wemetrix
we measure you manage

ppc data cleansing

Methodology Documentation of Project Deliverables

15, Eleftheriou Venizelou

PC 10565, Athens

Greece

info@wemetrix.com

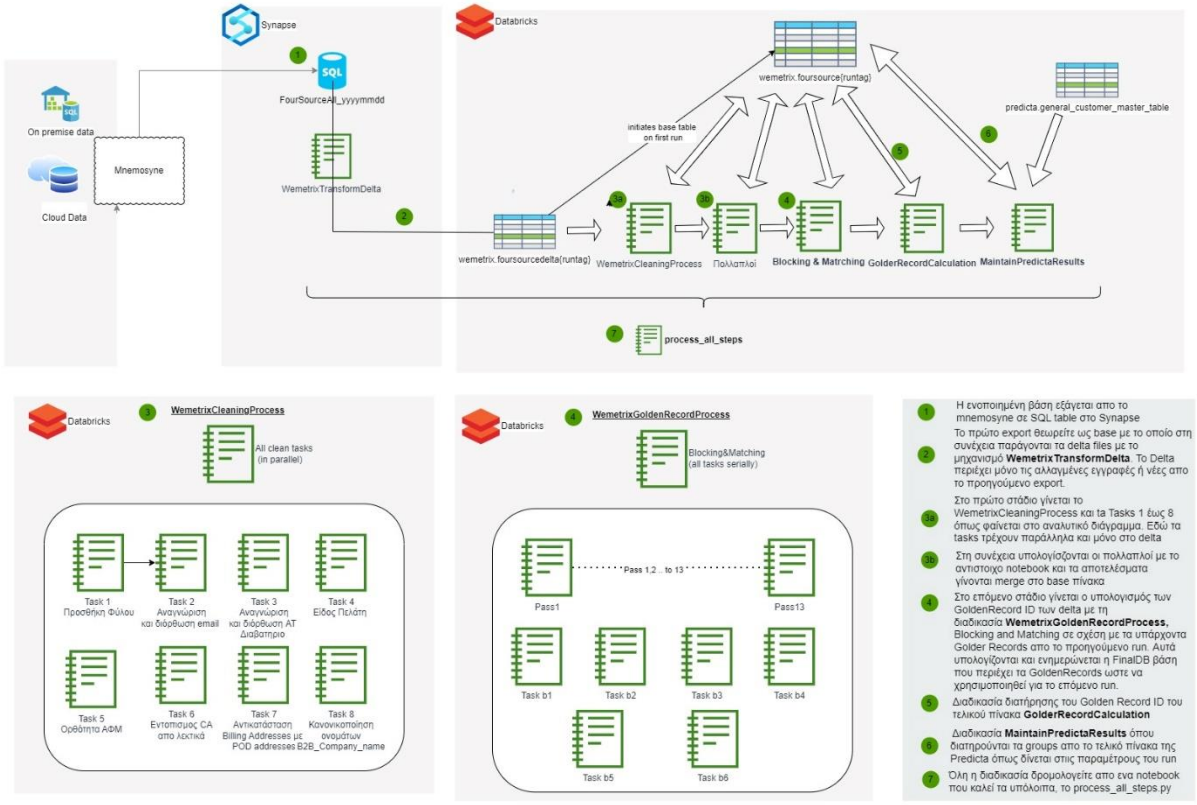
ΠΕΡΙΕΧΟΜΕΝΑ

ΣΧΕΔΙΑΣΜΟΣ DATACLEANSING – ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ

Συνέχεια των συζητήσεων με την ομάδα επιστήμης μεγάλων δεδομένων της ΔΕΗ και αναθεώρηση της υπάρχον υποδομής ακολουθεί ο παρακάτω σχεδιασμός της λύσης που αποσκοπεί :

1. Τη βέλτιστη και ταχύτερη επεξεργασία των δεδομένων για την εκτέλεση των σκοπών του παρόντος.
2. Την ελάχιστη δυνατή ανάγκη υποστήριξης και συντήρησης του συστήματος μελλοντικά.
3. Την προσαρμογή με την υπάρχουσα τεχνογνωσία της ομάδας της ΔΕΗ ώστε να είναι εύκολη η περαιτέρω εξέλιξή του μετά τη παράδοση του έργου.

Συγκεκριμένα οι απαιτούμενες υποεργασίες ξεκινούν από το Synapse workspace που αποθηκεύονται τα παραγωγικά δεδομένα και θα ολοκληρώνονται στην υπάρχον Azure DataBricks υποδομή μέσω κατάλληλων DataBricks notebooks. Η όλη διαδικασία δρομολογείτε από ένα κεντρικό DataBricks notebook που θα καλεί τόσο τις απαιτούμενες stored procedures του synapse όσο και τα υπόλοιπα notebook της διαδικασίας παραμετρικά.



Περιγραφή διαδικασίας:

Η όλη διαδικασία αποθηκεύει τα αποτελέσματα στο base πίνακα foursource{runtag} που δημιουργείτε με το πρώτο run.

Μπορούν να τρέξουν χωρίς να επηρεάζονται μεταξύ τους περισσότερα του ενός runs αλλάζοντας τη μεταβλητή “runtag” στις παραμέτρους του notebook που ακολουθεί. Κάθε ενδιάμεσος και τελικός πίνακας περιέχει τη παράμετρο αυτή.

1. Η ενοποιημένη βάση εξάγεται από το Mnemosyne σε SQL Table (FourSourceAll_yyymmdd)
2. Το πρώτο export θεωρείτε το base αρχείο με το οποίο στη συνέχεια παράγονται τα delta files με το μηχανισμό **WemetrixTransformDelta**. Το Delta περιέχει μόνο τις αλλαγμένες ή νέες εγγραφές απο το προηγούμενο export. Παράλληλα το νέο table αντικαθιστά το base για τον υπολογισμό του επόμενου delta. Τα delta data εξάγονται σε parquet αρχείο για τους επόμενους υπολογισμούς.
3. Στο πρώτο στάδιο γίνεται ο καθορισμός των data σύμφωνα με το **WemetrixCleaningProcess** και τα Tasks 1 έως 8 όπως φαίνεται στο αναλυτικό διάγραμμα παρακάτω. Εδώ τα tasks τρέχουν παράλληλα και μόνο στη delta βάση.
4. Στη συνέχεια υπολογίζονται οι “Πολλαπλοί” και ενημερώνεται ο βασικός πίνακας με τα αποτελέσματα.
5. Στο επόμενο στάδιο γίνεται ο υπολογισμός των Golden Record των delta με τη διαδικασία **WemetrixGoldenRecordProcess** που εμπεριέχει το blocking and

matching σε σχέση με τα υπάρχοντα Golder Records από το προηγούμενο run. Αυτά υπολογίζονται και ενημερώνεται η Foursource{runtag} βάση που περιέχει τα τελικά GoldenRecords.

6. Στη συνέχεια τρέχει η διαδικασία υπολογισμού του Golden Record ID του τελικού πίνακα **GolderRecordCalculation.** Η διαδικασία περιέχει sql reconciliation checks της μοναδικότητας του Grid σε όλα τα master records της βάσης αλλά και εντός των group.
7. Τέλος ανανεώνονται τα Golden Records ανα Group με τη διαδικασία MaintainPredictaResults η οποία επαναυπολογίζει τα Golden Record με βάση το πίνακα αναφοράς predicta που δηλώνεται στις παραμέτρους και διατηρεί τα νεα groups μόνο για το δελτα.

Στο Azure Databricks παρέχεται ένα οπτικό περιβάλλον για τον σχεδιασμό, την παρακολούθηση και τη διαχείριση των μετασχηματισμών των δεδομένων, εξασφαλίζοντας την ομαλή ροή κατά τη διάρκεια της αναλυτικής διαδικασίας. Επίσης τα notebooks έχουν σχεδιαστεί παραμετρικά και μπορούν να καθορίζονται από τις ρυθμίσεις στο configuration notebook.

Task	Notebook Path
A1. Gender	/Shared/wemetrix/cleansing/gender_identification
A2. Email	/Shared/wemetrix/cleansing/email_correction
A3. AT	/Shared/wemetrix/cleansing/AT_Passport_info
A5. VAT Discard	/Shared/wemetrix/cleansing/AFM_discard
A6. CA Discard	/Shared/wemetrix/cleansing/CAs_discard
A8. B2B Norm	/Shared/wemetrix/cleansing/B2B_names_normalization
B1. B&M	/Shared/wemetrix/blocking-and-matching/blocking_and_matching_enhanced
B2. GR	/Shared/wemetrix/blocking-and-matching/GoldenRecordGR
C1. Delta	/Shared/process_all_steps
C2. Maintain GR	/Shared/process_all_steps
D1. Multiples	/Shared/wemetrix/blocking-and-matching/pollaploi
D2. Multiple-mpampades-fuzzy-matching	/Shared/wemetrix/blocking-and-matching/pollaploi_mpampades_fuzzy

ΕΡΓΑΣΙΕΣ ΚΑΤΗΓΟΡΙΑΣ Α

ΕΡΓΑΣΙΑ Α.1 ΠΡΟΣΘΗΚΗ ΦΥΛΟΥ ΑΝΤΙ-ΣΥΜΒΑΛΛΟΜΕΝΟΥ GENDER

Διαδικασία εύρεσης φύλου

1. Εύρεση Gender από SAP (dbfs:/mnt/PRD/curated/SAP_ISU/but000)
2. Αντιστοίχιση "μικρού" ονόματος με βάση δεδομένων που περιλαμβάνει ελληνικά και ξένα ονόματα με ελληνικούς και λατινικούς χαρακτήρες
3. Εύρεση φίλου από την κατάληξη του επιθέτου και του μικρού ονόματος εφ' όσον είναι ελληνικά και έχουν τουλάχιστον 5 γράμματα

Οι εγγραφές που εξαιρούνται είναι:

1. Πελάτες που δεν έχουν ούτε PrFirst_Name ούτε PrLast_Name
2. PrCustomer_Type = "organization"
3. PrCustomer_Type_IIS με τιμές "municipalities"

Κατόπιν ολοκλήρωσης της διαδικασίας στα συμβόλαια του πίνακα "/dbfs/mnt/databricks/master_table_mnemosyne_2023_05_31.parquet":

