

Direction régionale de l'Environnement,
de l'Aménagement du Logement
Nord-Pas de Calais
Évolution d'une application de
cartographie dynamique CARMEN 1.6

Spécifications techniques

Auteurs: ALKANTE- Version V1.2- Mars 2009

SOMMAIRE

1. Objectif de ce document.....	3
2. Fonctionnalités de consultation de CARMEN 1.6.....	4
2.1. Initialisation du module de consultation cartographique.....	4
2.2. Initialisation de la page html.....	5
2.3. Structuration de la vue cartographique.....	5
2.4. Gestion de couches / Légende.....	6
2.5. Localisation grâce aux requêteurs	7
2.6. Changement du système de coordonnées de la barre d'information.....	7
2.7. Fonctionnalités de la barre d'outils.....	7
1.1.1. Outils cartographiques issus de OpenLayers.....	8
1.1.2. Outils issus de CARMEN 1.5.....	8
1.1.3. Autres outils spécifiques.....	8
2.8. Répartition des différents services sur le poste de navigation, le serveur applicatif et le serveur de téléchargement.....	11
3. Compléments d'information.....	12
3.1. Gestion des services WMS/WFS.....	12
3.2. Gestion des formats d'image de sortie.....	12

1. Objectif de ce document

Ce document précise les aspects techniques des principales fonctionnalités de l'application CARMEN 1.6 pour sa partie consultation. Il décrit pour chaque fonctionnalité quelles sont les technologies employées, les services web associés et les composants de l'application impliqués (ex. : poste de navigation, serveur applicatif, serveur de données ou encore serveur de téléchargement). Ce document vient compléter les documents de spécifications fonctionnelles et d'architecture générale de l'application CARMEN 1.6.

Il est accompagné de documents annexes :

- CARMEN_METADATA.ods: tableau décrivant les métadonnées des mapfiles de CARMEN,
- EspacesProtegesGUY.xml : exemple de contexte ows généré à partir d'un mapfile de CARMEN
- tagmapping_carmen.csv : tableau de correspondance entre les tags du mapfile et les balises du fichier ows.

