

**Titre:** Mise en œuvre de mod\_WebObjects

**Version:** 2.0

**Dernière modification:** 2010/09/06 20:00

**Auteur:** Aurélien Minet <aurelien dot minet at cocktail dot org>

**Statut:** version finale

**Licence:** Creative Commons - by-nc-sa 2.0

( <http://creativecommons.org/licenses/by-nc-sa/2.0/fr/> )

Remarques:

- les pré-requis sont de savoir installer et configurer Apache, maîtriser l'installation des paquets de la distribution que vous utilisez, avoir des notions de compilation en C et des l'outil: make, libtool.
- le document est "orienté" Apache sous Linux mais la procédure d'installation reste applicable à d'autres Unix
- rien dans ce document n'est supporté par Apple
- ce document dans sa globalité n'est pas spécifique Cocktail
- au niveau de la typographie: le code, les lignes de commandes, le contenu des fichiers sont en italique
- la dernière version est disponible [ici](#)

# Mise en œuvre de mod\_WebObjects



## **Versions:**

date	n°version	commentaires
10/07/2008	0	brouillon
30/07/2008	1.0	version initiale
06/09/2010	2.0	Mise à jours suite la création d'un nouveau package utilisant exclusivement apxs

Les version au format X.y.z ne sont pas listées, elles ne contiennent que des corrections (coquilles ....). L'information contenue ne change pas.

<b>Objectif:</b> .....	<b>5</b>
<b>Introduction:</b> .....	<b>5</b>
<b>1. Rôle et fonctionnement</b> .....	<b>6</b>
<b>2. Installation</b> .....	<b>9</b>
<b>3. Configuration</b> .....	<b>10</b>
<b>Conclusion</b> .....	<b>12</b>

## Objectif:

- Comprendre le fonctionnement de mod\_WebObjects
- Installer mod\_WebObjects
- Configurer mod\_WebObjects

## Introduction:

Apple livrait avec WebObjects 5.2 les sources de mod\_WebObjects sous "APPLE PUBLIC SOURCE LICENSE" en différentes versions: IIS, Apache 1.3 et CGI. La Licence permet la modification, c'est pourquoi le support d'Apache 2 et 2.2 a été rajouté par le projet Wonder en plus de divers corrections.

Avec WebObjects 5.4 Apple a livré une version reprenant les corrections et supportant officiellement apache 2 et 2.2, le projet Wonder a de nouveau apporté des corrections. L'installation peut se baser sur la version XCode 3.0, qui est compatible (niveau des URL) avec les anciennes versions de WebObjects, ou sur la version Wonder qui est une version patchée de la version XCode 3.0 comprenant des corrections de bugs notamment pour x86\_64. Nous utiliserons donc une version repackagé, pour faciliter la compilation, issue des sources de Wonder mais uniquement destinée a Apache 2.2

# 1. Rôle et fonctionnement

mod\_WebObjects peut aussi être appelé Adaptor (ou encore HTTP Adaptor) étant donné qu'il reçoit les requêtes http qu'il transmet à l'application afin d'obtenir une réponse qu'il renvoie au client. Et que cette dernière à un WOAdaptor qui par défaut est le WODefaultAdaptor, il fournit le support http à l'application c'est grâce à lui que l'on peut se connecter directement à l'application via son port via un navigateur web (il existe d'autres WOAdaptors).

Ainsi mod\_WebObjects fait interface entre les clients et les applications, il transforme donc le serveur http en proxy. Mais pas seulement car il peut découvrir applications avec leurs instances, c'est pourquoi le proxy est dynamique et qu'il fait load-balancer lorsqu'il y a différentes instances pour une même application.

Pour cela il fonctionne avec 2 listes:

