

DOI: <https://doi.org/10.32782/2524-0072/2021-34-102>

УДК 336.717:004.6

МОДЕЛЮВАННЯ ПРОЦЕСІВ АВТОМАТИЧНОЇ ЗВІРКИ ДАНИХ В ЕЛЕКТРОННИХ ПЛАТІЖНИХ СИСТЕМАХ НА ОСНОВІ МЕТОДІВ ОБРОБКИ ПРИРОДНОЇ МОВИ

MODELING OF AUTOMATIC DATA RECONCILIATION PROCESSES IN ELECTRONIC PAYMENT SYSTEMS BASED ON NATURAL LANGUAGE PROCESSING METHODS

Сідельов Павло Анатолійович

аспірант

ДВНЗ «Приазовський державний технічний університет»

ORCID: <https://orcid.org/0000-0001-5672-8189>**Sidelov Pavlo**State Higher Education Institution
«Pryazovskyi State Technical University»

Стаття присвячена актуальним проблемам звірки даних в електронних платіжних системах. З поширенням електронних платіжних систем в рамках концепції open banking зросли кількість транзакцій, різноманітність платіжних систем і відповідно складність звірки платіжних даних. Це збільшує частку платежів, які доводиться звіряти вручну, та відповідно витрати на платіжну інфраструктуру. В роботі проаналізовано генезис існуючих підходів до звірки платіжних даних. Досліджено основні етапи звірки, відмінності між бізнес-транзакціями та технічними, основні операції та сценарії процесу звірки даних. Розроблено модель процесу звірки в компанії, що є провайдером платіжних сервісів (ППС). Розроблено модель сервісу автоматичної звірки платіжних даних та схему його інтеграції до інформаційної системи ППС. Виокремлено проблеми автоматичної звірки даних. Окреслено шляхи їх вирішення через застосування методів розпізнавання природної мови.

Ключові слова: звірка, платіжні системи, мета-модель, розпізнавання природної мови, інтеграція.

The article is devoted to the actual problems of data reconciliation in electronic payment systems. The aim of the work is to develop a model for the process of data reconciliation in electronic payment systems based on natural language processing methods and ensure its integration into payment transaction processing systems. With the spread of electronic payment systems within the concept of Open Banking, the number of transactions, the variety of payment systems, and, accordingly, the complexity of reconciling payment data has increased dramatically. This is due to the fact that the data come from different sources and have a heterogeneous structure that is difficult to recover by formal methods. This increases the share of payments that have to be reconciled manually, which, in turn, leads to an increase in the costs of operating the payment system. Therefore, at present, research on the automation of reconciliation processes, based on the use of artificial intelligence methods, in particular, natural language processing, is becoming relevant. The existing approaches to a reconciliation of payment data are systematized and their genesis is analyzed. It turned out that in recent years the number of publications on reconciliation methods based on the analysis of big data and the use of semantic models has been growing, but a number of issues on building payment data reconciliation systems have not yet been disclosed. This concerns the integration of automatic reconciliation services into payment transaction processing systems. The main stages of payment data reconciliation, the differences between business and technical transactions, the main operations of the data reconciliation process, and reconciliation scenarios have been explored. A model of the reconciliation process in a company that is a Payment Service Provider (PSP) has been developed. A model for the operation of the service for automatic reconciliation of payment data and a scheme for its integration into the PSP information system has been developed. The problems of automatic data reconciliation are highlighted, namely: discrepancy between data exchange formats; a discrepancy in field names; data structure discrepancy; error in the content of the fields; different accuracy of numerical values; temporary gaps; splitting transaction amounts. The ways of solving these problems through the use of natural language recognition methods are outlined.

Keywords: reconciliation, payment system, meta-model, natural language processing, integration.

Постановка проблеми. Сучасна фінансова система має надзвичайно складну структуру, що складається з окремих автономних елементів і їх груп. Елементами фінансової системи можуть бути банки, Біржі, платіжні організації, підприємства, окремі торгові точки, агрегатори послуг, державні органи і багато інших організацій, які так чи інакше пов'язані з рухом грошових потоків.

Будь-який рух безготівкових грошових коштів, які станом на 2021 рік становлять понад 90% від загальної суми платежів в Україні [1] фактично являє собою, перш за все, рух інформації між учасниками платіжної системи. Наприклад, при забезпеченні транзакцій з використанням банківських платіжних карт тільки близько 15% операцій можна назвати «грошовими». Решта «інформаційні» [2].

Дані про здійснені фінансові транзакції в даний час мають самостійну цінність, як об'єкт дослідження. Аналіз таких даних дозволяє виявляти різні закономірності у фінансовій поведінці учасників платіжного процесу, що дає можливість зокрема поліпшити управління фінансовими потоками і ризиками, передбачати обсяги платежів та їх динаміку, запобігати платіжному шахрайству [3; 4; 5].

Однак, крім даних для складання подальшого аналізу, слід брати до уваги те, що ці дані мають мало цінності, якщо не досягається їх узгодженість (consistency) з іншими джерелами даних. Жодна система не працює в інформаційному вакуумі крім навмисне створених систем закритого типу, але навіть такі системи повинні мати інструментарій самоперевірки, щоб бути впевненим, що ніде немає витоків або помилок в розрахунках. У реальному ж світі всі фінансові організації якимось чином взаємодіють між собою і обмінюються даними. Отже, потрібно механізм, які дозволить бути впевненим що обидві сторони ведуть облік вірно, не втрачають транзакції, баланси сходяться і статуси транзакцій також синхронізовані. Основною функцією такого механізму є звірка даних, і він є обов'язковою частиною будь-якої безготівкової платіжної системи [2; 6; 7].

