



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA DOPRAVNÍ

VÍT ŘEZÁČ
MOŽNOSTI APLIKACE VÍCEÚČELOVÉHO MODELU
ŽELEZNIČNÍ INFRASTRUKTURY

BAKALÁŘSKÁ PRÁCE

2023



K620..... Ústav dopravní telematiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE (PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

Vít Řezáč

Studijní program (obor/specializace) studenta:

bakalářský – ITS – Inteligentní dopravní systémy

Název tématu (česky): **Možnosti aplikace Víceúčelového modelu železniční infrastruktury**

Název tématu (anglicky): Possibilities of Multipurpose Railway Infrastructure Model Application

Zásady pro vypracování

Při zpracování bakalářské práce se řiďte následujícími pokyny:

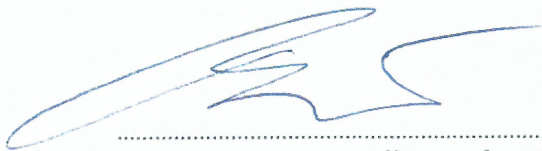
- Seznamte se s Víceúčelovým modelem železniční infrastruktury včetně vybraných stávajících specifických rozšíření a existujícími nástroji pro práci s ním.
- Posud'te možnosti aplikace Víceúčelového modelu pro potřeby popisu kolejíšť Dopravního sálu Fakulty dopravní Českého vysokého učení technického v Praze
- Navrhňte takové specifické rozšíření Víceúčelového modelu, které bude účelům popisu výše uvedeného kolejíšť vyhovovat.
- Navrhňte metodu naplňování Víceúčelového modelu daty, která umožní vybrané aspekty infrastruktury výše uvedeného kolejíšť s využitím struktury modelu popsat.
- Ověřte funkčnost navržené metody vytvořením datového popisu části výše uvedeného kolejíšť.

- Rozsah grafických prací: Dle pokynů vedoucího práce.
- Rozsah průvodní zprávy: minimálně 35 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)
- Seznam odborné literatury: INTERNATIONÁL UNIÓN OF RAILWAYS (UIC). IRS 30100: RailTopoModel Railway Infrastructure Topological Model (v 1.1). Paris: International Union of Railways (UIC), 2017.
HLUBUČEK, Adam. Víceúčelový model železniční infrastruktury (v. 12.2). Praha, 2022.

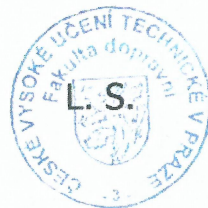
Vedoucí bakalářské práce: **doc. Ing. Martin Leso Ph. D.**

Datum zadání bakalářské práce: **7. října 2022**
(datum prvního zadání této práce, které musí být nejpozději 10 měsíců před datem prvního předpokládaného odevzdání této práce vyplývajícího ze standardní doby studia)

Datum odevzdání bakalářské práce: **7. srpna 2023**
a) datum prvního předpokládaného odevzdání práce vyplývající ze standardní doby studia a z doporučeného časového plánu studia
b) v případě odkladu odevzdání práce následující datum odevzdání práce vyplývající z doporučeného časového plánu studia

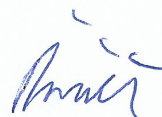


Ing. Zuzana Bělinová, Ph.D.
vedoucí
Ústavu dopravní telematiky



prof. Ing. Ondřej Příbyl, Ph.D.
děkan fakulty

Potvrzuji převzetí zadání bakalářské práce.



Vít Řezáč
jméno a podpis studenta

V Praze dne 7. října 2022

Poděkování

Chtěl bych poděkovat zvláště panu doc. Ing. Martinu Lesovi Ph.D., vedoucímu této práce, za jeho vedení a poskytnuté konzultace. Dále by autor chtěl poděkovat panu Ing. Adamu Hlubučkovi, autoru víceúčelového modelu železniční infrastruktury, za poskytnuté materiály, konzultace a zpětnou vazbu k návrhu. Nakonec by autor chtěl poděkovat své rodině a blízkým, kteří jej při tvorbě této práce i předchozím studiu podporovali.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Nemám závažný důvod proti užívání tohoto školního díla ve smyslu § 60 zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů.

V Praze dne 6. 8. 2023



.....
Podpis

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA DOPRAVNÍ

MOŽNOSTI APLIKACE VÍCEÚČELOVÉHO MODELU ŽELEZNIČNÍ
INFRASTRUKTURY

Bakalářská práce

6. 8. 2023

Vít Řezáč

ABSTRAKT

Práce popisuje Víceúčelový model železniční infrastruktury v rozsahu znalostí potřebných pro uživatele resp. vývojáře specifických rozšíření pro tento model. Dále popisuje implementaci specifického rozšíření pro potřeby dopravního sálu FD ČVUT v Praze a metodu tvorby modelu v rámci tohoto rozšíření vč. tvorby podkladů. Práce dále obsahuje informace potřebné pro případnou údržbu a další rozšíření navrženého specifického rozšíření.

ABSTRACT

This bachelors thesis describes Multipurpose model of railway infrastructure in level of detail required by user or developer od specific extensions designed for this model. This thesis also describes implementation of specific extension designed for use by railway simulation laboratory of Faculty of transportation sciences, CTU in Prague. Furthermore, this thesis also contains information necessary for possible maintenance of designed specific extension.

KLÍČOVÁ SLOVA

Víceúčelový model železniční infrastruktury, datový popis infrastruktury, RailTopoModel, railML, integrální zabezpečovací zařízení, radiobloková centrála, Železnice 4.0

KEYWORDS

Multipurpose model of railway infrastructure, data-centric description of infrastructure, RailTopoModel, railML, integral railway interlocking, radio block control, Railway 4.0

Obsah

Seznam použitých zkratk	6
1 Úvod	7
2 Cíle práce	8
3 Víceúčelový model železniční infrastruktury	9
3.1 Struktura modelu	9
3.2 Vztah mezi VMŽI a RailTopoModelem	11
3.3 Specifické rozšíření	12
3.3.1 Existující specifická rozšíření pro VMŽI	13
3.4 Aplikace VMŽI na DSFD	15
3.4.1 Nový software řízení modelových souprav	15
3.4.2 Integrovaná zabezpečovací zařízení pro DSFD	16
3.4.3 Tvorba 3D prostředí pro potřeby simulátorů na DSFD	19
3.4.4 Pasportizace kolejí DSFD	20
4 Návrh specifického rozšíření	23
4.1 Potřebné úkony pro vytvoření specifického rozšíření v rámci tvorby nástroje pro tvorbu VMŽI	25
4.2 Úpravy již existujících částí nástroje pro tvorbu VMŽI implementované pro potřeby specifického rozšíření pro DSFD	27
4.3 Popis řešení výhybek ve specifickém rozšíření pro DSFD	28
4.3.1 Popis jednoduché výhybky	29
4.3.2 Popis obloukové výhybky	29
4.3.3 Popis křižovatkové výhybky	30
4.3.4 Popis kolejového křížení	31
4.4 Popis řešení návěstidel ve specifickém rozšíření pro DSFD	32
4.5 Funkce vytvořené v rámci usnadnění tvorby modelu v rámci specifického rozšíření	33
5 Metoda získání dat pro potřeby modelu a jejich vkládání do modelu	35

5.1	Sběr dat o kolejišti DSFD	35
5.1.1	Měření a popis geometrie kolejiště	35
5.1.2	Sběr dat o prvcích infrastruktury	37
5.1.3	Zaměření prvků infrastruktury do připravené geometrie v prostředí běžného editoru CAD	38
5.1.4	Kontrola správnosti získaných informací	39
5.2	Metoda vkládání dat do VMŽI	39
6	Ověření funkčnosti tvorby modelu	41
6.1	Výstupy ověření	43
7	Závěr	45
	Reference	47
	Seznam obrázků	49
	Seznam příloh	50

Seznam použitých zkratk

- ATO - automatic train operation - automatické vedení vlaku
- CAD - Computer Aided Design - počítačem podporované projektování
- DSFD - Dopravní sál Fakulty dopravní
- ETCS - European Train Control System - evropský vlakový zabezpečovač
- FD - Fakulta dopravní Českého vysokého učení technického v Praze
- IZZ - integrální zabezpečovací zařízení
- RBC - Radio Block Centre - radiobloková centrála
- SR - specifické rozšíření
- SW - software
- UIC - Union Internationale des Chemins de fer - Mezinárodní železniční unie
- VMŽI - Víceúčelový model železniční infrastruktury

1 Úvod

Provoz a budoucí rozvoj Dopravního sálu Fakulty dopravní Českého vysokého učení technického v Praze je závislý jen na samotném vybavení, ale i na kvalitě a využitelnosti podkladů mapujících infrastrukturu kolejíště sálu. Současná dokumentace kolejíště je tvořena nesourodou kombinací textových dokumentů a výkresů s omezenou možností automatizovaného zpracování. Dále pak všechny již existující řídicí SW DSFD využívá uzavřený, proprietární model infrastruktury omezující využitelnost těchto dat jinými aplikacemi, čímž jsou znatelně omezeny možnosti budoucího rozvoje dopravního sálu, vč. projektů na něm realizovaných. Tyto projekty, často směřované k ověření koncepce **Železnice 4.0**[3], vyžadují v mnoha případech standardizovaný a strojově zpracovatelný model infrastruktury popisující všechny její důležité aspekty.

V rámci jednoho z projektů na DSFD vznikl Víceúčelový model železniční infrastruktury, modulární nástroj pro tvorbu počítačově zpracovatelného popisu infrastruktury vycházející z principů a struktury RailTopoModelu UIC[1]. Modularita tohoto modelu, založená na úpravě vlastností modelu prostřednictvím specifických rozšíření, datových struktur navázaných na jádro modelu a obsluhovaných vlastními částmi nástroje pro tvorbu tohoto modelu, umožňuje popsat železniční infrastrukturu pro projekty s často zcela odlišnými požadavky na popis této infrastruktury, přičemž však všechny moduly využívají společné jádro a principy popisu.

Tento model však nebyl doposud na popis kolejíště DSFD využit. V rámci této práce proběhne seznámení se s vlastnostmi a strukturou tohoto modelu včetně již existujících specifických rozšíření, ověření vybraných možností aplikace VMŽI na řešení popisu infrastruktury kolejíště pro potřeby konkrétních projektů plánovaných na DSFD, definice požadavků na specifické rozšíření přizpůsobující VMŽI potřebám popisu kolejíště DSFD pro vybrané projekty a následná tvorba tohoto rozšíření a doplnění nástrojů pro jeho obsluhu do nástroje pro tvorbu VMŽI. Dále dojde k navržení metody tvorby VMŽI s navrženým specifickým rozšířením a závěrem dojde i k ověření funkčnosti této metody i navrženého rozšíření vč. doplněných nástrojů pro tvorbu VMŽI s daným SR.

2 Cíle práce

Cílem této bakalářské práce je seznámení se s víceúčelovým modelem železniční infrastruktury, jeho základní vnitřní strukturou, způsoby jeho tvorby a nástroji pro jeho tvorbu vč. znalostí zdrojového kódu a programovacího jazyka potřebného pro jeho úpravu a tvorbu specifického rozšíření pro potřeby DSFD a toto specifické rozšíření vytvořit. Dalšími dílčími cíli práce je seznámení se s již existujícími specifickými rozšířeními vytvořenými pro jiné projekty tvořené ve VMŽI a možností využití jejich částí v nově vytvářeném specifickém rozšíření tvořeném pro DSFD. Před samotným zahájením návrhu a tvorby SR je cílem prověřit možnosti aplikace VMŽI na DSFD se zvláštním zřetelem na využití v projektu **Železnice 4.0** [3], vybrat vhodné způsoby využití a přizpůsobit toto navrhované specifické rozšíření těmto možnostem využití. Hlavním cílem práce je vytvoření specifického rozšíření a doplnění nástrojů pro jeho obsluhu v rámci nástroje pro tvorbu VMŽI a také popis navrhovaného rozšíření v rozsahu potřebném pro budoucí údržbu navrhovaného SR pro případ změn v jádře VMŽI a další informace potřebné pro tvorbu VMŽI kolejiště DSFD s pomocí daného SR. V rámci tohoto popisu je vytvořen seznam funkcí vytvořených v rámci tvorby specifického rozšíření obsahující vstupní a výstupní proměnné těchto funkcí, základní popis jejich vnitřního fungování, možné problémy vznikající během jejich běhu a jiné funkce využití v rámci tvořené funkce. Závěrem má být provedeno ověření navrhovaného SR vytvořením datového modelu části kolejiště DSFD. Cílem práce není samotná tvorba celkového modelu kolejiště DSFD v rozsahu potřebném pro využití navrhovanými způsoby ani návrh konkrétního navázání VMŽI do těchto systémů.

Seznam cílů bakalářské práce:

- Seznámení se s víceúčelovým modelem železniční infrastruktury, nástroji pro jeho tvorbu a již navrženými specifickými rozšířeními pro VMŽI.
- Posouzení možností užití víceúčelového modelu na dopravním sále FD s přihlédnutím k dlouhodobým potřebám DSFD a projektu **Železnice 4.0** [3] na DSFD.
- Provedení návrhu specifického rozšíření na základě seznámení se s VMŽI, existujícími specifickými rozšířeními a s ohledem na potřeby DSFD.
- Návrh metody plnění modelu sálu DSFD vytvořeného ve VMŽI s použitím navrženého SR s využitím existujících nástrojů.
- Ověření funkčnosti navržené metody pomocí vytvoření popisu vybrané části kolejiště DSFD ve SR navrženém v rámci této práce.

3 Víceúčelový model železniční infrastruktury

Víceúčelový model železniční infrastruktury je datový model strukturálně vycházející z doporučení UIC RailTopoModel [1]. VMŽI vznikl jako *výstup projektu Vnitřní soutěže IP2019 úkolu Implementace moderních přístupů k popisu železniční infrastruktury do výuky na pracovišti Dopravního sálu Fakulty dopravní ČVUT v Praze.* [2] Cílem VMŽI je umožnit popis železniční infrastruktury jednotným přístupem, avšak s možností úpravy vybraných aspektů podle potřeb konkrétního projektu, čemuž je přizpůsobena i struktura modelu, která je navržena jako modulární. Základem datového modelu VMŽI je relační databáze ukládající informace o popisované infrastruktuře formou relacemi navzájem propojených záznamů do tabulek modelu. Každý primární objekt v rámci VMŽI je identifikován svým unikátním ID, které je zaneseno a uchováváno ve třídě **BaseObject**, od níž jsou odvozeny všechny další terminální třídy objektů. Zpětné získání dat z modelu je umožněno formou SQL dotazů. [2]

Popis sítě v rámci VMŽI je (při využití současné verze nástroje pro jeho tvorbu) podporován ve třech rozlišovacích úrovních: **mikro**, **mezo** a **makro**. Úroveň **mikro** popisuje infrastrukturu v rozlišovací úrovni jednotlivých kolejí, výhybek a dalších prvků infrastruktury. V případě úrovně **mikro** lze též zvolit, zdali se jedná o zobrazení schématické, při němž je zaznamenáván pouze topologický popis infrastruktury a není vyžadována návaznost jednotlivých prvků na sebe, nebo realistické, kde je vedle topologického zaznamenáván i realistický popis vč. návaznosti prvků. [2] Úroveň **mezo** popisuje infrastrukturu v úrovni jednotlivých dopraven, popř. je-li to nutné částí dopraven, propojených traťovými úseky. Úroveň **makro** je nejméně podrobná úroveň zobrazující infrastrukturu v úrovni uzlů a dalších významných bodů spojených jednotlivými tratěmi, přičemž v této úrovni jsou mezilehlé dopravní bez vysokého významu zpravidla vynechány. Schématické zobrazení v úrovni **makro** lze přirovnat k plánu železniční sítě z knižního jízdního řádu. V rámci RailTopoModelu i VMŽI lze definovat i úroveň **nano**, která zobrazuje infrastrukturu v úrovni přesné topologické struktury jednotlivých prvků infrastruktury, např. body spojení a křížení v rámci křižovatkové výhybky. [7] Návrh SR pro potřeby DSFD a nástroje pro tvorbu modelu v něm byly navrženy tak, aby minimalizovaly nutnost práce v úrovni **nano** a ve většině případů umožňovaly práci přímo v úrovni **mikro**.

3.1 Struktura modelu

Model lze rozdělit na 2 základní moduly: jádro, tvořící společný základ pro všechny projekty bez ohledu na jejich potřeby a zajišťující základní funkcionalitu modelu (blíže popsáno dále), a specifická rozšíření umožňující přizpůsobit model potřebám daného projektu (pro bližší popis viz kap. 3.3). Strukturální popis VMŽI uvedený v následujících částech je z důvodu cíle a charakteru práce realizován v úrovni vyžadovaném uživatelem, popř. tvůrcem běžného specifického rozšíření a není všeobjímající.

Jádro lze dále dělit na tematické bloky myšlenkově vycházející ze struktury RailTopoModel UIC[1]. Jednotlivé tabulky jádra jsou rozděleny podle funkce a navázání na zbylé části modelu do těchto bloků bez přesahů mezi bloky. Funkcemi jádra jsou tvorba a uchovávaní vybraných informací o objektech, které jsou neměnné bez ohledu na potřeby jednotlivých projektů, např. unikátní ID objektu, které je zaneseno do tabulky **BaseObject**, a které identifikuje daný objekt ve všech dalších strukturách VMŽI. [2] Dále jádro zajišťuje funkce tvorby souřadnicových systémů, funkce lokalizace objektů, tvorby

základních struktur topologického popisu, které mohou být dále rozvinuty specifickými rozšířeními, funkce tvorby sítě z jednotlivých prvků na různých úrovních a jejich transformace mezi sebou. Konečně, funkcí jádra je i společná základní struktura **NetEntity** a funkce jejich propojení, od níž jsou dále odvozeny jednotlivé modulární entity (**ModularEntity**), popisující jednotlivé konkrétní prvky infrastruktury specifické pro každé specifické rozšíření. Struktura jádra v době dokončení specifického rozšíření pro DSFD je uvedena v Příloze 1 v Obr.: 1.