1. Από τις 18,220,805 εγγραφές οι 16,567,400 πληρούν τα κριτήρια ένταξης στη διαδικασία εύρεσης φύλου, όπως αναφέρεται παραπάνω
2. Ακολουθήθηκε η διαδικασία αντιστοίχισης όπως περιγράφεται παραπάνω
3. Τα αποτελέσματα μπορούν να περαστούν στον κύριο πίνακα κάνοντας "left join" πάνω στα πεδία "SOURCE" και "Contract_Account_ID"

Τελική κατανομή φύλλου:

Gender	count
M	9439910
F	5522796
null	1604694

Κατανομή χρησιμοποιούμενων μεθόδων

Gender_Info	count
First Name Match	9396299
SAP	5308811
null	1604694
First & Last Name Ending	257596

a. Η διαδικασία μπορεί να επαναλαμβάνεται και στα delta με τον παρακάτω κώδικα χρησιμοποιώντας τα κατάλληλα input_table, output_table
b. Αποτελέσματα στον "wemetrix.gender_match_results_20231102"
c. Κώδικας /Shared/wemetrix/cleansing/gender_identification

ΕΡΓΑΣΙΑ Α.2 ΑΝΑΓΝΩΡΙΣΗ ΚΑΙ ΔΙΟΡΘΩΣΗ EMAIL

Δημιουργία λίστας έγκυρων domain και ταξινόμηση με βάση τη συχνότητα εμφάνισης τους (Ο έλεγχος εγκυρότητας γίνεται με DNS queries - Python email_validator module)

1. Αντικαθίστανται ελληνικά γράμματα από greeklish πχ α --> a, θ --> th.
2. Γίνεται έλεγχος στο local κομμάτι (πριν το @) για άκυρους χαρακτήρες οι οποίοι αφαιρούνται
3. Έλεγχος αν το domain κομμάτι υπάρχει στη λίστα με τα έγκυρα domain που διατηρούμε. Αν όχι, τότε δίνεται νέο DNS query. Αν αυτό αποτύχει τότε γίνεται έλεγχος για άκυρους χαρακτήρες οι οποίοι αφαιρούνται και στη συνέχεια fuzzy matching με το πιο κοντινό από τα έγκυρα domains.

4. Σε περιπτώσεις που λείπει ο κωδικός χώρας από το domain πχ hotmail τότε συμπληρώνεται με βάση τα πιο συχνά εμφανιζόμενα όπως hotmail.com αντί για hotmail.gr
5. Οι διορθώσεις του domain γίνονται αποδεκτές μόνο αν διαφέρουν κατά 4 το πολύ χαρακτήρες από το αρχικό. Αυτός ο αυστηρός κανόνας έχει στόχο τη διόρθωση των προφανών λαθών και την αποφυγή αβέβαιων διορθώσεων.
6. Στις περιπτώσεις που δεν υπάρχει παπάκι ελέγχεται αν το τελευταίο κομμάτι της διεύθυνσης είναι (μέρος) έγκυρου domain, προστίθεται το παπάκι στην κατάλληλη θέση και εκτελούνται τα βήματα 1-4 κανονικά.
7. Η λίστα ενημερώνεται με τα νέα έγκυρα domain για να αποφεύγεται η επαναλαμβανόμενη εκτέλεση του ίδιου DNS query
8. Το σύνολο των αποτελεσμάτων αποθηκεύτηκε στον wemetrix.email_correction_results

SOURCE	Contract_Account_ID	Email	Email_Info	Email_Correct	B2B_Email	B2B_Email_Info	B2B_Email_Correct
EBILL	300004979390	nikosmpenetos1gmail.com	Unknown_Format	nikosmpenetos1@gmail.com	null	null	null
EBILL	300005501568	xalkikos@gmail .com	Unknown_Format	xalkikos@gmail.com	null	null	null
EBILL	300006046288	ir.furkan70@gmail. com	Unknown_Format	ir.furkan70@gmail.com	null	null	null
EBILL	300007429879	euaggelia-skylla@hotmail	Unknown_Format	euaggelia-skylla@hotmail.com	null	null	null
EBILL	300007821175	Îºalimnixk@.gmail	Unknown_Format	kalimnixk@gmail.com	null	null	null
EBILL	300008510709	sales@gb_bianco.gr	Unknown_Format	sales@agrinio.gr	null	null	null
EBILL	300012779100	gmak@#$/@gmail.com	Unknown_Format	gmak@gmail.com	null	null	null

9. Η διαδικασία τρέχει για τα πεδία Email και B2B_Email
10. Κώδικας /Shared/wemetrix/cleansing/email_correction

ΕΡΓΑΣΙΑ Α.3 ΑΝΑΓΝΩΡΙΣΗ ΚΑΙ ΔΙΟΡΘΩΣΗ ΑΤ (ΣΤΡΑΤΙΩΤΙΚΗΣ, ΑΣΤΥΝΟΜΙΚΗΣ, ΠΟΛΙΤΙΚΗΣ) & ΔΙΑΒΑΤΗΡΙΟ (ΕΛΛΗΝΙΚΟ Η ΞΕΝΟ)

Διαδικασία Αναγνώρισης & διόρθωσης ΑΤ

1. Δημιουργία νέας στήλης AT_Info_Wemetrix για αποθήκευση αποτελεσμάτων και σύγκριση με την υπάρχουσα κατηγοριοποίηση του AT_Info.
2. Για τη διαδικασία χρησιμοποιήθηκαν μόνο εγγραφές που προέρχονται από το SAP.
3. Για τη διαδικασία χρησιμοποιήθηκε η στήλη ClAT.
4. Αν το ClAT έχει AT_Format -2 γράμματα (Λατινικά & Ελληνικά) and 6 αριθμούς-, θα κατηγοριοποιηθεί ως AT_Format.
5. Αν το ClAT έχει AT_Format -1 γράμμα (Λατινικά & Ελληνικά) and 6 αριθμούς, θα κατηγοριοποιηθεί ως AT_Format.
6. Ταξινόμηση σύμφωνα με το AT_Institute:
    - α Από αυτή τη διαδικασία εξαιρούνται όσα έχουν ήδη ταξινομηθεί στα προηγούμενα βήματα και οι εγγραφές που δεν έχουν τιμή στο ClAT.
    - β Αν το AT_Institute περιέχει αυτά τα λεκτικά ['ΑΤ','ΤΑ','ΥΑ'] και το ClAT δεν έχει AT_Format -2 γράμματα (Λατινικά & Ελληνικά) and 6 αριθμούς- ή -1 γράμμα (Λατινικά & Ελληνικά) and 6 αριθμούς-, θα κατηγοριοποιηθεί ως Invalid_value.
    - γ Αν είτε το AT_Institute είτε το ClAT περιέχουν αυτά τα λεκτικά ['ΓΕΝ','ΓΕΣ','ΓΕΑ','ΓΕΕΘΑ','ΕΛΑΣ','ΕΛΛΑΣ','ΛΣ','ΥΑ','ΕΛ.ΑΣ','ΑΣΤΥΝΟΜΙΑ','ΛΙΜΕΝ','ΠΥΡ','ΣΩΜΑΤ','ΣΤΡΑΤ'] και το ClAT έχει Army_Police_Format -9 αριθμούς-, θα κατηγοριοποιηθεί ως Army_Police_Format αλλιώς null.
    - δ Αν είτε το AT_Institute είτε το ClAT περιέχουν αυτά τα λεκτικά ['ΔΙΑΒ.','ΔΙΑΒ','ΔΙΑΒΑΤΗΡΙΟ','ΑΔΕΙΑ','ΔΙΑΜΟΝΗ','ΑΣΥΛ','REPU','ΑΡΧΕΣ','Α.Ε.Α. /Δ.Δ','ΑΕΑ/ΔΔ','ΔΗΜΟΚΡΑΤΙΑ'], θα κατηγοριοποιηθεί ως Passport_Format.

ε Χρησιμοποιούμε βάση δεδομένων με τα ονόματα όλων των χωρών είτε στα Αγγλικά είτε στα Ελληνικά.

στ Αν το AT_Institute περιέχει το όνομα κάποια χώρας είτε στα αγγλικά είτε στα ελληνικά, θα κατηγοριοποιηθεί ως Passport_Format.

7. Διαχείριση των εγγραφών που δεν ταξινομήθηκαν μετά την παραπάνω διαδικασία (δλδ. δεν πήραν τιμή στη στήλη AT_Info_Wemetrix)

α Αν το AT_Info_Wemetrix είναι null και το AT_Info είναι ίσο είτε με Invalid_Value είτε με Passport_Format, θα κατηγοριοποιηθεί ως Invalid_Value ή Passport_Format αντίστοιχα

<u>Διαδικασία Αναγνώρισης & διόρθωσης Διαβατηρίου</u>

Οι παρακάτω διαδικασίες εκτελούνται σειριακά (για να υπάρχει μια προτεραιότητα στην κατηγοριοποίηση)

