

OpCall

Documentation technique

Table des matières

1-Introduction	2
2-Récupération des données MP3&CSV	3
2.1-Objectifs :	3
2.2-Principe de la mise en œuvre :	4
3-Création de la SGBD	5
3.1-Objectifs :	5
3.2-Principe de la mise en œuvre :	5
4-Création de l'application WEB	7
4.1-Objectifs :	7
4.2-Principe de la mise en œuvre :	7
5-Annexes	9
5.1 Référentiel Git Projet :	9
5.2 Lancements des batchs :	11
5.3 Pour la suite :	11
Mettre en place dans la SGBD OPCALL :	11

Mettre à jour par la suite la connexion ftp en sftp:

12

1-Introduction

Le projet opcall a pour but de récupérer les **écoutes téléphoniques de la DGO** qui sont gérées par le prestataire **Eloquent**, qui seront ensuite stockées dans notre **serveur de stockage** puis indexées grâce aux fichiers CSV dans la **base de données Opcall**.

Par la suite il y aura une **API web** ou certains **utilisateurs pourront consulter ces écoutes en indiquant leur raison**, pour **l'administrateur** il pourra à la fois consulter mais aussi voir les utilisateurs qui ont consultés ces écoutes.

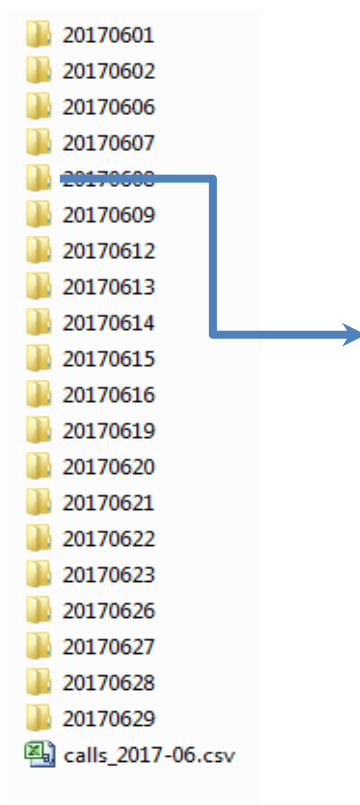
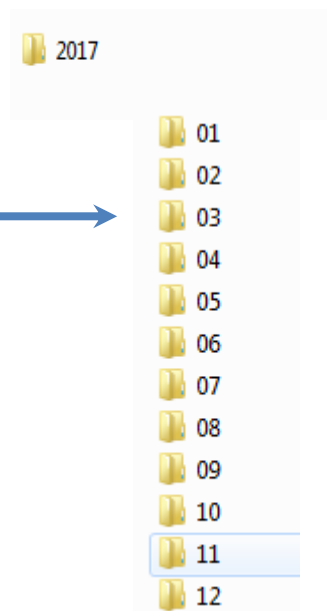
2-Récupération des données MP3&CSV

2.1-Objectifs :

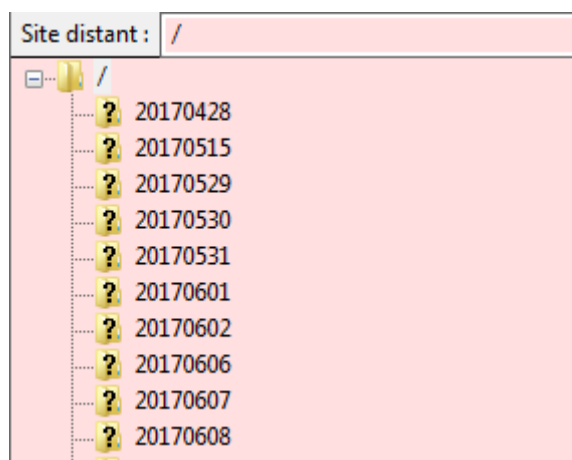
Tout d'abord la récupération des fichiers mp3 doit se faire avec le **PHP à travers des fonctions FTP** puisque le serveur du prestataire Eloquant est **un serveur FTP**.