Tematický blok **Base** tvoří základ všech dalších struktur VMŽI a tvoří jej dvě tabulky: **BaseObject** uchovávající unikátní ID každého objektu ve VMŽI a **NamedResource** uchovávající dvojici parametrů pojmenovávající každý objekt pomocí úplného, zpravidla pro koncového uživatele přívětivějšího jména (longname) a krátkého jména. Záznamy v tabulkách reprezentující tyto třídy nemohou existovat samostatně, ale vznikají pro každý objekt vytvořený jinou, typicky terminální, třídou. [2]

Tematický blok **Network** zajišťuje popis sítě a umožňuje definovat jednotlivé úrovně popisu sítě prostřednictvím třídy **NetworkLevel**, vytvořit samotnou síť, do níž budou zařazovány jednotlivé prvky této sítě a tabulky realizující potřebná přiřazení síťových vlastností popsaných ve třídě **NetworkLevel** této sítě. V rámci třídy **NetworkLevel** lze popsat vedle již zmíněné úrovně sítě i zdali se jedná o schématickou, nebo realistickou reprezentaci sítě a dále dimenzionalitu dané sítě a s tím spojenou souřadnicovou soustavu. V tomto tematickém bloku je též třída **NetworkResource**, která je základem pro všechny další objekty na dané síti. [2]

Dalším tematickým blokem je blok **Positioning System** zajišťující tvorbu souřadnicového systému a přiřazení **GeoPoint** danému souřadnicovému systému (Třída **GeoPoint** je součástí tematického bloku **Location**, viz níže.). Souřadnicový systém může být realizován buď lineární, kdy je pozice prvku infrastruktury vztažena vůči již existujícímu lineárnímu prvku sítě, typicky ose koleje, a pozice daného prvku je určena kombinací ID prvku, vůči kterému je pozice vztažena, informace, jestli je pozice vztažena vůči počátku nebo konci orientované křivky reprezentující lineární prvek a relativní polohu daného prvku na tomto lineárním elementu. Druhou metodou definice pozice **GeoPoint** je definice pomocí kartézské nebo sférické soustavy souřadnic. [2]

Blok **Location** tvoří mechanismus pro samotnou lokalizaci prvků infrastruktury v rámci dané sítě. K tomuto využívá třídu **GeoPoint**, která je základem pro další lokalizaci objektů prostřednictvím dalších tříd nazvaných v závislosti na druhu zvoleného souřadnicového systému pro lokalizaci daného objektu buď **AssociatedLocation** (pozice vztažena vůči existujícímu lineárnímu elementu), nebo **GeoLocation** (pozice v kartézské nebo sférické soustavě souřadnic). Finální přiřazení skutečné lokace probíhá pomocí lokalizačních entit (**AssociatedLocationEntity**, **GeoLocationEntity**), v rámci nichž lze určit horizontální i vertikální stranu a vzdálenost vůči lokaci, směr působnosti lokalizované entity, přiřazení a rozdělení funkční a fyzické entity (např. pro potřeby stožárových návštěvidel, kdy se funkční lokalizace nachází na daném lineárním prvku představující osu koleje, ale fyzická pozice návštěvidla je několik metrů od bodu funkční lokalizace).

Tematický blok **Topology** zajišťuje tvorbu topologického popisu infrastruktury prostřednictvím tříd **LinearElement** a **NonLinearElement**, dále zajišťuje jejich propojení prostřednictvím třídy **PositionedRelation**. Blok **Topology** poskytuje též aparát potřebný k tvorbě kolekcí využitelných např. při meziúrovňové transformaci. Třída **LinearElement** reprezentuje jednotlivé liniové síťové prvky. Liniové prvky jsou ve VMŽI vždy orientované [2] a mají stanovenou délku. Na liniových elementech jsou definovány relativní

souřadnice (0 = počátek elementu, 1 = konec elementu)[12][2]. Relativní souřadnici x na elementu lze snadno vypočíst dle vztahů (1), (2) a (3).

$$x = \frac{l_e}{l_c} = \frac{s_e - s_p}{s_k - s_p} \quad (1)$$

$$x = 1 - \frac{l_c - l_e}{l_c} = 1 - \frac{s_k - s_e}{s_k - s_p} \quad (2)$$

$$x = \frac{l_e}{l_c} + x_r = \frac{s_e - s_r}{s_k - s_p} + x_r \quad (3)$$

kde l_e je vzdálenost umístění entity od referenčního bodu, l_c je celková délka elementu, s_e je poloha bodu umístění entity, s_p je poloha počátku lineárního elementu, s_k je poloha konce lineárního elementu, s_r je poloha referenčního bodu a x_r je relativní vzdálenost referenčního bodu od počátku lineárního elementu. Vztah (1) je určen pro výpočet relativní pozice prvku vůči počátku elementu, (2) vůči konci elementu (přičemž všechny hodnoty jsou dosazovány kladné bez ohledu na směr) a vztah (3) je určen pro výpočet polohy entity vůči obecnému bodu na elementu, přičemž nachází-li se entita před referenčním bodem, je nutno dosadit l_e záporně. **NonLinearElement** slouží k popisu nelineárních prvků infrastruktury jako jsou výhybky, kolejová křížení apod., a nemají na sobě definovaný souřadnicový systém podobný **LinearElementu**. [2]

PositionedRelation je další důležitou terminální třídou tematického bloku **Topology**, protože propojuje jednotlivé instance třídy **NetElement** mezi sebou a umožňuje určit průjezdnost daného spojení způsobem [2] uvedeným níže. Uvedené hodnoty vychází z hodnot parametru navigability třídy **NetRelations** v jazyku railML [8], přičemž hodnoty tohoto parametru v jazyku railML vychází z požadavků popisu navigability v RailTopoModelu [9]:

- ab - prvek je sjízdný pouze směrem z prvního prvku v rámci **PositionedRelation** do druhého prvku
- ba - prvek je sjízdný pouze směrem z druhého prvku v rámci **PositionedRelation** do prvního prvku
- both - prvek je sjízdný oběma směry
- none - propojené prvky nejsou sjízdné ani v jednom směru

Směr none se používá např. u výhybkových konstrukcí, kde označuje nemožnost jízdy ze směru po hrotu do opačné větve směrem proti hrotu.

Posledním tematickým blokem jádra je **NetEntity**, který vytváří obecnou třídu stejného jména a propojení mezi konkrétními instancemi této entity pomocí třídy **NetEntityRelation**. Třída **NetEntity** reprezentuje jednotlivé prvky infrastruktury, přičemž samotné instance jsou zpravidla tvořeny specializacemi této třídy implementovanými v rámci specifického rozšíření. [2] Blok jádra **NetEntity** je tedy jedním z nejdůležitějších prvků jádra z hlediska zapojení SR k jádru VMŽI.

3.2 Vztah mezi VMŽI a RailTopoModelem

Víceúčelový model železniční infrastruktury myšlenkově vychází z RailTopoModelu a využívá obdobné struktury a metody popisu infrastruktury. Základní struktura VMŽI je

téměř totožná se strukturou RailTopoModelu, avšak vybrané konkrétní části jádra se svojí strukturou mohou od RailTopoModelu lišit, přičemž ale zajišťují podobnou funkcionalitu v rámci modelu. Důležitým rozdílem mezi VMŽI a RailTopoModelem je existence specifických rozšíření umožňujících přizpůsobit vybrané datové struktury požadavkům daného projektu [2], přičemž RailTopoModel specifikuje dané objekty pouze na vyšší úrovni abstrakce a má omezenou možnost přizpůsobení datových struktur pro potřeby konkrétního projektu. [1] Funkcionalita struktur přizpůsobujících popis danému projektu je však implementována v určitém rozsahu v railML (přesněji ve verzi railML 3) v podobě tzv. **Use cases**, v rámci nichž dochází k definici specifických požadavků na formát pro uchovávání a výměnu dat pro jednotlivé uživateli definované případy. [10] Již před zahájením tvorby SR pro potřeby DSFD vznikl **Use case** pro potřeby DSFD [22], který však nebyl pro definici požadavků na SR využit a až dodatečně s ním byly výsledné požadavky porovnány.

3.3 Specifické rozšíření

Specifické rozšíření je část struktury VMŽI uzpůsobujícího VMŽI pro potřeby daného projektu. V rámci SR dochází k implementaci nadstavby k existujícím nástrojům jádra sloužící k umožnění geometrického popisu infrastruktury a instalovaných prvků na této infrastruktuře podle požadavků konkrétního popisu. [2]

Struktura specifického rozšíření se skládá z tematických bloků, které obsahují jednotlivé třídy, obdobně jako je tomu u jádra VMŽI, přičemž tyto tematické bloky navazují na struktury jádra prostřednictvím návaznosti na vybrané terminální třídy, přičemž hlavním spojovacím článkem mezi jádrem a SR je třída **NetEntity**, od níž je v rámci existujících SR odvozena většina dalších entit a jejich specializací. Přímý přístup tříd implementovaných v rámci SR k třídám jádra VMŽI není znemožněn, čehož je využíváno vybranými SR k definici typů jiných entit v tematických blocích **Type** a navazujících specializací tohoto bloku, přičemž entity typů nejsou odvozeny od třídy **NetEntity**. [14] [15]

Jednotný popis struktury specifického rozšíření je obtížnější než v případě jádra z důvodu závislosti implementovaných tematických bloků a tříd v rámci těchto bloků na konkrétním specifickém rozšíření, lze však najít společné trendy v rámci specifických rozšíření, neboť všechna existující specifická rozšíření implementují tematické bloky **ModularEntity**, **Type**, **ModularEntityType** a **Geometry**, přičemž poslední jmenovaný blok je jako jediná struktura identický pro všechna existující specifická rozšíření, vč. navrhovaného SR pro potřeby DSFD. Následující popis struktury, podobně jako v případě popisu jádra, je uveden v úrovni potřebné pro uživatele, popř. tvůrce základního specifického rozšíření. [13] [14] [15]

Blok **Geometry** zajišťuje popis geometrických vlastností koleje v trojrozměrném prostoru. Všechny třídy tohoto tematického bloku jsou odvozeny od třídy **NetEntity** prostřednictvím třídy **GeometricEntity**. [11] Dále se blok dělí na 3 samostatné části, každá specializující se na vybraný aspekt geometrie koleje:

- **HorizontalCurve** a její specializace **StraightHC**, **CurveHC**, **ClothoidHC** a **CubicParabolaHC** zajišťují popis geometrie koleje v horizontální ploše (popis odpovídající situačnímu výkresu).
- **VerticalCurve** a její specializace **StraightVC** a **ParabolaVC** popisují geometrii koleje ve vertikální ploše (popis odpovídající podélnému řezu).

- **Superelevation** a její specializace **ConstantSuperelevation** a **LinearSuperelevationRamp** zajišťují popis převýšení v dané koleji v obloucích a přechodnicích.

Důležitou vlastností tohoto modulu je nezávislost popisu výše zmíněných složek geometrie koleje, díky čemuž lze popsat i složité geometrické parametry kolejí po jednotlivých složkách bez nárůstu obtížnosti popisu. Modul **Geometry** je součástí SR a nikoliv jádra VMŽI z důvodu nutnosti snadné upravitelnosti modulu pro potřeby speciálních projektů s atypickými geometrickými vlastnostmi, např. popis rozchodu (jako je navrženo v railML [21]), jiným způsobem implementace převýšení (nelineární vzestupnice), apod. Struktura tematického bloku **Geometry** je zobrazena ve schématu v Příloze 1 v Obr.: 2.

Dalšími bloky s velmi podobnou strukturou napříč všemi specifickými rozšířeními jsou bloky **Type** a **ModularEntityType**, které zajišťují definici typů a vybraných vlastností modulárních entit, které jsou zpravidla shodné pro celý projekt, popř. celé specifické rozšíření. Všechny existující SR využívají dané bloky pro definici typů vybraných návěstidel a jejich vybraných vlastností. Zvláštností tříd v těchto blocích je, že nejsou odvozeny od třídy **NetEntity**, ale jsou odvozeny přímo od třídy **NamedResource**, která je součástí tematického bloku **Base** jádra VMŽI. Predefinice typů probíhá zpravidla při prvotní tvorbě projektu v rámci specifického rozšíření buď ručním vkládáním typů uživatelem, nebo jako v případě (nejen) tohoto SR automatickým během SW po potvrzení uživatelem.

3.3.1 Existující specifická rozšíření pro VMŽI

V současné době existují tři specifická rozšíření vytvořena pro VMŽI. Každé specifické rozšíření bylo vytvořeno za jiným účelem a liší se nabídkou popisovaných prvků infrastruktury a informacemi o nich uchovávaných. V rámci seznámení se s těmito specifickými rozšířeními byly vytipovány možnosti využití již implementovaných tříd pro řešení požadavků kladených na SR vytvořené pro potřeby DSFD.

Specifické rozšíření ProETCS

Specifické rozšíření vzniklo pro potřeby popisu sítě vybavené jednotným evropským vlakovým zabezpečovačem ETCS. Z hlediska struktury SR se jedná o nejjednodušší ze všech SR, protože vedle výše zmíněných tematických bloků **Geometry**, **Type** a **ModularEntityType** shodných (nebo téměř shodných) pro všechny existující SR implementuje pouze tematický blok **ModularEntity**, sloužící pro popis vybavení infrastruktury, a v rámci něj implementuje pouze logické bloky operačních entit, obsahující terminální třídy **PlatformEdge** (označující polohu nástupních hran) a **SpeedSection** (poskytující informace o rychlostním profilu), a logický blok řídicích entit s terminálními třídami **Balise** (označující polohu a poskytující informace o balizách) a **TracksideSignal**, poskytující informace o poloze a typu návěstidla. [13]

Toto SR bylo identifikováno jako vhodný základ pro SR vytvářené pro potřeby DSFD z důvodu jeho relativní jednoduchosti a přehlednosti struktury a využitelnosti všech již implementovaných aspektů v rámci tohoto SR v rámci navrhovaného SR, přičemž rozsah úprav již implementovaných aspektů je nižší, než v případě jiných SR. Schéma struktury SR ProETCS (přesněji její část popisující modulární entity) je uvedeno v Příloze 1 v Obr.: 3.

Specifické rozšíření ProMetro

Dalším rozšířením vytvořeným pro víceúčelový model železniční infrastruktury je SR vytvořené pro potřeby popisu sítě metra v Praze. Z hlediska struktury je SR ProMetro složitější než SR pro ETCS, přestože vedle společných tematických bloků **Geometry**, **Type** a **ModularEntityType** implementuje také pouze tematický blok **ModularEntity**, v rámci nějž implementuje logický blok operačních entit, který je strukturálně identický se stejným blokem ve SR ProETCS a dále logický blok řídicích entit, v rámci nějž však neimplementuje terminální třídu **Balise**, ale nahrazuje ji třídou **TransmissionLoop**. Dále je v tomto logickém bloku zahrnuta třída **SwitchMachine** poskytující topologické informace o přestavníku. Třída **SwitchMachine** je též navázána dvojicí relací na třídu **PositionedRelation**, každá z nich reprezentující spojení do jedné z větví z výhybky a zároveň tak i koncovou polohu přestavníku. Navíc v porovnání se SR ProETCS implementuje navíc logický blok elektrifikačních entit, sloužící pro evidování míst vybavených napájecími zařízeními, a logický blok entit pro zjišťování přítomnosti kolejového vozidla, určený k popisu kolejových obvodů.[14]

V rámci tohoto SR byl identifikován logický blok entit pro zjišťování přítomnosti kolejového vozidla jako možný pro implementaci do SR pro potřeby DSFD. Obdobně byla identifikována třída **SwitchMachine** vzhledem k pravděpodobné nutnosti popisu výhybek a s nimi spojených přestavníků v rámci navrhovaného specifického rozšíření. Schéma struktury SR ProMetro (přesněji její část popisující modulární entity) je uvedeno v Příloze 1. v Obr.: 4.

Specifické rozšíření ProVEXA

Posledním existujícím specifickým rozšířením pro VMŽI je SR určené pro potřeby projektu VEXA, což je projekt vývoje vlakového expertního systému plnění úlohy autonomního vlaku. Cílem tohoto projektu je dosáhnout operace vlaku bez nutnosti lidské obsluhy využitím existujících i nových technologií. [16] Jedná se o strukturálně nejkomplicovanější SR, protože implementuje tematický blok **SurroundingEntity**, sloužící k popisu objektů okolí železniční infrastruktury, ale které nemají na infrastrukturu přímý vliv, jako jsou domy, stožáry, ploty apod. Tyto entity jsou na rozdíl od všech doposud zmíněných lokalizovány pomocí pozice geografické a nikoliv pozice asociované s lineárním elementem infrastruktury (viz kapitolu 3.1). V rámci tematického bloku **ModularEntity** byl, vedle logických bloků a tříd shodných s předchozími SR, rozšířen o logický blok strukturálních entit obsahující třídy pro popis struktury nástupiště (nejen nástupní hrany jako v ostatních SR) a třídy pro popis úrovněového křížení s pozemní komunikací a příchody k úrovněovým nástupišťům, i infrastrukturních prvků mimoúrovňového křížení jako jsou mosty a propustky. [15]

Na základě seznámení se s tímto specifickým rozšířením byl identifikován logický blok strukturálních entit, přesněji třídy pro popis křížení železniční infrastruktury s pozemními komunikacemi, vodotečemi apod. jako možný pro implementaci do SR pro VMŽI vzhledem k přítomnosti relativně vysokého počtu těchto staveb na kolejišti DSFD. Dále bylo zkoumané SR určeno jako možný zdroj informací o způsobu realizace v případě nutnosti implementace entit lokalizovaných s pomocí geografické lokalizace v rámci SR pro DSFD.

Schéma struktury specifického rozšíření ProVEXA (resp. jeho části popisující entity) je uvedeno v Příloze 1. v Obr.: 5.

3.4 Aplikace VMŽI na DSFD

V rámci přípravy na tvorbu specifického rozšíření pro DSFD bylo provedeno ověření možností využití VMŽI. Prověřované možnosti vychází z potřeb aplikace konceptu **Železnice 4.0**[3] na DSFD, jehož je tvorba datového modelu kolejiště součástí. V rámci této práce však bylo ověřeno i využití mimo tuto koncepci, která však stále vychází z potřeb DSFD. Nutnost prověření jednotlivých možností užití vyplývá z dramaticky rostoucí komplexnosti modelu, a s tím spojené časové náročnosti tvorby specifického rozšíření i samotného modelu kolejiště DSFD, nehledě na snížení přehlednosti a tím i jednoduchosti využití modelu pro praktické aplikace v případě, že by byl model tvořen bez ohledu na reálné potřeby a možnosti dalšího rozvoje DSFD. Tímto prověřením byly nalezeny požadavky na model a tím i na tvořené specifické rozšíření, např. požadavky na zaznamenání elementů kolejiště, informace o nich, a způsob jejich propojení. Dále byly na základě tohoto ověření vytvořeny požadavky na podklady pro tvorbu digitálního modelu, protože existující podklady z doby vzniku DSFD v určitých případech neobsahovaly požadované informace, nebo se neshodovaly s reálným stavem pozorovaným na kolejišti DSFD. Podobně se i tato absence vybraných přesných podkladů zpětně promítla do zavržení vybraných užití VMŽI na DSFD z důvodu neúměrné časové náročnosti tvorby podkladu nebo datové struktury v rámci SR VMŽI pro možnost implementace požadavků pro dané využití.

