

InpresZon

Raphaël Javaux

Résumé

Dossier d'aide à l'installation des différents composants de la plateforme d'e-commerce « InpresZon ».

Table des matières

1	Présentation des différents composants	2
2	Sites web	3
3	Bases de données	4
4	Importation CouchDB	5
5	Importation fnac.com	6
6	Nouveaux stocks	7
7	Data Warehouse	7
A	Sauvegardes	8

1 Présentation des différents composants

La plate-forme d'e-commerce « InpresZon » utilise divers logiciels pour ses différents composants :

Site web Les deux instances du site web (une américaine et une belge) sont conçues à l'aide du framework web Django¹, utilisant le langage de programmation Python².

Bases de données Les quatre bases de données (une américaine, une belge, une anglaise ainsi qu'un dépôt de Data Warehouse) utilisent le SGBDR Oracle 11g Release 2³. La communication entre les bases de données et les sites web se fait à l'aide du pilote cxOracle⁴.

Importation CouchDB Un programme console permet l'importation de produits stockés dans une base de données CouchDB⁵. Ce programme, écrit en Haskell⁶, est compilable à l'aide de GHC⁷.

Importation fnac.com Un second programme console permet l'importation d'articles (musiques, films ou livres) vendus sur le site <http://www.fnac.com>. Ce dernier est écrit en F#⁸ et extrait les informations des produits sous forme de documents XML. Un script Bash utilisant le programme ncftp⁹ a été créé pour transmettre ces documents aux différents dépôts XDB des bases de données.

Nouveaux stocks L'envoi des fichiers XML représentant les nouveaux arrivages en stock se fait à l'aide d'un script Bash utilisant, à nouveau, ncftp.

Data Warehouse Un dépôt de Data Warehouse, utilisé à des fins de data mining, est alimenté à l'aide :

- de deux procédures stockées PL/SQL et de DB/LINK homogènes pour les bases de données belge et anglaise ;

1. <https://www.djangoproject.com/>

2. <http://www.python.org/>

3. <http://www.oracle.com/us/products/database/enterprise-edition/>

4. <http://cx-oracle.sourceforge.net/>

5. <http://couchdb.apache.org/>

6. <http://www.haskell.org/>

7. <http://www.haskell.org/ghc/>

8. <http://research.microsoft.com/fsharp>

9. <http://www.ncftp.com/>

- d'un programme écrit avec le langage de programmation Clojure¹⁰ pour la base de données américaine, générant des fichiers XML pouvant être transmis au dépôt XDB de la base de données de Data Warehouse à l'aide d'un script Bash.

Les différentes opérations de data mining peuvent être effectuées à l'aide d'un programme console Scala¹¹.

2 Sites web

Python Il est nécessaire d'installer la version 2 de l'interpréteur Python¹². Celui-ci est déjà installé sur la plupart des distributions GNU/Linux (il se trouve dans les dépôts le cas échéant) ainsi que sur Mac OS X. L'installateur pour Microsoft Windows est disponible sur <http://www.python.org/>¹³.

Django Le framework Django est disponible dans les dépôts des distributions GNU/Linux. Si ce n'est pas le cas, on peut l'obtenir sur le site officiel : <https://www.djangoproject.com/>.

Serveur web Un serveur web est fourni avec Django. Pour lancer le serveur web, il est nécessaire de se positionner dans le répertoire du site web à servir (*be/* ou *usa/*) et d'exécuter la commande suivante :

```
$ python2 manage.py runserver 8080
```

Ceci va démarrer un serveur web sur le port 8080.

Il est également possible d'utiliser un site web créé avec Django sur le serveur web Apache à l'aide du module WSGI¹⁴.

10. <http://www.clojure.org/>

11. <http://www.scala-lang.org/>

12. Django et le pilote Oracle ne sont pas encore compatibles avec Python 3.

13. L'ensemble de l'installation n'a été testée que sous GNU/Linux. Il est fort probable que quelques démarches supplémentaires soient nécessaires sur d'autres systèmes.

14. <http://code.google.com/p/modwsgi/>

3 Bases de données

Installation La base de données Oracle 11g Release 2 est disponible en téléchargement sur le site d'Oracle¹⁵ pour une utilisation non commerciale.