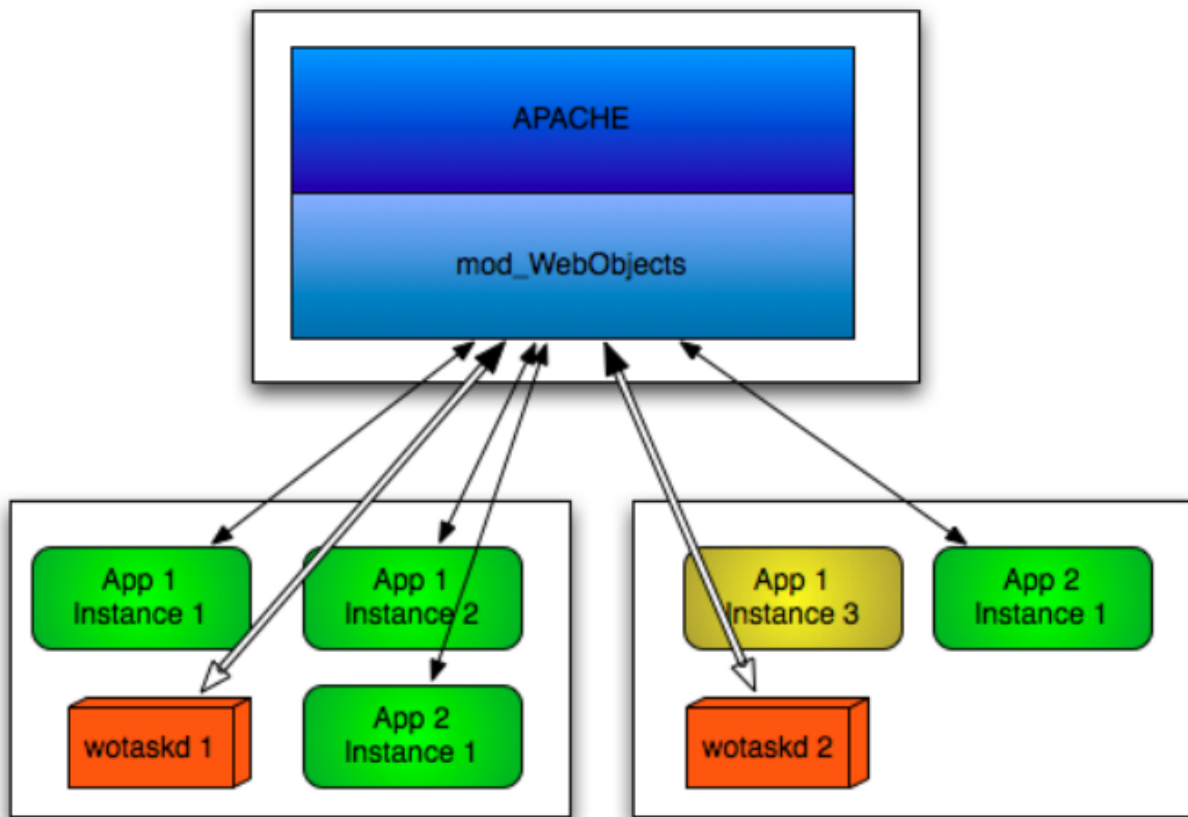
- une liste de serveurs d'applications
- une liste d'instances disponibles pour chacun des serveurs de la liste précédente.

Ces 2 listes peuvent d'être dynamiques ou statiques, on obtient 3 configurations possibles:

- multicast: découverte des serveurs dynamique (liste des instances forcément dynamique)
- multihosts: la liste de serveurs fixe (n wotaskd de déclarés) mais la liste des instances est dynamique
- statique: les 2 listes sont fixes (instances disponibles ou non)

La configuration multihosts est recommandée. Dans cette configuration mod\_WebObjects va donc régulièrement contacter chaque wotaskd pour obtenir la liste des instances disponibles.

Exemple:



Sur le site, il y a 2 deux serveurs, mod\_WebObjects contacte les wotaskds de chaque serveur et obtient un liste de 4 instances correspondant à 2 applications:

- App1 a 4 instances: 3 sont démarrées sur le premier serveur, la 4ième est déclarée mais non démarrée sur le deuxième serveur.
- App2 a 1 seule instance et elle est sur le deuxième serveur.

Exemple de ce que connaît au final mod\_WebObjects si toutes les instances était démarrées dans l'exemple précédent.

```
<?xml version="1.0" encoding="ASCII"?>
<adaptor>
  <application name="app1" retries="5" scheduler="ROUNDROBIN" urlVersion="4">
    <instance id="1" port="2001" host="appserver1.mydomain.com"
sendTimeout="3" recvTimeout="10"/>
    <instance id="2" port="2002" host="appserver1.mydomain.com"/>
    <instance id="3" port="2003" host="appserver1.mydomain.com"/>
    <instance id="4" port="2001" host="appserver2.mydomain.com"/>
  </application>
  <application name="app2" retries="1" scheduler="RANDOM" urlVersion="4">
    <instance id="1" port="2002" host="appserver2.mydomain.com"/>
  </application>
</adaptor>
```

Par instance il peut y avoir des paramètres spécifiques. Ce fichier XML correspond à la configuration statique.

Apache avec mod\_Webobjects peut être sur l'un des serveurs WebObjects ou sur un autre serveur. Il peut même y avoir plusieurs serveur pour Apache avec mod\_WebObjects, chacun contactant les wotaskds des serveurs d'applications indépendamment.

## 2. Installation

pré-requis:

- Apache avec le **mpm prefork** (et les headers correspondants)
- APR (Apache Portable Runtime) et APR-utils
- gcc >= 3.4
- glibc >= 2.2

L'ancienne méthode utilisait les versions issues d'Xcode 3.0 (/Developer/Examples/WebObjects/Sources) ou de Wonder (version patché de celle d'XCode 3.0), cela passait par l'utilisation d'un fichier make.config pas forcément très pratique, la compilation était gérée par des Makefile.

La version repackagée Cocktail est celle de WOnDer mais que pour Apache 2.2, les fichiers nécessaires sont dans un seul et même répertoire. Ainsi après avoir décompressé l'archive il faut lancer le script **build.sh** présent dans le répertoire des sources.

Il va lancer APache eXtenSion tool qui va se charger de toute la compilation.

