



PolishAPI

Rekomendacja dot. stosowania standardu PolishAPI w zakresie świadczenia usług inicjowania transakcji płatniczej (PIS) oraz dostępu do informacji o rachunku (AIS) – rozszerzenie danych adresowych

Dokument opracowany przez Grupę Projektową ds. PolishAPI

16 czerwca 2025 r.

Spis treści

1. Opis problemu	2
2. Propozycja rozwiązania	2
3. Uzupełnienia dodatkowe: wprowadzenie opcjonalnych pól dla zwiększenia interoperacyjności i spójności przekazywania informacji w systemach płatniczych	4
4. Uzasadnienie.....	4
5. Załączniki.....	6

1. Opis problemu

W związku z migracją kolejnych systemów płatniczych do standardu ISO20022, wystąpiło zapotrzebowanie na wykorzystanie w komunikatach płatniczych rozszerzonego zakresu danych. W szczególności dotyczy to nowej, ustrukturyzowanej formy danych adresowych. Stosowane do tej pory dane ograniczały się do długości 140 znaków, rozbitych na cztery linie po 35 znaków, i zawierały zarówno nazwę jak i adres strony transakcji. Dodatkowo informacje te były nieustrukturyzowane i jedynie praktyka rynkowa definiująca znaczenie poszczególnych linii pozwalała na ich właściwą interpretację – która też nie zawsze działała poprawnie.

W nowych standardach komunikatów XML opartych o normę ISO20022, dane te są w znaczącym stopniu rozszerzane i strukturyzowane. Oryginalna norma dopuszcza wykorzystanie 140 znaków na nazwę oraz sumarycznie 575 znaków adresu rozbitych po czternastu elementach. Przy dostosowywaniu standardu PolishAPI należy jednak pamiętać, że część systemów płatniczych nie jest jeszcze migrowana. Wymagane jest więc zachowane wsparcie dla dotychczasowego formatu.

Polskie banki zaczęły wraz z konsolidacją T2/T2S oraz migracją SWIFT wprowadzać w swoich interfejsach zmiany pozwalające na wykorzystanie, w pełni lub w części, nowej struktury. W związku z toczącym się projektem migracji SORBNET2 do SORBNET3 oraz planowanym projektem migracji Elixir na XML, coraz częściej klienci mogą spotkać się z nowymi interfejsami. Zakres informacji dozwolony przy inicjacji płatności oraz przy odczytywaniu informacji o rachunku w ramach standardu PolishAPI pozostał jednak niezmienny. Oznacza to, że ogranicza on zakres danych o stronie transakcji do dotychczasowych 140 znaków.

Taka sytuacja w konsekwencji mogłaby oznaczać wzrost ryzyka oskarżenia banków o dyskryminację wobec dostawców trzecich, ponieważ wykorzystywany standard API jest ograniczony w stosunku do natywnych interfejsów banku.

2. Propozycja rozwiązania

W celu wyrównania możliwości oferowanych przez PolishAPI do tych, oferowanych przez natywne interfejsy, prezentowana propozycja rozszerza usługi PIS i AIS w zakresie stron transakcji. Propozycji przyświecała idea minimalizacji zakresu zmian i jak największa kompatybilność wsteczna.

Proponowane zmiany mają polegać na rozszerzeniu klas RecipientPIS i SenderPIS w usługach PIS oraz SenderRecipient w usługach AIS o nowy atrybut nameAddressStructured. Atrybut ten z kolei powinien wskazywać nową klasę NameAddressStructured. Dodatkowo, w ramach klas RecipientPIS oraz SenderPIS, atrybut nameAddress przestanie być wymagany. W ramach zmienionej dokumentacji banków, niezbędne będzie wskazanie wymogu podania parametrów dla atrybuty nameAddress lub nameAddressStructured. Dodatkowo, w ramach konsultacji z sektorem, dodane zostały również atrybuty ultimateSender, ultimateRecipient, ultimateParty wskazujące na klasę UltimateParty oraz atrybut identifiers klasy Identifiers. Dodanie tych atrybutów jest związane z rozszerzaniem interfejsów natywnych o odpowiadające im elementy. Elementy te mogą nie być jednak wspierane przez część Dostawców i w takim wypadku w wewnętrznej dokumentacji niezbędne będzie wskazanie, że te elementy są ignorowane w ramach obsługiwanych usług.

