

SPIP

- développement de squelettes -



Partie 1

Fonctionnement du noyau



Organisation des données



Les objets principaux de la BDD

Manipulés par les utilisateurs :

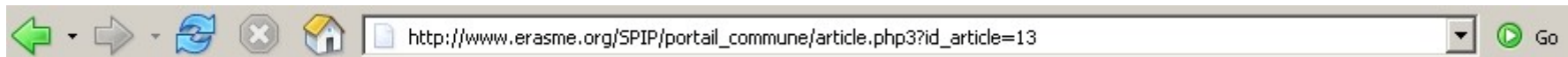
- les articles
- les brèves
- les images et les documents joints
- les rubriques
- les mots clefs
- les sites syndiqués
- les auteurs
- les messages internes / de calendrier
- les forums et les pétitions

Les autres objets

- Les objets créés automatiquement par le système :
 - **L'indexation du contenu** (moteur de recherche)
 - **Les statistiques** de visite
- Les méta-données :
 - **La version** du système
 - **Les options fonctionnelles** : forums, révisions, l'orthographe, prévisualisation, ect

Reconnaître un objet

- L'identifier parmi la liste des objets.
- Noter son numéro d'identifiant unique :
 - Visible dans l'URL du site public ou privé :



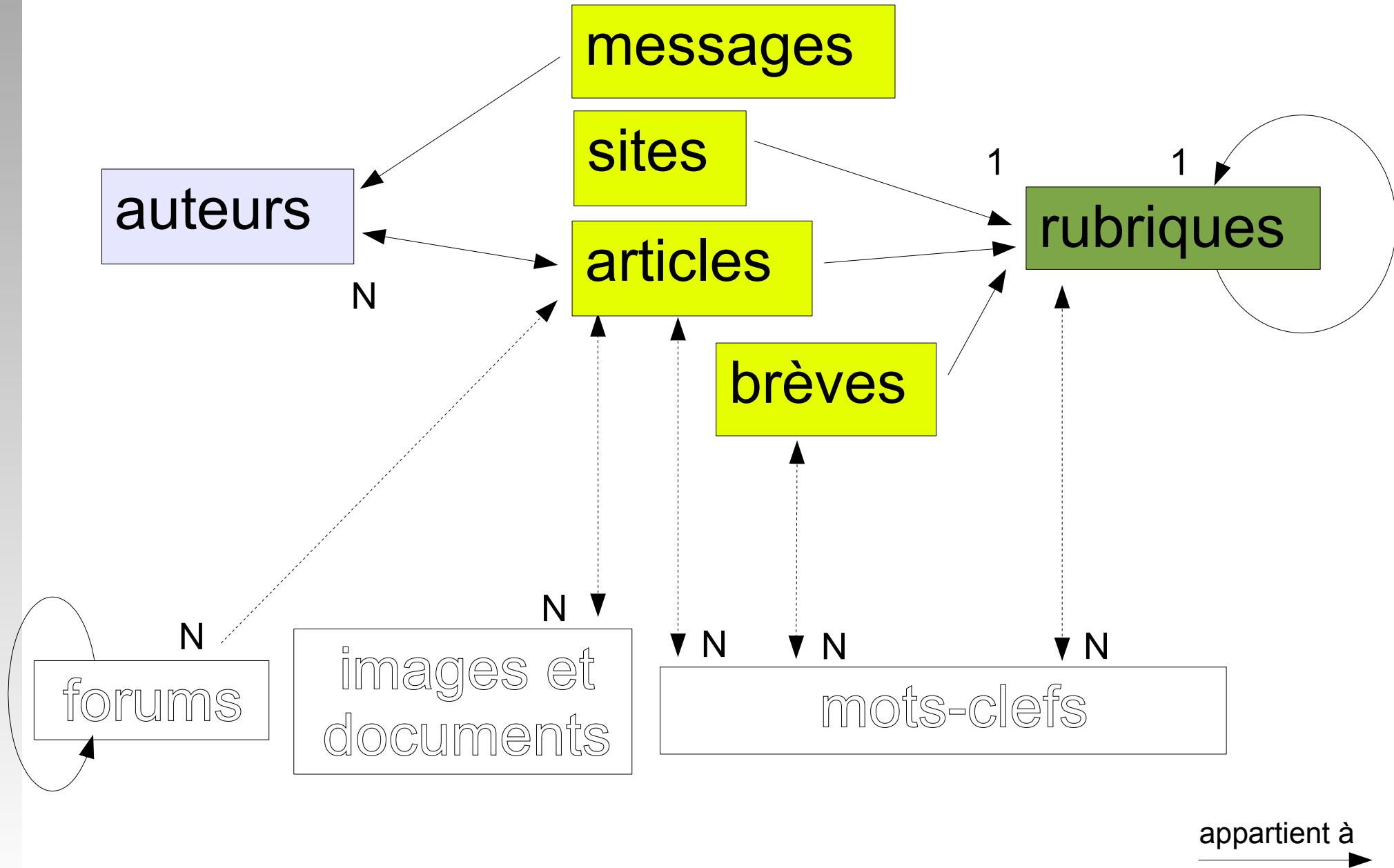
- Visible dans le Backoffice :

A screenshot of a backoffice interface for article management. It displays the following information:

- ARTICLE NUMÉRO :**
13
-  **Voir en ligne**
-  **Suivi du forum public :**
15 contribution(s)
-  **Évolution des visites**
20 visites

- Connaître l'interaction entre les objets de SPIP

Liens entre les objets SPIP



Extrait de la Base de Données

- table : spip_rubriques

	id_rubrique	id_parent	titre	descriptif	texte	id_secteur	maj	export	id_import	statut	date	lang
<input type="checkbox"/>	1	0	Contenu Editorial		[BLOB - 0 Octets]	1	2007-10-27 17:15:02	oui	0	publie	2007-02-20 16:52:45	fr
<input type="checkbox"/>	2	0	Vie pratique		[BLOB - 0 Octets]	2	2007-10-15 17:28:16	oui	0	publie	2007-10-15 17:28:16	fr
<input type="checkbox"/>	3	0	Vie culturelle		[BLOB - 0 Octets]	3	2007-08-02 16:52:45	oui	0	prive	0000-00-00 00:00:00	fr
<input type="checkbox"/>	5	0	Economie		[BLOB - 0 Octets]	5	2007-08-02 16:52:45	oui	0	prive	0000-00-00 00:00:00	fr
<input type="checkbox"/>	6	47	Tourisme		[BLOB - 0 Octets]	47	2007-08-02 16:52:45	oui	0	prive	0000-00-00 00:00:00	fr

- table : spip_articles

	id_article	s_titre	titre	sous_titre	id_rubrique	descriptif	chapo	texte	ps	date	statut	id_secteur	maj	export	date_redac	visites
<input type="checkbox"/>	1		Carte d'identité de la commune		1	Rapide descriptif de la commune		[BLOB - 160 Octets]		2007-02-20 16:52:45	publie	1	2007-10-27 17:09:36	oui	0000-00-00 00:00:00	44
<input type="checkbox"/>	2		Associations		2			[BLOB - 13,9 Ko]		2007-02-20 16:53:27	publie	2	2007-08-02 16:52:45	oui	0000-00-00 00:00:00	74
<input type="checkbox"/>	3		Les collègues		2			[BLOB - 1,6 Ko]		2007-02-20 16:53:28	publie	2	2007-10-27 17:09:50	oui	0000-00-00 00:00:00	62

- table : spip_auteurs

	id_auteur	nom	bio	email	nom si
<input type="checkbox"/>	1	pvincent		pvincent@erasme.org	
<input type="checkbox"/>	2	admin1			
<input type="checkbox"/>	9	Montpellier			

- table : spip_auteurs_articles (pivot)

	id_auteur	id_article
<input type="checkbox"/>	1	88
<input type="checkbox"/>	2	81
<input type="checkbox"/>	2	83
<input type="checkbox"/>	2	84
<input type="checkbox"/>	2	90

Du point de vue fonctionnel

- **les rubriques** déterminent l'organisation du contenu du site (structure logique)
- **les articles, brèves, sites, forums, documents** représentent ce contenu
- **les squelettes** déterminent la mise en forme du contenu sur le site public
- **les mots-clefs** peuvent servir à la fois à organiser le contenu de manière transversale et à moduler la mise en forme (si intégrés dans les squelettes)
- **les auteurs** structurent la chaîne rédactionnelle

Calcul des pages et modèles d'affichage



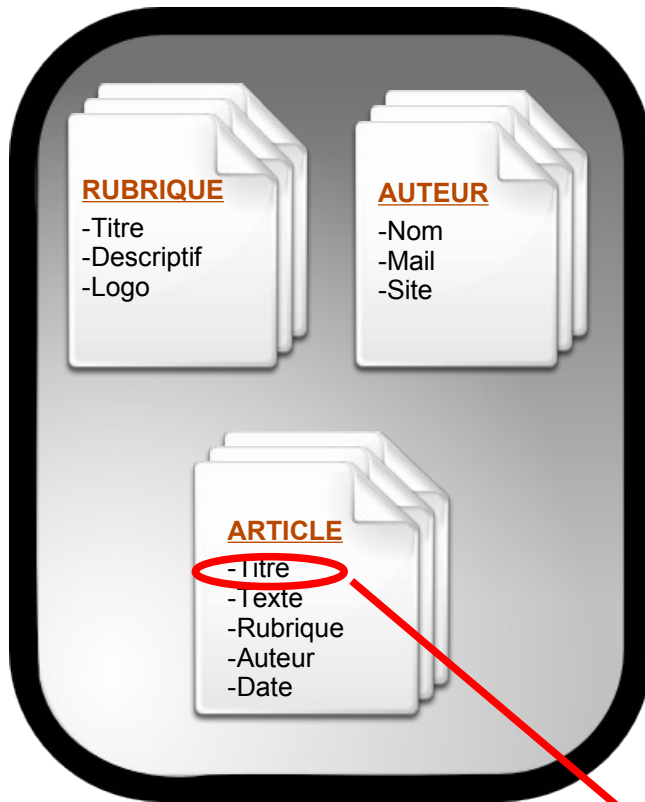
Deux principes pour l'affichage

- La mise en page du site est effectuée au moyen de pages HTML nommées **squelettes**, contenant des instructions simplifiées permettant d'indiquer où et comment se placent les informations tirées de la base de données dans la page.
- Un **système de cache** permet de stocker chaque page temporairement et ainsi d'éviter de provoquer des appels à la base de données à chaque visite.

