



Couches

Les couches sont les 'source de données' dans OpenLayers.

Couches de base et autres

OpenLayers a deux types de couches opérant au sein de votre application : les couches de base et les couches superposées. Cette différence détermine plusieurs aspects de la manière d'interagir avec une carte OpenLayers.

Couches de base

Les couches de base sont mutuellement exclusive, ce qui signifie que seule une couche peut être activée. La couche de base active détermine la projection disponible (le système de coordonnées) et les niveaux de zoom disponible sur la carte.

Le fait qu'une couche soit une couche de base ou non est déterminé par la propriété `isBaseLayer`. Cela peut être modifié par les options de la couche.

Les couches de base s'affichent toujours en dessous des couches de superposition.

Couches de superposition

Les couches qui ne sont pas de base – parfois appelées 'overlays' ou couches de superposition – sont des couches alternatives aux couches de base. Plusieurs couches de superposition peuvent être activée à la fois. Ces couches ne contrôlent pas les niveaux de zoom de la carte, mais peuvent être activée ou désactivée à certaines échelles par les paramètres de résolution/d'échelle min/max.

Certains types de couches overlays gèrent la reprojection vers la projection de la couche de base au moment du chargement de la couche. La plupart des couches de superposition ne sont pas

définies par défaut comme couches de base, comme le fait la classe `Layer`. Les couches qui ne sont pas de base s'affichent au dessus des couches de base.

couches raster

Les couches raster sont des couches d'images. Ces couches sont typiquement dans une projection fixée qui ne peut pas être changé côté client.

Google

Cette couche permet d'utiliser les données Google Maps dans OpenLayers. Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche Google](#). Pour un exemple d'utilisation, voyez l'[exemple Spherical Mercator](#).

Si vous superposez d'autres données sur une couche de base de Google Maps, vous voudrez interagir avec la couche de Google Maps dans des coordonnées projetées (ceci est particulièrement important si vous travaillez avec des données d'imagerie). Vous pouvez lire plus d'information sur la 'projection sphérique Mercator que Google Maps - et d'autres fournisseurs commerciaux - utilise dans la documentation *spherical Mercator*.

La classe de la couche Google est conçue pour être seulement utilisée en tant que couche de base.

Image

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche image](#).

KaMap

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche KaMap](#).

KaMapCache

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche KaMapCache](#).

MapGuide

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche MapGuide](#).

MapServer

Cette couche n'est pas nécessaire pour interagir avec MapServer. En général, la couche *WMS* est préférée à la couche MapServer. Puisque MapServer expose la plupart de ses fonctionnalités CGI également en mode WMS, la couche WMS est préférable. La couche MapServer peut souvent mener à des cartes qui semblent fonctionner, mais ne fonctionnent pas en raison de problèmes de projection ou d'autres erreurs de configuration similaire. Sauf si vous avez une bonne raison de ne pas le faire, vous devriez utiliser Layer.WMS au lieu de Layer.MapServer.

Si vous utilisez Layer.MapServer, et que votre carte est répétée plusieurs fois, cela indique que vous n'avez pas configuré correctement votre carte pour être dans une projection différente. OpenLayers ne peut pas lire cette information à partir de votre mapfile, et il doit être configuré explicitement. La [FAQ sur la configuration des différentes propriétés de projection](#) fournit des informations sur la manière de configurer différentes projection dans OpenLayers.

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche MapServer](#).

MultiMap

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche MultiMap](#).

TMS

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche TMS](#).

TileCache

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche TileCache](#).

VirtualEarth

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche VirtualEarth](#).

WMS

Type de couche pour accéder des données servie selon le standard Web Mapping Service.

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche WMS](#).

WorldWind

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche WorldWind](#).

Yahoo

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche Yahoo](#).

Couches de superposition

Les couches de superposition peuvent être n'importe quelles couches ayant leur source de données dans un autre format que sous forme d'image. Cela inclut les sous-classes des couches [Markers](#) et [Vecteur](#). Pour plus d'information sur les différences entre ces deux classes de base, lisez la documentation sur [Overlays](#).

Boxes

Basé sur une surclasse de markers. En général, il est plus intéressant d'implémenter cette fonctionnalité avec une couche vecteur et des géométries polygonales. Maintenu pour une compatibilité arrière.