En respectant l'arborescence côté serveur, ainsi de notre **serveur de stockage**, où seront placés les fichiers mp3 & d'indexation CSV.

Arborescence serveur de stockage :



Arborescence serveur Eloquant :



2.2-Principe de la mise en œuvre :

Pour cela nous mettons 2 batch de récupération, un pour le **mp3** et l'autre pour **CSV**.

Batch MP3 :

La solution mise en place est de respecter ces deux arborescences qui sont compléments différents pour l'adapter dans notre code, en utilisant les fonctions FTP, la récursivité, les expressions régulières et les futurs problèmes pouvant être occasionnés.

Config ftp mp3 : [config ftp.php](#)

```
$ftp_server = "ftp.eloquant.com";  
$ftp_port = 21;  
$user_ftp = "opcalim";  
$password_ftp = ".....";
```

[Schéma récupération mp3.docx](#)

Batch CSV :

De même il faut se connecter au serveur https, puisque les récupérations des fichiers csv ce passe en https via au PHP.

Config http csv : [config https.php](#)

```
$user_https="OPCALIM_ADM";  
$password_https=".....";
```

[Schéma récupération csv.docx](#)

3-Création de la SGBD

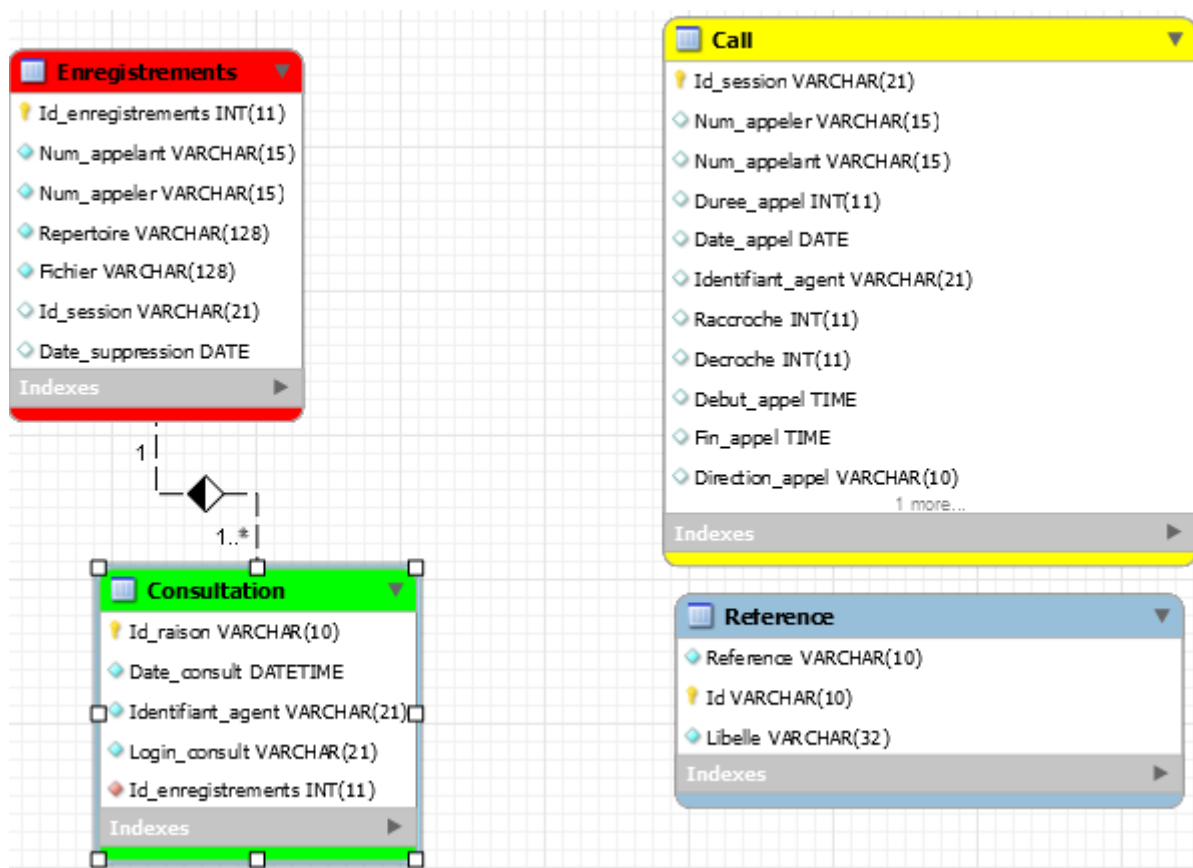
3.1-Objectifs :