Pilote cxOracle Il est nécessaire d'installer le pilote cxOracle pour faire communiquer les sites web avec les bases de données. Le pilote est téléchargeable à l'adresse <http://cx-oracle.sourceforge.net/>. Si la base de données Oracle n'est pas installée sur la même machine que celle hébergeant les sites web, il sera nécessaire d'installer l'InstantClient d'Oracle¹⁶.

Utilisateurs Il est nécessaire de créer quatre utilisateurs pour les quatre bases de données : *be*, *usa*, *uk* ainsi que *marketing*. Par défaut, les différents composants sont configurés pour utiliser le mot de passe *pass* pour l'ensemble de ces comptes. Celui-ci est modifiable dans les fichiers *settings.py* des répertoires *be*, *usa*, *uk* et *marketing*. Les utilisateurs *be*, *usa* et *marketing* doivent posséder les droits sur un dépôt XDB¹⁷.

Tables Les tables sont créées à partir de classes Python à l'aide de Django. Cette manière de procéder a plusieurs avantages :

- la description des tables est indépendante du logiciel de gestion de base de données utilisé (Django est officiellement compatible avec Oracle, SQLite, PostgreSQL et MySQL) ;
- Django est capable de générer des formulaires HTML à partir de ces classes ainsi que les vérifications associées (email valide, longueur des chaînes ...), ce qui évite certaines tâches répétitives ainsi que les erreurs de programmation qui y sont associées ;
- Django fournit un langage pour écrire des requêtes (utilisant des objets Python) indépendant du SGBD utilisé (fonctionnalité non employée pour InpresZon).

Pour créer les tables, il suffit de se positionner dans le répertoire de la base de données (*be/*, *usa/*, *uk/* ou *marketing/*) et d'exécuter la commande suivante :

```
$ python2 manage.py syncdb
```

15. <http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html>

16. <http://www.oracle.com/technetwork/database/features/instant-client/>

17. Voir documentation sur l'installation et la configuration d'un dépôt XDB.

Packages Le répertoire *packages/* contient les scripts SQL utilisés pour créer les différents packages des quatre bases de données. Le script *exceptions.sql* doit être exécuté sur les bases *be* et *usa* et contient le module de gestion des exceptions.

4 Importation CouchDB

Haskell Le programme d'importation doit être compilé avec un compilateur Haskell. GHC est le compilateur le plus utilisé et est disponible sur la plupart des systèmes d'exploitation. Un paquetage contenant GHC ainsi que divers outils est mis à disposition sous le nom d'« Haskell Platform » à cette adresse : <http://hackage.haskell.org/platform/>. La plupart du temps, ce dernier est directement disponible dans les dépôts des distributions GNU/Linux.

ODBC Le programme Haskell d'importation des articles utilise ODBC pour accéder aux bases de données Oracle. Sous GNU/Linux, c'est le programme unixODBC (disponible sur la plupart des distributions dans les dépôts) qui fournit la connexion ODBC.

Il est indispensable de configurer un driver nommé « oracle », car c'est celui-ci qui sera utilisé par le programme d'importation, dans le fichier */etc/odbcinst.ini* :

```
[ oracle ]
Description      = Oracle ODBC driver for Oracle 11g
Driver           = /opt/instantclient/libsqora.so.11.1
Setup            =
FileUsage        =
CPTimeout        =
CPReuse          =
```

La variable *Driver* doit pointer vers le pilote *libsqora.so* de l'InstantClient ou de l'installation d'Oracle.

Haskell Platform n'est pas fourni avec les bibliothèques permettant la communication avec les drivers ODBC et la base CouchDB, cependant on peut utiliser la commande *cabal* (le gestionnaire des libraires de Haskell) pour installer celles-ci :

```
$ cabal update
```

```
$ cabal install HDBC-odbc
$ cabal install CouchDB
```

Cabal va se charger de récupérer les sources des libraires, de les compiler et de les installer.

Utilisation Le programme d'importation CouchDB se nomme *alimentation.hs* et se compile avec GHC :

```
$ ghc -O2 alimentation.hs
```

Ceci va créer un exécutable nommé *alimentation*.

Il est possible de changer les paramètres de connexion aux serveurs Oracle et CouchDB dans les premières lignes du fichier *alimentation.hs*¹⁸.