1. Δημιουργία νέας στήλης Passport_Info_Wemetrix για αποθήκευση αποτελεσμάτων και σύγκριση με την υπάρχουσα κατηγοριοποίηση του Passport_Info.
2. Για τη διαδικασία χρησιμοποιήθηκαν μόνο εγγραφές που προέρχονται από το SAP.
3. Για τη διαδικασία χρησιμοποιήθηκε η στήλη ClPassport.
4. Για τις χώρες ALBANIA, GEORGIA, PAKISTAN, USA, UKRAINE, ROMANIA, GBR, CHINA, BANGLADESH, EGYPT, BULGARIA, DEUTSCHLAND, GERMANY, ITALIA δημιουργήθηκε ένα frequency table ώστε να βρούμε τους κανόνες που πρέπει να πληροί το εκάστοτε διαβατήριο(αναλογία γραμμάτων και αριθμών).Για κάθε μια από αυτές τις χώρες δημιουργήθηκε ένας κανόνας. Αν στο ADT_Institute υπάρχει κάποια από αυτές τις χώρες γίνεται ο έλεγχος με τον αντίστοιχο κανόνα και αν το ClPassport δεν είναι null και τον ικανοποιεί, ταξινομείται ως Passport_Format. Εάν όχι ταξινομείται ως Invalid_Value.
5. Ταξινόμηση σύμφωνα με το ADT_Institute:
    α. Από αυτή τη διαδικασία εξαιρούνται όσα έχουν ήδη ταξινομηθεί στα προηγούμενα βήματα και οι εγγραφές που δεν έχουν τιμή στο ClPassport.
    β. Αν το ADT_Institute περιέχει αυτά τα λεκτικά ['ΑΤ','ΤΑ','ΥΑ'] και το ClPassport έχει AT_Format -2 γράμματα (Λατινικά & Ελληνικά) and 6 αριθμούς- ή -1 γράμμα (Λατινικά & Ελληνικά) and 6 αριθμούς-, θα κατηγοριοποιηθεί ως AT_Format αλλιώς ως Invalid_value.
    γ. Αν το ADT_Institute περιέχει αυτά τα λεκτικά ['ΓΕΝ','ΓΕΣ','ΓΕΑ','ΓΕΕΘΑ','ΕΛΑΣ','ΕΛΛΑΣ','ΛΣ','ΥΑ','ΕΛ.ΑΣ','ΑΣΤΥΝΟΜΙΑ','ΛΙΜΕΝ','ΠΥΡ','ΣΩΜΑΤ','ΣΤΡΑΤ'] θα κατηγοριοποιηθεί ως Invalid_Value.
    δ. Αν είτε το ADT_Institute είτε το ClPassport περιέχει αυτά τα λεκτικά ['ΔΙΑΒ.','ΔΙΑΒ','ΔΙΑΒΑΤΗΡΙΟ','ΑΔΕΙΑ','ΔΙΑΜΟΝΗ','ΑΣΥΛ','REPU','ΑΡΧΕΣ','Α.Ε.Α./Δ.Δ','ΑΕΑ/ΔΔ','ΔΗΜΟΚΡΑΤΙΑ'], θα κατηγοριοποιηθεί ως Passport_Format.
    ε. Αν το ADT_Institute περιέχει το όνομα κάποια χώρας είτε στα αγγλικά είτε στα ελληνικά, θα κατηγοριοποιηθεί ως Passport_Format.
6. Διαχείριση των εγγραφών που δεν ταξινομήθηκαν μετά την παραπάνω διαδικασία (δλδ. δεν πήραν τιμή στη στήλη Passport_Info_Wemetrix)
    α. Αν το Passport_Info_Wemetrix είναι null και το Passport_Info είναι ίσο με Passport_Format, θα κατηγοριοποιηθεί ως Passport_Format.
    β. Αν το Passport_Info_Wemetrix και το ADT_INSTITUTE είναι null και το Passport_Info είναι ίσο με AT_Format και το ClPassport έχει AT_Format -2 γράμματα (Λατινικά & Ελληνικά) and 6 αριθμούς- ή -1 γράμμα (Λατινικά & Ελληνικά) and 6 αριθμούς-, θα κατηγοριοποιηθεί ως AT_Format αλλιώς ως Invalid_value.

Κατόπιν ολοκλήρωσης της διαδικασίας στα συμβόλαια του πίνακα "/dbfs/mnt/databricks/master_table_mnemosyne_2023_05_31.parquet":

1. **Πίνακας συχνοτήτων των ταυτοτήτων των συμβολαίων ανά κατηγορία**

	AT_Info_Wemetrix	count
1	AT_Format	11348291
2	Army_Police_Format	140866
3	Passport_Format	247829
4	null	838346
5	Invalid_value	2722589

2. **Συγκριτικός πίνακας που δείχνει πόσες εγγραφές έχουν ταξινομηθεί στην ίδια κατηγορία με αυτή του AT_Info και πόσες σε διαφορετική**

	AT_Info	AT_Info_Wemetrix	count
1	Unknown_Format	AT_Format	150777
2	Passport_Format	Passport_Format	146979
3	AT_Format	AT_Format	11197514
4	Unknown_Format	null	627734
5	Unknown_Format	Army_Police_Format	140866
6	Unknown_Format	Passport_Format	100841
7	Invalid_value	Invalid_value	2690787
8	Unknown_Format	Invalid_value	31802
9	null	null	210612
10	Invalid_value	Passport_Format	9

3. **Πίνακας συχνοτήτων των διαβατηρίων των συμβολαίων ανά κατηγορία**

	Passport_Info_Wemetrix	count
1	AT_Format	11654
2	null	15063279
3	Invalid_value	5936
4	Passport_Format	217052

4. **Συγκριτικός πίνακας που δείχνει πόσες εγγραφές έχουν ταξινομηθεί στην ίδια κατηγορία με αυτή του Passport_Info και πόσες σε διαφορετική.**

	Passport_Info	Passport_Info_Wemetrix	count
1	AT_Format	Invalid_value	176
2	AT_Format	Passport_Format	1218
3	null	null	15039931
4	Passport_Format	Passport_Format	130637
5	Unknown_Format	Invalid_value	5159
6	Unknown_Format	AT_Format	283
7	Passport_Format	Invalid_value	601
8	Unknown_Format	null	22880
9	AT_Format	AT_Format	11371
10	Invalid_value	null	468

5. **Τα αποτελέσματα αποθηκεύτηκαν στον "dbfs:/FileStore/wemetrix/ id_info_v2.parquet "**

7. **Πεδία πίνακα: "SOURCE", "Contract_Account_ID","AT","ClAT","AT_INSTITUTE","AT_Info","ClPassport","ADT_INSTITUTE","Passport_Info","AT_Info_Wemetrix","Passport_Info_Wemetrix"**

8. **Η διαδικασία μπορεί να επαναλαμβάνεται και στα delta.**

ΕΡΓΑΣΙΑ Α.4 ΔΗΜΙΟΥΡΓΙΑ ΕΙΔΟΥΣ ΠΕΛΑΤΗ (CUSTOMER TYPE) ΣΕ ΕΠΙΠΕΔΟ GOLDEN RECORD GROUP

Η εργασία περιγράφεται στην διαδικασία της κατηγορία Β που αφορά τα Golden Records

ΕΡΓΑΣΙΑ Α.5 ΑΝΑΓΝΩΡΙΣΗ ΟΡΘΟΤΗΤΑΣ ΕΥΡΕΩΣ ΧΡΗΣΙΜΟΠΟΙΟΥΜΕΝΩΝ ΑΦΜ ΜΕ ΒΑΣΗ ΤΑ ΛΕΚΤΙΚΑ

Διαδικασία Υλοποίησης

1. Μας δόθηκαν 17 ΑΦΜ για έλεγχο από την ομάδα της ΔΕΗ τα οποία είναι τα ακόλουθα:

   090000045, 090153025, 090169846, 090174291, 090283815, 094014201, 094014249, 094014298, 094019245, 094025817, 094026421, 094038689, 094444827, 094449128, 094493766, 099936189, 999510393.

2. Η ομάδα της ΔΕΗ παρέδωσε επίσης τα πιο κοινά λεκτικά τα οποία εμφανίζονται στα 4 παρακάτω πεδία : ClLast_Name , ClCompany_Name, B2B_Company_Name, ClFirst_name
3. Για κάθε ΑΦΜ δημιουργήθηκε ένα frequency table με τα πιο συχνά χρησιμοποιού-μενα ClLast_Name,ClCompany_Name,B2B_Company_Name.
4. Δημιουργήθηκε μια λίστα λεκτικών για έλεγχο για κάθε ένα ΑΦΜ ξεχωριστά και για τα 4 παραπάνω πεδία. Οι λίστες δημιουργήθηκαν με βάση:

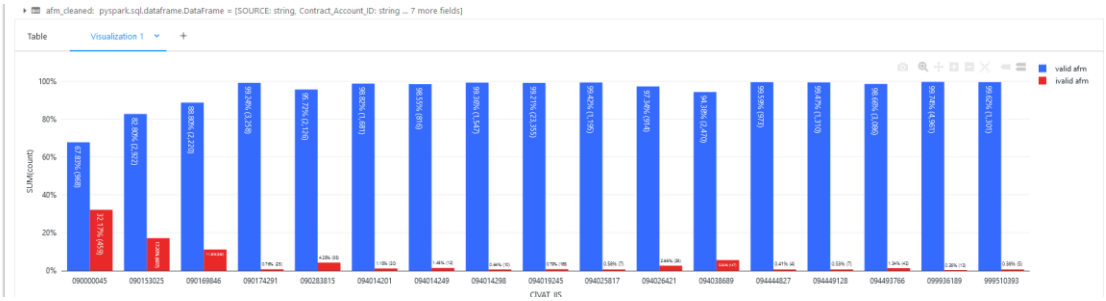
   - τα λεκτικά που μας στάλθηκαν -αρχείο most common vat ids_ΠΥ
   - από ανάλυση (frequency table) σε κάθε πεδίο

5. Κάθε ένα πεδίο περνάει από έλεγχο για να αφαιρεθούν special characters με σκοπό το καλύτερο και πιο ποιοτικό ταίριασμα με τις λίστες
6. Δημιουργείτε καινούριο πεδίο ClVAT_IIS_wemetrix
7. Όσες εγγραφές είχαν κάποιο από τα προαναφερθέντα ΑΦΜ συγκρίθηκαν με τη λί-στα των λεκτικών του χρησιμοποιώντας τη μέθοδο rlike (Similar to SQL regexp_like()).

Σε εγγραφές που κανένα από τα λεκτικά του ΑΦΜ τους δεν υπήρχαν είτε στα παρα-πάνω πεδία το πεδίο ClVAT_IIS_wemetrix έγινε null. Να σημειωθεί ότι βαρύτητα έ-χει δοθεί πρώτα στα πεδία ClFirst_name, ClLast_Name, ClCompany_Name και έπει-τα στο πεδίο B2B, δλδ αν το πεδίο B2B ταιριάζει με κάποιο λεκτικό της λίστας αλλά κανένα άλλο από τα πεδία δεν ταιριάζει, τότε η εγγραφή θεωρείται invalid (flag = 0 & ClVAT_IIS_wemetrix = null)

Αποτελέσματα

1. Το σύνολο των αποτελεσμάτων αποθηκεύτηκε στον "dbfs:/FileStore/wemetrix/afm_cleansing_results.parquet"
2. Στο notebook που παραδόθηκε υπάρχει και κατανομή των valid και invalid εγγρα-φών ανά ΑΦΜ



Η διαδικασία υλοποιήθηκε με spark και μπορεί να επαναλαμβάνεται και στα delta με τον παρακάτω κώδικα χρησιμοποιώντας τα κατάλληλα input_table, output_table.