3.4.1 Nový software řízení modelových souprav

Jedním z budoucích projektů zvažovaných pro realizaci je tvorba nového SW řízení souprav na modelovém kolejišti volně vycházející z principů vedení vlaků systémem ATO pod ETCS L2 nebo L3. Jedním ze základních požadavků tohoto řídicího SW je přesný popis infrastruktury z hlediska logických návazností jednotlivých prvků za sebou, přesné informace o rozmístění prvků infrastruktury ovlivňující jízdu vlaku po ní, zvláště pak vzdálenosti těchto prvků mezi sebou (z důvodu využití odometrie při určování přesné polohy vlaku na infrastruktuře). Jiné geometrické vlastnosti infrastruktury jsou z hlediska tohoto systému v podmínkách modelového kolejiště nedůležité a projevují se nepřímo jako rychlostní profil, který je naopak z hlediska řídicího SW velmi důležitý.

Identifikované požadavky na SR VMŽI a podklady pro jeho tvorbu nutné pro realizaci nového řídicího SW:

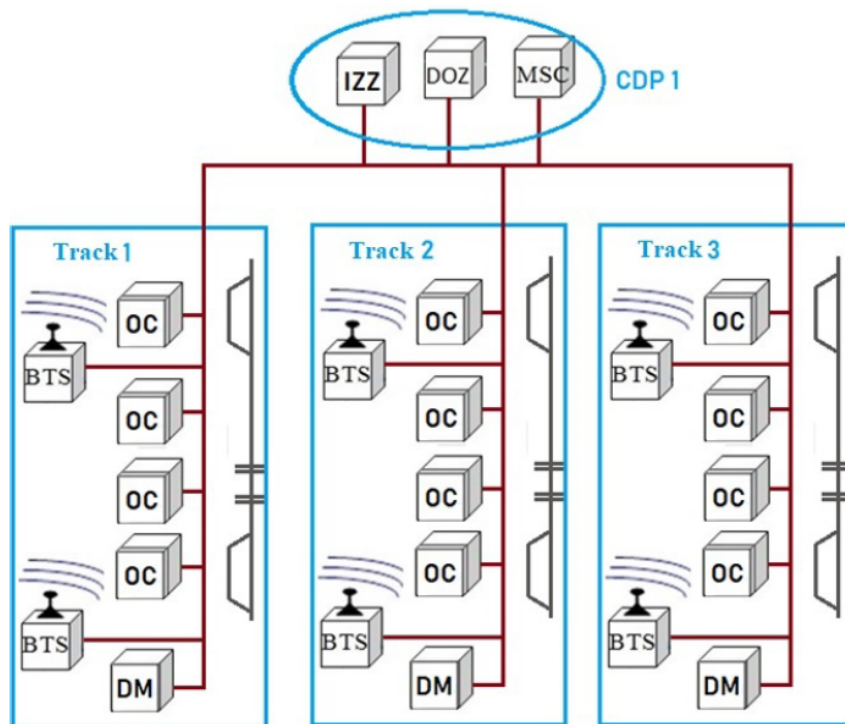
1. Topologický popis infrastruktury na úrovni mikro, zvláště pak návaznosti jednotlivých prvků infrastruktury na sebe a jejich sjízdnost v daném směru.
2. Přesné vzdálenosti mezi jednotlivými prvky infrastruktury zaznamenanými v rámci VMŽI s požadovanou přesností.
3. Maximální povolená rychlost na všech úsecích infrastruktury a druh rychlostního profilu pro nějž daná rychlost platí (např. V_{130} , V_{NS} , V_3 apod).
4. Podélný sklon každého úseku infrastruktury.
5. Zaznamenání polohy a vlastností následujících prvků infrastruktury:

- (a) Lokalizačních prvků sloužících k určení přesné polohy vlaku v systému a opravě chyby odometrie (simulujících zařízení eurobalise z reálné železnice) a informace o nich.
- (b) Začátků a konců nástupních hran, jejich výšky nad temenem kolejnice.
- (c) Návěstidel, jejich druhů a směrů platnosti.
- (d) Přejezdů a úrovnových přechodů.
- (e) Začátky a konce potenciálně rizikových úseků infrastruktury: nadjezdy, podjezdy, mosty, tunely.

Ověřením splnitelnosti požadavků kladených tímto systémem na VMŽI bylo zjištěno, že veškeré klíčové požadavky daného systému bylo možno splnit, protože všechny výše uvedené body jsou splněny buď na základě samotného konceptu a základních vlastností VMŽI, nebo již byly implementovány v rámci již existujícího SR. Z tohoto důvodu je VMŽI považován pro dané využití za vhodný a navrhované SR bylo přizpůsobeno potřebám tohoto rozšíření.

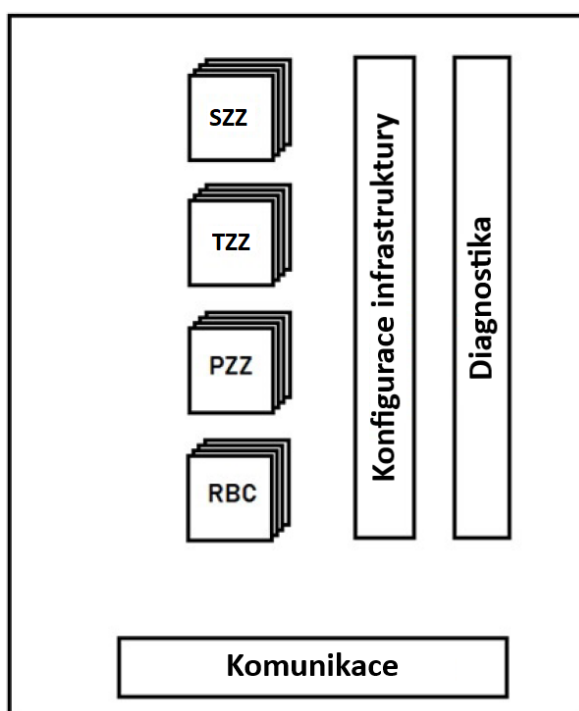
3.4.2 Integrovaná zabezpečovací zařízení pro DSFD

Dalším ze zvažovaných budoucích projektů je realizace integrovaného zabezpečovacího zařízení pro ovládání DSFD. V rámci IZZ dochází k funkčnímu sloučení staničního, traťového a přejezdového zabezpečovacího zařízení do jediného systému, který navíc umí plnit funkci radioblokové centrály ETCS [3] potřebné pro ETCS úroveň 2 a 3. [4] Důležitým aspektem IZZ je vysoká úroveň centralizace funkce zabezpečovacího zařízení a co nejvyšší míra přenesení logických a rozhodovacích funkcí do centra zabezpečovacího zařízení a využití na trhu běžně dostupných (a tedy výrazně levnějších) zařízení pro ovládání vnějších prvků zabezpečovacího zařízení (přestavníky, výstražníky přejezdů, ovládací prvky návěstidel, apod). V rámci IZZ se předpokládá agregace zabezpečovacího zařízení více tratí do jediného centra IZZ. [17] Schéma IZZ je uvedeno v Obr.: 1.

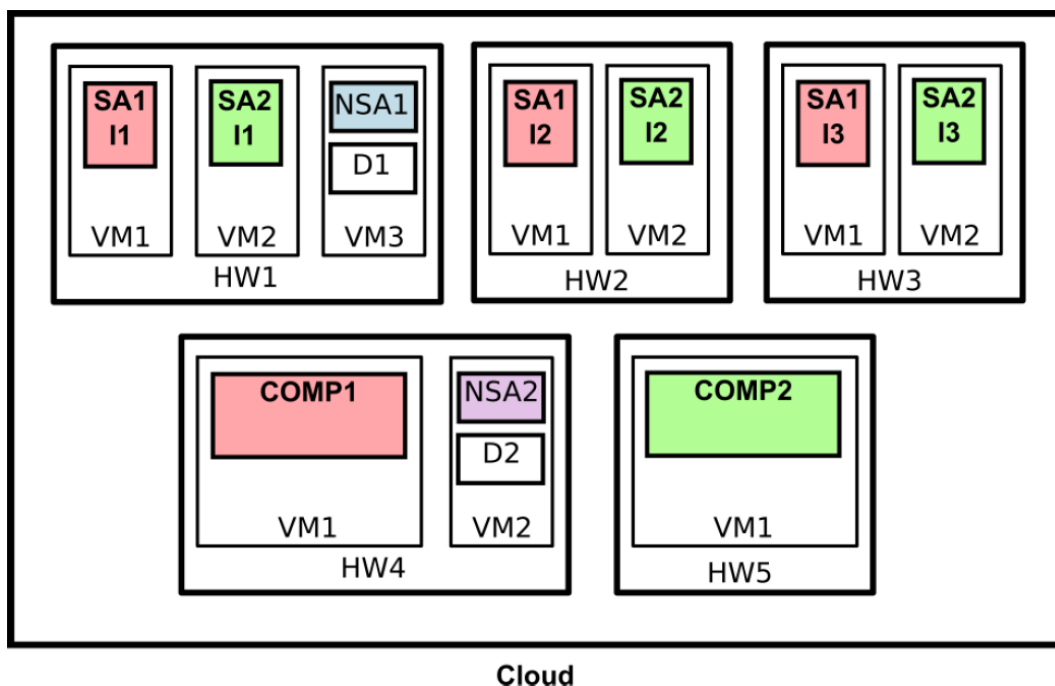


Obrázek 1: Schéma architektury možné implementace integrovaného zabezpečovacího zařízení. Převzato ze [17].

Dalším aspektem IZZ je využití generických aplikací pro jednotlivé funkce zabezpečovacího zařízení, tzn. takové, které v rámci své struktury neobsahují přímo vytvořený model infrastruktury, ale jsou schopny tyto informace získat ze standardizovaného modelu infrastruktury [17] (současné systémy jsou zpravidla navrženy tak, že každá implementace daného systému, ač typově shodného, je jedinečná a nepřenositelná mezi různými řízenými úseky, protože ve své struktuře obsahuje pevně naprogramovaný popis řízeného úseku). Jádro IZZ lze realizovat i v cloudovém prostředí s použitím vysokého počtu virtuálních počítačů s vysokým výkonem, každý nezávisle na sobě řešící úlohu identické úlohy, které jsou následně porovnávány a v případě souladu předávány po privátní IP síti objektovým kontrolerům na trase, které zajistí provedení požadovaných akcí (zahájení výstrahy na přejezdu, přestavení výhybky, apod.). [17] Struktura jednotlivých instancí virtuálních strojů IZZ v cloudu je uvedena v Obr.: 2 a výsledná struktura možné implementace IZZ v cloudu je uvedena v Obr.: 3.



Obrázek 2: Architektura virtuálního počítače cloudového řešení integrovaného zabezpečovacího zařízení. Převzato ze [17], přeloženo.



Obrázek 3: Architektura cloudového řešení integrálního zabezpečovacího zařízení. Převzato ze [17].

Základním požadavkem tohoto systému jsou stejné informace jako pro jednotlivé prvky zabezpečovacího zařízení, které IZZ nahrazuje, navíc však z důvodu funkce radioblokové centrály vyžaduje přesné pozice těchto prvků na infrastruktuře a nikoliv jen jejich přibližné polohy a vztahy mezi nimi. Proto je pro IZZ znalost přesné délky jednotlivých úseků nezbytně nutná z důvodu funkce radioblokové centrály, která musí vydat oprávnění k jízdě na přesně stanovenou vzdálenost). [5]

V průběhu prověřování možnosti využití VMŽI pro IZZ na DSFD byly zjištěny dva základní koncepty implementace IZZ na DSFD, které se liší požadavky na datový popis infrastruktury. Prvním způsobem je implementace IZZ jako systému zcela nezávislého na existujících systémech řízení dopravy na DSFD. V takovémto případě by postačoval popis, který klade menší důraz na dopravní a nepotřebuje informace o lokálním způsobu řízení a přiřazení prvků k řídicím místům (dopravním kancelářím, stavědlům apod.) a dopravnám. Druhý systém slouží jako náhrada současného systému řízení, a proto musí respektovat existující zařízení pro řízení provozu na DSFD a umožňovat jejich funkci v rámci systému, kvůli čemuž vyžaduje podrobnější popis, hlavně v oblasti přiřazení prvků řídicím místům, dopravnám apod. Následující popis identifikovaných požadavků odpovídá druhé variantě.

Identifikované požadavky na SR VMŽI a podklady pro jeho tvorbu nutné pro realizaci IZZ:

1. Topologický popis infrastruktury na úrovni mikro, zvláště pak návaznosti jednotlivých prvků infrastruktury na sebe a jejich sjízdnost v daném směru.
2. Přesné vzdálenosti mezi jednotlivými prvky infrastruktury zaznamenanými v rámci VMŽI s požadovanou přesností.
3. Maximální povolená rychlost na všech úsecích infrastruktury a druh rychlostního profilu pro nějž daná rychlost platí (např. V_{130} , V_{NS} , V_3 apod).
4. Podélný sklon každého úseku infrastruktury.

5. Popis dopraven a jejich lokalizace na základě prvků v infrastruktuře.
6. Přiřazení ovládaných prvků infrastruktury řídicím místům a dopravnám.
7. Zaznamenání polohy a vlastností následujících prvků infrastruktury:
 - (a) Lokalizačních prvků sloužících k určení přesné polohy vlaku v systému a opravě chyby odometrie (simulující zařízení eurobalise z reálné železnice) a informace o nich.
 - (b) Začátků a konců nástupních hran, jejich výšky nad temenem kolejnice.
 - (c) Návěstidel, jejich druhů, směrů platnosti, instalovaných návěstních svítilen, označení návěstidel a jejich příslušnost k dopravnám.
 - (d) Přejezdů a úrovnových přechodů a druh jejich zabezpečení.
 - (e) Začátky a konce potenciálně rizikových úseků infrastruktury: nadjezdy, podjezdy, mosty, tunely.
 - (f) Výhybek, jejich začátků a konců, jejich čísel a druh přestavníků.
 - (g) Výkolejek a jejich označení.
 - (h) Řídicích míst, jejich druh (stavědlo, stanoviště výhybkáře, dopravní kancelář apod.) a z nich ovládané prvky infrastruktury.
 - (i) Detekčních prvků kolejových vozidel a izolované styky kolejových obvodů.

Vzhledem k přítomnosti funkcionality radioblokové centrály v rámci integrovaného zabezpečovacího zařízení tvoří požadavky na nový SW pro řízení modelových souprav podmnožinu požadavků na IZZ, a proto již nebyly znovu ověřeny v rámci ověřování použitelnosti VMŽI pro tvorbu datového modelu pro IZZ. S výjimkou evidence výkolejek, druhu přestavníku u výhybek, řídicích míst a přiřazování prvků infrastruktury dopravnám a řídicím místům byly již všechny ostatní požadavky řešeny v rámci jádra nebo již existujícího specifického rozšíření. Evidence druhu přestavníku je v rámci SR řešena přidáním parametru entitě přestavníku. K řešení problematiky výkolejek, řídicích míst a dopraven bylo zvoleno vytvoření nových druhů třídy **NetEntity** umožňující reprezentaci daných prvků infrastruktury. Požadovaná propojení pak jsou řešena s pomocí propojovací třídy **NetEntityRelation**, která je součástí jádra. Vzhledem k charakteru se VMŽI dobře hodí k řešení problematiky datového popisu kolejiště pro potřebu IZZ pro DSFD, protože splňuje požadavky na topologický i geometrický popis infrastruktury, návaznost jednotlivých prvků a popis vztahů mezi nimi.

3.4.3 Tvorba 3D prostředí pro potřeby simulátorů na DSFD

Prověřovaným využitím VMŽI na DSFD bylo i možnost užití výsledného modelu jako vstupních dat pro tvorbu 3D prostředí pro simulátor vlaku. V porovnání s předchozími prověřovanými možnostmi využití vyžaduje tvorba 3D nejen přesnou funkční polohu v rámci koleje, ale i fyzickou pozici daného prvku v prostoru. Vzhledem k aplikaci se nabízí možnost ukládat do VMŽI i přítomnost prvků kolejiště, které nesouvisí přímo s provozem, ale umožňovaly by snadnější tvorbu přesné kopie modelového kolejiště DSFD i návrh tratí zcela nezávisle na modelovém kolejišti z velké části v prostředí VMŽI. Doplnění o prvky nesouvisející přímo s železniční infrastrukturou však nejsou nezbytně nutné pro fungování simulace a mohou být doplněny dodatečně v prostředí editoru prostředí v simulačním SW nebo jiném nástroji, neboť v těchto případech odchylky umístění nezmění funkcionalitu simulace.

Identifikované požadavky na SR VMŽI a podklady pro jeho tvorbu nutné pro realizaci prostředí pro 3D simulaci:

1. Topologický popis infrastruktury na úrovni mikro, zvláště pak návaznosti jednotlivých prvků infrastruktury na sebe a jejich sjízdnost v daném směru.
2. Zaznamenání geometrické polohy koleje s dostatečnou věrohodností.
3. Funkční i fyzické polohy jednotlivých prvků infrastruktury zaznamenané s dostatečnou přesností.
4. Maximální povolená rychlost na všech úsecích infrastruktury a druh rychlostního profilu pro nějž daná rychlost platí (např. V_{130} , V_{NS} , V_3 apod.).
5. Podélný sklon každého úseku infrastruktury.
6. Zaznamenání polohy a vlastností následujících prvků infrastruktury:
 - (a) Začátků a konců nástupních hran, jejich výšky nad temenem kolejnice.
 - (b) Návěstidel, jejich druhů, směrů platnosti, instalovaných návěstních svítlen, druhu jejich upevnění (stožár, krakorec apod.) a označení návěstidel.
 - (c) Přejezdů a úrovnových přechodů a druh jejich zabezpečení.
 - (d) Začátky a konce potenciálně rizikových úseků infrastruktury: nadjezdy, podjezdy, mosty, tunely.
 - (e) Výhybek, jejich začátků a konců.
 - (f) Výkolejek.

Identifikované požadavky se do určité míry shodují s požadavky na model pro tvorbu IZZ, ale navíc je vyžadováno zaznamenání přesné fyzické polohy prvků infrastruktury vůči koleji a i přesnější geometrický popis koleje samotné. VMŽI se svým charakterem dobře hodí pro záznam geometrických vlastností a podporuje uchování všech potřebných údajů v rámci jádra nebo rozšíření. I možnost popisu prvků kolejiště nesouvisející s železniční infrastrukturou vč. možnosti zaznamenávat je do souřadnicového systému nesouvisejícího s kolejí (např. kartézský souřadnicový systém) byl realizován v rámci jiných specifických rozšíření. Na základě výše uvedených skutečností je VMŽI vhodný pro tuto aplikaci, avšak z důvodu výrazného zvýšení komplexnosti není možnost zaznamenání staveb nesouvisejících s infrastrukturou implementována v rámci SR. Specifické rozšíření, podklady i následná tvorba modelu jsou přizpůsobeny požadavkům pro popis pouze železniční infrastruktury a s ní přímo souvisejících elementů v rozsahu a přesnosti potřebné pro tvorbu 3D prostředí pro simulátory.

