du portail renforce ce sentiment de « déjà-vu », car elle ne contient que 68 notices, regroupées autour d'un petit nombre de thèmes majeurs (et les films de Tim Burton en font partie, il est l'auteur de 6 notices).

Le risque statistique de retrouver la même notice à plusieurs endroits est donc forcément plus élevé dans la démo que dans une base normale ! Ce risque dépend surtout du contenu de votre base, et de la manière de paramétrer l'OPAC.

Je pense qu'il serait en général plus judicieux d'afficher les « notices du même auteur » dans la page du portail qui affiche une notice, afin de pouvoir contrôler l'apparition ou non de ce cadre.
Bien sûr, ceci n'est qu'un avis personnel, pas une règle que j'essayerais d'imposer …
Testez en tout cas dans la vraie base, n'hésitez pas à modifier votre choix initial si nécessaire.

N.B. Dans le chapitre intitulé « Exemples supplémentaires », j'expliquerai comment insérer le cadre « Notices du même auteur » dans une variante de la page du portail qui affiche une notice.

REQUÊTES SQL LIÉES AUX PANIERS ET AUX ÉTAGÈRES

Exemple de liste de paniers – Avec le numéro d'id et le nombre de notices par panier

Type de panier	Id	Nom du panier	Nombre	Commentaire	Utilisateurs
Bulletins	6	Exemple de panier de bulletins	0		1
Exemplaires	8	Exemple de panier d'exemplaires	0		1 4 3 2
Notices	11	Exemple de panier de notices	0		1
Notices	12	Nouveautés en page d'accueil	13		1
Notices	13	Nouveautés fiction adultes	9		1
Notices	14	Nouveautés fiction jeunesse	13		1
Notices	15	Nouveautés documentaires	20		1
Notices	16	Nouveautés CD	5		1
Notices	17	Nouveautés DVD	5		1
Notices	18	Nouveautés périodiques	13		1

Exemple de contenu de 2 paniers de notices – avec indication du type de notice

	A	B	C	D	E	F	G	H	I	J	K	L
1	Id	Panier	ISBN / ISSN	Auteur	Type de notice	Id	Titre de la notice (+ titre de série)	Libellé de période	Date de parution	Editeur	Cote	Numéro (ex)
2	14	Nouveautés fiction jeun	978-2-87833-390-9	Heitz, Bruno	Monographie	81	Jojo (série) : Jojo et le secret de la		2006	Circonflexe	RJP HEI	000045
3	14	Nouveautés fiction jeun	978-2-07-053882-9	Mahy, Margaret	Monographie	87	L'enlèvement de la bibliothécaire		2002	Gallimard jeunesse	RJ MAH	000037
4	14	Nouveautés fiction jeun	978-2-01-322406-2	Gudule,	Monographie	80	La bibliothécaire		2007	Hachette Jeunesse	RJ GUD	000046
5	14	Nouveautés fiction jeun	978-2-203-12937-5	Brisou-pellen, Evelyn	Monographie	85	Le grand amour du bibliothécaire		2005	Casterman	JR BRI	000039
6	18	Nouveautés périodique:		Bonnet, Jacques	Article	125	Bibliothécaire de soi-même in Bulle		01/11/2010			
7	18	Nouveautés périodique:			Article	133	Dark Shadows in Studio ciné live n	avril 2012	01/04/2012		PER STU	P000006
8	18	Nouveautés périodique:		Aïoutz, Renaud	Article	127	L'abeille et le bibliothécaire ou éthi		01/09/2012			
9	18	Nouveautés périodique:		Akkouche, Mouloud	Article	126	Voyages en BM in Bulletin des bib		01/11/2010			
10	18	Nouveautés périodique:			Notice de bulletin	122	Géo - N° 393 - novembre 2011 - L'Îl	novembre 2011	01/11/2011		PER GEO	P000002
11	18	Nouveautés périodique:			Notice de bulletin	123	Géo - N° 402 - août 2012 - Compos	août 2012	01/08/2012			
12	18	Nouveautés périodique:			Notice de bulletin	121	Géo - N° 403 - septembre 2012 - N	septembre 2012	01/09/2012		PER GEO	P000003
13	18	Nouveautés périodique:			Notice de bulletin	131	Georges - Numéro Cupcake		24/10/2012		PER J GEO	P000005
14	18	Nouveautés périodique:			Notice de bulletin	130	Georges - Numéro Moustache		24/10/2012		PER J GEO	P000004
15	18	Nouveautés périodique:	1292-8399		Périodique	124	Bulletin des bibliothèques de Franc			ENSSIB		
16	18	Nouveautés périodique:	0220-8245		Périodique	120	Géo			Participations éditio		
17	18	Nouveautés périodique:	2110-8749		Périodique	129	Georges			Éd. Grains de sel		
18	18	Nouveautés périodique:	1969-9441		Périodique	132	Studio ciné live magazine			Studio magazine		

Exemple de liste d'étagères – Avec le paramétrage pour l'OPAC

Id	Nom de l'étagère	Commentaire	OPAC ?	Validité	De … à …	Utilisateurs
3	Nouveautés		Invisible	Permanente		1
4	Nouveautés fiction adultes		Invisible	Permanente		1
5	Nouveautés fiction jeunesse		Invisible	Permanente		1
6	Nouveautés documentaires		Invisible	Permanente		1
7	Nouveautés CD		Invisible	Permanente		1
8	Nouveautés DVD		Invisible	Permanente		1
9	Nouveautés périodiques		Invisible	Permanente		1
10	Test – Nouveautés – Fiction		Invisible	Permanente		1

Exemple de liste d'étagères – Avec le numéro d'id et le nombre de notices par panier associé

Id	Nom de l'étagère	Id	Nom du panier	Nombre
3	Nouveautés	12	Nouveautés en page d'accueil	13
4	Nouveautés fiction adultes	13	Nouveautés fiction adultes	9
5	Nouveautés fiction jeunesse	14	Nouveautés fiction jeunesse	13
6	Nouveautés documentaires	15	Nouveautés documentaires	20
7	Nouveautés CD	16	Nouveautés CD	5
8	Nouveautés DVD	17	Nouveautés DVD	5
9	Nouveautés périodiques	18	Nouveautés périodiques	13
10	Test – Nouveautés – Fiction	13	Nouveautés fiction adultes	9
10	Test – Nouveautés – Fiction	14	Nouveautés fiction jeunesse	13

N.B. Pour avoir la liste complète des procédures disponibles, voir Requêtes SQL

Analyser les pages-outils du portail

Que de chemin parcouru depuis le début …
*Vous venez de voir comment fabriquer le **dernier** modèle de briques dont vous aviez besoin.*
En effet, vous savez désormais comment configurer les cadres issus des modules !
*En outre, vous avez vu **toutes** les instructions liées aux templates Django.*
Votre « imprimante 3D » est prête à fonctionne, vous savez comment la piloter ;-)

Mais il est inutile de rêver à l'étage supérieur (la page d'accueil du portail), si vous n'avez pas érigé des fondations solides et un rez-de-chaussée dont les murs sont porteurs !
Les « murs porteurs », ce sont les pages du portail autres que la page d'accueil, celles que j'appelle les « pages-outils ».

Liste des pages du portail

	A	B
1	Id	Nom de la page
2	3	Page d'accueil
3	4	Page qui affiche un article et une liste de notices
4	6	Page qui affiche une rubrique
5	7	Page qui affiche une notice
6	9	Page qui affiche une liste d'événements
7	11	Page qui affiche une liste d'articles (recherche)

Liste des pages-outils et des variables d'environnement

	A	B	C	D
1	Id	Nom de la page-outil	Variable d'environnement	Libellé
2	4	Page qui affiche un article et une liste de notices	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	id_date	Identifiant de date
6	11	Page qui affiche une liste d'articles (recherche)		

N.B. Il n'y a pas de variable d'environnement pour la page qui affiche une liste d'articles (le résultat d'une recherche dans le contenu éditorial).

Aussi longtemps que les pages-outils ne sont pas opérationnelles, cela ne sert à rien de mettre la plupart des cadres dans la page d'accueil du portail ou ailleurs, ils ne fonctionneront pas !!

> **A quoi bon mettre un carrousel ou une liste d'articles en page d'accueil si la page qui est censée afficher un article n'existe pas encore, ou bien si elle existe mais qu'on n'y a pas encore inséré un cadre issu du module « afficher un article » ?**
>
> **C'est comme si le GPS du portail dysfonctionnait et envoyait systématiquement les gens dans une « voie sans issue » …**

Le premier cadre à insérer est issu du « module de base », le module qui donne à chaque page-outil sa raison d'être :

- **Module « Rubrique »** dans la page qui affiche une rubrique
  - le **contenu de n'importe quelle rubrique** (résumé, sous-rubriques et/ou articles qui en dépendent)
- **Module « Article »** dans la page qui affiche un article et une liste de notices
  - le **contenu de n'importe quel article** (résumé et texte même de l'article)
  - **quel que soit le type de contenu** : coup de cœur, sélection, information, événement (date ou période)
- **Module « Notice »** dans la page qui affiche une notice
  - le **contenu de n'importe quelle notice**
- **Module « Agenda »** dans la page qui affiche une liste d'événements qui correspondent à la date sélectionnée dans l'agenda
  - les **activités qui ont lieu à n'importe quelle date ou période** - du moins si quelque chose est prévu !
- **Module « Liste d'articles »** dans la page qui affiche les résultats d'une recherche dans le contenu éditorial
  - le **résultat de n'importe quelle recherche** (une liste d'articles) – du moins si on a trouvé quelque chose !

Les quatre premiers modules cités font d'ailleurs appel à la variable d'environnement de la page correspondante.

- **id_rubrique, id_article, id_notice et id_date**

En outre, chaque cadre gère en général « la suite de la navigation », il stipule

- **où on ira ensuite** : on le détermine la plupart du temps en répondant à **la question « Construction du lien vers … ?** »
- **comment y aller** : le **template Django (ou le template de notices)** génère si nécessaire une URL de destination (sur base de la réponse à la question « Construction du lien vers ? », et ce template spécifie aussi ce qui est cliquable : titre, logo …

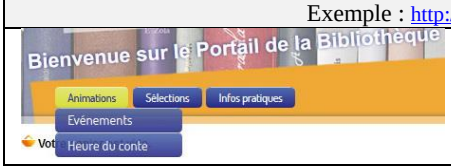
Si les pages-outils n'existaient pas, les cadres placés en page d'accueil ou sur d'autres pages ne mèneraient nulle part !

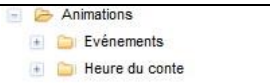
Un exemple significatif des « enchaînements » : le menu principal, qui est un cadre fixe, présent dans toutes les pages du portail.

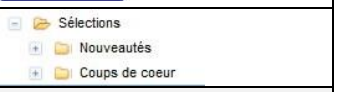
- Tel qu'il est conçu dans la démo, **ce menu affiche uniquement les titres des rubriques et des sous-rubriques**
- **Si la page qui affiche une rubrique n'existait pas ou n'était pas opérationnelle**, vous auriez beau cliquer (dans le menu) sur le titre d'une rubrique ou d'une sous-rubrique, rien ne se passerait, vous n'iriez nulle part ailleurs
- **Si cette page qui affiche une rubrique existe et fonctionne**, vous voyez les logos, titres et résumés des sous-rubriques et articles
- **Mais si la page qui affiche un article n'existait pas ou n'était pas opérationnelle**, cela ne servirait à rien de cliquer sur le logo d'un article (dans la page qui affiche une rubrique), vous ne pourriez pas changer de page pour lire le contenu de cet article !

PAGE DU PORTAIL QUI AFFICHE UNE RUBRIQUE

Comment accéder à la page du portail qui affiche une rubrique ?
Le « chemin habituel » part du menu principal qui présente les rubriques et sous-rubriques.
L'URL de destination contient le numéro d'id de la page (6) et la variable d'environnement liée à ce type de page : id_rubrique
Exemple : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=3







Comme vous le voyez, la page du portail qui affiche une rubrique peut prendre des aspects différents.

Votre panier est vide.

Animations

Evènements | Heure du conte

Votre panier est vide.

Evénements

Cultures d'Afrique noire contemporaine | Projections : journée Tim Burton

Votre panier est vide.

Sélections

Vous aimeriez sortir des sentiers battus, découvrir d'autres auteurs ou de nouvelles collections ? Ne cherchez plus, cette rubrique est faite pour vous !

Nous vous présentons un aperçu des acquisitions récentes disponibles dans les divers rayons, ainsi que les "Coups de coeur" des bibliothécaires.

Nouveautés | Coups de coeur

Animations est une rubrique parente sans résumé **On voit les 2 sous-rubriques qui en dépendent**	**Evénements n'a pas de sous-rubrique ni de résumé** **On voit le titre, le logo et le résumé des articles**	**Sélections est une rubrique parente avec résumé** **On voit le résumé et les sous-rubriques**
N.B. Dans la démo, la rubrique « Sélections » n'a pas de résumé, j'en ai « inventé » un.		Voir Rédiger les rubriques et articles

La page du portail qui affiche une rubrique n'est pas une page où on verrait « quelque part » le titre d'une rubrique, un titre sur lequel on pourrait cliquer, ni une page où on afficherait une rubrique précise (et pas les autres).

- **c'est une page où on voit le contenu d'une rubrique, quel que soit son numéro d'id**
  - on « voit le contenu d'une rubrique » signifie qu'on voit
    - les champs de la rubrique elle-même (titre + logo + résumé)
    - les sous-rubriques qui en dépendent (s'il y en a)
    - les articles qui en dépendent (s'il y en a)
  - **cette page a le numéro d'id 6 dans la démo** (ce numéro ne change pas)
  - la **variable d'environnement id_rubrique** transmet le **numéro d'id de la rubrique à afficher** (ce numéro varie)
    - ex. http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=3

- **en théorie, ce doit être une page polyvalente où on voit une rubrique, quel que soit son type de contenu**
  - **mais dans la démo**, il y a **un seul type de contenu** pour les rubriques, et ce type n'est lié à **aucun champ personnalisé**
  - donc la version originale du template Django de cette page n'est pas vraiment polyvalente, il faudra l'adapter si vous voulez créer un nouveau type de contenu pour les rubriques

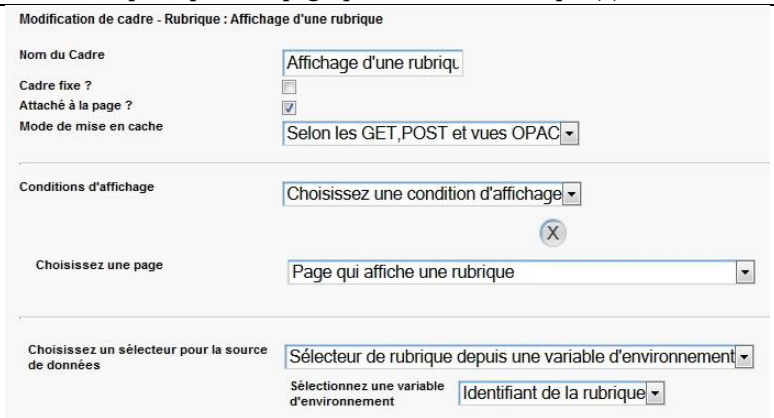
Formulaire pour le module « Rubrique » → cadre principal de la page qui affiche une rubrique (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche une rubrique

Sélecteur pour la source de données ?
Sélecteur de rubrique depuis une **variable d'environnement.**

L'identifiant de la rubrique (id_rubrique) est donc la seule possibilité.



Formulaire pour le module « Rubrique » → cadre principal de la page qui affiche une rubrique (2)

Afficher le contenu d'une rubrique, c'est montrer le titre et le résumé, **et aussi** les sous-rubriques et/ou articles qui en dépendent.

On pose deux fois la question « Construction du lien vers ?» :

où aller s'il s'agit d'une sous-rubrique ? où aller s'il s'agit d'un article ?

Ces deux questions ne s'excluent pas, car une rubrique peut contenir des sous-rubriques et des articles.

Construction du lien vers une rubrique
= où aller ensuite ? vers quelle page ?
Page qui affiche une rubrique

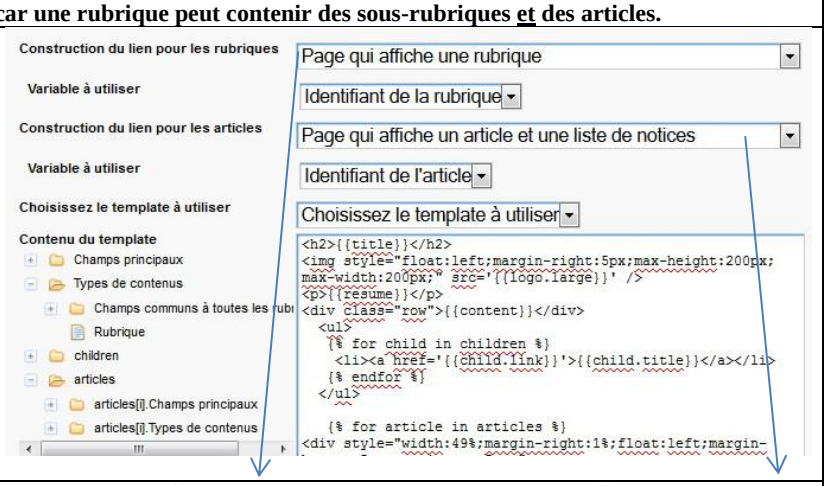
Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la rubrique est ici la seule option.

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Le template gère l'affichage des sous-rubriques et articles, il gère aussi la navigation vers « ailleurs », en précisant ce qui est cliquable (titre, logo, message)



Page qui affiche une rubrique

Page qui affiche un article

Template Django – Afficher le contenu d'une rubrique

```
<h2>{{title}}</h2>

<img style="float:left;margin-right:5px;max-height:200px;max-
width:200px;" src='{{logo.large}}' />

<p>{{resume}}</p>

<div class="row">{{content}}</div>

<ul>
  {% for child in children %}
     <li><a href='{{child.link}}'>{{child.title}}</a></li>
  {% endfor %}
</ul>

{% for article in articles %}
<div style="width:49%;margin-right:1%;float:left;margin-
bottom:5px;margin-top:5px;">
    <h3>{{article.title}}</h3>

    <a href='{{article.link}}' alt="{{article.title}}"
title="{{article.title}}"><img style="float:left;margin-right:5px;"
src='{{article.logo.vign}}' /></a>

    <p>{{article.resume}}</p>

    <a style="text-align:right;" href="{{article.link}}">Lire la suite...</a>
</div>
{% endfor %}
```

Titre de la rubrique
Logo (éventuel)
Résumé (éventuel) de la rubrique

Début de la liste non numérotée (<ul>)
- Début de la boucle : **{% for child in children %}**
  - Permet d'afficher les **sous-rubriques d'une rubrique parente**
  - Début de la définition d'un élément de la liste (<li>)
    - Début de la balise <a href=" … (lien URL)
      - URL de destination
      - On clique sur le titre de la sous-rubrique : {{child.title}}
    - Fin de la balise
  - Fin de la définition de cet élément (
- Fin de la boucle : **{% endfor %}**

Fin de la liste non numérotée ()

Début de la boucle : **{% for article in articles %}**
- Permet d'afficher les **articles d'une rubrique parente**
- Début de la balise
  - On affiche le titre de l'article
    - Début de la 1e balise <a href=" … (lien URL)
      - URL de destination
      - On clique sur le logo : {{article.logo.vign}}
    - Fin de la 1e balise
  - On affiche le résumé de l'article (chapeau)
    - Début de la 2e balise <a href=" … (lien URL)
      - URL de destination
      - **Ou** on clique sur le texte « Lire la suite »
    - Fin de la 2e balise
- Fin de la balise

Fin de la boucle : **{% endfor %}**

Explications sur les balises HTML : voir aide-mémoire HTML

Explications sur les templates : voir aide-mémoire - templates Django

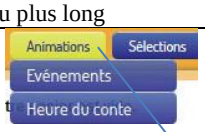
Les instructions entre {% %}
la boucle de répétition {% for child in children %}
Dans ce contexte, a child = une sous-rubrique
Les champs entre {{ }}
URL de l'article {{article.link}}

Une rubrique peut contenir des sous-rubriques et aussi des articles, le template Django en tient compte

Dans la démo, aucune rubrique ne contient à la fois des sous-rubriques **et** des articles, mais en soi, c'est tout à fait possible.
C'est ce que j'ai voulu illustrer en créant 3 sous-rubriques de « Evénements » :
Pour les adultes – Pour les enfants – Pour tous

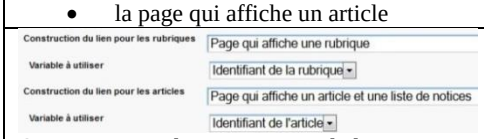
La version originale du template Django gère ce genre d'éventualité : on affiche d'abord la liste des sous-rubriques, et ensuite, celle des articles.

Accéder à la page qui affiche une rubrique … et poursuivre sa route …
On peut même passer plusieurs fois par la page qui affiche une rubrique avant d'atteindre un article

Un chemin direct		Un chemin un peu plus long	
Point de départ **Menu principal** Clic sur Evénements (id_rubrique = 4)		**Point de départ** **Menu principal** Clic sur Animations (id_rubrique = 1)	
On va **vers** la page qui affiche **une rubrique**	http://...&pageid=6&id_rubrique=4	On va **vers** la page qui affiche **une rubrique**	http://...&pageid=6&id_rubrique=1
Page → une rubrique **(pageid = 6)** Clic sur un article (Burton) (id_article = 11)		**Page → une rubrique** **(pageid = 6)** Clic sur Evénements (id_rubrique = 4)	
On va **vers** la page qui affiche **un article**	http://...&pageid=4&id_article=11	On va **vers** la page qui affiche **une rubrique**	http://...&pageid=6&id_rubrique=4
Page → un article **(pageid = 4)**		**Page → une rubrique** **(pageid = 6)** Clic sur un article (Burton) (id_article = 11)	
Page qui affiche un article (vue complète) 		On va **vers** la page qui affiche **un article**	http://...&pageid=4&id_article=11
		Page → un article **(pageid = 4)**	

Je n'avais montré qu'une « version réduite » de la page qui affiche un article. Mais le portail recèle souvent des surprises …
Si vous regardez cette page dans sa totalité, vous constatez la présence d'un **carrousel de notices sur le même sujet que l'article**.
Vous pouvez donc poursuivre l'exploration du catalogue …

La création automatique de ces interconnexions dépend de 2 éléments liés au cadre :
la réponse à la question « Construction du lien vers … » (= où aller ensuite ?) et le template Django du cadre.

Dans le formulaire de création du cadre	Dans le template associé à ce cadre : contenu de la balise <a href=" …>	
La question : Construction du lien vers … ? On a prévu ici 2 possibilités : aller vers • la page qui affiche une rubrique • la page qui affiche un article  On ne peut pas deviner sur quoi le lecteur va cliquer (rubrique ou article), il faut donc gérer les 2 cas dans le formulaire **et** dans le template.	**Exemple de lien à créer :** **vers la page qui affiche une rubrique (id = 6)** pour afficher une sous- rubrique (id = 4) Le template **crée l'URL de destination** et l'enregistre dans **{{child.link}}** http://...&pageid=6&id_rubrique=4 Le template gère aussi la **navigation entre les pages**, puisqu'il spécifie ce qui est cliquable : ici, le titre {{child.title}}	**Exemple de lien à créer :** **vers la page qui affiche un article (id = 4)** pour afficher un article (id = 11) Le template **crée l'URL de destination** et l'enregistre dans **{{article.link}}** http://...&pageid=4&id_article=11 Le template gère aussi la **navigation entre les pages**, puisqu'il spécifie ce qui est cliquable : le logo {{article.logo.vign}} ou le message « Lire la suite ».

En outre, la présence éventuelle d'un autre cadre sur la même page peut ouvrir d'autres perspectives …

PAGE DU PORTAIL QUI AFFICHE UN ARTICLE ET UNE LISTE DE NOTICES

Comment accéder à la page du portail qui affiche un article et une liste de notices ?
Le « chemin habituel » part d'une liste ou d'un carrousel d'articles, ou bien d'une recherche dans le contenu éditorial.
L'URL de destination contient le numéro d'id de la page (4) et la variable d'environnement liée à ce type de page : id_article
Exemple : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=9

N.B. Le mot recherché était « cinéma »

L'aspect de cette page varie en fonction du type de contenu de l'article, et des autres cadres présents éventuellement.

En effet, cette page ne contient pas uniquement le cadre « Afficher un article », il y en a deux autres.
Module Liste de notices → « Etagère associée à un article » et **« Notices dans les mêmes catégories qu'un article »**

Type Sélection	Type Coup de cœur	Type Evénement à une date
Ce type d'article est **lié à une étagère** **champ personnalisé : etagere**. **On voit deux cadres** : l'article et la liste des notices de cette étagère. En théorie, on pourrait aussi voir le 3e cadre : le carrousel de notices sur le même sujet.	Ce type d'article est **lié à une notice** **champ personnalisé : identifiant_notice** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.	Ce type d'article est **lié à une date précise** **champ personnalisé : date** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.
	Type Information	**Type Evénement sur une période**
Ce sont les 5 types de contenu définis dans la démo. Voir configuration préliminaire du contenu éditorial	Ce type d'article n'est **lié à aucun champ personnalisé**. **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.	Ce type d'article est **lié à deux dates précises : la date de début et la date de fin de l'activité** **champs personnalisés : date_debut, date_fin** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.

Le nom complet « page qui affiche un article et une liste de notices » se justifie souvent, mais pas dans tous les cas.
En effet, on voit toujours un article, mais pas nécessairement le contenu d'une étagère ou un carrousel de notices.

- **La présence de l'étagère liée à une liste de notices n'est possible que si l'article est de type Sélection**, si on a enregistré le numéro d'id d'une étagère (dans le champ personnalisé lié à ce type d'article) … et si l'étagère contient des notices !
- **Le carrousel de notices sur le même sujet peut apparaître de manière automatique, quel que soit le type de contenu de l'article. Sa présence éventuelle dépend des catégories communes à l'article et à certaines notices de la base**. Ce carrousel est donc créé de manière dynamique, sur base des liens entre le contenu éditorial et le contenu bibliographique.

Ne confondez pas ces variables ! Leurs noms sont similaires, et les questions de départ identiques : quelle notice ? ou quelle date ?
id_notice et **id_date** sont 2 **variables d'environnement → page** qui affiche **une notice ou page** qui affiche **une liste d'événements**
identifiant_notice et **date** sont 2 **champs personnalisés → types de contenu des articles : Coup de cœur ou Evénement à une date**

La page du portail qui affiche un article n'est pas une page où on verrait « quelque part » le titre d'un article, un titre sur lequel on pourrait cliquer, ni une page où on afficherait un article précis (et pas les autres).

- **c'est une page où on voit le contenu d'un article, quel que soit son numéro d'id**
  - on « voit le contenu d'un article » signifie qu'on voit l'ensemble de l'article
    - titre + logo + résumé (chapeau) + contenu (texte de l'article)
  - **cette page a le numéro d'id 4 dans la démo** (ce numéro ne change pas)
  - la **variable d'environnement id_article** transmet le **numéro d'id de l'article à afficher**, ce numéro varie donc
    - ex. http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=9

- **dans la démo, c'est une page polyvalente où on voit un article, quel que soit son type de contenu**
  - les **champs personnalisés de l'article** transmettent **d'autres paramètres liés au type de contenu de l'article** :
    - **numéro d'id de la notice** (coup de cœur), **numéro d'id de l'étagère** (sélection), **date** (événement à une date), **dates de début et de fin** (événement sur une période) – aucun champ personnalisé pour le type Information
  - même si la version originale du template Django de cette page est polyvalente (par rapport aux types de contenu qui existent dans la démo), il faudra l'adapter si vous voulez créer un nouveau type de contenu pour les articles

Formulaire pour le module « Article » → cadre principal de la page qui affiche un article (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche un article

Sélecteur pour la source de données ?
Sélecteur de rubrique depuis une **variable d'environnement.**

L'identifiant de l'article (id_article) est donc la seule possibilité.

Formulaire pour le module « Article » → cadre principal de la page qui affiche un article (2)

Tous les chemins mènent à Rome … c'est particulièrement vrai pour <u>une page</u> du portail
En effet, toutes les routes mènent à un moment ou l'autre vers la page qui affiche un article

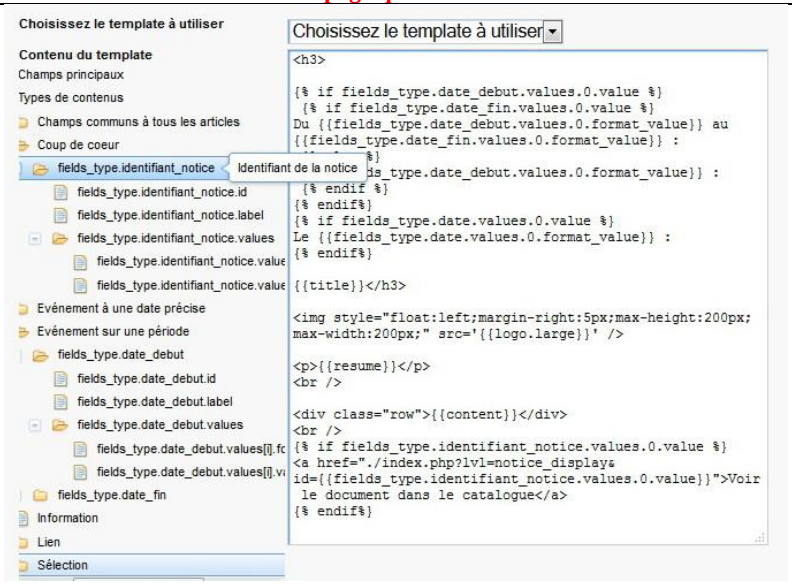
Dans le formulaire, on ne pose pas la question Construction du lien = où aller ensuite ? vers quelle page ? En effet, on est arrivé « au bout du parcours » : dans la page qui affiche un article.

Toutefois, il est malgré tout possible de poursuivre sa route, d'autres voies s'ouvrent ;-)

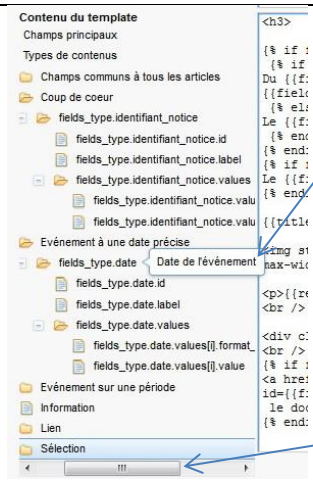
Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Le template gère l'affichage de l'article, quel que soit son type de contenu :

- coup de cœur (id d'une notice)
- sélection (id d'une étagère)
- événement à une date (date)
- événement sur une période (dates de début et de fin de l'activité)
- information (aucun champ personnalisé)



S'il s'agit d'un article « coup de cœur » lié à un identifiant de notice, le template gère aussi la navigation entre les pages, puisqu'il propose d'aller vers la page de l'OPAC qui affiche une notice.



Si vous dépliez les dossiers, vous voyez la liste des variables que vous pouvez utiliser dans les templates Django.
En cas de doute sur le sens d'une variable, déplacez le curseur : une infobulle s'affiche au survol de la souris, et vous explique la fonction de cette variable.

Exemples : les champs personnalisés liés aux types de contenu « Coup de cœur » et « Evénement à une date précise », ainsi que les noms des variables qui désignent ces champs.

Type coup de cœur
fields_type.identifiant_notice,
fields_type.identifiant_notice.id fields_type.identifiant_notice.label …

Type Evénement à une date précise
fields_type.date
fields_type.date.id fields_type.date.label …

N.B. La fenêtre de visualisation est assez étroite, donc il faut souvent scroller latéralement pour voir le nom complet d'une variable.

Template Django – Afficher le contenu d'un article

En fonction des types de contenu, on emploie les champs personnalisés qui y sont liés : date, date_debut, date_fin, identifiant_notice

Exemple : le champ date_debut correspond à la variable fields_type.date_debut.values.0.value dans le template Django

C'est pourquoi il vaut mieux restaurer la configuration de la démo : les templates sont ainsi immédiatement prêts à l'emploi.

Si ces champs personnalisés existaient dans votre base mais sous des noms différents, vous devriez adapter les templates !

```
<h3>

S'il s'agit d'un événement sur une période précise
{% if fields_type.date_debut.values.0.value %}

  {% if fields_type.date_fin.values.0.value %}
      Du {{fields_type.date_debut.values.0.format_value}}
      au {{fields_type.date_fin.values.0.format_value}} :

   {% else %}
      Le {{fields_type.date_debut.values.0.format_value}} :
  {% endif %}

{% endif%}

S'il s'agit d'un événement à une date précise
{% if fields_type.date.values.0.value %}
      Le {{fields_type.date.values.0.format_value}} :
{% endif%}

Quel que soit le type de contenu de l'article
{{title}}</h3>

<img style="float:left;margin-right:5px;max-height:200px;max-width:200px;"
src='{{logo.large}}' />

<p>{{resume}}</p>
<br />

<div class="row">{{content}}</div>
<br />

S'il s'agit d'un article « Coup de cœur » (lié à une notice)
 {% if fields_type.identifiant_notice.values.0.value %}
    <a href="./index.php?lvl=notice_display&id=
      {{fields_type.identifiant_notice.values.0.value}}">
      Voir le document dans le catalogue
    </a>
{% endif%}
```

```
S'il s'agit d'un événement sur une période précise
 Début de la 1e condition
   Si le champ date_debut a une valeur
      {% if fields_type.date_debut.values.0.value %}

     Début de la 2e condition
       Si le champ date_fin a une valeur
          {% if fields_type.date_fin.values.0.value %}
          On affiche : Du {{date_debut}} au {{date_fin}}
       Sinon {% else %}        (s'il n'y a pas de date de fin)
          On affiche : Le {{date_debut}}
       S'il n'y a pas de date de fin, on considère que l'événement a lieu
          à une date précise (celle du début), et on modifie le message
     Fin de la 2e condition {% endif%}
 Fin de la 1e condition   {% endif%}

S'il s'agit d'un événement à une date précise
 Début de la condition
   Si le champ date a une valeur
      {% if fields_type.date.values.0.value %}
       On affiche : Le {{date}}
 Fin de la condition   {% endif%}

Quel que soit le type de contenu de l'article, on affiche
   le titre de l'article        {{title}}
   le logo de l'article         <img style="… " src='{{logo.large}}' />
   le résumé de l'article       {{resume}}   (le chapeau)
   le contenu de l'article      {{content}}   (le texte complet)

S'il s'agit d'un article « Coup de cœur » (lié à une notice)
 Début de la condition
   Si le champ identifiant_notice a une valeur
     {% if fields_type.identifiant_notice.values.0.value %}
       On génère une balise <a href=" …">
          Le 1er élément est l'URL de destination (une adresse OPAC)
          Le 2e élément est la source = ce qui est cliquable
 Fin de la condition   {% endif%}
```

Vu la manière dont le template Django est rédigé (pour les coups de cœur), il génère une URL de destination qui est une adresse OPAC.

En fait, cette URL correspond au permalink de la notice dans l'OPAC.

Exemple : si identifiant_notice = 88 cela donne http://./index.php?lvl=notice_display&id=88

Voir le document dans le catalogue

Le template spécifie aussi ce qui est cliquable = le texte entre les balises et

Rappel : l'adresse http://./index.php?lvl=notice_display&id=88 **est une adresse relative par rapport à /pmb/opac_css**
elle correspond à une adresse absolue comme par exemple http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=88

Les conditions dans les templates Django : synthèse schématique

Condition « simple » : si … alors …

On n'envisage qu'un seul cas :
si la condition est vérifiée
Il n'y a rien à faire dans le cas contraire

```
{% if %}    Si la condition est vérifiée
      Instruction à exécuter
{% endif %}  Fin de la condition
```

Condition « si … alors … sinon … »
On envisage les 2 termes de l'alternative :
si la condition est vérifiée – si elle ne l'est pas

```
{% if %}    Si la condition est vérifiée
      Instruction n° 1
  {% else %}  Sinon
      Instruction n° 2
{% endif %}  Fin de la condition
```

On peut imbriquer 2 conditions l'une dans l'autre

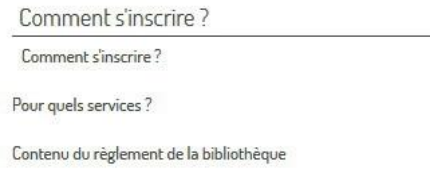
```
{% if %}    Si la condition n° 1 est vérifiée
      Instruction n° 1

   {% if %}    Si la condition n° 2 est vérifiée
         Instruction n° 2
   {% else %}  Sinon
         Instruction n° 3
   {% endif %}  Fin de la condition n° 2

{% endif %}  Fin de la condition n° 1
```

Dans ce template, il y a **un exemple d'imbrication entre 2 conditions** :
le cas des **événements qui se déroulent sur une période précise**.

Explications complémentaires : voir aide-mémoire HTML | Explications complémentaires : voir aide-mémoire – templates Django

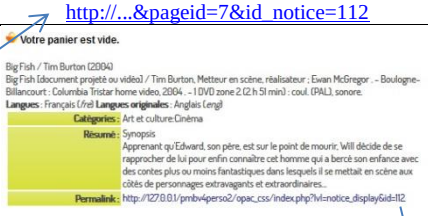
Accéder à la page qui affiche un article et une liste de notices … et poursuivre sa route …
Vue d'ensemble de cette page qui comprend trois cadres

- **le cadre « Affichage d'un article » est le cadre fondamental de cette page**
  - il est lié à la **variable d'environnement id_article**
  - il est forcément présent en toute circonstance
- **le cadre « Etagère associée à un article »** n'est visible que
  - si l'article est de **type Sélection**
  - si le **champ personnalisé lié à cet article (etagere)** contient une valeur qui correspond à **une étagère** remplie
- **le cadre « Notices dans les mêmes catégories qu'un article »**
  - n'est **lié à aucun type de contenu** d'article ; en théorie, il peut donc apparaître en toute circonstance
  - mais il n'est visible que s'il y a des **descripteurs communs à l'article et à certaines notices de la base**

Article - Type Information	**Article - Type Sélection**	
http://...&pageid=4&id_article=10	http://...&pageid=4&id_article=13	http://...index.php?lvl=notice_display&id=80
		
Un seul cadre : l'affichage d'un article. **Aucun champ personnalisé** n'est associé au **type de contenu Information** **Cet** article n'est **lié à aucune catégorie**, donc pas de carrousel de notices sur le même sujet. On est donc arrivé au bout du parcours, on n'ira nulle part ailleurs.	**Le type de contenu Sélection est lié à un champ personnalisé : etagere.** **Cet** article n'est **lié à aucune catégorie**, donc pas de carrousel de notices sur le même sujet. **Le template de notices qu'on associe au cadre Etagère liée à un article** prévoit 2 éléments cliquables : la vignette et le titre de la notice.	Si on clique sur la vignette ou le titre, on va vers la **page OPAC qui affiche la même notice.** **Le template de notices qu'on associe au cadre Etagère liée à un article** prévoit un lien vers l'OPAC : <a href="#permalink();">, ce qui explique l'URL de destination visible ci-dessus.

Conseil – Evitez de lier un article de type Sélection aux catégories, afin de ne pas avoir 2 cadres supplémentaires : le contenu de l'étagère **et** le carrousel de notices sur le même sujet, d'autant plus qu'on risque vraiment d'avoir des notices communes à ces 2 cadres.

Article - Type Coup de cœur		
http://...&pageid=4&id_article=17	http://...index.php?lvl=notice_display&id=88	**Template du cadre « Afficher un article »**
		URL de destination pour les Coups de cœur **Contenu de la balise <a href=" …>** Voir le document dans le catalogue* J'ai expliqué (en commentant le cemplate) que ceci générait une adresse OPAC comme http://...index.php?lvl=notice_display&id=88 Le template gère aussi la navigation entre les pages, puisqu'il spécifie ce qui est cliquable : le message « *Voir le document dans le catalogue* » est situé entre les balises et
Le type de contenu Coup de cœur est lié à un champ personnalisé : identifiant_notice. **Cet article est lié à une catégorie** (bibliothéconomie), donc on voit aussi le **carrousel de notices sur le même sujet.**	Si on clique sur ce lien « *Voir le document dans le catalogue* », on va **vers la page OPAC qui affiche la même notice.** **C'est prévu dans le template du cadre principal de cette page : Afficher un article.**	

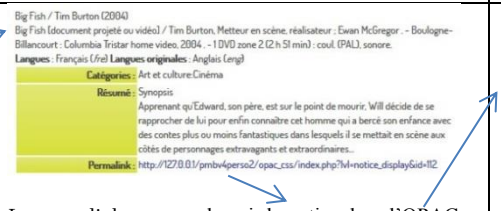
Article - Type Evénement à une date		**Article - Type Evénement sur une période**
http://...&pageid=4&id_article=11	http://...&pageid=7&id_notice=112	http://...&pageid=4&id_article=1
		
Ce type de contenu est lié à un champ personnalisé : date. **Cet article est lié à une catégorie** (cinéma), donc on voit aussi le **carrousel de notices sur le même sujet que l'article**. Un clic sur la 1e vignette (dans le carrousel mène vers la **page du portail qui affiche cette notice.**	Dans le **formulaire du cadre Notices dans les mêmes catégories qu'un article,** il y a la question « Construction du lien vers ? » et on a répondu : vers la page du portail qui affiche une notice, ce qui explique l'URL de destination http://...&pageid=7&id_notice=112 Le **template de ce cadre** prévoit 2 éléments cliquables : la vignette et le titre de la notice.	**Ce type de contenu est lié à 2 champs personnalisés : date_debut et date_fin.** **En outre, cet article est lié à une catégorie** (Afrique), donc on voit aussi le **carrousel de notices sur le même sujet** que l'article. On pourrait donc poursuivre sa route en cliquant sur un titre ou une vignette.

