

PRAKTYCZNE ZASTOSOWANIE RAILML¹

Iga Samkowska

Politechnika Warszawska, Wydział Transportu, ul. Koszykowa 75, 00-662 Warszawa, tel.: +48 22 234 7882, email: iga.samkowska@pw.edu.pl

Andrzej Kochan

dr inż., Politechnika Warszawska, Wydział Transportu, ul. Koszykowa 75, 00-662 Warszawa, tel.: +48 22 234 7882, email: andrzej.kochan@pw.edu.pl

***Streszczenie.** Niniejszy artykuł opisuje strukturę i możliwości zastosowania języka RailML do tworzenia cyfrowych modeli sieci kolejowych. Przedstawiono również analizę i porównanie wybranych przypadków użycia, ułatwiających praktyczne wykorzystanie RailML i pozwalających na wybór aspektów sieci uwzględnianych w modelu. Modelowanie w RailML to skomplikowany i złożony proces, dlatego poszukiwane są rozwiązania umożliwiające skrócenie czasu tworzenia modeli. Jednym z nich jest program Rail-AiD – narzędzie do tworzenia i edycji planów schematycznych. W artykule opisano funkcje programu oraz możliwości praktycznego wykorzystania, ze szczególnym uwzględnieniem kompatybilności ze standardem RailML.*

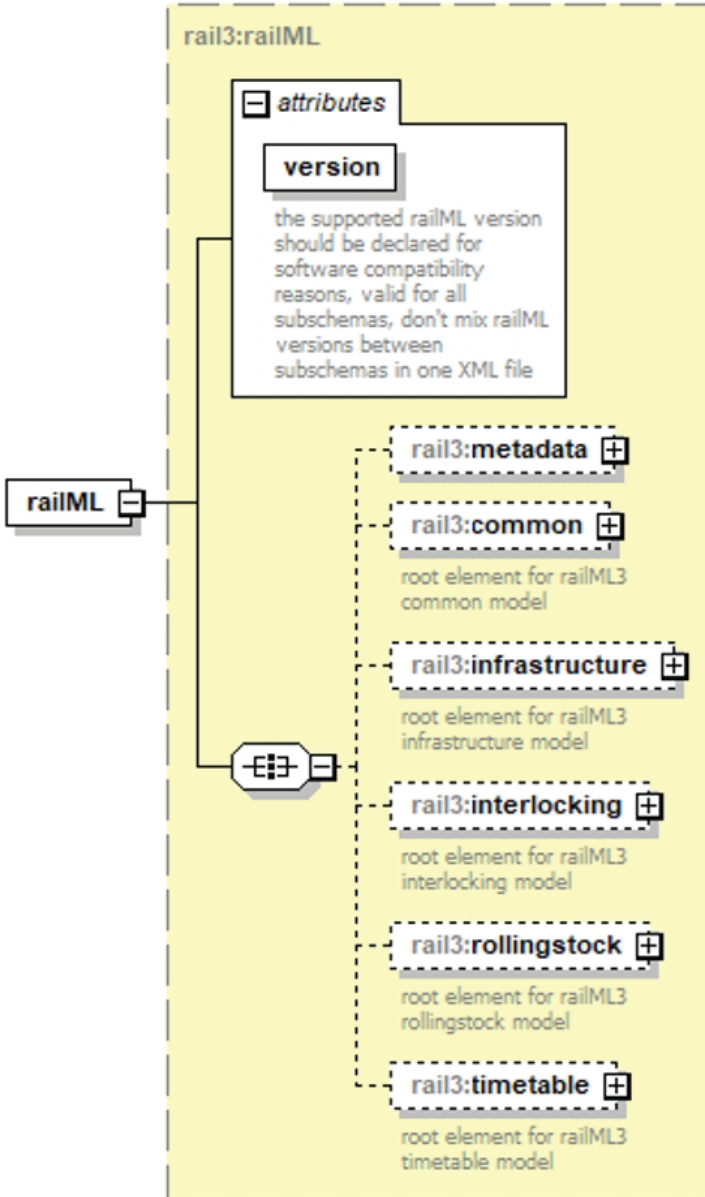
***Słowa kluczowe:** RailML, Rail-AiD, cyfrowy model sieci kolejowej*

