# KRAJOWY PROFIL STANDARDU NETEX

ANALIZA BIZNESOWA I SPECYFIKACJA
TECHNICZNA

Dokument przygotowany na zlecenie Ministerstwa Infrastruktury pzez:
Michał Wolański, Stawki 8/7, 00-193 Warszawa

# SPIS TREŚCI

1. Wprowadzenie ..... 12
2. Praktyki krajowe w zakresie profili krajowych ..... 13
2.1. Wprowadzenie ..... 13
3. Przegląd profili krajowych ..... 14
3.1. Niemcy ..... 14
3.2. Irlandia ..... 14
3.3. Włochy ..... 14
4. Realizacja polskiego profilu minimalnego ..... 15
5. Minimalny polski profil standardu NeTEx ..... 16
5.1. Zastosowania polskiego profilu NeTEx ..... 16
5.2. Organizacja dokumentu ..... 16
5.3. Jakość danych ..... 16
6. Konwencje zapisu ..... 17
6.1. Konwencje pochodzące z normy NeTEx ..... 17
6.2. Konwencje pochodzące z EPIP ..... 18
6.2.1. Typy wyliczeniowe ..... 18
6.2.2. Wskazywanie klas abstrakcyjnych ..... 18
6.2.3. Podkomponenty ..... 18
6.2.4. Wyróżnienia ..... 18
7. Profil wspólny — Framework ..... 19
7.1. Wprowadzenie ..... 19
7.2. Zarządzany obiekt danych, wersja,przestrzeń kodów ..... 19
7.2.1. Przegląd ..... 19
7.2.2. Encja — Entity (klasa abstrakcyjna) ..... 20
7.2.3. Wersjonowana encja — EntityInVersion (klasa abstrakcyjna) ..... 20
7.2.4. Zarządzany obiekt danych — DataManagedObject (klasa abstrakcyjna) ..... 21
7.2.5. Tekst alternatywny — AlternativeText (podkomponent DataManagedObject) .22

7.2.6. Klucz-wartość — KeyValue (podkomponent DataManagedObject) ..... 23
7.2.7. Wersja — Version ..... 23
7.2.8. Przestrzeń kodów — Codespace ..... 23
7.2.9. Warunek ważności — ValidityCondition ..... 26
7.2.10. Ważność pomiędzy — ValidBetween ..... 26
7.2.11. Zbiór odpowiedzialności — ResponsibilitySet ..... 26
7.2.12. Przyporządkowanie roli odpowiedzialności — ResponsibilityRoleAssignment (podkomponent ResponsibilitySet) ..... 27
7.2.13. Źródło danych — DataSource ..... 30
7.3. Relacje i odnośniki do encji ..... 30
7.3.1. Podstawowe zasady ..... 30
7.3.2. Odniesienie do wersji obiektów — VersionOfObjectRef ..... 32
7.3.3. Sprawdzanie integralności dokumentów XML ..... 33
7.3.4. Zakres unikalności identyfikatorów ..... 33
7.4. Ramki danych ..... 34
7.4.1. Przegląd ..... 34
7.4.2. Wersjonowana ramka — VersionFrame (klasa abstrakcyjna) ..... 35
7.4.3. Typ ramki — TypeOfFrame ..... 36
7.4.4. Ramka konkretna, ramka ogólna — SpecificFrame, GeneralFrame ..... 38
7.4.5. Ramka złożona — CompositeFrame ..... 38
7.4.6. Wartości domyślne ramki — FrameDefaults ..... 39
7.5. Elementy ogólne ..... 40
7.5.1. Wprowadzenie ..... 40
7.5.2. Typ wartości — TypeOfValue (klasa abstrakcyjna) ..... 40
7.5.3. Zbiór wartości — ValueSet ..... 41
7.5.4. Grupa encji — GroupOfEntities (klasa abstrakcyjna) ..... 42
7.5.5. Punkt — Point (klasa abstrakcyjna) ..... 43
7.5.6. Lokalizacja — Location (podkomponent Point) ..... 43
7.5.7. Połączenie punktów — Link (klasa abstrakcyjna) ..... 44

7.5.8. Sekwencja połączeń punktów — LinkSequence (klasa abstrakcyjna) ..... 44
7.5.9. Punkt sekwencji połączeń punktów — PointInLinkSequence (klasa abstrakcyjna) 45
7.5.10. Strefa — Zone (klasa abstrakcyjna) ..... 45
7.5.11. Środek transportu, podrzędny środek transportu — TransportMode, Submode46
7.5.12. Przypisanie — Assignment (klasa abstrakcyjna) ..... 50
7.5.13. Prezentacja — Presentation (podkomponent ogólny) ..... 50
7.5.14. Ustawienia regionalne — Locale (podkomponent ogólny) ..... 51
7.5.15. Ustalenia zamawiania — BookingArrangements (podkomponent ogólny) ..... 52
7.5.16. Informacje kontaktowe — ContactDetails (podkomponent ogólny) ..... 54
7.6. Dostępność ..... 54
7.6.1. Przyporządkowanie informacji o dostępności — AccessibilityAssessment (podkomponent ogólny) ..... 55
7.6.2. Ograniczenie dostępności — AccessibilityLimitation (podkomponent AccessibilityAssessment) ..... 56
7.7. Zbiory usług —Facility sets ..... 59
7.7.1. Przegląd ..... 59
7.7.2. Zbiór usług — FacilitySet (klasa abstrakcyjna) ..... 59
7.7.3. Zbiór usług kursu — ServiceFacilitySet ..... 60
7.7.4. Zakwaterowanie — Accommodation (podkomponent ServiceFacilitySet) ..... 60
7.7.5. Zbiór usług miejsca — SiteFacilitySet ..... 61
8. Profil przystanków — Stop Profile ..... 63
8.1. Przegląd ..... 63
8.2. Hierarchia ..... 64
8.3. Inne rodzajemiejsc ..... 65
8.3.1. Pozycja — Place (klasa abstrakcyjna) ..... 65
8.3.2. Pozycja z adresem — AddressablePlace (klasa abstrakcyjna) ..... 66
8.3.3. Składnik miejsca — SiteElement (klasa abstrakcyjna) ..... 66
8.3.4. Miejsce — Site (klasa abstrakcyjna) ..... 68
8.3.5. Adres — Address (klasa abstrakcyjna) ..... 69

8.3.6. Adres pocztowy — PostalAddress (podkomponent SITE) ..... 70
8.3.7. Adres drogowy — RoadAddress (podkomponent SITE) ..... 70
8.3.8. Pozycja topograficzna — Topographie Place ..... 72
8.3.9. Opis pozycji topograficznej — TopographicPlaceDescriptor (podkomponent TopographicPlace) ..... 73
8.3.10. Punkt zainteresowania — Point of Interest ..... 74
8.4. Miejsca zatrzymania (przystanki) ..... 74
8.4.1. Miejsce zatrzymania — StopPlace ..... 74
8.4.2. Komponent miejsca — SiteComponent (klasa abstrakcyjna) ..... 78
8.4.3. Przestrzeń miejsca zatrzymania — StopPlaceSpace (klasa abstrakcyjna) ..... 79
8.4.4. Stanowisko — Quay ..... 79
8.4.5. Wejście — Entrance (klasa abstrakcyjna) ..... 80
8.4.6. Wejście miejsca zatrzymania — StopPlaceEntrance ..... 81
8.5. Grupa miejsc zatrzymania — GroupOfStopPlaces ..... 81
8.6. Strefa taryfowa — TariffZone ..... 82
9. Profil sieci i rozkładów jazdy — Network and Timetable Profile ..... 84
9.1. Wstęp ..... 84
9.2. Podstawowa topologia sieci ..... 84
9.2.1. Przegląd ..... 84
9.2.2. Linia — Line ..... 84
9.2.3. Dozwolony kierunek linii — AllowedLineDirection (podkomponent Line) ..... 87
9.2.4. Grupa linii — GroupOfLines ..... 88
9.2.5. Sieć transportowa — Network ..... 88
9.2.6. Wyświetlenie kierunku — DestinationDisplay ..... 89
9.2.7. Element pośredni trasy — Via ..... 90
9.2.8. Wariant wyświetlenia kierunku — DestinationDisplayVariant ..... 90
9.2.9. Organizacja — Organisation (Abstract) ..... 92
9.2.10. Przewoźnik — Operator ..... 93
9.2.11. Podmiot odpowiedzialny — Authority ..... 93

9.3. Trasy ..... 94
9.3.1. Przegląd ..... 94
9.3.2. Punkt trasy — RoutePoint ..... 94
9.3.3. Połączenie punktów trasy — RouteLink ..... 94
9.3.4. Trasa — Route ..... 95
9.3.5. Punkt na trasie — PointOnRoute (podkomponent Route) ..... 96
9.4. Wzorce kursów — Service patterns ..... 97
9.4.1. Przegląd ..... 97
9.4.2. Wzorzec przejazdu — JourneyPattern (klasaabstrakcyjna w EPIP) ..... 97
9.4.3. Wzorzec kursu — ServicePattern ..... 99
9.4.4. Punkt we wzorcu przejazdu — PointlnJourneyPattern (klasa abstrakcyjna w EPIP) 99
9.4.5. Punkt zatrzymania we wzorcu przejazdu — StopPointlnJourneyPattern (podkomponent JourneyPattern) ..... 100
9.4.6. Kierunek — Direction ..... 103
9.4.7. Punkt pomiaru czasu — Timing point (klasa abstrakcyjna w EPIP) ..... 104
9.4.8. Rozkładowy punkt zatrzymania — ScheduledStopPoint ..... 105
9.4.9. Połączenie rozkładowych punktów zatrzymania — ServiceLink ..... 107
9.5. Przydzielanie zatrzymań ..... 107
9.5.1. Przegląd ..... 107
9.5.2. Przypisanie punktu zatrzymania — StopAssignment (klasa abstrakcyjna) .... 107
9.5.3. Pasażerskie przypisanie punktu zatrzymania — PassengerStopAssignment 108
9.5.4. Dynamiczne przypisanie punktu zatrzymania — DynamicStopAssignment ... 110
9.5.5. Przypisanie pociągu do punktu zatrzymania — TrainStopAssignment (podkomponent PassengerStopAssignment) ..... 111
9.6. Connections, transfers and interchanging ..... 112
9.6.1. Przegląd ..... 112
9.6.2. Przejście piesze — Transfer (klasa abstrakcyjna) ..... 113
9.6.3. Czas trwania przejścia pieszego — TransferDuration (podkomponent Transfer) 114
9.6.4. Przejście piesze pomiędzy punktami zatrzymania — Connection ..... 115

9.6.5. Kraniec przejścia pieszego pomiędzy punktami zatrzymania — ConnectionEnd (podkomponent Connection) ....115
9.6.6. Przejście piesze pomiędzy miejscami — SiteConnection ....115
9.6.7. Kraniec przejścia pieszego pomiędzy miejscami — SiteConnectionEnd (podkomponent SiteConnection) ....116
9.6.8. Domyślne przejście piesze pomiędzy środkami transportu — DefaultConnection 117
9.6.9. Kraniec domyślnego przejścia pieszego pomiędzy środkami transportu — DefaultConnectionEnd (podkomponentDefaultConnection) ....117
9.7. Kursy i czasy mijania — Vehicle journeys, passing times ....118
9.7.1. Przegląd ....118
9.7.2. Przejazd — Journey (klasa abstrakcyjna) ....118
9.7.3. Kurs — ServiceJourney ....119
9.7.4. Czas mijania — PassingTime (klasa abstrakcyjna) ....121
9.7.5. Rozkładowy czas mijania — TimetabledPassingTime (podkomponent ServiceJourney) ....122
9.7.6. Grupa kursów — GroupOfServices ....123
9.7.7. Element grupy kursów — GroupOfServicesMember (podkomponent GroupOfServices) ....124
9.8. Bardzo częste kursy ....125
9.8.1. Wprowadzenie ....125
9.8.2. Szablon kursu pojazdu — TemplateVehicleJourney ....125
9.8.3. Grupa kursów częstotliwościowych — JourneyFrequencyGroup (klasa abstrakcyjna) ....126
9.8.4. Grupa kursów z interwałem — HeadwayJourneyGroup (podkomponent TemplateJourney) ....127
9.9. Przesiadki pomiędzy kursami ....128
9.9.1. Wprowadzenie ....128
9.9.2. Przesiadka — Interchange (klasa abstrakcyjna) ....129
9.9.3. Przesiadka pomiędzy kursami — ServiceJourneyInterchange ....130
9.10. Pociągi i wagony komunikacji bezpośredniej ....132
9.10.1. Przegląd ....132

9.10.2. Numer pociągu — TrainNumber .....132
9.10.3. Część przejazdu — JourneyPart (podkomponent ServiceJourney).....133
9.10.4. Powiązanie części przejazdu — JourneyPartCouple.....135
9.10.5. Umiejscowienie części przejazdu — JourneyPartPosition (podkomponent JourneyPartCouple) ..... 137
9.11. Pojazdy i pociągi.....138
9.11.1. Przegląd .....138
9.11.2. Typ pojazdu — VehicleType .....138
9.11.3. Pociąg — Train .....139
9.11.4. Element pociągu — TrainElement (podkomponent Train) .....140
9.11.5. Komponent pociągu — TrainComponent (podkomponent Train).....141
9.12. Uwagi .....142
9.12.1. Wprowadzenie .....142
9.12.2. Uwaga — Notice (podkomponent ogólny) .....142
9.12.3. Przypisanie uwagi — NoticeAssignment (podkomponent ogólny) .....143
9.13. Kalendarze .....144
9.13.1. Przegląd .....144
9.13.2. Typ dnia — DayType .....144
9.13.3. Własność dnia — PropertyOfDAy (podkomponent DayType) .....145
9.13.4. Okres dnia — Timeband (podkomponent DayType) .....147
9.13.5. Kalendarz kursowania — ServiceCalendar .....148
9.13.6. Dzień kursowania — OperatingDay (podkomponent ServiceCalendar)..... 149
9.13.7. Przypisanie typu dnia — DayTypeAssignment (podkomponent ServiceCalendar) 150
9.13.8. Okres kursowania — OperatingPeriod (klasa abstrakcyjna w EPIP) .....151
9.13.9. Okres kursowania UIC — UicOperatingPeriod (podkomponent ServiceCalendar) 152
10. Wspólne zasady.....153
10.1. Zarządzanie wersjami.....153
10.2. Używanie warunków ważności
153

10.3. Używanie znaczników czasu stworzenia i modyfikacji ..... 154
10.4. Przestrzeń kodów i struktura identyfikatorów ..... 154
10.4.1. Wprowadzanie ..... 154
10.4.2. Używanie CODESPACE ..... 155
10.4.3. Struktura identyfikatorów obiektów danych ..... 155
10.4.4. Struktura identyfikatorów ramek wersji ..... 156
10.4.5. Wersje profilu ..... 157
10.4.6. Zależności pomiędzy ramkami ..... 157
10.5. Organizacja i zagnieżdżanie elementów w dokumencie ..... 157
10.6. Odpowiedzialność za dane ..... 158
10.7. Przetwarzanie dużych zbiorów danych ..... 159
10.8. Dostępne typy ramek danych ..... 159
10.8.1. Lista typów ramek danych ..... 159
10.8.2. Ramki pomocnicze ..... 160
10.8.3. Ramki główne ..... 161
10.8.4. Ramki złożone ..... 165
10.8.5. Ramki metadanych ..... 166
10.9. Kompletność danych ..... 166
10.9.1. Wprowadzenie ..... 166
10.9.2. Puste wartości ..... 167
10.10. Wymiana danych ..... 167
10.11. Obiekty PublicationDelivery ..... 167
10.11.1. PublicationDelivery ..... 167
10.11.2. PublicationRequest (podkomponent PublicationDelivery) ..... 168
10.11.3. NetworkFrameTopic (podkomponent PublicationDelivery) ..... 169
10.11.4. NetworkFilterByValue (podkomponent PublicationDelivery) ..... 169
10.12. Nazwy plików ..... 169
10.13. Rozszerzanie o dane dodatkowe ..... 170

11. Dodatek A 172

11.1. A.1FacilityList (udogodnienia) 172
11.2. A.2AccessibilityInfoFacilityList (dostępność informacji) 172
11.3. A.3AccessibilityToolList (środek dostępności) 172
11.4. A.4 AccommodationAccessList (rodzaj ulokowania podróżnych) ..... 172
11.5. A.5 AccommodationFacilityList (własności ulokowania podróżnych) ..... 173
11.6. A.6 AssistanceFacilityList (usługi pomocowe) ..... 173
11.7. A.7 BoardingPermission (zezwolenie na wsiadanie) ..... 173
11.8. A.8 BookingProcessFacilityList (proces zakupu) ..... 174
11.9. A.9 CarServiceFacilityList (usługi samochodowe) ..... 174
11.10. A.10CateringFacilityList (usługi gastronomiczne) 174
11.11. A.11CouchetteFacilityList (usługi w wagonach z miejscami do leżenia) 175
11.12. A.12FamilyFacilityList (usługi dla rodzin) 175
11.13. A.13FareClasses (klasy taryfowe) 175
11.14. A.14GenderLimitation (ograniczenia płci) 175
11.15. A.15GroupBookingFacility (rezerwacje grupowe) 175
11.16. A.16LuggageCarriageFacilityList (przewóz bagażu) 176
11.17. A.17MealFacilityList (posiłki) 176
11.18. A.18MedicalFacilityList (usługi medyczne)

11.30. A.30UicProductCharacteristicList (klasyfikacja produktów UIC) 179

11.31. A.31UicTrainRate (stawka UIC)

# 1. WPROWADZENIE

Minimalny polski profil standardu NeTEx (PL_NeTEx_Profile) bazuje na Europejskim profilu informacji pasażerskiej (EPIP), który jest opisany w normie PKD-CEN/TS 16614-4:2021. W związku z powyższym niniejszy dokument opiera się na wspomnianej normie i ma podobną strukturę, co pozwala na łatwe wskazanie różnic pomiędzy profilami.

EPIP jest zbudowany na podstawie pełnego standardu NeTEx, definiowanego przez PKN-CEN/TS 16614-1:2021, PKN-CEN/TS 16614-2:2021 i PKN-CEN/TS 16614-3:2021.

# 2. PRAKTYKI KRAJOWE W ZAKRESIE PROFILI KRAJOWYCH

## 2.1. Wprowadzenie

Zgodnie z Rozporządzeniem Delegowanym Komisji (UE) 2017/1926 z dnia 31 maja 2017 r. kraje członkowskie opracowują własne profile krajowe stanowiące podzbiór normy NeTEx.

Pierwszym etapem zazwyczaj jest opracowanie profilu minimalnego, który ma za zadanie zaznajomić podmioty biorące udział w wytwarzaniu i przetwarzaniu danych rozkładów z normą NeTEx i stanowić pierwszą, najprostszą do implementacji wersję schematu danych. Profil ten powinien zakładać możliwość późniejszego rozszerzania w miarę szerszego wykorzystywania przez podmioty krajowe i zgłaszane zapotrzebowanie na dodatkowe elementy.

Dlatego też punktem wyjścia jest Europejski Profil Informacji Pasażerskiej (EPIP), który został opracowany jako minimalny profil w kontekście europejskim i który powinien być rozszerzany przez wszystkie profile krajowe państw członkowskich.

W zależności od stopnia unormowania przekazywania danych o transporcie publicznym w danym państwie pierwszy profil minimalny może zostać zdefiniowany w różnym zakresie. Niektóre państwa, w szczególności te, które brały czynny udział w definiowaniu normy NeTEx w latach ubiegłych, posiadają własne, krajowe normy wymiany danych, które są szeroko wykorzystywane przez tamtejsze podmioty (przewoźników, organizatorów transportu i inne instytucje). W takim przypadku istnieje dobrze zdefiniowany zakres domenowych danych wykorzystywanych w transporcie publicznym. Opracowanie profilu krajowego sprowadza się zatem do stworzenia odwzorowania pomiędzy istniejącą normą krajową a normą NeTEx. Nie jest wymagana dodatkowa analiza potrzebnych danych, ponieważ dokładnie takie dane są przechowywane w dotychczasowej normie, która uwzględnia wszelkie krajowe subtelności.

W przypadku państw, które nie posiadają standardów krajowych, a organizacja transportu publicznego jest wysoce rozdrobniona, podczas tworzenia profilu krajowego można oprzeć się na profilu europejskim EPIP lub przeprowadzić analizę wielu niezależnych systemów wytwarzających te dane i przekazujących je w różny sposób. W drugim przypadku zagadnienie komplikuje dodatkowo fakt, iż wiele mniejszych podmiotów może w ogóle nie korzystać z żadnych narzędzie informatycznych generujących dane w formie czytelnej maszynowo. W takich sytuacjach wydaje się zatem sensowne rozpocząć pracę od profilu minimalnego realizującego wymagania EPIP i rozpoczęcie konsultacji i implementacji przez pierwsze przedsiębiorstwa. W kolejnych etapach, jeżeli okaże się to zasadne, należy rozszerzać profil krajowy o wymagane elementy specyficzne dla danego kraju.

# 3. PRZEGLĄD PROFILI KRAJOWYCH

Poniżej przedstawiono szczególne przypadki państw, które wytworzyły już profile krajowe NeTEx i je opublikowały.

## 3.1. Niemcy

W Niemczech realizację profilu krajowego stanowi dokument *VDV Schrift 462.* Powstał on na podstawie niemieckiego dokumentu *VDV Schrift 452* opublikowanego pierwotnie w 1999 roku, który opisuje schemat opisu danych transportu publicznego. W związku z tym, że opis ten stanowi standard danych używany szeroko w Niemczech przez ponad 20 lat oraz fakt, że sama norma NeTEx powstawała między innymi na podstawie tego dokumentu, niemiecki profil krajowy jest rozbudowany, dobrze zdefiniowany i odpowiada lokalnym potrzebom.

Dokument opisujący profil zawiera opisy wszystkich obiektów potrze bnych podczas wymiany danych. Dodatkowo zawarte są tam też tablice dokładnych odwzorowań pomiędzy systemami VDV i NeTEx, które pozwalają na dokładne przeniesienie danych bez zmiany semantyki.

## 3.2. Irlandia

Profil irlandzki EIRENP-NT powstał bezpośrednio na podstawie opisu profilu europejskiego (EPIP) z uwzględnieniem dodatkowych pól zdefiniowanych w normie NeTEx, ale nie używanych w podstawowym EPIP. Pola te co do zasady odpowiadają dodatkowym informacjom, które są dostępne w wykorzystywanym tam systemie krajowym opisującym miejsca i przystanki: NPTG *(National Public Transport Gazetteer)* oraz NaPTAN *(National Public Transport Access Node).*

## 3.3. Włochy

Opis włoskiego profilu zamieszczony w Linee guida per la compilazione del profilo italiano del NeTEx jest dużo bardziej zwięzły niż powyższe dwa. Ogranicza się on do krótkiego opisu poszczególnych elementów profilu europejskiego bez odniesienia do lokalnych systemów. Wynika to z dużego rozproszenia sektora transportu publicznego i zapewne braku dotychczasowych standardów. Zaproponowany profil ma stanowić bazę do późniejszych rozszerzeń po wdrożeniu pierwszej iteracji.

# 4. REALIZACJA POLSKIEGO PROFILU MINIMALNEGO

Na podstawie wymagań stawianych przez rozporządzenie, rozwiązań przyjętych przez inne państwa oraz obecnego sposobu organizacji transportu publicznego w Polsce i stopnia jego informatyzacji, została podjęta decyzja o stworzeniu polskiego profilu minimalnego maksymalnie zbieżnego z europejskim (EPIP).

Za taką decyzją przemawiało szereg czynników, takich jak brak dotychcz asowego standardu przekazywania takich danych w Polsce, duże rozdrobnienie organizatorów transportu i przewoźników oraz niski poziom informatyzacji mniejszych podmiotów.

Analiza profilu europejskiego wykazała, że zasadniczo zapewnia on wystarczający opis wszystkich aspektów transportu publicznego w Polsce bez dodatkowych rozszerzeń.

Profil został rozszerzony o możliwość zapewniania szczegółowych informacji o skomunikowaniach pomiędzy liniami (w szczególności: pociągami) wraz z podaniem maksymalnego czasu oczekiwania na opóźniony pojazd.

Praktyka skomunikowań jest stosowana na polskiej kolei dość szeroko, wynika to między innymi z dość małej częstotliwości ruchu pociągów i braku taktu w rozkładach. W krajach zachodnich z większym ruchem kolejowym często połączenia nie czekają na siebie, a pasażerowie mają do dyspozycji kolejne połączenie w danym kierunku po dość krótkim czasie (np. za 30 minut, za godzinę). W ten sposób opóźnione pociągi w ograniczonym stopniu wpływają na ruch innych i nie są przenoszone opóźnienia wtórne. W polskich warunkach nietrzymanie skomunikowań często wiązałoby się z wielogodzinnym oczekiwaniem na kolejny pociąg. Stąd też narodowy zarządca infrastruktury kolejowej — PKP PLK — w swoim systemie konstrukcji rozkładów jazdy przewiduje możliwość jawnego definiowania skomunikowań wraz z maksymalnym czasem, przez jaki skomunikowany pociąg może oczekiwać na ten opóźniony. Polski profil minimalny umożliwia zapisanie takich danych w instancjach klasy ***ServiceJourneyInterchange*** (zobacz Przesiadka pomiędzy kursami — ServiceJourneyInterchange).

Rozszerzenie jest zgodne z normą PKN-CEN/TS 16614-2:2021 i opiera się na zdefiniowanych tamże (sekcja 7.3.9.3 2) atrybutach: ***Guaranteed, StandardWaitTime, MaximumWaitTime*** oraz ***MaximumAutomatic WaitTime.***

Polski profil minimalny realizuje zatem wymagania rozporządzenia przez zastosowanie jako bazy profilu europejskiego i jest zbudowany w taki sposób, aby w przyszłości umożliwić proste rozszerzenie o dodatkowe elementy.

# 5. MINIMALNY POLSKI PROFIL STANDARDU NETEX

## 5.1. Zastosowania polskiego profilu NeTEx

Profile NeTEx tworzą podzbiory schematu NeTEx, które są użyteczne w kontekście transportu publicznego w danym kraju. Głównym celem profilu jest zapewnienie danych przydanych z punktu widzenia pasażerów i systemów informacji pasażerskiej. Z tego powodu, nie są zawarte tu informacje dotyczące szczegółów operacyjnych prowadzenia przewozów, które nie są istotne dla pasażerów.

Typowe zastosowania dla profilu:

- dostarczenie danych do wyszukiwarki połączeń,
- dostarczenie danych narzędzia wizualizującego sieć połączeń na mapie,
- dostarczenie danych narzędzia generującego drukowane rozkłady jazdy.

## 5.2. Organizacja dokumentu

Profil jest podzielony na trzy główne części:

- **profil wspólny** — zawierający wspólne pojęcia używane przez kolejne części,
- **profil przystanków** — opisujący miejsca zatrzymań i ich hierarchię,
- **profil sieci i rozkładów jazdy** — opisujący sieć transportową i rozkłady jazdy.

## 5.3. Jakość danych

Jednym z celów profilu jest zapewnienie interoperacyjności z innymi systemami. Aby to osiągnąć, konieczna jest wysoka jakość przekazywanych danych.

W związku z tym, dane XML zawierają wiele automatycznych narzędzi do weryfikacji poprawności, np. sprawdzanie dozwolonych wartości, użytych odniesień oraz wymaganych kształtów struktur (wymagane elementy i ich liczba).

Część weryfikacji nie może być jednak wykonana automatycznie, np. sprawdzenie, czy kurs ma przypisane odpowiednie nazwy przystanków zgodne ze stanem faktyczn ym.

# 6. KONWENCJE ZAPISU

Zapis w niniejszym dokumencie w miarę możliwości odzwierciedla ten użyty w normie NeTEx. Pewne różnice wymuszone są faktem, że tekst tego dokumentu napisany jest w języku polskim, a norma europejska — w angielskim. W związku z tym, że znaczenie poszczególnych pojęć jest ściśle zdefiniowane w słowniku stanowiącym cześć Transmodelu, te pojęcia zostały pozostawione w języku angielskim, aby nie naruszać ustalonej semantyki.

## 6.1. Konwencje pochodzące z normy NeTEx

NeTEx stosuje spójny sposób prezentacji pojęć technicznych:

- Modele koncepcyjne z Transmodelu są zapisywane WERSALIKAMI, np. "LINE", "SCHEDULED STOP POINT". W związku z różnicami językowymi nie są tworzone od nich liczby mnogie tak jak w tekście NeTEx. Zamiast tego, używane są rzeczowniki opisujące rodzaj encji w odpowiedniej odmianie, aby określić, czy chodzi o liczbę pojedynczą, czy mnogą w danym kontekście (np. "obiekty LINE").
- Wszystkie elementy schematu NeTEx XML są zapisywane pismem ***wytłuszczonym pochyłym,*** np. "***Line***". Nazwy złożone są zapisywane w formie PascalCase, np. ***"ScheduledStopPoinf.***
- Własności będące atrybutami klas XML składane są pismem ***wytłuszczonym pochyłym małymi literami,*** np. ***'"id", '"order'.***
- Własności będące elementami potomnymi klas XML, składane są jak inne elementy, to jest pismem ***wytłuszczonym pochyłym rozpoczynającym się wielką literą (PascalCase),*** np. ***"ScheduledStopPoint/ Name", "SiteElement/ Covered".***
- Zamknięte listy enumeracyjne składane się pismem pochyłym (zazwyczaj camel-case rozpoczynającym się małą literą) np. ***"****busStop'****",*** *"canalBarge"****,*** *"rail****".***
- Relacje jeden-do-jednego, które zaimplementowano jako wersjonowane odniesienia (referencje), są zapisywane jako proste nazwy atrybutów z sufiksem ***"Ref',*** np.: ***"ScheduledStopPointRef'.*** Jeżeli relacja jest zwrotna albo istnieje więcej niż jedna relacja pomiędzy tymi samymi dwoma klasami, nazwy posiadają prefiks rozróżniający, np.: ***" ParentZoneRef", " FromPointRef", "ToPointRef'.***
- Relacje jeden-do-wielu, które zaimplementowano jako kolekcje, są zapisywane małymi literami w liczbie mnogiej, np. *"****stopPoints****".* Nazwa relacji jest więc używana jako tag okalający. Takie relacje mogą definiować listę encji na dwa sposoby:
  - jako proste wersjonowane referencie (np. wiele instancji ***ScheduledStopPointRef***);
  - jako osadzone elementy, których cała definicja jest zamieszczona wewnątrz zawierającego elementu (np. wiele instancji ***ScheduledStopPoint***).
- Typy danych zapisywane są pismem *pochyłym:*
  - wbudowane typy XML zapisywane są tak samo, camel-case i zazwyczaj z prefiksem ***"****xsd:",*** np. *xsd:dateTime****,*** *xsd:normalizedString****;***
  - typy danych NeTEx zapisywane są wielką literą bez prefiksu, np. *LengthType****,*** *QuayTypeEnum****.***

- Złożone typy NeTEx, które mają osobną definicję, są dodatkowo podkreślone, np. *DataManagedObject, Line.* Wyjątek stanowią często wykorzystywane typy pomocnicze, które zapisywaną są bez podkreślenia, np. *MultilingualString*.
- Stereotypy UML zapisywane są wewnątrz cudzysłowów francuskich (szewronów), np. «enum», «FK».

## 6.2. Konwencje pochodzące z EPIP

### 6.2.1. Typy wyliczeniowe

Wszędzie, gdzie to możliwe EPIP standaryzuje listę dostępnych wartości dla typów wyliczeniowych. NeTEx przewiduje kilka różnych mechanizmów do tego celu.

- **Stałe wyliczenia**, które są zdefiniowane jako część schematu NeTEx (np. *busStop).* EPIP i profile krajowe definiują podzbiór tych wartości — zobacz Dodatek A.
- **Specjalizacje** encji TYPE OF VALUE, służące definiowaniu otwartych zbiorów wartości, które mogą być rozszerzane w miarę upływu czasu bez konieczności zmiany schematu. EPIP dodaje w ten sposób kilka przypadków; dla wyróżnienia umieszczone są one w CODESPACE "*epip_metadata*" i zawierają prefiks *"epip:",* np.: "*epip:monomodal*".
- Instancje **TypeOfFrame** służące do rozróżnienia zastosowań obiektów VERSION FRAME w profilu. Identyfikatory instancji **TypeOfFrame** zapisywane są wielkimi literami z podkreśleniem pomiędzy wyrazami, np. "*epip:EU_PI_LINE_OFFER*'.

### 6.2.2. Wskazywanie klas abstrakcyjnych

NeTEx szeroko korzysta z mechanizmu dziedziczenia, co pozwala uprościć definicję większości klas oraz uniknąć zbędnego powtarzania specyfikowania części wspólnych. Większość nadklas jest abstrakcyjna, a tylko końcowe są właściwe, tj. mogą posiadać instancje.

Aby ułatwić zrozumienie struktury podczas czytania dokumentu, klasy abstrakcyjne zostały oznaczone przez sufiks "**(klasa abstrakcyjna)**" w nagłówkach. Kolejność definicji została ustalona tak, aby nadklasa zawsze była zdefiniowana wcześniej, niż podklasa.

Część klas w normie NeTEx nie jest abstrakcyjna, ale w EPIP nigdy nie są tworzone jej instancje; takie klasy są wyróżniane sufiksem "**(klasa abstrakcyjna w EPIP)**".

### 6.2.3. Podkomponenty

Część klas w EPIP posiada podkomponenty, które tworzą ich definicje. Zapewniają one uszczegółowienie podstawowego modelu i zazwyczaj są częścią tylko jednej klasy. Dlatego też są oznaczane sufiksem "**(podkomponent XXX)**", gdzie XXX jest nazwą bazowej klasy. Ich definicja może zostać pominięta podczas pierwszego czytania dokumentu.

### 6.2.4. Wyróżnienia

Podkreślenie używane jest do wyróżnień ważnych fragmentów, ponieważ pismo pochyłe i wytłuszczone używane jest do wskazywania elementów w schemacie XML.

# 7. PROFIL WSPÓLNY — FRAMEWORK

## 7.1. Wprowadzenie

W celu ułatwienia implementacji wszystkie obiekty normy NeTEx używają jednolitego mechanizmu realizacji wspólnych funkcji potrzebnych do efektywnej wymiany danych, w szczególności:

- wersjonowanie;
- definiowanie ważności;
- przypisywanie odpowiedzialności za zbiory danych;
- grupowanie danych.

Niniejszy rozdział opisuje wspólny zrąb NeTEx (NeTEx Framework), który jest współdzielony przez wszystkie pozostałe części profilu.

## 7.2. Zarządzany obiekt danych, wersja, przestrzeń kodów

### 7.2.1. Przegląd

***DataManagedObject****,* wraz z ***Version*** i ***Codespace*** zapewniają wspólne własności dla wszystkich encji NeTEx.

***DataManagedObject*** jest abstrakcyjnym obiektem stanowiącym korzeń drzewa klas w implementacji NeTEx i wszystkie podstawowe encje są jego specjalizacjami.

ENTITY w Transmodelu reprezentuje obiekt w repozytorium, który posiada instancję w wymienianym zestawie danych. Każde ENTITY może posiadać kilka wersji — ENTITY podlega zmianą w trakcie czasu życia. Zazwyczaj zbiór danych zawiera tylko jedną konkretną wersję ENTITY, aczkolwiek możliwe jest załączenie więcej niż jednej wersji, jeżeli przypadek użycia tego wymaga.

Dane wyeksportowane z repozytorium do dokumentu XML reprezentują stan obowiązujący w konkretnym momencie (snapshot). NeTEx używa więc głównie danych będącymi instancjami ENTITY IN VERSION z Transmodelu. W dokumencie znajdują się zatem pojedyncze wersje każdego ENTITY, a nie pełna historia zmian. Podczas wykonywania eksportu tworzone są nowe instancje ENTITY IN VERSION odpowiadające czasowi eksportu; jeżeli bieżąca wersja w repozytorium zmieni się pomiędzy eksportami, zawierać one będą dane.