ΕΡΓΑΣΙΑ Α.6 ΑΝΑΓΝΩΡΙΣΗ ΚΑΙ ΑΠΟΜΟΝΩΣΗ ΕΓΓΡΑΦΩΝ CAS ΜΕ ΣΥΓΚΕΚΡΙΜΕΝΑ ΛΕΚΤΙΚΑ

Διαδικασία απομόνωσης εγγραφών ως αποτέλεσμα fuzzy matching λίστας συγκεκριμένων λεκτικών

1. Αναζήτηση των λεκτικών που μας παραδόθηκαν προς αναγνώριση από την ομάδα της ΔΕΗ πάνω στο πεδίο "ClLast_Name"
3. Σε περιπτώσεις που κάποιο από τα δοθέντα λεκτικά βρεθεί στο "ClLast_Name", το συμβόλαιο αυτό σημειώνεται ώστε να μη συμπεριληφθεί στη διαδικασία golden record group
4. Η διαδικασία σύγκρισης πραγματοποιείται με την χρήση regular expressions

Κατόπιν ολοκλήρωσης της διαδικασίας σε όλη τη βάση /dbfs/mnt/databricks/master_table_mnemosyne_2023_05_31.parquet":

1. Από τα 18,220,805 συμβόλαια τα 43073 περιείχαν στο "ClLast_Name" πεδίο κάποιο από τα λεκτικά
2. Το σύνολο των 43073 συμβολαίων που πρέπει να εξαιρεθούν αποθηκεύτηκε στον "dbfs:/FileStore/wemetrix/string_match_results.parquet"

(1) Spark Jobs

Table +

	SOURCE	Contract_Account_ID	ClLast_Name
1	SAP	300011097569	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
2	SAP	300007943445	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
3	SAP	300006066060	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
4	SAP	300007253154	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
5	SAP	300005007588	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
6	SAP	300011777876	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ
7	SAP	300003819246	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ

3. Σχήμα πίνακα

```
root
 |-- SOURCE: string (nullable = true)
 |-- Contract_Account_ID: string (nullable = true)
 |-- ClLast_Name: string (nullable = true)
```

Ένταξη καινούριου πεδίου «ClCompany_Name» στην διαδικασία

Η παραπάνω διαδικασία έχει επαναληφθεί και στη στήλη ClCompany_Name. Μετά την ολοκλήρωσή της προστέθηκαν 7 νέες εγγραφές. Αυτό οφείλεται στο ότι τα πεδία ClLast_Name & ClCompany_Name ταιριάζουν σχεδόν εξ' ολοκλήρου στις περιπτώσεις των λεκτικών της επιθυμητής λίστας.

Table +

	ClCompany_Name	ClLast_Name	count
1	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ	ΚΟΜΜΕΝΟ ΧΩΡΙΣ ΠΕΛΑΤΗ	39216
2	ΑΝΕΥ ΠΕΛΑΤΗ	ΑΝΕΥ ΠΕΛΑΤΗ	1466
3	ΑΝΕΥ ΠΕΛΑΤΟΥ ΠΑΡΑΒΙΑΣΜΕΝΟ	ΑΝΕΥ ΠΕΛΑΤΟΥ ΠΑΡΑΒΙΑΣΜΕΝΟ	410
4	ΧΩΡΙΣ ΠΡΟΜΗΘΕΥΤΗ	ΧΩΡΙΣ ΠΡΟΜΗΘΕΥΤΗ	355
5	VERMION	VERMION	300
6	ΑΝΕΥ ΠΕΛΑΤΗ ΠΑΡΑΒΙΑΣΜΕΝΟ	ΑΝΕΥ ΠΕΛΑΤΗ ΠΑΡΑΒΙΑΣΜΕΝΟ	224
7	null	TEST	140
8	ΑΛΛΑΓΗ ΣΕ ΚΟΜΜΕΝΟ	ΑΛΛΑΓΗ ΣΕ ΚΟΜΜΕΝΟ	82
9	ΑΝΕΥ ΠΕΛΑΤΟΥ	ΑΝΕΥ ΠΕΛΑΤΟΥ	72
10	ΠΑΡΑΒΙΑΣΜΕΝΟ ΔΕΗ	ΠΑΡΑΒΙΑΣΜΕΝΟ ΔΕΗ	53
11	ΜΕΤΡΗΤΗΣ ΠΑΡΑΒΙΑΣΜΕΝΟΣ	ΜΕΤΡΗΤΗΣ ΠΑΡΑΒΙΑΣΜΕΝΟΣ	52

Επιπλέον , σχετικά με τις 7 καινούριες εγγραφές παρατηρείται το εξής:

Table +

	SOURCE	Contract_Account_ID	ClLast_Name	ClCompany_Name
1	SAP	300002165375	null	ΚΟΙΝΟΤΗΤΑ ΚΟΜΜΕΝΟΥ (ΑΝΤΛΙΟΣΤΑΣΙΟ)
2	SAP	300002152032	null	ΦΟΠ ΚΟΜΜΕΝΟΥ
3	SAP	300002154547	null	ΚΟΙΝΟΤΗΤΑ ΚΟΜΜΕΝΟΥ *ΚΕΠ*
4	SAP	300002293141	null	ΚΟΙΝΟΤΗΤΑ ΚΟΜΜΕΝΟΥ *ΦΟΠ*
5	SAP	300002154239	null	Κ.ΓΡΑΦ.ΚΟΜΜΕΝΟΥ
6	SAP	300002520192	null	ΚΟΙΝΟΤΗΤΑ ΚΟΜΜΕΝΟΥ ΑΡΔΕΥΤ. ΑΝΤΛΙΟΣΤΑΣΙΟ
7	SAP	300007474397	null	ΧΩΡΙΣ ΠΡΟΜΗΘΕΥΤΗ

Μόνο 1 εγγραφή είναι η σωστή , καθώς οι υπόλοιπες αναφέρονται σε κοινότητα «ΚΟΜΜΕΝΟΥ» και κακώς εξαιρούνται. Προτείνουμε να παραμείνουμε στην αρχική υλοποίηση , χρησιμοποιώντας μόνο το πεδίο ClLast_Name.

Ωστόσο , τόσο το παραδοτέο notebook , όσο και ο τελικός πίνακας έχουν ανανεωθεί με την καινούρια υλοποίηση , για έλεγχο και επαλήθευση από την αρμόδια ομάδα

ΕΡΓΑΣΙΑ Α.8 ΚΑΝΟΝΙΚΟΠΟΙΗΣΗ ΟΝΟΜΑΤΩΝ B2B_COMPANY_NAME

Διαδικασία Υλοποίησης

1. Για τη διαδικασία χρησιμοποιήθηκε βάση δεδομένων με τα ΑΦΜ & τα επίσημα ονόματα ιδιωτικών και δημόσιων εταιριών. Η συγκεκριμένη βάση δημιουργήθηκε από λίστες του ΓΕΜΗ, του ΕΒΕΑ και του gov ώστε να διασφαλίζεται ότι κάθε ΑΦΜ έχει την επίσημη ονομασία του. Η εν λόγω βάση αποτελείται από 7958 ΑΦΜ και την επίσημη τους ονομασία και βρίσκεται στο dbfs:/FileStore/wemetrix/db_company_names.parquet
2. Δημιουργία νέας στήλης B2B_Company_Names_Wemetrix για αποθήκευση αποτελεσμάτων, η οποία περιέχει την επίσημη ονομασία των εταιριών με βάση το ΑΦΜ τους.

```
root
 |-- SOURCE: string (nullable = true)
 |-- Contract_Account_ID: string (nullable = true)
 |-- ClVat_IIS: string (nullable = true)
 |-- ClCompany_Name: string (nullable = true)
 |-- B2B_Company_Name: string (nullable = true)
 |-- EnB2B_Company_Name: string (nullable = true)
 |-- B2B_Company_Names_Wemetrix: string (nullable = true)
```

3. Για τη διαδικασία χρησιμοποιήθηκε το πεδίο ClVat_IIS. Στη διαδικασία συμπεριλάβαμε μόνο όσες εγγραφές έχουν τιμή διαφορετική του null στο πεδίο ClVat_IIS .
4. Συνδυάζοντας τους 2 πίνακες, κάνουμε αναζήτηση σχετικά με το ποια ΑΦΜ ταιριάζουν με τα διαθέσιμα ΑΦΜ που έχουμε στη δική μας βάση δεδομένων. Όσα ταιριάζουν παίρνουν στην στήλη B2B_Company_Names_Wemetrix την επίσημη ονομασία που έχουμε εμείς στη βάση για τα συγκεκριμένα ΑΦΜ. Τα υπόλοιπα παίρνουν τιμή null.
5. Κάθε ένα πεδίο περνάει από έλεγχο για να αφαιρεθούν special characters με σκοπό το καλύτερο και πιο ποιοτικό ταίριασμα με τις λίστες

Αποτελέσματα

1. Το σύνολο των αποτελεσμάτων αποθηκεύτηκε στον "dbfs:/FileStore/wemetrix/b2b_name_normalization.parquet"
2. Το σύνολο των εγγραφών που είχε τιμή στο πεδίο B2B_Company_Name είναι 273487. Μετά από αυτή τη διαδικασία οι εγγραφές ανέρχονται στις 351874 (με τιμή στο B2B_Company_Names_Wemetrix).
3. Παρακάτω παραθέτουμε δυο παραδείγματα για να συγκρίνουμε τα existing ονόματα με αυτά που προκύπτουν μετά την ολοκλήρωση της διαδικασίας.