Процеси звірки відбуваються прозоро для кінцевих користувачів платіжної системи. З результатами звірки вони стикаються досить рідко і тільки побічно – в тих випадках, коли звірка показує розбіжність в даних і необхідне коригування транзакцій. При цьому між фінансовими організаціями постійно відбувається велика кількість звірок, перш ніж дані відправляються в постійне зберігання.

З поширенням електронних платіжних систем в рамках концепції open banking, кількість транзакцій, різноманітність платіжних систем і відповідно складність звірки платіжних даних різко зросли. Це обумовлено тим, що дані надходять з різних джерел і мають неоднорідну структуру, яку важко відновити формальними методами. Саме остання обставина ускладнює використання існуючих механізмів звірки платіжних даних. Це збільшує частку платежів, які доводиться звіряти вручну, що в свою чергу призводить до збільшення витрат на функціонування платіжної системи. Тому в даний час актуальними стають дослідження з автоматизації процесів звірки, на підставі залучення методів штучного інтелекту, зокрема – обробки природної мови.

Аналіз останніх досліджень і публікацій. Задача звірки даних історично є однією з перших фінансових задач, для вирішення якої була використана комп'ютерна техніка. Програма для звірки платежів за чеками була впроваджена казначейством США в 1958 році і дозволила на 50% скоротити персонал, який займався обробкою платежів, при тому, що обсяги транзакцій за період її використання зросли на 60% [8]. Аналогічні рішення застосовувалися також для обслуговування банківських кореспондентських рахунків і розрахунків за платіжними картками [6]. Проте, до появи в другій половині 1990-х років систем електронних платежів, алгоритми звірки залишалися досить простими, оскільки уніфікація платіжних повідомлень забезпечувалася на рівні джерел.

Поява і розвиток електронних платежів актуалізує розробку ефективних алгоритмів звірки платіжних даних, що супроводжується збільшенням кількості публікацій на цю тематику. У ранніх публікаціях розкриваються роль і базові технології звірки даних в електронних платіжних системах [9]. Пізніше, зважаючи на зростаючу актуальність проблем звірки, з'являються спеціалізовані публікації, присвячені окремим аспектам і реалізації звірки в платіжних системах різного типу. Так, в [10] автори розглядають реалізацію звірки, засновану на технологіях блокчейн. У роботах [11; 12] розглядаються практичні реалізації технологій розумної (Smart) звірки в сучасних інформаційних системах і робиться висновок про перспективність і зручність використання цих технологій.

Також слід зазначити роботи, присвячені використанню аналогічних технологій в інших областях, де також потрібна звірка даних.

Зокрема, у роботі [13] досліджуються необхідність та особливості звірки даних у бухгалтерському та податковому обліку. У роботі [14] розглядаються питання звірки на рівні планування державного бюджету.

В останні кілька років також з'являються публікації, присвячені методам звірки на основі аналізу великих даних [15] і використання семантичних моделей [16]. Автори останньої статті послідовно аргументують необхідність використання саме таких моделей в рамках реалізації концепції Open Banking.

Разом з тим, слід зазначити, що тема моделювання процесів звірки даних в електронних платіжних системах на основі методів обробки природної мови тільки починає набирати актуальність, зважаючи на те, що реалізація концепції Open Banking тільки порівняно недавно отримала потужну законодавчу підтримку у вигляді директиви PSD2 [17]. Ряд питань щодо побудови систем звірки платіжних даних досі не розкритий, зокрема це стосується питань інтеграції сервісів автоматичної звірки до систем обробки платіжних транзакцій, витрат на супроводження таких систем тощо.

Таким чином, метою роботи є розробка моделі автоматичної звірки даних в електронних платіжних системах на основі методів обробки природної мови та забезпечення її інтеграції до систем обробки платіжних транзакцій.

Виклад основного матеріалу дослідження. Фінансові організації починають проводити звірки (reconciliation) після початку бізнес діяльності один з одним. Звірки є ключовим процесом, які породжують документи що підтверджують фактичну діяльність і фінансові відносини між організаціями. Звірка – це покроковий аналіз всіх рухів і операцій за рахунком (рахунками) клієнтів. Звірки можуть бути як внутрішніми так і зовнішніми.

Внутрішні звірки, мають на увазі собою певні дії, спрямовані на те, щоб, багаторазово (мінімум з двох джерел) переконатися в тому, що, облік фінансових транзакцій всередині компанії ведеться вірно. Наприклад, якщо всередині організації ведеться облік в одній таблиці з округленням до двох знаків після коми, а в інший до чотирьох знаків після коми, то очевидно, що з часом може виявитися розрив балансів навіть всередині однієї організації і однієї облікової системи.

Зовнішні звірки – це зіставлення транзакцій з власних баз даних із зовнішніми джерелами інформації: звірочних файлів, баз даних, API інтерфейсів.

Для того, щоб детально розібрати наявні способи звірок, необхідно позначити складові частини транзакції, та її життєвий цикл.

Як правило, транзакції несуть в собі наступний типовий набір супровідних даних: Client_ID, Organization_ID, Terminal_ID, Transaction_ID, Date, Time, Amount, Currency, Fee, MCC_Code, Card_Type і т.д. Досить часто в звірочувальні файли потрапляє далеко не вся інформація, яку організація зберігає, а тільки мінімально достатня.

Транзакції у фінансових системах умовно можна розділити на два види:

Бізнес-транзакції – відповідають рівню банківської операції і являють собою високорівневі дані які необхідні для фінансового обліку. Дані бізнес-транзакцій використовуються операційним менеджментом для організації виплат та ініціації розслідувань з причин розбіжностей.

Технічні транзакції – операції, які пов'язані з бізнес-транзакціями і виконуються в інформаційній системі. Дані з технічних транзакцій використовуються фахівцями для пошуку причин відмов і невідповідностей при звірках. Одна бізнес транзакція може складатися з великої кількості технічних.