W ramach modelu fizycznego NeTEx ENTITY IN VERSION jest implementowane przez ***EntityInVersion****,* które jest specjalizowane przez ***DataManagedObject****,* która to encja łączy pewne inne pojęcia z Transmodelu w pojedynczą abstrakcyjną klasę XML. Pozwala to na wygodne umieszczanie danych o wersjach, odpowiedzialności i ważności poprzez mechanizm rozszerzania w XML.

NeTEx używa obiektów ***Codespace****,* aby zapewnić unikalność identyfikatorów elementów w dokumentach XML, co jest istotne w szczególności, jeżeli dane pochodzą o różnych dostawców. Zobacz Przestrzeń kodów — Codespace oraz Przestrzeń kodów i struktura identyfikatorów.

***ValidityCondition*** jest używane do specyfikowania okoliczności, w których dany element powinien być używany (np. "w zimie"). VALIDITY CONDITION może być dołączone do poszczególnych elementów lub poprzez VERSION FRAME do większych zestawów danych, aby uniknąć powtórzeń.

***DataManagedObject*** pozwala na przydzielenie ***Version*** i atrybutów odpowiedzialności do ***EntityInVersion****,* a także do jednej lub więcej instancji ***ValidityCondition****.*

### 7.2.2. Encja — Entity (klasa abstrakcyjna)

ENTITY reprezentuje obiekt w wymienianym zbiorze danych.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *ObjectIdType* | 1:1 | Identyfikator obiektu ENTITY, unikalny wewnątrz CODESPACE. |

### 7.2.3. Wersjonowana encja — EntityInVersion (klasa abstrakcyjna)

Konkretna wersja ENTITY. Klasa ***EntityInVersion*** w NeTEx implementuje ENTITY IN VERSION poprzez dodanie informacji o VERSION i VALIDITY CONDITION (zobacz poniżej) do dowolnego elementu NeTEx.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Entity* | ::> | ENTITY IN VERSION dziedziczy po ENTITY. |
| «FK» | ***dataSourceRef*** | *DataSourceIdType* | 0:1 | System z którego pochodzą dane. |
| «atr» | ***created*** | *xsddateTime* | 0:1 | Data i czas stworzenia ENTITY. |
| «atr» | ***changed*** | *xsddateTime* | 0:1 | Data i czas ostatniej zmiany ENTITY. |
| «enum» | ***modification*** | *ModificationEnum* | 0:1 | Rodzaj modyfikacji. Dostępne wartości:<br>• *new* (nowa),<br>• *revise* (poprawka),<br>• *unchanged* (niezmienione)<br>Jeżeli puste, należy interpretować jako wartość z elementu nadrzędnego. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***version*** | *VersionIdType* | 0:1 | Numer wersji (VERSION) tej instancji ENTITY IN VERSION. |
| «enum» | ***status*** | *VersionStatusEnum* | 0:1 | Status ENTITY w VERSION. EPIP obsługuje tylko jedną wartość: *active.* |
| «enum» | ***status*** | *VersionStatusEnum* | 0:1 | Status ENTITY w VERSION. EPIP obsługuje tylko jedną wartość: *active.* |
| «cntd» | ***ValidBetween*** | *ValidBetween* | 0:1 | Uproszczona wersja VALIDITY CONDITION, zawierająca prosty zakres daty, bez warunku na unikalność. |

### 7.2.4. Zarządzany obiekt danych — DataManagedObject (klasa abstrakcyjna)

***DataManagedObject*** jest podklasą ***EntityInVersion****,* która implementuje własności wspólne dla wszystkich podstawowych encji w NeTEx, takie jak:

- ***ResponsibilitySet****,* określające odpowiedzialny podmiot za zarządzanie danymi,
- ***AlternativeText****,* stanowiący zbiór alternatywnych wariantów dowolnych danych tekstowych powiązanych z ENTITY; główny cel tego elementu to zapewnienie tłumaczeń na inne języki urzędowe.

***DataManagedObject*** jest klasą wynikającą z implementacji i nie pojawia się w Transmodelu.

| Klasyfikacja ::> | Nazwa ::> | Typ *EntityInVersion* | Kardynalność ::> | Opis DATA MANAGED OBJECT dziedziczy po ENTITY IN VERSION. |
|---|---|---|---|---|
| «FK» | ***responsibilitySetRef*** | *ResponsibilitySetIdType* | 1:1 | Odniesienie do RESPONSIBILITY SET określającego własność i |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | odpowiedzialność za zarządzanie danymi. |
| «cntd» | ***keyList*** | *KeyValue* | 0:* | Lista par ***KeyValue****,* które można powiązać z obiektem. Pozwala to na przechowywanie prostych dodatkowych danych przez użytkownika. |
| «FK» | ***BrandingRef*** | *BrandingRef* | 0:1 | Odniesienie do BRANDING, zapewniającego dodatkowe informacje handlowe o obiekcie. |
| «cntd» | ***AlternativeTexts*** *AlternativeText* | | 0:* | Dodatkowe tłumaczenia elementów tekstowych obiektu. |

### 7.2.5. Tekst alternatywny — AlternativeText (podkomponent DataManagedObject)

***AlternativeText*** zapewnia możliwość podania różnych wariantów danych tekstowych obiektu, w szczególności tłumaczeń na inne języki, np. "en.Warsaw", "de.Warschau", "hu.Varsó". Ta własność może być postrzegana jako komplementarna do ***AlternativeName*** opisanej poniżej, która służy do zapewniania aliasów nazw, np. "Warszawa Centralna", "Dworzec Centralny".

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionedChild* | ::> | ***AlternativeText*** dziedziczy po VERSIONED CHILD. |
| «PK» | ***attributeName*** | *xsd:NCName* | 0:1 | Nazwa atrybutu tekstowego, do którego niniejszy obiekt stanowi alternatywę. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***usedForLanguage*** | *xsd:language* | 0:1 | Nazwa języka, z którym należy użyć tekstu. |
| «atr» | ***order*** | *xsd:integer* | 0:1 | Liczba porządkowa tekstu w kolekcji. |
| | ***Text*** | *MultilingualString* | 1:1 | Wariant tekstu w określonym języku. |

### 7.2.6. Klucz-wartość — KeyValue (podkomponent DataManagedObject)

***KeyValue*** pozwala to na przechowywanie prostych dodatkowych danych przez użytkownika nieujętych w normie NeTEx. Listy "klucz-wartość" mogą być przypisane do dowolnego obiektu ***DataManagedObject***. W EPIP rozszerzenia powinny być realizowane tylko przez ten mechanizm. Należy jednak zauważyć, że nie powinien być ony wykorzystywany do przekazywania informacji, które można zapisać wewnątrz struktur NeTEx.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***typeOfKey*** | *xsd:normalizedString* | 0:1 | Typ pary KEY VALUE. |
| «PK» | ***Key*** | *xsd:normalizedString* | 1:1 | Klucz pary KEY VALUE. |
| | ***Value*** | *xsd:normalizedString* | 1:1 | Wartość pary KEY VALUE. |

### 7.2.7. Wersja — Version

NeTEx obsługuje wymianę danych pomiędzy wieloma systemami i definiuje możliwość śledzenia każdej zmiany w każdym elemencie poprzez atrybut ***version*** w ***DataManagedObject***. Jednakże nie wszystkie istniejące systemy zapewniają tak rozbudowaną kontrolę, dlatego w EPIP podawanie wersji dla poszczególnych obiektów jest opcjonalne. Wymagane jest natomiast specyfikowanie wersji na poziomie ramki danych.

VERSION w Transmodelu jest obiektem pierwszej kategorii i może zawierać opis wyjaśniający ważność danej wersji. Te dane mogą być przekazywane w ramach instancji ***Version*** w normie NeTEx. Instancje ***Version*** nie są zawarte w EPIP i do identyfikowania wersji jest używany wyłącznie jej numer.

### 7.2.8. Przestrzeń kodów — Codespace

CODESPACE jest pojęciem podobnym do przestrzeni nazw w XML *(namespace),* która definiuje kontekst, w ramach którego nazwy elementów i atrybutów w schemacie powinny być unikalne. Pozwala to na łączenie elementów z różnych schematów w jeden zagregowany dokument. W podobny sposób NeTEx używa obiektów ***Codespace*** do zapewnienia, że identyfikatory instancji w dokumencie XML są unikalne, nawet jeżeli pochodzą z różnych systemów i są umieszczone w jednym dokumencie (zobacz przykłady poniżej). Pozwala to na agregowanie bez konfliktów danych z różnych państw, regionów i podmiotów.

Obiekty ***Codespace*** używane są przede wszystkim do:

1. definiowania kontekstu, w którym należy zapewnić unikalność identyfikatorów;
2. wskazywanie systemom zależnym, którego systemu identyfikatorów używać do interpretacji danych;
3. zarządzania rozproszonym nadawaniem nowych identyfikatorów dla nowych instancji obiektów.

Schemat NeTEx zawiera wymogi unikalności i integralności odniesień XML, które sprawdzają, czy każdy identyfikator jest unikalny oraz, czy każda wersjonowana referencja do encji jest spełniona (to jest, że obiekt, na który wskazuje odniesienie, również znajduje się w dokumencie). Te sprawdzenia mogą być wykonywane automatycznie przez dowolny walidator XML na dowolnym dokumencie NeTEx i są kluczowe do zapewnienia podstawowego poziomu jakości wymienianych danych.

Obiekty CODESPACE zapobiegają konfliktom identyfikatorów pomiędzy danymi pochodzącymi z różnych krajów, regionów lub podmiotów. Różne systemy identyfikatorów mogą być używane przez różne CODESPACE.

W celu zarządzania wspólnym systemem identyfikatorów (kodów) przez wiele podmiotów, pożądany może być podział CODESPACE na mniejsze ADMINISTRATIVE ZONES oparte na kryterium topograficznym, środka transportu albo innym stosownym do sytuacji takie, że do każdej można przypisać odpowiedzialny podmiot (ORGANISATION). W takim przypadku, wewnątrz jednego CODESPACE, użycie odpowiedniego prefiksu może być zarezerwowane do identyfikatorów obiektów poszczególnych stref. Pozwala to na rozdzielenie kompetencji przydzielania nowych identyfikatorów do zarządców poszczególnych stref w sposób rozproszony. Rola centralnego zarządcy sprowadza się do wyznaczenia stef i przydzielenia odpowiednich prefiksów. Obiekty ADMINISTRATIVE ZONE nie są wymieniane bezpośrednio w EPIP, ale mogą być wskazywane przez zewnętrzne odniesienia w ***ResponsibilitySet***.

***Codespace*** jest definiowane przez ścieżkę używającą domeny internetowej, np. "*tfl.gov.uk*", *"sbb.ch",* "wtp.waw.pl". Ten adres może być zadeklarowany wewnątrz atrybutu ***XmlnsUrl*** w ***Codespace***.

Każde ***Codespace*** posiada przypisany prefiks, który może być używany jako substytut ***Codespace*** wewnątrz dokumentu i który jest zadeklarowany w ***VersionFrame*** za pomocą elementu ***Codespace***. Te prefiksy pozwalają na efektywne kodowanie unikalnych identyfikatorów w dokumencie XML, ponieważ mogą być znacząco krótsze niż kwalifikowany ciąg znaków reprezentujący ***Codespace***. W szczególności prefiks może być inny niż atrybut ***id*** w ***Codespace***, na przykład elementy z ***Codespace*** *"epip_data"* używają prefiksu "epd", a te z *"epip_metadata"* — *"epip".* Technicznie prefiks jest lokalnie definiowany w danym dokumencie i różne prefiksy mogą być używane dla jednego ***Codespace*** w różnych dokumentach, ale użyteczną konwencją jest używanie jednego prefiksu do jednego ***Codespace***.

Wewnątrz ***Codespace*** podmiot dostarczający dane może użyć dowolnie przez siebie wybranego sposobu przedzielania identyfikatorów. Jedynym wymaganiem jest zapewnienie stałego, unikalnego identyfikatora dla każdej osobnej instancji encji wewnątrz ***Codespace***.

Przykładowa deklaracja ***Codespace*** w XML:

```
<CompositeFrame>
  <codespace>
    <Codespace id="pkp_plk">
      <Xmlns>plk</Xmlns>
      <XmlnsUrl>https://www.plk-sa.pl/netex</XmlnsUrl>
      <description>PL PKP PLK</description>
    </Codespace>
    <Codespace id="era_stops">
      <Xmlns>era</Xmlns>
      <XmlnsUrl>https://era.org.eu/stops</XmlnsUrl>
      <Description>European Rail Authority</Description>
    </Codespace>
  </codespace>
</CompositeFrame>
```

Przykłady elementów, które używają powyższych ***Codespace**:*

```
<StopPlace id="plk:5900"/>
```

```
<StopPlaceRef ref="plk:60103"/>
```

```
<StopPlace id="era:5100001"/>
```

Od aplikacji przetwarzających dokumenty profilu oczekiwane jest, aby rozumiały zasady weryfikowania lub interpretowania identyfikatorów w poszczególnych ***Codespace**.* Na przykład identyfikatory stacji nadawane przez PKP PLK zawierają jako ostatnią cyfrę samokontroli potwierdzającą poprawność numeru. Z kolei numery stacji nadawane przez UIC mają stałą długość siedmiu cyfr, z czego pierwsze dwie stanowią unikalny kod kraju (np. 51 — Polska). Walidator XML nie wymusza tych reguł, ale dedykowane aplikacje mogą je implementować, aby zapewnić wyższą jakość danych. W kontekście generowania i parsowania dokumentów NeTEx powyższe zasady mogą być pominięte, kluczowe jest jedynie zapewnienie unikalności identyfikatorów i przypisanie ich wszystkim instancjom.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Entity* | ::> | CODESPACE dziedziczy po ENTITY. |
| «PK» | ***id*** | *CodespaceIdType* | 1:1 | Identyfikator obiektu CODESPACE. Unikalny w ramach wszystkich dokumentów. |
| «AK» | ***Xmlns*** | *xsdNMTOKEN* | 1:1 | Prefiks CODESPACE, unikalny wewnątrz danego dokumentu XML. |
| | ***XmlnsUrl*** | *xsd:anyURI* | 0:1 | Adres URL CODESPACE, globalnie unikalny. |
| | ***Description*** | *xsd:string* | 0:1 | Opis CODESPACE. |

### 7.2.9. Warunek ważności — ValidityCondition

***ValidityCondition*** jest używane do definiowania warunkowej ważności ***DataManagedObject.*** EPIP używa VALIDITY CONDITION tylko jako nazwy; pełny NeTEx pozwala na zaawansowane konstruowanie reguł ważności.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | VALIDITY CONDITION dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *ValidityConditionIdType* | 1:1 | Identyfikator obiektu VALIDITY CONDITION. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis VALIDITY CONDITION. |

### 7.2.10. Ważność pomiędzy — ValidBetween

***ValidBetween*** to uproszczona wersja ***ValidityCondition*** określająca jedynie początkową i końcową datę ważności. Początek albo koniec może zostać pominięty, jeżeli jest nieznany, ale co najmniej jeden kraniec ważności musi być zdefiniowany.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | Dziedziczy po DATA MANAGED OBJECT. |
| | ***FromDate*** | *xsd:dateTime* | 0:1 | Początkowa data ważności AVAILABILITY CONDITION. Rozpoczęcie ważności od wskazanej daty włącznie. |
| | ***ToDate*** | *xsd:dateTime* | 0:1 | Końcowa data ważności AVAILABILITY CONDITION. Zakończenie ważności do wskazanej daty włącznie. |

### 7.2.11. Zbiór odpowiedzialności — ResponsibilitySet

***ResponsibilitySet*** może być zdefiniowany dla ***DataManagedObject****,* aby określić organizacje odpowiedzialne za zarządzanie danymi z podziałem na odpowiednie role, takie jak tworzenie, utrzymywanie, dystrybucja lub licencjonowanie.

EPIP co do zasady zaleca, aby ***ResponsibilitySet*** był definiowany na poziomie VERSION FRAME i dotyczył wszystkich elementów w danej ramce, ale stosowanie osobnych reguł dla poszczególnych elementów jest również możliwe.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | RESPONSIBILITY SET dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *ResponsibilitySetIdType* | 1:1 | Identyfikator obiektu RESPONSIBILITY SET. |
| «AK» | *PrivateCode* | *PrivateCode* | 0:1 | Wewnętrzny kod identyfikujący w sposób unikalny element. Może zostać wykorzystany w komunikacji z innymi systemami. |
| «cntd» *roles* | | *ResponsibilityRoleAssignment* | 1 :* | Obiekty RESPONSIBILITY ROLE ASSIGNMENT tworzące RESPONSIBILITY SET. |

### 7.2.12. Przyporządkowanie roli odpowiedzialności — ResponsibilityRoleAssignment (podkomponent ResponsibilitySet)

*ResponsibilitySet* składa się z co najmniej jednej instancji *ResponsibilityRoleAssignment,* każda z nich przypisuje co najmniej jedną rolę do podmiotu (ORGANISATION) lub jego części (ORGANISATION PART).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VesionedChild* | ::> | RESPONSIBILITY ROLE ASSIGNMENT dziedziczy po VERSIONED CHILD. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | *id* | *ResponsibilityRoleAssignmentIdType* | 1:1 | Identyfikator obiektu RESPONSIBILITY ROLE ASSIGNMENT. |
| «FK» | *ResponsibilitySetRef* | *ResponsibilitySetRef* | 0:1 | Odniesienie do RESPONSIBILITY SET, do którego należy RESPONSIBILITY ROLE ASSIGNMENT. |
| | *Description* | *MultilingualString* | 0:1 | Opis RESPONSIBILITY ROLE ASSIGNMENT. |
| «enum» | *DataRoleType* | *DataRole TypeEnum* | 0:1 | Rodzaj roli danych. W EPIP wybrano podzbiór wartości z NeTEx:<br>• *creates* (pierwotne wytwarzanie danych),<br>• *collects* (zbieranie danych z różnych źródeł bez ingerencji); |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *aggregates* (zbieranie danych z różnych źródeł z działaniami integrującymi je, dodającymi poprawki, itp.). |
| «enum» | ***StakeholderRoleTyp*** *StakeholderRoleTypeEnum* 0:1 ***e*** | | | Rodzaj roli podmiotu. W EPIP wybrano podzbiór wartości z NeTEx:<br>• *Planning* (zazwyczaj zarząd planujący);<br>• *Operation* (podmiot wykonujący działania, np. przewoźnik, zarządc |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | a infrastruktury, właściciel obiektu) |
| «FK» | ***ResponsibleOrganisationRef*** | *OrganisationRef* | 0:1 | Odniesienie do ORGANISATION, do której przypisano RESPONSIBILITY ROLE. |
| «FK» | ***ResponsibleAreaRef*** | *AdministrativeZoneRef* | 0:1 | Odniesienie do ADMINISTRAT IVE ZONE, do której przypisano RESPONSIBILITY ROLE. |

#### 7.2.13. Źródło danych — DataSource

Każde ENTITY jest powiązane z referencją DATA SOURCE, która identyfikuje system, z którego pochodzi dana wersja encji. Zazwyczaj wszystkie elementy w zbiorze danych pochodzą z jednego źródła, więc ***DataSource*** jest definiowane na poziomie VERSION FRAME za pomocą atrybutu ***DefaultDataSource*** w ***VersionFrame / FrameDefaults.***

## 7.3. Relacje i odnośniki do encji

### 7.3.1. Podstawowe zasady

Podczas serializacji danych z innych systemów do dokumentu XML należy zapewnić odwzorowanie relacji pomiędzy encjami, które we wspomnianych systemach mogą mieć formę kluczy w bazie danych lub wskaźników. W dokumentach XML stosuje się co do zasady dwie metody:

1. *Zawieranie* — ciała powiązanych encji są zawarte wewnątrz obiektu nadrzędnego. Takie rozwiązanie dobrze implementuje relacje kompozycji, gdzie jest dobrze zdefiniowania hierarchia elementów. Na przykład perony wewnątrz stacji: każdy peron zawarty jest w dokładnie jednej stacji.
2. *Referencje* — odniesienia, które zrealizowane są za pomocą atrybutu zawierającego identyfikator powiązanego elementu. Stosuje się je w przypadku obiektów, pomiędzy

którymi występuje powiązanie, ale nie jest to prosta hierarchia. W zależności od semantyki powiązania i możliwości przechodzenia po niej, odniesienie (referencja) może być zdefiniowana na jednym lub obu końcach relacji.

#### 7.3.1.1. Przykłady XML

Poniżej przedstawiono przykład zserializowanej informacji o peronach *(StopPlace ^ quays)* na stacji Gdynia Główna. Zwraca uwagę dodatkowy element *quays* grupujący potomne instancje *Quay.*

```
<StopPlace version="8.6" id="plk:5900">
  <quays>
    <Quay version="8.4" id="plk:5900-platform2">
      <Name>Peron 2</Name>
    </Quay>
    <Quay version="8.4" id="plk:5900-platform3">
      <Name>Peron 3</Name>
    </Quay>
  </quays>
</StopPlace>
```

Przykład referencji: powiązanie pomiędzy *StopPlace* a *TopographicPlace* jest realizowane za pomocą atrybutu definiującego odniesienie do powiązanego elementu.

```
<StopPlace version="001" id="plk:5900">
  <name>Gdynia Główna</name>
  <TopographicPlaceRef version="any" ref="teryt:2262011">
    Gdynia
  </ TopographicPlaceRef>
</StopPlace>
```

W celu większej elastyczności schemat NeTEx zakłada, że każdy element potomny zawiera atrybut z referencją do elementu nadrzędnego. W momencie, kiedy definicje potomków są zawarte wewnątrz elementu nadrzędnego, takie odniesienia nie są konieczne i EPIP zaleca, aby w tym przypadku nie były wypełniane.

Poniżej zaprezentowano takie odniesienia, które są zgodne ze schematem, ale nie powinny być serializowane, ponieważ nie stanowią wartości dodanej i tylko zwiększają objętość dokumentu XML.

```
<StopPlace version="8.6" id="plk:5900">
  <quays>
    <Quay version="8.4" id="plk:5900-platform2">
      <Name>Peron 2</Name>
      <ParentZoneRef version="8.6" ref="plk:5900">
        ZBĘDNE
      </ ParentZoneRef>
    </Quay>
    <Quay version="8.4" id="plk:5900-platform3">
      <Name>Peron 3</Name>
```

```
      <ParentZoneRef version="8.6"
        ref="plk:5900"> ZBĘDNE
      </ParentZoneRef >
    </Quay>
  </quays>
</StopPlace>
```

### 7.3.2. Odniesienie do wersji obiektów — VersionOfObjectRef

Wszystkie identyfikatory w odniesieniach w NeTEx nie są zwykłą wartością, lecz są typowane zgodnie z typem, na który wskazują.

Każdy z tych typów jest specjalizacją typu ***VersionOfObjectRef***. odniesienie do obiektu ***Line*** jest typu ***LineRef***, do ***StopPlace_StopPlaceRef*** itd.

Takie podejście pozwala na zapewnienie kontroli typów podczas automatycznego sprawdzania poprawności danych. Dodatkowo typ ***VersionOfObjectRef*** pozwala określić wersję obiektu, której oczekuje dane odniesienie.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***NameOfRefClass*** | *NameOfClass* | 0:1 | Odniesienie do typu obiektu encji, której dotyczy to odniesienie i która jest zdefiniowana w VERSION FRAME. |
| «FK» | ***ref*** | *(ObjectIdType)* + | 1:1 | Odniesienie do prostego identyfikatora encji, która jest zdefiniowana w VERSION FRAME |
| «FK» | ***version*** | *VersionIdType* | 0: 1 | Odniesienie do VERSION encji, która istnieje w VERSION FRAME. Definicja tego atrybutu jest wzajemnie wykluczająca się z ***versionRef***. Jeżeli wersja nie jest znana, należy użyć wartości *"any.* |
| «FK» | ***version Ref*** | *VersionRef* | 0:1 | Odniesienie do VERSION encji, gdy referencja jest zewnętrzna i konkretna wersja jest wymagana. Definicja tego atrybutu jest wzajemnie |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | wykluczająca się z ***version***. Jeżeli wersja nie jest znana, należy użyć wartości *"any'.* |

### 7.3.3. Sprawdzanie integralności dokumentów X ML

Brakujące lub błędne odniesienia w danych stanowią poważny problem w każdym systemie. NeTEx dzięki opisanym wyżej mechanizmom pozwala na automatyczne sprawdzanie i wymuszanie spójności odniesień.

Odniesienia mogą być wewnętrzne (do elementów zdefiniowan ych w tym samym dokumencie) lub zewnętrzne (do obiektów zdefiniowanych w osobnym dokumencie). Podczas definiowania odniesienia rozróżnienie to jest dokonywane przez użycie odpowiednio atrybutu ***version*** albo ***versionRef***.

- Odniesienia używające atrybutu ***version*** mogą zostać sprawdzone automatycznie przez walidator XML na podstawie reguł zdefiniowanych w schemacie danych.
- Dla odniesień zewnętrznych, czyli używających atrybutu ***versionRef*** takie sprawdzenie nie jest wykonywane.

Na przykład poniższe odniesienie zostanie sprawdzona pod kątem istnienia wskazywanego obiektu w przetwarzanym dokumencie:

<**TopographicPlaceRef** version="2.2" ref="teryt:2262011"/>

Natomiast tutaj sprawdzenie zostanie odpuszczone, ponieważ odniesienie wskazuje, że definicja powiązanego elementu znajduje się poza przetwarzanym dokumentem:

<**TopographicPlaceRef** versionRef="2.2" ref="teryt:2262011"/>

W celu poprawienia czytelności i ułatwienia debugowania konwencja stosowana w standardzie zakłada dodawanie adnotacji w ciele elementu opisującej docelowy obiekt, ponieważ nie zawsze identyfikator jasno wskazuje, o jaki konkretnie obiekt chodzi:

<**TopographicPlaceRef** versionRef="2.2" ref="teryt:2262011">
Gdynia
</ **TopographicPlaceRef**>

### 7.3.4. Zakres unikalności identyfikatorów

Wszystkie podklasy ***DataManagedObject*** posiadają ***id***. Wewnątrz jednego dokumentu, wartość tych identyfikatorów powinna być unikalna w ramach klasy elementu. Na przykład, w dokumencie może istnieć tylko jedna instancja ***SiteFrame*** z atrybutem ***id*** równym "xyz". W większości przypadków instancje różnych klas mogą mieć taki sam identyfikator, ale ze względu na czytelność nie jest to zalecane.

Schemat XML normy NeTEx zapewnia powyższe wymagania co do unikalności. Ponad powyższe, pewne klasy mają bardziej restrykcyjne wymagania dotyczące unikalność —

wymagana jest unikalność identyfikatorów instancji w zakresie nadklasy. Na przykład, dowolne PLACE musi mieć unikalny identyfikator, co uniemożliwia istnienie ***PointOfInterest*** i ***StopPlace*** z takim samym ***id.***

W poniższej tabeli przedstawiono klasy używane w EPIP, które mają współdzielony zakres unikalności.

| Nadklasa | Klasy właściwe |
|---|---|
| POINT | ***ScheduledStopPoint, TimingPoint*** |
| LINK | ***ServiceLink, RouteLink, PathLink*** |
| ZONE, PLACE | ***TopographicPlace, Address, TariffZone, StopPlace, Quay, StopPlaceEntrance, PointOfInterest*** |
| LINK SEQUENCE | ***ServicePattern, JourneyPattern*** |
| LINE | ***Line, FlexibleLine*** |
| JOURNEY | ***ServiceJourney, TemplateServiceJourney*** |
| ORGANISATION | ***Operator, Authority*** |
| FACILITY SET | ***SiteFacilitySet, ServiceFacilitySet*** |

Identyfikatory części elementów potomnych są dodatkowo rozróżnialne poprzez szeregujący atrybut ***order****.* Mogą one w takim przypadku współdzielić ***id*** z elementem nadrzędnym. Na przykład ***ServicePattern*** "xyz" może mieć potomne elementy ***StopPointInSequence*** z takim samym identyfikatorem:

```
< ServicePattern id="xyz" version="1.1">
<StopPointInSequence id="xyz" version="1.1" order="1"/>
<StopPointInSequence id="xyz" version="1.1" order="2"/>
<StopPointInSequence id="xyz" version="1.1" order="3"/>
</ServicePattern>
```

Elementy z atrybutem ***order*** powinny być serializowane w kolejności zgodnej z rosnącą wartością tego atrybutu.

## 7.4. Ramki danych

### 7.4.1. Przegląd

***VersionFrame*** zapewnia wersjonowany kontener (ramkę danych), który pozwala na wymianę koherentnych zestawów danych, np. rozkładu jazdy jednej linii, stacji i przystanków osobowych w kraju.

Ramki są ściśle powiązane z informacji o wersjach. Każda instancja VERSION FRAME zawiera co najmniej jeden element ***ValidityCondition****,* który jest używane do wyrażania ważności całej ramki wraz ze wszystkimi obiektami zdefiniowanymi wewnątrz niej. Zobacz: Zarządzanie wersjami.

***VersionFrame*** jest klasą abstrakcyjną. NeTEx zawiera wyspecjalizowane podklasy do konkretnych zastosowań (np. ***ServiceFrame***, ***SiteFrame***), które dotyczą funkcyjnych podzbiorów danych. Dodatkowo zdefiniowana jest również ***GeneralFrame***, która może zawierać dowolne dane.

### 7.4.2. Wersjonowana ramka — VersionFrame (klasa abstrakcyjna)

Abstrakcyjna klasa zawierająca współdzielone atrybuty ramki z wersjonowanymi danymi.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | VERSION FRAME dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *VersionFrameIdType* | 1:1 | Identyfikator obiektu VERSION FRAME. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa VERSION FRAME. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis VERSION FRAME. |
| «FK» | ***TypeOfFrameRef*** | *TypeOfFrameRef* | 0:1 | Odniesienie do TYPE opisywanej VERSION FRAME. |
| «FK» | ***BaselineVersionFrameRef*** | *VersionRef* | 0:1 | Poprzednia podstawowa ramka, której obiekty z bieżącej wymagają. |
| «cntd» | ***codespaces*** | *Codespace* | 0:* | Obiekty CODESPACE używane w bieżącej ramce. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***FrameDefaults*** | *FrameDefaults* | 0:1 | Zbiór domyślnych wartości dla elementów w ramce, dla których nie zapewniono konkretnych wartości. |
| «cntd» | ***prerequisites*** | *VersionFrameRef* | 0:* | Odniesienia do innych ramek, od których bieżąca zależy. |
| «cntd» | ***contentValidityCondition*** *ValidityCondition* ***s*** | | 0:* | Współdzielone VALIDITY CONDITION używane przez elementy w ramce. |

### 7.4.3. Typ ramki — TypeOfFrame

TYPE OF FRAME klasyfikuje VERSION FRAME. Opisuje, jaka jest dozwolona zawartość profilu. EPIP definiuje standardowe instancje ***TypeOfFrame***; każda z nich odpowiada konkretnemu zestawowi danych, który może być wymieniany w dokumentach XML. Identyfikatory są zdefiniowane w ***Codespace*** EPIP z prefiksem *"epip:".*

Dostępne typy ramek:

- ***EU_PI_COMMON****:* wspólne i ogólne typy używane w innych ramkach (np. ***TypeOfValue****,* ***DayType****).*
- ***EU_PI_STOP****:* obiekty niezbędne do opisów STOP PLACE i SITE (np. STOP PLACE, QUAY, ENTRANCE, POINT OF INTEREST).
- ***EU_PI_NETWORK****:* obiekty niezbędne opisu NETWORK (np. LINE, ROUTE, JOURNEY PATTERN).
- ***EU_PI_TIMETABLE****:* obiekty związane z danymi rozkładowymi — czasami (np. VEHICLE JOURNEY, PASSING TIME).
- ***EU_PI_CALENDAR****:* obiekty definiujące kalendarze i typy dni.

Typy *EU_PI_STOP, EU_PI_NETWORK* i *EU_PI_TIMETABLE* zawierają główny wsad danych w wymianie EPIP. *EU_PI_COMMON* i *EU_PI_CALENDAR* zawierają dodatkowe dane słownikowe używane przez poprzednie trzy typy.

*CompositeFrame* jest ramką zawierającą zbiór innych VERSION FRAME, do których zastosowano takie same VALIDITY CONDITION. Jest to zatem mechanizm grupujący. EPIP definiuje trzy złożone ramki:

- *EU_PI_LINE_OFFER:* zawiera wszystkie informacje związane z pojedynczą linią (LINE); składają się na nią dane z ramek: *EU_PI_COMMON, EU_PI_STOP, EU_PI_NETWORK, EU_PI_CALENDAR* oraz *EU_PI_TIMETABLE.*
- *EU_PI_NETWORK_OFFER:* zawiera wszystkie informacje związane z jedną instancją NETWORK (zatem dotyczy wielu linii w jednej sieci); składają się na nią dane z ramek: *EU_PI_COMMON, EU_PI_STOP, EU_PI_NETWORK, EU_PI_CALENDAR* oraz *EU_PI_TIMETABLE.* W ramach głównej ramki może być wiele instancji ramek poszczególnych typów, np. osobna *EU_PI_TIMETABLE* dla każdej linii w sieci.
- **_EU_PI_STOP_OFFER**: zawiera wszystkie informacje związane z obiektami STOP dla wybranego obszaru; składają się na nią dane z ramek: *EU_PI_COMMON* i *EU_PI_STOP*

Szczegółowy opis tych ramek znajduje się w sekcji Dostępne typy ramek danych.

Dodatkowo istnieje jeszcze ramka *EU_PI_METADATA,* która może zostać załączona do dokumentu, aby zdefiniować statyczne metadane, do których mogą się odnosić walidatory XML.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *TypeOfValue* | ::> | TYPE OF FRAME dziedziczy po TYPE OF VALUE. |
| «PK» | *id* | *TypeOfFrameIdType* | 1:1 | Identyfikator obiektu TYPE OF FRAME, powinien być jednym z:<br>• *EU_PI_COMMON,*<br>• *EU_PI_STOP,*<br>• *EU_PI_NETWORK,*<br>• *EU_PI_ TIMETABLE,*<br>• *EU_PI_LINE_ OFFER,*<br>• *EU_PI_NETWORK_OFFER,*<br>• *EU_PI_CALENDAR.* |
| «cntd» | *classes* | *ClassInContextRef* 0:* | | Klasy znajdujące się w ramkach zgodnych z TYPE OF FRAME. |
| «cntd» | *includes* | *TypeOfFrameRef* 0:* | | Lista TYPE OF FRAME zawartych w danym TYPE OF FRAME. Nie może być cykliczne |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | (musi stanowić strukturę drzewiastą). |

7.4.4. Ramka konkretna, ramka ogólna — SpecificFrame, GeneralFrame

Ramka ***VersionFrame*** jest specjalizowana jako konkretne ramki do określonych zbiorów danych (np. ***SiteFrame***, ***ServiceFrame***) lub jako ***GeneralFrame***, która może przetrzymywać dowolne obiekty w heterogenicznej liście.