1. Wprowadzenie

RailML (ang. Railway Markup Language) jest opartym na formacie XML językiem opisu sieci kolejowych zapewniającym możliwość przechowywania danych oraz ich wymiany między różnymi systemami. Pliki w standardzie RailML składają się ze znaczników oraz przypisanych im atrybutów, co umożliwia nadanie opisywanym elementom dowolnej ilości cech oraz budowanie ich hierarchii. Podział elementów na obowiązkowe i opcjonalne daje użytkownikowi dużą swobodę i umożliwia dostosowanie modelu do własnych potrzeb. W strukturze RailML wyodrębnionych jest pięć głównych schematów [3][4] skupiających się na opisie różnych aspektów sieci kolejowej. Każdy z nich zawiera kolejne podschematy i elementy (rys. 1). Szósty widoczny na rysunku schemat, Metadata, zawiera jedynie informacje na temat pliku RailML, takie jak wersja i data utworzenia. W wersji RailML 3.1 możliwe jest korzystanie z trzech schematów. Pozostałe schematy nie posiadają jeszcze zdefiniowanej zawartości, jednak cały czas trwają prace nad ich rozwojem. W celu otrzymania kompletnego opisu sieci kolejowej często niezbędne jest wykorzystanie kilku schematów i zastosowanie odniesień między nimi. Możliwości wykorzystania schematów RailML w praktyce i dostosowania ilości wykorzystywanych elementów opisane są w przypadkach użycia.

¹ Wkład autorów w publikację: Samkowska I.: 80%, Kochan A.: 20%

Oprócz ręcznego wprowadzania kodu istnieją również inne metody tworzenia modeli RailML. Jedną z nich jest wykorzystanie programu Rail-AiD. Jest to zaawansowane narzędzie do tworzenia planów schematycznych sieci kolejowych, które posiada funkcje eksportu plików do formatu RailML, a także importu i edycji gotowych modeli.



Rys.1. Struktura schematów RailML
Źródło: dokumentacja RailML

2. Schematy RailML

2.1. *Infrastructure*

Schemat Infrastructure wykorzystywany jest do szczegółowego opisu elementów infrastruktury kolejowej [2]. Do definiowania różnych aspektów infrastruktury służą wyodrębnione podschematy.

Podschemat Topology umożliwia opis topologii kolejowej, która określa, jak fragmenty układu torowego (nazywane Elementami Sieciowymi) łączą się ze sobą poprzez rozjazdy, jakie istnieją możliwości przemieszczania się między nimi, a także, w jaki sposób posterunki ruchu są połączone liniami kolejowymi. Prawidłowo zdefiniowana topologia stanowi podstawę do wyznaczenia położenia elementów opisywanych w innych podschematach.

Podschemat Geometry służy do opisu geometrii toru. Pozwala na zdefiniowanie gradientów i łuków poziomych wraz z ich parametrami oraz umiejscowienie ich we wcześniej zadeklarowanej sieci topologii.

Podschemat Functional Infrastructure zawiera wszystkie elementy infrastruktury funkcjonalnej niezbędne do stworzenia kompletnego modelu sieci kolejowej. Pozwala między innymi na opis torów, posterunków ruchu, a także urządzeń sterowania ruchem kolejowym (srk) – sygnalizacji, urządzeń kontroli nie zajętości, itd. Odpowiednie atrybuty i elementy umożliwiają zdefiniowanie parametrów takich jak położenie lub szczegóły konstrukcji, a także relacje obiektu z warstwą topologii.

Podschemat Infrastructure Visualizations umożliwia stworzenie graficznej reprezentacji modelowanej sieci kolejowej. Pozwala na określenie parametrów wizualizacji, takich jak umiejscowienie elementu na ekranie oraz wykorzystywane symbole.

Ostatnim dostępnym podschematem jest podschemat Infrastructure States umożliwiający określenie stanu technicznego zdefiniowanych elementów infrastruktury. Pozwala to na odróżnienie urządzeń działających od tych planowanych lub wyłączonych z użytku. Istnieje także podschemat Physical Facilities, jednak w obecnej wersji RailML nie jest on wykorzystywany i nie zawiera żadnych dodatkowych elementów, a jego zastosowanie nie jest sprecyzowane.

2.2. *Interlocking*

Schemat Interlocking [1] wykorzystywany jest do opisu kolejowych układów zależnościowych. Pozwala on na modelowanie działania urządzeń zadeklarowanych w schemacie Infrastructure, a także definiowanie zależności między nimi. Schemat Interlocking umożliwia także modelowanie informacji pozyskiwanych z tablic zależności, określanie przebiegów i ich parametrów. Schemat wciąż się rozwija, a w kolejnych wersjach RailML rozszerzany będzie o dodatkowe funkcje, głównie związane z systemem ETCS.

