Mode opératoire pour les scripts de facturation Passerelles

Table des matières

[2 Historique des versions 2](#_Toc77085715)

[3 Répertoire d’archive du script 3](#_Toc77085716)

[4 Fichiers d’entrée 4](#_Toc77085717)

[4.1 Liste des clients à ne pas facturer. 4](#_Toc77085718)

[4.2 Liste des SIMs de prêt. 4](#_Toc77085719)

[4.3 Fichier de correspondance des noms de clients sur PassPort et sur PayView 4](#_Toc77085720)

[4.4 Fichiers suffixés 4](#_Toc77085721)

[5 Description des étapes du script 5](#_Toc77085722)

[6 Configuration du script 6](#_Toc77085723)

[6.1 Mettre à jour les mots de passe 6](#_Toc77085724)

[6.1.1 PayView - FactuPayview.py (line 18) 6](#_Toc77085725)

[6.1.2 PassPort - mshAPI.py (line 70) 6](#_Toc77085726)

[6.2 Mettre à jour les chemins des fichiers d'entrée - FactuPayview.py (line 18) 6](#_Toc77085727)

[6.3 Mettre à jour les chemins des fichiers de sortie - FactuPayview.py (line 18) 6](#_Toc77085728)

[6.4 Mettre à jour le mois d'exécution du script - FactuPayview.py (line 537) 6](#_Toc77085729)

[6.5 Mettre à jour les fichiers des données à traiter - FactuPayview.py (line 547) 7](#_Toc77085730)

[6.6 Mettre à jour la liste des clients, fichier correspPayViewPassPort.xlsx 7](#_Toc77085731)

[6.7 Lancer le script 7](#_Toc77085732)

[7 Fichier “mois\_AAAA\_facturationGlobale.xlsx” 8](#_Toc77085733)

[7.1 Mise en forme des fichiers avant envoi sur chaque onglet 8](#_Toc77085734)

[7.2 Corrections des fichiers avant envoi sur le 1er onglet 'Facturation' 8](#_Toc77085735)

[8 Envoi des fichiers 9](#_Toc77085736)

[8.1 ADV 9](#_Toc77085737)

[8.1.1 Destinataire 9](#_Toc77085738)

[8.1.2 Objet 9](#_Toc77085739)

[8.1.3 Corps 9](#_Toc77085740)

[8.2 AVT 9](#_Toc77085741)

[8.2.1 Destinataire 9](#_Toc77085742)

[8.2.2 Objet 9](#_Toc77085743)

[8.2.3 Corps 9](#_Toc77085744)

[8.3 HM 10](#_Toc77085745)

[8.3.1 Destinataire 10](#_Toc77085746)

[8.3.2 Objet 10](#_Toc77085747)

[8.3.3 Corps 10](#_Toc77085748)

[9 Sauvegarde des fichiers de facturation 11](#_Toc77085749)

# Historique des versions

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Auteur** | **Changements** |
| 1 | 23/06/2021 | Laure BROEGG | Première version |
| 2 | 13/07/2021 | Laure BROEGG | Modification suite à l’exécution du 28/06/21, le script s’exécute sur le dernier mois révolu  Ajout du paragraphe 5 décrivant les étapes du script |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Répertoire d’archive du script

Les fichiers actuels sont archivés dans un outil de gestion de conf :

<https://github.com/stephane2K/payview-billing.git>

# Fichiers d’entrée

Le script a besoin de plusieurs fichiers en entrée :

## Liste des clients à ne pas facturer.

Car ils ne sont pas dans notre scope ou parce qu’on ne les facture pas:

* Une majorité de clients sur PassPort
* INGENICO TEST sur PayView
* PROGECARTE sur PassPort et PayView

Ce fichier n’a pas été modifié depuis le début de PayView.

Ce fichier est mis à jour par Laure.

Ce fichier est sur le réseau : France - Marketing And Communication - 14. Projets\Payview\11.Scripts\Facturation\01.Inputs\clients\_non\_facturés.txt

<https://ingenicogroup.sharepoint.com/:t:/r/sites/France-MarketingAndCommunication/Shared%20Documents/14.%20Projets/Payview/11.Scripts/Facturation/01.Inputs/clients_non_factur%C3%A9s.txt?csf=1&web=1&e=Qhiwvd>

## Liste des SIMs de prêt.

Ces SIMs présentes sur PassPort et PayView ne sont pas à facturer. Le script de facturation ne regarde pas la date, uniquement l’iccid. Evolution à prévoir : facturer quand la date d’expiration est passée

Ce fichier est mis à jour par Maeva, Abdoulaye et Alyaa, en fonction des cartes SIMs prêtées aux clients et des fin de période de prêt.

