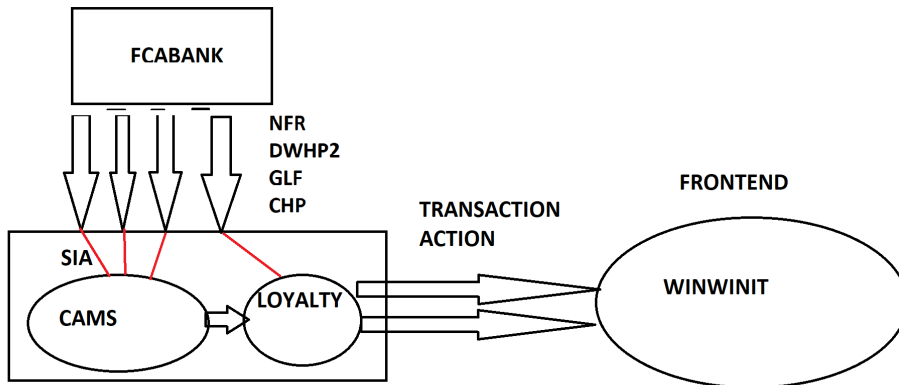


Funzionamento Loyalty FcaBank



FCABank invia a SIA 4 flussi consistenti di file posizionali e csv di cui 2 di anagrafica e 2 di transazioni.

NFR, DWHP2 e GLF sono posizionali e passano prima dal CAMS, invece **CHP** è un csv e arriva direttamente al server della Loyalty

SIA si occupa della storicizzazione e del processamento di questi flussi con obiettivo di scrivere dei file di output per WinWinIT.

Il codice della Loyalty di FcaBank è sulla 10.92.1.180 (TEST) e sulla 10.92.1.80 (PRODUZIONE)

L'applicazione, sviluppata in Python, si trova qui, sia test che prod:

/home/hdfs/lbd/loyaltyFcaBank/app_loyalty

Qui su sharepoint, oltre a documenti funzionali utili, trovi uno zip Loyalty_FCABank del 25/8 con il codice più aggiornato: <https://offices.sia.local/sites/U00125/Shared%20Documents/Digital%20Services/FCABank>

Sui server sopra citati è stata creata un'alberatura di cartelle su cui il sistema si aspetta i file di input, scrive i file di output e sposta i file arrivati in input in una cartella di file già elaborati.

CAMS invia i suddetti file nelle rispettive cartelle sul server loyalty 10.92.1.80:

/ftp/loyalty_fca/ftp/cams/nfr/in --giornaliero

/ftp/loyalty_fca/ftp/cams/dwhp2/in --giornaliero

/ftp/loyalty_fca/ftp/cams/glf/in --di solito con cadenza mensile il primo del mese

FCAB invia il CHP_transaction_yyyymmddhhmmss giornalmente in questa cartella:

/ftp/loyalty_fca/ftp/fcab/transaction/in

L'elaborazioni varie vengono effettuate su strutture chiamate DataFrame.

L'applicazione è collegata ad Hadoop, quindi ha accesso ad Hdfs e il database loyalty_fca di Hive.

Su Hive sono presenti le tabelle contenenti lo storico dei flussi, oltre che tabelle temporanee usate per le elaborazioni.

Processi caricamento file

I 4 processi di caricamento file partono lanciando rispettivamente gli script:

- ProcessNFR.sh **ProcessNFR.py**
- ProcessDWHP2.sh **ProcessDWHP2.py**
- ProcessGLF.sh **ProcessGLF.py**
- ProcessFCABTransaction.sh **ProcessFCABTransaction.py**

I process richiamano il Process.py che gestisce il lancio dei 3 processi di Read, Elaboration e Refresh, occupandosi di inviare le email in caso di OK o KO.

ReadNFR.py si occupa della lettura del file posizionale proveniente dal CAMS e scrive le tabelle loyalty_fca.e_nfr_cams (in overwrite) e loyalty_fca.p_nfr_cams (in append)

ElaborationNFR.py legge loyalty_fca.e_nfr_cams per scrivere in append loyalty_fca.l_anag e loyalty_fca.l_balance. Usa come tabelle intermedie i_nfr_cams e i_nfr_cams_scarti.

RefreshNFR.py svuota la tabella loyalty_fca.e_nfr_cams e sposta il file NFR dalla cartella ../in/ alla cartella ../worked/

ReadGLF.py si occupa della lettura del file posizionale proveniente dal CAMS e scrive le tabelle loyalty_fca.e_glf_cams (in overwrite) e loyalty_fca.p_glf_cams (in append)

ElaborationGLF.py legge loyalty_fca.e_glf_cams per scrivere in append loyalty_fca.l_transaction e aggiornare loyalty_fca.l_balance. Usa come tabelle intermedie i_glf_cams e i_glf_cams_scarti

RefreshGLF.py svuota la tabella loyalty_fca.e_glf_cams e sposta il file GLF dalla cartella ../in/ alla cartella ../worked/

ReadDWHP2.py si occupa della lettura del file posizionale proveniente dal CAMS e scrive le tabelle loyalty_fca.e_dwHP2_cams (in overwrite) e loyalty_fca.p_dwHP2_cams (in append)

ElaborationDWHP2.py legge loyalty_fca.e_dwHP2_cams per aggiornare tabelle loyalty_fca.l_anag e loyalty_fca.l_balance. Usa come tabelle intermedie i_dwHP2_cams e i_dwHP2_cams_scarti