3.4.4 Pasportizace kolejiště DSFD

V rámci ověřování byla prověřena možnost využití VMŽI pro potřebu pasportizace vybavení modelového kolejiště DSFD. Funkcí takového pasportu by bylo koncentrovat do sebe veškeré informace o železničním svršku, vybraných stavbách železničního spodku modelového kolejiště, dále prvky zabezpečovacího zařízení a skutečná elektroinstalace modelového kolejiště DSFD vč. prvků napájení a ovládání kolejiště. Vytvoření úplného pasportu vybavení vyžaduje přesné zaznamenání geometrické polohy koleje, druhu využitého kolejiva, pozice styků mezi jednotlivými úseky koleje a jejich délce. Dále je nutno zaznamenat všechny prvky infrastruktury, jejich fyzickou polohu a jejich stavební vlastnosti, připojení těchto prvků z hlediska modelového i z hlediska reálného světa. Na toto navazující by byla

část obsahující skutečnou elektroinstalaci modelového kolejiště, tj. ovládacích prvků, např. skutečných (nikoliv simulovaných) přestavníků výhybek, detekčních prvků přítomnosti vozidel a obvodů napájení. Vhodný by byl i popis rozhraní s navazujícími částmi DSFD, jako jsou řídicí počítače, elektrická síť apod.

Identifikované požadavky na SR VMŽI a podklady pro jeho tvorbu nutné pro realizaci pasportu modelového kolejiště DSFD:

1. Zaznamenání geometrické polohy koleje s dostatečnou přesností.
2. Informace o úsecích kolejí, pozice styků s navazujícími úseky, informace o vlastnostech kolejiva.
3. Fyzické polohy jednotlivých prvků infrastruktury zaznamenané s dostatečnou přesností.
4. Podélný sklon každého úseku infrastruktury.
5. Zaznamenání polohy a informací o řídicích prvcích modelového kolejiště a přiřazení prvků k těmto řídicím prvkům.
6. Zaznamenání polohy a informací o napájecích prvcích kolejiště a přiřazení kolejí jednotlivým napájecím prvkům.
7. Zaznamenání informací o rozhraních modelového kolejiště s okolními zařízeními.
8. Zaznamenání polohy a vlastností následujících prvků infrastruktury:
 - (a) Lokalizačních prvků sloužících k určení přesné polohy vlaku v systému a opravě chyby odometrie (simulující zařízení eurobalise z reálné železnice) a informace o nich.
 - (b) Návěstidel, jejich druhů, směrů platnosti, instalovaných návěstních svítilen, druhu jejich upevnění (stožár, krakorec apod.).
 - (c) Přejezdů a úrovnových přechodů a druh jejich zabezpečení a konstrukce.
 - (d) Vybraných staveb železničního spodku: zarážedel, mostů, tunelů, nástupišť.
 - (e) Výhybek, jejich začátků a konců.
 - (f) Výkolejek.
 - (g) Detekčních prvků kolejových vozidel a izolovaných styků kolejových obvodů.

Ověřením požadavků bylo zjištěno, že by v rámci VMŽI bylo možné vytvořit specifické rozšíření poskytující požadovanou funkcionalitu, protože funkcionalita popsaná body 1., 3. a 4. je součástí jádra VMŽI, všechny části bodu 8. byly již realizovány v rámci jiných specifických rozšíření, nebo řešeny v rámci jiných prověřovaných možností užití. Body 5. a 6. nejsou obsahem jádra ani žádného z existujících rozšíření VMŽI, ale bylo by je možné realizovat jako nově vytvořenou kategorií entit třídy **NetElement** lokalizované s pomocí na koleji nezávislého souřadnicového systému, podobně jako tomu je u vybraných objektů nesouvisejících přímo s železniční infrastrukturou u jiných specifických rozšíření. Požadovaná propojení by bylo možno realizovat s pomocí propojovací třídy **NetEntityRelation**, popřípadě, vzhledem k velkému počtu propojení, z ní odvozené třídy specifické pro propojení jednotlivých prvků. Zaznamenání informací o rozhraních (bod 7.) by bylo možno realizovat s pomocí nelokalizované kategorie třídy **NetEntity**. Požadavky bodu 2. by bylo možno uspokojit doplněním atributů do tříd logického bloku **Geometry** zajišťující požadovaný popis.

Na základě výše provedeného rozboru je zřejmé, že je možné vytvořit SR VMŽI pro potřeby DSFD, které by dokázalo pokrýt potřeby úplného pasportu vybavení kolejiště DSFD, avšak za cenu úplného přizpůsobení SR potřebám tohoto pasportu, nebo za cenu neúměrného zvýšení komplexnosti tohoto SR, všech modelů vytvořených tímto specifickým rozšířením (a to i v situacích, kdy není cílem tohoto modelu pasportizace; tento problém by bylo možno odstranit za cenu ještě dalšího zvýšení komplexnosti SR) a snížení přehlednosti a další využitelnosti modelů vytvořených s tímto specifickým rozšířením. Dalším problémem by byla nutnost odlišit skutečné a simulované prvky infrastruktury, které ale existují zároveň a jsou navázány na identický prvek. Typickým příkladem tohoto jsou přestavníky, které existují v simulované formě, ale i ve skutečné formě a různé možné způsoby využití digitálního modelu vyžadují informace o simulovaném přestavníku, přičemž pasport vyžaduje informace o tom skutečném. Tato a podobné situace by byly řešitelné vytvořením samostatných **NetEntity** pro různé situace, což by však značnou měrou zvýšilo komplexnost SR.

Z výše popsaných důvodů není navržené SR přizpůsobeno potřebám pasportizace modelového kolejiště DSFD, přestože by návrh takového SR byl možný, avšak za cenu ztráty jiné funkcionality, nebo za cenu neúměrně vysoké komplexnosti SR, popř. i modelu samotného.

4 Návrh specifického rozšíření

Návrh specifického rozšíření vychází z požadavků na SR identifikovaných v rámci prověření možností využití VMŽI na DSFD a ze zkušeností získanými při seznámení se s existujícími specifickými rozšířeními. Ze zkoumaných možností využití VMŽI byly pro aplikaci v rámci SR vybrány požadavky na datový popis infrastruktury pro potřeby IZZ a pro potřeby tvorby prostředí pro 3D simulátory. Na základě definovaných požadavků a v rámci jiných SR již implementovaných byl vytvořen seznam terminálních tříd tematického bloku **ModularEntity** pro implementaci v rámci struktury SR pro potřeby DSFD. Z důvodu komplexnosti bylo rozhodnuto o neimplementaci žádné ze tříd tematického bloku **SurroundingEntity**. Seznam implementovaných terminálních tříd v rámci SR je uveden níže:

Seznam implementovaných terminálních tříd entit tematického bloku ModularEntity:

- Logický blok **ControlEntity**:
 - Balise - reprezentující simulované balízy systému ETCS v kolejišti
 - SwitchMachine - představující simulované přestavníky
 - TracksideSignal - reprezentující všechny druhy návěstidel vyskytujících se na DSFD
 - Derail - reprezentující výkolejky
 - SignalBox - reprezentující simulovaná místa v infrastruktuře z nichž se obsluhuje zabezpečovací zařízení
- Logický blok **OperationalEntity**:
 - SpeedSection - definující rychlostní profil kolejového úseku
 - PlatformEdge - reprezentující nástupní hranu
 - TrackCircuit - reprezentující kolejové obvody
 - ControlArea - reprezentující dopravní (přepravní stanoviště apod.)
- Logický blok **StructuralEntity**:
 - BridgeUC - reprezentující most železniční trati
 - CulvertUC - reprezentující propustek
 - RailwayCrossingLC - reprezentující železniční přejezd
 - PlatformAccessLC - reprezentující úroňový příchod k úroňovému nástupišti
 - BridgeOC - reprezentující most nad železniční tratí
 - TunnelOC - reprezentující tunel

Dále je v rámci SR implementována třída **SignalType** z tematického bloku **ModularEntityType** definující typy návěstidel (navazuje na třídu **TracksideSignal**). Pro zajištění jednotného názvu tříd a celkově zachování jednotného způsobu pojmenování v rámci VMŽI, bylo u již existujících i zcela nově navržených tříd i jejich atributů použito anglických názvů v souladu s konvencemi VMŽI.

Již existující **Use case** pro potřeby DSFD (UC:IS:Railway Simulation Laboratory) [22] definoval požadavky na popis uvedené v seznamu níže.

Seznam požadavků na popis definovaný v Use case pro potřeby DSFD: (Převzato ze [22], přeloženo)

- *“Topologie - pohled v úrovni **mikro** i **makro**”*
- *Výhybky*
- *Návěstidla*
- *Rychlostní profily a omezení - různé druhy*
- *Nástupní hrany - vč. ozn. míst zastavení a úrovniových přechodů*
- *Změny sklonů*
- *Balízy - dva různé druhy (fyzické i virtuální) jsou vyžadovány*
- *Kolejové obvody*
- *Horizontální popis geometrie koleje*
- *Přejezdy*
- *Tunely*
- *Mosty*
- *Změny elektrifikace - vč. návěstidel pro elektrický provoz (pro pozdější implementaci)*
- *Výstražné kolíky”*

Ačkoliv byly požadavky na model vytvářeny nezávisle na sobě, **Use case** definuje velmi podobné požadavky na popis infrastruktury k těm, které byly implementovány v rámci tvořeného SR. V rámci SR pro potřeby DSFD nebyly zahrnuty změny elektrifikace a výstražné kolíky, a to z důvodu, že se na modelovém kolejišti DSFD ve fyzické podobě nevyskytují a v rámci navrhovaného využití modelu vytvořeného v navrhovaném SR není jejich implementace nutná.

Následně byly v závislosti na požadavcích navržených v rámci zkoumání využitelnosti definovány atributy ukládané pro jednotlivé terminální třídy, přičemž snahou bylo vedle popisnosti zachovat jednoduchost a v maximální možné míře využít existujícího aparátu VMŽI. Cílem tohoto rozhodnutí bylo redukovat dobu potřebnou pro seznámení se uživateli znalých práce s jinými specifickými rozšířeními s navrhovaným SR, popř. i zvýšit kompatibilitu již existujících řešení pro získávání dat s tímto SR. Takto navržené třídy a jejich atributy jsou doplněny o abstraktní třídy jednotlivých logických bloků a zařazeny do struktury SR. Schéma této struktury, obsahující též výčet všech atributů jednotlivých tříd, je uvedeno v Příloze 1 v Obr.: 6.

Takto navržená struktura byla následně implementována do nástroje pro tvorbu VMŽI vytvořeném v prostředí editoru skriptů v prostředí grafického editoru Rhinoceros. V souladu s již existujícími konvencemi VMŽI bylo navrhované SR pojmenováno **ProDSFD2023**, přičemž pokud byly tvořeny funkce specificky nutné pro tvorbu tohoto SR, byly tvořené funkce na konci jména doplněny o **DSFD**, pro příklady viz kap. 4.5 nebo přílohu 2.

4.1 Potřebné úkony pro vytvoření specifického rozšíření v rámci tvorby nástroje pro tvorbu VMŽI

V rámci doplnění nástroje pro tvorbu VMŽI o specifické rozšíření je nutné v závislosti na konkrétním SR nutno provést založení a zprovoznění specifického rozšíření a popř. doplnění tříd a vytvoření funkcí pro tvorbu a obsluhu tvořených tříd v rámci tohoto nástroje. Následující sekce obsahují popis kroků nutných k založení samotného specifického rozšíření a kroků nutných k založení doposud neexistující třídy. Proces je pro přehlednější popsán v následujících krocích.

Seznam kroků potřebných pro založení specifického rozšíření v nástroji pro tvorbu VMŽI (pouze SR samotné):

1. Bez ohledu na strukturu samotného specifického rozšíření je nutno provést založení názvu rozšíření jako platné hodnoty parametru **modelExtensionSubset_G**, což je globální proměnná definující specifické rozšíření používané každým konkrétním projektem tvořeným ve VMŽI. Toto se musí provést v rámci funkce **NacistStrukturuTabulky(tabulka)**. Zde se též musí uvést všechny třídy implementované v rámci daného SR a atributy těchto tříd, vč. jejich typu prostřednictvím zadání jména funkce pro jejich plnění daty.
2. Dále je nutno založit seznam terminálních tříd každého z tematických bloků implementovaného v rámci SR ve funkcích **SeznamTerminalnichTridModularEntity()**, resp. **SeznamTerminalnichTridSurroundingEntity()** a **SeznamTerminalnichTridModularEntityType()**.
3. Dále je nutno v rámci funkce **PridatInstanci(trida_C, lokalniVektorPGA, lokalniMaticePHA)** v sekci **zpusobZiskaniHodnotyAtributu_C** v části **volba-ModelExtensionSubset** doplnit jméno tvořeného SR do seznamu přípustných hodnot, čímž bude při založení projektu zobrazeno jako možnost v dialogovém okně výběru SR pro daný projekt.
4. Závěr tvorby samotného SR je nutno doplnit do funkce **VkladatNetEntity(trida_I, lokalizacniTrida)** v části definic **vektorTrid_X** a **vektorTrid_Y**, sloužící v případě vektoru tříd X k odstranění uvedených NetEntit ze seznamu nabízených terminálních modifikací třídy NetEntity k vytvoření, a v případě vektoru Y k doplnění uvedených tříd do tohoto seznamu. Přestože v současné době není mechanismus tříd X a Y aktivně využíván, je stále nutné vytvořit v dané sekci záznamy specifické danému SR, typicky (z důvodu nevyužití této části) vyplněny hodnotou **Null**, z důvodu správného běhu funkce pro vkládání tříd odvozených od třídy NetEntity do modelu tvořeném v daném specifickém rozšíření.

Při tvorbě třídy doposud neimplementované v žádném z existujících SR, je vedle výše uvedeného založení dané třídy v rámci kroků 1. a 2. nutno definovat danou třídu na obecné úrovni v rámci funkce **NacistParametryVrstvy(tabulka)** v rámci části **Případy tabulek specifických pro jednotlivé moduly modelu**. ve formátu uvedeném v Src.: 1. a ve funkci **NacistParametryTridy(trida)** v části **Případy tříd specifických pro jednotlivá rozšíření modelu** ve formátu uvedeném v Src.: 2.

Case “SignalBox” ’ProDSFD2023

```
vrstva_G = “SignalBox”  
nadrazenaVrstva_G = “ControlEntity”  
barvaVrstvy_G = RGB(50, 60, 150)  
viditelnostVrstvy_G = 1  
uzamceniVrstvy_G = 1
```

Src. 1: Část zdrojového kódu nástroje pro tvorbu VMŽI, v němž dochází k implementaci třídy SignalBox specifického rozšíření pro DSFD ve funkci pro řešení grafických vrstev SW Rhinoceros. První řádek je zahájen klíčovým slovem Case, neboť každá definice je součástí výběru a je jedním z případů daného výběru, následuje název třídy v uvozovkách a popis je vhodné doplnit i názvem SR pro nějž je třída implementována. Ve druhém řádku je uveden název vrstvy v SW Rhinoceros, v níž se budou objekty dané třídy nacházet a je zavedenou praxí, že je název třídy a vrstvy stejný. Další řádek určuje nadřazenou třídu v rámci struktury SR. Zbývající řádky definují určité parametry vykreslení v rámci SW Rhinoceros a nesouvisí tak přímo s VMŽI, přesto je vhodné je nastavit pro snazší přehlednost vizualizace tvořeného modelu.

Case “SignalBox” ’ProDSFD2023

```
trida_G = “SignalBox”  
zkratkaTridy_G = “XCSgBx”  
vektorTabulekTridy_G = Array(“BaseObject”, “NamedResource”, “NetworkResource”, “NetEntity”, “ModularEntity”, “ControlEntity”, “SignalBox”)  
metodaVytvoreniGeometrickehoObjektu_G = “normalNetEntity”  
metodaUzivatelскеIdentifikace_G = “standardNamedResource”
```

Src. 2: Část zdrojového kódu nástroje pro tvorbu VMŽI, v němž dochází k implementaci třídy SignalBox specifického rozšíření pro DSFD ve funkci tvorby struktury VMŽI. První řádek je zahájen klíčovým slovem Case, neboť každá definice je součástí výběru a je jedním z případů daného výběru, následuje název třídy v uvozovkách a popis je vhodné doplnit i názvem SR pro nějž je třída implementována. Ve druhém řádku je uveden název třídy, ve třetím pak krátké jméno třídy, přičemž standardním formátem je zahájit název písmenem X, druhé písmeno označuje tematický, popř. logický blok entity (C = řídicí, G = geometrická, O = operační, E = elektrifikační, D = detekční, S = strukturní, U = entita okolí a Y = entita definice typu) a poslední čtyři písmena jsou samotným názvem. Další řádek obsahuje vektor popisující odvození dané třídy od třídy BaseObject. Následující dva řádky popisují způsob vytvoření daného objektu a způsob identifikace dané třídy.

Dále je této třídě nutno přiřadit parametry lokalizačního atributu ve funkci **NacistParametryLokalizacnihoAtributu(lokalizacniTrida, terminalniTridaNetEntity, lokalizacniAtribut)**. Zde je možné danou třídu připojit buď k již existujícímu způsobu přiřazení prostým doplněním názvu dané třídy k již existujícímu způsobu, přičemž od předchozí třídy je nově doplňovaná třída oddělena prostřednictvím čárky, nebo vytvořením nového způsobu analogicky k již existujícím případům. V případě tvorby navrhovaného SR bylo využito pouze prvního způsobu. Dále je pro tvořenou třídu vhodné vytvořit grafický objekt, který ji bude reprezentovat v grafické reprezentaci modelu při tvorbě v prostředí Rhinoceros. Toto se neprovádí v případě nelokalizovaných entit a ani pro lokalizované entity není tento krok nezbytně nutný, ale je vhodný pro zvýšení přehlednosti modelu při tvorbě. Návrh grafické reprezentace modelu se provede pomocí funkcí pro vykreslení poskytovaných SW Rhinoceros ve funkci **ZalozitDefiniciBlokuSymboluFunkcniEntity(nazevDefiniceBloku)** v části **nazevDefiniceBloku** jakožto samostatný případ pod názvem dané třídy. Je nutné upozornit, že každý blok je generován pouze při prvním vytvoření projektu a následné změny vykreslovaného bloku nejsou možné.