Ce fichier est sur le réseau : France - Marketing And Communication - 14. Projets\Passerelles\SIM\SIM DE PRET\SIMs\_de\_pret.xlsx

<https://ingenicogroup.sharepoint.com/:x:/r/sites/France-MarketingAndCommunication/Shared%20Documents/14.%20Projets/Passerelles/SIM/SIM%20DE%20PRET/SIMs_de_pret.xlsx?d=w222cd9036ad24789ab5f83cd31ea5dda&csf=1&web=1&e=6LrScD>

## Fichier de correspondance des noms de clients sur PassPort et sur PayView

Ce fichier permet

* d’associer un nom de client PassPort et un nom de client PayView pour regrouper le nombre de TPEs sur PassPort et PayView et le nombre de SIMs par forfait sur PassPort et PayView d’un meme client,
* d’indiquer la BU d’appartenance du client,
* de fournir le CODE CLIENT SAP et le CONTRAT SAP,
* d’indiquer si le client est exonéré de la surconsommation des cartes SIMs.

Ce fichier est mis à jour à chaque nouvelle création de client ou sous client sur PayView.

Ce fichier est mis à jour par Laure.

Ce fichier est sur le réseau : France - Marketing And Communication - 14. Projets\Payview\11.Scripts\Facturation\01.Inputs\correspPayViewPassPort.xlsx

<https://ingenicogroup.sharepoint.com/:x:/r/sites/France-MarketingAndCommunication/Shared%20Documents/14.%20Projets/Payview/11.Scripts/Facturation/01.Inputs/correspPayViewPassPort.xlsx?d=w874e6a9d4ac64b239bae913c2a79d8ab&csf=1&web=1&e=42bWdb>

## Fichiers suffixés

Les fichiers suffixés “\_PassPort” servent a récuperer l’integralité des données de PassPort, pour que le résultat soit exact, il ne faut pas récuperer de données PayView, cette partie du code est à commenter.

Les fichiers suffixés “\_PayView” servent a récuperer l’integralité des données de PayView et comparer les données avec Preludd pour leur facturation envers Ingenico, pour que le resultat soit exact, il ne faut pas récupérer de données de PassPort, cette partie du code est à commenter.

# Description des étapes du script

1. Récupération la liste des organisations sur PayView

(if PayViewClientToFactData)

1. Vérification de la présence des organisations dans le fichier d’entrée “Fichier de correspondance des noms de clients sur PassPort et sur PayView”

(for l in readCsvOrExcel(FICHIER\_CORRESPONDANCE\_NOMS\_CLIENTS))

-> Si une organisation existe sur PayView mais pas dans le fichier, le script est interrompu et demande si on souhaite traiter ce client.

Attention, le script ne met pas à jour le fichier de correspondance.

1. Lecture du fichier de correspondance des noms de clients sur PassPort et sur PayView

(readCorrespondanceNomClients())

-> Enregistrement en mémoire des noms PassPort et des noms PayView pour faire correspondre les données

1. Récupération des données de PassPort

(ap.RécupèreFichiersFactu)

-> Création d’un répertoire DataPassPort contenant des fichiers CSV (1 par client pour ses TPE, 1 par client pour ses SIM)

1. Récupération des données de PayView

(getFactuFilesPayView)

-> Création d’un répertoire DataPayView contenant des fichiers CSV (1 par client pour ses TPE, 1 par client pour ses SIM)

1. Génération complète de la facturation

(génèreFacturationComplete)

-> Les clients sont pris les uns après les autres.

1. Un test vérifie que le client est accepté (s’il ne fait pas partie de la liste des clients à ne pas facturer)
2. Le fichier SIM de prêt est parcouru pour mettre un tag sur les SIMs de prêt trouvées dans les fichiers SIM sur PassPort et PayView
3. Grace au fichier de correspondance des noms de client, les données TPE et SIM sur PassPort et sur PayView sont mises en commun.
4. Le fichier xls de facturation est rempli et s’affiche

# Configuration du script

## Mettre à jour les mots de passe

### PayView - FactuPayview.py (line 18)

*##################### EMAIL MOT DE PASSE PAYVIEW ADMIN ###########################*

*PAYVIEW\_ADMIN\_EMAIL= #remplacer par l'email admin PayView de Laure*

*PAYVIEW\_ADMIN\_MDP= #Remplacer par le mot de passe admin PayView de Laure*

Évolué : parametres.ini dans Inputs/ contient ces données.