2.3. *Common*

Schemat Common wykorzystywany jest do opisu wszystkich elementów, które nie pasują do pozostałych schematów, ale są przez nie wykorzystywane. Zawarte są w nim między innymi informacje na temat zarządców infrastruktury, sieci trakcyjnej, profili prędkości oraz wykorzystywanych w modelu systemów pozycjonowania. Zdefiniowanie przynajmniej jednego systemu pozycjonowania jest niezbędne do określenia położenia elementów opisywanych w schemacie Infrastructure.

2.4. *Rollingstock*

Schemat Rollingstock nie jest dostępny w wersji RailML 3.1, jednak planowane jest wprowadzenie go w kolejnych aktualizacjach. Jego celem jest umożliwienie szczegółowego opisu taboru kolejowego. Gotowy schemat będzie pozwalał na modelowanie pojedynczych pojazdów trakcyjnych i wagonów oraz łączenie ich w gotowe składy z określonymi parametrami.

2.5. *Timetable and Rostering*

Również Schemat Timetable and Rostering nie jest dostępny w aktualnej wersji RailML, ale trwają prace nad jego przygotowaniem. Będzie on umożliwiał wprowadzanie informacji niezbędnych do tworzenia rozkładów jazdy pociągów, a także definiowanie przeprowadzanych operacji i relacji między pociągami. Więcej szczegółów dotyczących dostępnych funkcji i elementów zostanie udostępnionych podczas kolejnych etapów rozwoju schematu.

3. Przypadki użycia

Przypadki użycia (ang. use cases) dla RailML tworzone są w celu opisu funkcjonalności języka w zależności od tego, na jakich aspektach sieci kolejowej ma skupiać się opracowywany model. Przypadek użycia ogranicza dostępność niepotrzebnych elementów i wykorzystuje jedynie niezbędne z punktu widzenia tworzonego modelu schematy i podschematy. Zapobiega to umieszczaniu w modelu niepotrzebnych elementów oraz gromadzeniu nadmiaru niewykorzystywanych danych. Obecnie do użytku udostępnione są trzy przypadki użycia, jednak wciąż trwają prace nad kolejnymi.

3.1. *SCTP*

Przypadek SCTP (ang. Schematic Track Plan) [5] skupia się na dokładnym odwzorowaniu planu schematycznego, opisie układu torowego wraz z urządzeniami srk oraz ich wizualizacji. W tym celu wykorzystywane są elementy ze schematów

Infrastructure oraz Common. W wytycznych SCTP zalecane jest przedstawienie sieci kolejowej przynajmniej na poziomie mikroskopowym, ze szczególnym uwzględnieniem lokalizacji elementów określonej poprzez kilometrąż linii. Jako że w tym przypadku użycia istotne są jedynie podstawowe informacje zawarte na planie schematycznym, SCTP nie uwzględnia geometrii torów ani informacji na temat układów zależnościowych. Wykorzystywany jest za to podschemat Infrastructure Visualizations, który poprzez przypisanie modelowanym elementom współrzędnych ekranowych oraz symboli umożliwia ich wizualizację na ekranie.

3.2. *NEST*

Przypadek NEST (ang. Network Statement) [6] umożliwia opis sieci kolejowej i jej elementów pod kątem przeprowadzanych operacji. Pozwala na określenie wszystkich informacji niezbędnych do prowadzenia ruchu kolejowego, które wykraczają poza standardowy plan schematyczny. NEST, podobnie jak SCTP, wykorzystuje jedynie schematy Common i Infrastructure, jednak występują różnice w zawartości poszczególnych elementów, a także w zalecanym poziomie szczególności. Rekomendowane jest zastosowanie mezoskopowego lub makroskopowego poziomu opisu sieci kolejowej. Istotne są tu informacje na temat sieci trakcyjnej oraz współrzędne geograficzne opisywanych elementów, niezbędne jest także zdefiniowanie geometrii toru. Nie występuje jednak możliwość utworzenia wizualizacji infrastruktury. Również zestaw wymaganych elementów infrastruktury funkcjonalnej jest inny niż w pierwszym opisywanym przypadku użycia – większy nacisk położony jest na elementy związane z ograniczeniami ruchu i prędkości oraz parametrami toru. NEST nie zawiera żadnych informacji ze schematu Interlocking.