Приклад бізнес транзакції:

Клієнт А, з рахунку В, перевів суму Х у валюті Y на рахунок Z.

Приклад відповідної технічної транзакції:

1. Заблокувати операції над рахунком В поки виконується транзакція.
2. Встановити статус транзакції в Pending.
3. Перевірити баланс на рахунок В.
4. Якщо сума на рахунку В достатня, то перевірити рахунок Z на предмет можливих обмежень або блокувань.
5. Заблокувати інші операції крім поточної над рахунком Z.
6. Списати кошти з рахунку В на рахунок Z.
7. Провести необхідні записи в історії транзакцій по обох рахунках.
8. Встановити статус транзакції в Success.
9. Розблокувати рахунок В.
10. Розблокувати рахунок Z.

До завершення бізнес-транзакції практично кожна технічна транзакція може бути скасована (виконаний відкат, або rollback). Якщо на будь-якому з кроків технічних транзакцій виникають проблеми або відмова-транзакція отримує статус Fail і виконання закінчується. У підсумку – баланси по рахунках не зміняться, але з'явиться бізнес-транзакція з статусом Fail і посиланням на технічну транзакцію на якій бізнес-транзакція переривається.

Нижче представлені основні статуси транзакцій в облікових системах (найменування в різних системах і організаціях може відрізнятися).

Init – транзакція ініційована і пішла в обробку.

Pending – транзакція оброблена і чекає свого завершення.

Capture – кошти по транзакції списані з рахунку платника.

Reconciliation – транзакція пройшла процедуру звірки, зазвичай приймає значення Success / Fail.

Після успішного завершення звірки, в залежності від діяльності фінансової організації можуть бути зроблені додаткові кроки по транзакціях:

Settlement – розрахунок значень, наприклад збір комісій з прийнятих платежів.

Payout – виплата зібраних платежів на банківський рахунок торговця.

Розглянемо сутність проблеми звірки і причини її виникнення.

Як відомо, всі фінансові операції ґрунтуються на понятті рахунку. Транзакцією називається така операція над рахунком, яка мутує його стан (змінює рахунок в більшу або меншу сторону). Різні клієнти генерують різну кількість транзакцій, яке може змінюватися від одиниць (для приватних осіб) до сотень мільйонів штук (для банків і великих онлайн магазинів) в день.

Запис про кожну транзакцію включає певний набір полів і параметрів що складають «тіло» транзакції, статус транзакції, «тимчасові позначки» (timestamp) коли транзакція змінювала свій статус, та іншу додаткову інформацію.

Оскільки платіжні операції найчастіше зачіпають контрагентів, які не ведуть загальний баланс (наприклад – компанія А і банк Б), платіжний сервіс кожного з них породжує власну історію транзакцій.

Історії транзакцій компанії А і банку Б можуть не збігатися з безлічі технічних або організаційних причин. Сенс звірок полягає саме в тому, щоб переконатися, що система знаходиться в збалансованому стані і немає ніяких розбіжностей або невідповідностей між історіями транзакцій зацікавлених сторін. Якщо невідповідності знаходяться – робочі групи обох сторін повинні знайти своє пояснення для конкретного випадку (наприклад, технічний збій або помилка оператора) і зробити конкретні кроки щодо усунення наслідків і, при можливості, недопущення подібної ситуації в майбутньому.

Наприклад, клієнт ініціював процедуру покупки в магазині і платіжний сервіс А ініціював транзакцію по картковому рахунку. Банкеквайер В заблокував гроші на картці платника, але через розрив зв'язку платіжний сервіс А не отримав про це повідомлення і відповідно не завершив транзакцію на своєму боці.

При проведенні звірки обидві сторони побачать розбіжність в балансах і зможуть виявити проблемну транзакцію, а також закінчити її або навпаки – скасувати.

Таким чином, *звірка* (reconciliation) – це механізм зіставлення наявної історії транзакцій із зовнішнім джерелом звірки, наданим контрагентом. Звірка передбачає проведення ряду операцій: *фетчинг* (fetching), *парсинг* (parsing), *маппінг* (mapping) і *матчинг* (matching).

Фетчинг – процес отримання даних, які будуть використовуватися для звірки з будь-якого джерела. Дані можуть потрапляти в систему різними шляхами-по електронній пошті, через файл-сервер, через окремий сервіс, доступний за допомогою API.

Парсинг – процес обробки вмісту для приведення у відповідність з внутрішньою структурою даних перевіряючої системи. Проводиться розбір отриманих даних, можливі конвертації або пост-процесінг отриманої структури даних для приведення у відповідність з внутрішньою структурою перевіряючого.

В процесі парсингу виконується послідовний аналіз кожного запису на пошук відповідних ключових слів або значень, які конвертуються у внутрішній формат даних перевіряючої системи. Наприклад поле *trx_id* в таблиці зовнішнього джерела даних приводиться в стан *Transaction_ID* і зберігається в конвертованому вигляді на перевіряючій стороні.

Маппінг – накладання перетворених на попередній стадії даних із зовнішнього джерела на дані що містяться у внутрішній базі. При цьому на кожну транзакцію застосовується процедура *матчинг*.

Матчинг – при збігу вмісту, транзакція відзначається як перевірена. Але при виявленні розбіжностей – спрацьовує обробка виняткової ситуації. Транзакції, вміст яких не співпало, потрапляють в спеціальний звіт по звірці для подальшої обробки. Під обробкою можна розуміти ручне втручання оператора або запуск автоматичного сценарію, який в залежності від коду помилки виконає ту чи іншу стратегію вирішення проблеми. В результаті *матчинг* породжує один або два звіти. Один якщо всі транзакції збіглися і два, якщо є розбіжності.