En outre, **si on est dans la page du portail qui affiche une notice**, on peut même **aller vers la page OPAC qui affiche la même notice** !
En effet, à la dernière ligne de la page du portail qui affiche une notice, on suggère de cliquer sur le permalink …

Conclusion : la page qui affiche un article et une liste de notices est loin d'être une voie sans issue !!

PAGE DU PORTAIL QUI AFFICHE UNE NOTICE

Comment accéder à la page du portail qui affiche une notice ?
Le « chemin habituel » part d'un carrousel de notices.
L'URL de destination contient le numéro d'id de la page (7) et la variable d'environnement liée à ce type de page : id_notice
Exemple : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=112

Carrousel de nouveautés (accueil) Carrousel « sur le même sujet »	Adresse : page du portail qui affiche une notice http://... &pageid=7&id_notice=112	Adresse OPAC = permalink http://...?lvl=notice_display&id=112
	 Le permalink propose de voir la notice dans l'OPAC	

Dans la démo, la page du portail qui affiche une notice contient (à la dernière ligne) un lien vers le permalink de la notice.
Dès qu'on passe dans l'OPAC, on voit tous les champs de la notice, ainsi que les données des exemplaires liés à cette notice.
Données liées aux exemplaires : cote de rangement, numéro d'exemplaire, localisation, section, disponibilité …

Toutefois, on ne « transite » pas toujours par la page du portail qui affiche une notice.
Dans certains cas, le lien se fait directement vers la page de l'OPAC qui affiche une notice.

Comment accéder directement à la page de l'OPAC qui affiche une notice ?
Le « chemin habituel » part d'une liste de notices, ou d'un article « Coup de cœur » - dans la page qui affiche un article.
L'URL de destination correspond donc au permalink de la notice dans l'OPAC.
Exemple : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=notice_display&id=80

Liste de nouveautés (fiction jeunesse)	http://...?lvl=notice_display&id=80	
 L'affichage de cette liste est géré par un template de notices qui crée un lien vers l'OPAC.		**Lorsque je parle de « chemin habituel », je fais référence à la manière dont la démo est paramétrée.** **Ce n'est pas un parcours obligatoire !** Vous ne devez pas nécessairement: • aller vers la page du portail qui affiche une notice • aller vers la page de l'OPAC qui affiche une notice dans le même genre de circonstances que dans la démo. Je rappelle que le choix de la page de destination se détermine dans le cadre **précédent**. Par exemple, dans le cadre qui affiche une liste ou un carrousel de notices, ou dans le cadre qui affiche un article. Vous pouvez paramétrer autrement le chemin qui mène d'une page à l'autre. Vous pouvez aussi modifier la mise en forme des données dans la page du portail qui affiche une notice.
Article de type Coup de cœur	http://...?lvl=notice_display&id=88	
 Le lien propose de voir la notice dans l'OPAC		

La page du portail qui affiche une notice n'est pas une page où on verrait « quelque part » le titre d'une notice, un titre sur lequel on pourrait cliquer, ni une page où on afficherait une notice précise (et pas les autres).

- **c'est une page où on voit le contenu d'une notice, quel que soit son numéro d'id**
  - **ce qu'on voit dépend du paramétrage du cadre « afficher une notice », plus précisément, du template Django**
    - dans la démo, cette page n'affiche pas tous les champs de la notice, ni les données liées aux exemplaires
    - c'est pourquoi il y a un lien « permalink » afin d'accéder aisément à la version OPAC de cette page
  - **cette page a le numéro d'id 7 dans la démo** (ce numéro ne change pas)
  - la **variable d'environnement id_notice** transmet le **numéro d'id de la notice à afficher** (ce numéro varie)
    - ex. http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=88

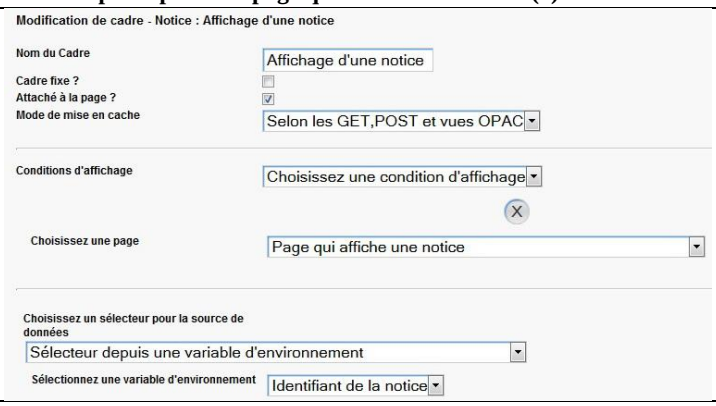
Formulaire pour le module « Notice » → cadre principal de la page qui affiche une notice (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche une notice

Sélecteur pour la source de données ?
Sélecteur de rubrique depuis une **variable d'environnement.**

L'identifiant de la notice (id_notice) est donc la seule possibilité.



Formulaire pour le module « Notice » → cadre principal de la page qui affiche une notice (2)

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Un seul template Django est disponible : l'affichage d'une notice, et il brille par sa concision : 2 lignes !

Choix du template de notice : template par défaut.

Une liste de notices ou un carrousel (notices ou articles) sont toujours bâtis sur le modèle de la boucle {% for record in records %}

- Une boucle qui va du premier au dernier enregistrement (a record), et cet enregistrement peut être un article ou une notice.

PMB pose d'office la question « Quel template de notice employer ? » dans deux cas :

- Si le template Django contient la **boucle {% for record in records %}** **<u>ou bien</u>** s'il faut **afficher une seule notice.**

Toutefois, cette possibilité de recourir à un template prédéfini n'existe concrètement que si 2 conditions sont remplies :

- **Le cadre doit être créé à partir des modules Liste de notices** (carrousel ou liste) **ou Afficher une notice, mais pas à partir du module Liste d'articles** (carrousel ou liste). Choisir un template de notice pour afficher un article n'aurait d'ailleurs aucun sens !
- Le template Django doit contenir « quelque part » la **variable {{record.content}}** (ce qui exclut par ex. le module Carrousel)

Dans ce cas-ci, la question « quel template de notice employer ? » a un sens, car toutes les conditions sont vérifiées.

Si le template par défaut ne vous convient pas, vous pouvez créer un nouveau template de notice, ce qui vous permettra de modifier l'aspect général de la page du portail, et/ou d'augmenter le nombre de champs visibles à l'écran.
Le template qui existe déjà (Portail – Liste de notices) n'affiche en effet qu'un nombre fort réduit de champs, c'est celui qu'on avait employé pour le cadre : Notices d'une étagère associée à un article.

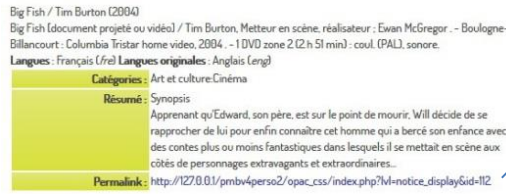
Template Django – Afficher le contenu d'une notice

```
<p>{{record.header}}</p>
<blockquote>{{record.content}}</blockquote>
```

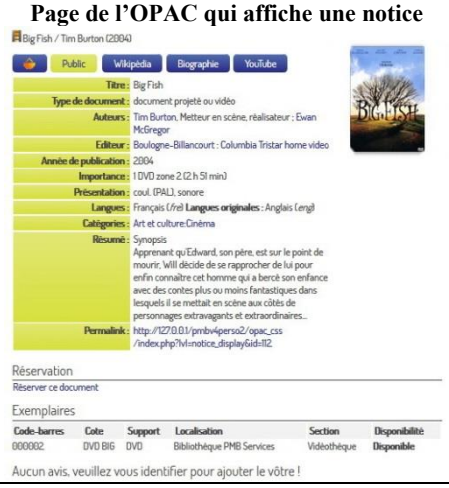
{{record.header}} = l'en-tête
Titre, auteur et date de publication.

{{record.content}}
Les données essentielles de la notice : titre, auteur, éditeur, date de publication, collation, langues, catégories, résumé et permalink.

Page du portail qui affiche une notice – Template par défaut



Page de l'OPAC qui affiche une notice



Rappel : un clic sur le permalink mène vers la page de l'OPAC qui affiche la même notice.

PAGE DU PORTAIL QUI AFFICHE UNE LISTE D'ÉVÉNEMENTS (SÉLECTION D'UNE DATE DANS L'AGENDA)

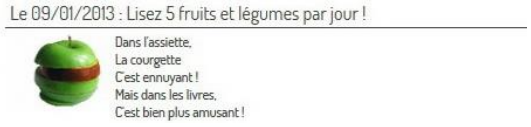
Comment accéder à la page du portail qui affiche une liste d'événements ?
Le « chemin habituel » part de l'agenda : on clique sur une date dans le calendrier, qui est affiché en page d'accueil.

Une liste d'événements est une liste d'articles dont le type de contenu est Evénement (à une date ou sur une période).

Si plusieurs événements correspondent à la date sélectionnée, on va vers la page qui affiche une liste d'événements.
L'URL de destination contient le numéro d'id de la page (9) et la variable d'environnement liée à ce type de page : id_date
Exemple : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9&id_date=2013-01-09

Toutefois, on ne va pas toujours vers la page qui affiche une liste d'événements.
Si un seul événement correspond à la date sélectionnée, on va directement vers la page qui affiche un article.
L'URL de destination contient le numéro d'id de la page (4) et la variable d'environnement liée à ce type de page : id_article
Exemple : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=1

Pour comprendre ces captures d'écran : 2 activités ont été annoncées dans 2 articles de type Evénement
Un événement à une date précise (Lisez 5 fruits …), et un autre qui s'étale sur une période (Cultures d'Afrique …)



Date sélectionnée : 09/01/2013
2 événements à afficher

Date sélectionnée : 01/02/2013
Un seul événement à afficher

Date sélectionnée : 01/04/2013
Aucun événement à afficher

Comme vous le voyez, 3 cas de figure peuvent se présenter ensuite.

Page du portail qui affiche une liste d'événements → **une liste = au moins 2 !**

Page du portail qui affiche l'article lié à l'unique événement concerné

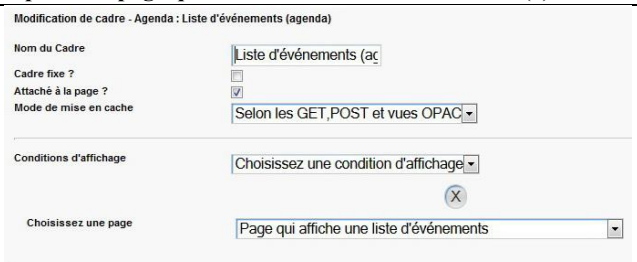
Le clic sur la date n'a rien déclenché
Pas de changement de page

N.B. Si on clique sur le logo d'un événement dans cette liste, on va vers la page du portail qui affiche l'article lié à cette activité.

Formulaire pour le module « Agenda » → cadre principal de la page qui affiche une liste d'événements (1)

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche une liste d'événements

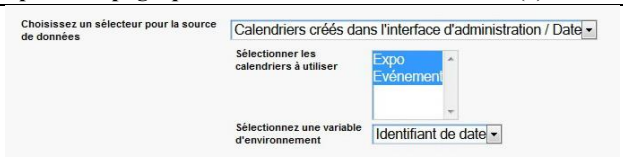


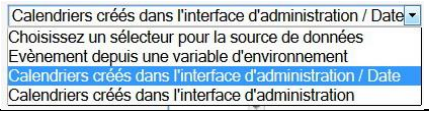
Formulaire pour le module « Agenda » → cadre principal de la page qui affiche une liste d'événements (2)

Sélecteur pour la source de données ?
Calendriers créés dans l'interface d'administration/Date.
C'est la bonne option dans ce cas-ci, car le critère de date doit déterminer quelles activités il faut afficher.

On sélectionne les **2 calendriers** qui ont été **définis dans l'interface d'administration : Expo et Evénement**

Sélecteur depuis une **variable d'environnement.**
Une seule possibilité : **identifiant de date** (id_date)





Page du portail page qui affiche une liste d'événements : 2 sélecteurs possibles pour la source de données

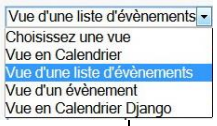
- **Si on choisit « Calendriers créés dans l'interface d'administration »**, PMB va afficher la liste de **tous les articles de type Evénement**, quelle que soit la date de l'événement
- **Si on choisit « Calendriers créés dans l'interface d'administration / Date »**, PMB va afficher la liste de **certains articles de type Evénement**, **sélectionnés d'après la date sur laquelle on a cliqué dans l'agenda**. On ne verra donc que
  - les articles qui correspondent aux activités à cette date (le calendrier appelé « Evénement »)
  - les articles qui correspondent aux activités sur une période qui inclut cette date (le calendrier appelé « Expo »)

Dans ce contexte-ci, il faut donc choisir « Calendriers créés dans l'interface d'administration / Date »

N.B. Je présente d'abord ce cadre-ci dans sa totalité, ensuite j'expliquerai comment créer un calendrier dans l'interface d'administration.

Formulaire pour le module « Agenda » → cadre principal de la page qui affiche une liste d'événements (3)

Choisir une vue = comment afficher la liste ?
Vue d'une liste d'événements



Construction du lien vers un événement
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django à utiliser ?
quels champs afficher ? comment gérer la mise en forme et les liens éventuels vers « ailleurs » ?

Attention ! Le mot « événement » a plusieurs significations dans ce contexte !

Un **Evénement** est un **type de contenu d'article** : événement à une date précise, ou événement qui s'étale sur un période déterminée (les autres types de contenu éditorial sont le coup de cœur, la sélection et l'information). Voir configuration du contenu éditorial.

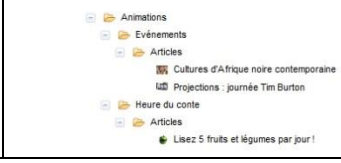
- Une **liste d'événements** est une **liste d'articles de type Evénement :** suivant le cas, **une liste complète ou une liste sélective** (sur base de la date choisie dans l'agenda). Voir explications ci-dessus : différence entre 2 sélecteurs pour la source de données.
- Le cadre **Liste d'événements** est généré par le **module Agenda.**

« **Evénement** » est le **nom donné dans la démo à un calendrier qu'on a défini au moment de créer l'agenda (dans l'interface d'administration)**

- Dans cet exemple, on a décidé d'afficher les 2 calendriers qui avaient été prévus : Expo (événement sur une période) et Evénement (événement à une date précise).
- Ces calendriers doivent être définis sur base des types de contenu des articles.

« **Evénements** » est aussi le **nom donné dans la démo au titre d'une rubrique**

- Dans cet exemple, la rubrique « Evénements » contient 2 articles de type Evénement (« Cultures d'Afrique noire contemporaine » et « Journée Tim Burton »), mais un autre article qui est lui aussi de type Evénement (« Lisez 5 fruits et légumes par jour ») dépend de la rubrique « Heure du conte »



Voir OPAC « style portail » : les exemples n° 4 (afficher une liste d'événements) et n° 5 (afficher le contenu d'une rubrique)

Template Django – Afficher une liste d'événements

```
<div>

{% for event in events %}
<h3>

Qu'il s'agisse d'un événement à une date ou sur une période
Si le champ date de début a une valeur
{% if event.event_start.format_value %}

Si le champ date de fin a une valeur
 {% if event.event_end.format_value %}
      Du {{event.event_start.format_value}}
      au {{event.event_end.format_value}}

   {% else %}

      Le {{event.event_start.format_value}}
    {% endif %}

{% endif%}

 : {{event.title}}     N.B. Ce titre est précédé du signe de ponctuation :
</h3>

<blockquote>
      <a href='{{event.link}}'><img style="float:left;margin-right:5px;"
src='{{event.logo.vign}}'/></a>

<p>{{event.resume}}</p>
<br/>

<div class="row">{{event.content}}</div>

</blockquote>

<br />

{% endfor %}

</div>
```

Explications complémentaires : voir aide-mémoire HTML

Début de la balise

Début de la boucle : **{% for event in events %}**
Permet d'afficher **les événements prévus**
on a donc une "boucle" qui va du premier au dernier événement

Qu'il s'agisse d'un événement à une date ou sur une période
Début de la 1e condition
Si le champ date de début a une valeur
{% if event.event_start.format_value %}

Début de la 2e condition
Si le champ date de fin a une valeur
{% if event.event_end.format_value %}
On affiche **Du** {{event.event_start.format_value}}
au {{event.event_end.format_value}}

Sinon {% else %} (s'il n'y a pas de date de fin)
On affiche : **Le** {{event.event_start.format_value}}
S'il n'y a pas de date de fin, on considère que l'événement a lieu à une date précise (celle du début), et on modifie le message
Fin de la 2e condition {% endif%}
Fin de la 1e condition {% endif%}

On affiche le titre de l'article {{event.title}}

On génère une balise
 Lien pour un événement
URL de destination : page du portail qui affiche un article

la source = ce qui est cliquable : le logo

On affiche
le résumé de l'article {{event.resume}} (le chapeau)
le contenu de l'article {{event.content}} (le texte complet)

Fin de la boucle : **{% endfor %}**

Fin de la balise

Explications complémentaires : voir aide-mémoire – templates Django

Dans ce template, on a « fusionné » les 2 types d'événements dans la boucle {% for event in events %} … {% endfor %}

- {{event.event_start.format_value}} reprend les valeurs du champ date (événement à une date précise) ou du champ date_debut (le début d'un événement qui s'étale sur une période)
- {{event.event_end.format_value}} reprend la valeur du champ date_fin (la fin d'un événement qui s'étale sur une période)

Cela correspond d'ailleurs au paramétrage des calendriers dans l'interface d'administration.

Remarques à propos de la version d'origine de ce template : variantes pour la balise <a href='{{event.link}}'>

Version abrégée d'une partie du template d'origine

```
: {{event.title}}
<a href='{{event.link}}'><img style="float:left;margin-right:5px;" src='{{event.logo.vign}}'/></a>
{{event.resume}}
{{event.content}}
```

Variante n° 1 : ne pas afficher le contenu de l'article

Remplacer {{event.content}} par ceci
Lire la suite …

Variante n° 2 : clic possible aussi sur le titre
Adaptation conseillée aux distraits qui pourraient oublier de prévoir un logo pour ce genre d'article !

Remplacer **:** {{event.title}} par ceci
: {{event.title}}

Cette balise … gère la navigation vers la page qui affiche un article, mais on aura parfois l'impression de « déjà-vu ».

En effet, la page qui affiche une liste d'événements montre déjà tous les champs de l'article : titre, logo, résumé et contenu. Donc, aller vers la page qui affiche un article n'apportera « un plus » que si on a lié cet article aux catégories, et qu'on voit en outre le carrousel de notices sur le même sujet. Sinon, changer de page ne sert à rien.

Vous pourriez peut-être remplacer la variable {{event.content}} par un lien vers l'article complet. Mais cela obligerait le lecteur à cliquer pour lire la suite, alors qu'avec le template d'origine, il dispose dès le début de toutes les infos sur l'activité qui a été organisée. A vous de juger !!

Aller ailleurs, vers la page qui affiche un article ne sera pas toujours aisé !
En effet, seul le logo de l'événement est cliquable, donc si vous n'avez pas prévu de logo pour cet article, rien n'est cliquable ! Impossible d'aller ailleurs …

Par « mesure de sécurité », vous pourriez veiller à ce que le titre de l'article soit lui aussi cliquable, on ne sait jamais !

CRÉER ET DÉFINIR DES CALENDRIERS DANS L'INTERFACE D'ADMINISTRATION

Module Agenda – Etape préliminaire : création et paramétrage des calendriers

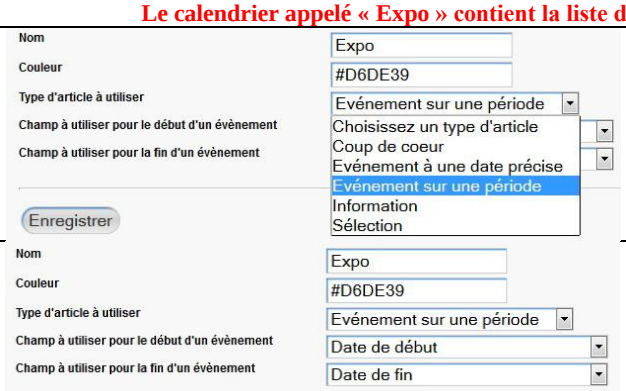
Pour accéder à l'interface d'administration d'un module spécifique, cliquez sur le nom de ce module dans le menu latéral de gauche.

Général → Gestion du module – dans ce cas-ci, Agenda
Création et définition des calendriers

Vues → Gestion des templates associés à ce module

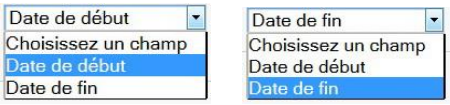
Clic sur le bouton « Général » : la 1e chose à faire est de créer des calendriers, sur base des types de contenu des articles.
Il faut bien sûr choisir des types de contenu liés à une ou plusieurs dates, vu que le but est de publier un agenda des activités.

Le calendrier appelé « Expo » contient la liste des événements qui s'étalent sur une période déterminée.

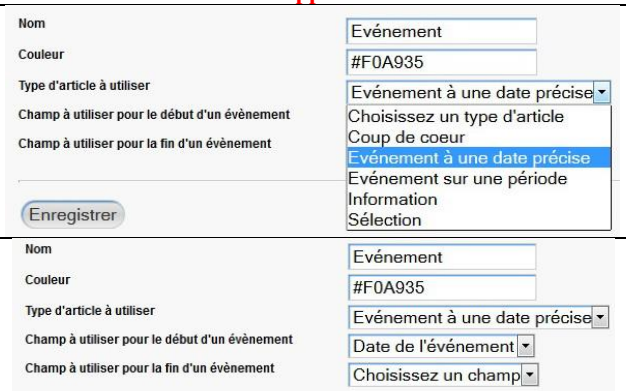


Puisqu'il s'agit d'un événement sur une période, il faut ensuite préciser les dates de début et de fin de l'activité : le contenu des champs personnalisés appelés date_debut et date_fin.

De toute façon, dès que vous avez sélectionné un événement sur une période, la liste des champs personnalisés que vous pourriez choisir se réduit automatiquement, vu la configuration du contenu éditorial.



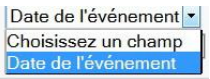
Le calendrier appelé « Evénement » contient la liste des événements qui se déroulent à une date précise.



Bien qu'il s'agisse d'un événement à une date précise, on vous propose de fournir le même genre d'infos que si c'était un événement sur une période : les dates de début et de fin de l'activité.

Il suffit bien entendu de compléter le 1er champ, et de sélectionner le champ personnalisé appelé date.

De toute façon, dès que vous avez sélectionné un événement à une date, la liste des champs personnalisés que vous pourriez choisir se réduit automatiquement, vu la configuration du contenu éditorial.



On envisage donc les 2 types d'événements de la même manière : avec des dates de début et de fin. Ceci correspond d'ailleurs à une remarque que je venais de faire en commentant le template de la page qui affiche une liste d'événements.
On a « fusionné » les 2 types d'événements dans la boucle {% for event in events %} … {% endfor %}

- {{event.event_start.format_value}} reprend les valeurs du champ date (événement à une date précise) ou du champ date_debut (le début d'un événement qui s'étale sur une période)
- {{event.event_end.format_value}} reprend la valeur du champ date_fin (la fin d'un événement qui s'étale sur une période)

Gestion des couleurs affectées aux calendriers

Chaque calendrier est associé à une couleur, que vous pouvez choisir (en fonction de votre charte graphique).
Par convention, en HTML, les codes des couleurs sont indiqués en format hexadécimal (en base 16).

Ces codes commencent par le symbole # et 16 caractères sont utilisables : les chiffres de 0 à 9 et les lettres des A à F (obligatoirement en majuscules).

De nombreux sites vous aident à déterminer quel code il faut employer, par exemple http://html-color-codes.info/Codes-couleur-HTML/
Vous pouvez travailler « dans les deux sens » :

- Pour connaître le code HTML d'une couleur, il suffit de cliquer sur celle-ci dans une palette de couleurs
- Pour savoir à quelle couleur un code HTML correspond, il suffit d'introduire ce code dans un champ de saisie

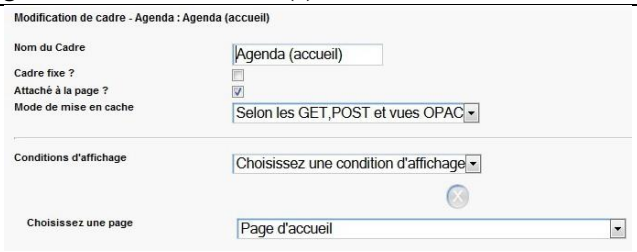
AGENDA : CALENDRIER EN PAGE D'ACCUEIL

Vous venez de franchir les 2 étapes préliminaires :

- Vous avez configuré la page qui affiche une liste d'événements : elle contient un cadre issu du module Agenda
- Vous avez créé et paramétré 2 « calendriers » dans l'interface d'administration : « Expo » et « Evénement »

Vous pouvez employer à nouveau le module Agenda, mais cette fois-ci, pour afficher un calendrier en page d'accueil.

Le module « Agenda » a généré le cadre « agenda » : le calendrier visible en page d'accueil.		
Date sélectionnée : 09/01/2013 **2 événements** à afficher	Date sélectionnée : 01/02/2013 **Un seul événement** à afficher	Date sélectionnée : 01/04/2013 **Aucun événement** à afficher

Formulaire pour le cadre « Agenda » : vue en calendrier (1)	
Nom précis donné au cadre ? **Fixe (toutes les pages) ou attaché à une page ?** **Mode de mise en cache ?** **Conditions d'affichage ?** **Dans quelle page afficher le cadre ?** Page d'accueil	

Formulaire pour le cadre « Agenda » : vue en calendrier (2)	
Sélecteur pour la source de données ? **Calendriers créés dans l'interface d'administration**. C'est la bonne option dans ce cas-ci, car le critère de date ne peut pas avoir d'impact sur les événements à afficher. On doit voir, mois par mois, toutes les activités qui ont été annoncées dans des articles de type Evénement. On sélectionne les **2 calendriers** qui ont été **définis dans l'interface d'administration : Expo et Evénement**	

Pour la page du portail qui affiche une liste d'événements, il faut choisir Calendriers créés dans l'interface d'administration / Date
On obtient la liste de certains articles de type Evénement, sélectionnés d'après la date sur laquelle on a cliqué dans l'agenda.
L'option « Date » agit comme un filtre, on ne verra donc que

- les articles qui correspondent aux activités à cette date (le calendrier appelé « Evénement »)
- les articles qui correspondent aux activités sur une période qui inclut cette date (le calendrier appelé « Expo »)

Il faut donc une URL avec variable de date pour appeler la page qui affiche une liste d'événements et obtenir une liste sélective.
Exemple : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9&id_date=2013-01-09

Attention au paramétrage et aux URL
Si on appelle cette page en spécifiant seulement son numéro identifiant, mais sans transmettre le paramètre de date
Exemple : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9
Ou si appelle cette page en transmettant son numéro identifiant et le paramètre de date, mais en ayant mal configuré le sélecteur
Exemple : choisir « Calendriers créés dans l'interface d'administration » (donc sans l'option Date)
Dans un cas comme dans l'autre, on verra la liste de toutes les activités annoncées dans des articles, au lieu d'une liste sélective.

Pour l'agenda (vue en calendrier), il faut choisir Calendriers créés dans l'interface d'administration (sans l'option Date)
En effet, le calendrier doit permettre de repérer, mois par mois, toutes les dates qui correspondent à des activités annoncées dans des articles de type Evénement :

- une couleur pour les activités à une date précise (c'est le calendrier appelé « Evénement »)
- une autre couleur pour les activités sur une période (c'est le calendrier appelé « Expo »)

Formulaire pour le cadre « Agenda » : vue en calendrier (3)

On pose deux fois la question « Construction du lien vers ? » :
où aller s'il y a un seul événement ? où aller s'il y a plusieurs événements ?

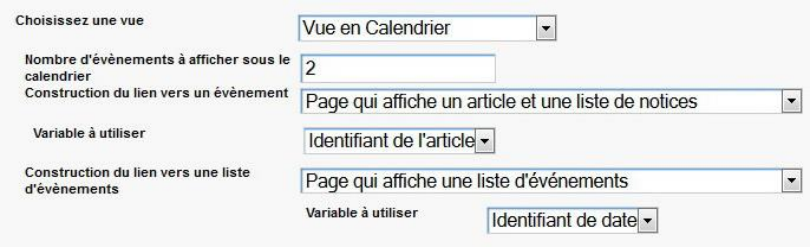
Vue ? Nombre d'événements à afficher ?
Vue en calendrier – 2 événements

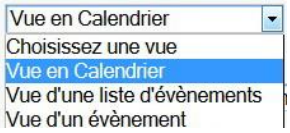
Construction du lien vers un seul événement
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article ici la seule option.

Construction du lien vers plusieurs événements
= où aller ensuite ? vers quelle page ?
Page qui affiche une liste d'événements

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de date est ici la seule option.





N.B. Les deux questions « Construction du lien vers ? » ne s'excluent pas, car le cadre doit être polyvalent, et traiter les deux cas possibles.

Vous pouvez cliquer sur n'importe quelle date visible dans l'agenda (en page d'accueil)
La page du portail qui affiche une liste d'événements est une page où on voit une série d'événements
quelle que soit la date choisie … mais à condition qu'il y ait des articles de type Evénement pour annoncer les activités !!

Attention ! On ne va pas nécessairement vers la page qui affiche une liste d'événements !
En effet, la date sélectionnée dans l'agenda peut être liée à zéro, une ou à plusieurs activités qui se déroulent précisément à la date choisie, ou qui s'étalent sur une période qui englobe la date choisie.

Si **aucun événement** n'est prévu pour cette date, rien ne se passe !
On reste dans la page où on voit le calendrier : la page d'accueil, dans ce cas-ci.

Si **un seul événement** est prévu, on va directement **vers la page qui affiche un article**

- Dans la démo, la page qui affiche un article a le numéro d'id 4 (ce numéro ne change pas)
- **la variable d'environnement id_article** transmet le numéro d'id de l'article à afficher (ce numéro varie donc)
- Exemple pour l'article « Cultures d'Afrique noire contemporaine »
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=1

Ensuite, si on clique sur une vignette (dans le carrousel), on va vers la page qui affiche une notice.

- Ex. http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=70

Si **plusieurs événements** sont prévus, on va **vers la page qui affiche une liste d'événements**

- Une **liste d'événements** comporte **au minimum deux événements** !
- Dans la démo, la page qui affiche une liste d'événements a le numéro d'id 9 (ce numéro ne change pas)
- **la variable d'environnement id_date** transmet la date sélectionnée (cette date varie donc)
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9&id_date=2013-01-09
- Rappel : le cadre principal de cette page est créé à partir du **module agenda**

Ensuite, si on clique sur le logo d'un événement, on va vers la page qui affiche l'article adéquat.

- Exemple pour l'article « Lisez 5 fruits et légumes par jour »
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=2

Les dates des événements ne sont pas le seul critère qui détermine l'affichage ou non de tel ou tel article de type Evénement !
Le statut de publication d'un article et ses dates de publication ont aussi un impact filtrant ! Voir configuration du contenu éditorial
Vous ne voyez donc pas tous les articles, seulement ceux dont le statut est « publié » et dont les dates de publication sont valides.

PAGE DU PORTAIL QUI AFFICHE UNE LISTE D'ARTICLES (LE RÉSULTAT D'UNE RECHERCHE DANS LE CONTENU ÉDITORIAL)

Comment accéder à la page qui affiche le résultat d'une recherche ?
Le « chemin habituel » part de la barre de recherche (visible dans la zone introduction).
Ce champ de saisie offre 2 possibilités : recherche dans le catalogue ou recherche dans le contenu éditorial.

Si on effectue une recherche dans les articles, on va vers la page du portail qui affiche une liste d'articles (résultat de recherche)
L'URL de destination contient seulement le numéro d'id de cette page (11) qui n'est liée à aucune variable d'environnement
Exemple : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=11

Toutefois, on ne va pas toujours vers la page du portail qui affiche une liste d'articles (résultat de recherche).
Si on effectue une recherche dans le catalogue, on va vers la page de l'OPAC qui affiche le résultat d'une recherche
L'URL de destination correspond donc à une adresse de type OPAC
Exemple : http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=more_results

Recherche dans le contenu éditorial	Recherche dans le catalogue

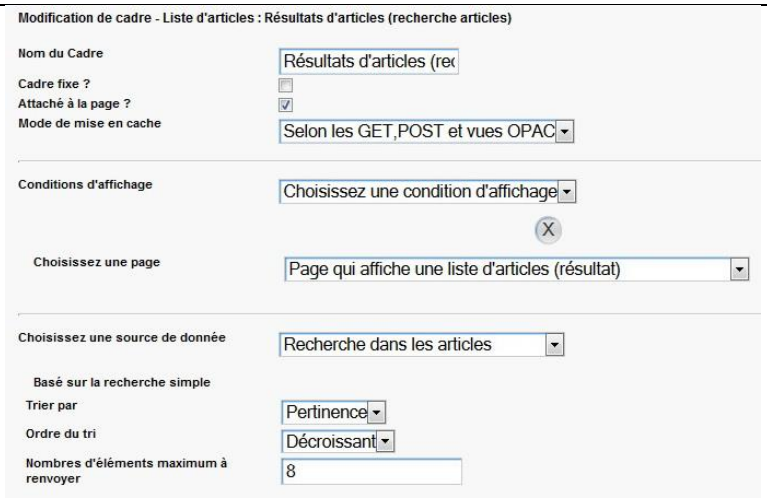
Formulaire pour le module « Liste d'articles » → cadre principal de la page qui affiche un résultat (1)
Il s'agit du résultat d'une recherche dans le contenu éditorial

Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Mode de mise en cache ?
Conditions d'affichage ?

Dans quelle page afficher le cadre ?
Page qui affiche une liste d'articles (résultat)
Source de données ?
Recherche simple dans les articles
Critère et ordre de tri ?
Par pertinence – ordre décroissant

Nombre maximum d'éléments à renvoyer ?
= nombre maximum d'articles à afficher

Dans le cas d'une liste ou d'un carrousel d'articles, la valeur zéro affiche tous les articles qui répondent au critère de sélection.



Choix de la source de données	Choix du critère de tri

Il s'agit ici d'une liste dynamique, créée au moment même, en fonction de ce qu'on a pu trouver dans les articles.
La liste des articles à afficher peut changer de jour en jour, en fonction de l'évolution du contenu éditorial.

Formulaire pour le module « Liste d'articles » → cadre principal de la page qui affiche un résultat (2)
Il s'agit du résultat d'une recherche dans le contenu éditorial

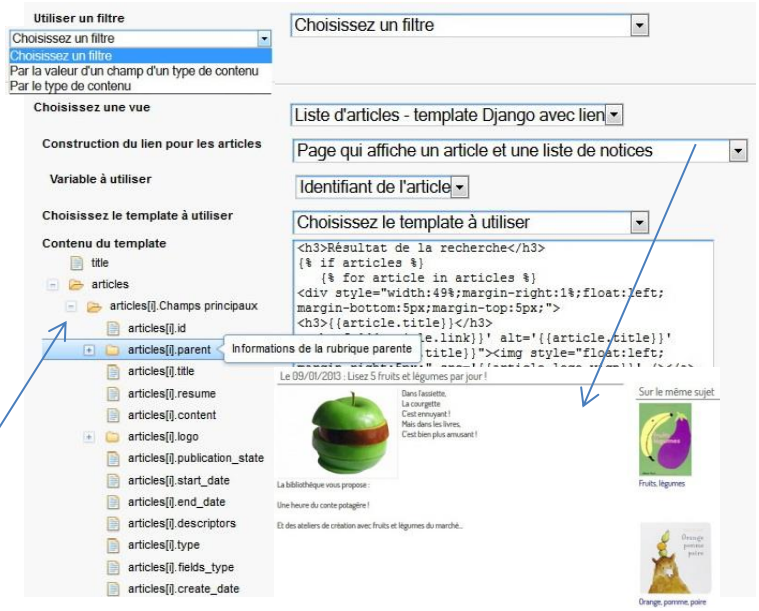
Filtre
Aucun filtre sélectionné

Choisir une vue = comment afficher la liste ?
Liste d'articles – template Django avec lien
Autre possibilité : Vue en carrousel

Construction du lien vers un article
= où aller ensuite ? vers quelle page ?
Page qui affiche un article et une liste de notices

Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de l'article est ici la seule option.

Template Django
Rappels : si vous dépliez les dossiers, vous pouvez voir la liste des variables utilisables dans le template.
En outre, une infobulle (affichée au survol de la souris) vous explique la fonction de chaque variable.



Un cadre issu du module « Liste d'articles » mène toujours vers la page qui affiche un article et une liste de notices.
Je rappelle que c'est une page multifonctions, conçue pour afficher tous les types de contenu d'articles de la démo.

Template Django – Afficher une liste d'articles : résultat d'une recherche dans le contenu éditorial

```
<h3>Résultat de la recherche</h3>

{% if articles %}

  {% for article in articles %}
  <div style="width:49%;margin-right:1%;float:left;margin-bottom:5px;margin-top:5px;">

<h3>{{article.title}}</h3>

<a href='{{article.link}}' alt='{{article.title}}' title="{{article.title}}"><img style="float:left;margin-right:5px;" src='{{article.logo.vign}}' /></a>

<p>{{article.resume}}</p>

<a style="text-align:right;" href="{{article.link}}">Lire la suite...</a>
</div>
  {% endfor %}

<div class="row"></div>

 {% else %}

<br />
Pas de résultats
<br />

<a href="./index.php?search_type_asked=simple_search">>> Rechercher dans le catalogue</a>

{% endif %}
```

Explications complémentaires : voir aide-mémoire HTML

Titre : Résultat de la recherche

Début de la condition **{% if articles %}**
Si la recherche a donné des résultats, s'il y a des articles à afficher

Début de la boucle : **{% for article in articles %}**
Permet d'afficher **les articles adéquats**
on a donc une "boucle" qui va du premier au dernier article

On affiche le titre de l'article {{article.title}}

On génère une balise ….
<a href='{{article.link}}'> Lien pour un article
URL de destination : page du portail qui affiche un article

la source = ce qui est cliquable : le logo

On affiche le résumé de l'article {{article.tresume}}

On génère une balise ….
<a href='{{article.link}}'> Lien pour un article
URL de destination : page du portail qui affiche un article
la source = ce qui est cliquable : le message « Lire la suite »

Fin de la boucle **{% endfor %}**

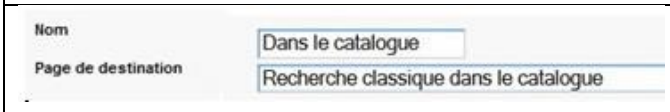
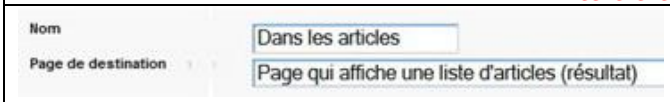
Sinon **{% else %}**

On affiche le message « Pas de résultats »
On affiche un lien → une recherche simple dans le catalogue

Fin de la condition : **{% endif %}**

Explications complémentaires : voir aide-mémoire – templates Django

CONFIGURER LA RECHERCHE DANS L'INTERFACE D'ADMINISTRATION

Module Recherche – Etape préliminaire : configuration de la recherche	
	Pour accéder à l'interface d'administration d'un module spécifique, cliquez sur le nom de ce module dans le menu latéral de gauche. **Général** → Gestion du module – dans ce cas-ci, Recherche Paramétrage de la recherche Choix des pages de destination
Recherche dans le catalogue	
	Donner un nom à ce type de recherche Choisir la page de destination : la page de l'OPAC qui affiche le résultat d'une recherche dans le catalogue
Recherche dans les articles	
	Donner un nom à ce type de recherche Choisir la page de destination : la page du portail qui affiche une liste d'articles = le résultat d'une recherche dans les articles
C'est fini, vous venez de configurer le module recherche ! **Particularité de ce module : la question « Construction du lien vers … ? » n'est pas posée dans un cadre issu de ce module.** **C'est dans l'interface d'administration que vous déterminez les pages de destination (OPAC et portail).**	

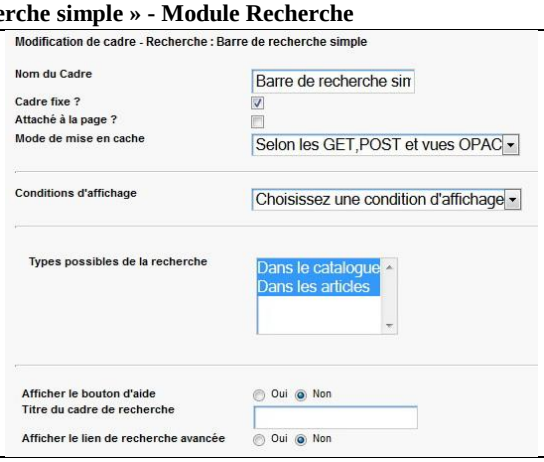
INSÉRER UN CADRE « BARRE DE RECHERCHE » (MODULE RECHERCHE)

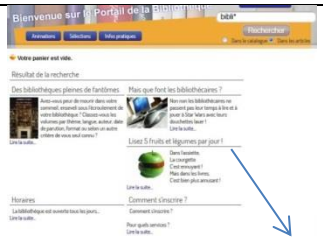
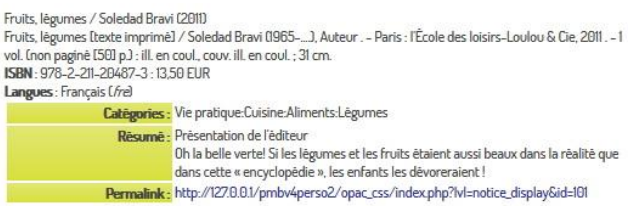
Vous venez de franchir les 2 étapes préliminaires :

- Vous avez configuré la page qui affiche le résultat d'une recherche dans le contenu éditorial : elle contient un cadre issu du module Liste d'articles
- Vous avez configuré les 2 aspects de la recherche dans l'interface d'administration (recherche dans le catalogue et recherche dans le contenu éditorial), et déterminé la page de destination dans chaque cas.