EPIP używa jedynie ***ResourceFrame***, ***SiteFrame***, ***ServiceFrame***, ***TimetableFrame*** i ***ServiceCalendarFrame***, których opis znajduje się w

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionFrame* | ::> | GENERAL FRAME dziedziczy po VERSION FRAME. |
| «PK» | ***id*** | *GeneralFrameIdType* | 1:1 | Identyfikator obiektu GENERAL FRAME. |
| «cntd» ***members*** *(EntityInFrame)* | | | 0:* | Obiekty należące do GENERAL FRAME: DATA MANAGED OBJECT lub referencje do DATA MANAGED OBJECT. |

7.4.5. Ramka złożona — CompositeFrame

***CompositeFrame*** jest ramką zawierającą zbiór innych VERSION FRAME, do których zastosowano takie same VALIDITY CONDITION. Jest to zatem mechanizm grupujący.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionFrame* | ::> | COMPOSITE FRAME dziedziczy po VERSION FRAME. |
| «PK» | ***id*** | *CompositeFrameIdType* | 1:1 | Identyfikator obiektu COMPOSITE FRAME. |
| «cntd» | ***frames*** | *VersionFrame* | 1:* | Ramki zawarte w opisywanej COMPOSITE FRAME. Mogą być wyłącznie typów: *EU_PI_STOP*, *EU_PI_NETWORK*, *EU_PI_ TIMETABLE*, |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | *EU_PI_COMMON* i *EU PI CALENDAR.* |

### 7.4.6. Wartości domyślne ramki — FrameDefaults

***FrameDefaults*** stanowi zbiór domyślnych wartości dla elementów w ramce, dla których nie zapewniono konkretnych wartości. Użycie takiego mechanizmu zarówno upraszcza dokument XML, jak i zmniejsza jego rozmiar.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***DefaultCodespaceRef*** | *CodespaceRef* | 0:1 | Domyślny obiekt CODESPACE, który należy założyć dla encji w ramce, które explicite nie definiują CODESPACE. |
| «FK» | ***DefaultDataSourceRef*** | *DataSourceRef* | 0:1 | Domyślny obiekt DATA SOURCE, który należy założyć dla encji w ramce, które explicite nie definiują DATA SOURCE. |
| «FK» | ***DefaultResponsibilitySetRef*** | *ResponsibilitySetRef* | 0: 1 | Domyślny obiekt RESPONSIBILITY SET, który należy założyć dla encji w ramce, które explicite nie definiują RESPONSIBILITY SET. |
| «cntd» | ***DefaultLocale*** | *Locale* | 0:1 | Domyślny obiekt LOCALE, który należy założyć dla encji w ramce, które explicite nie definiują LOCALE. |
| | ***DefaultLocationSystem*** | *xsd:normalizedString* | 0: 1 | Domyślny obiekt LOCATION SYSTEM, który |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | należy założyć dla encji w ramce, które explicite nie definiują LOCATION SYSTEM. Zalecany jest WGS84. |
| «enum» | ***DefaultSystemOfUnits*** | *SystemOfUnitsEnum* | 0:1 | Domyślny obiekt SYSTEM OF UNITS, który należy założyć dla atrybutów bez jednostek w ramce. Jedyna dopuszczalna wartość to *SiMetresAndKilometres.* |
| | ***DefaultCurrency*** | *CurrencyType* | 0:1 | Domyślny obiekt CURRENCY, który należy założyć dla atrybutów cen bez jednostek w ramce. |

## 7.5. Elementy ogólne

### 7.5.1. Wprowadzenie

Schemat NeTEx opiera się na zrębie abstrakcyjnych typów ogólnych wziętych z Transmodelu, które są dalej specjalizowane w konkretne elementy XML. To podejście ma na celu uspójnienie częstych konstrukcji i ułatwić utrzymywanie normy i systemów z niej korzystających. Część elementów z Transmodelu jest właściwymi klasami i jest używana w profilu NeTEx. Poniżej znajdują się opisy używanych modeli.

### 7.5.2. Typ wartości — TypeOfValue (klasa abstrakcyjna)

TYPE OF VALUE opisuje prostą wartość opartą na dostępnych kodach. Może być specjalizowania do tworzenia zbiorów wartości do opisywania innych encji, np. TYPE OF POINT, TYPE OF LINK. TYPE OF VALUE może być używane do wymiany metadanych służących do weryfikowania danych.

Normy zalecają stałe, enumeratywne wartości, ale w niektórych przypadkach używane są te dziedziczące po ***TypeOfValue****,* w szczególności tam, gdzie przewiduje się możliwość rozszerzania dostępnych opcji w późniejszym terminie.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManaaedübiect* | ::> | TYPE OF VALUE dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *TypeüfValueldType* | 1:1 | Identyfikator obiektu TYPE OF VALUE. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa TYPE OF VALUE. |
| | *Description* | *MultilingualString* | 0:1 | Opis TYPE OF VALUE. |

### 7.5.3. Zbiór wartości — ValueSet

*ValueSet* jest używany do grupowania wartości należący do danek kategorii (TYPE OF VALUE), tak aby komunikujący się system mógł rozpoznać je jako zbiór dozwolonych wartości.

Na przykład EPIP definiuje dwa typy STOP PLACE: *monomodal* i *general.* Mogą one zostać zakodowane w XML jak poniżej i w ten sposób umożliwić walidację obiektów używających tych wartości.

```
<ValueSet version="1.0" id="epip:Types_of_StopPlace" classOfValues="TypeOfPlace">
  <Name>Type of EPIP Stop Place</Name>
  <values>
    <TypeOfPlace version="1.0" id="epip:general">
      <Name>General stop place</Name>
    </TypeOfPlace>
    <TypeOfPlace version="1.0" id="epip:monomodal">
      <Name>Single mode stop place</Name>
    </TypeOfPlace>
  </values>
</ValueSet>
```

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedübiect* | ::> | VALUE SET dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *ValueSetIdType* | 1:1 | Identyfikator obiektu VALUE SET. |
| «atr» | *classOfValues* | *xsd:NCName* | 0:1 | Nazwa klasy XML dla elementów zawartych w VALUE SET. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa VALUE SET. |
| «cntd» | *values* | *(TypeüfValue)* | 0:* | Dostępne wartości w VALUE SET. |

7.5.4. Grupa encji — GroupOfEntities (klasa abstrakcyjna)

Zbiór ENTITY połączony na podstawie kryterium PURPOSE OF GROUPING, np. grupa przystanków ze wspólną nazwą (zespół przystankowy).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *DataManagedObject* | ::> | GROUP OF ENTITIES dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *GroupOfEntitiesIdType* | 1:1 | Identyfikator obiektu GROUP OF ENTITIES. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu GROUP OF ENTITIES. |
| | ***ShortName*** | *MultilingualString* | 0:1 | Krótka nazwa obiektu GROUP OF ENTITIES. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu GROUP OF |
| «FK» | ***PurposeOfGroupingRef*** | *PurposeOfGroupingRef* | 0:1 | Kryterium, powód grupowania GROUP OF |
| «AK» | ***PrivateCode*** | *PrivateCode* | 0:1 | Wewnętrzny kod powiązany z GROUP OF |
| «cntd» | ***members*** | *(VersionOfObjectRef)+ / (GroupMember)+* | 1:1 | Odniesienie do elementów zawartych w grupie. Dla klas właściwych grup powinny być zdefiniowane odpowiednie klasy tych odniesień: albo w formie prostych odniesień dziedziczących po ***Version OfObjectRef*** lub w przypadku istnienia dodatkowych własności — dziedziczących po ***GroupMember****,* np. ***GroupOfServicesMember***. |

### 7.5.5. Punkt — Point (klasa abstrakcyjna)

Bezwymiarowy węzeł sieci używany do przestrzennego opisu. POINT jest lokalizowany za pomocą LOCATION w zadanym LOCATION SYSTEM.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | POINT dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *PointIdType* | 1:1 | Identyfikator obiektu POINT. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu POINT. |
| «cntd» | ***Location*** | *Location* | 1:1 | Lokalizacja obiektu POINT. |
| | ***PointNumber*** | *xsd:normalizedString* | 0:1 | Arbitralny identyfikator alternatywny. |
| «cntd» | ***projections*** | *Projection* | 0:* | Projekcje obiektu POINT. |

### 7.5.6. Lokalizacja — Location (podkomponent Point) Pozycja

obiektu POINT z referencją do LOCATION SYSTEM.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***srsName*** | *LocatingSystemName Typ e* | 0:1 | Identyfikator GML systemu lokalizacji, zobacz http://www.epsg.org . *WGS84* jest wysoce zalecaną opcją. |
| | ***Longitude*** | *LongitudeType* | 1:1 | Długość geograficzna. |
| | ***Latitude*** | *LatitudeType* | 1:1 | Szerokość geograficzna. |
| | ***Altitude*** | *AltitudeType* | 0:1 | Wysokość nad poziomem morza. |
| «cntd» | ***Coordinates*** | *gml:pos* | 0:1 | Koordynaty GML. |
| | ***Precision*** | *xsd:decimal* | 0:1 | Precyzja koordynatów. |

7.5.7. Połączenie punktów — Link (klasa abstrakcyjna)

LINK to skierowana krawędź — obiekt jednowymiarowy, w opisie sieci odpowiadający za połączenie pomiędzy dwoma POINT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManaaedObiect* | ::> | LINK dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *LinkIdType* | 1:1 | Identyfikator obiektu LINK. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa obiektu LINK. |
| | *Distance* | *DistanceType* | 0:1 | Odległość wzdłuż obiektu LINK. |
| «cntd» | *LineString* | *gml:LineString* | 0:1 | Obiekt GML LineString opisujący geometrię obiektu. |
| «FK» | *(FromPointRef)* | *PointRef* | 1:1 | Odniesienie do początkowego punktu obiektu LINK. |
| «FK» | *(ToPointRef)* | *PointRef* | 1:1 | Odniesienie do końcowego punktu obiektu LINK. |

7.5.8. Sekwencja połączeń punktów — LinkSequence (klasa abstrakcyjna)

LINK SEQUENCE opisuje ścieżkę w sieci, taką jak linia autobusowa. Może zawierać uporządkowaną listę obiektów POINT IN SEQUENCE albo LINK IN SEQUENCE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | LINK SEQUENCE dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *LinkSequenceIdType* | 1:1 | Identyfikator obiektu LINK SEQUENCE. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa obiektu LINK SEQUENCE. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *Distance* | *DistanceType* | 0:1 | Długość (odległość) ścieżki. |
| «cntd» | *(pointsInSequence)* | *(XxxxPointInSequence)* + | 1:* | Uporządkowana lista elementów POINT IN SEQUENCE, z których każdy posiada odniesienie do POINT odpowiedniego typu dla LINK SEQUENCE. |

### 7.5.9. Punkt sekwencji połączeń punktów — PointInLinkSequence (klasa abstrakcyjna)

Obiekt POINT w LINK SEQUENCE wskazujący swoją pozycję w danej LINK SEQUENCE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionedChild* | ::> | POINT IN LINK SEQUENCE dziedziczy po VERSIONED CHILD. |
| «atr» | *order* | *xsdpositiveInteger* | 1:1 | Pozycja POINT IN LINK SEQUENCE na liście. |
| «FK» | *LinkSequenceRef* | *LinkSequenceRef* | 1:1 | Odniesienie do LINK SEQUENCE. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu POINT IN LINK SEQUENCE. |

### 7.5.10. Strefa — Zone (klasa abstrakcyjna)

ZONE definiuje dwuwymiarowy obszar związany z obsługą przez transport publiczny (np. ADMINISTRATIVE ZONE, TARIFF ZONE, ACCESS ZONE).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *GroupOfPoints* | ::> | ZONE dziedziczy po GROUP OF POINTS. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *ZoneIdType* | 1:1 | Identyfikator obiektu ZONE. |
| «cntd» | ***members*** | *PointRef* | 0:* | Lista punktów zawartych w ZONE. |
| «cntd» | ***Centroid*** | *Point* | 0:1 | Centralny POINT leżący w ZONE. |
| «cntd» | ***gml:Polygon*** | *gml:Polygon* | 0:1 | Wielokąt reprezentujący ZONE. |

7.5.11. Środek transportu, podrzędny środek transportu — TransportMode, Submode

TRANSPORT MODE klasyfikuje środki transportu publicznego, np. autobus, tramwaj, metro, pociąg dalekobieżny, prom.

VEHICLE TYPE powinien należeć do dokładnie jednego TRANSPORT MODE. Na przykład TRANSPORT MODE *bus* zawiera dowolny typ autobusu — standardowy, przegubowy, piętrowy, itd.

Linia (LINE) jest co do zasady obsługiwana przez tylko jeden TRANSPORT MODE: linia autobusowa, tramwajowa, metra.

EPIP zezwala na użycie jedynie wyliczonych w normie środków transportu *(VehicleModeEnum):*

- *air* (transport lotniczy),
- *bus* (autobus),
- *coach* (autokar),
- *funicular* (kolej linowo-terenowa),
- *metro* (metro),
- *rail* (transport kolejowy),
- *trolleyBus* (trolejbus),
- *tram* (tramwaj),
- *water* (transport wodny),
- *cableway* (kolej linowa),
- *other* (inny).

SUBMODE pozwala na dalszy podział TRANSPORT MODE.

1. *TelecabinSubmodeEnum*:
    - *telecabin* (kolej gondolowa),
    - *cableCar* (kolej linowa),
    - *lift* (dźwig osobowy),
    - *chairLift* (wyciąg krzesełkowy),
    - *dragLift* (wyciąg orczykowy),
    - *telecabinLink* (połączenie kolei gondolowych).

2. *AirSubmodeEnum:*
    - *internationalFlight* (lot międzynarodowy),
    - *domesticFlight* (lot krajowy),
    - *intercontinentalFlight* (lot międzykontynentalny),
    - *domesticScheduledFlight* (rozkładowy lot krajowy),
    - *shuttleFlight* (lot wahadłowy),
    - *intercontinentalCharterFlight* (czarterowy lot międzykontynentalny),
    - *internationalCharterFlight* (czarterowy lot międzynarodowy),
    - *roundTripCharterFlight* (czarterowy lot okrężny),
    - *sightseeingFlight* (lot widokowy),
    - *helicopterService* (przelot śmigłowcem),
    - *domesticCharterFlight* (czarterowy lot krajowy),
    - *SchengenAreaFlight* (lot w obrębie strefy Schengen),
    - *airshipService* (lot sterowcem),
    - *shortHaulInternationalFlight* (krótki lot międzynarodowy).
3. *WaterSubmodeEnum:*
    - *internationalCarFerry* (międzynarodowy prom samochodowy),
    - *nationalCarFerry* (krajowy prom samochodowy),
    - *regionalCarFerry* (regionalny prom samochodowy),
    - *localCarFerry* (lokalny prom samochodowy),
    - *internationalPassengerFerry* (międzynarodowy prom pasażerski),
    - *nationalPassengerFerry* (krajowy prom pasażerski),
    - *regionalPassengerFerry* (regionalny prom pasażerski),
    - *localPassengerFerry* (lokalny prom pasażerski),
    - *postBoat* (łódź pocztowa),
    - *trainFerry* (prom kolejowy),
    - *roadFerryLink* (przeprawa promowa w ciągu drogi),
    - *airportBoatLink* (wodne połączenie z lotniskiem),
    - *highSpeedVehicleService* (prom samochodowy dużych prędkości),
    - *highSpeedPassengerService* (prom pasażerski dużych prędkości),
    - *sightseeingService* (łódź widokowa),
    - *schoolBoat* (łódź szkolna),
    - *cableFerry* (prom linowy),
    - *riverBus* (autobus rzeczny),
    - *scheduledFerry* (prom rozkładowy),
    - *shuttleFerryService* (prom wahadłowy),
    - *canalBarge* (barka kanałowa).
4. *BusSubmodeEnum:*
    - *localBus* (autobus lokalny),
    - *regionalBus* (autobus regionalny),

- *expressBus* (autobus ekspresowy),
- *nightBus* (autobus nocny),
- *postBus* (autobus pocztowy),
- *specialNeedsBus* (autobus dla osób ze specjalnymi wymaganiami),
- *mobilityBus* (autobus dla osób z ograniczoną mobilnością),
- *mobilityBusForRegisteredDisabled* (autobus dla zarejestrowanych osób z ograniczoną mobilnością),
- *sightseeingBus* (autobus widokowy),
- *shuttleBus* (autobus wahadłowy),
- *schoolBus* (autobus szkolny),
- *schoolAndPublicServiceBus* (autobus szkolny z dozwolonym przewozem innych osób),
- *railReplacementBus* (autobus kolejowej komunikacji zastępczej),
- *demandAndResponseBus* (autobus na żądanie),
- *airportLinkBus* (autobus dowozowy na lotnisko).
- *highFrequencyBus* (autobus kursujący z wysoką częstotliwością),
- *dedicatedLaneBus* (autobus kursujący po wydzielonych pasach).

5. *FunicularSubmodeEnum:*
   - *funicular* (kolej linowo-terenowa),
   - *streetCableCar* (tramwaj linowy),
   - *allFunicularServices* (dowolna usługa koleją linowo-terenową).
6. *TramSubmodeEnum:*
   - *cityTram* (tramwaj miejski),
   - *localTram* (tramwaj lokalny),
   - *regionalTram* (tramwaj regionalny),
   - *sightseeingTram* (tramwaj widokowy),
   - *shuttleTram* (tramwaj wahadłowy),
   - *trainTram* (tramwaj dwusystemowy).
7. *SelfDriveSubmodeEnum:*
   - *hireCar* (wypożyczony samochód),
   - *hireVan* (wypożyczony samochód dostawczy),
   - *hireMotorbike* (wypożyczony motocykl),
   - *hireCycle* (wypożyczony rower),
   - *allHireVehicles* (dowolny wypożyczony pojazd).
8. *MetroSubmodeEnum:*
   - *metro* (metro),
   - *tube* (metro głębokie, małoprofilowe),
   - *urbanRailway* (kolej miejska).
9. *RailSubmodeEnum:*
   - *local* (pociąg lokalny),

- *highSpeedRail* (pociąg dużych prędkości),
- *suburbanRailway* (pociąg podmiejski),
- *regionalRail* (pociąg regionalny),
- *interregionalRail* (pociąg międzyregionalny),
- *longDistance* (pociąg dalekobieżny),
- *international* (pociąg międzynarodowy),
- *nightTrain* (pociąg nocny),
- *sleeperRailService* (pociąg hotelowy),
- *carTransportRailService* (pociąg z przewozem samochodów osobowych),
- *touristRailway* (pociąg turystyczny),
- *railShuttle* (pociąg wahadłowy),
- *replacementRailService* (pociąg zastępczej komunikacji kolejowej),
- *specialTrain* (pociąg specjalny),
- *airportLinkRail* (pociąg lotniskowy),
- *crossCountryRail* (pociąg przecinający kraj),
- *rackAndPinionRailway* (kolej zębata).

10. *SnowAndIceSubmodeEnum:*
    - *snowMobile* (skuter śnieżny),
    - *snowCat* (ratrak),
    - *snowCoach* (autobus gąsienicowy poruszający się po śniegu),
    - *terraBus* (autobus na ogumionych kołach poruszający się po śniegu),
    - *windSled* (sanie wiatrowe).
11. *TaxiSubmodeEnum*:
    - *communalTaxi* (taksówka komunalna),
    - *waterTaxi* (taksówka wodna),
    - *railTaxi* (taksówka szynowa),
    - *bikeTaxi* (taksówka rowerowa),
    - *blackCab* (czarna taksówka — w Londynie),
    - *miniCab* (mała taksówka — w Londynie, w odróżnieniu od *blackCab),*
    - *charterTaxi* (wynajęta taksówka),
    - *allTaxiServices* (usługa dowolna taksówką).
12. *CoachSubmodeEnum:*
    - *internationalCoach* (autokar międzynarodowy),
    - *nationalCoach* (autokar krajowy),
    - *shuttleCoach* (autokar wahadłowy),
    - *regionalCoach* (autokar regionalny),
    - *specialCoach* (autokar specjalny),
    - *schoolCoach* (autokar szkolny),
    - *sightseeingCoach* (autokar widokowy),
    - *touristCoach* (autokar turystyczny),

- *commuterCoach* (autokar dowozowy).

### 7.5.12. Przypisanie — Assignment (klasa abstrakcyjna)

ASSIGNMENT definiuje zestaw własności, które mają zostać powiązane z inną ENTITY w sposób tymczasowy lub stały. Na przykład STOP ASSIGNMENT jest używane do powiązania przystanku w rozkładzie jazd z fizycznym przystankiem.

| Klasyfikacja ::> | Nazwa ::> | Typ *DataManagedObject* ::> | Kardynalność | Opis ASSIGNMENT dziedziczy po DATA MANAGED OBJECT. |
|---|---|---|---|---|
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu ASSIGNMENT. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu ASSIGNMENT. |

### 7.5.13. Prezentacja — Presentation (podkomponent ogólny)

PRESENTATION definiuje w sposób ujednolicony zestaw wartości kontrolujących wygląd elementu (na mapie, schemacie, wydruku, wyświetlaczu). Może zostać powiązany z LINE, TARIFF ZONE i innymi encjami.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***Colour*** | *ColourValueType* | 0:1 | Kolor RGB do użycia w interfejsie użytkownika. |
| | ***ColourName*** | *xsd:String* | 0:1 | Tekstowa nazwa koloru do użycia w interfejsie użytkownika. |
| | ***BackgroundColour*** | *ColourValueType* | 0:1 | Kolor RGB do użycia jako tło w interfejsie użytkownika. |
| | ***BackgroundColourName*** | *xsd:String* | 0:1 | Tekstowa nazwa koloru do użycia jako tło w interfejsie użytkownika. |
| | ***TextColour*** | *ColourValueType* | 0:1 | Kolor RGB do użycia jako kolor tekstu w interfejsie użytkownika |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***TextColourName*** | *xsd:String* | 0:1 | Tekstowa nazwa koloru do użycia jako kolor tekstu w interfejsie użytkownika. |
| | ***TextFont*** | *xsd:normalizedString* | 0:1 | Identyfikator kroju pisma. |
| | ***TextFontName*** | *xsd:normalizedString* | 0:1 | Nazwa kroju pisma. |
| «cntd» | ***infoLinks*** | *InfoLink* | 0:* | URL obrazu powiązanego z obiektem, np. ikony. |
| | ***ColourSystem*** | *xsd:String* | 0:1 | Nazwa systemu nazw kolorów używanych w atrybutach ***ColourName, BackgroundColourName*** i ***TextColourName.*** Np. *RAL* albo *Pantone****.*** |

7.5.14. Ustawienia regionalne — Locale (podkomponent ogólny)

Własności definiujące dane dotyczącego kraju lub regionu (strefa czasowa, język).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***TimeZoneOffset*** | *TimeZoneOffset* | 0:1 | Przesunięcie strefy czasowej wyrażone w godzinach różnic w stosunku do czasu GMT. |
| | ***TimeZone*** | *xsd:normalizedString* | 0:1 | Nazwa strefy czasowej. |
| | ***SummerTimeZoneOffset*** | *TimeZoneOffset* | 0:1 | Przesunięcie letniej strefy |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | czasowej wyrażone w godzinach różnic w stosunku do czasu GMT. |
| «enum» | ***DefaultLanguage*** | *xsd:language* | 0:1 | Domyślny język (kod ISO 639-1, np. *pl, en, de).* |
| «enum» | ***languages*** | *LanguageUsageEnum* | 0:* * | Inne języki dostępne w LOCALE. |

7.5.15. Ustalenia zamawiania — BookingArrangements (podkomponent ogólny)

Szczegóły rezerwowania (zgłaszania) LINE lub SERVICE JOURNEY w przypadku, kiedy są one transportem na żądanie.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***BookingContact*** | *Contact* | 0:1 | Szczegóły kontaktu dla utworzenia rezerwacji. |
| «enum» | ***BookingMethods*** | *BookingMethodEnum* | 0:* | Metoda rezerwacji dla FLEXIBLE LINE. Dostępne wartości:<br>• *callDriver* (telefon do prowadzącego),<br>• *callOffice* (telefon do biura),<br>• *online* (online). |
| «enum» | ***BookingAccess*** | *BookingAccessEnum* | 0:1 | Kto może zrobić rezerwację? Dostępne wartości: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *other* (inne). |
| «enum» | *BookWhen* | *PurchaseWhenEnum* | 0:1 | Kiedy może być zrobiona rezerwacja. Dostępne wartości:<br>• *advanceOnly* (tylko z wyprzedzeniem),<br>• *untilPreviousDay* (tylko dnia poprzedzającego),<br>• *dayOfTravelOnly* (tylko w dniu podróży),<br>• *advanceAndDayOfTravel* (z wyprzedzeniem oraz w dniu podróży),<br>• *time OfTra velOnly* (tylko w czasie podróży),<br>• *subscriptionChargeMoment* (podczas zakupu biletu okresowego),<br>• *other* (inne). |
| «enum» | *BuyWhen* | *PurchaseMomentEnum* | 0:* | Kiedy może być zrobiony zakup. Dostępne wartości:<br>• *inAdvanceOnly* (tylko z wyprzedzeniem),<br>• *onReservation* (podczas rezerwacji),<br>• *beforeBoarding* (przed wsiadaniem),<br>• *beforeBoardingOnl y* (tylko przed wsiadaniem), |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *onBoarding* (podczas wsiadania),<br>• *onBoardingOnly* (tylko podczas wsiadania),<br>• *afterBoarding* (po wejściu na pokład),<br>• *onCheckIn* (podczas meldowania),<br>• *onCheckOut* (podczas wymeldowania),<br>• *other* (inne). |
| | ***LatestBookingTim*** *MultilingualString* ***e*** | | 0:1 | Najpóźniejszy czas w ciągu dnia w którym można dokonać rezerwacji. |
| | ***MinimumBooking*** *xsd:duration* ***Period*** | | 0:1 | Minimalny odstęp czasu przed dniem lub godziną wyjazdu, w którym można zamówić usługę. |
| | ***BookingUrl*** | *xsd:anyURI* | 0:1 | Adres URL do rezerwacji. |
| | ***BookingNote*** | *MultilingualString* | 0:1 | Notatka dotycząca rezerwacji FLEXIBLE LINE. |

7.5.16. Informacje kontaktowe — ContactDetails (podkomponent ogólny) Szczegóły kontaktu z ORGANISATION.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***Email*** | *EmailAddressType* | 0:1 | Adres e-mail w formacie ISO. |
| | ***Phone*** | *PhoneType* | 0:1 | Numer telefonu. |
| | ***Url*** | *xsd:anyURI* | 0:1 | Adres strony internetowej. |

## 7.6. Dostępność

NeTEx używa jednolitego zestawu komponentów do opisu dostępności miejsc i pojazdów za pomocą ACCESSIBILITY ASSESSMENT. Użycie tego mechanizmu pozwala na obliczenie

przecięcia usług na różnych etapach podróży i zaprezentowania wspólnych dostępnych udogodnień.

Obiekt ACCESSIBILITY ASSESSMENT opisuje części SITE, LINE lub SERVICE JOURNEY; złożony jest z obiektów ACCESSIBILITY LIMITATION, z których każdy opisuje dostępność dotyczącą konkretnej potrzeby, np. *Wheelchair* (poruszanie się na wózku), *StepFree* (bez stopni), *Escalator free* (bez schodów ruchomych), *Lift free* (bez dźwigów osobowych).

Należy rozróżnić sytuację braku danych o dostępności od braku dostępności.

Przy ustalaniu wartości atrybutów dla obiektów STOP PLACE (podklasa SITE), należy kierować się następującymi zasadami:

- STOP PLACE jest dostępne (wartość *true)* zgodnie z danym kryterium, jeżeli istnieje co najmniej jedna dostępna droga dojścia do każdej instancji QUAY wewnątrz danego STOP PLACE;
- STOP PLACE nie jest dostępne (wartość *false*) zgodnie z danym kryterium, jeżeli istnieje co najmniej jedna instancja QUAY wewnątrz danego STOP PLACE, do której nie istnieje żadna dostępna droga dojścia;
- w przypadku braku dokładnych danych należy użyć wartości *unknown.*

Przy wartościach dla obiektów SITE COMPONENT (w szczególności QUAY) wewnątrz STOP PLACE zasady są analogiczne:

- SITE COMPONENT jest dostępny zgodnie z danym kryterium, jeżeli istnieje co najmniej jedna dostępna droga dojścia do wyjścia z obiektu;
- SITE COMPONENT nie jest dostępny zgodnie z danym kryterium, jeżeli nie istnieje żadna dostępna droga dojścia do wyjścia z obiektu;
- w przypadku braku dokładnych danych należy użyć wartości *unknown.*

### 7.6.1. Przyporządkowanie informacji o dostępnoś ci — AccessibilityAssessment (podkomponent ogólny)

Ocena dla ułatwień dostępu pod kątem ograniczeń środowiskowych lub udogodnień (SUITABILITY) dla jednego lub większej liczby USER NEED względem komponentu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManaaedObject* | ::> | ACCESSIBILITY ASSESSMENT dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *AssessmentIdType* | 0:1 | Identyfikator obiektu ACCESSIBILITY ASSESSMENT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *MobilityimpairedAccess* | *LimitationStatusEnum* | 1:1 | Czy w ogólnej ocenie istnieje dostęp dla osób z ograniczoną mobilnością. Dostępne wartości:<br>• *true* (prawda),<br>• *false* (fałsz)<br>• *partial* (częściowe),<br>• *unknown* (nieznane) |
| | «cntd» *limitations* | *AccessibilityLimitation* | 0:1 | Ograniczenia (LIMITATION) składające się na ACCESSIBILITY ASSESSMENT. |

### 7.6.2. Ograniczenie dostępności — AccessibilityLimitation (podkomponent AccessibilityAssessment)

Kategoryzacja charakterystyki ACCESSIBILITY w celu wskazania użyteczności dla pasażerów ze specyficznymi potrzebami, dla przykładu, osób wymagających dostępu dla wózków inwalidzkich, dostępu bez schodów lub tych chcących uniknąć ciasnych pomieszczeń takich jak windy.

Stosuje się niewielką liczbę dobrze zdefiniowanych kategorii, które są wybierane w celu umożliwienia spójnego przechwytywania danych i wydajnego obliczania tras dla różnych klas użytkowników.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> ::> | | *VersionedChild* | | ACCESSIBILITY LIMITATION dziedziczy z VERSIONED CHILD. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *LimitationIdType* | 0:1 | Identyfikator ACCESSIBILITY LIMITATION. |
| «enum» | ***WheelchairAccess*** | *LimitationStatusEnum* | 1:1 | Czy istnieje dostęp dla osób poruszających się na wózku inwalidzkim. Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |
| «enum» | ***StepFreeAccess*** | *LimitationStatusEnum* | 0:1 | Czy istnieje dostęp dla osób wymagających dostępu bez schodów. Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |
| «enum» | ***EscalatorFreeAccess*** | *LimitationStatusEnum* | 0:1 | Czy istnieje dostęp bez użycia schodów ruchomych. Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |
| «enum» | ***LiftFreeAccess*** | *LimitationStatusEnum* | 0:1 | Czy istnieje dostęp bez użycia windy. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |
| «enum» | ***AudibleSignsAvailabl*** | *LimitationStatusEnu* ***e*** *m* | 0:1 | Czy dostępne są oznaczenia dźwiękowe. Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |
| «enum» | ***VisualSignsAvailable*** | *LimitationStatusEnu m* | 0:1 | Czy dostępne są oznaczenia wizualne. Wartości zamieszczone w tabeli powyżej. Przy braku wartości przyjmujemy *unknown* |

Wartości, które dostępne są w **LimitationStatus** to: *true* (prawda), *false* (fałsz), *unknown* (nieznane) lub *partial* (częściowe); wartość *partial* może mieć różne znaczenia:

- *Temporal,* dla przykładu: dostęp dla wózka inwalidzkiego zależny jest od osobistej obsługi, dostępnej jedynie w wyznaczonych porach lub dniach;
- *Geographic,* dla przykładu: stacja gdzie istnieje dostęp dla wózków inwalidzkich, jednak tylko dla wybranych QUAY (nie wszystkich);
- *Contextual,* dla przykładu: zapowiedzi megafonowe tylko w przypadku zakłóceń, lub dla wybranej klasy usług.

W kontekście EPIP każdorazowo, gdy użyta zostaje wartość *partial,* powinno zostać zamieszczone ***ValidityCondition / Description*** w celu wyjaśnienia, dlaczego dostępność jest tylko częściowa. Zawarte informacje tekstowe powinny być odpowiednie do zaprezentowania użytkownikom bez dalszego przetwarzania.

## 7.7. Zbiory usług — Facility sets

### 7.7.1. Przegląd

Informacja pasażerska może zawierać szczegóły dotyczące usług dostępnych na przystanku lub na pokładzie pojazdu. FACILITY to nazwa takiej usługi, np. *buffetCar, leftLuggage.* NeTEx udostępnia stały zestaw zestandaryzowanych wartości dla słowników usług, które opisane są w dodatku A.

FACILITY SET opisuje zestaw usług, który może zostać powiązany z ENTITY, może zostać do niego przyporządkowane również VALIDITY CONDITION, aby wskazać ograniczenia w ważności.

FACILITY SET jest specjalizowany przez SITE FACILITY SET dla instancji ***StopPlace*** oraz SERVICE FACILITY SET dla instancji ***ServiceJourney*** lub ***JourneyPart****.*

### 7.7.2. Zbiór usług — FacilitySet (klasa abstrakcyjna)

FACILITY SET określa ogólne właściwości SITE FACILITY SET oraz SERVICE FACILITY SET.

Wewnątrz instancji FACILITY SET określone wartości są koniunkcją, dla przykładu: ' ***FareClass***_ = firstClass_' i ***CateringFacility***_ = restaurant_'.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| XGRP | ***CommonFacilityGroup*** | *(xxxFacilityList)* | Nx(0:*) | Wybór z ***FacilityLists***; każda ***FacilityList*** pozwala na wybranie jednego lub więcej FACILITY (każda zakodowana jest jako wyliczeniowa wartość) dla konkretnego tematu. Zobacz dodatek A. Ogólne listy mogą być użyte w każdym FACILITY SET. Dostępne są dodatkowe udogodnienia, które są specyficzne dla SERVICE FACILITY SET i SITE FACILITY SET. |

### 7.7.3. Zbiór usług kursu — ServiceFacilitySet

SERVICE FACILITY SET opisuje udogodnienia (FACILITY) dostępne na pokładzie podczas usługi. Dodatkowo pozwala na opisanie typów ACCOMMODATION (miejsce siedzące, do sypialne, itp.)

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *FacilitySet* | ::> | SERVICE FACILITY SET dziedziczy po FACILITY SET. |
| «PK» | *id* | *ServiceFacilitySetIdType* | 1:1 | Identyfikator obiektu SERVICE FACILITY SET. |
| XGRP | *ServiceFacilityGroup* | *xmlGmup of choices* | 0:1 | Udogodnienia specyficzne dla SERVICE FACILITY SET. Zobacz dodatek |
| «cntd» | *accommodations* | *Accommodation* | 0:* | Dostępne ACCOMMODATION w SERVICE FACILITY SET. |

### 7.7.4. Zakwaterowanie — Accommodation (podkomponent ServiceFacilitySet)

Opis charakterystyki klasy i rodzaju miejsca dostępnego podczas usługi, na przykład: *"Wagon z miejscami do leżenia pierwszej klasy z prysznicem".*

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionedChild* | ::> | ACCOMMODATION dziedziczy po VERSIONED CHILD. |
| «PK» | *id* | *AccommodationIdType* | 1:1 | Identyfikator obiektu ACCOMMODATION. |
| «enum» | *FareClass* | *FareClassEnum* | 0:1 | FARE CLASS dla ACCOMMODAT |

| Klasyfikacja | Nazwa | Typ | Kardynaln ość | Opi |
|---|---|---|---|---|
| | | | | ION. Zobacz dodatek A dla sprawdzenia dostępnych wartości. |
| «enum» | *AccommodationFacility* | *AccommodationFacilityEnum* | 0:1 | Rodzaj miejsca. Zobacz dodatek A dla sprawdzenia dostępnych wartości. |
| «enum» | *NuisanceFacility* | *NuisanceFacilityEnum* | 0:1 | Typ uciążliwego obiektu w ACCOMMODATION. Zobacz dodatek A dla sprawdzenia dostępnych wartości. |
| «enum» | *PassengerCommsFacility* | *PassengerCommsFacilityEnum* | 0:1 | Udogodnienia komunikacyjne w obiekcie umiejscowienia. Zobacz dodatek A dla sprawdzenia dostępnych wartości. |

### 7.7.5. Zbiór usług miejsca — SiteFacilitySet

SITE FACILITY SET opisuje obiekty (FACILITY) dostępne w stałej lokalizacji.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *FacilitySet* | ::> | SITE FACILITY SET dziedziczy po FACILITY SET. |
| «PK» | *id* | *SiteFacilitySetIdType* | 1:1 | Identyfikator obiektu SITE FACILITY SET. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| XGRP | *SiteFacilityGroup* | *xmlGroup* | 0:1 | Udogodnienia specyficzne dla SITE FACILITY SET. Zobacz dodatek A. |

# 8. PROFIL PRZYSTANKÓW — STOP PROFILE

## 8.1. Przegląd

Profil przystanków (miejsc postoju, Stop Profile) jest częścią PL_NeTEx_Profile odpowiedzialną za opis przystanków (stacji) dowolnego środka transportu publicznego wraz z ich właściwościami.

Centralnym pojęciem jest tutaj STOP PLACE. Zgodnie z Transmodelem ta encja dziedziczy po wielu innych uszeregowanych w dość głębokiej hierarchii. Do zrozumienia modelu należy mieć podstawową wiedzę na temat tej hierarchii.