### PassPort - mshAPI.py (line 70)

*def loginAsIngenico(self):*

*#Remplacer par login/mdp Laure PassPort*

*self.login(os.environ['PASSPORT\_ADMIN\_LOGIN'],os.environ['PASSPORT\_ADMIN\_MDP'])*

Évolué : parametres.ini dans Inputs/ contient ces données.

## Mettre à jour les chemins des fichiers d'entrée - FactuPayview.py (line 18)

*##################### CHEMINS DES FICHIERS D'ENTREES ##############################*

*CLIENTS\_IGNORES = r'C:\Users\squack\Documents\\_PAYVIEW\FACTURATION\Inputs\clients\_non\_facturés.txt'*

*FICHIER\_SIM\_PRET = r"C:\Users\squack\Documents\\_PAYVIEW\FACTURATION\Inputs\SIMs\_de\_pret.xlsx"*

*FICHIER\_CORRESPONDANCE\_NOMS\_CLIENTS = r'C:\Users\squack\Documents\\_PAYVIEW\FACTURATION\Inputs\correspPayViewPassPort.xlsx'*

Évolué : parametres.ini dans Inputs/ contient ces données.

EVOLUTION : pointer vers les fichiers sur le réseau, cette action sera à faire 1 seule fois

## Mettre à jour les chemins des fichiers de sortie - FactuPayview.py (line 18)

*##################### CHEMINS DES FICHIERS GENERES ##############################*

*DOSSIER\_GENERATION\_RESULTATS = r'C:\Users\squack\Documents\\_PAYVIEW\FACTURATION\Generated'*

Évolué : parametres.ini dans Inputs/ contient ces données.

EVOLUTION : sauvegarder les fichiers d’entrée dans un répertoire « Inputs »

## Mettre à jour le mois d'exécution du script - FactuPayview.py (line 537)

*#Début du script*

*if \_\_name\_\_ == "\_\_main\_\_":*

*locale.setlocale(locale.LC\_ALL, '') #set local en Français*

*setLogger()*

*mois =12*

*année = 2020*

~~EVOLUTION : remplir cette variable depuis un paramètre, passer le mois et l’année en parametre du script par exemple sous la forme MMAAAA, ou s’il y a une interface graphique via un menu déroulant~~

Modification du 28/06/2021 : Le script s’exécute sur le dernier mois révolu, sans avoir besoin de renseigner le mois en paramètre ou dans le code.

EVOLUTION ajoutée le 13/07/2021: Ces variables restent néanmoins disponibles pour un lancement manuel

## Mettre à jour les fichiers des données à traiter - FactuPayview.py (line 547)

*# Remplacer les valeurs par None sur les 2 lignes pour récupérer les données de PassPort et PayView*

*PASSPORT\_ZIP\_DIRECTORY = r'C:\Users\...\DataPassPort' ou None*

*PAYVIEW\_DATA\_DIRECTORY = r'C:\Users\...\DataPayView' ou None*

Ces variables indiquent ou aller chercher les données de PassPort et PayView.

* Si c’est None, le script recupere les données sur les serveurs respectifs via les API
* Si c’est un chemin, le script récupere les données dans ces repertoires. Cela veut dire que les données ont déjà été récupérées précédemment.

NOTE: Je pense que par défaut, on va tout le temps laisser à None, ca prend du temps mais c’est plus simple

EVOLUTION ajoutée le 13/07/2021: Ces variables restent néanmoins disponibles pour un lancement manuel

## Mettre à jour la liste des clients, fichier correspPayViewPassPort.xlsx

## Lancer le script

Ouvrir une invite de commande (executer / cmd)

se placer dans le répertoire de travail (cd C:/ …..

taper FactuPayview.py

~~EVOLUTION : taper FactuPayview.py 012021~~

Modification du 28/06/2021 : Le script s’exécute sur le dernier mois révolu, sans avoir besoin de renseigner le mois en paramètre ou dans le code.

Lors de l’exécution, le script s’arrête pour demander si un client existant sur PayView qui n’apparait pas dans le fichier correspPayViewPassPort.xlsx doivent être ajoutés

EVOLUTION : arrêter le script et indiquer qu’il faut demander à Laure

EVOLUTION : remplir dynamiquement le fichier correspPayViewPassPort.xlsx

# Fichier “mois\_AAAA\_facturationGlobale.xlsx”

## Mise en forme des fichiers avant envoi sur chaque onglet

* Élargir toutes les colonnes
* Faire des blocs aux bordures extérieures épaisses autour des données des sous clients des grossistes
* Supprimer la ligne REV Test d’AVT

## Corrections des fichiers avant envoi sur le 1er onglet 'Facturation'

* Suppression des surconso de SIM pour tous les clients MS sauf CAP INFO
* Suppression des surconso de SIM pour les clients grossistes AVT et HM Telecom mais pas pour leurs sous-clients
* Il ne faut pas mettre à 0 la surconso des clients BPMED, CORSE MONETIQUE, ESPACE MONETIQUE et BRED (Bred n’est pas un grossiste, c’est une banque avec plusieurs mainteneurs)

EVOLUTION : ajouter un bout de code qui supprime les surconsommations des forfaits SIMs (données de la colonne PAS\_SIM\_OVERFEE) des clients qui sont éxonérés de surconsommation. La liste des clients se trouve dans le fichier correspPayViewPassPort.xlsx, colonne Exoneration de la surconso.