	EnB2B_Company_Name	B2B_Company_Name	B2B_Company_Names_Wemetrix	count
1	OTE AE LIP ELADAS	ΟΤΕ ΑΕ ΛΟΙΠΕΛΛΑΔΑΣ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	11152
2	OTE A E ATTIKIS	ΟΤΕ ΑΕ ΑΤΤΙΚΗΣ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	6850
3	null	null	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	5525
4		null	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	8
5	IP PROSTASIAS POLITI	ΥΠ ΠΡΟΣΤΑΣΙΑΣ ΠΟΛΙΤΗ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	1
6	KIN ARTEMIS	ΚΟΙΝ ΑΡΤΕΜΗΣ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	1
7	GENIKO EPITELIO STRATU	ΓΕΝΙΚΟ ΕΠΙΤΕΛΕΙΟ ΣΤΡΑΤΟΥ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	1
8	KIN KOKARIU	ΚΟΙΝ ΚΟΚΚΑΡΙΟΥ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	1
9	KIN DORIKU	ΚΟΙΝ ΔΩΡΙΚΟΥ	ΟΡΓΑΝΙΣΜΟΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΤΗΣ ΕΛΛΑΔΟΣ ΑΝΩΝΥΜΗ ΕΤΑΙΡΕΙΑ	1

	EnB2B_Company_Name	B2B_Company_Name	B2B_Company_Names_Wemetrix	count
1	null	null	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	2345
2	GENIKO EPITELIO AEROPORIAS	ΓΕΝΙΚΟ ΕΠΙΤΕΛΕΙΟ ΑΕΡΟΠΟΡΙΑΣ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	554
3	GENIKO EPITELIO STRATU	ΓΕΝΙΚΟ ΕΠΙΤΕΛΕΙΟ ΣΤΡΑΤΟΥ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	520
4	GENIKO EPITELIO NAVTIKU	ΓΕΝΙΚΟ ΕΠΙΤΕΛΕΙΟ ΝΑΥΤΙΚΟΥ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	60
5		null	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	35
6	PEDIO VOLIS KRITIS	ΠΕΔΙΟ ΒΟΛΗΣ ΚΡΗΤΗΣ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	8
7	ETHNIKI METEOROLOGIKI IPIRESIA	ΕΘΝΙΚΗ ΜΕΤΕΩΡΟΛΟΓΙΚΗ ΥΠΗΡΕΣΙΑ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	5
8	KIN ASIRU	ΚΟΙΝ ΑΣΣΗΡΟΥ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	1
9	KIN KAVILIS	ΚΟΙΝ ΚΑΒΥΛΗΣ	ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΑΜΥΝΑΣ	1

Η διαδικασία υλοποιήθηκε με spark και μπορεί να επαναλαμβάνεται και στα delta με τον παρακάτω κώδικα χρησιμοποιώντας τα κατάλληλα input_table, output_table.

ΕΡΓΑΣΙΕΣ ΚΑΤΗΓΟΡΙΑΣ Β

ΕΡΓΑΣΙΑ BLOCKING AND MATCHING

Η διαδικασία B&M χρησιμοποιεί το μηχανισμό execute_pass και επεξεργάζεται επαναλαμβανόμενα για κάθε pass το σύνολο των δεδομένων ως εξής:

a. Αποκλείονται οι πολλαπλοί, τα κοινόχρηστα και οι πελάτες εκτός SAP
b. Σε κάθε pass επιλέγονται οι πελάτες που είτε δεν έχουν ομαδοποιηθεί είτε έχουν και είναι master record και ανήκουν στην κατηγορία πελατών που αφορά στο εκάστοτε pass
c. Οι πελάτες του κάθε block συγκρίνονται όλοι μεταξύ τους ως προς συγκεκριμένα πεδία και γίνεται η επιλογή των σχέσεων που πληρούν τους παρακάτω hard rules:
    1. Όμοιο VAT με εξαιρέσεις κενού VAT μόνο στον ένα πελάτη ή διαφορές μέχρι και 2 ψηφία ή 1 αντιμετάθεση σε περιπτώσεις λανθασμένης καταχώρισης
    2. Ίδιο customer type με εξαίρεση περιπτώσεις όπου ο ένας πελάτης έχει null
    3. Όμοια λεκτικά
    4. Όμοια τα υπόλοιπα πεδία που ορίζονται σε κάθε pass αλλά με χαμηλότερη βαρύτητα από τα λεκτικά
d. Από τα παραπάνω ζευγάρια δημιουργούνται τα γκρουπ εκείνα που παρουσιάζουν μεγαλύτερη συνοχή (matching ratio) και μέγεθος
e. Στη συνέχεια υπολογίζεται το matching ratio του κάθε πελάτη ως προς το σύνολο του γκρουπ αλλά και ο μέσος όρους του γκρουπ συνολικά
f. Για την επιλογή του master record επιλέγεται η εγγραφή του κάθε γκρουπ με το μεγαλύτερο matching ratio. Σε περιπτώσεις ισοπαλίας επιλέγεται ο πελάτης με τα περισσότερα συμπληρωμένα πεδία ή το πιο μεγάλο αριθμητικά CA. Εξαίρεση αποτελούν γκρουπ στα οποία υπάρχει πελάτης που ορίστηκε master record σε προηγούμενο pass, οπότε και παραμένει master record του νέου γκρουπ ενώνοντας τις προηγούμενες εγγραφές. Κάθε γκρουπ έχει ένα και μόνο master record.

Τελικά πεδία:

- Wemetrix_group_number: Αύξοντας αριθμός γκρουπ εντός pass με καθαρά αλγοριθμική χρήση χωρίς business νόημα

- Wemetrix_mp_customer_code: Το CA του Master Record του γκρουπ στο οποίο ανήκει ένας πελάτης
- Wemetrix_pass: Ο αριθμός του pass στο οποίο ομαδοποιήθηκε ο πελάτης
- Wemetrix_match_type: Το είδος ομαδοποίησης MP/DA/null
- Wemetrix_matching_ratio: Το ποσοστό ομοιότητας του πελάτη με τις υπόλοιπες εγγραφές του γκρουπ που ανήκει

Τα blocking πεδία, τα matching πεδία και τα customer types που αφορούν στο κάθε pass ορίζονται στο dict_parameters.

Το παραγωγικό notebook είναι το /Shared/wemetrix/blocking-and-matching/blocking_and_matching_enhancedπου περιλαμβάνει τα «καθαρά» πεδία ταυτοτήτων και διαβατηρίων ClPassport_IIS_Wemetrix και ClAT_IIS_Wemetrix.

ΕΡΓΑΣΙΑ GOLDEN RECORD GROUPS

Ζητούμενο σενάριο

Η διαδικασία εφαρμογής της λογικής των main records ανά group και η δημιουργία των Golden πεδίων ανά γκρουπ. Κάθε γκρουπ θα πρέπει να έχει ένα μόνο main record, καθώς και τα golden πεδία υιοθετούνται από όλες τις εγγραφές του γκρουπ. Ως γκρουπ ορίζεται ένα σύνολο λογαριασμών. Σε κάθε σύνολο υπάρχει μία master εγγραφή και τα παιδιά της. Η master εγγραφή είναι ένας λογαριασμός Contract Account, ο οποίος έχει λάβει την ένδειξη “MP” από την σχετική διαδικασία blocking and matching. Αν σε μία ομάδα (σύνολο) εγγραφών δεν υπάρχει ξεκάθαρο contract με την ένδειξη “MP” (βλ. 15) τότε το group των εγγραφών έχει ως καθορίζεται από το main record που έχει δοθεί σε αυτό το group/σύνολο λογαριασμών.

Υλοποίηση

Η υλοποίηση βασίστηκε στα παρακάτω δύο αρχεία:

- IBM DATASTAGE - 4 GOLDEN RECORD.pdf (1)
- Development and Operational Document Data Cleansing 13122022.pdf (2)

Με μεγαλύτερη βαρύτητα να δοθεί στο πρώτο pdf αρχείο (1).

Αρχικά, για την υλοποίηση των main records ακολουθήσαμε τους παρακάτω κανόνες βάσει και των οδηγιών στο αρχείο (1).

1. If the master record (wemetrix_match_type = MP) is active (Active_Contract = 1) and SOURCE =SAP, then is the Main Record.
2. If not, use Active_Contract =1 and maximum weight and SOURCE = SAP then is Main Record.
3. Otherwise, take the maximum weight and source=SAP then is Main Record.

Ωστόσο, στους κανόνες (2) και (3) παρατηρήθηκαν εγγραφές/groups όπου το main record κατέληγε σε τουλάχιστον (>=) 2 Contract Account IDs. Οπότε η τελική επιλογή για τους κανόνες (2) και (3) με πολλαπλά Contracts που πληρούσαν την προδιαγραφή για main record έγινε βάση του μεγαλύτερου Contract Account ID. Το αποτέλεσμα ήταν κάθε group

να διαθέτει ένα μοναδικό record ως το main record. Με τιμή 1 χαρακτηρίζονται τα main records των group ενώ τα υπόλοιπα records λαμβάνουν την τιμή 0. Εγγραφές με NULL στο πεδίο wemetrix_main_record, δεν πληρούσαν την προϋποθέσεις για τον υπολογισμό του main record, κυριώς επειδή ανήκαν εκτός SAP.

Στην συνέχεια, εφόσον υλοποιήθηκε το main record και το κάθε group απέκτησε από ένα, προχώρησε και η δημιουργία των GR πεδίων βάση των κανόνων που δόθηκαν στα προαναφερθέντα PDF αρχεία. Στην ίδια λογική των Main Records, όταν εντοπίζονταν ισοπαλίες στην Most Frequent λογική, η επιλογή γινόταν βάσει του μεγαλύτερου length, ή του μεγαλύτερου weight (wemetrix_matching_ratio) ή/και του μεγαλύτερου Contract Account ID. Η επιλογή εξαρτάται από την φύση του πεδίου. Να τονιστεί ότι στις κολώνες που περιγράφονται ως buffer column στα εν λόγω pdf αρχεία προστέθηκε η λέξη GR_ στην αρχή του ονόματος του πεδίου. Παραδείγματος χάριν, το πεδίο ClFirst_Name έλαβε νέα ονομασία ως GR_ClFirst_Name αφού εφαρμόστηκε η most frequent logic, σύμφωνα με το pdf αρχείο IBM DATASTAGE - 4 GOLDEN RECORD.

Η most frequent λογική ακολουθεί τα παρακάτω βήματα:

1. Filter out NULL values. Βάσει του μεγέθους/length των τιμών του εκάστοτε πεδίου δίνεται προτεραιότητα στις non-null εγγραφές.
2. Group qsMatchSetID. Ομαδοποίηση βάση των groups (wemetrix_mp_customer_code).
3. Υπολογισμός του frequency των εγγραφών ανάλογα την target κολώνα. Ως target ορίζεται η κολώνα της οποίας θέλουμε να υπολογίσουμε το golden record. Για παράδειγμα, PrCustomer_Type_IIS, PrFirst_Name, ClAT_IIS, κλπ.
4. Order by frequency in descending order.
5. Επιλογή της πρώτης γραμμής με την μέγιστη συχνότητα (frequency) ανά group.