Vous pouvez enfin employer le module Recherche pour insérer une barre de recherche dans la zone Introduction.

Le module « Recherche » a généré le cadre « Barre de recherche », visible dans la zone Introduction	
Recherche dans les articles	Recherche dans le catalogue

Formulaire pour le cadre « Barre de recherche simple » - Module Recherche	
Nom précis donné au cadre ? **Fixe (toutes les pages) ou attaché à une page ?** **Un cadre fixe !! Dans la zone Introduction** **Mode de mise en cache ?** **Conditions d'affichage ?** **Typos possibles de recherche ?** Sélectionner les 2 types de recherche prédéfinis : dans le catalogue – dans les articles **Afficher le bouton d'aide ?** **Titre du cadre de recherche ?** **Afficher le lien de recherche avancée ?**	

Accéder aux résultats de recherche … et poursuivre sa route …	
Recherche dans le contenu éditorial	**Recherche dans le catalogue**
	
Clic sur un article → page qui affiche un article	On peut déplier une notice
http://.../ opac_css/index.php?lvl=cmspage&pageid=4&id_article=2	
Clic sur une vignette (carrousel) → page qui affiche une notice	On peut aussi découvrir d'autres voies à explorer …
http://.../opac_css/index.php?lvl=cmspage&pageid=7&id_notice=101 	

Vous pouvez effectuer une <u>recherche dans le contenu éditorial</u>, si la barre de recherche a été configurée en ce sens
La question « Construction du lien vers ? » n'est pas posée dans le formulaire de création de ce cadre.
C'est dans l'interface d'administration que vous sélectionnez la page du portail qui affiche le résultat d'une recherche.
C'est une page où on voit <u>la liste des articles qui répondent au critère de recherche.</u>

- Dans la démo, la page qui affiche le résultat d'une recherche dans les articles a le numéro d'id 11 (ce numéro ne change pas)
- **Il n'y aucune variable d'environnement liée à cette page**
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=11

- Si un ou plusieurs articles contiennent le(s) mot(s) recherché(s), on voit le titre et le résumé de cet article (ou de ces articles)
- Un clic sur un de ces articles mène vers la page qui affiche un article.
- Sinon, on voit le message « Pas de résultats » et on propose d'effectuer une recherche dans le catalogue.

Le mot recherché dans le contenu éditorial n'est pas le seul critère qui détermine l'affichage ou non de tel ou tel article !
Le statut de publication d'un article et ses dates de publication ont aussi un impact filtrant ! Voir configuration du contenu éditorial
Vous ne voyez donc pas tous les articles, seulement ceux dont le statut est « publié » et dont les dates de publication sont valides.

Vous pouvez effectuer une <u>recherche dans le contenu bibliographique</u>, si la barre de recherche a été configurée en ce sens
La question « Construction du lien vers ? » n'est pas posée dans le formulaire de création de ce cadre.
C'est dans l'interface d'administration que vous sélectionnez la page de l'OPAC qui affiche le résultat d'une recherche.
C'est une page où on voit <u>la liste des notices qui répondent au critère de recherche.</u>

- La page qui affiche le résultat d'une recherche dans les notices correspond forcément à une adresse de type OPAC
- http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=more_results

- Si une ou plusieurs notices contiennent le(s) mot(s) recherché(s), on voit la liste des notices, qu'on peut déplier si nécessaire.
- Un clic sur le permalink d'une notice mène vers la page de l'OPAC qui affiche cette notice dans son intégralité.
- Sinon, on propose de faire une suggestion d'achat, ou bien d'interroger des sources externes.

Menu basé sur le contenu éditorial

Il y a deux manières de créer un menu :

- Se baser sur une partie de l'arbre du contenu éditorial (certaines rubriques)
- Construire ce menu dans l'interface d'administration

Je traite ici la 1e méthode : un menu basé sur le contenu éditorial.

C'est la méthode employée dans la démo, et en outre, c'est le système le plus simple à mettre en œuvre au début.

Formulaire pour le cadre « Menu » - Module Menu (1)

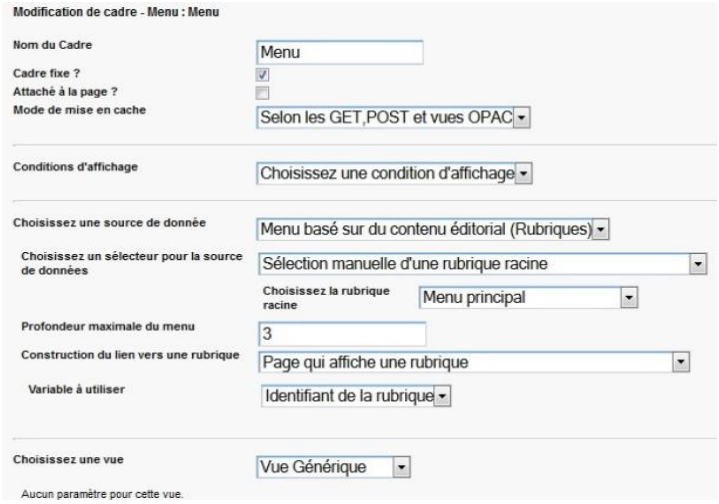
Nom précis donné au cadre ?
Fixe (toutes les pages) ou attaché à une page ?
Un cadre fixe !! Dans la zone Introduction
Mode de mise en cache ?
Conditions d'affichage ?

Source de données ?
Menu basé sur du contenu éditorial (Rubriques)
Sélecteur pour la source de données ?
Sélection manuelle d'une rubrique racine
La rubrique racine sera le « Menu principal ».

Profondeur maximale du menu ? 3

Construction du lien vers une rubrique
= où aller ensuite ? vers quelle page ?
Page qui affiche une rubrique
Que faire dans cette page ? Comment le signaler ?
= quelle variable transmettre ?
L'identifiant de la rubrique est ici la seule option.

Quelle vue ? Vue générique



Formulaire pour le cadre « Menu » - Module Menu (2)
N.B. Je reprends les questions les plus importantes, et je commente les réponses possibles

Source de données ?
Menu basé sur du **contenu éditorial** (Rubriques)
Créer un menu dans l'interface d'administration offre plus de souplesse, mais est plus complexe.

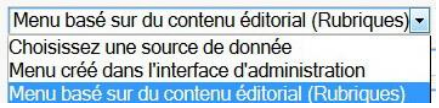
Sélecteur pour la source de données ?
Sélection manuelle d'une rubrique racine
Quelle rubrique racine ?
Menu principal → rubriques Animations, Sélections et Infos pratiques

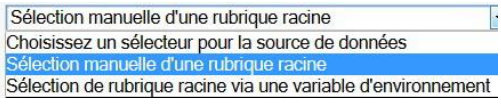
Une « rubrique racine » est le parent d'au moins une autre rubrique, et éventuellement de sous-rubriques → d'où le terme « rubrique racine »
En outre, elle peut aussi être éventuellement l'enfant d'une autre rubrique → c'est une question de point de vue, c'est relatif !

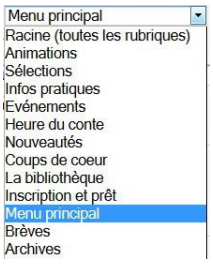
La « racine de l'arbre du contenu éditorial » est la seule rubrique qui n'a pas de parent, mais qui est le parent de toutes les autres
→ c'est une vérité générale.

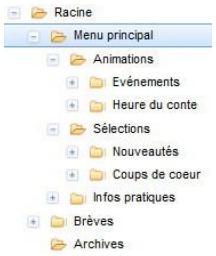
Attention à la gestion de l'arbre du contenu éditorial !

La position que les rubriques occupent par rapport à leur rubrique racine (ici, le « Menu principal »), et **la position que les articles occupent dans une rubrique** déterminent **l'ordre d'affichage** dans le menu et dans la page qui affiche le contenu d'une rubrique. C'est pourquoi j'avais expliqué en détail comment déplacer si nécessaire les rubriques et articles.









Formulaire pour le cadre « Menu » - Module Menu (3)
N.B. Je reprends les questions les plus importantes, et je commente les réponses possibles

Profondeur maximale du menu ?
2 niveaux suffisent en théorie pour la démo.
On peut prévoir un 3e niveau « en réserve » (si on ajoute un niveau de sous-rubriques).

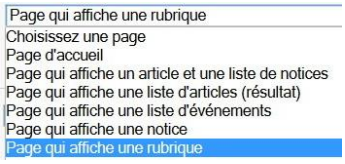
Exemple : j'ai créé 3 sous-rubriques qui dépendent de « Evénements » :
Pour les adultes – Pour les enfants – Pour tous

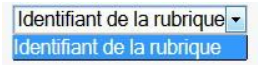
Construction du lien vers une rubrique = où aller ensuite ? vers quelle page ?
Page qui affiche une rubrique

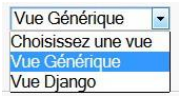
Vu qu'on a choisi le menu basé sur le contenu éditorial (rubriques), **c'est la seule page de destination possible !**

Que faire dans cette page ? Comment le signaler ? = quelle variable transmettre ?
L'identifiant de la rubrique est ici la seule option.

Quelle vue ?
Vue générique
Il est plus facile d'adapter la profondeur maximale du menu si on a choisi la vue générique (il suffit de changer la valeur). Dans le cas de la vue Django, il faut adapter le template.







A partir du menu principal, accéder aux rubriques, ensuite aux articles … et poursuivre sa route …

Une rubrique qui n'est le parent d'aucune autre rubrique

Page → une rubrique (pageid = 6)

Rubrique Evénements
Id_rubrique = 4

http://...&pageid=6&id_rubrique=4

Une rubrique qui est le parent de 2 autres rubriques

Page → une rubrique (pageid = 6)

Rubrique Animations
Id_rubrique = 1

http://...&pageid=6&id_rubrique=1

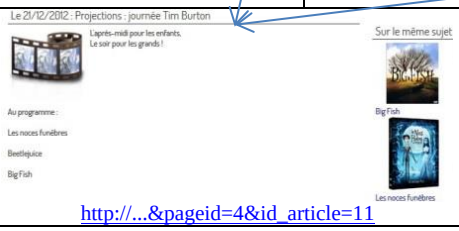
Page → une rubrique (pageid = 6)

Rubrique Evénements
Id_rubrique = 4

http://...&pageid=6&id_rubrique=4

Page → un article (pageid = 4)

Article Tim Burton
id_article = 11



http://...&pageid=4&id_article=11

Le carrousel de notices sur le même sujet que l'article vous ouvre d'autres perspectives.

Vous pouvez donc poursuivre l'exploration du catalogue …

Configuration du contenu éditorial – **Tenez compte de la visibilité des rubriques et articles → statut et dates de publication.**
Ex. Si une rubrique n'apparaît pas dans le menu, vérifiez son statut et ses dates de publication : ils pourraient expliquer son invisibilité !

Ce n'est pas parce qu'une rubrique ou un article ont le statut « Publié – Visible par tous » qu'on les voit automatiquement !
Or, vous n'allez sûrement pas créer un nombre anormalement élevé de carrousels ou de listes, afin d'être sûr que les rubriques et articles soient tous accessibles d'une manière ou d'une autre.
En outre, il faut faciliter la navigation du lecteur, lui fournir un menu bien structuré, dont les titres et sous-titres sont explicites.

Un menu est un système efficace de liens logiques vers les rubriques et articles, un GPS pour orienter le lecteur

- Insérer un menu dans la zone Introduction - il s'agit ici d'un menu basé sur le contenu éditorial (les rubriques)
- Organiser (à partir d'une rubrique racine visible) une arborescence cohérente de rubriques visibles
- Mettre les articles visibles dans des rubriques parentes visibles qui dépendent de la rubrique racine qu'on a choisie

C'est ainsi que le lecteur pourra naviguer dans votre portail sans risquer de se perdre, et en ayant le maximum de chances de trouver ce qu'il était venu chercher … et de découvrir en même temps d'autres aspects du catalogue ou les activités que vous organisez régulièrement;-)

Analyser la page d'accueil du portail

En introduisant un des chapitres précédents, j'avais dit qu'il était inutile de rêver à l'étage supérieur (le portail), aussi longtemps qu'on n'avait pas érigé des fondations solides et un rez-de-chaussée dont les murs sont porteurs : les pages-outils.

Mission accomplie : vous avez construit les « murs porteurs ».
Chaque page-outil contient désormais un cadre issu du module fondamental qui lui est associé.
Vous pouvez enfin passer à la dernière étape : insérer des cadres dans la page d'accueil ou ailleurs ...

Liste de quelques modules et cadres du portail visibles dans la page d'accueil – dans les zones introduction et page centrale

En noir (police standard) : le nom du cadre	***En rouge et en italiques : le nom du module***

Menu
Menu

Carrousel d'articles (accueil)
Liste d'articles

Liste d'articles - sélections (accueil)
Liste d'articles

Liste de brèves (accueil)
Liste d'articles

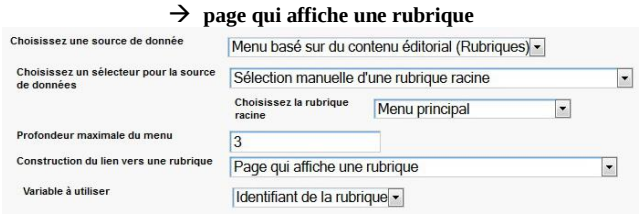
Barre de recherche simple
Recherche

Agenda (accueil)
Agenda

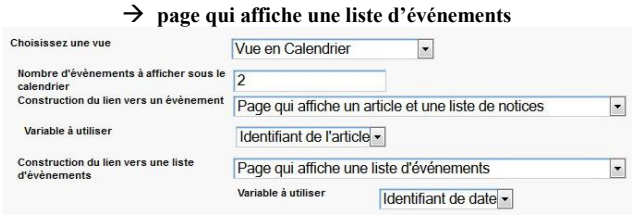
Carrousel de notices (nouveautés - accueil)
Liste de notices

Je n'ai pas l'intention d'analyser à nouveau ces cadres, mais je voudrais insister sur la **chronologie des actions**.
Pour tous ces cadres, vous déterminez où il faut aller ensuite, en répondant à la question « Construction du lien vers ... ? »
Cette question est posée en général au moment de créer un cadre, sauf pour la barre de recherche simple.
Mais si la page de destination n'existe pas encore, vous ne pouvez forcément pas la sélectionner dans la liste déroulante ...
Par conséquent, il faut créer et configurer les 5 pages-outils indispensables <u>avant</u> d'insérer ces cadres dans une page !
Dans le cas contraire, vous devriez éditer à nouveau ces cadres plus tard, afin de configurer le « lien vers » !

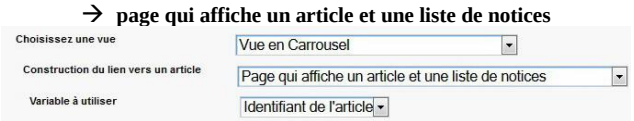
Menu basé sur le contenu éditorial
→ page qui affiche une rubrique



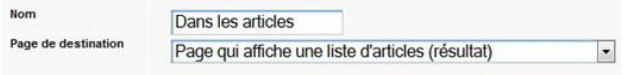
Agenda
→ page qui affiche un article et une liste de notices
→ page qui affiche une liste d'événements



Carrousel d'articles d'actualité
→ page qui affiche un article et une liste de notices

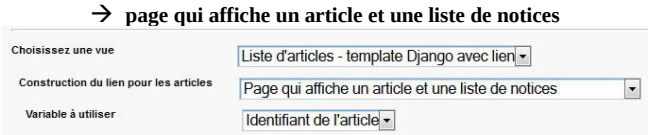


Barre de recherche simple
→ page qui affiche une liste d'articles (résultat d'une recherche)

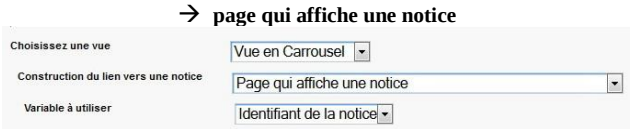


Page de destination : question posée dans l'interface d'administration

Liste d'articles – Sélections
→ page qui affiche un article et une liste de notices



Carrousel de notices – Nouveautés
→ page qui affiche une notice



En outre, ces cadres contiennent un template Django ou un template de notices afin de gérer la navigation, la « suite du parcours ».
<u>Mais</u> si ce template ne crée pas d'URL de destination (sur base de la réponse à la question « où aller ensuite ? »), si ce template ne spécifie pas ce qui est cliquable (texte et/ou image), on n'a aucune possibilité concrète de changer de page !

C'est d'ailleurs le cas pour les Brèves. <u>En théorie</u>, on pourrait aller vers la page qui affiche un article et une liste de notices.
En effet, le formulaire de ce cadre contient la même réponse à la question « Lien vers ? » que la Liste d'articles – Sélections.
Mais le template Django de ce cadre ne crée pas d'URL de destination ni d'élément cliquable, donc on ne peut pas aller ailleurs !

Il est techniquement impossible de paramétrer correctement ces cadres si on n'a pas d'abord fait une série d'actions :

- **Configurer le contenu éditorial : créer divers types d'articles et des champs personnalisés pour transmettre des valeurs**
  - Ex. Sans le champ personnalisé etagere, impossible de lier un article de type Sélection à une étagère !
- **Créer la page-outil vers laquelle on ira ensuite**
  - Si la page-outil dont on a besoin n'existe pas, elle n'apparaît forcément pas dans la liste déroulante, parmi les réponses possibles à la question « construction du lien vers … ? », donc on ne peut pas la sélectionner !
- **Lier cette page-outil à une variable d'environnement – Du moins si elle est censée avoir une variable d'environnement !**
  - Si la variable d'environnement (id_rubrique, id_article, id_notice ou id_date) n'existe pas, on ne peut pas transmettre à la page suivante le numéro d'id de la rubrique, de l'article … à afficher
  - Il n'y a pas de variable d'environnement pour la page qui affiche le résultat d'une recherche dans le contenu éditorial
- **Insérer dans chaque page-outil un cadre issu du module fondamental lié à cette page**
  - Ex. si la page qui est censée afficher un article ne contient pas un cadre issu du module « Article », PMB peut aller vers cette page (car elle existe), mais PMB ne peut rien faire, rien afficher, faute de cadre adéquat !

En résumé

Si les pages-outils n'existaient pas, si elles ne contenaient pas le cadre fondamental lié à leur fonction, s'il n'y avait pas des variables d'environnement et des champs personnalisés pour transmettre des paramètres, si les templates Django ne créaient pas des URL de destination et ne spécifiaient pas ce qui est cliquable, il serait inutile d'insérer des cadres en page d'accueil, soit on n'irait nulle part ailleurs, soit on irait quelque part mais on ne pourrait rien y faire !

C'est pourquoi je vous ai présenté le contenu éditorial, les modules, les cadres et les pages-outils <u>avant</u> de voir les choses sous un autre angle : à partir de la page d'accueil. Il faut respecter la chronologie des actions à exécuter.

Interconnexion entre ces cadres visibles en page d'accueil et les pages-outils du portail

Rappel
Il n'y a pas de variable d'environnement liée à la page qui affiche une liste d'articles (le résultat d'une recherche dans le contenu éditorial).

	A	B	C	D
1	Id	Nom de la page-outil	Variable d'environnement	Libellé
2	4	Page qui affiche un article et une liste de notices	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	id_date	Identifiant de date
6	11	Page qui affiche une liste d'articles (recherche)		

On ne va jamais « spontanément » vers une page, quelle qu'elle soit
C'est dans chaque cadre qu'on construit le lien vers « la suite du parcours ».

Cadres →	Variable d'environnement	→ Pages de destination	
Menu (Le menu de la démo n'affiche que des titres de rubriques) Clic sur un titre de rubrique	**id_rubrique**	Page qui affiche une **rubrique** → Si on clique sur une sous-rubrique → page qui affiche une rubrique → Si on clique sur un article → page qui affiche un article	
Carrousel d'articles (accueil) Clic sur un article du carrousel **Liste d'articles - Sélections (étagères)** Clic sur un article de la liste	**id_article**	Page qui affiche un **article** L'aspect varie en fonction du **type de contenu de l'article** (ce paramètre est transmis par un **champ personnalisé**)	Coup de cœur : **identifiant_notice** Sélection : **etagere** Evénement (date précise) : **date** Evénement (période) : **date_debut, date_fin** Information : pas de champ personnalisé
Liste de brèves – pas de clic possible	Pas de lien	Aucune page de destination prévue dans le template de ce cadre	
Barre de recherche dans les articles Si on effectue une recherche	Pas de variable	Page qui affiche une **liste d'articles : le résultat d'une recherche** → Si on clique sur un article de la liste → page qui affiche un article	
Agenda (calendrier : choix d'une date) S'il y **a un seul** événement	**id_article**	Page qui affiche un **article**	
S'il y a **plusieurs** événements	**id_date**	Page qui affiche **une liste d'événements (module agenda)** → Si on clique sur un événement de la liste → page qui affiche un article	
Carrousel de notices (nouveautés) Clic sur une notice du carrousel	**id_notice**	Page du portail qui affiche une **notice** → Si on clique sur le permalink → page de l'OPAC qui affiche une notice	

Ces variables se ressemblent : leurs noms sont similaires, et les questions de départ identiques : quelle notice ? <u>ou</u> quelle date ?
Mais elles n'ont pas la même fonction, ne les confondez pas !

id_notice et **id_date** sont 2 **variables d'environnement** → **page** qui affiche **une notice <u>ou</u> page** qui affiche **une liste d'événements**
identifiant_notice et **date** sont 2 **champs personnalisés** → **types de contenu des articles : Coup de cœur <u>ou</u> Evénement à une date**

Templates Django – Synthèse

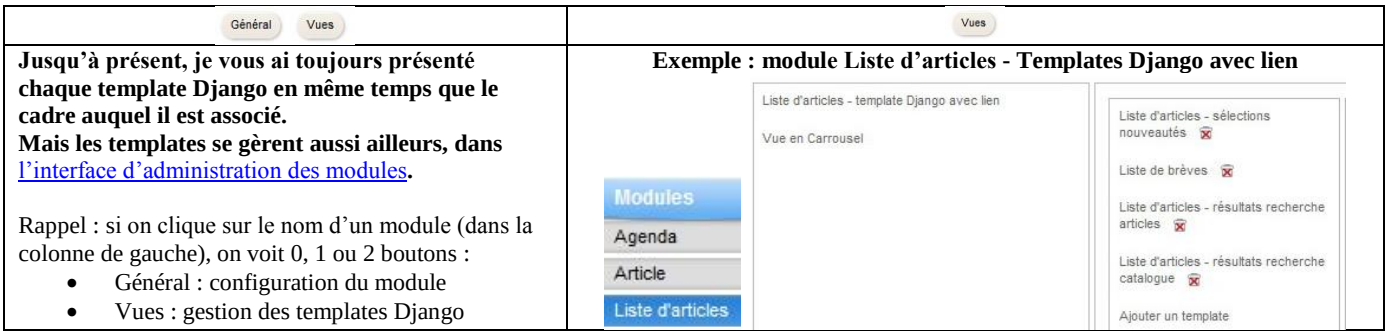
Dans ce chapitre consacré aux modules et cadres, j'ai parlé à de nombreuses reprises des templates Django. J'ai essayé d'y aller « à petits pas », de passer progressivement du plus simple au plus complexe. Le moment est venu d'en faire une synthèse.

Un template Django est un script créé à l'aide d'un langage de programmation appelé Django H2O.

Ce template remplit de multiples fonctions :

- **Déterminer quelles données il faut afficher**
  - tout l'article ou seulement le titre et le chapeau (le champ résumé) ?
  - toute la notice ou seulement certains champs ?
  - on dispose en effet de variables pour identifier les données bibliographiques ou éditoriales
- **Gérer la mise en forme de ces données**
  - Spécifier la police de caractères, la taille des images … grâce aux balises HTML
- **Définir une structure générale, un algorithme**
  - Créer des boucles pour gérer les répétitions
    - Ex. afficher toutes les notices d'une étagère ou afficher tous les articles d'une rubrique
  - Formuler des conditions
    - Ex. Si on a trouvé des articles qui répondent au critère de recherche, on les affiche
      Sinon, on voit le message « Aucun résultat »
- **Gérer la navigation entre les pages, indiquer vers quelle page on ira ensuite**
  - Générer une balise <a href= …>
    - l'URL de destination est créée sur base de la réponse à la question « Construction du vers … ? »
    - le script spécifie aussi quels éléments sont cliquables (ex. titre, logo)

GESTION DES TEMPLATES DANS L'INTERFACE D'ADMINISTRATION



Jusqu'à présent, je vous ai toujours présenté chaque template Django en même temps que le cadre auquel il est associé.
Mais les templates se gèrent aussi ailleurs, dans l'interface d'administration des modules**.**

Rappel : si on clique sur le nom d'un module (dans la colonne de gauche), on voit 0, 1 ou 2 boutons :

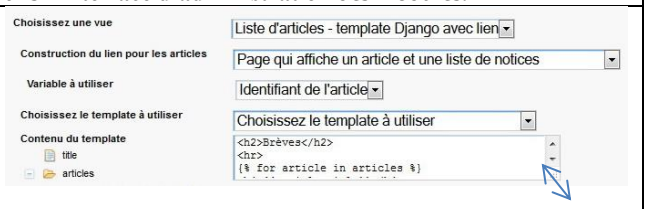
- Général : configuration du module
- Vues : gestion des templates Django

Exemple : module Liste d'articles - Templates Django avec lien

Comment lire aisément un template Django dans son intégralité
Cette astuce fonctionne aussi bien pour les cadres que dans l'interface d'administration des modules.

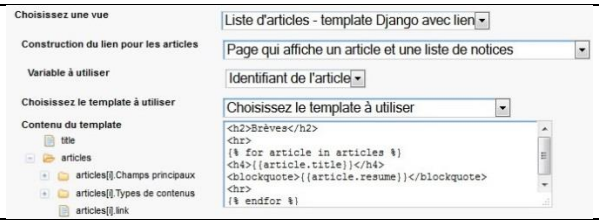
Au départ, le cadre du template est « minuscule » !!!

En effet, on ne voit en général que 3 lignes du template dans la petite fenêtre de lecture, ce qui ne permet vraiment pas d'avoir une vue d'ensemble.



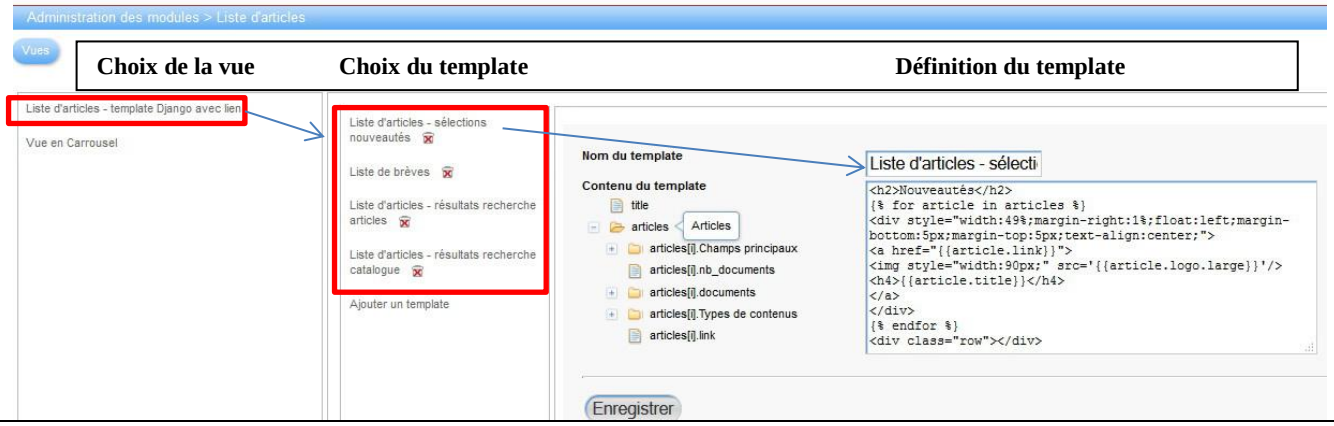
Si vous amenez le curseur de la souris dans le coin inférieur droit, vous verrez une flèche bidirectionnelle, qui vous indique qu'il est possible d'agrandir ce cadre, afin de faciliter la lecture.

N.B. Le template que j'ai choisi comme exemple est fort court
7 lignes au total – c'est d'ailleurs celui de la Liste de brèves ;-)

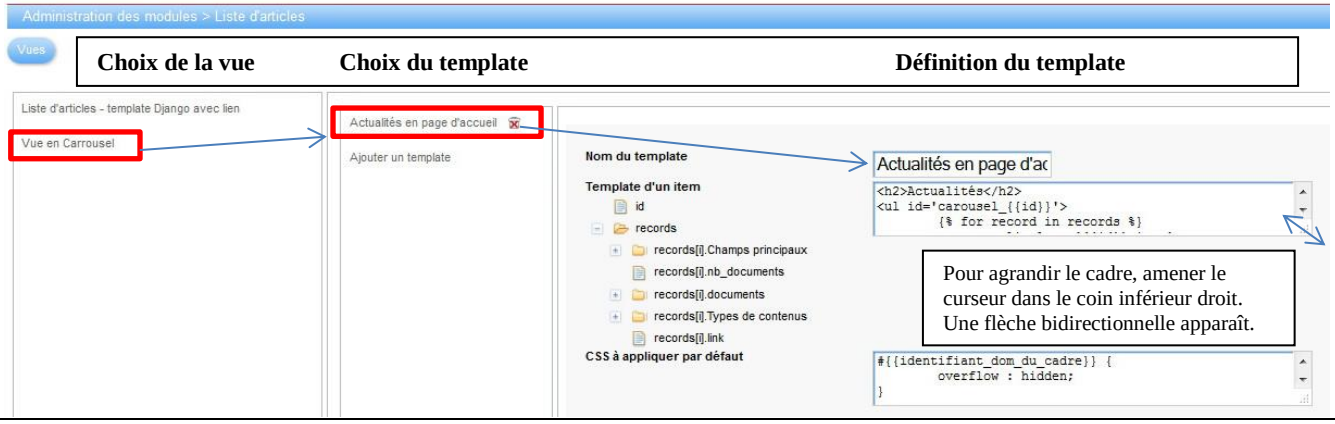


Interface d'administration - Vues et templates

Exemple : module Liste d'articles – Vue : template Django avec lien - On a enregistré 4 modèles de template



Exemple : module Liste d'articles – Vue : carrousel - On a enregistré 1 modèle de template



La « vue » a un sens plus global que le template : c'est une manière générale d'afficher les données.
Pour chacune de ces vues, on peut prévoir un ou plusieurs templates, afin de diversifier la présentation.

DÉMO DU PORTAIL - LISTE DES MODULES, DES TEMPLATES DJANGO ET DES CADRES OÙ ILS SONT EMPLOYÉS		
Modules	**Templates Django**	**Cadres où ces templates sont employés**
3 modules de base : un article - une notice - une rubrique		
Article	Affichage d'un article	→ Cadre principal de la page qui affiche un article
Notice	Affichage d'une notice	→ Cadre principal de la page qui affiche une notice
Rubrique	Affichage d'une rubrique	→ Cadre principal de la page qui affiche une rubrique
Vues	**Templates Django**	**Cadres où ces templates sont employés**
Module liste d'articles		
Templates Django avec lien	Liste d'articles – Sélections – Nouveautés	Sélection de nouveautés (symbolisées par les étagères)
	Liste de brèves	Liste de brèves
	Liste d'articles – Résultat d'une recherche dans les articles	→ Cadre principal de la page qui affiche le résultat d'une recherche dans le contenu éditorial
	Liste d'articles – Recherche dans le catalogue	Articles dans les mêmes catégories qu'une notice
Carrousel	Actualités en page d'accueil	Carrousel d'articles d'actualité
Module liste de notices		
Vue Django	Liste de notices	Notices d'une étagère associée à un article
Vue en carrousel	Notices du même auteur	Notices du même auteur qu'une notice
	Notices similaires	N.B. Ces 2 templates ne sont pas illustrés dans la démo.
	Notices sur la même étagère (cote de rangement proche)	
	Nouveautés en page d'accueil	Notices – Carrousel de nouveautés
	Notices dans les mêmes catégories qu'un article	Notices dans les mêmes catégories qu'un article (même sujet)
Module agenda		
Liste d'événements	Affichage d'une liste d'événements à partir de l'agenda	→ Cadre principal de la page qui affiche une liste d'événements
Conclusion : à chaque cadre illustré dans la démo correspond au moins un template sauvegardé dans l'interface d'administration **Comme vous pouvez le constater, tous les cadres et templates illustrés dans la démo ont été commentés dans ce manuel !**		

SYSTÈME DE CLASSEMENT DES TEMPLATES DANS L'INTERFACE D'ADMINISTRATION

En guise d'exemples, je reprends les deux modules qui contiennent le plus grand nombre de templates.

Liste de notices – Vue : template Django

Liste de notices – Vue : carrousel

Attention ! Ces carrousels ont été créés avec le module Liste de notices, c'est pourquoi les templates correspondants figurent sous le titre « Liste de notices », et pas sous le titre « Carrousel » !

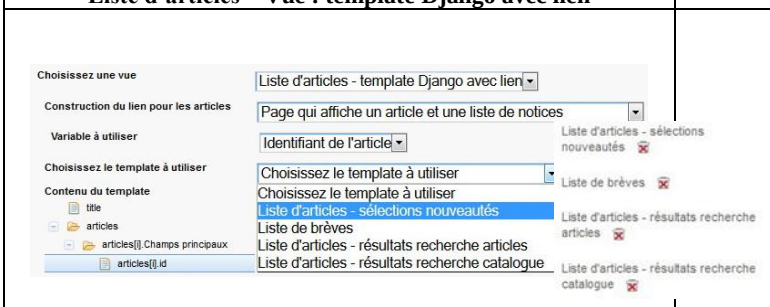
Liste d'articles – Vue : template Django avec lien

Liste d'articles – Vue : carrousel

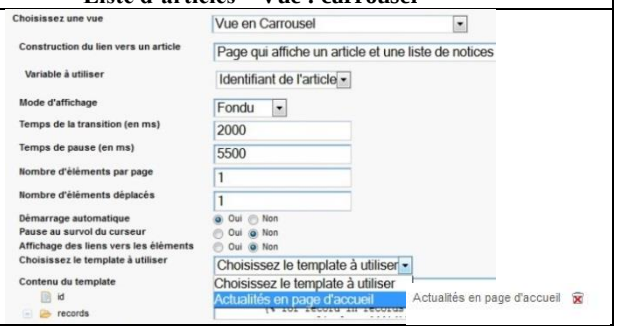
Conclusion : les templates sont classés sur base du module qui a servi à générer un cadre, et ensuite par vue.

Ce système de classement facilite le choix d'un template adéquat, lorsqu'on paramètre un cadre.
On ne voit que la liste des templates réellement utilisables, en fonction du module concerné et de la vue qu'on a choisie.

Liste d'articles – Vue : template Django avec lien



Liste d'articles – Vue : carrousel



Vous pouvez ajouter des templates à ceux qui ont déjà été enregistrés, mais veillez à respecter ce mode de classement.
Sauvegardez ces nouveaux templates « avec le bon module », sinon ils ne vous seront pas proposés « au bon moment » !!

A ce propos, je vous rappelle quelques indications du chapitre « Distinguer les modules et les cadres ».

Comment savoir avec certitude quel module est à l'origine de tel ou tel ou cadre ?
Cliquez sur l'icône du crayon pour éditer le cadre.

En premier lieu, le nom du module qui a servi à créer le cadre

Ensuite, le nom donné au cadre

Si un carrousel de notices a été créé sur base du module Liste de notices, le template correspondant est sauvegardé dans Liste de notices > Vue en carrousel !

Pour créer un nouveau template, sélectionnez le bon module et la bonne vue, et cliquez ensuite sur *ajouter un template*. Vous pourrez **adapter si nécessaire le template par défaut** qui vous sera proposé, **ou le remplacer par le template d'un cadre spécifique** (il suffit de copier-coller).

Pour modifier un template préexistant, cliquez sur son nom et adaptez le code.

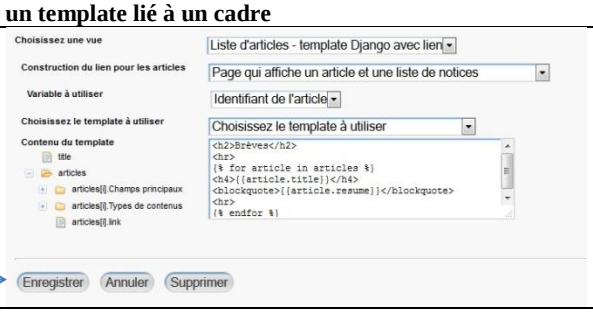
Dans un cas comme dans l'autre, n'oubliez pas d'enregistrer !

GESTION DES TEMPLATES DANS LES CADRES

Rappel : comment sauvegarder un template lié à un cadre

A ce point de vue-là, il n'y a aucune différence entre un cadre qu'on vient de créer et qu'on enregistre pour la 1e fois, et un cadre préexistant qu'on édite afin de compléter ou de corriger le paramétrage initial.

La méthode de sauvegarde est identique et toute simple : **lorsque vous enregistrez le contenu d'un cadre, vous enregistrez en même temps le template qui lui est associé !**

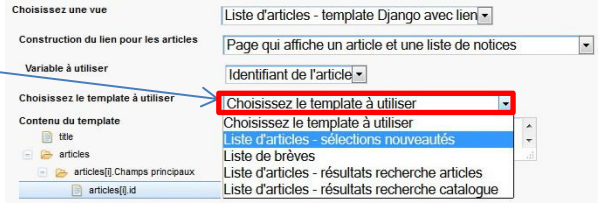


Choix d'un template précis ou recours au template par défaut ?

Quand *peut-on* choisir le template à utiliser ?

Jusqu'à présent, vous avez vu des captures d'écran comme celle-ci, où vous pouviez choisir le template à utiliser pour tel ou tel cadre. Mais ce n'est pas une règle générale.

Ce choix éventuel d'uu template n'est possible que s'il y a au moins un template adéquat qui a été enregistré dans l'interface d'administration, donc un **template qui est relié au module concerné et aussi à la vue choisie**.



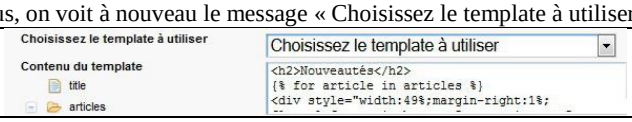
Il faut respecter le système de classement dans l'interface d'administration. (voir page précédente)
Sinon, vous ne disposerez pas du « bon » template au moment où vous en aurez besoin.

Si vous parcourez à nouveau la liste des templates enregistrés dans l'interface d'administration, vous comprendrez aisément pourquoi vous avez toujours vu jusqu'à présent ce message « Choisissez le template à utiliser ». Je rappelle la conclusion qui figure à la fin de cette liste.
A chaque cadre illustré dans la démo correspond au moins un template sauvegardé dans l'interface d'administration

Ne soyez pas surpris par la modification du message – Cela reflète un changement important

Quand on choisit pour la 1e fois un template ou dès qu'on change le template d'un cadre préexistant, le nom du nouveau template est visible aussi longemps qu'on n'a pas sauvegardé le cadre.

Les choses changent dès qu'on a sauvegardé le cadre. Si vous éditez à nouveau ce cadre, le nom du template que vous aviez choisi n'apparaît plus, on voit à nouveau le message « Choisissez le template à utiliser »

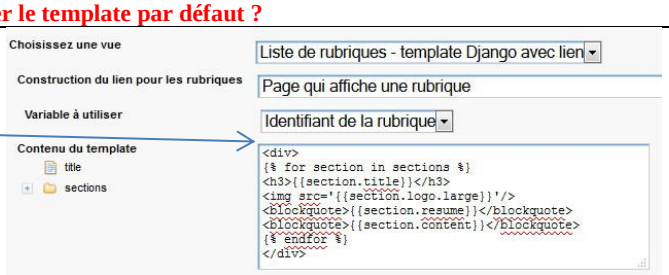


Le message a changé, le nom du modèle choisi disparaît dès que vous sauvegardez le template en même temps que le cadre.
Vous avez sélectionné un template dans la liste de ceux qui sont enregistrés dans l'interface d'administration.
Soit vous avez employé ce template tel quel dans le cadre, soit vous l'avez modifié, mais peu importe, finalement.
Ce modèle de template a perdu son statut de modèle général et réutilisable, il est devenu le template d'un cadre précis.

Quand *doit-on* employer le template par défaut ?

Si aucun template dans l'interface d'administration ne correspond au module concerné et à la vue choisie, une seule possibilité : le template par défaut ! On ne voit dès lors aucun message suggérant un choix éventuel.

Ex. Module Liste de rubriques (non illustré dans la démo).
2 vues possibles : Template Dajngo avec lien ou carrousel
Mais aucun template adéquat n'est sauvegardé dans l'interface d'administration, donc pas de choix de template possible.



Conclusion : même si aucun template ne semble exister (en administration) pour un module et une vue spécifiques, PMB ne vous abandonne pas à votre triste sort ! Il vous propose de toute façon un template par défaut ;-)

Même si vous sauvegardez les changements apportés à un template dans un cadre ou dans l'interface d'administration, il n'y a aucune répercussion automatique ni dans un sens, ni dans l'autre.
La modification d'un modèle de template dans l'interface d'administration n'a aucun impact sur les templates des cadres préexistants
La modification d'un template dans un cadre n'a aucun impact sur les modèles de template dans l'interface d'administration
Il faut donc faire des copier-coller si on veut mettre les templates à jour des deux côtés.

FAUT-IL CRÉER OU MODIFIER DES TEMPLATES DANS L'INTERFACE D'ADMINISTRATION ?

Gérer les templates dans l'interface d'administration permet de créer des « modèles de templates », à réutiliser, à améliorer et à partager, donc cela facilite vraiment l'existence !