5 Importation fnac.com

F# Le programme d'importation est écrit en F#. Le compilateur F# est disponible sur le site de Microsoft : <http://research.microsoft.com/fsharp>. Celui-ci est déjà inclus avec Visual Studio 2010. Sous GNU/Linux, il faudra en plus installer Mono¹⁹ (disponible dans la plupart des dépôts) pour exécuter des binaires .NET. Microsoft supporte F# sous Windows et GNU/Linux.

ncftp ncftp est un programme console permettant de se connecter à un serveur FTP depuis le terminal. Il est installable depuis les dépôts de la plupart des distributions GNU/Linux.

Utilisation Le programme d'importation se nomme *alimentation_indep.fs* et se compile avec cette commande (ou via Visual Studio pour Windows) :

```
$ fsharpc -O -r System.Xml.Linq.dll alimentation_indep.fs
```

Ceci va créer un exécutable console nommé *alimentation_indep.exe* exécutable directement sous Windows ou avec la commande *mono* sous GNU/Linux.

L'envoi des fichiers XML sur le dépôt XDB se fait avec le script Bash *alimentation_indep.xml.sh*.

18. Toute modification exigera une re-compilation du fichier *alimentation.hs*.

19. <http://www.mono-project.com/>

6 Nouveaux stocks

L'arrivage de nouveaux stocks se fait à l'aide du script Bash utilisant ncftp. Le script se nomme *nouveaux_stocks.sh* et s'utilise comme suit :

```
$ ./nouveaux_stocks.sh fichier_arrivage.xml
```

7 Data Warehouse

Chargement des données depuis BE et UK Le package *Chargement-Donnees* contient deux procédures, *ChargementBe* et *ChargementUk* pour importer dans le Data Warehouse les données des bases belge et anglaise, respectivement.

Chargement des données depuis USA

Clojure Le chargement des données se trouvant dans USA se fait à l'aide d'un programme écrit pour le langage de programmation Clojure. Clojure utilise la JVM. Le compilateur Clojure ainsi que la JVM sont disponibles dans la plupart des dépôts GNU/Linux. Il est de plus nécessaire d'installer et de configurer JDBC pour Oracle²⁰.

Utilisation Le programme de chargement se nomme *chargement_usa.clj* et s'exécute avec Clojure :

```
$ clj chargement_usa.clj
```

Consultation des résultats

Scala Le programme console écrit en Scala permet d'effectuer diverses opérations de forage de données sur le Data Warehouse. Tout comme Clojure, Scala utilise la JVM et JDBC et son compilateur est disponible sur la plupart des dépôts GNU/Linux.

Utilisation Le programme de forage se nomme *forage.scala* et s'exécute avec Scala :

```
$ scala forage.scala
```

20. Voir syllabus sur JDBC.

A Sauvegardes

Etant donné la nature financière des transactions effectuées sur la plate-forme d'e-commerce, il est nécessaire d'établir des méthodes de sauvegardes robustes.

Cependant, deux remarques viennent à l'esprit :

- les produits des bases de données américaine et anglaise sont recopiés sur le dépôt belge à une cadence de vingt articles ;
- les commandes et les utilisateurs sont recopiés en temps réel entre les bases belge et américaine.

Ainsi, certaines données sont déjà dupliquées à deux endroits différents²¹. Ceci est particulièrement intéressant dans le cas des commandes et des utilisateurs, car il s'agit clairement des données les plus délicates du système, et elles sont répliquées en temps réel (c'est à dire sans risque de perte pour ces tables lorsque seulement l'un de ces deux dépôts rencontre un problème).

Néanmoins, ça n'est pas suffisant, car même s'il est très peu probable que les deux dépôts rencontrent une panne simultanément, il serait assez aisé à un pirate de s'attaquer à la seconde base s'il a réussi à s'introduire dans la première (celles-ci sont quasi identiques et de plus liées par un DBLINK).

La première chose à faire semble être de configurer les bases de données en mode ARCHIVELOG afin de permettre des sauvegardes à chaud²² incrémentales cohérentes. La sauvegarde à chaud devrait se faire durant les heures creuses. Les tables liées aux utilisateurs et leurs commandes devront être sauvegardées plus régulièrement, même si cela se fera au détriment des performances.

21. A la condition que les différentes bases soient installées sur des machines distinctes.

22. Il est impensable de stopper la plate-forme durant plusieurs minutes lors des sauvegardes.