Pokud třída zaváděná v rámci SR (ať již existující nebo nově zaváděná) obsahuje atribut používající doposud neexistující doménu (typicky se jedná o výběr z možností specifický pro daný atribut), tak je nutno vytvořit funkce řešící dané přiřazení. Tvorba této funkce je zahájena v místě definice daného atributu v rámci zaváděné třídy (bod 1. v seznamu kroků potřebných pro založení SR) použít jako jméno procedury pro vkládání dat (druhá položka vektoru daného atributu) použít jméno vlastní rutiny, stejné jméno je pak nutno vytvořit jako možnost volby ve funkci **PridatInstanci(trida_C, lokalniVektorPGA, lokalniMaticePHA)** v části **zpusobZiskaniHodnotyAtributu_C**. V této části je vhodné vytvořit si vlastní sekci pro potřeby všech podobných výběrů v daném SR s použitím komentáře nadepisujícím tuto sekci. Jedinným výstupem implementované funkce má být požadovaná hodnota atributu uložená do proměnné **hodnotaAtributu_C**.

4.2 Úpravy již existujících částí nástroje pro tvorbu VMŽI implementované pro potřeby specifického rozšíření pro DSFD

Při implementaci specifického rozšíření pro potřeby DSFD bylo se souhlasem autora VMŽI provedeno několik drobných změn v jádrové části nástroje pro tvorbu VMŽI (část tohoto nástroje obsahující funkce, které nejsou určeny k obsluze pouze jednoho specifického rozšíření). Tyto změny se netýkaly přímo struktury jádra, která musela z důvodu kompatibility s předchozími SR zůstat neměnná.

Nejdůležitější změnou implementovanou v rámci tvorby SR pro DSFD je vytvoření třídy **NetEntityRelation**, která sice v rámci jádra existovala, avšak nebyla doposud implementována v rámci nástroje pro tvorbu a nebyla ani doposud použita v žádném z dříve realizovaných projektů. Avšak popis kolejiště DSFD vyžadoval propojení jednotlivých instancí tříd odvozených od třídy **NetEntity**, a proto bylo se souhlasem autora VMŽI přistoupeno k implementaci této třídy, a obsluhujících funkcí do jádra nástroje pro tvorbu VMŽI, tak aby obsluhovaly struktury již navržené v současné verzi jádra VMŽI. V rámci této implementace byly vytvořeny následující obsluhující funkce a části funkcí:

- Vytvoření funkce **PriraditSitovouEntituSitoveEntite(tridaA, goIdNetEntityA, tridaB, goIdNetEntityB, relationType)** sloužící k vytvoření propojení libovolných instancí tříd odvozených od třídy **NetEntity**, a která je základní funkcí pro tvorbu těchto propojení.
- Doplnění Case “**vyberRelationType**” ve funkci **PridatInstanci(trida_C, lokalniVektorPGA, lokalniMaticePHA)** sloužící pro výběr možností typu vztahu vytvářeného propojení.
- Doplnění položky hlavního menu nástroje generované funkcí **ZvolitUrovenUceleTvorbySite (opakovat)** o volbu **vzájemné přiřazování síťových entit**.
- Vytvoření funkce **PrirazovatSitoveEntity(opakovat)** generující menu zaštiťující jednotlivé funkce obsluhující přiřazení instancí tříd **NetEntity**, vč. specializovaných funkcí implementovaných v rámci SR.

Dále byly v rámci tvorby SR doplněny vybrané funkce jádra nástroje (i vybrané funkce pro obsluhu jiných existujících SR) o parametr vracející ID (**goId**) nově vytvořené instance, aby mohla být předána v rámci nadstavbové funkce, která vyvolala její tvorbu, předána další funkci zajišťující tvorbu dalších objektů nebo propojení nad tímto vytvořeným objektem.

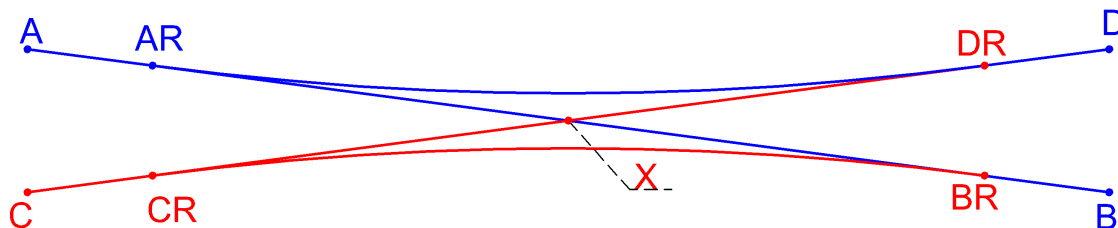
4.3 Popis řešení výhybek ve specifickém rozšíření pro DSFD

Způsob popisu a vkládání výhybek se v rámci navrženého SR liší od existujících rozšíření vyšší úrovně automatizace vkládání i určitými rozdíly v rámci topologického popisu výhybky. Toto zvýšení automatizace snižuje uživatelskou náročnost tvorby topologie a geometrie výměnou za mírné zvýšení komplexity modelu. Z topologického hlediska existují v DSFD 3 základní typy výhybkových konstrukcí:

- Jednoduchá výhybka
- Křižovatková výhybka
- Prosté křížení

Jednoduché výhybky jsou dále děleny na jednoduché výhybky a výhybky obloukové, které se liší z hlediska geometrického popisu, ale z topologického hlediska a z hlediska navázání na předcházející a následující úseky koleje jsou shodné s výhybkami jednoduchými.

Všechny výhybkové konstrukce jsou popisovány systémem odvozeným od nejsložitější z vyskytujících se konstrukcí, tj. úplnou křižovatkovou výhybkou; další výhybkové konstrukce se tvoří vypouštěním částí této konstrukce, přičemž zachované části si zachovávají popis shodný s úplnou křižovatkovou výhybkou. Úplná topologie s naznačenou topologií jednoduché výhybky je uvedena v Obr. 4.



Obrázek 4: Topologické schéma úplné křižovatkové výhybky. Modrou barvou jsou vykresleny části sdílené jednoduchou výhybkou a úplnou křižovatkovou výhybkou, červeně pak zbývající části úplné křižovatkové výhybky. Označené body jsou z hlediska popisu důležité, protože v nich dochází k napojení jednotlivých částí konstrukce na následující úseky koleje, nebo zde dochází k odbočení nebo křížení jednotlivých částí výhybkové konstrukce. Zdroj autor.

Tento návrh byl použit z důvodu standardizace popisu konstrukce a umožnění konzistentního navázání dalších kolejí na výhybkou konstrukci, stejně jako navázání dalších prvků výstroje výhybkové konstrukce na daný topologický popis.

V porovnání s jinými rozšířeními existujícími pro VMŽI umožnil způsob popisu návrh specifických funkcí umožňujících uzavírání instancí třídy **LinearNetElement**, určení průjezdnosti na jednotlivých stycích **LinearNetElement** s pomocí instancí třídy **PositionedRelation** a v případech, kdy toto dává smysl připojit k instancím třídy **PositionedRelation** instance třídy **SwitchMachine** (přestavník).

4.3.1 Popis jednoduché výhybky

Popis jednoduché výhybky se z hlediska topologie liší od popisu používaných u jiných rozšíření připojením úseku před jazyky (krátkého úseku mezi začátkem výhybky a začátkem jazyků, viz úsek **A-AR** na Obr. 4). Tento popis byl zvolen z důvodu snazšího zaměření dalších prvků modelu, protože kolejnicové styky jsou v kolejišti DSFD dobře patrné a zvláště na zhlavích stanic slouží jako orientační prvek a bylo žádoucí, aby začátek výhybkové konstrukce byl ztotožněn s jedním z těchto styků. Dalším důvodem byla zvýšená automatizace tvorby modelu.

Topologický popis

Z topologického hlediska je výhybka tvořena úsekem před jazyky, přímou a odbočnou větví, které se setkávají v bodě označovaném **AR**. Tyto úseky jsou v tomto bodě propojeny třemi vazbami s pomocí instancí třídy **PositionedRelation**, určující možnost směru pojezdu na dané konstrukci, dvě určují pojízditelnost výměny z a do odbočky po i proti hrotu. Třetí pak určuje nemožnost pojezdu výhybky ve směru z odbočky (po hrotu) do přímého směru (proti hrotu) a naopak. K instancím třídy **PositionedRelation** určující možnost obousměrného pojezdu v přímém směru a do odbočky je připojena instance třídy **SwitchMachine** (přestavník).

Popis geometrie

Jednoduchá výhybka je po prostém křížení nejjednodušší výhybkovou konstrukcí na kolejišti DSFD. Společná část je tvořena přímou kolejí, stejně tak i přímá větev. Odbočná větev je tvořena obloukem bez přechodnic a krátkým přímým úsekem na konci výhybkové konstrukce. Pro tvorbu modelu je jednoduchá výhybka popsána **délkou úseku před jazyky**, **délkou tečny oblouku odbočné větve** (spojnice pomyslného bodu **X** a konce oblouku), **stavební délkou výhybky** (vzdálenost od počátečního kolejnicového styku před jazyky až po kolejový styk na konci přímé větve) a **úhlem odbočení**. Výkres tohoto popisu je uveden v Obr.: 5.



Obrázek 5: Výkres geometrického popisu jednoduché výhybky používané na kolejišti DSFD. Zdroj autor.

4.3.2 Popis obloukové výhybky

Popis obloukové výhybky je z hlediska topologie téměř shodný s popisem jednoduché výhybky, a proto není uveden jako samostatná část popisu této výhybky. Jedinou změnou je změna polohy instance třídy **GeoPoint** z odbočné větve přímé výhybky na přímou větev obloukové výhybky. Tato změna je vynucena čistě geometrií výhybky.

Popis geometrie

Oblouková výhybka v SR pro DSFD nevzniká, na rozdíl od skutečného světa, z jednoduché výhybky pomocí transformace, ale vzniká z prototypu vytvořeného na základě potřeb popisu sálu. Tento prototyp je kompromisem mezi potřebami popisu (zachování osové vzdálenosti na konci výhybky, odbočného úhlu, použití křivek definovatelných ve VMŽI, hladkým napojení jednotlivých křivek na sebe apod.) a skutečným stavem v kolejišti, kdy nepřesnostmi při stavbě a dlouhodobým působením sil v kolejišti dochází ke vzniku deformací a zlomů v napojení jednotlivých dílů koleje. V prostředí DSFD se nachází 2 typy obloukových výhybek, menší a větší, obě však mají shodný geometrický popis a liší se pouze konkrétními velikostmi jednotlivých parametrů popisu.

Společná část obloukové výhybky (úsek před jazyky) je tvořena obloukem o poloměru odbočné větve o úhlu odbočení 1° . Na tento v odbočné větvi navazuje oblouk o stejném poloměru. V přímém směru na krátký oblouk ve společném úseku navazuje přímý úsek a následně oblouk přímého směru. V rámci modelu je tvar výhybky popsán pomocí **odbočného úhlu úseku před jazyky**, **délky velké tečny**, což je přímka spojující osu koleje v místě napojení a konce oblouku odbočné větve, přičemž tato přímka je vedena pod úhlem odbočení (30° - Na nákresu v Obr.: 6 je pro snadnější orientaci ozn. $Tv.$), parametr ozn. jako **délka malé tečny**, což je přímka spojující prodloužení přímého úseku na přímé větvi výhybky (vedená v úhlu 1° vůči ose původní koleje v místě napojení výhybky) s koncem oblouku přímé větve vedená pod úhlem odbočení vůči ose původní koleje v místě napojení (délka tzv. malé tečny je pro snadnější orientaci na výkresu v Obr.: 6 ozn. Tm), dále **šikmý posun**, což je délka přímé koleje v přímé větvi výhybky (na výkresu v Obr.: 6 označena S) a posledním parametrem je **úhel odbočení**. Výkres tohoto popisu je uveden v Obr.: 6.



Obrázek 6: Výkres geometrického popisu větší z dvojice prototypů používaných pro popis obloukových výhybek na DSFD. Tm označuje délku malé tečny, Tv označuje délku velké tečny a S označuje délku šikmého posunu (délka přímé koleje v přímé větvi výhybky). Zdroj autor.

4.3.3 Popis křižovatkové výhybky

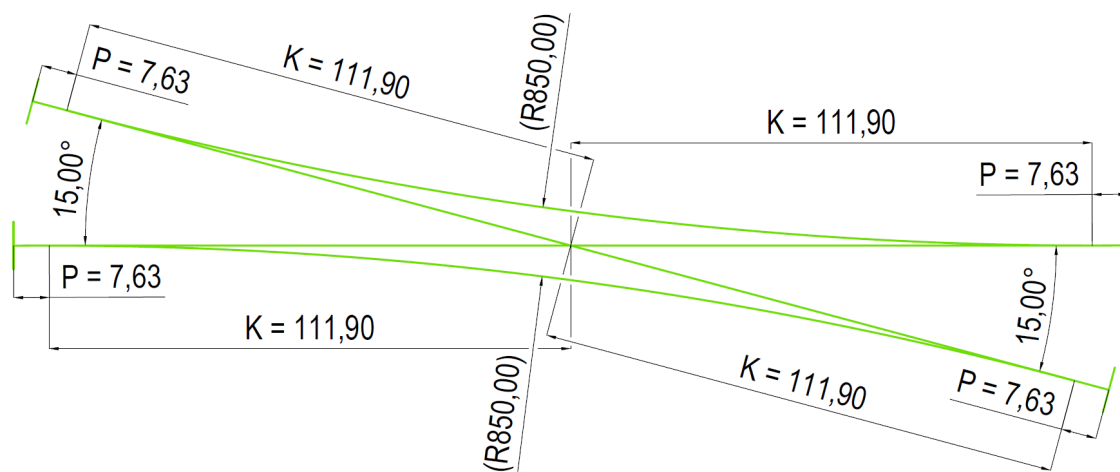
Topologický popis

Z hlediska topologického popisu je (úplná) křižovatková výhybka nejsložitějším útvarem vyskytujícím se na DSFD. Schéma topologického popisu křižovatkové výhybky je uvedeno v Obr.: 4. Na začátku každého z možných vjezdů do křižovatkové výhybky se nachází krátký úsek před jazyky (např. **A - AR**, na něj vždy navazuje jedna z dvojice odbočných větví křižovatkové výhybky (např. **AR - DR** a jeden přímý úsek vedoucí do bodu křížení (např. úsek **AR - X**). V každém z bodů **AR**, **BR**, **CR** a **DR** jsou jednotlivé instance **LinearNetElement** propojeny pomocí **PositionedRelation** stejným způsobem, jako u jednoduché výhybky. V bodě křížení (bod **X**) je vytvořeno šest instancí

třídy **PositionedRelation** tak, že dvě z nich definují sjízdnost daného křížení z každého směru přímým směrem a další dvě definují nesjízdnost křížení do odbočky a zbylé dvě definují nesjízdnost křížení do druhé větve stejné části křižovatkové výhybky (např. směr jízdy **A - B**). K **PositionedRelation** v bodech **AR**, **BR**, **CR** a **DR** jsou podobně jako u jednoduché výhybky připojeny instance třídy **SwitchMachine**, které jsou navíc propojeny pomocí třídy **NetEntityRelation** s pomocí vztahu **aIsPartOfB** tak, aby reprezentovaly jeden skutečný přestavník (jsou vždy propojovány přestavníky **AR + BR** a **CR + DR**).

Popis geometrie

Úsek na začátku každé z větví křižovatkové výhybky je tvořen krátkým úsekem přímé koleje (ve výkresu geometrické konstrukce je označen P), na jehož konci uvnitř konstrukce na něj navazují přímé úseky spojující tento bod s bodem křížení a oblouk odbočné větve, který je vždy (na rozdíl od přímých úseku u křížení) společný pro obě větve, které spojuje. V rámci modelu je křižovatková výhybka geometricky popsána pomocí následujících parametrů: **délka úseku před jazyky** (ve výkresu ozn. P), **délka od počátku jazyků po křížení** (ve výkresu ozn. K) a **úhel odbočení**.



Obrázek 7: Výkres geometrického popisu křižovatkové výhybky používané na DSFD. K označuje úsek mezi odbočením **odbočné větve** a bodem křížení (tento úsek zároveň reprezentuje tečnu k oblouku odbočné větve) a P označuje úsek před jazyky. Zdroj autor.

4.3.4 Popis kolejového křížení

Nejjednodušším prvkem vycházejícím topologicky z výhybkové konstrukce je prosté křížení. Jako jediná tato konstrukce neumožňuje změnit směr jízdy vlaku a nachází se na ní pouze jediný bod křížení. Navrhovaný popis zcela odebrává body odbočení (v Obr. 4 označených písmeny **R**) a vyskytují se na ní pouze body pro připojení navazujících kolejí (v Obr. 4 označených písmeny **A, B**, apod.) a bod křížení (v Obr. 4 označen písmenem **X**).

Z hlediska topologického i geometrického popisu se jedná o křižovatkovou výhybku bez odbočných větví, úseků před jazyky a výše zmíněných bodů **R** (např. **AR**) a s tím spojených absencí instancí tříd **PositionedRelation** a **SwitchMachine**, které se v těchto bodech lokalizují. Z hlediska geometrie je kolejové křížení popsáno pouze parametry **úhel odbočení** a **délka od počátku jazyků po křížení** (z důvodu konzistence je název parametru zachován shodně s křižovatkovou výhybkou, přestože se v dané konstrukci jazyky vůbec nevyskytují).

4.4 Popis řešení návěstidel ve specifickém rozšíření pro DSFD

V rámci tvorby specifického rozšíření pro potřeby DSFD byla dána zvýšená pozornost popisu návěstidel, přesněji jejich polohy a vlastností z hlediska provozního i z hlediska funkce zabezpečovacího zařízení. V neposlední řadě bylo řešeno jejich vkládání do modelu v nástroji pro tvorbu, při jehož úpravě bylo dbáno na to, aby vkládání návěstidel bylo v rámci možností co nejjednodušší a nejintuitivnější pro běžného uživatele. V rámci popisu byly řešeny i vlastnosti návěstidel z hlediska tvorby 3D prostřední pro potřeby simulátoru, tj. relativní poloha vůči koleji a druh jejich upevnění (stožár, krakorec apod.).

