



Installation du générateur de bordereaux

v 1.0.1



Table des matières

1	Préambule	1
2	Prérequis	2
3	Installation	3
3.1	Module de génération de bordereaux	3
3.1.1.	Installation d'un JRE Java (s'il n'est pas installé)	3
3.1.2.	Installation et préparation de PostgreSQL 9.1	3
3.1.3.	Installation et paramétrage du module de génération	3
3.1.4.	Utilisation du module de génération	4

1 Préambule

Ce document présente les prérequis et l'installation du générateur de bordereaux.

2 Prérequis

Prérequis logiciels :

- VM Ubuntu server 12.04 LTS (64 bits)
 - o connexion internet,
 - o l'installation du générateur nécessite les droits administrateur de la machine, pour installer la base de données.
- JRE Java 1.7 min.
- Editeur de texte vim.
- Logiciel de décompression unzip (apt-get install unzip).
- (PostgreSQL 9.1 : installation dans cette procédure).

Prérequis fonctionnel :

- Conformément au modèle de conception défini, ce générateur de bordereaux ne traite que la génération des bordereaux un par un.

Fichiers nécessaires à l'installation du générateur de bordereaux :

- JavaSedaProfileGenerator-*version*.zip contenant :
 - o lib/* : fichiers jar utilisés par le générateur.
 - o sql/1structure/* : fichiers sql utiles pour la préparation de la base de données.
 - o sql/2data/* : fichier d'exemple d'insertion d'une entrée dans la table SAE.
 - o sedaGeneratorUnit.properties : fichier de propriétés.
 - o JavaSedaProfileGenerator_unit.sh (et .bat) : script de lancement du générateur pour une génération de bordereau de versement.
 - o JavaSedaProfileGenerator_config.sh (et .bat) : script pré-paramétré pour le dossier RACINE/, utilisant le fichier de configuration job.config.
 - o RACINE/* : dossiers pré-paramétrés pour JavaSedaProfileGenerator_config.sh (et.bat).

3 Installation

Ce chapitre présente l'installation du générateur de bordereaux, en utilisant une base de données pour incrémenter les identifiants de transferts. Il est aussi possible de générer un bordereau en lançant `JavaSedaProfileGenerator_config.sh` (ou.bat), qui s'appuie sur un fichier de configuration.

3.1 Module de génération de bordereaux

3.1.1. Installation d'un JRE Java (s'il n'est pas installé)

1. Récupérer la version JRE Java (par exemple `jre-8u51-linux-x64.tar.gz`) et le déposer par exemple dans `~/Downloads`.
2. `#: tar -xvf ~/Downloads/jre-8u51-linux-x64.tar.gz`
3. `#: mkdir -p /usr/lib/jvm/jre1.8.0_51`
4. `#: mv jre1.8.0_51/* /usr/lib/jvm/jre1.8.0_51/`
5. `#: update-alternatives --install /usr/bin/java java /usr/lib/jvm/jre1.8.0_51/bin/java 0`
6. vérifier la bonne installation : `#: java -version` : Java™ SE Runtime Environment (build 1.8.0_51-b16)

3.1.2. Installation et préparation de PostgreSQL 9.1

Notations : Les trois valeurs suivantes sont celles utilisées pour cette procédure d'installation, elles peuvent être adaptées au besoin :

- DATABASE : sedagen
- USER : sedagen
- PASSWORD : sedagen

1. `# apt-get update`
2. `# sudo adduser postgres`
3. `# aptitude install postgresql-9.1 postgresql-client-9.1 postgresql-contrib-9.1 postgresql-doc-9.1 postgresql-plperl-9.1 postgresql-plpython-9.1 postgresql-pltcl-9.1`
4. `# vim /etc/postgresql/9.1/main/pg_hba.conf`
5. Ajout de la ligne suivante dans les catégories désignées.

`## Database administrative login by UNIX sockets`

`local sedagen sedagen md5`

6. `# /etc/init.d/postgresql restart`
7. `# su postgres`
8. `# psql`
9. Taper les commandes suivantes

`CREATE USER sedagen;`

`CREATE DATABASE sedagen OWNER sedagen ;`

`ALTER USER sedagen WITH ENCRYPTED PASSWORD 'sedagen';`

Quitter l'éditeur PSQL:

10. `#\q`
11. `# exit`

3.1.3. Installation et paramétrage du module de génération

1. Déposer `JavaSedaProfileGenerator-version-bin.zip` à l'endroit souhaité que nous notons `[app_root]`.
2. `# cd [app_root]`
3. `# unzip JavaSedaProfileGenerator-version-bin.zip`
4. `# cd JavaSedaProfileGenerator-version`
5. Paramétrage du fichier `sedaGeneratorUnit.properties` :
 - a. `DATABASE_URL = jdbc:postgresql://127.0.0.1:5432/sedagen`

- b. DATABASE_USER=sedagen
- c. DATABASE_PASSWD=sedagen

```
6. # psql sedagen -U sedagen
7. # \i ../JavaSedaProfileGenerator-
   version/sql/1structure/20150723_2sedaGenerator_createTable_SAE.sql
8. #\q
```

3.1.4. Utilisation du module de génération

La génération d'un bordereau se fait par rapport au profil d'archivage RNG, relié à un couple (SAE_AccordVersement/SAE_Serveur) en base de données. Ce profil doit être accessible en lecture par l'utilisateur lançant le générateur. Les actions suivantes permettent d'insérer en base de données la relation entre (SAE_AccordVersement/SAE_Serveur) et le chemin vers le profil RNG à utiliser lorsque le générateur est lancé avec (SAE_AccordVersement/SAE_Serveur).