- **Les templates par défaut sont fort brefs, quelque peu « rudimentaires »**
  - Cela vaut vraiment la peine de récupérer et de sauvegarder dans votre portail les templates disponibles dans la démo, car ils sont beaucoup plus élaborés, et vous montrent des exemples concrets de ce qu'on peut faire.
  - **Ces templates sont enregistrés dans la table cms_managed_modules.**
- **Cela vous donne aisément accès à de nouveaux modèles de templates**
  - Si vous avez modifié un des modèles de templates lorsque vous étiez en train d'éditer un cadre, cette nouvelle version du modèle n'est disponible que si vous éditez le cadre concerné.
  - Or, si vous voulez créer un nouveau cadre en partie similaire, il est plus pratique de sélectionner directement ce nouveau template, ce qui implique de l'avoir sauvegardé préalablement dans l'interface d'administration.
  - Suivant le cas, vous pouvez soit conserver le modèle préexistant et sauvegarder la nouvelle version sous un nom différent, soit faire les changements dans le modèle préexistant.
- **Cela facilite la collaboration**
  - Si vous ne vous sentez pas capable de modifier ou de créer vous-même un template Django**,** vous pouvez demander à quelqu'un d'autre de vous aider, intégrer ensuite ce nouveau template dans l'interface d'administration et l'employer ultérieurement dans un cadre.
  - Même si personnellement vous maîtrisez les templates Django, il est toujours plus fructueux de travailler à plusieurs, de partager et d'échanger vos « productions » !
- **C'est utile si vous travaillez en multi-localisations**
  - Chaque bibliothèque sera amenée à publier ses propres articles, mais risque du coup de développer son propre mode de présentation.
  - Pour maintenir une cohérence dans l'affichage, pour éviter de rompre l'harmonie graphique de l'ensemble, il vaut mieux prévoir des templates identiques, au moins pour les cadres fondamentaux du portail.

LES TEMPLATES DE LA DÉMO SONT-ILS RÉUTILISABLES PARTOUT ?

Globalement, oui.

- **La plupart des variables utilisées dans les templates ont un lien avec les champs de la base de données ou avec le contenu éditorial** (ex. titre / résumé d'un article ou d'une rubrique)**, et elles ont donc une portée générale.**
- **Toutefois, certaines variables désignent des champs personnalisés qu'on a pu créer.**
  - Ex. Date de début d'un événement sur une période {{fields_type.date_debut.values.0.format_value}}
  - Si vous avez donné d'autres noms à ces champs personnalisés, vous devrez faire quelques changements dans certains templates
  - Si vous avez restauré le contenu éditorial minimum, ou si vos champs personnalisés sont identiques à ceux de la démo (mêmes noms, types de données et types de saisie), les templates sont prêts à l'emploi.

N.B. Je rappelle que vous pouvez restaurer dans votre base la configuration éditoriale minimum de la démo, ainsi que la table cms_managed_modules, qui contient les templates Django.

Voir Sauvegarder - restaurer la configuration minimum du portail

Les tableaux de synthèse des pages suivantes vous donnent des exemples de variables utilisables dans les templates Django, ainsi que la liste des instructions et fonctions disponibles.

LISTE DES INSTRUCTIONS ET FONCTIONS DES TEMPLATES DJANGO

Les instructions entre {% %}

Les boucles de répétition **Exemple** **{% for article in articles %} …** {{article.title}} … {{article.resume}} … {{article.content}} … **{% endfor %}**		
Liste d'articles (template Django avec lien)	**{% for article in articles %}**	Article du contenu éditorial
Liste de notices (vue Django ou carrousel) Carrousel de notices Carrousel d'articles Carrousel de rubriques	**{% for record in records %}** Pour toutes les listes de notices Pour tous les carrousels	A record = enregistrement Suivant le cas, record = une notice, un article ou une rubrique
Liste de rubriques (template Django avec lien)	**{% for section in sections %}**	Section = rubrique du contenu éditorial
Liste des sous-rubriques d'une rubrique parente	**{% for child in children %}**	Dans ce contexte, a child = une sous-rubrique
Agenda (vue d'une liste d'événements)	**{% for event in events %}**	An event = un événement
Liste d'étagères	**{% for shelve in shelves %}**	A shelf (pluriel shelves) = une étagère
Liste de bannettes	**{% for bannette in bannettes %}**	Les bannettes de la DSI

Les conditions : 2 exemples concrets		
Contexte : un article est lié à une étagère. On affiche l'article, et en-dessous, la liste des notices contenues dans cette étagère N.B. J'ai abrégé le contenu du template, j'ai gardé les instructions essentielles pour comprendre le sens de la condition.	**{% if records %}** Toutes les nouveautés **{% endif %}** {% for record in records %} {{record.content}} {% endfor %}	S'il y a des enregistrements (= notices) à afficher On affiche le titre « Toutes les nouveautés » Fin de la condition Boucle qui va de la 1e à la dernière notice On affiche le contenu de la notice Fin de la boucle
La condition {% if records %} {% endif %} est nécessaire dans cet exemple Inutile d'afficher le titre « Toutes les nouveautés » s'il n'y a pas d'étagère associée à l'article ou si elle est vide !		
Contexte : page → résultat de recherche On cherche un mot dans les articles. Si on le trouve, on affiche la liste des articles concernés. N.B. J'ai abrégé le contenu du template, j'ai gardé les instructions essentielles pour comprendre le sens de la condition.	**{% if articles %}** {% for article in articles %} …. {% endfor %} **{% else %}** Pas de résultats **{% endif %}**	S'il y a des articles à afficher Boucle qui va du 1er au dernier article On affiche la version abrégée (titre + résumé) Fin de la boucle Sinon On affiche le message « Pas de résultats » Fin de la condition
La condition {% if articles %} {% else %} {% endif %} est nécessaire dans cet exemple On a une boucle qui va du 1er au dernier article seulement si on a trouvé le mot quelque part dans le contenu éditorial ! Et en toute logique, le message « Pas de résultats » n'a de sens que dans le cas d'une recherche infructueuse !		

Les conditions : synthèse schématique		
Condition « simple » : si … alors … **On n'envisage qu'un seul cas :** **si la condition est vérifiée** **Il n'y a rien à faire dans le cas contraire** **{% if true %} Si la condition est vérifiée** Instruction à exécuter **{% endif %} Fin de la condition**	**Condition « si … alors … sinon … »** **On envisage les 2 termes de l'alternative :** **si la condition est vérifiée** **si elle ne l'est pas** **{% if true %} Si la condition est vérifiée** Instruction n° 1 **{% else %} Sinon** Instruction n° 2 **{% endif %} Fin de la condition**	**On peut imbriquer 2 conditions l'une dans l'autre** **{% if true %} Si la condition n° 1 est vérifiée** Instruction n° 1 **{% if true %} Si la condition n° 2 est vérifiée** Instruction n° 2 **{% else %} Sinon** Instruction n° 3 **{% endif %} Fin de la condition n° 2** **{% endif %} Fin de la condition n° 1**
On symbolise en général une condition Django de cette manière : (% if true %) → si c'est vrai, si la condition est vérifiée		
On peut effectuer les tests classiques de comparaison		
if {{variable1}} == {{variable2}} if {{variable1}} != {{variable2}}	if {{variable1}} < {{variable2}} if {{variable1}} <= {{variable2}}	if {{variable1}} > {{variable2}} if {{variable1}} >= {{variable2}}
Par convention == signifie "égal à" (double signe =)	**Par convention != signifie "différent de"**	
On peut employer les opérateurs booléens habituels : and or not if {{variable2}} >= {{variable1}} and {{variable2}} <= {{variable3}}		

Infos complémentaires : https://docs.djangoproject.com/en/1.8/ref/templates/builtins/

Une fonction supplémentaire : limitstring	
PMB a ajouté une fonction aux templates Django de base, la fonction limitstring	
{{record.title | limitstring,45,"..." }}	**Attention à la syntaxe ! Le tout doit être entre {{ }}** **Il faut une barre verticale | entre la variable et limitstring**
La fonction limitstring permet de limiter la longueur d'une chaîne de caractères. Exemple {{record.title | limitstring,45,"..." }} Cela signifie que la longueur du texte à afficher (le titre de la notice, record.title) sera limitée à 45 caractères, et que les caractères manquants seront remplacés par le texte entre guillemets (les 3 petits points). Le découpage se fait à l'aveugle, et le résultat final est plus ou moins heureux suivant le cas = suivant le texte à afficher ! Si vous avez une série de titres assez longs dans un carrousel de notices, cette fonction est fort utile. Bien entendu, son rôle n'est pas limité aux carrousels, elle peut aussi servir en d'autres circonstances.	

VARIABLES UTILISABLES DANS LES TEMPLATES DU PORTAIL

Les noms des variables entre {{ }}		
Les variables utilisables dans les templates ont un lien avec les champs de la base de données ou avec le contenu éditorial		
Le sens de certaines variables comme {{article}} ou {{section}} est évident. Mais … **Tous les carrousels (notices, articles, rubriques) emploient la boucle {% for record in records %}** **Toutes les listes de notices (vue Django ou carrousel) emploie la boucle {% for record in records %}** **La variable {{record}} a donc plusieurs significations : elle peut désigner une notice, un article ou une rubrique.**		
{{title}}	{{article.title}}	Le titre de l'article
	{{section.title}}	Le titre de la rubrique
	{{record.title}}	Le titre de l'enregistrement Suivant le cas, titre de la notice, de l'article ou de la rubrique
	{{event.title}}	Le titre de l'événement (Le terme *événement* a un sens global dans ce contexte : article qui décrit une activité à une date ou sur une période)
{{resume}}	{{article.resume}}	Le « résumé » de l'article = le chapeau de l'article
	{{section.resume}}	Le « résumé » de la rubrique
	{{record.resume}}	Le « résumé » de l'enregistrement Suivant le cas, résumé de la notice, chapeau de l'article ou résumé de la rubrique
	{{event.resume}}	Le « résumé » de l'événement = le chapeau de l'article
{{content}}	{{article.content}}	Le contenu de l'article = le texte même de l'article
	{{record.content}}	Le contenu de l'enregistrement - Suivant le cas • le contenu de la notice = tous les champs de la notice • le contenu de l'article = le texte même de l'article N.B. Rappel : une rubrique n'a pas de champ « contenu »
	{{event.content}}	Le contenu de l'événement = le texte même de l'article
{{link}}	{{article.link}}	Le lien = URL de l'article
	{{section.link}}	Le lien = URL de la rubrique
	{{record.link}}	Le lien = URL de l'enregistrement Suivant le cas, URL de la notice, de l'article ou de la rubrique
	{{event.link}}	Le lien = URL de l'événement
{{vign}}	{{article.vign}}	La vignette de l'article
	{{section.vign}}	La vignette de la rubrique
	{{record.vign}}	La vignette de l'enregistrement Suivant le cas, la vignette de la notice, de l'article ou de la rubrique
	{{event.vign}}	La vignette de l'événement
Exemples de champs personnalisés, qui correspondent aux types de contenu définis pour les articles **N.B. Les noms des variables et fonctions associées à un champ personnalisé contiennent toujours ceci : {{fields_type}}**		
{{fields_type.date.values.0.format_value}}	Evénement à une date précise	
{{fields_type.date_debut.values.0.format_value}}	Evénement sur une période : date de début	
{{fields_type.date_fin.values.0.format_value}}	Evénement sur une période : date de fin	
{{fields_type.etagere.values.0.value}}	Sélection : identifiant de l'étagère associée à un article	
{{fields_type.identifiant_notice.values.0.value}}	Coup de cœur : identifiant de la notice associée à un article	

Propriétés CSS des cadres

LE FLUX D'UN DOCUMENT

Le **flux normal d'un document correspond au comportement naturel des éléments d'une page Web**.

Les éléments sont placés les uns après les autres dans le code HTML, et c'est cet ordre qui détermine l'ordre d'affichage des éléments dans la page. C'est donc le navigateur qui décide où il va placer les cadres (comme si le code HTML était un texte qu'on lisait de haut en bas).

Les cadres du portail sont des éléments de type « block » : des espèces de « briques ».

Par défaut, les blocs se placent naturellement les uns en-dessous des autres, dans le flux normal, et ils occupent toute la largeur de la zone où ils se trouvent. C'est ce qu'on appelle la position « statique ».

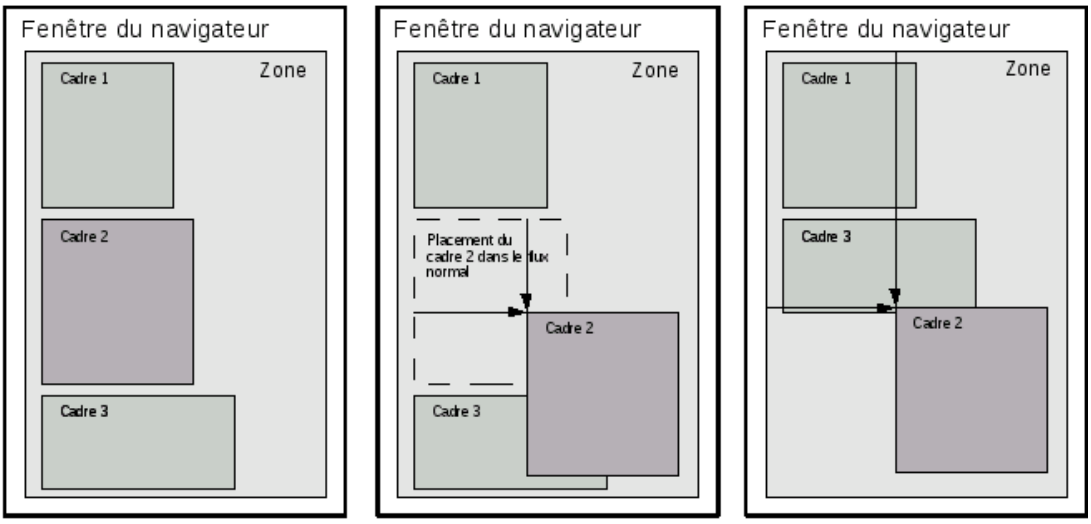
Pour modifier ce comportement par défaut, on peut agir sur 2 propriétés des cadres : position et float.

La propriété « position »

- **Position statique** : c'est le placement par défaut, le cadre est dans le flux du document, à sa place « normale »
- **Position relative** : le cadre sort du flux et est décalé par rapport à la position qu'il aurait normalement dû avoir, c'est pourquoi l'espace que ce cadre aurait dû occuper reste vide
- **Position absolue** : le cadre sort du flux et est positionné par rapport au coin supérieur gauche de la zone en position relative ou absolue qui le contient, sa place ne dépend donc plus de l'ordre dans lequel les éléments sont déclarés dans le code HTML
- **Position fixe** : le cadre sort du flux et est positionné par rapport à la fenêtre du navigateur, il ne bouge pas même si on scrolle

La propriété « float »

- Le cadre sort partiellement du flux, il est déplacé vers l'extrême gauche ou droite de la zone qui le contient, ce qui permet par exemple d'avoir deux cadres côte à côte.



Les 3 cadres sont en position statique.

Le cadre 2 est en position relative. Les cadres 1 et 3 sont en position statique.

Le cadre 2 est en position absolue. Les cadres 1 et 3 sont en position statique.

Espace réservé

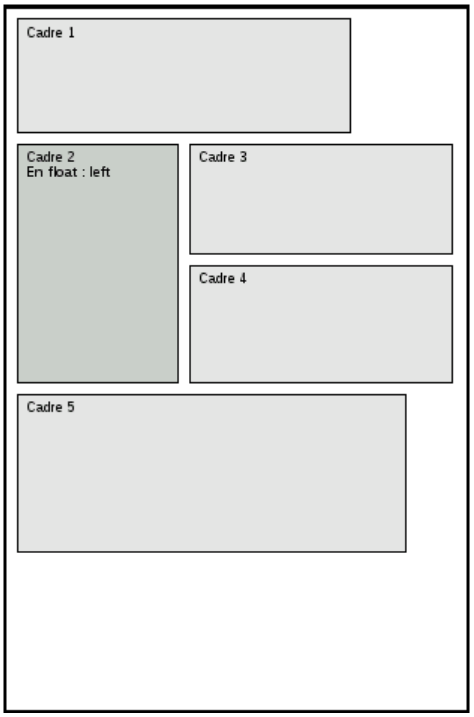
- En **position relative**, le navigateur conservera l'espace que le cadre occuperait s'il était dans le flux normal.
- En **position absolue**, le cadre n'a plus d'espace réservé dans le flux normal

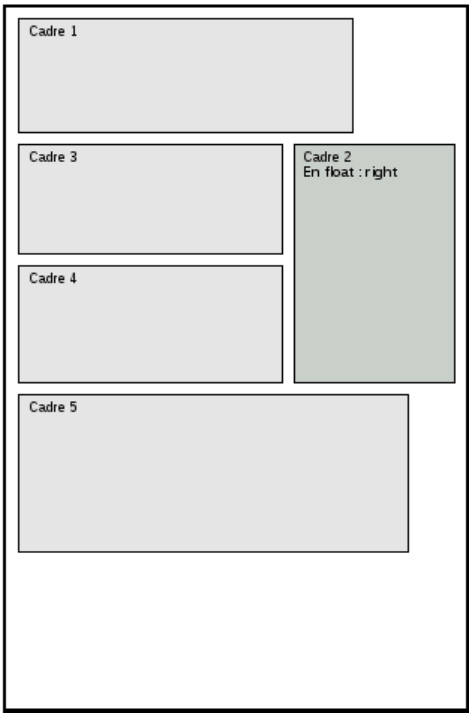
Source : Premiers pas avec le portail (document en ligne, publié par PMB Services)
http://doc.sigb.net/portail_pmb/co/description_deplacement.html

La propriété « float » permet d'entourer le cadre par d'autres cadres.

Float : left et float : right

- left : le cadre se place à gauche, les suivants se placent à sa droite.
- right : le cadre se place à droite, les suivants se placent à sa gauche.
- none : pas de float attribué au cadre.
- inherit : le cadre hérite du float de sa zone parente.





Insérer un clear (*ndlr* : lorsqu'on édite les propriétés CSS d'un cadre)
Cocher cette case annule l'effet du float left **ou du** float right **du cadre précédent.**

Source : Premiers pas avec le portail (document en ligne, publié par PMB Services)
http://doc.sigb.net/portail_pmb/co/utilisation_construction_editioncss.html

Commentaires à propos des deux exemples ci-dessus

- **C'est le cadre n° 2 qui « flotte » à gauche ou à droite**
- **Les cadres n° 3 et n° 4 ne peuvent se placer à côté du cadre n° 2 que si on a réduit leur largeur** (sinon, ils seraient en-dessous)
- Un cadre qui contient un « clear » ne va ni à gauche ni à droite du cadre flottant, il est dans le flux normal du document, en-dessous du cadre qui flottait. C'est comme si on traçait une ligne horizontale virtuelle à la fin de tout ce qui flotte.
- **C'est donc pour le cadre n° 5 qu'il faut cocher la case « clear »**

Autres ressources intéressantes sur le positionnement CSS : Site de Alsacréations : http://www.alsacreations.com/

Initiation au positionnement CSS – partie 1
http://www.alsacreations.com/article/lire/533-initiation-au-positionnement-en-css-partie-1.html

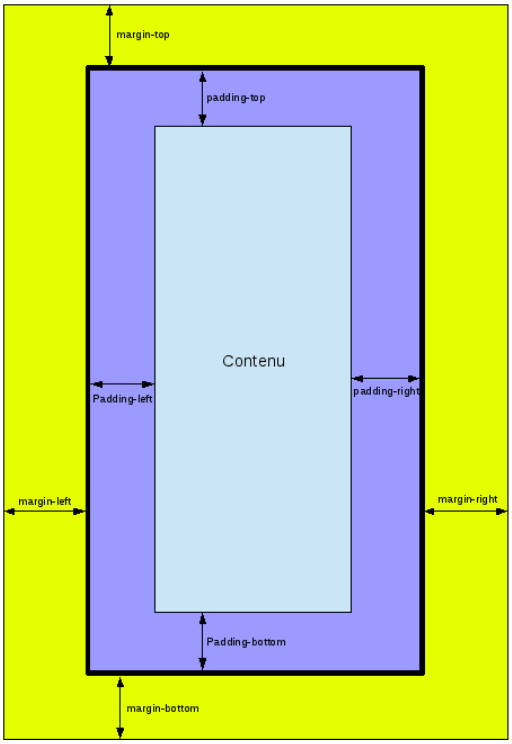
Initiation au positionnement CSS – partie 2
http://www.alsacreations.com/tuto/lire/608-initiation-au-positionnement-css-partie-2.html

Guide de survie du positionnement CSS
http://www.alsacreations.com/article/lire/53-guide-de-survie-du-positionnement-css.html

LES MARGES : MARGIN ET PADDING

Pour positionner les cadres les uns par rapport aux autres, on peut aussi employer les propriétés margin et padding.

Le padding d'un cadre représente l'espace entre la bordure du cadre et son contenu
Margin représente la marge entre la bordure du cadre et les autres cadres.



Padding et margin d'un cadre

auto : le cadre prend le padding ou le margin dont il a besoin.
px : définit en nombre de pixels le padding ou le margin.
% : définit le padding ou le margin du cadre en pourcentage de la largeur ou de la hauteur de la zone contenant le cadre
inherit : le cadre hérite du padding ou du margin de sa zone parente.

Si le **margin-left** et le **margin-right** sont en auto, le cadre est **centré**.

Source : Premiers pas avec le portail (document en ligne, publié par PMB Services)
http://doc.sigb.net/portail_pmb/co/utilisation_construction_editioncss.html

Dans un premier temps, j'ai surtout voulu expliquer la notion de flux d'un document, et présenter les propriétés CSS qui permettent de positionner les cadres les uns par rapport aux autres.
Je vous donnerai toutefois une liste complète de ces propriétés dans le chapitre suivant.

N'oubliez pas qu'un cadre est rarement seul dans une zone !
Comprendre comment le système fonctionne facilite « l'assemblage des pièces du puzzle ».

Je sais que ces concepts semblent fort théoriques pour le moment, pourtant il est nécessaire de les maîtriser.
Je vous rassure, vous aurez une vision plus concrète grâce aux explications et aux captures d'écran des pages suivantes ;-)

Manipuler les cadres

Le premier point important est d'attacher si nécessaire un cadre à une page

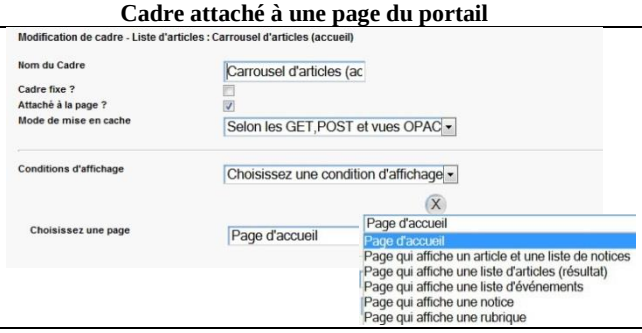
Si le cadre est fixe (visible dans toutes les pages), il suffit de cocher la case « fixe », et ensuite, on place ce cadre en général dans une zone fixe, comme l'introduction, le bandeau de gauche ou le pied de page.
Ainsi, il sera visible en toute circonstance, peu importe qu'on ait affiché une page de l'OPAC ou du portail.

Si le cadre doit être attaché à une page spécifique (du portail ou de l'OPAC), il faut procéder autrement.
Mais les captures d'écran des pages précédentes ne montrent pas exactement comment faire.
En effet, je vous ai montré jusqu'à présent des cadres déjà configurés, mais **l'écran n'est pas le même si on est en mode édition d'un cadre existant, ou en mode création d'un nouveau cadre.**

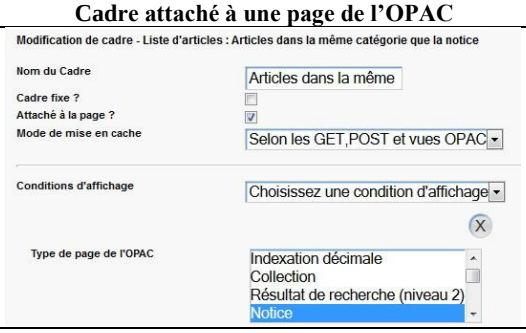
Modification d'un cadre existant

Si on est en mode édition du cadre, on ne voit qu'**une seule option** : soit la liste des pages du portail, soit la liste des pages de l'OPAC.

Cadre attaché à une page du portail

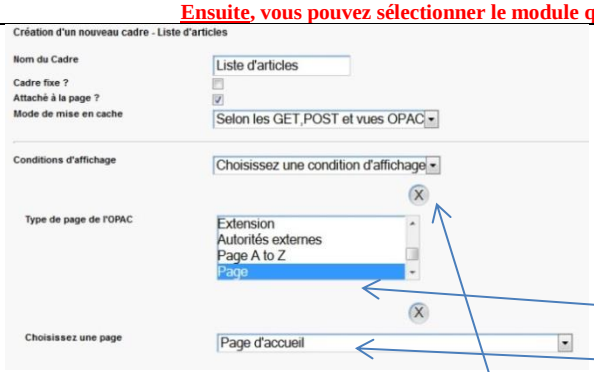


Cadre attaché à une page de l'OPAC



Création d'un nouveau cadre : comment l'attacher à une page

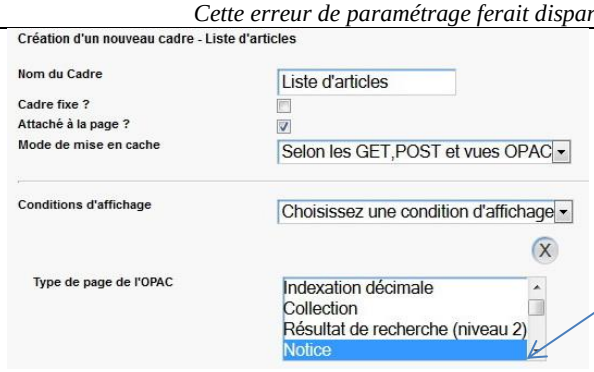
Pour attacher un cadre « solidement » à une page, il faut d'abord naviguer dans le volet de droite, afin d'afficher la bonne page !!
Ensuite, vous pouvez sélectionner le module qui servira à générer le cadre, et compléter le formulaire.



Page signifie qu'on n'a sélectionné aucune page de l'OPAC.
La page d'accueil du portail est sélectionnée automatiquement, puisqu'elle était visible dans le volet de droite.

Vous pouvez cliquer sur la croix, pour fermer la liste des pages de l'OPAC, mais ce n'est pas obligatoire.

Evitez bien sûr de sélectionner une page du portail et une page de l'OPAC !
PMB n'apprécierait pas votre indécision et vous le montrerait clairement ;-)
Cette erreur de paramétrage ferait disparaître le cadre dans les cadres « hors de la page » !!



La page de l'OPAC qui affiche une notice est sélectionnée automatiquement, puisqu'elle était visible dans le volet de droite.

On n'a même pas accès à la liste des pages du portail.

Création d'un nouveau cadre dans une page du portail

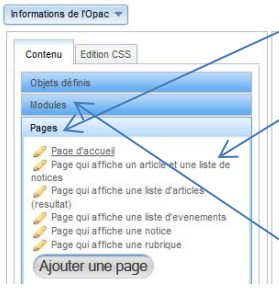
Comment faire s'il n'est pas possible de naviguer dans le volet de droite ?

Supposons que vous vouliez travailler dans la page qui affiche un article et une liste de notices, et y ajouter un cadre.

Si vous venez à peine de démarrer la construction de votre portail, naviguer dans le volet de droite sera impossible. Le doute est permis, surtout au début.

Toutes ces pages se ressemblent comme des gouttes d'eau : elles sont vides !

Comment être sûr que ceci est bien la page qui affiche un article et que les cadres qu'on veut placer iront bien là et pas ailleurs ?

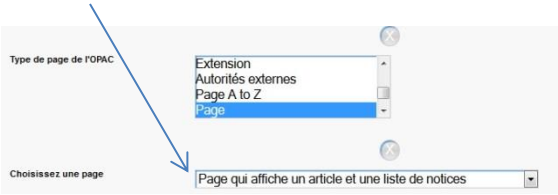


Cliquez sur « Pages » et sélectionnez la page concernée dans la liste.

Si c'est la « bonne » page du portail (celle que vous vouliez), vous pouvez y positionner des cadres.
Si ce n'est pas correct, refaites une sélection et l'écran se rafraîchira dans le volet de droite.

Cliquez ensuite sur Modules pour sélectionner le module qui servira à générer le cadre.

Le formulaire du nouveau cadre montre clairement que la bonne page du portail a été sélectionnée.



Après enregistrement du formulaire, le cadre apparaît en haut de la page centrale.
Ne cliquez surtout pas tout de suite sur « Enregistrer » (tout en bas du volet de droite) !!

Cadres de l'Opac:
Cadres du portail:
Cadres hors de la page:
Zones de l'Opac:

Les cadres du portail aboutissent « hors de la page » dans plusieurs cas de figure :

- Cela peut être tout à fait normal, s'ils sont liés à une autre page que celle visible dans le volet de droite.
- Cela peut être lié à une erreur de paramétrage, si on ajoute un cadre et que ses conditions d'affichage ne correspondent pas au type de page sélectionnée, ou si on modifie les conditions d'affichage d'un cadre préexistant et que cela déclenche un problème
- Cela peut révéler la maladresse des débutants … si on n'a pas bien « accroché » le cadre à une zone ;-)

Un petit mot à propos de la « maladresse des débutants »

- Ce genre de problème peut survenir si on inséré un cadre dans une page et qu'on l'a enregistré ou actualisé sans avoir d'abord positionné le cadre correctement dans une zone.

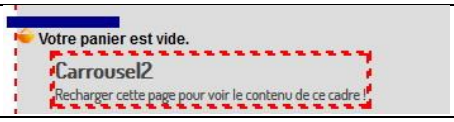
Méfiez-vous, certains cadres sont parfois récalcitrants et ont tendance à disparaître dans les cadres hors de la page …
Méfiez-vous aussi de certains cadres têtus, qui acceptent d'aller dans la bonne zone, mais pas à l'endroit que vous aviez choisi !

Positionner un cadre correctement dans une zone

Un nouveau cadre se place par défaut en haut de la page centrale, il occupe toute la largeur de la page centrale (puisque c'est un « bloc »), mais en réalité il n'est positionné dans aucune zone au départ. *Les apparences sont trompeuses !*

Il faut bien sûr déplacer un cadre s'il doit aller dans une autre zone, <u>mais aussi</u> s'il doit rester dans la zone « page centrale ».

<u>Avant</u> d'enregistrer et d'actualiser la page, il faut déplacer ce cadre à la souris, et voir le trait bleu qui indique l'endroit approximatif où on peut le déposer – même si cette position n'est pas parfaite au 1er essai, et même si vous le ramenez finalement à son point de départ … Vous pouvez redimensionner le cadre tout de suite ou plus tard.



« Baladez votre cadre » à la souris avant de cliquer sur « Enregistrer ».
Ne « lâchez » pas le cadre aussi longtemps que vous n'avez pas vu le trait bleu qui indique l'endroit où le déposer.
C'est la meilleure manière d'accrocher un cadre solidement à une zone !

Editer le contenu d'un cadre – Editer les propriétés CSS d'un cadre

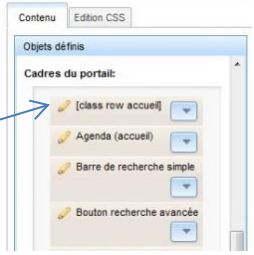
2 onglets : contenu - CSS

On peut cliquer sur deux onglets :

- « **Contenu** » → pour voir la liste des cadres (de l'OPAC, du portail, hors de la page)
- « **Edition CSS** » → pour éditer les propriétés CSS d'un cadre préalablement sélectionné

Pour éditer le contenu d'un cadre, cliquez sur l'icône du crayon (à gauche du nom donné au cadre)
Définir les propriétés générales du cadre (nom, page, source de données, template Django ...)

Pour éditer les propriétés CSS d'un cadre, cliquez sur son nom, et ensuite, sur l'onglet CSS.
Définir les attributs CSS du cadre sélectionné (dimensions, positionnement, marges …)
Si vous voulez retourner ensuite à la liste des cadres, cliquez sur l'onglet « **Contenu** »



Edition CSS des cadres – Liste complète des propriétés CSS		
Carrousel d'articles d'actualité **Page d'accueil** 	Position Left, top	**Position statique** : c'est le placement par défaut, le cadre est dans le flux du document, à sa place « normale » **Position relative** : le cadre sort du flux et est décalé par rapport à la position qu'il aurait normalement dû avoir, c'est pourquoi l'espace que ce cadre aurait dû occuper reste vide **Position absolue** : le cadre sort du flux et est positionné par rapport au coin supérieur gauche de la zone en position relative ou absolue qui le contient, sa place ne dépend plus de l'ordre dans lequel les éléments sont déclarés dans le code HTML **Position fixe** : le cadre sort du flux et est positionné par rapport à la fenêtre du navigateur, il ne bouge pas même si on scrolle **Left** et **top** permettent de préciser l'écart souhaité par rapport au bord latéral gauche ou supérieur. **Il se peut donc que des cadres se chevauchent**, en fonction de leur paramétrage respectif**.**
		Espace réservé pour un cadre qui est en position relative ou absolue En **position relative**, le navigateur conservera l'espace que le cadre occuperait s'il était dans le flux normal. En **position absolue**, le cadre n'a plus d'espace réservé dans le flux normal.
	Zindex	**Si des cadres se chevauchent, il peut être utile de savoir lequel se trouve « au-dessus » des autres.** Un cadre possédant un attribut **Zindex** supérieur à un autre se trouve « au-dessus ».
	Visibility	Cette propriété permet de rendre un cadre **visible ou invisible** : Visible : le cadre est visible Hidden : le cadre est invisible (hidden = caché)
	Height	Cette propriété permet de définir la **hauteur** d'un cadre : auto : le cadre prend la hauteur dont il a besoin. px : définit la hauteur du cadre en nombre de pixels. % : définit la hauteur du cadre en pourcentage de la hauteur de la zone contenant le cadre. inherit : le cadre hérite de la hauteur de sa zone parente.
	Width	Cette propriété permet de définir la **largeur** d'un cadre : auto : le cadre prend la largeur dont il a besoin. px : définit la largeur du cadre en nombre de pixels. % : définit la largeur du cadre en pourcentage de la largeur de la zone contenant le cadre. inherit : le cadre hérite de la largeur de sa zone parente
	Float	Cette propriété permet d'**entourer le cadre par d'autres cadres,** d'**en avoir plusieurs côte à côte**. left : le cadre se place à gauche, les suivants se placent à sa droite. right : le cadre se place à droite, les suivants se placent à sa gauche. none : pas de float attribué au cadre. inherit : le cadre hérite du float de sa zone parente.
	Padding Margin	Le **padding** d'un cadre représente l'espace entre la bordure du cadre et son contenu **Margin** représente la marge entre la bordure du cadre et les autres cadres. auto : le cadre prend le padding ou le margin dont il a besoin. px : définit le padding ou le margin en nombre de pixels. % : définit le padding ou le margin du cadre en pourcentage de la largeur ou de la hauteur de la zone contenant le cadre inherit : le cadre hérite du padding ou du margin de sa zone parente. Si le **margin-left** et le **margin-right** sont en auto, le cadre est **centré**.
	Display	**block** : le cadre est visible. **none** : le cadre est invisible.
	Visibility Display	Différence entre ces deux propriétés **Visibility : hidden** rend le cadre invisible mais conserve son emplacement. **Display : none** rend le cadre invisible et place les cadres suivants comme s'il n'existait pas.
	Insérer un clear	Cocher la case « **clear** » annule l'effet du float left ou du float right du cadre précédent.
Source : Premiers pas avec le portail (document en ligne, publié par PMB Services) http://doc.sigb.net/portail_pmb/co/utilisation_construction_editioncss.html		

On peut aussi éditer, de la même manière, les propriétés CSS des zones, mais je vous renvoie au chapitre Cascading Style Sheets.

Attention ! Les conflits de CSS peuvent générer de l'instabilité dans le portail, surtout s'ils concernent les dimensions et la position des zones (introduction, bandeau de gauche et de droite, page centrale, pied de page) ! Veillez à définir les zones dans la feuille de style même, et essayez de ne pas changer leurs propriétés principales à d'autres niveaux de CSS.

Propriétés CSS – Exemples			
Carrousel d'articles d'actualité **Attaché à la page d'accueil** **Zone page centrale**	**Barre de recherche rapide** **Cadre fixe** **Zone introduction**	**Carrousel de notices** **Attaché à la page d'accueil** **Zone page centrale**	**Menu** **Cadre fixe** **Zone introduction**
Contenu / Edition CSS Edition du Style cms_module_articleslist_21 Purger Position: static Left: 0 px Top: 0 px zIndex: Visibility: visible Height: 221 px Width: 370 px float: left margin-top: 0 px padding-top: 0 px margin-right: 0 px padding-right: 0 px margin-bottom: 15 px padding-bottom: 5 px margin-left: 0 px padding-left: 0 px display: block Insérer un clear	Contenu / Edition CSS Edition du Style cms_module_search_22 Purger Position: absolute Left: 719 px Top: 57 px zIndex: Visibility: visible Height: 51 px Width: 250 px float: none margin-top: 0 px padding-top: 0 px margin-right: 0 px padding-right: 0 px margin-bottom: 0 px padding-bottom: 0 px margin-left: 0 px padding-left: 0 px display: block Insérer un clear	Contenu / Edition CSS Edition du Style cms_module_recordslist_23 Purger Position: absolute Left: 595 px Top: 77 px zIndex: Visibility: visible Height: 520 px Width: 150 px float: none margin-top: 0 px padding-top: 0 px margin-right: 0 px padding-right: 0 px margin-bottom: 25 px padding-bottom: 0 px margin-left: 0 px padding-left: 0 px display: block Insérer un clear	Contenu / Edition CSS Edition du Style cms_module_menu_13 Purger Position: relative Left: 243 px Top: 90 px zIndex: Visibility: visible Height: 24 px Width: 463 px float: none margin-top: 0 px padding-top: 0 px margin-right: 0 px padding-right: 0 px margin-bottom: 0 px padding-bottom: 0 px margin-left: 0 px padding-left: 0 px display: block Insérer un clear

Quelques conseils pratiques

- Adaptez les dimensions de certains cadres, comme des carrousels de notices ou d'articles, en fonction du contenu à y mettre
- Si vous voulez qu'un cadre s'étende « naturellement » vers le bas, définissez sa hauteur en « auto »
- Si vous voulez que le portail s'adapte facilement aux diverses résolutions d'écran (style responsive), définissez la hauteur, la largeur, padding et margin en pourcentage plutôt qu'en pixels.
- Si un cadre doit être positionné et/ou dimensionné de manière précise, et que vous êtes « presque » arrivé au bon paramétrage, pensez à noter les valeurs actuelles ou à enregistrer une version du CMS avant de tenter une nouvelle modification ! Sinon, vous pourriez éprouver quelques difficultés à retrouver « la configuration qui était presque correcte, à un détail près » ;-)

Vous pouvez changer la position d'un cadre de deux manières :

- **Modifier certaines propriétés CSS : position, left et top, margin et padding –** Voir pages précédentes
- **Déplacer le cadre à la souris**

Objets définis – Sélection des objets – Déplaçables - Récepteurs

Déplaçables : en général, on coche « **cadres** », afin de pouvoir les déplacer
Récepteurs : en général, on coche « **zones** », on déplace le cadre par rapport à une zone

N.B. Si on voulait déplacer une « zone », ce serait par rapport au « fond » (l'ensemble de la page), mais je rapppelle qu' il vaut mieux gérer le placement des zones dans la feuille de style, afin d'éviter des conflits de CSS à différents niveaux.

Deux mini-cours simultanés – vous êtes bien entendu multitâches, cela va de soi ;-)
Un rappel de grammaire française et une révision du code de la route …

- La différence entre l'infinitif et le participe passé des verbes en –er
  - « Activer » à l'infinitif signifie que quelque chose est à activer
  - « Activé » (le participe passé) signifie que quelque chose a été activé
- La différence entre les feux verts et rouges
  - On a ou on n'a pas le droit de faire quelque chose

Posons d'abord la question de base : activer … mais que faut-il activer ?
Il s'agit du droit de déplacer un objet à la souris.

Par défaut, ce droit n'est pas activé, le feu est rouge, sinon dès que vous cliqueriez sur un cadre (volontairement ou non), il pourrait se balader ! C'est donc une mesure de sécurité bien utile, pour éviter les déplacements intempestifs de cadres.
Vous sélectionnez un cadre, vous cliquez sur « Activer », et vous pouvez agir (le message change, le feu est vert).
Par conséquent, dès que vous avez changé la position du cadre concerné, n'oubliez pas de cliquer sur « Activé » afin de bloquer à nouveau les cadres ! (il faut voir le message « Activer » et le feu rouge)

Attention à l'ordre des opérations !

Sélectionnez d'abord le cadre que vous voulez déplacer en cliqaunt sur son nom !

Lorsque vous cliquerez ensuite sur « Activer » afin que le message devienne « Activé », un seul cadre sera déplaçable : celui que vous aviez sélectionné. Ce qui est beaucoup plus sûr, les autres cadres ne risquent pas de bouger.