RefreshDWHP2.py svuota la tabella loyalty_fca.e_dwHP2_cams e sposta il file DWHP2 dalla cartella ../in/ alla cartella ../worked/

ReadFCAB.py si occupa della lettura del file csv (con nome CHP_transaction_yyyymmddhhssmm) proveniente da FCABANK direttamente e scrive le tabelle loyalty_fca.e_transaction_fcab (in overwrite) e loyalty_fca.p_transaction_fcab (in append)

ElaborationFCAB.py legge loyalty_fca.e_transaction_fcab per aggiornare tabelle loyalty_fca.l_transaction e loyalty_fca.l_balance. Usa come tabelle intermedie i_transaction_fcab e i_transaction_fcab_scarti

RefreshFCAB.py svuota la tabella loyalty_fca.e_transaction_fcab e sposta il file CHP di input dalla cartella ../in/ alla cartella ../worked/

ElaborationNFR.py scrive file di action per WinWinIT

ElaborationDWHP2.py scrive file di action per WinWinIT

ElaborationGLF.py scrive file di transaction per WinWinIT

ElaborationFCAB.py scrive file di transaction per WinWinIT

Alle 14 è schedulata l'elaborazione del file NFR (Anagrafica Utente) ;

alle 14.30 l'elaborazione del file DWHP2 (Cambio stato carta) ,

alle 16 quella del file GLF(Transazioni-VirtualCredit),

alle 17 quella del file CHP(Transazioni-VirtualCredit&Extrabonus)

Questo l'elenco preciso dei crontab schedulati (utente hdfs) in ambiente di Produzione:

```
0 3 * * * sh /home/hdfs/lbd/loyaltyFcaBank/app_loyalty/BackupDB.sh
```

```
0 14 * * * sh /home/hdfs/lbd/loyaltyFcaBank/app_loyalty/ProcessNFR.sh
```

```
30 14 * * * sh /home/hdfs/lbd/loyaltyFcaBank/app_loyalty/ProcessDWHP2.sh
```

```
0 16 * * * sh /home/hdfs/lbd/loyaltyFcaBank/app_loyalty/ProcessGLF.sh
```

```
0 17 * * * sh /home/hdfs/lbd/loyaltyFcaBank/app_loyalty/ProcessFCABTransaction.sh
```

Crontab come utente tomadm , questi script si occupano di inviare i file ACTION e TRANSACTION per WinWinIT:

```
30 16 * * * sh /ftp/loyalty_fca/SftpScript/sftpActionToWinwinit.sh &>> /ftp/loyalty_fca/SftpScript/sftp.log
```

```
00 18 * * * sh /ftp/loyalty_fca/SftpScript/sftpTransactionToWinwinit.sh &>> /ftp/loyalty_fca/SftpScript/sftp.log
```

Backup

Ogni notte viene effettuato il backup delle seguenti tabelle con lo script **BackupDB.sh**:

- l_anag
- l_balance
- l_transaction

Il backup consiste nel creare una copia delle tabelle con suffisso _yyyymmdd

C'è la possibilità di effettuare un restore delle tabelle con lo script **RestoreDB.py** e passando il parametro yyyymmdd della data dai cui restaurare le tabelle.

FLUSSO VIRTUAL CREDIT ED EXTRABONUS (FCAB>SIA>WINWINIT): file CHP_transaction_yyyymmddhhmmss

Il flusso consiste in un file .csv inviato al massimo una volta al giorno da FCABank su server già ora utilizzati, ma su un path nuovo.

L'elaborazione lato SIA si articola in 3 fasi.

Nella prima fase il file viene letto e le righe vengono memorizzate in una tabella temporanea chiamata e_transaction_fcab che verrà usata per le elaborazioni successive. Le nuove informazioni vengono salvate in una tabella p_transaction_fcab contenente lo storico dei file inviati da FCAB.
Questa fase viene svolta dallo script ReadFCAB.py.

Nella seconda fase di elaborazione vera e propria (svolta dallo script ElaborationFCAB.py), dopo aver filtrato le transazioni solo degli utenti attivi e presenti nella tabella anagrafica (l_anag), vengono eseguite tre operazioni principali:

-aggiornamento del saldo di Extrabonus e Virtual Credit del cliente (tabella l_balance), se presente tra le nuove transazioni

-scrittura del file di output per WinWinIT

-storicizzazione delle nuove transazioni nella tabella l_transaction del database loyalty_fca

Nella terza e ultima fase svolta dallo script RefreshFCAB.py viene eliminato il contenuto della tabella temporanea e_transaction_fcab e spostato il file arrivato in input in un'altra cartella contenente i file già elaborati.

Nel link seguente puoi trovare un esempio di Release Note per questo progetto relativamente alla gestione di un nuovo flusso da zero (CHP da FCAB) e la modifica di script per un flusso già esistente (DHWP2) per l'aggiunta di nuove informazioni : [RELEASE NOTE_DCH_Loyalty FCAB 02-04-2020](#)

Questo rilascio invece riguardava l'invio a FCAB di un file di risposta in seguito all'invio da parte loro del CHP_Transaction: [RELEASE NOTE_DCH_Loyalty FCAB FLUSSO RESPONSE 11-06-2020](#)