Návrh popisu vychází z existujících struktur navržených v rámci existujících SR, které byly dále upraveny pro potřeby DSFD. Struktura popisu (atributy jednotlivých tříd sloužících popisu, jejich propojení je uvedeno ve schématu v Příloze 1. v Obr.: 6. V rámci přípravných prací na tvorbu SR došlo k inventarizaci existujících návěstidel v kolejišti DSFD a na základě zjištěných návěstidel byly navrženy kategorie návěstidel, které byly zavedeny do třídy **SignalType**, a které se tak automaticky (po souhlasu uživatele) generují při prvním spuštění projektu v navrhovaném specifickém rozšíření. Kategorie jsou volně odvozeny od kategorizace návěstidel v předpisu SŽ D1 [18], ale vzhledem k absenci mnoha typů návěstidel na DSFD jsou vybrané kategorie sloučeny, popř. dále rozděleny, aby lépe odpovídaly požadavkům na popis návěstidel na DSFD. Kategorie návěstidel zavedené v rámci třídy **SignalType** jsou:

1. **hlavní** - popisují hlavní návěstidla bez permissivního významu návěsti stůj fyzicky přítomná v kolejišti.
2. **hlavní_virtuální** - popisují hlavní návěstidla bez permissivního významu návěsti stůj, která jsou implementována pouze v logice řízení sálu a souprav.
3. **předvěst** - popisují samostatné předvěsti.
4. **seřad'ovací** - popisují seřad'ovací návěstidla.
5. **autoblok** - popisují hlavní návěstidla s permissivním významem návěsti stůj fyzicky přítomná v kolejišti.
6. **autoblok_virtuální** - popisují hlavní návěstidla s permissivním významem návěsti stůj, která jsou implementována pouze v logice řízení sálu a souprav.
7. **přejezdník** - popisují přejezdníky.
8. **rychlostník** - popisují rychlostníky.
9. **posun_pevné** - popisují neproměnná návěstidla platná pouze pro posun.
10. **námezník** - popisují námezníky, hraničníky a koncovníky.
11. **doplňkové** - popisují návěstidla, která jsou zpravidla doplňkem jiných návěstidel, nebo jiné potřebné návěstidla nezařaditelná do jiné z předchozích skupin.

Každé z výše uvedených kategorií je přiřazeno číslo reprezentující tuto kategorii ve všech dalších třídách a funkcích (shodné s číslem v seznamu) a typ konstrukce návěstidla dle railML [19] (z tohoto důvodu jsou v rámci výše uvedeného seznamu rozdělena fyzická a virtuální návěstidla) v parametru **signalConstructionType**.

Popis konkrétního návěstidla v infrastruktuře je proveden s pomocí třídy **TracksideSignal**, která obsahuje atributy ID, následně přiřazení instance třídy **SignalType** definující typ a charakter návěstidla, na základě přiřazení k typu je následně omezen výběr rozložení (podtypu) návěstidla. V rámci tohoto atributu je formou předdefinovaných textových řetězců uloženo rozložení světél u světelných návěstidel ve směru od nejvýše umístěného až po nejnižší umístěné světlo, v případě neproměnných návěstidel je v této kategorii uložen podtyp nebo určení konkrétního návěstidla v kategoriích (např. v typu **námezník** se zde určuje, zdali jde o námezník, hraničník, nebo koncovník). Pro uchování rozložení návěstních světél se používá zkratk odvozených od prvního písmene barvy světla, velkého písmene pro běžnou kulatou svítlnu, malého písmene pro indikační pruh rychlostní návěstní soustavy (běžné hlavní návěstidlo na DSFD nabývá rozsahu **ŽŽČBŽ** + **zžz**, což odpovídá návěstním světlům v pořadí žlutá, zelená, červená, bílá a žlutá, doplněné zeleným, žlutým a zeleným indikačním pruhem v tomto pořadí).

Třída **TracksideSignal** dále obsahuje atributy **label**, který slouží k zadání jména návěstidla (např. Lc3a) (v případě neproměnných návěstidel vyjadřujících určitou hodnotu (např. rychlostníky) je tato hodnota uváděna v tomto atributu), dále **defaultIndication** popisující základní indikaci návěstidla, přičemž možnosti výběru při tvorbě byly vytvořeny na základě provedené inventarizace. Pro popis světelných návěstidel jsou k dispozici nabídky svítlen, v tomto případě bez omezení výběru typu návěstidla, opět popsány stejným způsobem jako u výběru rozložení svítlen. Pro neproměnná návěstidla je rezervována hodnota **N**, která zároveň slouží u světelných návěstidel k vyjádření v základním stavu zhaslého návěstidla. Posledním parametrem v rámci třídy **TracksideSignal** je atribut **signalMountingType**, který popisuje způsob upevnění návěstidla, opět formou textového řetězce, který je při tvorbě vybírán z předem definovaných možností.

Světelná návěstidla typů 1 - 4 budou přiřazována konkrétní dopravně, popsané třídou **ControlArea**, a ovládacímu stanovišti, popsaném třídou **SignalBox** (existuje-li v dané dopravně více stanovišť umožňující ovládat návěstidla), s pomocí přiřazovací třídy **NetEntityRelation**. Návěstidla autobloku se automaticky považují za vlastní dopravnu a nebudou přiřazovány vlastní instancí třídy **ControlArea**. V případě nutnosti doplnit návěstidlo o indikátor, nebo jiné návěstidlo z typu **doplňkové**, bude toto provedeno pomocí stejné přiřazovací třídy. Dále budou tímto způsobem připojeny samostatné předvěsti předvěstěným hlavním návěstidlům.

4.5 Funkce vytvořené v rámci usnadnění tvorby modelu v rámci specifického rozšíření

V rámci tvorby specifického rozšíření byly vytvořeny funkce obsluhující specifické třídy implementované v rámci tohoto rozšíření, nebo které zjednodušují tvorbu modelu zvýšením automatizace tvorby (např. výhybkové konstrukce). V rámci tvorby specifického rozšíření byly pro jeho obsluhu v nástroji pro tvorbu VMŽI do tohoto nástroje doplněny následující funkce.

Seznam funkcí doplněných do nástroje pro tvorbu VMŽI pro potřeby specifického rozšíření pro potřeby DSFD:

- `VlozitHorizontalCurveProJednoduchouVyhybkuDSFD`
- `VlozitHorizontalCurveProObloukovouVyhybkuDSFD`

- VlozitHorizontalCurveProKrizovatkovouVyhybkuDSFD
- VlozitHorizontalCurveProKrizeniDSFD
- PridatPrestavnikDSFD
- ZalozitPreddefinovaneObjektyProDSFD2023
- ZalozitSignalTypeDSFD
- VlozitHorizontalCurveProVyhybkyDSFD
- PriraditCastNastupniHranyNastupniHrane

V příloze 2. je uveden popis těchto funkcí z důvodu snadnějšího seznámení se případného uživatele s nástroji vytvořenými v rámci tvorby SR pro DSFD a dále pro případ nutnosti údržby vytvořených nástrojů pro tvorbu SR v důsledku změn jádra VMŽI, resp. jádrové části nástroje pro tvorbu VMŽI a s tím spojených změn v navázání funkcí (jiné vstupní parametry těchto funkcí apod.).

5 Metoda získání dat pro potřeby modelu a jejich vkládání do modelu

V rámci přípravných prací pro tvorbu modelu kolejiště DSFD byl proveden návrh způsobu vytváření podkladů, který vychází z existujících podkladů a nástrojů dostupných na DSFD. Navržená metoda se sestává ze dvou hlavních kroků: krok **sběr dat o kolejišti** (v prostředí běžně používaných tabulkových editorů a CAD nástrojů) a krok **vkládání dat do VMŽI v prostředí nástroje pro tvorbu**, každý dále dělený na dílčí úkony, z nichž některé jsou volitelné, jejichž provedením se zvyšuje rychlost a komfort při finální tvorbě modelu. Navržená metoda v určité míře využívá měřících prací na kolejišti provedených v roce 2016 a dokumentace z dob výstavby DSFD a přímo navazuje na měřící práce provedené v roce 2019, přičemž v oblasti měření a sběru dat v kolejišti vychází z měřících metod použitých při tomto měření.

5.1 Sběr dat o kolejišti DSFD

Prvním krokem při tvorbě datového popisu kolejiště DSFD je sběr dat přímo v kolejišti a v existujících materiálech. Tento krok umožňuje využití existujících podkladů, které jsou zpravidla uloženy v běžně používaných editorech, CAD aplikacích apod., a dále tyto editory umožňují úpravu již zadaných dat, což v případě nástroje pro tvorbu VMŽI není ve většině případů možné. Z těchto důvodů je vhodné připravit veškeré podklady pro tvorbu ještě před zahájením vkládání dat do VMŽI, vč. ověření správnosti a přesnosti. Sběr dat obsahuje následující dílčí kroky:

1. Měření a popis geometrie kolejiště v prostředí běžného editoru CAD. Tento krok se pro každý měřený úsek rozpadá na 3 další dílčí kroky:
 - (a) Měření a popis geometrie koleje úseku ve 2D
 - (b) Popis podélného sklonu úseku
 - (c) Transformace vzdáleností ze 3D do 2D
2. Sběr dat o prvcích infrastruktury jako jsou návěstidla, nástupiště, přejezdy, apod. a jejich popis v prostředí tabulkového editoru. (volitelný úkon)
3. Zaměření prvků infrastruktury do připravené geometrie v prostředí běžného editoru CAD. (V případě jednodušších úseků lze spojit s bodem 1.)
4. Nezávislá kontrola správnosti získaných informací. (volitelný úkon)

5.1.1 Měření a popis geometrie kolejiště

Měření a popis geometrie koleje úseku ve 2D

Prvním dílčím krokem popisu kolejiště je zaměření a zakreslení geometrie koleje v editoru CAD. Při tomto kroku je vhodné dávat prioritu přesnému popisu délky úseků nad jinými vlastnostmi koleje. Toto vychází z požadavků kladených na model, zvláště požadavky popisu pro RBC ETCS a IZZ. Měření je navrhováno ruční s použitím běžných pomůcek, jako jsou pevná a ohebná pravítka, tužka, olovnice a libela. Popsaný způsob měření je nutno aplikovat postupně pro každý změřený úsek. V rámci provedeného zkušeb-

ního měření bylo zjištěno, že oblouky na kolejišti DSFD nemají přechodnice a změny sklonu úseku jsou bez zaoblení. Navrhovaná metoda toto respektuje a způsob měření těchto prvků není zaveden.

Měření v přímém úseku je prováděno s pomocí pevného pravítka, samotné měření je vždy prováděno po jedné ze dvojice kolejnic, následné měření po druhé kolejnici slouží pro kontrolu. Z důvodu ne zcela přesně spojených styků kolejí, drobných deformací kolejí způsobených léty opotřebení apod. je možné, že se naměřené hodnoty délky jednotlivých kolejnic mohou rozcházet i několik milimetrů. Dosáhne-li odchylka 2 mm je vhodné měření zopakovat. Pokud je výsledek měření stejný, je nutné vypočítat aritmetický průměr (dle vztahu (4) uvedeného níže) délky kolejnic, a až ten následně zakreslit do výkresu jako délku koleje.

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^n x_i, \quad (4)$$

kde x_i jsou jednotlivá měření a n je počet měření.

Měření v oblouku je prováděno v obou kolejnicích zároveň s pomocí ohebného pravítka po straně kolejnic blíže k ose koleje a výsledná délka osy koleje je opět aritmetický průměr obou délek kolejnic (vztahu (4)). Toto platí vzhledem k linearitě závislosti délky kružnicového oblouku na poloměru:

$$l = \frac{\pi \cdot r \cdot \alpha}{180}, \quad (5)$$

kde l je délka oblouku v mm r je poloměr v mm a α je úhel oblouku ve stupních. Pro zakreslení oblouku v SW CAD je vedle počátečního bodu nutno znát i poloměr a úhel oblouku. Jako zdroj úhlu oblouku je doporučeno využít existujících podkladů, které sice nejsou přesné z hlediska vzdáleností, ale úhly jednotlivých oblouků jsou popsány přesně. Úpravou vztahu (5) lze získat vztah pro výpočet poloměru daného oblouku:

$$r = \frac{l \cdot 180}{\pi \cdot \alpha}, \quad (6)$$

kde r je výsledný poloměr v mm, l je délka oblouku v mm a α je úhel oblouku ve stupních. Na základě znalosti počátečního bodu, úhlu oblouku a poloměru lze již oblouk snadno zakreslit.

Zaměření výhybkových konstrukcí je velmi jednoduché. V rámci přípravy na tvorbu SR byly vytvořeny bloky odpovídající jednotlivým typům výhybkových konstrukcí užívaných na DSFD. K jejich zavedení do modelu v prostředí CAD postačí mít definice bloků, znát bod připojení (konec předchozího úseku) a identifikovat danou konstrukci a na základě ní vybrat správný blok pro vložení. Vzhledem k nízkému počtu typů výhybkových konstrukcí a zřetelnému rozdílu mezi nimi (a to vč. obloukových výhybek) je navrhována identifikace pohledem, popř. zběžným přeměřením.

Popis podélného sklonu úseku

Dalším dílčím krokem popisu je získat popis podélného sklonu. Nejvýhodnějším způsobem zaměření výšky pro popis kolejiště DSFD je přímé měření bodů změny podélného sklonu vůči pevné podložce na níž kolejiště spočívá. Body změny podélného sklonu lze nalézt pomocí krátkého pevného pravítka umístěného na temeno kolejnice. V případě zvýšení sklonu (údolnicový oblouk) se pravítko bude dotýkat pouze na svých okrajích, v případě snížení sklonu (vrcholový oblouk) se bude dotýkat pouze na jedné straně, avšak

větší částí své délky. Body změny sklonu se zpravidla nacházejí v, nebo velmi blízko kolejnicových styků. Nalezení přesného bodu je vhodné provádět postupným posouváním pravítka postupně po obou kolejnicích z obou stran a sledování prvního bodu narušení dotyku celé plochy pravítka s kolejnicí a následné označení tohoto bodu v koleji. Tento postup je vhodné několikrát opakovat a z výsledků následně vypočítat aritmetický průměr (vzdáleností jednotlivých bodů vůči referenčnímu bodu). Výška bodu se měří pravítkem podél olovnice (nebo jiného nástroje pro zjištění vertikální roviny) vůči referenční hladině, kterou může být buď pevný objekt v kolejišti, u nějž byla potvrzena jeho vodorovnost vůči desce nesoucí kolejiště, která v celém rozsahu kolejiště slouží jako referenční hladina 0, nebo vůči rovině definované libelou.

Zaznamenání podélného sklonu do prostředí CAD je možné provést dvěma způsoby. Jednodušším a z hlediska vkládání dat do VMŽI vhodnějším způsobem je přímé označení výšky konkrétního bodu vůči referenční rovině (desce nesoucí kolejiště). Tento postup je však vhodný pouze v místech, kde je možné tuto výšku změřit. V oblastech, kde toto možné není je nutno uvádět sklon mezi dvěma body v promilích (avšak pro potřeby následujícího úkonu je vhodné zaznamenat i absolutní rozdíl výšek počátečního a koncového bodu úseku). Tento sklon vypočteme dle vztahu:

$$s = \frac{h \cdot 1000}{l}, \quad (7)$$

kde h je rozdíl výšek měřených bodů a l je jejich vzdálenost (všechny hodnoty musí být dosazeny ve stejných jednotkách). Doporučeným způsobem označení bodů změny podélného sklonu a těchto sklonů do výkresu je vložení dvojice sklonovníků do každého z bodů změny sklonu.

Transformace vzdáleností ze 3D do 2D

Posledním dílčím krokem při tvorbě výkresu geometrie koleje je provedení transformace všech vzdáleností z trojrozměrného prostředí do vodorovné plochy. Při měření délky v koleji v podélném sklonu se naměřená vzdálenost ve skutečnosti nachází na přeponě pravoúhlého trojúhelníku (jeho odvěsny pak tvoří rozdíl výšek začátku a konce úseku a délka úseku v ploše - požadovaná hodnota). Přepočet délek do plochy je požadován, protože výpočet délek kolejí ve VMŽI vzniká na základě popisu podélného sklonu a délek ve 2D opačnou transformací. Z tohoto důvodu je nutno přepočítat každou délku změřenou v podélném sklonu pomocí vztahu

$$l_h = \sqrt{l_m^2 - v^2}, \quad (8)$$

kde l_h je požadovaná délka ve vodorovné rovině, l_m je změřená délka a v je rozdíl výšek mezi počátkem a koncem měřeného úseku (všechny hodnoty musí být dosazeny ve stejných jednotkách). V případě oblouku je nutno provést nový výpočet poloměru na základě takto upravené délky a nové vykreslení. V případě výhybkových konstrukcí se tato oprava neprovádí. Stejně není nutno provádět tuto opravu v případě velmi malých sklonů (do deseti promile) na kratších úsecích (do jednoho metru). V případě výhybkové konstrukce je možno chybu zanedbat z důvodu malé délky konstrukce a v případě velmi malých sklonů je rozdíl opravené a neopravené délky velmi malý (pro uvedené mezní hodnoty je chyba 0,05 mm, což je v porovnání s chybou ručního měření (na stejné vzdálenosti) minimálně 0,5 mm zanedbatelné).

5.1.2 Sběr dat o prvcích infrastruktury

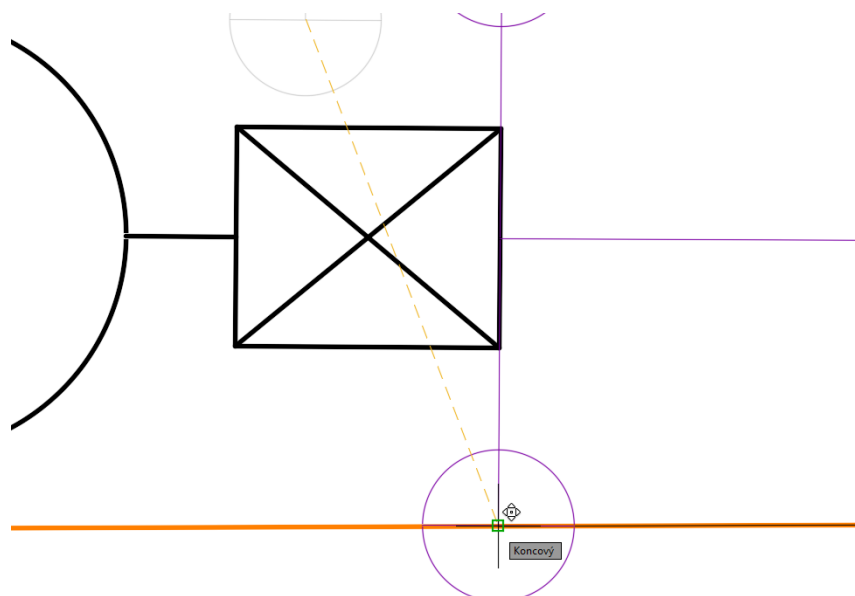
Dalším dílčím krokem navrhované metody je sběr informací o jednotlivých prvcích infrastruktury a jejich ukládání do prostředí tabulkového editoru. Tento krok vznikl na

základě zkušeností z měření z roku 2019, v jehož průběhu byly vybrané informace (např. název) vkládány přímo do výkresu v prostředí CAD formou popisku u vybraných prvků infrastruktury, což u komplikovanějších částí infrastruktury vedlo k nepřehlednosti. Z toho důvodu je doporučeno před zahájením tvorby samotného VMŽI vytvořit strukturovanou tabulku, do níž jsou zanášeny informace o prvcích infrastruktury odpovídající jejich datovému popisu v navrhovaném SR jednotlivě pro každý objekt, které by měly být dále rozděleny podle typu objektu a příslušnosti k dopravně/traťovému úseku na němž se nacházejí. Vhodnost realizace tohoto kroku vyplývá ze složitěho způsobu změny už jednou zadaných informací, a je proto vhodné provést revizi zjištěných informací v přehlednější a snadno opravitelné formě ještě před zahájením vkládání dat do VMŽI. Tento dílčí krok je však potenciálně časově náročný a v určité míře duplikuje práci následujícího kroku. Proto je možné v případě dobré znalosti kolejiště popisující osobou a přímého přístupu ke zdrojům informací nebo kolejišti samotnému v průběhu tvorby samotného modelu v nástroji pro tvorbu VMŽI tento dílčí krok vynechat, popř. jej omezit pouze na vybrané prvky infrastruktury se složitější datovou strukturou.