2. Fonctionnalités de consultation de CARMEN 1.6

2.1. Initialisation du module de consultation cartographique

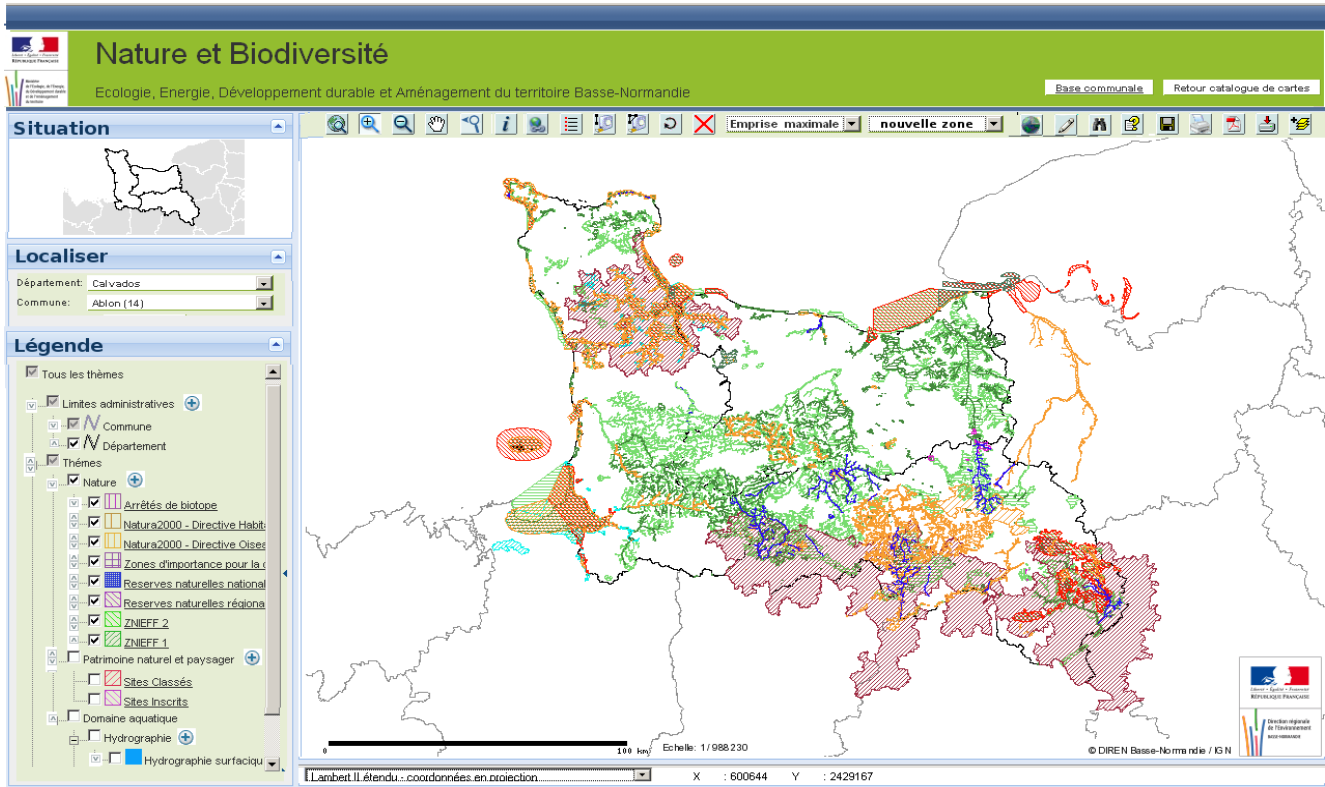


Figure 1: Interface de consultation

L'appel au module de consultation se fait par une URL ou sont passés en paramètre un numéro de service et un fichier mapfile décrivant la carte et l'interface de consultation associée (outils de l'interface disponibles pour la carte considérée).

http://carto.ecologie.gouv.fr/10W/Nature_Paysage.map

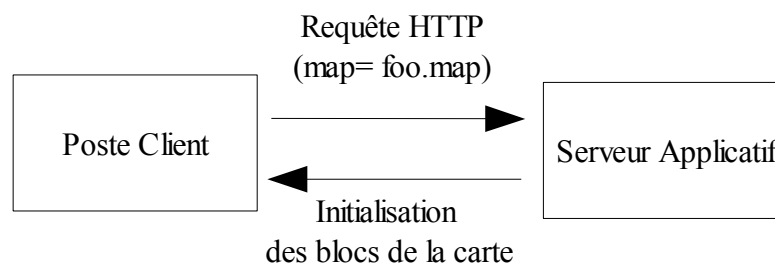


Figure 2: Service de Lecture de configuration

Lors du chargement de la page de consultation cartographique sur le poste client, le service web de *Lecture de configuration* est appelé. Ce service prend en paramètre le nom du mapfile

de la carte et renvoie au poste client les différentes informations sous la forme d'un flux XML. Ces informations comprennent :

- les informations générales sur la carte (ex. : nom, projection)
- la description des différentes couches du mapfile (ex. : nom, adresse de la couche, projection, metadata du layer, urls des images de la légende),
- les informations de structuration des couches en groupes techniques (information stockée dans les métadonnées du mapfile),
- un lien vers l'image de la vue générale,
- la liste des fonctionnalités actives (paramétrage de l'interface),
- les paramètres d'initialisation du requêteur de localisation (ex. : nombre de niveaux du requêteur, couche et champ associés à chaque niveau).

Le service de Lecture de configuration s'appuie sur les fonctions de la librairie PHPMapscript au niveau du serveur applicatif pour générer le fichier XML de configuration à partir du mapfile. Ce fichier de configuration s'inspirera de la proposition OWS Context¹ de l'OGC pour la description d'un contexte de carte. Le fichier de configuration partagera la même structure que le fichier de sauvegarde du contexte de la carte (voir Enregistrement/Sauvegarde de contexte). Un exemple de fichier ows est fourni en annexe à ce document (EspacesProtegesGUY.xml).

2.2. Initialisation de la page html

La page html sera générée à partir d'un template en se basant sur la librairie smarty (<http://www.smarty.net/>). Ce template comprendra les différents blocs de la page, à savoir : le bandeau, la carte, la barre d'outils, la vue générale, le requêteur de localisation, la légende/liste des couches, la barre d'information.

