

MILOŠ OSTOJIĆ*

RAILML STANDARD ZA RAZMENU PODATAKA U ŽELEZNIČKOM SAOBRAĆAJU

RAILML STANDARD FOR DATA EXCHANGE IN RAILWAY TRAFFIC

Datum prijema rada: 7.8.2020.

UDK: 656.2+004:656.2+33

REZIME:

RailML je format razmene podataka koji je razvio evropski konzorcijum železničkih kompanija, akademskih institucija i konsultantskih firmi zbog nedostataka i neefikasnog načina povezivanja različitih železničkih IT aplikacija. Njegov glavni cilj je da omogući heterogenim železničkim aplikacijama da međusobno komuniciraju i razmenjuju ulazne i izlazne podatke. Zbog složenosti podataka o železnici i mnogih sintaktičkih razlika među železnicama, softver koji koristi železnički format mora da bude sertifikovan što dalje obezbeđuje kvalitet RailML interfejsa. RailML protokol formiran je da poboljša razmenu podataka između železničkih informacionih tehnologija. Od 2002. god. RailML razvija se i unapređuje. Primenjuje se za kreiranje i primenu sistematskih XML orijentisanih rešenja razmene podataka između železničkih aplikacija. RailML jezik, na primer, daje univerzalno primenljive formate razmene podataka između vlasnika infrastrukture i raznih operatera, ali i omogućava dalji napredak primene neophodnih IT rešenja na železnici.

Ključne reči: RailML protokol, železnički saobraćaj, razmena podataka, železničke IT aplikacije

SUMMARY:

RailML is a data exchange format developed by a European consortium of railway companies, academic institutions and consulting firms due to the shortcomings and inefficient way of connecting different railway IT applications. Its main objective is to enable heterogeneous railway applications to communicate with each other. Due to the complexity of railway data and the many syntactic differences among the railways, software using the RailML format have to be certified. The certification ensures the quality of the RailML interfaces, which further ensures the quality of the RailML interface. The RailML protocol was formed to improve the exchange of data between railway information technologies. Since 2002. RailML is evolving and improving. It is used to create and implement systematic XML-oriented data exchange solutions between railway applications. The RailML language, for example, provides universally applicable data exchange formats between infrastructure owners and various operators, but also enables further progress in the implementation of the necessary IT solutions on the railway.

Key words: RailML protocol, railway traffic, data exchange, railway IT applications

* Miloš Ostojić, dipl. inž. saobr, Saobraćajni fakultet, Beograd, Vojvode Stepe 305, ostoja_95_cacak@hotmail.com

1. UVOD

Danas je računar postao osnovni alat za planiranje i rad, a preduzeća i železnice u tome nisu izuze-tak. Poslednjih nekoliko godina porastao je broj specijalizovanih kompjuterskih programa osmišljenih da adresiraju sve aspekte poslova u vezi sa prugama, od izrade voznog reda, preko rasporeda osoblja do plana infrastrukture. U većini slučajeva ovi programi imaju različite strukture podataka i protokole. Kako broj programa i aplikacija raste, tako i raznolikost različitih struktura podataka postaje sve veća. U cilju efikasnog korišćenja nekoliko različitih aplikacija za poboljšanje planiranja zajedno sa železničkim operacijama, podaci se moraju izvesti iz jednog programa i uvesti u drugi radi dalje obrade. To se može učiniti kopiranjem podataka između aplikacija (dug i mučan proces za programe sa bilo kojom količinom podataka) ili korišćenjem programa automatskog prenosa podataka (transparentno za korisnika).

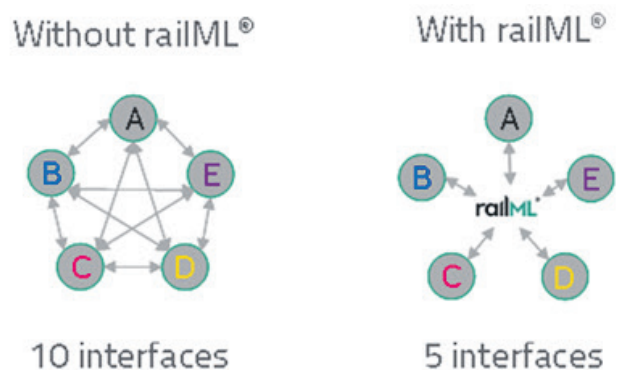
U svom radu [1] autori predlažu B2B model u oblasti železnica, zasnovan na kombinovanju integracije informacija i integracije portala u tehnološkom okruženju računarstva u oblaku. Integracija informacija vrši se u zajedničkoj bazi podataka SKL Azure. Integracija portala je omogućena uslugom hostovanom u sistemu Windows Azure. Predloženi model primenjen je u studiji slučaja integracije informacionih sistema, koji se koriste za upravljanje železničkim prelazima na železnicama Srbije.

Prema autorima rada [2] primena EU standarda na srpski železnički sistem, jedan je od glavnih preduslova za postizanje interoperabilnosti sa železničkim sistemom EU. Ovaj rad opisuje veb uslugu za razmenu podataka o voznom redu na osnovu vodiča za primenu Evropske železničke agencije. Opisani su detalji implementacije stvarnih formata železničkog reda vožnje i odgovarajuće softverske arhitekture, što potvrđuje prednosti SOA pristupa u poređenju sa drugim postojećim rešenjima u ovoj oblasti.

Nekoliko specijalizovanih interfejsa između programa trenutno postoje, ali kako broj aplikacija za železnice raste ovo rešenje postaje manje efikasno. Prenos podataka između dva određena programa zahteva razvoj dva različita interfejsa (po jedan za svaki pravac prenosa). Kako se broj programa povećava broj mogućih prenosa podataka značajno

raste. Na primer, pod pretpostavkom da postoji pet programa u teoriji treba razviti dvadeset aplikacija za potpunu razmenu podataka između svih tih programa. Generalno gledano, ako se broj programa, n , linearno povećava, broj interfejsa $[n * (n-1)]$ povećava se kvadratno. Kako broj novih tehničkih rešenja, kao što su nacrt i praćenje, vremenska procena i prilagođavanja, raste, tako se i broj programa za železnice povećava. Jasno, broj novih interfejsa koje je potrebno razviti za ove nove aplikacije predstavlja veliki izazov železničkom sektoru.

RailML.org je inicijativa koja podržava dalji razvoj otvorenog koda železničkog jezika za označavanje, pod nazivom RailML. Kao svaki softver otvorenog koda, RailML šeme su generalno karakterisane pristupačnošću izvornom kodu i dokumentaciji, koja je javnosti dostupna bez ikakvih troškova. Svi ovi atributi ispunjeni su od 2002. godine inicijativom railML.org. Njegov glavni cilj je da omogući heterogenim železničkim aplikacijama međusobnu komunikaciju [3]. Pored toga, ima za cilj da ispuni potrebe zainteresovanih grupa korisnika u održavanju međunarodnog standarda industrije za razmenu podataka o železnicama. Danas je povezivanje različitih železničkih softverskih paketa prepuno problema. Svrha inicijative railML.org ogleda se u pronalaženju, diskusiji i predstavljanju sistemskih rešenja zasnovanih na XML-u za uprošćenu razmenu podataka između železničkih aplikacija (slika 1).



Slika 1. Sa RailML i bez RailML (Izvor: [3])

RailML standard nije proizvod pojedinca ni centralizovanog tela. On je razvijen u procesu rada i diskusije između uključenih partnera. Sve zainteresovane institucije i kompanije pozvane su da učestvuju u railML.org inicijativi.

Zbog složenosti podataka o železnici i mnogih sintaktičkih razlika između železnica, softver koji koristi železnički format mora da bude sertifikovan. Sertifikacija obezbeđuje kvalitet RailML interfejsa i ne donosi dodatne troškove osim jednokratne sertifikacije. Nakon što se izvrši provera da li podležu uslovima licenciranja, rezultati se mogu preuzeti na veb-stranici railML.org. Tamo se mogu pronaći XML šeme, uzorci datoteka podataka i detaljnih specifikacija RailML-a.

Članovi railML.org trenutno su:

- železnice,
- menadžeri infrastrukture,
- železnička preduzeća,
- kompanije,
- softverski programeri,
- vlasti,
- istraživački instituti i univerziteti,
- RailML protokol za razmenu podataka u železničkim aplikacijama.