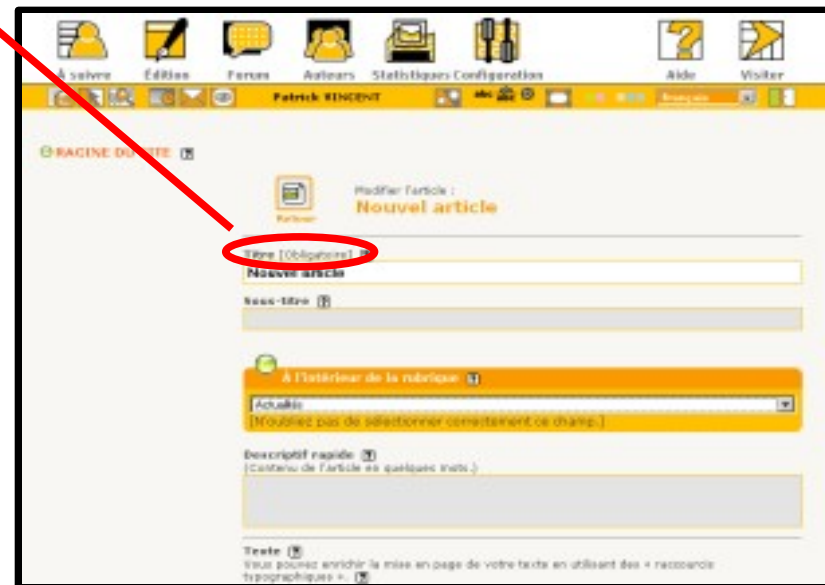
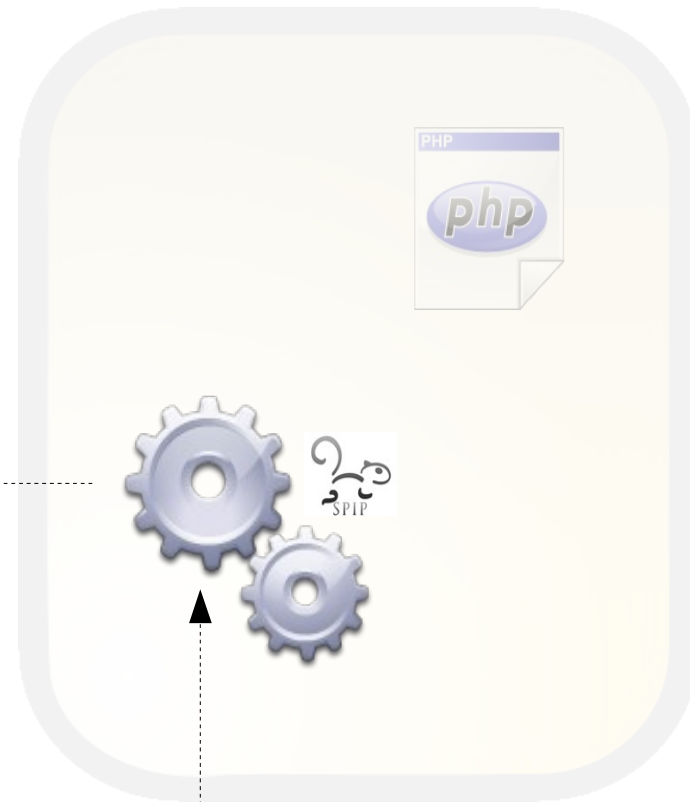
Les squelettes

- L'aspect graphique et la navigation des pages du site sont définis par des *squelettes* appelés aussi *modèles*, *canevas* ou «*templates*».
- Ils définissent chacun une «**vue**» :
 - une pour le **sommaire**
 - une pour une **rubrique** et le résumé de son contenu
 - une pour le détail d'un **article**
 - une pour le détail d'une **brève...**
- **Le système fabrique chaque page** individuelle à partir du canevas **en plaçant automatiquement le contenu lié à l'objet** (titre, texte, liens de navigation...) dans des **blocs**
 - En savoir plus : http://www.spip.net/fr_rubrique143.html

base de données

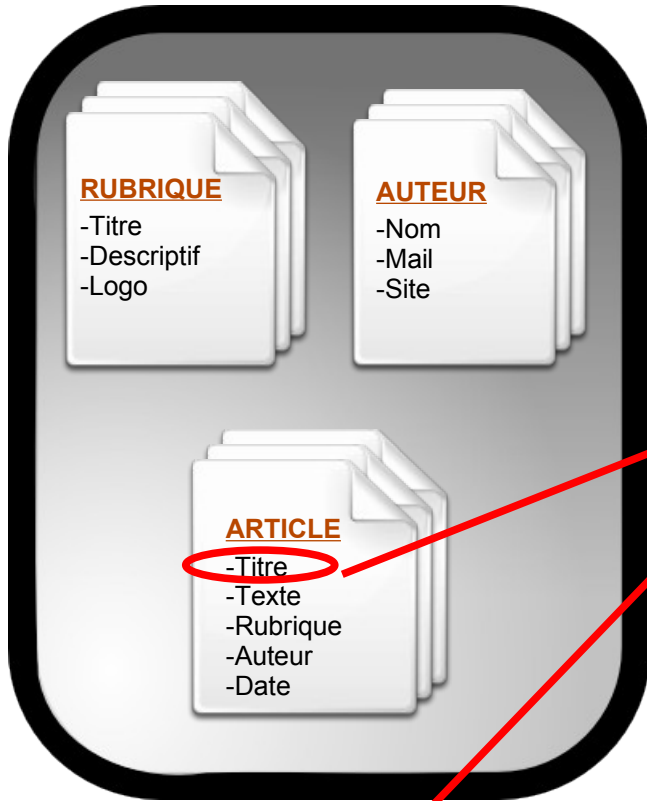


serveur web



Publication

base de données



squelette

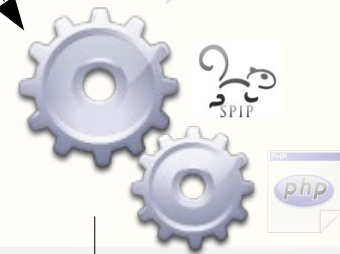


sommaire.html

feuille de style

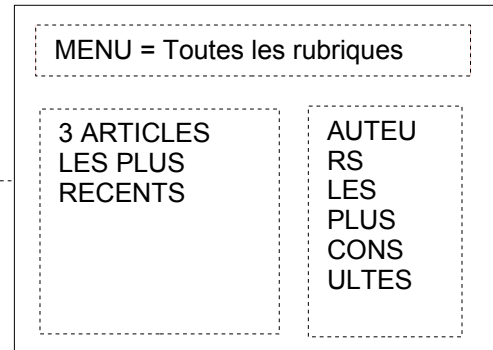
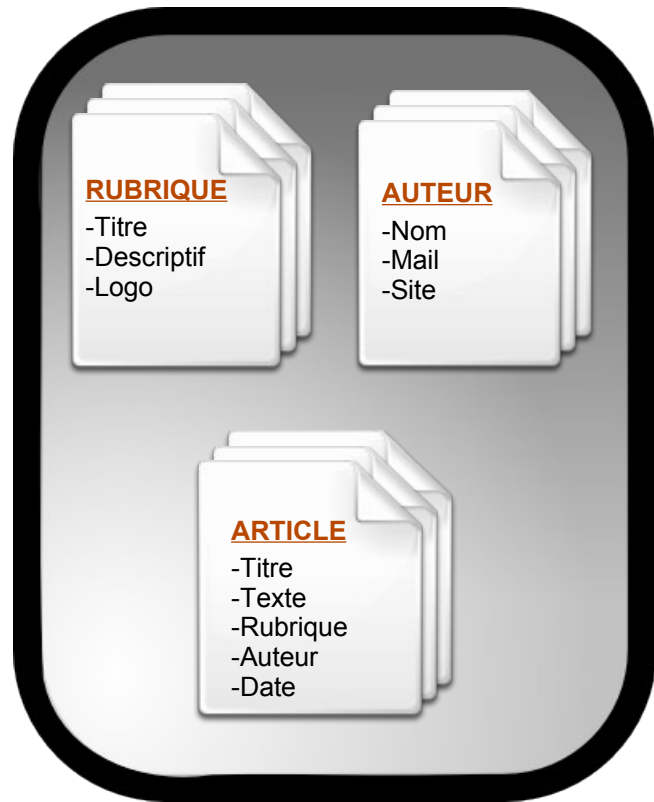


style.css

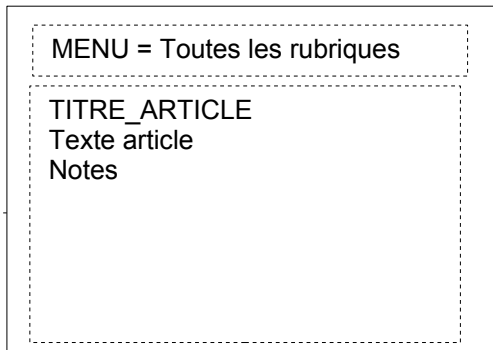


page affichée sur le site public

Une base pour plusieurs squelettes



sommaire



article

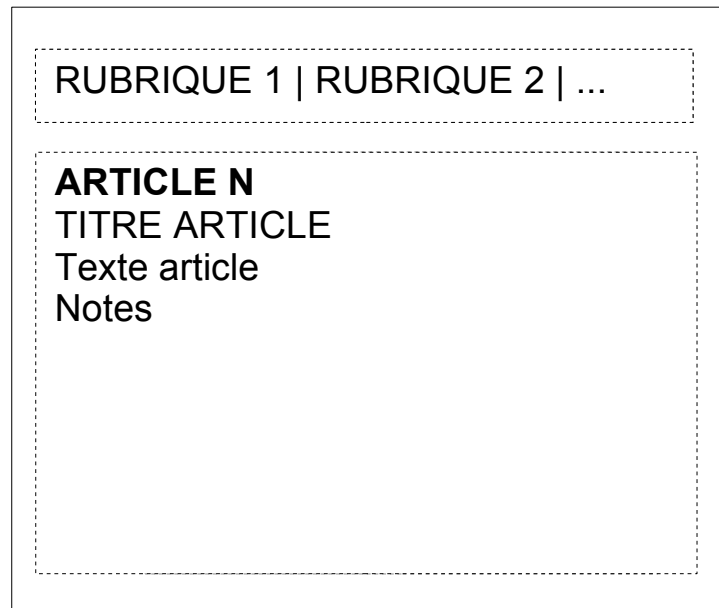


base de données

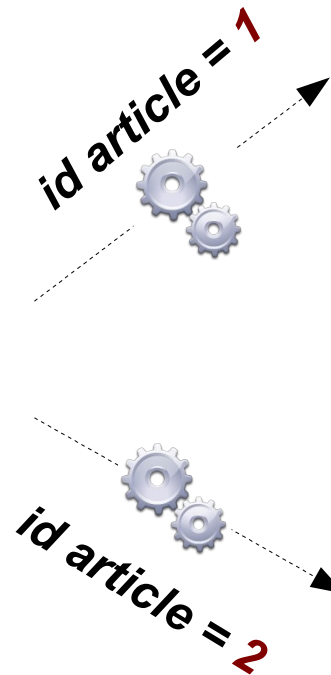
squelettes

pages HTML calculées

Un squelette pour plusieurs pages



squelette *article*



A final page for article 1. It has a red header bar with the text "CUISINE | CINEMA | SPORT". Below it is a yellow content area with the title "La tarte à la praline" and the text "Tout d'abord prendre 500 g de farine. La mélanger avec 100 g de beurre en ajoutant progressivement de l'eau. Puis (...)"

A final page for article 2. It has a red header bar with the text "CUISINE | CINEMA | SPORT". Below it is a yellow content area with the title "Wallace et Gromit et le lapin Garou" and the text "Le dernier film d'animation des réalisateurs anglais adeptes de pâte à modeler est (...)"

pages finales affichées

Correspondance des champs



espace privé

base de données

squelette

page publique

ARTICLES

[ACCUEIL](#) >> [Vie au vert](#) >> [Environnement](#)

Développement Durable

samedi 5 mars 2005 , par Mr Le Maire

Cet article regroupe plusieurs documents destinés à sensibiliser les visiteurs aux enjeux du développement durable et à réfléchir ensemble aux actions à mener pour infléchir la tendance actuelle.

Format PDF 

Format Imprimable 

Quelle planète pour les générations futures ?

Développement durable : un développement qui répond aux besoins du présent sans compromettre la capacité des générations futures à répondre aux leurs.