Nowa klasa NameAddressStructured składa się z obowiązkowego atrybutu name o typie string i maksymalnej długości 140 znaków, opcjonalnego atrybutu addressStructured referującego do klasy AddressStructured oraz dodanego po konsultacjach rynkowych opcjonalnego atrybutu adrLine. Klasa AddressStructured z kolei będzie odwzorowaniem pełnego adresu ustrukturyzowanego w standardzie ISO20022. Atrybut adrLine z o długości 140 znaków został dodany na potrzeby prezentacji danych otrzymanych w formie nieustrukturyzowanej lub hybrydowej. Może być jednak wykorzystany również w ramach usług PIS, do wprowadzania adresu w formie hybrydowej – jeżeli Dostawca wspiera ten typ adresu.

Klasa UltimateParty będzie wywoływana z poziomu właściwej strony transakcji – elementy SenderPIS i RecipientPIS dla usług PIS oraz atrybut sender lub recipient odwołujący się do klasy SenderRecipient dla usług AIS. Klasa wykorzystuje atrybut nameAddressStructured do wywołania klasy NameAddressStructured oraz atrybut identifiers do wywołania klasy Identifiers. Zaznaczyć należy, że ze względu na strukturę danych UltimateDebtor i UltimateCreditor w komunikatach ISO, nie ma możliwości wprowadzenia danych tych podmiotów w formie nieustrukturyzowanej. W związku z tym klasa nie ma odwołania do atrybutów klasy NameAddress, które pozostają dostępne dla obiektów sender i recipient.

Klasa Identifiers składa się z dwóch obowiązkowych atrybutów idType oraz idValue i opcjonalnego atrybutu issuer. Zadaniem klasy jest umożliwienie podania parametrów identyfikatora LEI lub innego równoważnego identyfikatora, zgodnie z wymogami rozporządzenia 1113/2023. Atrybuty identifiers są zaprojektowane zawsze jako opcjonalne, ponieważ podanie tych danych jest warunkowe. W opisie klasy w dokumentacji wewnętrznej Dostawców niezbędne będzie wskazanie ograniczeń formatowych dla poszczególnych formatów identyfikatora. Wstępnie klasa dopuszcza dwa typy identyfikatora, czyli LEI oraz TXID, co wynika z ustaleń sektora w stosunku do stosowania innych równoważnych identyfikatorów.

Poprzez dodanie nowej klasy w ramach atrybutu klasy SenderRecipient w usługach AIS, bez ingerencji w dotychczasową klasę NameAddress, możliwe będzie odczytanie informacji zarówno w dotychczasowym, jak i w nowym formacie. W celu minimalizacji zakresu zmian, sugerowane jest, aby w ramach klasy NameAddress zawsze zwracana była nazwa strony transakcji. Dla transakcji z ograniczonym zakresem danych, pełny zestaw nazwy i adresu będzie prezentowany jak dotychczas. Dla nowego zakresu, w klasie NameAddress wszystkie dopuszczalne linie będą uzupełniane nazwą strony, a elementy adresu ustrukturyzowanego będą zwracane w ramach NameAddressStructured. W przypadku stosowania formy hybrydowej, do decyzji Dostawcy będzie, czy scalony i przycięty do 140 znaków zestaw nazwy i adresu nieustrukturyzowanego umieszczać w NameAddress, czy też należy ten

element przeznaczyć jedynie na nazwę, a wszystkie elementy adresowe, zarówno ustrukturyzowane jak i nieustrukturyzowane, wskazać w ramach NameAddressStructured. Na potrzeby instytucji trzecich, które chcą korzystać, kiedy to możliwe, tylko z danych ustrukturyzowanych, nazwa strony będzie powtórzona również w NameAddressStructure w atrybucie name. Zapewni to obsługę najważniejszych elementów danych nawet w przypadku braku zmiany po stronie dostawcy trzeciego.