L'objectif est de mettre en place une solution pour faire **des indexations** de données par l'intermédiaire des **fichiers mp3 et csv** stockés dans le serveur de **stockage Eloquant**.

3.2-Principe de la mise en œuvre :

Premièrement il y a la conception du modèle de la base de données **opcall** avec l'outil MYSQL WORKBENCH.

Modèle conceptuel d'Opcall :



opcall.mwb (Prérequis : disposer de WORKBENCH)

Ensuite création de la base de données opcall pour faire l'indexation des données des mp3 associées à celles des fichiers CSV.

Script opcall : [opcall.sql](#)

Config opcall : [config_opcall.php](#)

```
$MySQLSvr="vm-035";  
$MySQLUser="opcall";  
$encode="..."  
$MySQLPasswd=base64_decode($encode);  
$MySQLDB="opcall";
```

Pour finir, application du code PHP pour automatiser les insertions des données mp3 et csv à la SGBD opcall.

Indexation MP3 : [Schéma indexation mp3.docx](#)

Indexation CSV : [Schéma indexation csv.docx](#)

4-Création de l'application WEB

4.1-Objectifs :

L'objectif a pour but de créer une API WEB, il y aura partie ou **les utilisateurs** pourront **consulter les écoutes téléphoniques** en donnant un motif avant d'écouter, ensuite il y aura une **partie administrateur** qui pourront seulement voir **qui à écouter cette enregistrement téléphonique** (les utilisateurs)

Pour finir une autre partie avec **un groupe de personnes** sera à la fois **administrateur et utilisateur**.

4.2-Principe de la mise en œuvre :

Les prérequis dans cette API WEB est l'utilisation des Frameworks de jQuery et jQuery ui (javascripts) et de Bootstrap (css).

Voir: [head include.php](#) (head des pages).

Le contrôle d'accès des pages se font par authentification Windows via annuaire d'opcalim.

Un groupe global :

APP-OPCALL->index.html

GROUPE

Deux sous groups pour gérer des droits distincts:

APP-OPCALL-Admins (user_opcalim / user_opcalim) ->
Page_admin.php

APP-OPCALL-Consultations (user_opcalim / user_opcalim) -
>Page_utilisateur.php

Formulaire de saisie utilisateur:

The screenshot shows a user registration form with the following fields and constraints:

- Numéro appelé:** Input field containing "062241458821747444". Constraint: "Pas plus de 15 caractères".
- Numéro appelant:** Input field containing "066524545425421212". Constraint: "Pas plus de 15 caractères".
- Identifiant agent:** Input field containing "nafuedlzoxlxdoideldzdsdzs". Constraint: "Pas plus de 21 caractères".
- Date d'appel, de * :** Input field for the start date.
- à * :** Input field for the end date.
- Calendar icons are present next to the date fields.
- Buttons: "Validez" and "Reset".

Voici les conditions de validation des champs pour l'utilisateur, cette vérification se fait par l'intermédiaire du *jquery validate* ([saisie.js](#)).

