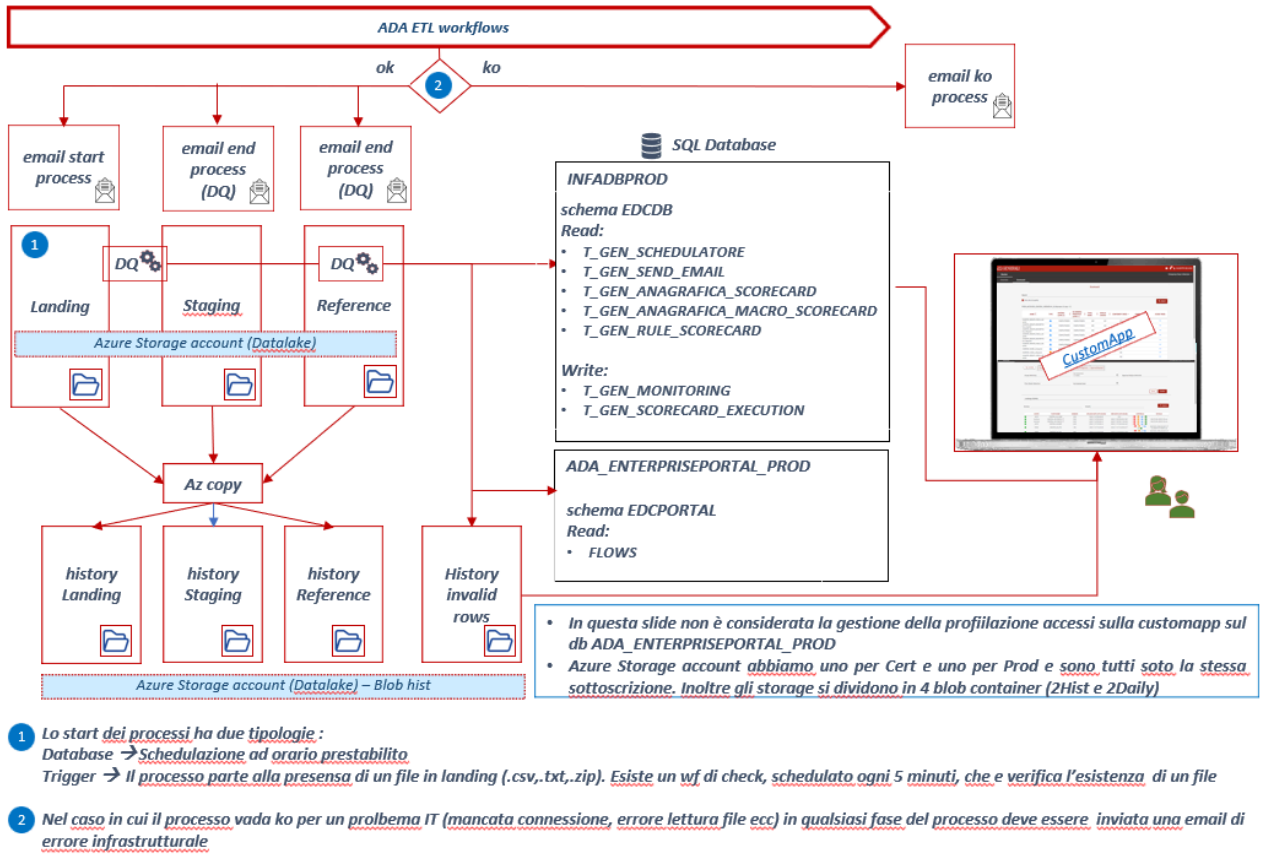
Framework DQ ADA - Informatica

Last updated by | NTT Data Stipo | 1 Mar 2024 at 23:47 CET

In questa sezione viene descritta la procedura ETL ADA sul prodotto Informatica.
Di seguito il flow di processo:



Processo ETL Informatica ADA

Il processo di Ada è progettato per gestire sia la lettura da

- [ ] File
- [ ] Database

e lo si può suddividere in queste macro fasi :

1. Landing to analisys
2. analysis to staging
3. stanging to reference

NOTA_1: Per l'ambito IFRS9 abbiamo più fasi che verranno descritte nel paragrafo dedicato.

1. Landing to analisys

In questa fase il processo recupera i dati da

- [ ] File
- [ ] Database

Case - File

Il processo di etl verifica presenza ***file*** nel path di landing , landing/project_name/flow_name/,

1. invia una email ***start processo***
2. ***Predisposizione ambiente*** (opzionale)
3. Applicazione ***controlli referenziali*** (opzionale)
4. Storicizzazione file input
5. Step din to per:
   spostamento del file dal path di landing al path landing/project_name/flow_name/analysis/ con eventuale arricchimento file di input con campi calcolati, o recuperati in lkp o in left join, necessari per i controlli di qualità