Plik yaml z proponowaną strukturą usług oraz model danych dla usług AIS zostały przedstawione w załącznikach dołączonych do pisma.

3. Uzupełnienia dodatkowe: wprowadzenie opcjonalnych pól dla zwiększenia interoperacyjności i spójności przekazywania informacji w systemach płatniczych

Dodatkowo, w celu zachowania spójności z SORBNET 3, propozycja została uzupełniona o klasę Purpose, z dwoma elementami: Code i Proprietary. Powyższe wpisano do klas TransferDataBase dla usług PIS oraz TransactionInfo dla usług AIS, gdyż pozwala to na zmapowanie praktycznie każdej usługi. Należy podkreślić, że zaproponowane rozwiązanie w tym zakresie posiada charakter opcjonalny i wynika przede wszystkim z potrzeby standardowego i spójnego przekazywania informacji o przyczynach transakcji między stronami systemu bankowego oraz w kontekście raportowania do organów nadzorczych.

Dodano również opcjonalną pozycję EndToEndId dla przelewów krajowych w systemie SORBNET, przelewów SEPA oraz przelewów SWIFT. Celem tej zmiany jest zwiększenie zgodności PolishAPI z systemami międzynarodowymi, takimi jak SEPA i SWIFT, w których EndToEndId stanowi istotny element identyfikacyjny płatności. Również SORBNET zaczyna wspierać tę referencję, co uzasadnia jej uwzględnienie w standardzie.

Nowe pole zostało dodane w taki sposób, by nie naruszać istniejących implementacji i zachować pełną kompatybilność wsteczną, a jego obsługa zależy od konkretnego typu płatności oraz wewnętrznych preferencji danego banku.

Zmiana została zaimplementowana w klasach bazowych TransferDataBase i TransactionInfo, co pozwala na objęcie nią szerokiego zakresu usług PIS oraz AIS bez konieczności tworzenia nowych klas ani modyfikowania płatności specjalistycznych, takich jak ZUS, podatki czy transakcje kartowe, które i tak nie obsługują tej referencji. Dzięki temu rozwiązaniu możliwe jest spójne i efektywne rozszerzenie standardu przy minimalnym wpływie na istniejące struktury.

Powyższe zmiany zostały zaproponowane do dwóch najczęściej stosowanych na rynku wersji standardu, czyli wersji 3.0 oraz wersji 2.1.1.

4. Uzasadnienie

Ze względu na pilną potrzebę zaadresowania opisanych powyżej problemów, sugerowane zmiany skupiają się na wprowadzeniu niezbędnego minimum modyfikacji, zapewniającego ochronę przed oskarżeniami o dyskryminację. Jednocześnie, dzięki ograniczeniu zmian i zasadzie jak największej kompatybilności wstecznej, ww. zmiany są na tyle małe, że wymagają dotrzymania jedynie trzymiesięcznego okresu informowania o nich dostawców. Ryzyka oskarżeń o dyskryminację, opisane

powyżej, zmaterializują się wraz z uruchomieniem systemu SORBNET3. Uruchomienie systemu SORBNET3 pociągnie za sobą szerokie zmiany w interfejsach natywnych banków, w ramach zleceń płatności krajowych, co znacząco uwidoczni dysproporcję pomiędzy interfejsami natywnymi a API i zwiększy istniejące już ryzyko.

Warto również zaznaczyć, że wprowadzenie teraz ograniczonych zmian w PolishAPI, pozwoli w niedalekiej przyszłości, gdy adresy nieustrukturyzowane przestaną być akceptowane, na poprawne działanie usług PIS oraz AIS. Zmiany są bowiem zaplanowane w taki sposób, aby w pełni wspierały wymóg stosowania adresów ustrukturyzowanych bez konieczności zmian technologicznych w standardzie. Konieczna będzie jedynie zmiana dokumentacji, zakazująca stosowania elementów nieustrukturyzowanych w specyficznych lub wszystkich typach płatności – w zależności od stanu migracji poszczególnych systemów na polskim rynku.

5. Załączniki

1. Zmiany_PolishAPI-ver3_0.yaml
2. Zmiany_PolishAPI-ver2_1_1.yaml