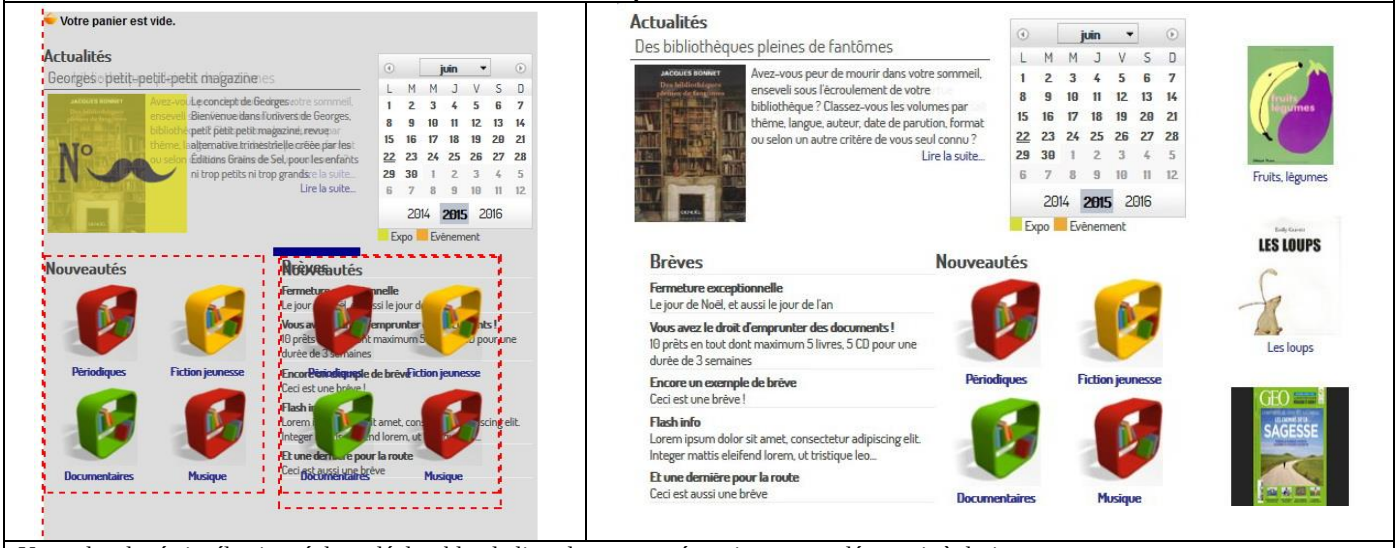
Ne restez en mode Activé (feu vert) que le temps strictement nécessaire !!
Retournez en mode Activer (feu rouge) dès que possible

Si vous cliquez d'abord sur « Activer » afin que le message devienne « Activé », tous les cadres seront déplaçables.

Ce qui est évidemment beaucoup plus dangereux !!
Le moindre clic de souris peut déclencher des catastrophes …

Ici, je n'avais déplacé qu'un seul cadre – intentionnellement - mais vous avez déjà un aperçu des dégâts potentiels ;-)

Déplacer un cadre



Un seul cadre était sélectionné donc déplaçable : la liste de nouveautés, qui se trouve désormais à droite.
Comme lors de la création d'un cadre, il faut absolument voir le trait bleu qui indique l'endroit où vous pourrez déposer le cadre.

Comment enregistrer ces modifications graphiques ?

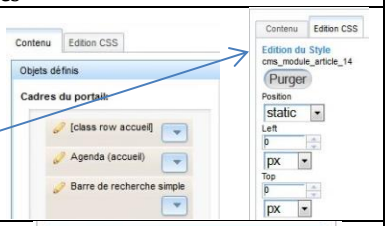
Les choses ne se passent pas de la même manière que pour d'autres éléments du portail.

- **Contenu éditorial**
  - Lorsque vous paramétrez les types de contenu ou les statuts de publication pour les rubriques et les articles, vous voyez un bouton « Enregistrer » dans le formulaire (onglet Administration).
  - Lorsque vous ajoutez ou éditez un article ou une rubrique, vous voyez un bouton « Enregistrer » dans le formulaire (onglet Portail)
- **Gestion des cadres**
  - Lorsque vous créez un nouveau cadre ou éditez un cadre existant, vous voyez un bouton « Enregistrer » dans le formulaire. (onglet Portail > Construire)
- **Gestion des templates Django**
  - Lorsque vous créez un nouveau template ou éditez un template existant, vous voyez un bouton « Enregistrer » dans le formulaire. (Onglet Portail > Modules > Interface d'administration)

On peut modifier les propriétés CSS des objets de deux manières

Pour éditer les propriétés CSS d'un cadre, cliquez sur son nom, et ensuite, sur l'onglet CSS.
Définir les attributs CSS du cadre sélectionné (dimensions, positionnement, marges …)

Mais comme vous avez pu le constater dans les captures d'écran des pages précédentes,
le formulaire d'édition des propriétés CSS ne contient pas de bouton « Enregistrer »



On peut déplacer un cadre à la souris, après avoir activé ce droit.

Mais le bouton « Enregistrer » brille là aussi par son absence …

Si vous avez modifié les propriétés CSS d'un objet d'une des deux manières expliquées ci-dessus, et si vous voulez sauvegarder ces changements, il faut cliquer sur « Enregistrer »

La sauvegarde est dans ce cas un peu moins spontanée, moins intuitive que quand on voit le bouton « Enregistrer » dans un formulaire ! Il faut y penser …
Enregistrez régulièrement ! Cela permet de retourner en arrière si nécessaire.

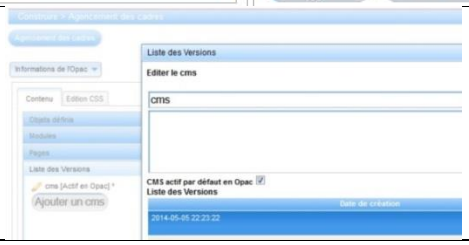
Enregistrer une « version » du portail ne signifie pas « enregistrer tout le portail » ni « enregistrer tout l'aspect graphique » !

Lorsqu'on a déplacé ou modifié des éléments, il faut enregistrer les nouvelles propriétés CSS des zones et des cadres afin de préserver les modifications – du moins si on est content de cette mise en page !

Cliquer sur « enregistrer » (dans le volet de droite, en bas).
Cela crée une « version » : une copie de la mise en page.

Deux précautions valent mieux qu'une !
Vous pouvez aussi adopter une « conduite préventive » : si la mise en page actuelle vous convient bien dans l'ensemble, mais que vous voulez essayer d'améliorer « un petit quelque chose », cliquez sur « enregistrer » **avant** de faire des modifications !

Si le résultat ne vous plaît pas, restaurez la version antérieure ;-)
Il suffit de choisir dans la liste la « bonne » version à restaurer.



Ce système de « versions » sauvegarde seulement la mise en page, la disposition des zones et des cadres et leurs propriétés CSS,

- **pas les cadres en tant que tels** (le paramétrage du contenu est enregistré quand on sauvegarde le cadre)
- **ni le contenu éditorial** géré dans l'onglet Administration ou dans l'onglet Portail

Si vous avez entretemps supprimé un cadre, restaurer une « version » antérieure ne va pas le ressusciter !
Ce système de « version » permet uniquement de repositionner les cadres qui existent encore à l'endroit qu'ils occupaient auparavant, en rendant aux zones et aux cadres les propriétés CSS antérieures qui avaient été enregistrées.

PMB vous propose d'actualiser la page

- **après la création (ou la modification) d'un cadre, et son enregistrement**
- **ou après l'enregistrement d'une « version » du CMS**

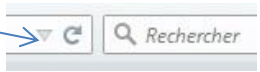
Il y a deux manières d'actualiser une page

Si on clique sur un des deux boutons « Actualiser la page », on va actualiser la page courante, celle qui est visible dans le volet de droite.

Donc, on rafraîchit la bonne page.

Si on clique sur « Agencement des cadres » ou si on rafraîchit la page dans l'onglet du navigateur, on retourne à la page d'accueil.

Il faudra donc éventuellement 1, 2 ou 3 clics de souris en plus pour parvenir à la page qu'on souhaitait actualiser.



Signification précise du terme « Actualiser une page » dans le contexte : enregistrer une version du CMS
Ce commentaire ne concerne donc pas l'actualisation d'une page après la sauvegarde d'un cadre

Jusqu'à présent, j'ai présenté les choses comme si vous aviez à chaque fois cliqué sur « Enregistrer » afin de sauvegarder les nouvelles propriétés CSS d'un cadre ou sa nouvelle position. Mais que se passe-t-il si vous n'avez pas cliqué ?

- **Si vous n'avez pas cliqué sur « Enregistrer », aucun changement n'a été enregistré, forcément !**
- **Si vous cliquez sur « Actualiser la page » sans avoir cliqué préalablement sur « Enregistrer », la page est rafraîchie, mais par rapport à la dernière version enregistrée du CMS** (il y a 5 minutes, il y a 2 heures, hier …)

Conséquence … et astuces ;-)

- **Si la modification que vous vouliez faire dans les propriétés CSS est tout compte fait une mauvaise idée, ou si vous avez déplacé un cadre mais que sa nouvelle position est une catastrophe graphique, n'enregistrez rien du tout, cliquez sur « Actualiser la page », et vous reviendrez à la situation antérieure !**
- **Evitez toutefois que la situation « antérieure » ne soit « trop ancienne » !**
- **Pensez à enregistrer régulièrement lorsque vous adaptez la mise en page ou que vous déplacez des cadres.**

Quelle version du CMS est active dans l'OPAC ?	Par défaut, c'est la dernière version en date qui est active.
On voit d'ailleurs son numéro identifiant dans les informations de l'OPAC.	Informations de l'Opac Url de l'Opac: /pmbv4perso2/opac_css/index.php?database=bibliv4perso2&cms_build_activate=1&build_id_version=207

Restaurer une version antérieure – Supprimer une version antérieure

Pour restaurer une version antérieure, cliquez sur la date et de l'heure de la version à restaurer. Si vous avez sélectionné la bonne version, je vous conseille de faire immédiatement une nouvelle sauvegarde.

En effet, il est beaucoup plus facile de gérer les choses si on est sûr que la dernière version en date est « la bonne », plutôt que de devoir retenir que c'est la 2e ou la 3e dans la liste.

Pour supprimer une version, cliquez sur la croix à droite.

Si le nombre de versions à supprimer est nettement plus élevé, je vous conseille d'employer une requête SQL.
Voir Requêtes SQL utiles

Requête SQL - Supprimer une ou plusieurs versions du CMS antérieure(s) à la version sélectionnée

Au fil du temps, le nombre de versions du CMS augmente dans la table cms_version, et il devient utile de lancer un « grand nettoyage de printemps ». C'est particulièrement le cas pour la démo du portail, où le nombre de versions du CMS est fort élevé.

La requête de suppression fonctionne selon le principe suivant

- on sélectionne une version du CMS
- la requête supprime la (les) version(s) du CMS antérieure(s) à la version choisie
- elle conserve la version sélectionnée et les versions postérieures

Cette requête est surtout conçue pour lancer « un nettoyage par le vide » dans des versions anciennes du portail, mais **soyez prudent dès qu'il s'agit de supprimer des versions relativement récentes !!**

Pour que le portail fonctionne, il vous faut « la bonne version » du CMS : celle qui est active pour le moment (celle dont on voit le numéro dans les informations de l'OPAC). Or, c'est en général la version la plus récente qui est active (sauf si vous venez de restaurer une version antérieure).

Si on veut vérifier à l'aise, avant de lancer la suppression, et savoir à quelle version du CMS un numéro d'id correspond, il suffit d'employer une autre procédure qui donne la liste des versions du CMS, et leur numéro identifiant.	En effet, la liste des versions du CMS fournie par PMB mentionne la date et l'heure de sauvegarde, mais pas le numéro d'id de la version.

Attention donc, ne lancez pas cette procédure de manière inconsidérée ! Vérifiez d'abord !!

En rédigeant cette requête, j'ai veillé à ce qu'il soit techniquement impossible de supprimer la dernière version en date, puisque cette suppression déclencherait probablement de sérieux dysfonctionnements (selon toute vraisemblance, c'est celle qui est active).
Assurez-vous que la version du CMS qui est active pour le moment ne fasse pas partie de celles que la requête pourrait supprimer.

Ne soyez pas surpris par le nombre de lignes affectées par la requête, il est largement supérieur au nombre de versions supprimées.

En effet, à chaque ligne de la table cms_version correspondent plusieurs lignes dans la table cms_build (une ligne par cadre).
Or, la requête supprime les traces d'une version dans les deux tables concernées.

Dans cet exemple, j'ai demandé la suppression des 3 versions antérieures à la version 177, donc la requête va supprimer 3 lignes dans la table cms_version, mais 138 lignes dans cms_build : 141 au total.

ADAPTER SON MODE DE TRAVAIL AU PORTAIL

Le champ « résumé »

DANS LE CONTENU ÉDITORIAL

Articles : veillez à bien répartir le texte entre le résumé et le contenu = l'article lui-même
Le résumé, c'est le chapeau d'un article, il doit donc être conçu pour accrocher le lecteur.

Il se peut aussi qu'un article n'ait qu'un seul champ complété, le résumé.
Ex. Un article qui explique les spécificités d'une étagère virtuelle, les points communs entre les notices

Les rubriques n'ont qu'un seul champ, le résumé, qui doit lui aussi être « accrocheur ».

Faites également attention aux titres des rubriques pour le menu : ils doivent être clairs et aussi concis que possible.

DANS LE CONTENU BIBLIOGRAPHIQUE

L'absence de contenu pour le champ résumé d'une notice (monographie, périodique, article de périodique ou notice de bulletin) peut parfois créer une impression de « vide » dans l'affichage abrégé de cette notice.

Voir exemples de périodiques (Géo)

Mais d'autre part, un résumé beaucoup trop détaillé, qui n'est pas conçu dans une optique de portail, va créer une impression de « surcharge », de longueur excessive.

Si vous voulez présenter certaines notices dans le portail, il faudra peut-être transférer provisoirement le contenu de certains résumés trop longs vers un autre champ, et créer un autre résumé à employer temporairement. A méditer !

Le logo des rubriques et articles

Le logo des rubriques et articles est en général un élément cliquable dans les templates Django, comme le titre ou des messages (ex. lire la suite). Donc il faut penser à en prévoir un.

D'ailleurs, c'est parfois même le seul élément cliquable dans le template d'origine !
Ex. Module agenda : affichage d'une liste d'événements à partir d'une date sélectionnée dans l'agenda
Ex. Template du module Liste de rubriques (vue : template Django avec lien) – non illustré dans la démo

Pensez à standardiser la taille de certains logos susceptibles d'être affichés sur la même page.
Si on voit une liste d'articles / de rubriques avec logos, il est plus esthétique d'avoir des logos de proportions similaires.

Le dernier sujet que je vais aborder dans ce chapitre ne concerne pas uniquement le portail, mais l'OPAC en général : il s'agit des notices de bulletins.

Un bulletin « ordinaire » n'est pas une notice, donc on ne peut pas le mettre dans un panier de notices pour le lier à une étagère. Il faut d'abord transformer le bulletin en « notice de bulletin ».

J'ai donc jugé utile de rappeler quelques notions de base à propos de ces notices de bulletins, sinon vous ne pourrez pas les mettre en évidence dans le portail.

Bulletins et notices de bulletins

Le mot « notice » a plusieurs significations dans PMB.

- En ce qui concerne la **notice d'une monographie** (livre, CD, DVD …), on voit en général tout de suite de quoi il s'agit : titre, complément de titre, ISBN, auteur, éditeur, nombre de pages, année de parution, indexation décimale, catégories …
- La **notice d'un périodique** est la **notice-mère de ce périodique** (titre, ISSN, éditeur …). Cette notice de périodique est globale, elle n'a pas de numéro ni de date de parution, et n'est liée à aucun exemplaire.
- Le terme **notice d'article** désigne les **divers articles publiés dans un bulletin** (numéro d'un journal ou d'une revue). Dépouiller un périodique, c'est dresser la liste de ces articles (titre, auteur, page de début et de fin, catégories …)
- Et finalement, il y a la **notice de bulletin** : elle comporte au départ les mêmes données que le bulletin, mais on y a ajouté des infos complémentaires (note de contenu, résumé, catégories …)

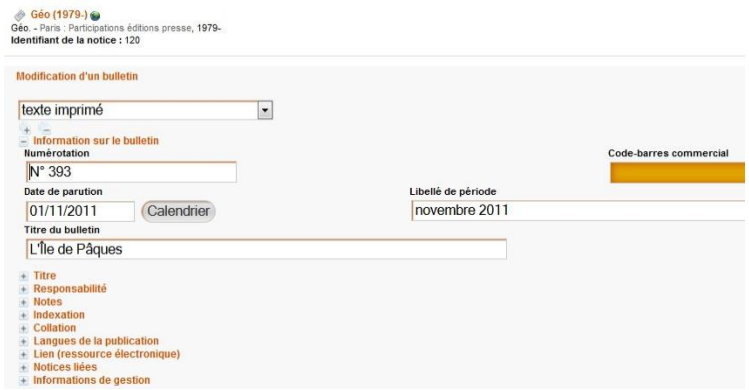
Toutes ces notices sont enregistrées dans la table des notices. Le contenu du champ niveau_biblio permet de les différencier.
niveau_biblio = m (monographie) niveau_biblio = a (article) niveau_biblio = s (périodique) niveau_biblio = b (notice de bulletin)

Différences entre un bulletin et une notice de bulletin

Dans PMB, un bulletin « simple, ordinaire » ne comprend que les champs visibles au début : numérotation, code-barres commercial, date de parution, libellé de période et titre du bulletin. Rien d'autre n'est prévu dans la table bulletins. **Impossible de le mettre dans un panier de notices.**

Seul un bulletin peut avoir des données d'exemplaire, comme un numéro, une cote de rangement, une localisation, un statut …

Dès que vous ajoutez d'autres infos (auteur, notes, résumé, collation, catégories, langues ...), ces infos ne peuvent techniquement pas être stockées dans la table des bulletins, car celle-ci ne contient pas les champs adéquats.



A ce moment-là, dès que vous avez complété un autre champ, PMB crée automatiquement et discrètement une notice de bulletin, une espèce de « clone » de ce bulletin dans la table des notices. **Mais si vous effacez les infos supplémentaires ajoutées entretemps, cette notice de bulletin sera supprimée automatiquement (et tout aussi discrètement) par PMB, le bulletin redeviendra un simple bulletin.**
Attention ! N'essayez pas d'interférer (par le biais de requêtes SQL), avec le processus automatique de création ou de suppression des notices de bulletins, vous risquez de provoquer des dégâts collatéraux dans la base !

N.B. Ne confondez pas la date de parution et le libellé de période

Le libellé de période est un champ dont la formulation est libre, ce qui empêche tout tri cohérent. Quelques exemples
Comment PMB pourrait-il « deviner » que « mars » précède « mai » dans le calendrier, alors que le tri alphabétique dit l'inverse ?
Comment PMB pourrait-il « deviner » que pour un trimestriel, « 1er trimestre 2015 » a le même sens que « Janvier –Mars 2015 » ?
En outre, si vous avez des périodiques en plusieurs langues, la diversité dans les libellés de période augmentera encore.

La date de parution est une date à saisir au format SQL, elle seule permet de trier les bulletins par ordre chronologique inverse.
Attention ! Ne laissez pas ce champ vide, car PMB a absolument besoin d'une date de parution pour le tri (une « vraie » date SQL). Or, si vous ne complétez pas ce champ, PMB va y insérer d'office la date où vous avez catalogué le bulletin, ce qui fausse tout !

Le fait qu'un bulletin soit « simplement » un bulletin empêche de faire certaines choses, par exemple :

- Vous ne pouvez pas mettre le bulletin dans un panier de notices afin qu'il apparaisse dans une étagère virtuelle ou dans la DSI
- Vous ne pouvez pas effectuer de recherche sur le titre du bulletin
- Vous ne pouvez pas lui donner un statut équivalent à celui des notices (visible ou non dans l'OPAC, réservé aux abonnés …)

L'impact est direct dans un OPAC classique ou dans un OPAC de style portail.
Vous ne pouvez pas mettre un bulletin en évidence dans une liste de notices ou un carrousel de notices, vous ne pouvez même pas le mettre (via un panier de notices) dans une étagère virtuelle de l'OPAC, puisqu'un bulletin n'est pas une notice !

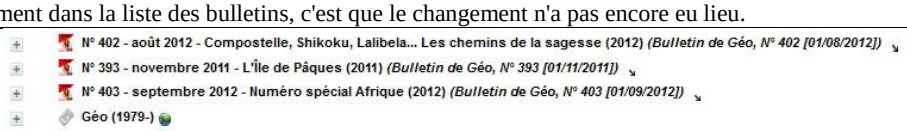
Il est donc parfois recommandé de "forcer" manuellement le processus de changement, afin de transformer ce simple bulletin en une notice de bulletin.

Impossible de le faire par une requête SQL, (il faudrait un bout de code en PHP), il faut donc le faire soi-même.

Dès que vous complétez une des infos supplémentaires (complément de titre, résumé, notes, auteur, langues, peu importe), la transformation du bulletin en notice de bulletin s'effectue, et celle-ci sera visible dans le catalogue.

Si votre bulletin apparaît dans une liste de notices (comme la liste des « dernières notices »), ou si vous voyez le terme « notice de bulletin », c'est que le bulletin a subi une « mutation génétique », et qu'il est en effet devenu une notice !
Si vous continuez à voir le bulletin uniquement dans la liste des bulletins, c'est que le changement n'a pas encore eu lieu.

Ex. : liste des dernières notices
On voit 3 notices de bulletin de Géo, ainsi que la notice-mère de ce périodique.



Après ce changement, vous pouvez mettre ce bulletin dans un panier de notices, et l'afficher dans le portail ou l'OPAC.

Le titre d'un bulletin est en général significatif, puisqu'il reflète le dossier principal qui a été traité dans ce numéro.
Ce titre de bulletin revêt une importance encore plus grande si vous n'avez pas eu le temps de dépouiller les articles de ce numéro, puisqu'il permet d'avoir malgré tout une idée du thème majeur qui a été développé.

Toutefois, si un bulletin reste « un simple bulletin », et ne devient pas une notice, il est impossible d'effectuer une recherche sur le titre du bulletin. En effet, la recherche se fait sur les différents champs de la notice.

Pour transformer ce bulletin en notice, il vaut mieux ne pas compléter n'importe quel champ pris au hasard.

Autant agir efficacement et faire d'une pierre deux coups !

Je vous conseille de copier-coller le titre du bulletin dans le champ « note de contenu », champ dans lequel vous pourriez aussi ajouter un aperçu des articles les plus importants de ce numéro (si vous n'avez pas eu le temps de dépouiller les articles un par un).

Vous pourriez aussi choisir un autre champ, mais n'oubliez pas que **dans le contexte du portail, le champ « résumé » a une fonction particulière.** Si vous y copiez-collez le titre du bulletin, pensez à y ajouter quelques phrases qui vont attirer l'attention des lecteurs.

Un petit copier-coller du titre du bulletin, par exemple dans le champ note de contenu, remplit plusieurs fonctions :

- lancer le processus de « mutation génétique » : transformer le bulletin en notice de bulletin
- permettre que la recherche dans tous les champs fonctionne, on « trouve » le titre du bulletin, qui est la plupart du temps significatif (sinon, cette info reste inaccessible)
- permettre de placer cette notice de bulletin dans un panier de notices, afin de l'afficher dans une étagère virtuelle de l'OPAC, dans un cadre du portail, ou bien dans la DSI (Diffusion Sélective de l'Information)
- permettre d'agir sur le statut de cette notice, et sur sa visibilité éventuelle dans l'OPAC (pour tous, pour les abonnés …)

Exemple : le contenu de 2 paniers de notices (avec indication du type de notice) – Voir requêtes SQL utiles

	A	B	C	D	E	F	G	H	I	J	K	L
1	Id	Panier	ISBN / ISSN	Auteur	Type de notice	Id	Titre de la notice (+ titre de série)	Libellé de période	Date de parution	Editeur	Cote	Numéro (ex)
2	14	Nouveautés fiction jeun	978-2-87833-390-9	Heitz, Bruno	Monographie	81	Jojo (série) : Jojo et le secret de la		2006	Circonflexe	RJP HEI	000045
3	14	Nouveautés fiction jeun	978-2-07-053882-9	Mahy, Margaret	Monographie	87	L'enlèvement de la bibliothécaire		2002	Gallimard jeunesse	RJ MAH	000037
4	14	Nouveautés fiction jeun	978-2-01-322406-2	Gudule,	Monographie	80	La bibliothécaire		2007	Hachette Jeunesse	RJ GUD	000046
5	14	Nouveautés fiction jeun	978-2-203-12937-5	Brisou-pellen, Evelyn	Monographie	85	Le grand amour du bibliothécaire		2005	Casterman	JR BRI	000039
6	18	Nouveautés périodique:		Bonnet, Jacques	Article	125	Bibliothécaire de soi-même in Bulle		01/11/2010			
7	18	Nouveautés périodique:			Article	133	Dark Shadows in Studio ciné live n	avril 2012	01/04/2012		PER STU	P000006
8	18	Nouveautés périodique:		Aïoutz, Renaud	Article	127	L'abeille et le bibliothécaire ou éthi		01/09/2012			
9	18	Nouveautés périodique:		Akkouche, Mouloud	Article	126	Voyages en BM in Bulletin des bib		01/11/2010			
10	18	Nouveautés périodique:			Notice de bulletin	122	Géo - N° 393 - novembre 2011 - L'Îl	novembre 2011	01/11/2011		PER GEO	P000002
11	18	Nouveautés périodique:			Notice de bulletin	123	Géo - N° 402 - août 2012 - Compos	août 2012	01/08/2012			
12	18	Nouveautés périodique:			Notice de bulletin	121	Géo - N° 403 - septembre 2012 - N	septembre 2012	01/09/2012		PER GEO	P000003
13	18	Nouveautés périodique:			Notice de bulletin	131	Georges - Numéro Cupcake		24/10/2012		PER J GEO	P000005
14	18	Nouveautés périodique:			Notice de bulletin	130	Georges - Numéro Moustache		24/10/2012		PER J GEO	P000004
15	18	Nouveautés périodique:	1292-8399		Périodique	124	Bulletin des bibliothèques de Franc			ENSSIB		
16	18	Nouveautés périodique:	0220-8245		Périodique	120	Géo			Participations éditio		
17	18	Nouveautés périodique:	2110-8749		Périodique	129	Georges			Éd. Grains de sel		
18	18	Nouveautés périodique:	1969-9441		Périodique	132	Studio ciné live magazine			Studio magazine		

EXEMPLES SUPPLÉMENTAIRES

Le point commun de ces exemples supplémentaires est qu'ils ne sont pas illustrés dans la démo du portail, contrairement aux exemples du chapitre précédent.
A partir du moment où on a compris les concepts de base liés aux modules et cadres, et les connexions entre les pages du portail, on peut se mettre à créer des variantes et à innover ;-)

Variantes de quelques cadres et pages

Les cadres et les pages (portail ou OPAC) de la démo ne sont pas « intangibles », vous pouvez par exemple

- conserver un cadre mais **modifier le critère de sélection des données ou changer de « vue » (le mode d'affichage)**
- **ajouter un cadre** dans une page existante
- **créer de nouveaux types de contenu (articles)** pour **diversifier les critères de sélection**

PAGE DU PORTAIL QUI AFFICHE UNE NOTICE

AJOUTER LE CADRE « NOTICES DU MÊME AUTEUR PRINCIPAL »

Dans la démo, le cadre « Notices du même auteur principal » est inséré dans la page de l'OPAC qui affiche une notice.

Dans le commentaire de ce cadre, j'avais montré que cela risquait de surcharger cette page de l'OPAC, et de renforcer une impression de « déjà-vu », surtout si on y affichait déjà des suggestions comme « Dans le même rayon » ou « Peut-être aimerez-vous ». En effet, cela augmenterait probablement le nombre de notices et donc de vignettes identiques.

« Notices du même auteur » (version OPAC)

Rappel - Ces cadres sont visibles dans la page OPAC qui affiche une notice, à la fin des cadres habituels.

A la fin de mes explications initiales, j'avais suggéré d'insérer ce cadre dans la page du portail qui affiche une notice. Ainsi, ce cadre ne serait pas visible en permanence dans l'OPAC pour toutes les notices de la base, on ne le verrait que si la notice est affichée dans la page adéquate du portail. Voici comment procéder.

Page du portail qui affiche une notice
Paramétrage de la démo - Un seul cadre : affichage d'une notice

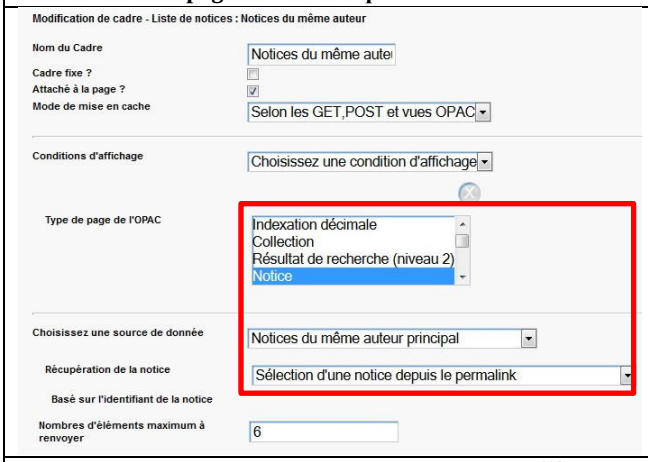
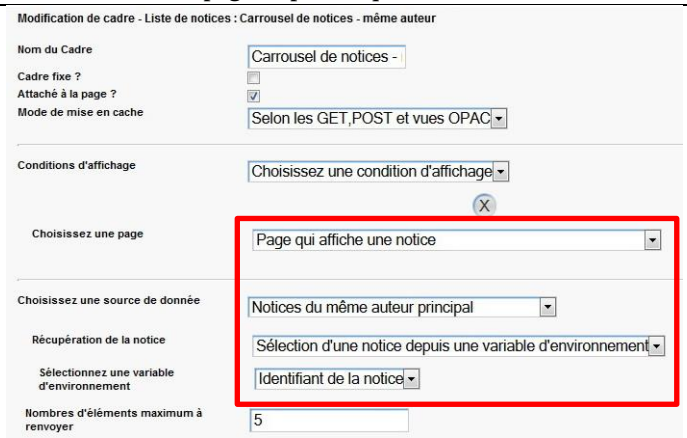
Cette page est peu remplie, on peut facilement y ajouter un cadre supplémentaire, sans nuire à l'esthétique de l'ensemble.

Je vais montrer comment ajouter le cadre : Notices du même auteur.
Je garde l'idée du carrousel horizontal (comme dans la version OPAC).

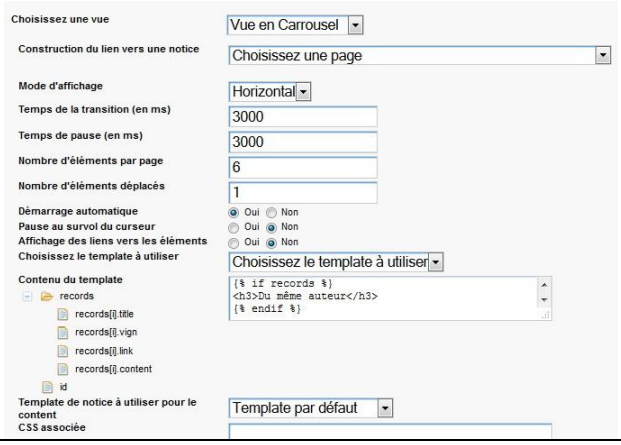
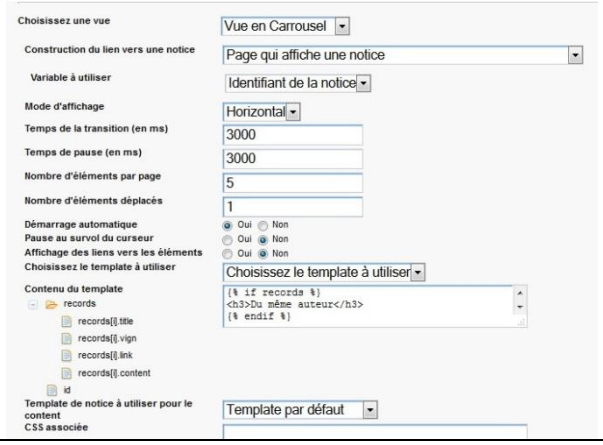
Mais vous pourriez aussi employer d'autres critères de sélection, et donc d'autres modules pour générer un nouveau cadre dans cette page :
Notices similaires
Notices dont la cote de rangement est proche

Je ne vais pas détailler le paramétrage de ce cadre en version « page du portail », vu qu'il quasi identique à la version OPAC.
Je vais seulement relever quelques différences. N.B. Si nécessaire, relire « Notices du même auteur principal ».

Formulaire pour le carrousel de notices du même auteur qu'une notice– Module « Liste de notices »

Version : page de l'OPAC qui affiche une notice	Version : page du portail qui affiche une notice
	

Différences entre ces deux formulaires

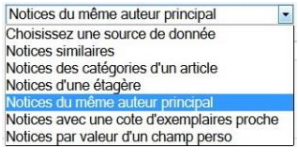
Cadre attaché à une page de l'OPAC Sélection d'une notice depuis le permalink	Cadre attaché à une page du portail Sélection d'une notice depuis une variable d'environnement
URL de la page de l'OPAC qui affiche une notice Le permalink contient l'dentifiant de la notice. http://127.0.0.1/... /opac_css/index.php?lvl=notice_display&id=112	**URL de la page du portail qui affiche une notice** La variable d'environnement est ici l'identifiant de la notice (id_notice) http://127.0.0.1/..../opac_css/index.php?lvl=cmspage&pageid=7&id_notice=112
	

Aucune différence sgnificative (j'ai par hasard réduit à 5 le nombre d'éléments à renvoyer et à afficher, mais c'est un détail).

Résultat final

Rappel : on aurait aussi pu employer un autre critère de sélection des données que les notices du même auteur principal, par exemple les notices similaires ou celles dont la cote de rangement est proche.

Des choix comme « Notices des catégories d'un article » ou « Notices d'une étagère » seraient bien sûr impossibles, vu que la page courante est celle qui affiche une notice, et qu'il n'y a dès lors aucun lien avec les articles.

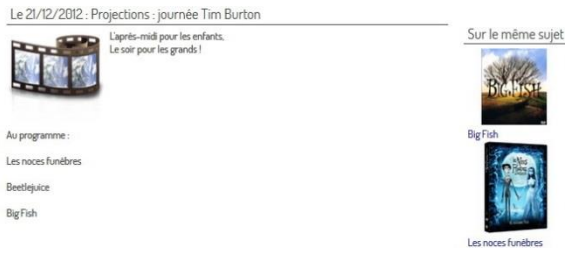
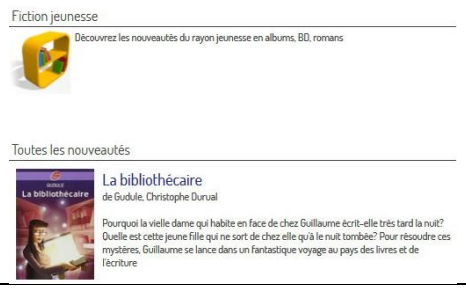


PAGE DU PORTAIL QUI AFFICHE UN ARTICLE

UN AUTRE USAGE DU CADRE « LISTE DE NOTICES – ETAGÈRE LIÉE À UN ARTICLE »

La page qui affiche un article et une liste de notices comprend trois cadres, mais tous ne sont pas nécessairement affichés.

- **Un cadre issu du module Affichage d'un article**
  - Visible en toute circonstance – c'est le cadre principal de cette page
- **Carrousel de notices dans les mêmes catégories qu'un article** – Etagère créée de manière dynamique
  - Visible si certaines catégories sont communes à l'article et aux notices - Peu importe le type de contenu de l'article
- **Liste de notices - Une étagère thématique associée à un article** : ex. Nouveautés du rayon Fiction jeunesse
  - Visible seulement si l'article est de type Sélection et si l'étagère préalablement choisie contient des notices

Carrousel de notices dans les mêmes catégories qu'un article	Liste de notices d'une étagère associée à un article
	

Dans les commentaires qui suivent l'analyse du cadre « Notices du même auteur qu'une autre notice », **j'avais montré que ce mode de sélection n'affichait pas toujours les notices que vous auriez voulu mettre en évidence, et que ce type de problème pouvait aussi se poser pour d'autres cadres dont le contenu est créé de manière dynamique.**

Si vous laissez faire les choses sans intervenir, vos lecteurs risquent parfois de voir

- des notices fort anciennes ou peu pertinentes – si le choix se fait dans une base bien étoffée
- souvent les mêmes notices - si le choix se fait dans une base de taille réduite

S'il s'agit simplement d'offrir quelques suggestions de lecture complémentaire, vous pouvez opter sans souci pour la création dynamique du contenu de l'étagère.
Toutefois, si la pertinence des résultats est à vos yeux le critère essentiel pour certains carrousels ou listes spécifiques, choisissez vous-même les notices à afficher, via le sélecteur manuel d'une étagère thématique.

Ce que je vous suggère de faire n'est pas une innovation en soi : **les étagères thématiques existent déjà dans le portail.**

En effet, dans la page du portail qui affiche un article, on voit les Notices d'une étagère associée à un article.
Dans la démo, il s'agit des étagères qui contiennent les nouveautés dans divers rayons, ex. Fiction jeunesse, Fiction adultes …

L'objectif est de remplacer le carrousel de « Notices dans les mêmes catégories qu'un article » par une liste de notices liées à une étagère spécifique. Le but principal est donc de changer le mode de sélection des notices à afficher.

Le fait de remplacer un carrousel par une liste a aussi de l'importance, mais les questions à se poser sont autres :

- Faut-il afficher des données relativement détaillées sur la notice ?
  - Si la réponse est oui, la liste s'impose (on peut paramétrer les champs à afficher par un template de notice)
  - Si la réponse est non, si le titre et la vignette suffisent, le carrousel convient mieux

- Quel type de présentation est préférable en termes d'ergonomie, d'équilibre graphique de la page et du portail ?
  - Une liste ?
  - Un carrousel ? Horizontal, vertical ou en fondu enchaîné ?

Il est facile d'éviter l'affichage du carrousel des « Notices dans les mêmes catégories qu'un article ».

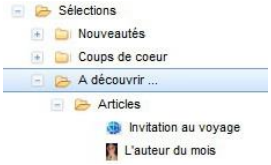
Veillez à ce que l'article ne soit lié à aucune catégorie, et le tour est joué !
On ne verra pas le carrousel ...

Il est tout aussi facile d'obtenir une liste des « Notices d'une étagère associée à un article ».

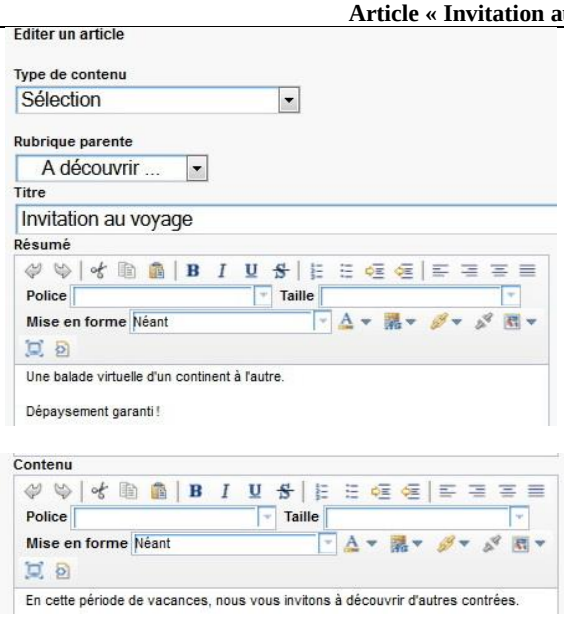
Veillez à ce que le type de contenu de l'article soit « Sélection », et le tour est joué !
Du moins si vous avez associé l'article à une étagère qui contient effectivement des notices !
On verra la liste des notices que vous vouliez afficher, et seulement celles-là.

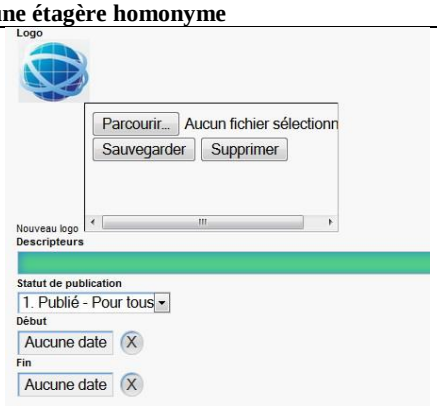
On n'a jamais dit que le type de contenu « Sélection » était réservé aux articles qui présentent les nouveautés ;-)
Ce que vous voyez dans la démo est une illustration d'une des possibilités, mais il y en a d'autres à explorer …

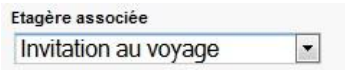
Exemple concret d'un autre usage du type de contenu « Sélection »
J'ai créé une rubrique « A découvrir » qui contient deux articles thématiques.



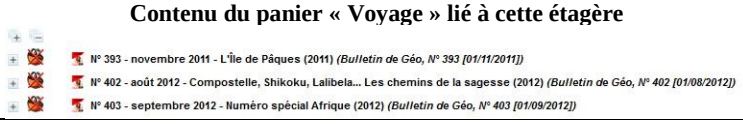
Article « Invitation au voyage » - Lié à une étagère homonyme







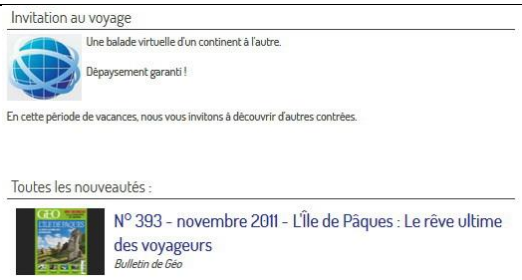
Contenu du panier « Voyage » lié à cette étagère



Premier test

De prime abord, tout semble correct, le cadre « Notices d'une étagère associée à un article » affiche les notices de l'étagère sélectionnée.

Toutefois, il subsiste un petit problème : le titre affiché au-dessus de la liste des notices !
Le titre « Toutes les nouveautés » reflète l'ancienne fonction du cadre, il ne convient donc plus.



Si vous voulez rendre le cadre « Notices d'une étagère associée à un article » vraiment polyvalent, il y a une seule petite modification à faire dans le template Django : changer le titre !

Il faut trouver un titre neutre, utilisable en toute circonstance, aussi bien pour les nouveautés des divers rayons que pour une étagère thématique. Je propose « Quelques suggestions » comme nouveau titre.