Formulaire de saisie administrateur:

The screenshot shows an administrator registration form with the following fields and constraints:

- Identifiant Agent:** Input field containing "nzhduznduznsuznsuzaaassdz". Constraint: "Pas plus de 21 caractères".
- Login:** Input field.
- Id enregistrements:** Input field.
- Date de consultation, de * :** Input field for the start date.
- à * :** Input field for the end date.
- Calendar icons are present next to the date fields.
- Buttons: "Validez" and "Reset".

Même chose côté administrateur (jquery validate).

Schéma API WEB :

[Schéma Application Web Opcall.docx](#)

Pour les utilisateurs lors qu'ils voudront écouter les fichiers MP3, ils seront obligés d'indiquer leur raison.

5-Annexes

5.1 Référentiel Git Projet :

Prérequis : Installer git_ <https://git-for-windows.github.io/>

Ensuite aller dans un dépôt pour cloner le projet dans le référentiel git opcall via à la commande « git clone <http://infradsi.opcalim.org/Git/OPCall.git> »

Après le git clone, faire un « git commit » puis « git push »

Mettre à jour le référentiel git via au serveur :

Name	Last Modified
2016 - Configuration	08/06/2017 11:57:31
AD-Modification-Champs	31/07/2017 14:40:35
MetaAnnuaire - 001 - Setup Instance	03/03/2017 14:24:32
MetaAnnuaire - 999 - Remove Instance	03/03/2017 09:16:51
Sites - Eloquant - 001 - Creation	05/04/2017 11:06:56
Sites - OPCall	26/07/2017 15:49:18
Sites - OPCALL - Recuperation Eloquant	26/07/2017 16:20:34
Sites - Portail RH et Annuaire	26/07/2017 14:30:21
Taches - EDI POLF	18/05/2017 08:49:32



Create Job

✕

Plan name:

Windows::Sites - OPCall

Job name:

automatic

Server targets:

Target servers directly

Server filter:

none

Start time:

31/07/2017

at

14:45

Simulation:

☐ Simulation only; do not modify any servers

Variables: ?