Σε περιπτώσεις ισοπαλίας στο 5ο βήμα, η επιλογή έγινε με βάση το μέγιστο length, μέγιστο weight ή/και το μέγιστο contract account id. Επιπλέον σε μεμονωμένες περιπτώσεις, ανάλογα και της περιγραφής του spec, χρησιμοποιήθηκαν και οι κολώνες Active_Contract (δίνεται προτεραιότητα στις Active εγγραφές/λογαριασμούς), ή/και main_record (δίνεται προτεραιότητα στο main record του εκάστοτε group).

Το παραγωγικό notebook βρίσκεται στο Databricks και συγκεκριμένα στην τοποθεσία: **/Workspace/Shared/wemetrix/blocking-and-matching/GoldenRecordGR.**

Το παραγωγικό notebook χρησιμοποιεί μεθόδους (functions) οι οποίες βρίσκονται στον φάκελο /src και συγκεκριμένα στο αρχείο: **/Workspace/Shared/wemetrix/blocking-and-matching/src/py_functions.py**

Στο αρχείο *py_functions.py* μπορούν να γίνονται όλες οι αλλαγές που αφορούν τον τρόπο υπολογισμού των golden πεδίων. Κάθε μέθοδος έχει μια αναλυτική περιγραφή σε μορφή “””DOCSTRING””” για την αναφορά του σκοπού της μεθόδου, των μεταβλητών που δίνονται ως όρισμα στην μέθοδο καθώς και το αποτέλεσμα της μεθόδου.

Τέλος, να τονιστεί ότι κάθε μέθοδος για τον υπολογισμό του golden record ακολουθεί την παρακάτω λογική:

1. Φιλτράρισμα του αρχικού dataset με τις κολώνες που θα χρησιμοποιηθούν στην μέθοδο για τον υπολογισμό ενός golden record. Εφαρμογή φίλτρων αναλόγως της περιγραφής. Παραδείγματος χάριν, στην πλειοψηφία των golden record χρησιμοποιήθηκαν εγγραφές από το σύστημα SAP.

2. Από το snapshot του dataset γίνεται μία ομαδοποίηση των εγγραφών ανά group/σύνολο και της κολώνας που θα εξάγουμε το golden record. Παραδείγματος χάριν, για τον υπολογισμό του πεδίου GR_Customer_Type, ο snapshot πίνακας ομαδοποιείται ανά group και PrCustomer_Type_IIS.

3. Μετά την ομαδοποίηση εφαρμόζονται aggregations στις υπόλοιπες κολώνες και το σύνολο των γραμμών για τον υπολογισμό των frequent εγγραφών ανά group.

4. Στην συνέχεια δημιουργούνται τα windows, τα οποία χωρίζουν τις εγγραφές ανά group εφαρμόζοντας ένα είδους προτεραιότητας/σειράς βάσει της σειράς που έχουν δοθεί οι κολώνες στο window function. Παραδείγματος ένα window μπορεί να έχει την παρακάτω δομή *(desc(length), desc(count)).*

5. Εφαρμόζεται το window function με μια λογική *dense_rank()*, *rank()* και δίνεται με αυτό τον τρόπο στις εγγραφές του κάθε group ένας αριθμός προτεραιότητας (1, 2, 3,…N). Η προτεραιότητα καθορίζεται από το window function. Παραδείγματος χάριν, εάν λάβουμε υπόψιν το παράδειγμα στο παραπάνω βήμα τότε οι λογαριασμοί ανά group θα πάρουν προτεραιότητα πρώτα βάσει του μεγέθους τους (length) και μετά βάσει της συχνότητας τους (count). Είναι σημαντικό να τονιστεί πως σε κάθε group εγγραφών θα πρέπει να υπάρχει **1 και μόνο** επικρατέστερη τιμή. Αν δεν υπάρχει τότε σημαίνει ότι θα πρέπει να προστεθούν επιπλέον κολώνες προτεραιότητας στο window function καθώς οι υπάρχουσες κολώνες **ΔΕΝ** αρκούν για να καθοριστεί η επικρατέστερη τιμή ανά group.

6. Τέλος, εφόσον διασφαλιστεί ότι κάθε group έχει μία και μόνο επικρατέστερη τιμή τότε εξάγεται ο τελικός snapshot πίνακας με τα groups και την κολώνα που θα καθορίσει το golden record.

7. Ο πίνακας, αποτέλεσμα της εφαρμοσμένης μεθόδου, γίνεται join με τον αρχικό πίνακα στο πεδίο των groups. Σε αυτό το σημείο ολοκληρώνεται η μεθοδολογία υπολογισμού του golden record.

Να σημειωθεί σε αυτό το σημείο ότι στην υλοποίηση του notebook έχει προστεθεί η λογική αποθήκευσης του βασικού πίνακα με τα gr πεδία σε parquet αρχεία σε διάφορα σημεία της υλοποίησης. Το γράψιμο του πίνακα εξυπηρετεί στην απόδοση εκτέλεσης των μεθόδων και την βέλτιστη εκτέλεση του κώδικα. Η μεθοδολογία υλοποιείται με τα λεγόμενα ***checkpoints*** τα οποία αποτελούν σημείο αναφοράς σε διάφορα στιγμιότυπα του παραγωγικού κώδικα. Κάθε σημείο αναφοράς (checkpoint) έχει μία εικόνα του αρχικού πίνακα εμπλουτισμένο με ορισμένα από τα gr πεδία. Να τονιστεί ότι στο πέρας της εκτέλεσης κάθε προσωρινός πίνακας για κάθε checkpoint διαγράφεται. Εξασφαλίζοντας την διατήρηση του παραγωγικού περιβάλλοντος καθαρό από περιττούς πίνακες.

ΕΡΓΑΣΙΕΣ ΚΑΤΗΓΟΡΙΑΣ C

ΕΡΓΑΣΙΑ C.1 ΔΗΜΙΟΥΡΓΙΑ ΑΥΤΟΜΑΤΟΥ ΜΗΧΑΝΙΣΜΟΥ DELTA ΕΠΙΚΑΙΡΟΠΟΙΗΣΗΣ

Διαδικασία DELTA μηχανισμού

1. Ο μηχανισμός DELTA υλοποιήθηκε στο synapse καθώς εκεί βρίσκονται οι πίνακες SQL των data που χρησιμοποιούνται. Έτσι επιτυγχάνεται παραγωγή του delta στη πηγή και μεταφορά στο databricks για περαιτέρω υπολογισμούς πολύ μικρότερου όγκου δεδομένων.
2. Το DELTA αναφέρεται στα νέα ή αλλαγμένα records του export των πινάκων FOURSOURCE. Το DELTA υπολογίζεται μεταξύ 2 ημερομηνιών καθώς δέχεται σας input τα ονόματα των πινάκων που θα συγκριθούν καθώς και το όνομα του πίνακα DELTA.
3. Παράδειγμα κλήσης:

```
exec dataanalysis.createdeltatable
 [predicta].[FOURSOURCE_ALL_26_05_2023]',
'[predicta].[FOURSOURCE_ALL_31_05_2023]','DELTAFOURSOURCE_ALL_26_05_2023_31_05_2023'
```

Το συγκεκριμένο execution παράγει το πίνακα DELTA με ~268k rows.

4. Αλγόριθμος υπολογισμού
   α. Δημιουργία 2 index πινάκων για αποθήκευση των κλειδιών και του hash των εγγραφών των 2 πινάκων του input (αρχικός και τελικός).
   β. Υπολογισμός του hash των columns των 2 πινάκων λαμβάνοντας υπόψιν όλες τις κολώνες. Αυτές μπορούν να περιοριστούν μελλοντικά αν δεν θέλουμε το DELTA να προκύπτει από συγκεκριμένες κολώνες μόνο. Λόγο ότι οι κολώνες δηλώνονται αποκλειστικά για του πίνακες αυτού του τύπου ο μηχανισμός DELTA μπορεί να χρησιμοποιηθεί μόνο για πίνακες του ίδιου schema.
   γ. Left join των 2 index πινάκων πάνω στα keys (SOURCE kai Contract_id) και το hashid.
   δ. Φιλτράρισμα του αποτελέσματος για hashid=null που σημαίνει ότι κρατάμε τα records του τελικού πίνακα που έχουν διαφορετικό hash οπότε είναι νέα ή υπάρχοντα που έχουν αλλάξει μια ή περισσότερες κολώνες.
   ε. Inner Join του αποτελέσματος με το τελικό full πίνακα οπότε και δημιουργείτε ο DELTA πίνακας που περιέχει όλα τα records.
   στ. Οι ενδιάμεσοι πίνακες αρχικά γίνονται truncate σε κάθε εκτέλεση και ο τελικός αν υπάρχει γίνεται drop και επαναδημιουργείτε.
5. Το execution του DELTA μπορεί να γίνεται και από το databricks με odbc connection και είναι το τρίτο βήμα του overall procedure μετά το τρέξιμο των includes των configuration και common_functions notebooks. (Shared/wemetrix/process_all_steps).

ΕΡΓΑΣΙΑ C.2 ΔΙΑΤΗΡΗΣΗ ΣΤΑΘΕΡΟΥ GOLDEN RECORD ID ΑΠΟ ΠΙΝΑΚΑ ΑΝΑΦΟΡΑΣ (PREDICTA) ΜΕΤΑΞΥ ΔΙΑΦΟΡΕΤΙΚΩΝ RUNS ΤΟΥ ΜΗΧΑΝΙΣΜΟΥ ΤΑΚΤΙΚΗΣ (ΕΒΔΟΜΑΔΙΑΙΑΣ) ΑΝΑΝΕΩΣΗΣ

Ο μηχανισμός διατήρησης του Golden Record ID βασίζεται στο μηχανισμό DELTA. Με την ολοκλήρωση του blocking and matching κάποια records αλλάζουν group ή και matching type (MP ή DA) λόγο ότι νέα records γίνονται match σε άλλα group από αυτά που ήταν ή με records του DELTA. Για να είναι valid κάθε νέο group πρέπει να έχει μοναδικό GRid για το master record το οποίο θα είναι ίδιο για τα υπόλοιπα records του group. Επίσης θα πρέπει τα records που δεν έχουν αλλαχθεί να διατηρούν τα αρχικά groups του πίνακα αναφοράς (predicta) και οι μεταβολές τις νέας διαδικασίας να εφαρμοστούν μόνο για τα δεδομένα του δελτα.