Je sais que ce n'est guère original, mais je manquais d'inspiration ;-)

Template Django du cadre « Notices d'une étagère associée à un article »	
Ancienne version	**Nouvelle version**
{% if records %} Toutes les nouveautés {% endif %} {% for record in records %} <blockquote>{{record.content}}</blockquote> > <hr> {% endfor %}	{% if records %} Quelques suggestions {% endif %} {% for record in records %} <blockquote>{{record.content}}</blockquote> > <hr> {% endfor %}
Il suffit d'éditer ce cadre, de remplacer l'ancien titre par un autre de votre choix, et de sauvegarder à nouveau le cadre. N.B. Si nécessaire, relire Notices d'une étagère associée à un article pour comprendre exactement de quoi il s'agit.	

Le nouveau titre convient pour une étagère thématique et pour les nouveautés d'un rayon	

Je voudrais également montrer que les « coups de cœur des bibliothécaires » ne se limitent pas nécessairement à la présentation d'un livre. Si on emploie un article de type Sélection lié à une étagère, on peut créer aisément un « coup de cœur » pour un auteur, une collection, un thème spécifique, etc.

La démo du portail contient déjà une œuvre de Ray Bradbury, *Fahrenheit 451*, un magnifique roman de science-fiction qui nous interpelle autant qu'au moment de sa parution initiale (en 1953), que nous soyons bibliothécaires ou non …

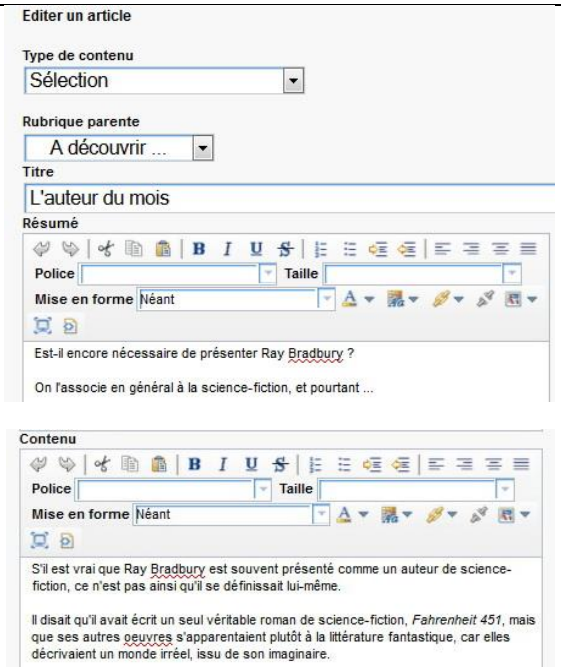
Imaginez un monde où la TV règne en maître et où la lecture est devenue interdite, car elle incite les gens à réfléchir ...
Un monde où on dénonce à la police ceux qui ont commis un « acte antisocial » : ils lisent et ont caché quelques livres chez eux …
Un monde où un corps spécial de pompiers a été créé afin de brûler les livres et les maisons des « criminels » qui en possèdent ...
Un monde où quelques individus se réfugient dans les bois et deviennent la « mémoire vivante des livres » : chacun a appris une œuvre par cœur, afin que celle-ci ne disparaisse jamais …

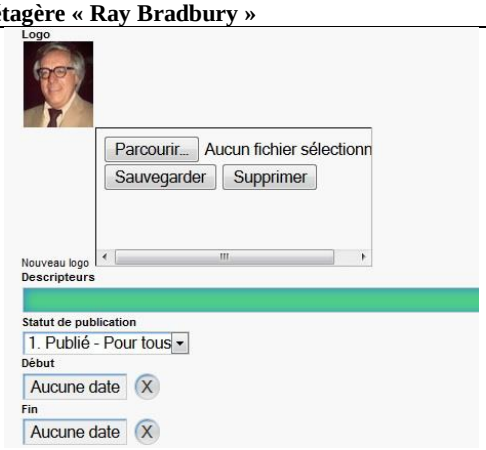
Etapes préliminaires

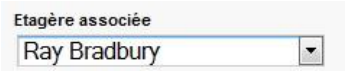
- J'ai ajouté dans la base une 2e œuvre de Ray Bradbury, un recueil de nouvelles : *Les pommes d'or du soleil*.
- J'ai créé une étagère et un panier « Ray Bradbury » qui contiennent ces deux documents

Deux livres, c'est vraiment le strict minimum requis pour illustrer le concept d'une étagère ;-)

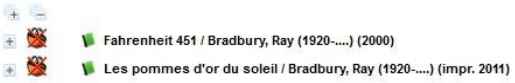
Article « L'auteur du mois » - Lié à l'étagère « Ray Bradbury »







Contenu du panier « Ray Bradbury » lié à cette étagère



Résultat final et conseils

Un « coup de cœur » pour un auteur
Article de type « Sélection » lié à une étagère

Techniquement, rien n'empêche la présence simultanée des deux cadres suivants dans la même page :

« Notices dans les mêmes catégories qu'un article »
« Notices d'une étagère associée à un article »

Toutefois, il vaut mieux éviter que les deux cadres ne soient activés en même temps, sinon on risque de voir deux fois les mêmes notices.

En effet, vous pourriez « instinctivement » mettre dans l'étagère des notices qui seraient apparues de toute façon dans l'autre cadre, car elles sont liées aux mêmes catégories que l'article.

Pour avoir <u>seulement</u> les notices dans les mêmes catégories qu'un article et être sûr que l'autre cadre n'apparaîtra pas pour un article précis

- Si cet article était par hasard initialement de type Sélection, supprimer d'abord le contenu éventuel du champ personnalisé lié au type précédent d'article (l'étagère). Il faut que l'ancien champ personnalisé ne contienne plus de valeur (choix vide).
- Choisir un (nouveau) type de contenu : sûrement pas Sélection ! N'importe quel autre type convient.
- Choisir ensuite les catégories adéquates

Pour avoir <u>seulement</u> les notices d'une étagère associée à un article et être sûr que l'autre cadre n'apparaîtra pas pour un article précis

- Si cet article était lié initialement à une ou plusieurs catégories, supprimer d'abord ces liens vers les descripteurs
- Si cet article était par hasard initialement d'un autre type que Sélection, supprimer d'abord le contenu éventuel du champ personnalisé lié au type précédent d'article. Il faut que l'ancien champ personnalisé ne contienne plus de valeur (choix vide).
- Choisir un (nouveau) type de contenu : Sélection
- Choisir ensuite l'étagère adéquate

Je rappelle que vous ne pouvez effacer le contenu d'un champ personnalisé via l'interface de PMB que si vous avez prévu un choix vide.

PAGE DU PORTAIL QUI AFFICHE UN ARTICLE

UN AUTRE MODE D'AFFICHAGE DU CADRE « NOTICES DANS LES MÊMES CATÉGORIES QU'UN ARTICLE »

La page qui affiche un article et une liste de notices comprend trois cadres, mais tous ne sont pas nécessairement affichés.

- **Un cadre issu du module Affichage d'un article**
  - Visible en toute circonstance – c'est le cadre principal de cette page
- **Carrousel de notices dans les mêmes catégories qu'un article** – Etagère créée de manière dynamique
  - Visible si certaines catégories sont communes à l'article et aux notices - Peu importe le type de contenu de l'article
- **Liste de notices - Une étagère thématique associée à un article** : ex. Nouveautés du rayon Fiction jeunesse
  - Visible seulement si l'article est de type Sélection et si l'étagère préalablement choisie contient des notices

Carrousel de notices dans les mêmes catégories qu'un article	Liste de notices d'une étagère associée à un article

Le « Carrousel de notices dans les mêmes catégories qu'un article » a été créé sur base du module « Liste de notices », donc 2 vues sont disponibles : un carrousel ou une liste (template Django).

Dans la démo, on a choisi le carrousel, mais on pourrait préférer une liste pour diverses raisons :

- Le carrousel ne montre que les vignettes et les titres, tandis **qu'avec une liste, on peut afficher davantage de champs, ce qui permet de comprendre pourquoi telle ou telle notice a été sélectionnée**
- Le choix des champs à afficher et de leur mode de présentation se gère facilement dans le **template de notice** associé à ce cadre (Portail – Liste de notices), on peut par exemple ajouter des champs.

N.B. Si nécessaire, relire « Carrousel de notices dans les mêmes catégories qu'un article »

L'objectif est de remplacer ce carrousel de notices dans les mêmes catégories qu'un article par une liste de notices dans les mêmes catégories qu'un article. Il faut donc changer le mode d'affichage et la position du cadre dans la page.

J'ai affiché cette liste de notices « sur le même sujet » de deux manières, afin de faciliter la comparaison et le choix

Bien entendu, en situation réelle, vous ne gardez qu'une seule option pour les notices sur le même sujet : le carrousel ou la liste !!

Techniquement, rien n'empêche la présence simultanée des deux cadres suivants dans la même page :

« Notices dans les mêmes catégories qu'un article »
« Notices d'une étagère associée à un article »

Toutefois, il vaut mieux éviter que les deux cadres ne soient activés en même temps, sinon on risque de voir deux fois les mêmes notices.

Si vous choisissez d'afficher les notices dans les mêmes catégories qu'un article sous forme de liste, il faut prendre toutes les précautions utiles afin que les notices d'une étagère associée à un article ne soient pas visibles.

Sinon vous auriez deux longues listes en partie similaires affichées l'une en-dessous de l'autre, et la page deviendrait « kilométrique » !

Première précaution : ne supprimez pas tout de suite le carrousel de notices sur le même sujet qu'un article !
Ni la liste de notices que vous pourriez créer ensuite !

Sûrement pas quand vous êtes en période de test, et même pas après les tests initiaux.
Vous pourriez changer d'avis plus tard, et il faudrait recréer ce cadre !!

Il est simple de masquer en permanence un cadre basé sur le module Liste de notices. Il suffit d'**indiquer la valeur 0 pour le nombre d'éléments à renvoyer**, et ce cadre ne sera jamais affiché.	Choisissez une source de donnée: Notices des catégories d'un article Choisissez un sélecteur pour la source de données: Sélecteur depuis une variable d'environnement Sélectionnez une variable d'environnement: Identifiant de l'article Nombres d'éléments maximum à renvoyer: 0
N.B. Je rappelle que cette astuce ne fonctionne pas pour les cadres basés sur le module Liste d'articles. Dans ce cas, la valeur zéro signifie : afficher tous les articles !	

Je ne vais pas détailler le paramétrage de ce cadre en version « liste », vu qu'il ressemble fort à la version « carrousel ».
Je vais seulement relever quelques différences. N.B. Si nécessaire, relire « Notices dans les mêmes catégorie qu'un article ».

Formulaire pour la variante du carrousel de notices dans les mêmes catégories qu'un article – Module « Liste de notices »	
Vue en carrousel	**Vue en liste de notices**
Modification de cadre - Liste de notices : Notices dans la même catégorie qu'un article Nom du Cadre: Notices dans la même Cadre fixe ? / Attaché à la page ? / Mode de mise en cache: Selon les GET,POST et vues OPAC Conditions d'affichage: Choisissez une condition d'affichage Choisissez une page: Page qui affiche un article et une liste de notices Choisissez une source de donnée: Notices des catégories d'un article Choisissez un sélecteur pour la source de données: Sélecteur depuis une variable d'environnement Sélectionnez une variable d'environnement: Identifiant de l'article Nombres d'éléments maximum à renvoyer: 15	Modification de cadre - Liste de notices : Notices dans les mêmes catégories (liste) Nom du Cadre: Notices dans les même Cadre fixe ? / Attaché à la page ? / Mode de mise en cache: Selon les GET,POST et vues OPAC Conditions d'affichage: Choisissez une condition d'affichage Choisissez une page: Page qui affiche un article et une liste de notices Choisissez une source de donnée: Notices des catégories d'un article Choisissez un sélecteur pour la source de données: Sélecteur depuis une variable d'environnement Sélectionnez une variable d'environnement: Identifiant de l'article Nombres d'éléments maximum à renvoyer: 15
Choisissez une vue: Vue en Carrousel Construction du lien vers une notice: Page qui affiche une notice Variable à utiliser: Identifiant de la notice Mode d'affichage: Vertical Temps de la transition (en ms): 3000 Temps de pause (en ms): 4000 Nombre d'éléments par page: 2 Nombre d'éléments déplacés: 2 Démarrage automatique: Oui / Non Pause au survol du curseur: Oui / Non Affichage des liens vers les éléments: Oui / Non Choisissez le template à utiliser: Choisissez le template à utiliser Contenu du template: records, records[i].title, records[i].vign, records[i].link, records[i].content, id Template de notice à utiliser pour le content: Template par défa CSS associée Choisissez le template à utiliser / Choisissez le template à utiliser / Notices du même auteur / Notices similaires / Notices sur la même étagère / Nouveautés en page d'accueil / Notices dans la même catégorie	Choisissez une vue: Vue Django Construction du lien vers une notice: Page qui affiche une notice Variable à utiliser: Identifiant de la notice Choisissez le template à utiliser: Choisissez le template à utiliser Contenu du template: title, records, records[i].id, records[i].header, records[i].content, records[i].link Choix du template: Portail - liste de notices Choisissez le template à utiliser / Choisissez le template à utiliser / Liste de notices Vu que la liste de notices (vue Django) avait été conçue au départ pour les nouveautés des divers rayons, PMB ne propose qu'un seul template.
Différences entre ces deux formulaires	
Il faut paramétrer le carrousel dans ce cas-ci. On dispose dès le départ d'un template prévu pour les Notices dans les mêmes catégories qu'un article.	Le paramétrage du carrousel a bien sûr disparu. Il faut changer le titre dans le template, et remplacer « Toutes les nouveautés » par « Quelques suggestions », puisque ce cadre est devenu polyvalent (nouveautés + étagère thématique). Voir Autre usage du cadre « Liste de notices liées à une étagère »

Après sauvegarde du cadre « Notices dans les mêmes catégories qu'un article », il apparaît en 2e position dans la page qui affiche un article, entre « Afficher un article » et « Liste de notices – Etagère associée à un article ».
Vous pouvez éventuellement le déplacer, afin qu'il soit le dernier de la page.
De toute façon, qu'il soit en 2e ou 3e position n'a guère d'importance, puisqu'on voit soit le cadre n° 2, soit le cadre n° 3.

Résultat final et conseils

N.B. Je sais que je répète les conseils donnés dans l'analyse de la variante précédente :
un autre usage du cadre « Liste de notices – Etagère associée à un article ».
Mais je ne sais pas si vous lirez ces pages dans l'ordre où elles se présentent.
Je ne suis donc pas sûre que vous ayez déjà lu cet avertissement.

Article de type « Coup de cœur »
Notices dans les mêmes catégories que l'article
Affichage en liste (et non en carrousel)

Techniquement, rien n'empêche la présence simultanée des deux cadres suivants dans la même page :
« Notices dans les mêmes catégories qu'un article »
« Notices d'une étagère associée à un article »

Toutefois, il vaut mieux éviter que les deux cadres ne soient activés en même temps, sinon on risque de voir deux fois les mêmes notices.

En effet, vous pourriez « instinctivement » mettre dans l'étagère des notices qui seraient apparues de toute façon dans l'autre cadre, car elles sont liées aux mêmes catégories que l'article.

Le problème serait encore plus crucial si les 2 cadres affichaient les notices sous forme de liste.
Vous auriez alors 2 longues listes en partie similaires l'une en-dessous de l'autre, et la page deviendrait « kilométrique » !

Pour avoir seulement les notices dans les mêmes catégories qu'un article et être sûr que l'autre cadre n'apparaîtra pas pour un article précis

- Si cet article était par hasard initialement de type Sélection, supprimer d'abord le contenu éventuel du champ personnalisé lié au type précédent d'article (l'étagère). Il faut que l'ancien champ personnalisé ne contienne plus de valeur (choix vide).
- Choisir un (nouveau) type de contenu : sûrement pas Sélection ! N'importe quel autre type convient.
- Choisir ensuite les catégories adéquates

Pour avoir seulement les notices d'une étagère associée à un article et être sûr que l'autre cadre n'apparaîtra pas pour un article précis

- Si cet article était lié initialement à une ou plusieurs catégories, supprimer d'abord ces liens vers les descripteurs
- Si cet article était par hasard initialement d'un autre type que Sélection, supprimer d'abord le contenu éventuel du champ personnalisé lié au type précédent d'article. Il faut que l'ancien champ personnalisé ne contienne plus de valeur (choix vide).
- Choisir un (nouveau) type de contenu : Sélection
- Choisir ensuite l'étagère adéquate

Faites des tests comparatifs entre ces divers modes d'affichage dès que possible, évitez de changer trop tard de vue ! Surtout si vous devez modifier le type de contenu de certains articles, et les champs personnalisés qui y sont liés.

Sinon, vous risquez de conserver les traces des valeurs antérieures de ces champs personnalisés « quelque part » dans les tables, et de ne pas obtenir la mise en page souhaitée.

Je rappelle que vous ne pouvez effacer le contenu d'un champ personnalisé via l'interface de PMB que si vous avez prévu un choix vide. Si le choix vide n'existe pas, il n'y a que 2 manières de supprimer le contenu d'un champ personnalisé :

- passer par l'interface de **PhpMyAdmin**
- rédiger une **requête SQL** spécifique et à usage unique, adaptée au champ personnalisé que vous avez créé

Soyez donc prudent !

Liste de sites Web

Le site d'une bibliothèque scolaire ou académique présente souvent des ressources externes en ligne : des références de base (encyclopédies, dictionnaires généraux ou spécifiques), des sites bibliographiques, des aides méthodologiques, etc.

Le portail permet d'afficher ces ressources autrement que sous la forme d'une « simple liste ».

Divers modes de présentation sont possibles, en fonction du nombre de sites à cataloguer, de leur importance, et de leur préexistence éventuelle dans la base (sous forme de notices – type « document électronique »).

Puisqu'il s'agit de ressources externes à la bibliothèque, il vaut mieux (en termes d'ergonomie) que ces sites s'ouvrent dans un autre onglet du navigateur.

Hypothèse n° 1 : un article du portail contient un ou plusieurs liens vers des documents externes

S'il s'agit de liens occasionnels, sans connexion directe avec votre institution ou avec les spécificités de votre bibliothèque, ces sites ne seront probablement pas catalogués : employez donc l'éditeur HTML pour gérer ces URL.

Rappel – Mode d'emploi de l'éditeur html intégré au portail

Comment insérer ou supprimer un lien et spécifier si l'URL doit s'ouvrir dans un autre onglet du navigateur

Sélectionner un ou plusieurs mots - Cliquer sur l'**icône « Lien »**

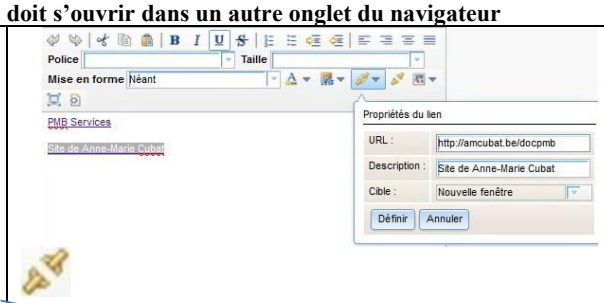
Saisir **l'URL de destination** et une brève description

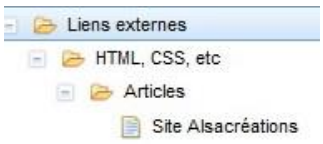
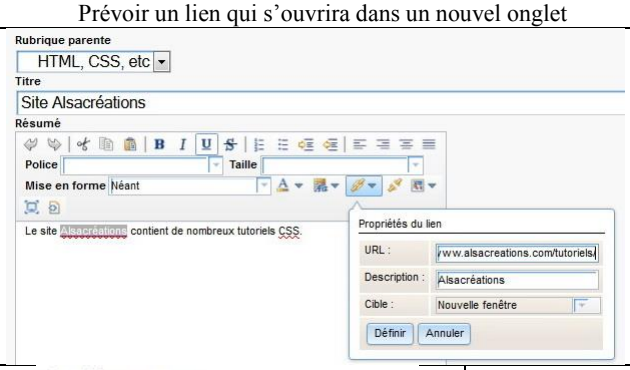
Choisir la cible :

- *Fenêtre actuelle* pour un lien vers une page de votre portail
- *Nouvelle fenêtre* pour un lien vers un autre site (le lien s'ouvrira ainsi dans un autre onglet du navigateur)

Cliquer sur « Définir » pour créer et enregistrer ce lien.

Pour supprimer un lien URL : positionner le curseur quelque part dans le texte sélectionné, et cliquer sur l'icône à droite de « Lien »



Arborescence	Prévoir un lien qui s'ouvrira dans un nouvel onglet	Le lien est cliquable
		
Cet article comprend 4 liens Un lien vers la section Tutoriels 3 liens vers des tutoriels CSS		Le site s'ouvre en effet dans un nouvel onglet

Hypothèse n° 2 : un article du portail décrit un site externe de manière relativement détaillée

Si vous proposez déjà une liste assez longue de sites dans votre catalogue (par exemple dans les infopages ou via les catégories), vous n'aurez sûrement pas le temps de créer un article par site ! En outre, cela ne se justifierait probablement pas pour un certain nombre de sites moins intéressants. Dernier obstacle : cela créerait un nombre anormalement élevé d'articles et de rubriques (pour les classer), et cela alourdirait donc la gestion du contenu éditorial.

Toutefois, certains sites de ressources méritent d'être mis à l'honneur ! Autant leur consacrer un article …

Puisqu'il s'agit de ressources externes à la bibliothèque, il vaut mieux que ces sites s'ouvrent dans un autre onglet du navigateur : c'est pourquoi **il faut créer un nouveau type de contenu pour les articles**.

Ce nouveau type de contenu (appelé « Lien ») relie un article et l'URL d'une page Web, et permet aussi de choisir si cette autre page s'ouvrira dans un autre onglet du navigateur ou non. 2 champs personnalisés sont dès lors nécessaires.

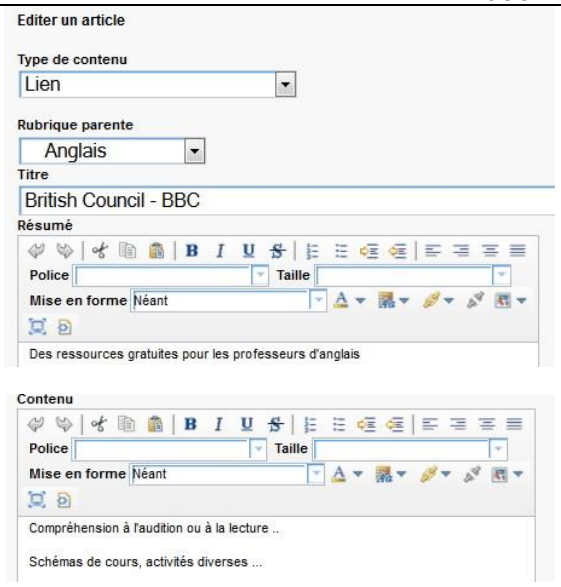
Définition des champs pour le type de contenu "Lien"

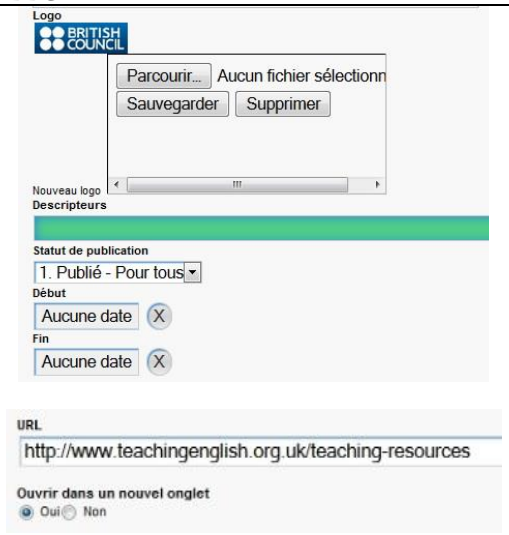
	Nom du champ	Titre à l'affichage	Type de saisie	Type de données
- +	url	URL	Texte	Texte
- +	nouvel_onglet	Ouvrir dans un nouvel onglet	Liste de choix	Entier

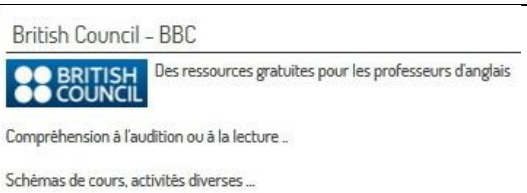
Type de contenu : lien		1er champ personnalisé : URL
Edition du champ url Nom du champ: url Titre à l'affichage: URL Type de saisie: Texte Options Texte Liste de choix Liste de choix à partir d'une requête Valeurs à partir d'un sélecteur d'autorités Sélection d'une date Zone de commentaire Sélecteur externe URL Résolveur de lien Zone de saisie HTML Annuler Enregistrer	**Cliquer sur « Options » pour paramétrer le type de saisie.** Options pour le paramètre Texte Taille du champ à l'écran: 80 Nombre maximum de caractères à saisir: 255 Répétable Html autorisé Enregistrer Ne pas oublier d'enregistrer ensuite !	**Type de données** Edition du champ url Nom du champ: url Titre à l'affichage: URL Type de saisie: Texte Options Type de données: Texte Texte Texte large Entier Date Nombre à virgule 100 Annuler Enregistrer
Le type de saisie « texte » signifie « caractères saisis au clavier ». **Plusieurs types de données sont donc possibles.**		**Le type de données « texte »** peut contenir tout type de caractère : lettres, chiffres ou divers symboles (mathématiques, de ponctuation)

Type de contenu : lien		2e champ personnalisé : ouvrir dans un nouvel onglet
Cliquer sur « Options » pour paramétrer le type de saisie. Edition du champ nouvel_onglet Nom du champ: nouvel_onglet Titre à l'affichage: Ouvrir dans un nouvel onglet Type de saisie: Liste de choix Options Texte Liste de choix Liste de choix à partir d'une requête Valeurs à partir d'un sélecteur d'autorités Sélection d'une date Zone de commentaire Sélecteur externe URL Résolveur de lien Zone de saisie HTML Annuler Enregistrer	Liste de choix Liste multiple Affichage sous forme d'autorité Affichage sous forme de cases à cocher ou boutons radio: Nombre de choix par ligne : Numérotation automatique Choix vide: Valeur : Libellé : Choix par défaut: Valeur : Liste des options de la liste Valeur / Libellé / Ordre 1 / Oui / 0 2 / Non / 0	**Type de données** Edition du champ nouvel_onglet Nom du champ: nouvel_onglet Titre à l'affichage: Ouvrir dans un nouvel onglet Type de saisie: Liste de choix Options Type de données: Entier Texte Texte large Entier Date Nombre à virgule 100 Annuler Enregistrer
Le type de saisie « liste de choix » se base sur une liste préétablie. Chaque choix possible (ici, « oui ou non ») correspond à une valeur (ici, « 1 ou 2 »). **Plusieurs types de données sont donc possibles.**		**Le type de données est un « entier »** car le type correspond à la valeur choisie : des nombres (1,2). Mais on aurait aussi pu choisir le type « texte » pour enregistrer ces nombres. Si on avait choisi des lettres (« o » ou « n »), le type aurait été texte.

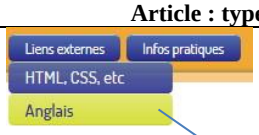
Article « British Council - BBC »





Arborescence	Rubrique « Anglais »	Article « Birtish Council – BBC »
		

Article : type de contenu = Lien URL

Point de départ **Menu principal** Clic sur Anglais (id_rubrique = 25)		**C'est un beau schéma théorique de navigation ;-)** **Mais au stade actuel, rien ne se passe dès qu'on arrive dans la page qui affiche un article …** Certes, on a prévu un nouveau type de contenu, mais vu que le template Django n'en tient pas encore compte, il est impossible de quitter le portail. Aucun élément (logo ou titre) n'est cliquable, et la variable qui contient l'URL de destination ne sert provisoirement à rien. **Il faut adapter le template Django du cadre « Afficher un article ».** *Sinon, le site du British Council resterait inaccessible !!*
On va **vers** la page qui affiche **une rubrique**	http://...&pageid=6&id_rubrique=25	
Page → une rubrique **(pageid = 6)** Clic sur un article (British Council) (id_article = 36)		
On va **vers** la page qui affiche **un article**	http://...&pageid=4&id_article=36	
Page → un article **(pageid = 4)** Type de contenu : Lien URL		
On peut ouvrir le lien dans un nouvel onglet		

N.B. Je vous renvoie si nécessaire au commentaire du cadre « Afficher un article », car je ne vais pas le reprendre de manière détaillée. En effet, la seule chose qui change, c'est le template Django.

Template Django – Afficher le contenu d'un article

Ancienne version

```
<h3>

 S'il s'agit d'un événement sur une période précise
{% if fields_type.date_debut.values.0.value %}

  {% if fields_type.date_fin.values.0.value %}
      Du {{fields_type.date_debut.values.0.format_value}}
      au {{fields_type.date_fin.values.0.format_value}} :

   {% else %}
      Le {{fields_type.date_debut.values.0.format_value}} :
  {% endif %}

{% endif%}

S'il s'agit d'un événement à une date précise

{% if fields_type.date.values.0.value %}
      Le {{fields_type.date.values.0.format_value}} :
{% endif%}

Quel que soit le type de contenu de l'article
{{title}}

</h3>

<img style="float:left;margin-right:5px;max-height:200px;max-
width:200px;" src='{{logo.large}}' />

<p>{{resume}}</p>
<br />

<div class="row">{{content}}</div>
<br />

S'il s'agit d'un article « Coup de cœur » (lié à une notice)

 {% if fields_type.identifiant_notice.values.0.value %}
     <a href="./index.php?lvl=notice_display&id=
        {{fields_type.identifiant_notice.values.0.value}}">
        Voir le document dans le catalogue
     </a>
{% endif%}
```

Nouvelle version

```
<h3>

S'il s'agit d'un événement sur une période précise
{% if fields_type.date_debut.values.0.value %}

   % if fields_type.date_fin.values.0.value %}
      Du {{fields_type.date_debut.values.0.format_value}}
      au {{fields_type.date_fin.values.0.format_value}} :

   {% else %}
      Le {{fields_type.date_debut.values.0.format_value}} :
  {% endif %}

{% endif%}

S'il s'agit d'un événement à une date précise
{% if fields_type.date.values.0.value %}
      Le {{fields_type.date.values.0.format_value}} :
{% endif%}

Si le champ URL a une valeur (type de contenu = lien URL)
 {% if fields_type.url.values.0.value %}
    <a href="{{fields_type.url.values.0.value}}"
      {% if fields_type.nouvel_onglet.values.0.value==1 %}
          target="_blank"
 {% endif %}>
  {{title}}</a>

 {% else %}
   {{title}}

{% endif %}

Quel que soit le type de contenu de l'article


</h3>

<img style="float:left;margin-right:5px;max-height:200px;max-width:200px;"
src='{{logo.large}}' />

<p>{{resume}}</p>
<br />

<div class="row">{{content}}</div>
<br />

S'il s'agit d'un article « Coup de cœur » (lié à une notice)

{% if fields_type.identifiant_notice.values.0.value %}
     <a href="./index.php?lvl=notice_display&id=
        {{fields_type.identifiant_notice.values.0.value}}">
        Voir le document dans le catalogue
     </a>
{% endif%}
```

Template Django – Afficher le contenu d'un article – Nouvelle version

```
<h3>
S'il s'agit d'un événement sur une période précise
{% if fields_type.date_debut.values.0.value %}
   % if fields_type.date_fin.values.0.value %}
      Du {{fields_type.date_debut.values.0.format_value}}
      au {{fields_type.date_fin.values.0.format_value}} :
   {% else %}
      Le {{fields_type.date_debut.values.0.format_value}} :
   {% endif %}
{% endif%}

S'il s'agit d'un événement à une date précise
{% if fields_type.date.values.0.value %}
      Le {{fields_type.date.values.0.format_value}} :
{% endif%}

Si le type de contenu est un lien URL
 {% if fields_type.url.values.0.value %}
    <a href="{{fields_type.url.values.0.value}}"
      {% if fields_type.nouvel_onglet.values.0.value==1 %}
            target="_blank"
 {% endif %}>
  {{title}}</a>

 {% else %}
   {{title}}

{% endif %}

Quel que soit le type de contenu de l'article

</h3>
<img style="float:left;margin-right:5px;max-height:200px;max-width:200px;"
src='{{logo.large}}' />

<p>{{resume}}</p><br />

<div class="row">{{content}}</div>
<br />

S'il s'agit d'un article « Coup de cœur » (lié à une notice)

{% if fields_type.identifiant_notice.values.0.value %}
      <a href="./index.php?lvl=notice_display&id=
         {{fields_type.identifiant_notice.values.0.value}}">
         Voir le document dans le catalogue
      </a>
{% endif%}
```

S'il s'agit d'un événement sur une période précise

Début de la 1e condition
Si **le champ date_debut** a une valeur
{% if fields_type.date_debut.values.0.value %}

Début de la 2e condition
Si **le champ date_fin** a une valeur
{% if fields_type.date_fin.values.0.value %}
On affiche : **Du** {{date_debut}} **au** {{date_fin}}
Sinon {% else %} (s'il n'y a pas de date de fin)
On affiche : **Le** {{date_debut}}
S'il n'y a pas de date de fin, on considère que l'événement a lieu à une date précise (celle du début), et on modifie le message
Fin de la 2e condition {% endif%}
Fin de la 1e condition {% endif%}

S'il s'agit d'un événement à une date précise

Début de la condition
Si **le champ date** a une valeur
{% if fields_type.date.values.0.value %}
On affiche : **Le** {{date}}
Fin de la condition {% endif%}

Si le type de contenu est un lien URL

Faute de place, voir explications détaillées page suivante !

Quel que soit le type de contenu de l'article, on affiche

le logo de l'article	
le résumé de l'article	{{resume}} (le chapeau)
le contenu de l'article	{{content}} (le texte complet)

S'il s'agit d'un article « Coup de cœur » (lié à une notice)

Début de la condition
Si **le champ identifiant_notice** a une valeur
{% if fields_type.identifiant_notice.values.0.value %}
On génère une balise
Le 1er élément est l'URL de destination (une adresse OPAC)
Le 2e élément est la source = ce qui est cliquable
Fin de la condition {% endif%}

Avant tout, une petite leçon – ou une petite révision ? – de HTML ! Comment ouvrir une URL dans un nouvel onglet ?

Comment rendre un texte cliquable et ouvrir l'URL de destination dans un autre onglet du navigateur ?	
<a href="… : URL de destination	**Texte entre <a> et : source, d'où on part : on clique sur le texte**
Dans la partie qui contient l'URL de destination, ajouter l'instruction target="_blank" Site de Anne-Marie Cubat devient Site de Anne-Marie Cubat	

Comment rendre une image cliquable et ouvrir l'URL de destination dans un autre onglet du navigateur ?	
<a href="… : URL de destination	**<img src="… : source, d'où on part : on clique sur l'image**
Dans la partie qui contient l'URL de destination, ajouter l'instruction target="_blank" devient 	

Template Django – Afficher le contenu d'un article – Lien URL

```
{% if fields_type.url.values.0.value %}

  <a href="{{fields_type.url.values.0.value}}"

 {% if fields_type.nouvel_onglet.values.0.value==1 %}
        target="_blank"
  {% endif %}>

{{title}} </a>

 {% else %}
  {{title}}

{% endif %}
```

Par convention, dans les templates Django :

On emploie un double signe = pour vérifier si 2 valeurs sont égales.
{% if fields_type.nouvel_onglet.values.0.value==1 %}

On emploie les signes != pour vérifier si 2 valeurs sont différentes.
{% if fields_type.nouvel_onglet.values.0.value !=1 %}

Début de la 1e condition
Si le champ URL a une valeur {% if fields_type.date.values.0.value %}
On génère une balise <a href="…"
URL de destination = le contenu de la variable {{fields_type.url.values.0.value}}

Début de la 2e condition
Si le champ nouvel_onglet a la valeur 1
{% if fields_type.nouvel_onglet.values.0.value==1 %}
On ajoute target="_blank" à l'intérieur de la balise
Fin de la 2e condition {% endif%}

On ajoute > pour fermer la balise
Ce symbole > doit se trouver en-dehors de la 2e condition
En effet, il faut fermer cette balise, peu importe la valeur du champ nouvel_onglet

On affiche le titre, qui sera donc cliquable dans ce cas-ci
En effet, la balise qui suit ce titre marque la fin de ce qui est cliquable

Sinon = Si le champ URL n'a pas de valeur
= Si l'article a un autre type de contenu ou si on a oublié de compléter ce champ
On affiche le titre, qui ne sera donc pas cliquable
car il est externe à la balise <a href=" …">

Fin de la 1e condition {% endif%}

Hypothèse n° 3 : ces sites externes sont déjà catalogués dans votre base

Il faut trouver un système pratique pour regrouper ces notices et les présenter de manière structurée.
Choisissez des méthodes aussi peu chronophages que possible !

Solution n° 1 :
- regrouper ces notices de sites Web dans un certain nombre d'étagères virtuelles
- lier chaque étagère à un article de type sélection qui présente ce sous-ensemble de sites

Vous avez déjà dans le portail de PMB tous les éléments dont vous avez besoin pour mettre cette solution en œuvre, et en outre, je vous ai donné des « tuyaux » dans le chapitre précédent. En effet, j'ai expliqué comment employer le cadre « Liste de notices – Etagère associée à un article » à un autre usage que celui prévu initialement.

Tenez quand même compte des aspects suivants
- s'il y beaucoup de sites catalogués dans votre base, cela peut augmenter le nombre d'articles, de rubriques et d'étagères
- il faudra vérifier de temps en temps le contenu des étagères virtuelles, ajouter des notices dans les paniers ou en supprimer

Solution n° 2 :
- prévoir des catégories bien structurées, et lier ces notices de sites Web aux « bons » descripteurs
- prévoir pour chaque sous-ensemble de sites un article de présentation (n'importe quel type de contenu sauf sélection !)
- lier chaque article aux « bonnes » catégories, afin que la liste affichée corresponde à vos critères de sélection

Vous avez déjà dans le portail de PMB tous les éléments dont vous avez besoin pour mettre cette solution en œuvre, et en outre, je vous ai donné des « tuyaux » dans le chapitre précédent. En effet, j'ai expliqué comment employer le cadre « Notices dans les mêmes catégories qu'un article » à un autre usage que celui prévu initialement.

Tenez quand même compte des aspects suivants
- s'il y beaucoup de sites catalogués dans votre base, et/ou des sites dans des domaines fort diversifiés, cela peut augmenter le nombre d'articles et de catégories
- il faudra vérifier de temps en temps les liens vers les descripteurs, ajouter ou supprimer des catégories

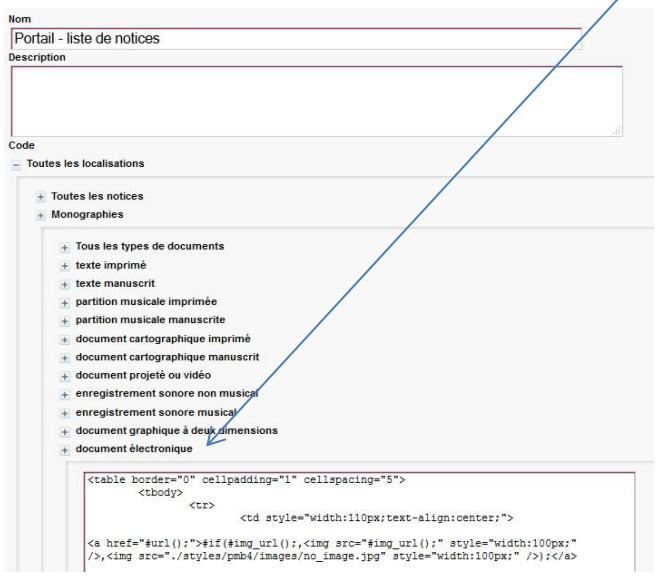
Il n'y a pas de panacée, pas de solution-miracle unique, valable en toute circonstance !
Il y a de grandes chances que vous soyez confronté aux 3 hypothèses que j'ai évoquées, et vous allez probablement combiner les diverses solutions suggérées ici, et même en trouver d'autres ;-)

Adaptation du template de notices (onglet Editions)

Lorsque j'avais analysé le cadre « Liste de notices – Etagère associée à un article » dans sa version originale, j'avais commenté le template de notice qui gère l'affichage.

Il faut compléter ce template afin qu'il affiche correctement les documents électroniques.

Editions > Templates de notices



Template adapté pour les sites Web (documents électroniques)

```
<table border="0" cellpadding="1" cellspacing="5">
        <tbody>
                <tr>
                        <td style="width:110px;text-align:center;">

<a href="#url();">#if(#img_url();,<img src="#img_url();"
style="width:100px;" />,<img src="./styles/pmb4/images/no_image.jpg"
style="width:100px;" />);</a>

                </td>
                        <td style="text-align:left;">

<a href="#url();"><font size=3>#title();#a_empty(#complement_title();, :
);</font></a>

#str_replace(  , ,#a_empty(#authors_by_type_with_tpl(1,3,\, ,\,
,0,0,70\,71\,72,{rejete} {name});,<br />de ););

#a_empty(#resume();,<br /><br />);

                </td>
                        </tr>
        </tbody>
</table>
```

Ce chapitre consacré aux sites Web se termine de manière impromptue, et j'en suis consciente.

J'aurais voulu ajouter ici quelques exemples, afin d'illustrer ces diverses options. Les exemples sont déjà conçus dans ma base de test, mais j'en suis à peu près au stade zéro en ce qui concerne les captures d'écran et les explications.

La situation est analogue pour le chapitre suivant, le menu créé dans l'interface d'administration : c'est un brouillon impubliable, dans l'état actuel des choses.

Un peu de patience, cela viendra bientôt … et merci de votre compréhension ;-)

Menu créé dans l'interface d'administration

TRANSFERT DU PORTAIL VERS LA VRAIE BASE

Préparatifs à effectuer dans la vraie base

N'attendez pas la dernière minute ! Il y a déjà un certain nombre de choses que vous pouvez faire avant le jour J.