NOTA file: il naming FILE deve rispettare formato previsto dallo stream e ad oggi viene tramite goanyware, sftp teams e manuale, e solitamente contiene la versione vYYYYMMDDHHMMSS (esempio v29991231235959);
NOTA Start processo: per la metodologia vedere i dettagli vedere paragrafo **Schedulazione**;
Nota Email: per la metodologia vedere i dettagli vedere paragrafo **Email**;
Nota Controlli referenziali :Se previsto check bigger length (check su struttura file in input, es. Nome colonne, ordine colonne, lunghezza colonne);
Nota Predisposizione ambiente : Se l'input è di tipo zip o xlsx e deve subire trasformazioni prima di essere processato Copia da datalake landing a server di informatica , Unzip/conversione file, copia file da server informatica a datalake landing;

1. analysis to staging

in questa fase  vengono applicate le regole di qualità tramite:

- Scorecard
- mapping lnd to stg
- Storicizzazione file
- Storicizzazione invalid rows

Scorecard
L'esecuzione consente la visualizzazione tramite l'analyst che applica tutte le regole di qualità sul file presente in landing/project_name/flow_name/analysis/

mapping lnd to stg

in questa fase applicazione invece le regole di qualità vengono applicate in modo fisico su informatica, spostamento del file dal path di analysis ad uno temporaneo (es. technical/elaboration/staging/ project_name/flow_name/), convertendolo in file parquet e salvataggio esiti check qualità.

E' in questa fase che viene effettuato un check dei risultati dei controlli di qualità.

- BLOCKING
- ALERT WITH DISCARD
- ALERT WITHOUT DISCARD

e tramite la combinazione dell'applicativo di informatica e una shell (AWK) consetne di eventuali invalid rows per i record che falliscono i controlli di qualità.

(file csv divisi per nome regola fallita contenenti i relativi record di input che hanno fallito la specifica regola).

toricizzazione invalid rows zippando i csv prodotti e copiandoli sul datalake nel container di history al path h_invalidrows/project_name/flow_name/ZIP/YYYY/MM/DD/
Se NON ci sono controlli con esito KO di tipo BLOCKING
copia file da path technical al path di staging: staging/project_name/flow_name/
Storicizzazione file di staging nel container di history h_staging/project_name/flow_name/yyyy/mm/dd/

1. stanging to reference

Step reference Copia file da path staging a path reference reference/project_name/flow_name/ con eventuali trasformazioni/arricchimento dei dati
Storicizzazione file di reference nel container di history h_reference/project_name/flow_name/yyyy/mm/dd/
Rimozione file dal path di landing (questo step andrà eseguito in qualsiasi caso, per evitare che riparta lo stesso file)
Step mail finale
Dopo la scrittura sulle tabelle del monitor, viene mandata una mail di fine processo, che può indicare il termine dell'esecuzione in Success, la presenza di errori bloccanti o di warning con o senza scarti o il fallimento del processo.
Gestione errori generici/infrastrutturali
Se qualsiasi step fallisce per errori infrastrutturali o non gestiti viene eseguito lo step di t_gen_monitoring KO e viene mandata una mail di failed per errore generico

In questa sezione paragrafi verranno effettuati dei focus su seguenti processi

- **Schedulazione**
- **Storicizzazione**
- **Configurazione ambiente**
- **Scrittura log**
- **Gestione Email**

Schedulazione

Ada usa come strumento di schedulazione il l'applicativo fornito da **Informatica DEI**, gestisce un approccio di schedulazione leggermente diverso in base alla fonte dei dati, in particolare abbiamo due approcci:

- **Schedulazione Combinata Timing e Trigger**: E' una pianificazione specifica basata sia sul tempo (timing) che su eventi specifici (trigger) quando si tratta di leggere dati da file. Ad esempio, abbiamo un wf di che è configurato per verificare l'esistenza di file (.csv, .zip, .txt) ogni giorno ogni 5 minuti (timing) e scatena l'esecuzione del processo al verificarsi dell'evento (trigger) presenza file (vedi figura*)

- **Schedulazione di Tipo Timing**: E' una pianificazione basata esclusivamente sul tempo (timing). Potrebbe essere configurato per recuperare dati dal database ad intervalli regolari, come ogni ora o ogni giorno.

In sostanza, Ada usa la schedulazione combinata timing e trigger per lettura da file visto che l'esecuzione deve essere attivata da eventi specifici, oltre a intervalli temporali regolari.
mentre usa la schedulazione basata solo sul timing per i databas, visto che i dati nel database sono aggiornati a intervalli regolari e prevedibili.

Per entambi i casi vi è un ulteriore check sulla apertura/chiusura della schedulazione dell'ambito (esepio. IFRS9, IFRS17, ecc ecc). L'attvità prevede l'interrogazione della tabella

EDCDB.T_GEN_SCHEDULATORE

sulla quale viene fatto un check per verificare l'apertura/chiusura della schedulazione di quel processo, se la schedulazione è chiusa viene mandata una mail ai destinatari indicati nella tabella con l'indicazione della schedulazione chiusa.