Populate from Template

Create Job

Close

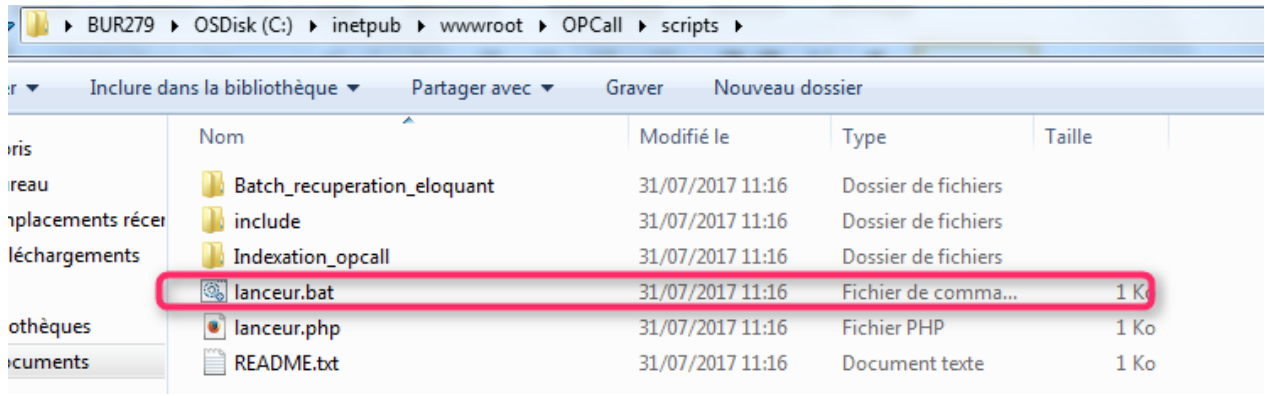


Jobs			
Count:		Server:	Status:
25		any server	any state
<div>Upcoming</div> <div>Recurring</div> <div>History</div>			
Date	Duration	Job Name	Status
31/07/2017 14:42	1,4s	Windows::AD-Modification-Champs	✓ Normal
31/07/2017 14:34	0,86s	Windows::AD-Modification-Champs	! Error
31/07/2017 14:33	2,4s	Windows::AD-Modification-Champs	✓ Normal
31/07/2017 14:31	0,59s	Windows::AD-Modification-Champs	! Error
31/07/2017 14:28	5s	Windows::AD-Modification-Champs	! Error
31/07/2017 14:13	0,016s	Windows::AD-Modification-Champs	! Error
31/07/2017 11:58	3,1s	Windows::Sites - OPCall	✓ Normal
31/07/2017 06:07	3,8s	Default::OPUS - Restart Tomcat	✓ Normal
30/07/2017 17:15	3s	Default::AD - Contrôle Champs Adresses	✓ Normal
30/07/2017 06:06	4,5s	Default::OPUS - Restart Tomcat	✓ Normal
29/07/2017 17:15	3,5s	Default::AD - Contrôle Champs Adresses	✓ Normal

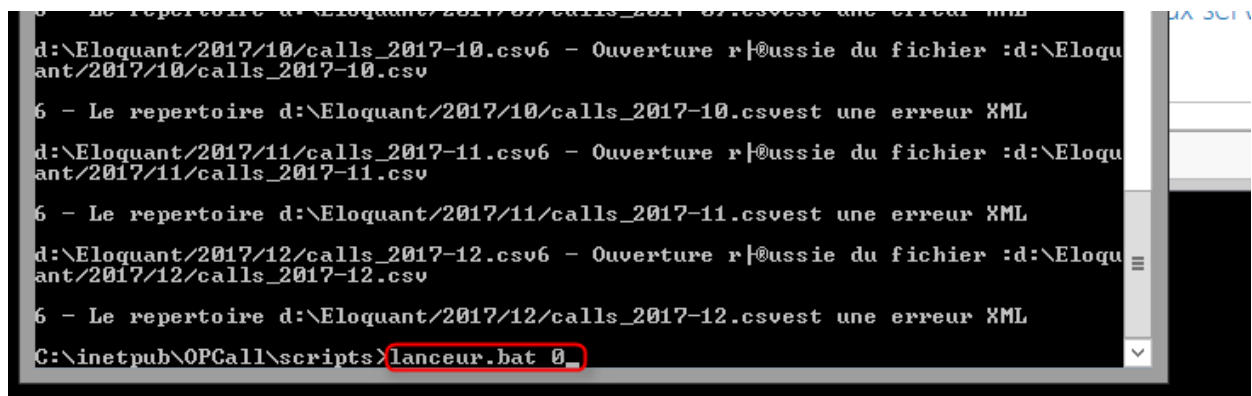
Le référentiel git côté serveur est maintenant à jour.

5.2 Lancements des batchs :

Lancement de batch se fait à partir d'un fichier lanceur.bat qui regroupe toute la commande pour lancer le fichier lanceur.php.



Ensuite pour lancer le fichier lanceur.bat, il faut le faire côté serveur (vm-035) en invite de commande.



Le dossier opcall se trouve « C:\inetpub\OPCall\ », l'encadrer est la commande à exécuter en mettant le paramètre 0 pour confirmer les lancements des batchs.

Le dépôt des fichiers mp3 et CSV côté serveur se trouve :

« d:\Eloquent » et pour regarder en base les indexations des données utiliser mysqlworkbench en entrant les paramètres pour se connecter à la base (voir page 6).

5.3 Pour la suite :

Mise en place de la suppression des fichiers agés de plus 2ans :

Une table enregistrements , de rajouter une colonne statut du fichier (si il est supprimer etc ..). Puisque il y aura prochainement un batch lancé quotidiennement de suppression sur les fichiers d'enregistrements de plus de 2 ans.

Mettre à jour de la connexion ftp en sftp:

Puisque Eloquant a prévu de mettre son serveur FTP en sftp à la demande d'Opcalim, il y aura simplement à changer le code dans [connexionftp.php](#) et de passer les fonctions ftp en SFTP dans le code PHP.