1.1. Istorija RailML-a

RailML je format razmene podataka koji je razvio evropski konzorcijum železničkih kompanija, akademskih institucija i konsultantskih firmi usled hroničnih poteškoća povezivanja različitih železničkih IT aplikacija. Projekt railML.org, osnovan 2002. godine, nastoji da kontinuirano razvija ovaj format kako bi se olakšala njegova upotreba u širokom spektru železničkih aplikacija. Projekat je započeo kao partnerstvo između Frauenhofer instituta za transportne sisteme i infrastrukturu (FhG-IVI) i švajcarskog Federalnog tehnološkog instituta "Institut za planiranje i sisteme transporta" [4], a trenutno ga vodi mali nezavisni tim. Pored njih uključili su se univerzitetski istraživači, kompanije koje rade na železnici, privatni istraživački instituti i konsultantske firme. Institut Fraunhofer služi kao tehnički partner koordinator koji pruža resurse, kao što su veb-stranica i forum za diskusiju. Razvoj RailML-a zasnovan je na predlozima, komentarima i kritikama partnera. Pojedinačni programeri aplikacija rade zajedno na koordiniran način kako bi dovršili rad na razvoju elemenata RailML-a. Ovi programeri formiraju grupe (otvorene za sve zainteresovane partnere) na osnovu individualnih interesa i zatim komuniciraju kroz moderirane onlajn korisničke forume, grupe za

vesti i redovne sastanke. Dok grupe završavaju sa radom na RailML elementima, članovi konzorcijuma, koji služe kao tehnički koordinatori, ih dovršavaju i objavljuju standarde (slično kao RFC za Internet). Ukratko, RailML standardi razvijeni su u kontekstu tehničkih diskusija koje su otvorene za sve zainteresovane za razvoj aplikacija za železnički sektor. RailML je objavljen kao niz XML šema i podšema, od kojih svaka obuhvata određeno područje železničkih domena. Prva šema RailML-a razvijena je 2005. godine. Nakon prvih iskustava sa razvojem i upotrebom railML 1.k, railML 2.0. objavljen je 2009. godine. Od 2012. godine railML.org je registrovano udruženje po nemačkom zakonu (lokalni sud Dresden VR 5750). U 2014. godini započeta je saradnja sa UIC-om, nakon čega je mnogo železničkih preduzeća u državnom vlasništvu (OBB, DB, SNFC, itd) postalo železnički partner. Sam model podataka trenutno je fokusiran na predstavljanje informacija koje se koriste u železničkim operacijama i voznom redu, ali mogu se proširiti i na druge oblasti.

1.2. Troškovi korišćenja RailML-a, uslovi licenciranja i nabavke

RailML® je format za razmenu podataka otvorenog koda. To znači da je dostupan svima besplatno. Creative Commons licenca, primenjuje se na RailML od 2009. godine. Međutim, razvoj i održavanje standarda zahteva napore koje trenutno ulaže zajednica railML.org.

RailML.org koordinira posao u vezi sa RailML i nije komercijalno preduzeće sa namerom da ostvari profit. Iako razne železnice, programeri softvera i druge organizacije (poput UIC-a) velikodušno podržavaju ideju besplatnog, zajedničkog i jedinstvenog modela razmene podataka za železnički sektor, neke posebno opterećene usluge ne mogu se pružiti besplatno ili se uopšte ne mogu pružiti.

U nastavku dat je kratak pregled besplatnih, naplaćenih i neponuđenih usluga railML.org-a u domenu RailML.

Besplatne usluge:

- upotreba šema RailML u softverskom interfejsu (vaš dobavljač softvera može da naplati implementaciju, ugovor o održavanju ili ažuriranja, ali railML.org neće),

- pristup informacijama i podacima zajednice putem korisničkog naloga na www.railML.org,
- članstvo u svojstvu partnera railML.org (podrška, korisnik ili programer),
- prava pristupa i upotrebe u svrhu testiranja uzoraka podataka RailML,
- pristup železničkom forumu sa pet odbora za diskusiju (TT, IS, IL, RS, CO),
- učestvovanje u radnim grupama za razvoj TT, IS ili IL za železničke programe (za dugoročne saradnike koji se pridržavaju pravila radne grupe railML.org),
- pristup železničkom viki-u,
- podnošenje predloga za proširenje šinskih šema,
- učestvovanje na railML.org konferencijama i događajima (zahvaljujući pojedinačnom sponzorisanju partnera railML.org svi putni troškovi i troškovi biće nadoknađeni),
- preuzimanje i upotreba railVIVID: (pregledač i validator RailML - pokreće UIC),
- bitne informacije o RailML-u, railML.org, upotrebi RailML-a i najboljim postupcima.

Naplative usluge:

- sastanci za prezentaciju proizvoda kao i opšte i posebne radionice o RailML-u,
- konsultacije sa stručnjacima railML.org u vezi sa najboljom praksom (upravljanja podacima / strategijom razmene),
- nabavka obaveznog sertifikovanja (jednokratna naknada) za softver ili RailML interfejs pre upotrebe,
- troškovi za prekid i odustajanje od upozorenja i pravne takse izazvane nepravilnim korišćenjem intelektualnog vlasništva railML.org i kršenjem licencnih uslova.

Nije ponuđeno:

- programiranje za implementaciju za izvoz ili uvoz RailML® u softveru (ove usluge će pružati samo softverski programer ili dobavljač),
- Pristup voznom redu, infrastrukturi, bazama podataka za zaključavanje ili voznom parku u formatu [railML](http://railML.org) datoteke.

Sve šeme RailML-a dostupne su za razmenu podataka za celi železnički sektor, bez obzira na njihovu upotrebu. To mogu biti razvoj interfejsa, doprinos daljem razvoju šema RailML ili upotreba softvera

sa RailML interfejsima. Šeme RailML-a koriste se besplatno nakon prihvatanja uslova licence. Registrovani korisnici RailML-a su dobrodošli da doprinesu daljem razvoju šema.

Kako je RailML kontinuiran razvoj, koji traje od 2002. godine, verzije RailML objavljene su pod malo drugačijim uslovima licenciranja, i to:

- **railML 0.x – 1.0** railML.org omogućava vam trajno, nezabranjujuće, besplatno, svetsko pravo i licencu za korišćenje specifikacije railML.org i kopiranje, objavljivanje ili deljenje u skladu sa autorskim pravima u specifikaciji. RailML.org je spreman da, pod uslovima autorskih prava autora, izda besplatnu licencu za sprovođenje i upotrebu oznaka i principa šeme railML.org za razvoj računarskih programa koji slede ove principe. Ova licenca daje se pod uslovom da se slažete da nećete zahtevati od railML.org ili drugih strana bilo koja prava intelektualnog vlasništva u vezi sa implementacijom specifikacije railML.org. RailML.org izričito zadržava sva ostala prava koja se tiču sadržaja i predmeta specifikacije railML.org. RailML.org izričito odbija bilo kakvu garanciju za specifikaciju, uključujući i bilo kakvu garanciju da ova specifikacija ili njena primena ne krše nikakva prava treće strane. Ukoliko dođe do izmena u samim šemama, na sajtu se mogu naći obaveze koje korisnici moraju da sprovedu;
- **railML 2.0 – 2.1 interfejsi registrovani do juna 2013. godine.** Za te interfejse primenjuje se licenca Creative Commons "Attribution-NoDerivs 2.0 Generic" (CC BY-ND 2.0);
- **railML 2.0 – 2.x interfejsi registrovani posle juna 2013. godine.** Šeme [railML](http://railML.org) (Verzija 2) moraju se koristiti u skladu s ograničenom Creative Commons Attribution-NonCommercial-NoDerivs 3.0 neprijavljenom licencom.

Sledeći zahtevi moraju biti ispunjeni:

- besplatna registracija na railML.org uslov je za započinjanje radova na implementaciji (tj. kreiranje bilo kojeg programa uvoza, izvoza ili transformacije za podatke RailML; takođe za potrebe testiranja) shema RailML,
- uspešna sertifikacija putem railML.org obavezna je pre bilo koje produktivne ili komercijalne upotrebe interfejsa,

- šeme RailML-a moraju se distribuirati samo putem <http://www.railML.org>. i svaka dalja distribucija je zabranjena,
- railML.org mora biti spomenut u bilo kojoj softverskoj komponenti, priručniku ili listiću gde je softver opisan sa svojim karakteristikama.