(définition extraite du rapport Notre Avenir à tous de la Commission Mondiale sur l'Environnement et le Développement -Commission dite Brundtland, du nom de Mme Gro Harlem Brundtland, ancienne Premier Ministre de Norvège, qui l'a présidée- 1987)

RECHERCHE

AUTRES ARTICLES

Nouvel article

Le mercredi 12 octobre 2005 par Patrick VINCENT

Tri sélectif

Le mercredi 19 janvier 2005 par Mr Le Maire

SUR LE WEB

[reseau.erasme.org]

- Consultation pour un proxy filtrant > 21 juin
- Proposer une fiche d'aide technique > 17 avril
- Comment redimensionner une image > 15 février



Commune de St-Clément les places

AGENDA

PLAN DU SITE

CONTRIBUER

S'INSCRIRE

Vie municipale

Vie pratique

Vie culturelle

Vie Associative

Vie économique

Vie au vert

Actualités

ARTICLES

ACCUEIL >> Vie au vert >> Environnement

ARBORESCENCE

Développement Durable

samedi 5 mars 2005, par Mr Le Maire

DATE AUTEUR

Cet article regroupe plusieurs documents destinés à sensibiliser les visiteurs aux enjeux du développement durable et à réfléchir ensemble aux actions à mener pour infléchir la tendance actuelle.

DESCRIPTIF

Format PDF

Format Imprimable

RECHERCHE

AUTRES ARTICLES

**AUTRES ARTICLES
DANS LA
MEME RUBRIQUE**

SUR LE WEB

**SITES
SYNDIQUES
DANS CETTE
BRANCHE**

Quelle planète pour les générations futures ?

Développement durable : un développement qui répond aux besoins du présent sans compromettre la capacité des générations futures à répondre aux leurs.



TEXTE

(définition extraite du rapport Notre Avenir à tous de la Commission Mondiale sur l'Environnement et le Développement -Commission dite Brundtland, du nom de Mme Gro Harlem Brundtland, ancienne Premier Ministre de Norvège, qui l'a présidée- 1987)

[Vie municipale](#)[Vie pratique](#)[Vie culturelle](#)[Vie Associative](#)[Vie économique](#)[Vie au vert](#)[Actualités](#)

Vie Pratique

Cette rubrique vous informera sur l'ensemble des démarches que vous pouvez effectuer auprès de notre municipalité. Nous espérons que les informations qu'elle contient vous permettront de vous orienter plus facilement dans vos choix et de gagner du temps.



Nous restons bien entendu à votre disposition par téléphone au 04.78.00.00.00

L'équipe municipale

RECHERCHE

SOUS-RUBRIQUES

[Démarches administratives](#)[Evènements estivaux](#)[Logement](#)[Nouvelles Technologies](#)[N° Utiles](#)[Plan](#)[Prévention et sécurité](#)[Réservation des salles municipales](#)[Social](#)[Transports](#)[Vie scolaire - Petite enfance](#)

ARTICLES

ACCUEIL >> Vie pratique

Social-Santé

Le mercredi 19 janvier 2005 par Mr Le Maire

Le premier devoir de la Mairie est d'améliorer les services rendus aux grassois et d'accompagner ceux qui en ont besoin. Le C.C.A.S. (Centre Communal d'Action Sociale), présidé par Monsieur le Maire, est l'outil privilégié de la politique sociale municipale : il gère des services nombreux et variés qui s'adressent aussi bien aux plus (...)

[Lire la suite >](#)

Le temps d'une chanson

Le mardi 18 janvier 2005 par Mr Le Maire



Commune de St-Clément les places

RUBRIQUE

AGENDA

PLAN DU SITE

CONTRIBUER

S'INSCRIRE

Vie municipale

Vie pratique

Vie culturelle

Vie Associative

Vie économique

Vie au vert

Actualités

Vie Pratique

Cette rubrique vous informera sur l'ensemble des démarches que vous pouvez effectuer auprès de notre municipalité. Nous espérons que les informations qu'elle contient vous permettront de vous orienter plus facilement dans vos choix et de gagner du temps.



Nous restons bien entendu à votre disposition par téléphone au 04.78.00.00.00

L'équipe municipale

RECHERCHE

SOUS-RUBRIQUES

Démarches administratives

Evènements estivaux

Logement

Nouvelles Technologies

N° Utiles

Autres

Prévention et sécurité

Reservation des salles municipales

Social

Transports

Vie scolaire - Petite enfance

SOUS-ARBORESCENCE DE LA RUBRIQUE

ARTICLES

ACCUEIL >> Vie pratique

ARBORESCENCE

Social-Santé

Le mercredi 19 janvier 2005 par Mr Le Maire

Le premier devoir de la Mairie est d'améliorer les services rendus aux grassois et d'accompagner ceux qui en ont besoin. Le C.A.S. (Centre Communal d'Action Sociale), présidé par Monsieur le Maire, est l'outil privilégié de la politique sociale municipale : il gère des services nombreux et variés qui s'adressent aussi bien aux plus (...)

[Lire la suite >](#)

DERNIERS ARTICLES DE LA RUBRIQUE

Le temps d'une chanson

Le mardi 18 janvier 2005 par Mr Le Maire

Le système de Cache

Le **système de cache** « construit » et stocke chaque page à des intervalles réguliers. Il évite ainsi son recalcul à chaque visite d'un utilisateur.

- La base de données n'est pas sollicitée à chaque visite
- La charge sur le serveur est réduite et la vitesse d'affichage des pages très largement accélérée
- Le site, par exemple sous SPIP, reste consultable même lorsque la base mySQL est en panne.

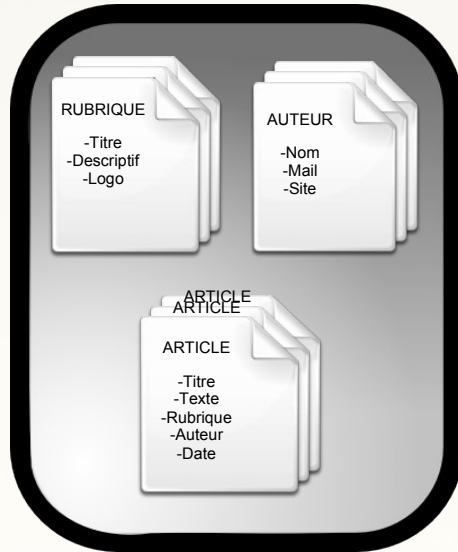
Vider le cache manuellement :



Vider le cache

[Recalculer cette page](#)