Na samej górze drzewa dziedziczenia znajduje się encja ZONE, która opisuje ogólny dwuwymiarowy obiekt. Może ona być zdefiniowana jako GROUP OF POINTS należąca do ZONE albo jako geometryczny obszar ograniczony wielokątem (na podstawie GML Linear Ring).

ZONE może zawierać inne mniejsze encje ZONE (możliwe jest wielopoziomowe zagnieżdżenie).

Drugim poziomem jest PLACE, które reprezentuje dowolne znaczące miejsce z punktu widzenia modelu transportu. PLACE może być specjalizowane przez TOPOGRAPHIC PLACE (nazwana miejscowość albo obszar, jednostka administracyjna) albo ADDRESSABLE PLACE posiadające ADDRESS, który przyjmuje jedną z dwóch form: ROAD ADDRESS albo POSTAL ADDRESS.

SITE ELEMENT specjalizuje ADDRESSABLE PLACE poprzez dodanie informacji od dostępności (zobacz ACCESSIBILITY ASSESSMENT) i innych własności typowych dla miejsc, po których poruszają się podróżni.

SITE specjalizuje SITE ELEMENT, dodając opis własności dotyczących fizycznego miejsca, takiego jak stacja lub punkt zainteresowania. Te własności to na przykład wejścia, piętra, wyposażenia, dostępne drogi i dostępność. Podklasami SITE są na przykład STOP PLACE, POINT OF INTEREST, PARKING. SITE składa się z następujących elementów:

- SITE COMPONENT: opisuje część obiektu SITE, taką jak peron, hala dworcowa, hala peronowa, obszar handlowy. Powyższe
- mogą być specjalizowane przez odpowiednie podklasy (np. QUAY).
- LEVEL: określone piętro (poziom) budynku, na którym znajdują się encje SITE COMPONENT.
- ENTRANCE: fizyczne wejście/wyjście z SITE. Każda podklasa SITE ma zdefiniowaną podklasę wejścia (np. STOP PLACE ENTRANCE).

STOP PLACE opisuje fizyczny punkt dostępu do transportu, taki jak przystanek albo stacja. Dla złożonego obiektu, takiego jak duża stacja kolejowa, model przewiduje opis wszystkich elementów składowych: wejścia, hale, perony, piętra. Pozwala również na szczegółowy opis dostępności dla poszczególnych elementów lub całości.

STOP PLACE reprezentuje fizyczny przystanek lub stację. Należy zauważyć, że to pojęcie różne od zatrzymania wyszczególnionego w rozkładzie jazdy, które reprezentuje encia SCHEDULED STOP POINT. Powyższe mogą zostać powiązane poprzez STOP ASSIGNMENT.

Fizycznie, SCHEDULED STOP POINT może odpowiadać albo całemu STOP PLACE, albo konkretnemu QUAY (np. rozkład może przypisywać ogólnie stację lub konkretny peron). STOP PLACE albo QUAY mogą być rozpatrywane w różnych kontekstach funkcjonalnych i być przypisanymi do wielu różnych SCHEDULED STOP POINT (zobacz Przypisanie punktu zatrzymania).

Fizyczny punkt dostępu do środka transportu zawsze odbywa się w QUAY. STOP PLACE ENTRANCE opisuje wejścia do STOP PLACE.

W związku z odziedziczeniem z ZONE relacji zawierania STOP PLACE mogą być zorganizowane w hierarchię, aby opisać skomplikowane węzły przesiadkowe, takie jak wspólne pętle autobusowo-tramwajowe lub powiązane stacje metra i kolei.

QUAY mogą być zagnieżdżone, co pozwala na reprezentację wielu krawędzi peronowych na jednym peronie albo wyznaczenie nazwanych sektoró w.

STOP PLACE ENTRANCE zazwyczaj opisuje miejsce, w którym pasażer może dostać się do miejsca zatrzymania pieszo. ACCESS MODE może zostać użyty, aby opisać inne dozwolone formy przemieszczenia się takie jak rower lub samochód.

## 8.2. Hierarchia

Definicja przystanków pozwala na tworzenie dowolnie głęboko zagnieżdżonych struktur. W celu zwiększenia zgodności z istniejącymi systemami EPIP zaleca nałożenie pewnych wymagań dotyczących tej struktury:

- dozwolone są maksymalnie dwie warstwy STOP PLACE;
- każdy poziom ma określoną semantykę:
  - ogólny STOP PLACE (EPIP "General") może zawierać encje typu QUAY lub STOP PLACE dla pojedynczego środka transportu (EPIP "Monomodal");
  - STOP PLACE jednego środka transportu (EPIP "Monomodal") powinien mieć co najmniej jedną encję QUAY i nie może zawierać innych STOP PLACE; wszystkie QUAY powinny być używane przez ten sam środek transportu;
- istnieje niejawne SITE CONNECTION pomiędzy wszystkimi QUAY zawartymi w jednym STOP PLACE, co oznacza, że zawsze istnieje możliwość przesiadki pomiędzy dowolnymi dwoma QUAY w ramach jednego STOP PLACE.

Poniższa tabela przedstawia mapowanie pomiędzy środkami transportu a dostępnymi typami QUAY zgodne z EPIP.

| Typ QUAY | Inne kompatybilne typy QUAY | Możliwe środki transportu |
|---|---|---|
| *railPlatform* (peron kolejowy) | | *rail, intercityRail, urbanRail* |
| *metroPlatform* (peron metra) | | *metro, funicular* |
| *tramPlatform* (peron tramwajowy) | *tramStop* | *tram* |

| Typ QUAY | Inne kompatybilne typy QUAY | Możliwe środki transportu |
|---|---|---|
| *tramStop* (przystanek tramwajowy) | *tramPlatform* | *tram* |
| *busStop* (przystanek autobusowy) | *busBay* | *bus, coach, trolleyBus* |
| *busBay* (zatoka autobusowa) | *busStop* | *bus, coach, trolleyBus* |
| *boatQuay* (nabrzeże) | *ferryLanding* | *water* |
| *ferryLanding* (nabrzeże promowe) | *boatQuay* | *water* |
| *telecabinePlatform* (peron koeli linowej) | - | *cableway* |
| *airlineGate* (braka lotniskowa) | - | *air* |

Dla bardzo dużych grup przystanków, np. w ramach węzła przy głównym dworcu kolejowym albo oddalonych od siebie terminali portu lotniczego, może zostać zdefiniowana dodatkowa warstwa organizacji: GROUP OF STOP PLACES.

## 8.3. Inne rodzaje miejsc

Poza STOP PLACE standard definiuje również inne encje miejsc (PLACE), które mają zastosowanie przy ustalaniu lokalizacji w wyszukiwarkach połączeń.

TOPOGRAPHIC PLACE reprezentuje nazwaną miejscowość lub jednostkę administracyjną, do której mogą zostać przyporządkowane znajdujące się tam obiekty STOP PLACE.

TOPOGRAPHIC PLACE może zostać umiejscowione wewnątrz COUNTRY, a dwa różne TOPOGRAPHIC PLACE mogą się nakładać na siebie lub zawierać się wewnątrz innego TOPOGRAPHIC PLACE, mogą zostać również zdefiniowane oficjalne nazwy w różnych językach urzędowych.

POSTAL ADDRESS i ROAD ADDRESS mogą zostać powiązane z SITE albo jego częścią (SITE ELEMENT), aby zintegrować wyszukiwanie przystanków z systemami nawigacji.

POINT OF INTEREST reprezentuje znane obiekty takie jak muzea, parki i atrakcje turystyczne.

### 8.3.1. Pozycja — Place (klasa abstrakcyjna)

PLACE definiuje miejsce, które może być warte opisane z punktu widzenia modelu transportu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Zone* | ::> | PLACE dziedziczy po ZONE. |
| «cntd» | *placeTypes* | *TypeOfPlaceRef* | 0:* * | Kategoria PLACE. Zgodnie z EPIP miejsca powinny mieć przydzielone jedną z dwóch wartości:<br>• 'epip:monomodal',<br>• *'epip:general'.* |

8.3.2. Pozycja z adresem — AddressablePlace (klasa abstrakcyjna)

ADDRESSABLE PLACE to PLACE posiadające adres, który może być w jednej z dwóch postaci: ROAD ADDRESS albo POSTAL ADDRESS.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Place* | ::> | ADDRESSABLE PLACE dziedziczy po PLACE. |
| «PK» | *id* | *AddressablePlaceIdType* | 1: 1 | Identyfikator obiektu ADDRESSABLE PLACE. |
| | *Uri* | *xsd:anyURI* | 0:1 | Domyślny adres URL dla ADDRESSABLE PLACE. |
| | *Image* | *xsd:anyURI* | 0:1 | Domyślny obraz (zdjęcie) dla ADDRESSABLE PLACE. |
| «cntd» *PostalAddress* | | *PostalAddress* | 0:1 | POSTAL ADDRESS, na który może zostać wysłana poczta. |
| «cntd» *RoadAddress* | | *RoadAddress* | 0:1 | ADDRESS budynku z numerem przy nazwanej drodze. |

8.3.3. Składnik miejsca — SiteElement (klasa abstrakcyjna) SITE

ELEMENT opisuje wspólne własności miejsca w przestrzeni.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *AddressablePlace* | ::> | SITE ELEMENT dziedziczy po ADDRESSABLE PLACE. |
| «PK» | *id* | *SiteElementIdType* | 0:1 | Identyfikator obiektu SITE ELEMENT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***AccessibilityAssessment*** | *AccessibilityAssessment* | 0:1 | ACCESSIBILITY obiektu SITE ELEMENT. |
| «cntd» | ***AccessModes*** | *AccessModeEnum* | 0:* | Metody dostępu do SITE ELEMENT:<br>• *foot* (pieszo),<br>• *bicycle* (rowerem),<br>• *boat* (łodzią),<br>• *car* (samochodem),<br>• *taxi* (taksówką),<br>• *shuttle* (transfer),<br>• *ski* (na nartach),<br>• *skate* (na łyżwach). |
| | ***CrossRoad*** | *MultilingualString* | 0:1 | Nazwa najbliższego skrzyżowania lub drogi, przy którym znajduje się SITE. Używane głównie przy komunikacji autobusowej, gdzie taka informacja dodaje kontekst użytkownikowi. |
| «cntd» | ***SiteElementProperties Group*** | ***xmlGroup*** | 0:1 | Dalsze własności elementu — zobacz poniżej. |

**SiteElementPropertiesGroup**

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «enum» | *Covered* | *CoveredEnum* | 0:1 | Czy obiekt jest zadaszony. Dostępne wartości:<br>• *indoors* (w pomieszczeniu zamkniętym),<br>• *outdoors* (na otwartej przestrzeni),<br>• *covered* (zadaszenie),<br>• *mixed* (mieszany),<br>• *unknown* (nieznany). |
| «enum» | *Gated* | *GatedEnum* | 0:1 | Czy obiekt jest ogrodzony (dostęp poprzez bramki). Dostępne wartości:<br>• *gatedArea* (ogrodzony obszar),<br>• *openArea* (otwarty obszar),<br>• *unknown* (nieznane). |
| «enum» | *Lighting* | *LightingEnum* | 0:1 | Sposób oświetlenia obiektu. Dostępne wartości:<br>• *wellLit* (dobrze oświetlony),<br>• *poorlyLit* (słabo oświetlony),<br>• *unlit* (nieoświetlony),<br>• *other* (inne),<br>• *unknown* (nieznane). |

### 8.3.4. Miejsce — Site (klasa abstrakcyjna)

Podklasa PLACE, stanowiąca miejsce, do którego pasażerowie mogą chcieć podróżować, np. STOP PLACE, POINT OF INTEREST.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *SiteElement* | ::> | SITE dziedziczy po SITE ELEMENT. |
| «PK» | *id* | *SiteIdType* | 1:1 | Identyfikator obiektu SITE. |
| «FK» | *TopographicPlaceRef* | *TopographicPlaceRef* | 0:1 | Odniesienie do TOPOGRAPHIC PLACE, z którym SITE |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | jest powiązane. |
| «cntd» | ***additionalTopographicPlaces*** | *TopographicPiaceRef* | 0:* | Dodatkowe TOPOGRAPHIC PLACE, z którymi SITE jest powiązane. |
| «cntd» | ***Locale*** | *Locale* | 0:1 | Lokalizacja, która pozwala określić strefę czasową, domyślny język itp. dla SITE. |
| «FK» | ***OrganisationRef*** | *OrganisationRef* | 0:1 | Odniesienie do operatora (OPERATOR) danego SITE. |
| «FK» | ***ParentSiteRef*** | *SiteRef* | 0:1 | Odniesienie do nadrzędnego SITE. |
| «cntd» | ***levels*** | *Level* | 0:* | Istniejące LEVEL w SITE. |
| «cntd» | ***entrances*** | *Entrance* | 0:* | Istniejące ENTRANCE da SITE. |

8.3.5. Adres — Address (klasa abstrakcyjna) Wspólne atrybuty dla obiektu ADDRESS.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Place* | ::> | ADDRESS dziedziczy po PLACE. |
| «PK» | ***id*** | *AddressidType* | 1:1 | Identyfikator obiektu ADDRESS. |
| «FK» | ***CountryRef*** | *CountryEnum* | 0:1 | COUNTRY dla ADDRESS. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***CountryName*** | *MultilingualString* | 0:1 | Nazwa kraju używana w adresie. |

8.3.6. Adres pocztowy — PostalAddress (podkomponent SITE)

Reprezentacja adresu pocztowego.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Address* | ::> | POSTAL ADDRESS dziedziczy po ADDRESS. |
| «PK» | ***id*** | *PostalAddressIdType* | 1:1 | Identyfikator obiektu POSTAL ADDRESS. |
| | ***HouseNumber*** | *xsd:normalizedString* | 0:1 | Numer budynku. |
| | ***BuildingName*** | *xsd:normalizedString* | 0:1 | Nazwa budynku. |
| | ***AddressLine1*** | *xsd:normalizedString* | 0:1 | Pierwsza linia adresu. |
| | ***AddressLine2*** | *xsd:normalizedString* | 0:1 | Druga linia adresu. |
| | ***Street*** | *xsd:normalizedString* | 0:1 | Nazwa ulicy. |
| | ***Town*** | *xsd:normalizedString* | 0:1 | Miasto. |
| | ***Suburb*** | *xsd:normalizedString* | 0:1 | Osiedle. |
| | ***PostCode*** | *PostCodeType* | 0:1 | Kod pocztowy. |
| | ***PostCodeExtension*** | *xsd:normalizedString* | 0:1 | Rozszerzenie kodu pocztowego. |
| | ***PostalRegion*** | *xsd:normalizedString* | 0:1 | Region pocztowy. |
| | ***Province*** | *xsd:normalizedString* | 0:1 | Prowincja pocztowa. |
| «FK» | ***RoadAddressRef*** | *RoadAddressRef* | 0:1 | ROAD ADDRESS powiązany z danym POSTAL ADDRESS. |

8.3.7. Adres drogowy — RoadAddress (podkomponent SITE) Adres

określający lokalizację przy nazwanej ulicy.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *Address* | ::> | ROAD ADDRESS dziedziczy po ADDRESS. |
| «PK» | ***id*** | *RoadAddressIdType* | 1:1 | Identyfikator obiektu ROAD ADDRESS. |
| «EK» | ***GisFeatureRef*** | *xsd:normalizedString* | 0:1 | Odniesienie do identyfikatora obiektu GIS. |
| | ***RoadNumber*** | *xsd:normalizedString* | 0:1 | Numer drogi (ROAD). |
| | ***RoadName*** | *xsd:normalizedString* | 0:1 | Nazwa drogi (ROAD). |
| «enum» | ***BearingCompass*** | *CompassEnum* | 0:1 | Namiar kompasowy ROAD w punkcie definiowanym przez ADDRESS. |
| | ***BearingDegrees*** | *xsd:integer* | 0:1 | Namiar w stopniach w punkcie definiowanym przez ADDRESS. |
| | ***OddNumberRange*** | *RoadNumberRangeStructure* | 0:1 | Zakres nieparzystych numerów obiektu ADDRESS. |
| | ***EvenNumberRange*** | *RoadNumberRanaeStructure* | 0:1 | Zakres parzystych numerów |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | obiektu ADDRESS. |

8.3.8. Pozycja topograficzna — Topographic Place

TOPOGRAPHIC PLACE stanowi geograficzne miejsce, które określa kontekst podczas wyszukiwania lub prezentacji informacji o podróży, np. jako początek lub koniec przejazdu. Może być zróżnicowanej wielkości i stopnia administracyjnego, np. *'Województwo mazowieckie', 'Powiat wyszkowski', 'Wyszków'.*

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Place* | ::> | TOPOGRAPHIC PLACE dziedziczy po PLACE. |
| «PK» | ***id*** | *TopographicPlaceIdType* | 1:1 | Identyfikator obiektu TOPOGRAPHIC PLACE. |
| | ***Descriptor*** | *TopographicPlaceDescriptor* | 1:1 | Element opisujący TOPOGRAPHIC PLACE. |
| «enum» | ***TopographicPlace Type*** | *TopographicTypeEnum* | 0:1 | Rodzaj TOPOGRAPHIC PLACE. |
| | ***PostCode*** | *xsd:normalizedString* | 0:1 | Kod pocztowy przypisany do danego obszaru. |
| «FK» | ***CountryRef*** | *CountryRef* | 0:1 | Odniesienie do COUNTRY, w którym znajduje się bieżące TOPOGRAPHIC PLACE. |
| «FK» | ***ParentTopographicPlaceRef*** | *TopographicPlaceRef* | 0:1 | Odniesienie do |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | nadrzędnego TOPOGRAPHIC PLACE, które zawiera bieżące TOPOGRAPHIC PLACE. |
| | *containedIn* | *TopogmphicPlaceRef* 0:* | | Odniesienia do innych TOPOGRAPHIC PLACE, wewnątrz których bieżące TOPOGRAPHIC PLACE się zawiera. |

8.3.9. Opis pozycji topograficznej — TopographicPlaceDescriptor (podkomponent TopographicPlace)

Element definicji TOPOGRAPHIC PLACE określający nazwę danego obiektu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> *VersionedChild* | | | TOPOGRAPHIC PLACE DESCRIPTOR dziedziczy po VERSIONED CHILD. |
| «PK» | *id* | *TopographicPlaceIdDescriptorType* | 1:1 | Identyfikator obiektu TOPOGRAPHIC PLACE DESCRIPTOR. |
| | *Name* | *MultilingualString* | 1:1 | Nazwa obiektu TOPOGRAPHIC PLACE DESCRIPTOR. |

### 8.3.10. Punkt zainteresowania — Point of Interest

POINT OF INTEREST (POI) to rodzaj PLACE, do którego lub przez które pasażerowie mogą chcieć podróżować. Zazwyczaj stanowi obiekt użyteczności publicznej lub atrakcję turystyczną.

Standard nie oczekuje, że dane o obiektach POINT OF INTEREST będą przykazywane razem z danymi rozkładowymi NeTEx; w większości przypadków będą one definiowane w zewnętrznych bazach danych (np. INSPIRE, OSM) i poprzez SITE CONNECTION będą tworzone do nich odniesienia

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *Site* | ::> | POINT OF INTEREST dziedziczy po SITE. |
| «PK» | ***id*** | *PointOfInterestIdType* | 1:1 | Identyfikator obiektu POINT OF INTEREST. |
| «cntd» | ***classifications*** | *PointOfInterestClassificationView* | 0:* | Klasyfikacja obiektu POINT OF INTEREST. Zgodnie z normami NeTEx i EPIP, w rozpatrywanym zastosowaniu będzie wykorzystywany jedynie atrybut ***Name*** z enci ***PointOfInterestClassificationView.*** |
| «cntd» | ***nearTopographicPlaces*** | *TopographicPlaceRef* | 0:* | Obiekty TOPOGRAPHIC PLACE znajdujące się niedaleko danego POINT OF INTEREST. |

## 8.4. Miejsca zatrzymania (przystanki)

### 8.4.1. Miejsce zatrzymania — StopPlace

STOP PLACE, tak jak opisano wyżej, reprezentuje stację, przystanek, port, przystań, port lotniczy lub dowolny fizyczny punkt dostępu do środków transportu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Site* | ::> | STOP PLACE dziedziczy po SITE. |
| «PK» | ***id*** | *StopPlaceIdType* | 1:1 | Identyfikator obiektu STOP PLACE. |
| «AK» | ***PublicCode*** | *StopPlace Code Type* | 0:1 | Kod używany publicznie do identyfikacji STOP |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | PLACE jako alternatywa do nazwy. |

**STOP PLACE COMPONENT PROPERTY GROUP (własności wspólne dla wszystkich STOP PLACE COMPONENT)**

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «enum» | ***TransportMode*** | *VehicleModeEnum* | 0:1 | Główny środek transportu (TRANSPORT MODE) danego STOP PLACE. Dostępne wartości:<br>• *air* (transport lotniczy),<br>• *bus* (autobus),<br>• *trolleyBus* (trolejbus),<br>• *coach* (autokar),<br>• *funicular* (kolej linowo-terenowa),<br>• *metro* (metro),<br>• *tram* (tramwaj),<br>• *cableway* (kolej linowa),<br>• *rail* (transport |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | kolejowy)<br>• *water* (transport wodny),<br>• *ferry* (prom),<br>• *lift* (dźwig osobowy),<br>• *taxi* (taksówka),<br>• *other* (inny). |
| «enum» | *submode* | *TransportSubmodeEnum* | 0:1 | Podrzędne środki transportu powiązane z głównym. |
| «enum» | *OtherTransportModes* | *VehicleModeEnum* | 0:* * | Inne dostępne środki |
| «cntd» | *tariffZones* | *TariffZoneRef* | 0:* | Obiekty TARIFF ZONE obejmujące dany STOP PLACE. |

**STOP PLACE PROPERTY GROUP (własności dostępne tylko dla encji STOP PLACE)**

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «enum» | *StopPlaceTyp e* | *StopPlaceTypeEnum* | 1:1 | Rodzaj STOP PLACE.<br>Dostępne wartości:<br>• *onstreetBus* (przystanek autobusowy),<br>• *onstreetTram* (przystanek<br>• *busSt* |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *airport* (port lotniczy),<br>• *railStation* (stacja kolejowa),<br>• *metroStation* (stacja metra),<br>• *coachStation* (dworzec autobusów dalekobieżnych),<br>• *ferryPort* (port promowy),<br>• *harbourPort* (nabrzeże portowe),<br>• *ferryStop* (przystanek promu),<br>• *liftStation* (przystanek dźwigu osobowego)<br>• *tramStation* (dworzec tramwajowy),<br>• *vehicleRailInterchange* (kolejowy węzeł przesiadkowy),<br>• *taxiStand* (postój taksówek),<br>• *other* (inny). |
| | *BorderCrossing* | *xsd:boolean* | 0:1 | Czy STOP PLACE jest przejściem granicznym. |

**STOP PLACE TOPOGRAPHIC GROUP**

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «enum» | *Weighting* | *InterchangeUseEnum* | 0:1 | Waga STOP PLACE w kontekście preferowania przesiadek. Dostępne wartości: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *noInterchange* (przesiadki nie są dozwolone),<br>• *interchangeAllowed* (przesiadki są dozwolone),<br>• *preferredInterchange* (przesiadki są preferowane w tym STOP PLACE — jest szczególnie dogodne do przesiadek, ma odpowiednie oznaczenia, krótkie przejście piesze itp). |
| «enum» | *StopPlaceWeight* | *StopPlaceWeightEnum* | 0:1 | Waga STOP PLACE w kontekście oczekiwanego rodzaju przesiadek. Dostępne wartości:<br>• *international* (międzynarodowe),<br>• *national* (krajowe),<br>• *regional* (regionalne),<br>• *local* (lokalne). |

**STOP PLACE PASSENGER GROUP**

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *quays* | *Quay* | 0:* | Obiekty QUAY zawarte w danym STOP PLACE,<br><br>to jest perony, zatoki, stanowiska odjazdowe i inne punkty fizycznego dostępu do pojazdów. |

8.4.2. Komponent miejsca — SiteComponent (klasa abstrakcyjna)

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *SiteElement* | ::> | SITE COMPONENT dziedziczy po SITE ELEMENT. |
| «PK» | *id* | *SiteComponentIdType* | 1:1 | Identyfikator obiektu SITE COMPONENT. |
| «FK» | *LevelRef* | *LevelRef* | 0:1 | Odniesienie do encji LEVEL przypisanej do SITE COMPONENT. |

8.4.3. Przestrzeń miejsca zatrzymania — StopPlaceSpace (klasa abstrakcyjna)

STOP PLACE SPACE to warstwa abstrakcji dodająca etykietę do SITE COMPONENT. Stanowi bazę dla QUAY oraz STOP PLACE ENTRANCE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *SiteComponent* | ::> | STOP PLACE SPACE dziedziczy po SITE COMPONENT. |
| | *Label xsd.normalizedString* 0:1 | | | Lokalna etykieta komponentu, np. numer przystanku w obrębie zespołu przystankowego. |

8.4.4. Stanowisko — Quay

QUAY to obiekt taki jak peron, zatoka przystankowa, stanowiska odjazdowe lub inny punkt fizycznego dostępu do pojazdów transportu publicznego, taksówek, samochodów lub innych pojazdów. QUAY może obsługiwać jeden lub więcej VEHICLE STOPPING PLACE i być powiązany z co najmniej jednym SCHEDULED STOP POINT.

QUAY może zawierać podrzędne QUAY, które muszą fizycznie zawierać się w nadrzędnym obiekcie. To pozwala na reprezentację złożonych peronów z większą liczbą krawędzi peronowych albo nazwanymi sektorami.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *StopPlaceSpace* | ::> | QUAY dziedziczy po STOP PLACE SPACE. |
| «PK» | *id* | *QuayIdType* | 1:1 | Identyfikator obiektu QUAY. |
| «AK» | *PublicCode* | *xsd:normalizedString* | 0:1 | Kod używany publicznie do |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | identyfikacji obiektu QUAY. |
| «enum» | *QuayType* | *QuayTypeEnum* | 0:1 | Klasyfikacja QUAY. |
| «FK» | *ParentQuayRef* | *QuayRef* | 0:1 | Odniesienie do nadrzędnego obiektu QUAY. |

8.4.5. Wejście — Entrance (klasa abstrakcyjna)

ENTRANCE opisuje dowolne fizyczne wejście lub wyjście do/z obiektu SITE. Mogą to być drzwi, bramka albo inny punkt dostępu. Istnieją specjalizowane klasy dla odpowiednich obiektów, np. STOP PLACE ENTRANCE. ˙

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *SiteComponent* | ::> | ENTRANCE dziedziczy po SITE COMPONENT. |
| «PK» | *id* | *EntranceIdType* | 1:1 | Identyfikator obiektu ENTRANCE. |
| «AK» | *PublicCode* | *xsd:normalizedString* | 0:1 | Kod używany publicznie do identyfikacji obiektu ENTRANCE. |
| | *Label* | *xsd:normalizedString* | 0:1 | Etykieta obiektu ENTRANCE. |
| «enum» | *EntranceType* | *Entrance TypeEnu m* | 0:1 | Rodzaj ENTRANCE. Dostępne wartości: |

| Klasyfikacj | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *automaticDoor* (drzwi automatyczne),<br>• *ticketBarrier* (bramka biletowa),<br>• *gate* (brama),<br>• *other* (inne). |
| | *IsExternal* | *xsd:boolean* | 0:1 | Czy ENTRANCE otwiera się na zewnątrz. |
| | *IsEntry* | *xsd:boolean* | 0:1 | Czy ENTRANCE może zostać użyte jako wejście do SITE. |
| | *IsExit* | *xsd:boolean* | 0:1 | Czy ENTRANCE może zostać użyte jako wyjście z SITE. |
| | *Width* | *LengthType* | 0:1 | Szerokość wejścia. |
| | *Height* | *LengthType* | 0:1 | Wysokość wejścia. |
| | *DroppedKerbOutside* | *xsd:boolean* | 0:1 | Czy na zewnątrz wejścia znajduje się obniżony krawężnik. |

### 8.4.6. Wejście miejsca zatrzymania — StopPlaceEntrance

STOP PLACE ENTRANCE to fizyczne wejście lub wyjście do/ze STOP PLACE dostępne dl a pasażerów. Przydatne w przypadku dużych węzłów przesiadkowych.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Entrance* | ::> | STOP PLACE ENTRANCE dziedziczy po ENTRANCE. |
| «PK» | *id* | *StopPlaceEntranceIdType* | 1:1 | Identyfikator obiektu STOP PLACE ENTRANCE. |

## 8.5. Grupa miejsc zatrzymania — GroupOfStopPlaces

GROUP OF STOP PLACES to grupa przystanków, które często będą traktowane wspólnie do celów wyszukiwania, np. przystanki przy porcie lotniczym, główne stacje w mieście.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *GroupOfEntities* | ::> | GROUP OF STOP PLACES dziedziczy po GROUP OF ENTITIES. |
| «PK» | ***id*** | *GmupOfStopPlacesIdType* | 1:1 | Identyfikator obiektu GROUP OF STOP PLACES. |
| «cntd» | ***members*** | *StopPlaceRef* | 0:* | Obiekty STOP PLACE będące częścią GROUP OF STOP PLACES. |
| «enum» | ***TransportMode*** | *VehicleModeEnum* | 0:1 | Podstawowy TRANSPORT MODE grupy. Dostępne wartości — zobacz Miejsce zatrzymania. |
| «enum» | ***TransportSubmode*** | *SubmodeEnum* | 0:1 | Podrzędne środki transportu powiązane z głównym. |

## 8.6. Strefa taryfowa — TariffZone

TARIFF ZONE definiuję strefę (ZONE), która służy do opisywania taryf strefowych.

Relacja pomiędzy TARIFF ZONE a SCHEDULED STOP POINT jest dwukierunkowa i zgodnie z NeTEx, może być definiowana zarówno po stronie ***TariffZone****,* jak i ***ScheduledStopPoint****.* EPIP dla uporządkowania zaleca, aby była definiowana tylko w ***ScheduledStopPoint****.*

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Zone* | ::> | TARIFF ZONE dziedziczy po ZONE. |
| «PK» | ***id*** | *TariffZoneidType* | 1:1 | Identyfikator obiektu TARIFF ZONE. |
| «cntd» | ***Presentation*** | *Presentation* | 0:1 | Cechy prezentacyjne takie jak kolor przypisany do TARIFF ZONE. |

# 9. PROFIL SIECI I ROZKŁADÓW JAZDY — NETWORK AND TIMETABLE PROFILE

## 9.1. Wstęp

Profil sieci i rozkładów jazdy definiuje dane potrzebne do opisu rozkładu jazdy dla pasażerów. Opis podzielony jest na moduły:

- topologia sieci (linie, trasy);
- dostępne wzorce obsługi (schematy obsługi przystanków, itp.);
- konkretne kursy transportu zbiorowego wraz z informacjami o czasach;
- rodzaje dni i kalendarzy definiujące okresy kursowania poszczególnych przejazdów.

## 9.2. Podstawowa topologia sieci

### 9.2.1. Przegląd

Topologia opisuje stałe linie i trasy w sieci transportowej.

Transmodel definiuje LINE jako grupę obiektów ROUTE, która jest znana publicznie pod wspólną nazwą lub numerem. Obiekty ROUTE zazwyczaj są podobne do siebie i stanowią warianty podstawowej trasy, w szczególności mogą to być dwie trasy, które używają tej samej ścieżki, ale w przeciwnych kierunkach (DIRECTION).

LINE jest powiązana z głównym TRANSPORT MODE i SUBMODE, ale może mieć dodatkowe środki transportu, np. linia kolejowa, która w określonych okolicznościach może być obsługiwana przez komunikację autobusową.

Kolejnym atrybutem LINE jest podmiot OPERATOR lub AUTHORITY (dodatkowe podmioty są również dozwolone).