Звірки можуть бути як пакетні так і транзакційні.

У пакетному сценарії, це може бути пакет транзакцій за певний період (наприклад – 24 години) і завдання компанії полягає в тому, щоб звірити всі транзакції, фінальні суми по ним і провести взаєморозрахунки за конкретний день.

У транзакційному сценарії, кожна транзакція і її звітність можуть бути самостійним і автономним процесом, що не впливає на звірки інших транзакцій.

Проблематика звірок

З масовим виходом на ринок FinTech компаній і фінансових сервісів, альтернативних банківським, звітність платіжних даних почала перетворюватися на серйозну проблему. Кількість електронних транзакцій і кількість учасників в ланцюжку зростає багаторазово, при цьому методологія звірок істотно не змінилася.

Відсутність єдиної стратегії при звірці даних, що зберігаються в масі розподілених систем, систематично споживає відчутні ресурси всіх учасників ланцюжків транзакцій, що прямо впливає на доходи компаній учасників. Іноді десятки бухгалтерів і технічних фахівців в кожному конкретному банку або фінансовій організації займаються тільки тим, що по шматочках збирають дані з різних джерел, «склеюють» їх, проводять звірки і при виявленні розбіжностей так само болісно шукають причину розбіжностей.

Кожна компанія, по своєму визначає власні структури і формат даних, найчастіше сліпо покладаючись на пропозиції технічних фахівців, не особливо вникаючи до суті довгострокової стратегії роботи з даними. Незважаючи

на наявність стандартів ISO по роботі з даними, рішення на їх основі часто виявляються занадто дорогі в придбанні і обслуговуванні, особливо для невеликих компаній.

Для фінансової організації, що надає платіжний сервіс, затримки в звітках можуть викликати несвоєчасні виплати ланцюжку одержувачів, наприклад торговцям або мерчантам (Merchant). Далі ці затримки поширюються по ланцюжку на всіх учасників платіжного процесу. У Європейському Союзі, затримки виплат (Late Payments) іноді може досягати до 70 бізнес днів на рік. Ця проблема є настільки гострою, що робить істотний вплив навіть на ВВП.

Розглянемо можливі підходи до автоматизації звірки платіжних даних на прикладі роботи двох компаній.

На рис. 1, відображений потік даних умовної фінансової компанії, що працює як PSP (Payment Service Provider) і умовної платіжної системи.

Розглянемо роботу моделі, що показана на рис. 1.

1. Оператор або сервіс Компанії 1 (платіжна система), робить експорт даних в будь-якому форматі, наприклад CSV (Comma Separated Value) файл.
2. Дані імпортує в систему попередньої обробки Компанії 2 (PSP).
3. Дані перетворюються у внутрішній формат Компанії 2.
4. Дані розбираються по транзакціях і складаються в таблицю звірки для Компанії 1 в базі даних Компанії 2.
5. Дані Компанії 1 звіряються з наявними даними в таблицях Компанії 2.

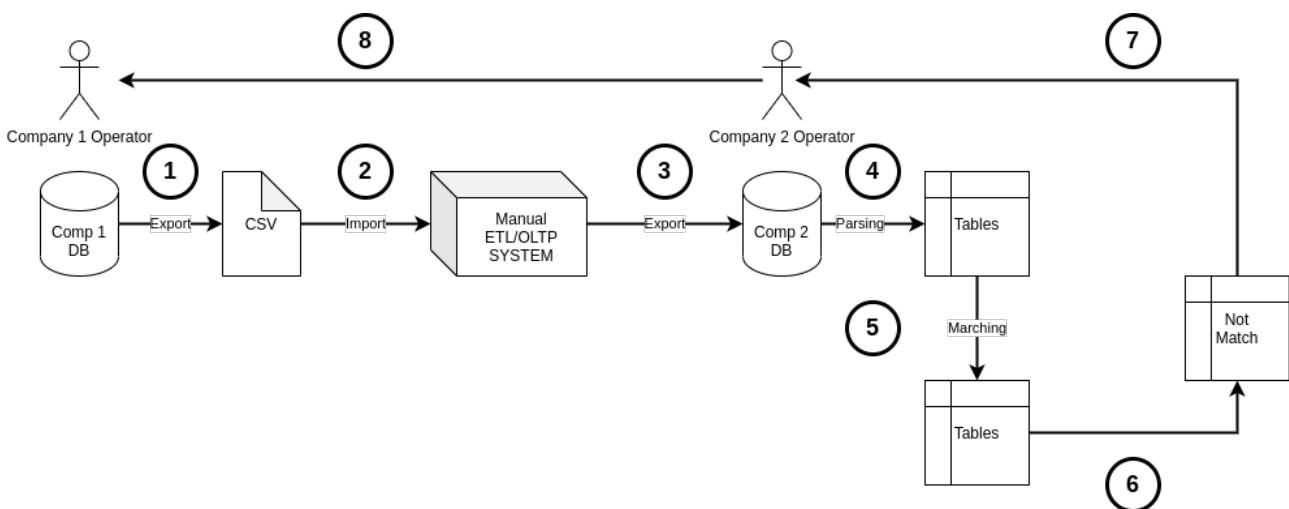


Рис. 1. Модель процесу звірки транзакцій PSP прийнятих від мерчантів і даних картової платіжної системи

Джерело: розроблено автором

6. Дані, які не пройшли звірку зберігаються в спеціальній таблиці «винятків», інші транзакції змінюють свій статус на «звірено» і готові до виплат.

7. Дані про транзакції які потрапили в виключення якимось чином доставляються Оператору Компанії 2, наприклад по e-mail.

8. Компанія 2 повідомляє Оператора (або менеджера) Компанії 1 про виявлені розбіжності.