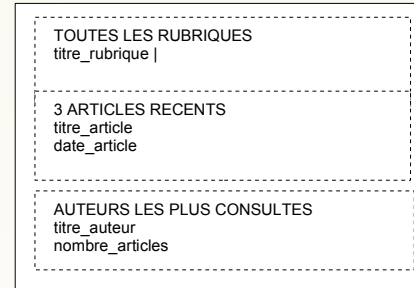
base de données



feuille de style



squelette



sommaire.html



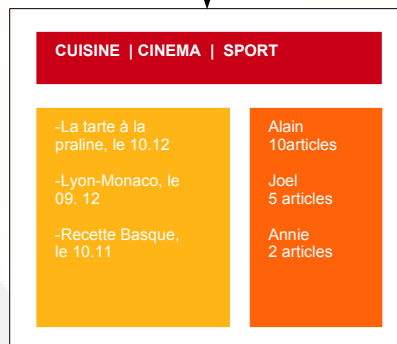
l'utilisateur appelle la page

article12

SPIP vérifie dans le cache si le fichier existe et n'est pas périmé

non

oui



page HTML / stockage dans le cache



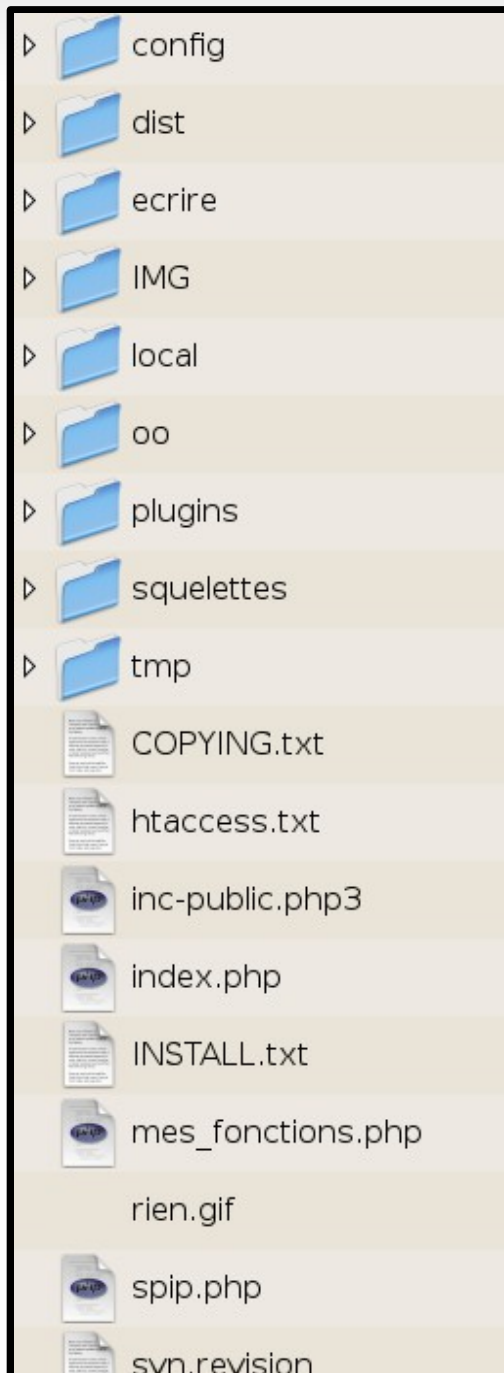
3

1

Le système de fichiers du noyau

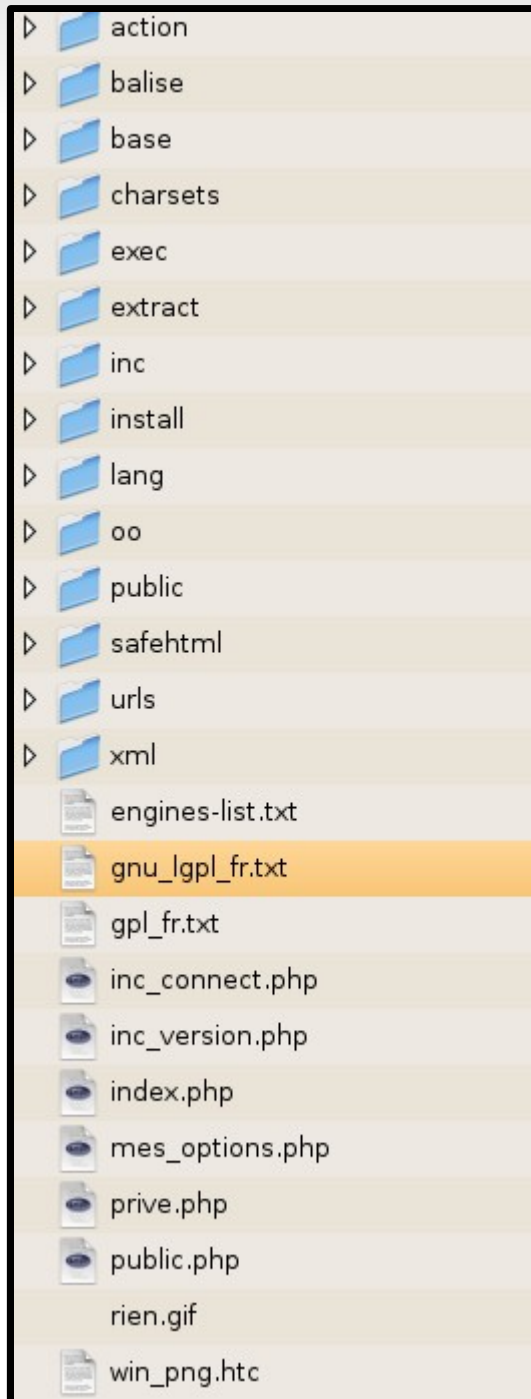


Le système de fichiers : racine



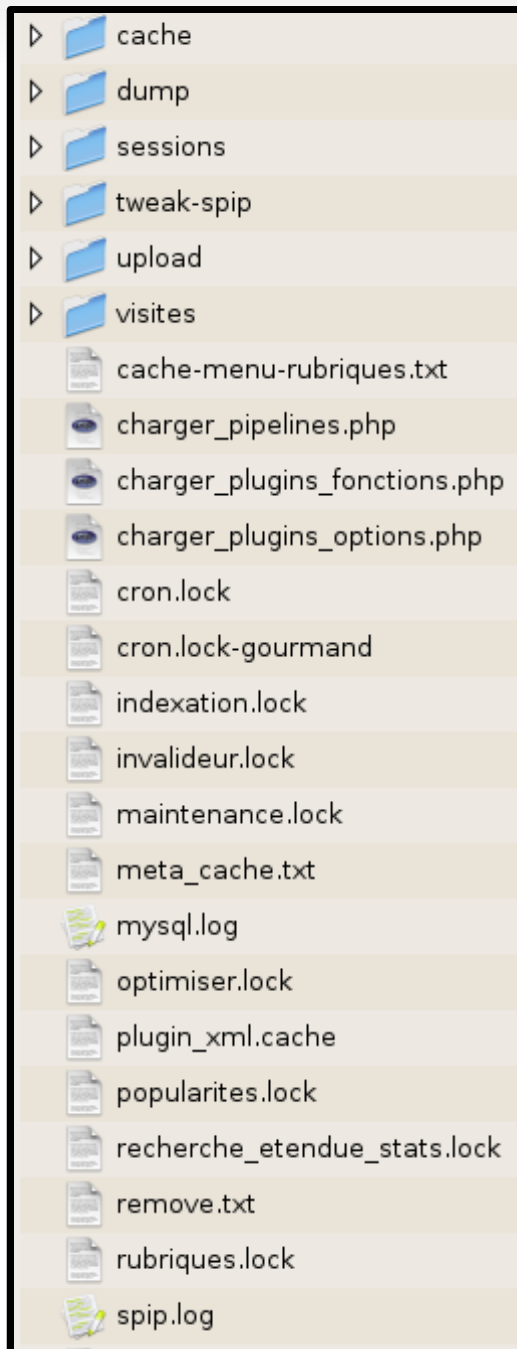
- ***/config*** : Fichiers de configuration : *mes_options.php, connect.php*
 - ***/dist*** (<v2.0>) ***/squelettes-dist*** : Jeu de squelettes et css par défaut
 - ***/ecrire*** : Noyau du système de publication
 - ***/prive*** (v2.0 >) : Jeu de squelettes du backoffice
 - ***/oo*** : Affichage texte seul
 - ***/IMG*** : Documents multimédias liés aux articles
 - ***/plugins*** : Plugins du site
 - ***/squelettes*** : Squelettes utilisateurs
 - ***/tmp*** : Fichiers temporaires
-
- ***index.php*** > ***spip.php*** : fichiers d'index
 - **svn.revision** : n° de version
 - ****.txt*** : à lire...
 - ***(mes_fonctions.php)*** : fonctions utilisateur

Le système de fichiers : /ecrire



- ***/action /exec*** : fichiers de fonctionnement de l'espace privé
 - ***/public /balise*** : calcul des pages du site public
 - ***/inc*** : bibliothèque de fonctions
 - ***/lang*** : fichiers de langue et de traduction (backoffice et frontoffice)
-
- ***Version < 1.9***
 - ***inc_connect.php*** : codes d'accès à la BDD
 - ***mes_options.php*** : options d'utilisation / définition des champs extra

Le système de fichiers : /tmp



- **/cache** : fichiers de cache du site
 - **/dump** : copies des bases de données (import / export)
 - **/upload** : stockage des arborescences de documents disponibles dans l'espace privé
 - **/visites** : données temporaires liées aux statistiques de consultation du site
-
- ***.log** : fichiers de log d'exécution du site
 - pour le php : voir dans `/var/log/apache2`

Squelettes disponibles nativement

description	nom	paramètre
page de détail d'un article	article	id_article
flux RSS	backend	id_rubrique (opt)
flux RSS brèves	backend-breves	id_rubrique (opt)
page de détail d'une brève	breve	id_breve
page de contribution au forum	forum	id_forum
page de détail d'un mot clef	mot	id_mot
plan du site	plan	
page de résultat d'une recherche	recherche	recherche
page de détail d'une rubrique	rubrique	id_rubrique
page de détail d'un site syndiqué	site	id_syndic
page d'accueil du site	sommaire	

Accès aux pages

- sans paramètres :
spip.php?page=plan
- avec paramètres :
spip.php?page=article&id_article=3
OU
spip.php?article3

Accès aux pages

- Appel du squelette *vue1*

spip.php?page=vue1

- Appel du squelette *vue1* avec le paramètre *param1=xxx* :

spip.php?page=vue1¶m1=xxx

- Appel du squelette *vue1* avec l'identifiant *id_vue1=xxx* :

spip.php?vue1xxx

- Paramètres spécifiques

- recalcul de la page : *var_mode=calcul*
- mode debug : *var_mode=debug*
- mode prévisualisation : *var_mode=preview*
- mode recherche : *var_recherche=xxx*

Fonctionnement du squelette

- Fichier HTML ajouté d'un langage de boucles et de balises interprété sur le serveur

Temps
de mise
en cache

```
#CACHE{24*3600}
```

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
Titre :
```

Boucle
Balise

```
<BOUCLE_un(ARTICLES) {id_article}>
```

```
#TITRE
```

```
</BOUCLE_un>
```

```
<p>texte texte texte</p>
```

```
</body>
```

```
</html>
```

article.html

- N'importe où dans la page : <head>, <body>, ...

Le webmaster et les squelettes

- **Peut-on utiliser un éditeur textuel pour créer et modifier ses squelettes ?**
 - Oui, comme on le ferait pour du HTML classique.
- **Peut-on utiliser un éditeur graphique (WYSIWYG) pour créer et modifier ses squelettes ?**
 - Oui, néanmoins, certains éditeurs graphiques modifient les tags SPIP et « corrigent » automatiquement les tags qu'ils ne comprennent pas. La plupart ont toutefois une option permettant de désactiver cette fonctionnalité.

Choix du squelette

- **SPIP cherche le fichier .html dans l'ordre suivant :**
 - 1. dans le dossier (ou la liste) désignés dans la variable ***\$dossier_squelettes*** si elle existe
 - 2. dans le dossier ***squelettes/*** situé à a racine
 - 3. dans la liste des **dossiers de plugins** installés (variable *\$plugins*)
 - 4. à la ***racine du site***
 - 5. dans le répertoire ***dist/***
 - 5. dans le répertoire ***ecrire/***
- **La surcharge de tous les éléments du site s'effectue dans l'ordre suivant:**
 - **ecrire > dist > racine > plugins > squelettes > mes_options**

Un squelette pour une branche

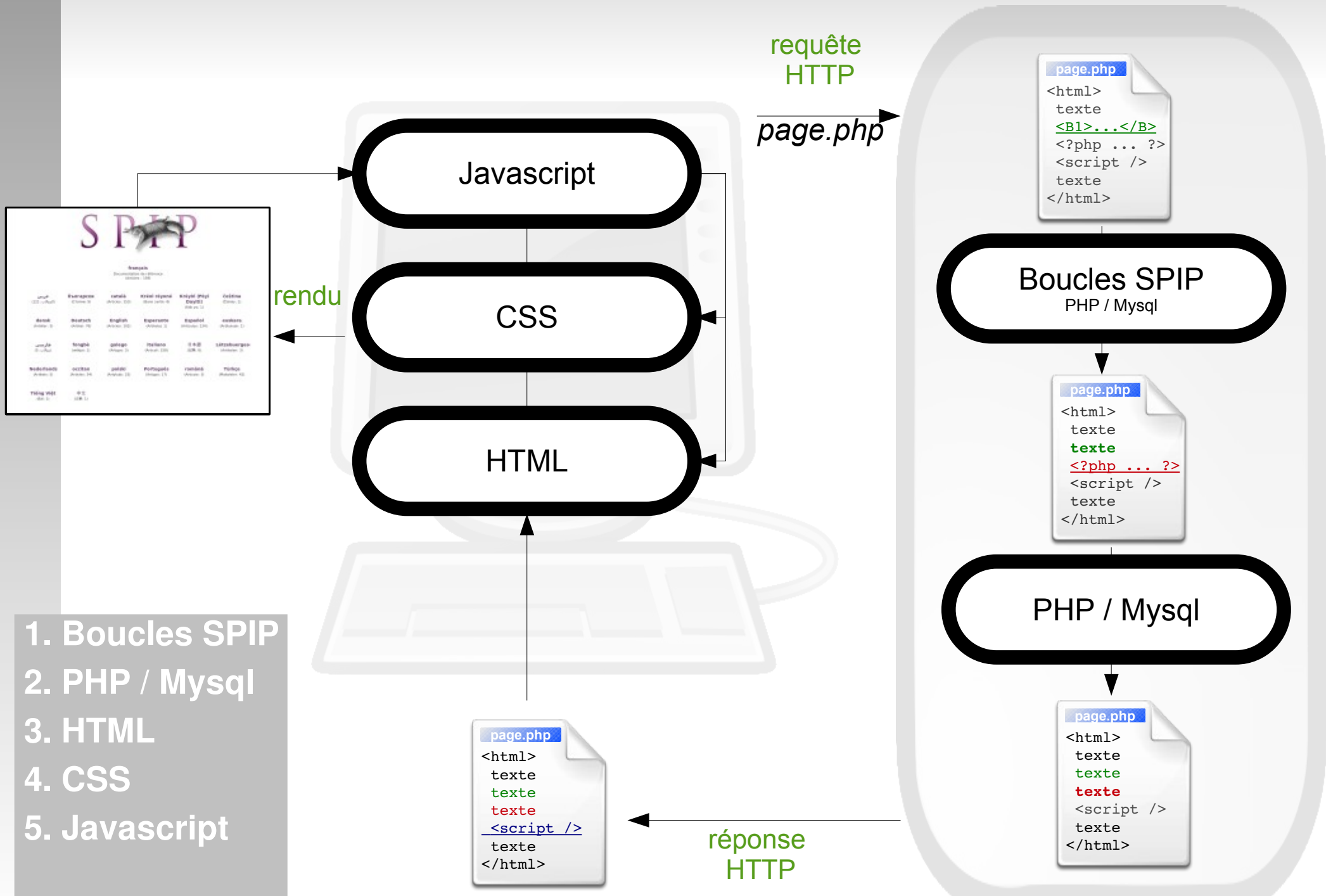
- **On peut créer un squelette spécifique pour une branche du site. Par exemple :**
 - Si le fichier *article-15.html* existe, il sera utilisé pour tous les articles de la rubrique 15 et de ses filles à la place du fichier *article.html*.
 - De même pour *breve-15.html*, *rubrique-15.html*,
- **Si on a seulement besoin d'une modification du style de la page (et non des blocs fonctionnels), préférer un simple changement de la feuille CSS**
 - test à faire dans le `<head>`

Ordre de traitement des instructions

- **Au niveau du serveur**
 - **1. Langage SPIP** : Le noyau SPIP interprète les boucles à l'intérieur des fichiers HTML et remplace les balises par leur contenu tiré de la base de données.
 - **2. Langage PHP** : les instructions PHP éventuelles s'exécutent dans la page et Apache sert la page html au navigateur client.
- **Au niveau du navigateur**
 - **3. Langages clients** : Javascript et les langages clients s'exécutent dans le navigateur

Note : On ne peut pas inclure de variables PHP dans le langage SPIP (critères de boucles par exemple).

Ordre d'exécution des instructions



Partie 3

Le langage de boucles de SPIP



Boucles, critères, filtres et objets



Syntaxe d'une boucle

OBJET
DE LA REQUÊTE

CRITERES
DE REQUÊTE

SELECTION ET ORDRE
DES RESULTATS

```
<BOUCLE_nom(OBJET) {critères} {filtres}>
```

...