3.3. *RSIM*

Przypadek RSIM (ang. Routes for Timetable Simulation) [7] służy do opisu zależności między urządzeniami srk, pozwala także na definiowanie przebiegów. RSIM wykorzystuje schematy Infrastructure oraz Interlocking. Skupia się na mikroskopowym opisie topologii sieci kolejowej, nie uwzględnia jednak geometrii toru, ani nie pozwala na wizualizację infrastruktury. Podczas modelowania wykorzystywane są jedynie elementy infrastruktury funkcjonalnej oddziałujące na ruch kolejowy lub bezpośrednio go kontrolujące. Za pomocą schematu Interlocking możliwe jest sprecyzowanie parametrów działania urządzeń srk oraz zachodzących między nimi relacji. Duży nacisk położony jest także na opis realizowanych przebiegów wraz z ich uzależnieniem od stanu urządzeń.

3.4. *Porównanie*

Poniżej przedstawiono porównanie trzech wybranych przypadków użycia dostępnych w wersji RailML 3.1. Zestawienie zawiera szczegółowe informacje na temat wykorzystywanych schematów, podschematów oraz elementów.

Tabela 1. Schematy

	SCTP	NEST	RSIM
Infrastructure (infrastruktura)	wymagane	wymagane	wymagane
Interlocking (urządzenia zależnościowe)	brak	brak	wymagane
Common (różne)	wymagane	wymagane	brak

Tab. 2 Podszchematy wchodzące w skład schematu Infrastructure

	SCTP	NEST	RSIM
Topology (topologia)	wymagane	wymagane	wymagane
Geometry (geometria)	brak	wymagane	brak
Functional Infrastructure (infrastruktura funkcjonalna)	wymagane	wymagane	wymagane
Infrastructure Visualizations (wizualizacje infrastruktury)	wymagane	brak	brak
Infrastructure States (stany infrastruktury)	opcjonalne	opcjonalne	brak

Tab. 3 Elementy wchodzące w skład podszchematu Functional Infrastructure

	SCTP	NEST	RSIM
Balisy	opcjonalne	brak	opcjonalne
Granice	opcjonalne	wymagane	wymagane
Sieć trakcyjna	brak	wymagane	brak
Kozły oporowe	wymagane	brak	wymagane
Skrzyżowania	wymagane	brak	wymagane
Wykolejnice	brak	brak	wymagane
Przejazdy kolejowo-drogowe	wymagane	opcjonalne	wymagane
Linie	wymagane	wymagane	opcjonalne
Skrajnia ładunkowa	opcjonalne	wymagane	brak
Punkty operacyjne	wymagane	wymagane	wymagane
Przejazdy nadziemne	wymagane	wymagane	brak
Przejazdy podziemne	wymagane	wymagane	brak
Perony	wymagane	wymagane	brak
Ograniczenia ruchu	brak	wymagane	brak
Odcinki serwisowe	wymagane	wymagane	brak
Sygnalizatory	wymagane	brak	wymagane
Rozjazdy	wymagane	brak	wymagane
Prędkości	brak	wymagane	wymagane
Tory	wymagane	wymagane	wymagane
Szerokość toru	brak	wymagane	brak
Urządzenia kontroli niezajętości	wymagane	opcjonalne	wymagane
Urządzenia systemu automatycznego ograniczania prędkości	brak	brak	opcjonalne
Łączność radiowa	brak	wymagane	brak
Ograniczenia wagowe	brak	wymagane	brak

łączące je linie kolejowe oraz przypisywanie im wcześniej utworzonych elementów mikroskopowych. Funkcja ta pozwala na budowanie złożonych modeli o dwóch różnych poziomach szczegółowości.

4.2. Zastosowania praktyczne przy tworzeniu modelu RailML

Dzięki możliwości eksportu gotowego modelu do formatu RailML, program RaIL-AiD może znacznie ułatwić tworzenie modeli infrastruktury kolejowej w tym standardzie. Jednakże, mimo dużych możliwości, RaIL-AiD, zwłaszcza w wersji demonstracyjnej, posiada pewne ograniczenia i wyeksportowane pliki nie zawsze zawierają wszystkie elementy niezbędne do stworzenia kompletnego modelu RailML.