Cependant, la mise en page sera également réalisée dans une librairie javascript spécifique pour réaliser la mise en forme « extjs ».

2.3. Structuration de la vue cartographique

La vue cartographique générale s'appuie sur OpenLayers. Dans OpenLayers, chaque couche est associé à un objet <div>. Le rendu cartographique est construit par superposition des différents objets <div>.

Dans CARMEN 1.6, pour minimiser le nombre de couches gérées par OpenLayers, les couches de données de la carte issues d'un même groupe technique sont représentées dans la même couche OpenLayers. Cela signifie qu'au niveau des serveurs de données une seule image est générée et associée ensuite à un unique objet <div> dans OpenLayers.

Les couches OpenLayers sont construites à partir de données retournées par le serveur de diffusion de référentiels : flux WMS et images retournées par Mapserver (mode CGI).

¹ Le format OWS Context est annoncé comme une extension du format WMC. Le format WMC permet d'associer un contexte cartographique aux services WMS. L'OWS Context doit prendre en compte les autres services OGC (ex. : WFS, WCS).

2.4. Gestion de couches / Légende

L'interface graphique de la gestion de légende (voir Figure 3) s'appuie sur une classe Javascript spécifique enrichissant les classes Tree de la librairie ExtJS. La librairie ExtJS dispose de classes Tree permettant le réorganisation des noeuds de l'arbre.

Les actions sur les couches (activation/désactivation, ouverture/fermeture de groupes, transparence des couches ou des groupes techniques) sont donc gérées au niveau du client en utilisant les fonctionnalités d'OpenLayers.

Les images des caissons de légende seront chargées soit en faisant appel à la requête `getlegendgraphic` pour les couches WMS, soit en faisant appel à une URL CGI de mapserver (`mode=legendicon`) pour les couches issues de Mapserver. Les liens seront déterminés dans le fichier `ows`.

Dans l'arbre de couches du cadre légende, chaque noeud, représentant une couche ou un groupe de couches, est lié à la couche OpenLayers dans laquelle il est représenté. Lorsqu'un groupe de couches dans la légende correspond à un groupe technique de couches (issues du même mapfile et représentées dans une même couche OpenLayers), la réorganisation de ces couches dans la légende (changement d'ordre d'affichage, activation/désactivation) entraîne un rechargement de la couche OpenLayers associée.

Aucun traitement serveur n'est nécessaire pour déterminer quels sont les groupes techniques. La composition est déterminée par une metadata présente dans le fichier `ows`.

La réorganisation de groupes ou de couches associées à différents groupes techniques, et donc à différentes couches OpenLayers, est réalisée en local sur le client en utilisant les fonctionnalités d'OpenLayers, sans nécessiter d'appel au serveur.

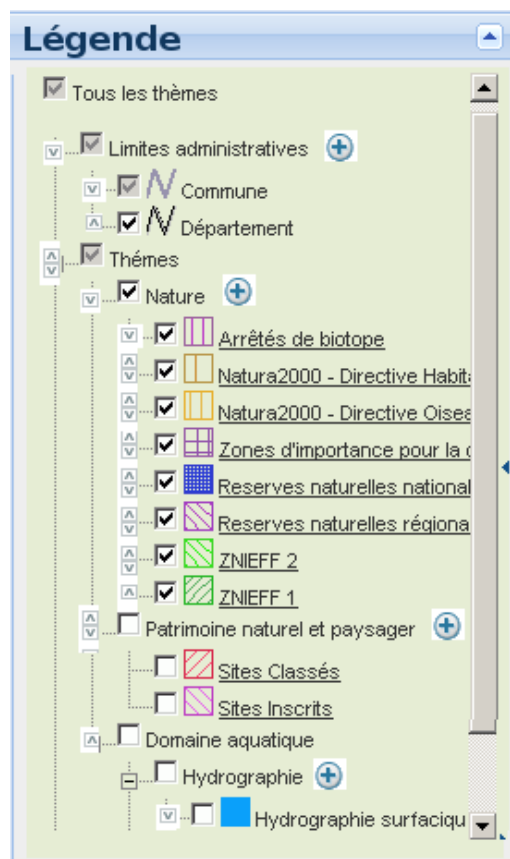


Figure 3: Interface de gestion de couches / Légende

2.5. Localisation grâce aux requêteurs

Le requêteur de localisation fait appel à un service web spécifique *GetAreas* localisé sur le serveur applicatif pour remplir les différentes listes déroulantes (ex. : régions, départements, communes). Ce service prend en paramètre le mapfile de la carte. Il renvoie une liste d'emprises (nom et emprise spatiale) correspondant soit à un ensemble des zones (ex. : départements), soit à un ensemble de sous-zones (communes d'un département) d'une zone passée en paramètre supplémentaire. Ce service s'appuie sur PhpMapScript.