```
<strong>#BALISE</strong>
```

...

```
</BOUCLE_nom>
```

RESULTATS

Mon premier squelette

Je recherche tous les
ARTICLES publiés

```
#CACHE { 0 }
```

```
...
```

```
<body>
```

```
<BOUCLE_une (ARTICLES) >
```

```
<h4>#TITRE</h4>
```

J'affiche le TITRE

```
</BOUCLE_une>
```

```
</body>
```

squelette1.html

- Tout le code situé dans la boucle est affiché dans la page autant de fois que d'occurrences sont trouvées (0 à n)
- Les balises *#BALISE* sont remplacées par leurs valeurs issues de la base de données

- A l'aide de votre éditeur de texte créer votre squelette *prenom1.html* avec un cache nul.
- Sur le site de travail, déposer le par FTP dans le nouveau dossier ***/squelettes/formation*** après avoir modifié le répertoire d'appel dans *mes_options.php*
- Appeler le squelette depuis un navigateur :
<http://www.monsite.net/spip.php?page=squelette>
- Afficher la source du fichier html calculé par le serveur.
Le comparer au fichier de squelette avant calcul.
Conclusion ?

Des balises propres à chaque objet

```
<BOUCLE_nom(OBJET)>  
  #BALISE1  
  #BALISE2  
</BOUCLE_nom>
```

Balises principales pour l'objet ARTICLES

- #ID_ARTICLE
- #TITRE
- #SOUSTITRE
- #TEXTE
- [(#TEXTE|couper{50})]
- #DATE
- #ID_RUBRIQUE
- #URL_ARTICLE
- #LESAUTEURS
- #LOGO_ARTICLE
- #LOGO_ARTICLE_RUBRIQUE

Exemple

- ajout des liens cliquables vers les articles du site

```
<body>
```

```
  <BOUCLE_une(ARTICLES)>
```

```
    <h4>
```

```
    <a href="#URL_ARTICLE">#TITRE</a>
```

```
    </h4>
```

```
  </BOUCLE_une>
```

```
</body>
```

squelette2.html

- Compléter votre premier squelette :
 - pour chaque article :
 - ajouter l'affichage de la date de publication
 - ajouter le logo de l'article
 - ajouter le début du texte de l'article (100 premiers caractères)
 - ajouter le nom de l'auteur ou des auteurs
 - dans chaque cas penser à encadrer les balises SPIP par des balises xhtml adaptées au contenu

Les critères et les filtres de boucles

prendre les 10 premiers

classés par ordre de publication

Je recherche les ARTICLES

publié(s) directement dans la rubrique 1

```
<BOUCLE_premiere(ARTICLES) {id_rubrique=1}
  {0,10}{par date}>
```

...

```
<h1>#TITRE</h1>
```

Affiche le TITRE

...

```
</BOUCLE_premiere>
```

- Les critères de boucles précisent la requête
- Les filtres de sélection et de tri organisent le résultat

Les critères de boucles (opérateurs)

```
<BOUCLE_nom(OBJET) {critère1} {critère2} ... >
```

- Hérité du contexte (égalité)

- {critère}

```
{id_rubrique}
```

- Comparaison directe

- {critère=valeur}

avec <,=,>=,<=,==,!=

```
{id_rubrique>3}
```

- Union de valeurs

- {critère IN v1,v2,...}

```
{id_rubrique IN 1,2,45}
```

- Comparaison à des variables

- {critère=#BALISE}
- {critère=#GET{var}}
- {critère=#ENV{var}}

```
{id_rubrique=#ID_AUTEUR}
```

```
{id_rubrique=#ENV{id_rubrique}}
```

Les critères de boucles *(nature)*

```
<BOUCLE_nom(OBJET) {critère1} {critère2?} ... >
```

- Des critères généraux
 - Arborescence
 - {id_rubrique=}
 - {id_secteur=}
 - {branche} *
 - Date
 - {age<n} (en jours)
 - {mois=}
 - {annee=}
- Des critères propres à chaque objet
 - ARTICLES
 - {id_article}
 - {id_rubrique}
 - {titre_mot==}
 - RUBRIQUES
 - {id_rubrique}
 - {id_parent}
 - {id_enfant}

* : pas d'opérateur direct possible -> hérité du contexte

Exemple avec plusieurs critères

- seulement les articles de la rubrique du contexte dont l'auteur n'est pas 1 :

```
<body>
  ...

  <BOUCLE_une(ARTICLES) {id_rubrique} {id_auteur!=1}>
    <h4>
      <a href="#URL_ARTICLE">#TITRE</a>
    </h4>
  </BOUCLE_une>

  ...
</body>
```

Les expressions régulières

- **{critere=[expr]}**
- Exemple : seulement les articles dont le titre ne commence par a ou A :

```
<body>  
  
  <BOUCLE_une(ARTICLES) {titre!=[^aA]}>  
    <h4>  
      <a href="#URL_ARTICLE">#TITRE</a>  
    </h4>  
  </BOUCLE_une>  
  
</body>
```

Les filtres de sélection

```
<BOUCLE_nom(OBJET) {selection} {tri} >
```

- {début,nombre}

- affiche *nombre* éléments à partir de *début*

```
{0,5}
```

```
{4,2}
```

- {debut_xxx,nombre}

- idem avec *debut_xxx* passé dans l'URL

```
{debut_photo,1}
```

- {page/total}

- affiche *1/total* des éléments à partir de *page*

```
{1/5}
```

```
{2/5}
```

- {doublons}

- n'affiche pas tous les éléments de la même nature rencontrés dans d'autres boucles avec {doublons}

```
{doublons}
```

- {pagination}

- pagine les résultats

```
<BOUCLE_page(ARTICLES) {pagination}>#TITRE</BOUCLE_page>  
#PAGINATION  
</B_page>
```

Les filtres de tri

```
<BOUCLE_nom(OBJET) {selection} {tri} >
```

- Les filtres de tri généraux :
 - {par titre}
 - {par hasard}
 - {par date}
- Les critères de tri particuliers (aux articles) :
 - {par visites}
 - {par date_redac}
 - {par popularite}
- Opérateurs
 - {par num critère}
 - {!par critère}
 - {par critère} {inverse}
 - {par critère1, critère2, ...}

Exemple avec sélection et tri

- affiche les dix derniers articles classés du plus récent au plus ancien :

```
<body>

    <BOUCLE_une(ARTICLES) {0,10} {par date} {inverse}>
        <h4>
            <a href="#URL_ARTICLE">#TITRE</a>
        </h4>
    </BOUCLE_une>

</body>
```

- Compléter votre squelette
- avec des critères de boucle :
 - Afficher les articles dont vous êtes l'auteur
 - Afficher les articles de votre rubrique de publication
 - Afficher les articles de votre branche de publication et de celle de votre voisin de droite
- avec des filtres de tri et de sélection :
 - Afficher les 3 derniers articles, par ordre de publication inverse
 - Afficher le 2ème et le 3ème article les plus populaires
 - Afficher le premier tiers des articles classés par ordre alphabétique

Les filtres de balises

```
<BOUCLE_premiere(OBJETS) {critere}{filtre}>
  [ ( #TITRE | couper{50} ) ]
  [ <p> ( #SOUSTITRE ) </p> ]
  [ <p> ( #SURTITRE | sinon{ 'nosurtitre' } ) </p> ]
</BOUCLE_premiere>
```

- Le code situé avant et après la parenthèse du filtre n'est affiché que si la balise est non vide (renseignée dans l'espace privé)
- Les filtres s'appliquent à la balise avec la syntaxe :
|filtre{paramètre}
- Si il y a plusieurs filtres, ils s'appliquent de gauche à droite

Les filtres de balises

```
[ (#BALISE|filtre1|filtre2...)]
```

```
[ texte_optionnel_avant (#BALISE|filtre) texte_optionnel_après ]
```

- Filtres de texte
 - majuscules
 - justifier, aligner_droite, aligner_gauche
 - centrer
 - textebrut
 - couper{taille}
 - ligne_longue{taille}
 - supprimer_numero
 - nl2br (et toute autre fonction php...)
- Les filtres des dates
 - affdate
 - jour
 - mois
 - annee
 - nom_jour
 - nom_mois
 - saison

Filtres d'images

- Note : les filtres appliqués aux logos nécessitent ||
 - `image_reduire{largeur, hauteur}`
 - rééchantillonnage et stockage dans /local
 - la plus petite des valeurs atteintes est prise en compte
 - `image_recadre{largeur, hauteur, position}`
 - `image_aplatir{format, couleur_fond}`
 - `image_gamma{valeur}`
 - `image_typo{police, couleur, taille}`
- } nécessitent GD2

```
[ (#LOGO_ARTICLE | left | | reduire_image{50, 100} ) ]
```

```
[ (#TITRE | image_typo{police=stencil.ttf, couleur=000000, taille=40} ) ]
```

Filtres conditionnels

- `|?{valeursioui}`
- `|sinon{valeursinon}`
- `|?{valeursioui,valeursinon}`
- `|=={valeur}`
- `|!={valeur}`
- `|>{valeur}`
- `|>={valeur}`
- `|<{valeur}`
- `|<={valeur}`

```
[ (#NOM_SITE_SPIP|=={'site1'}|?{style.css}) ]
```

```
[ (#TITRE|=={'edito'}|?{edito#ID_RUBRIQUE,texte}) ]
```

```
[ (#TITRE|=={'edito'}|?{' ',''})  
Bloc complexe incluant d'autres opérations ]
```

Filtres techniques

- `insérer_attribut{attribut,val}`
 - insere un attribut XHTML aux balises calculées par SPIP
- `extraire_attribut{attribut}`
 - extrait l'attribut XHTML des balises calculées par SPIP
- `parametre_url{parametre,val}`
 - rajoute un paramètre à l'URL passée en argument

```
[ (#LOGO_ARTICLE | |insérer_attribut{'width', '50%' }) ]
```

```
[ (#LOGO_ARTICLE | |extraire_attribut{'src' }) ]
```

```
[ (#SELF |parametre_url{'id_article', '12' }) ]
```

Filtres techniques (2)

Rendre le texte :

- **entites_html**
 - utilisable dans un formulaire
- **texte_script**
 - utilisable entre ' ' pour js ou php
- **attribut_html**
 - utilisable entre " " pour xhtml
- **texte_backend**
 - utilisable dans un fichier XML (RSS)
- **abs_url**
 - transforme une URL rel. en absolu