Pored toga, trebalo bi imati na umu uslove CC-BI-NC-ND-3.0-licence. Komercijalna upotreba:

Nakon uspešne sertifikacije railML interfejsa, upotreba šema RailML u ovom programu/servisu biće prebačena sa prethodne CC-BI-NC-ND-3.0 u besplatnu licencu CC-BI-ND-3.0, koja omogućava komercijalnu upotrebu upotreba railML interfejsa (gore navedeni dodatni uslovi 3. i 4. ostaju na snazi). Partner RailML dobiće pisanu potvrdu u izveštaju o sertifikaciji.

railML 3.1 – 3.x. Šeme railML (verzija 3) moraju se koristiti u skladu s ograničenom Creative Commons Attribution-NonCom Commercial-NoDerivs 4.0 neprijavljenom licencom. Sledeći zahtevi moraju biti ispunjeni:

- besplatna registracija na railML.org preduslov je za započinjanje radova na implementaciji (tj. kreiranje bilo kog programa uvoza, izvoza ili transformacije za podatke MLML, ali i za potrebe testiranja) šema RailML,
- sertifikacija putem railML.org obavezna je pre bilo koje produktivne ili komercijalne upotrebe interfejsa,
- šeme RailML-a moraju se distribuirati samo putem <http://www.railML.org>. i svaka dalja distribucija je zabranjena,
- railML.org mora biti spomenut u bilo kojoj softverskoj komponenti, priručniku ili listiću gde je softver opisan sa svojim karakteristikama.

Pored toga, trebalo bi imati na umu uslove CC-BI-NC-ND-4.0-licence. Bitno je da se obrati pažnja na licence u sažetom izdanju i puni i obavezujući zakonski kod pre upotrebe bilo kog RailML® šema. Za komercijalnu upotrebu, nakon uspešne sertifikacije RailML interfejsa, upotreba šema RailML u ovom programu/servisu biće prebačena sa prethodne CC-BI-NC-ND-4.0 u besplatnu licencu CC-BI-ND-4.0, koja omogućava komercijalnu upotrebu RailML interfejsa (gore navedeni dodatni uslovi 3. i 4. ostaju na snazi). Partner RailML dobiće pisanu potvrdu u izveštaju o sertifikaciji.

Uslovi licence se prihvataju preuzimanjem šeme i njenog dodatnog materijala ili upotrebom railML datoteka. Dodatni ili različiti uslovi za licencu moraju se pismeno dogovoriti sa koordinatorima železničke pošte. Trebalo bi imati na umu da sertifikati i komercijalna upotreba šema RailML trenutno nisu dostupni za sve verzije (npr. railML 1.0, 1.1, 2.0, 2.1; od jula 2021 i railML 2.2).

1.3. Korišćenje i preuzimanje RailML-a

Preuzimanje RailML-a:

1. Prvo je potrebno registrovati se na railML.org. Registracija je besplatna. Registracija je važna da pratimo upotrebu RailML® jer se njegove prednosti oslanjaju na nesmetanu razmenu podataka.
2. Prijaviti se na railML.org i sačuvati na: <http://www.railML.org/en/developers/download.html>.
3. Pristanak na uslove licence. RailML® se licencira dvostruko ili sa ograničenom CC-BI-NC-ND licencom ili komercijalnom licencom.
4. Preuzeti zip datoteku s trenutnom verzijom datoteka šeme RailML®.

Za dobijanje verzije programera (ne preporučuje se početnicima i za aplikacije na terenu) treba se obratiti koordinatorima railML.org.

Korišćenje RailML-a može se olakšati zu sledeće smernice:

- RailML®-Wiki (Wikipedia) je osnova dokumentacije. Zbirku vodiča za korisnike i programere (npr. kako doprineti ovom vikiju) možete pronaći na ovom sajtu.
- Ukoliko postoje pitanja koja su bez odgovora na viki stranici, može se prijaviti na forum, gde programeri i korisnici RailML-a mogu razmeniti svoja iskustva i razgovarati o problemima ili poboljšanjima ili, takođe, odgovoriti na sva vaša pitanja.
- Na konferenciji railML.org možete upoznati druge korisnike, programere i koordinate. Konferencije se obično održavaju dva puta godišnje.
- Proverite i testirajte svoj model interfejsa. Iako se RailML® može koristiti u oflajn izolovanim modelima, npr. za vežbanje, u većini slučajeva produktivna upotreba zavisi od razmene podataka sa drugim jedinicama (druga aplikacija;

drugi podsistem unutar vaše kompanije; drugo preduzeće). Kako biste omogućili nesmetan rad, licenca zahteva da sertifikujete svoj interfejs ako ga komercijalno koristite (razmenjujući podatke sa drugima).

Razvoj RailML-a odvija se u radnim grupama eksperimenta iz oblasti železničkog softvera, modeliranja podataka i IT stručnjaka. Radne grupe su definisane sa četiri različite podšeme RailML-a. Svaka radna grupa sastaje se odvojeno jednom u dva do tri meseca kako bi razgovarali o pitanjima sadržaja i daljim koracima razvoja. Celokupna zajednica RailML sastaje se dva puta godišnje na zvaničnim konferencijama railML.org. Prolećna konferencija uglavnom se održava u nemačkoj govornoj zajednici u Evropi. Jesenja konferencija se najčešće održava u Parizu i obraća se međunarodnoj železničkoj zajednici. Ona se uglavnom održava na engleskom jeziku.

RailML.org je industrijska asocijacija, koju vode njeni članovi. To znači da je i format razmene podataka RailML® industrijski standard koji koristi širok spektar železnica, računarskih programa i projekata. Sav razvojni rad na railML.org odvija se po principu odozdo prema gore, što garantuje da se sve prijavljene potrebe industrije uzmu u obzir i da se mogu implementirati na brz i dugotrajan način. Primeri uspešnih i široko rasprostranjenih industrijskih standarda su USB (univerzalan serijski sabirnik), Bluetooth ili GSM (Globalni sistemi za mobilne komunikacije).

Vladine norme kao što su ISO ili CEN razvijaju stručne grupe i nakon toga se primenjuju odozgo prema dole. Državne institucije, koje nisu nužno usklađene sa potrebama industrije, razvijaju i objavljuju norme, kojih će se industrija nakon toga pridržavati. Mogu se javiti problemi jer glas industrije nije u primarnom fokusu tokom razvojnih procesa i proces je zavisn od vladinih pravila i životnih ciklusa.

Uporedo sa razvojem programa sa svojim partnerima railML.org saraduje i sa drugim organizacijama u sektoru železnice, saobraćaja i standardizacije. A to su :

- Međunarodna železnička unija (UIC),
- Evropska organizacija za bezbednost vazdušne plovidbe (EUROCONTROL),

- Evropska železnička agencija (ERA).

1.4. Razvojno partnerstvo

RailML.org je razvojno partnerstvo nezavisnih preduzeća i institucija. Njihovi predlozi, komentari i kritike su osnova za razvoj i konsolidaciju šema RailML®. Partneri se definišu u tri različite grupe: korisnici, pristalice i programeri. Trenutno se radi na pružanju detaljnih informacija o svakom partneru, a to su:

- Programeri - RailML programerski partneri su kompanije sa dugogodišnjim iskustvom u stvaranju RailML interfejsa i utiču na razvoj RailML standarda na Forumu, u Wikiju ili na redovnim RailML konferencijama. Prethodno članstvo kao podrška RailML-a je od suštinske važnosti;
- Pristalice RailML-a su partneri koji napreduju ka cilju jedinstvenog rešenja za razmenu podataka sa RailML interfejsima. Nije potrebno aktivno učestvovanje u razvoju šema RailML-a;
- Korisnici RailML su uglavnom železničke kompanije i operatori koji prihvataju RailML interfejse. Iako ne poboljšavaju standard RailML-a, imaju veliko iskustvo u svakodnevnoj razmeni RailML podataka.

2. PROŠIRIVI JEZIK ZA OZNAČAVANJE - XML

XML je standardni skup pravila za definisanje formata podataka u elektronskoj formi. Propisala ih je W3C. Pravila XML standarda su: korisnici definišu sopstvene (XML) formate podataka, koje mogu koristiti za njihovo skladištenje, obradu i razmenu. XML je skraćenica za Extensible Markup Language, odnosno proširivi (meta) jezik za označavanje tekstualnih dokumenata. Ideja je bila da se stvori jezik koji će ljudi i računarski programi moći jednostavno da čitaju. XML definiše opštu sintaksu za označavanje podataka pomoću odgovarajućih etiketa (tagova) koje imaju poznato i razumljivo značenje. Format koji obezbeđuje XML za računarske elemente može se prilagoditi najrazličitijim oblastima, kao što su elektronska razmena podataka, čuvanje podataka, odvajanje podataka od prezentacije, vektorska grafika, sistemi glasovne pošte, izrada novih specijalizovanih jezika za označavanje.

Pošto se svi XML formati podataka kreiraju pomoću istog skupa pravila, moguće je napraviti univerzalne

alate za njihovu obradu. Tako postoji veliki broj besplatnih biblioteka na različitim jezicima na kojima se može pisati i čitati XML, a mogu se upotrebiti i gotovi softveri (kao što su programi za obradu teksta ili čitači veba). Neke alatke mogu da rade sa svim XML dokumentima, dok su druge prilagođene za podršku XML aplikacijama u različitim oblastima, ali se uvek koristi ista sintaksa. Mnogi formalno definisani jezici za označavanje kao što su RSS, MathML, GraphML, XHTML, Scalable Vector Graphics, MusicXML i sl. bazirani su na XML-u.