1. Modifier l'exemple JavaSedaProfileGenerator-version/sql/2data/20151221_insertExample.sql afin d'insérer une ligne dans la table SAE.
 - a. [SAE_AccordVersement] : le nom de l'accord de versement (exemple : « MP »).
 - b. [SAE_Serveur] : l'URI du SAE de destination. (exemple : « org.cgi.asalae »).
 - c. [TransferIdPrefix] : préfixe des identifiants de transferts incrémentés automatiquement (exemple : « MP_ »)
 - d. [SAE_ProfilArchivage] : lien vers le profil RNG à utiliser lorsque le générateur de bordereau sera lancé avec le couple (SAE_AccordVersement, SAE_Serveur) (absolu ou relatif par rapport au jar JavaSedaProfileGenerator-version.jar).
 - e. [TransferringAgencyId] : Identifiant du service versant.
 - f. [TransferringAgencyName] : Nom du service versant.
 - g. [TransferringAgencyDesc] : Description du service versant.
 - h. [ArchivalAgencyId] : Identifiant du service d'archives.
 - i. [ArchivalAgencyName] : Nom du service d'archives.
 - j. [ArchivalAgencyDesc] : Description du service d'archives.

Remarque : les champs c, e, f, g, h, i, j se retrouveront dans les bordereaux dont la génération sera lancée avec SAE_AccordVersement et SAE_Serveur.

```
2. # psql sedagen -U sedagen
3. # select count(*) from sae; (Résultat1).
4. # \i ../JavaSedaProfileGenerator-version/sql/2data/20151221_insertExample.sql
5. #SELECT count(*) FROM SAE; (Le résultat doit être Résultat1 + 1).
6. #\q
```

7. Se placer dans le dossier JavaSedaProfileGenerator-version

```
8. # chmod 755 JavaSedaProfileGenerator_unit.sh (si nécessaire)
```

9. Lancer JavaSedaProfileGenerator_unit.sh, celui-ci permettant d'appeler la classe unit.SedaGeneratorUnit en lui passant en paramètres :

- a. **PROPERTIES_LOCATION** : chemin vers un fichier renseignant "DATABASE_URL=" "DATABASE_USER=" et "DATABASE_PASSWD="

exemple : sedaGeneratorUnit.properties.

- b. **LOGBACK_DIR** : chemin vers le dossier contenant logback.xml.

exemple : logback/

- c. **URI** (Colonne SAE_Serveur de la table SAE): première partie de la clé (SAE_AccordVersement, SAE_Serveur) permettant de récupérer le chemin vers le profil en base de données.

exemple : org.cgi.asalae

- d. **AGREEMENT** (Colonne SAE_AccordVersement de la table SAE) : deuxième partie de la clé (SAE_AccordVersement, SAE_Serveur) permettant de récupérer le chemin vers le profil en base de données.

exemple : MP

- e. **ARCHIVE_FOLDER** : chemin vers le dossier racine de l'archive (non compressé).

exemple : « [app_root]/FOLDERS/Marché public »

- f. **DATA_FILE** : chemin vers le fichier de données métiers à utiliser pour la génération du bordereau.

exemple : « [app_root]/DATA/Marché public.txt »

- g. **OUT_SUMMARY** : chemin vers le bordereau de sortie.

exemple : « [app_root]/OUT/bordereau_MP.xml »

- h. **OUT_SUMMARY_ERROR** : chemin vers le fichier d'erreur à créer, plutôt que le bordereau, si la génération de ce bordereau est en erreur.

exemple : « [app_root]/ERRORS/bordereau_MP.xml.err »

Pour le lancement de la génération d'un bordereau avec ces exemples, voici la commande :

```
10. # ./JavaSedaProfileGenerator_unit.sh sedaGeneratorUnit.properties logback/ org.cgi.asalae MP
    "[app_root]/FOLDERS/Marché public" "[app_root]/DATA/Marché public.txt"
    [app_root]/OUT/bordereau_MP.xml [app_root]/ERRORS/bordereau_MP.xml.err
```

Explication du fonctionnement du générateur :

Le générateur de bordereaux se connecte à la base de données grâce aux informations récupérées dans le fichier **PROPERTIES_LOCATION**. Les propriétés des logs sont dans le fichier **LOGBACK_DIR/logback.xml**. Le générateur récupère en base de données le chemin vers le profil RNG à utiliser pour la clé primaire (SAE_Serveur = **URI**, SAE_AccordVersement = **AGREEMENT**). A partir du profil, du dossier **ARCHIVE_FOLDER** et du fichier de données métiers **DATA_FILE**, le générateur créera le bordereau **OUT_SUMMARY**, ou le fichier d'erreur **OUT_SUMMARY_ERROR** s'il y a eu des erreurs. Chaque jour, un fichier logging-*jour*.log est généré avec l'ensemble des logs du générateur.