```
[<textarea>(#DESCRIFTIF | entites_html)</textarea>]
```

```
txt = '[(#TEXTE | texte_script)]';
```

```
<a href="#URL_ARTICLE" [title="( #DESCRIFTIF | textebrut | attribut_html)"] >
```

```
[ (#TEXTE | texte_backend) ]
```

```
[ (#LOGO_ARTICLE | abs_url) ]';
```

- A l'intérieur de votre squelette mettre en forme les informations de chaque article grâce aux filtres :
 - Mettre en forme la date : le mercredi 20 avril 2006
 - Afficher le titre des articles en majuscules et sans le numéro précédent l'article
 - Ajouter l'affichage du sous-titre précédé de « sous-titre : »
 - Rendre le titre cliquable vers l'url de l'article avec le texte coupé à 50 en titre
 - Si il y a un logo :
 - redimensionner l'image en 100 pixels de large
 - afficher l'image à droite
 - insérer une bordure de 2 pixels
 - Si il n'y a pas de logo, mettre le logo du site en petit

Les balises communes à tout le site

- #URL_SITE_SPIP
 - ne contient pas le /
- #NOM_SITE_SPIP
- #EMAIL_WEBMASTER
- #LOGO_SITE_SPIP
- #DOSSIER_SQUELETTE
ou #CHEMIN{fichier}

A l'intérieur ou hors d'un
contexte de boucle :

```
[ (#BALISE) ]
```

```
src=' #CHEMIN{image.jpg} '
```

```
src=' #DOSSIER_SQUELETTE/image.jpg '
```

- #CHARSET
- #SELF
- [(#ENV{paramètre_url})]

```
charset=#CHARSET
```

```
<form action=' #SELF' >
```

```
[(#ENV{affiche_detail})]
```

Paramètres et URLs

- **Récupération de paramètre**

- **Dans le contexte du squelette**
#ENV{parametre,defaut}
- **Hors du contexte du squelette**
#EVAL{\$_request('parametre')}

```
#SET{couleur,#ENV{couleur,noir}}
```

- **Construction d'URLs**

- **Récupération de tous les éléments de l'url courante**
#SELF
- **Construction de l'URL pour un squelette**
#URL_PAGE{squelette}

```
[ (#URL_PAGE{vue1} | parametre_url{param,valeur}... ) ]
```

Les formulaires

- [(#FORMULAIRE_RECHERCHE)]
 - accès au moteur de recherche
- [(#FORMULAIRE_INSCRIPTION)]
 - inscription rédacteur ou {forum} pour le forum
- [(#FORMULAIRE_FORUM)]
 - dans un contexte ARTICLES ou FORUM
- [(#FORMULAIRE_ECRIRE_AUTEUR)]
 - dans un contexte AUTEURS, ARTICLES, FORUM
- [(#FORMULAIRE_SITE)]
 - permet au visiteur de proposer des sites

A l'intérieur ou hors d'un
contexte de boucle :

[(#FORMULAIRE)]

- A l'intérieur de votre squelette apporter les modifications suivantes :
 - Afficher la bannière du site avec un lien vers le squelette *sommaire* et le parametre *revisite=oui*
 - Ajouter un lien vers le mail du Webmaster
 - Ajouter le formulaire d'inscription au site
 - Ajouter le formulaire de recherche
 - Ajouter un formulaire de proposition de site
 - Ajouter un formulaire pour qu'on puisse vous écrire (repérer votre `id_auteur`)
 - Modifier dynamiquement le jeu de caractères utilisé par le site dans le `<head>`

Boucles avancées



Syntaxe complète

<B_nom**>**

Code HTML exécuté une fois avant la boucle SI la boucle a au moins une occurrence.

<BOUCLE_nom**(OBJET) {critères} {tri}>**

Code HTML exécuté à chaque occurrence

[texte optionnel avant (#BALISE|filtre) texte optionnel après]

</BOUCLE_nom**>**

Code HTML exécuté une fois après la boucle SI la boucle renvoie des réponses.

</B_nom**>**

Code HTML exécuté une fois si la boucle ne renvoie rien.

</ /B_nom**>**

Syntaxe complète

`<B_nom>`

Code HTML exécuté une fois avant la boucle SI la boucle renvoie au moins une occurrence.

`<BOUCLE_nom(OBJET){critères} {tri}>`

Code HTML exécuté à chaque occurrence avec [(#BALISE|filtre)]

`</BOUCLE_nom>`

Code HTML exécuté une fois après la boucle SI la boucle renvoie des réponses.

`</B_nom>`

Code HTML exécuté une fois si la boucle ne renvoie rien.

`</B_nom>`

`<B_nom>`

Code HTML optionnel avant

`<BOUCLE_nom(TYPE){critère1}{critère2}...{critèrex}>`

Code HTML avec

`[texte optionnel avant (#BALISE|filtre|filtre...)]`

`texte optionnel après]`

`<BOUCLE_recursive(boucle_nom)>`

`</BOUCLE_recursive>`

`</BOUCLE_nom>`

Code HTML optionnel après

`</B_nom>`

Code HTML alternatif

`</B_nom>`

Exemple de boucle conditionnelle

- Si la boucle fournit des résultats ou non :

```
<body>
  <B_une>
    <p>Voici la liste des articles :
  <BOUCLE_une(ARTICLES) {id_rubrique=2} {id_auteur!=1}>
    <h4>
    <a href="#URL_ARTICLE">#TITRE</a>
    </h4>
  </BOUCLE_une>
  </p>
  </B_une>
  <p>Il n'y a pas d'articles correspondant à ces
critères.
  </p>
  </B_une>
</body>
```

Des boucles successives

- On peut développer des boucles multiples dans un même squelette
- Le nom des boucles doit être différent

```
<body>
  <BOUCLE_une(ARTICLES) {id_rubrique=2}
  {id_auteur!=1}>
    <h4>
    <a href="#URL_ARTICLE">#TITRE</a>
    </h4>
  </BOUCLE_une>

  <BOUCLE_deux(ARTICLES) {par hasard} {0,10}>
    <h4>
    <a href="#URL_ARTICLE">#TITRE</a>
    </h4>
  </BOUCLE_deux>
</body>
```

Boucles exclusives

- Pour éviter l'affichage en double du même article dans des boucles différentes

```
<body>
    <BOUCLE_une(ARTICLES) {id_rubrique=2} {id_auteur!=1}
    {doublons x}>
        <h4>
        <a href="#URL_ARTICLE">#TITRE</a>
        </h4>
    </BOUCLE_une>

    <BOUCLE_deux(ARTICLES) {par hasard} {0,10} {doublons x}>
        <h4>
        <a href="#URL_ARTICLE">#TITRE</a>
        </h4>
    </BOUCLE_deux>
</body>
```


Pour exclure un article avec un mot-clef

- Les mots clefs n'autorisent pas de critère négatif : il faut avoir recours à l'exclusion.

```
<BOUCLE_ex(ARTICLES) {titre_mot=pas_une} {doublons x}>  
</BOUCLE_ex>
```

```
<BOUCLE_une(ARTICLES) {id_rubrique=2} {id_auteur!=1} {doublons x}>  
  <h4>  
  <a href="#URL_ARTICLE">#TITRE</a>  
  </h4>  
</BOUCLE_une>
```

```
<BOUCLE_deux(ARTICLES) {par hasard} {0,10} {doublons x}>  
  <h4>  
  <a href="#URL_ARTICLE">#TITRE</a>  
  </h4>  
</BOUCLE_deux>
```

- Reprendre la version projet.html finale. La renommer *projet_nom.html*
- Intégrer les données dynamiques suivantes dans le squelette *projet_nom.html* :
- Bloc Edito
 - Boucle affichant les articles avec le mot-clef *sommaire_edito*
 - #TITRE, #TEXTE, #LOGO_ARTICLE
- Bloc A la une
 - Boucle affichant les articles avec le mot-clef *sommaire_centre*
 - #TITRE, #TEXTE, #LOGO_ARTICLE_RUBRIQUE

TP

- Bloc Derniers Articles
 - Boucle affichant les 10 derniers articles
 - #LOGO_RUBRIQUE_ARTICLE, #TITRE, #DATE
- Bloc Menu
 - Boucle affichant les rubriques de niveau 1 ne comportant par le mot-clef : pas_menu
 - #TITRE, #URL_RUBRIQUE
- Bloc Entete
 - #URL_SITE_SPIP, #NOM_SITE_SPIP, #EMAIL_WEBMASTER, #LOGO_SITE_SPIP

Vers les boucles imbriquées...

- Les boucles transportent le contexte de l'objet en cours sous la forme de **variables d'environnement** (id_rubrique, id_article...)
- On peut donc appeler une boucle à l'intérieur d'une autre à partir du moment où les variables d'environnement de la boucle mère sont compatibles avec les critères de la boucle fille.
- La seconde boucle, dépendra du contexte de la première. Elle sera répétée à chaque appel de la première boucle.

Des boucles imbriquées

- Afficher les articles de chaque rubrique

```
<BOUCLE_principale(RUBRIQUES)>

  <B_une>
    Voici la liste des articles dans la rubrique #TITRE :

    <BOUCLE_une(ARTICLES) {id_rubrique} {par titre}>
      <h4><a href="#URL_ARTICLE">#TITRE</a></h4>
    </BOUCLE_une>

  </B_une>
    Désolé, il n'y a pas d'articles dans la rubrique #TITRE
  </B_une>

</BOUCLE_principale>
```

... avec balises non ambiguës

- pour lever le doute sur les balises homonymes

```
<BOUCLE_principale(RUBRIQUES)>
```

```
<B_une>
```

```
Voici la liste des articles dans la rubrique #TITRE :
```

```
<BOUCLE_une(ARTICLES) {id_rubrique} {par titre}>
```

```
<h4>
```

```
Rubrique : #_principale:TITRE
```

```
Article : <a href="#URL_ARTICLE">#TITRE</a>
```

```
</h4>
```

```
</BOUCLE_une>
```

```
</B_une>
```

```
Désolé, il n'y a pas d'articles dans la rubrique  
#TITRE
```

```
</B_une>
```

```
</BOUCLE_principale>
```

... des critères contextuels

- seulement les articles dont le titre le même que celui d'une rubrique de niveau 1

```
<BOUCLE_principale(RUBRIQUE)>  
    <BOUCLE_une(ARTICLES) {titre=#TITRE}  
        <h4>  
            Article : <a href="#URL_ARTICLE">#TITRE</a>  
        </h4>  
    </BOUCLE_une>  
</BOUCLE_principale>
```

- Ajouter les contenus suivants :
 - Bloc Derniers Articles
 - Boucle imbriquée AUTEURS affichant le nom des auteurs (balise *#NOM*)
 - Bloc Derniers Articles
 - Indiquer le nom de la rubrique dans laquelle ils sont publiés à côté du nom l'article (balise non ambiguë)

Un critère de boucle peut donc avoir une valeur

- contextuelle
 - dans le contexte d'une boucle

```
<BOUCLE_une(RUBRIQUES)>  
  <BOUCLE_deux(ARTICLES){id_rubrique}>  
  </BOUCLE_deux>  
</BOUCLE_une>
```

- dans le contexte d'une page avec un paramètre passé dans l'url

```
url : spip.php?page=rubrique&id_rubrique=xxx
```

```
<BOUCLE_deux(ARTICLES){id_rubrique}>  
</BOUCLE_deux>
```

Un critère de boucle peut avoir une valeur :

- fixé dans le critère de la boucle