XML omogućava stvaranje formata podataka koji su nezavisni od platforme. Često se dokumenti pisani na jednoj platformi ne mogu čitati na drugoj platformi, niti u različitim programima na istoj platformi, čak ni u ranijoj verziji jednog programa na istoj platformi. U XML dokumentima tekstualni su ne samo podaci već i etikete koje su smeštene u samoj XML dokumentaciji. To znači da ih može čitati svaka alatka koja je u stanju da čita tekstualne datoteke. Tako se podaci mogu prenositi sa jednog na drugi sistem. Tako XML daje prenosive podatke.

Osnovna svrha XML-a je da olakša deljenje podataka kroz različite informacione sisteme, posebno kroz sisteme povezane sa internetom.

2.1. Prednosti i nedostaci

XML je jednostavan i čitljiv i čoveku i računaru. Pošto je XML dokument obično tekstualna datoteka, čitljiva je svakoj platformi koja čita tekstualne podatke. Tako postaje neosetljiv na tehnološke promene bez obzira na napredak tehnologije, tekstualne podatke će još dugo svaki računar moći da pročita. XML podržava UNICOD i omogućava prikaz teksta na svim poznatim jezicima.

Takođe, ovaj format je samodokumentujući: etikete opisuju sadržaj koji se nalazi unutar njih. Ispravnost nastalog XML dokumenta vrlo je jednostavno kontrolisati, jer XML ima stroga sintaksička pravila. Računarski programi za obradu dokumenata mogu jednostavno da obrađuju XML sadržaj.

XML je međunarodno prihvaćen standard. Prihvatili su ga mnogi proizvođači programa i koriste ga u svojim proizvodima. Može se reći da je za XML najvažnije to što predstavlja jezik za metaoznačavanje, što podrazumeva fiksni skup oznaka i elemenata. U

imenu XML-a X potiče od reči EXtensible (proširiv). To znači da se jezik može prilagođavati i proširivati kako bi zadovoljio različite potrebe korisnika.

Jedan od osnovnih nedostataka je redundantna i opširna sintaksa. Ona može zbunjivati čoveka, a računarski program koji obrađuje dokument može delimično usporiti. Ovi računarski programi su veoma složeni jer moraju da obrađuju velike količine ugnežđenih podataka na više nivoa.

Treba imati u vidu da je XML samo jezik za označavanje, a ne programski jezik. XML dokument samo postoji, ne radeći ništa. Kao ni HTML, tako ni XML ne šalje podatke preko mreže. Poslati podaci mogu biti kodirani u XML-u, ali izvan XML dokumenta mora da postoji neki protokol (http, ftp, nfs i slično) koji će poslati dokument. To znači na XML nije protokol za mrežni prenos.

2.2. Sintaksa

U XML dokumentu podaci su smešteni u obliku znakovnih nizova, koji se nalaze između tekstualnih etiketa koje ih opisuju. U XML-u se osnovne jedinice etiketa i podataka nazivaju elementi.

Korisnik se pri pisanju etiketa mora pridržavati precizno definisane sintakse. Strukturu XML dokumenata opisuju njegove etikete, pomoću kojih se može videti koji su elementi pridruženi drugim elementima. Ako je XML dokument dobro projektovan, etikete opisuju i njegovu semantiku (npr. etikete ukazuju da li je element naslov, ime osobe, datum ili slično). Ali, etikete u dobro projektovanim XML aplikacijama ne govore ništa o načinu prikazivanja dokumenata (npr. da li je element ispisan polucrno ili kurzivom i slično) jer XML jezik ne označava način prikazivanja, već strukturu i semantiku.

Bazična sintaksa za jedan dokument u XML-u je:

```
<osoba>Miloš Ostojić</osoba>
```

2.2.1. Elementi, oznake i znakovni podaci

XML elementi opisuju određeni deo XML dokumenta. Jedan element je razgraničen početnom i završnom oznakom. Početna oznaka se sastoji od imena elementa okruženog uglastom zagradom (osoba). Završna oznaka se sastoji od istog imena okruženog uglastim zagradama, ali i od crtice koja

prethodi imenu elementa koja označava zatvaranje elementa tj. završavanje. Sve što se nalazi između početne i završne oznake predstavlja sadržaj elementa. U ovom primeru sadržaj je Miloš Ostojić. Iako su razmaci deo sadržaja, mnoge aplikacije ih zanemaruju. Etikete `<osoba>` i `</osoba>` čine markiranje dokumenata, dok sadržaj između njih predstavlja znakovne podatke.

XML obezbeđuje posebnu sintaksu za predstavljanje elemenata praznog sadržaja. Ova oznaka počinje `<`, a završava se znakovima `/>`, ovaj element označava prekidanje reda.

Korenski element je deo svakog XML dokumenta koji uokviruje kompletan sadržaj dokumenata. U dokumentu, on je prvi element i u sebi sadrži sve druge elemente. Ponekad se korenski element naziva i element dokumenta. Na primeru se može videti kako to izgleda u vidu "stabla":

```
<osoba>
  <ime_i_prezime>
    <ime>Miloš</ime>
    <prezime>Ostojić</prezime>
  </ime_i_prezime>
  <zanimanje>WEB DEVELOPER</zanimanje>
  <zanimanje>INŽENJER</zanimanje>
</osoba>
```

Ovaj dokument još uvek sadrži samo jedan element - osoba. Ali, ovaj element ima tri elementa - potomka: jedan element `<ime_i_prezime>` i dva elementa `<zanimanje>`. Element `<ime_i_prezime>` ima dva elementa - potomka: element `<ime>` i element `<prezime>`. Element `<osoba>` je roditeljski element `<ime_i_prezime>` i `<zanimanje>`, dok su `<ime_i_prezime>` i `<zanimanje>` bratski elementi, tako su `<ime >` i `<prezime>` bratski elementi.

2.2.2. Atributi

XML atributi daju podatke koji dodatno opisuju elemente XML dokumenata. Atributi imaju svoj naziv i vrednost. Ime atributa je razdvojeno od vrednosti znakom jednakosti i razmakom, koji je neobavezan. Vrednost atributa mora biti zatvorena u navodnike ili polunavodnike, npr:

```
<osoba rođena="1893" umrla="1977">
  Miloš Crnjanski
</osoba>
```

Postoje dve struje mišljenja u vezi sa korišćenjem atributa u okviru XML dokumenata. Po jednoj, atributi ne bi smeli koristiti obogaćivanje smisla sadržaja, tj. dozvoljeni su samo oni atributi koji ne dodaju ništa u osnovnu informaciju (kao na primer atributi koji opisuju vrstu i veličinu fonta). Prema drugom shvatanju, atributi se mogu slobodno koristiti jer njima se smanjuje broj elemenata.

2.2.3. Reference

Unutar elemenata XML dokumenata ne sme se naći znak `<` koji nema odgovarajuću izlaznu sekvencu `</>`, jer se znak `<` tumači kao početak etikete. Ovaj znak se, ukoliko korisniku zatreba u tekstu, može pretvoriti u izlaznu sekvencu pomoću reference entiteta (eng. entity reference) `<`, numeričke reference znaka `<`. Reference entiteta i numeričke reference znaka spadaju u markiranje; pri analiziranju XML dokumenata, markiranje se zamenjuje onim znakom na koji upućuje referenca.

XML poseduje pet unapred definisanih entiteta, a to su:

- `&`; (&, ampersend);
- `<`; (<, znak "manje od");
- `>`; (>, znak "veći od");
- `"`; (" ,običan navodnik);
- `'`; (ravan polunavodnik, apostrof).

3. STANDARDNI FORMATI ZA EFIKASNIJU RAZMENU PODATAKA I RAZVOJ RAILML-A

Situacija u železničkom sektoru nije nesvakidašnja, kako računari postaju sve neophodniji u društvu tako raste sve veća potreba za efikasnom razmenom podataka. Danas ovi tipovi problema sa nekompatibilnošću podataka rešavaju se pomoću "otvorenih i univerzalno upotrebljivih formata podataka". Takvi formati su uglavnom razvijeni pomoću ekstenzibilnog jezika (XML) kao osnove (npr. MathML za matematički izraz i GML za geografske podatke). XML je razvio World Wide Web za upotrebu u WWW aplikacijama (najpoznatiji označni jezik je XHTML). XML nije jezik aplikacije, već skup pravila koja se mogu koristiti za definisanje drugih "označnih" jezika i zato funkcioniše kao meta jezik. Glavna prednost dokumenata baziranih na XML-u je u tome što oni opisuju i podatke, a i strukture podataka.