Le zoom sur l'emprise spatiale de la zone sélectionnée par les listes déroulantes fait appel aux fonction de zoom d'OpenLayers.

2.6. Changement du système de coordonnées de la barre d'information

Dans la barre d'information, l'utilisateur a la possibilité de modifier la projection dans laquelle sont affichées les coordonnées courantes du pointeur de la souris. La conversion de la position courante du système de coordonnées de la carte (fournie par OpenLayers) dans le système de coordonnées sélectionné par l'utilisateur est réalisée par une classe Javascript localement sur le poste client.

Les différents noms de projection et paramètres de conversion associés sont chargés lors du chargement du module de consultation cartographique.

Un système de coordonnées pivot (latitude/longitude) sera utilisé pour les conversions. Par exemple, si le système de coordonnées de la carte est Lambert2 étendu et que les coordonnées doivent être affichées en Lambert 93, la conversion se fera en 2 temps :

Lambert2 étendu → pivot (lat/long) → Lambert 93

Ainsi, seuls les paramètres de conversion et de conversion inverse d'un système de coordonnées proposé vers le système pivot ont besoin d'être connus au niveau du poste client. Pour ne pas surcharger celui-ci, cette approche suppose qu'un nombre limité de projections (ne dépassant pas 20) soient prises en charge.

La librairie utilisée pour la gestion des projections est proj4js. Celle-ci sera utilisée avec les fichiers de configuration des projections isolés.

2.7. Fonctionnalités de la barre d'outils

Les outils de la barre d'outils soit s'appuient sur les outils cartographiques disponibles dans OpenLayers, soit enrichissent des outils existants de CARMEN 1.5, soit font appel à des services spécifiques développés pour l'application.



- Outil issu d'OpenLayer
- Outil s'appuyant sur le moteur GIMS
- Outil développé spécifiquement

Figure 4: Barre d'outils

1.1.1. Outils cartographiques issus de OpenLayers

Zoom, Déplacement, Mesure

Les outils de navigation cartographiques de zoom (vue générale, zoom avant, zoom arrière, zoom précédent), de déplacement et de mesure (périmètre et superficie) sont repris sans changement de OpenLayers.

Annotations

L'outil d'annotation de CARMEN 1.6 s'appuie sur les fonctionnalités d'annotation d'OpenLayers. OpenLayers permet la saisie d'annotations géométriques ponctuelles, linéaires et surfaciques sur la carte, ainsi que leur modification et leur suppression. Dans CARMEN 1.6, une classe Javascript spécifique permet en plus de paramétrer la couleur de chaque annotation saisie. Les annotations seront stockées au format KML, le format KML permettant d'associer un style graphique propre pour chaque annotation géométrique. OpenLayers lit nativement le KML en prenant en compte le style associé à chaque objet.

Un enregistrement des annotations au format GML est une alternative possible, mais nécessite d'une part d'enregistrer la définition des styles graphiques dans le fichier GML (en s'appuyant sur un schéma XML spécifique) et d'autre part, d'enrichir OpenLayers avec une classe de lecture de ces styles graphiques.

1.1.2. Outils issus de CARMEN 1.5

Export / Impression de la carte

Les outils d'export (format image) et d'impression de la carte (depuis le navigateur ou au format PDF) s'appuient sur le module GIMS de CARMEN 1.5 pour la génération de la carte centrale. La génération de la légende est assurée par un module spécifique identique à celui utilisé pour l'affichage de la légende sur le poste client (voir Affichage de la légende de la carte).

Ces fonctionnalités sont réalisées sous la forme de services localisés sur le serveur applicatif : *GetIMG*, *GetPDF*, *GetHTML*. Ces services sont paramétrés par le fichier contexte décrivant la carte.