```
<BOUCLE_deux(ARTICLES) {id_rubrique=1} >  
</BOUCLE_deux>
```

- conditionnel

-> ne tient compte du critère que si il est présent

```
<BOUCLE_deux(ARTICLES) {id_rubrique?} >  
</BOUCLE_deux>
```

- multiple

```
url : spip.php?page=rubrique&id_rubrique=xxx
```

```
<BOUCLE_deux(ARTICLES) {id_rubrique}{id_rubrique=1}>  
</BOUCLE_deux>
```

Lever l'ambiguïté d'un critère

- Par défaut, le contexte de boucle est prioritaire

```
url : spip.php?page=rubrique&id_rubrique=xxx
```

```
<BOUCLE_une(RUBRIQUES)>  
  <BOUCLE_deux(ARTICLES) {id_rubrique} >  
  </BOUCLE_deux>  
</BOUCLE_une>
```

- Pour forcer l'utilisation du paramètre d'url : #ENV{ }

```
url : spip.php?page=rubrique&id_rubrique=xxx
```

```
<BOUCLE_une(RUBRIQUES)>  
  <BOUCLE_deux(ARTICLES) {id_rubrique=#ENV{id_rubrique}} >  
  </BOUCLE_deux>  
</BOUCLE_une>
```

- Créer un nouveau squelette *article_nom.html*
 - Ce squelette devra récupérer le critère de sélection passé dans l'url : *id_article* (boucle ARTICLES)
 - Il affichera le titre de l'article, l'auteur, la date de publication ainsi que le texte de l'article.
 - Il redimensionnera les images du texte de l'article à 500 pixels de large maximum.
 - Il affichera les documents joints à l'article (boucle DOCUMENTS).

Les boucles récursives

- Les Boucles récursives s'appellent elles mêmes jusqu'à ne plus rencontrer d'occurrence
- Elles sont souvent utilisées pour dérouler des structures arborescentes

```
<BOUCLE_rubrique(RUBRIQUES){id_parent}{par titre}>  
  
  #TITRE  
  
  <BOUCLE_ssRubriques(BOUCLE_rubrique)>  
  </BOUCLE_ssRubriques>  
  
</BOUCLE_Rubriques>
```

- Afficher l'arborescence du site en utilisant une boucle de rubriques et de sous-rubriques récursives :

```
<ul>
  <li>
    Rubrique 1
    <ul>
      <li>Sous-rubrique 1</li>
      <li>Sous-rubrique 2</li>
    </ul>
  </li>
  <li>
    Rubrique 2
  </li>
</ul>
```

La boucle hierarchie

- La boucle HIERARCHIE retourne la liste des RUBRIQUES qui mènent de la racine du site à la rubrique ou à l'article en cours.

```
<B_hie>
  ACCUEIL
  <BOUCLE_hie(HIERARCHIE){id_rubrique?}{id_article?}>
    &nbsp;>&nbsp;>
    <a href="#URL_RUBRIQUE">
      [ (#TITRE|supprimer_numero) ]
    </a>
  </BOUCLE_hie>
</B_hie>
```

TP

- Proposer plusieurs feuilles de style pour le site :
style1.css, style2.css, style3.css...
- Choisir automatiquement la feuille de style incluse au squelette *article.html*
 - en fonction du secteur du site dans lequel on se trouve
 - en fonction du paramètre style passé dans l'url
 - (en fonction du nom du site)
- Vérifier que la feuille de style existe autrement utiliser la feuille de style par défaut

Manipulation de variables



Les compteurs

- affiche le nombre d'occurrences et l'index de l'occurrence en cours dans un contexte de boucle :

```
<B_une>
  <p>Voici la liste des articles :
<BOUCLE_une(ARTICLES) {id_rubrique=2} {id_auteur!=1}>
  Article n° : [ (#COMPTEUR_BOUCLE) ]
  <h4>
  <a href="#URL_ARTICLE">#TITRE</a>
  </h4>
</BOUCLE_une>
  <p>Nombre total d'articles : [ (#TOTAL_BOUCLE) ] .
  </p>
</B_une>
  <p>Désolé, il n'y a pas d'articles.</p>
</B_une>
```

VARIABLES D'ENVIRONNEMENT

■ #ENV{}

- permet de voir le tableau associatif des variables disponibles dans le contexte de la page (url ou passage de paramètres)

```
a:4{s:4:"page";s:7:"article";s:10:"id_article";s:1:"3";s:4:"truc";s:1:"oui";s:4:"date";s:19:"2007-11-09 21:55:58";}
```

[url : spip.php?page=article&id_article=3&truc=oui](http://spip.php?page=article&id_article=3&truc=oui)

■ #ENV{'variable'}

- permet de récupérer la variable *variable* manipulée par SPIP
- ex : #ENV{'truc'} -> 'oui'

■ #VARIABLE

- est aussi disponible dans le contexte de la page
- ex : #TRUC -> 'oui'

Manipulation de variables

- #SET{var,val}
 - stocke une valeur dans une variable (portée : squelette)
- #GET{var}
 - restitue la valeur
- #ARRAY{index0,val0,index1,val1...}
 - tableau associatif
- #EVAL{exp}
 - évalue une expression php

```
<BOUCLE_aut(AUTEURS){0,1}>#SET{aut,#NOM}</BOUCLE_aut>  
[ (#LOGO_ARTICLE || sinon{#GET{aut}}) ]
```

quel est le rôle de cette utilisation de variables ?

Les variables globales (php)

- Tableau associatif php des variables disponibles dans la page : **\$GLOBALS**

```
foreach ($GLOBALS as $index=>$val)
    echo $index." : ".$val."<br />";
```

- Ex : **\$GLOBALS['auteur_session']** (tableau associatif)
 - id_auteur : 1
 - nom : pvincent
 - login : pvincent
 - email : pvincent@erasme.org
 - statut : 0minirezo
 - lang

```
<BOUCLE_auteur(AUTEURS)
{id_auteur=#EVAL{$GLOBALS['auteur_session']['id_auteur']}}>
    #NOM, vous êtes connecté.
</BOUCLE_auteur>
```

- Afficher le nombre d'articles du site publiés en ligne
- Déterminer la date de la dernière mise à jour sur le site
- Afficher l'auteur connecté
- Un exemple peu élégant de #GET / #SET permettant de résoudre les requêtes non natives à SPIP sans passer par php :

```
<BOUCLE_articlesT(ARTICLES){branche}{0,6}>
    #SET{#COMPTEUR_BOUCLE,#ID_ARTICLE}
</BOUCLE_articlesT>

#SET{critere_in,
#ARRAY{0,#GET{1},1,#GET{2},2,#GET{3},3,#GET{4},4,#GET{5},5,#GET{6}}

<BOUCLE_forums_liens(FORUMS){id_article IN #GET{critere_in}} {tout} {par
date}{inverse}>
    [(#DATE|affdate_court) : ]<a href="#URL_FORUM">#TITRE</a><br />
</BOUCLE_forums_liens>
```

récupère les forum des 6 articles de la branche

Inclusion de squelettes



Inclure des squelettes

- Des sous-squelettes peuvent être inclus dans les squelettes principaux.
- Deux syntaxes
 - le sous-squelette est réellement inclus à la page appelante et leur calcul est lié

```
[ (#INCLURE {fond=entete}) ]  
[ (#INCLURE {fond=agenda} {id_rubrique}) ]
```

- le sous-squelette est géré de manière distincte et recalculé indépendamment de la page appelante (plus de fichiers de cache autonomes)

```
<INCLURE {fond=bloc-connexion}>  
<INCLURE {fond=agenda} {id_rubrique}>
```


Le passage de paramètres

- Les sous-squelettes n'héritent pas des variables d'environnement ou de contexte du squelette
 - Ces paramètres doivent être indiqués lors de l'appel
 - Ils sont équivalents à un passage de paramètre dans l'url : #ENV{variable}

```
[ (#INCLUDE {fond=entete}) ]  
[ (#INCLUDE {fond=agenda} {id_rubrique} {truc=oui}) ]
```

- dans le sous-squelette agenda.html :

```
<BOUCLE_evenements2 (BREVES) {id_rubrique?}>  
  #TITRE  
  [ (#TRUC |= {oui} | ? {#DATE}) ]  
</BOUCLE_evenements2>
```

- Depuis votre squelette *sommaire.html* extraire les sous-squelettes qui peuvent être mutualisés :
 - entete
 - menu
 - footer
- Créer et compléter les squelettes
 - article
 - rubrique
 - plan
 - forum
- Discussion autour des solutions

Requête générales sur les tables

- Depuis la version 1.9, on peut faire des requêtes de boucle sur **n'importe quelle table de la BDD**

```
<BOUCLE_nom(table) {champ1=xxx}>  
  #CHAMP2  
  #CHAMP3  
</BOUCLE_nom>
```

- Exemple : récupérer la liste des rubriques restreintes d'un auteur

```
<BOUCLE_auteur(spip_auteurs_rubriques) {id_auteur=1}>  
  #ID_RUBRIQUE  
</BOUCLE_auteur>
```

SPIP 2



Nouvelles balises

- #SESSION{index}
- #AUTORISER{action, objet,id}
- #URL_ACTION_AUTEUR{action,param,url}
- #FORMULAIRE_action_objet{id_objet,param,url}

```
[ (#AUTORISER{modifier, rubrique, #ID_RUBRIQUE})  
  <a  
href="#URL_ACTION_AUTEUR{editer_#OBJET,nan,#SELF}&id_parent  
=#  
ID_RUBRIQUE&statut=publie&titre=nouveau+contenu">  
  lien</a>  
  #FORMULAIRE_EDITER_SITE{#ID_SYNDIC, #ID_RUBRIQUE,  
#SELF}  
]
```

Manipulation de tableaux

- #SET{tab,#ARRAY{cle1,item1,..}}
- [(#GET{tab}|table_valeur{1})]
- [(#GET{tab}|table_valeur{cle})]
- #SET{tab, #GET{tab}|array_merge{#ARRAY{clef, val}}}

ou

- Utiliser le plugin SPIP BONUX avec la boucle POUR

```
<BOUCLE_test(POUR){tableau #GET{tab}}>  
#CLE=>#VALEUR<br />  
</BOUCLE_test>
```

Références

- <http://programmer.spip.org>
- <http://www.plugandspip.com/>
- <http://marcimat.magraine.net/Ajouter-des-champs-dans-une-table>

- <http://www.spip-contrib.net/SPIP-Bonux>