- si nécessaire, changer de feuille de style dès que possible, cela vous laissera le temps de la tester à l'aise, et vos lecteurs s'y accoutumeront
- importer dans la base certaines requêtes SQL liées spécifiquement au portail, et qui ne sont disponibles pour le moment que dans votre base de test

- si le contexte le permet, prévoir une journée pour le transfert du portail et les tests, et annoncer aux lecteurs qu'il sera malheureusement impossible de consulter le catalogue tel ou tel jour
- choisir cette date judicieusement – pas à une période de grand stress comme la rentrée scolaire !

- sans vouloir être exagérément pessimiste, il vaut mieux prévoir une possibilité de faire marche arrière
- ne pas supprimer les étagères et paniers employés actuellement dans la base, ni en gestion, ni pour l'OPAC
- si le passage au portail s'avérait plus compliqué que prévu, il faut que vous puissiez retourner à la version antérieure de l'OPAC
- ne pas employer les anciennes étagères pour le portail, créer de *nouvelles* étagères et les paniers correspondants
- vous pouvez déjà commencer à remplir certaines de ces étagères
- avant le transfert, faire une copie de certaines tables, comme celle des paramètres, les infopages, la liste des paniers et étagères et leur contenu, afin de pouvoir les restaurer au cas où …

- compléter cette liste-type à l'aide du journal de bord que vous avez tenu à jour, et dans lequel vous avez sûrement noté la liste des changements à faire ultérieurement ;-)

Eléments à transférer depuis la base de travail

Au début du processus, il y a déjà eu un transfert depuis la vraie base vers la base de test.

Le moment est venu de transférer dans l'autre sens.

- Si vous changez de feuille de style en dernière minute (contrairement à ce que je venais de conseiller …), n'oubliez pas de copier la nouvelle feuille de style au bon endroit, et de modifier le paramétrage de l'OPAC (default_style)
- le transfert concerne bien entendu l'ensemble des tables liées au portail, toutes celles dont le nom commence par cms (le jeu de sauvegarde que j'ai appelé « cms »)
- si nécessaire, relire Organiser les sauvegardes
- normalement, votre base de test ne comprend qu'un nombre réduit de notices, et tout le contenu bibliographique est déjà présent dans la vraie base. Vérifiez malgré tout.
- vous avez peut-être créé de nouvelles notices dans la base de travail, et associé certaines à des documents numériques enregistrés dans des répertoires d'upload
- en ce qui concerne les templates de notice, cela dépend. Les tables liées aux templates contiennent tous les templates qu'on a définis, et pas uniquement celui dont on a parlé dans ce manuel
    - Si vous pouvez transférer l'intégralité, ce sera plus simple
    - Sinon, il faudra recréer dans la vraie base le template employé dans la base de test. N'oubliez pas qu'il comporte plusieurs parties (monographies, périodiques, articles de périodiques et notices de bulletins)

Autres ajustements nécessaires

Il faut maintenant tester l'ensemble

- modifier le paramétrage spécifique du portail : cache_ttl, URL de base, cms, first_page, params, etc
- si nécessaire, relire Paramétrage du portail, paramètres cms et first_page_params
- se balader un peu partout dans le portail, et voir si tout fonctionne correctement
- modifier certains numéros identifiants, par exemple pour les articles de type « Coup de cœur ».
- ils sont liés au numéro d'id d'une notice spécifique, et ce numéro n'est sûrement pas le même dans la base de travail et dans la base en production
- vérifier le résultat obtenu avec certains carrousels dynamiques dans une base complète : ils ne vont peut-être pas afficher les notices que vous voudriez mettre en évidence (ex. notices dans les mêmes catégories qu'un article)
- remplacer peut-être certains carrousels ou listes dynamiques par des listes ou carrousels « statiques » = basés sur une étagère sélectionnée manuellement

Si par malheur les tests n'étaient pas vraiment concluants

- ne pas se décourager, les débuts sont parfois difficiles !
- reprendre les choses calmement, travailler en équipe

Astuce pour retourner à l'OPAC traditionnel

- si vous avez bien conservé les étagères et paniers de l'OPAC, les infopages, etc, une marche arrière est possible, même si la « machine à remonter le temps » connaît parfois quelques défaillances !
- si vous videz le contenu du paramètre first_page_params, les lecteurs verront à nouveau l'OPAC classique
- mais certains éléments du portail resteront visibles, je pense par exemple aux cadres affichés dans les zones fixes comme le bandeau de gauche, l'introduction (la zone supérieure) ou le pied de page
- on continuera donc à voir le menu ou la barre de recherche rapide, il faudra essayer de les masquer

CONCLUSION

Nous voilà arrivés au terme de ce manuel. Le sujet est complexe, mais j'ai essayé de le présenter aussi clairement que possible, même si je sais que je n'en maitrise pas tous les aspects.

J'espère que ce tutoriel vous aura en tout cas appris comment « piloter votre imprimante 3D » afin de construire un portail.

Bien sûr, ce n'est pas un « thriller » palpitant qui va vous tenir en haleine, ni votre futur livre de chevet, mais je vous souhaite malgré tout une agréable lecture et beaucoup de plaisir lorsque vous manipulerez votre « jeu de construction ».

Cordialement,

Anne-Marie Cubat

BIBLIOGRAPHIE

Si j'avais uniquement mentionné les documents cités dans ce manuel, vous n'auriez eu que deux titres !

Premiers pas avec le portail de PMB (en ligne)
Bénard, Camille, Portail PMB. *Introduction aux concepts du portail et à l'utilisation du contenu éditorial*, 2014

J'ai donc complété cette « mini-mini-liste » de départ, afin de vous donner d'autres références utiles ;-)
En outre, j'ai opté pour une présentation thématique de la bibliographie.

Logiciel PMB –	
Infos sur PMB en général Lien pour le téléchargement de PMB	http://www.sigb.net/index.php?lvl=cmspage&pageid=2&id_logiciel=18 http://forge.sigb.net/redmine/projects/pmb/files Versions employées pour ce manuel : 4.0.13, 4.1.4 et 4.1.5
Le portail de PMB	
Premiers pas avec le portail de PMB	http://doc.sigb.net/portail_pmb/co/guide%20web%20portail.html
Bénard, Camille, Portail PMB. *Introduction aux concepts du portail et à l'utilisation du contenu éditorial* – Support non finalisé, 2014 (23 p.) Document non publié	
Guide en ligne de PMB	
Guide en ligne de PMB Templates de notices Fonctions utilisables dans les templates	http://doc.sigb.net/pmb/co/guide_complet_web_1.html http://doc.sigb.net/pmb/co/edit_tpl_notice.html http://doc.sigb.net/pmb/co/edit_tpl_notice_fonctions.html
PMB Services	
Site de PMB Services	http://www.sigb.net/index.php

Informations sur PMB	
Mon site personnel : http://amcubat.be/docpmb/	
Installation de PMB (Windows) Personnaliser PMB – Logos – Feuilles de style Afficher les vignettes des couvertures Requêtes SQL à importer dans PMB	http://amcubat.be/docpmb/pmb-installation-windows-easyphp121 http://amcubat.be/docpmb/personnaliser-pmb-nom-logo-feuilles-style http://amcubat.be/docpmb/afficher-vignettes-couvertures http://amcubat.be/docpmb/requetes-sql/telecharger-et-importer

HTML – CSS	
Site de Alsacréations : http://www.alsacreations.com/	
Nombreux tutoriels disponibles sur ce site Initiation au positionnement CSS – partie 1 Initiation au positionnement CSS – partie 2 Guide de survie du positionnement CSS	http://www.alsacreations.com/tutoriels/ http://www.alsacreations.com/article/lire/533-initiation-au-positionnement-en-css-partie-1.html http://www.alsacreations.com/tuto/lire/608-initiation-au-positionnement-css-partie-2.html http://www.alsacreations.com/article/lire/53-guide-de-survie-du-positionnement-css.html
Site de w3scchools : http://www.w3schools.com/default.asp	
Tutoriels HTML Tutoriels CSS	http://www.w3schools.com/html/default.asp http://www.w3schools.com/css/default.asp
Codes-couleurs en HTML	
Guide des codes-couleurs en HTML	http://html-color-codes.info/Codes-couleur-HTML/

PHP - MySQL – Templates Django	
PHP	
Langage PHP – Références	http://php.net/manual/fr/langref.php
MySQL	
SQL 5.5 Tutoriels SQL	http://dev.mysql.com/doc/refman/5.5/en/index.html http://www.w3schools.com/sql/default.asp
Templates Django	
Templates Django H2O	https://docs.djangoproject.com/en/1.8/ref/templates/builtins/

AIDE-MÉMOIRE ET TABLEAUX DE SYNTHÈSE

Mon but n'est pas d'augmenter artificiellement le nombre de pages, et je sais que je n'ajoute ici que peu d'infos nouvelles par rapport à ce que j'ai déjà dit. Toutefois, j'ai décidé de fournir un résumé schématique sur certains sujets fondamentaux :

Les **liens entre les modules, cadres et pages** du portail
Les **templates Django**
Les balises **HTML**
Les **requêtes SQL**

En effet, les commentaires sur ces thèmes sont disséminés un peu partout, c'est pourquoi j'ai préféré regrouper les infos essentielles dans une synthèse. Ainsi, vous ne devrez pas parcourir plusieurs chapitres à la recherche de l'une ou l'autre explication ;-)

Liens entre modules, cadres et pages

LISTE DES MODULES DISPONIBLES

J'ai regroupé ces modules en diverses catégories, afin de donner une vue d'ensemble.
Les chapitres Modules, cadres et pages **et** Exemples supplémentaires **contiennent des explications détaillées sur de nombreux modules.**

PMB 4.0.13

PMB 4.1

Certains modules sont fondamentaux, car ils génèrent le cadre principal d'une des pages-outils du portail ; ils sont en général liés à une variable d'environnement

- Module rubrique → **variable id_rubrique** → page qui affiche **une rubrique**
- Module article → **variable id_article** → page qui affiche **un article**
- Module notice → **variable id_notice** → page qui affiche **une notice**
- Module agenda → **variable id_date** → page qui affiche **une liste d'événements**
- Module liste d'articles → **pas de variable** → page qui affiche le **résultat d'une recherche** dans le contenu éditorial : **une liste d'articles**

Listes présentées de manière « classique » ou en carrousel

- liste d'articles
- liste de notices
- liste de rubriques

} au choix : affichage sous forme de liste ou de carrousel (défilement horizontal, vertical, en fondu enchaîné)

- liste de bannettes
- liste d'étagères

} un seul mode d'affichage : la liste

- carrousel de notices → un seul mode d'affichage : le carrousel

Aide à la navigation

- menu créé sur base du contenu éditorial ou dans l'interface d'administration
- fil d'Ariane (fournir des liens en arrière sur les pages déjà parcourues)

Aide à la recherche interne ou externe

- recherche interne (dans le contenu bibliographique et/ou éditorial)
- recherche externe : interrogation SPARQL (accès au Web sémantique)

Insertion directe

- agenda (calendrier des activités organisées par la bibliothèque)
- code HTML (ajouter un cadre qui contient du code HTML)
- éléments de l'OPAC (ex. infopages, navigateur de périodiques)

Insertion d'éléments externes à la base

- flux RSS
- portfolio (gérer une collection de documents numériques non catalogués)

5 modules doivent être configurés avant usage

- **Agenda**
  - Définir les calendriers
- **Interrogation SPARQL (recherche externe)**
  - Définir les sources de données
- **Menu : seulement s'il s'agit du menu créé dans l'interface d'administration**
  - Inutile pour un menu basé sur le contenu éditorial (rubriques) : on peut l'employer tout de suite
- **Portfolio**
  - Paramétrer un répertoire d'upload (onglet Administration) et définir les Collections numériques (onglet Portail)
- **Recherche interne**
  - Définir les types de recherche (contenu éditorial et bibliographique) et les pages de destination

INTERCONNEXION ENTRE LA PAGE D'ACCUEIL ET LES AUTRES PAGES

Liste de quelques modules et cadres du portail visibles dans la page d'accueil – dans les zones introduction et page centrale

En noir (police standard) : le nom du cadre | ***En rouge et en italiques : le nom du module***

Menu
Menu

Carrousel d'articles (accueil)
Liste d'articles

Liste d'articles - sélections (accueil)
Liste d'articles

Liste de brèves (accueil)
Liste d'articles

Barre de recherche simple
Recherche

Agenda (accueil)
Agenda

Carrousel de notices (nouveautés - accueil)
Liste de notices

Si les pages-outils n'existaient pas, si elles ne contenaient pas le cadre fondamental lié à leur fonction, s'il n'y avait pas des variables d'environnement et des champs personnalisés pour transmettre des paramètres, si les templates Django ne créaient pas des URL de destination et ne spécifiaient pas ce qui est cliquable, il serait inutile d'insérer des cadres en page d'accueil, soit on n'irait nulle part ailleurs, soit on irait quelque part mais on ne pourrait rien y faire !

Interconnexion entre ces cadres visibles en page d'accueil et les pages-outils du portail

Il n'y a pas de variable d'environnement liée à la page qui affiche une liste d'articles (le résultat d'une recherche dans le contenu éditorial).

	A	B	C	D
1	Id	Nom de la page-outil	Variable d'environnement	Libellé
2	4	Page qui affiche un article et une liste de notices	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	id_date	Identifiant de date
6	11	Page qui affiche une liste d'articles (recherche)		

On ne va jamais « spontanément » vers une page, quelle qu'elle soit
C'est dans chaque cadre qu'on construit le lien vers « la suite du parcours ».

Cadres →	Variable d'environnement	→ Pages de destination
Menu (Le menu de la démo n'affiche que des titres de rubriques) Clic sur un titre de rubrique	**id_rubrique**	Page qui affiche une **rubrique** → Si on clique sur une sous-rubrique → page qui affiche une rubrique → Si on clique sur un article → page qui affiche un article
Carrousel d'articles (accueil) Clic sur un article du carrousel **Liste d'articles - Sélections (étagères)** Clic sur un article de la liste	**id_article**	Page qui affiche un **article** L'aspect varie en fonction du **type de contenu de l'article** (ce paramètre est transmis par un **champ personnalisé**) Coup de cœur : **identifiant_notice** Sélection : **etagere** Événement (date précise) : **date** Événement (période) : **date_debut, date_fin** Information : pas de champ personnalisé
Liste de brèves – pas de clic possible	Pas de lien	Aucune page de destination prévue dans le template de ce cadre
Barre de recherche dans les articles Si on effectue une recherche	Pas de variable	Page qui affiche une **liste d'articles : le résultat d'une recherche** → Si on clique sur un article de la liste → page qui affiche un article
Agenda (calendrier : choix d'une date) S'il y **a un seul** événement	**id_article**	Page qui affiche un **article**
S'il y a **plusieurs** événements	**id_date**	Page qui affiche **une liste d'événements (module agenda)** → Si on clique sur un événement de la liste → page qui affiche un article
Carrousel de notices (nouveautés) Clic sur une notice du carrousel	**id_notice**	Page du portail qui affiche une **notice** → Si on clique sur le permalink → page de l'OPAC qui affiche une notice

Ces variables se ressemblent : leurs noms sont similaires, et les questions de départ identiques : quelle notice ? ou quelle date ?
Mais elles n'ont pas la même fonction, ne les confondez pas !

id_notice et **id_date** sont 2 **variables d'environnement** → **page** qui affiche **une notice ou page** qui affiche **une liste d'événements**
identifiant_notice et **date** sont 2 **champs personnalisés** → **types de contenu des articles : Coup de cœur ou Evénement à une date**

SYNTHÈSE DES PAGES-OUTILS DU PORTAIL

Vous pouvez cliquer sur <u>n'importe quelle rubrique</u> visible par exemple dans le menu principal
Le cadre du menu principal contient la question « Construction du lien vers une rubrique ? »
Et on a répondu « aller vers la page du portail qui affiche une rubrique »
C'est une page où on voit le contenu d'une rubrique, quel que soit son numéro d'id

- Dans la démo, la page qui affiche une rubrique a le numéro d'id 6 (ce numéro ne change pas)
- la **variable d'environnement id_rubrique** transmet le numéro d'id de la rubrique à afficher (ce numéro varie donc)
- http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=6&id_rubrique=3

Afficher le contenu d'une rubrique, c'est montrer le titre et le résumé, et aussi <u>les sous-rubriques et/ou articles qui en dépendent</u>.
Le cadre « Afficher une rubrique » contient dès lors <u>deux fois</u> la question « Construction du lien vers ? »
« Construction du lien vers <u>une rubrique</u> ? » - « Construction du lien vers <u>un article</u> ? »

- Un clic sur une sous-rubrique vous mène vers la page qui affiche une rubrique (id_rubrique change bien sûr de valeur)
- Un clic sur un article vous mène vers la page qui affiche un article

Vous pouvez cliquer sur <u>n'importe quel article</u> visible dans une liste ou un carrousel d'articles
(sauf les Brèves en page d'accueil : ce template Django est particulier, car il ne prévoit pas de lien vers une autre page)
Le cadre de la liste d'articles ou du carrousel d'articles contient la question « Construction du lien vers un article ? »
Et on a répondu « aller vers la page du portail qui affiche un article et une liste de notices »
C'est une page où on voit le contenu d'un article, quel que soit son numéro d'id, et quel que soit son type de contenu éditorial

- Dans la démo, la page qui affiche un article a le numéro d'id 4 (ce numéro ne change pas)
- **la variable d'environnement id_article** transmet le numéro d'id de l'article à afficher (ce numéro varie donc)
- http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=9

- Le **template Django du cadre « Afficher un article » tient compte de la diversité des types de contenu éditorial**

La page du portail qui affiche un article peut prendre des aspects différents.
Cela varie en fonction du type de contenu de l'article, et des autres cadres présents éventuellement dans cette page.

En effet, cette page ne contient pas uniquement le cadre « Afficher un article », il y en a deux autres.
Module Liste de notices → « Etagère associée à un article » et **« Notices dans les mêmes catégories qu'un article »**

Type Sélection	Type Coup de cœur	Type Evénement à une date
Ce type d'article est **lié à une étagère** **champ personnalisé : etagere**. **On voit <u>deux</u> cadres** : l'article et la liste des notices de cette étagère. En théorie, on pourrait aussi voir le 3e cadre : le carrousel de notices sur le même sujet.	Ce type d'article est **lié à une notice** **champ personnalisé : identifiant_notice** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.	Ce type d'article est **lié à <u>une</u> date précise** **champ personnalisé : date** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.
	Type Information	**Type Evénement sur une période**
Ce sont les 5 types de contenu définis dans la démo. Voir configuration préliminaire du contenu éditorial	Ce type d'article n'est **lié à aucun champ personnalisé**. **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.	Ce type d'article est **lié à <u>deux</u> dates précises : la date de début et la date de fin de l'activité** **champs personnalisés : date_debut, date_fin** **On voit 1 ou 2 cadres** : l'article et (parfois) un carrousel de notices sur le même sujet.

Le nom complet « page qui affiche un article et une liste de notices » se justifie souvent, mais pas dans tous les cas.
En effet, on voit toujours un article, mais pas nécessairement le contenu d'une étagère ou un carrousel de notices.

- **La présence de l'étagère liée à une liste de notices n'est possible que si l'article est de type Sélection**, si on a enregistré le numéro d'id d'une étagère (dans le champ personnalisé lié à ce type d'article) … et si l'étagère contient des notices !
- **Le carrousel de notices sur le même sujet peut apparaître de manière automatique, quel que soit le type de contenu de l'article. Sa présence éventuelle dépend des catégories communes à l'article et à certaines notices de la base**. Ce carrousel est donc créé de manière dynamique, sur base des liens entre le contenu éditorial et le contenu bibliographique.

Ne confondez pas ces variables ! Leurs noms sont similaires, et les questions de départ identiques : quelle notice ? ou quelle date ?
id_notice et **id_date** sont 2 **variables d'environnement → page** qui affiche **une notice <u>ou</u> page** qui affiche **une liste d'événements**
identifiant_notice et **date** sont 2 **champs personnalisés → types de contenu des articles : Coup de cœur <u>ou</u> Evénement à une date**

Vous pouvez cliquer sur n'importe quelle notice visible dans un carrousel de notices.
Le cadre de ce carrousel contient la question « Construction du lien vers une notice ? »
Et on a répondu « aller vers la page du portail qui affiche une notice »
C'est une page où on voit le contenu d'une notice, quel que soit son numéro d'id

- Dans la démo, la page du portail qui affiche une notice a le numéro d'id 7 (ce numéro ne change pas)
- **la variable d'environnement id_notice** transmet le numéro d'id de la notice à afficher (ce numéro varie donc)
- http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=88

- Dans la démo, le cadre « Afficher une notice » montre une version abrégée de la notice, et aucun champ lié aux exemplaires
- C'est pourquoi on affiche (à la dernière ligne) le permalink afin de consulter la notice et les exemplaires dans l'OPAC

La démo propose parfois aussi d'aller directement vers la page de l'OPAC qui affiche une notice

La liste de notices (étagère associée) et l'article de type « Coup de cœur » suggèrent de voir le document dans le catalogue, et contiennent donc un lien (le permalink) vers la page de l'OPAC qui affiche une notice.

- http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=notice_display&id=88

Mais ce n'est pas un parcours obligatoire ! Ce que je viens d'expliquer ci-dessus, c'est le mode de fonctionnement dans la démo.
Dans votre portail, vous ne devez pas nécessairement faire les mêmes choix que dans la démo.
Ex. vous pouvez décider qu'un carrousel de notices mène directement vers la page de l'OPAC qui affiche une notice, sans transiter par la page du portail qui affiche une notice. Il suffit de modifier le paramétrage du carrousel de notices.

Vous pouvez cliquer sur n'importe quelle date visible dans l'agenda (en page d'accueil)
La page du portail qui affiche une liste d'événements est une page où on voit une série d'événements,
quelle que soit la date choisie … mais à condition qu'il y ait des articles de type Evénement pour annoncer les activités !!

Attention ! On ne va pas nécessairement vers la page qui affiche une liste d'événements !
En effet, la date sélectionnée dans l'agenda peut être liée à zéro, une ou à plusieurs activités qui se déroulent précisément à la date choisie, ou qui s'étalent sur une période qui englobe la date choisie.

Le cadre « agenda » visible en page d'accueil contient dès lors deux fois la question « Construction du lien vers ? »
« Construction du lien vers un événement ? » - « Construction du lien vers plusieurs événements ? »

- Si **aucun événement** n'est prévu pour cette date, rien ne se passe ! On reste dans la page où on voit le calendrier.

- Si **un seul événement** est prévu pour cette date, on va directement **vers la page qui affiche l'article lié à cet événement**
- Dans la démo, la page qui affiche un article a le numéro d'id 4 (ce numéro ne change pas)
- **la variable d'environnement id_article** transmet le numéro d'id de l'article à afficher (ce numéro varie donc)
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=4&id_article=1

- Si **plusieurs événements** sont prévus pour cette date, on va **vers la page qui affiche une liste d'événements**
- Le **cadre Liste d'événements** est créé à partir du **module agenda.**
- Dans la démo, la page qui affiche une liste d'événements a le numéro d'id 9 (ce numéro ne change pas)
- **la variable d'environnement id_date** transmet la date sélectionnée (cette date varie donc)
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=9&id_date=2013-01-09
- **Ensuite, si on clique sur le logo d'un événement**, on ira vers la **page qui affiche l'article adéquat**

Les dates des événements ne sont pas le seul critère qui détermine l'affichage ou non de tel ou tel article de type Evénement !
Le statut de publication d'un article et ses dates de publication ont aussi un impact filtrant ! Voir configuration du contenu éditorial
Vous ne voyez donc pas tous les articles, seulement ceux dont le statut est « publié » et dont les dates de publication sont valides.

Vous pouvez effectuer une recherche dans le contenu éditorial, si la barre de recherche a été configurée en ce sens
La question « Construction du lien vers ? » n'est pas posée dans le formulaire de création de ce cadre.
C'est dans l'interface d'administration que vous sélectionnez la page du portail qui affiche le résultat d'une recherche.
C'est une page où on voit la liste des articles qui répondent au critère de recherche.

- Dans la démo, la page qui affiche le résultat d'une recherche dans les articles a le numéro d'id 11 (ce numéro ne change pas)
- **Il n'y aucune variable d'environnement liée à cette page**
- http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=cmspage&pageid=11

- Si un ou plusieurs articles contiennent le(s) mot(s) recherché(s), on voit le titre et le résumé de cet article (ou de ces articles)
- Un clic sur un de ces articles mène vers la page qui affiche un article.
- Sinon, on voit le message « Pas de résultats » et on propose d'effectuer une recherche dans le catalogue.

Le mot recherché dans le contenu éditorial n'est pas le seul critère qui détermine l'affichage ou non de tel ou tel article !
Le statut de publication d'un article et ses dates de publication ont aussi un impact filtrant ! Voir configuration du contenu éditorial
Vous ne voyez donc pas tous les articles, seulement ceux dont le statut est « publié » et dont les dates de publication sont valides.

Vous pouvez effectuer une recherche dans le contenu bibliographique, si la barre de recherche a été configurée en ce sens
La question « Construction du lien vers ? » n'est pas posée dans le formulaire de création de ce cadre.
C'est dans l'interface d'administration que vous sélectionnez la page de l'OPAC qui affiche le résultat d'une recherche.
C'est une page où on voit la liste des notices qui répondent au critère de recherche.

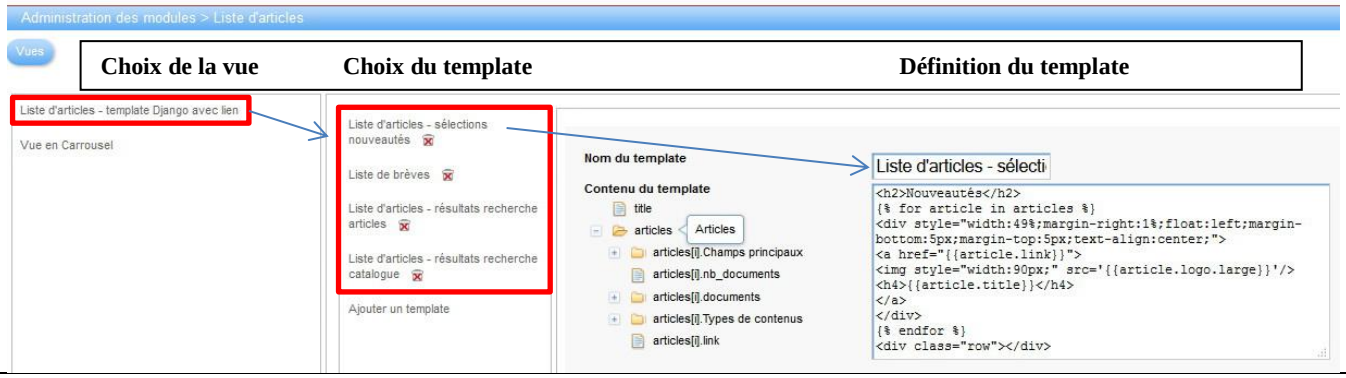
- La page qui affiche le résultat d'une recherche dans les notices correspond forcément à une adresse de type OPAC
- http://127.0.0.1/pmbv4perso2/opac_css/index.php?lvl=more_results

- Si une ou plusieurs notices contiennent le(s) mot(s) recherché(s), on voit la liste des notices, qu'on peut déplier si nécessaire.
- Un clic sur le permalink d'une notice mène vers la page de l'OPAC qui affiche cette notice dans son intégralité.
- Sinon, on propose de faire une suggestion d'achat, ou bien d'interroger des sources externes.

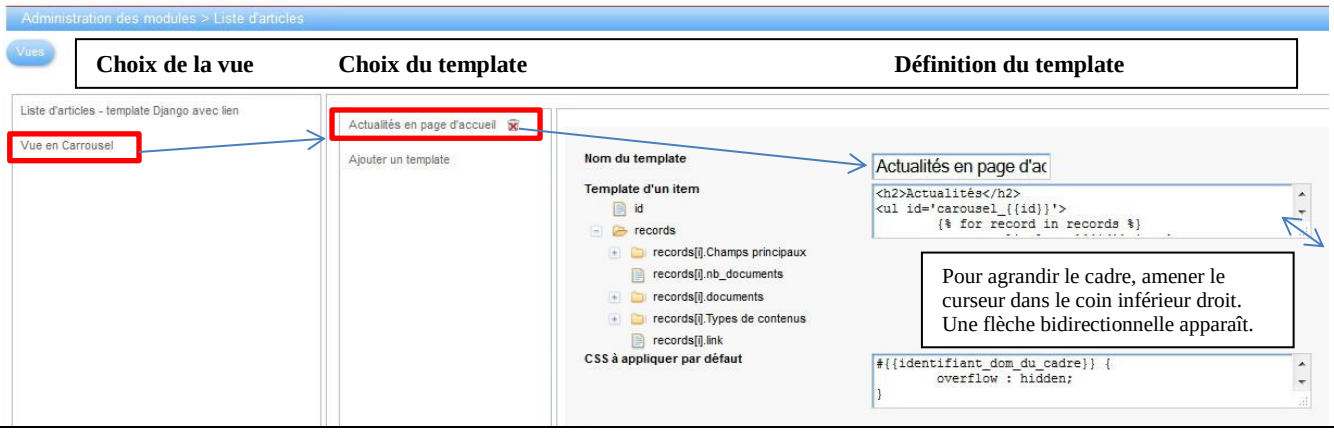
Templates Django

Interface d'administration - Vues et templates

Exemple : module Liste d'articles – Vue : template Django avec lien - On a enregistré 4 modèles de template



Exemple : module Liste d'articles – Vue : carrousel - On a enregistré 1 modèle de template



La « vue » a un sens plus global que le template : c'est une manière générale d'afficher les données.
Pour chacune de ces vues, on peut prévoir un ou plusieurs templates, afin de diversifier la présentation.

DÉMO DU PORTAIL - LISTE DES MODULES, DES TEMPLATES DJANGO ET DES CADRES OÙ ILS SONT EMPLOYÉS		
Modules	**Templates Django**	**Cadres où ces templates sont employés**
3 modules de base : un article - une notice - une rubrique		
Article	Affichage d'un article	→ Cadre principal de la page qui affiche un article
Notice	Affichage d'une notice	→ Cadre principal de la page qui affiche une notice
Rubrique	Affichage d'une rubrique	→ Cadre principal de la page qui affiche une rubrique
Vues	**Templates Django**	**Cadres où ces templates sont employés**
Module liste d'articles		
Templates Django avec lien	Liste d'articles – Sélections – Nouveautés	Sélection de nouveautés (symbolisées par les étagères)
	Liste de brèves	Liste de brèves
	Liste d'articles – Résultat d'une recherche dans les articles	→ Cadre principal de la page qui affiche le résultat d'une recherche dans le contenu éditorial
	Liste d'articles – Recherche dans le catalogue	Articles dans les mêmes catégories qu'une notice
Carrousel	Actualités en page d'accueil	Carrousel d'articles d'actualité
Module liste de notices		
Vue Django	Liste de notices	Notices d'une étagère associée à un article
Vue en carrousel	Notices du même auteur	Notices du même auteur qu'une notice
	Notices similaires	N.B. Ces 2 templates ne sont pas illustrés dans la démo.
	Notices sur la même étagère (cote de rangement proche)	
	Nouveautés en page d'accueil	Notices – Carrousel de nouveautés
	Notices dans les mêmes catégories qu'un article	Notices dans les mêmes catégories qu'un article (même sujet)
Module agenda		
Liste d'événements	Affichage d'une liste d'événements à partir de l'agenda	→ Cadre principal de la page qui affiche une liste d'événements
Conclusion : à chaque cadre illustré dans la démo correspond au moins un template sauvegardé dans l'interface d'administration. **Comme vous pouvez le constater, tous les cadres et templates illustrés dans la démo ont été commentés dans ce manuel !**		

LISTE DES INSTRUCTIONS ET FONCTIONS DES TEMPLATES DJANGO

Les instructions entre {% %}

Les boucles de répétition **Exemple** **{% for article in articles %} …** {{article.title}} … {{article.resume}} … {{article.content}} … **{% endfor %}**		
Liste d'articles (template Django avec lien)	**{% for article in articles %}**	Article du contenu éditorial
Liste de notices (vue Django ou carrousel) Carrousel de notices Carrousel d'articles Carrousel de rubriques	**{% for record in records %}** Pour toutes les listes de notices Pour tous les carrousels	A record = enregistrement Suivant le cas, record = une notice, un article ou une rubrique
Liste de rubriques (template Django avec lien)	**{% for section in sections %}**	Section = rubrique du contenu éditorial
Liste des sous-rubriques d'une rubrique parente	**{% for child in children %}**	Dans ce contexte, a child = une sous-rubrique
Agenda (vue d'une liste d'événements)	**{% for event in events %}**	An event = un événement
Liste d'étagères	**{% for shelve in shelves %}**	A shelf (pluriel shelves) = une étagère
Liste de bannettes	**{% for bannette in bannettes %}**	Les bannettes de la DSI

Les conditions : 2 exemples concrets		
Contexte : un article est lié à une étagère. On affiche l'article, et en-dessous, la liste des notices contenues dans cette étagère N.B. J'ai abrégé le contenu du template, j'ai gardé les instructions essentielles pour comprendre le sens de la condition.	**{% if records %}** Toutes les nouveautés **{% endif %}** {% for record in records %} {{record.content}} {% endfor %}	S'il y a des enregistrements (= notices) à afficher On affiche le titre « Toutes les nouveautés » Fin de la condition Boucle qui va de la 1e à la dernière notice On affiche le contenu de la notice Fin de la boucle
La condition {% if records %} {% endif %} est nécessaire dans cet exemple Inutile d'afficher le titre « Toutes les nouveautés » s'il n'y a pas d'étagère associée à l'article ou si elle est vide !		
Contexte : page → résultat de recherche On cherche un mot dans les articles. Si on le trouve, on affiche la liste des articles concernés. N.B. J'ai abrégé le contenu du template, j'ai gardé les instructions essentielles pour comprendre le sens de la condition.	**{% if articles %}** {% for article in articles %} …. {% endfor %} **{% else %}** Pas de résultats **{% endif %}**	S'il y a des articles à afficher Boucle qui va du 1er au dernier article On affiche la version abrégée (titre + résumé) Fin de la boucle Sinon On affiche le message « Pas de résultats » Fin de la condition
La condition {% if articles %} {% else %} {% endif %} est nécessaire dans cet exemple On a une boucle qui va du 1er au dernier article seulement si on a trouvé le mot quelque part dans le contenu éditorial ! Et en toute logique, le message « Pas de résultats » n'a de sens que dans le cas d'une recherche infructueuse !		

Les conditions : synthèse schématique		
Condition « simple » : si … alors … **On n'envisage qu'un seul cas :** **si la condition est vérifiée** **Il n'y a rien à faire dans le cas contraire** **{% if true %} Si la condition est vérifiée** Instruction à exécuter **{% endif %} Fin de la condition**	**Condition « si … alors … sinon … »** **On envisage les 2 termes de l'alternative :** **si la condition est vérifiée** **si elle ne l'est pas** **{% if true %} Si la condition est vérifiée** Instruction n° 1 **{% else %} Sinon** Instruction n° 2 **{% endif %} Fin de la condition**	**On peut imbriquer 2 conditions l'une dans l'autre** **{% if true %} Si la condition n° 1 est vérifiée** Instruction n° 1 **{% if true %} Si la condition n° 2 est vérifiée** Instruction n° 2 **{% else %} Sinon** Instruction n° 3 **{% endif %} Fin de la condition n° 2** **{% endif %} Fin de la condition n° 1**
On symbolise en général une condition Django de cette manière : (% if true %) → si c'est vrai, si la condition est vérifiée		
On peut effectuer les tests classiques de comparaison		
if {{variable1}} == {{variable2}} if {{variable1}} != {{variable2}}	if {{variable1}} < {{variable2}} if {{variable1}} <= {{variable2}}	if {{variable1}} > {{variable2}} if {{variable1}} >= {{variable2}}
Par convention == signifie "égal à" (double signe =)	**Par convention != signifie "différent de"**	
On peut employer les opérateurs booléens habituels : and or not if {{variable2}} >= {{variable1}} and {{variable2}} <= {{variable3}}		

Infos complémentaires : https://docs.djangoproject.com/en/1.8/ref/templates/builtins/

Une fonction supplémentaire : limitstring	
PMB a ajouté une fonction aux templates Django de base, la fonction limitstring	
{{record.title | limitstring,45,"..." }}	**Attention à la syntaxe ! Le tout doit être entre {{ }}** **Il faut une barre verticale | entre la variable et limitstring**
La fonction limitstring permet de limiter la longueur d'une chaîne de caractères. Exemple {{record.title | limitstring,45,"..." }} Cela signifie que la longueur du texte à afficher (le titre de la notice, record.title) sera limitée à 45 caractères, et que les caractères manquants seront remplacés par le texte entre guillemets (les 3 petits points). Le découpage se fait à l'aveugle, et le résultat final est plus ou moins heureux suivant le cas = suivant le texte à afficher ! Si vous avez une série de titres assez longs dans un carrousel de notices, cette fonction est fort utile. Bien entendu, son rôle n'est pas limité aux carrousels, elle peut aussi servir en d'autres circonstances.	

VARIABLES UTILISABLES DANS LES TEMPLATES DU PORTAIL

Les noms des variables entre {{ }}		
Les variables utilisables dans les templates ont un lien avec les champs de la base de données ou avec le contenu éditorial		
Le sens de certaines variables comme {{article}} ou {{section}} est évident. Mais … **Tous les carrousels (notices, articles, rubriques) emploient la boucle {% for record in records %}** **Toutes les listes de notices (vue Django ou carrousel) emploie la boucle {% for record in records %}** **La variable {{record}} a donc plusieurs significations : elle peut désigner une notice, un article ou une rubrique.**		
{{title}}	{{article.title}}	Le titre de l'article
	{{section.title}}	Le titre de la rubrique
	{{record.title}}	Le titre de l'enregistrement Suivant le cas, titre de la notice, de l'article ou de la rubrique
	{{event.title}}	Le titre de l'événement (Le terme *événement* a un sens global dans ce contexte : article qui décrit une activité à une date ou sur une période)
{{resume}}	{{article.resume}}	Le « résumé » de l'article = le chapeau de l'article
	{{section.resume}}	Le « résumé » de la rubrique
	{{record.resume}}	Le « résumé » de l'enregistrement Suivant le cas, résumé de la notice, chapeau de l'article ou résumé de la rubrique
	{{event.resume}}	Le « résumé » de l'événement = le chapeau de l'article
{{content}}	{{article.content}}	Le contenu de l'article = le texte même de l'article
	{{record.content}}	Le contenu de l'enregistrement - Suivant le cas • le contenu de la notice = tous les champs • le contenu de l'article = le texte même de l'article N.B. Rappel : une rubrique n'a pas de champ « contenu »
	{{event.content}}	Le contenu de l'événement = le texte même de l'article
{{link}}	{{article.link}}	Le lien = URL de l'article
	{{section.link}}	Le lien = URL de la rubrique
	{{record.link}}	Le lien = URL de l'enregistrement Suivant le cas, URL de la notice, de l'article ou de la rubrique
	{{event.link}}	Le lien = URL de l'événement
{{vign}}	{{article.vign}}	La vignette de l'article
	{{section.vign}}	La vignette de la rubrique
	{{record.vign}}	La vignette de l'enregistrement Suivant le cas, la vignette de la notice, de l'article ou de la rubrique
	{{event.vign}}	La vignette de l'événement
Exemples de champs personnalisés, qui correspondent aux types de contenu définis pour les articles **N.B. Le nom d'une variable associée à un champ personnalisé contient toujours ceci : {{fields_type}}**		
{{fields_type.date.values.0.format_value}}		Evénement à une date précise
{{fields_type.date_debut.values.0.format_value}}		Evénement sur une période : date de début
{{fields_type.date_fin.values.0.format_value}}		Evénement sur une période : date de fin
{{fields_type.etagere.values.0.value}}		Sélection : identifiant de l'étagère associée à un article
{{fields_type.identifiant_notice.values.0.value}}		Coup de cœur : identifiant de la notice associée à un article
{{fields_type.url.values.0.value}}		Lien : URL de la page à ouvrir
{{fields_type.nouvel_onglet.values.0.value}}		Ouvrir une page dans un autre onglet du navigateur ?