Instancje LINE mogą być grupowane w GROUP OF LINES w celu porządkowania sieci (np. autobusowe linie nocne)

### 9.2.2. Linia — Line

LINE jest grupą obiektów ROUTE, która jest znana publicznie pod wspólną nazwą lub numerem.

| Klasyfikacja Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|
| ::> ::> | *DataManaaedObiect* | ::> | LINE dziedziczy po DATA MANAGED OBJECT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *LineIdType* | 1:1 | Identyfikator obiektu LINE. |
| | ***Name*** | *MultilingualString* | 1:1 | Nazwa obiektu LINE. |
| | ***ShortName*** | *MultilingualString* | 0:1 | Krótka nazwa obiektu LINE. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu LINE. |
| «enum» | ***TransportMode*** | *AllVehicleModesOfTransportEnum* | 0:1 | Identyfikator podstawowego TRANSPORT MODE dla LINE. Zobacz STOP PLACE dla dozwolonych wartości. |
| «enum» | ***TransportSubmode*** | *TransportSubmodeEnum* | 0:1 | Tryb podrzędny dla TRANSPORT MODE obiektu LINE. |
| | ***Url*** | *any* | 0:1 | Adres URL powiązany z obiektem LINE. |
| «AK» | ***PublicCode*** | *xsd:normalizedString* | 0:1 | Identyfikator publiczny obiektu LINE. |
| «AK» | ***PrivateCode*** | *xsd:normalizedString* | 0:1 | Identyfikator alternatywny obiektu LINE. |
| «FK» | ***OperatorRef*** | *OperatorRef* | 0:1 | Odniesienie do OPERATOR obiektu LINE. Może zostać zdefiniowane wyłącznie |

| Klasyfikacja | Nazwa | Typ | Kardynaln ość | Opis |
|---|---|---|---|---|
| | | | | OperatorRef albo AuthorityRef. |
| «FK» | ***AuthorityRef*** | *AuthorityRef* | 0:1 | Odniesienie do AUTHORITY obiektu LINE. Może zostać zdefiniowane wyłącznie OperatorRef albo AuthorityRef. |
| «cntd» | ***additionalOperators*** | *OperatorRef* | 0:* | Odniesienie do dodatkowego TRANSPORT ORGANISATION. (Może to być OPERATOR lub AUTHORITY). |
| «cntd» | ***otherModes*** | *VehicleModeEnum* | 0:* | Dodatkowy środek (MODE) transportu dla obiektu LINE. |
| | ***TypeOfLineRef*** | *TypeOfLineRef* | 0:1 | Odniesienie do TYPE dla obiektu LINE |
| | ***Monitored*** | *xsdboolean* | 0:1 | Czy są zwykle dostępne dane w czasie rzeczywistym dla obiektu LINE. |
| «cntd» | ***routes*** | *RouteRef* | 0:* | Trasy którymi podąża LINE. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***Presentation*** | *Presentation* | 0:1 | Preferowane atrybuty prezentacji używane podczas przedstawiania obiektów LINE na mapach, itp. |
| «cntd» | ***AccessibilityAssessment*** | *AccessibilityAssessment* | 0:1 | Ogólna charakterystyka ułatwień dostępu dla obiektu LINE |
| «cntd» | ***allowedDirections*** | *AllowedLineDirection* | 0:* | Dozwolone kierunki dla linii. |
| «cntd» | ***noticeAssignments*** | *NoticeAssignment* | 0:* | Uwagi (NOTICE) przypisane do obiektu LINE. |
| «cntd» | ***documentLinks*** | *InfoLinks* | 0:* | Dokumenty rozkładu jazdy powiązane z obiektem LINE, na przykład pliki PDF. |

### 9.2.3. Dozwolony kierunek linii — AllowedLineDirection (podkomponent Line)

Dozwolony kierunek (DIRECTION) dla danej trasy (ROUTE). Używany do przyporządkowania DIRECTION do LINE i weryfikacji wyboru dozwolonych wartości.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | ALLOWED LINE DIRECTION dziedziczy po DATA MANAGED OBJECT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | *id* | *AllowedDirectionIdType* | 1:1 | Identyfikator obiektu LINE ALLOWED DIRECTION. |
| «FK» | *DirectionRef* | *DirectionRef* | 1:1 | Odniesienie do DIRECTION. |

9.2.4. Grupa linii — GroupOfLines

GROUP OF LINES to grupa instancji LINE utworzona w celu porządkowania sieci. Jedna linia może należeć do wielu grup.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *GroupOfEntities* | ::> | GROUP OF LINES dziedziczy po GROUP OF ENTITIES. |
| «PK» | *id* | *GroupOfLinesIdType* | 1:1 | Identyfikator obiektu GROUP OF LINES. |
| «cntd» | *members* | *LineRef* | 0:* | Odniesienie do członków GROUP OF LINES. |
| «FK» | *MainLineRef* | *LineRef* | 0:1 | Główny obiekt LINE w GROUP OF LINES. |
| «enum» | *TransportMode* | *VehicleModeEnum* | 0:1 | Podstawowy tryb transportu (MODE) dla GROUP OF LINES. Zobacz STOP PLACE dla dozwolonych wartości |

9.2.5. Sieć transportowa — Network

NETWORK to nazwana GROUP OF LINES, np. linie uruchamiane przez jeden zarząd transportu w danym mieście.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *GroupOfLines* | ::> | NETWORK dziedziczy po GROUP OF LINES. |
| «PK» | *id* | *NetworkIdType* | 1:1 | Identyfikator obiektu NETWORK. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *Name* | *MultilingualString* | 1: 1 | Nazwa obiektu NETWORK. Ze względu na konwencję, nazwa jest obowiązkowa |
| «FK» | *TransportOrganisationRe* | *OperatorRef / AuthorityRef* | 0:1 *f* | Organizacja transportu odpowiedzialna za NETWORK. |
| «cntd» | *groupsOfLines* | *GroupOfLines* | 0:* | Grupy obiektów LINE w NETWORK. Może zostać użyte w celu podzielenia lub klasyfikowania linii wewnątrz sieci. |

### 9.2.6. Wyświetlenie kierunku — DestinationDisplay

DESTINATION DISPLAY stanowi definicję pokazywanego pasażerom opisu krańca, do którego zmierza pojazd. Z punktu widzenia pasażerów ważne jest, aby ten opis był spójny i pokazywany wszędzie w tej samej formie. DESTINATION DISPLAY może być w EPIP przyporządkowany jedynie do elementów STOP POINT IN JOURNEY PATTERN.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | DESTINATION DISPLAY dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *DestinationDisplayIdType* | 1:1 | Identyfikator obiektu DESTINATION DISPLAY. |
| | *SideText* | *MultilingualString* | 0:1 | Tekst wyświetlany na boku pojazdu powiązanego z DESTINATION DISPLAY. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *FrontText* | *MultilingualString* | 0:1 | Tekst wyświetlany na przodzie pojazdu powiązanego z DESTINATION DISPLAY. |
| «AK» | *PublicCode* | *xsd:normalizedString* 0:1 | | Kod publiczny (do powszechnego prezentowania) powiązany z DESTINATION DISPLAY. |
| «cntd» | *vias* | *Via* | 0:* | Tekst do wyświetlania dla VIA. |
| «cntd» | *variants* | *DestinationDisplayVariant* | 0:* | Teksty wariantów wyświetlania dla różnych mediów. |

### 9.2.7. Element pośredni trasy — Via

VIA to opcjonalny element DESTINATION DISPLAY mający na celu rozróżnienie wielu możliwych tras pomiędzy początkiem i końcem trasy.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | *VersionedChild* | ::> | VIA dziedziczy po VERSIONED CHILD. |
| | *id* | *ViaIdType* | 1:1 | Identyfikator obiektu VIA. |
| | *Name* | *MultilingualString* | 1:1 | Nazwa obiektu VIA. |

### 9.2.8. Wariant wyświetlenia kierunku — DestinationDisplayVariant

W niektórych przypadkach wymagane jest zastosowanie różnych wariantów DESTINATION DISPLAY dla poszczególnych nośników informacji (wyświetlacz, drukowany rozkład, zapowiedzi przystankowe, itd.).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | DESTINATION DISPLAY VARIANT dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *DestinationDisplayIdType* | 1:1 | Identyfikator obiektu DESTINATION DISPLAY VARIANT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | DESTINATION DISPLAY VARIANT powinien być osadzony w linii z rodzicem DESTINATION DISPLAY |
| «enum» | ***DestinationDisplayVariantMediaType*** | *DeliveryVariantTypeEnum* | 1: 1 | Rodzaj dostarczania DESTINATION DISPLAY VARIANT. Dostępne wartości:<br>• *printed* (wydruk),<br>• *textToSpeech* (zamiana tekstu na mowę),<br>• *recordedAnnouncement* (nagrany komunikat),<br>• *web* (strona internetowa),<br>• *mobile* (aplikacje mobilne),<br>• *other* (inne). |
| | ***SideText*** | *MultilingualString* | 0: 1 | Tekst wyświetlany na boku pojazdu powiązanego z DESTINATION DISPLAY VARIANT. |
| | ***FrontText*** | *MultilingualString* | 0: 1 | Tekst wyświetlany na przodzie pojazdu powiązanego z DESTINATION DISPLAY VARIANT. |
| «cntd» | ***vias*** | *Via* | 0:* | Tekst dla wyświetlacza stacji pośrednich (VIA). |

### 9.2.9. Organizacja — Organisation (Abstract)

ORGANISATION reprezentuje podmiot, który jest zaangażowany w planowanie, przetwarzanie lub udostępnianie informacji o transporcie publicznym. Na przykład zarząd transportu, albo przewoźnik.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManaaedObiect* | ::> | ORGANISATION dziedziczy po DATA MANAGED OBJECT. |
| «AK» | ***PublicCode*** | *xsd:normalizedString* | 0:1 | Alternatywny publiczny identyfikator obiektu ORGANISATION. |
| «AK» | ***CompanyNumber*** | *xsd:normalizedString* | 0:1 | Numer rejestracyjny podmiotu prawnego. |
| | ***Name*** | *xsd:normalizedString* | 0:1 | Nazwa obiektu ORGANISATION. |
| | ***ShortName*** | *MultilingualString* | 0:1 | Krótka nazwa obiektu ORGANISATION. |
| | ***LegalName*** | *MultilingualString* | 0:1 | Nazwa prawna obiektu ORGANISATION. |
| «cntd» | ***alternativeNames*** | *AlternativeName* | 0:* | Alternatywne nazwy obiektu ORGANISATION. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu ORGANISATION. |
| «cntd» | ***ContactDetails*** | *ContactDetails* | 1:1 | Informacje kontaktowe. |
| «FK» | ***OrganisationType*** | *TypeOfOrganisationEnum* | 1:1 | Rodzaj ORGANISATION.<br>Dostępne wartości:<br>• *authority* (podmiot odpowiedzialny),<br>• *operator* (przewoźnik),<br>• *railOperator* |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | przewoźnik kolejowy),<br>• *statutoryBody* (organ ustawowy),<br>• *facilityOperator* (zarządca obiektu),<br>• *travelAgent* (biuro podróży),<br>• *servicedOrganisation* (obsługiwana organizacja),<br>• *other* (inny). |

9.2.10. Przewoźnik — Operator

OPERATOR definiuje przewoźnika — podmiot odpowiedzialną za uruchamianie przewozów.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Organisation* | ::> | OPERATOR dziedziczy po ORGANISATION. |
| «PK» | *id* | *OperatorIdType* | 1:1 | Identyfikator obiektu OPERATOR. |
| «FK» | *CountryRef* | *CountryRef* | 0:1 | Odniesienie do kraju (ISO 3166-1). |
| «cntd» | *Address* | *PostalAddress* | 0:1 | ADDRESS pocztowy dla ORGANISATION. |
| «enum» | *PrimaryMode* | *VehicleModeEnum* | 0:1 | Podstawowy TRANSPORT MODE dla ORGANISATION. Zobacz STOP PLACE dla dozwolonych wartości. |

9.2.11. Podmiot odpowiedzialny — Authority

AUTHORITY definiuje podmiot odpowiedzialny za organizację przewozów na danym obszarze.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Organisation* | ::> | AUTHORITY dziedziczy po ORGANISATION. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *AuthorityIdType* | 1:1 | Identyfikator obiektu AUTHORITY. |
| «cntd» | ***Address*** | *PostalAddress* | 1:1 | ADDRESS pocztowy dla ORGANISATION. |

## 9.3. Trasy

### 9.3.1. Przegląd

Obiekty ROUTE opisują ścieżki po fizycznej infrastrukturze używane przez regularne przewozy transportu publicznego. ROUTE to liniowa struktura złożona z punktów i połączeń (LINK) zdefiniowanych do tego celu.

ROUTE reprezentuje abstrakcyjne pojęcie, które samo w sobie nie ma znaczenia operacyjnego. Jego zadaniem jest opis ścieżki niezależnie od konkretnych elementów infrastruktury (ROAD ELEMENT, RAILWAY ELEMENT) oraz wzorców obsługi przystanków (SCHEDULED STOP POINT).

Może się zdarzyć, że niektóre podmioty OPERATOR nie zapewniają danych dotyczących ROUTE. W takim wypadku te dane wewnątrz LINE mogą zostać pominięte lub stworzone na podstawie SERVICE PATTERN (z użyciem instancji SCHEDULED STOP POINT do stworzenia ROUTE POINT).

### 9.3.2. Punkt trasy — RoutePoint

ROUTE POINT definiuje punkt (POINT) kształtujący ROUTE w sieci. Relacja pomiędzy ROUTE POINT a ROUTE jest typu wiele-do-wielu, do ich połączenia służą instancje POINT ON ROUTE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Point* | ::> | ROUTE POINT dziedziczy po POINT. |
| «PK» | ***id*** | *RoutePointidType* | 1:1 | Identyfikator obiektu ROUTE POINT. |
| | ***BorderCrossing*** | *xsd:boolean* | 0:1 | Czy punkt jest przejściem granicznym (punktem granicznym). |

### 9.3.3. Połączenie punktów trasy — RouteLink

ROUTE LINK to skierowane połączenie pomiędzy dwoma obiektami ROUTE POINT pozwalające na definicję unikalnej ścieżki po sieci. Jedno ROUTE LINK może być używane przez różne ROUTE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Link* | ::> | ROUTE LINK dziedziczy po LINK. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | *id* | *RouteLinkIdType* | 1:1 | Identyfikator obiektu ROUTE LINK. |
| | *Distance* | *DistanceType* 1:1 | | Odległość ROUTE LINK. Jednostki są takie jak określane dla struktury (domyślnie metry z układu SI). Dziedziczy po LINK, jednak ze względu na konwencję wartość jest wymagana w ROUTE LINK. |
| «FK» | *FromPointRef* | *RoutePointRef* | 1: | Odniesienie do ROUTE POINT w którym ROUTE LINK ma swój początek. |
| «FK» | *ToPointRef* | 1 | | Odniesienie do ROUTE POINT w którym ROUTE LINK ma swój koniec. |

### 9.3.4. Trasa — Route

ROUTE to uporządkowana lista punktów definiująca trasę przez sieć drogową lub kolejową. Może przebiegać przed jeden punkt wielokrotnie.

| Klasyfikacja ::> | Nazwa ::> | Typ *LinkSequence* | Kardynalność ::> | Opis ROUTE dziedziczy po LINK SEQUENCE. |
|---|---|---|---|---|
| «PK» | *id* | *RouteIdType* | 1:1 | Identyfikator obiektu ROUTE. |
| «FK» | *LineRef* | *LineRef* | 0:1 | Odniesienie do LINE do którego ROUTE przynależy. |
| «enum» *DirectionType* | | *TypeOfDirectionEnum* | 0:1 | Typ kierunku dla ROUTE. Dostępne wartości:<br>• *outbound* (wyjeżdżający) |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | się zgodnie z ruchem wskazówek zegara),<br>• *anticlockwise* (poruszający się przeciwnie do ruchu wskazówek zegara). |
| «FK» | *DirectionRef* | *DirectionRef* | 0:1 | Odniesienie do DIRECTION dla ROUTE. |
| «cntd» | *pointsInSequence* | *PointOnRoute* | 2:* | Punkty składające się na ROUTE. |
| «FK» | *InverseRouteRef* | *RouteRef* | 0:1 | Jeżeli występuje: odniesienie do odpowiedniego pasującego ROUTE dla przeciwległego kierunku. |

9.3.5. Punkt na trasie — PointOnRoute (podkomponent Route)

POINT ON ROUTE wiąże instancje ROUTE POINT z ROUTE.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *PointInSequence* ::> | | POINT ON ROUTE dziedziczy po POINT IN SEQUENCE. |
| «PK» | *id* | *PointOnRouteIdType* | 1: 1 | Identyfikator obiektu POINT ON ROUTE. |
| «PK» | *(order)* | *xsd:positiveInteger* | 1: 1 | Kolejność POINT ON ROUTE w ramach ROUTE (dziedziczy po POINT IN LINK SEQUENCE). |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***LineSequenceRef*** | *LinkSequenceRef* | 0:1 | Odniesienie do LINK SEQUENCE. |
| «FK» | ***RoutePointRef*** | *RoutePointRef* | 1:1 | Odniesienie do ROUTE POINT którego POINT ON ROUTE używa. |

## 9.4. Wzorce kursów — Service patterns

### 9.4.1. Przegląd

Model SERVICE PATTERN opisuje obsługę przez kursy tras w sieci transportowej i stanowi bazę do opisu rozkładów jazdy w obiektach SERVICE JOURNEY, które kursują zgodnie z SERVICE PATTERN.

W przeciwieństwie do ROUTE i LINE, które opisują możliwe ścieżki pojazdów w sieci, JOURNEY PATTERN opisuje faktyczny sposób obsługi linii przez pojazdy: zawiera informacje o miejscach zatrzymania oraz czasach.

SERVICE PATTERN jest częścią JOURNEY PATTERN ograniczającą się do informacji istotnych dla pasażerów, czyli punktów, gdzie mogą wsiadać i wysiadać z pojazdów (SCHEDULED STOP POINT) i z pominięciem punktów istotnych tylko z punktu widzenia konstrukcji rozkładu jazdy.

SERVICE PATTERN składa się z uporządkowanej listy STOP POINT IN JOURNEY PATTERN z możliwymi SERVICE LINK pomiędzy każdą parą kolejnych SCHEDULED STOP POINT. Jeden SCHEDULED STOP POINT może być odwiedzany wielokrotnie.

Możliwość przesiadki pomiędzy dwoma punktami zatrzymania w sieci transportowej jest zapisywana w instancjach CONNECTION.

### 9.4.2. Wzorzec przejazdu — JourneyPattern (klasa abstrakcyjna w EPIP)

JOURNEY PATTERN opisuje pracę pojazdów transportu publicznego. Aspekt tego opisu różni się od tego zapewnianego przez klasy ROUTE i LINE, które opisują schematyczne trasy, a JOURNEY PATTERN opisuje, w jaki sposób praca pojazdu jest wykonywana: zawiera listę punktów zatrzymań pojazdu oraz punktów, które są istotne z punktu widzenia trasowania i ustalania rozkładu jazdy.

JOURNEY PATTERN składa się z następujących elementów:

- uporządkowana lista SCHEDULED STOP POINT, na których pojazdy się zatrzymują;
- uporządkowana lista TIMING POINT, dla których definiowane są czasy przejazdu. Zgodnie z NeTEx mogą to być również
- punkty inne niż SCHEDULED STOP POINT, ale w kontekście EPIP wszystkie takie punkty są pomijane, a wymieniane dokumenty
- zawierają wyłącznie TIMING POINT, które są jednocześnie SCHEDULED STOP POINT.

JOURNEY PATTER zazwyczaj jest powiązany z ROUTE, po której się porusza. Kilka JOURNEY PATTERN może używać tej samej trasy (np. przyspieszone połączenie pomijające większość przystanków i zwykłe, które zatrzymuje się na każdym z nich).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *LinkSequence* | ::> | JOURNEY PATTERN dziedziczy po LINK SEQUENCE. |
| «FK» | *RouteRef* | *RouteRef*<br>*RouteView* | \| 0:1 | Odniesienie do ROUTE, którą porusza się JOURNEY PATTERN. Jeżeli ROUTE nie jest dostępne, *RouteRef* powinno zostać zastąpione *RouteView* (zdefiniowanym w XSD), gdzie *LineRef* powinno zostać uzupełnione odpowiednią instancją LINE. Jeżeli JOURNEY PATTERN nie jest powiązane z żadną linią, pole *RouteRef* należy pozostawić puste. |
| «FK» | *DirectionRef* | *DirectionRef* | 0:1 | Odniesienie do DIRECTION, w którym zmierza |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | dany obiekt JOURNEY PATTERN. |
| «FK» | *DestinationDisplayRef* | *DestinationDisplayRe* | 0:1 *f* | Odniesienie do DESTINATION DISPLAY powiązanego z JOURNEY PATTERN. |
| «cntd» | *pointsInSequence* | *PointInJourneyPatter* | 0:* *n* | Instancje POINT składające się na dany obiekt JOURNEY PATTERN. |

9.4.3. Wzorzec kursu — ServicePattern

SERVICE PATTERN dziedziczy po JOURNEY PATTERN zawierający listę STOP POINT IN JOURNEY PATTERN. Każdy STOP POINT IN JOURNEY PATTERN odnosi się do konkretnego SCHEDULED STOP POINT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *JourneyPattern* | ::> | SERVICEPATTERN dziedziczy po JOURNEY PATTERN. |
| «PK» | *id* | *ServiceJourneyPatternIdType* | 1:1 | Identyfikator obiektu SERVICE PATTERN. |

9.4.4. Punkt we wzorcu przejazdu — PointInJourneyPattern (klasa abstrakcyjna w EPIP)

Każdy punkt POINT IN JOURNEY PATTERN opisuje odwiedzenie instancji POINT przez SERVICE z określonym porządkiem we wzorcu (JOURNEY PATTERN).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *PointInLinkSequence* | ::> | POINT IN JOURNEY PATTERN dziedziczy po POINT IN LINK SEQUENCE. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***(order)*** | *xsd:positiveInteger* | 1:1 | Kolejność POINT IN JOURNEY PATTERN w ramach JOURNEY PATTERN (Dziedziczone po POINT IN LINK SEQUENCE). |
| «FK» | ***JourneyPatternRef*** | *JourneyPatternRef* | 0:1 | JOURNEY PATTERN odnoszący się do danego POINT (specjalizacja ***LinkSequenceRef).*** |
| «FK» | ***DestinationDisplayRef*** | *DestinationDisplayRef* | 0:1 | DESTINATION DISPLAY powiązane z POINT IN JOURNEY PATTERN. |
| «cntd» | ***FlexiblePointProperties*** | *FlexiblePointProperties* | 0:1 | Właściwości elastyczne dla POINT. |
| | ***ChangeOfDestinationDisplay*** | *xsd:boolean* | 0:1 | Czy DESTINATION DISPLAY zmienia się w tym punkcie. |

### 9.4.5. Punkt zatrzymania we wzorcu przejazdu — StopPointInJourneyPattern (podkomponent JourneyPattern)

Każdy STOP POINT IN JOURNEY PATTERN opisuje odwiedzenie instancji SCHEDULED STOP POINT przez SERVICE z określonym porządkiem we wzorcu JOURNEY PATTERN).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *PointInLinkSequence* | ::> | STOP POINT IN JOURNEY PATTERN |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | dziedziczy po POINT IN JOURNEY PATTERN. |
| «PK» | *id* | *StopPointInJourneyPatternIdType* | 1: 1 | Identyfikator obiektu STOP POINT IN JOURNEY PATTERN. |
| «PK» | *(order)* | *xsd:positiveInteger* | 1: 1 | Kolejność POINT IN JOURNEY PATTERN w ramach JOURNEY PATTERN (Dziedziczone po POINT IN LINK SEQUENCE). |
| «FK» | *ScheduledStopPointRef* | *ScheduledStopPointRef* | 1: 1 | Odniesienie do SCHEDULED STOP POINT który STOP POINT w ramach JOURNEY PATTERN odwiedza, nadpisuje *TimingPointRef* odziedziczone po TIMING POINT IN JOURNEY PATTERN. |
| «FK» | *OnwardServiceLinkRef* | *ServiceLinkRef* | 0:1 | Następny SERVICE LINK po STOP POINT w ramach JOURNEY PATTERN. Ten atrybut powinien być użyty wtedy i tylko wtedy, gdy wiele SERVICE LINK jest dostępnych pomiędzy dwoma kolejnymi SCHEDULED STOP POINT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***ForAlighting*** | *xsd:boolean* | 0:1 | Czy przystanek może służyć do wysiadania. |
| | ***ForBoarding*** | *xsd:boolean* | 0:1 | Czy przystanek może służyć do wsiadania. |
| «FK» | ***DestinationDisplayRef*** | *DestinationDisplayRef* | 0:1 | Wartości DESTINATION DISPLAY do pokazania na tym punkcie (Także dostępne jako ***DestinationDisplayView***). |
| «cntd» | ***FlexiblePointProperties*** | *FlexiblePointProperties* | 0:1 | Właściwości elastyczne dla POINT. |
| | ***ChangeOfDestinationDisplay*** | *xsd:boolean* | 0:1 | Czy DESTINATION DISPLAY zmienia się w tym punkcie. |
| | ***noticeAssignments*** | *NoticeA ssignment* | 0:* | NOTICE dla POINT IN JOURNEY PATTERN. |
| | ***RequestStop*** | *xsd:boolean* | 0:1 | Czy przystanek jest przystankiem na żądanie dla tego kursu. Domyślna wartość to *false.* |
| «enum» | ***RequestMethod*** | *RequestMethodTypeEnum* | 0:1 | Metoda żądania zatrzymania w tym konkretnym wzorze usługi. Jeśli nie określono, tak jak dla przystanku. |
| «enum» | ***StopUse*** | *StopUseEnum* | 0:1 | Charakterystyka użycia przystanku, domyślna wartość to *'access'.* Dostępne wartości: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *access* (wolny dostęp),<br>• *interchangeOnly* (wyłącznie przesiadanie się),<br>• *passthrough* (przejazd bez zatrzymania),<br>• *noBoardingOrAlighting* (bez wsiadania albo wysiadania — postój techniczny). |
| | *BookingArrangeme BookingArrangements nts* | | 0:1 | Ustalenia dotyczące rezerwacji dla przystanku, jeżeli inne niż te dla SERVICE JOURNEY. |

### 9.4.6. Kierunek — Direction

DIRECTION wskazuje względny kierunek linii. Jest to dodatkowa informacja, nie jest wymagana i może być używana głównie do celów prezentacji rozkładu, filtrowania.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *DataManagedObject* | ::> | DIRECTION dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *DirectionIdType* | 1:1 | Identyfikator obiektu DIRECTION. |
| «enum» | *DirectionType* | *Direction TypeEnu m* | 0:1 | Stała wartość powiązana z kierunkiem: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *outbound* (wyjeżdżający),<br>• *inbound* (przyjeżdżający),<br>• *clockwise* (poruszający się zgodnie z ruchem wskazówek zegara),<br>• *anticlockwise* (poruszający się przeciwnie do ruchu wskazówek zegara). |
| «FK» | *OppositeDirection DirectionRef Ref* | | 0:1 | Odniesienie do przeciwnego kierunku. |

9.4.7. Punkt pomiaru czasu — Timing point (klasa abstrakcyjna w EPIP)

TIMING POINT istnieje w profilu jedynie jako klasa abstrakcyjna, która jest specjalizowana przez SCHEDULED STOP POINT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *Point* | ::> | TIMING POINT dziedziczy po POINT. |
| «PK» | *id* | *TimingPointIdType* | 1:1 | Identyfikator obiektu TIMING POINT. |
| «enum» | *TimingPointStatus* | *TimingPointStatusEnum* | 0:1 | Rodzaj TIMING POINT.<br>Dostępne wartości:<br>• *timingPoint* (podstawowy punkt pomiaru czasu, wartość domyślna), |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *secondaryTimingPoint* (drugorzędny punkt pomiaru czasu),<br>• *notTimingPoint* (bez pomiaru |

9.4.8. Rozkładowy punkt zatrzymania — ScheduledStopPoint Punkt, w którym pasażerowie mogą wsiąść lub wysiąść z pojazdu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *TimingPoint* | ::> | SCHEDULED STOP POINT dziedziczy po TIMING POINT. |
| «PK» | *id* | *ScheduledStopPointIdType* | 1:1 | Identyfikator obiektu SCHEDULED STOP POINT. |
| «cntd» | *tariffZones* | *TariffZoneRef* | 0:* | Instancje TARIFF ZONE , do których SCHEDULED STOP POINT należy. |
| | *ShortName* | *MultilingualString* | 0:1 | Krótka nazwa obiektu SCHEDULED STOP POINT. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu SCHEDULED STOP POINT. |
| «AK» | *PublicCode* | *xsd:normalizedString* | 0:1 | Publiczny kod obiektu SCHEDULED STOP POINT. |
| «AK» | *PrivateCode* | *xsd:normalizedString* | 0:1 | Alternatywny kod obiektu SCHEDULED STOP POINT, może zostać wykorzystany do integracji z istniejącymi systemami. |
| «enum» | *StopType* | *StopPlace TypeEnum* | 0:1 | Rodzaj STOP PLACE. Dostępne |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *onstreetBus* (przystanek autobusowy),<br>• *onstreetTram* (przystanek tramwajowy),<br>• *busStation* (dworzec autobusowy),<br>• *airport* (port lotniczy),<br>• *railStation* (stacja kolejowa),<br>• *metroStation* (stacja metra),<br>• *coachStation* (dworzec autobusów dalekobieżnych),<br>• *ferryPort* (port promowy),<br>• *harbourPort* (nabrzeże portowe),<br>• *ferryStop* (przystanek promu),<br>• *liftStation* (przystanek dźwigu osobowego)<br>• *tramStation* (dworzec tramwajowy),<br>• *vehicleRailInterchange* (kolejowy węzeł przesiadkowy),<br>• *other* (inny). |
| | *Presentation* | *Presentati on* | 0:1 | Własności prezentacyjne określające sposób |

| Klasyfikac ja | Nazwa | Typ | Kardynalno ść | Opis |
|---|---|---|---|---|
| | | | | wyświetlania obiektu SCHEDULED STOP POINT. |

9.4.9. Połączenie rozkładowych punktów zatrzymania — ServiceLink

Połączenie (LINK) pomiędzy uporządkowaną parą obiektów SCHEDULED STOP POINT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Link* | ::> | SERVICE LINK dziedziczy po LINK. |
| «PK» | ***id*** | *ServiceLinkIdType* | 1:1 | Identyfikator obiektu SERVICE LINK. |
| «FK» | ***FromPointRef*** | *ScheduledStopPointRef* | 1:1 | Odniesienie do SCHEDULED STOP POINT na którym SERVICE LINK zaczyna trasę. |
| «FK» | ***ToPointRef*** | *ScheduledStopPointRef* | 1:1 | Odniesienie do SCHEDULED STOP POINT na którym SERVICE LINK kończy trasę. |

## 9.5. Przydzielanie zatrzymań

9.5.1. Przegląd

Model przydzielania zatrzymań odpowiada za powiązanie rozkładowych zatrzymań z fizycznie istniejącymi obiektami STOP PLACE i QUAY.

SCHEDULED STOP POINT występujący w rozkładzie może być przypisany do całego obiektu STOP PLACE albo do konkretnego QUAY znajdującego się wewnątrz.

9.5.2. Przypisanie punktu zatrzymania — StopAssignment (klasa abstrakcyjna)

STOP ASSIGNMENT to przyporządkowanie SCHEDULED STOP POINT do fizycznego przystanku. Jest specjalizowane jako PASSENGER STOP ASSIGNMENT oraz DYNAMIC STOP ASSIGNMENT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| :> | :> | *DataManagedObject* ASSIGNMENT dziedziczy po ASSIGNMENT | ::> | STOP |
| «PK» | *id* | *StopAssignmentIdType* | 1: 1 | Identyfikator obiektu STOP ASSIGNMENT |
| «PK» | *(order)* | *xsd:positiveInteger* | 1:1 | Odniesienie do zamówienia dla ASSIGNMENT (odziedziczone po ASSIGNMENT). Może zostać użyte do wskazania pierwszeństwa reguły. |
| «FK» | *ScheduledStopPointRef* | *ScheduledStopPointRef* | 1:1 | Odniesienie do SCHEDULED STOP POINT do którego obiekt ma być przypisany. |

### 9.5.3. Pasażerskie przypisanie punktu zatrzymania — PassengerStopAssignment

PASSENGER STOP ASSIGNMENT przydziela SCHEDULED STOP POINT do STOP PLACE i opcjonalnie QUAY.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *StopAssignment* | ::> | PASSENGER STOP ASSIGNMENT dziedziczy po STOP |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | ASSIGNMENT. |
| «PK» | *id* | *PassengerStopAssignmentIdType* | 1:1 | Identyfikator obiektu PASSENGER STOP ASSIGNMENT. |
| «FK» | *StopPlaceRe* | *StopPlaceRef* | *f* 1:1 | Odniesienie do STOP PLACE do którego SCHEDULED STOP POINT jest przypisany. |
| «FK» | *QuayRef* | *QuayRef* | 0:1 | Odniesienie do QUAY w ramach STOP PLACE do którego SCHEDULED STOP POINT jest przypisany. |
| «cntd» | *trainElement* | *TrainStopAssignment s* | 0:* | TRAIN STOP ASSIGNMENT powiązany z PASSENGER STOP ASSIGNMENT. Mogą one dostarczyć bardziej szczegółowe informacje o indywidualnych BOARDING POSITION i ustawieniu pojazdu. |

### 9.5.4. Dynamiczne przypisanie punktu zatrzymania — DynamicStopAssignment

Dynamiczna zmiana przypisania SCHEDULED STOP POINT do STOP PLACE, np. zmiana peronu. Obiekt nadpisuje istniejący PASSENGER STOP ASSIGNMENT dla wszystkich kursów przebiegających przez SCHEDULED STOP POINT lub tylko z wybranego JOURNEY PATTERN.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> ::> | | *PassengerStopAssignment* | ::> | DYNAMIC STOP ASSIGNMENT dziedziczy po PASSENGER STOP ASSIGNMENT. |
| «PK» | *id* | *DynamicStopAssignmentIdType* | 1: 1 | Identyfikator obiektu DYNAMIC STOP ASSIGNMENT. |
| «FK» | *JourneyPatternRef* | *JourneyPatternRef* | 0:1 | Odniesienie do obiektu JOURNEY PATTERN, do którego odnosi się DYNAMIC STOP ASSIGNMENT. |
| «FK» | *PassengerStopAssignmentRef* | *PassengerStopAssignmentRef* | 0:1 | Odniesienie do obiektu PASSENGER STOP ASSIGNMENT, który jest nadpisywany przez |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | DYNAMIC STOP ASSIGNMENT. |

9.5.5. Przypisanie pociągu do punktu zatrzymania — TrainStopAssignment (podkomponent PassengerStopAssignment)

TRAIN STOP ASSIGNMENT wiąże TRAIN COMPONENT w danym SCHEDULED STOP POINT z konkretnym STOP PLACE i ewentualnie QUAY. Element używany to wskazania miejsc wsiadania do części pociągu lub konkretnego wagonu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *StopAssignment* | ::> | TRAIN STOP ASSIGNMENT dziedziczy po STOP ASSIGNMENT. |
| «PK» | *id* | *TrainStopAssignmentIdType* | 1:1 | Identyfikator obiektu TRAIN STOP ASSIGNMENT. |
| «FK» | *ScheduledStopPointRef* | *ScheduledStopPointRef* | 0:1 | Odniesienie do obiektu SCHEDULED STOP POINT. |
| «FK» | *PassengerStopAssignmentRef* | *PassengerStopAssignmentRef* | 0:1 | Odniesienie do obiektu PASSENGER STOP ASSIGNMENT, którego jest to część. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***TrainRef*** | *TrainRef* | 0:1 | Odniesienie do obiektu TRAIN, do którego TRAIN STOP ASSIGNMENT ma zastosowanie. |
| | ***PositionOfTrainElement*** *xsd:positiveInteger* | | 0:1 | Względne położenie TRAIN ELEMENT w ramach obiektu TRAIN. |
| «FK» ***TrainComponentRef*** *TrainComponentRef* | | | 0:1 | Odniesienie do obiektu TRAIN COMPONENT, do którego TRAIN STOP ASSIGNMENT ma zastosowanie. |
| | ***EntranceToVehicle*** | *MultilingualString* | 0:1 | Konkretne wejście do pojazdu, np. *front* (przednie), *rear* (tylne). |

## 9.6. Connections, transfers and interchanging

### 9.6.1. Przegląd

Większość wyszukiwarek połączeń pozwala na określenie czasów na przesiadki. Model NeTEx zapewnia definiowanie obiektów TRANSFER DURATION służące wymianie informacji o

wymaganych czasach na przesiadkę na różnych poziomach szczegółowości (ogólny, na konkretnej stacji, pomiędzy parami przystanków).

### 9.6.2. Przejście piesze — Transfer (klasa abstrakcyjna)

TRANSFER reprezentuje parę punktów, które są dostatecznie blisko siebie, aby możliwe było przejście piesze pomiędzy nimi.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | TRANSFER dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *TransferIdType* | 1:1 | Identyfikator obiektu TRANSFER. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu TRANSFER. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu TRANSFER. |
| «FK» | ***TypeOfTransferRef*** | *TypeOfTransferRef* | 1:1 | Odniesienie do obiektu TYPE OF TRANSFER. |
| | ***Distance*** | *DistanceType* | 0:1 | Odległość pokonywana w ramach obiektu TRANSFER. |
| «cntd» | ***WalkTransferDuration*** | *TransferDuration* | 0:1 | Czas pieszego pokonania obiektu TRANSFER. |
| | ***BothWays*** | *xsd_boolean* | 0:1 | Czy TRANSFER może być pokonany w obydwu kierunkach? |
| «cntd» | ***(From)*** | *(TransferEnd)* | 1:1 | Początkowy kraniec obiektu TRANSFER, |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | specjalizowany w podklasach. |
| «cntd» | ***(To)*** | *(TransferEnd)* | 1: 1 Końcowy | kraniec obiektu TRANSFER, specjalizowany w podklasach. |

9.6.3. Czas trwania przejścia pieszego — TransferDuration (podkomponent T ransfer)

TRANSFER DURATION opisuje wymagany czas przesiadki dla różnych klas użytkowników.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***DefaultDuration*** | *xsd:duration* | 1:1 | Domyślny czas potrzebny na dokonanie przesiadki. |
| | ***FrequentTravellerDuration*** | *xsd:duration* | 0:1 | Potrzebny na dokonanie przesiadki przez pasażerów często podróżujących na danej trasie. |
| | ***OccasionalTravellerDuration*** | *xsd:duration* | 0:1 | Potrzebny na dokonanie przesiadki przez pasażerów rzadko podróżujących na danej trasie. |
| | ***MobilityRestrictedTravellerDuration*** | *xsd:duration* | 0:1 | Potrzebny na dokonanie przesiadki przez pasażerów o |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis ograniczone |
|---|---|---|---|---|

9.6.4. Przejście piesze pomiędzy punktami zatrzymania — Connection

CONNECTION opisuje fizyczną możliwość przesiadki z jednego pojazdu transportu publicznego na inny, indukowaną przez dwa obiekty SCHEDULED STOP POINT. Różne czas mogą być wymagane w zależności od rodzaju pasażera.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Transfer* | ::> | CONNECTION dziedziczy po TRANSFER. |
| «PK» | *id* | *ConnectionIdType* | 1:1 | Identyfikator obiektu CONNECTION. |
| «cntd» | *From* | *ConnectionEnd* | 1:1 | Własności początku przesiadki. |
| «cntd» | *To* | *ConnectionEnd* | 1:1 | Własności końca przesiadki. |

9.6.5. Kraniec przejścia pieszego pomiędzy punktami zatrzymania — ConnectionEnd (podkomponent Connection)

CONNECTION END opisuje każdy z krańców CONNECTION (którymi są obiekty SCHEDULED STOP POINT).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | *ScheduledStopPointRef* | *ScheduledStopPointRef* | 0:1 | Odniesienie do obiektu SCHEDULED STOP POINT, który stanowi kraniec CONNECTION. |

9.6.6. Przejście piesze pomiędzy miejscami — SiteConnection

SITE CONNECTION jest pojęciem podobnym do CONNECTION, ale opisuje fizyczne przejście pomiędzy miejscami (STOP PLACE lub POINT OF INTEREST), a nie zatrzymaniami rozkładowymi (SCHEDULED STOP POINT).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| :> | :> | *Transfer* | ::> | SITE CONNECTION dziedziczy po TRANSFER. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *ConnectionIdType* | 1:1 | Identyfikator obiektu SITE CONNECTION. |
| «cntd» | ***From*** | *Site ConnectionEnd* | 1:1 | Własności początku przesiadki. |
| «cntd» | ***To*** | *Site ConnectionEnd* | 1:1 | Własności końca przesiadki. |

### 9.6.7. Kraniec przejścia pieszego pomiędzy miejscami — SiteConnectionEnd (podkomponent SiteConnection)

***SiteConnectionEnd*** opisuje każdy z krańców SITE CONNECTION. Może odnosić się do STOP PLACE albo POINT OF INTEREST. W przypadku STOP PLACE, może być dodatkowo wsk atrybut MODE lub QUAY, jeżeli nie podano QUAY, to SITE CONNECTION opisuje przejście pomiędzy dowolnym QUAY, którego MODE zgadza się z tym podanym.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *ConnectionEnd* | ::> | SITE CONNECTIO N END dziedziczy po CONNECTIO N END. |
| «FK» | ***StopPlaceRef*** | *StopPlaceRef* | 0:1 | Odniesienie do STOP PLACE. |
| «FK» | ***StopPlaceSpaceRef*** | *StopPlaceSpaceRef* | 0:1 | Odniesienie do przestrzeni w STOP PLACE, może to być QUAY lub ACCESS SPACE. |
| «FK» | ***QuayRef*** | *QuayRef* | 0:1 | Odniesienie do QUAY. |
| «FK» | ***StopPlaceEntranceRef*** | *StopPlaceEntranceRef* | 0:1 | Odniesienie do STOP PLACE ENTRANCE. |
| «FK» | ***PointOfInterestRef*** | *PointOfInterestRef* | 0:1 | Odniesienie do POINT OF INTEREST. |

### 9.6.8. Domyślne przejście piesze pomiędzy środkami transportu — DefaultConnection

DEFAULT CONNECTION definiuje domyślny czas na przesiadkę pomiędzy środkami transportu w dowolnym miejscu — tam, gdzie nie zdefiniowano konkretnej wartości.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Transfer* | ::> | DEFAULT CONNECTION dziedziczy po TRANSFER. |
| «PK» | *id* | *ConnectionIdType* | 1: 1 | Identyfikator obiektu DEFAULT CONNECTION. |
| «cntd» | *From* | *DefaultConnectionEn* | 1:1 *d* | Własności początku przesiadki. |
| «cntd» | *To* | *DefaultConnectionEn* *TopographicPlaceRe* | 1:1 *d* 0:* *f* | Własności końca przesiadki. |
| «FK» | *TopographicPlace Vie w* | | | Obiekty TOPOGRAPHIC PLACE, dla których DEFAULT CONNECTION ma zastosowanie. |
| «FK» | *SiteElementRef* | *SiteElement* | 0:* | Obiekty SITE ELEMENT, dla których DEFAULT CONNECTION ma zastosowanie. |

### 9.6.9. Kraniec domyślnego przejścia pieszego pomiędzy środkami transportu — DefaultConnectionEnd (podkomponent DefaultConnection)

*DefaultConnectionEnd* opisuje kraniec DEFAULT CONNECTION.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «enum» | *TransportMonde VehicleModeEnum* | | 0:1 | Środek transportu. |

## 9.7. Kursy i czasy mijania — Vehicle journeys, passing times

### 9.7.1. Przegląd

VEHICLE JOURNEY (przejazd) to ruch pojazdu zgodnie z określonym JOURNEY PATTERN podążającym za ROUTE. VEHICLE JOURNEY jest zdefiniowane dla określonego DAY TYPE, zatem opisuje ruch pojazdu, który odbywa się o tej samej porze każdego dnia, który spełnia warunki określone w DAY TYPE.