Il peut y avoir quelques warning notamment pour la fonction xmlTokenizerNextToken, ne sont pas problématique.

S'il n'y a pas eu de problème mod\_WebObjects.so est dans le sous-répertoire "caché" .libs

Il faut le copier dans dans /usr/lib/apache2/modules ou plus exactement dans celui contenant les modules d'apache, l'emplacement peut varier d'une distribution a l'autre. La fin de *build.sh* vous indique la commande de copie et ce qu'il reste a faire.

### 3. Configuration

Tout d'abord vérifier qu'Apache est configuré avec comme "moteur", i.e. Multi-Processing Modules (MPMs), le `mpm_prefork` et non le `worker`, cela en cas de forte charge peut causer des problèmes de stabilité.

Copier le fichier `mod_webobjects.conf` dans le sous répertoire contenant la configuration des modules ( `/etc/apache2/modules.d/` ou `/etc/httpd/conf.d` : fonction de votre distribution).

Ensuite il faut éditer `mod_webobjects.conf` :

- commenter la ligne "`LoadModule WebObjects_module modules/mod_WebObjects.so`" si votre distribution centralise tout les `LoadModule` dans un fichier unique (`httpd.conf` ou `apache.conf`)
- éventuellement corriger la valeur pour `WebObjectsDocumentRoot` même si l'adaptor n'utilise pas cette valeur.
- choisir le mode de fonctionnement en changeant la valeur de `WebObjectsConfig`, il est recommandé d'utiliser le `multihost` qui a la forme:  
`WebObjectsConfig http://<name-of-a-host>:<port-on-a-host>,http://<name-of-another-host>:<port-on-a-host> <interval>`
- activer la page d'information <http://webservice/cgi-bin/WebObjects/WOAdaptorInfo?user+password> en décommentant/changeant `WebObjectsAdminUsername` et `WebObjectsAdminPassword`
- décommenter `WebObjectsLog` afin de pouvoir activer les logs lorsque `/tmp/logWebObjects` existe
- on peut ajouter: `WebObjectsOptions redir=http://.../erreur.html` pour personnaliser la page d'erreur lorsque l'application demandée n'existe pas ou qu'elle n'a pas d'instance démarrée.

Puis il faut modifier la configuration d'apache:

- il faut charger le module `WebObjects` (soit dans `mod_webObjects.conf` soit dans `httpd.conf` ou autre selon votre distribution)

```
LoadModule WebObjects_module modules/mod_WebObjects.so
```

- éventuellement si vous utilisez `mod_Rewrite`, il est préférable d'avoir les `LoadModule` dans cette ordre (si les `LoadModule` sont centralisés dans un seul fichier)

```
LoadModule WebObjects_module modules/mod_WebObjects.so
LoadModule rewrite_module modules/mod_rewrite.so
```

- il faut commenter la ligne `ScriptAlias /cgi-bin/ "/path/to/cgi/dir..."` si elle existe.
- il faut éventuellement changer les droits d'accès aux répertoires:  
par défaut il y peut y avoir:

```
<Directory / >
    Order deny,allow
    Deny from all
</Directory>
```

(ce qui est très sécuritaire)

- donc il faut inverser les droits

```

<Directory / >
    Order allow,deny
    allow from all
</Directory>
```

(ce n'est pas vraiment l'idéal)

ou ajouter:

```

<LocationMatch /cgi-bin/WebObjects/.*>
    Order allow,deny
    Allow from all
</LocationMatch>

<Location /WebObjects>
    Order allow,deny
    Allow from All
</Location>
```

(ce qui mieux niveau sécurité)

- enfin il faut s'assurer que mod\_webobjects.conf est bien chargé par une directive Include ( Include /.../webobjects.conf ou Include /etc/apache2/modules.d/\* .conf )

Il ne reste plus qu'a vérifier que l'Adaptor est bien chargé après avoir redémarré Apache

- vérifier que <http://webserver/cgi-bin/WebObjects/WOAdaptorInfo?user+password> fonctionne et indique la bonne configuration.
- une URL du type <http://webserver/cgi-bin/WebObjects/AppQuiNExistePas> doit afficher "The requested application was not found on this server." (ou la page d'erreur maison) si le mod\_WebObjects tourne et s'il est accessible.
- après `touch /tmp/logWebObjects` le fichier /tmp/WebObjects.log doit être créé par mod\_WebObjects ce qui indique qu'il est chargé. Le contenu peut aider à l'analyse de certains problèmes.

Aussi il ne faudra pas oublier de copier les WebServerResources dans /.../htdocs/WebObjects, signer les jar dans /.../htdocs/WebObjects/Java ...

## Conclusion

La dernière version de mod\_WebObjects est plus simple à compiler, la configuration est bête et méchante: il faut respecter certains points précis. Il y a encore pas mal de limitations (pas de gestion virtual host ...) mais ce module est critique pour le déploiement et la disponibilité des applications, prendre un peu de temps pour l'installer est nécessaire. Aussi il est fortement recommandé de lire la documentation officielle d'Apple sur l'Adaptor dans le [WebObjects Deployment Guide](#) .