Si VRAI : le client est exonéré de surconso, remplacer la valeur par 0

Si FAUX ou vide : le client peut etre facturé de la surconso, ne pas modifier la valeur

# Envoi des fichiers

## ADV

L’ADV n’a pas besoin des répertoires DataPassPort et DataPayView

Faire un zip nommé AAAAMM\_factuPayView.zip contenant

* le fichier mois\_AAAA\_facturationGlobale.xlsx
* le répertoire Détails

### Destinataire

Les données de facturation sont à envoyer

A [adv.services@ingenico.com](mailto:adv.services@ingenico.com)

CC Jerome, Sylvain, Laure, Maeva, Alyaa

### Objet

PAYVIEW - Données de facturation du mois de MOIS AAAA

### Corps

*Bonjour Marie-Rose et Sarah,*

*voici les données de facturation de PayView pour MOIS AAAA.*

*Les données grossistes ont déjà été fournies à AVT et HM Telecom.*

*Je reste à votre disposition pour tout complément d’information.*

*Cordialement,*

*Laure*

*EVOLUTION : ajouter l’info comme quoi ce mail est automatique*

## AVT

Le fichier zip mois\_AAAAgrossiste\_AVT Grossiste.zip du répertoire « Détails » est à envoyer à AVT (Sebastien Lico) tel quel, sans modifier les surconso de SIM du fichier mois\_AAAA\_BILAN\_grossiste\_AVT Grossiste.xlsx

### Destinataire

Les données de facturation sont à envoyer

A [sebastien@avtsesam.com](mailto:sebastien@avtsesam.com)

CC Jerome, Sylvain, Laure

### Objet

PAYVIEW / AVT - Données de facturation du mois de MOIS AAAA

### Corps

*Monsieur Lico, bonjour,*

*Je vous transmets les données de facturation de PayView pour mois AAAA.*

*Je reste à votre disposition pour tout complément d’information.*

*Cordialement,*

*Laure BROEGG*

*EVOLUTION : ajouter l’info comme quoi ce mail est automatique+ une adresse à qui ecrire en cas de questions*

## HM

Le fichier zip mois\_AAAAgrossiste\_HM TELECOM Grossiste.zip est à envoyer à HM Telecom (Léa Zeitoun) tel quel, sans modifier les surconso de SIM du fichier mois\_AAAA\_BILAN\_grossiste\_HM TELECOM Grossiste.xlsx

### Destinataire

Les données de facturation sont à envoyer

A [lea@hmtelecom.com](mailto:lea@hmtelecom.com)

CC Jerome, Sylvain, Laure

### Objet

PAYVIEW / HM Telecom - Données de facturation du mois de MOIS AAAA

### Corps

*Bonjour Léa,*

*Voici les données de facturation de PayView pour mois AAAA.*

*Je reste à votre disposition pour tout complément d’information.*

*Cordialement,*

*Laure BROEGG*

*EVOLUTION : ajouter l’info comme quoi ce mail est automatique+ une adresse à qui ecrire en cas de questions*

# Sauvegarde des fichiers de facturation

Le répertoire de facturation est à sauvegarder sur le réseau

EVOLUTION : sauvegarder directement le résultat quelquepart ou créer un bouton ou script pour suavegarder un repertoire, car parfois, il faut lancer plusieurs fois la script. Il ne faudrait qu’une sauvegarder par mois

Déplacer le zip dans le répertoire de facturation (factuPayView\_janvier\_2021\_pourADV.zip dans 20210202\_17h34\_facturation\_janvier\_2021)

Copier le tout sur le réseau :

* D’octobre 2020 à Décembre 2020

France - Marketing And Communication - 14. Projets\Passeport\Billing\2020

* A partir de janvier 2021

France - Marketing And Communication - 14. Projets\Payview\32.Facturation\_Billing\AAAA

<https://ingenicogroup.sharepoint.com/:f:/r/sites/France-MarketingAndCommunication/Shared%20Documents/14.%20Projets/Payview/32.Facturation_Billing/2021?csf=1&web=1&e=qNnX9u>