Stoga je XML bio idealno rešenje za prenos i skladištenje podataka o železnici. Problem sa prenosom podataka nastao je povećavanjem broja aplikacija i može se rešiti pomoću EAI (Enterprise Application Integration).

EAI omogućava saradnju različitih programa kreiranjem standardnih interfejsa, koji su nezavisni od bilo koje aplikacije, ali se fokusiraju na objekte koji će se razmenjivati. Odgovarajući opisi objekata zasnivaju se na XML sintaksi. Efektivna razmena podataka između dve aplikacije zahteva da program uključuje funkciju za generisanje XML podataka (izbacivanje) i funkciju za čitanje i interpretiranje XML datoteka (unos). Zbog obostrane upotrebe XML-a i standardizacije koju je izazvao World Wide Web, većina računarskih platformi za razvoj aplikacija uključuju unapred definisane biblioteke i funkcije za obradu podataka zasnovanih na XML-u, što značajno smanjuje vreme potrebno za razvoj funkcija izbacivanja i unosa podataka za stvaranje specijalizovanih aplikacija kao što su one koje su namenjene za železnice.

3.1. Šeme za skladištenje metapodataka o izvoru ili aplikaciji

Model RailML definisan je u tri XML šeme, koje su opisane u sledećim odlomcima, a osim toga postoji i osnovna ('uobičajena') šema za skladištenje metapodataka o izvoru podataka ili aplikaciji. Ove teme su dalje podeljene u dodatne podšeme koje se bave specifičnijim oblastima. Planira se i raspravlja o dodatnim železničkim podšemama kao što su lista članova posade, upravljanje imovinom ili podaci u realnom vremenu. Jedna RailML datoteka može sadržati nekoliko podšema. Tri glavne šeme su:

- infrastruktura (IS),
- vozni park železnica (RS) (Rollingstock),
- red vožnje (TT) (Timetable).

3.1.1. Šema za predstavljanje železničke infrastrukture

Šema infrastrukture definiše elemente i strukturu za predstavljanje železničke infrastrukture i pridruženih upravljačkih informacija. Ova šema je sada visoko razvijena i ima mogućnost da efektivno izrazi evropsku železničku infrastrukturu. Ključni pojmovi:

- topologija mreže i ruta (poznatija kao "makroskopski" grafikoni u RailML-u),
- praćenje odseka, skretnica i prelaza i topologije linija ("makroskopski" čvorovi i lukovi),
- elementi za operacije i upravljanje kao što su signali, balize i kontrolna područja (uključujući postrojenje za identifikaciju blokada),
- infrastruktura (mostovi, tuneli, prelazi u nivou),
- propratni atributi/mogućnosti i granice između njih (elektrifikacija, gradijent, vlasnik, merač, brzina linije),
- dalji koncepti za modeliranje vizuelne infrastrukture na simulatorima.

Podaci o dokumentaciji i primeri upotrebe za većinu koncepata dostupni su iz redovno ažuriranih RailML vikija, zajedno s detaljima dizajnerskih odluka donesenih u razvoju modela železničkih podataka RailML [5].

3.1.2. Šema uvida u železnička vozila i vozne parkove

Šema železničkog voznog parka omogućava uvid u železnička vozila i vozne parkove u različitom stepenu apstrakcije i široko se koristi u razmeni podataka o simulatoru železnice. Na najopštijem nivou, može se definisati vozilo bez drugih podataka osim identifikatora ili nekih osnovnih vučnih karakteristika. Model, međutim, takođe omogućava specifikaciju različitih vagona da označe voz, kategoriju vozila i naprednije vučne proračune. Faktori kao što su tip pogona, sposobnost kočenja i mehanički gubici vuče mogu takođe biti uz mnoge druge atribute koji omogućavaju vozovima da se fizički modeliraju do visokog nivoa detalja.

3.1.3. Šema za efikasno predstavljanje modela i rasporeda vozova

Šema rasporeda i raspored vozova su takvi da omogućavaju efikasno predstavljanje modela i rasporeda. Omogućavaju i predstavljanje brojnih operativnih ideja, poput spajanja vozova, i pružaju mogućnost povezivanja infrastrukture definisane u "IS" šemi sa informacijama o voznom redu [6]. Oslanjajući se na podatke železničke voznog reda vožnje infrastrukture železnice i šemu "TT" kretanja vozova mogu se sačuvati (slika 2).

```

<ocpTT ocpRef="_85ZUE" ocpType="begin">
  <times scope="scheduled"
departure="08:37:18.0" />
  <times scope="published"
departure="08:37:00.0" />
  <sectionTT section="85ZUE-85ZVIA"
lineRef="_70000" trackInfo="1" distance="1.4">
    <trackRef ref="t70000-1" />
    <trackRef ref="t70000-1" />
    <runTimes minimalTime="PT2M" />
  </sectionTT>
</ocpTT>
<ocpTT ocpRef="_85ZVIA"
ocpType="pass">
  <times scope="scheduled"
arrival="08:39:54.0" departure="08:39:54.0" />
  <sectionTT section="85ZVIA-85ZWIP"
lineRef="_70005" trackInfo="1"
distance="0.907">
    <trackRef ref="t70005-1" />
    <runTimes minimalTime="PT0M42S" />
  </sectionTT>
</ocpTT>

```

Slika 2. Izgled šeme resporeda vozova

U radu [7] autori se bave preraspoređivanjem vozova, ali raspored se ne beleži u XML formatu.

3.1.4. Uobičajena ili obična šema za dodavanje metapodataka

Osnovna ili "uobičajena" šema omogućava dodavanje metapodataka (informacije o aplikaciji za koju su podaci ili koje treba iskoristiti) modelu. U najnovijem izdanju RailML-a to se navodi pomoću Dublin Core-a, široko korišćenog vokabulara metapodataka. RailML omogućava specifikaciju autora, datuma, izvora, identifikatora i raznih drugih informacija.

Trenutno se RailML primarno koristi kao format podataka za uvoz / izvoz za brojne sisteme za železničku simulaciju. Izuzetak je SBB-ov sistem informisanja o kupcima usluga koji omogućava upotrebu RailML-a kao ulaznog i izlaznog formata u proizvodnji. Softverski alati koji podržavaju RailML za prenos podataka uključuju simulatore kao što su OpenTrack i HER-MES (simulacijski paket koji se koristi u ON-TIME), kao i softver za vremenski bling, kao što su Viriato i OpenTimetable. Projekat railML.org podržava veliki broj evropskih železničkih operatera, proizvođača i menadžera infrastrukture;

tu su između ostalih Alstom, Deutsche Bahn, SBB, Siemens i Infrabel.

3.1.5. Podšeme voznog reda, informacije i karakteristike vozova

Podšema voznog reda RailML-a fokusirana je na opis reda vožnje vozova, uključujući sve njegove različite aspekte koji su potrebni aplikacijama za razmenu podataka. Konkretno, šema voznog reda RailML sadrži sledeće informacije:

- Periodi rada: Radni dani za usluge vozova;
- Delovi voza: Osnovni delovi voza kao niz operativnih ili kontrolnih tačaka s istim karakteristikama, kao što su formiranje i razdoblje rada. Deo voza uključuje stvarne informacije u vezi sa vozom, kao i odgovarajuće informacije o voznom redu;
- Vozovi: Jedan ili više grupa kola čine voz i predstavljaju ili operativni ili komercijalni prikaz kretanja voza;
- Operativno planiranje: Delovi vozova mogu se povezati tako da formiraju nizove neophodne za roster (raspored vagona).

Podšema železničkog voznog parka RailML-a fokusirana je na opis železničkog voznog parka, uključujući sve njegove različite aspekte za koje se smatra da su potrebne aplikacije za razmenu podataka. Konkretno, šema RailML vagoni (rollingstock) sadrži sledeće informacije:

- Vozila: karakteristike pojedinačnih železničkih kola ili serije vozila opisane su u ovom delu šeme. Opis kola uzima u obzir neke opšte podatke koji se koriste za organizovanje imovine kao što su naziv, klasifikacija ili brojevi kola koje je dao njihov operater. Veći deo šeme je konstrukcija za skladištenje različitih tehničkih aspekata železničkih kola s obzirom na njihov pogonski sistem, karakteristike karoserije kola, kočnice ili usluge instalirane u kola;
- Sastav: u ovom delu opisane su karakteristike garnitura vozova ili njegovih delova koji se sastoje od više različitih ili sličnih kola. Ova kombinacija kola koristi se za opisivanje karakteristika voza po potrebi, npr. u voznom redu. Međutim, šema ne nameće logičnu konzistentnost formacije i kola od koje je načinjena. To mora osigurati aplikacija koja proizvodi podatke.