Після кроку 8 відбувається той чи інший процес або процедура реакції на виявлені «виключення». Він може займати досить тривалий проміжок часу і вимагати значних витрат людських ресурсів.

Слід зазначити, що наведена на рис. 1 схема, відповідає реальним процесам обробки платежів. Так, відповідний функціонал має Сервіс eCommerce компанії Mastercard.

Автоматизація даного процесу дозволить прискорити процес звірок між компаніями і скоротити ресурси, які йдуть на організацію проміжних кроків.

На рис. 2 представлена схема звірки, в якій введено зворотний зв'язок між учасниками через впровадження сервісу системи автоматичної звірки (САЗ), що дозволяє отримати

той же результат з меншими витратами часу і людських ресурсів.

Розглянемо модель, показану на рис. 2:

1. Оператор або сервіс, Компаній 1-2, виробляє експорт даних в будь-якому форматі.

2. Дані завантажуються в будь-яке проміжне сховище.

3. Дані автоматично завантажуються до системи попередньої обробки

4. Дані проходять приведення до того чи іншого стандарту згідно виявленої класифікації.

5. Дані завантажуються в проміжну базу даних, яка виконує роль Мета моделі всіх можливих типів даних і їх шаблонів перетворень.

6. Нові дані потрапляють в сервіс звірки (reconciliation), де очікують документа з подібним набором даних від Компанія 1 і Компанія 2.

7. При виявленні подібних наборів складається уніфікований документ по звірці в єдиному форматі даних ISO 20022.

8. Доступ до документів можливий з Особистих кабінетів операторів Компаній 1-2.

9. Оператори Компаній 1-2 завжди мають доступ до всієї історії раніше звірених документів.

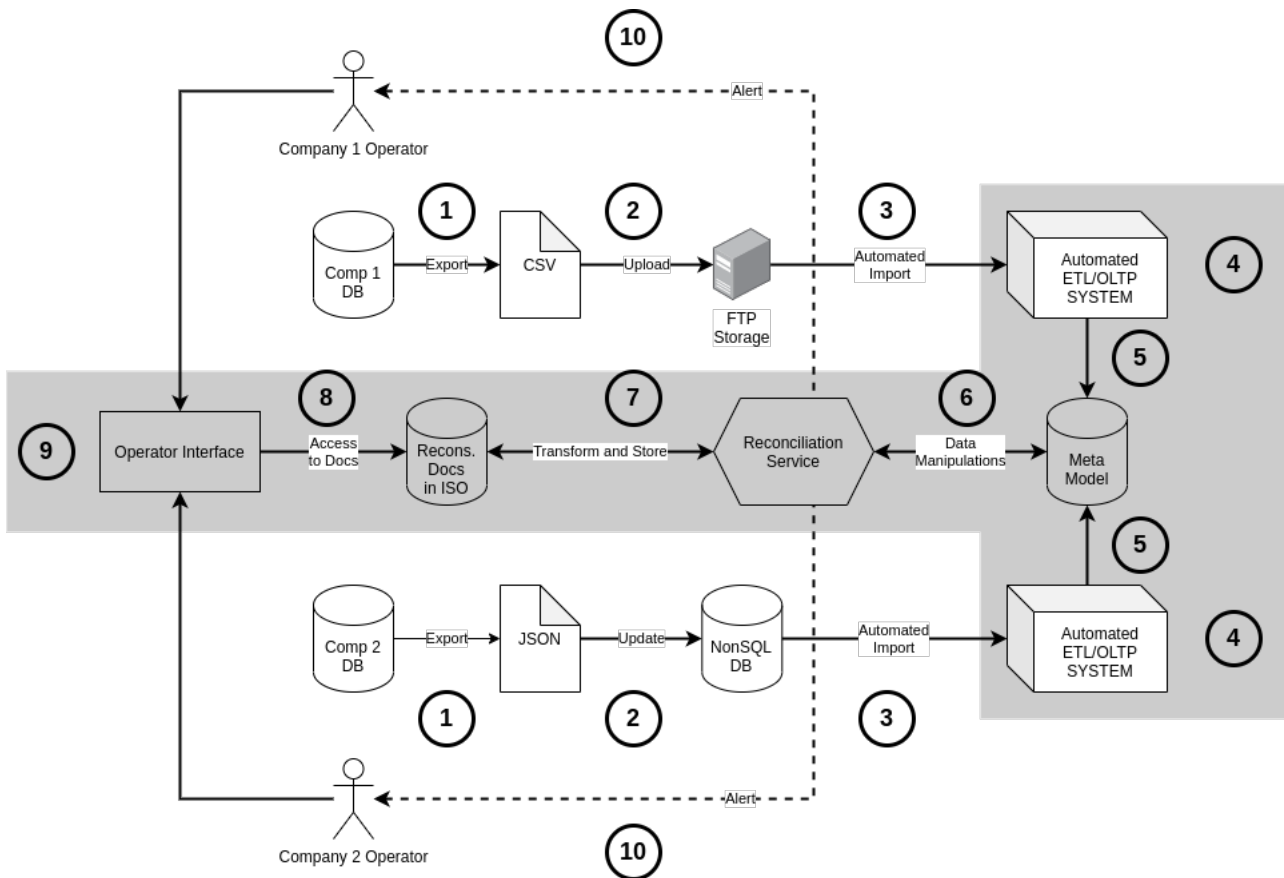


Рис. 2. Інтеграція сервісу САЗ в модель звірки транзакцій

Джерело: розроблено автором

10. У разі виявлення «винятків» провести повідомлення операторів Компанії 1-2.

Якщо порівнювати зі схемою, наведеною на рис. 1, то збіги можна знайти тільки на першому етапі. Далі робота механізмів звірки принципово відрізняється.