Pierwszym ograniczeniem jest brak możliwości bezpośredniej edycji elementów topologii za pomocą interfejsu RaIL-AiD. Podczas modelowania topologii kluczowe jest przypisanie elementom sieciowym odpowiedniego położenia, a także nadanie im odpowiednich nazw. Interfejs RaIL-AiD nie udostępnia żadnej z tych opcji – nazwy elementów sieciowych przypisywane są automatycznie, bez możliwości edycji, a ich położenie można zadeklarować jedynie pośrednio, jeśli pokrywa się z położeniem elementów infrastruktury funkcjonalnej. Niedogodności te jednak nie zmieniają faktu, że wykorzystywanie edytora znacznie skraca czas potrzebny na zadeklarowanie elementów topologii i zachodzących między nimi relacji.

Kolejnym ograniczeniem aktualnej wersji RaIL-AiD jest brak możliwości zrezygnowania z niewykorzystywanych elementów. Tworzenie kodu RailML daje użytkownikowi dużą dowolność przy wyborze odpowiednich elementów i atrybutów, natomiast edytor RaIL-AiD posiada stałą bazę opcji i nie zawsze umożliwia rezygnację ze zbędnego w danym modelu elementu. Dla wielu elementów infrastruktury funkcjonalnej dostępne parametry pochodzą zarówno ze schematu Infrastructure, jak i Interlocking, co nie sprawdza się, gdy model ma w założeniu wykorzystywać tylko jeden schemat. Często niemożliwe jest także usunięcie obiektów automatycznie umieszczanych na planie schematycznym, takich jak rozjazdy wstawiane w miejscach połączenia elementów sieciowych lub kozły oporowe na ich końcach.

Ponadto, mimo że RaIL-AiD posiada bardzo szeroki wybór elementów infrastruktury możliwych do rozmieszczenia na planie schematycznym, nie wszystkie są zgodne ze standardem RailML i w efekcie nie jest możliwe ich wyeksportowanie. Przykład stanowią niektóre typy sygnalizatorów. Ta niekompatybilność znacznie utrudnia tworzenie kompletnych modeli jedynie przy pomocy edytora RaIL-AiD.

5. Podsumowanie

Język RailML umożliwia tworzenie złożonych cyfrowych modeli sieci kolejowych. Dzięki swojej budowie zapewnia dużą dowolność w wyborze wykorzystywa-

nych elementów i pozwala na dostosowanie wyglądu modelu do indywidualnych potrzeb użytkownika. Najnowsza wersja RailML pozwala na szczegółowe odwzorowanie infrastruktury kolejowej i systemów zależnościowych, a planowane aktualizacje wprowadzą możliwość opisu taboru kolejowego i modelowanie rozkładów jazdy. Opracowane przypadki użycia zapewniają możliwość wyboru, jakiego aspektu sieci ma dotyczyć model oraz przedstawiają schematy i elementy niezbędne do jego zrealizowania. Narzędziem ułatwiającym tworzenie skomplikowanych układów jest oprogramowanie Rail-AiD. Pozwala ono na tworzenie i edycję planu schematycznego za pomocą interfejsu graficznego oraz eksport gotowego modelu do formatu RailML. Liczne funkcje Rail-AiD znacząco ułatwiają modelowanie, jednakże program posiada też pewne ograniczenia narzucające użytkownikowi konkretny wygląd modelu i uniemożliwiające pełne wykorzystanie zalet RailML. Program jest jednak nieustannie rozwijany i kolejne wersje mogą przynieść rozwiązanie wyżej wymienionych problemów.

Badania były finansowane ze środków Politechniki Warszawskiej w ramach Programu Inicjatywa Doskonałości – Uczelnia Badawcza (IDUB)

Bibliografia

- [1] Ciszewski T., Komaszewski M., Nowakowski W., Zastosowanie RailML do opisu kolejowych systemów nastawczych, 2018.
- [2] Hlubuček A., RailTopoModel and RailML 3 in overall context, 2017.
- [3] RailML: schema documentation <https://www.railML.org/files/download/schemas/3.1/documentation/railML3.html>
- [4] RailML: the RailML subschemas <https://www.railML.org/en/user/subschemes.html>
- [5] Rahmig Ch., RailML Use Case Definition Schematic Track Plan, 2019.
- [6] Rahmig Ch., RailML Use Case Definition Network Statement, 2019.
- [7] von Lingen J., RailML Use Case Definition Routes for Timetable Simulation, 2019.