Podšema infrastrukture železničkog sistema RailML-a fokusirana je na opis infrastrukture železničke mreže, uključujući sve njene različite aspekte koji su potrebni aplikacijama za razmenu podataka. Konkretno, železnička infrastrukturna šema sadrži sledeće informacije:

- Topologija;
- Koordinate: svi elementi železničke infrastrukture mogu se nalaziti u proizvoljnom dvo ili trodimenzionalnom koordinatnom sistemu, npr. VRG84 koji se široko koristi u današnjem softveru za navigaciju;
- Geometrija pruge može se opisati u obliku poluprečnika krivine i nagiba. Elementi železničke infrastrukture obuhvataju različita sredstva u vezi sa železnicom koja se mogu naći ispod, iznad ili pored železničke pruge, npr. ivice platforme i prelazi u nivou;
- Dalje, obuhvaćeni su elementi koji su usko povezani sa železničkom infrastrukturom, ali koji "nisu opipljivi", npr. profili, brzine i uslovi pruge;
- Šema povezivanja RailML® fokusirana je na informacije koje upravljači infrastrukturom obično održavaju u planovima signala i tabelama povezivanja ruta. Glavni korisnici ove opreme isporučuju i simulatore koji povezuju navedene elemente. Korisnici trenutno ručno kopiraju konfiguraciju podataka sa planom na papir;
- Priprema podataka je formiranje puta vožnje i sistema signalizacije za određeno polje. Greške u podacima ugrožavaju sigurnost; previše je očigledno da pogrešan aspekt signala može izazvati zastrašujuće nesreće. Zbog toga ljudi na železnici ulažu mnogo vremena i truda u testiranje. Standardni format razmene podataka omogućava automatizaciju prenosa podataka i smanjuje broj grešaka uklaňanjem ljudskog faktora iz petlje. Ovo će stvoriti više nivoa sigurnosti uz, zasigurno, niže troškove;
- Programi simulacije izračunavaju postavljanje puteva vožnje i konfiguriraju signale. Stvari poput prebacivanja signala, korišćenja veće brzine vožnje ili skraćivanja blokova mogu imati značajni uticaj. Algoritmi za simulaciju postaju sve bolji i dostižu visok nivo tačnosti tamo gde su sekunde važne. Šema RailML IL (RailML Interlocking) dozvoljava modelu

da brzo prerade informacije o sistemima za postavljanje puteva vožnje, kao što su ponašanja vremena i ruta, i analiziraju uticaj na kapacitet železnice.

Očigledan je slučaj razmene informacija koje se međusobno spajaju u zajedničkom formatu. Šema za međusobno povezivanje koristi mnoštvo podataka koji se nalaze u šemi infrastrukture.

3.2. Prenos podataka pomoću RailML

RailML je jednostavan i efikasan način da se prenesu podaci između računarskih programa, koji se koriste za modeliranje različitih aspekata železničkih operacija. Programi koji koriste RailML jezik prave izvozne datoteke sa RailML strukturom, ove datoteke mogu se zatim direktno koristiti u drugim programima. Prijemni program analizira dolazni fajl, kako bi dobio samo one podatke koji su mu potrebni, što omogućava da mnogo različitih programa koriste istu datoteku podataka. RailML pruža dva načina za razmenu podataka. Prvi se sastoji od pokretanja aplikacije odvojeno, što proizvodi izlazne datoteke koje se zatim koriste kao ulazne za druge aplikacije. Drugi pristup, trenutno u fazi razvoja, sastoji se od direktnog prenosa podataka između dve aplikacije sredstvima interprocesne komunikacije (npr. veb servisima preko TCP/IP-a), tj. bez zaobilaženja datoteke.

3.3. Struktura RailML-a

U ovom poglavlju opisana je opšta struktura RailML datoteke kroz primer šeme voznog reda. RailML je jezik zasnovan na XML-u, što znači da datoteke podataka sadrže i podatke i opise svih sadržanih podataka. Svi jezici izvedeni u XML-u koriste vrlo jednostavan i fleksibilan ASCII format naslova za svoje dokumente. U svim slučajevima dokumenti su hijerarhijski, oni grade oblik stabla, tj. svaki dokument ima jasan korenski element iz kojeg navigacija može početi da koristi opštu strukturu dokumenata (zajednička svim RailML dokumentima) kao vodič. Korenski element RailML dokumenta naziva se < railml >. Podšeme (infrastruktura, vozni park i vozni red) koje sadrže podatke u vezi sa železnicom izvode se iz ove korenske strukture. Koristeći ovaj fleksibilni pristup svaka pojedinačna aplikacija kompatibilna RailML-u određuje koje od koje tačno vrste treba koristiti podatke.

Podaci u XML dokumentima se organizuju i upravljaju pomoću elementa i atributa. Element RailML počinje početnom oznakom "< railml >", i završava se završnom oznakom "< /railml >". Element može imati atribute za detaljniji opis, a može sadržati i dodatne elemente. Pod elementom "< timetable >" navedeni su pojedinačni kursevi u elementu "< train >". Najvažniji atribut toka/pravca, broj pravca (train ID), neophodan je; ostali atributi nisu obavezni. Svaki voz koji se kreće (tok) sadrži onoliko unosa voznog reda koliko neko želi "< entry >", koji su navedeni pod elementom "< timetableentries >". Pojedinačne stavke voznog reda sadrže atribute kao što su skraćenica za stanicu (ID), vreme dolaska (arrival), vreme odlaska (departure), ili minimalno vreme zaustavljanja u stanici (min stop time).

3.4. SDEF (2009)

Format razmene podataka za signale (Signalling Data Exchange Format - SDEF) kreiran je od strane Net-work Rail-a kako bi se omogućio razvoj signalnih sistema iz zajedničkog izvora i smanjila ljudska greška nastala dupliranjem signalnih podataka iz jednog sistema u drugi [8]. Planirano je da SDEF bude sveobuhvatni infrastrukturni standard podataka, omogućavajući bilo kojem učesniku u Velikoj Britaniji železničku razmenu informacija o temama kao što su topologija pruge i lokacija opreme. SDEF definiše sledeće karakteristike sadržaja u svojoj XML šemi:

- vremenski niz,
- opisuje relevantne konfiguracije sa poštovanjem tačnog vremena na železnici. Višestruki signalni sistemi mogu postojati za isti set u različito vreme (na primer, prilikom dizajniranja nadogradnji signalizacije). Iako ovo nije potpuna reprezentacija domena tokom neisprekidanog vremena (4D), kao u drugim formatima, to bar olakšava prikaz podataka u diskretnim vremenskim okvirima, što nedostaje funkcija RailML,
- fizička geografija (topologija trasa),
- uključuje grafikone višeg nivoa povezivanja rasporeda trasa, kao i niže nivoe specifičnih informacija o lokaciji trase i fizičkim atributima,
- položaj i atributi opreme koji definišu entitete kao što su signali, brojači osovina i balize,
- specifikacija logičke interakcije,

- komunicira signalizaciju i logiku za upotrebu u šemama ponovnog signaliziranja,
- izvori obezbeđenih podataka,
- atributi koji se odnose na izvor podataka u modelu SDEF,
- prikazi izgleda.

Ovo su samo indikativni podaci. Različite aplikacije koje koriste SDEF podatke unose sopstvene prikaze na svoj način.

3.5. TAF/TAP TSI (2006/2011)

Tehnički standardi za interoperabilnost telematičkih aplikacija za teretne/putničke usluge dizajnirani su tako da omoguće razmenu informacija, koje opisuju prekogranične železničke usluge između deoničara unutar EU. Razvoj standarda je u toku, iako je TAF trenutno u mnogo naprednijoj fazi od TAP-a. Zajednički koncepti koje standard deli, uključuju identifikatore vozila, rute i redove usluga, spiskove uključenih aktera (operatera, lizing kompanija itd.) i manifestacije (broj putnika, vrste i mase tereta itd.).

3.6. RailVIVID alat za validaciju

RailVIVID® je alatka za validaciju i preglednik RailML®, čiju su prvu alfa verziju predstavili TU Dresden i railML.org na četvrtoj konferenciji UIC RailTopoModel u Parizu, 29. aprila 2015. godine.

RailVIVID je rezultat projekta usled UIC-ovog konkursa iz novembra 2014. godine, koji je finansiralo nekoliko evropskih menadžera železničke infrastrukture. TU Dresden, Institut za saobraćajnu telematiku, pobedio je u ovom pozivu i započeo početni razvoj železničkog alata RailVIVID. Kasnije je RailVIVID dodeljen RailML zajednici i ta zajednica će ga dalje razvijati.