Η διαδικασία εκτελείτε στο section “Produce output maintaining predicta” του process_all_steps και έχει τη παρακάτω αλγοριθμική προσέγγιση:

1. Αρχικά με join με το πίνανα αναφοράς διατηρούνται τα groups των εγγραφών που δεν είναι δελτα. (Κάθε εγγραφή χαρακτηρίζεται αν είναι δελτα στο συγκεκριμένο run με το πεδίο ‘isdelta’.
2. Στη συνέχεια για τις δελτα εγγραφές εκτελούνται 2 διαδοχικά pass οπου βρίσκεται το σωστό GRid (σε περίπτωση που είναι invalid μετά το update με τα predicta results) ή αν το GRid είναι κενό. Τα 2 passes χρειάζονται γιατί υπάρχει περίπτωση όχι μόνο η εγγραφή αλλά και το master record ενός golden group να έχουν αλλάξει οπότε πρέπει να τρέξει και σε δευτερο επίπεδο για τα master records των groups αυτών ώστε να διορθωθούν με τα σωστά GRids. Υπάρχουν οι εξής περιπτώσεις:
   - Αν το GRid του ενός group δεν είναι master θα βρεθεί αυτό με valid GRid (master) ή θα γίνει το ίδιο master
   - Τα master record με null GRid θα ενημερωθούν με το Contract id. Τα υπόλοιπα records του ίδιου group το κρατούν και ενημερώνονται, διαφορετικά αποκτούν νέο.

Μετά το τέλος της διαδικασίας ενσωματώνεται έλεγχος για την ορθότητα όλων των GR groups του base πίνακα.

ΕΡΓΑΣΙΕΣ ΚΑΤΗΓΟΡΙΑΣ D

ΠΟΛΛΑΠΛΟΙ ΚΑΙ ΠΟΛΛΑΠΛΟΙ ΠΟΛΛΑΠΛΩΝ

1. Εργασία d.1: Υλοποίηση ΠΟΛΛΑΠΛΟΥΣ – Ιδιώτες
2. Εργασία d.2: Υλοποίηση ΠΟΛΛΑΠΛΟΥΣ - Δημόσιο
3. Εργασία d.3: Υλοποίηση ΠΟΛΛΑΠΛΟΥΣ – Δήμοι
4. Εργασία d.4: Υλοποίηση Πολλαπλούς «Πολλαπλών»
5. Εργασία d.5: Αναγνώριση περιπτώσεων Πολλαπλών με «λάθος καταχωρημένο ΑΦΜ»
6. Εργασία d.6: Δημιουργία πεδίου «Μέθοδος Ομαδοποίησης» για την περίπτωση Πολλαπλών.

Το παραγωγικό notebook για την εκτέλεση των πολλαπλών βρίσκεται στην τοποθεσία:
/Workspace/Shared/wemetrix/blocking-and-matching/pollaploi

Το πρώτο βήμα είναι να υπολογιστούν οι πολλαπλοί για περίπου 6400 «ΜΠΑΜΠΑΔΕΣ».
Ενδεικτικά βρέθηκαν,

- ΜΠΑΜΠΑΔΕΣ ΔΗΜΟΣΙΟ: 342
- ΜΠΑΜΠΑΔΕΣ ΔΗΜΟΣ: 6034
- ΜΠΑΜΠΑΔΕΣ ΙΔΙΩΤΗΣ: 79

Το query για την απόκτηση του πίνακα των μπαμπάδων (μας δόθηκε έτοιμο):

```
df_mpampades = spark.sql(
  """
  SELECT
  a.VKONT , vbund, TAXNUM, name_org1, name_org2, name_org3,name_org4
  FROM fkkvk a
  left join fkkvkp b on a.VKONT = b.VKONT
  left join dfkkbptaxnum c on b.GPART = c.PARTNER
  left join but000 d on d.partner = c.partner
  where vktyp = '02'
  AND TAXTYPE = 'GR2'
  and a.vkont like '001%'
  """
)
```

Επίσης, διαβάζονται τα δεδομένα από το excel με τους ΔΗΜΟΥΣ ΚΑΛΛΙΚΡΑΤΗ. Ο πίνακας με τα σχετικά δεδομένα βρίσκεται στο schema *wemetrix.dimoi_kallikrati.* Το CSV αρχείο των ΔΗΜΩΝ ΚΑΛΛΙΚΡΑΤΗ βρίσκεται στο παραγωγικό περιβάλλον, στην τοποθεσία **/Workspace/Shared/wemetrix/blocking-and-matching/ΔΗΜΟΙ_ΚΑΛΛΙΚΡΑΤΗ_utf8.csv**

Εφόσον αποκτήσουμε τους «ΜΠΑΜΠΑΔΕΣ» ανά κατηγορία τότε προχωράμε στην παρακάτω υλοποίηση:

1. Υπολογίζουμε τρεις βασικές κολώνες.
   - Customer_Type_<type>_entagmenos [Τιμή: «Υ»]
   - Wemetrix_ar_pollaplou_<type>_entagmenos [Τιμή: «VKONT» του «ΜΠΑΜΠΑ»]
   - Wemetrix_pollaplos_type_<type>_entagmenos [Τιμή: «Ενταγμένος»]

, όπου type = ΔΗΜΟΣ, ΔΗΜΟΣΙΟ ή ΙΔΙΩΤΗΣ
Η αναγνώριση των ενταγμένων βασίστηκε στον παρακάτω κανόνα:
Όποια εγγραφή έχει ίδιο αριθμό πολλαπλού [πεδίο: Ar_Pollaplou] με τον ΜΠΑΜΠΑ [πεδίο: VKONT] τότε αναγνωρίζεται ως ενταγμένος.

Με την ολοκλήρωση υπολογισμού των τριών προαναφερθέντων πεδίων και την αναγνώριση των *Ενταγμένων* εγγραφών, ο πίνακας γράφεται σε ένα ***checkpoints*** (σημείο αναφοράς). Στη συνέχεια, προχωράμε στο δεύτερο στάδιο της υλοποίησης, στην αναγνώριση των *Ανένταχτων* εγγραφών.

2. Για την αναγνώριση των ανένταχτων εγγραφών έχει σχεδιαστεί και εφαρμοστεί η μέθοδος **find_anentaxtoi_pollaploi().** Η μέθοδος παίρνει συγκεκριμένα ορίσματα και χρησιμοποιεί τα τρία πεδία που δημιουργήθηκαν ανά τύπο στο παραπάνω βήμα (1). Στη συνέχεια δημιουργούνται 4 νέα πεδία.

- Customer_Type_<type>_anentaxtos [Τιμή: «Υ»]
- Wemetrix_ar_pollaplou_<type>_anentaxtos [Τιμή: «VKONT του ΜΠΑΜΠΑ με τους περισσότερους ενεργούς ενταγμένους λογαριασμούς]
- Wemetrix_pollaplos_type_<type>_anentaxtos [Τιμή: «Ανένταχτος»]
- Wemetrix _<type>_anentaxtos_afm [Τιμή: «TAXNUM»]

Ο κάθε ένας από τους πίνακες που παράγεται ανά τύπο (ΔΗΜΟΣ, ΔΗΜΟΣΙΟ, ΙΔΙΩΤΗΣ) χρησιμοποιείται για την δημιουργία των 4 παραπάνω πεδίων. Η σύνδεση των παραγόμενων πινάκων (από την μέθοδο **find_anentaxtoi_pollaploi)** με τον αρχικό πίνακα γίνεται στο κλειδί Contract Account ID για τους ανένταχτους. Ο νέος πίνακας με τα επιπλέον 4 πεδία ανά τύπο γράφεται σε ένα ***checkpoints*** (σημείο αναφοράς).

Στην συνέχεια εφαρμόζουμε ένα concatenation στις κολώνες των βημάτων (1) και (2) ώστε να κρατήσουμε μία τιμή ανά εγγραφή. Εφαρμόζουμε 2 διαφορετικούς τρόπους concatenation ανά εγγραφή σύμφωνα με την παρακάτω λογική:

- Στο πρώτο concatenation λαμβάνουμε υπόψιν:
    1. Τα πρώτα 12 ψηφία από τα πεδία *wemetrix_ar_pollaplou_<type>_entagmenos* και *wemetrix_ar_pollaplou_<type>_anentaxtos* που θέλουμε να ομαδοποιήσουμε για το πεδίο **wemetrix_ar_pollaplou.** Υπενθυμίζεται ότι ο αριθμός πολλαπλού λαμβάνει τιμή από το πεδίο VKONT το οποίο έχει σταθερά 12 ψηφία.
    2. Τα 10 πρώτα ψηφία από τα πεδία *wemetrix_pollaplos_type_<type>_entagmenos* και *wemetrix_pollaplos_type_<type>_anentaxtos* που θέλουμε να ομαδοποιήσουμε για το πεδίο **wemetrix_pollaplos_type.** Υπενθυμίζεται ότι ο τύπος πολλαπλού λαμβάνει δύο τιμές «Ενταγμένος» ή «Ανένταχτος». Κάθε τιμή έχει 10 ψηφία.
    3. Το πρώτο ψηφίο από τα πεδία *Customet_Type_<type>_anentaxtos* και *Customet_Type_<type>_entagmenos* για το πεδίο **Customer_Type_<type>.** Υπενθυμίζεται ότι το πεδίο Customer_Type λαμβάνει μόνο την τιμή «Υ». Κάθε τιμή έχει 1 ψηφίο.
    4. Τα 9 πρώτα ψηφία από τα πεδία *wemetrix_<type>_anentaxtos_afm* για το πεδίο **wemetrix_anentaxtos_afm.** Υπενθυμίζεται ότι το ΑΦΜ έχει σταθερό μέγεθος στα 9 ψηφία.
- Το αποτέλεσμα της πρώτης μεθόδου concatenation είναι μία τιμή ανά εγγραφή/λογαριασμό.