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche Boxes](#).

GML

La couche GML est une surclasse de la couche vecteur pour lire les données à partir d'un fichier

et l'afficher. Elle est idéale pour travailler avec plusieurs formats, pas seulement avec GML, et peut être configuré pour lire d'autres formats via l'option 'format' de la couche.

Le cas d'utilisation le plus simple des couches GML est simplement de charger un fichier GML. L'exemple de couche GML montre cela : ajoutez simplement :

```
var layer = new OpenLayers.Layer.GML("GML", "gml/polygon.xml")
map.addLayer(layer);
```

Si vous voulez ajouter un type de format différent, vous pouvez changer l'option format de la couche au moment de l'initialisation. L'exemple KML montre cela :

```
var layer = new OpenLayers.Layer.GML("KML", "kml/lines.kml", {
  format: OpenLayers.Format.KML
})
map.addLayer(layer);
```

Vous pouvez aussi ajouter formatOption à la couche : ces options sont utilisées lors de la création de la classe format en interne à la couche :

```
var layer = new OpenLayers.Layer.GML("KML", "kml/lines.kml", {
  format: OpenLayers.Format.KML,
  formatOptions: {
    'extractStyles': true
  }
});
map.addLayer(layer);
```

Les options du format sont déterminées par la classe format.

Pour toutes informations sur l'API, lisez la [documentation de l'API sur la couche GML](#).

GeoRSS

La couche GeoRSS utilise le format GeoRSS et affiche les résultats comme 'markers' cliquable. C'est une surclasse de la couche Markers qui ne gère pas les lignes ni les polygones. Plusieurs de ses comportements sont codés en dur, et il est en général préférable d'utiliser une couche GML avec un contrôleur SelectFeature à la place de la couche GeoRSS si vous voulez pouvoir

configurer le comportement de votre application (Pour plus d'information sur comment réaliser cette transition, lisez *Transition d'une couche Texte ou GeoRSS vers une couche vecteur.*)

Pour toutes informations sur l'API, lisez la documentation de l'API sur les couches GeoRSS.

Markers

La couche de base Markers est assez simple et permet d'utiliser la fonction `addMarkers` pour ajouter des markers à la couche. Elle gère seulement les points, pas les lignes et les polygones.

Pour toutes informations sur l'API, lisez la documentation de l'API sur les couches Markers.

PointTrack

Pour toutes informations sur l'API, lisez la documentation de l'API sur les couches PointTrack.

Texte

La couche Texte utilise le format Texte et affiche les résultats comme marqueurs cliquables. C'est une surclasse de la couche Markers qui ne gère pas les lignes ni les polygones. Plusieurs de ses comportements sont codés en dur, et il est en général préférable d'utiliser une couche GML avec un Contrôleur `SelectFeature` à la place de la couche Text si vous voulez pouvoir configurer le comportement de votre application (Pour plus d'information sur comment réaliser cette transition, lisez *Transition d'une couche Texte ou GeoRSS vers une couche vecteur.*)

Pour toutes informations sur l'API, lisez la documentation de l'API sur les couches Texte.

Vecteur

La couche Vecteur est la base de la gestion avancées des géométries dans OpenLayers. Les classes comme GML et WFS sont des surclasses de la couche Vecteur. Lors de la création d'objets dans le code JavaScript, la bonne pratique est généralement d'utiliser directement la couche Vecteur.

À partir d'OpenLayers 2.7, le développement de l'extension de la couche Vecteur a débuté pour avoir des fonctionnalités supplémentaires pour charger des données, remplacer le grand nombre

de surclasse de couches. Ce travail sur les classe Strategy et Protocol est pensé pour faciliter l'interaction avec les données à partir de sources distantes. Pour plus d'information sur les stratégies et les protocoles, lisez la documentation de l'API d'OpenLayers.

Pour toutes informations sur l'API, lisez la [documentation de l'API sur les couches Vecteur](#).

WFS

Pour toutes informations sur l'API, lisez la [documentation de l'API sur les couches WFS](#).

Sur-classes génériques

- EventPane
- FixedZoomLevels
- Grid
- HTTPRequest
- SphericalMercator