Balises HTML

Presque toutes les balises HTML vont deux par deux : il y a une balise ouvrante et une balise fermante, ex. <h2> et </h2>
N.B. HTML = HyperText Markup Language

**La balise la plus importante : … **
Elle permet de créer des liens URL et comprend 2 éléments : • **l'URL de destination : la page vers laquelle on ira** • **la source, le point de départ : les éléments sur lesquels on clique (texte et/ou image)**

L'URL de destination peut être exprimée sous la forme d'une adresse absolue ou d'une adresse relative
Exemple : la page de l'OPAC qui affiche une notice dont le numéro d'id est 88 **Adresse absolue de cette page :** http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=88 **Adresse relative de cette page – relative par rapport au répertoire /pmb/opac_css/ :** http://./index.php?lvl=notice_display&id=88

Comment rendre un texte cliquable en HTML ?	
<a href="… : URL de destination	**Texte entre <a> et : source, d'où on part : on clique sur le texte**
Site de Anne-Marie Cubat	

Comment rendre une image cliquable en HTML ?	
<a href="… : URL de destination	**<img src="… : source, d'où on part : on clique sur l'image**
	
N.B. Pour les balises alt=" " et title=" ", employez des guillemets, mais pas l'apostrophe. **alt=" " : texte alternatif à afficher si l'image n'est pas disponible** **title=" " : infobulle - texte à afficher au survol de la souris sur l'image** **L'apostrophe posera problème dès que le texte (dans le alt ou le title) contient des apostrophes.** **Modifiez donc quelques templates Django si nécessaire.** En effet, certains d'entre eux créent le contenu des balises alt ou title avec le titre d'une notice, or ces titres peuvent contenir des apostrophes, donc ils seraient tronqués dès la 1e apostrophe. ***L'Avare* de Molière se réduirait ainsi à la seule lettre "L" !!** Exemples corrects : **alt="{{record.title}}" ou title="{{record.title}}"**	

Comment rendre le texte et l'image cliquables en <u>une</u> seule balise HTML ? On combine le tout !
Site de PMB Site de PMB
C'est d'ailleurs ce qui se passe dans beaucoup de templates : une seule balise <a href=" pour le texte et l'image
 {{article.title}} **Des variables comme {{article.link}} ou {{record.link}} désignent l'URL de destination, leur contenu est créé de manière dynamique.** Exemples de contenu pour un lien vers la page qui affiche une notice (dans sa version portail et dans sa version OPAC classique) Lien vers la page du portail : http://127.0.0.1/pmb4perso/opac_css/index.php?lvl=cmspage&pageid=7&id_notice=110 Lien vers la page de l'OPAC : http://127.0.0.1/pmbv4perso/opac_css/index.php?lvl=notice_display&id=110

Comment rendre un texte cliquable et ouvrir l'URL de destination dans un autre onglet du navigateur ?	
<a href="… : URL de destination	**Texte entre <a> et : source, d'où on part : on clique sur le texte**
Dans la partie qui contient l'URL de destination, ajouter l'instruction target="_blank" Site de Anne-Marie Cubat devient Site de Anne-Marie Cubat	

Comment rendre une image cliquable et ouvrir l'URL de destination dans un autre onglet du navigateur ?	
<a href="… : URL de destination	**<img src="… : source, d'où on part : on clique sur l'image**
Dans la partie qui contient l'URL de destination, ajouter l'instruction target="_blank" devient 	

Quelques balises HTML employées fréquemment dans les templates de notices ou les templates Django	
Balises « simples » = qui n'ont pas de balise fermante	**La balise ** provoque un saut de ligne (br = break) **La balise <hr>** trace une ligne horizontale (hr = horizontal rule)
Mise en forme du texte	**Les balises <h2> </h2> (<h3> <h4>)** reflètent la hiérarchie titre / sous-titres et déterminent la taille des caractères **Les balises <div style="** … > et ** ** marquent le début et la fin d'un « bloc de texte » à mettre en forme. L'affichage du texte situé entre ces 2 balises peut être personnalisé : marges, alignement, police de caractères … **La balise <blockquote>** détermine un mode d'affichage (bloc de citation) **Les balises et ** marquent le début et la fin d'un paragraphe
Image	**La balise <img>** signifie image. **<img … src=' …'>** signifie la source de l'image = l'URL de l'image (img src = image source). **** La partie style="… "** permet de définir le mode d'affichage (ex. hauteur et largeur limitées à 120px – centré)
Liste non ordonnée (liste à puces) …. …. 	**Les balises et ** marquent le début et la fin d'une liste non ordonnée = non numérotée (ul = unordered list) Une liste non ordonnée peut prendre la forme d'une liste à puces, mais cela peut aussi signifier tout simplement une liste d'éléments non numérotés, ou une liste d'éléments qui figurent dans un carrousel. **N.B. Les balises et ** marquent le début et la fin d'une liste numérotée (ol = ordered list = liste ordonnée) **Les balises et ** marquent le début et la fin de chaque élément de la liste (non) ordonnée (li = list)

Les tableaux en HTML– Cet exemple est extrait du template de notices appelé « Portail – Liste de notices »
Voir Notices d'une étagère associée à un article

Les balises **<table> et </table>** marquent le début et la fin d'un tableau.
Les balises **<tbody> et </tbody>** marquent le début et la fin de la définition du tableau.

Les balises **<tr> et </tr>** marquent le début et la fin de chaque ligne.
Les balises **<td> et </td>** marquent le début et la fin de chaque cellule.

En résumé

```
<table>
  <tbody>
      <tr> Début de l'unique ligne de ce tableau
              <td> 1e cellule : la vignette  </td>
              <td> 2e cellule : titre, auteur et résumé de la notice  </td>
      </tr> Fin de la ligne
  </tbody>
</table>
```

On ne le croirait pas à première vue, mais cette notice est affichée dans un tableau dont les bordures sont invisibles !
Je trace ces bordures pour clarifier les choses ;-)

Ce tableau comprend donc
- une seule ligne : l'ensemble des infos sur la notice
  - <tr> = table row (ligne, rangée)
- 2 cellules par ligne
  - <td> = table division (une cellule, dans ce contexte)
  - à gauche : la vignette
  - à droite : titre, auteur et résumé de la notice

Requêtes SQL utiles

J'ai repris ici la liste complète des requêtes et beaucoup de captures d'écran, mais pas toutes les explications !
En effet, ces procédures et les concepts qui y sont liés ont déjà fait l'objet de commentaires détaillés ailleurs :
Voir surtout Contenu éditorial Pages de l'OPAC et du portail Modules, cadres et pages

Si vous avez restauré le jeu de sauvegarde cms_min (configuration du contenu éditorial), la plupart de ces requêtes sont immédiatement prêtes à l'emploi, bien que certaines fassent appel aux champs personnalisés qu'on a pu créer.

Tout d'abord, un petit mot sur l'origine de ces requêtes

Toutes ces requêtes sont « de mon cru », mais je tiens quand même à remercier PMB Services, qui a fourni un point de départ utile, surtout un « petit bloc d'instructions » que j'ai pu insérer dans mes procédures.

Au terme de la formation initiale, nous avons en effet reçu de Camille Bénard une série de 8 requêtes liées au portail. Je les ai regardées attentivement, et j'y ai trouvé des idées intéressantes, surtout quelques lignes qui permettent d'afficher une partie de l'arbre du contenu éditorial, afin de situer exactement chaque article (voir ci-dessous). J'ai donc intégré ce bloc d'instructions dans mes procédures.

On a une meilleure vue de l'arborescence si on sait par exemple qu'un article est dans
Menu principal > Sélections > Coups de cœur
C'est moins clair si on sait seulement qu'un article est dans la rubrique Coups de cœur.

C'est donc ce bloc-là d'instructions que j'ai repris des requêtes que nous avons reçues.

Rubrique	Titre de l'article
Menu principal > Sélections > Coups de cœur	Des bibliothèques pleines de fantômes
Menu principal > Sélections > Coups de cœur	Georges : petit-petit-petit magazine
Menu principal > Sélections > Coups de cœur	Poisson rouge
Menu principal > Animations > Evénements	Cultures d'Afrique noire contemporaine
Menu principal > Animations > Evénements	Projections : journée Tim Burton
Menu principal > Animations > Heure du conte	Lisez 5 fruits et légumes par jour !

Les requêtes SQL spécifiques au portail sont disponibles sur mon site, il y a un zip à télécharger à cette adresse :
http://amcubat.be/docpmb/portail-construction-concepts-exemples

N.B. Lorsqu'on décompresse le zip, cela crée automatiquement des dossiers dans lesquels les requêtes sont réparties.
Voici la liste des requêtes de chaque répertoire, et des copies d'écran pour illustrer les plus importantes.

Les requêtes qui figurent dans les deux premiers groupes (Utilitaires – Gestion de la base et Requêtes générales du portail) sont tout à fait standards, et fonctionnent dans toutes les bases. Aucune adaptation n'est nécessaire.

Utilitaires – Gestion de la base en général
Etagères - Liste avec les autorisations accordées
Etagères - Liste des étagères et des paniers qui y sont associés - Avec les numéros d'id
Paniers (Catalogue : bulletins, exemplaires et notices) - Liste avec les autorisations accordées
Paniers – Contenu d'un panier de notices (monographies, périodiques, articles ou notices de bulletins)
Paramètres - Liste complète : type, nom, valeur, commentaire et section
Utilisateurs - Liste : login, nom, prénom et n° d'id

Exemple de liste des utilisateurs – avec le numéro d'id

N° id	Login	Nom	Prénom
1	admin	Super User	
4	Eric	Eric	
5	Florent	Florent	
6	Camille	Camille	

Exemple de liste de paniers – Avec le numéro d'id et le nombre de notices par panier

Type de panier	Id	Nom du panier	Nombre	Commentaire	Utilisateurs
Bulletins	6	Exemple de panier de bulletins	0		1
Exemplaires	8	Exemple de panier d'exemplaires	0		1 4 3 2
Notices	11	Exemple de panier de notices	0		1
Notices	12	Nouveautés en page d'accueil	13		1
Notices	13	Nouveautés fiction adultes	9		1
Notices	14	Nouveautés fiction jeunesse	13		1
Notices	15	Nouveautés documentaires	20		1
Notices	16	Nouveautés CD	5		1
Notices	17	Nouveautés DVD	5		1
Notices	18	Nouveautés périodiques	13		1

Exemple de contenu de 2 paniers de notices - avec indication du type de notice

	A	B	C	D	E	F	G	H	I	J	K	L
1	Id	Panier	ISBN / ISSN	Auteur	Type de notice	Id	Titre de la notice (+ titre de série)	Libellé de période	Date de parution	Editeur	Cote	Numéro (ex)
2	14	Nouveautés fiction jeun	978-2-87833-390-9	Heitz, Bruno	Monographie	81	Jojo (série) : Jojo et le secret de la		2006	Circonflexe	RJP HEI	000045
3	14	Nouveautés fiction jeun	978-2-07-053882-9	Mahy, Margaret	Monographie	87	L'enlèvement de la bibliothécaire		2002	Gallimard jeunesse	RJ MAH	000037
4	14	Nouveautés fiction jeun	978-2-01-322406-2	Gudule,	Monographie	80	La bibliothécaire		2007	Hachette Jeunesse	RJ GUD	000046
5	14	Nouveautés fiction jeun	978-2-203-12937-5	Brisou-pellen, Evelyn	Monographie	85	Le grand amour du bibliothécaire		2005	Casterman	JR BRI	000039
6	18	Nouveautés périodique:		Bonnet, Jacques	Article	125	Bibliothécaire de soi-même in Bulle		01/11/2010			
7	18	Nouveautés périodique:			Article	133	Dark Shadows in Studio ciné live n	avril 2012	01/04/2012		PER STU	P000006
8	18	Nouveautés périodique:		Aïoutz, Renaud	Article	127	L'abeille et le bibliothécaire ou éthi		01/09/2012			
9	18	Nouveautés périodique:		Akkouche, Mouloud	Article	126	Voyages en BM in Bulletin des bib		01/11/2010			
10	18	Nouveautés périodique:			Notice de bulletin	122	Géo - N° 393 - novembre 2011 - L'Îl	novembre 2011	01/11/2011		PER GEO	P000002
11	18	Nouveautés périodique:			Notice de bulletin	123	Géo - N° 402 - août 2012 - Compos	août 2012	01/08/2012			
12	18	Nouveautés périodique:			Notice de bulletin	121	Géo - N° 403 - septembre 2012 - N	septembre 2012	01/09/2012		PER GEO	P000003
13	18	Nouveautés périodique:			Notice de bulletin	131	Georges - Numéro Cupcake		24/10/2012		PER J GEO	P000005
14	18	Nouveautés périodique:			Notice de bulletin	130	Georges - Numéro Moustache		24/10/2012		PER J GEO	P000004
15	18	Nouveautés périodique:	1292-8399		Périodique	124	Bulletin des bibliothèques de Franc			ENSSIB		
16	18	Nouveautés périodique:	0220-8245		Périodique	120	Géo			Participations éditio		
17	18	Nouveautés périodique:	2110-8749		Périodique	129	Georges			Éd. Grains de sel		
18	18	Nouveautés périodique:	1969-9441		Périodique	132	Studio ciné live magazine			Studio magazine		

Exemple de liste d'étagères – Avec le paramétrage pour l'OPAC

Id	Nom de l'étagère	Commentaire	OPAC ?	Validité	De ... à ...	Utilisateurs
3	Nouveautés		Invisible	Permanente		1
4	Nouveautés fiction adultes		Invisible	Permanente		1
5	Nouveautés fiction jeunesse		Invisible	Permanente		1
10	Test – Nouveautés - Fiction		Invisible	Permanente		1

Exemple de liste d'étagères - Avec le numéro d'id et le nombre de notices par panier associé

Id	Nom de l'étagère	Id	Nom du panier	Nombre
3	Nouveautés	12	Nouveautés en page d'accueil	13
4	Nouveautés fiction adultes	13	Nouveautés fiction adultes	9
5	Nouveautés fiction jeunesse	14	Nouveautés fiction jeunesse	13
6	Nouveautés documentaires	15	Nouveautés documentaires	20
7	Nouveautés CD	16	Nouveautés CD	5
8	Nouveautés DVD	17	Nouveautés DVD	5
9	Nouveautés périodiques	18	Nouveautés périodiques	13
10	Test - Nouveautés - Fiction	13	Nouveautés fiction adultes	9
10	Test - Nouveautés - Fiction	14	Nouveautés fiction jeunesse	13

Exemple de résultat obtenu dans un PMB standard grâce à la requête : contenu de la table des paramètres

Paramètres - Liste complète : type, nom, valeur, commentaire et section				Tri par type de paramètre, ensuite par nom du paramètre	
N° id	**Type**	**Nom**	**Valeur**	**Commentaire**	**Section**
741	Cms	Active	0	Module 'Portail' activé. 0 : Non. 1 : Oui.	
758	Cms	cache_ttl	1800	durée de vie du cache des cadres du portail (en secondes)	
797	Cms	url_base_cms_build		url de construction du CMS de l'OPAC	
106	Opac	biblio_adr1	ZI de Mont/Loir BP 10023	Adresse 1 de la bibliothèque ou du centre de ressources dans l'opac	b_aff_general
107	Opac	biblio_town	CHATEAU DU LOIR	Ville dans l'opac	b_aff_general

Portail – Requêtes générales
Articles - Liste complète - Aucun champ personnalisé
Champs personnalisés du portail - Liste avec les numéros d'id
Pages - Liste de toutes les pages avec les numéros d'id
Pages - Liste des pages qui ont une variable globale - Aves les numéros d'id de la page et de la variable
Rubriques - Liste complète - Aucun champ personnalisé
Types de contenu éditorial - Liste des types avec les numéros d'id

Liste des pages du portail

	A	B
1	Id	Nom de la page
2	3	Page d'accueil
3	4	Page qui affiche un article et une liste de notices
4	6	Page qui affiche une rubrique
5	7	Page qui affiche une notice
6	9	Page qui affiche une liste d'événements
7	11	Page qui affiche une liste d'articles (recherche)

Liste des variables d'environnement et des pages correspondantes

	A	B	C	D	E
1	Id (page)	Nom de la page	Id (variable)	Variable globale	Libellé
2	4	Page qui affiche un article et une liste de notices	6	id_article	Identifiant de l'article
3	6	Page qui affiche une rubrique	4	id_rubrique	Identifiant de la rubrique
4	7	Page qui affiche une notice	7	id_notice	Identifiant de la notice
5	9	Page qui affiche une liste d'événements	10	id_date	Identifiant de date

Les valeurs affichées (numéros d'id et noms des variables) correspondent à la démo du portail.

Tableau de bord : vue d'ensemble des articles, classés par rubrique – N.B. Liste abrégée, je ne montre que 9 articles sur 21 !

Rappel : l'actualité est éternelle dans une démo, les articles n'ont donc pas de dates de parution et tous ont le statut « publié ». J'ai donc dû inventer des dates de publication, et changer certains statuts de publication, pour diversifier les exemples.

Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
24/03/2015	Menu principal > Animations > Evénements	Cultures d'Afrique noire contemporaine	Evénement sur une période	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
24/03/2015	Menu principal > Animations > Evénements	Projections : journée Tim Burton	Evénement à une date précise	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
24/03/2015	Menu principal > Animations > Heure du conte	Lisez 5 fruits et légumes par jour !	Evénement à une date précise	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
24/03/2015	Menu principal > Infos pratiques > Inscription et prêt	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
24/03/2015	Menu principal > Infos pratiques > La bibliothèque	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
24/03/2015	Menu principal > Infos pratiques > La bibliothèque	Mais que font les bibliothécaires ?	Information	4. A valider			25/10/2012	7
24/03/2015	Menu principal > Sélections > Coups de coeur	Georges : petit-petit-petit magazine	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
24/03/2015	Menu principal > Sélections > Coups de coeur	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
24/03/2015	Menu principal > Sélections > Coups de coeur	Des bibliothèques pleines de fantômes	Coup de coeur	1. Publié - Pour tous			26/10/2012	17

Au départ, les données sont triées sur base de l'arbre du contenu éditorial, donc regroupées par rubrique.
Transférez le résultat vers le tableur, et supprimez d'abord la 1e ligne (celle qui contient le nom de la procédure) **avant** de trier.

Dans un tableur, il est très facile de trier les données dans l'ordre qui vous convient, par exemple par statut de publication.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
3	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
4	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
5	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
6	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
7	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
8	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
9	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7
10	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légume	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2

On peut aussi **trier les données par type de contenu**.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
3	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
4	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
5	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
6	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légume	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
7	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1
8	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
9	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
10	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7

Le tri par type de contenu permet de vérifier si les dates de parution sont cohérentes. On voit ici des anomalies pour les Coups de cœur. En théorie, ce type de contenu a une durée de vie limitée, or les dates de début et/ou de fin de publication manquent pour 2 articles.
La situation est normale pour les Evénements (dates mentionnées) et les Informations (il n'y a aucune date car la publication est permanente)

On peut aussi **trier les données par numéro identifiant (du plus grand au plus petit)**, afin de **voir d'abord les articles les plus récents**.

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	24/03/2015	Menu principal > Sélections > Cou	Des bibliothèques pleine	Coup de coeur	1. Publié - Pour tous			26/10/2012	17
3	24/03/2015	Menu principal > Sélections > Cou	Georges : petit-petit-pet	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	26/10/2012	12
4	24/03/2015	Menu principal > Animations > Evé	Projections : journée Tir	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/10/2012	11
5	24/03/2015	Menu principal > Infos pratiques >	Comment s'inscrire ?	Information	1. Publié - Pour tous			25/10/2012	10
6	24/03/2015	Menu principal > Infos pratiques >	Horaires	Information	1. Publié - Pour tous			25/10/2012	9
7	24/03/2015	Menu principal > Infos pratiques >	Mais que font les bibliot	Information	4. A valider			25/10/2012	7
8	24/03/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		25/10/2012	4
9	24/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légume	Evénement à une date	5. Archives	23/03/2014	23/03/2014	25/10/2012	2
10	24/03/2015	Menu principal > Animations > Evé	Cultures d'Afrique noire	Evénement sur une péri	1. Publié - Pour tous	13/04/2015	30/05/2015	25/10/2012	1

Tableau de bord : vue d'ensemble des rubriques

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique parente	Titre de la rubrique	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	27/03/2015	Racine	Menu principal	Rubrique	1. Publié - Pour tous			26/10/2012	12
3	27/03/2015	Racine	Série de brèves	Rubrique	1. Publié - Pour tous			26/10/2012	13
4	27/03/2015	Racine	Archives	Rubrique	1. Publié - Pour tous			17/11/2012	14
5	27/03/2015	Racine > Menu principal	Animations	Rubrique	1. Publié - Pour tous			25/10/2012	1
6	27/03/2015	Racine > Menu principal	Sélections	Rubrique	1. Publié - Pour tous			25/10/2012	2
7	27/03/2015	Racine > Menu principal	Infos pratiques	Rubrique	1. Publié - Pour tous			25/10/2012	3
8	27/03/2015	Racine > Menu principal > Animatior	Evénements	Rubrique	1. Publié - Pour tous			25/10/2012	4
9	27/03/2015	Racine > Menu principal > Animatior	Heure du conte	Rubrique	1. Publié - Pour tous			25/10/2012	5
10	27/03/2015	Racine > Menu principal > Infos prati	La bibliothèque	Rubrique	1. Publié - Pour tous			25/10/2012	9
11	27/03/2015	Racine > Menu principal > Infos prati	Inscription et prêt	Rubrique	1. Publié - Pour tous			25/10/2012	10
12	27/03/2015	Racine > Menu principal > Sélection	Nouveautés	Rubrique	1. Publié - Pour tous			25/10/2012	7
13	27/03/2015	Racine > Menu principal > Sélection	Coups de coeur	Rubrique	1. Publié - Pour tous			25/10/2012	8

Outils - Liste des champs personnalisés – enregistrés dans la table cms_editorial_custom – triés par numéro d'id

	A	B	C	D	E	F	G	H
1	Type de contenu	Id	Titre (affichage)	Champ (nom SQL)	Type de saisie	Type de données	Valeur	Contenu
2	Sélection	1	Etagère associée	etagere	Liste à partir d'une requête	Entier		
3	Evénement sur une période	2	Date de début	date_debut	Sélection de date	Date		
4	Evénement sur une période	3	Date de fin	date_fin	Sélection de date	Date		
5	Coup de coeur	4	Identifiant de la notice	identifiant_notice	Texte (saisie au clavier)	Texte court		
6	Evénement à une date précise	5	Date de l'événement	date	Sélection de date	Date		
7	CP pour Rubrique	8	Commentaires de gestion - Rubriques	comment_rubriques	Texte (saisie au clavier)	Texte court		
8	CP pour Article	9	Commentaires de gestion - Articles	comment_articles	Zone de commentaire	Texte court		
9	CP pour Article	10	Rédacteur	redacteur	Liste de choix préétablie	Entier	1	XX
10	CP pour Article	10	Rédacteur	redacteur	Liste de choix préétablie	Entier	2	XY
11	CP pour Article	11	Rédacteur (user)	redacteur_user	Liste à partir d'une requête	Entier		
12	CP pour Article	12	Dernière mise à jour	mise_a_jour	Sélection de date	Date		
13	Lien	13	URL	url	Texte (saisie au clavier)	Texte court		
14	Lien	14	Ouvrir dans un nouvel onglet	nouvel_onglet	Liste de choix préétablie	Entier	1	Oui
15	Lien	14	Ouvrir dans un nouvel onglet	nouvel_onglet	Liste de choix préétablie	Entier	2	Non

N.B. Ce sont les valeurs dans ma base de travail ; selon toute probabilité, certaines seront différentes dans votre base.
Toutefois, si vous avez restauré le jeu de sauvegarde cms_min, les 5 premiers champs sont identiques (ce sont les valeurs de la démo).

Pour les types de contenu, la signification de 2 libellés (« CP pour … ») pourrait intriguer de prime abord.

CP = champ personnalisé

CP pour Article = champ générique pour article
→ commun à tous les articles
CP pour Rubrique = champ générique pour rubrique
→ commun à toutes les rubriques

L'explication se trouve dans la table qui définit ces types de contenu : cms_editorial_types

	A	B	C
1	Id	Type de contenu	Type (nom SQL)
2	1	Rubrique	section
3	2	Sélection	article
4	3	Evénement sur une période	article
5	4	Information	article
6	5	Coup de coeur	article
7	6	Evénement à une date précise	article
8	7	CP pour Article	article_generic
9	8	CP pour Rubrique	section_generic

Sélection d'articles selon les types de contenu standards
Articles - Type de contenu : Coups de cœur - Titre abrégé de la notice (25 caractères) et numéro d'id
Articles - Type de contenu : Evénement à une date précise - Date de l'activité
Articles - Type de contenu : Evénement sur une période - Dates de début et de fin
Articles - Type de contenu : Information
Articles - Type de contenu : Sélection de notices - Nom de l'étagère associée et numéro d'id
Articles - Type de contenu : Vérification - Attention ! Le type de contenu n'est pas défini

Si vous avez restauré le jeu de sauvegarde cms_min (configuration du contenu éditorial), vous avez des types de contenu standards, et les 5 premiers éléments de la liste des champs personnalisés sont également standards.
Dans ce cas, ces requêtes sont immédiatement prêtes à l'emploi ! Sinon, il faudrait adapter les procédures.

Mêmes types de contenu

	A	B	C
1	Id	Type de contenu	Type (nom SQL)
2	1	Rubrique	section
3	2	Sélection	article
4	3	Evénement sur une période	article
5	4	Information	article
6	5	Coup de coeur	article
7	6	Evénement à une date précise	article
8	7	CP pour Article	article_generic
9	8	CP pour Rubrique	section_generic

5 champs personnalisés identiques : numéro d'id, nom SQL, type de saisie et de type de données

Il s'agit seulement des 5 premiers champs de la liste.

	A	B	C	D	E	F
1	Type de contenu	Id	Titre (affichage)	Champ (nom SQL)	Type de saisie	Type de données
2	Sélection	1	Etagère associée	etagere	Liste à partir d'une requête	Entier
3	Evénement sur une période	2	Date de début	date_debut	Sélection de date	Date
4	Evénement sur une période	3	Date de fin	date_fin	Sélection de date	Date
5	Coup de coeur	4	Identifiant de la notice	identifiant_notice	Texte (saisie au clavier)	Texte court
6	Evénement à une date précise	5	Date de l'événement	date	Sélection de date	Date

Ces requêtes sélectionnent les articles selon les types de contenu standards, et permettent des vérifications bien utiles.
Pour un article de type « coup de cœur », on voit l'identifiant de la notice, et son titre abrégé (25 caractères).
Pour un article de type « sélection », on voit l'identifiant et le nom de l'étagère associée à cet article.
Pour un article de type « événement », on voit la date de l'activité, ou bien les dates de début et de fin.
Une autre requête permet de repérer la présence éventuelle d'un article dont le type de contenu n'aurait pas été défini.

Coups de cœur

Id de la notice ? Annales ? Ou La 8e couleur ?

	A	B	C	D	E	F	G	H	I	J	K
1	Edité le	Rubrique	Titre de l'article	Type de conten	Id	Titre abrégé de la notice	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Sélec	Des bibliothèques pl	Coup de coeur	88	Des bibliothèques pleines	1. Publié - Pour tous			26/10/2012	17
3	12/04/2015	Menu principal > Sélec	Georges : petit-petit-	Coup de coeur	129	Georges	1. Publié - Pour tous			26/10/2012	12
4	12/04/2015	Menu principal > Sélec	Les annales du disque	Coup de coeur	104	La huitième couleur	1. Publié - Pour tous			26/10/2012	18

Après vérification, on voit que le n° d'id de la notice est correct : « La 8e couleur » est un des titres de la série « Les Annales ».

Sélection
Etagère associée à l'article

	A	B	C	D	E	F	G	H	I	J	K
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Id	Etagère	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Séle	Cinéma	Sélection	8	Nouveautés DVD	1. Publié - Pour			26/10/2012	16
3	12/04/2015	Menu principal > Séle	Documentaires	Sélection	6	Nouveautés documentaires	1. Publié - Pour			26/10/2012	14

Evénement sur une période
Dates ?

	A	B	C	D	E	F	G	H	I	J	K
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Date (début)	Date (fin)	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Anim	Cultures d'Afrique noire	Evénement sur une pér	01/12/2012	31/03/2013	1. Publié - Pour			25/10/2012	1
3	12/04/2015	Tests divers	test - événement 2	Evénement sur une pér			1. Publié - Pour			25/03/2015	26

Evénement à une date précise

Date ?

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Date (activité)	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Menu principal > Anim	Projections : journée Tim	Evénement à une date	21/12/2012	1. Publié - Pour tous			25/10/2012	11
3	12/04/2015	Menu principal > Anim	Lisez 5 fruits et légumes	Evénement à une date	09/01/2013	1. Publié - Pour tous			25/10/2012	2
4	12/04/2015	Tests divers	test - événement 1	Evénement à une date		1. Publié - Pour tous			25/03/2015	25

Vérification
Article sans type de contenu !!!

	A	B	C	D	E	F	G	H	I
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Créé le	Id
2	12/04/2015	Tests divers	Test n° 4		1. Publié - Pour tous			3/04/2015	30

Ces requêtes servent aussi à la fin du processus, quand on prépare la mise en ligne du portail, par exemple pour vérifier les n° d'id des notices ou des étagères (ces numéros ne sont probablement pas tous les mêmes dans la base de test et dans la base en production).

Sélection d'articles selon un champ personnalisé supplémentaire
Articles - Sélection : seulement les articles dont le rédacteur est connu - sur base d'une liste de choix préétablie
Articles - Sélection : seulement les articles dont le rédacteur est connu - sur base des logins de la table users
Articles - Sélection : seulement les articles qui ont un commentaire de gestion
Articles - Sélection : seulement les articles qui ont une date de mise à jour

Les procédures de cette série-ci sont « spécifiques », elles sont liées à des champs personnalisés supplémentaires dont les numéros d'id peuvent différer d'une base à l'autre (commentaire de gestion, date de la dernière mise à jour, nom du rédacteur).
Toutefois, si vous avez défini les champs personnalisés comme je l'ai fait (même type de saisie et même type de contenu), et si vous leur avez donné le même nom SQL (ex. nouvel_onglet), il n'y a aucun changement à faire ;-)

Sélection : liste partielle des articles – seulement s'il y a une date de mise à jour

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Mis à jour le	Créé le	Id
2	04/04/2015	Menu principal > Anima	Projections : journée Tin	Evénement à une date	2. Publié - Abonnés	11/05/2015	24/05/2015	25/03/2015	25/10/2012	11
3	04/04/2015	Menu principal > Infos	Mais que font les bibliotl	Information	4. A valider			26/03/2015	25/10/2012	7

Sélection : liste partielle des articles – seulement si le rédacteur est connu (liste préétablie de choix)

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Rédacteur	Créé le	Id
2	29/03/2015	Menu principal > Animations > Evér	Cultures d'Afrique noire c	Evénement sur une pério	1. Publié - Pour tous	13/04/2015	30/05/2015	XX	25/10/2012	1
3	29/03/2015	Menu principal > Animations > Evér	Projections : journée Tim	Evénement à une date pı	2. Publié - Abonnés	11/05/2015	24/05/2015	XY	25/10/2012	11
4	29/03/2015	Menu principal > Animations > Heu	Lisez 5 fruits et légumes	Evénement à une date pı	5. Archives	23/03/2014	23/03/2014	XX	25/10/2012	2

Sélection : liste partielle des articles – seulement si le rédacteur est connu (liste à partir d'une requête – contenu de la tables users)

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Rédacteur	Créé le	Id
2	04/04/2015	Menu principal > Sélections > Cou	Des bibliothèques pleines	Coup de coeur	1. Publié - Pour tous			Eric	26/10/2012	17
3	04/04/2015	Menu principal > Sélections > Cou	Les annales du disque-m	Coup de coeur	1. Publié - Pour tous			Florent	26/10/2012	18
4	04/04/2015	Menu principal > Sélections > Cou	Georges : petit-petit-petit	Coup de coeur	3. Brouillon	13/04/2015	30/04/2015	admin	26/10/2012	12
5	04/04/2015	Menu principal > Sélections > Cou	Poisson rouge	Coup de coeur	1. Publié - Pour tous	06/04/2015		Camille	25/10/2012	4

Sélection : liste partielle des articles – seulement s'il y a des commentaires de gestion

	A	B	C	D	E	F	G	H	I	J
1	Edité le	Rubrique	Titre de l'article	Type de contenu	Statut	A publier de	Jusqu'à	Commentaire de gestion	Créé le	Id
2	04/04/2015	Menu principal > Infos	Mais que font les bibliothé	Information	4. A valider			Il manque 2 photos pour illustrer l'article	25/10/2012	7
3	04/04/2015	Menu principal > Séle	Des bibliothèques pleines	Coup de coeur	1. Publié - Pour tous			Id de la notice dans la base en ligne : 1689	26/10/2012	17

Les requêtes qui figurent dans le groupe (Portail - Utilitaires de gestion) sont tout à fait standards, et fonctionnent dans toutes les bases. Aucune adaptation n'est nécessaire.

Portail - Utilitaires de gestion
Liste des versions du CMS – avec leur numéro identifiant
Supprimer une ou plusieurs versions du CMS antérieure(s) à la version sélectionnée – A exécuter dans l'onglet Administration
Vider le cache du portail (source : PMB Services) – A exécuter dans l'onglet Administration

Requête SQL –Liste des versions du CMS (avec leur numéro identifiant)

Ceci n'est pas une requête d'action ; elle affiche simplement la liste des versions enregistrées (avec leur numéro identifiant).
Cette requête permet de consulter cette liste à l'aise, avant de lancer la procédure qui supprime plusieurs versions du CMS.

Requête SQL - Supprimer une ou plusieurs versions du CMS antérieure(s) à la version sélectionnée

Au fil du temps, le nombre de versions du CMS augmente dans la table cms_version, et il devient utile de lancer un « grand nettoyage de printemps ». C'est particulièrement le cas pour la démo du portail, où le nombre de versions du CMS est fort élevé.

La requête de suppression fonctionne selon le principe suivant

- on sélectionne une version du CMS
- la requête supprime la (les) version(s) du CMS antérieure(s) à la version choisie
- elle conserve la version sélectionnée et les versions postérieures

Cette requête est surtout conçue pour lancer « un nettoyage par le vide » dans des versions anciennes du portail, mais **soyez prudent dès qu'il s'agit de supprimer des versions relativement récentes !!**

Pour que le portail fonctionne, il vous faut « la bonne version » du CMS : celle qui est active pour le moment (celle dont on voit le numéro dans les informations de l'OPAC). Or, c'est en général la version la plus récente qui est active (sauf si vous venez de restaurer une version antérieure).

Si on veut vérifier à l'aise, avant de lancer la suppression, et savoir à quelle version du CMS un numéro d'id correspond, il suffit d'employer une autre procédure qui donne la liste des versions du CMS, et leur numéro identifiant.

En effet, la liste des versions du CMS fournie par PMB mentionne la date et l'heure de sauvegarde, mais pas le numéro d'id de la version.

Attention donc, ne lancez pas cette procédure de manière inconsidérée ! Vérifiez d'abord !!

En rédigeant cette requête, j'ai veillé à ce qu'il soit techniquement impossible de supprimer la dernière version en date, puisque cette suppression déclencherait probablement de sérieux dysfonctionnements (selon toute vraisemblance, c'est celle qui est active).
Assurez-vous que la version du CMS qui est active pour le moment ne fasse pas partie de celles que la requête pourrait supprimer.

Ne soyez pas surpris par le nombre de lignes affectées par la requête, il est largement supérieur au nombre de versions supprimées.
En effet, à chaque ligne de la table cms_version correspondent plusieurs lignes dans la table cms_build (une ligne par cadre).
Or, la requête supprime les traces d'une version dans les deux tables concernées.

Exemple concret : s'il y a 20 cadres dans le portail, dès qu'on enregistre une version du portail, on aura

- une ligne supplémentaire dans cms_version : les « références » de cette version (numéro d'id, date et heure de sauvegarde)
- 20 lignes supplémentaires dans la table cms_build : une ligne par cadre qui fait partie de cette version-là

Donc si on supprime 3 versions du portail, on supprimera 63 lignes au total !

Dans cet exemple, j'ai demandé la suppression des 3 versions antérieures à la version 177, donc la requête va supprimer 3 lignes dans la table cms_version, mais 138 lignes dans cms_build : 141 au total.

Requête SQL – Vide le cache du portail (source : PMB services)

N.B. Je reprends seulement les infos essentielles, car j'ai déjà parlé du mode de mise en cache des cadres du portail.

C'est dans la table cms_cache_cadres que certains cadres sont mis en cache, afin de ne pas devoir être rechargés lors du prochain affichage.

Si vous avez effectué des changements relativement importants dans le portail, si vous avez ajouté ou modifié des articles, il est parfois utile que les lecteurs soient prévenus le plus tôt possible.
Mais si la durée de vie de votre cache est d'une heure ou deux, les lecteurs ne verront pas tout de suite l'impact de ces changements.

Dans ce genre de circonstances, il vaut mieux vider le cache « manuellement ». C'est le rôle de cette requête.

Les requêtes SQL spécifiques au portail sont disponibles sur mon site, il y a un zip à télécharger à cette adresse :
http://amcubat.be/docpmb/portail-construction-concepts-exemples

Lorsqu'on décompresse le zip, cela crée automatiquement des dossiers dans lesquels les requêtes sont réparties.

Pour savoir comment les télécharger et les importer dans PMB
http://amcubat.be/docpmb/telecharger-requetes-sql
http://amcubat.be/docpmb/importer-requetes-sql

N.B. Il faut importer toutes ces requêtes liées au portail dans l'onglet Administration.
Presque toutes les procédures peuvent être exécutées dans les onglets Administration ou Editions, sauf les 2 dernières que j'ai citées.

Il y a aussi sur mon site un ensemble de requêtes « générales » que vous pouvez télécharger, et importer dans PMB.
Vous trouverez ici 3 zips : http://amcubat.be/docpmb/requetes-sql/telecharger-et-importer

Lorsqu'on décompresse le zip, cela crée automatiquement des dossiers dans lesquels les requêtes sont réparties.

Il y a 3 zips car les requêtes sont groupées en fonction de l'onglet de PMB dans lequel il faut les importer :
administration, catalogue et circulation.

Pour savoir comment les télécharger et les importer dans PMB
http://amcubat.be/docpmb/telecharger-requetes-sql
http://amcubat.be/docpmb/importer-requetes-sql

Pour consulter la liste complète de cet ensemble de requêtes
http://amcubat.be/docpmb/wp-content/uploads//pmb-liste-requetes-sql.pdf

Pour savoir à quoi ces requêtes peuvent servir, suivre les liens suivants et parcourir les diverses rubriques
http://amcubat.be/docpmb/requetes-sql/sommaire-requetes-circulation
http://amcubat.be/docpmb/requetes-sql/sommaire-requetes-catalogue
http://amcubat.be/docpmb/requetes-sql/sommaire-requetes-administration

TABLE DES MATIÈRES

Construction du portail – Concepts et exemples

PMB – Construction du portail – Concepts et exemples 1
Préambule 3
Introduction 5
Conseils pratiques (1) 6
Travailler sur plusieurs bases 6
La version de démo du portail 9
Plusieurs bases sur le même serveur 9
Installer une base PMB 10
Organiser les sauvegardes 11
Sauvegarder – restaurer la configuration minimum 13
Particularités de la démo 14
Paramétrage du portail 16
Activer le module Portail 16
Autoriser l'accès à cet onglet 16
Paramétrer les URL de base 16
Paramétrage de la démo 17
Les étapes du processus 19
Conseils pratiques (2) 22
Phase d'apprentissage et de pré-tests 22
Journal de bord 23
Liste des changements à faire ultérieurement 23
Conseils pratiques (3) 24
Pendant la construction du portail 24
Transfert de la vraie base vers la base de travail 25
Notices, exemplaires et étagères 26
Configuration globale d'une base 27
Feuilles de style 28
Généralités 28
Choisir une feuille de style 29
Cascading Style Sheets 29
Conseils pratiques (4) 30
Changer de feuille de style 30
Quelques exemples d'OPAC 34
Un OPAC « classique » 35
Un OPAC « style portail » 36

Contenu éditorial 42
Configuration préliminaire 42
Numéros identifiants 47
Rédiger les rubriques et articles 49
Gérer l'arbre du contenu éditorial 55
Gérer globalement le contenu éditorial 58
Sauvegarder – restaurer le contenu éditorial 64
Pages de l'OPAC et du portail 65
Zones et pages 65
Pages de l'OPAC 66
Créer les pages du portail 67
Paramètres cms et first_page_params 69
URL avec variables et navigation 70
Dans quel ordre configurer les pages du portail ? 71
Modules, cadres et pages 72
Cadres de l'OPAC et du portail 72
Identifier et repérer les zones et cadres 75
Distinguer les modules et les cadres 77
Modules disponibles 78
Administration des modules 79
Analyser quelques exemples de cadres 80
Analyser les pages-outils du portail 112
Menu basé sur le contenu éditorial 132
Analyser la page d'accueil du portail 134
Templates Django – Synthèse 136
Propriétés CSS des cadres 143
Manipuler les cadres 146
Adapter son mode de travail au portail 154
Le champ « résumé » 154
Le logo des rubriques et articles 154
Bulletins et notices de bulletins 155
Exemples supplémentaires 157
Variantes de quelques cadres et pages 157
Liste de sites Web 166
Menu créé dans l'interface d'administration 172
Transfert du portail vers la vraie base 173
Préparatifs à effectuer dans la vraie base 173
Eléments à transférer depuis la base de travail 173
Autres ajustements nécessaires 174
Conclusion 174
Bibliographie 175
Aide-mémoire et tableaux de synthèse 176
Liens entre modules, cadres et pages 176

Templates Django 181
Balises HTML 184
Requêtes SQL utiles 186
Table des matières 193

Modules et cadres analysés de manière détaillée dans ce manuel

Chapitre : Modules, cadres et pages

- Cadres créés à partir du Module « Liste d'articles » → liste ou carrousel d'articles
  - Carrousel d'articles d'actualité
  - Sélection de nouveautés (étagères)
  - Liste de brèves
  - Articles dans les mêmes catégories qu'une notice
- Cadres créés à partir du Module « Liste de notices » → liste ou carrousel de notices
  - Carrousel de notices - Nouveautés
  - Notices d'une étagère associée à un article
  - Notices dans les mêmes catégories qu'un article
  - Notices du même auteur qu'une notice
- Les cadres principaux associés aux pages-outils du portail
  - Affichage d'une rubrique
  - Affichage d'un article
  - Affichage d'une notice
  - Affichage d'une liste d'événements – Module agenda
    - Configuration de l'agenda - insertion d'un agenda (vue en calendrier)
  - Affichage d'une liste d'articles = résultat d'une recherche dans le contenu éditorial
    - Configuration de la recherche - insertion d'une barre de recherche (catalogue + articles) - Module recherche
- Menu basé sur le contenu éditorial (titres des rubriques)

Chapitre : Exemples supplémentaires (non illustrés dans la démo)

- Variantes de quelques cadres et pages
  - Page du portail qui affiche une notice : ajouter un cadre « Notices du même auteur »
  - Page du portail qui affiche un article : un autre usage du cadre « Liste de notices – Etagère associée à un article »
  - Page du portail qui affiche un article : un autre mode d'affichage de « Notices dans les mêmes catégories qu'un article »
- Liste de sites Web
  - Ouvrir un lien externe dans un nouvel onglet
- Menu créé dans l'interface d'administration