```
<BOUCLE_test(CONDITION){si #ENV{test}|>{5}}>  
vrai  
</BOUCLE_test>  
faux  
<>//B_test>
```

Partie 4

Méthodologie



Mentions légales

- Créer un article indiquant les mentions légales du site
 - Conditions d'utilisation et responsable de publication
 - Propriété intellectuelle sur les contenus du site
 - indiquer le type de licence s'appliquant aux contenus du site :
copyright, créative commons, GPL
 - Protection des données personnelles et droit de rectification de l'internaute
 - indiquer le n° d'enregistrement CNIL du site
 - citer l'article 34 de la loi "Informatique et Libertés" du 6 janvier 1978 pour droits de rectification
 - Hébergeur
 - Liens vers des sites extérieurs
- http://www.villes-internet.net/UPLoad/rubrique/pages/33/33_rubrique.php

Chaîne de développement du site



Les différentes étapes

- Analyse et schémas fonctionnels
 - Conception graphique
 - Développement des pages dynamiques / BDD
 - Installation et aspects réseau
 - Administration et ligne éditoriale
- > Idéalement les rôles doivent être séparés. Dans la pratique, ils sont souvent cumulés au niveau d'un Webmaster et d'un comité rédactionnel.

Vie du site

Étape	Rôle	Déliverable
Analyse fonctionnelle	Chef de projet / Maître d'ouvrage / Comité de pilotage	Cahier des charges, spécifications et schéma fonctionnel = <u>dessin fonctionnel du squelette</u>
Conception des pages Web / design	Webmaster (Graphiste)	Templates = <u>Pages statiques HTML</u> et <u>Feuilles de style CSS</u>
Développement des pages dynamiques / BDD	Webmaster (Développeur)	Intégration du langage dynamique dans les templates : <u>Langage SPIP</u> , PHP, JAVA, ASP...
Installation réseau et sécurité	Administrateur système et réseaux	Mise en ligne du système de publication et des templates
Administration et ligne éditoriale	Chef de projet	Ligne éditoriale / Options / Animation et modération / Tests et retours fonctionnels
Publication	Comité de rédaction	Contenu et mise en forme

Conception

Exploitation

Création d'un site SPIP

- 1. Liste des vues utiles
 - déterminer la liste des pages de squelettes du site
- 2. Maquette fonctionnelle
 - faire un schéma sur table de chaque squelette en charge : blocs, objets de la base et règles d'affichage
- 3. Développement statique : templates
 - pour chaque squelette, réaliser une page XHTML, composée de la structure et du contenu du site statique en charge
 - externaliser les styles d'affichage dans une feuille CSS unique
- 4. Développement dynamique : boucles SPIP
 - remplacer le contenu statique de la page HTML par des boucles SPIP
 - gérer les conditions d'affichage optionnelles des blocs

Ne pas partir from scratch

- S'inspirer des squelettes existants et des boucles disponibles dans le répertoire ***dist*** de la distribution :
 - boucles forum
 - boucles documents joints
 - boucles menu
 - boucles sites syndiqués...
 - -> éventuellement développer une **feuille de style pour dist**
- Utiliser des jeux de squelettes existants (et bien écrits)
 - alternatives, sarka, erasme, etc
- Utiliser des templates existants (et bien écrits)
 - freetemplates, csszengarden, etc

Migrer un site SPIP ***<=1.9.1 vers 1.9.2***



Procédure de migration

- Faire une sauvegarde de la base
- Déplacez tous les fichiers et dossiers de l'ancienne installation dans un sous-répertoire */old*
- Installez le nouveau noyau de SPIP 1.9 à la racine
 - droits d'écriture sur *IMG*, *tmp*, *local*, *config*
 - créer */squelettes* et */plugins*
- Depuis votre ancienne version */old*, copiez :
 - votre dossier de squelettes dans */squelettes*
 - votre dossier */IMG* à la racine
 - *mes_fonctions.php3* -> */squelettes/sq/mes_fonctions.php*
 - */ecrire/mes_options.php3* -> */config/mes_options.php*

Procédure de migration (2)

- Spécifier dans *mes_options.php* le chemin vers votre jeu de squelettes :
 - `$GLOBALS['dossier_squelettes'] = 'squelettes/repertoire';`
- Se rendre dans l'espace privé
 - réinstallation des connexions mysql
 - mise à jour de la base de données + authentification ftp
- Notes
 - on peut supprimer */IMG/icone_barre* et */IMG/icones*
 - ne pas restaurer la BDD
 - si les squelettes sont <1.9 il faut les adapter
 - ne pas faire de downgrade de site

Adapter les squelettes

Remplacement des urls statiques	objet.php3?id_objet=#ID_OBJET	#URL_ARTICLE, #URL_RUBRIQUE, #URL_BREVE, #URL_AUTEUR, #URL_MOT
Remplacement des autres urls	objet.php3	spip.php?page=objet (plan, forum, contact, recherche, backend...)
Remplacement des chemins statiques	"chemin_vers_squelette/..." ou pas de chemin	#CHEMIN}, #DOSSIER_SQUELETTE/
Amélioration des balises statiques	utf-8, nom_site, url_site ...	#CHARSET, #NOM_SITE_SPIP, #URL_SITE_SPIP ...
Amélioration des formulaires statiques	recherche, contact, forum, ect	#FORMULAIRE_RECHERCHE, #FORMULAIRE_ECRIRE_AUTEUR, #FORMULAIRE_FORUM, ect
Mises en conformité W3C	 , <hr>, <input>, , <meta> ...	 , <hr />, <input />, , <meta /> ...

Voir script reseau

Reprise d'un site existant
-> SPIP



Reprise des templates

- Page de l'ancien site -> Enregistrer sous
- Nettoyage XHTML
- Extraction des styles CSS + organisation des images intégrées
- Mappage fonctionnel de la structure
 - quel objet fonctionnel (ou champ) de l'ancien site deviendra quel objet de SPIP : article, breve, rubrique, auteur, site, ect
- Intégration des boucles dans le canevas et développement des squelettes

Reprise de contenu

- Si le contenu est statique (peu important)
 - publication manuelle des objets dans le back-office
 - copier/coller de chaque contenu -> bloc-note pour nettoyage de mise en forme -> raccourcis typo
 - mappage avec les champs (titre, texte, emplacement)
- Si le contenu est statique (important)
 - création d'un tableau .csv de contenu après nettoyage
 - intégration massive avec le plugin *import_csv*
- Si le contenu est dynamique
 - développement d'une moulinette en php

import_csv

- Type de fichier à créer

titre;descriptif;texte;statut;id_rubrique

"01-titre1";"descriptif

descriptif";"texte texte texte texte texte texte texte texte texte texte texte texte texte
texte

texte texte texte texte texte texte texte texte texte texte texte texte texte texte texte
texte texte ";publie;9

"02-titre2";"descriptif descriptif";"texte texte texte texte texte texte texte texte texte
texte texte texte texte textetexte texte texte texte texte texte texte texte texte
texte texte texte texte texte texte texte ";publie;9

- Interface

Prévisualisation

Caractère de séparation

1^{ère} ligne d'en-tête

Champ CSV Champ Table

titre	<input type="text" value="titre"/>
descriptif	<input type="text" value="descriptif"/>
texte	<input type="text" value="texte"/>
statut	<input type="text" value="statut"/>
id_rubrique	<input type="text" value="id_rubrique"/>

Intégrer GoogleMaps à SPIP



Généralités

- Google Maps fonctionne sous forme d'API (bibliothèque de fonctions) Javascript intégrable à une page HTML
- Le site doit accéder au Web pour télécharger les données de la carte
- Il faut commencer par obtenir une clef d'utilisation pour l'API Google :
<http://www.google.fr/apis/maps/signup.html>



La page GoogleMaps par défaut

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
    <title>Google Maps JavaScript API Example</title>
    <script src="http://maps.google.com/maps?file=api&v=2&key=clef" />

    <script type="text/javascript">

      function load()
      {
        if (GBrowserIsCompatible())
        {
          var map = new GMap2(document.getElementById("map"));
          map.setCenter(new GLatLng(45.7384,4.4643), 12);
        }
      }

    </script>

  </head>
  <body onload="load()" onunload="GUnload()">
    <div id="map" style="width: 500px; height: 300px"></div>
  </body>
</html>
```

Ajouter des marqueurs

```
/*Ajoute des marqueurs avec texte s'ouvrant lors des
clics. La fonction createMarker est à déclarer une fois*/

function createMarker(point, texte)
{
  var marker = new GMarker(point);
  map.addOverlay(marker);
  GEvent.addListener(marker, "click", function() {
    marker.openInfoWindowHtml(texte);
  });
}

/*Déclaration d'un marqueur*/

var point = new GLatLng(45.76201437142477,
4.4968414306640625);
createMarker(point, "Bruliolles et ses beaux paturages");
```

Liens utiles

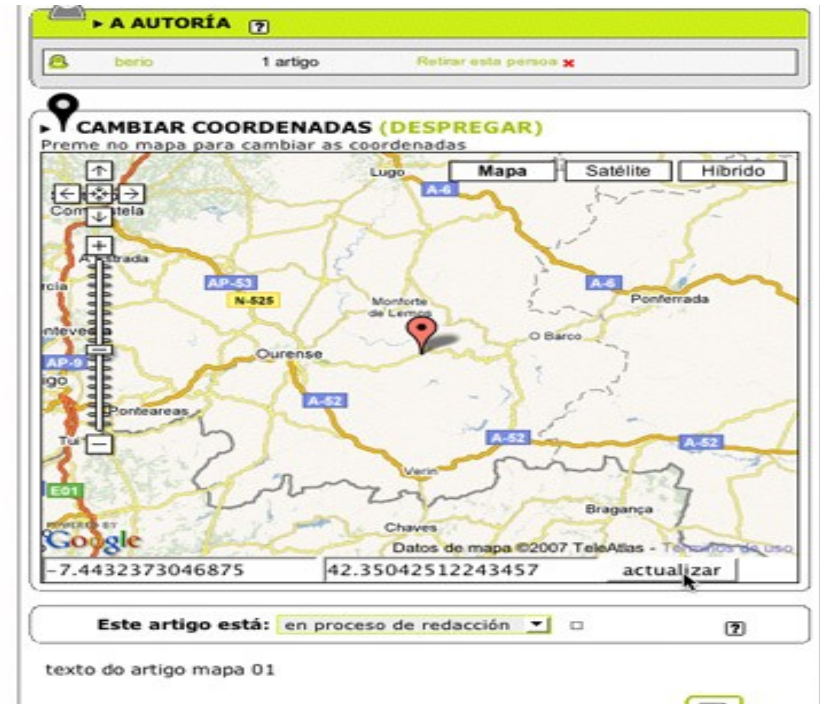
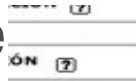
- Manuel de Référence de l'API Gmap2
 - <http://www.google.fr/apis/maps/documentation/reference.html>
- Concepts et Exemples
 - <http://www.google.com/apis/maps/documentation/index.html>
- Tutoriaux
 - <http://www.econym.demon.co.uk/googlemaps/>
- Le Wiki Mapki
 - http://www.mapki.com/index.php?title=Main_Page
- Trouver les coordonnées d'une ville
 - <http://www.multimap.com/>

Interfacer Googlemaps avec SPIP

- Créer une bibliothèque de mots-clefs « localisation » avec en descriptif les coordonnées : "Lat,Long"
- Créer un sous-squelette *bloc-localisation.html*
 - prenant en paramètre *nom* et *coordonnees*
 - qui affiche le bloc carte avec un marqueur reprenant les paramètres
- Dans le squelette article, inclure *bloc-localisation* si un mot-clef *localisation* est joint

Plugins Google-Maps

- <http://www.spip-contrib.net/Plugin-Google-Maps-API>
 - permet de intégrer la clé et les entêtes Gmaps à vos pages avec #CONFIG
- <http://www.spip-contrib.net/Plugin-Gis-escoitar>
 - en complément du premier, permet de gérer des cartes liées à chaque article
 - mais pas intuitif, préfère avec du temps le dvpt donné en TP précédent



Références

- Documentation officielle de SPIP
 - <http://www.spip.net/fr>
- Supports de formation sur SPIP-Contrib
 - <http://www.spip-contrib.net/Supports-de-formations-SF>
- Documentation des plugins
 - <http://www.spip-contrib.net/Plugin>
- Merci aux auteurs de ces contributions

Crédits

- Patrick VINCENT
 - pvincent@erasme.org
 - pvincent4@gmail.com
- Contenu
 - non garanti exempt d'erreurs
 - sous licence **FDL**
- Merci !