VEHICLE JOURNEY dzieli się dwa podstawowe rodzaje:

- SERVICE JOURNEY (kurs) — dotyczy przewozu pasażerów;
- DEAD RUN (przejazd techniczny) — opisujące wszelkiego rodzaju przejazdy techniczne bez pasażerów.

SERVICE JOURNEY INTERCHANGE opisuje zaplanowaną możliwość przesiadki pomiędzy dwoma SERVICE JOURNEY na jednym lub różnych STOP POINT, pomiędzy konkretnymi STOP POINT IN JOURNEY PATTERN należącymi do SERVICE JOURNEY.

Do SERVICE JOURNEY można przypisać dostępne usługi za pomocą SERVICE FACILITY SET.

### 9.7.2. Przejazd — Journey (klasa abstrakcyjna)

Element *Journey* zawiera wspólne własności dla dowolnego kursu.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *LinkSequence* | ::> | JOURNEY dziedziczy po LINK SEQUENCE. |
| «enum» | *TransportMode VehicleModeEnum* | | 0:1 | Główny środek transportu (TRANSPORT MODE). |
| «enum» | *submode* | *TransportSubmodeEnu* | 0:1 *m* | Podrzędne środki transportu powiązane z głównym. |
| «cntd» | *noticeAssignment NoticeAssignmentView* | | 0:* *s* | Obiekty NOTICE |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opi |
|---|---|---|---|---|
| | | | | ASSIGNMEN T mające zastosowanie do całego obiektu JOURNEY. |

9.7.3. Kurs — ServiceJourney

SERVICE JOURNEY to kurs VEHICLE JOURNEY, podczas którego dozwolone jest wsiadanie lub wysiadanie pasażerów na przystankach.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Journey* | ::> | SERVICE JOURNEY dziedziczy po JOURNEY. |
| «PK» | ***id*** | *ServiceJourneyIdType* | 1:1 | Identyfikator obiektu SERVICE JOURNEY. |
| | ***DepartureTime*** *xsd:time* | | 0:1 | Czas odjazdu. |
| | ***DepartureDayOffset*** *DayOffsetType* | | 0:1 | Przesunięcie dnia odjazdu SERVICE JOURNEY w stosunku do OPERATING DAY. |
| | ***JourneyDuration*** *xsd:duration* | | 0:1 | Całkowity czas trwania podróży. |
| «cntd» ***dayTypes*** | | *DayTypeRef* | 0:1 | Obiekty DAY TYPE związane z SERVICE JOURNEY. |
| «FK» | ***JourneyPatternRef*** *JourneyPatternRef* | | 1:1 | Odniesienie do obiektu JOURNEY PATTERN. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | *Vehicle TypeRef* | *VehicleTypeRef* | 0:1 | Odniesienie do obiektu TYPE OF VEHICLE używanego przez SERVICE JOURNEY. |
| «FK» | *OperatorRef* | *OperatorView* | 0:1 | Odniesienie do obiektu OPERATOR. |
| «cntd» | *trainNumbers* | *TrainNumberRef* | 0:* | Obiekty TRAIN NUMBER związane z wszystkimi JOURNEY PART danej podróży — do wykorzystania w przypadkach podróży złożonych z wielu części. |
| «cntd» | *passingTimes* | *TimetabledPassing Time* | 1:* | Obiekty PASSING TIME związane z SERVICE JOURNEY. |
| «cntd» | *parts* | *JourneyPart* | 0:* | Obiekty JOURNEY PART związane z podróżą — do wykorzystania w przypadkach podróży złożonych z wielu części. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***facilities*** | *ServiceFacilitySet* | 0:* | Usługi dostępne w ramach SERVICE JOURNEY. |
| | ***TrainSize*** | *TrainSize* | 0:1 | Wymagany TRAIN SIZE. |

9.7.4. Czas mijania — PassingTime (klasa abstrakcyjna)

Wiele systemów informacji pasażerskiej wymaga danych o planowanych i faktycznych czasach przyjazdu i odjazdu pojazdów. Zazwyczaj nie są wymagane jednak szczegółowe dane używane do pełnej konstrukcji rozkładu jazdy. W modelu koncepcyjnym wszystkie te czasy reprezentowane są przez PASSING TIME, które wskazują czas pojazdu w danym punkcie JOURNEY PATTERN. Te czasy mogą być różnego rodzaju, np. przyjazd, odjazd, oczekiwany przyjazd, potwierdzony odjazd.

PASSING TIME wiąże czas z POINT IN JOURNEY PATTERN.

PASSING TIME, który jest rezultatem skonstruowanego rozkładu i jest ogłoszony publicznie to TIMETABLED PASSING TIME.

TIMETABLED PASSING TIME są co do zasady utworzone na długo przed dniem kursowania i ważne są przez dłuższy okres (np. rozkład letni, zimowy).

Szczególnym przypadkiem brzegowym jest dzień kursowania, kiedy następuje zmiana czasu z zimowego na letni lub odwrotnie. Norma zakłada, że dla jednego SERVICE JOURNEY wszystkie PASSING TIME muszą odnosić się do jednego referencyjnego czasu. Zatem jeżeli zmiana czasu następuje o godzinie 2.00, poprawne PASSING TIME to 1.45 - 1.55 - 2.05 - 2.15, a nie: 1.45 - 1.55 - 3.05 - 3.15 lub 1.45 - 1.55 - 1.05 - 1.15.

Aby uniknąć niejasności, mogą być przygotowywane specjalne rozkłady na dzień zmiany czasu, które poprawnie opisują faktyczny ruch pojazdu. W ten sposób można zapobiec fikcyjnym godzinnym opóźnieniom wszystkich pojazdów w noc zmiany czasu zimowego na letni.

| Klasyfikacja ::> | Nazwa ::> | Typ ::> *VersionedChild* | Kardynalność ::> | Opis |
|---|---|---|---|---|
| | | | | PASSING TIME dziedziczy po VERSIONED CHILD. |
| «FK» | ***PointInJoumeyPattemRef*** | *PointInLinkSequenceR ef* | 1:1 | POINT IN SEQUENC |

| Klasyfikacja | Nazwa | Typ | | Kardynalność | Opis |
|---|---|---|---|---|---|
| | | | | | E, którego dotyczy dany PASSING TIME. |

9.7.5. Rozkładowy czas mijania — TimetabledPassingTime (podkomponent ServiceJourney)

| Klasyfikacja | Nazwa | Typ | | Kardynalność | Opis |
|---|---|---|---|---|---|
| ::> | ::> | *PassingTime* | | ::> | TIMETABLED PASSING TIME dziedziczy po PASSING |
| «PK» | *id* | *TimetablePassingpe* | *TimeldTy* | 1:1 | Identyfikator obiektu TIMETABLED PASSING TIME. |
| | *ArrivalTime* | *xsd:time* | | 0:1 | Czas przyjazdu. |
| | *ArrivalDayOffset* | *DayOffsetType* | | 0:1 | Liczba dni po rozpoczęciu podróży po której następuje przyjazd, jeżeli jest to ten sam dzień: |
| | *DepartureTime* | *xsd:time* | | 0:1 | Czas odjazdu. |
| | *DepartureDayOffset* | *DayOffsetType* | | 0:1 | Liczba dni po rozpoczęciu podróży po której |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | odjazd, jeżeli jest to ten sam dzień: wartość 0. |
| «cntd» | *Headway* | *HeadwayInterval* | 0:1 | Częstotliwość odjazdów w POINT IN PATTERN. |

9.7.6. Grupa kursów — GroupOfServices

Grupa obiektów SERVICE JOURNEY, używana w szczególności do organizacji w zbiory podobnych kursów w celu prezentacji tabelarycznej lub w systemach informacji pasażerskiej.

Zbiór wszystkich SERVICE JOURNEY składających się na obsługę jednej LINE (kursy we wszystkie dni, w obu kierunkach) zazwyczaj będą podzielone na pewną liczbę różnych GROUP OF SERVICES (np. z podziałem na rodzaje dni, kierunki, warianty).

| Klasyfikacja :> | Nazwa :> | Typ *GroupOfEntities* | Kardynalność ::> | Opis |
|---|---|---|---|---|
| | | | | GROUP OF SERVICES dziedziczy po GROUP OF ENTITIES. |
| «PK» | *id* | *GroupOfServicesIdTyp* | 1: 1 *e* | Identyfikator obiektu GROUP OF SERVICES. |
| | *origin* | *EndPoint DerivedView* | | Widok pochodnych danych dotyczących początku podróży. |
| | *destination* | *EndPoint_DerivedView* | | Widok pochodnych danych dotyczących końca podróży. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | *destinationDisplays* | *DestinationDisplay* | 0:* | DESTINATION DISPLAY powiązany z GROUP OF SERVICES. |
| «cntd» | *members* | *VehicleJoumeyRef* | 0:* | Obiekty należące do GROUP OF SERVICES. |
| «cntd» | *noticeAssignments* | *NoticeAssignment View* | 0:* | Obiekty NOTICE ASSIGNMENT mające zastosowanie do GROUP OF SERVICES. |

9.7.7. Element grupy kursów — GroupOfServicesMember (podkomponent GroupOfServices)

Powiązanie relacji należenia do GROUP OF SERVICES przez obiekty SERVICE JOURNEY.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *GroupOfEntitiesMember* | ::> | GROUP OF SERVICES MEMBER dziedziczy po GROUP OF ENTITIES MEMBER. |
| «PK» | *id* | *GroupOfServicesMemberIdType* | 1:1 | Identyfikator obiektu GROUP OF SERVICES MEMBER. |
| «PK» | *(order)* | *xsd:positiveInteger* | 1:1 | Liczba porządkowa. |
| «FK» | *JourneyRef* | *JourneyRef* | 1:1 | Odniesienie do JOURNEY. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***noticeAssignments*** | *NoticeAssignment View* | 0:* | NOTICE ASSIGNMENT dotyczące GROUP OF SERVICE. |

## 9.8. Bardzo częste kursy

### 9.8.1. Wprowadzenie

W przypadkach, kiedy pojazdy obsługujące daną linię kursują bardzo często, a ich rozkład jest zdefiniowany poprzez częstotliwość, a nie sztywne godziny odjazdów poszczególnych kursów, systemy informacji pasażerskiej podają tę częstotliwość: "kursuje co 3 minuty". Należy przy tym zauważyć, że taki rozkład ma na celu uproszczenie informacji dla pasażerów, rozkłady wewnętrzne bez przeszkód mogą opierać się na konkretnych godzinach.

Norma przewiduje opcję agregowania wielu kursów w obiekt TEMPLATE JOURNEY, który ma zdefiniowaną częstotliwość następowania po sobie kolejnych kursów za pomocą HEADWAY JOURNEY GROUP.

Zapewnienie kursowania linii w takcie ułatwia pasażerom planowanie podróży i zapamiętywanie rozkładu oraz dodatkowo zmniejsza objętość przekazywanych danych rozkładowych.

### 9.8.2. Szablon kursu pojazdu — TemplateVehicleJourney

TEMPLATE VEHICLE JOURNEY to powtarzalne VEHICLE JOURNEY, dla których zdefiniowano częstotliwość jako JOURNEY FREQUENCY GROUP. Może reprezentować wiele kursów.

Częstotliwość jest definiowana przez JOURNEY FREQUENCY GROUP, które może być jednego z dwóch rodzajów:

- *headway* (bazujące na interwale, np. "co 15 minut");
- *rhythmic* (minuty po pełnych godzinach, np. "7, 22, 37, 52 minut po pełnej godzinie", obecnie niewykorzystywane).

| Klasyfikacja Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|
| ::> ::> | *Journey* | ::> | TEMPLATE VEHICLE JOURNEY dziedziczy po JOURNEY |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | *id* | *TemplateVehicleJourneyIdType* | 1:1 | Identyfikator obiektu TEMPLATE VEHICLE JOURNEY |
| «enum» | *TemplateVehicleJourneyType* | *TemplateVehicleJourneyTypeEnum* | 0:1 | Rodzaj TEMPLATE VEHICLE JOURNEY, obecnie jedyna wartość to *"headway".* |
| «cntd» | *frequencyGroups* | *JourneyFrequencyGroup* | 0:* | Instancje FREQUENCY GROUP definiujące TEMPLATE VEHICLE JOURNEY |

### 9.8.3. Grupa kursów częstotliwościowych — JourneyFrequencyGroup (klasa abstrakcyjna)

JOURNEY FREQUENCY GROUP definiuje zbiór instancji VEHICLE JOURNEY oraz początek i koniec okresu, podczas którego odbywają się kursy oparte o częstotliwość.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> ::> | | *GroupOfEntities* | ::> | JOURNEY FREQUENCY GROUP dziedziczy po GROUP OF ENTITIES. |
| «PK» | *id* | *FrequencyGroupIdType* | 1: 1 | Identyfikator obiektu |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | JOURNEY FREQUENCY GROUP. |
| | ***FirstDepartureTime*** | *xsd:time* | 0:1 | Czas pierwszego odjazdu w obiekcie FREQUENCY GROUP. |
| | ***LastDepartureTime*** | *xsd:time* | 0:1 | Czas ostatniego odjazdu w obiekcie FREQUENCY GROUP. |
| | ***DayOffset*** | *xsd:integer* | 0:1 | Przesunięcie daty ostatniego odjazdu w stosunku do pierwszego odjazdu. |
| | «cntd» ***journeys*** | *VehicleJoumeyRef* | 1:* | Instancje VEHICLE JOURNEY należące do JOURNEY FREQUENCY GROUP. |

### 9.8.4. Grupa kursów z interwałem — HeadwayJourneyGroup (podkomponent TemplateJourney)

HEADWAY JOURNEY GROUP to grupa instancji VEHICLE JOURNEY, które podążają tym samym JOURNEY PATTERN i mają taki sam interwał pomiędzy sobą.

| Klasyfikac ja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> ::> | | *JourneyFrequencyGroup* | ::> | HEADWAY JOURNEY GROUP dziedziczy po JOURNEY |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | FREQUENCY GROUP. |
| «PK» *id* | | *HeadwayJourneyGroupIdType* | 1:1 | Identyfikator obiektu HEADWAY JOURNEY GROUP. |
| | *ScheduledHeadwayInterval* | *xsd:duration* | 0:1 | Rozkładowy czas następstwa pomiędzy kolejnymi kursami. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu HEADWAY JOURNEY GROUP. |

## 9.9. Przesiadki pomiędzy kursami

### 9.9.1. Wprowadzenie

Jeżeli nie istnieje bezpośrednie SERVICE JOURNEY pomiędzy początkiem i końcem podróży, pasażer musi przesiąść się pomiędzy pojazdami. Pasażer wysiądzie z pojazdu obsługującego SERVICE JOURNEY na określonym SCHEDULED STOP POINT i przesiądzie się na pokład innego pojazdu, wykonującego inne SERVICE JOURNEY (zazwyczaj innej LINE) na tym samym lub innym SCHEDULED STOP POINT.

Przesiadka odbywa się w CONNECTION, które wskazuje, że dane dwa (w szczególności jeden, ten sam) SCHEDULED STOP POINT są odpowiednio blisko siebie, aby możliwe było przejście piesze pomiędzy nimi.

Jeżeli dwa konkretne SERVICE JOURNEY mają założone skomunikowanie w CONNECTION, może to zostać wyrażone za pomocą obiektu SERVICE JOURNEY INTERCHANGE.

Każde SERVICE JOURNEY INTERCHANGE dotyczy dwóch SERVICE JOURNEY. Pasażer przesiada się z dostarczającego (feeder) SERVICE JOURNEY do odbierającego *(distributor)* SERVICE JOURNEY.

Czasami SERVICE JOURNEY INTERCHANGE wyraża "przesiadkę" pomiędzy dwoma kursami, które są zaplanowane do obsługi przez ten sam pojazd. Taka sytuacja zdarza się w przypadku linii okrężnych lub połączonych kursów (wagony bezpośrednie). W takim przypadku informacja

pasażerska powinna wskazać, że pasażer nie musi się przesiadać (zobacz atrybut ***StaySeated*** poniżej).

### 9.9.2. Przesiadka — Interchange (klasa abstrakcyjna)

INTERCHANGE opisuje wspólne własności przesiadek, w szczególności czy przesiadka jest oficjalnie polecana, gwarantowana (skomunikowanie) wraz z czasem, jaki maksymalnie może czekać drugi pojazd na opóźniony.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| :> | :> | *DataManagedObject* ::> | | INTERCHANGE dziedziczy po DATA MANAGED OBJECT. |
| «PK» ***id*** | | *InterchangeIdType* | 1: 1 | Identyfikator obiektu INTERCHANGE. |
| «FK» ***ConnectionRef*** | | *ConnectionRef* | 0:1 | CONNECTION, którego dotyczy dana przesiadka. |
| | ***StaySeated*** | *xsd:boolean* | 0:1 | Czy pasażer może pozostać na swoim miejscu (nie przesiadać się do innego pojazdu)? |
| | ***Guaranteed*** | *xsd:boolean* | 0:1 | Czy przesiadka jest gwarantowana? |
| | ***CrossBorder*** | *xsd:boolean* | 0:1 | Czy przesiadka wymaga przekroczenia granicy państwowej? |
| | ***StandardTransferTime*** | *xsd:duration* | 0:1 | Standardowy czas potrzebny na przesiadkę. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *StandardWaitTime* | *xsd:duration* | 0:1 | Standardowy czas oczekiwania na INTERCHANGE. |
| | *MaximumWaitTime* | *xsd:duration* | 0:1 | Maksymalny czas oczekiwania na INTERCHANGE. |
| | *MaximumAutomaticWaitTime* | *xsd:duration* | 0:1 | Maksymalny czas automatycznego oczekiwania na INTERCHANGE. |

9.9.3. Przesiadka pomiędzy kursami — ServiceJourneyInterchange

SERVICE JOURNEY INTERCHANGE specjalizuje INTERCHANGE: określa, które obiekty POINT na PATTERN odpowiednich SERVICE JOURNEY biorą udział w przesiadce. W związku z tym, że jeden przystanek może być odwiedzany przez SERVICE JOURNEY wielokrotnie, SERVICE JOURNEY INTERCHANGE odnosi się do POINT IN PATTERN, a nie SCHEDULED STOP POINT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *Interchange* | ::> | SERVICE JOURNEY INTERCHANGE dziedziczy po INTERCHANGE. |
| «PK» | *id* | *ServiceJourneyInterchangeId Type* | 1:1 | Identyfikator obiektu SERVICE JOURNEY INTERCHANGE. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***FromPointRef*** | *ScheduledStopPointRef* | 1:1 | Odniesienie do obiektu SCHEDULED STOP POINT, na którym rozpoczyna się przesiadka. |
| | ***FromVisitNumber*** | *xsd:integer* | 0:1 | Numer wizyty pierwotnej podróży (konieczny tylko w przypadku wielorotnego odwiedzania tego samego przystanku). |
| «FK» | ***ToPointRef*** | *ScheduledStopPointRef* | 1:1 | Odniesienie do obiektu SCHEDULED STOP POINT, na którym kończy się przesiadka. |
| | ***ToVisitNumber*** | *xsd: integer* | 0:1 | Numer wizyty docelowej podróży (konieczny tylko w przypadku wielorotnego odwiedzania tego samego przystanku). |
| «FK» | ***FromJourneyRef*** | *ServiceJourneyRef* | 1:1 | Odniesienie do pierwotnej podróży (SERVICE JOURNEY), z której odbywa się przesiadka. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | *ToJourneyRef* | *ServiceJourneyRef* | 1:1 | Odniesienie do docelowej podróży (SERVICE JOURNEY), na którą odbywa się przesiadka. |

## 9.10. Pociągi i wagony komunikacji bezpośredniej

### 9.10.1. Przegląd

Obsługa połączeń kolejowych wymaga dodatkowych przypadków i pojęć, które nie występują w innych środkach transportu. Na przykład wagony bezpośrednie, dzielenie pociągów i numery pociągów, które są pojęciem różnym od identyfikatorów SERVICE JOURNEY i mogą zmieniać się w trakcie biegu pociągu.

SERVICE JOURNEY może zostać podzielone na pewną liczbę obiektów JOURNEY PART, każdy z własnym TRAIN NUMBER. JOURNEY PART należące do różnych SERVICE JOURNEY mogą być połączone poprzez encję JOURNEY PART COUPLE, w celu wskazania, że na danym odcinku są połączone w jeden pociąg. Ten sam TRAIN NUMBER może zostać przydzielony obu obiektom SERVICE JOURNEY poprzez przypisanie go do JOURNEY PART COUPLE.

### 9.10.2. Numer pociągu — TrainNumber

TRAIN NUMBER specyfikuje numer pociągu, który może zostać przypisany do VEHICLE JOURNEY obsługiwanego przez TRAIN. W komunikacji kolejowej, w szczególności w pociągach objętych rezerwacją miejsc, kluczowe jest zapewnienie pasażerom informacji o numerze

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ::> | *DataManaaedübiect* | ::> | TRAIN NUMBER dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *TrainNumberldType* | 1:1 | Identyfikator obiektu TRAIN NUMBER. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu TRAIN NUMBER. |
| | *ForAdvertisement* | *xsd:normalizedStrina* | 0:1 | Number pociągu używany publicznie, jeżeli |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | inny niż identyfikator. |

9.10.3. Część przejazdu — JourneyPart (podkomponent ServiceJourney)

JOURNEY PART stanowi część VEHICLE JOURNEY, którą tworzy się w celu rozróżnienia charakteru VEHICLE JOURNEY w trakcie jego biegu, np. w przypadku dzielenia lub łączenia pociągów. JOURNEY PART może być zdefiniowane przez początkowy i końcowy punkt SCHEDULED STOP POINT. Wymusza to, że dany SCHEDULED STOP POINT może być odwiedzony tylko jednokrotnie w ciągu biegu VEHICLE JOURNEY.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | JOURNEY PART dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *JourneyPartIdType* | 1:1 | Identyfikator obiektu JOURNEY PART. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu JOURNEY PART. |
| «FK» | *MainPartRef* | *JourneyPartCoupleRef* | 1:1 | Odniesienie do JOURNEY PART COUPLE głównego JOURNEY |
| «FK» | *JourneyPartCoupleRef* | *JourneyPartCoupleRef* | 0:1 | Odniesienie do JOURNEY PART COUPLE należącego do łączonego JOURNEY |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «FK» | ***TrainNumberRef*** | *TrainNumberRef* | 0:1 | Odniesienie do TRAIN NUMBER obiektu JOURNEY PART. |
| «FK» | ***FromStopPointRef*** | *ScheduledStopPointRef* | 0:1 | Odniesienie do początkowego SCHEDULED STOP POINT obiektu JOURNEY PART. |
| «FK» | ***ToStopPointRef*** | *ScheduledStopPointRef* | 0:1 | Odniesienie do końcowego SCHEDULED STOP POINT obiektu JOURNEY PART. |
| | ***StartTime*** | *xsd:time* | 1:1 | Czas początku obiektu JOURNEY PART. |
| | ***StartTimeDayOffset*** | *DayOffsetType* | 0:1 | Liczba dni od rozpoczęcia podróży do momentu początkowego JOURNEY PART. |
| | ***EndTime*** | *xsd:time* | 1:1 | Czas końca obiektu JOURNEY PART. |
| | ***EndTimeDayOffset*** | *DayOffsetType* | 0:1 | Liczba dni od rozpoczęcia podróży do momentu końcowego |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | JOURNEY PART. |
| «cntd» | *facilities* | *ServiceFacilitySet* | 0:* | Usługi dostępne podczas trwania JOURNEY PART. |
| «cntd» | *journeyPartPositions* | *JourneyPartPosition* | 0:* | Umiejscowienie obiektu JOURNEY PART w pociągu. |

9.10.4. Powiązanie części przejazdu — JourneyPartCouple

JOURNEY PART COUPLE definiuje co najmniej dwa obiekty JOURNEY PART należące do różnych VEHICLE JOURNEY, które są obsługiwane jednocześnie przez jeden pociąg (TRAIN) złożony z połączonych pojazdów (wagonów) poszczególnych VEHICLE JOURNEY.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> ::> | | *DataManagedObject* | ::> | JOURNEY PART COUPLE dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *JourneyPartCoupleIdType* | 1:1 | Identyfikator obiektu JOURNEY PART COUPLE. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu JOURNEY PART COUPLE. |
| | *StartTime* | *xsd:time* | 1:1 | Czas początku obiektu |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | JOURNEY PART COUPLE. |
| | ***StartTimeDayOffse*** | *DayOffsetType* ***t*** | 0:1 | Liczba dni od rozpoczęcia podróży do momentu początkowego JOURNEY PART COUPLE. |
| | ***EndTime*** | *xsd:time* | 1:1 | Czas końca obiektu JOURNEY PART COUPLE. |
| | ***EndTimeDayOffset*** | *DayOffsetType* | 0:1 | Liczba dni od rozpoczęcia podróży do momentu końcowego JOURNEY PART COUPLE. |
| «FK» | ***FromStopPointRef*** | *ScheduledStopPointRef* 0:1 | | Odniesienie do początkowego SCHEDULED STOP POINT obiektu JOURNEY PART COUPLE. |
| «FK» | ***ToStopPointRef*** | *ScheduledStopPointRef* 0:1 | | Odniesienie do końcowego SCHEDULED STOP POINT obiektu JOURNEY |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | PART COUPLE. |
| «FK» | *MainPartRef* | *JourneyPartCoupleRef* | 1:1 | Odniesienia do JOURNEY PART COUPLE głównej podróży JOURNEY PART COUPLE. |
| «cntd» | *journeyParts* | *JoumeyPartRef* | 0:* | Odniesienia do instancji JOURNEY PART, które łączy JOURNEY PART COUPLE. |
| «FK» | *TrainNumberRef* | *TrainNumberRef* | 0:1 | Odniesienia do TRAIN NUMBER. |

9.10.5. Umiejscowienie części przejazdu — JourneyPartPosition (podkomponent JourneyPartCouple)

JOURNEY PART POSITION opisuje względną pozycję JOURNEY PART w pociągu od określonego zatrzymania. W trakcie biegu pociągu pozycja ta może się zmieniać w związku ze zmianami kierunku jazdy, sprzęganiem i rozsprzęganiem części pociągu (grup wagonowych).

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VersionedChild* | ::> | JOURNEY PART POSITION dziedziczy po VERSIONED CHILD. |
| «PK» | *id* | *JoumeyPartPositionIdTyp e* | 1:1 | Identyfikator obiektu JOURNEY |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | PART POSITION. |
| «FK» | ***ScheduledStopPoint*** | *ScheduledStopPointRef* | 0:1 | Odniesienie do obiektu SCHEDULED STOP POINT od którego podana pozycja jest ważna. |
| | ***PositionInTrain*** | *xsd:integer* | 0:* | Pozycja JOURNEY PART wewnątrz pociągu od wskazanego wyżej SCHEDULED STOP POINT do odwołania. |

## 9.11. Pojazdy i pociągi

### 9.11.1. Przegląd

Modele dotyczące pojazdów i pociągów pozwalają na definiowanie własności pojazdów i zestawień pociągów. W EPIP używane są one do opisu obiektów SERVICE JOURNEY.

### 9.11.2. Typ pojazdu — VehicleType

Obiekty VEHICLE mogą być klasyfikowane przez VEHICLE TYPE na podstawie spełniania wymagań dotyczących pojemności i udogodnień dostępnych na pokładzie (np. *standard bud, double decker).* VEHICLE TYPE może być dołączony do konkretnego SERVICE JOURNEY, aby wskazać, że pojazd tego typu jest wymagany na danym kursie.

VEHICLE TYPE jest szczególnie istotny dla osób z ograniczoną mobilnością, które mogą wybierać połączenia tylko z pojazdami zapewniającymi pewien zestaw potrzebnych udogodnień, np. miejsce dla wózka inwalidzkiego.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | VEHICLE TYPE dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *Vehicle TypeldType* | 1:1 | Identyfikator obiektu VEHICLE TYPE. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu VEHICLE TYPE. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu VEHICLE TYPE. |
| | ***SelfPropelled*** | *xsd:boolean* | 0:1 | Czy pojazd posiada własny napęd? |
| «PK» | ***TypeOfFuel*** | *TypeOfFuelEnum* | 0:1 | Rodzaj paliwa. |
| | ***EuroClass*** | *xsd:normalizedString* | 0:1 | Norma EURO spełniana przez pojazd. |
| «cntd» | ***capacities*** | *PassengerCapacitv* | 0:* | Pojemność pojazdu z podziałem na FARE CLASS. |
| | ***LowFloor*** | *xsd:boolean* | 0:1 | Czy pojazd jest niskopodłogowy? |
| | ***HasLiftOrRamp*** | *xsd:boolean* | 0:1 | Czy pojazd jest wyposażony w podnośnik lub ramkę dla wózków inwalidzkich? |
| | ***HasHoist*** | *xsd:boolean* | 0:1 | Czy pojazd jest wyposażony w wciągarkę dla wózków inwalidzkich? |
| | ***Length*** | *LengthType* | 0:1 | Długość pojazdu. |
| | ***Width*** | *LengthType* | 0:1 | Szerokość pojazdu. |
| | ***Height*** | *LengthType* | 0:1 | Wysokość pojazdu. |
| | ***Weight*** | *WeightType* | 0:1 | Masa pojazdu. |

### 9.11.3. Pociąg — Train

TRAIN to pojazd złożony z obiektów TRAIN ELEMENT w określonej kolejności, np. wagony połączone ze sobą i ciągnięte przez lokomotywę albo zespół trakcyjny.

Część systemów informacji pasażerskiej na kolei udostępnia szczegółowe informacje dotyczące zestawienia pociągu i lokalizacji poszczególnych wagonów w obrębie pociągu, co umożliwia pasażerom szybsze zajęcie miejsc w odpowiedniej części pociągu (poprzez udaje się do odpowiednich sektorów peronu), znalezienie wagonu restauracyjnego, itp.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *VehicleTvpe* | ::> | TRAIN dziedziczy po VEHICLE TYPE. |
| «PK» | ***id*** | *TrainIdType* | 1:1 | Identyfikator obiektu TRAIN. |
| | ***NumberOfCars*** | *xsd:nonNegativeInteger* | 0:1 | Liczba wagonów w pociągu. |
| «enum» | ***TrainSizeType*** | *TrainSizeEnum* | 0:1 | Kategoryzacja rozmiaru pociągu. Dostępne wartości: • *normal* (standardowy) • *short* (krótki), • *long* (długi). |
| «cntd» ***components*** *TrainComponent* | | | 0:* | Części (komponenty) pociągu. |

9.11.4. Element pociągu — TrainElement (podkomponent Train)

TRAIN ELEMENT to elementarna część pociągu, np. wagon, lokomotywa, człon zespołu trakcyjnego.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | TRAIN ELEMENT dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *TrainElementIdType* | 1:1 | Identyfikator obiektu TRAIN ELEMENT. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu TRAIN ELEMENT. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu TRAIN ELEMENT. |
| «enum» | ***TrainElementType*** | *TypeOfTrainElementEnum* | 1:1 | Rodzaj TRAIN ELEMENT. Dostępne wartości: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *buffetCar* (wagon barowy),<br>• *carriage* (wagon),<br>• *engine* (lokomotywa),<br>• *carTransporter* (wagon do przewozu samochodów),<br>• *sleeperCarriage* (wagon sypialny),<br>• *luggageVan* (wagon bagażowy),<br>• *restaurantCarriage* (wagon restauracyjny),<br>• *other* (inny). |
| | ***Length*** | *LengthType* | 0:1 | Długość obiektu TRAIN ELEMENT. |
| | ***Width*** | *LengthType* | 0:1 | Szerokość obiektu TRAIN ELEMENT. |
| | ***FareClasses*** | *FareClassEnum* | 0:* | FARE CLASS dostępne w TRAIN ELEMENT. |

### 9.11.5. Komponent pociągu — TrainComponent (podkomponent Train)

TRAIN COMPONENT specyfikuje kolejność obiektów TRAIN ELEMENT wewnątrz instancji TRAIN. W EPIP TRAIN ELEMENT powinny być wbudowane w obiekt TRAIN COMPONENT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| :> | :> | *VersionedChild* | :> | TRAIN COMPONENT dziedziczy po VERSIONED CHILD. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | *id* | *Train ComponentIdType* | 1:1 | Identyfikator obiektu TRAIN COMPONENT. |
| | *order* | *xsd:positiveInteger* | 0:1 | Kolejność komponentu w obrębie pociągu. |
| | *Label* | *MultilingualString* | 0:1 | Etykieta obiektu TRAIN COMPONENT. |
| | *Description* | *MultilingualString* | 0:1 | Opis obiektu TRAIN COMPONENT. |
| «cntd» | *TrainElement* | *TrainElement* | 1:1 | TRAIN ELEMENT powiązany z TRAIN COMPONENT. |

## 9.12. Uwagi

### 9.12.1. Wprowadzenie

Uwaga (NOTICE) definiuje teksy wielokrotnego użytku, które mogą być dołączone do rozkładów jazdy i innych encji jako przypisy, użyte jako zapowiedzi itp. NOTICE jest powiązane z encjami za pomocą obiektów NOTICE ASSIGNMENT. NOTICE może być sklasyfikowane za pomocą TYPE OF NOTICE.