Sa RailVIVID-om imate potpunu kontrolu nad svojim RailML datotekama: Možete vizualizovati infrastrukturu, vozni red i elemente voznog parka koji su uključeni u RailML datoteku čak i bez potpunog razumevanja sinteze KSML i railML. Integrisani RailVIVID validator omogućava vam da proverite RailML datoteku u pogledu ispravnosti sintakse, kao i nekoliko semantičkih aspekata. Dakle, RailVIVID vam pruža svu udobnost potrebnu za uspešan rad sa sučeljima za uvoz i izvoz softvera RailML i vašeg softvera.

RailVIVID sadrži:

- grafički preglednik podataka o voznom redu,
- tabelarni preglednik za podatke o voznom redu sa izvozom proračunskih tabela,
- pregled podataka vagona (Rolling Stock),
- topološki preglednik za podatke o infrastrukturi,
- geografski preglednik za podatke o infrastrukturi,
- validator šeme za RailML,
- softver na engleskom i nemačkom jeziku sa korisničkim priručnikom (moguć je prevod na druge jezike).

Detaljan opis i uputstvo za upotrebu možete pronaći u RailVIVID Viki. Više informacija, ideja za implementaciju ili ako želite da učestvujete u diskusijama o RailVIVID, pogledajte forum RailVIVID. Možete preuzeti zvanične i beta verzije RailVIVID-a i izvorni kod za vaše lične svrhe nakon prihvatanja uslova licence.

Upotreba RailVIVID-a je besplatna za registrovane članove railML.org sa svrhom validacije podataka i predstavljanja podataka RailML-a.

Ni UIC, ni railML.org, ni podizvođač niti treće strane ne preuzimaju odgovornost za gubitak podataka, gubitak prihoda, gubitak poslovnog prekida ili bilo kakve druge štete / štete koje prouzrokuje RailVIVID.

Upotreba railVIVID-a ne obavezuje railML.org za sertifikaciju softverskih interfejsa ili upotrebu zaštitnih znakova ili logotipa RailML.

RailVIVID se mora distribuirati samo putem <http://railVIVID.railML.org>. Ne nude dalje mogućnosti preuzimanja.

4. PRIMER RAILML-A

RailML zasniva se na istom načinu pisanja kao već pomenuti gore XML. RailML se služi takođe sintaksama koje se koriste kod XML-a. Iz tog razloga je prvo objašnjen XML, onda RailML, zbog načina razumevanja samog RailML koda. Za primer je dat izgled trase koja se opisuje i sam kod za tu trasu (slika 3). Na slici možete videti element `<railml version>` što predstavlja uvodni element koji je "parent" svim elementima koda. U tom elementu postoje dva atributa koja bliže opisuju sam kod, a to su verzija i lokacija. Verzija koda je bitna iz razloga da kada drugi developer pristupi kodu, jasno zna u kojoj verziji je kod pisan [9]. Što se tiče samog koda, kod je podeljen na dva dela, prvi deo čini "glava" koda u kome su smešteni neki osnovni podaci o samom RailML-u, a u drugom delu zvanom "telo" nalazi se kod RailML-a. Podaci koji se nalaze u glavi koda su podaci o verziji RailML-a, format dokumenta, vreme i datum, naslov dokumenta, jezik, kreator dokumenta, objašnjenje o dokumentu, obezbeđenje prava.

Kao što smo rekli u XML poglavlju, ceo kod predstavlja jedno "stablo", koje se sastoji od "roditelja" i "dece". U ovom primeru "roditelj" celog koda je element pod nazivom `<infrastructure>` koji sadrži atribut čija vrednost je ime ili naziv koji dodeljujemo trasi, u ovom slučaju to je "inf01" (slika 4). Element

```
<?xml version="1.0" encoding="UTF-8"?>
<railml version="2.3" xsi:schemaLocation="http://www.railml.org/schemas/2016
http://www.railml.org/schemas/2016/railML-2.3/schema/railML.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns="http://www.railml.org/schemas/2016">
  <metadata>
    <dc:format>2.3</dc:format>
    <dc:identifier>1</dc:identifier>
    <dc:source>railML.org</dc:source>
    <dc:title xml:lang="en">Simple Example railML 2.4 (infrastructure)</dc:title>
    <dc:language>en</dc:language>
    <dc:date>2018-09-17T09:57:00+01:00</dc:date>
    <dc:creator xml:lang="de">Christian Rahmig</dc:creator>
    <dc:description>This example file has been coded manually and therefore may not be free of
      errors.</dc:description>
    <dc:rights>Copyright (c) railML.org, Dresden, Germany. All Rights Reserved. For more information, visit
      https://www.railml.org/en/user/licence.html. Content of this file: railML 2.3 Simple Example</dc:rights>
  </metadata>
  <infrastructure id="inf01">
    <tracks>
    <trackGroups>
    <operationControlPoints>
  </infrastructure>
</railml>
```

Slika 3. Izgled stabla koda "glave" i "tela"

<infrastructure> je sačinjen od tri elementa koja predstavljaju njegovu "decu" [10].

```
- <infrastructure id="inf01">
  + <tracks>
  + <trackGroups>
  + <operationControlPoints>
</infrastructure>
```

Slika 4. "Roditelj" elementima trase

Prvi element je <tracks>, koji predstavlja skup koloseka i elemenata koji ih opisuju. U ovom slučaju imamo sedam koloseka, svaki kolosek ima atribute kojima je dodeljeno ime koloseka, odnosno redni broj koloseka. Pored toga atributom "type" je dodeljena vrednost-važnost koloseka U ovom slučaju imamo tri glavna koloseka, tri pomoćna i jedan kolosek sa strane. Sada ćemo objasniti samo jedan kolosek, pošto ih ima sedam (slika 5).

```
- <tracks>
  + <track id="tr01" mainDir="none" type="secondaryTrack" name="track a02">
  + <track id="tr02" mainDir="none" type="mainTrack" name="track a01">
  + <track id="tr03" mainDir="none" type="mainTrack" name="track a03 x01 b03">
  + <track id="tr04" mainDir="none" type="sidingTrack" name="track b05">
  + <track id="tr05" mainDir="none" type="secondaryTrack" name="track b04">
  + <track id="tr06" mainDir="none" type="secondaryTrack" name="track b02">
  + <track id="tr07" mainDir="none" type="mainTrack" name="track b01">
</tracks>
```

Slika 5. Broj koloseka u šemi

Svaki kolosek u kodu sadrži elemente koje ga bliže opisuju <trackTopology >, <TrackElement >, <oscElements >. Prvi element <trackTopology > topologiju koloseka. U to spadaju elementi u vezi sa početkom i krajem koloseka koji se obeležavaju elementima <trackBegin>, <trackEnd> i element <crossSection > koji predstavlja stanicu (slika 6).

```
- <trackTopology>
  - <trackBegin id="tr01_tb" pos="0">
    <bufferStop id="tr01_bs01"/>
  </trackBegin>
  - <trackEnd id="tr01_te" pos="500">
    <connection id="tr01_c01" ref="tr03_c03"/>
  </trackEnd>
  - <crossSections>
    <crossSection xml:lang="de" id="tr01_cs01" type="station" name="Gleis 1" pos="300.0" absPos="300.0" ocpRef="ocp02"/>
  </crossSections>
</trackTopology>
```

Slika 6. Toplogija koloseka

Drugi element <TrackElement > opisuje elemente trase, u ovom slučaju to je peron sa podacima gde se

on nalazi, njegovoj visini, dužini i položaju u odnosu na kolosek (slika 7).

```
- <trackElements>
  - <platformEdges>
    <platformEdge xml:lang="de" dir="up" id="tr01_pe01" name="Gleis 2" pos="200" absPos="200" length="200" height="550" side="left"/>
  </platformEdges>
</trackElements>
```

Slika 7. Elementi koloseka

Treći Element <oscElements > daje informacije u vezi sa signalom na koloseku i voznim detektorima. Što se tiče signala, atributima se daju informacije o poziciji, njihova funkcija i tip signala. Isto tako i za detekciju vozova, mogu se videti osnovne informacije o signalu (slika 8).

```
- <ocsElements>
  - <signals>
    - <signal dir="up" code="68N1" id="tr01_si01" type="main" pos="450" absPos="450" ocpStationRef="ocp02" function="exit">
      <etcs level_2="true" switchable="false"/>
    </signal>
  </signals>
  - <trainDetectionElements>
    <trainDetector id="tr01_td01" pos="475" absPos="475" medium="inductive" directionDetection="true" axleCounting="true"/>
  </trainDetectionElements>
</ocsElements>
```

Slika 8. Informacije o signalima na trasi

Sledeći element je <trackGropus >. Element <trackGropus> predstavlja koloseke udružene u grupe-objekte, u ovom slučaju to su glavni koloseci kojima je dodeljen atribut sequence koji predstavlja redni broj objekta.