1.1.3. Autres outils spécifiques

Outil d'information sur les couches / accès aux liens hypertextes des couches

L'outil d'accès aux liens hypertexte étant une version spécialisée de l'outil d'information, nous détaillons uniquement le premier outil.

L'outil d'interrogation des couches s'appuie sur mapserver en mode CGI pour l'interrogation des couches vectorielles et sur le service *getFeatureInfo* pour les couches WMS de la carte. Les couches au format *TileCache* ne sont pas interrogées.

La position (interrogation par un point) ou les limites de l'étendue (tracé d'un rectangle sur la carte) de l'interrogation sont récupérées en utilisant les fonctions d'OpenLayers. Ensuite, un service spécifique *GetInformation* sur le serveur applicatif est appelé avec en paramètre un résumé du contexte de la carte (liste des couches actives dans la carte) et la zone interrogée. Ce service lance une requête CGI à mapserver (en mode *NQUERY*) pour récupérer la liste des entités vectorielles sélectionnées par l'outil d'interrogation. Le principe est d'utiliser les possibilités de mise en forme de mapserver (*TEMPLATE*) pour structurer au maximum les informations résultats dans un format proche du JSON. Le service *GetInformation* interroge également les différentes couches WMS actives de la carte avec la directive *getFeatureInfo*. L'ensemble des résultats (issus de mapserver et des couches WMS) sont structurés et transcrits au format JSON avant d'être transmis sous forme d'une requête ajax au poste client.

Sur le poste client, les données sont présentées à l'utilisateur dans une fenêtre construite à partir des composants ExtJS.

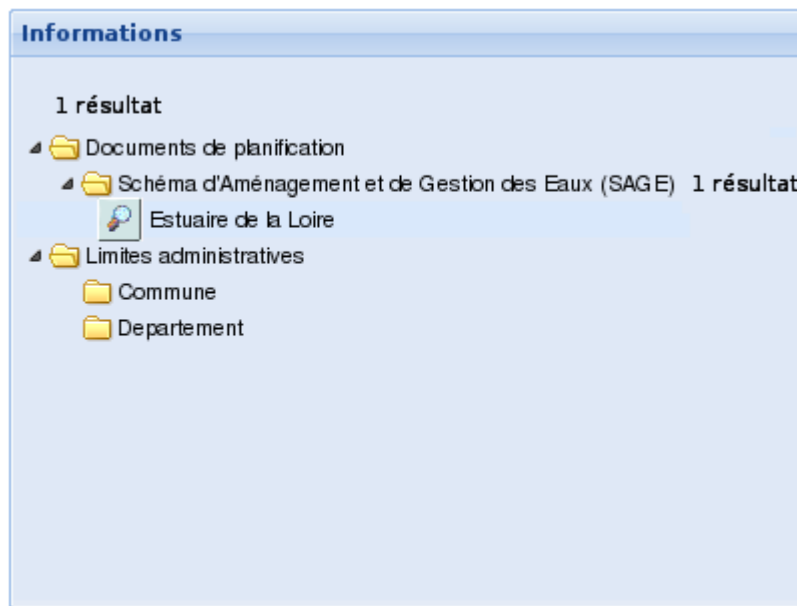


Figure 5: Exemple de fenêtre de l'interface de consultation

Pour chaque résultat fourni, un outil de recentrage permet de repositionner la carte sur une zone pertinente associée. Le recentrage est opéré en utilisant les fonctionnalités de zoom d'OpenLayers.

La symbologie de l'objet géographique associé au résultat est également modifiée : la couche OpenLayers associée au résultat est redessinée en utilisant le mode QUERYMAP de Mapserver-cgi.

Affichage de la légende de la carte

L'affichage de la légende de la carte est réalisée par un module Javascript spécifique chargé sur le poste client. Ce module reproduit au format HTML une version simplifiée de l'arbre de couches du cadre de légende (seules les couches actives sont représentées).

Changement d'échelle

L'outil de changement d'échelle est réalisé par une classe Javascript spécifique coté client. Cette classe interagit avec les fonctionnalités de zoom d'OpenLayers.

Définition/sélection de zones préférentielles

L'outil permettant d'associer un nom à l'emprise courante de la carte et de cadrer sur les zones préférentielles définies est réalisé par une classe Javascript spécifique qui interagit avec les fonctionnalités de zoom d'OpenLayers.