Każde NOTICE może zawierać kilka alternatywnych formatów specyfikowanych przez DELIVERY VARIANT, np. inne treści dla wyświetlaczy, drukowanych rozkładów, zapowiedzi megafonowych, do użytku w aplikacjach mobilnych.

### 9.12.2. Uwaga — Notice (podkomponent ogólny)

NOTICE przenosi dodatkową informację dla pasażerów, która może pomóc im w planowaniu podróży lub podczas niej. NOTICE może być za pomocą NOTICE ASSIGNMENT powiązane z zatrzymaniami jednego kursu (POINT IN JOURNEY PATTERN), zatrzymań wielu kursów (COMMON SECTION), do całego JOURNEY PATTERN, do jednego VEHICLE JOURNEY albo GROUP OF SERVICES, a także do INTERCHANGE. Każde NOTICE ASSIGNMENT może być ograniczone ważnością przez VALIDITY CONDITION.

Jedno NOTICE może być użyte przez wiele instancji NOTICE ASSIGNMENT.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | NOTICE dziedziczy po DATA MANAGED OBJECT. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «PK» | ***id*** | *NoticeIdType* | 1:1 | Identyfikator obiektu NOTICE. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu NOTICE. |
| | ***Text*** | *MultilingualString* | 0:1 | Treść obiektu NOTICE. Dla tłumaczeń można użyć ***A lternativeText.*** |
| «AK» | ***PublicCode*** | *xsd:normalizedString* | 0:1 | Kod publiczny obiektu NOTICE. |
| «FK» | ***TypeOfNoticeRef*** | *TypeOfNoticeRef* | 0:1 | Odniesienie do obiektu TYPE OF NOTICE. |

9.12.3. Przypisanie uwagi — NoticeAssignment (podkomponent ogólny)

Przypisanie NOTICE do ENTITY. W EPIP ***NoticeAssignment*** powinno być zawsze wbudowane w element, który opisuje.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | NOTICE ASSIGNMENT dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *NoticeAssignmentIdType* | 1:1 | Identyfikator obiektu NOTICE ASSIGNMENT |
| «FK» | ***NoticeRef*** | *NoticeRef* | 0:1 | Odniesienie do obiektu NOTICE. Atrybut używany tylko w przypadku współdzielonyc h obiektów NOTICE, w |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | przeciwnym wypadku należy używać atrybutu ***Notice*** dla osadzonego obiektu NOTICE. |
| | ***Notice*** | *Notice* | 0:1 | Obiekt NOTICE. |
| «FK» | ***NoticedObjectRef*** | *Version OfObjectRef* | 0:1 | Odniesienie do obiektu, którego dotyczy NOTICE. |
| «FK» | ***StartPointInPatternRef*** | *PointInSequenceRef* | 0:1 | Odniesienie do punktu, od którego obiekt NOTICE ma zastosowanie. |
| «FK» | ***EndPointInPatternRef*** | *PointInSequenceRef* | 0:1 | Odniesienie do punktu, do którego obiekt NOTICE ma zastosowanie. |

## 9.13. Kalendarze

### 9.13.1. Przegląd

Kalendarz kursowania (SERVICE CALENDAR) zawiera informacje dotyczące dni kursowania. Transmodel oddziela koncepcje typów dni, dla których odpowiedni jest poziom obsługi (rozkład, tabor) od konkretnych dat. SERVICE JOURNEY jest powiązane wyłącznie z DAY TYPE. DAY TYPE ASSIGNMENT wiąże DAY TYPE z konkretną datą kalendarzową.

Takie podejście pozwala, aby jedna definicja SERVICE JOURNEY była używana z różnymi SERVICE CALENDAR w kolejno następujących okresach.

### 9.13.2. Typ dnia — DayType

DAY TYPE jest definiowany jako kombinacja różnych własności, które może posiadać konkretny dzień i które wpływają na zapotrzebowane na transport i warunki jego prowadzenia (np. natężenie ruchu drogowego wpływające na czas jasy autobusów).

Każdy pojedynczy warunek istotny dla popytu jest zapisywany jako PROPERTY OF DAY. Na przykład każdy z warunków: "*workday*■", "*Sunday*■", "*school holidays*", *"market day* będzie osobnym PROPERTY OF DAY. DAY TYPE może zostać utworzone poprzez kombinację wielu warunków, na przykład "dzień roboczy w trakcie wakacji szkolnych będący dniem targowym" powstanie poprzez dodanie do DAY TYPE odpowiednich PROPERTY OF DAY.

Najbardziej podstawowym PROPERTY OD DAY jest DAY OF WEEK (np. *"Wednesday"),* inne popularne to *"public holiday*", *"school term day*. Jeden DAY TYPE może mieć przyporządkowane więcej niż jedną wartość danego typu własności, np. wtorek lub czwartek (" *Tuesday* OR *Thursday*[1]*').*

Wiele wartości ***PropertyOfDay*** w jednym ***DayType*** opisują złożony warunek:

- wartości zdefiniowane w jednej instancji są iloczynem logicznym (koniunkcją), np. *"(Monday* AND *public holiday* AND *winter)";*
- osobne instancje własności są sumą logiczną (alternatywą), np. *"((Sunday* AND *not holiday* AND *winter)* OR *(Monday* AND *public holiday* AND *summer))".*

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObject* | ::> | DAY TYPE dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *DayTypeIdType* | 1:1 | Identyfikator obiektu DAY TYPE. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu DAY TYPE. |
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu DAY TYPE. |
| | ***EarliestTime*** | *xsd:time* | 0:1 | Najwcześniejsze rozpoczęcie dnia. |
| | ***DayLength*** | *xsd:duration* | 0:1 | Długość dnia. |
| «cntd» | ***properties*** | *PropertyOfDay* | 0:* | Własności obiektu DAY TYPE. |
| «cntd» | ***timebands*** | *Timeband* | 0:* | Instancje TIME BAND obiektu DAY TYPE. |

### 9.13.3. Własność dnia — PropertyOfDAy (podkomponent DayType)

Własność, którą może posiadać dzień, np. wakacje szkolne, święta państwowe, dni tygodnia, pora roku. Może zostać użyta do wygenerowania opisów w językach naturalnych.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu PROPERTY OF DAY. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***Description*** | *MultilingualString* | 0:1 | Opis obiektu PROPERTY OF DAY. |
| «enum» | ***DaysOfWeek*** | *DayOfWeekEnum* | 0:7 | Dni tygodnia przypisane do własności. |
| «enum» | ***WeeksOfMonth*** | *WeekOfMonthEnum* | 0:5 | Tygodnie miesiąca (1-5) przypisane do własności. |
| | ***MonthOfYear*** | *xsd:gMonth* | 0:1 | Miesiąc roku dla typów dni występujących tego samego miesiąca każdego roku. |
| | ***DayOfMonth*** | *xsd:gDay* | 0:1 | Dzień miesiąca dla typów dni występujących tego samego dnia każdego miesiąca. |
| | ***DayOfYear*** | *xsd:gMonthDay* | 0:1 | Dzień roku dla typów dni występujących tego samego dnia każdego roku. |
| «enum» | ***CountryRef*** | *CountryEnum* | 0:* | Państwo zapewniające kontekst dla dni świątecznych. |
| «enum» | ***HolidayTypes*** | *HolidayTypeEnum* | 0:5 | Rodzaje dni świątecznych. Dostępne wartości: |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | • *RegionalHoliday* (święto regionalne),<br>• *NationalHoliday* (święto państwowe),<br>• *HolidayDisplacement Day* (dzień odbioru za święto)<br>• *EveOfHoliday* (wigilia święta). |
| «enum» *Seasons* | | *SeasonEnum* | 0:4 | Pora roku. Dostępne wartości:<br>• *Spring* (wiosna),<br>• *Summer* (lato),<br>• *Autumn* (jesień),<br>• *Winter* (zima),<br>• *Perennially* (cały rok). |
| «enum» *Crowding* | | *CrowdingEnum* | 0:1 | Względne zatłoczenie danego dnia. Dostępne wartości:<br>• *veryQuiet* (bardzo niskie),<br>• *quiet* (niskie),<br>• *normal* (normalne),<br>• *busy* (wysokie),<br>• *veryBusy* (bardzo wysokie). |

### 9.13.4. Okres dnia — Timeband (podkomponent DayType)

Okres dnia, istotny z punktu widzenia transportu publicznego, np. podobne warunki ruchowe lub kategoria taryfowa.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | TIME BAND dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *TimebandIdType* | 1:1 | Identyfikator obiektu TIME BAND. |
| | *StartTime* | *xsd:time* | 0:1 | Czas rozpoczęcia (włącznie). |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | *EndTime* | *xsd:time* | 0:1 | |
| | *DayOffset* | *xsd:integer* | 0:1 | Czas zakończenia (włącznie). Różnica dni pomiędzy czasem zakończenia a rozpoczęcia, jeśli ten sam dzień: wartość 0. |

### 9.13.5. Kalendarz kursowania — ServiceCalendar

SERVICE CALENDAR grupuje encje związane z kalendarzem i zapewnia określony zbiór przypisań instancji OPERATING DAY do DAY TYPES.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| :> :> | | *DataManagedObject* ::> | | SERVICE CALENDAR dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *ServiceCalendarIdTyp* | 1:1 *e* | Identyfikator obiektu SERVICE CALENDAR. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa obiektu SERVICE CALENDAR. |
| | *FromDate* | *xsd:date* | 0:1 | Początek ważności SERVICE CALENDAR (włącznie). |
| | *ToDate* | *xsd:date* | 0:1 | Koniec ważności SERVICE CALENDAR (włącznie). |
| «cntd» *dayTypes* | | *DayType* | 0:* | Instancje DAY TYPE w SERVICE CALENDAR. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***operatingDays*** | *OperatingDay* | 0:* | Instancje OPERATING DAY w SERVICE CALENDAR. |
| «cntd» | ***operatingPeriods*** | *OperatingPeriod* | 0:* | Instancje OPERATING PERIOD w SERVICE CALENDAR. |
| «cntd» | ***dayTypeAssignment*** | *DayTypeAssignment* 0:* ***s*** | | Instancje DAY TYPE ASSIGNMENT w SERVICE CALENDAR. |

### 9.13.6. Dzień kursowania — OperatingDay (podkomponent ServiceCalendar)

OPERATING DAY w większości przypadku odpowiada dniu kalendarzowemu, to jest konkretnej dacie w kalendarzu, np. 2 maja 2022, ale może mieć zdefiniowane różne czasy początku i końca.

Proces przypisywania DAY TYPE do poszczególnych OPERATING DAY pozwala na wybór odpowiednich rozkładów dla spodziewanego popytu i sytuacji ruchowej

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | OPERATING DAY dziedziczy po DATA MANAGED OBJECT. |
| «PK» | ***id*** | *OperatingDayIdType* | 1:1 | Identyfikator obiektu OPERATING DAY. |
| | ***CalendarDate*** | *xsd:date* | 1:1 | Data kalendarzowa. |
| «FK» | ***ServiceCalendarRef*** | *CalendarRef* | 0:1 | SERVICE CALENDAR do którego należy obiekt OPERATING DAY. |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu OPERATING DAY. |
| | ***EarliestTime*** | *xsd:time* | 1:1 | Rozpoczęcie dnia. |
| | ***DayLength*** | *xsd:duration* | 1:1 | Długość dnia. |

9.13.7. Przypisanie typu dnia — DayTypeAssignment (podkomponent ServiceCalendar)

Przypisanie DAY TYPE do konkretnej daty, np. OPERATING DAY.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | Typ _____ *VersionedChild* | ::> | DAY TYPE ASSIGNMENT dziedziczy po VERSIONED CHILD. |
| «PK» | ***id*** | *ShortTermAssignmentIdType* | 1:1 | Identyfikator obiektu DAY TYPE ASSIGNMENT. |
| | ***Name*** | *MultilingualString* | 0:1 | Nazwa obiektu DAY TYPE ASSIGNMENT. |
| «FK» | ***ServiceCalendarRef*** | *CalendarRef* | 0:1 | Odniesienie do SERVICE CALENDAR do którego należy obiekt DAY TYPE ASSIGNMENT. |
| «FK» | ***OperatingPeriodRef*** | *OperatingDayRef* | 1:1 | Odniesienie do OPERATING PERIOD do |

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | | | którego należy obiekt DAY TYPE ASSIGNMENT. |
| | *Date* | *xsddate* | 1:1 | Data, może być użyta zamiast OPERATING DAY. |
| «FK» | *DayTypeRef* | *DayTypeRef* | 1:1 | Odniesienie do DAY TYPE do którego należy obiekt DAY TYPE ASSIGNMENT. |
| | *IsAvailable* | *xsdboolean* | 0:1 | Czy dostępny danego dnia. |

9.13.8. Okres kursowania — OperatingPeriod (klasa abstrakcyjna w EPIP)

Ciągły okres pomiędzy dwoma instancjami OPERATING DAY, który używany jest do definiowania ważności rozkładu jazdy.

W EPIP za każdym razem używana będzie specjalizacja tej klasy w postaci *UicOperatingPeriod.*

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *DataManagedObiect* | ::> | OPERATING PERIOD dziedziczy po DATA MANAGED OBJECT. |
| «PK» | *id* | *PeriodIdType* | 1:1 | Identyfikator obiektu OPERATING PERIOD. |
| | *Name* | *MultilingualString* | 0:1 | Nazwa obiektu OPERATING PERIOD. |
| | *FromDate* | *xsd:dateTime* | 1:1 | Data początkowa (włącznie). |
| | *ToDate* | *xsd:dateTime* | 1:1 | Data końcowa (włącznie). |

### 9.13.9. Okres kursowania UIC — UicOperatingPeriod (podkomponent ServiceCalendar)

*UicOperatingPeriod* to zoptymalizowana forma reprezentacji *OperatingPeriod*, która używa ciągu bitów (ciąg znaków "0" i "1") do wskazania, w których dniach określone *DayType* ma zastosowanie, analogiczna do formatu zapisu używanego przez UIC.

Ta forma jest zalecana w EPIP, ponieważ jest zwięzła, prosta w implementacji i zgodna z wieloma dotychczas istniejącymi systemami.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| ::> | ::> | *OperatingPeriod* ::> | | UIC OPERATING PERIOD dziedziczy po UIC OPERATING PERIOD. |
| | *ValidDayBits* | *xsd:normalizedString* | 1: 1 | Ciąg znaków "0" i "1", jeden dla każdego dnia w okresie: wartość informuje, czy DAY TYPE jest ważne danego dnia. Zasadniczo długość powinna odpowiadać liczbie dni pomiędzy datami: początkową i końcową. Jeżeli jest krótsza, należy założyć, że DAY TYPE jest dostępne. |
| | *DaysOfWeek* | *DaysOfWeek* | 0:1 | Dni tygodnia odpowiadające bitom danych (do pierwszych siedmiu). |

# 10. WSPÓLNE ZASADY

## 10.1. Zarządzanie wersjami

Zarządzanie wersjami w profilu opiera się na encjach VERSION FRAME.

Podstawowe zasady przedzielania wersji:

- Wszystkie obiekty powinny posiadać unikalny identyfikator *id.* Dla niektórych komponentów potomnych, zamiast *id,* może być definiowany unikalny wewnątrz elementu nadrzędnego *order,* gdzie **id+order** jest unikalne. Identyfikator powinien być stały podczas całego cyklu życia obiektu: wszystkie kolejne wymiany danych i aktualizacje obiektu nie powinny zmieniać identyfikatora.
- Każdy obiekt posiadający *id,* powinien posiadać również *version.* Jeżeli wersja jest nieznana lub zarządzana poza profilem, należy użyć wartości *"any'.*
- Wszystkie wersje obiektów zawartych w *VersionFrame* powinny być spójne, używalne jako całość. Nie oznacza to jednak, że wartości pola *version* dla każdego obiektu muszą być takie same.
- Każda ramka *VersionFrame* musi mieć konkretną wartość w polu *version* (inną niż *"any).*
- Wersja powinna być liczbą (ciągiem cyfr).
- Numer każdej kolejnej wersji dla danej *VersionFrame* musi być ostro większy od numeru poprzedniej (funkcja monotonicznie rosnąca).

Modyfikacja obiektu powinna skutkować powiększeniem wersji danego obiektu oraz obiektów nadrzędnych, których to dotyczy. Takie działanie jest podyktowane efektywnością wykrywani a zmian przez systemy czerpiące dane. Zasady modyfikacji wersji są następujące:

- Jeżeli którakolwiek wartość atrybutu obiektu się zmienia, jego wersja powinna zostać zwiększona.
- Jeżeli którekolwiek odniesienie wewnątrz obiektu się zmienia (zostaje dodane, usunięte lub zmienia się obiekt, na który wskazuje), jego wersja powinna zostać zwiększona.
- Jeżeli zmienia się numer wersji wskazany w którymkolwiek odniesieniu wewnątrz obiektu (tj. odniesienie wskazuje wciąż na ten sam obiekt, ale na inną jego wersję), we rsja obiektu nadrzędnego również powinna zostać zwiększona.
- Jeżeli obiekt jest złożony, tj. zawiera agregację potomków tworzących jego definicję (np. SERVICE CALENDAR jest złożony z instancji DAY TYPE), to każda zmiana podkomponentu powinna powodować zwiększenie wersji obiektu.

W skrócie każda zmiana obiektu powoduje zwiększenie jego wersji oraz populację tej zmiany i podwyższenia wersji na wszystkie nadrzędne obiekty.

## 10.2. Używanie warunków ważności

Pełny NeTEx pozwala na opisywanie skomplikowanych reguł i warunków ważności. Na potrzeby profilu krajowego wybrano prosty podzbiór zgodny z EPIP.

Stosowanie VALIDITY CONDITION podlega poniższym zasadom:

- Do instancji FRAME dołączany jest dominujący warunek ważności (**dominant** VALIDITY CONDITION). Jeżeli obiekty zawarte wewnątrz ramki mają zdefiniowane VALIDITY CONDITION sprzeczne z dominującym, zostaną zignorowane.
- Każda instancja FRAME powinna zawierać co najmniej jeden dominujący warunek ważności, który z kolei powinien mieć zdefiniowaną datę początkową.
- Jeżeli okresy ważności kolejnych wersji tej samej instancji FRAME nachodzą na siebie, okres zdefiniowany w nowszej wersji całkowicie nadpisuje okres wynikający z poprzedniej. Na przykład, jeżeli pierwsza wersja jest ważna od 1 stycznia 2021 do 31 grudnia 2022, a druga od 10 grudnia 2022 do 31 grudnia 2023, to okres ważności pierwszej zostaje ograniczony do 9 grudnia 2022, a okres drugiej nie ulega zmianie.
- Warunek ważności dołączony do COMPOSITE FRAME jest aplikowany do wszystkich zawartych FRAME i jest dominujący wobec ich VALIDITY CONDITION.
- Warunki zawarte wewnątrz jednego obiektu nie mogą być sprzeczne, np. jeden obiekt nie może mieć jednocześnie *"only in summer'* i *"only in winter'.*

## 10.3. Używanie znaczników czasu stworzenia i modyfikacji

Wszystkie instancje ***DataManagedObject*** posiadają znaczniki czasu opisujące czas stworzenia ***(created)*** oraz modyfikacji ***(changed)***. Ten drugi oznacza datę ostatniej zmiany znanej systemowi eksportującemu dane w czasie tego eksportu.

Umieszczanie tych danych w dokumencie zgodnym z EPIP jest opcjonalne. Należy jednak rozróżnić je od atrybutu ***PublicationDelivery ^ PublicationTimestamp***, który określa datę utworzenia pliku XML.

- Atrybut ***created*** określa datę stworzenia encji w pierwotnym systemie. W przypadku referencji jest to data utworzenia odniesienia.
- Atrybut ***changed*** określa datę ostatniego przypisania nowej wersji, czyli co do zasady faktycznej modyfikacji obiektu.

## 10.4. Przestrzeń kodów i struktura identyfikatorów

### 10.4.1. Wprowadzanie

Dane wymieniane w ramach profilu pochodzą z wielu źródeł i są zarządzane przez wiele podmiotów. Wymiana jest procesem wielokrotnie powtarzanym i może być integrowana z innymi danymi na wiele sposobów. Bardzo istotne jest więc zapewnienie, aby identyfikatory były unikalne i nie zmieniały się podczas cyklu życia każdego z obiektów.

Norma NeTEx pozwala na arbitralną strukturę identyfikatorów definiowaną przez obiekty CODESPACE, które zapewniają dodatkowo kontekst, w którym dany zestaw identyfikatorów jest unikalny (zobacz Przestrzeń kodów — Codespace). W jednym dokumencie może być zdefiniowanych wiele CODESPACE. Jeżeli CODESPACE obiektu jest tożsame z CODESPACE zdefiniowanym w ramce zawierającej ten obiekt, może zostać pominięte w definicji obiektu.

CODESPACE może również definiować dodatkowe reguły dotyczące danych, takie jak minimalne i maksymalne długości elementów, dozwolone znaki i określona struktura.

Celem istnienia CODESPACE jest umożliwienie naturalnego używania istniejących systemów identyfikatorów i jednoczesnego używania wielu różnych takich systemów bez ryzyka istnienia konfliktów identyfikatorów.

### 10.4.2. Używanie CODESPACE

Wszystkie identyfikatory używane w dokumencie zgodnym z EPIP muszą mieć przydzielone CODESPACE i być unikalne wewnątrz danego CODESPACE.

Współdzielone *epip_data* CODESPACE jest dostępne do definiowania unikalnych identyfikatorów w paneuropejskim kontekście.

W związku z powyższym struktura tych identyfikatorów jest rozbudowana i koduje znaczenie w strukturze, co pozwala na tworzenie jednolitych identyfikatorów i zapewnia proste w użyciu rozwiązanie dla sytuacji, gdzie nie istniał dotychczas system nadawania identyfikatorów.

Użycie identyfikatorów zgodnych z *epip_data* jest obowiązkowe dla ramek danych (FRAME). Dla obiektów zawartych wewnątrz ramek jest opcjonalne, ale zalecane.

### 10.4.3. Struktura identyfikatorów obiektów danych

CODESPACE *epip_data* dla elementów danych zgodnie z konwencją oznaczane jest w dokumentach XML przez prefiks *epd.* Prefiks może zostać pominięty, jeżeli jest zdefiniowany jako domyślny w nadrzędnej ramce.

Struktura identyfikatora dla obiektu wygląda następująco:

(epd:)[country code]:([local code]):[object type](_[epip type]):[technical identifier](:[ID provider]) gdzie:

- [country code] — dwuliterowy kod państwa ISO 3166-1 (np. PL);
- [local code] — opcjonalny kode regionu państwa (np. europejski kod NUTS, kod jednostki administracyjnej), zalecany jest kod NUTS; kod może być pominięty, jeżeli reszta identyfikatora zapewnia jego unikalność;
- [object type] — typ obiektu w postaci dokładnej nazwy elementu XML (np. *ServiceJourney, Line);*
- [epip type] — opcjonalny kwalifikator [object type], pozwalający na używanie tego samego [technical identifier] dla blisko powiązanych obiektów; w szczególności może byc używany do rozróżnienia typów *StopPlace* na dwie klasy: *StopPlace_monomodal* oraz *StopPlace_general;*
- [technical identifier] — unikalny identyfikator dla obiektu wewnątrz CODESPACE i innych parametrów, co do zasady jest to dowolny ciąg znaków spełniających typ danych *NCName* XML;
- [ID provider] — opcjonalny identyfikator dostawcy identyfikatorów (pierwotnej bazy danych), jeżeli istnieje.

Przykłady (fikcyjne, ilustrujące mechanizm):

<**ScheduledStopPoint** id="epd:FR:SMIRT:ScheduledStopPoint:454e74"/>

<**Line** id="epd:SI:LLP:Line:22"/>

<**Route** id="epd:NO::Route:985a:ENTUR"/>

<**ServiceJourney** id="epd:DE:DB:ServiceJourney:985a"/>

<**StopPlace** id="epd:PL::StopPlace_monomodal:5900:PLK"/>

### 10.4.4. Struktura identyfikatorów ramek wersji

Identyfikatory ramek również muszą być unikalne w celu zapewnienia kontroli wersji podczas wymiany danych.

Struktura identyfikatora jest podobna:

(epd:)[country code]:[local code]:[frame type]_[epip type]:[frame topic] gdzie:

- [country code] i [local code] — jak wyżej;
- [frame type] — jeden z dostępnych typów ramek: ***CompositeFrame, GeneralFrame, ResourceFrame, SiteFrame, ServiceFrame, TimetableFrame*** lub ***GeneralFrame;***
- [epip type] — identyfikator instancji EPIP ***TypeOfFrame:***
  - konkretne:
    - *EU_PI_COMMON*
    - *EU_PI_STOP*
    - *EU_PI_NETWORK*
    - *EU_PI_ TIMETABLE*
    - *EU_PI_CALENDAR*
    - *EU_PI_METADA TA*
  - złożone:
    - *EU_PI_LINE_ OFFER*
    - *EU_PI_NETWORK_OFFER*
    - *EU_PI_ STOP_ OFFER*
- [frame topic] — oznaczenie zawartości ramki w zależności od typu ramki, powinna to być krótka nazwa lub kod (poniżej 15 znaków) złożona z liter, cyfr i dywizu (-):
  - dla *EU_PI_LINE_OFFER* — numer lub nazwa linii,
  - dla *EU_PI_NETWORK_OFFER* — krótka nazwa sieci,
  - dla *EU_PI_STOP_OFFER* — nawa lub identyfikator instancji OPERATOR, ADMINISTRATIVE ZONE lub TOPOGRAPHIC PLACE, dla której zawarto informacje o zatrzymaniach.

W dokumentach XML elementy składające się na [frame topic] zewnętrznych ***CompositeFrame*** powinny być również zawarte jako ***topics*** w ***PublicationDelivery.*** Pozwala to na szybką interpretację zawartości dokumentu przez człowieka lub komputer.

### 10.4.5. Wersje profilu

Zarówno EPIP, jak i profil krajowy są w założeniu przystosowane do rozbudowy w przyszłości. Stąd też każda kolejna wersja profilu ma nadany numer wersji począwszy od 1.0. Numer wersji zapisywany jest w atrybucie ***version*** w instancjach ***TypeOfFrameRef****.*

Numery wersji profilu zapisywane są w formacie a.b, gdzie a to główny numer wersji, a b to numer podwersji.

```
<CompositeFrame version="1.4" id="epd:FR:RAPT:CompositeFrame:EU_PI_LINE_OFFER:L22">
   <TypeOfFrameRef ref="epip:EU_PI_LINE_OFFER" versionRef="1.0"/>
</CompositeFrame>
```

### 10.4.6. Zależności pomiędzy ramkami

Dane różnego rodzaju są podzielone pomiędzy ramki odpowiednich rodzajów, mogą pochodzić też od różnych dostawców. Na przykład dane o przystankach mogą pochodzić z ***SiteFrame*** w *EU_PI_STOP_OFFER,* a dane o liniach komunikacyjnych z ***ServiceFrame*** i ***TimetableFrame*** z *EU_PI_NETWORK_OFFER.*

Zależności pomiędzy ramkami mogą być sformalizowane przez relację ***prerequisites*** w ***VersionFrame****.* Te zależności są szczególnie przydatne w przypadku, kiedy zależna ramka jest umieszczona w oddzielnym pliku XML, co uniemożliwia klasyczne sprawdzanie poprawnoś ci odniesień wewnątrz pliku.

Ramki mogą być umieszczone wewnątrz dokumentu w dowolnej kolejności.

## 10.5. Organizacja i zagnieżdżanie elementów w dokumencie

Kolejność i zagnieżdżanie danych w dokumentach XML jest głównie sterowane przez schemat danych (XSD). Jednakże w celu uelastycznienia normy NeTEx, jej schemat pozwala dla niektórych elementów na pewien stopień wyboru, czy dane mają być zagnieżdżone, czy podane przez odniesienie. Na przykład ***Notice*** może być zdefiniowane wewnątrz obiektu nadrzędnego lub bezpośrednio w ramce i potem użyte w poszczególnych obiektach poprzez ***NoticeAssignment***.

W celu zredukowania możliwych alternatyw dla implantacji profilu zazwyczaj zdefiniowana jest jedna zalecana opcja:

- Każdy typ elementu jest przyporządkowany do tylko jednego rodzaju konkretnej ramki. Ogólne modele pomocnicze, takie jak ***AlternativeName*** mogą się jednak pojawiać w wielu ramkach w kontekście ich elementów nadrzędnych.
- Pomocnicze obiekty muszą być wbudowane w miejscu ich pierwszego wystąpienia (ułatwia to ręczną inspekcję danych), na przykład:
  - instancje ***TrainNumber*** wbudowane są w ***ServiceJourney****;*
  - instancje ***ServiceFacilitySet*** wbudowane są w ***ServiceJourney****;*
  - instancje ***SiteFacilitySet*** wbudowane są w ***StopPlace****;*

- instancje ***NoticeAssignment*** wbudowane są w ***ServiceJourney*** (albo w inny element nadrzędny);
- instancje ***Notice*** wbudowane są w ***NoticeAssignment***.

• Jeżeli istnieje wiele odniesień do jednego obiektu (np. do często wykorzystywanej uwagi ***Notice***), kolejne odniesienia mogą używać referencji do pierwszego wystąpienia, zamiast wbudowywać powtarzalne kopie elementu.

## 10.6. Odpowiedzialność za dane

Dokumenty wskazują podmiot odpowiedzialny za dostarczenie danych za pomocą atrybutu ***responsibilitySetRef*** w ***DataManagedObject***. Ramka danych może zawierać dane od wielu podmiotów. Instancja OPERATOR lub AUTHORITY zapewniająca dane powinna być zdefiniowana w ramce *EU_PI_COMMON,* która może zawierać jedną lub więcej definicji ***ResponsibilitySet***, które wskazują na konkretne role.

Przykład XML:

```
<CompositeFrame version="3.0" id="UK::CompositeFrame-EU_PI_STOP_OFFER:490:NaPTAN"
        responsibilitySetRef="UK:TfL:ResponsibilitySet:082:NPTG">
  <TypeOfFrameRef ref="epip:EU_PI_STOP_OFFER" versionRef="1.0"/>
  <FrameDefaults>
    <DefaultCodespaceRef ref="epip_data"/>
    <DefaultResponsibilitySetRef version="any" ref="UK:TfL:ResponsibilitySet:082:NaPTAN"/>
  </FrameDefaults>
  <frames>
    <SiteFrame version="2.1" id="UK:TfL:SiteFrame-EU_PI_STOP:490:NaPTAN">
      <TypeOfFrameRef ref="epip:EU_PI_STOP" versionRef="1.0"/>
      [...]
    </SiteFrame>
    <ResourceFrame version="2.1" id="UK:TfL:ResourceFrame-EU_PI_COMMON:490:NaPTAN">
      <TypeOfFrameRef ref="epip:EU_PI_COMMON" versionRef="1.0"/>
      [...]
      <responsibilitySets>
        <ResponsibilitySet version="any" id="UK:TfL:ResponsibilitySet:082:NaPTAN"
                responsibilitySetRef="nptg:RS_nptg">
          <Name>TfL provides London data</Name>
          <roles>
            <ResponsibilityRoleAssignment version="any" id="UK:TfL:ResponsibilitySet:082:Na
PTAN">
              <Description>490 London data - Managed centrally</Description>
              <DataRoleType>collects</DataRoleType>
              <ResponsibleOrganisationRef ref="noc:TfL" versionRef="any"/>
            <ResoonsibleAreaRef ref="nptg:AdministrativeZone:82" versionRef="any"/>
            </ResponsibilityRoleAssignment>
          </roles>
        </ResponsibilitySet>
      </responsibilitySets>
```

```
      […]
    </ResourceFrame>
  </frames>
</CompositeFrame>
```

## 10.7. Przetwarzanie dużych zbiorów danych

Niektóre zbiory danych mogą być bardzo duże, np. w Polsce znajduje się około 175 tysięcy przystanków autobusowych[1], a rozkład pojedynczej linii może zawierać kilkadziesiąt zatrzymań, wiele różnych wersji i wiele rozkładów dla różnych rodzajów dni.

[1]: GUS, 2021, Czynne przystanki autobusowe (z trolejbusowymi).

W celu zapewnienia modularności danych oraz ograniczenia wielkości poszczególnych plików norma przewiduje poniższe zalecenia:

- Duże zbiory danych powinny być dzielone na mniejsze pliki, np. podział według linii (rozkłady jazdy) lub obszaru (dane o przystankach bez rozkładów).
- W dokumentach zgodnych z profilem dane powinny być podzielone na odpowiednie ramki. Na przykład w *EU_PI_NETWORK_OFFER* złożona ramka przechowuje dane o wielu liniach; wewnątrz niej powinny być osobne ***ServiceFrame*** dotyczące opisu SERVICE JOURNEY poszczególnych linii oraz ***TimetableFrame*** zawierające rozkłady.
- W przypadku rozdzielenia danych na wiele plików należy korzystać z atrybutu ***VersionFrame ^ prerequisites*** w celu wskazanie zależności.

## 10.8. Dostępne ty py ramek danych

### 10.8.1. Lista typów ramek danych

Poniższe TYPE OF FRAME są dostępne do użycia w profilu:

- pomocnicze:
  - *EU_PI_COMMON,*
  - *EU_PI_CALENDAR;*
- główne:
  - *EU_PI_STOP,*
  - *EU_PI_NETWORK,*
  - *EU_PI_ TIMETABLE;*
- złożone:
  - *EU_PI_LINE_ OFFER,*
  - *EU_PI_NETWORK_OFFER,*
  - *EU_PI_STOP_OFFER;*
- metadanych:
  - *EU_PI_METADA TA*.

W kolejnych sekcjach zaprezentowano tabelaryczne przedstawienie danych zawartych w poszczególnych TYPE OF FRAME.

Wszystkie opisy korzystają ze wspólnej konwencji zapisu:

- Jeżeli przed nazwą klasy nie umieszczono strzałki, elementy należy zawrzeć bezpośrednio w ramce.
- Elementy klasy będącej podkomponentem klasy nadrządnej, które należy wbudować wewnątrz elementu nadrzędnego są poprzedzone podwójną strzałką (^).
- Elementy, które należy łączyć z elementem nadrzędnym poprzez referencję (odniesienie), są poprzedzone pojedynczą strzałką (^).
- Elementy, których klasa nadrzędna jest abstrakcyjna i element nadrzędny może być wielu typów, poprzedzone są gwiazdką (*).
- Klasy w nawiasach okrągłych są abstrakcyjne.

### 10.8.2. Ramki pomocnicze

#### 10.8.2.1. TypeOfFrame: ALL

| Nazwa Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|
| ***classes*** *ClassInContextRef* | * ^ ***ValidityCondition*** | VALIDITY CONDITION |
| | * ^ ***ValidBetween*** | [tylko NeTEx] |
| | * ^ ***AlternativeText*** | ^ ALTERNATIVE TEXT |
| | * ^ ***KeyValue*** | [tylko NeTEx] |

Powyższe dane mogą być powiązane z dowolnym ***DataManagedObject*** w dowolnym typie ramki.