5.1.3 Zaměření prvků infrastruktury do připravené geometrie v prostředí běžného editoru CAD

Dalším krokem ve tvorbě podkladů pro tvorbu VMŽI je zakreslení prvků infrastruktury do výkresu s geometrií. Postupy zaměřování jsou shodné jako v případě tvorby geometrické části výkresu, a proto zde nejsou znovu uvedeny (pro postup viz kap.: 5.1.1). Vzhledem ke shodnosti postupu je možné (zvláště v případě úseků s nízkou složitostí) provést tento dílčí krok spolu se zaměřováním geometrie.

V rámci přípravných prací byla definována řada bloků pro evidenci jednotlivých zaznamenaných prvků infrastruktury, obsahujících prvky pro jejich přesnou a snadnou lokalizaci v rámci tvorby VMŽI v nástroji pro jeho tvorbu. V rámci návrhu postupu byl navržen následující postup přesné lokalizace (prvku jehož polohu jsme již zaměřili): po vložení bloku korespondujícího s daným typem prvku infrastruktury do výkresu je doporučeno vytvořit kopii části geometrie, na kterou má být objekt navázán, tuto kopii opatřit délkovou kótou odpovídajícího typu (šikmou, obloukovou apod.) s vysokým počtem desetinných míst a následně tuto kopii pomocí dostupných příkazů zkracovat (se zachováním ostatních geometrických vlastností), dokud nebude její délka odpovídat naměřené hodnotě. Následně blok přesunout za střed lokalizační značky tak, aby se tato značka nacházela v koncovém bodě upravené kopie geometrie koleje (s výhodou lze k tomuto využít funkci přichycení). Pokud je to u daného prvku nutné, umístění prvku musí respektovat stranu a směr působení (reprezentace by měla být otočena tak, jako by vznikla pomyslným sklopením **ve** směru jízdy vlaku, pro který platí). V případě atypické pozice daného prvku od osy koleje je vhodné poznačit vzdálenost středu prvku od osy koleje a výšku bodu, který je běžně v úrovni temene kolejnice, proti úrovni temene kolejnice, vedle lokalizovaného bloku. Nakonec je provedeno odstranění vytvořené kopie geometrie. Správně umístěnou lokalizační značku lze vidět v Obr.: 8.



Obrázek 8: Ukázka správného využití lokalizačních prvků bloků navržených pro tvorbu výkresu sloužícího jako podklad pro tvorbu VMŽI. Zdroj autor.

5.1.4 Kontrola správnosti získaných informací

Je vhodné aby součástí tvorby přípravy pro tvorbu VMŽI byla i kontrola tvořených podkladů. V průběhu tvorby může být prováděna vizuální kontrola tvořeného výkresu i ověření informací vkládaných do tabulky popisu prvků. Ověření změřených vzdáleností je možno provést s pomocí provozované vlakové soupravy, přesněji odometrie na ní instalované. V případě neshody výsledku je doporučeno provést v daném úseku nové ruční měření i nové měření s pomocí odometrie s jinou soupravou. V případě, že nové ruční měření potvrdí původní hodnoty, je doporučeno považovat výsledek tohoto měření za správný.

5.2 Metoda vkládání dat do VMŽI

Druhým hlavním krokem je tvorba samotného VMŽI v prostředí nástroje pro jeho tvorbu. Postup tohoto hlavního kroku vychází z logiky nástroje pro tvorbu VMŽI. Prvním dílčím krokem je provedení importu výkresu do prostředí SW Rhinoceros, což lze provést pomocí otevření výkresu v tomto prostředí a následném uložení tohoto výkresu ve vlastním formátu SW. Přitom je vhodné, aby se výkres nacházel mimo okolí bodu 0,0, a to z důvodu přítomnosti oblasti pro ukládání systémových proměnných nástroje pro tvorbu VMŽI v této oblasti. Následně je nutné zkontrolovat a případně zajistit změnu měřítka výkresu tak, aby měřítko výkresu bylo 1:1. Toho lze docílit pomocí nástroje změny měřítka a využití známé délky přímého elementu a ručním vytvořením referenční přímky, nebo lze měřítko změnit až po založení projektu (viz dále) v nástroji pro tvorbu s využitím v nástroji předdefinované geometrické reprezentace, např. vytvoření jednoduché výhybky se stejným směrem a bodem začátku a následným upravením měřítka podle tohoto útvaru.

Založení projektu probíhá otevřením souboru nástroje v editoru skriptů SW Rhinoceros a jeho spuštěním. Při prvním spuštění dojde k vytvoření a nadefinování systémových proměnných a založení struktury vrstev, do nichž jsou jednotlivé objekty vkládány, a které

svojí strukturou kopírují databázi VMŽI. V průběhu zakládání bude uživatel dotázán na zadání názvů proměnných a popř. jejich hodnot. Pro potřeby projektu v SR pro DSFD je vhodné využít přednastavených hodnot pro většinu parametrů. U následujících hodnot je však nutno zvolit jinou hodnotu: **modelExtensionSubset** - volba **proDSFD2023** (zvolí SR pro DSFD), **LevelNetworkAssignment** - atribut **dimension** - volba **xyz** (zvolí možnost vygenerování 3D reprezentace geometrie na základě výškových dat). V případě, že není nabídnuta žádná přednastavená hodnota, je možné doplnit libovolný textový řetězec. Na konci zakládání je uživatel vyzván k výběru, jestli požaduje založení instancí třídy **SignalType**. Po proběhnutí celého vytvoření projektu se vedle kurzoru myši objeví základní nabídka nástroje pro tvorbu pojmenovaná Zvol základní skupinu operací. Volbou storno dojde k řádnému ukončení běhu nástroje a uložení provedených změn. Správné založení projektu lze zkontrolovat podle přítomnosti bodů okolo nastaveného bodu počátku a existence vrstev reprezentující třídy jádra i SR VMŽI. V této fázi je vhodné, pokud je uživatelem nástroje začátečník, aby došlo k provedení části tvorby nanečisto v jejímž průběhu se uživatel seznámí se způsobem a možnostmi vkládání jednotlivých objektů (viz níže), způsoby opravy chyb apod. (viz níže), nejlépe za přítomnosti osoby seznámené s funkcemi nástroje.

Před zahájením tvorby je důležité, aby byl uživatel seznámen se specifiky nástroje, které jsou dány charakteristikami a úrovní vývoje jádra nástroje (tento stav je společný pro všechny SR a nesouvisí přímo s obsahem implementovaným v rámci této práce). Veškeré úkony prováděné v rámci nástroje jsou ukládány v průběhu jeho běhu, avšak jsou SW Rhinoceros chápány jako jeden atomický úkon a je s nimi tak nakládáno (např. při použití tlačítka zpět je odstraněn celý poslední běh nástroje bez ohledu na počet operací provedených v rámci tohoto běhu). Dále je nutné, aby byl uživatel seznámen s nedostatky nástroje a s tím spojenými problémy. V případě zadání nesprávných hodnot při tvorbě objektu, nesprávného ukončení běhu nástroje, nebo neobvyklého běhu funkce je jedinou možností odstranění chyby ukončení běhu nástroje a odstranění posledního běhu pomocí volby **zpět** v SW Rhinoceros. V případě, že při nesprávném ukončení běhu nástroje ke zrušení běhu nedojde, může při dalším běhu nástroje dojít k nestandardnímu běhu nástroje a poškození tvořeného modelu. Z důvodů výše uvedených je vhodné, aby uživatel pravidelně (po provedení přibližně 3-5 jednoduchých a před každou komplexnější akcí) běh nástroje ukončoval, čímž vytvoří zálohu, k níž se lze v případě chybného zadání hodnoty nebo nesprávného běhu nástroje vrátit.

Dalším dílčím krokem je převedení výkresu geometrie z podkladů do prostředí VMŽI. V rámci nástroje je doporučeno k tomuto využívat volby v sekci ucelená tvorba sítě → micro 2D → vkládání instancí HorizontalCurve s možností uzavření v LinearElement. Pro vkládání jednoduchých konstrukcí, jako jsou úseky přímé koleje a směrové oblouky, slouží volba postupně vkládat navazující instance HorizontalCurve. Pro vkládání výhybkových konstrukcí slouží volba vložit instance HorizontalCurve pro lokalizaci výhybek a výhybkových konstrukcí DSFD. V průběhu vkládání je uživatel veden volbami v rámci nástroje. Při ručním navazování jednotlivých křivek je nezbytně nutné, aby zvolený počáteční bod byl totožný s koncovým bodem již existujícího objektu geometrie. Při grafickém výběru je vhodné využít funkce uchopení bodů v SW Rhinoceros. Správně uchopená reprezentace instance třídy **GeoPoint** se pozná podle toho, že při uchopení daného bodu není zvýrazněna žádná křivka do něj vedoucí a při uzavírání instance třídy **LinearElement** nad danou geometrií je uživatel informován o nevytvoření nové instance dané třídy, ale o ztotožnění počátečního bodu s již existující instancí. Pokud se vytvářená instance napojuje na více již existujících prvků geometrie, musí dojít ke ztotožnění instancí třídy **GeoPoint** ve všech bodech navázání.

Dalším dílčím krokem je vkládání objektů infrastruktury prostřednictvím volby vkládání a přidružené lokalizace instancí `ModularEntity` v sekci `micro 2D`. Při vkládání je vhodné postupovat po jednotlivých dopravních a traťových kolejích. V rámci každé dopravní je doporučeno vždy nejprve vložit instanci třídy `ControlArea` reprezentující dopravnu a (existuje-li, popř. má-li existovat) instance třídy `SignalBox` reprezentující místa z nichž je doprava (simulovaně) ovládána a ty spolu propojit pomocí volby `ucelená tvorba sítě` → `vzájemné propojování síťových entit`. Následně je doporučeno do modelu vložit vždy všechny instance dané třídy, ty (dává-li to z hlediska popisu smysl) přiřadit k dané dopravně a popř. k již existujícím instancím tříd, a až poté pokračovat vkládáním instancí další třídy.

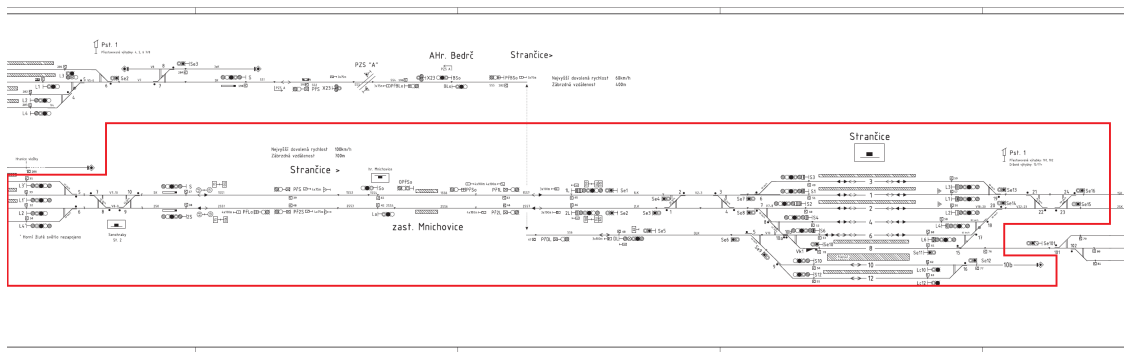
Předposledním navrženým krokem je vkládání instancí třídy `TracksideSignal` s funkcí rychlostníků a instance třídy `SpeedSection` pro každou instanci třídy `AssociatedSection` (stejným způsobem jako vkládání ostatních tříd tematického bloku `ModularEntity`) jejich vzájemné propojení (postup viz výše). Dále je v tomto dílčím kroku navrženo vkládání instancí třídy `VerticalCurve` volbou `ucelená tvorba sítě` → `micro 2D` → `vkládání a lokalizace instancí VerticalCurve`. Po dokončení všech předchozích akcí před dokončením projektu jsou pak na základě instancí této třídy tvořeny instance třídy odvozené od třídy `VerticalCurve` tvořící trojrozměrnou reprezentaci geometrie modelu.

Posledním dílčím krokem je založení databáze se strukturou odpovídající jádru i navrhovanému SR a naplnění této databáze daty exportovanými z nástroje pro tvorbu VMŽI. Dokončením tohoto exportu je tvorba VMŽI ve specifickém rozšíření ukončena.

6 Ověření funkčnosti tvorby modelu

Ověření funkcionality nástroje a metody tvorby modelu bylo provedeno vytvořením části modelového kolejiště DSFD podle podkladů vytvořených podle navrženého postupu v kap.: 5.1 s pomocí nástroje pro tvorbu doplněného o funkce a další elementy vytvořené pro potřebu obsluhy SR. Tato tvorba byla provedena metodou popsanou v kap.: 5.2. Jako ověřovací úsek byl zvolen úsek Strančice (včetně) - Senohraby (mimo). Tento úsek byl vybrán z důvodu vyšší než obvyklé složitosti kolejiště a výskytu většiny kolejových konstrukcí a prvků infrastruktury implementovaných ve SR. Funkčnost implementovaných prvků, které se v daném úseku nevyskytují, bylo provedeno samostatně.

Popisovaný úsek začíná na říčanském zhlaví stanice Strančice, která je největší standardní stanicí na kolejišti DSFD. V obvodu stanice Strančice se nachází sedm dopravních a dvě manipulační koleje (v rámci ověření nebyl popsán obvod depa). Dále byl popsán navazující úsek dvoukolejně trati do stanice Senohraby se zastávkou a hradlem Mnichovice. Konec popisovaného úseku byl z důvodu přítomnosti obloukových výhybek (metoda jejichž tvorby byla implementována v rámci návrhu nástroje pro tvorbu SR v rámci nástroje pro tvorbu VMŽI) posunut až do úrovně odjezdových návěstidel stanice Senohraby ve směru do stanice Strančice. Schéma popisovaného úseku lze vidět v Obr.: 9.



Obrázek 9: Schéma části kolejiště DSFD s vyznačeným úsekem na němž proběhlo ověření metody tvorby VMŽI. Převzato ze [20] a upraveno.

V průběhu ověřování byly zjištěny 2 nedostatky nástroje pro tvorbu a navržené metody. Prvním nedostatkem je využití v základu implementovaných bloků graficky reprezentující jednotlivé instance tříd vložených do modelu v nástroji pro jeho tvorbu a neprovedení změny měřítka. Toto způsobuje, že vzhledem k použitému měřítku při tvorbě bloku je velikost těchto bloků v porovnání s velikostí modelu malá, což znesnadňuje jejich výběr. Využití základních grafických bloků dále snižuje přehlednost modelu, protože dochází k využití grafického bloku reprezentujícího návěstidlo pro více typů návěstidel, než je u dalších SR obvyklé. Vhodnějším řešením by bylo definovat pro každý typ návěstidla vlastní blok v požadované velikosti. Tento problém však nebrání užití nástroje ani nenižuje využitelnost navržené metody.

Druhým nedostatkem byla vyšší než očekávaná časová náročnost sběru dat o prvcích infrastruktury. V rámci ověření bylo zahájeno zpracování daného podkladu v rozsahu potřebném pro tvorbu modelu na daném úseku, avšak vzhledem k vysoké časové náročnosti, přítomnosti již existujících obdobných podkladů a možnosti trvalého přístupu ke kolejišti DSFD v průběhu ověření, bylo od tohoto v průběhu tvorby podkladu pro stanici Strančice upuštěno. Vzhledem k této časové náročnosti lze tento úkon doporučit pouze při absenci spolehlivých zdrojů informací a omezenému přístupu k popisované infrastruktuře. V průběhu návrhu metody bylo zvažována možnost ukládat informace o infrastruktuře přímo do nelokalizovaných instancí reprezentujících jednotlivé prvky infrastruktury, do nichž by byly tyto informace ukládány, a které by v rámci tvorby byly pouze lokalizovány, čímž by odpadl tento krok. V době návrhu byl tento způsob evidence zamítnut z důvodu uživatelské nepřívětivosti tohoto způsobu, zaviněné složitějším způsobem úpravy chybně zadaných dat a nutností hledat danou instanci v seznamu, což vzhledem k absenci možnosti třídění seznamu či omezení položek podle příslušnosti ke třídě, přestává být praktické již při nižších desítkách vložených instancí. Tento počet přibližně odpovídá popisu jedné z menších stanic na kolejišti DSFD. Z dlouhodobého hlediska by bylo vhodné zvážit (ve spolupráci s autorem VMŽI a jádra nástroje pro jeho tvorbu) implementaci daných funkcí (hlavně funkce třídění a výběru zobrazovaných tříd v seznamu vložených instancí tříd odvozených od třídy **NetEntity**), čímž by bylo možné aplikovat tento postup a umožnit tento krok zcela přeskočit. Tento nedostatek však, vzhledem k jeho charakteru, nebrání využitelnosti dané metody ani navrženého nástroje.

Ověření potvrdilo funkčnost navrhované metody i navržených doplnění nástroje pro tvorbu VMŽI obsluhujících navržené SR. Odhalené nedostatky nebyly charakteru znemožňující využití metody ani nástroje, avšak z dlouhodobého hlediska by bylo vhodné implementovat navrhované změny odstraňující tyto nedostatky.