У першому випадку (див. рис. 1) Компанія 1 і Компанія 2 мають власні реалізації механізмів звірок що вимагає істотних витрат обома компаніями на реалізацію подібного процесу. Далі результати звірок між компаніями повинні бути оброблені оператором-співробітником відповідальним за звірки. Результати звірок якимось чином передаються оператору-контрагенту і починається процес розбору інцидентів – транзакцій випали зі звірки. Всі перераховані вище дії виконуються в ручному режимі, з істотними часовими затримками і фактично є пасивною реакцією на проблеми, відкладеною в часі.

У другому випадку (рис. 2) САЗ, в проактивній формі, виявить невідповідності і проінформує обидві сторони негайно. Проактивне

повідомлення сторін про виявлені невідповідності дозволить істотно скоротити час реакції на інциденти і мінімізувати потенційні ризики. Ризиками в цьому випадку можна вважати втрату коштів через можливі збої або втручання в роботу систем. Чим довше час реакції на інцидент, тим вище ризики фінансових, ресурсних і репутаційних втрат.

З точки зору практичного використання, великою перевагою моделі, показаної на рис. 2, є те, що САЗ є окремим модулем (сервісом). В рамках інформаційної системи САЗ працює як «чорний ящик», а робота з ним здійснюється через стандартизовані системні процедури. Це дозволяє організовувати обслуговування і оновлення цього модуля, зміну його функціональності без перерви в роботі платіжних сервісів.

Далі розглянемо окремі аспекти реалізації САЗ в рамках моделі, наведеної на рис. 2.

В рамках традиційного підходу до алгоритмічного забезпечення процесів звірки даних,

Таблиця 1

Основні проблеми звірки даних в платіжних системах

| № п/п | Проблеми при звірці даних: | Вирішено / не вирішено |
|-------|--|--|
| 1. | Розбіжність форматів обміну даними (одні сервіси можуть виробляти дані в форматі XML, інші в JSON або CSV) | Вирішено. Існують конвертери одного формату в інший, що працюють з усіма популярними форматами даних. |
| 2. | Розбіжність найменувань полів (Номер чека / payment_id) | Частково вирішена введенням парсингу. Якщо ж заголовки таблиць і ключових полів відрізняються від заголовків, заданих при налаштуванні парсингу, то процес звірки зупиняється, а транзакції направляються для ручного розбору. |
| 3. | Розбіжність структури даних (в одному наборі кілька полів об'єднані в одне, або якісь поля відсутні) | Не вирішено. Транзакції направляються для ручного розбору. |
| 4. | Помилка в змісті полів | Не вирішено. Помилки можуть бути на рівні Ім'я, Прізвище, та інші, що може створювати юридичні складнощі при виплаті переказів. Числова інформація (номери рахунків, карт і т.п.) зазвичай має контрольну суму, що зменшує проблеми. |
| 5. | Різна точність числових значень (до 2 або 4 знаків після коми) | Частково вирішено. Процедуру звірки можна налаштувати так щоб вона враховувала тільки 2 знака після коми, якщо там 4 знака, але тоді може бути помилковою фінальна сума звірки. |
| 6. | Дані, які звіряються, відображають стан об'єктів звірки на момент вивантаження даних і не завжди відображають реальну ситуацію | Частково вирішено. Вимагає обробки в ручному режимі (повторне вивантаження даних для звірки). |
| 7. | Дроблення сум транзакцій (сплата одного рахунку кількома платежами та навпаки) | Не вирішено. Потребує пошуку відповідних транзакцій в ручному режимі. |

Таблиця 2

Приклад даних для звірки, Сторона 1

| Trx_ID | Company | Amt | Itm | Dscr |
|--------|--------------|-------|-----|-------------|
| 123 | Green Energy | \$100 | 10 | Solar panel |
| 124 | Dark energy | €200 | 20 | Battery |

Таблиця 3

Приклад даних для звірки, Сторона 2

| Transaction ID | Company | Amount | Currency | Items | Details |
|----------------|-------------------|--------|----------|-------|-------------|
| 123 | Green Energy Ltd. | 100 | USD | 10 | Solar panel |
| 124 | Dark Energy | 200 | EUR | 20 | Battery |

пошук відповідності між транзакціями ведеться або по чіткій відповідності, або по заданій базі підстановок. Поряд з перевагами (передбачуваність, хороша опрацювання алгоритмів, велика кількість реалізацій) такий підхід пов'язаний також з рядом недоліків (табл. 1).

Розглянемо приклад платіжних даних і проблемами, які можуть виникати при їх звірці.

В табл. 2 і табл. 3 наведено приклади даних для звірки.

Як видно з порівняння табл. 2 і табл. 3, при їх звірці система зіткнеться з наступними проблемами (нумерація проблем дана по табл. 1):

Проблема 2. Розбіжність найменувань полів (Trx_ID vs Transaction ID; Amt vs Amount; Dscr vs Details і т. д.).

Проблема 3. Розбіжність структури даних (в табл.2 в поле Amt присутні і валюта і сума транзакції, а в табл. 3 вони рознесені по різних полях).

Дані проблеми є досить поширеними, і можуть бути вирішені алгоритмічно на етапі парсингу даних. Так, при виникненні проблеми 2 (не знайдено поле з відповідною назвою), програма звірки може виконати пошук відповідностей за наявними колонками і якщо значення осередків збігаються, припустити, що в цьому полі відбулася зміна назви.

Проблема 4. Помилка в змісті полів (з точки зору комп'ютерної системи обробки, значення в Company табл. 2 і табл. 3 є різними).

Дана проблема практично не вирішується алгоритмічно. Тому транзакції, в яких не збігаються поля, що перевіряються, потрапляють в винятки і обробляються вручну.