Sauvegarde/Chargement de contexte

L'enregistrement de contexte consiste à sauvegarder au format XML les caractéristiques de la carte (ex. : étendue de la carte, caractéristiques des couches, structure de l'arbre de couches, zones préférentielles, annotations). Le schéma XML adopté s'inspire du modèle OWS Context proposé par l'OGC. Cette action est réalisée côté client par une classe Javascript spécifique.

Le chargement d'un contexte s'appuie sur le service de *Lecture de configuration* (voir Initialisation du module de consultation cartographique). Cependant, seule la partie d'interprétation du fichier de contexte (*XMLMapfileParser*) est sollicitée.

Outil de requêtes attributaire

Tout comme l'outil d'interrogation, l'outil de requêtes attributaires s'appuie sur Mapserver utilisé en mode CGI. Plus particulièrement, un service *Query Attribute* est appelé sur le serveur applicatif avec en paramètre les éléments de la requête saisis par l'utilisateur. Ce service envoie une requête CGI à Mapserver : mode ITEMQUERY, avec les critères de requête passés dans le paramètre QSTRING. De la même façon que pour l'outil d'information, le résultat est renvoyé au poste client dans une structure JSON via une requête ajax. L'affichage sur le poste client s'appuie sur les composants de la librairie ExtJS.

Téléchargement

Le téléchargement de données (fichier couche ou table PostGIS) fait appel à un service spécifique *AddDownloadLayerToQueue* localisé sur le serveur de téléchargement. Ce service est paramétré par la liste des options de projection et de format (shp, tab, MIF/MID, KML) pour chaque couche sélectionnée par l'utilisateur. Il s'appuie sur l'outil de conversion ogr.

Le service de téléchargement est organisé sous forme de pile. Toute nouvelle demande de téléchargement (requête ajax) est positionnée en haut de la pile. Lorsque la demande arrive en bas de la pile, la commande ogr est exécutée et un fichier zip regroupant les couches demandées est généré. L'url de ce fichier sur le serveur de téléchargement est envoyée au commanditaire par mail. Les fichiers zip générés restent accessibles X jours sur le serveur avant d'être supprimés.

L'envoi du l'url par mail est réalisée par un service spécifique (classe php).

Ajout de couches WMS depuis le catalogue WMS de CARMEN

L'outil d'ajout de couches WMS externes à la carte initiale s'appuie sur un module *AddWMS* du serveur applicatif. Ce module renvoie une fenêtre listant les serveurs WMS accessibles par l'adhérent. Il interagit avec les librairies OpenLayers (ajout d'une couche à la carte) et LayerTree (ajout en légende).

2.8. Répartition des différents services sur le poste de navigation, le serveur applicatif et le serveur de téléchargement.