Esempio wf Informatica :

con naming e formato previsto, controlla eventuali altri processi già in running ed avvia, se verificate le condizioni, l'etl di caricamento principale.

NOTA: Per evitare spam di mail o problematiche relative a una possibile mail non vista generalmente le schedulazioni vengono gestite prevalentemente lato scheduler di informatica aprendole e chiudendole tramite sr.

Storicizzazione

Processo di storicizzazione prevede la creazione nei conteiner di hist di una folder /project_name/flow_name/yyyy/mm/dd/ dove yyyy/mm/dd/ è la data in cui è statop eseguito il processo:

- Storicizzazione file input (copia nel relativo path nel container di history) h_landing/project_name/flow_name/yyyy/mm/dd/
- Storicizzazione file di staging nel container di history h_staging/project_name/flow_name/yyyy/mm/dd/
- Storicizzazione file di reference nel container di history h_reference/project_name/flow_name/yyyy/mm/dd/

Configurazione ambiente

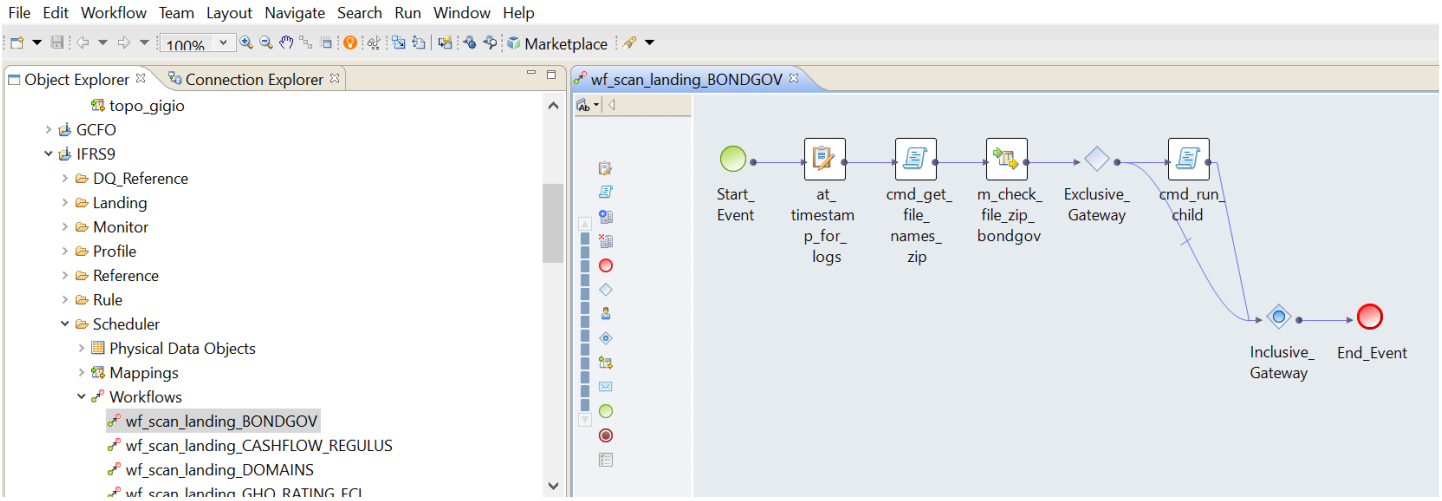
Schema EDCPORTAL

- FIELDS (una riga per stream progettuale es. IFRS9, IFRS17)
- FLOWS (una riga per singolo flusso, FK verso FIELDS)

Schema EDCDB

- T_GEN_ANAGRAFICA_MACRO_SCORECARD (1 riga per scorecard)
- T_GEN_ANAGRAFICA_SCORECARD (1 riga per scorecard/domain, FLOW_ID = NAME di FLOWS, FK verso T_GEN_ANAGRAFICA_MACRO_SCORECARD)
- T_GEN_rule_scorecard (anagrafica delle regole di qualità, una riga per ogni regola di ogni macro_scorecard, FK logica verso T_GEN_ANAGRAFICA_MACRO_SCORECARD)
- T_GEN_SEND_EMAIL (1 riga per chiave flusso/domain, contiene i destinatari delle mail di quel flusso)
- T_GEN_ SCHEDULATORE (1 riga per flusso, NOME_FLUSSO = NAME di FLOWS)

Fifura:


File Edit Workflow Team Layout Navigate Search Run Window Help
Marketplace
Object Explorer
Connection Explorer
topo_gigio
GCFO
IFRS9
DQ_Reference
Landing
Monitor
Profile
Reference
Rule
Scheduler
Physical Data Objects
Mappings
Workflows
wf_scan_landing_BONDGOV
wf_scan_landing_CASHFLOW_REGULUS
wf_scan_landing_DOMAINS
wf_scan_landing_BONDGOV
Start_ Event
at_ timestamp_for_ logs
cmd_get_ file_ names_ zip
m_check_ file_zip_ bondgov
Exclusive_ Gateway
cmd_run_ child
Inclusive_ Gateway
End_Event