Розглянемо спосіб вирішення даної проблеми, заснований на методах розпізнавання природної мови.

Слід зазначити, що в даному випадку вхідна інформація вже знаходиться у текстовій, або числовій формі, тобто не потребується додаткових дій для розпізнавання аудіосигналів, чи друкованого тексту.

Серед багатьох методів, що були розроблені для розпізнавання природної мови звернемо увагу на такі, як проблема синонімів та пошук подібностей.

Проблема синонімів виникає в тому випадку, коли одному поняттю може відповідати кілька різних лексичних одиниць. Так, в прикладі з табл. 2, 3 це «Dscr» та «Details»; «\$» та «USD».

Подібностями будемо вважати лексичні одиниці, які є різними варіантами одне одного. В тому ж прикладі це «Green Energy Ltd.» та «Green Energy»; «Trx_ID» та «Transaction ID» та інші.

В загальному вигляді визначення синонімів та подібностей – це різні задачі. Перша традиційно вирішується із допомогою словника, який необхідно постійно оновлювати. Друга вирішується із допомогою алгоритмів пошуку математичної міри відстані між лексичними одиницями.

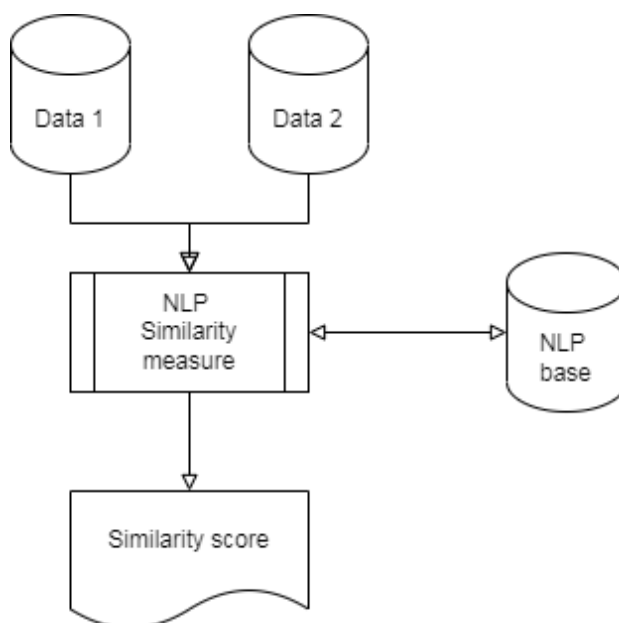


Рис. 3. Визначення подібності між даними в процесі звірки

Джерело: розроблено автором

На рис. 3 показано блок-схему визначення подібності між даними в процесі звірки.

Як видно з рис. 3, центральним ланцюгом в даній схемі є процес визначення подібності. Наразі існує багато алгоритмів для опрацювання цього завдання – від суто статистичного k-means та його варіацій до алгоритмів з використанням нечіткої логіки та штучного інтелекту.

У процесі звірки відбувається пошук подібностей не тільки між даними, що звіряються але й між ними та базою даних NLP, що дозволяє відшукувати у вхідних даних синоніми. Якщо в процесі роботи системи знаходиться новий синонім, він доповнює базу.

Таким чином забезпечується здатність системи автоматичної звірки до самонавчання, що дозволяє підвищити її адаптованість, точність роботи та знизити витрати на супроводження.

Висновки. В роботі було вирішено актуальне науково-практичне завдання з розробки моделі автоматичної звірки даних в електронних платіжних системах на основі методів обробки природної мови та забезпечення її інтеграції до систем обробки платіжних транзакцій.

Аналіз генезису підходів до звірки платіжних даних показав, що наразі ускладнення платіжних відносин потребує використання

більш складних методів і для звірки даних. Крім того практичну значущість наукових доробок обумовлює не лише їх ефективність щодо розв'язання основної задачі (звірки даних) а й ефективність інтеграції до існуючих інформаційних систем.

Систематизація понятійного апарату та аналіз основних етапів і сценаріїв звірки даних дозволив виділити основні проблеми цього процесу та запропонувати базову модель процесу звірки транзакцій в компанії, що є провайдером платіжних сервісів.

Розроблено модель сервісу автоматичної звірки платіжних даних та схему його інтеграції до інформаційної системи платіжного сервісу. Виокремлено проблеми автоматичної звірки даних, а саме: розбіжність форматів обміну даними; розбіжність найменувань полів; розбіжність структури даних; помилка в змісті полів; різна точність числових значень; часові розриви; дроблення сум транзакцій.

Розроблено блок-схему розв'язання задачі визначення подібності між даними в процесі звірки із застосуванням методів розпізнавання природної мови. Її використання забезпечує здатність системи автоматичної звірки до самонавчання та дозволяє підвищити її адаптованість, точність роботи та знизити витрати на супроводження