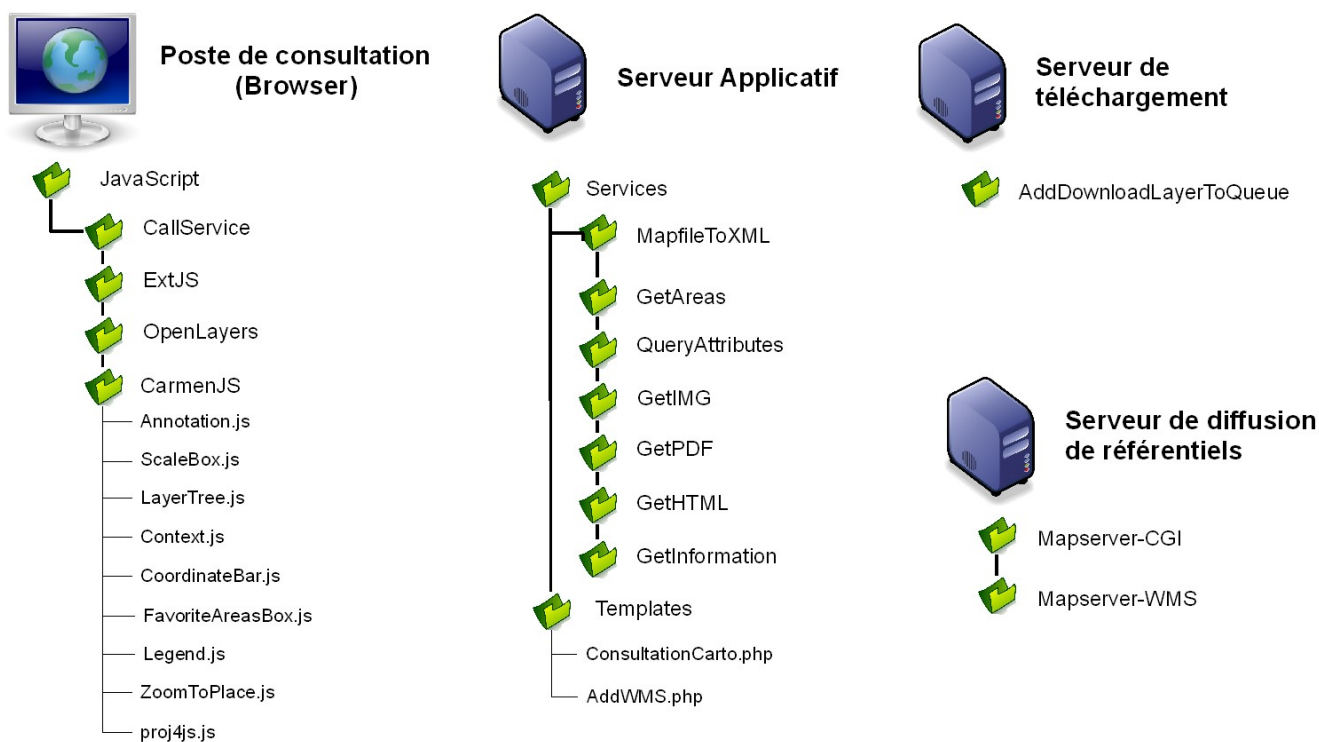


Figure 6: Architecture logicielle des serveurs (application de consultation de cartes)

3. Compléments d'information

3.1. Gestion des services WMS/WFS

Les appels de services WMS dans CARMEN 1.6 se font par le biais d'une URL directe :

[http://carto.ecologie.gouv.fr/WMS/\[SERVICE_IDX\]/\[NOM_CARTE\]](http://carto.ecologie.gouv.fr/WMS/[SERVICE_IDX]/[NOM_CARTE]).

Comme dans la version 1.5, à chaque mapfile peut correspondre un mapfile WMS (wms_[NOM_CARTE].map) offrant un service wms relatif à la carte.

L'URL d'accès du WMS redirige la requête vers un fichier wms.php ([http://carto.ecologie.gouv.fr/wms.php?map=\[NOM_CARTE\]&service_idx=\[SERVICE_IDX\]](http://carto.ecologie.gouv.fr/wms.php?map=[NOM_CARTE]&service_idx=[SERVICE_IDX])) qui lui-même fait appel à la requête WMS :

[http://carto.ecologie.gouv.fr/mapserv?map=/mnt/.../wms_\[NOM_CARTE\]...](http://carto.ecologie.gouv.fr/mapserv?map=/mnt/.../wms_[NOM_CARTE]...)

Les URLs des WMS doivent être libres d'accès pour tout le monde. En revanche, une restriction porte sur l'accès aux mapfiles CARMEN qui ne doivent être accessibles que par l'application elle-même.

Deux éléments garantiront cette sécurité :

- L'accès direct aux URLs WMS sera refusé (dans la configuration d'apache) en dehors des appels provenant du fichier wms.php
- Le fait de passer par le fichier wms.php empêche l'accès vers les mapfiles CARMEN (en raison de la concaténation systématique du nom du mapfile avec le terme « wms »).

Le principe est identique pour les accès WFS.

3.2. Gestion des formats d'image de sortie

Mapserver 5 offre des fonctionnalités d'antialiasing grâce à la librairie AGG. Cette fonction sera implémentée dans CARMEN 1.6. Aussi de nouveaux formats de sortie possibles compléteront les formats existants (png24, png, gif, jpg):

- AGG/JPG,
- AGG/PNG.

Lorsque l'adhérent choisit un format de sortie, l'ensemble des couches mapserver sera généré dans ce format de sortie.