#### 10.8.2.2. TypeOfFrame: EU_PI_COMMON

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_COMMON* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***ResourceFrame*** | RESOURCE FRAME |
| ***classes*** | *ClassInContextRef* | ***ResponsibilitySet*** | RESPONSIBILITY SET |
| | | ***ResponsibilityRoleAssignment*** | RESPONSIBILITY ROLE ASSIGNMENT |
| | | ***Operator*** | OPERATOR |
| | | ***Authority*** | AUTHORITY TYPE |
| | | ***TypeOfFrame*** | OF FRAME [tylko |
| | | ***Branding*** | NeTEx] DATA |
| | | ***DataSource*** | SOURCE [tylko |
| | | ***ValueSet*** | NeTEx] |

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| | | ^ (TypeOfValue) | TYPE OF VALUE |

Instancja ***ResourceFrame*** zgodna z *EU_PI_COMMON* powinna zawierać pomocnicze dane, do których odnoszą się główne ramki *EU_PI_STOP* oraz *EU_PI_TIMETABLE.*

10.8.2.3. TypeOfFrame: EU_PI_CALENDAR

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_COMMON* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***ServiceCalendarFrame*** | SERVICE CALENDAR FRAME |
| ***classes*** | *ClassInContextRef* | ***ServiceCalendar*** | SERVICE SET |
| | | ^ ***OperatingDay*** | ^ OPERATING DAY |
| | | ^ ***OperatingPeriod*** | ^ OPERATING PERIOD |
| | | ^ ***UicOperatingDay*** | [tylko NeTEx] |
| | | ^ ***DayTypeAssignment*** | ^ DAY TYPE ASSIGNMENT |
| | | ^ ***DayType*** | DAY TYPE |

Instancja ***ServiceCalendarFrame*** zgodna z *EU_PI_CALENDAR* powinna zawierać pomocnicze dane, do których odnoszą dane w ramkach *EU_PI_TIMETABLE.*

10.8.3. Ramki główne

10.8.3.1. TypeOfFrame: EU_PI_STOP

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu TYPE |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_STOP* | OF FRAME SITE |
| ***FrameClassRef*** | *NameOfClass* | ***SiteFrame*** | FRAME STOP |
| ***classes*** | *ClassInContextRef* | ***StopPlace*** | PLACE |
| | | ^ ***StopPlaceEntrance*** | ^ STOP PLACE ENTRANCE |
| | | ***Quay*** | ^ QUAY |
| | | ***TopographicPlace*** | TOPOGRAPHIC PLACE |
| | | ***TopographicPlaceDescriptor*** | ^ TOPOGRAPHIC PLACE DESCRIPTOR |
| | | ***PointOfInterest*** | POINT OF INTEREST |

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| | | * ^ ***RoadAddress*** | ROAD ADDRESS |
| | | * ^ ***PostalAddress*** | POSTAL ADDRESS |
| | | * ^ ***SiteFacilitySet*** | SITE FACILITY SET |
| | | * ^ ^ ***Facility*** | ^ FACILITY |
| | | * ^ ***AccessibilityAssessment*** | ACCESSIBILITY ASSESSMENT |
| | | * ^ ^ ***AccessibilityLimitation*** | ^ ACCESSIBILITY LIMITATION |
| | | ***GroupOfStopPlaces*** | (GROUP OF ENTITIES) |

Instancja ***SiteFrame*** zgodna z *EU_PI_STOP* powinna zawierać dane wymagane przez ***CompositeFrame*** *EU_PI_LINE_OFFER* lub *EU_PI_NETWORK_OFFER**.*** Zatem odpowiednio: obiekty STOP PLACE dotyczące pojedynczej linii (LINE) lub całej sieci (NETWORK). Sieć może być zdefiniowana jako ograniczona regionalnie (np. ZTM Poznań) lub według środka transportu (np. połączenia kolejowe).

10.8.3.2. TypeOfTrame: EU_PI_NETWORK

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_NETWORK* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***ServiceFrame*** | SERVICE FRAME |
| ***classes*** | *ClassInContextRef* | ***Line*** | LINE |
| | | ^ ***AllowedLineDirection*** | ^ ALLOWED LINE DIRECTION |
| | | ***FlexibleLine*** | FLEXIBLE LINE |
| | | ***Direction*** | DIRECTION |
| | | ***GroupOfLines*** | GROUP OF LINES |
| | | ***Network*** | NETWORK |
| | | ***TariffZone*** | TARIFF ZONE |
| | | ***DestinationDisplay*** | DESTINATION DISPLAY |
| | | ^ ***Via*** | ^ VIA |
| | | ***DestinationDisplayVariant*** | ^ DESTINATION DISPLAY VARIANT |
| | | ***RoutePoint*** | ROUTE POINT |

| Nazwa Typ XML Zawartość typu XML | Encje Transmodelu |
|---|---|
| ***RouteLink*** | ROUTE LINK |
| ***Route***<br>^ ***PointOnRoute*** | ROUTE |
| ***ScheduledStopPoint*** | ^ POINT ON ROUTE<br>SCHEDULED STOP POINT |
| ***ServiceLink*** | SERVICE LINK |
| * ^ ***TransferDuration***<br>***Connection*** | (TRANSFER) CONNECTION |
| * ^ ***ConnectionEnd***<br>***SiteConnection*** | [tylko NeTEx]<br>SITE CONNECTION |
| * ^ ***SiteConnectionEnd***<br>***DefaultConnection*** | [tylko NeTEx]<br>DEFAULT CONNECTION |
| * ^ ***DefaultConnectionEnd***<br>***ServiceJourneyPattern*** | [tylko NeTEx]<br>SERVICE JOURNEY PATTERN |
| ***StopPointInJourneyPattern*** | ^ STOP POINT IN JOURNEY PATTERN |
| ***ServicePattern*** | SERVICE PATTERN |
| ***StopPointInJourneyPattern*** | ^ STOP POINT IN JOURNEY PATTERN |
| * ^ ***FlexiblePointProperties*** | FLEXIBLE POINT PROPERTIES |
| ***PassengerStopAssignment*** | PASSENGER STOP ASSIGNMENT |
| ^ ***TrainStopAssignment*** | ^ TRAIN STOP ASSIGNMENT |
| ***DynamicStopAssignment*** | DYNAMIC STOP ASSIGNMENT |
| * ^ ***Notice***<br>* ^ ***NoticeAssignment*** | NOTICE<br>NOTICE ASSIGNMENT |

Instancja ***ServiceFrame*** zgodna z *EU_PI_NETWORK* powinna zawierać dane rozkładowe dla dokładnie jednej linii.

10.8.3.3. TypeOfTrame: EU_PI_TIMETABLE

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_ TIMETABLE* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***TimetableFrame*** | TIMETABLE FRAME |
| ***classes*** | *ClassInContextRef* | ***GroupOfServices*** | GROUP OF SERVICES |
| | | ^ ***GroupOfServicesMember*** | [tylko NeTEx] |
| | | ***ServiceJourney*** | SERVICE JOURNEY |
| | | ^ ***JourneyPart*** | ^ JOURNEY PART |
| | | ^ ***TimetabledPassingTime*** | ^ TIMETABLED PASSING TIME |
| | | ——— | FLEXIBLE |
| | | ***FlexibleServiceProperties*** | SERVICE PROPERTIES |
| | | ***TemplateServiceJourney*** | TEMPLATE SERVICE JOURNEY |
| | | * ^ ***HeadwayJourneyGroup*** | ^ HEADWAY JOURNEY GROUP |
| | | * — ***ServiceFacilitySet*** | SERVICE FACILITY SET |
| | | ***JourneyPartCouple*** | JOURNEY PART COUPLE |
| | | ^ ***JourneyPartPosition*** | [tylko NeTEx] |
| | | ***ServiceJourneyInterchange*** | SERVICE JOURNEY INTERCHANGE |
| | | ***VehicleType*** | VEHICLE TYPE |
| | | * — ***TrainNumber*** | TRAIN NUMBER |
| | | ***Train*** | TRAIN |
| | | ^ ***TrainComponent*** | ^ TRAIN COMPONENT |
| | | * ^ ***TrainElement*** | TRAIN ELEMENT |
| | | * ^ ***Notice*** | NOTICE |

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| | | * ^ ***NoticeAssignment*** | NOTICE ASSIGNMENT |

Instancja ***TimetableFrame*** zgodna z *EU_PI_TIMETABLE* powinna zawierać dane rozkładowe dla dokładnie jednej linii.

10.8.4. Ramki złożone 10.8.4.1. TypeOfFrame: EU_PI_LINE_OFFER

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_LINE_OFFER* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***CompositeFrame*** | COMPOSITE FRAME |
| ***includes*** | *TypeOfFrameRef* | *epip:EU_PI_COMMON* | RESOURCE FRAME |
| | | *epip:EU_PI_ S TOP* | SITE FRAME |
| | | *epip:EU_PI_NETWORK* | SERVICE FRAME |
| | | *epip:EU_PI_ TIMETABLE* | TIMETABLE FRAME |
| | | *epip:EU_PI_ CALENDA R* | SERVICE CALENDAR FRAME |

Instancja ***CompositeFrame*** zgodna z *EU_PI_LINE_OFFER* powinna zawierać dane dotyczące dokładnie jednej linii.

10.8.4.2. TypeOfFrame: EU_PI_NETWORK_OFFER

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_NETWORK_ OFFER* | TYPE FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***CompositeFrame*** | COMPOSITE FRAME |
| ***includes*** | *TypeOfFrameRef* | *epip:EU_PI_COMMON* | RESOURCE FRAME |
| | | *epip:EU_PI_ S TOP* | SITE FRAME |
| | | *epip:EU_PI_NETWORK* | SERVICE FRAME |
| | | *epip:EU_PI_ TIMETABLE* | TIMETABLE FRAME |
| | | *epip:EU_PI_ CALENDA R* | SERVICE CALENDAR FRAME |

Instancja ***CompositeFrame*** zgodna z *EU_PI_NETWORK_OFFER* powinna zawierać dane dotyczące co najmniej dwóch linii, które tworzą sieć transportową (NETWORK).

10.8.4.3. TypeOfFrame: EU_PI_STOP Nazwa Typ XML Zawartość typu XML Encje Transmodelu

| | | | |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_ S TOP_ OFFER* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***CompositeFrame*** | COMPOSITE FRAME |
| ***includes*** | *TypeOfFrameRef* | *epip:EU_PI_COMMON* | RESOURCE FRAME |
| | | *epip:EU_PI_ S TOP* | SITE FRAME |

Instancja ***CompositeFrame*** zgodna z *EU_PI_STOP_OFFER* powinna zawierać co najmniej jedną instancję ramki ***SiteFrame*** *EU_PI_STOP.* Każda z nich przekazuje informacje o STOP PLACE dla jednego lub więcej środka transportu (MODE), operato ra (OPERATOR) lub obszaru (ADMINISTRATIVE AREA). Duże zbiory danych mogą być podzielone na mniejsze instancje, które zawierają dane z podziałem na przykład na operatora, obszar.

10.8.5. Ramki metadanych

10.8.5.1. TypeOfFrame: EU_PI_METADATA

| Nazwa | Typ XML | Zawartość typu XML | Encje Transmodelu |
|---|---|---|---|
| ***id*** | *TypeOfFrameIdType* | *epip:EU_PI_ ME TA DA TA* | TYPE OF FRAME |
| ***FrameClassRef*** | *NameOfClass* | ***ResourceFrame*** | RESOURCE FRAME |
| ***includes*** | *TypeOfFrameRef* | ***ResponsibilitySet*** | RESPONSIBILITY SET |
| | | ***ResponsibilityRoleAssignment*** | RESPONSIBILITY ROLE ASSIGNMENT |
| | | ***TypeOfFrame ValueSet*** | TYPE OF FRAME [tylko NeTEx] |
| | | ^ ***TypeOfPlace*** | TYPE OF PLACE |

Instancja ***ResourceFrame*** zgodna z *EU_PI_METADATA* powinna zawierać definicje stałych elementów używanych w profilu: instancji ***TypeOfFrame*** oraz ***TypeOfPlace.***

## 10.9. Kompletność danych

### 10.9.1. Wprowadzenie

NeTEx przewiduje wymianę pełnych zbiorów danych lub tylko różnic (delt). Rodzaj przekazywanych danych jest wskazywany przez atrybut ***modification*** w ***DataManagedObject.*** W przypadku danych różnicowych przekazywane są tylko te elementy, które się zmieniły.

W celu uproszenia profil krajowy i EPIP zakładają wymianę wyłącznie pełnych zbiorów danych. Nowa wersja zastępuje starą całkowicie. Stąd też dozwolone wartości atrybutu *modification* to *new*, *revise*, *unchanged* i *delete*. Wartość *delta* jest niedozwolona. Atrybut *modification* jest wymagany tylko dla najbardziej zewnętrznego kontenera danych, zazwyczaj będzie to *CompositeFrame*.

Wewnątrz ramki wszystkie wartości wymagane przez profil powinny być uzupełnione. Wytwórca eksportowanych danych musi zatem obsłużyć cztery przypadki:

1. Elementy wymagane przez schemat NeTEx (muszą zostać uzupełnione).
2. Elementy opcjonalne w schemacie NeTEx, ale obowiązkowe w profilu (muszą zostać uzupełnione).
3. Elementy opcjonalne w profilu powinny być uzupełnione, jeżeli wytwórca posiada te dane, ale mogą zostać pominięte, jeżeli ich nie posiada.
4. Elementy leżące poza zakresem profilu powinny być uzupełnione analogicznie jak w punkcie 3.

### 10.9.2. Puste wartości

Opcjonalne elementy, dla których wytwórca nie zna wartości, powinny być pomijalne całkowicie w XML, a nie stosowane puste tagi (np. <Name></Name> lub <Name/>).

## 10.10. Wymiana danych

Dane wymieniane są w postaci plików XML, których korzeniem jest element *PublicationDelivery*, który zawiera dokładnie jedną ramkę danych jednego z dozwolonych typów.

W ramach profilu dozwolone są trzy rodzaje plików:

1. **Line XML** — zawierające pojedynczą ramkę *EU_PI_LINE_OFFER CompositeFrame*, która opisuje przystanki, topologię i rozkład pojedynczej linii.
2. **Network XML** — zawierające pojedynczą ramkę *EU_PI_NETWORK_OFFER CompositeFrame* opisującą wiele linii. Wewnątrz tej ramki może znajdować się wiele potomnych ramek, np. *TimetableFrame* dla każdej linii w sieci.
3. **Stop XML** — zwierające pojedynczą ramkę *EU_PI_STOP_OFFER CompositeFrame*, która opisuje wszystkie przystanki danego państwa, regionu, organizacji lub przewoźnika. Wewnątrz tej ramki może znajdować się jedna lub więcej ramek *SiteFrame*.

Należy zauważyć, że te same dane mogą być wymagane w więcej niż jednym zbiorze danych. Może to powodować pewną redundancję danych. Jest to poprawne i zamierzone, system przetwarzający dane może użyć identyfikatorów i wersji opisanych powyżej, aby ustalić najbardziej aktualny opis poszczególnych elementów.

## 10.11. Obiekty PublicationDelivery

### 10.11.1. PublicationDelivery

*PublicationDelivery* stanowi korzeń dokumentów XML przekazywanych w ramach EPIP.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| | | xsd:NMTOKEN | | Identyfikator wersji schematu NeTEx, która jest używana w dokumencie. |
| | ***PublicationTimestamp*** | *xsd:dateTime* | 1:1 | Czas eksportu danych. Zalecane podanie w UTC, np. 202110-06T13:25:00Z. |
| «FK» | ***ParticipantRef*** | *ParticipantCode Type* | 1:1 | Identyfikator systemu dostarczającego dane. Zazwyczaj będzie tożsamy z DATA SOURCE. |
| | ***Description*** | *xsd:normalizedString* | 0:1 | Opis zawartości. |
| «cntd» | ***dataObjects*** | *data Objects_RelStructu re* | 0:1 | VERSION FRAME zawierające dane publikacji. |

### 10.11.2. PublicationRequest (podkomponent PublicationDelivery)

***PublicationRequest*** opisuje, które obiekty zostały zażądane do włączenia do dokumentu wraz z poziomem szczegółowości.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***topics*** | *NetworkFrameTopic* | 0: | Filtry zapytania opisujące zakres żądanych danych. |

### 10.11.3. NetworkFrameTopic (podkomponent PublicationDelivery)

***NetworkFrameTopic*** opisuje kryteria wyboru danych do załączenia w dokumencie NeTEx. W EPIP używane są wyłącznie ***NetworkFilterByValue,*** które służą do wskazywania interesujących linii, sieci lub obszarów.

| Klasyfikacj a Nazwa | Typ | Kardynalnoś ć | Opis |
|---|---|---|---|
| «cntd» ***NetworkFilterByValu e*** | *NetworkFilterByValueStructu re* | 0:1 | Wartości , które należy załączyć |

### 10.11.4. NetworkFilterByValue (podkomponent PublicationDelivery)

***NetworkFilterByValue*** specyfikuje kryteria wyboru obiektów do załączenia w dokumencie NeTEx.

| Klasyfikacja | Nazwa | Typ | Kardynalność | Opis |
|---|---|---|---|---|
| «cntd» | ***objectReferences*** | *(VersionOfObjectRef)* | 0:* | Odniesienia do encji, które należy zwrócić. |
| «cntd» | ***NetworkRef*** | *NetworkRefStructure* | 0:1 | Odniesienie do NETWORK. |
| «cntd» | ***places*** | *(PlaceRef)* | 0:* | Odniesienie do PLACE, z którego mają zostać wszystkie elementy. |

## 10.12. Nazwy plików

W celu usprawnienia przetwarzania danych zaproponowano konwencję nazywania plików XML zawierających dane NeTEx. Dla plików zgodnych z EPIP wygląda ona następująco:

[prefix]-[epip version]_[country code]_[provider code]_[profile type]_[doc topic]_[creation date]

gdzie:

- [prefix] — stały prefiks ustalony pomiędzy partnerami, tutaj: ***"NX-PI";***
- [epip version] — wersja profilu EPIP, obecnie: "01";
- [country code] — kod ISO państwa;
- [provider code] — kod dostarczyciela danych, należy użyć "NAP" w przypadku krajowego punktu dostępu (KPD);
- [profile type] — typ profilu: *LINE, NETWORK* albo *STOP;*

- [doc topic] — zawartość dokumentu, nazwa zgodna z elementem [frame topic] opisanym w sekcji Struktura identyfikatorów ramek wersji;
- [creation date] — data stworzenia dokumentów w formacie YYYYMMDD.

Przykłady:

- NX-PI-01_PL_PLK_STOP_RAILWAY_20211210.xml
- NX-PI-01_PL_ZTMWAW_NETWORK_ZTMWAW_20220215.xml
- NX-PI-01_PL_ZTMWAW_LINE_169_20220215.xml

## 10.13. Rozszerzanie o dane dodatkowe

Zarówno EPIP, jak i minimalny profil krajowy mają za zadanie zapewnić minimalny funkcjonalny podzbiór normy NeTEx. W przypadkach, kiedy użytkownicy chcą rozszerzyć przekazywane dane, mogą to zrobić na kilka sposobów:

- dodanie atrybutów istniejących w NeTEx, ale nie wybranych do profilu krajowego: takie atrybuty są dozwolone, ale dane muszą pozostać spójne w przypadku przetwarzania przez narzędzia ignorujące atrybuty spoza profilu;
- dodanie obiektów istniejących w NeTEx, ale nie objętych profilem krajowym: istnienie tych obiektów nie może być kluczowe do semantyki przekazywanych danych; instancje tych obiektów powinny być przekazane w osobnej ramce;
- dane nieobjęte ani profilem krajowym, ani schematem NeTEx powinny być dodawane poprzez mechanizm ***KeyValue;*** jeżeli konkretne dane mogą być wyrażone w postaci elementów profilu, użycie ***KeyValue*** nie może zastępować wprowadzenia danych bezpośrednio do odpowiednich encji NeTEx.

Mając na uwadze powyższe, można wyróżnić trzy modele używania danych profilowych:

1. **Strict conformance** — używane są wyłącznie obiekty i atrybuty wyspecyfikowanie w profilu;
2. **Augmented conformance** — używane są wyłącznie obiekty zdefiniowane w profilu, dodatkowe dane są przekazywane w postaci:
   - dodatkowych atrybutów (zgodnych z NeTEx)
   - dodatkowych wartości list enumeracyjnych
   - instancje ***KeyValue***
3. **Extended conformance** — używane są dodatkowe obiekty z NeTEx oraz wszystkie powyższe rozszerzenia.

Wytwórca danych może oznaczyć oczekiwaną zgodność danych w ramce za pomocą instancji ***KeyValue***:

```
<SiteFrame version="2.3" id="FR:RATP:SiteFrame-EU_PI_STOP:RATP-1">
  <ValidBetween>
    <FromDate>2020-12-17T09:30:47Z</FromDate>
    <FromDate>2021-12-17T09:30:47Z</FromDate>
  </ValidBetween>
```

```
  <keyList>
    <KeyValue>
      <Key>epip:conformance</Key>
      <Value>strict</Value>
    </KeyValue>
  </keyList>
  <TypeOfFrameRef ref="epip:EU_PI_STOP"
versionRef="1.0"/> </SiteFrame>
```

# 11. DODATEK A

## 11.1. A.1 FacilityList (udogodnienia)

- *unknown* (nieznane)
- *lift* (dźwig osobowy)
- *wheelchairLift* (dźwig dla wózków inwalidzkich)
- *escalator* (schody ruchome)
- *travelator* (chodnik ruchomy)
- *ramp* (rampa)
- *automaticRamp* (rampa automatyczna)
- *steps* (stopnie)
- *stairs* (schody)
- *slidingStep* (stopień wysuwany)
- *shuttle* (transfer)
- *narrowEntrance* (wąskie wejście)
- *barrier* (barierka)
- *lowFloorAccess* (dostęp niskopodłogowy)
- *validator* (kasownik)

## 11.2. A.2 AccessibilityInfoFacilityList (dostępność informacji)

- *audioInformation* (informacja głosowa)
- *audioForHearingImpaired* (informacja głosowa dla osób słabosłyszących)
- *visualDisplays* (wyświetlacze)
- *displaysForVisuallyImpaired* (wyświetlacze dla osób słabowidzących)
- *largePrintTimetables* (duże drukowane rozkłady jazdy)
- *other* (inne)

## 11.3. A.3 AccessibilityToolList (środek dostępności)

- *wheelchair* (wózek inwalidzki)
- *walkingstick* (laska)
- *audioNavigator* (nawigacja dźwiękowa)
- *visualNavigator* (nawigacja wizualna)
- *passengerCart* (pojazd do transportu pasażerów)
- *pushchair* (wózek spacerowy)
- *umbrella* (parasol)
- *buggy* (łazik)
- *other* (inne)

## 11.4. A.4 AccommodationAccessList (rodzaj ulokowania podróżnych)

- *other* (inne)

- *freeSeating* (dowolne zajmowanie miejsc)
- *reservation* (rezerwacja miejsc)
- *standing* (miejsca stojące)

## 11.5. A.5 AccommodationFacilityList (własności ulokowania podróżnych)

- *unknown* (nieznane)
- *seating* (miejsca siedzące)
- *sleeper* (miejsca sypialne)
- *singleSleeper* (jednoosobowy przedział sypialny)
- *doubleSleeper* (dwuosobowy przedział sypialny)
- *specialSleeper* (specjalny przedział sypialny)
- *couchette* (miejsce do leżenia)
- *singleCouchette* (jednoosobowy przedział do leżenia)
- *doubleCouchette* (dwuosobowy przedział do leżenia)
- *specialSeating* (specjalne miejsce siedzące)
- *recliningSeats* (fotele rozkładane — wypoczynkowe)
- *babyCompartment* (przedział dla dzieci)
- *familyCarriage* (wagon rodzinny)
- *recreationArea* (przestrzeń rekreacyjna)
- *panoramaCoach* (wagon panoramiczny)
- *pullmanCoach* (wagon pulmanowski)
- *standing* (miejsca stojące)

## 11.6. A.6 AssistanceFacilityList (usługi pomocowe)

- *personalAssistance* (pomoc osobista)
- *boardingAssistance* (pomoc przy wsiadaniu)
- *wheelchairAssistance* (pomoc dla osób na wózkach inwalidzkich)
- *unaccompaniedMinorAssistance* (pomoc dla małoletnich bez opieki)
- *wheelchairUse* (użycie wózka inwalidzkiego)
- *conductor* (drużyna konduktorska)
- *information* (informacja)
- *other* (inne)
- *none* (brak)
- *any* (dowolne)

## 11.7. A.7 BoardingPermission (zezwolenie na wsiadanie)

- *normal* (normalne)
- *earlyBoardingPossibleBeforeDeparture* (dozwolone wczesne zajmowanie miejsc przed odjazdem)
- *lateAlightingPossibleAfterArrival* (dozwolone późne opuszczenie pokładu po przyjeździe)

- *overnightStayOnboardAllowed* (dozwolony pobyt nocny na pokładzie)

## 11.8. A.8 BookingProcessFacilityList (proces zakupu)

- *productNotAvailable* (produkt niedostępny)
- *productNotBookable* (zakup niemożliwy)
- *bookableThroughInternationalSystem* (zakup tylko przez system międzynarodowy)
- *bookableThroughNationalSystem* (zakup tylko przez system krajowy)
- *bookableManually* (zakup manualny)
- *other* (inne)

## 11.9. A.9 CarServiceFacilityList (usługi samochodowe)

- *unknown* (nieznane)
- *carWash* (myjnia)
- *valetPark* (usługa parkowania)
- *carValetClean* (usługa czyszczenia samochodu)
- *oilChange* (wymiana oleju)
- *engineWarming* (ogrzewanie silnika)
- *petrol* (paliwo)

## 11.10. A.10 CateringFacilityList (usługi gastronomiczne)

- *bar* (bar)
- *bistro* (bistro)
- *buffet* (bufet)
- *noFoodAvailable* (posiłki niedostępne)
- *noBeveragesAvailable* (napoje niedostępne)
- *restaurant* (restauracja)
- *firstClassRestaurant* (restauracja pierwszej klasy)
- *trolley* (wózek mini-bar)
- *coffeeShop* (kawiarnia)
- *hotFoodService* (dania gorące)
- *selfService* (samoobsługa)
- *snacks* (przekąski)
- *foodVendingMachine* (automat z przekąskami)
- *beverageVendingMachine* (automat z napojami)
- *miniBar* (mini-bar)
- *breakfastInCar* (śniadanie w wagonie)
- *mealAtSeat* (posiłek dostarczany do zajmowanego miejsca)
- *other* (inne)
- *unknown* (nieznane)

## 11.11. A.11 CouchetteFac ilityList (usługi w wagonach z miejscami do leżenia)

- *unknown* (nieznane)
- *T2*
- *T3*
- *C1*
- *C2*
- *C4*
- *C5*
- *C6*
- *wheelchair* (dostosowane dla osób na wózkach)
- *other* (inne)

## 11.12. A.12 FamilyFacilityList (usługi dla rodzin)

- *none* (brak)
- *servicesForChildren* (usługi dla dzieci)
- *servicesForArmyFamilies* (usługi dla rodzin wojskowych)
- *nurseryService* (opieka dla dzieci)

## 11.13. A.13 FareClasses (klasy taryfowe)

- *unknown* (nieznane)
- *firstClass* (pierwsza klasa)
- *secondClass* (druga klasa)
- *thirdClass* (trzecia klasa)
- *preferente* (używane w Hiszpanii)
- *premiumClass* (klasa premium)
- *businessClass* (klasa biznesowa)
- *standardClass* (klasa standardowa)
- *turista* (używane w Hiszpanii)
- *economyClass* (klasa ekonomiczna)
- *any* (dowolna)

## 11.14. A.14 GenderLimitation (ograniczenia płci)

- *both* (dowolne)
- *femaleOnly* (wyłącznie dla kobiet)
- *maleOnly* (wyłącznie dla mężczyzn)
- *sameSexOnly* (tylko jednej płci)

## 11.15. A.15 GroupBookingFacility (rezerwacje grupowe)

- *groupsAllowed* (grupy dozwolone)
- *groupsNotAllowed* (grupy niedozwolone)

- *gmupsAllowedWithReservation* (grupy tylko z rezerwacją)
- *groupBookingsRestricted* (ograniczona możliwość rezerwacji dla grup)
- *unknown* (nieznane)

## 11.16. A.16 LuggageCarriageFacilityList (przewóz bagażu)

- *unknown* (nieznane)
- *noBaggageStorage* (brak przechowalni bagażu)
- *baggageStorage* (przechowalnia bagażu)
- *luggageRacks* (półki bagażowe)
- *extraLargeLuggageRacks* (półki na duży bagaż)
- *baggageVan* (wagon bagażowy)
- *noCycles* (zakaz przewozu rowerów)
- *cyclesAllowed* (przewóz rowerów dozwolony)
- *cyclesAllowedInVan* (przewóz rowerów dozwolony w specjalnym wagonie rowerowym)
- *cyclesAllowedInCarriage* (przewóz rowerów dozwolony w wagonie)
- *cyclesAllowedWithReservation* (przewóz rowerów dozwolony z rezerwacją)
- *vehicleTransport* (przewóz pojazdów)

## 11.17. A.17 MealFacilityList (posiłki)

- *breakfast* (śniadanie)
- *lunch* (lunch)
- *dinner* (obiad)
- *snack* (przekąska)
- *drinks* (napoje)

## 11.18. A.18 MedicalFacilityList (usługi medyczne)

- *unknown* (nieznane)
- *defibrillator* (defibrylator)
- *alcoholTest* (alkomat)

## 11.19. A.19 MobilityFacilityList (usługi dostępności)

- *unknown* (nieznane)
- *lowFloor* (niska podłoga)
- *stepFreeAccess* (dostęp bez stopni)
- *suitableForWheelchairs* (dostępne dla wózków inwalidzkich)
- *suitableForHeavilyDisabled* (dostępne dla osób z dużą niepełnosprawnością)
- *boardingAssistance* (pomoc przy wsiadaniu)
- *onboardAssistance* (pomoc napokładzie)
- *unaccompaniedMinorAssistance* (pomoc dla małoletnich bez opieki)
- *tactilePlatformEdges* (wyczuwalne krawędzie peronowe dla niewidomych)
- *tactileGuidingStrips* (ścieżki wyczuwalne dla niewidomych)

## 11.20. A.20 Nu isanceFacilityList (niedogodności)

- *unknown* (nieznane)
- *smoking* (dla palących)
- *noSmoking* (dla niepalących)
- *familyArea* (strefa rodzinna)
- *childfreeArea* (strefa bez dzieci)
- *noAnimals* (zakaz zwierząt)
- *breastfeedingFriendly* (przyjazne karmieniu piersią)
- *mobilePhoneUseZone* (strefa z dozwolonym użyciem telefonów komórkowych)
- *mobilePhoneFreeZone* (strefa z zakazem używania telefonów komórkowych)

## 11.21. A.21 PassengerCommsFacilityList (usługi komunikacyjne dla pasażerów)

- *unknown* (nieznane)
- *freeWifi* (bezpłatne Wi-Fi)
- *publicWifi* (publiczne Wi-Fi)
- *powerSupplySockets* (gniazdka zasilające)
- *telephone* (telefon)
- *audioEntertainment* (rozrywka audio)
- *videoEntertainment* (rozrywka wideo)
- *businessServices* (usługi biznesowe)
- *internet* (internet)
- *postOffice* (urząd pocztowy)
- *postBox* (skrzynka pocztowa)

## 11.22. A.22 PassengerInformationEquipmentList (wyposażenie informacji pasażerskiej)

- *timetablePoster* (plat rozkładu jazdy)
- *fareInformation* (informacje taryfowe)
- *lineNetworkPlan* (schemat linii sieci transportowej)
- *lineTimetable* (rozkład linii komunikacyjnej)
- *stopTimetable* (rozkład przystankowy)
- *journeyPlanning* (planowanie podróży)
- *interactiveKiosk* (kiosk interaktywny)
- *informationDesk* (punkt informacyjny)
- *realTimeDepartures* (informacja o odjazdach w czasie rzeczywistym)
- *other* (inne)

## 11.23. A.23 PassengerInformationFacilityList (informacja pasażerska)

- *nextStopIndicator* (wskazanie następnego zatrzymania)

*stopAnnouncements* (zapowiedzi przystankowe) *passengerInformationDisplay* (wyświetlacz informacji pasażerskiej) *realTimeConnections* (informacja o przesiadkach na inne połączenia w czasie rzeczywistym) *other* (inne)

### A.24 RetailFacilityList (usługi handlowe)

*unknown* (nieznane) *food* (artykuły spożywcze) *newspaperTobacco* (prasa i tytoń)

11.24. *recreationTravel* (podróże i rekreacja) *hygieneHealthBeauty* (higiena, zdrowie i uroda) *fashionAccessories* (moda)

*bankFinanceInsurance* (banki, finanse, ubezpieczenia) *cashMachine* (bankomat) *currencyExchange* (wymiana walut) *tourismService* (usługi turystyczne) *photoBooth* (budka fotograficzna)

### A.25 SafetyFacilityList (usługi zapewniania bezpieczeństwa)

*ccTv* (monitoring)

*mobileCoverage* (zasięg komórkowy) *sosPoints* (punkty SOS) *staffed* (obsługa nam miejscu)

### A.26 SanitaryFacilityList (usługi sanitarne)

*none* (brak) *toilet* (toaleta)

*wheelChairAccessToilet* (toaleta dostępna dla osób na wózkach inwalidzkich) *shower* (prysznic)

11.25. *washingAndChangeFacilities* (umywalnia i przebieralnia) *babyChange* (przewijak) *wheelchairBabyChange* (przewijak dostępny dla osób na wózkach inwalidzkich) *shoeShiner* (polerowanie butów) *other* (inne)

### A.27 ServiceReservationFacilityList (usługi rezerwacji miejsc)

*reservationsCompulsory* (rezerwacja obowiązkowa) *reservationsCompulsoryFoGroups* (rezerwacja obowiązkowa dla grup) *reservationsCompulsoryForFirstClass* (rezerwacja

11.26. obowiązkowa w pierwszej klasie)

11.27.

- *reservationsCompulsoryFromOriginStation* (rezerwacja obowiązkowa ze stacji początkowej)
- *reservationsRecommended* (rezerwacja zalecana)
- *reservationsPossible* (rezerwacja możliwa)
- *reservationsPossibleOnlyInFirstClass* (rezerwacja możliwa tylko w pierwszej klasie)
- *reservationsPossibleOnlyInSecondClass* (rezerwacja możliwa tylko w drugiej klasie)
- *reservationsPossibleForCertainClasses* (rezerwacja możliwa tylko dla wybranych klas)
- *groupBookingRestricted* (ograniczona możliwość rezerwacji dla grup)
- *noGroupsAllowed* (grupy niedozwolone)
- *noReservationsPossible* (brak możliwości rezerwacji)
- *wheelchairOnlyReservations* (rezerwacja tylko miejsc na wózki inwalidzkie)
- *bicycleReservationsCompulsory* (rezerwacja miejsc na rowery obowiązkowa)
- *reservationsSupplementCharged* (rezerwacja z dopłatą)
- *unknown* (nieznane)

## 11.28. A.28 TicketingFacilityList (sposoby sprzedaży biletów)

- *unknown* (nieznane)
- *ticketMachines* (biletomaty)
- *ticketOffice* (kasa biletowa)
- *ticketOnDemandMachines*
- *mobileTicketing* (zakup biletów za pomocą telefonu komórkowego)

## 11.29. A.29 TicketingServiceFacilityList (usługi dostępne w ramach sprzedaży biletów)

- *purchase* (zakup)
- *collection* (odbiór)
- *cardTopUp* (doładowanie karty przejazdowej)
- *reservations* (rezerwacje)
- *exchange* (wymiana)
- *refund* (zwrot)
- *renewal* (odnowienie)
- *excessFares* (opłaty dodatkowe)
- *other* (inne)
- *all* (wszystkie)

## 11.30. A.30 UicProductCharacteristicList (klasyfikacja produktów UIC)

- *tariffCommunVoyageurs* (taryfa TCV)
- *allIInclusivePrice* (cena globalna)
- *eastWestTariff* (taryfa Wschód-Zachód — TWZ)
- *trainWithTcvAndMarketPrice* (pociąg z taryfą TCV i ceną globalną)
- *noPublishedTariff* (brak opublikowanej taryfy)

## 11.31. A.31 UicTrainRate (stawka UIC)

*normal* (normalna)
*discountInTrainOtherThanTGV* (zniżka w pociągu innym niż TGV)
*specialFare* (taryfa specjalna)
*supplement* (dopłata)
*noPublishedTariff* (brak opublikowanej taryfy)

## 11.32. A.32 VehicleFacilityList (dostępność pojazdu)

*unknown* (nieznana)
*wheelchairLift* (podnośnik dla wózków inwalidzkich) *manualRamp* (rampa ręczna) *automaticRamp* (rampa automatyczna) *steps* (stopnie)
*slidingStep* (stopień wysuwany) *narrowEntrance* (wąskie wejście)
*validator* (kasownik)