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

1. Національний банк України. Безготівкові розрахунки. URL: <https://bank.gov.ua/ua/payments/nocash>
2. Sidelov P. *The World of Digital Payments*. Kyiv : Publisher Ostap Khanko, 2017. 304 p.
3. Bahrami M., Bozkaya B., Balcisoy S. Using behavioral analytics to predict customer invoice payment. *Big data*. 2020. Т. 8. №. 1. С. 25–37.
4. Kolodiziev O., Mints A., Sidelov P., Pleskun I., Lozynska O. Automatic Machine Learning Algorithms for Fraud Detection in Digital Payment Systems. *Eastern-European Journal of Enterprise Technologies*. 2020. № 5(9 (107)), pp. 14–26. DOI: <https://doi.org/10.15587/1729-4061.2020.212830>
5. Mathuva D. M. The Fading role of bank reconciliation in fraud prevention and detection. *The accountant*. 2016. № 2. P. 32–33.
6. Stott J. R. *Mastering Principles of Accounts*. Palgrave, London, 1982. 248 p.
7. Adala A. Reconciliation of Electronic Remittances: A Multi-Level Approach // *Proceedings of the 11th International Conference on Theory and Practice of Electronic Governance*. 2018. P. 659–661.
8. Stickney G. F. The check payment and reconciliation program of the US Treasury: present status and future prospects. *Fall joint computer conference*. 1966. P. 479–499.
9. Furche A., Wrightson G. Computer Money: a systematic overview of electronic payment systems. *Verlag fur digitale Technologie GmbH*. 1996. 107 p.
10. Zhu X., Wang D. Application of blockchain in document certification, asset trading and payment reconciliation. *Journal of Physics: Conference Series*. – IOP Publishing. 2019. Vol. 1187. № 5. P. 52–80.
11. Sunarya P. A., Nurhaeni T., Haris H. Bank Reconciliation Process Efficiency Using Online Web Based Accounting System 2.0 in Companies. *APTISI Transactions on Management (ATM)*. 2017. Vol. 1. № 2. P. 124–129.
12. Rahardja U. et al. Financial management system integrated by web-based payment cash link solution to invent smart reconciliation. *International conference on industrial engineering and operations management*. 2021. P. 4733–4743.
13. Răscolean I., Rakos I. S., Calotă T. O. Analysis of the reconciliation of the accounting result with the fiscal result. case study: *Annals of Constantin Brancusi'University of Targu-Jiu. Economy Series*. 2015. P. 297–304.

14. Richardson J., Outlaw J. Implications of Budget Reconciliation for Commodity Programs. *Choices*. 2004. Vol. 19. № 4. P. 43–45.
15. Xing X. Financial Big Data Reconciliation Method. *2021 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE)*. IEEE. 2021. P. 260–263.
16. Paneque M., Roldán-García M. M., García-Nieto J. A Semantic Model for Enhancing Data-Driven Open Banking Services. SSRN preprints. № 4151598. 11 p.
17. Минц А., Сиделев П. Анализ глобального уровня готовности банковской системы к имплементации концепции «Open Banking». *Вісник Приазовського державного технічного університету. Серія : Економічні науки*. 2019. № 37. С. 57–63.

REFERENCES:

1. National Bank of Ukraine. *Cashless Payments*. Available at: <https://bank.gov.ua/ua/payments/nocash>
2. Sidelov P. (2017) *The World of Digital Payments*. Kyiv, 334 p.
3. Bahrami, M., Bozkaya, B., & Balcisoy, S. (2020) Using behavioral analytics to predict customer invoice payment. *Big data*, 8(1), 25–37.
4. Kolodiziev O., Mints A., Sidelov P., Pleskun I., Lozynska O. (2020) Automatic Machine Learning Algorithms for Fraud Detection in Digital Payment Systems. *Eastern-European Journal of Enterprise Technologies*, 5 (9 (107)), pp. 14–26. DOI: <https://doi.org/10.15587/1729-4061.2020.212830>
5. Mathuva, D. M. (2016) The Fading role of bank reconciliation in fraud prevention and detection. *The accountant*, no. 2, pp. 32–33.
6. Stott, J. R. (1982) *Mastering Principles of Accounts*. Palgrave, London, 248 p.
7. Adala, A. (2018, April). Reconciliation of Electronic Remittances: A Multi-Level Approach. *In Proceedings of the 11th International Conference on Theory and Practice of Electronic Governance*, pp. 659–661.
8. Stickney, G. F. (1966) The check payment and reconciliation program of the US Treasury: present status and future prospects. *In Proceedings of the November 7-10, 1966, fall joint computer conference*, pp. 479–499.
9. Furche, A., & Wrightson, G. (1996) *Computer Money: a systematic overview of electronic payment systems*. Verlag fur digitale Technologie GmbH, 107 p.
10. Zhu, X., & Wang, D. (2019, April) Application of blockchain in document certification, asset trading and payment reconciliation. *Journal of Physics: Conference Series*, vol. 1187, no. 5, pp. 52–80.
11. Sunarya, P. A., Nurhaeni, T., & Haris, H. (2017) Bank Reconciliation Process Efficiency Using Online Web Based Accounting System 2.0 in Companies. *APTISI Transactions on Management (ATM)*, vol. 1(2), pp. 124–129.
12. Rahardja, U., Aini, Q., Santoso, N. P. L., Hardini, M., & Edliyanti, A. (2021) Financial management system integrated by web-based payment cash link solution to invent smart reconciliation. *International conference on industrial engineering and operations management*, pp. 4733–4743.
13. Răscolean, I., Rakos, I. S., & Calotă, T. O. (2015) Analysis of the reconciliation of the accounting result with the fiscal result. Case study. *Annals of Constantin Brancusi University of Targu-Jiu. Economy Series*, pp. 297–304.
14. Richardson, J., & Outlaw, J. (2004) Implications of Budget Reconciliation for Commodity Programs. *Choices*, 19(4), pp. 43–45.
15. Xing, X. (2021, December) Financial Big Data Reconciliation Method. *2021 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE)*, pp. 260–263. IEEE.
16. Paneque, M., Roldán-García, M. D. M., & García-Nieto, J. A (2021) Semantic Model for Enhancing Data-Driven Open Banking Services. Available at SSRN 4151598, 11 p.
17. Mints, A., & Sidelov, P. (2019) Analiz global'nogo urovnya gotovnosti bankovskoy sistemy k implementatsii kontseptsii «Open Banking» [Analysis of the global level of readiness of the banking system for the implementation of the "Open Banking" concept]. *Вісник Приазовського державного технічного університету. Серія: Економічні науки*, vol. 37, pp. 57–63.