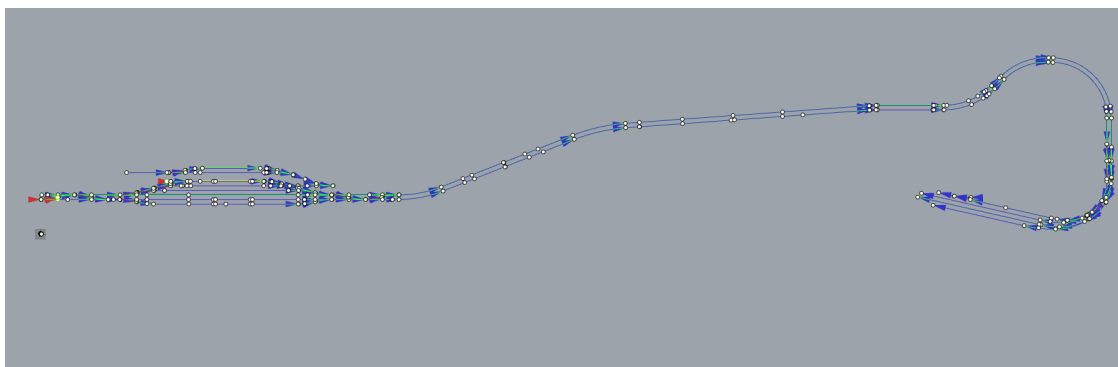
6.1 Výstupy ověření

V rámci ověření byly vytvořeny následující výstupy:

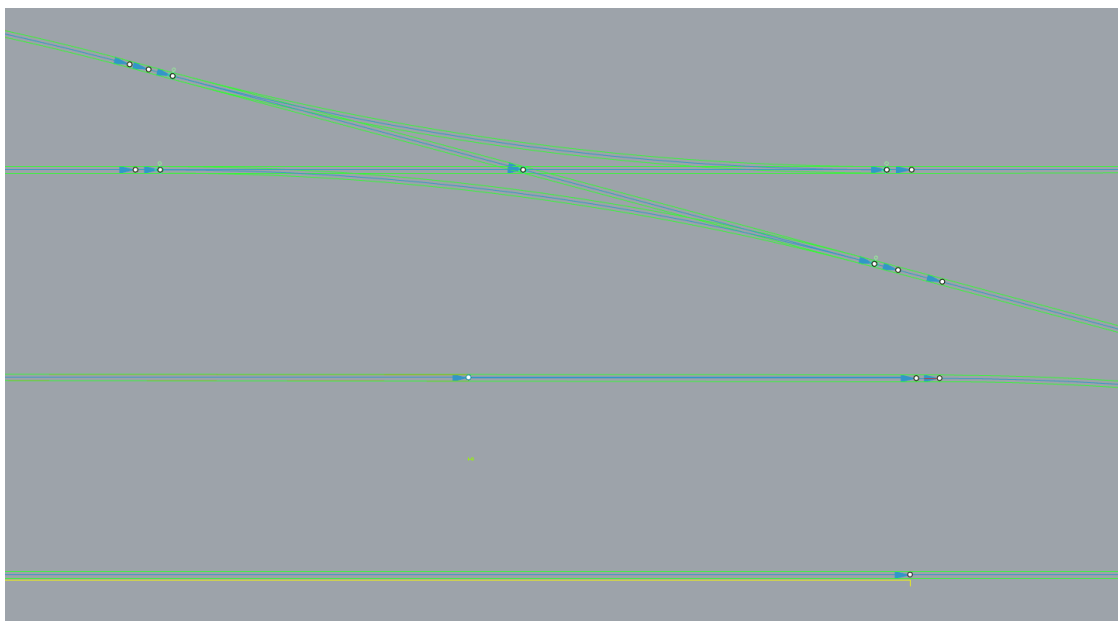
1. Výkres úseku kolejiště v SW AutoCad
2. VMŽI v prostředí nástroje pro jeho tvorbu

První z výstupů reprezentuje navrhovaný výsledný stav podkladů pro tvorbu VMŽI před zahájením tvorby výsledného modelu v prostředí nástroje pro tvorbu VMŽI. Vzhledem k časové náročnosti přípravných kroků bylo využito již existujících podkladů vytvořených v letech 2016 a 2019, které byly v rámci ověřování metody pouze dále rozšiřovány, přičemž však bylo dbáno, aby byly ověřeny všechny aspekty navrhovaného postupu. Vzhledem k mnohem vyšší, než očekávané časové náročnosti nebyl dokončen přípravný krok pro tvorbu VMŽI v podobě vytvoření soupisu informací o prvcích infrastruktury a informace o jednotlivých prvcích byly přímo vkládány do VMŽI při jeho tvorbě (viz předchozí kapitole).

Druhý z výstupů reprezentuje konec fáze tvorby VMŽI v navrženém rozšíření nástroje pro tvorbu VMŽI. Jedná se o model kolejiště ve VMŽI s užitím navrženého specifického rozšíření a postupu vkládání dat v rámci této práce. Na rozdíl od předchozího výstupu byl tento výstup zcela vytvořen v průběhu ověření. Ukázka z výsledku tvorby je uvedena v Obr.: 10, v Obr.: 11 je pak uveden detail grafické reprezentace vytvořeného modelu.



Obrázek 10: Grafická reprezentace modelu uvedeného úseku v prostředí nástroje pro tvorbu VMŽI běžícím v SW Rhinoceros. Zdroj Autor.



Obrázek 11: Detail modelu části stanice Strančice. Je možné si povšimnout nedostatečnou velikost bloků pro vizualizaci vybraných specializací třídy **NetEntity**. Modré šipky představují instance třídy **HorizontalCurve**, zelené orámování okolo nich reprezentuje instance třídy **SpeedSection**, pod nimi jsou skryta orámování představující instance třídy **TrackCircuit**. Žlutou čarou jsou reprezentovány instance třídy **PlatformEdge**, Dále jsou k vidění instance tříd **PositionedRelation**, **SwitchMachine** a **TracksideSignal**. Zdroj Autor.

7 Závěr

V rámci první části realizace této práce proběhlo seznámení se s Víceúčelovým modelem železniční infrastruktury a existujícími specifickými rozšířeními v rozsahu nejméně v úrovni znalostí požadovaných pro uživatele a tvůrce vlastního specifického rozšíření. Vzhledem ke spíše praktickému charakteru této práce byl při seznámení kladen zvláštní důraz na strukturu VMŽI a již existujících specifických rozšíření, neboť znalost této struktury, implementaci tříd a jejich provázání a návaznost této struktury na existující nástroje, je nezbytně nutná pro návrh specifického rozšíření. Popis základních vlastností VMŽI a jeho struktury (vč. struktury dostupných specifických rozšíření) je uveden v kap.: 3, 3.1 a 3.3.

Druhou částí práce bylo ověření využitelnosti VMŽI na konkrétní projekty plánované k realizaci na DSFD. Byly ověřeny čtyři konkrétní projekty, jmenovitě: **Nový software řízení modelových souprav**, **Integrální zabezpečovací zařízení pro DSFD**, **Tvorba 3D prostředí pro potřeby simulátorů na DSFD** a **Pasportizace kolejí DSFD**. V rámci každého projektu byly zjištěny požadavky na popis z pohledu VMŽI a byla prověřována splnitelnost požadavků na popis a vzájemná kompatibilita případných popisů mezi sebou. Bylo zjištěno, že projekty **Nový software řízení modelových souprav**, **Integrální zabezpečovací zařízení pro DSFD**, **Tvorba 3D prostředí pro potřeby simulátorů na DSFD** sdílí značnou část požadavků na popis a případné rozdílné požadavky na popis vzájemně významněji nekolidují a tím nesnižují využitelnost modelu pro jiné využití. Naopak bylo zjištěno, že požadavky projektu **Pasportizace kolejí DSFD** se zásadně odlišují z hlediska požadavků i charakteru popisu a tyto odlišnosti by vedly k výraznému zvýšení komplexnosti navrhovaného specifického rozšíření i nižší využitelnosti popisu zhotoveného podle požadavků tohoto projektu. Přestože bylo shledáno, že VMŽI by bylo možné přizpůsobit i tomuto využití, bylo rozhodnuto o nezahrnutí požadavků tohoto projektu do specifikace navrhovaného specifického rozšíření. Ověření, vč. zjištěných požadavků jednotlivých prověřovaných možností užití je uvedeno v kap.: 3.4. Dále je na počátku kap.: 4 uveden seznam terminálních tříd navržených k uspokojení požadavků popisu vybrané trojice způsobů užití.

Třetí část práce se věnuje samotnému návrhu specifického rozšíření a jeho implementaci. Zvláštní důraz je kladen na oblasti, které se charakterem implementace liší oproti již existujícím řešením ve VMŽI a jeho SR. V rámci této části je též uveden seznam změn jádra nástroje pro tvorbu VMŽI provedených se souhlasem autora VMŽI umožňující implementování některých požadovaných funkcí, popř. vylepšující funkci nástroje pro obsluhu navrhovaného SR, a dále byl vytvořen seznam a popis funkcí implementovaných v rámci tohoto navrhovaného nástroje (doplňovaného do nástroje pro tvorbu VMŽI), který je zde uveden pro potřeby případné údržby tohoto nástroje v případě změn jádra VMŽI nebo nástroje pro jeho tvorbu. Výše popsaná část práce je uvedena v kap.: 4 a příloze 2.

Poslední částí práce byl návrh metody plnění modelu daty vč. návrhu způsobu tvorby potřebných podkladů pro tuto tvorbu a ověření funkčnosti navrženého SR, vytvořeného nástroje pro jeho tvorbu i metody jeho tvorby navrhované v této práci. Při návrhu metody byly využity metody a podklady vytvořené v dřívější době. Následně bylo provedeno ověření. Z časových důvodů proběhlo ověření metody tvorby podkladů (prvního kroku tvorby VMŽI) současně s implementací navrhovaného specifického rozšíření, což však vzhledem ke zcela odlišným nástrojům a charakteru práce neovlivňuje výsledky tohoto ověření. V rámci ověření byly odhaleny dva nedostatky v navrhované metodě a nástroji pro tvorbu v daném specifickém rozšíření. Tyto nedostatky byly charakteru, který ne-

znemožňoval využití navrhovaných postupů, měl však vliv na časové zatížení a komfort uživatele. V rámci této části byla též navržena možná řešení těchto problémů. Tato část práce je popsána v kap.: 5 a 6.

Autor v rámci této práce získal znalosti v oblasti popisu infrastruktury principy definovanými v **RailTopoModel**. Autor dále považuje za důležité získání zkušeností s tvorbou SW v rámci rozsáhlejších projektů, vč. spolupráce s dalšími programátory, dále zkušenosti s odhalováním, identifikací a detailním popisem chyb v kódu jiného programátora, které autor této práce doposud neměl a v průběhu práce je získal.

Reference

- [1] INTERNATILONAL UNION OF RAILWAYS (UIC). IRS 30100: RailTopoModel Railway Infrastructure Topological Model (v 1.1). Paris: International Union of Railways (UIC), 2017.
- [2] HLUBUČEK, Adam. Víceúčelový model železniční infrastruktury (v. 12.2). Praha, 2022.
- [3] Koncept železnice 4.0: Koncept digitální železnice v ČR. Železnice 4.0 [online]. c2023 [cit. 2023-07-23]. Dostupné z: <https://zeleznice40.cz/>
- [4] SUBSET-026-2. ERTMS/ETCS System Requirements Specifications: Chapter 2 - Basic System Description. 3.6.0. Brussels: UNISIG, 2016.
- [5] SUBSET-026-7. ERTMS/ETCS System Requirements Specifications: Chapter 7 - ERTMS/ETCS language. 3.6.0. Brussels: UNISIG, 2016.
- [6] HLUBUČEK, Adam. Jádru Víceúčelového modelu železniční infrastruktury. In: *Team Víceúčelového modelu železniční infrastruktury* [online]. Praha, [2022] [cit. 2023-07-19]. Dostupné z: https://teams.microsoft.com/_#/school/FileBrowserTabApp/Dokumentace?threadId=19:15a7b2de84514db0a38fdb55a89c368@thread.tacv2&ctx=channel
- [7] Levels of detail. *RailTopoModel* [online]. Paris: ERIM group (UIC), 2017, 13. 3. 2017 [cit. 2023-07-25]. Dostupné z: https://wiki.railtopomodel.org/index.php?title=Levels_of_detail
- [8] IS:netRelation. railML Wiki [online]. 2019, 19. 12. 2019 [cit. 2023-07-12]. Dostupné z: <https://wiki3.railml.org/wiki/IS:netRelation>
- [9] Topological structure (network). *RailTopoModel* [online]. Paris: ERIM group (UIC), 2017, 13. 3. 2017 [cit. 2023-07-29]. Dostupné z: [https://wiki.railtopomodel.org/wiki/Topological_structure_\(network\)](https://wiki.railtopomodel.org/wiki/Topological_structure_(network))
- [10] UC:Use cases. railML Wiki [online]. 2023, 16. 3. 2023 [cit. 2023-07-29]. Dostupné z: https://wiki3.railml.org/wiki/UC:Use_cases
- [11] HLUBUČEK, Adam. Geometrie - Víceúčelový model železniční infrastruktury. In: *Team Víceúčelového modelu železniční infrastruktury* [online]. Praha, 2022 [cit. 2023-07-17]. Dostupné z: https://teams.microsoft.com/_#/school/FileBrowserTabApp/Dokumentace?threadId=19:15a7b2de84514db0a38fdb55a89c368@thread.tacv2&ctx=channel
- [12] Intrinsic positioning / referencing. *RailTopoModel* [online]. Paris: ERIM group (UIC), 2017, 14. 3. 2017 [cit. 2023-07-22]. Dostupné z: https://wiki.railtopomodel.org/wiki/Intrinsic_positioning_-_referencing
- [13] HLUBUČEK, Adam. ModularEntity v rozšíření ProETCS - Víceúčelový model železniční infrastruktury. In: *Team Víceúčelového modelu železniční infrastruktury* [online]. Praha, 2022 [cit. 2023-07-21]. Dostupné z: https://teams.microsoft.com/_#/school/FileBrowserTabApp/Dokumentace?threadId=19:15a7b2de84514db0a38fdb55a89c368@thread.tacv2&ctx=channel
- [14] HLUBUČEK, Adam. ModularEntity v rozšíření ProMetro - Víceúčelový model železniční infrastruktury. In: *Team Víceúčelového modelu železniční infrastruktury* [online]. Praha, 2022 [cit. 2023-07-21]. Dostupné z: https://teams.microsoft.com/_#/school/FileBrowserTabApp/Dokumentace?threadId=19:15a7b2de84514db0a38fdb55a89c368@thread.tacv2&ctx=channel

- [15] HLUBUČEK, Adam. ModularEntity v rozšíření ProVEXA - Víceúčelový model železniční infrastruktury. In: *Team Víceúčelového modelu železniční infrastruktury* [online]. Praha, 2022 [cit. 2023-07-21]. Dostupné z: https://teams.microsoft.com/_#/school/FileBrowserTabApp/Dokumentace?threadId=19:15a7b2de84514db0a38fdb55a89c368@thread.tacv2&ctx=channel
- [16] VEXA: *FD ČVUT* [online]. Praha: ČVUT v Praze, c2020 [cit. 2023-07-25]. Dostupné z: <https://vexa.fd.cvut.cz/>
- [17] BRANDEJSKÝ, Tomáš, Vít FÁBERA a Martin LESO. Integral railway interlocking systems and its assessment according to european standards. *Acta Polytechnica*. 2023, 1 - 5.
- [18] SŽ D1 ČÁST PRVNÍ: *Dopravní a návěštní předpis pro tratě nevybavené evropským vlakovým zabezpečovačem (ve znění opravy č. 1)*. Praha: Správa železnic, 2022.
- [19] TSignalConstructionType: Infrastructure3. railML Wiki [online]. 2022, 5. 2. 2022 [cit. 2023-07-11]. Dostupné z: <https://wiki3.railml.org/DataModel/3.2/IS/EARoot/EA414.htm>
- [20] KOUTECKÝ, Petr a Dušan KAMENICKÝ. Schéma kolejíště Dopravního sálu Fakulty dopravní ČVUT. Praha, 2017.
- [21] IS:trackGauge. railML Wiki [online]. 2019, 19. 12. 2019 [cit. 2023-07-25]. Dostupné z: <https://wiki3.railml.org/wiki/IS:trackGauge#3.2-0>
- [22] UC:IS:Railway Simulation Laboratory. railML Wiki [online]. 2022, 27. 06. 2022 [cit. 2023-08-01]. Dostupné z: https://wiki3.railml.org/wiki/UC:IS:Railway_Simulation_Laboratory

Seznam obrázků

1	Schéma architektury možné implementace integrálního zabezpečovacího zařízení. Převzato ze [17].	16
2	Architektura virtuálního počítače cloudového řešení integrálního zabezpečovacího zařízení. Převzato ze [17], přeloženo.	17
3	Architektura cloudového řešení integrálního zabezpečovacího zařízení. Převzato ze [17].	18
4	Topologické schéma úplné křižovatkové výhybky. Modrou barvou jsou vykresleny části sdílené jednoduchou výhybkou a úplnou křižovatkovou výhybkou, červeně pak zbývající části úplné křižovatkové výhybky. Označené body jsou z hlediska popisu důležité, protože v nich dochází k napojení jednotlivých částí konstrukce na následující úseky koleje, nebo zde dochází k odbočení nebo křížení jednotlivých částí výhybkové konstrukce. Zdroj autor.	28
5	Výkres geometrického popisu jednoduché výhybky používané na kolejišti DSFD. Zdroj autor.	29
6	Výkres geometrického popisu větší z dvojice prototypů používaných pro popis obloukových výhybek na DSFD. Tm označuje délku malé tečny, Tv označuje délku velké tečny a S označuje délku šikmého posunu (délka přímé koleje v přímé větvi výhybky. Zdroj autor.	30
7	Výkres geometrického popisu křižovatkové výhybky používané na DSFD. K označuje úsek mezi odbočením odbočné větve a bodem křížení (tento úsek zároveň reprezentuje tečnu k oblouku odbočné větve délku a P označuje úsek před jazyky. Zdroj autor.	31
8	Ukázka správného využití lokalizačních prvků bloků navržených pro tvorbu výkresu sloužícího jako podklad pro tvorbu VMŽI. Zdroj autor.	39
9	Schéma části kolejiště DSFD s vyznačeným úsekem na němž proběhlo ověření metody tvorby VMŽI. Převzato ze [20] a upraveno.	42
10	Grafická reprezentace modelu uvedeného úseku v prostředí nástroje pro tvorbu VMŽI běžícím v SW Rhinoceros. Zdroj Autor.	43
11	Detail modelu části stanice Strančice. Je možné si povšimnout nedostatečnou velikost bloků pro vizualizaci vybraných specializací třídy NetEntity . Modré šipky představují instance třídy HorizontalCurve , zelené orámování okolo nich reprezentuje instance třídy SpeedSection , pod nimi jsou skryta orámování představující instance třídy TrackCircuit . Žlutou čárou jsou reprezentovány instance třídy PlatformEdge , Dále jsou k vidění instance tříd PositionedRelation , SwitchMachine a TracksideSignal . Zdroj Autor. . .	44

Seznam příloh

1. Schémata Víceúčelového modelu železniční infrastruktury
2. Popis funkcí vytvořených v rámci usnadnění tvorby modelu v nástroji pro jeho tvorbu v navrženém specifickém rozšíření
3. Model databáze pro SR pro DSFD
4. Výkres s podklady pro tvorbu VMŽI v SW AutoCAD
5. Projekt v SW Rhinoceros s vytvořeným modelem kolejiště DSFD ve VMŽI a navrženém SR.