Sledeći element je <operationControlPoints>. U ovom elementu date su informacije da li se radi o stanici ili ne i za koje operacije je stanica predviđena, kao i tipu stanica, tipu osiguranja. Predstavlja se kroz skup niza elemenata i njegovih atributa.

Prvi atribut je operationalType operativna funkcija date ocp (kontrolne tačke ili mesta). Moguće opcije:

- station (klasična stanica za polaske, dolaske, prolaske vozova bez zaustavljanja),
- stoppingPoint Stop (signalna tačka bez signala, skretnica i nezavisni koloseci gde vozovi mogu polaziti i dolaziti),
- depot (depo),
- crossover (povezanost između dva koloseka ili duplih koloseka),

- junction (razdvajanje dva koloseka),
- blockPost (signalna tačka ili mesto sa signalima gde je vozni prostor ekranizovan),
- blockSignal (blok signal za vozni prostor sa automatskim upravljanjem),
- siding (kontrolne tačke za skupljanje cargo dostavnih vozova, ali ne i za vozne operacije),

Drugi atribut je trafficType koji predstavlja za koju vrstu prevoza je stanica. Moguće opcije:

- passenger (putničke),
- freight (teretne),
- shunting (manevarske),
- operationla (operativne).

Treći atribut je orderChangeable i postoje dve mogućnosti true i false. Ako je izabrana vrednost true, onda signalna tačka ima mogućnost promene niza vozova na otvorenom koloseku.

Poslednji atribut ensuresTrainSequence takođe ima dve vrednosti, a to su true i false. Ako je true, signalna tačka obezbeđuje i osigurava niz vozova na otvorenom koloseku. Drugim rečima, ako je signalna tačka zauzeta vozom, sledeći voz može sigurno da priđe signalnoj tački. Vozni niz je osiguran i to se pretežno obavlja signalima (blok i glavnim signalima).

To su osnovne informacije, a pored njih mogu se saznati i druge. U našem slučaju obe stanice su putničke. U ovom elementu nalaze se informacije u vezi sa signalima tih stanica, kao i koordinate i položaj stanica. Svaki element poseduje svoje attribute, koje opisuju sam kolosek i stanice.

```
<propOperational trafficType="passenger"
operationalType="station"
orderChangeable="true"
ensuresTrainSequence="true"/>
```

U ovom elementu je opisan tip stanice, u ovom slučaju je to putnička. Sledeći blok element

```
<summary signalBox="mechanical"
hasSwitches="true"
hasStarterSignals="true"
hasHomeSignals="true"/>
```

opisuje tip signala, način prebacivanja signala, da li ima ulazni signal i da li postoji glavni signal.

Poslednji element je prikaz geografskog položaja (slika 9).

```
- <operationControlPoints>
- <ocp xml:lang="de" code="CS" id="ocp01" type="operationalName"
name="Bf Cstadt">
<additionalName xml:lang="cz" type="localName"
name="Bouzov"/>
<propOperational trafficType="passenger" operationalType="station"
orderChangeable="true" ensuresTrainSequence="true"/>
- <propEquipment>
<summary signalBox="mechanical" hasSwitches="true"
hasStarterSignals="true" hasHomeSignals="true"/>
</propEquipment>
<geoCoord epsgCode="urn:ogc:def:crs:EPSG::4326"
coord="51.0000 16.0000"/>
<designator entry="OCS" register="RL100"/>
</ocp>
- <ocp xml:lang="de" code="AR" id="ocp02" type="operationalName"
name="Bf Arnau">
<additionalName xml:lang="cz" type="localName"
name="Adamov"/>
<propOperational operationalType="station"/>
<propService goodsSiding="true" service="true" passenger="true"/>
- <propEquipment>
<trackRef ref="tr01"/>
<trackRef ref="tr02"/>
</propEquipment>
<geoCoord epsgCode="urn:ogc:def:crs:EPSG::4326"
coord="52.0000 15.0000"/>
<designator entry="OAR" register="RL100"/>
</ocp>
</operationControlPoints>
```

Slika 9. Tipovi stanica, geografski položaj

5. ZAKLJUČAK

Cilj RailML projekta je da pojednostavi prenos podataka između rastućeg broja softverskih sistema, koji se koriste na železnici pomoću upotrebe zajedničkih struktura podataka. Ovo je izuzetno korisno jer omogućava projektantima železnica i operaterima lako korišćenje nekoliko različitih vrsta računarskih aplikacija, omogućavajući da svaki radi ono što najbolje ume. U prošlosti specijalizovani programi interfejsa su bili razvijeni za prenos podataka između aplikacija, ali to se ispostavilo kao neefikasan i dugotrajan proces. Pored toga, kako se broj aplikacija povećava, razvoj i održavanje ovih pojedinačnih interfejsa postaje ne tako praktičan. Već sada, nekoliko različitih aplikacija za železnice uključuju interfejse koji koriste RailML format za unos i izlaz podataka u vezi sa rasporedom između aplikacija. Ovi programi uključuju: FBS, Viriato, OpenTrack i OpenTimeTable.

LITERATURA:

[1] Radonjić, V., S. Janković, S. Mladenović, S. Vesko-
vić and A. Kostić-Ljubisavljević, "B2B Integration
of Rail Transport Systems in Cloud Computing
Environment", ISABEL 20[1, 4th International
Symposium on Applied Sciences in Biomedical
and Communication Technologies, Barcelona,
Spain, Oct 25–29, 2011, Proceedings of the

- Symposium, doi: 10.1145/2093698.2093833, Article 135, ISBN: 978-1-4503-0913-4.
- [2] Branović, I., S. Vesković, S. Mladenović, S. Milinković and S. Janković "SOA Architecture for Complying with EU Railway Timetable Data Exchange Format", TELSIXS '11, 10th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services, Niš, Serbia, Oct 5-8, 2011, Proceedings of Papers - Volume 2 of 2 volumes, pp. 630-631, ISBN: 978-1-4577-2016-1/11/\$26.00 ©2011 IEEE.
- [3] Hoffmann, R., Krauss, V. P., 2001, RailML als Datenaustauschformat für Eisenbahnsoftware (Projektbeschreibungen und-berichte sowie Konzepte, Fraunhofer Gesellschaft. V., Institut für Verkehrs- und Infrastruktur-systeme, Dresden)
- [4] Ostojić, Miloš, Railml standard za razmenu podataka u železničkom saobraćaju, Završni rad, Univerzitet u Beogradu – Saobraćajni fakultet, 2020.
- [5] Fries, N., 2003, Modellierung und Abbildung einer Eisenbahn-Infrastruktur auf ein Schema der XML-basierten Schnittstelle RailML (Diplomarbeit, TU Dresden, Professur für Verkehrssicherungstechnik, Dresden).
- [6] Montigel, M., 2002, Time-trigger edex change of train route data between train control systems, in Computer in Railways, edited by J. Allan, R. J. Hill, C. A. Brebbia, G. Scitutto & S. Sone (WIT Press, Southampton), pp. 33-41.
- [7] Mladenović, S., S. Vesković, I. Branović, S. Janković and S. Aćimović, "Heuristic Based Real-Time Train Rescheduling System", Networks, Vol. 67, Issue 1, 2016, pp. 32-48, ISSN: 0028-3045, <https://doi.org/10.1002/net.21625>.
- [8] Andrew Nash, Daniel Huerlimann, Prof. Joerg Schuette and Vasco Paul Krauss, RailML – A standard data interface for railroad applications, Computers in Railways IX, J. Allan, C. A. Brebbia, R. J. Hill, G. Scitutto & S. Sone (Editors), © 2004 WIT Press, www.witpress.com, ISBN 1-85312-715-9, COMPRAIL 2004, Dresden, Germany
- [9] Hengartner, M., 2003, Grafikeditor für RailML-basierte Infrastrukturdaten (Diplomarbeit, ETH Zürich, Institut für Verkehrsplanung und Transportsysteme, Zürich).
- [10] Hürlimann, D., 2002, Objektorientierte Modellierung von Infrastrukturelementen und Betriebsvorgängen im Eisenbahnwesen, 2002, Institut für Verkehrsplanung und Transportsysteme, ETH Zürich, Schriftenreihe des IVT, Nr. 125.
- [11] <https://www.railml.org/en/user/example-data.html>
- [12] <https://en.wikipedia.org/wiki/XML>
- [13] <http://www.ontime-project.eu/>
- [14] <https://www.railml.org/index.php/en/>
- [15] https://wiki2.railml.org/index.php?title=Main_Page