Ωστόσο, παρατηρώντας τα αποτελέσματα της διαδικασίας διαπιστώσαμε ότι υπήρχαν εγγραφές που ήταν εξίσου ενταγμένες ή ανένταχτες σε παραπάνω από ένα τύπο. Δηλαδή

λογαριασμού που ήταν ταυτόχρονα από μπαμπά ΔΗΜΟ, ή ΔΗΜΟΣΙΟ ή/και ΙΔΙΩΤΗ. Για την αναγνώριση αυτών των εγγραφών δημιουργήσαμε τον δεύτερο τρόπο concatenation των αποτελεσμάτων.

Στην δεύτερη λογική concatenation απλά ενώσαμε τα πεδία που προαναφέραμε με ένα «,» (κόμμα) ενδιάμεσα και δόθηκε νέα ονομασία στα αντίστοιχα πεδία με κατάληξη *_all_concat*. Παραδείγματος χάριν, το πεδίο **wemetrix_ar_pollaplou** ονομάστηκε **wemetrix_ar_pollaplou_all_concat.** Σε περίπτωση ποιοτικών ελέγχων θα μπορούσαν να ελεγχθούν οι εγγραφές με μεγάλες τιμές στα πεδία με κατάληξη ***_all_concat*.**

Υπενθυμίζεται, ότι αν μία εγγραφή έχει πολλές comma-separated τιμές σε κάποιο από τα πεδία ***_all_concat*** τότε η εγγραφή έχει αναγνωριστεί σε πολλαπλούς ΜΠΑΜΠΑΔΕΣ από διάφορα types (ΔΗΜΟΣ, ΔΗΜΟΣΙΟ, ΙΔΙΩΤΗΣ). Θα μπορούσατε να δώσετε προσοχή στην εγγραφή με αριθμό λογαριασμού «300012680341» η όποια ανήκει ως «Ανένταχτος» πολλαπλός σε ΔΗΜΟ και ΙΔΙΩΤΗ.

Τα νέα concatenated πεδία γράφονται και αυτά με την σειρά τους στο επόμενο σημείο αναφοράς (checkpoint) για την βέλτιστη εκτέλεση του κώδικα.

Έως τώρα έχουμε προσθέσει στον αρχικό πίνακα τα πεδία:

- **wemetrix_ar_pollaplou & wemetrix_ar_pollaplou_all_concat**
- **wemetrix_pollaplos_type & wemetrix_ar_pollaplou_all_concat**
- **Customer_Type_<type> & Customer_Type_<type>_all_concat**
- **wemetrix_anentaxtos_afm & wemetrix_anentaxtos_afm_all_concat**

Μαζί και παρεμφερή πεδία από τις εκτελέσεις των μεθόδων.

Στην συνέχεια προσθέτουμε στην υλοποίηση ορισμένα πεδία που ζητήθηκαν και έχουν σχέση με τον τύπο ΔΗΜΟΣ. Συγκεκριμένα τα πεδία αυτά είναι,

- LOG.SYMB_SAP
- DIMOS/KOINOTITA ERMI
- KOD.DIMOU KALLIKRATI
- DIMOS KALLIKRATI / KLEISTHENI

Στην συνέχεια ακολουθεί ο υπολογισμός των Contract Account ID ανά ΜΠΑΜΠΑ. Σύμφωνα με τον αριθμό πολλαπλού και το πεδίο VKONT εξάγεται ο αριθμός λογαριασμού του ΜΠΑΜΠΑ. Ωστόσο, όπως σημειώνεται και στο παραγωγικό notebook, βρέθηκαν αρκετοί ΜΠΑΜΠΑΔΕΣ με πολλαπλά Contract Accounts. Εφόσον δεν υπήρχε ξεκάθαρη κολώνα που να υποδεικνύει τον αριθμό λογαριασμού του ΜΠΑΜΠΑ, επιλέξαμε ως επικρατέστερη τιμή λογαριασμού για κάθε ΜΠΑΜΠΑ την μέγιστη τιμή λογαριασμού *max(contract_account_id)*. Συνδέοντας τον πίνακα με τους λογαριασμούς ανά ΜΠΑΜΠΑ (1 προς 1 λογική -> κάθε ΜΠΑΜΠΑΣ πρέπει να αντιστοιχεί σε έναν λογαριασμό CA), δημιουργήθηκε το πεδίο **wemetrix_pollaplos_group.** Στην ουσία αυτό το πεδίο δείχνει τον αριθμό λογαριασμού του ΜΠΑΜΠΑ πολλαπλού.

Τέλος δημιουργείτε το πεδίο **wemetrix_multiple_pollaplos.** Το πεδίο λαμβάνει έναν αριθμό group (group_number) από τον πίνακα *wemetrix.multiple_pollaploi.* Ο πίνακας δημιουργείται από το notebook που βρίσκεται στην τοποθεσία **/workspace/shared/wemetrix/blocking-and-matching/pollaploi_mpampades_fuzzy.** Πραγματοποιείται υλοποίηση fuzzy matching μεταξύ των μπαμπάδων ως προς την ονομασία τους και κοινοί μπαμπάδες λαμβάνουν έναν αριθμό. Αυτός ο αριθμός είναι και το group_number που θα αντιστοιχεί στο πεδίο wemetrix_multiple_pollaplos.

Να σημειωθεί ότι ο πίνακας *wemetrix.multiple_pollaploi* με τα αποτελέσματα της fuzzy matching διαδικασίας, υλοποιείται και γράφεται μία φορά καθώς δεν είναι συχνές οι αλλαγές στους πίνακες των μπαμπάδων.
Αν επιθυμείτε να κάνετε συχνές αλλαγές/διορθώσεις στους πίνακες των μπαμπάδων τότε θα μπορούσατε να αυτοματοποιήσετε την εκτέλεση του notebook **/workspace/shared/wemetrix/blocking-and-matching/pollaploi_mpampades_fuzzy** βάζοντας το στο pipeline εκτέλεσης των ροών (process_all_steps).
Ο πίνακας των μπαμπάδων προέρχεται από το παρακάτω query

```
df_mpampades = spark.sql(
  """
  SELECT
  a.VKONT , vbund, TAXNUM, name_org1, name_org2, name_org3,name_org4
  FROM fkkvk a
  left join fkkvkp b on a.VKONT = b.VKONT
  left join dfkkbptaxnum c on b.GPART = c.PARTNER
  left join but000 d on d.partner = c.partner
  where vktyp = '02'
  AND TAXTYPE = 'GR2'
  and a.vkont like '001%'
  """
)
```

, το οποίο μας δόθηκε έτοιμο. Επομένως, οπουδήποτε αλλαγή στους συμβεβλημένους πίνακες θα επηρεάσει τον πίνακα των μπαμπάδων και άρα το notebook **/workspace/shared/wemetrix/blocking-and-matching/pollaploi_mpampades_fuzzy** θα πρέπει να τρέξει εκ νέου για να υπολογιστούν τα νέα group_numbers.

Τα παρακάτω πεδία εξάγονται από την διαδικασία των πολλαπλών:

- "wemetrix_ar_pollaplou",
- "wemetrix_pollaplos_type",
- "wemetrix_anentaxtos_afm",
- "Customer_Type_Dimosio",
- "Customer_Type_Dimos",
- "Customer_Type_Idiotis",
- "LOG_SYMB_SAP_",
- "DIMOS_KOINOTITA_ERMI_",
- "KODIKOS_DIMOU_KALLIKRATI_",
- "DIMOS_KALLIKRATI_KLEISTHENI_",
- "wemetrix_ar_pollaplou_all_concat", -> για διαδικασία ελέγχων
- "wemetrix_pollaplos_type_all_concat", -> για διαδικασία ελέγχων
- "wemetrix_pollaplos_group",
- "wemetrix_multiple_pollaplos"

Τέλος να σημειωθούν τα παρακάτω:

Για τη κατηγορία Αλγόριθμοι (= records που δεν έχουν ar_pollaplou και δεν έχουν ΑΦΜ ίδιο με κάποιο από τους ΜΠΑΜΠΑΔΕΣ και ταιριάζουν με fuzzy matching στο λεκτικό) έγινε έρευνα για τις επιλογές υλοποίησης καθώς είναι τεχνικώς αδύνατον να πραγματοποιηθεί κατευθείαν το fuzzy matching των λεκτικών όλης της βάσης. Για την βέλτιστη εύρεση της

κατηγορίας αυτών των πολλαπλών προτείνεται η πρότερη κατηγοριοποίηση των λεκτικών βάση κοινών των company names και η σταδιακή εκτέλεση του fuzzy matching με διατήρηση των αποτελεσμάτων ώστε να ολοκληρωθεί εντός κάποιων ωρών και να επαναχρησιμοποιούνται σε μελλοντικά runs. Το λεκτικό που χρησιμοποιήθηκε είναι company name για εταιρίες, first/last name για Ιδιώτες.

Η διαδικασία των πολλαπλών μπήκε στο process_all_steps και γίνεται πριν το Blocking & Matching και τα group πολλαπλών εξαιρούνται απο το blocking and Matching.
Τα groups των πολλαπλών παίρνουν σαν GRid (wemetrix_GR_groupid) το CAid του ΜΠΑΜΠΑ.

Πολλαπλοί πολλαπλών:

- Οι ar_pollaplou συνδέονται με βάσει το πεδίο VKONT και Fuzzy matching με Levenstain distance 0.5 στα λεκτικά.
  (source: notebook /Workspace/Shared/wemetrix/blocking-and-matching/pollaploi_mpampades_fuzzy).
- Ο ar_pollaplou_pollaplou παίρνει τον ar_pollaplou του πολλαπλού (από όλους του πολλαπλούς με τους οποίους ταιριάζει) με τα περισσότερα records στο group του.
- Αναγνώριση πολλαπλών με λάθος ΑΦΜ.
- Λάθος αντιστοίχιση σε μη ενταγμένους. Έγινε σύγκριση με levenstein distance 0.8 στο PRCompanyName για την αναγνώριση των λάθος ΑΦΜ. Τα αποτέλεσμα χρειάζεται έλεγχο και επιβεβαίωση και πιθανά δοκιμές με διαφορετικό βάρος.