Les anciens formats de sortie sont conservés car même si la librairieAGG offre des possibilités de représentation accrues, elle présente également des performances inférieures à la librairie GD.

A titre d'exemple, dans une carte issue de Carmen 1.5, générée au format gd/png, une requête mapserver cgi aboutit au bout de 393 ms (figure 7) avec le format GD/PNG et 1,17s avec le format AGG/PNG (figure 8).

MapServer Layer - Mozilla Firefox

Fichier Édition Affichage Historique Marque-pages Outils Aide

http://192.168.0.4:23000/openlayers/examples/mapserver_untiled_carmen.htm

MapServer Layer
Shows MapServer Layer

Inspector Effacer | Tous HTML CSS JS XHR Images Flash

Console HTML CSS Script DOM Réseau

Request	Status	Response	Size	Time
GET EditingToolbar.j	200 OK	192.168.0.4:23000	3 KB	1ms
GET Lang.js	200 OK	192.168.0.4:23000	5 KB	1ms
GET en.js	200 OK	192.168.0.4:23000	5 KB	1ms
GET north-mini.png	200 OK	192.168.0.4:23000	484 B	2ms
GET west-mini.png	200 OK	192.168.0.4:23000	453 B	0ms
GET east-mini.png	200 OK	192.168.0.4:23000	451 B	1ms
GET south-mini.png	200 OK	192.168.0.4:23000	481 B	1ms
GET zoom-plus-mini.	200 OK	192.168.0.4:23000	489 B	1ms
GET zoom-world-min	200 OK	192.168.0.4:23000	2 KB	1ms
GET zoom-minus-mir	200 OK	192.168.0.4:23000	359 B	1ms
GET layer-switcher-r	200 OK	192.168.0.4:23000	451 B	1ms
GET layer-switcher-r	200 OK	192.168.0.4:23000	249 B	1ms
GET mapserv?map=/	200 OK	192.168.0.4:23001	?	393ms

183 requêtes 2 MB (34 KB à partir du cache) 1.95s

Terminé

Figure 7: Image générée au format GD/PNG

MapServer Layer - Mozilla Firefox <2>

Eichier Édition Affichage Historique Marque-pages Outils Aide

http://192.168.0.4:23000/openlayers/examples/mapserver_untitled_c

Les plus visités Intranet A L K A N ... tutorial Irisa divers Labos Biblio SIG Mail

MapServer Layer
Shows MapServer Layer

Inspector Effacer Tous HTML CSS JS XHR Images Flash

Console HTML CSS Script DOM Réseau Options

Request	Status	Response	Size	Time
GET EditingToolba	200 OK	192.168.0.4:23000	3 KB	2ms
GET Lang.js	200 OK	192.168.0.4:23000	5 KB	1ms
GET en.js	200 OK	192.168.0.4:23000	5 KB	1ms
GET north-mini.pn	200 OK	192.168.0.4:23000	484 B	1ms
GET west-mini.pn	200 OK	192.168.0.4:23000	453 B	1ms
GET east-mini.png	200 OK	192.168.0.4:23000	451 B	1ms
GET south-mini.pn	200 OK	192.168.0.4:23000	481 B	1ms
GET zoom-plus-mi	200 OK	192.168.0.4:23000	489 B	1ms
GET zoom-world-n	200 OK	192.168.0.4:23000	2 KB	1ms
GET zoom-minus-r	200 OK	192.168.0.4:23000	359 B	1ms
GET layer-switche	200 OK	192.168.0.4:23000	451 B	1ms
GET layer-switche	200 OK	192.168.0.4:23000	249 B	1ms
GET mapserv?map	200 OK	192.168.0.4:23001	?	1.17s

183 requêtes 2 MB (34 KB à partir du cache) 2.93s

Terminé

Figure 8: Image générée au format AGG/PNG

Table des figures

Figure 1: Interface de consultation.....	4
Figure 2: Service de Lecture de configuration.....	4
Figure 3: Interface de gestion de couches / Légende.....	6
Figure 4: Barre d'outils.....	7
Figure 5: Exemple de fenêtre de l'interface de consultation.....	9
Figure 6: Architecture logicielle des serveurs (application de consultation de cartes).....	11
Figure 7: Image générée au format GD/PNG.....	13
Figure 8: Image générée au format AGG/PNG.....	14